

Tutorials for Origin 8.5.1

Table of Contents

Table of Contents	iii
1 Data Analysis	1
1.1 Curve Fitting.....	1
1.1.1 Linear and Polynomial Fitting	1
1.1.2 Nonlinear Fitting	5
1.2 Peak Analysis.....	80
1.2.1 Peak Finding.....	80
1.2.2 Peak Integration	83
1.2.3 Peak Fitting	85
1.3 Data Manipulation	96
1.3.1 Setting Column Values.....	96
1.3.2 Worksheet Data Operations.....	109
1.4 Analysis Templates.....	112
1.4.1 Creating and Using Analysis Templates	112
1.4.2 Creating Analysis Templates using Set Column Value	115
1.4.3 Creating a Custom Report Sheet	120
1.5 Analysis Themes	125
1.5.1 Summary.....	125
1.5.2 What you will learn.....	125
1.5.3 Steps.....	125
1.6 Batch Processing	128
1.6.1 Analysis Template and Batch Processing	128
1.6.2 Batch Processing with Summary Report in External Excel File	135
2 Statistics	137
2.1 Descriptive Statistics	137
2.2 ANOVA.....	144
2.2.1 One Way ANOVA.....	144
3 Graphing	151
3.1 Graphing Data From Multiple Sheets.....	151
3.1.1 Summary.....	151
3.1.2 Create a Plot Quickly by Selecting Data.....	151
3.1.3 Add Data to an Existing Graph with Drag and Drop Plotting	152
3.1.4 Create a Plot using Plot Setup.....	156
3.1.5 Customize and Save a Graph Template	159
3.1.6 Plot into a Saved Custom Template	163
3.1.7 Plot Groups	164
3.2 Line Symbol	166
3.2.1 Basic 2D Plotting.....	166
3.2.2 Line Graph with Recession Bars	173
3.2.3 Scatter Plot of Decay and Recovery Curves	177
3.2.4 Micro-Raman spectroscopy of complex nanostructured mineral systems	186
3.2.5 Scatter Central Plot	193
3.2.6 Line Graph with Masked Data	199
3.2.7 Mark out a segment of plot with different plot style	203
3.2.8 Adding a Data Plot to Existing Graph	207
3.3 Column Bar Pie	209
3.3.1 2D Pie Chart of a Population Study	209

3.3.2	Stack Column and Bar Charts of Rechargeable Battery Statistics	214
3.3.3	Bar Graph of African Population	218
3.3.4	Bar and Scatter Plot with Error Bar	222
3.3.5	Column Graph with Error Bars	227
3.4	Multi-Axis Multi-Panel	229
3.4.1	Double Y	229
3.4.2	3Ys Y-YY	233
3.4.3	Multiple Layers with Linked Axis	240
3.4.4	Stack Lines by Y Offsets	250
3.4.5	Vertical 2 Panel Line	255
3.4.6	Multiple Axis Plot	262
3.4.7	Energy-Level Structure of the Er ³⁺ Ion in A YAG Crystal	270
3.4.8	Create an 8 layer multi-panel plot template	277
3.5	Contour	280
3.5.1	XYZ Contour	280
3.5.2	Contour Plot with Major and Minor Levels Filled by Using Color Palette	285
3.5.3	Contour Plots and Color Mapping	292
3.5.4	Polar Contour	306
3.5.5	Ternary Contour	311
3.5.6	Combining Line and Contour Plots	316
3.6	Statistical	322
3.6.1	Box Plot	322
3.6.2	Simple Dot Chart	325
3.6.3	Multi-Data Dot Chart	331
3.6.4	Add multiple fitted curves in a Histogram	346
3.6.5	Weibull Probability Plot	351
3.6.6	Q-Q Plot	354
3.7	Polar	357
3.7.1	Polar Contour	357
3.8	3D	362
3.8.1	Basic 3D Plotting	362
3.8.2	3D Pie Chart	369
3.8.3	3D Scatter Plot with Line Projections of Core Drilling Locations	371
3.8.4	Emission Intensity of Single-walled Nanotubes Plotted as a surface Graph	380
3.8.5	Surface with Missing Values	386
3.8.6	Creating Intersecting Surface Plots from Worksheets	390
3.8.7	Surface with Contour Projection	401
3.8.8	Surface with Symbols and Droplines	410
3.8.9	Intersecting Color Surfaces	420
3.8.10	Colormap from Second Matrix	426
3.8.11	Color Map Surface Graph	433
3.9	Vector	438
3.9.1	Vector Graph	438
3.9.2	3D Vector	440
3.10	Ternary	443
3.10.1	Ternary Contour	443
3.11	Waterfall	448
3.11.1	Waterfall with Y and Z Color Mapping	448
3.11.2	3D Waterfall Graph	455
3.12	Specialized	459
3.12.1	Windrose Graph	459

3.12.2	Plot Functions with Parameters Defined in a Worksheet.....	466
3.12.3	Open-High-Low-Close-Volume Stock Chart.....	472
4	Customizing Graphs.....	475
4.1	Customizing a Graph.....	475
4.1.1	Summary.....	475
4.1.2	Resize Layer.....	475
4.1.3	Customize Data Plot and Axes.....	477
4.1.4	Save and Reuse a Template.....	478
4.1.5	Changing graph using theme.....	479
4.1.6	Plotting Order.....	481
4.1.7	Customize points.....	483
4.1.8	Group plots.....	485
4.1.9	Plot a Color-mapped Waterfall.....	492
4.2	Themes.....	498
4.2.1	Copy and Apply a Plot Format to Another Plot.....	498
4.3	Color.....	499
4.3.1	Plotting Overlapping Data and Setting Transparency.....	499
4.4	Layers.....	504
4.4.1	Adding and Arranging Layers.....	504
4.4.2	Merging and Arranging Graphs.....	506
5	Programming.....	519
5.1	LabTalk.....	519
5.1.1	Advanced Scripting using LabTalk.....	519
5.2	Origin C.....	533
5.2.1	Introduction to Origin C and Code Builder.....	533
5.2.2	The Code Builder Workspace.....	536
5.2.3	Adding New Origin C Functions to Origin.....	539
5.2.4	Organizing and Accessing Origin C Functions.....	541
5.2.5	Accessing Internal Origin Objects by Origin C.....	545
5.3	X-Functions.....	549
5.3.1	Command Window and X-Functions.....	549
5.3.2	Introduction to X-Functions.....	557
5.3.3	How to Create a Wizard.....	561
6	Importing Data.....	585
6.1	ASCII.....	585
6.1.1	Importing Data.....	585
6.1.2	Single ASCII.....	595
6.1.3	Import Wizard.....	598
6.1.4	Import Time Data.....	602
6.1.5	Post Processing with Import Filter.....	607
7	Collaboration and Connectivity.....	609
7.1	Connectivity.....	609
7.1.1	Excel.....	609

1 Data Analysis

1.1 Curve Fitting

1.1.1 Linear and Polynomial Fitting

Linear Fitting and Outlier Removal

Summary

An outlier is typically described as a data point or observation in a collection of data points that is "very distant" from the other points and thus could be due to, for example, some fault in the measurement procedure. Identification and removal of outliers is often controversial, and is typically "more acceptable" in situations where the model used to describe the data is well known and well accepted.

Minimum Origin Version Required: Origin 8.1 SR2

What you will learn

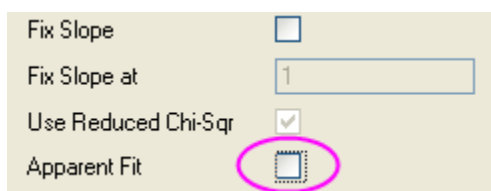
This tutorial will show you how to:

- Perform linear regression on a set of data points
- Examine the Residuals Table in the output and "identify" outliers
- Use the Masking Tool to remove the outlier points
- Use the Recalculation mechanism to automatically update the result after outlier removal

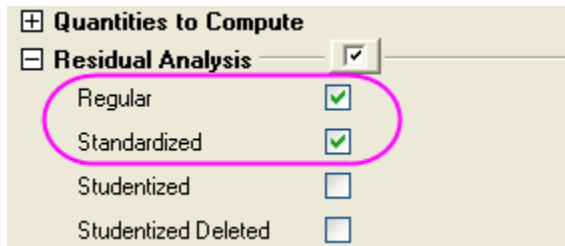
The procedure described in this tutorial is also applicable to other fitting tools such as Polynomial and Nonlinear Fitting

Steps

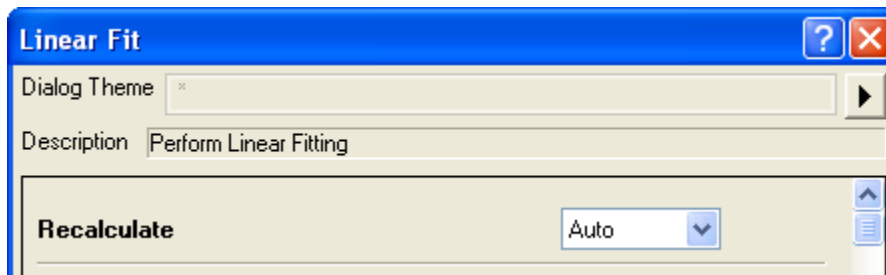
1. Start with a new workbook and import the file `\Samples\Curve Fitting\Outlier.dat`.
2. Click and select the second column and use the menu item **Plot: Symbol: Scatter** to create a scatter plot.
3. With the graph active, use the menu item **Analysis: Fitting: Fit Linear** to bring up the Linear Fit dialog. Note that if you have used the Linear Fit dialog before, there will be a fly-out menu and you need to select the **Open Dialog...** sub menu.
4. Under the **Fit Options** branch, clear the **Apparent Fit** check box.



5. Expand the **Residual Analysis** tree node in the dialog, and check the **Standardized** check box.



6. Change the **Recalculate** drop-down at the top of the dialog to **Auto** and press the OK button at the bottom of the dialog. The dialog will close and linear regression will be performed on the data.

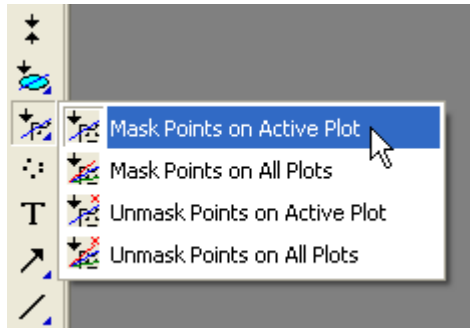


7. Select the **FitLinearCurve1** result sheet in the data workbook and scroll to the right side to view the **Standardized Residual** column. You will note that the value in row 6 in this column is -2.54889:

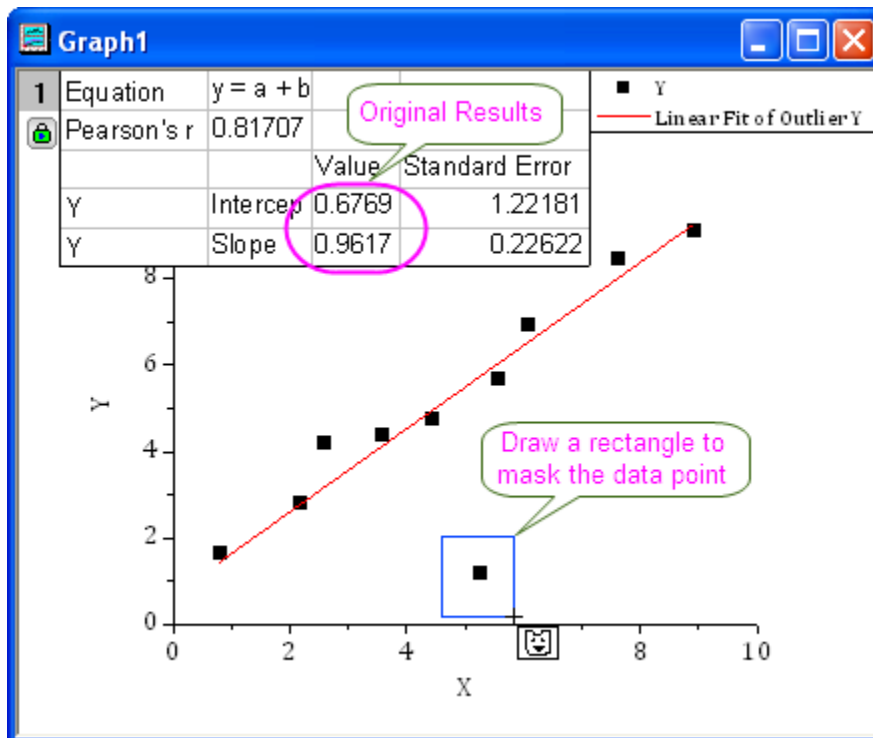
The image shows a data workbook window titled "Outlier - Outlier.dat". The "FitLinearCurve1" sheet is selected. The table below shows the data for the "Standardized Residual" column, with row 6 highlighted in pink.

	A(X1)	B(Y1)	C1(X2)	C2(Y2)	C3(Y2)
Long Name	Independ	Linear Fit of	Independ	Regular R	Standardize
Parameters	Fitted Curves Plot				
1	0.79	1.43673	0.79	0.23327	0.13281
2	0.87202	1.51561	2.16	0.08563	0.04875
3	0.95404	1.5945	2.56	1.08092	0.61539
4	1.03606	1.67339	3.57	0.28951	0.16483
5	1.11808	1.75227	4.43	-0.14762	-0.08404
6	1.2001	1.83116	5.23	-4.47705	-2.54889
7	1.28212	1.91004	5.55	-0.31482	-0.17923
8	1.36414	1.98893	6.06	0.44467	0.25316
9	1.44616	2.06781	6.67	2.41798	1.37662
10	1.52818	2.1467	7.61	0.48391	0.2755
11	1.6102	2.22558	8.91	-0.09641	-0.05489

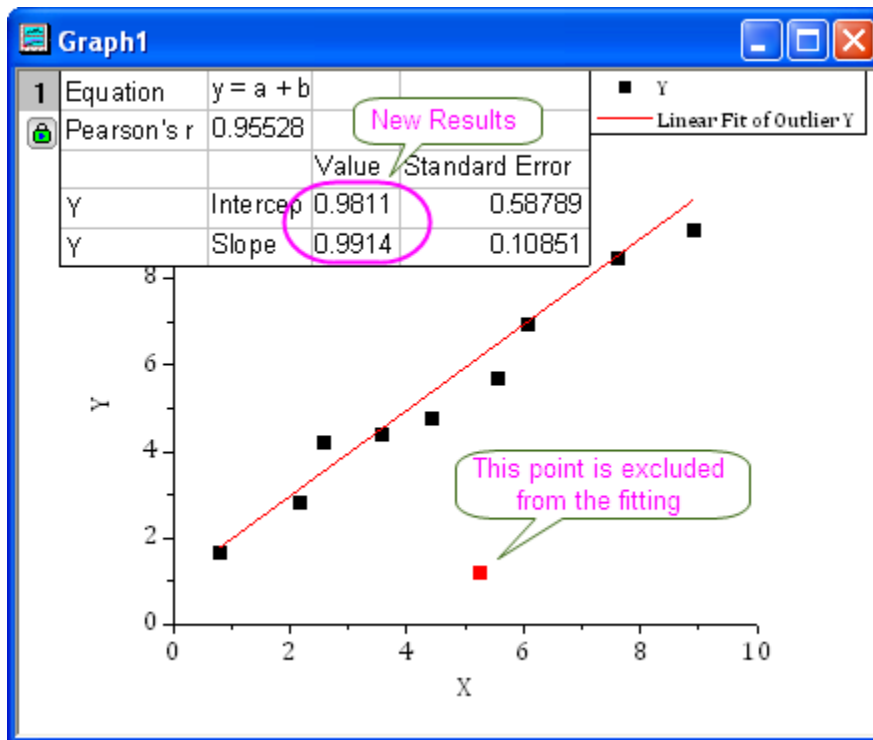
8. Make the graph active and then click and hold down the mouse left button on the "Regional Mask Tool" button in the Tools toolbar. Select the "Add Masked Points on Active Plot" submenu which will be the first item in the fly-out menu:



9. With the above submenu selected, go to the graph and click on the 6th data point to mask the point.



This changes the input data to the linear fit operation and the auto update mechanism will trigger. The linear fit will be repeated with this particular masked point left out. The fit curve in the graph and the pasted parameters will automatically update. Your result graph should then look like below:



1.1.2 Nonlinear Fitting

Nonlinear Curve Fitting

Summary

The Nonlinear Curve Fitter in Origin provides close to 200 built-in fitting functions, plus the ability to fit with user-defined functions. This tutorial will demonstrate how to use the fitter for a basic fitting task, including flexible selection of data.

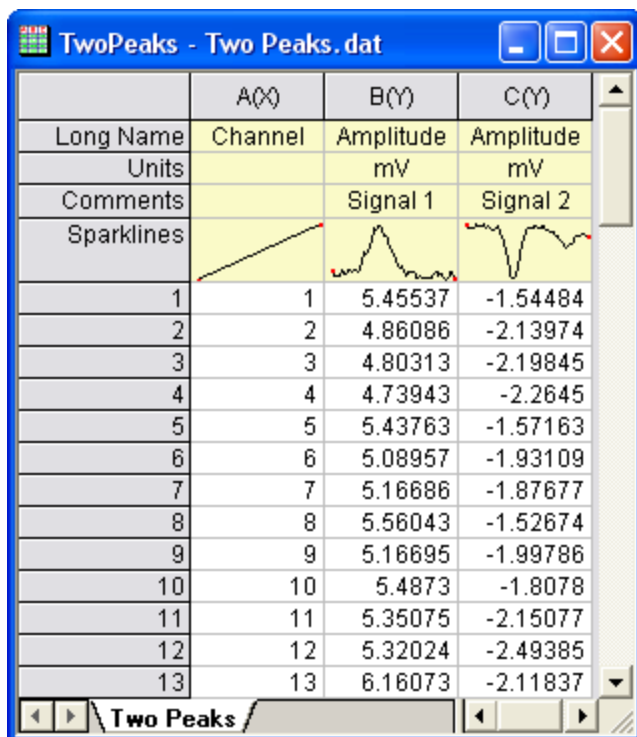
This tutorial will show you how to:


- Use the Regional Data Selector tool to select a specific range of a data plot to fit
- Use the Nonlinear Curve Fitter (NLFit) dialog to perform curve fitting
- Use Change Parameters to modify the data range to be fit


Graphically selecting the input data range to be fitted

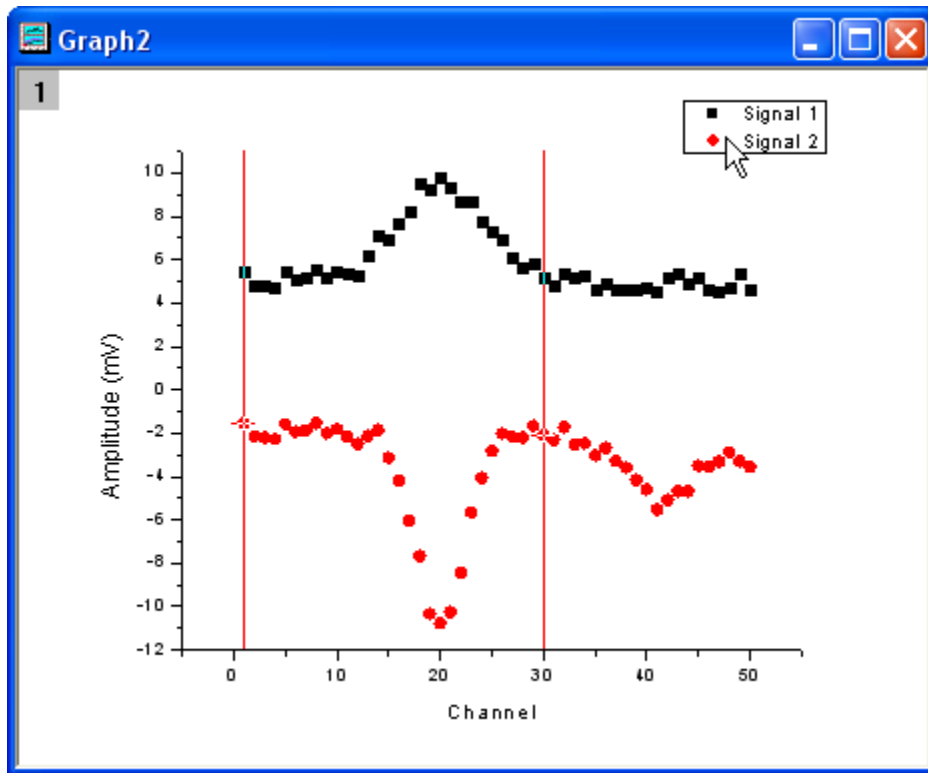
In this section, we will show you how to select a specific range of a dataset and perform curve fitting on it.

1. Start with a new project or create a new workbook and import the data file `\samples\curve fitting\Two Peaks.dat`, using **File: Import: Single ASCII**. Your workbook will look like this:

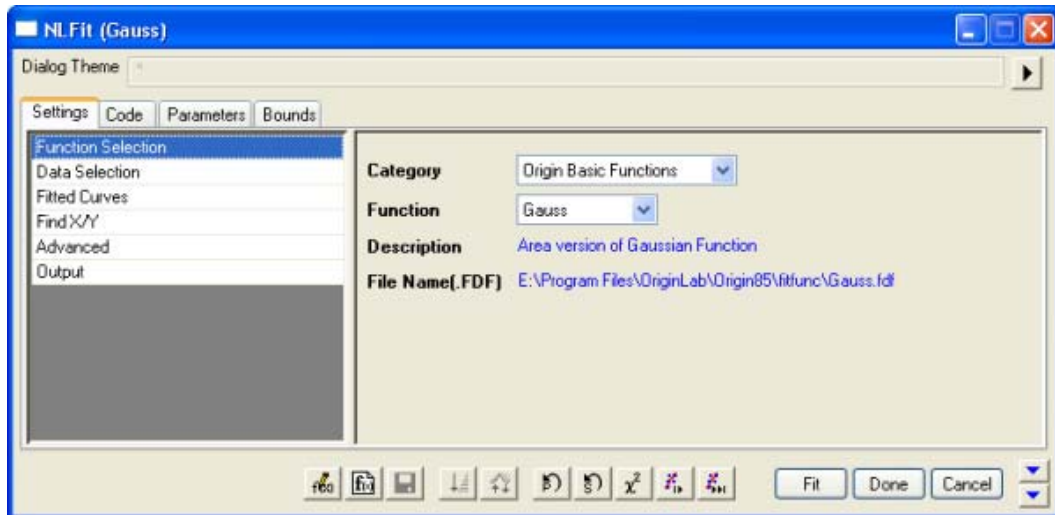


2. Highlight columns **B** and **C** and click the **Scatter** button, , from the **2D Graphs** toolbar to create a scatter graph. Note: Origin will automatically plot the two Y columns against the associated X column, column A, so there is no need to highlight column A.

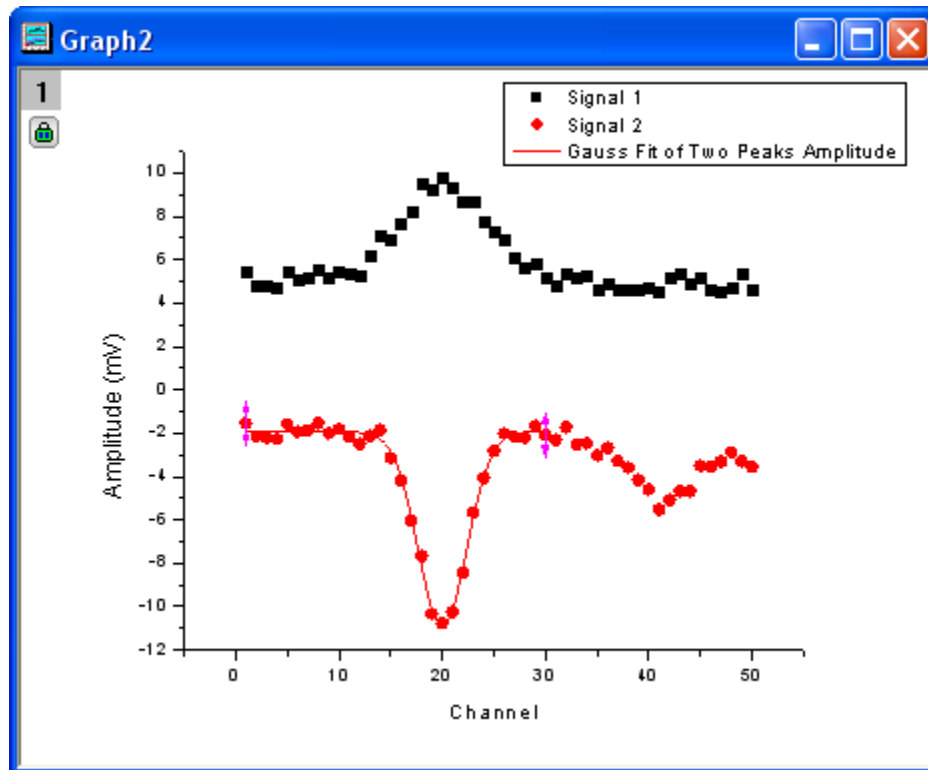
- The graph you just created includes multiple data plots, but only the first data plot, **Signal 1**, is active, meaning automatically picked up as input when an analysis dialog such as the Nonlinear Curve Fitting tool is opened. To fit the large peak on the left side of the second data plot, **Signal 2**, you can graphically select this range as input using the **Regional Data Selector** tool, , from the **Tools** toolbar to draw a rectangle around the big peak in the lower plot on the graph (you can move the two vertical lines to appropriate positions by dragging). The graph should look like this:




- Select **Analysis: Fitting: Nonlinear Curve Fit** to open the **Nonlinear Curve Fitting** dialog, and select **Gauss** from the **Function** drop-down list:

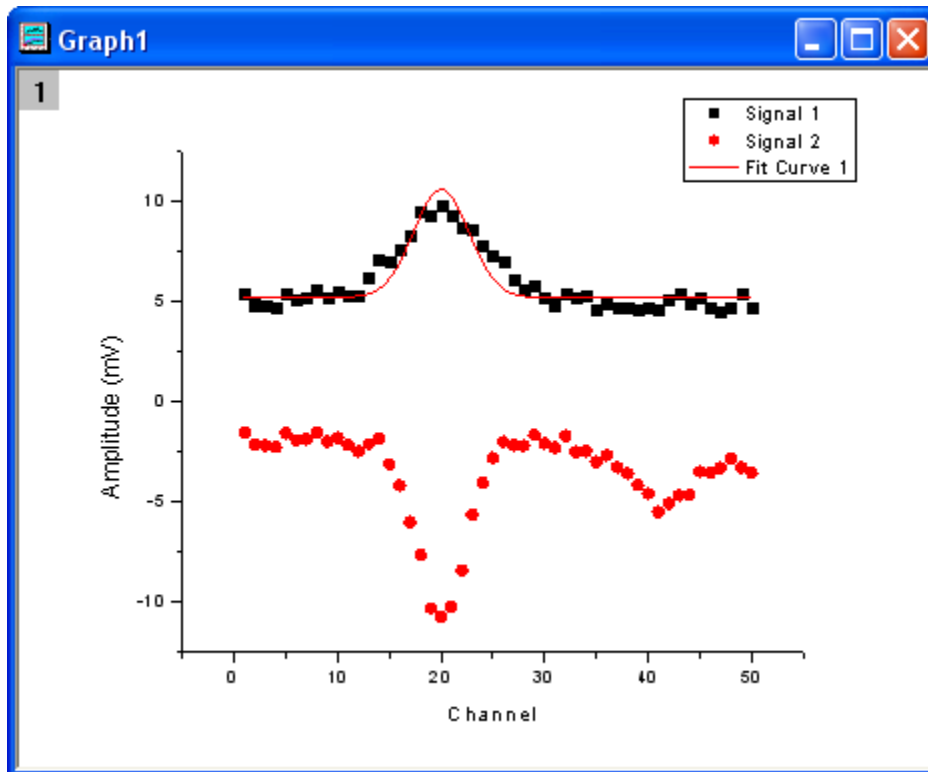


- The fit line is drawn *only* for the range selected. Note that if you select **Data Selection** on the **Settings** tab, you can view the input data range that the dialog automatically picked up from the graph window.
- Click **Fit** to finish the fitting. Note the analysis markers on the graph, marking the range that the analysis was performed on. These markers, as well as the green lock that appears, can be used to *Change Parameters* of the fitting. Delete the table on the graph. The graph will look like this:

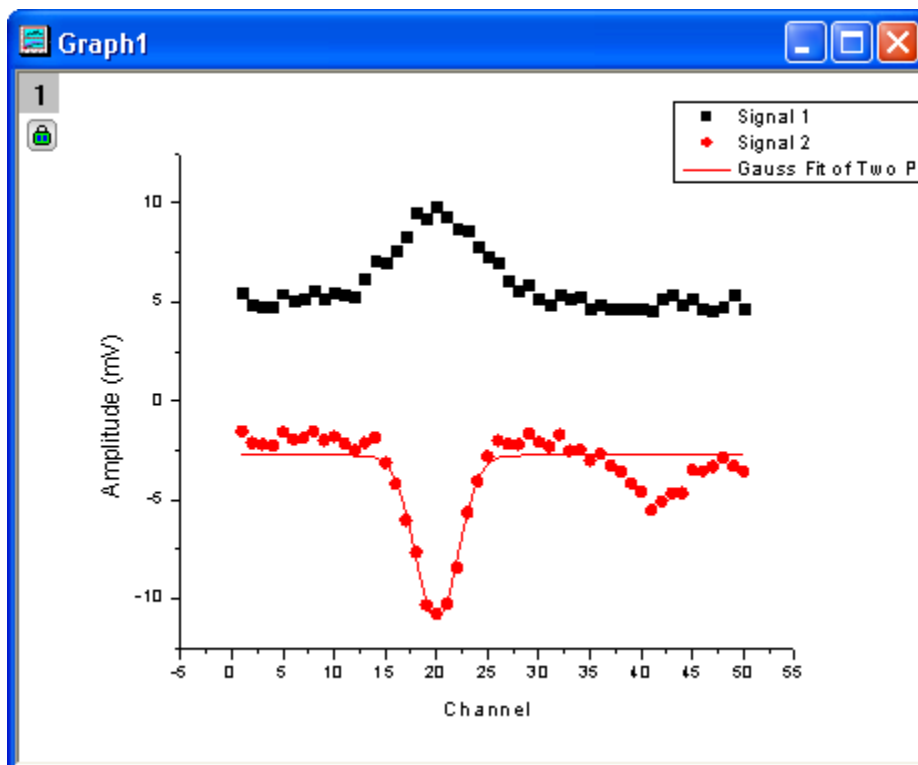


Assigning an Input Data Range inside the NLFit dialog

- Begin by creating a new graph as follows: In the TwoPeaks worksheet, highlight columns **B** and **C** and click the **Scatter** button, , from the **2D Graphs** toolbar to create a scatter graph.
- Select **Analysis: Fitting: Nonlinear Curve Fit** to open the **Nonlinear Curve Fitting** dialog, and select **Gauss** from the **Function** drop-down list. Please note that a graph in Origin can include multiple data plots, but only one of them will be active. In the current graph, **Signal 1**, the first data plot, is the active data plot. If you select **Data Selection** on the **Settings** tab, you can see that it has been automatically assigned as the **Input Data** range.



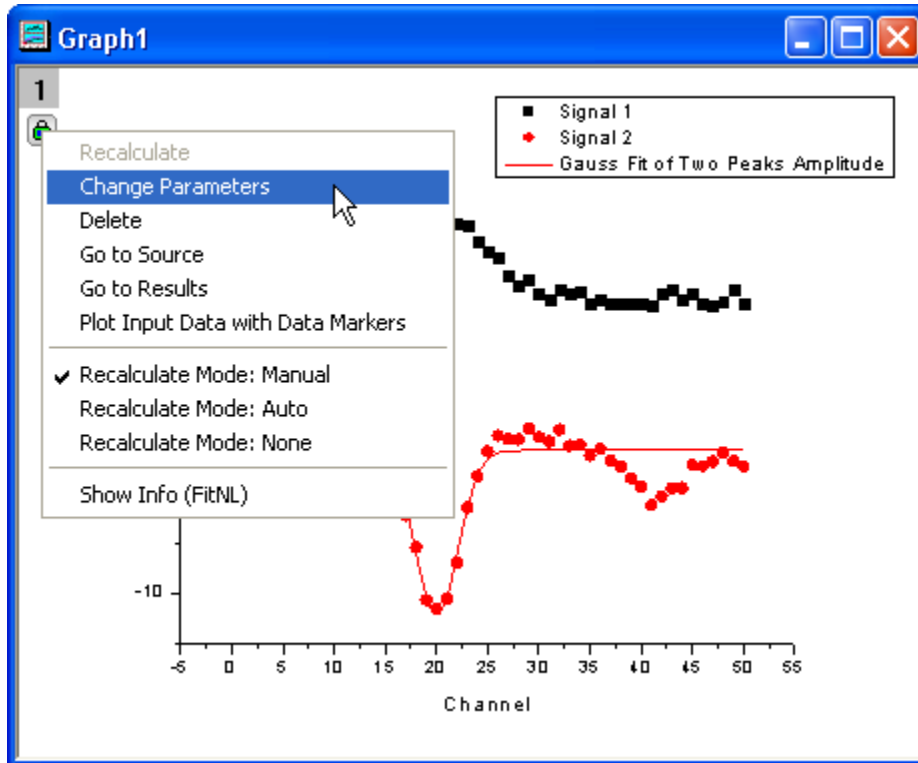
- You can reassign the input from within the dialog, by clicking the triangular button next to **Range 1** and selecting **Signal 2**. The fit line should now be tracking the **Signal 2** data plot.
3. Click the **Fit** button. The tool will automatically fit until convergence. It will also close the dialog and create the fitting report.



Using Change Parameters, modify the data range to be fitted

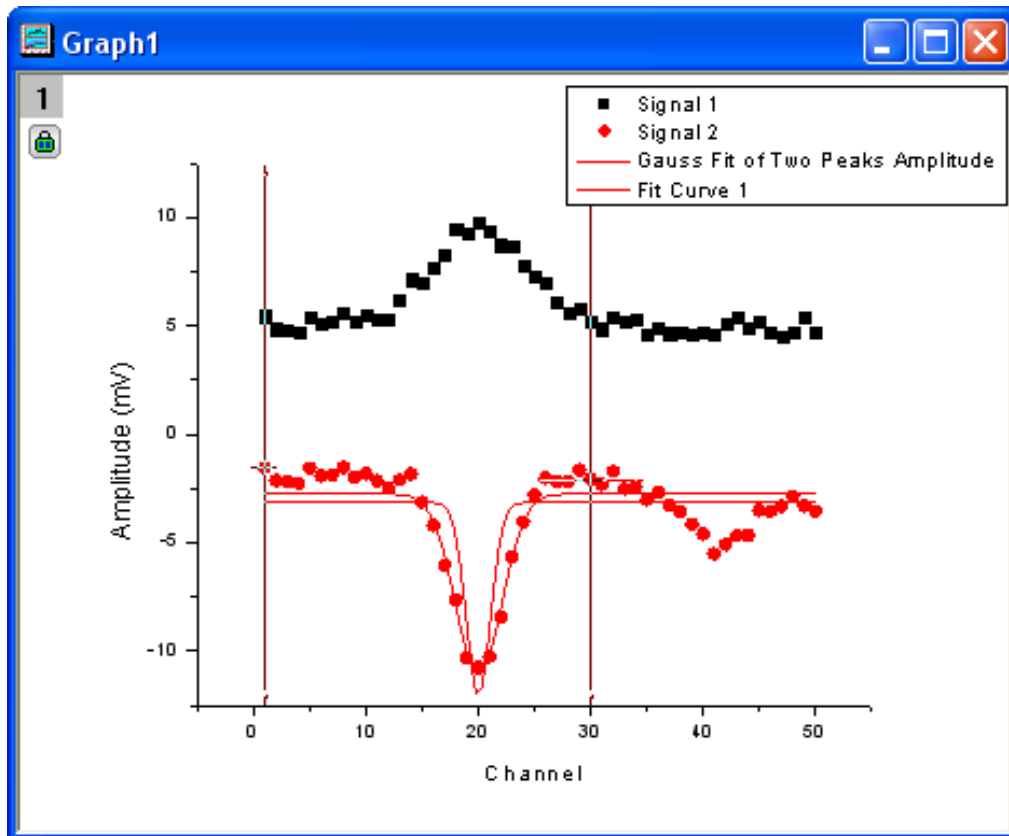
After fitting the entire range of **Signal 2**, you may want to update the fit just to fit a part of the curve. You can see that there are two peaks in **Signal 2** and we only want to fit the larger one on the left hand side of the graph. This section demonstrates how to do this.


1. First reopen the **Nonlinear Curve Fitting** dialog. You can click on the green lock on the upper left corner of the graph and select **Change Parameters** to open it.

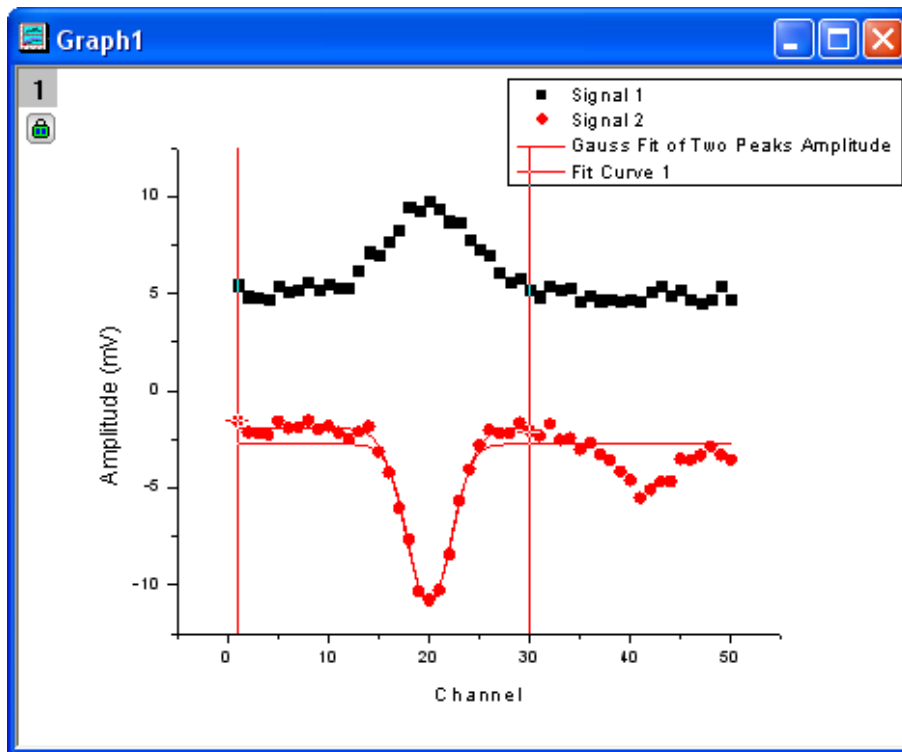


2. Change the fitting range by selecting **Data Selection** in the **Settings** tab, and then clicking the triangular button next to **Range 1** to choose **Select Range from Graph**. The Nonlinear Curve Fitting dialog will automatically be minimized.

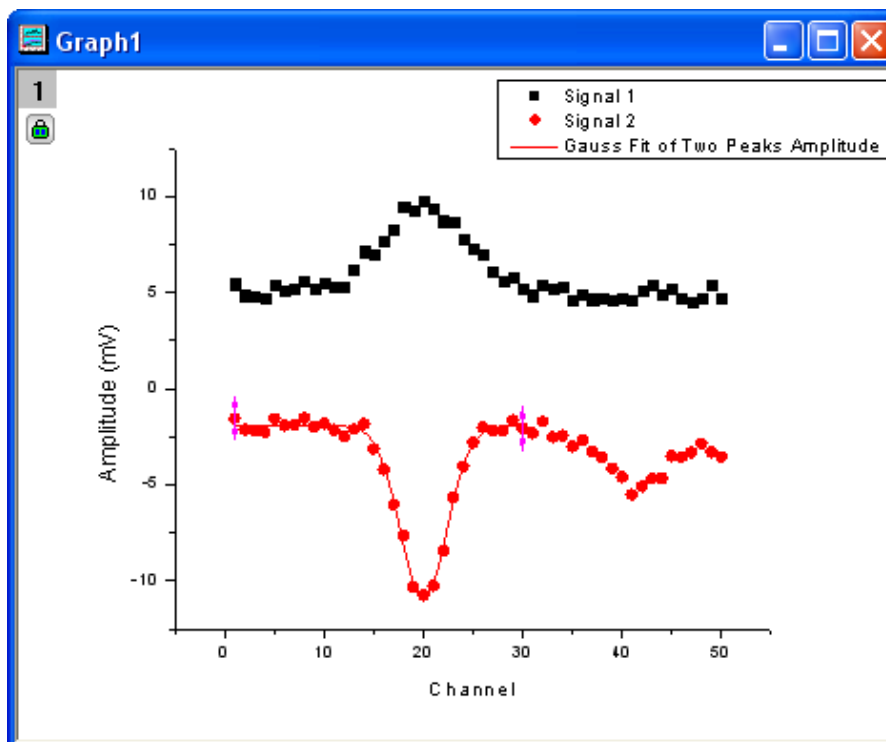
3. Drag the vertical line on the *right* side of the graph to leave out the small peak, as shown in the following image.



4. Maximize the **Nonlinear Curve Fitting** tool and click the **Fit Until Converged** button . As in the image below, you will find that the new fit curve is now only over the selected range, and it fits the baseline better.



5. Click the **OK** button and you will get the following result:



Nonlinear Fitting with System Function

Summary

The NLFit dialog is an interactive tool which allows you to monitor the fitting procedure during the non-linear fitting process. This tutorial fits the Michaelis-Menten function, which is a basic model in Enzyme Kinetics, and shows you some basic features of the NLFit dialog. During the fitting, we will illustrate how to perform a Global Fit, which allows you to fit two datasets simultaneously and share some parameter values.

Minimum Origin Version Required: Origin 8.0 SR6


What you will learn

This tutorial will show you how to:

- Import a single ASCII file
- Perform a global fit with shared parameters
- Select a fitting range and fit part of the data
- Use the Command Window to perform simple calculation

Steps

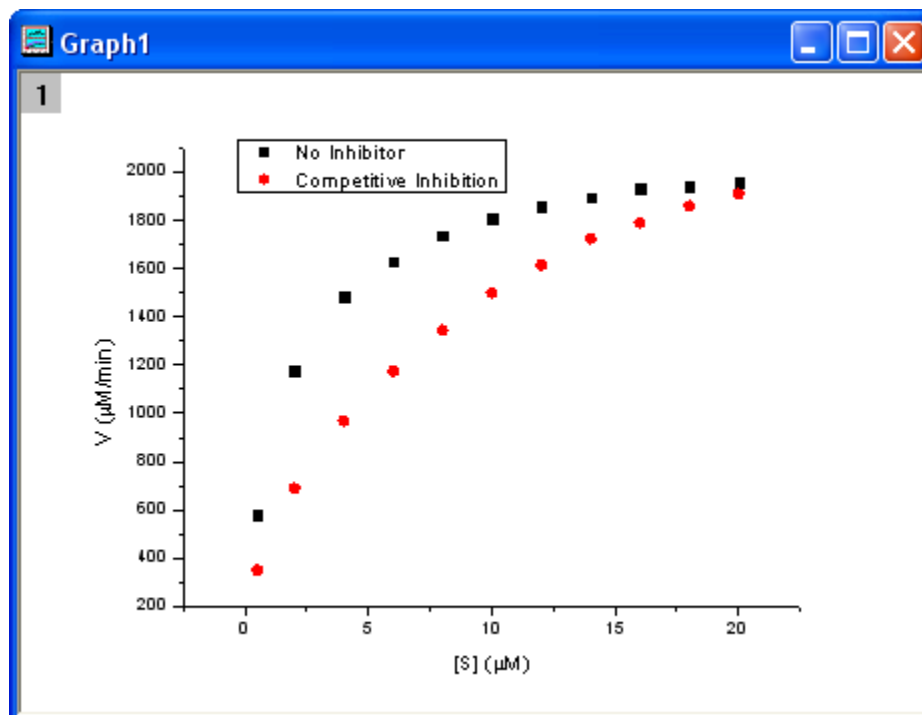
Import the file

- Open a new workbook.
- Click the *Import Single ASCII* button  to bring up the **Open** dialog. Browse to `\Samples\Curve Fitting` folder and select the file *Enzyme.dat*. Make sure to check the **Show Options Dialog** checkbox at the bottom of the dialog, and then click **Open**.
- In the **impASC** dialog, expand **Import Options: Header Lines** nodes, and select **3** from *Comments From* drop down.
- Click OK to import the file.

	A(X)	B(Y)	C(Y)
Long Name	[S]	V	V
Units	μM	$\mu\text{M}/\text{min}$	$\mu\text{M}/\text{min}$
Comments	Substrate	No Inhibitor	Competitive Inhibition
Sparklines			
1	0.5	580	350
2	2	1180	690
3	4	1485	970
4	6	1630	1175
5	8	1740	1345
6	10	1810	1500

Plotting the Data

- Highlight columns B & C and plot as a scatter plot by clicking the button.



Fitting Michaelis-Menten Function

The single-substrate Michaelis-Menten function:

$$v = \frac{V_{max}[S]}{K_m + [S]}$$

is a basic model in enzyme kinetics study, where v is the reaction velocity, $[S]$ is the substrate concentration, V_{max} is the maximal velocity and K_m represents the Michaelis constant. We can determine the V_{max} and K_m value, which are important enzyme properties, by fitting M-M function on v vs. $[S]$ curve.

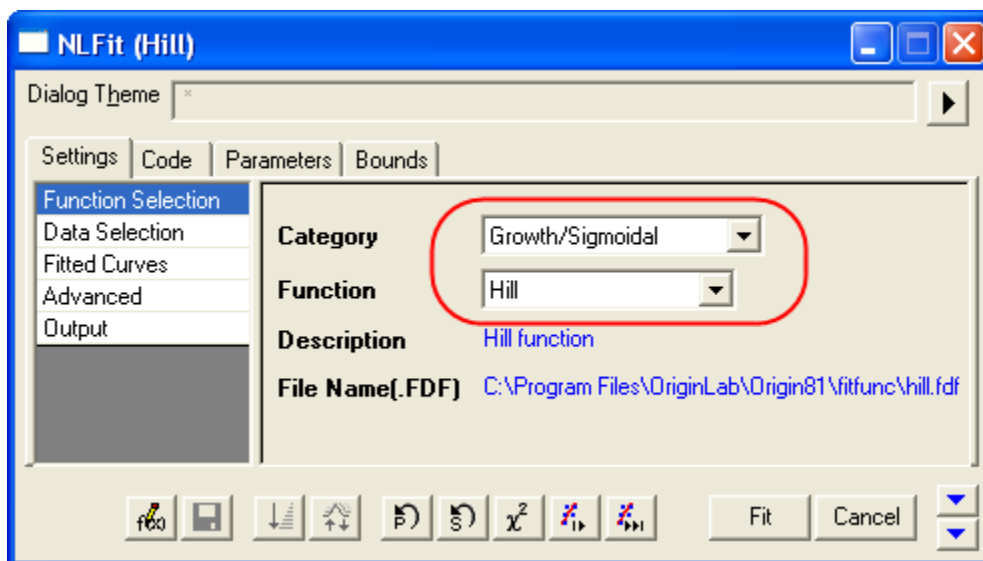
There is no M-M fitting function in Origin; however, we can use a more general model, the built-in *Hill* function to fit:

$$v = V_{max} \frac{x^n}{k^n + x^n}$$

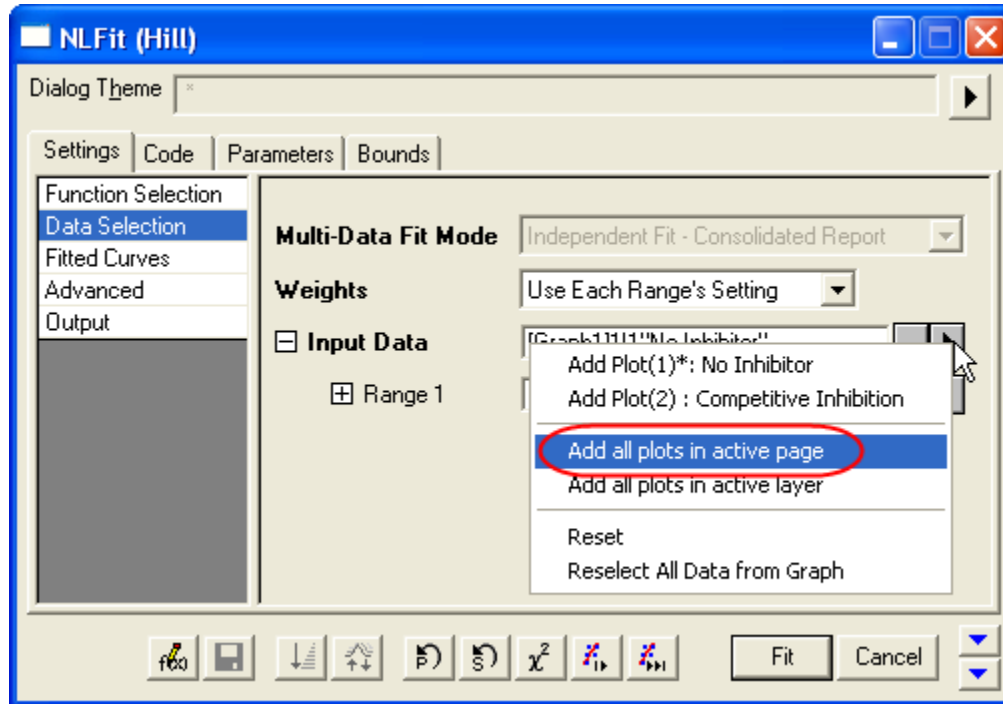
where n means the cooperative sites. For single-substrate model, we can just fix $n = 1$ during fitting and it will become the simplest form, the M-M function.

There are two curves, reaction without Inhibitor and reaction with Competitive Inhibitor in the graph, and the NLFit tool can fit these two curves simultaneously. Since for competitive inhibition reaction, the maximum velocity is the same with no inhibition reaction, we can share the V_{max} value during the fitting procedure, which can be implemented by a *Global Fit*.

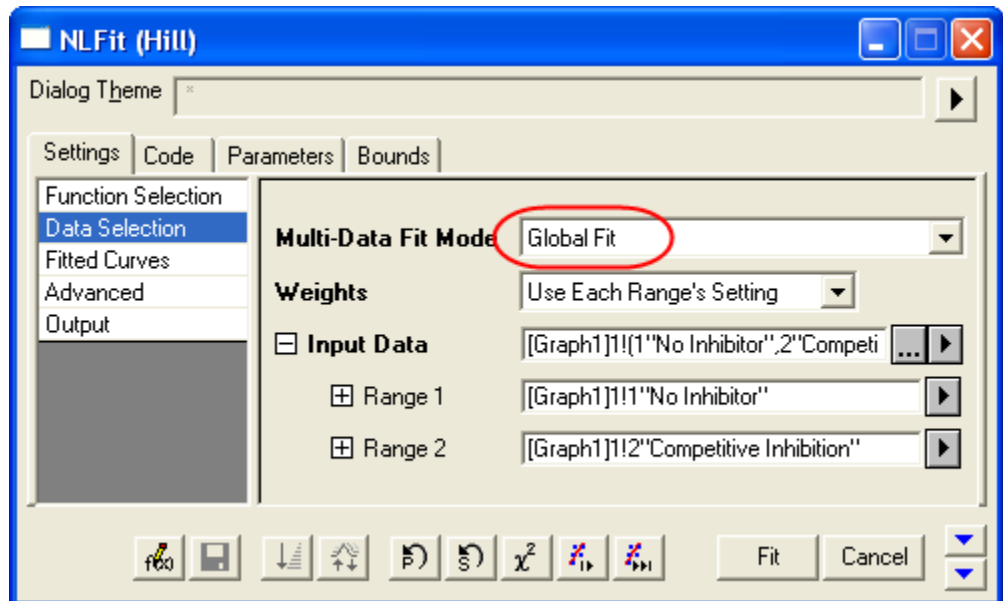
- With the graph active, select the menu item **Analysis: Fitting: Nonlinear Curve Fit** to bring up the NLFit dialog. Select *Hill* function from *Growth/Sigmoidal* category on the **Settings: Function Selection** page.



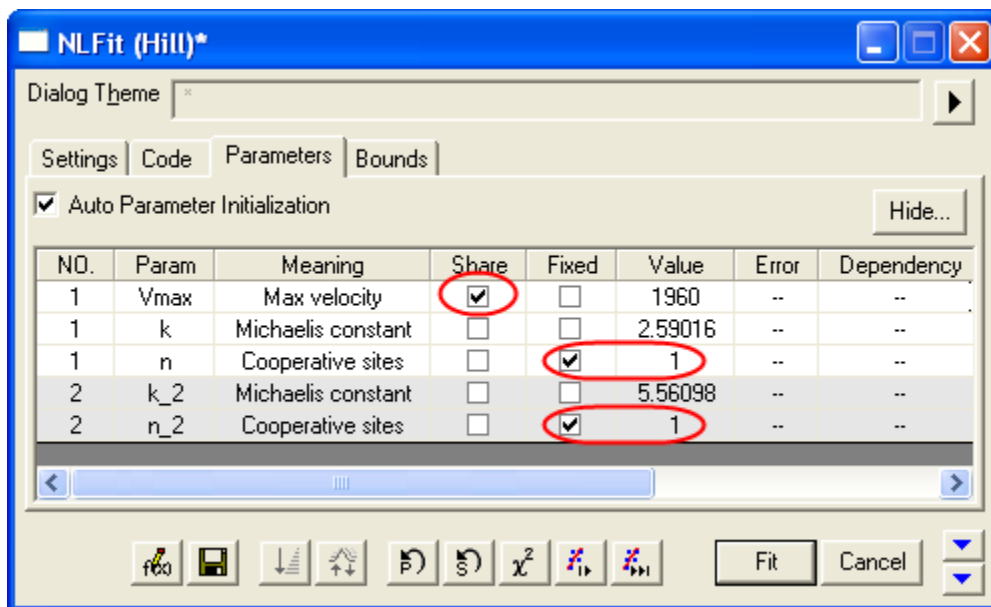
- On **Settings: Data Selection** page, click the triangular button next to the **Input Data** and choose **Add all plots in active page** to set the data.



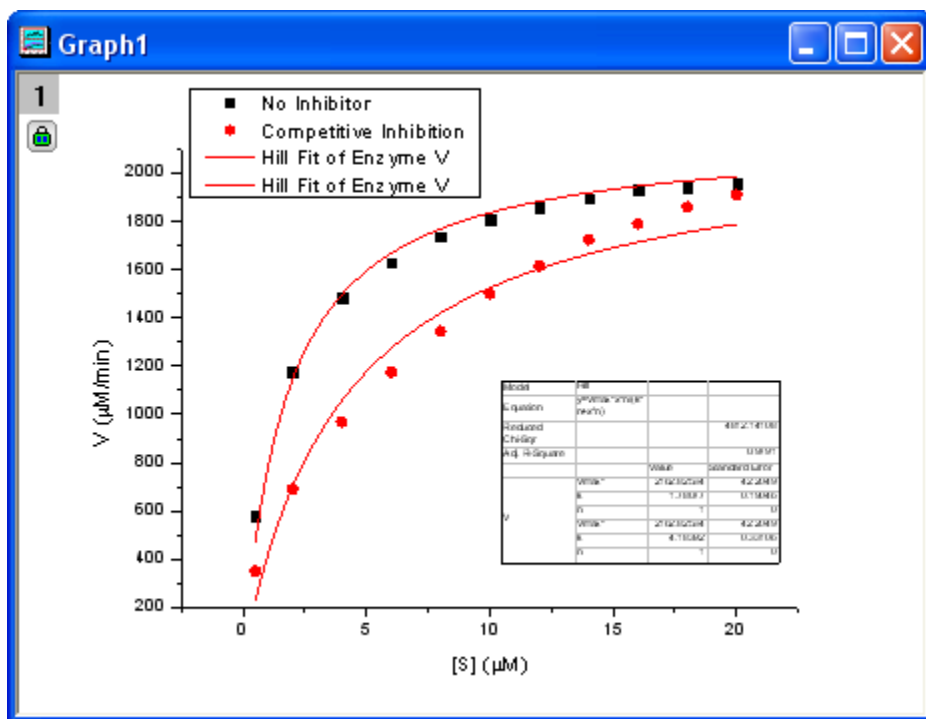
- Select *Global Fit* from **Multi-Data Fit Mode** drop-down list on the **Settings: Data Selection** page.



- Switch to the **Parameters** tab, check the **Share** box on the row V_{max} . These *Share* check boxes are only available when using *Global Fit* mode. Check the **Fixed** box for n and n_2 , and make sure their values are 1.



After that, click the *Fit* button to generate reports. The fit result will also be pasted on the original graph. (We just show the parameter values in the following figure.)



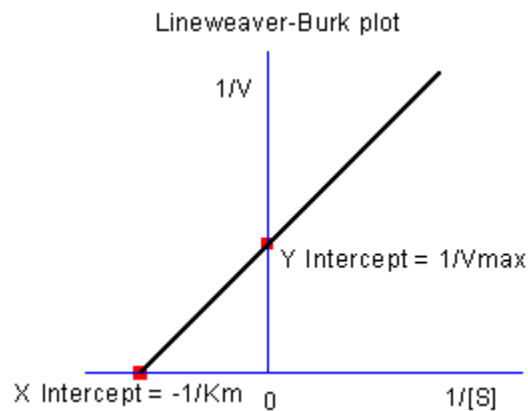
From the fit result, we can conclude that the maximum velocity is about $2160 \mu\text{M} / \text{min}$. and K_m for no inhibitor and competitive inhibitor model is $1.78 \mu\text{M}$ and $4.18 \mu\text{M}$, respectively.

Fitting Lineweaver-Burk Plot


As we know, the model parameters can also be estimated by the Lineweaver-Burk or double-reciprocal plot. The Lineweaver-Burk plot takes the reciprocal of both sides of the M-M function and plots by $1/v$ vs. $1/[S]$:

$$\frac{1}{v} = \frac{1}{V_{max}} + \frac{K_m}{V_{max}[S]}$$

This is actually a linear function:



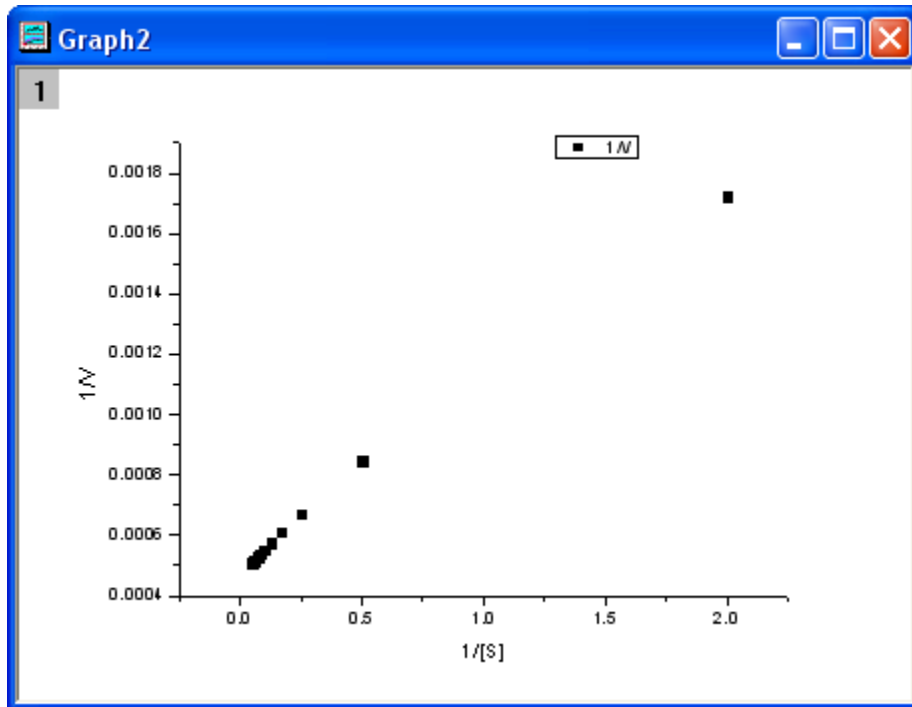
We will use the No Inhibitor data to illustrate how to calculate K_m and V_{max} by L-B plot.

- Go back to the raw data worksheet and add two more columns by clicking the  button. Right-click on column D and select **Set As: X** from the context fly-out menu to set it as an X column. Right-click on column D again and select **Set Column Values** to bring up the **Set Values** dialog. In the dialog edit box, enter: $1/Col(A)$ and set the **Recalculate** mode as *None*, since we don't need to auto update the reciprocal values in this example.


Similarly, set column E's values as $1/Col(B)$. Enter the long name for column D & E as $1/[S]$ & $1/V$, respectively. And then we have:

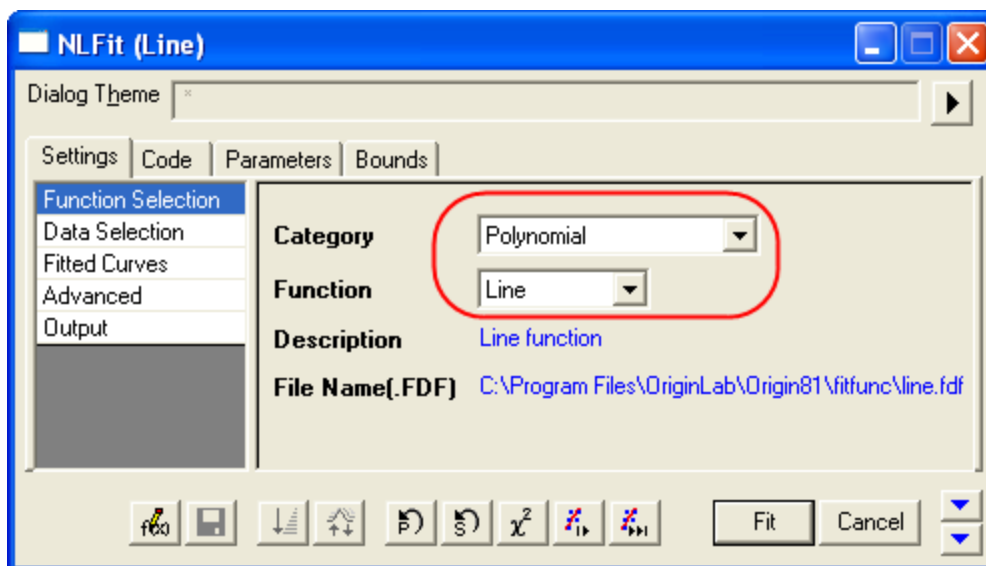
	A(X1)	B(Y1)	C(Y1)	D(X2)	E(Y2)
Long Name	[S]	V	V	1/[S]	1/V
Units	μM	$\mu\text{M}/\text{min}$	$\mu\text{M}/\text{min}$		
Comments	Substrate	No Inhibitor	Competitive Inhibition		
Sparklines					
1	0.5	580	350	2	0.00172
2	2	1180	690	0.5	8.47458E-4
3	4	1485	970	0.25	6.73401E-4
4	6	1630	1175	0.16667	6.13497E-4
5	8	1740	1345	0.125	5.74713E-4
6	10	1810	1500	0.1	5.52486E-4

- Highlight columns D & E and click  button to create a scatter plot.

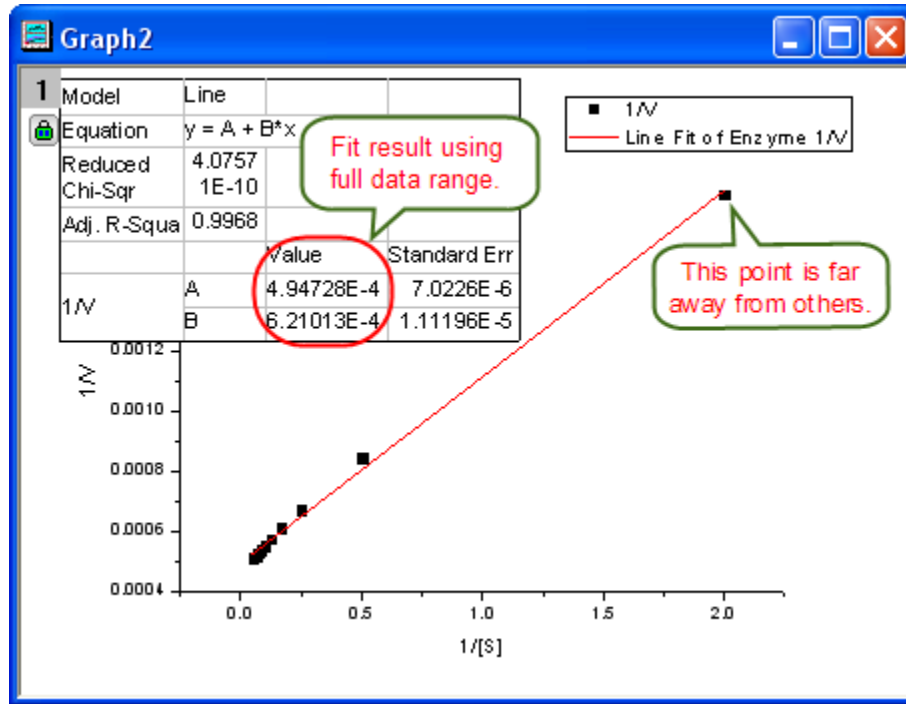


From the above equation, we know there is a linear relationship between $1/v$ and $1/[S]$, so we can use the NLFit tool to fit a straight line on this plot. (You can also use the Fit Linear tool from **Analysis: Fitting: Fit Linear**)

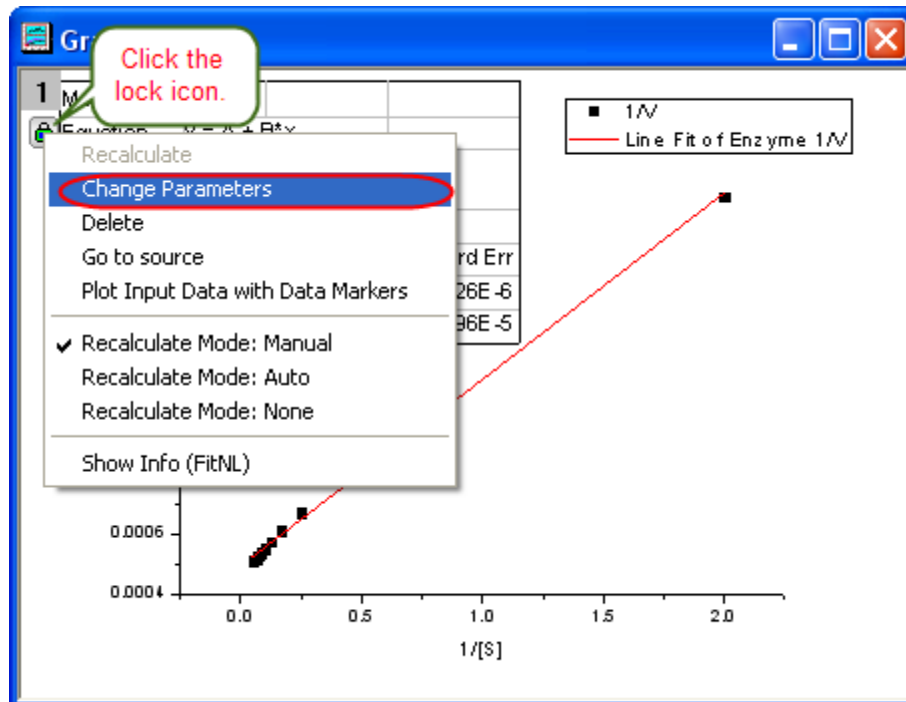
- Bring up the NLFit dialog again, select *Line* function from *Polynomial* category, and then click the *Fit* button  directly to generate results.




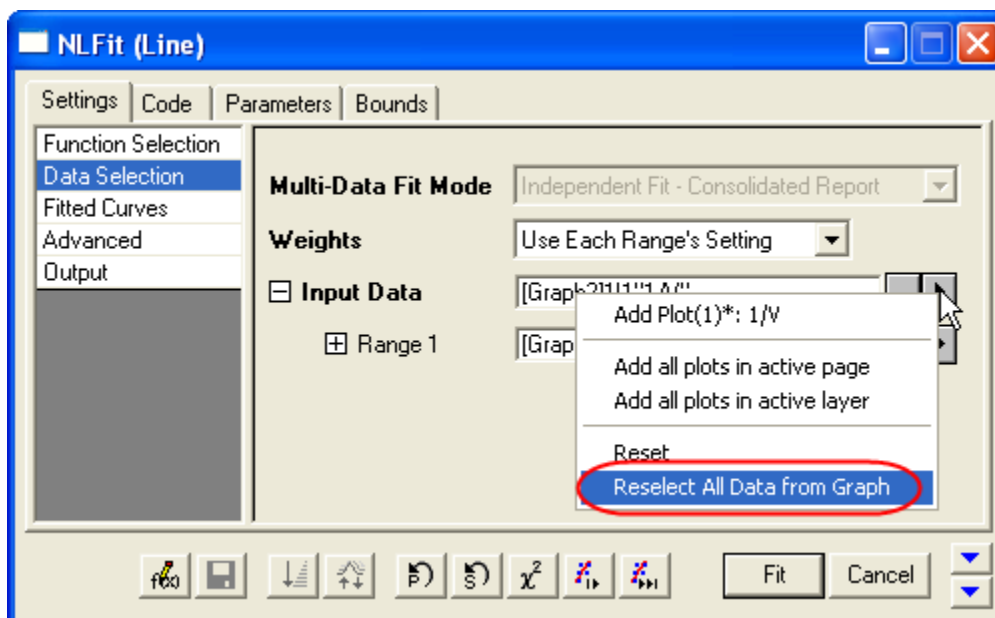
From the plot, one may doubt that this is the best fit curve since there is a point located far away. Actually, the right side of L-B plot is low substrate concentrations area, the measurement error may be large, so we'd better exclude these points during fitting.




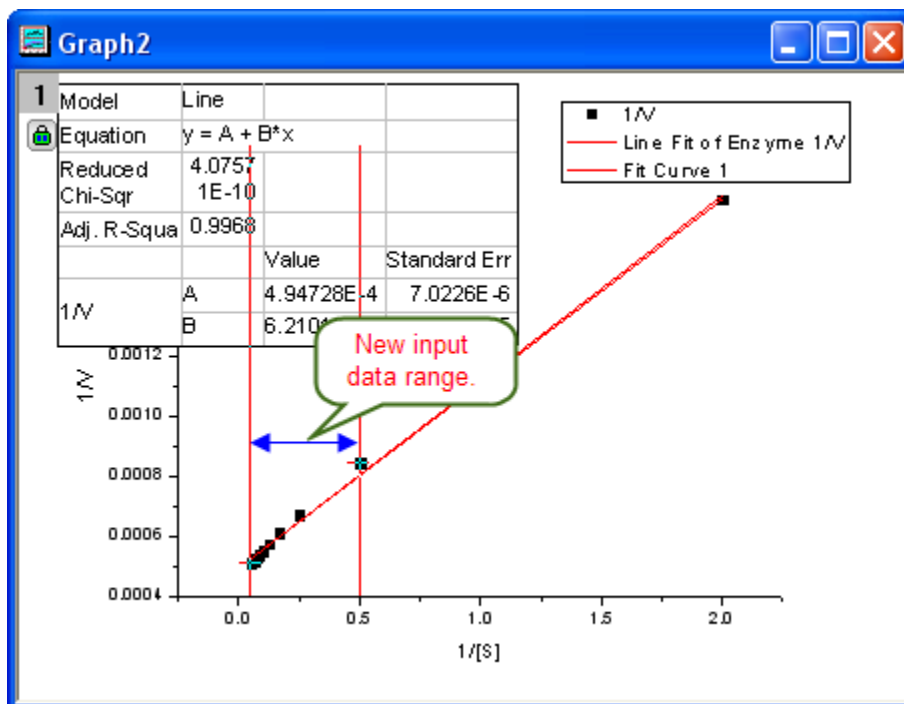
- Click the lock icon on the graph upper-left corner, and select *Change Parameters* to bring back NLFit dialog.



In **Settings: Data Selection** page, click the  button on **Input Data** node, and then choose *Reselect All Data from Graph* from fly-out menu.



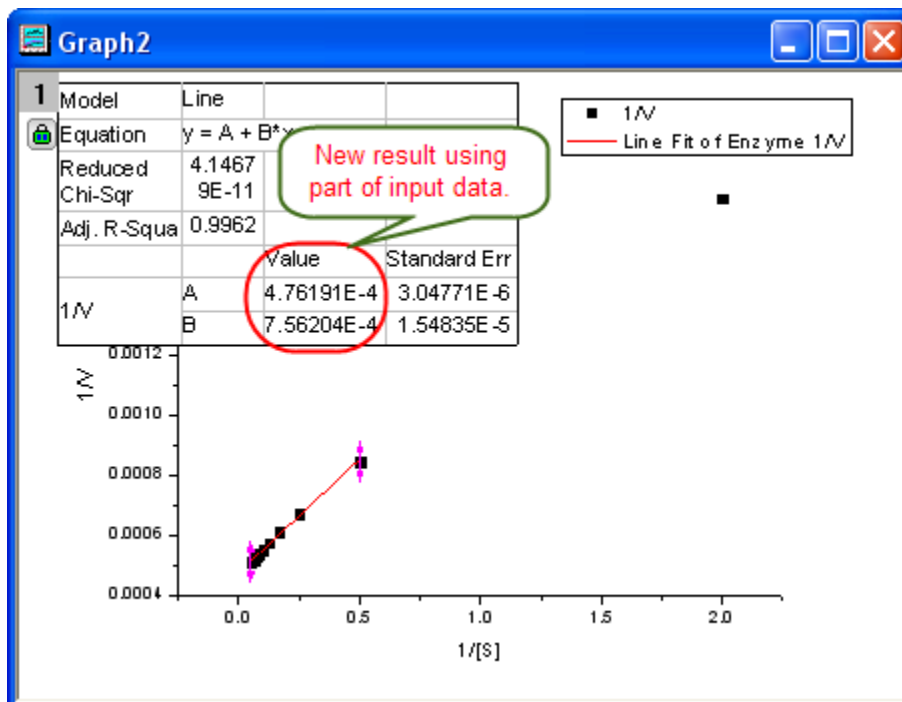
Then the NLFit dialog rolls up and your cursors become  when you move to the graph page. Click and draw a rectangle to select data points you want to fit. The input range is labeled by vertical lines. You can also click-and-move these lines to change the input range.



Click the  button on **Select Data in Graph** window to go back to NLFit dialog.



- Click the *Fit* button on the NLFit dialog to recalculate the result. You can see from the graph that the report table was updated.



- Since the intercept of the fitted curve is $1/V_{max}$, it is equal to $4.76191E-4$ in this example. To get the V_{max} value, select **Window: Command Window** to open the command window, type

$1/4.76191E-4 =$
and press ENTER:

```
//--2009-8-19 12:56
1/4.76191E-4=;
>>1/4.76191E-4=
1/4.76191E-4=2099.9976900025
>>
```

Origin returns the value 2099, which is close to what we got above, 2160. (When fitting the hill function above, we shared V_{max} when fitting two datasets. If you fit the No Inhibitor data only, this value will be closer.)

Global Fitting with Parameter Sharing

Summary

Global fit is one of the fit modes in Origin when fitting multiple curves. It will fit all datasets simultaneously, allowing parameter sharing. Compared to concatenate fit, which combine all datasets into one, global fitting performs chi-square minimization in a combined parameter space, so the parameter errors, DOF, npts and even parameter values may be different from a concatenated fit. Therefore global fitting is only appropriate/necessary if you want to share parameters.

Minimum Origin Version Required: Origin 8.0 SR6

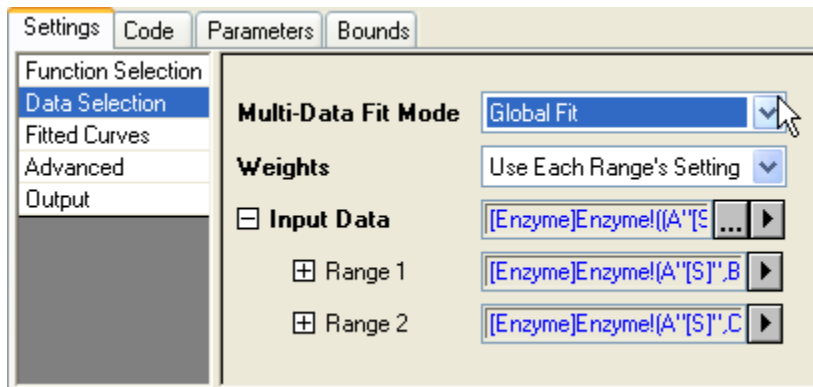
What you will learn

This tutorial will show you how to:

- Select multiple datasets for fitting.
- Select different fitting modes.
- Perform global fit with parameter sharing.

Steps

1. Start with a new workbook and import the file `\Samples\Curve Fitting\Enzyme.dat`.
2. Highlight column B and C and bring up the **NLFit** dialog from **Analysis: Fitting: Nonlinear Curve Fitting**. In the **Function Selection** page of **NLFit** dialog, choose **Hill** function from **Growth/Sigmoidal** category. Go to **Data Selection** page, and select **Global Fit** mode from **Multi-Data Fit Mode** drop-down list:



Then make sure the **Recalculate** mode is **Manual** in the **Advanced** page.

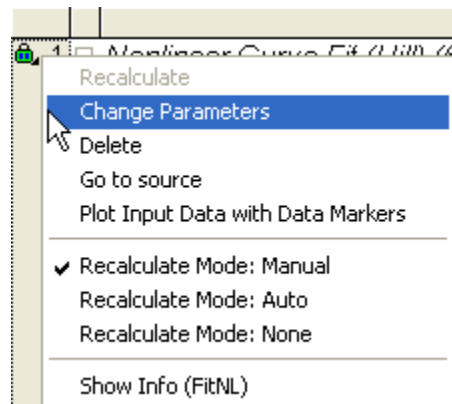
3. Active the **Parameters** tab. Check the **Fixed** checkbox for n and n_2 to fix their values to 1.

Param	Meaning	Share	Fixed	Value	Error
Vmax	Max velocity	<input type="checkbox"/>	<input type="checkbox"/>	1960	--
k	Michaelis constant	<input type="checkbox"/>	<input type="checkbox"/>	2.59016	--
n	Cooperative sites	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1	--
Vmax_2	Max velocity	<input type="checkbox"/>	<input type="checkbox"/>	1910	--
k_2	Michaelis constant	<input type="checkbox"/>	<input type="checkbox"/>	5.56098	--
n_2	Cooperative sites	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1	--

Click the **Fit** button to fit curves. You can see these results from the report worksheet:

		Value	Standard Error
V	Vmax	2091.96109	33.62032
	k	1.52432	0.13786
	n	1	0
	Vmax	2428.49265	81.97136
	k	5.86377	0.56737
	n	1	0

- Since the maximum velocity, V_{max} in this case, maybe the same. We now want to share this parameter value to fit. Click the lock icon in the report worksheet and select **Change Parameters** to bring back the **NLFit** dialog.



- In the **Parameters** tab, check the **Shared** checkbox for V_{max} .

Param	Meaning	Share	Fixed	Value	Error
Vmax	Max velocity	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1960	--
k	Michaelis constant	<input type="checkbox"/>	<input type="checkbox"/>	2.59016	--
n	Cooperative sites	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1	--
k_2	Michaelis constant	<input type="checkbox"/>	<input type="checkbox"/>	5.56098	--
n_2	Cooperative sites	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1	--

And then click the **Fit** button again to generate new results, you can see the *Vmax* values for both curves are the same. The asterisk in parameter name means that this parameter is shared:

		Value	Standard Error
V	Vmax*	2162.82534	42.2049
	k	1.78077	0.19046
	n	1	0
	Vmax*	2162.82534	42.2049
	k	4.18392	0.33106
	n	1	0

User Defined Fitting Function using Origin C

Summary

All the fitting functions in Origin are organized by **Fitting Function Organizer**. Beside the build-in functions, you can also create user-define functions in function organizer. Once a function is created, it can be accessed in the NLFit dialog. We will illustrate how to fit by user-define function below.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

- Create a user-define fitting function.

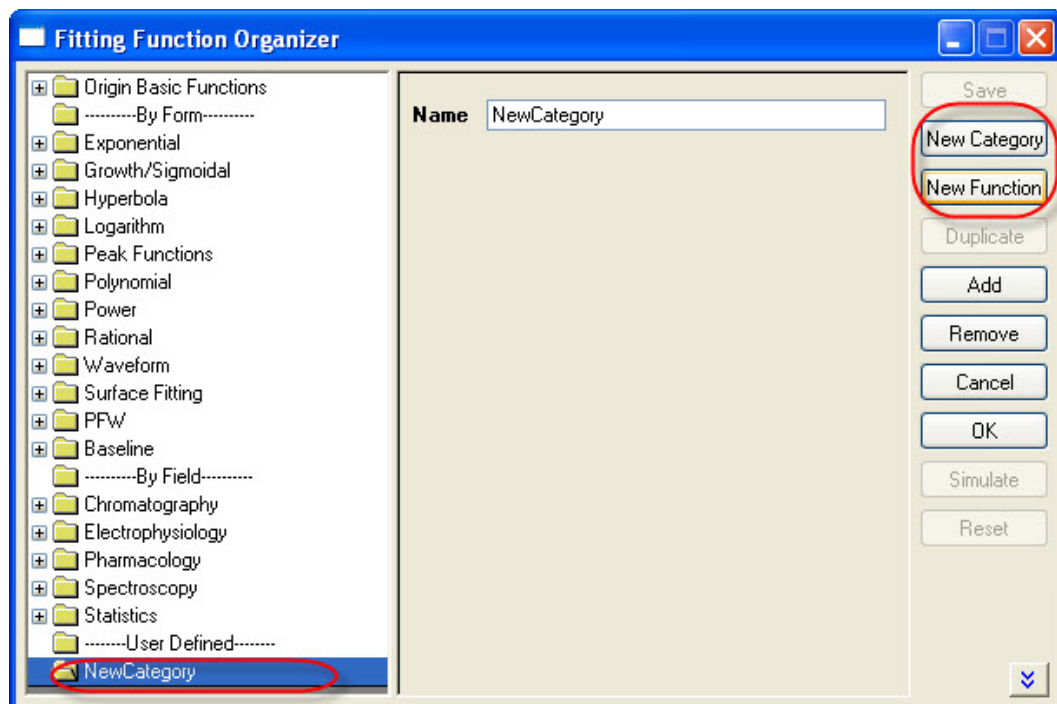
Example

We will illustrate how to define the following fitting function:

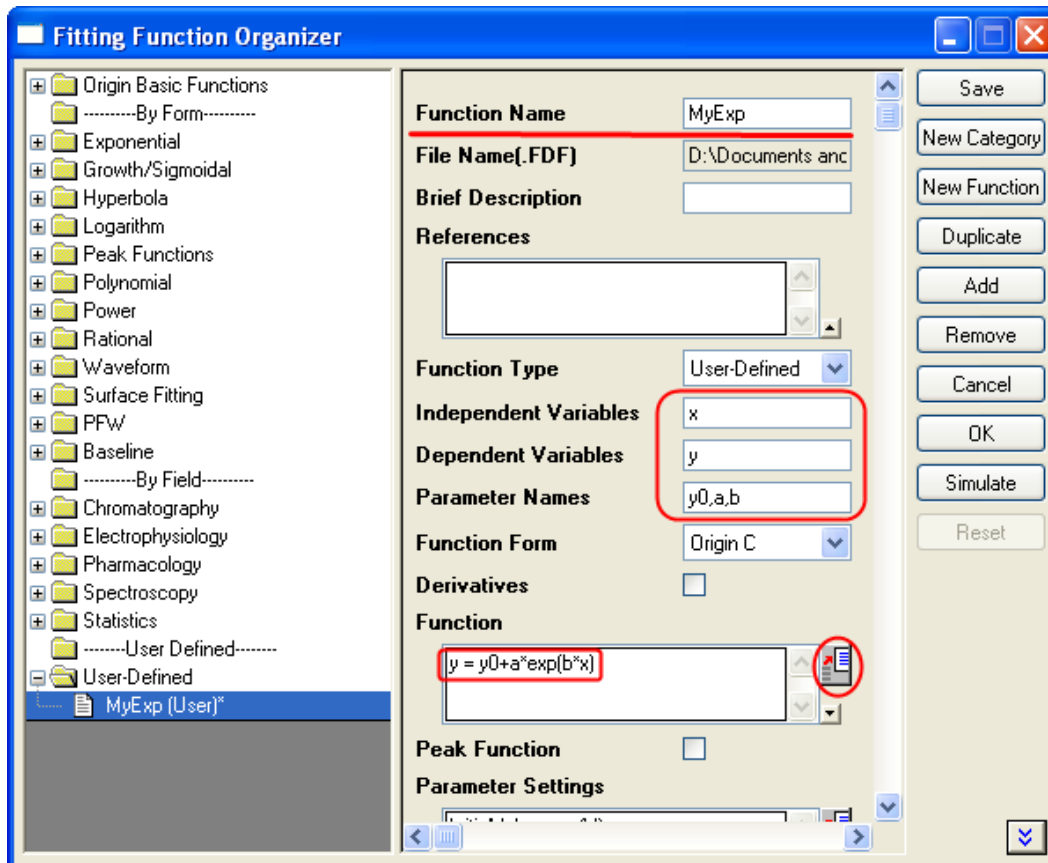
$$y = y_0 + ae^{bx}$$

Steps to define the function:

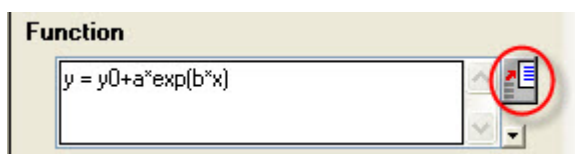
1. Select **Tools: Fitting Function Organizer** from menu (or press **F9**) to open the function organizer. Click the **New Category** button to create a function category, rename as *User-Defined* for example. Then press **New Function** button to create a **new function** under this category:



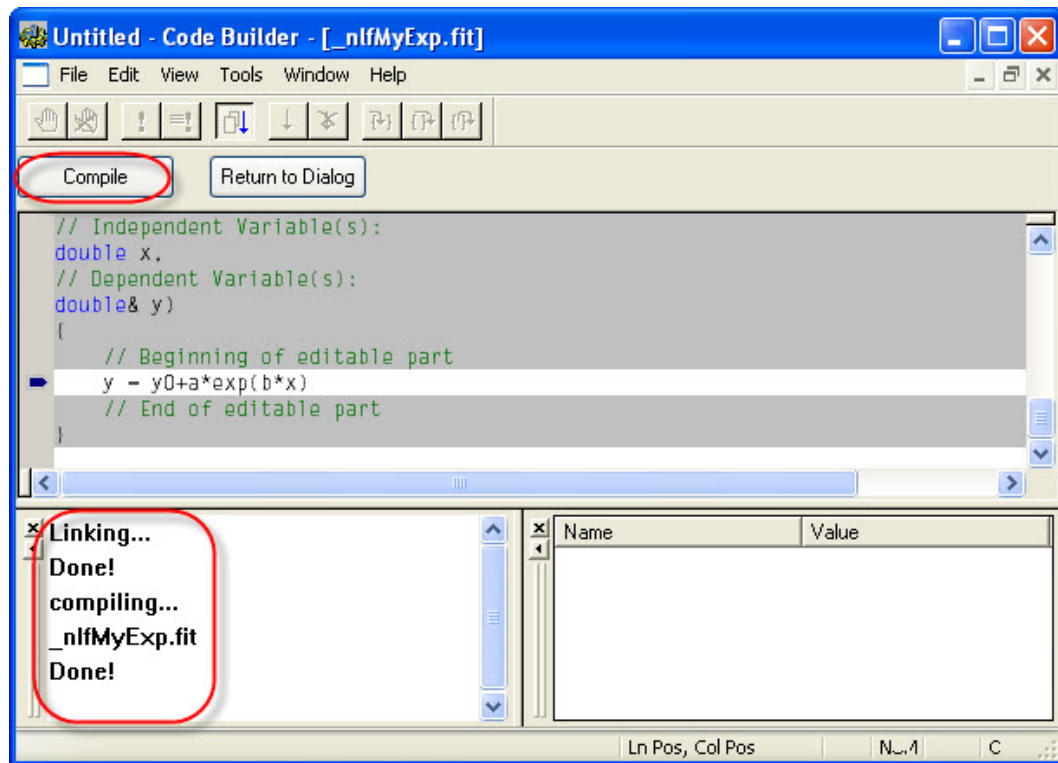
2. Enter function definition like the following image and **Save**:



3. To verify the correctness of the function, click the button beside the **Function** box to open Origin Code Builder:



In the **Code Builder**, click **Compile** button to compile the function. If passed, click **Return to Dialog** button to return **Fitting Function Organizer**.

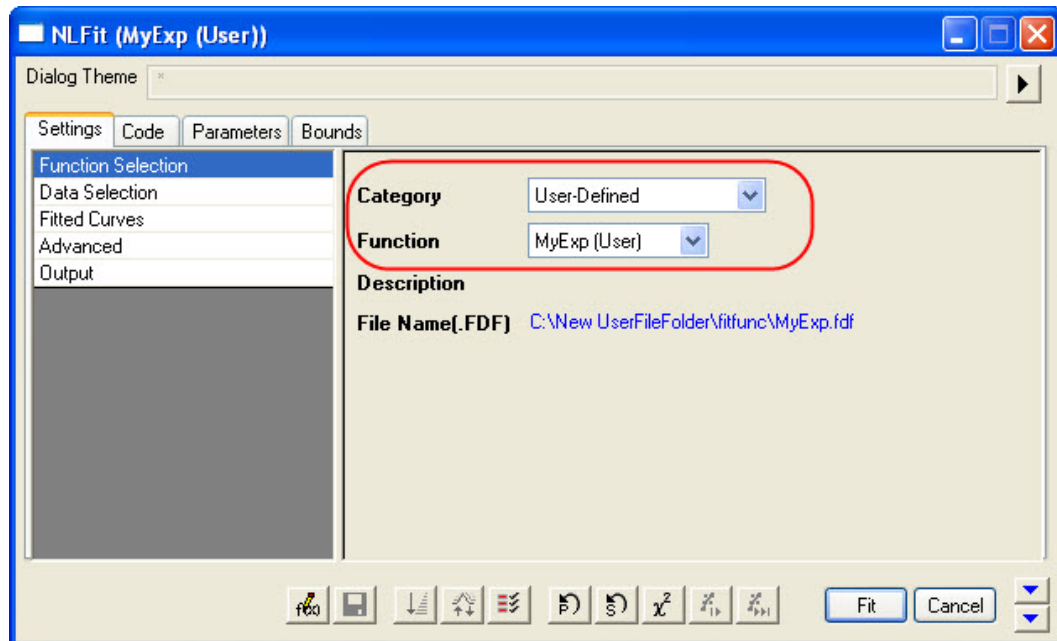


4. Click **Save** and **OK** to save the function and quite **Fitting Function Organizer**.

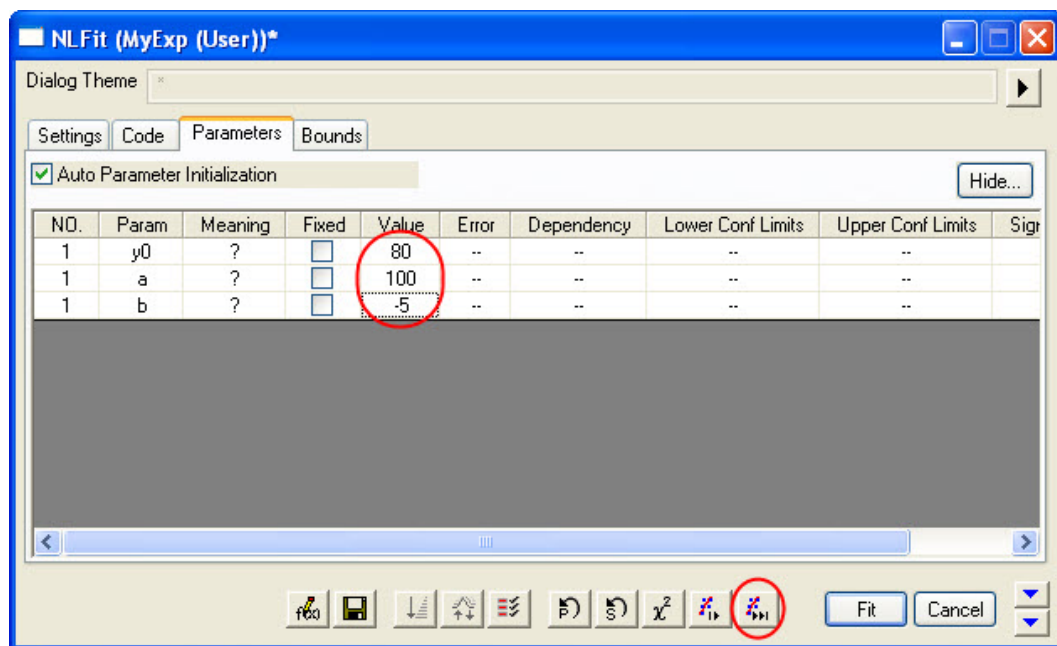
Fit data by the function:

1. Import \Samples\Curve Fitting\Exponential Decay.dat to Origin worksheet.
2. Highlight column B and select **Analysis: Fitting: Non-linear Curve Fit** from menu to bring up the **NLFit** dialog.

3. Select the function just defined in **Settings** tab, **Function Selection** page:

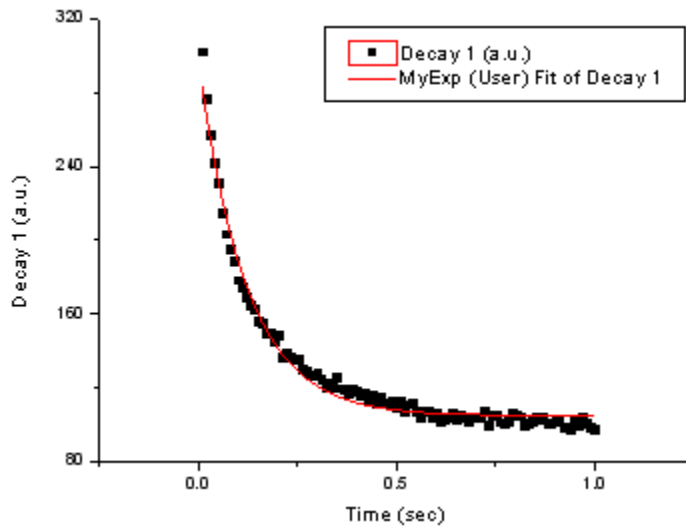


4. Switch to **Parameters** tab, enter 80, 100, -5 on the **Value** column as initial values for y_0 , a , b :



5. When the fit converged, click **OK** button to generate fitting reports.

From the Fitted Curves Plot we see the fitting is fine.



And the fitting function is $y = 104.85968 + 193.3244 * \exp(-8.273 * x)$

Parameters

		Value	Standard Error
Decay 1	y0	104.85968	0.69005
	a	193.3244	3.10614
	b	-8.273	0.21726

Iterations Performed = 4

Total Iterations in Session = 4

Fit converged - tolerance criterion satisfied.

Fitting One Dataset as a Function of Other Datasets

Summary

Sometimes, one may want to perform "Dataset Fitting", that is, the output may be composed of one or several datasets, like:

$$\text{Output} = A1 * \text{Dataset1} + A2 * \text{Dataset2}$$

For example, you may want to analyze a composite spectrum to find the contributions/ratio from individual component spectra. This can be accomplished either by defining multiple independent variables or by calculating the "combination" inside the fitting function.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Perform "Dataset Fitting"
- Define multiple independent variable fitting function

Steps

Import the *Composite Spectrum.dat* file from the `\Samples\Curve Fitting\` folder. In this sample data, we can see that column A is the index, columns B and C are the values for the spectrum of components A and B. Column D contains values obtained after reading a composite spectrum of components A and B. By fitting column D to an equation determined by the component spectra of the pure forms of columns B and C, the coefficients for the contributions of B and C (call them c1 and c2 respectively) can be found. (Note: In this case, we supposed the independent and dependent variables have the same size. If not, interpolation is need.)

Bring up the **Fitting Function Organizer** and define a new fitting function as follow:

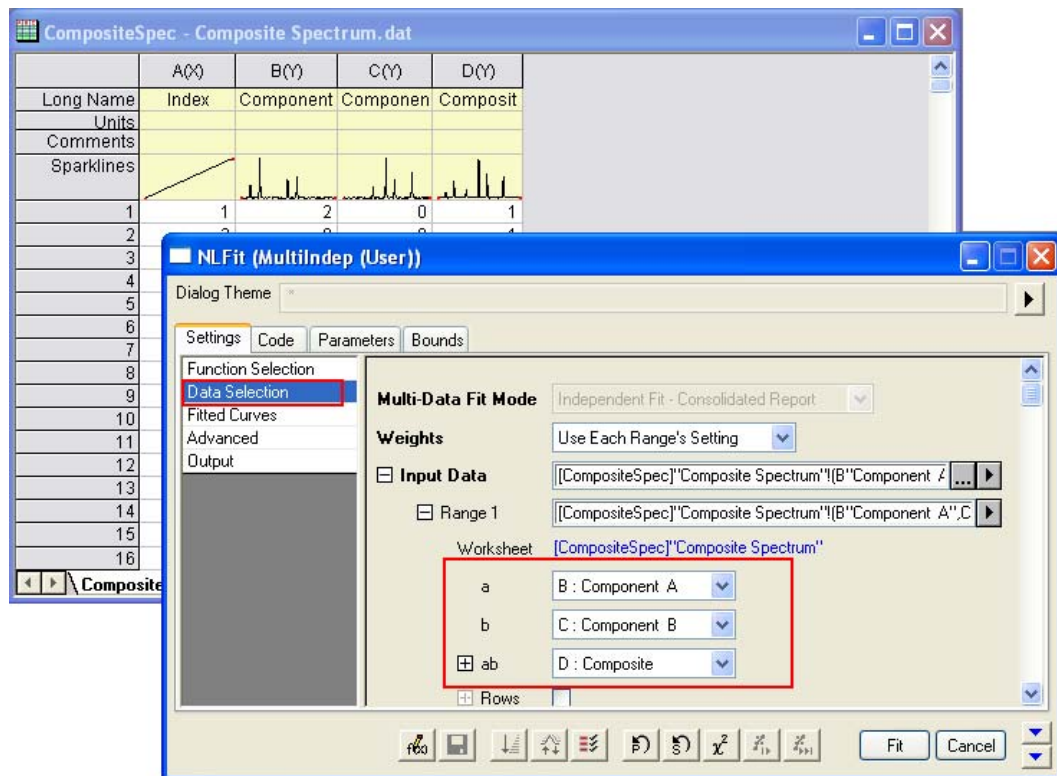
Function Name:	MultiIndep
Function Type:	User-Defined
Independent Variables:	a, b
Dependent Variables:	ab
Parameter Names:	C1, C2
Function Form:	Origin C
Function:	$ab = C1*a + C2*b;$

Initialize both C1 and C2 to 1 in the **Parameter Initialization** edit box by entering:

C1=1;

C2=1;

Save the fitting function and close **Fitting Function Organizer**. Highlight **ONLY** Column D and bring up the **NLFit** dialog, specify the input datasets in the **Data Selection** page as follow:



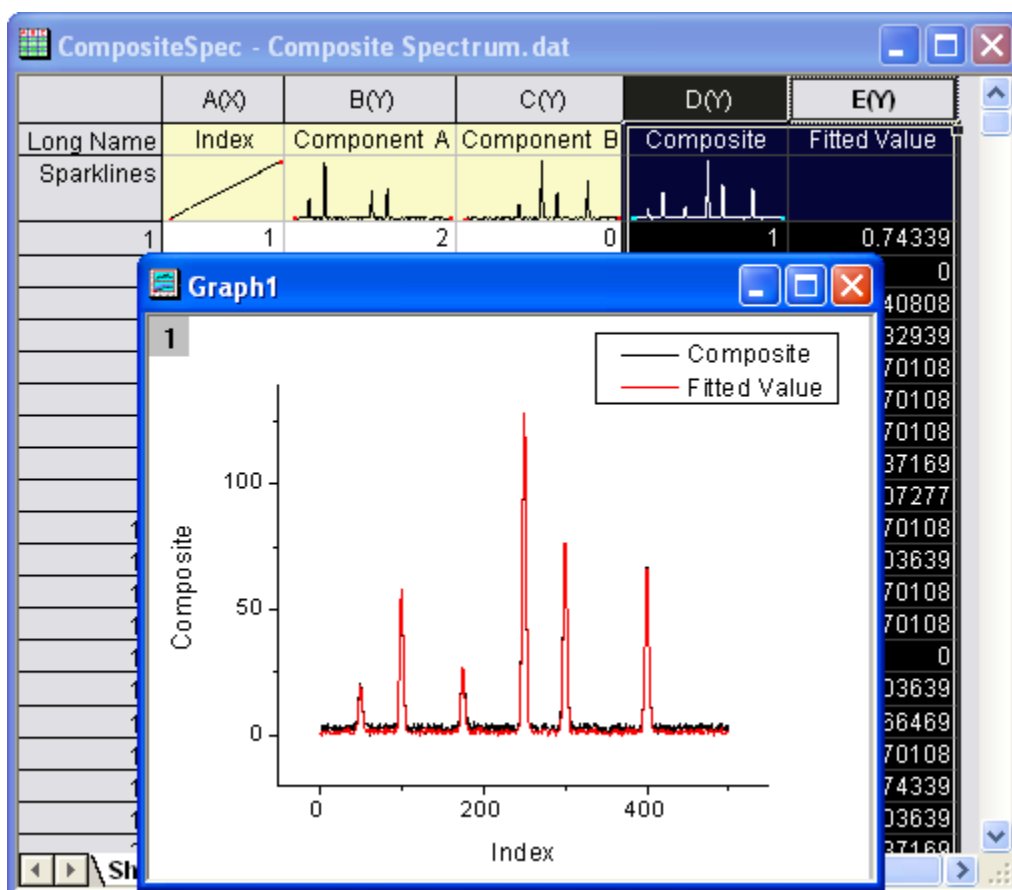
Then you can click the **Fit** button to generate results.

Results

You are supposed to get these results:

	Value	Standard Error
C1	0.37169	0.00483
C2	0.66469	0.0047

To verify the fitted results, you can add a new column and Copy + Paste the fitted value, which comes from the fitted Y in the worksheet *FitNLCurve1*, into it. Then Highlight the *Composite* and the fitted data and plot a line graph to see how good the fit is:



Fitting With Multiple Independent Variables

Summary

The **Function Organizer** tool can be used to create user-defined functions with more than one independent or dependent variable. The **NLFit dialog** can then be used to fit with such functions. The preview window in the fitter dialog is capable of plotting only one quantity versus another, however even if the preview does not *make sense*, the fitting process will correctly proceed once proper data and parameter assignments have been made.

Note that if you wish to fit multiple independent variables with an equation of the type $y = A0 + A1 * x1 + A2 * x2 + \dots$ you can make use of the Multiple Regression tool instead of the nonlinear fitter dialog.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a user-defined fitting function with two independent variables and one dependent variable
- Fit with that function in NLFit

Steps

1. Start with a new workbook and import the file `\Samples\Curve Fitting\Activity.dat`.
2. Select **Tools: Fitting Function Organizer** from menu (or press **F9**) to bring up the **Fitting Function Organizer** and define a new fitting function named **MultiIndep** in **NewCategory** (create the category if not exist) as follow:

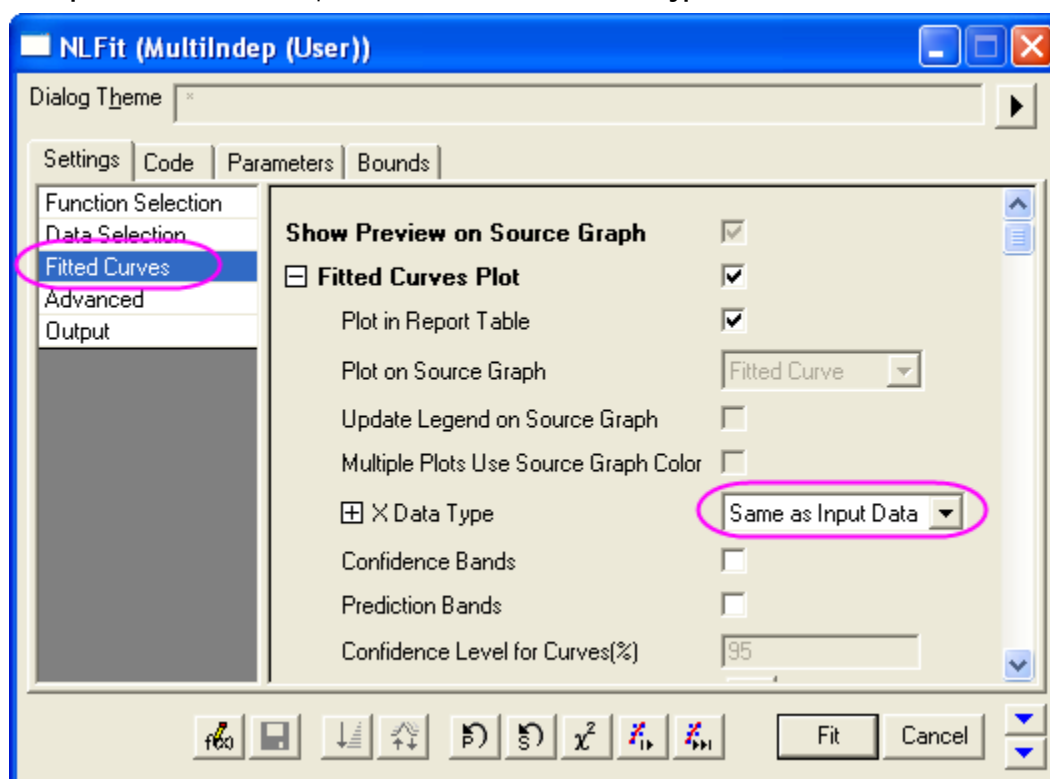
Function Name:	MultiIndep
Function Type:	User-Defined
Independent Variables:	substr, inhib
Dependent Variables:	act
Parameter Names:	ki, km, vm
Function Form:	Origin C
Function:	double mix = inhib / ki; act = vm * substr / (km + (1 + mix) * substr);

3. For more details about User Defined Fitting Function please refer to User Defined Fitting Function using Origin C.
4. Highlight **ONLY** column C and select **Analysis: Fitting: Non-linear Curve Fit** from menu to bring up the NLFit dialog. Select the function **MultiIndep** from **NewCategory** on the **Settings:**

Function Selection page. Set the input datasets in the **Data Selection** page as follow:



5. Select the **Fitted Curves** page and expand the **Fitted Curves Plot** branch. Then select **Sample as Input Data** from the drop-down list next to the **X Data Type** branch.



6. Select **Parameters Tab** and set the initial values as follow:

Settings	Code	Parameters	Bounds	
Automatic Parameter Initialization is enabled.				
Param	Meaning	Fixed	Value	Error
ki	?	<input type="checkbox"/>	0.01	--
km	?	<input type="checkbox"/>	1	--
vm	?	<input type="checkbox"/>	100	--

7. Click **Fit** button to generate the fitting reports. You can see these results from the report worksheet as below:

Parameters

		Value	Standard Error
Activity	ki	0.0373	0.00233
	km	7.30567	0.71748
	vm	653.1116	22.39698

Statistics

	Activity
Number of Points	18
Degrees of Freedom	15
Reduced Chi-Sqr	155.36102
Residual Sum of Squares	2330.41531
Adj. R-Square	0.98357
Fit Status	Succeeded(100)

From the Statistics table we can know that the fitting is fairly successful.

User Defined Fitting Function using GNU Scientific Library

This article demonstrates how to use GSL function as fit function.

Minimum Origin Version Required: Origin 8.0 SR6

1. We will fit the sample Data below by the following model:

0.1	0.10517
0.2	0.2214
0.3	0.34986
0.4	0.49182
0.5	0.64872
0.6	0.82212
0.7	1.01375
0.8	1.22554
0.9	1.4596
1	1.71828
1.1	2.00417
1.2	2.32012
1.3	2.6693
1.4	3.0552
1.5	3.48169
1.6	3.95303
1.7	4.47395
1.8	5.04965
1.9	5.68589
2	6.38906
2.1	7.16617
2.2	8.02501
2.3	8.97418
2.4	10.02318
2.5	11.18249
2.6	12.46374
2.7	13.87973
2.8	15.44465
2.9	17.17415
3	19.08554
3.1	21.19795
3.2	23.53253

2. Add the file `ocgsl.h` in **(Origin installation folder)\OriginC**, before next step, first make sure the `gsl` dlls are copied to this same location, see Calling GNU Scientific Library.

`ocgsl.h`

```
#pragma dll(libgsl, header)
// this is OC special pragma,
// header keyword is to indicate libgsl.dll is in same
location as this file

#define GSL_EXPORT          // for OC, this is not needed, so make
it empty

// you can directly search and copy gsl function prototypes
here

typedef double (* FUNC)(double x, void * params);

struct gsl_function_struct
{
    FUNC function;
    void * params;
};

typedef struct gsl_function_struct gsl_function ;

typedef struct
{
    size_t limit;
    size_t size;
    size_t nrmax;
    size_t i;
    size_t maximum_level;
    double *alist;
    double *blist;
    double *rlist;
    double *elist;
    size_t *order;
    size_t *level;
}
gsl_integration_workspace;

GSL_EXPORT gsl_integration_workspace
*gsl_integration_workspace_alloc (const size_t n);

GSL_EXPORT void gsl_integration_workspace_free
(gsl_integration_workspace * w);

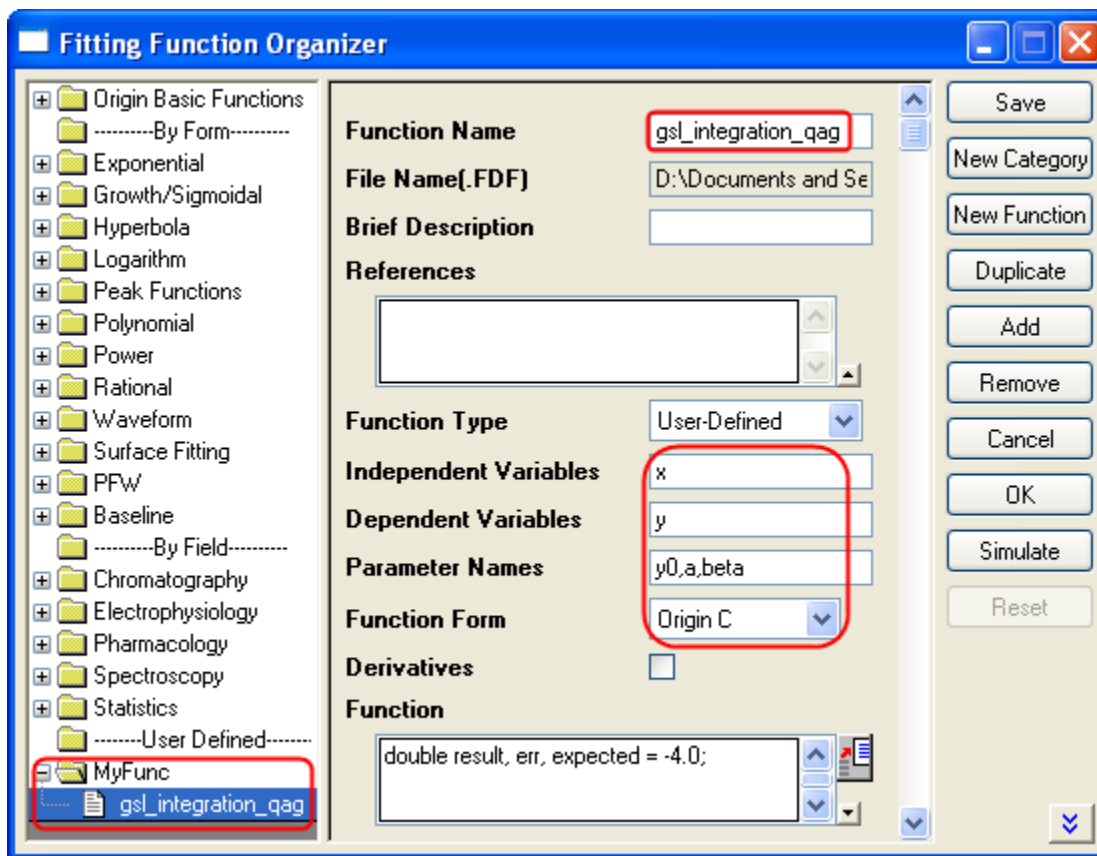
GSL_EXPORT int gsl_integration_qag (const gsl_function * f,
double a, double b,
double epsabs, double
epsrel, size_t limit,
int key,
gsl_integration_workspace
* workspace,
```

```

double *result, double
*abserr);

```

3. Press F9 to open the **Fitting Function Organizer** and then add a new function as follows:



4. Press the button on the right hand side of the **Function** Field to open the code builder and add the following codes and compile: **_nlfgsl_integration_qag.fit**

```

#include <..\ocgsl.h>

static double f_callback(double x, void * params)
{
    double alpha = *(double *)params;
    return exp(alpha*x);
}

void _nlsgsl_integration_qag(
// Fit Parameter(s):
double y0, double a, double beta,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)
{
    // Beginning of editable part
    double result, err, expected = -4.0;

    // Allocates a workspace sufficient to hold 1000 double
    precision intervals,

```

```

    // their integration results and error estimates
    gsl_integration_workspace *ww =
    gsl_integration_workspace_alloc(1000);

    gsl_function F;
    F.function = f_callback;
    F.params = &beta ;

    // integral interval (0, x), within the desired absolute
    // error 0 and relative error 1e-7
    gsl_integration_qag(&F, 0, x, 0, 1e-7, 1000, 0, ww, &result,
    &err);

    // frees the memory associated with the workspace w
    gsl_integration_workspace_free (ww);

    y = y0 + a*result;

    // End of editable part
}

```

Furthermore, a more elaborate but efficient version of the fitting function is given as follows

```

//-----
//
#include <ONLSF.h>
#include <..\ocgsl.h>

static double f_callback(double x, void * params)
{
    double alpha = *(double *)params;
    return exp(alpha*x);
}

void _nlsfgsl_integration_qag(
// Fit Parameter(s):
double y0, double a, double beta,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)
{
    // Beginning of editable part

    NLFitContext *pCtxt = Project.GetNLFitContext();
    if ( pCtxt )
    {
        static vector vInteg;
        NLSFCURRINFO stCurrInfo;
        pCtxt->GetFitCurrInfo(&stCurrInfo);
        int nCurrentIndex = stCurrInfo.nCurrDataIndex;

        BOOL bIsNewParamValues = pCtxt->IsNewParamValues();
        if ( bIsNewParamValues )
        {
            vector vx;
            pCtxt->GetIndepData(&vx);

```

```

        int nSize = vx.GetSize();
        vInteg.SetSize(nSize);

        // Allocates a workspace sufficient to hold
1000 double precision intervals,
        // their integration results and error
estimates
        gsl_integration_workspace *ww =
gsl_integration_workspace_alloc(1000);

        gsl_function F;
        F.function = f_callback;
        F.params = &beta ;

        double result, err, expected = -4.0;
        for(int ii=0; ii<nSize; ++ii)
        {
            // integral interval (0, vx[ii]),
within the desired absolute
            // error 0 and relative error 1e-7
            gsl_integration_qag(&F, 0, vx[ii], 0,
1e-7, 1000, 0, ww, &result, &err);
            vInteg[ii] = result;
        }

        // frees the memory associated with the
workspace w
        gsl_integration_workspace_free (ww);

    }

    y = y0 + a*vInteg[nCurrentIndex];
    x;
}

// End of editable part
}

```

5. Add the following initialization codes:

Parameter Init

```

//Code to be executed to initialize parameters
sort( x_y_curve );
double coeff[2];
fitpoly( x_y_curve, 1, coeff);
a = coeff[0];
y0 = coeff[1];
beta=1.0

```

6. Fit using the user-defined function `gsl_integration_qag`, here are the results:

y0 = -1.06363E-6

a = 1

beta =1

Fitting with NAG Special Function

Summary

Origin allows user to define an Origin C fitting function using NAG special functions. You can call NAG routine to evaluate the special function.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create fitting function using Fitting Function Organizer
- Create fitting function using NAG special function

Example and Steps

We will fit the following model:

Here A , td and $t0$ are the model parameters we want to obtain from the data fitting. $I0$ and $I1$ are the first kind of Modified Bessel function of order 0 and order 1, respectively. For current example, we use the sample data in the end of this tutorial. The fitting procedure can be outlined into the following steps:

Press **F9** to open the **Fitting Function Organizer** and then create a new Category named **FittingWithNAGSpecialFunc**. Define a new fitting function **FittingWithBessel** in the new category as follow:

```

Function Name:      FittingWithBessel
Function Type:     User-Defined
Independent Variables: t
Dependent Variables: inorm
Parameter Names:  A,t0,td
Function Form:     Origin C
Function:

```

Click the button (icon) beside the **Function** box to open the code builder and define and compile and save the fitting function as follows:

```

#include <origin.h>

// Add your special include files here.
// For example, if you want to fit with functions from the NAG
library,
// add the header file for the NAG functions here.
#include <OC_nag8.h>

// Add code here for other Origin C functions that you want to
define in this file,
// and access in your fitting function.

// You can access C functions defined in other files, if those
files are loaded and compiled

```

```

// in your workspace, and the functions have been prototyped
// in a header file that you have
// included above.

// You can access NLSF object methods and properties directly
// in your function code.

// You should follow C-language syntax in defining your
// function.
// For instance, if your parameter name is P1, you cannot use
// p1 in your function code.
// When using fractions, remember that integer division such
// as 1/2 is equal to 0, and not 0.5
// Use 0.5 or 1/2.0 to get the correct value.

// For more information and examples, please refer to the
// "User-Defined Fitting Function"
// section of the Origin Help file.

//-----
//
void _nlsfFittingWithBessel(
// Fit Parameter(s):
double A, double t0, double td,
// Independent Variable(s):
double t,
// Dependent Variable(s):
double& inorm)
{
    // Beginning of editable part
    //inorm= A* exp(-td/2.0/(t-t0)) * ( s18aec(td/2.0/(t-
t0),NAGERR_DEFAULT)+s18afc(td/2.0/(t-t0),NAGERR_DEFAULT)
);

    static NagError fail1;
    static NagError fail2;
    double dtemp = td/2.0/(t-t0);
    inorm= A* exp(-dtemp) * (
s18aec(dtemp,&fail1)+s18afc(dtemp,&fail2) );
    if(fail1.code !=NE_NOERROR)
        printf( "%s\n",fail1.message);
    if(fail2.code !=NE_NOERROR)
        printf( "%s\n",fail2.message);

    // End of editable part
}

```

Simulate the Function

After the function body is defined, you can click the **Compile** button in **Code Builder** to check syntax errors. And then click **Return to Dialog** button to go back **Fitting Function Organizer** dialog box. Now click the **Save** button to generate the .FDF file (Function definition file).

Once you have a .FDF file, you can click the **Simulate** button to simulate a curve, this will be very helpful to evaluate the initial values. In the **simcurve** dialog, enter some proper parameter values and X range, and see what the curve looks like in the **Preview** panel.

Set the Initial Values for the Parameters

As it is a user-defined fitting function, you have to supply the initial guess values for the parameters before performing your fitting task for the data. You may do it by set them manually in the **Parameter** tab in **Nonlinear Curve Fit** dialog. For the sample data shown below, you can just set the initial values for the parameters $A = 1$, $t_d = 1$, $t_0 = 1$. After the parameters are initialized, you can then do the fitting to obtain the fitting result, as shown to the right of the sample data.

Sample Data

Copy the below sample data and use **Import Wizard** to import the data from **Clipboard**, then do the fitting using the given initial values for the parameters: $A = 1$, $t_d = 1$, $t_0 = 1$.

Sample Data		Results															
X	Y																
2	0.7868954118	Parameters <table border="1"> <thead> <tr> <th></th> <th></th> <th>Value</th> <th>Standard Error</th> </tr> </thead> <tbody> <tr> <td rowspan="3">"Y"</td> <td>A</td> <td>0.96431</td> <td>0.06562</td> </tr> <tr> <td>t₀</td> <td>1.39545</td> <td>0.40134</td> </tr> <tr> <td>t_d</td> <td>0.53711</td> <td>0.54076</td> </tr> </tbody> </table> <p>Reduced Chi-sqr = 1.02442755048E-4 COD(R²) = 0.92247024814828 Iterations Performed = 11 Total Iterations in Session = 11 Fit converged - tolerance criterion satisfied.</p>			Value	Standard Error	"Y"	A	0.96431	0.06562	t ₀	1.39545	0.40134	t _d	0.53711	0.54076	
			Value	Standard Error													
"Y"	A		0.96431	0.06562													
	t ₀		1.39545	0.40134													
	t _d		0.53711	0.54076													
2.080808081	0.8133022141																
2.161616162	0.8178216765																
2.242424242	0.8427866729																
2.323232323	0.8315815363																
2.404040404	0.8484657180																
2.565656566	0.8618233553																
2.646464646	0.8745962570																
2.727272727	0.8921620316																
2.808080808	0.8687399759																

Fitting Integral Function with parametric limit using NAG Library

Summary

Before you start delving into this tutorial, you are recommended to read the relevant tutorial in Fitting with Integral using NAG Library. And as far as programming is concerned, the two tutorials are basically the same, except that here you will learn to define Origin C fitting function with fitting parameters in the integral limit, while in the previous tutorial we in fact define a fitting independent variable in the integral limit. Also note that **a different NAG integrator** is used here.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a fitting function with Definite Integral using the NAG integration routine
- Create a fitting function with a parametric integral limit
- Use a log function to scale a large return value from the fitting function

Example and Steps

For example, we will fit the sample data at the bottom of this page with the following model:

$$y = \int_c^d \frac{\cosh((x_i + b^2 \cdot x^2)/(b + x))}{a + (x_i^2 + x^2)} dx_i$$

Note that we use x_i to indicate the integral independent variable while x indicates the fitting independent variable. The model parameters a , b , c , and d are fitted parameters we want to obtain from the sample data. To prepare the data, you just need to copy the sample data to an Origin Work sheet. The fitting procedure is similar to the previous tutorial:

Define Fitting Function in Fitting Function Organizer

Press **F9** to open the Fitting Function Organizer and add the User-Defined integral fitting function *nag_integration_fitting_cosh* to the Category *FittingWithIntegral*, similar to the first tutorial.

Function Name:	nag_integration_fitting_cosh
Function Type:	User-Defined
Independent Variables:	x
Dependent Variables:	y
Parameter Names:	a, b, c, d
Function Form:	Origin C
Function:	

Click the button (icon) beside the **Function** box to open the code builder and define and compile the fitting function as follows: (**Note**: Remember to save the Function after compiling it and returning to the Function Organizer Dialog):

```
#include <origin.h>

// Add your special include files here.
// For example, if you want to fit with functions from the NAG
library,
```

```

// add the header file for the NAG functions here.
#include <oc_nag8.h>

// Add code here for other Origin C functions that you want to
define in this file,
// and access in your fitting function.
struct user
{
    double a, b, fitX; // fitX the independent variable of
fitting function
};
static double NAG_CALL f_callback(double x, Nag_User *comm)
// x is the independent variable of the integrand
{
    struct user *sp = (struct user *) (comm->p);

    double aa, bb, fitX; // temp variable to accept the
parameters in the Nag_User communication struct
    aa = sp->a;
    bb = sp->b;
    fitX = sp->fitX;

    return
cosh((x*x+bb*bb*fitX*fitX)/(bb+fitX))/(aa+(x*x+fitX*fitX));
}

// You can access C functions defined in other files, if those
files are loaded and compiled
// in your workspace, and the functions have been prototyped
in a header file that you have
// included above.

// You can access NLSF object methods and properties directly
in your function code.

// You should follow C-language syntax in defining your
function.
// For instance, if your parameter name is P1, you cannot use
p1 in your function code.
// When using fractions, remember that integer division such
as 1/2 is equal to 0, and not 0.5
// Use 0.5 or 1/2.0 to get the correct value.

// For more information and examples, please refer to the
"User-Defined Fitting Function"
// section of the Origin Help file.

//-----
//
void _nlsfnag_integration_fitting_cosh(
// Fit Parameter(s):
double a, double b, double c, double d,
// Independent Variable(s):
double x,

```

```

// Dependent Variable(s):
double& y)
{
    // Beginning of editable part
    double epsabs = 0.00001, epsrel = 0.0000001, result, abserr;
    Integer max_num_subint = 500;
        // you may use epsabs and epsrel and this quantity to
enhance your desired precision
        // when not enough precision encountered

    Nag_QuadProgress qp;
    static NagError fail;

    // the parameters parameterize the integrand can be input to
the call_back function
// through the Nag_User communication struct
    Nag_User comm;
    struct user s;
    s.a = a;
    s.b = b;
    s.fitX = x;
    comm.p = (Pointer)&s;

    d01sjc(f_callback, c, d, epsabs, epsrel, max_num_subint,
&result, &abserr, &qp, &comm, &fail);

        // you may want to exam the error by printing out
error message, just uncomment the following lines
    // if (fail.code != NE_NOERROR)
        // printf("%s\n", fail.message);

    // For the error other than the following three errors which
are due to bad input parameters
    // or allocation failure NE_INT_ARG_LT NE_BAD_PARAM
NE_ALLOC_FAIL
    // You will need to free the memory allocation before calling
the integration routine again to
        // avoid memory leakage
    if (fail.code != NE_INT_ARG_LT && fail.code != NE_BAD_PARAM
&& fail.code != NE_ALLOC_FAIL)
    {
        NAG_FREE(qp.sub_int_beg_pts);
        NAG_FREE(qp.sub_int_end_pts);
        NAG_FREE(qp.sub_int_result);
        NAG_FREE(qp.sub_int_error);
    }

    y = log(result);
        // note use log of the integral result as return as
the integral result is large,
        // you are not necessary to do so

    // End of editable part
}

```

In the above code, we define the integrand as a callback function *f_callback* just outside the fitting function body *_nlsfnag_integration_fitting_cosh*. Note that we parametrize the integrand function with the variables *a*, *b* and *fitX*, and pass them into the callback function through the *Nag_User* struct. After that we perform the integration using NAG integrator *d01sjc*. Besides, you can also use other Quadrature Routines as you want. In the current example, we also use a log scale for the fitting function. (The sample data are already scaled by a log function)

Compile the code, return to the dialog and then Save the fitting function in the function Organizer and open the **Nonlinear Curve Fit** dialog in the **Analysis-Fitting** menu. You can then select this user-defined fitting function in the **Function Selection** page under **Setting** Tab.

Set the Initial Values for the Parameters

Similarly, as it is a user-defined fitting function, you have to supply the initial guess values for the parameters. You may manually set them in the **Parameter** tab in **Nonlinear Curve Fit** dialog. For current example, you can just set the initial values for the parameters $a = 1$, $b = 10$, $c = 3$, $d = 4$. After the parameters are initialized, you can perform the fitting to obtain the fitting result, as shown in the following.

Sample Data

X	Y
-5	498.19046
-4.33333	329.43196
-3.66667	210.28005
-3	126.55799
-2.33333	69.01544
-1.66667	31.3555
-1	9.1393
-0.33333	-0.84496
0.33333	-0.99914
1	6.86736

Results:		
	Value	Standard Error
a	0.99303	0.06577
b	10	5.3108E-5
c	3.00083	0.0062
d	4.00022	9.38713E-4

Fitting with Integral using NAG Library

Summary

Origin allows user to define an Origin C fitting function which involves an integral. You can call NAG functions to perform the integration while defining the fitting function. There are built-in functions in Origin C which perform integration. For the current example, the NAG solution is recommended. It has a better performance compared to the built-in integration algorithm. Note that an **infinite NAG integrator** is used here.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a fitting function using the Fitting Function Organizer
- Create a fitting function with a Definite Integral using a NAG integration routine
- Set up the Initial Code for the fitting function

Example and Steps

We will fit the following model:

Here y_0 , A , x_c and w are the model parameters we want to obtain from the data fitting. The fitting procedure can be outlined into the following steps:

Define the Function

Press **F9** to open the **Fitting Function Organizer** and then create a new Category named **FittingWithIntegral**. Define a new fitting function **nag_integration_fitting** in the new category as follow:

Function Name: nag_integration_fitting
Function Type: User-Defined
Independent Variables: x
Dependent Variables: y
Parameter Names: y_0 , A, x_c , w
Function Form: [Origin C](#)
Function:

Click the button (icon) beside the **Function** box to open the code builder and define and compile and save the fitting function as follows:

```
#include <origin.h>
// Add your special include files here.
// For example, if you want to fit with functions from the NAG
library,
// add the header file for the NAG functions here.
#include <oc_nag8.h>
```

```

// Add code here for other Origin C functions that you want to
define in this file,
// and access in your fitting function.

struct user // parameters in the integrand
{
    double amp, center, width;
};
// Function supplied by user, return the value of the
integrant at a given x.
static double NAG_CALL f_callback(double x, Nag_User *comm)
{
    struct user *sp = (struct user *)(comm->p);

    double amp, center, width; // temp variable to
accept the parameters in the Nag_User communication struct
    amp = sp->amp;
    center = sp->center;
    width = sp->width;

    return amp * exp( -2*(x - center)*(x - center)/width/width )
/ (width*sqrt(PI/2));
}

// You can access C functions defined in other files, if those
files are loaded and compiled
// in your workspace, and the functions have been prototyped
in a header file that you have
// included above.

// You can access NLSF object methods and properties directly
in your function code.

// You should follow C-language syntax in defining your
function.
// For instance, if your parameter name is P1, you cannot use
p1 in your function code.
// When using fractions, remember that integer division such
as 1/2 is equal to 0, and not 0.5
// Use 0.5 or 1/2.0 to get the correct value.

// For more information and examples, please refer to the
"User-Defined Fitting Function"
// section of the Origin Help file.

//-----
//
void _nlsfnag_integration_fitting(
// Fit Parameter(s):
double y0, double A, double xc, double w,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)

```

```

{
    // Beginning of editable part

    // Through the absolute accuracy epsabs, relative accuracy
    epsrel and max_num_subint you can
    // control the precision of the integration you need
    // if epsrel is set negative, the absolute accuracy will be
    used.
    // Similarly, you can control only relative accuracy by set
    the epsabs negative
    double epsabs = 0.0, epsrel = 0.0001;

    // The max number of sub-intervals needed to evaluate the
    function in the integral
    // The more difficult the integrand the larger max_num_subint
    should be
    // For most problems 200 to 500 is adequate and recommended
    Integer max_num_subint = 200;

    // Result keeps the approximate integral value returned by
    the algorithm
    // abserr is an estimate of the error which should be an
    upper bound for the |I - result|
    // where I is the integral value
    double result, abserr;

    // The structure of type Nag_QuadProgress,
    // it contains pointers allocated memory internally with
    max_num_subint elements
    Nag_QuadProgress qp;

    // The NAG error parameter (structure)
    static NagError fail;

    // Parameters passed to integrand by Nag_User communication
    struct
        Nag_User comm;
    struct user s;
        s.amp = A;
        s.center = xc;
        s.width = w;
        comm.p = (Pointer)&s;

    // Perform integration
    // There are 3 kinds of infinite boundary types you can use
    in Nag infinite integrator
    // Nag_LowerSemiInfinite, Nag_UpperSemiInfinite, Nag_Infinite
    d01smc(f_callback, Nag_LowerSemiInfinite, x, epsabs, epsrel,
    max_num_subint, &result, &abserr, &qp, &comm, &fail);

        // you may want to exam the error by printing out
    error message, just uncomment the following lines
    // if (fail.code != NE_NOERROR)
        // printf("%s\n", fail.message);

    // For the error other than the following three errors which
    are due to bad input parameters

```



```

// or allocation failure NE_INT_ARG_LT NE_BAD_PARAM
NE_ALLOC_FAIL
// You will need to free the memory allocation before calling
the integration routine again to avoid memory leakage
if (fail.code != NE_INT_ARG_LT && fail.code != NE_BAD_PARAM
&& fail.code != NE_ALLOC_FAIL)
{
    NAG_FREE(qp.sub_int_beg_pts);
    NAG_FREE(qp.sub_int_end_pts);
    NAG_FREE(qp.sub_int_result);
    NAG_FREE(qp.sub_int_error);
}

// Calculate the fitted value
y = y0 + result;

// End of editable part
}

```

In the above code, we firstly define the integrand as a callback function *f_callback* just outside the fitting function body *_nlsfnag_integration_fitting*. Note that we parametrize the integrand function with the variables *amp*, *center* and *width*, and pass them into the callback function through the *Nag_User* struct. Inside the fitting function, we perform the integration using NAG integrator *d01smc*.

Calling NAG functions should be more efficient than writing your own routines. Using an analogous method, you can perform finite, infinite, one-dimension and multi-dimension quadrature in your fitting function. Please read the NAG Quadrature page and select a proper routine.

Simulate the Function

After entering the function body codes, you can click the **Compile** button in **Code Builder** to check syntax errors. And then click **Return to Dialog** button to go back **Fitting Function Organizer** dialog box. Now click the **Save** button to generate the .FDF file (Function definition file).

Once you have a .FDF file, you can click the **Simulate** button to simulate a curve, this will be very helpful to evaluate the initial values. In the **simcurve** dialog, enter some proper parameter values and X range, and see what the curve looks like in the **Preview** panel.

Fit the Curve

Before you start to fit the curve, it is very helpful to simulate the function first. Performing integration may take some time, if there is any mistake, you may see Origin "freeze" after you click the **Fit** button. So in the **Fitting Function Organizer** dialog, select the function we defined and click the **Simulate** button. This will bring up the **simcurve** X-Function. Enter some "guess" values and click the **Apply** button. If the simulated curve looks like your source data, you can go further to fit.

To test the fitting function, import *\Samples\Curve Fitting\Replicate Response Data.dat* to Origin. Set $Col(A) = \log(Col(A))$ in the **Set Column Values** dialog. This will make a sigmoid curve. Highlight column A and B and create a scatter plot. Then bring up the **NLFit** dialog from **Analysis: Fitting: Nonlinear Curve Fit** menu item. Select the fitting function we just defined and go to the **Parameters** tab, initialize all parameters by 1 and fit. You are supposed to see these results:

	Value	Standard Error
y0	-0.00806	0.18319
A	3.16479	0.39624
xc	-0.19393	0.10108
w	1.77252	0.33878

Fitting with Summation

Summary

We have showed you how to perform fitting with an integral using the NAG Library, and now you'll learn how to do that without calling NAG functions. In this tutorial, we will show you how to do integration by the trapezoidal rule and include the summation procedure in the fitting function.

Minimum Origin Version Required: Origin 8.0 SR6

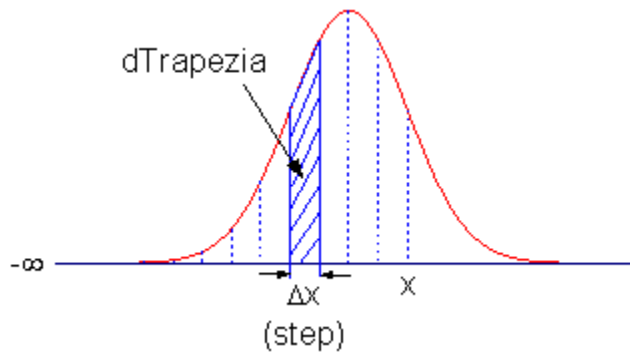
What you will learn

- How to include summation in your fitting function.
- Trapezoidal rule for integration.

Example and Steps

We will fit the same model as the integral fit using NAG:

The difference is that we will perform the integration within the fitting function. Using the trapezoidal rule, we will first divide the curve into pieces and then approximate the integral area by multiple trapezoids. The precision of the result then depends on how many trapezoids will be used. Since this is a semi-infinite integration, we will set an increment (steps) and construct trapezoids from the upper integral limit, x , to the lower integral limit, negative infinity and then accumulate the area of these trapezoids. When the increment of the area is significantly small, we will stop the summation. Before doing the summation, you should guarantee that the function is **CONVERGENT**, or you should include a convergence check in your code.



Define the Function

Select **Tools:Fitting Function Organizer** or alternatively press the **F9** key to open the **Fitting Function Organizer** and then define the function as follows:

Function Name:	summation
Function Type:	User-Defined
Independent Variables:	x
Dependent Variables:	y
Parameter Names:	y0, A, xc, w
Function Form:	Origin C

Function:

Click the button (icon) beside the **Function** box to open Code Builder. Define, compile and save the fitting function as follows:

```
#pragma warning(error : 15618)
#include <origin.h>

// Subroutine for integrand
double f(double x, double A, double xc, double w)
{
    return A * exp(-2*(x-xc)*(x-xc)/w/w) / w / sqrt(PI/2);
}

//-----
//
void _nlsfsummation(
// Fit Parameter(s):
double y0, double A, double xc, double w,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)
{
    // Beginning of editable part
    // Set the tolerance for stop integration.
    double dPrecision = 1e-12;
    // Initialization
    double dIntegral = 0.0;
    double dTrapezia = 0.0;
    // Steps, or Precision.
    double dStep = 0.01;
    // Perform integrate by trapezoidal rule.
    // Note that you should guarantee that the function is
    CONVERGENT.
    do
    {
        // Trapezia area.
        dTrapezia = 0.5 * ( f(x, A, xc, w) + f((x-dStep), A,
xc, w) ) * dStep;
        // Accumulate area.
        dIntegral += dTrapezia;
        x -= dStep;
    }while( (dTrapezia/dIntegral) > dPrecision );
    // Set y value.
    y = y0 + dIntegral;
    // End of editable part
}
}
```

Fit the Curve

We can use the same data to test the result.

1. Import `\Samples\Curve Fitting\Replicate Response Data.dat`.
2. Highlight the first column, right-click on it, and select **Set Column Values** from the context menu.
3. Set $Col(A) = \log(Col(A))$ in the **Set Column Values** dialog. This will make a sigmoidal curve.
4. Highlight columns A and B and create a scatter plot.

5. Then bring up the **NLFit** dialog by pressing **Ctrl + Y**. Select the fitting function we just defined and go to the **Parameters** tab, initialize all parameters to 1 and fit. You should see these results:

	Value	Standard Error
y0	-0.00806	0.18319
A	3.16479	0.39624
xc	-0.19393	0.10108
w	1.7725	0.33878

Fitting Complex Function

Summary

When fitting with a complex function, we can easily separate the complex function to two functions: one corresponding to its real part and the other corresponding to its imaginary part. With these two functions, we can define the complex fitting function with two dependent variables by **Fitting Function Organizer** and can access it in **NLFit** dialog. We will illustrate how to fit with complex function below. More details about fitting with multiple dependent or independent variable please refer to Fitting with Multiple Independent Variables.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a user-defined complex fitting function with two dependent variables and one independent variable
- Fit with such function in NLFit

Steps

1. Select whole form below (including header line) and right click to choose **Copy** to put the data in clipboard.

Omega	Y1	Y2
0	3	0
0.01	2.88462	-0.28846
0.02	2.58621	-0.51724
0.03	2.20588	-0.66176
0.04	1.82927	-0.73171
0.05	1.5	-0.75
0.06	1.22951	-0.7377
0.07	1.01351	-0.70946
0.08	0.8427	-0.67416
0.09	0.70755	-0.63679
0.1	0.6	-0.6
0.11	0.5137	-0.56507

2. Select **Import/ Import Wizard** to open Import Wizard dialog. Then choose **Clipboard** in **Data Source** group and click **Finish** to import the data.

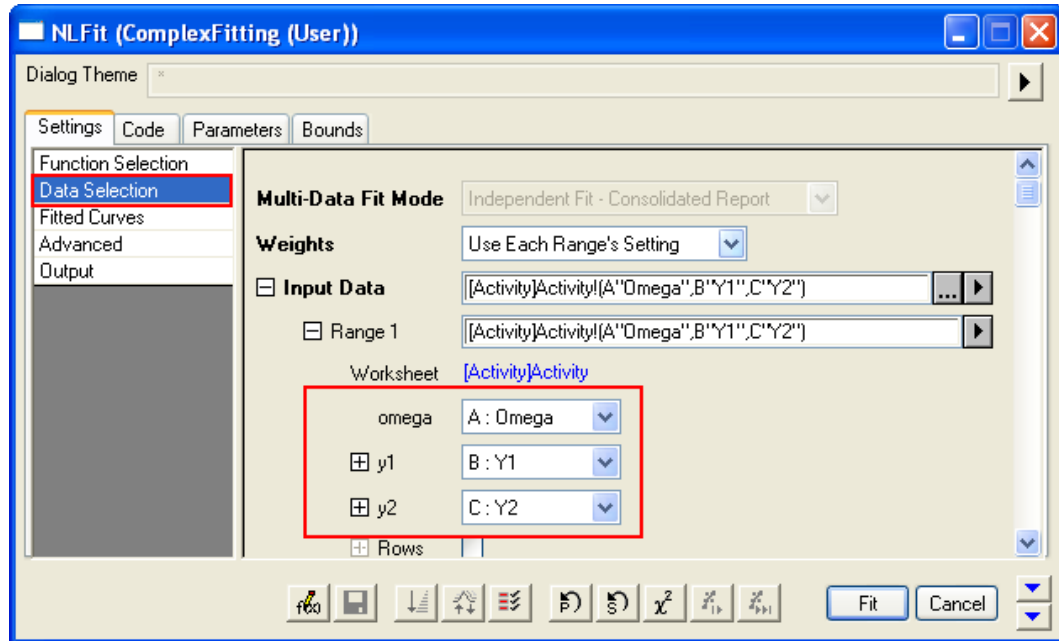


3. Select **Tools: Fitting Function Organizer** from menu (or press **F9**) to bring up the **Fitting Function Organizer** and define a new fitting function named **ComplexFitting** in **NewCategory** (create the category if not exist) as follow:

Function Name:	ComplexFitting
Function Type:	User-Defined
Independent Variables:	omega
Dependent Variables:	y1,y2
Parameter Names:	A,tau
Function Form:	Origin C
Function:	<pre>complex cc = A/(1+1i*omega*tau); y1 = cc.m_re; y2 = cc.m_im;</pre>

4. Note: To use the imaginary unit "i" for creating complex numbers, you need to write it as "1i" in Origin C, as in the above **Function** row. And **complex** is a class that implements a complex number data type. It contains both a Real and an Imaginary component.
5. For more details about creating user-defined fitting function, please refer to User Defined Fitting Function using Origin C.

6. Highlight all the columns and select **Analysis: Fitting: Non-linear Curve Fit** from menu to bring up the NLFit dialog. Select the function **ComplexFitting** from **NewCategory** on the **Settings: Function Selection** page. Set the input datasets in the **Data Selection** page as follow:



7. Select **Parameters Tab** and set the initial values as follows:

Param	Meaning	Fixed	Value	Error
A	?	<input type="checkbox"/>	1	--
tau	?	<input type="checkbox"/>	1	--

8. Click **Fit** to generate the fitting report sheet. You can see the results from the report worksheet as below:

Parameters

		Value	Standard Error
Y1,Y2	A	2.36712	0.15413
	tau	15.84746	1.94844

Statistics

	Y1,Y2
Number of Points	24
Degrees of Freedom	22
Reduced Chi-Sqr	0.12339
Residual Sum of Squares	2.71451
Adj. R-Square	0.92387
Fit Status	Succeeded(100)

From the Statistics table, we can see that the fitting is fairly successful.

Fitting with Convolution

Summary

When performing curve fitting to experimental data, one may encounter the need to account for instrument response in the data. One way to do this is to first perform deconvolution on the data to remove the instrument response, and then perform curve fitting as a second step. However deconvolution is not always reliable as the results can be very sensitive to any noise present in the data. A more reliable way is to perform convolution of the fitting function with the instrument response while performing the fitting. This tutorial will demonstrate how to perform convolution while fitting.

Minimum Origin Version Required: Origin 8 SR6.

What you will learn

This tutorial will show you how to:

- access fitting information during iterations.
- perform convolution while fitting.

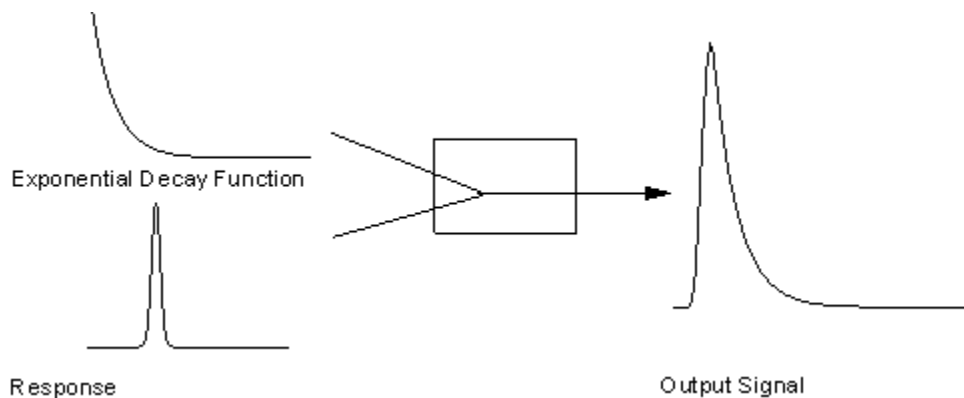
Example and Steps

Background

Let's start this example by importing `\Samples\Curve Fitting\FitConv.dat`.

A(X)	B(Y)	C(Y)
Sampling	Signal	Impulse
0	-0.19775	0
0.1	-0.32893	0
0.2	0.10055	0
0.3	0.09394	0
0.4	-0.1292	0
0.5	0.06346	1.48672E-6
0.6	0.19453	1.3383E-4

The source data includes sampling points, output signal and the impulse response. This experiment assumes that the output signal was the convolution of an exponential decay function with a Gaussian response:



Now that we already have the output signal and response data, we can get the exponential decay function by fitting the signal to the below model:

Define the Function

Obviously, column 1 and column 2 are x and y respectively in the function. How about column 3, the impulse response? We will access this column within the fitting function, and compute the theoretical exponential curve from the sampling points. Then we can use fast Fourier transform to perform the convolution.

Press **F9** to open the **Fitting Function Organizer** and define a function like:

Function Name: FitConv
Function Type: User-Defined
Independent Variables: x
Dependent Variables: y
Parameter Names: y0, A, t
Function Form: [Origin C](#)
Function:

Click the button (icon) beside the **Function** box and write the function body in **Code Builder**:

```
#pragma warning(error : 15618)
#include <origin.h>
// Header files need to be included
#include <ONLSF.H>
#include <fft_utils.h>
//
//
void _nlsfTestConv(
// Fit Parameter(s):
double y0, double A, double t,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)
{
// Beginning of editable part
Worksheet wks = Project.ActiveLayer();
NLFitContext *pCtxt = Project.GetNLFitContext();
if ( pCtxt )
{
// Vector for the output signal in each iteration.
static vector vSignal;
// If parameters were updated, we will recalculate
the convolution result.
BOOL bIsNewParamValues = pCtxt->IsNewParamValues();
if ( bIsNewParamValues )
{
// Read sampling and response data from
worksheet.

Dataset dsSampling(wks, 0);
Dataset dsResponse(wks, 2);
int iSize = dsSampling.GetSize();

vector vResponse, vSample;

vResponse = dsResponse;
```

```

vSample = dsSampling;

vSignal.SetSize(iSize);
vResponse.SetSize(iSize);
vSample.SetSize(iSize);

// Compute the exponential decay curve
vSignal = A * exp( -t*vSample );
// Perform convolution
int iRet = fft_fft_convolution(iSize, vSignal,
vResponse);
}

NLSFCURRINFO    stCurrInfo;
pCtxt->GetFitCurrInfo(&stCurrInfo);
// Get the data index for the iteration
int nCurrentIndex = stCurrInfo.nCurrDataIndex;
// Get the evaluated y value
y = vSignal[nCurrentIndex] + y0;
// For compile the function, since we haven't use x
here.
x;
}
// End of editable part
}

```

Traditionally, for a particular x , the function will return the corresponding y value. However, when convolution is involved, we need to perform the operation on the entire curve, not only on a particular data point. So from Origin 8 SR2, we introduced the `NLFitContext` class to achieve some key information within the fitter. In each iteration, we use `NLFitContext` to monitor the fitted parameters; once they are updated, we will compute the convolution using the fast Fourier transform by the `fft_fft_convolution` method. The results are saved in the `vSignal` vector. Then for each x , we can get the evaluated y from `vSignal` with the current data index in `NLSFCURRINFO`.

Fit the Curve

In the fitting function body, we read the response data directly from the active worksheet. So you should perform the fit from the worksheet. Highlight column B and press **Ctrl + Y** to bring up the **Nonlinear Fitting** dialog. Select the *FitConv* function, choose *X Data Type* from *Fitted Curves* page as **Same as Input Data**, and then initialize the parameters as $y_0=0$, $A=10$, $t=1$. Click the **Fit** button to generate the results.

Quoting Built-in Functions in Your New Function

Summary

This tutorial will show you how to reference a built-in function when creating a user-defined fitting function.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

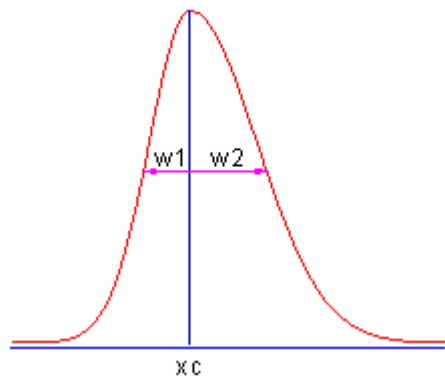
- Define a piecewise fitting function
- Access built-in functions in you new function
- Auto initialize the parameters

Steps

Data

Start by importing the file `\Samples\Curve Fitting\Asymmetric Gaussian.dat` into a new workbook.

Highlight column B and create a graph. The peak in the data is slightly skewed to the right. How to fit such a curve? One idea is to divide the curve into two parts - We can consider this curve to be composed of two Gaussian function as below. These two Gaussian curves share the same baseline and peak center, but differ in peak width and amplitude.



Define the Function

Press **F9** to open the **Fitting Function Organizer** and define the function as below:

Function Name: AsymmetricGauss
Function Type: User-Defined
Independent Variables: x
Dependent Variables: y
Parameter Names: y0, xc, w1, w2, A1, A2
Function Form: [Origin C](#)
Function: `y = x<xc? nlf_Gauss(x, y0, xc, w1, A1) : nlf_Gauss(x, y0, xc, w2, A2);`

Note:

For versions before **Origin 8.1**, the function body should be defined as:

```
y = x<xc? nlfxGauss(x, y0, xc, w1, A1) : nlfxGauss(x, y0,
xc, w2, A2);
x; y0; xc; w1; w2; A1; A2;
```

Listing the parameters at the end is used to avoid the "parameter not used inside the function body" error, although you already use these parameters. This is required to compile the function successfully.

When calling **nlf_FuncName** to reuse built-in functions, the syntax is:

```
nlf_FuncName( independent variable, parameter list ... )
```

where **FuncName** is the fitting function name. Besides, the old notation, **nlfxFuncName** also supported.

The *Parameter List* follows the parameter order in function definition file for the built-in function (the FDF file. You can open the FDF file in Notepad. The files are located in the \\Origin EXE Folder\FitFunc\). Note that, the function name we use is the DLL interface name. The actual name in the [General Information] section of the FDF file. Look at the *Function Source* item and the value is **fgroup.FuncName**, and use the *FuncName*. In most cases, this function name is consistent with the function name visible in the NLFit dialog. For a few functions such as Voigt, these names are different.

For parameter initialization of this skewed gaussian function, we can simply copy the initialization code of the built-in gauss function, and make a few minor modifications:

```
xc = peak_pos(x_y_curve, &w1, &y0, &A1);
w2 = w1;
A2 = A1;
```

The final function body should be as below:

Function Name AsymmetricGauss

File Name(.FDF) C:\Documents and Settings\Administrator\My C

Brief Description

References

Function Type User-Defined

Independent Variables x

Dependent Variables y

Parameter Names y0,xc,w1,w2,A1,A2

Function Form Origin C

Derivatives

Function

```
y = x<xc? nlf_Gauss(x, y0, xc, w1, A1) : nlf_Gauss(x, y0, xc, w2, A2);
```

Peak Function

Parameter Settings

```
InitialValues = --(V)
Meanings = ?
```

Enable Auto Initialization

Use OriginC

Parameter Initialization

```
xc = peak_pos(x_y_curve, &w1, &y0, &A1);
w2 = w1; A2 = A1;
```

Once compiled successfully, save the function and fit the curve. The results should be as below:

		Value	Standard Error
Amplitude 1	y0	1.8	4.79E-5
	xc	4.5	3.45E-5
	w1	1.8	4.5E-5
	w2	3	4.88E-5
	A1	30	0
	A2	50	0

Fit Function with Non-constant Background

Summary

Many of the Origin built-in functions are defined as:

Where y_0 can be treated as the "constant background". How about fitting a curve with a non-constant background? One option is to use the Peak Analyzer we provide. The Peak Analyzer includes multiple methods to subtract the baseline, including exponential or polynomial backgrounds. In this tutorial, we will show you how to fit such curves without using the Peak Analyzer.

Minimum Origin Version Required: Origin 8.0 SR6

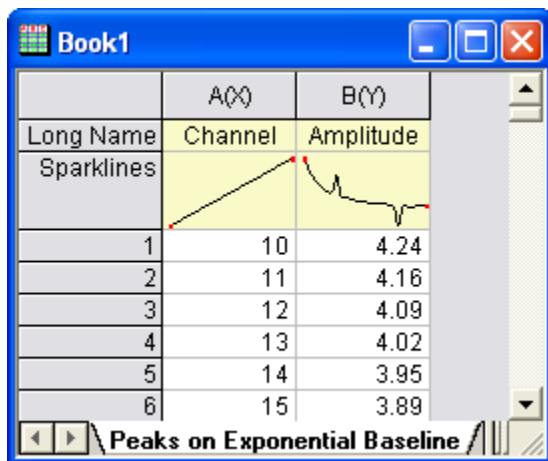
What you will learn

- Review Worksheet Query.
- Quote a built-in function by `nlfxFuncName` method.
- Auto Initialize the parameters.

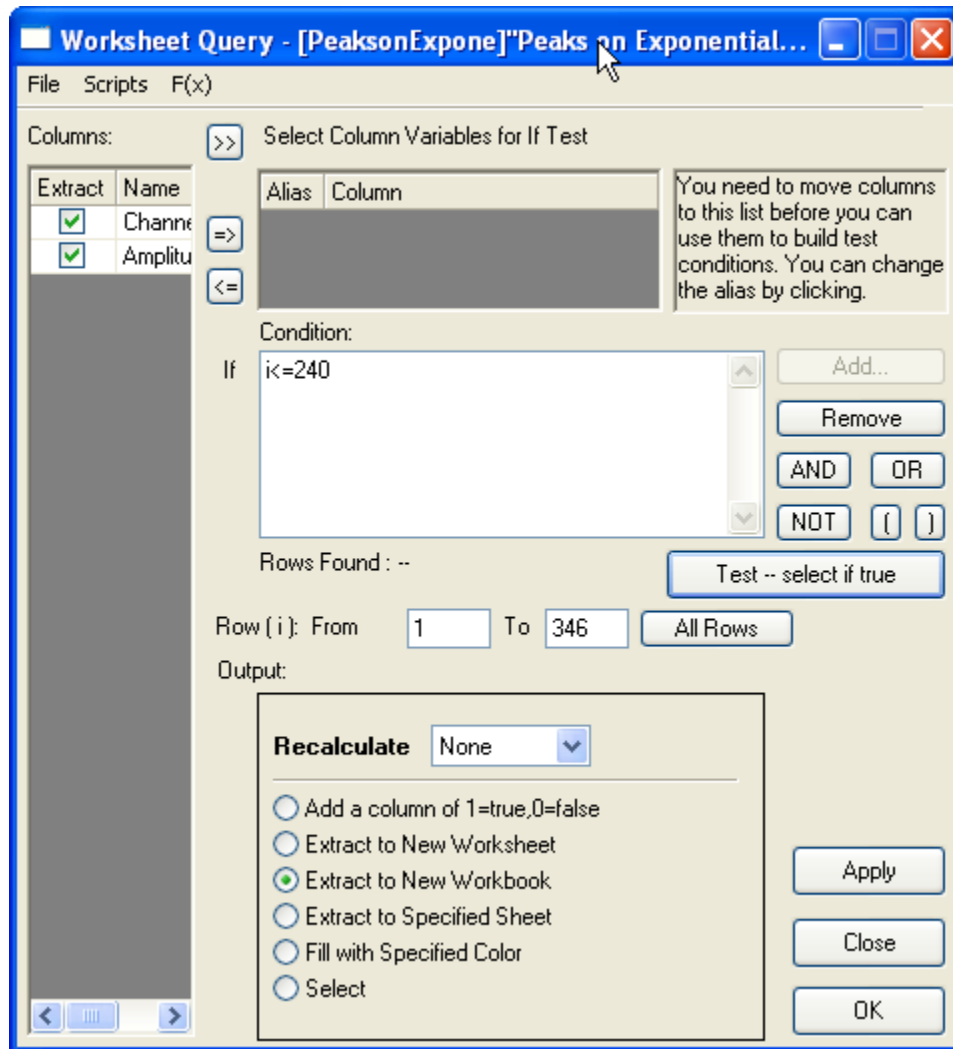
Example and Steps

Prepare the Data

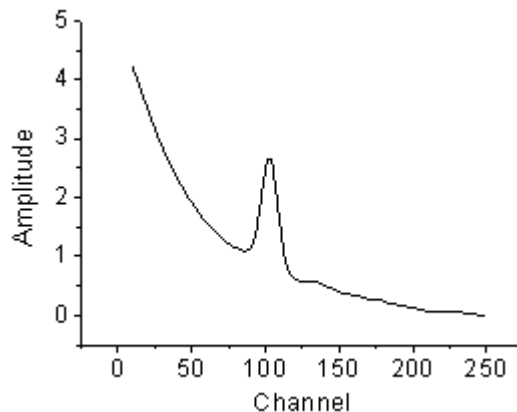
Let's start this tutorial by importing `\Samples\Spectroscopy\Peaks on Exponential Baseline.dat`. From the worksheet sparkline, we can see that there are two peaks in the curve. To simplify the problem, we will fit just one peak in this example.



Now bring up the Worksheet Query dialog from **Worksheet : Worksheet Query**. And we will extract data from row 1 to row 240:

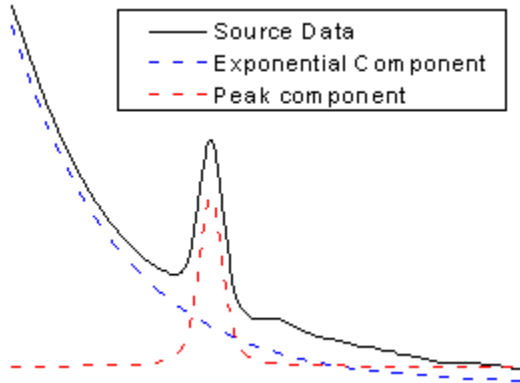


So the curve we will fit should look like this:



Define the Function

As illustrated below, we can consider the source curve is the combination of an exponential decay component (the background) with a Voigt peak:



So should we write down the whole equation to define the function? Like:

Well, this is a complicated equation and it includes infinite integration. Writing such an equation directly is painful. Now that we already have these two built-in functions:

ExpDec1:

Voigt:

we can simply use the **nlfFuncName** method to quote these two built-in functions and create a new one. Press **F9** to open the **Fitting Function Organizer** and define a function as below:

Function Name: ExpVoigt
Function Type: User-Defined
Independent Variables: x
Dependent Variables: y
Parameter Names: y0, A1, t1, xc, A2, wG, wL
Function Form: [Origin C](#)
Function: $y = \text{nlf_ExpDec1}(x, y0, A1, t1) + \text{nlf_Voigt}(x, y0, xc, A2, wG, wL) - y0;$

Note:

Some of the built-in function names do not consistent with the actual DLL function name. Just like this Voigt function, it's defined in Voigt5.FDF, and if you open the FDF file by Notepad, you can see a line under [GENERAL INFORMATION] section says:

Function Source=fgroup.Voigt5

The name after "fgroup" is the actual name we should put into **nlf_FuncName**.

Besides, for versions before **Origin 8.1 SR2**, the function body should use old **nlfFuncName** notation and define as:

```
y = nlfExpDec1(x, y0, A1, t1) + nlfVoigt(x, y0,
xc, A2, wG, wL) - y0;
x; xc; A1; t1; A2; wG; wL;
```

Listing the parameters at the end is done to avoid the "parameter not used inside the function body" error, although you already use these parameters. If not, you will not compile the function successfully.

Click the  button on the right of the **Parameter Settings** and enter these parameter initial values:

y0: 0
A1: 5
t1: 50
xc: 100
A2: 50
wG: 10
wL: 10

So the final function definition part should look like:

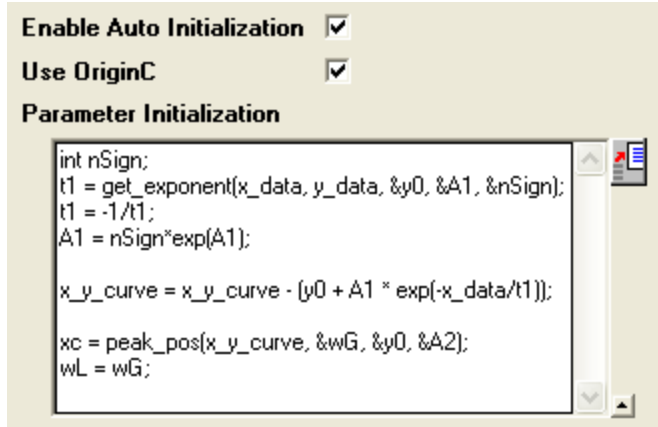
Function Type User-Defined
Independent Variables x
Dependent Variables y
Parameter Names y0,A1,t1,xc,A2,wG,wL
Function Form Origin C
Derivatives
Function
`y = nlf_ExpDec1(x, y0, A1, t1) + nlf_Voigt(x, y0, xc, A2, wG, wL) - y0;`
Peak Function
Parameter Settings
 InitialValues = 0(V),5(V),50(V),100(V),50(V),10(V),10(V)
 Meanings = ?,?,?,?,?,??

Auto Parameter Initialization

In the above section, we set fixed parameter initial values. If you know the possible fitted results, you can set the initial values in this way. But how about when the data is changed? Origin provides an Origin C interface to "guess" the initial values. To use the parameter initialization code, make sure to check the **Enable Auto Initialization** and **Use OriginC** checkboxes, and edit the code in Code

Builder by clicking the  icon.

(P.S: If you know the initial values very well, or you don't like coding, please skip this section.)



Now that the curve is composed by two components, we can guess the parameter values by separating these two parts, the initialization code includes:

1. Use the `get_exponent` function to fit the curve and get the parameter values for exponential component.
2. Remove the background -- exponential component -- from source data.
3. Approaching the peak by Gaussian peak using `peak_pos` function and set the initial values for peak component

So, the initialization code in Code Builder should look like this:

```
void _nlsfParamExpVoigt(
// Fit Parameter(s):
double& y0, double& A1, double& t1, double& xc, double& A2,
double& wG, double& wL,
// Independent Dataset(s):
vector& x_data,
// Dependent Dataset(s):
vector& y_data,
// Curve(s):
Curve x_y_curve,
// Auxiliary error code:
int& nErr)
{
// Beginning of editable part
int nSign;
// Evaluates the parameters' value, y0, ln(A) and R for y =
y0+A*exp(R*x).
t1 = get_exponent(x_data, y_data, &y0, &A1, &nSign);
// Set the exponential component values for the fitting
function.
t1 = -1/t1;
A1 = nSign*exp(A1);
// Remove the exponential component from the curve;
x_y_curve = x_y_curve - (y0 + A1 * exp(-x_data/t1));
// Fit to get peak values.
xc = peak_pos(x_y_curve, &wG, &y0, &A2);
wL = wG;
// End of editable part
}
```

Note:

When you check the **Enable Auto Initialization** and enter the initialization code, this code will cover the initial values in **Parameter Settings**.

Fit the Curve

No matter what kind of parameter initialization method you used, highlight column B and press **Ctrl + Y** to bring up the NLFit dialog, select the ExpVoigt function and fit. The result should be:

		Value	Standard Error
Amplitude	y0	0.04862	0.00724
	A1	5.08841	0.02599
	t1	50.67104	0.51939
	xc	102.81043	0.07241
	A2	32.9109	0.92012
	wG	9.65266	0.67731
	wL	5.75276	0.81022

Fitting with Piecewise Functions

Summary

We will show you how to define piecewise fitting function in this tutorial.

Minimum Origin Version Required: Origin 8.0 SR6

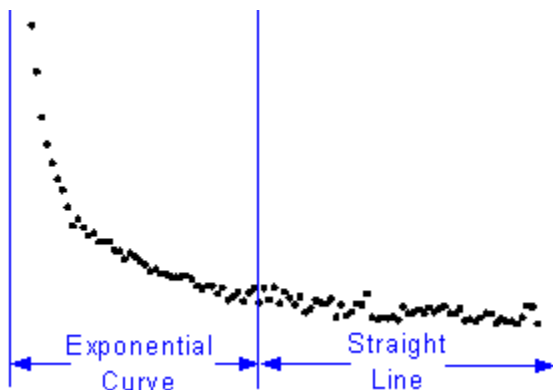
What you will learn

This tutorial will show you how to:

- Define piecewise (conditional) fitting functions.

Example and Steps

We can start this tutorial by importing the sample `\Samples\Curve Fitting\Exponential Decay.dat` data file. Highlight column D and plot a Scatter Graph. You can fit this curve using built-in functions under Growth/Sigmoidal category, however, in this tutorial, we will separate the curve into two parts by a piecewise function.



So the equation will be:

Define the Function

Press **F9** to open the **Fitting Function Organizer** and define a function like:

Function Name: piecewise
Function Type: User-Defined
Independent Variables: x
Dependent Variables: y
Parameter Names: xc, a, b, t1
Function Form: Origin C
Function:

Click the  button on the right of the **Function** edit box and define the fitting function in Code Builder using:

```
void _nlsfpiecewise(
```

```

// Fit Parameter(s):
double xc, double a, double b, double t1,
// Independent Variable(s):
double x,
// Dependent Variable(s):
double& y)
{
  // Beginning of editable part
  // Divide the curve by if condition.
  if(x<xc) {
    y = a+b*x+exp(-(x-xc)/t1);
  } else {
    y = a+b*x;
  }
  // End of editable part
}

```

Fit the Curve

Press **Ctrl + Y** to bring up NLFit dialog with the graph window active. Select the *piecewise* function we defined and initialize the parameter values:

```

xc: 1
a: 1
b: -1
t1: 0.1

```

Click **Fit** button to generate the results:

```

xc: 0.24
a: 36.76585
b: -24.62876
t1: 0.04961

```

Note that this function is sensitive to *xc* and *t1*, different initial values could generate different results.

Fit Curve Through Certain Points

Summary

This tutorial shows you three methods to force a fit curve to go thru a particular point. Choice of method depends on function expression and the data point you want the fit curve to go through.


What you will learn

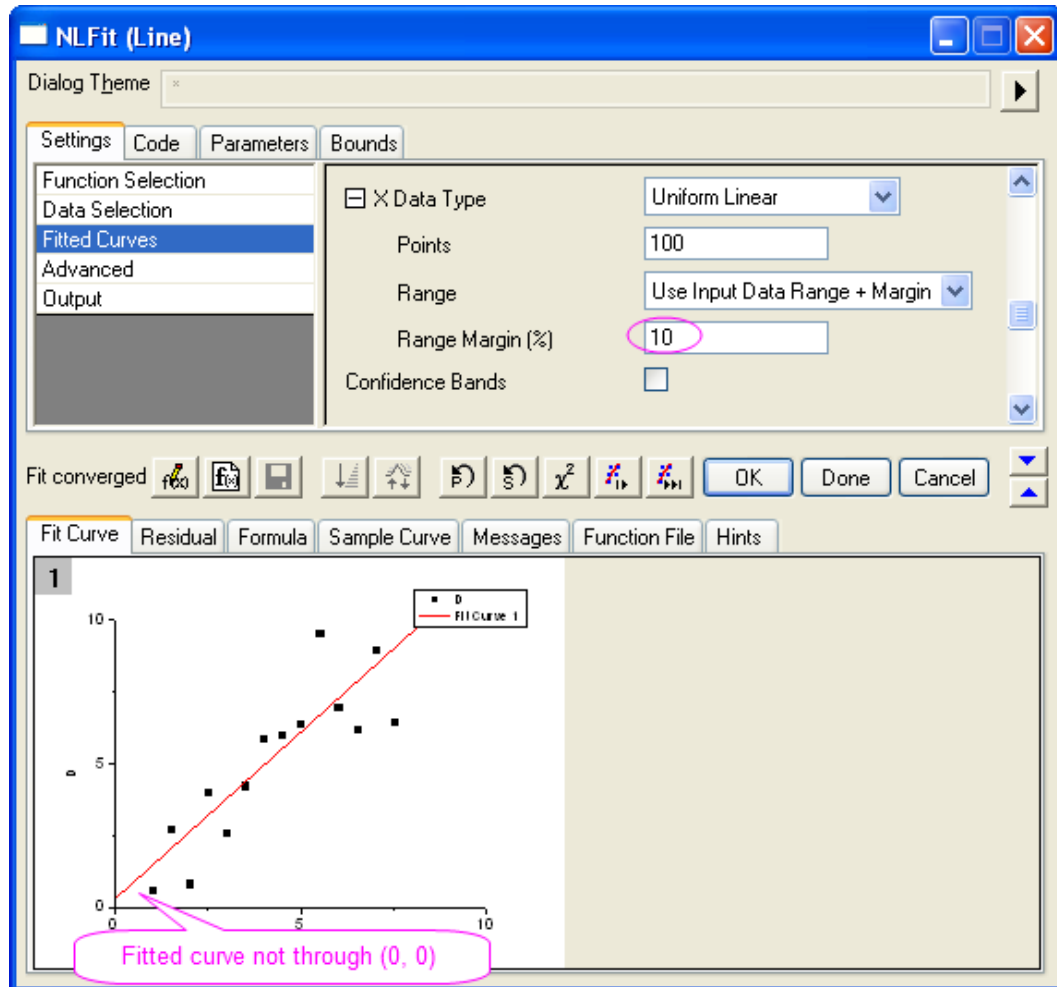
- Learn different methods to force a curve to go through a point.
- Fix fitting parameters during nonlinear fitting.
- Use general linear constraints in nonlinear fitting.
- Fit with weights.

Example and Steps

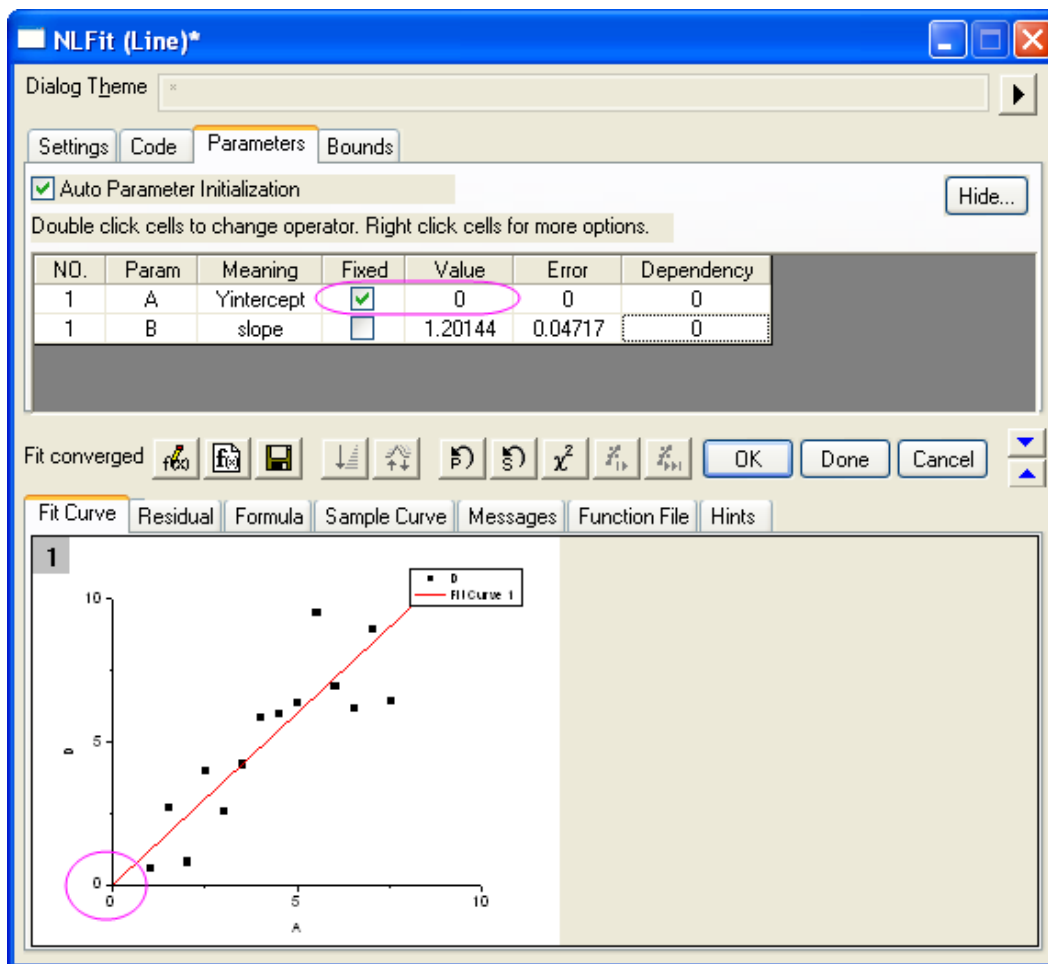
Fixing Function Parameters

This method works only when the point you want to fit through is related to a function parameter. One typical example: force the fitted line to go through the origin point, (0, 0), when fitting a straight line, $y = a + b * x$. In this particular case we know that if we let $a = 0$, the line will go through (0,0).

1. Import the data "*\Samples\Curve Fitting\Replicate Response Data.dat*" into an Origin worksheet.
2. Highlight one of the Y column, column D for example, and select **Analysis: Fitting: Nonlinear Curve Fit** to bring up the **NLFit** dialog.
3. Choose the **Line** function after selecting the **Polynomial** category.
4. Click the **Fitted Curves** page on **Settings** tab. Under the **X Data Type** branch, make sure the **Range** option is **Use Input Data Range + Margin**, and then enter **10** in the **Range Margin(%)** edit box. This option will lengthen the fitted curve.
5. Click the **Fit until converged** button . You can see from the **Fit Curve** tab that the curve does not go through the origin point.



- Now go to the **Parameters** tab, check the **Fixed** checkbox for parameter **A** and fix the value to **0**. Click the **Fit until converged** to fit the curve again. Now you can see the curve go through zero.



Note: You can also use the **Fix Intercept** option in the **Linear Fit** dialog to force the linear fitted line to go through the origin point.


Use Linear Constraint

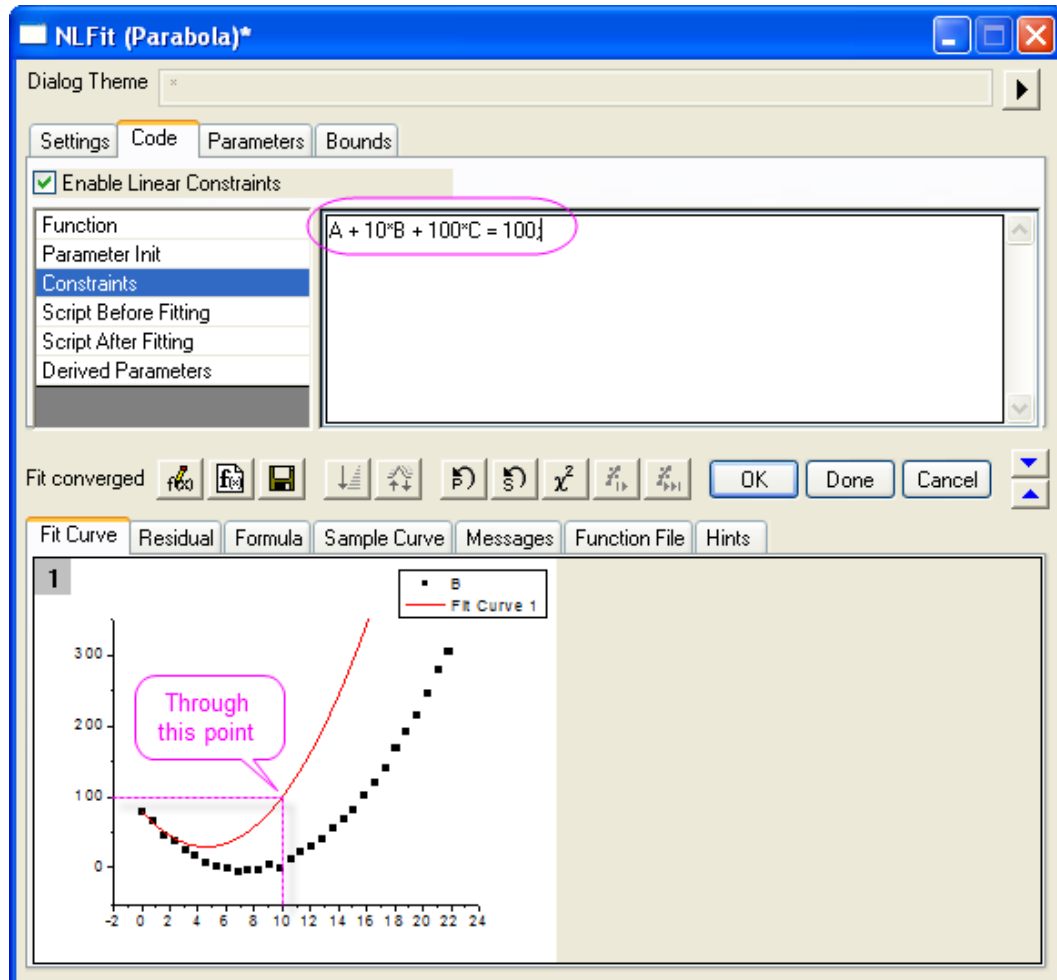
This method works when the fitting function is based on a LINEAR model, such as *Line*, *Parabola*, or *Cubic*, etc.

We will show you how to force the fitted curve to go through a particular point by using linear constraint:

1. Import the data "*\Samples\Curve Fitting\Polynomial Fit.dat*" into Origin worksheet.
2. Highlight column B and press **Ctrl + Y** to bring up the **NLFit** dialog.
3. Select **Parabola** ($y = A + B * x + C * x^2$) from the **Polynomial** category. From the **Fit Curve** tab, we can see the initial value already fits the data very well.
4. Suppose we want to force the curve through (10,100). Substitute (10,100) to the fitting function ($y = A + B * x + C * x^2$). we then have $100 = A + 10 * B + 100 * C$. We can use this equation as a general linear constraint condition. Select the **Constraints** page on **Code** tab. Check the **Enable Linear Constraints** checkbox, and enter the following expression into the edit box.

$$A + 10*B + 100*C = 100$$

5. Click the **Fit until converged** button . We can see the fitted curve deviates from the data points, but it goes through the specified point.



Use weighting

If the parameters are some eigenvalues such as upper or lower asymptotes, and your raw data includes the points you want to fit through, you can fit the curve by assigning larger weights to these particular points. This is not an analytical solution, but you can assign larger weights to reduce the error:


1. Prepare data by running the following script:

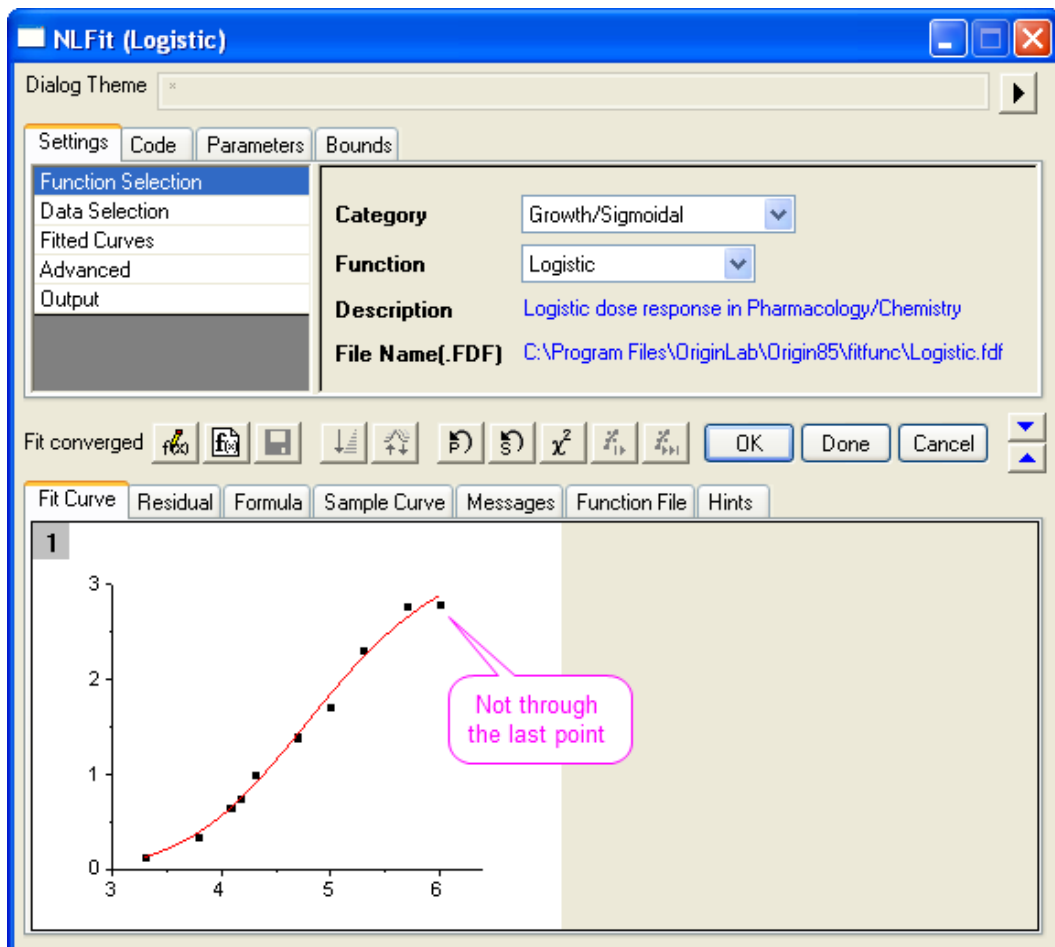
```
newbook;
string fname$ = system.path.program$;
fname$ += "Samples\Curve Fitting\Replicate Response
Data.dat";
impasc fname$ options.PartImp.Partial:=1
```

```

options.PartImp.LastCol:=2;
wks.addcol();
col(a) = log(col(a)) + 5;
col(c)[1] = 100;
for(int ii = 2; ii < wks.maxrows; ii++)
{
    col(c)[ii] = 1;
}
col(c)[wks.maxrows] = 100;

```

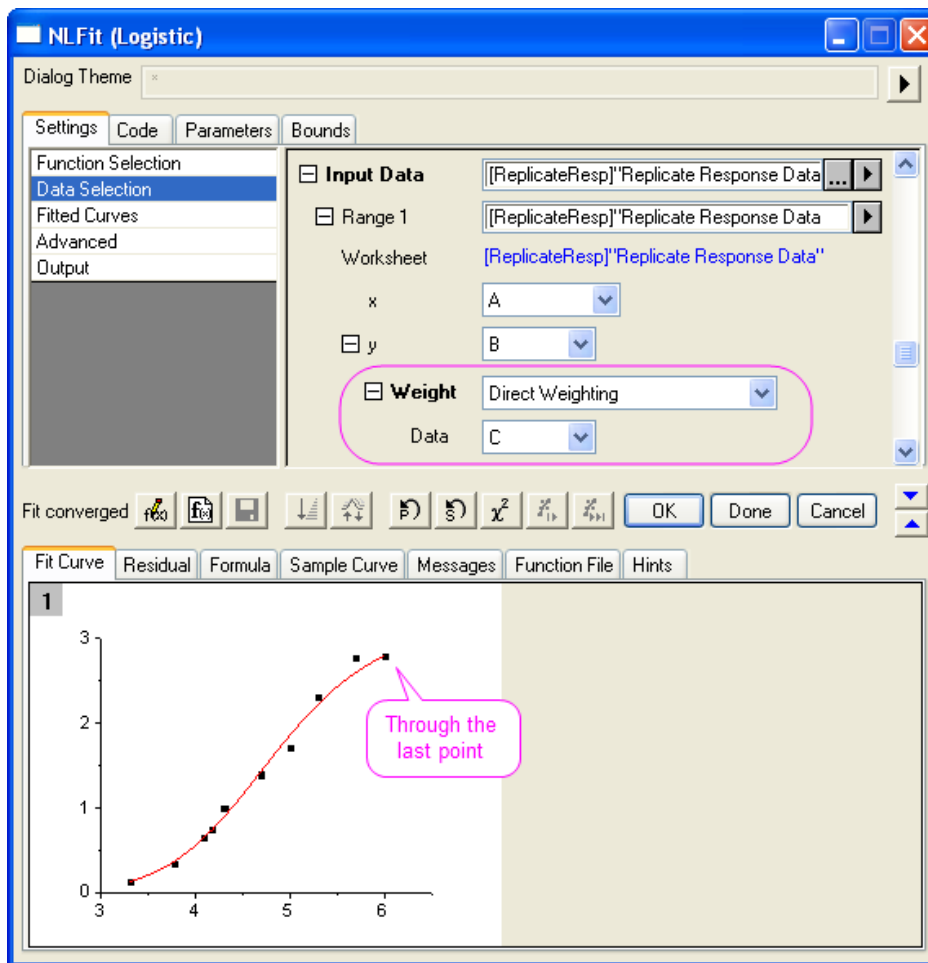
- First, let's see how the fitted curve looks when there is no weighting. Highlight column B and bring up NLFit dialog from **Analysis: Fitting: Nonlinear Curve Fit**. Select the **Logistic** function from **Growth/Sigmoidal** category. Then click the **Fit until converged** button . From the **Fit Curve** tab, we can see the curve does not go through any points near the top.



- Note that in the raw data worksheet, we have prepared column C and assigned large values for the first and last data points. If we use this column as weights, these two points will contribute more impact on the fitted curve and hence force the curve to go through these two points.

	A(X)	B(Y)	C(Y)
Long Name			
1	3.30103	0.13136	100
2	3.77815	0.34384	1
3	4.07918	0.6554	1
4	4.17609	0.73699	1
5	4.30103	1.00157	1
6	4.69897	1.39062	1
7	5	1.70785	1
8	5.30103	2.31437	1
9	5.69897	2.77326	1
10	6	2.79321	100

Now, activate the **Data Selection** page on the **Settings** tab. Expand the **Input Data** branch as below to expose the weighting option. Choose the **Direct Weighting** method and assign column **C** as the weighting dataset.



From the preview result, we can see that the fitted curve goes through the first and last data points.

Peak Fitting on Frequency Count Result

Summary

To know the location or scale parameters of a sample distribution, one can perform statistical tests on the data. However, you can also fit a probability density function on the binned data to get these values. This tutorial shows you how to estimate these parameters by curve fitting.

What you will learn

- Perform simple descriptive statistics.
- Perform frequency counts on dataset.
- Curve fitting on binned data.

Example and Steps

1. Run the following script to create sample dataset

```
newbook ;
col(2) = normal(1000) * 2 + 5;
```

2. This script generates 1000 normally distributed points where $mean \approx 5$ and $\sigma \approx 2$.
3. We can first perform simple descriptive statistics on this column to see the corresponding *Moments* output.

Highlight the data column and select **Statistics: Descriptive Statistics: Statistics on Column** to open the dialog. Make sure the *Mean* and *Standard Deviation* checkboxes are selected. And the click **OK** to generate report.

	N total	Mean	Standard Deviation
B	1000	5.03576	2.02236

From the report worksheet, we can see the Mean and Standard Deviation are very close to the value we just set.

4. Besides, you can also estimate the moments by fitting a proper probability density function on binned dataset. For example, highlight the source data column and select **Statistics: Description Statistics: Frequency Counts** from menu. This dialog will count the number of data points on specified bins.

- Expand the **Computation Control** branch, and adjust the bin size. Set the bin **Increment** to *0.5*. Make sure the **Bin Center** and **Count** check boxes under **Quantities to Compute** are selected. Then click **OK** to count the data.
5. In the frequency counts result sheet, the bin center value is set to X, while the bin counts is set to Y. You can fit the density function using these data.

Highlight the *Counts* column on the Frequency Counts result worksheet, and press **Ctrl + Y** to open the **NLFit** dialog. Then select the **Gauss** function from the *Origin Basic Function* category. Leave other options as defaults and click the **Fit** button directly to output fitting report.

		Value	Standard Error
Count	y0	-0.46104	3.25663
	xc	4.99845	0.0701
	w	4.1663	0.23397
	A	507.19423	38.34373
	sigma	2.08315	
	FWHM	4.90544	
	Height	97.1324	

From the fitting report, we can see that the fitted *xc* and *sigma* are close to 5 and 2.

1.2 Peak Analysis

1.2.1 Peak Finding

Picking and Marking Peaks

Summary

The Peak Analyzer provides several methods to pick peaks automatically. Also, user can opt to add/delete/modify the peaks manually.

Labels are added to the peak centers after they are found or added, to show user the positions of the current peaks.

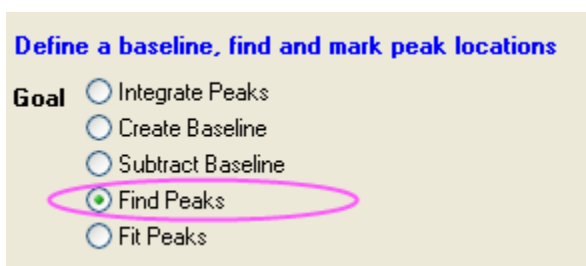
Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

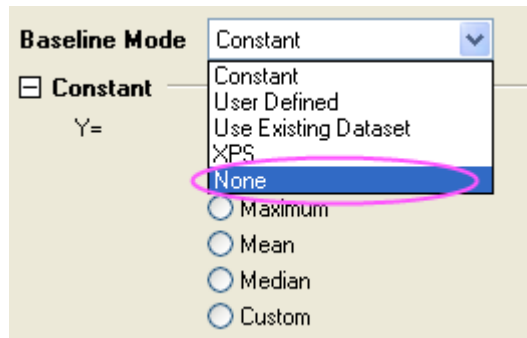
- How to use and customize the auto peak finding in the Peak analyzer
- How to customize the labels for the peak centers

Steps

1. Start a new workbook and import the file <Origin Program Folder>\Samples\Spectroscopy\HiddenPeaks.dat.
2. Highlight the second column.
3. Create a line plot by selecting **Plot: Line: Line**.
4. With the graph active, select **Analysis: Peaks and Baseline: Peak Analyzer** to open the dialog of the Peak Analyzer.
5. In the first page (the **Start** page), select the **Find Peaks** radio button in the **Goal** group. Then click the **Next** button to go to the next page.



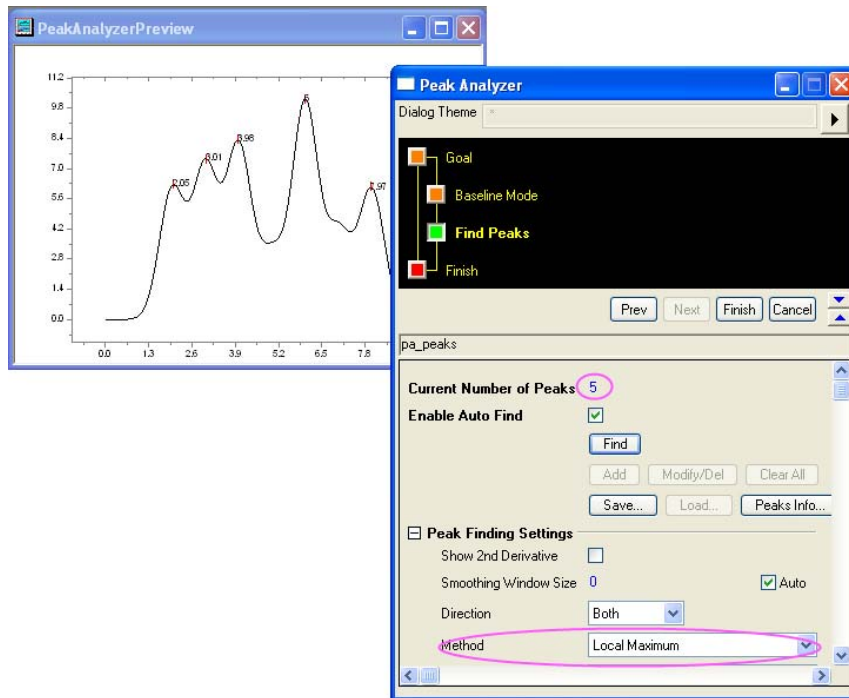
6. In the **Baseline Mode** page, select **None** for **Baseline Mode**.



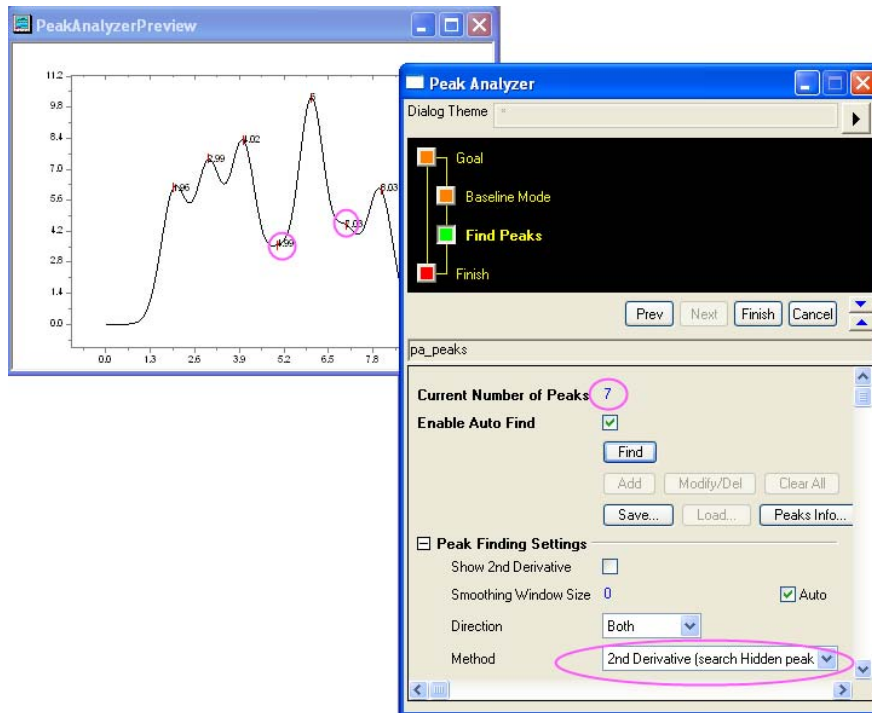
Click the **Next** button to go to the **Find Peaks** page.

7. In the **find Peaks** page:

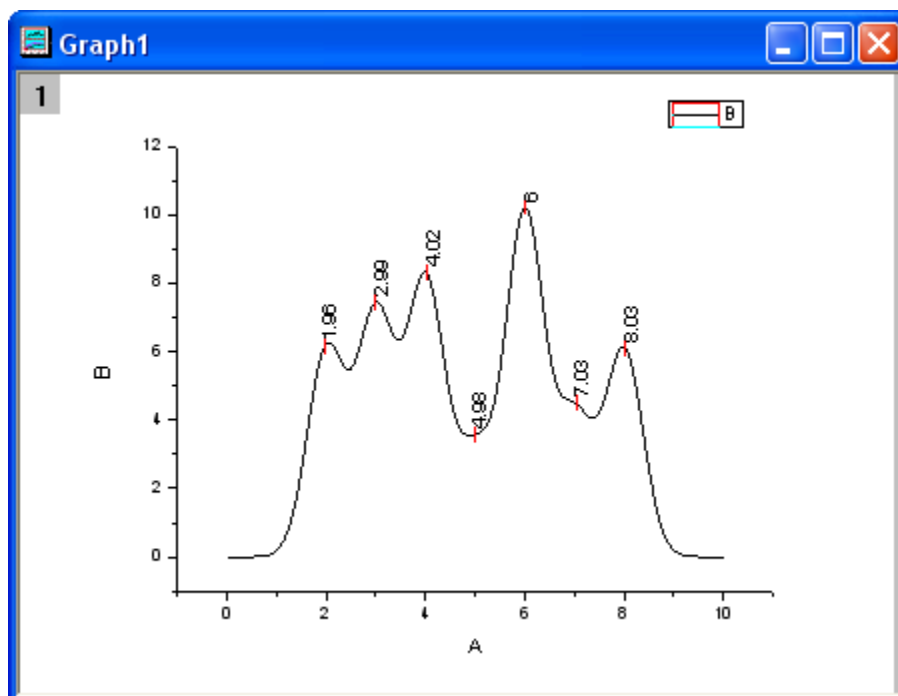
1. Expand the **Peak Finding Settings** branch. Make sure that **Local Maximum** is selected for **Method**. Then click the **Find** button. Only five peaks are detected.



2. Change **Method** to **2nd Derivative (Search Hidden Peaks)**. Click the **Find** button again. This time, seven peaks are detected.



3. Click **Finish** to complete the analysis. We will get this final graph:



1.2.2 Peak Integration

Integrating Peaks

Summary

The Peak Analyzer is capable of integrating peaks to find their areas.

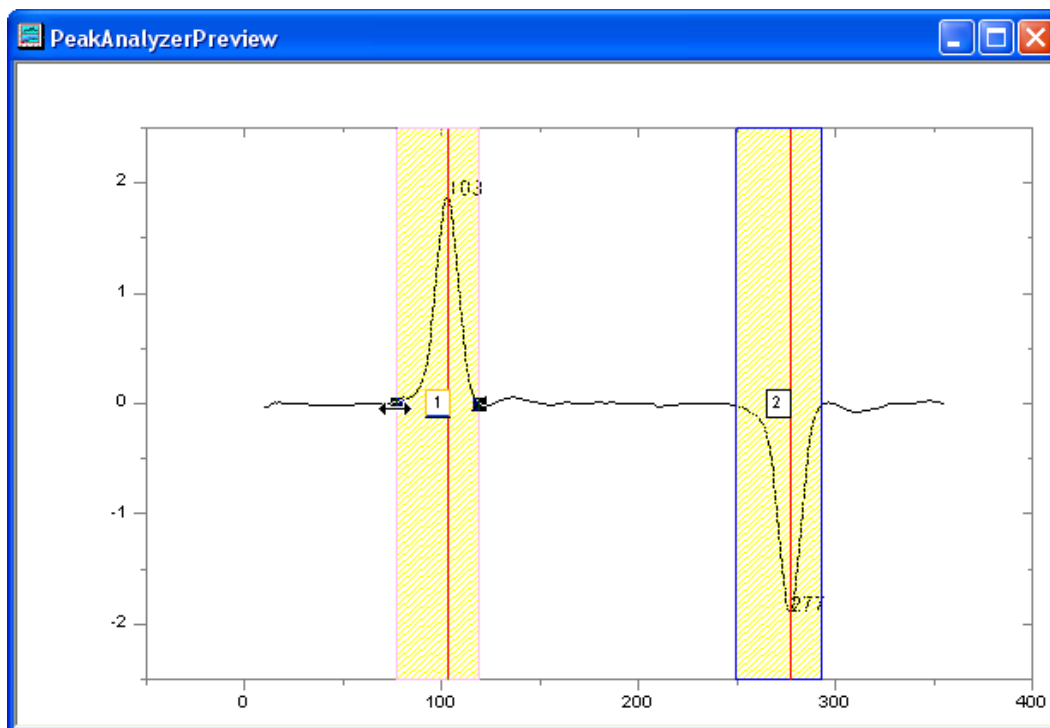
Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

- How to pick an existing dataset as baseline
- How to subtract a baseline from the spectrum data
- How to calculate peak areas with the Peak Analyzer

Steps

1. Start a new workbook and import the file `\Samples\Spectroscopy\Peaks with Base.DAT`.
2. Highlight the second column.
3. Select **Analysis: Peaks and Baseline: Peak Analyzer** from the main menu to open the **Peak Analyzer**.
4. In the first page (the **Start** page), select the **Integrate Peaks** radio button in the **Goal** group. Click **Next** to go to the **Baseline Mode** page.
5. In the **Baseline Mode** page, select **Use Existing Dataset** with the **Baseline Mode** drop-down list. Click the triangular button which next to the **Use Existing Dataset** to get the context menu and then select **Select Columns** to open the **Dataset Browser** to select the baseline dataset. In the **Dataset Browser** dialog, select column **C**, click **Add** button and then **OK** button to select the dataset as the baseline dataset. Click **Next** to go to the **Baseline Treatment** page.
6. Select the **Auto Subtract Baseline** check box. Click the page icon for the **Integrate Peaks** page in the wizard map to go to the last page (or you can click the **Next** button twice to directly go to the **Integrate Peaks** page). Please note that two peaks should be found in the **Find Peaks** page by using the default settings. You can see two numbered yellow rectangles added to them on the preview graph.
7. Set the **Integration Window Width** item as **Adjust on Preview Graph**, then click inside the rectangle marked with "1". A pair of handles (small black rectangles) appears on the left and right sides, allowing you to resize the yellow rectangle so as to change the range to perform integration on the first peak.



Similarly, you can resize the yellow rectangle around the second peak to modify the integration range for this peak.

- In the last page of the Peak Analyzer, make sure all the desired quantities to compute have been selected in the **Quantities** group. For example, if you want to calculate the peak centroid for each peak, select the **Peak Centroid** check box. If you don't want to output the percent areas, clear the **Percent Area** check box. When you are done, click **Finish** to perform the analysis. The result is in a worksheet named *Integration_Result1*.

	Index	P0(Y)	P1(Y)	P2(Y)	P3(Y)	P4(Y)	P5(Y)	P6(Y)
Comments	Integral R	Integral Re	Integral	Integral	Integral R	Integral	Integral R	Integral Res
Long Name	Index	Area	Beginnin	Ending	FWHM	Center	Height	Centroid
1	1	26.62295	77	119	13.16915	103	1.87906	102.12429
2	2	-27.62137	249	293	13.28994	277	-1.9043	275.88972
3								
4								

If the **Area** check box was selected in the **Quantities** group, you can see the peak areas in the **Area** column of this worksheet.

1.2.3 Peak Fitting

Peak Fitting with Baseline

Summary

In OriginPro, the **Peak Analyzer** is capable of performing multiple peak fitting with several baseline subtraction options.

There are various ways to create a baseline for your spectrum data. You can select a few anchor points and then fit them with a function. The fitting of the baseline can be done along with the peak fitting.

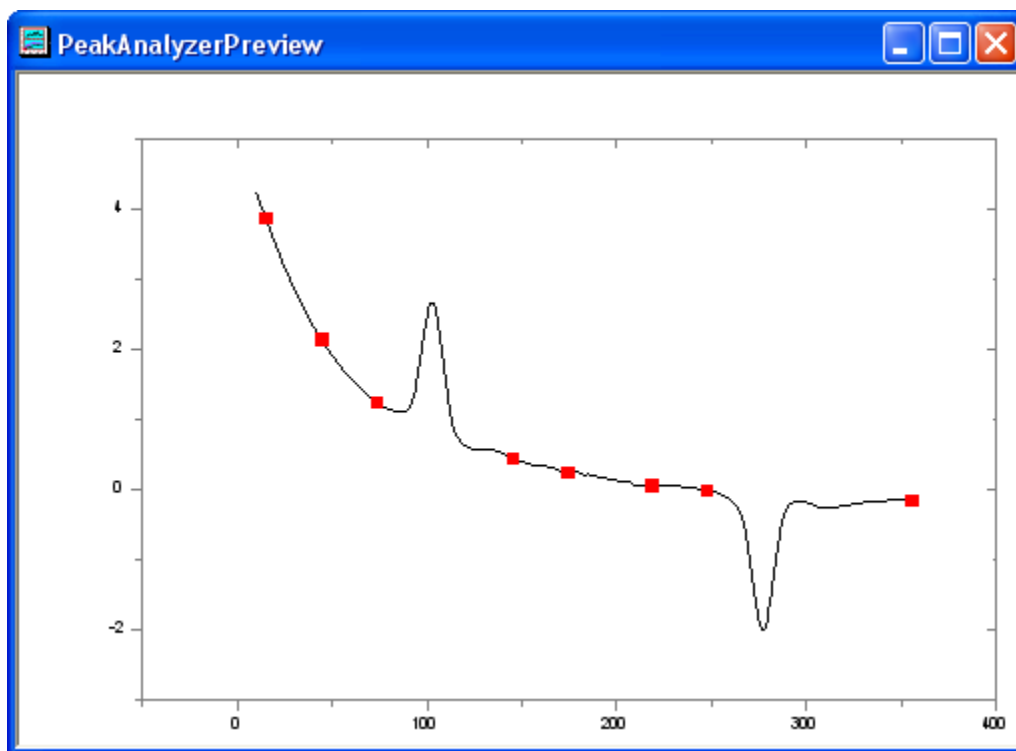
Minimum Origin Version Required: OriginPro 8.0 SR6

What You Will Learn

- How to perform fitting of peaks
- How to fit the baseline

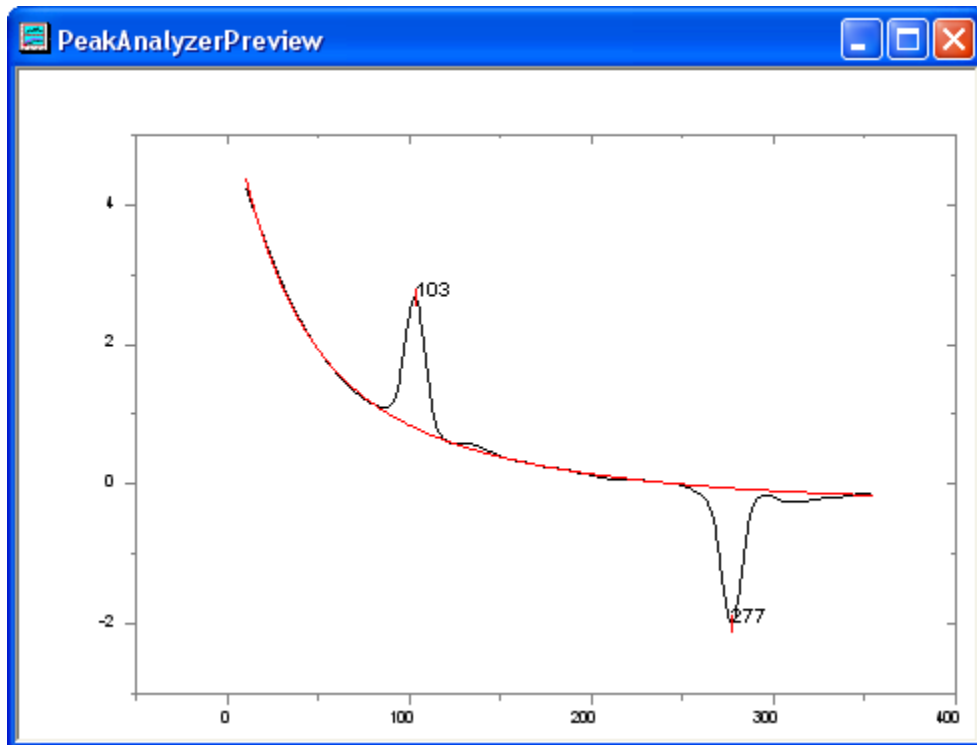
Steps

1. Start a new worksheet and import the file <Origin Program Folder>\Samples\Spectroscopy\Peaks on Exponential Baseline.dat.
2. Highlight the second column in the worksheet.
3. Select **Analysis: Peaks and Baseline: Peak Analyzer** from the main menu to open the **Peak Analyzer**.
4. Select the **Fit Peaks** radio button in the **Goal** group on the first page. Click **Next** to go to the **Baseline Mode** page.
5. In the **Baseline Mode** page, select **User Defined** from the **Baseline Mode** drop-down list. Click the **Find** button in the **Baseline Anchor Points** group. Eight anchor points should be found.



Click **Next** to go to the **Create Baseline** page.

6. In the **Create Baseline** page, select **Fitting** with the **Connect By** drop-down list. In the **Fitting** group, select **ExpDec2** from the **Function** drop-down list. Click **Next** to go to the **Baseline Treatment** page.
7. In the **Baseline Treatment** page, select the **Fit Baseline with Peaks** check box. Click **Next** to go to the **Find Peaks** page.
8. In the **Find Peaks** page, click the **Find** button to search peaks. Two peaks should be found.



Click **Next** to go to the **Fit Peaks** page.

9. In the **Fit Peaks** page, click the **Fit Control** button to open the **Peak Fit Parameters** dialog.
10. In the **Peak Fit Parameters** dialog, make sure that both peak types are Gaussian. Click the **Fit Until Converge** button. When the fitting is done, click **OK** to close the dialog.
11. Back in the **Fit Peaks** page, click **Finish** to complete the analysis. See the results in the source workbook and the graph report.

Peak Fitting with Preset Peak Parameters

Summary

In some cases, you may want to perform peak fitting with preset peak parameters. For example, you may have many datasets with fixed numbers of peaks and the centers of these peaks do not vary from dataset to dataset. What you are interested in is mainly other parameters of the peaks, for example, heights. Using the theme feature of the Peak Analyzer, you may carry out peak fitting with fixed peak parameters easily.

Minimum Origin Version Required: OriginPro 8.0 SR6

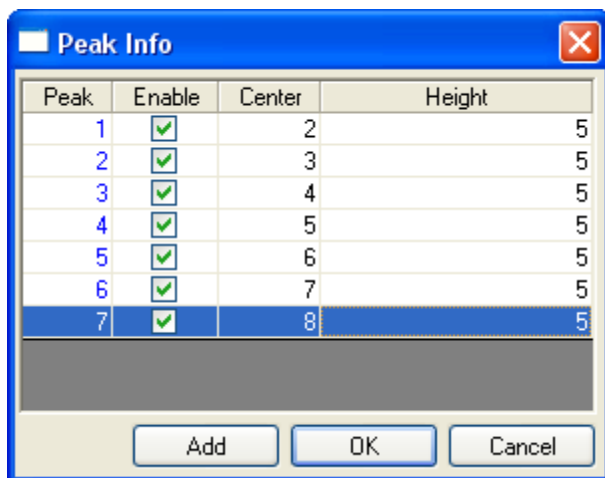
What You Will Learn

1. How to save Peak Analyzer settings in a theme and reuse them
2. How to fix peak parameters

Steps

Save a theme with peak positions and peak parameters

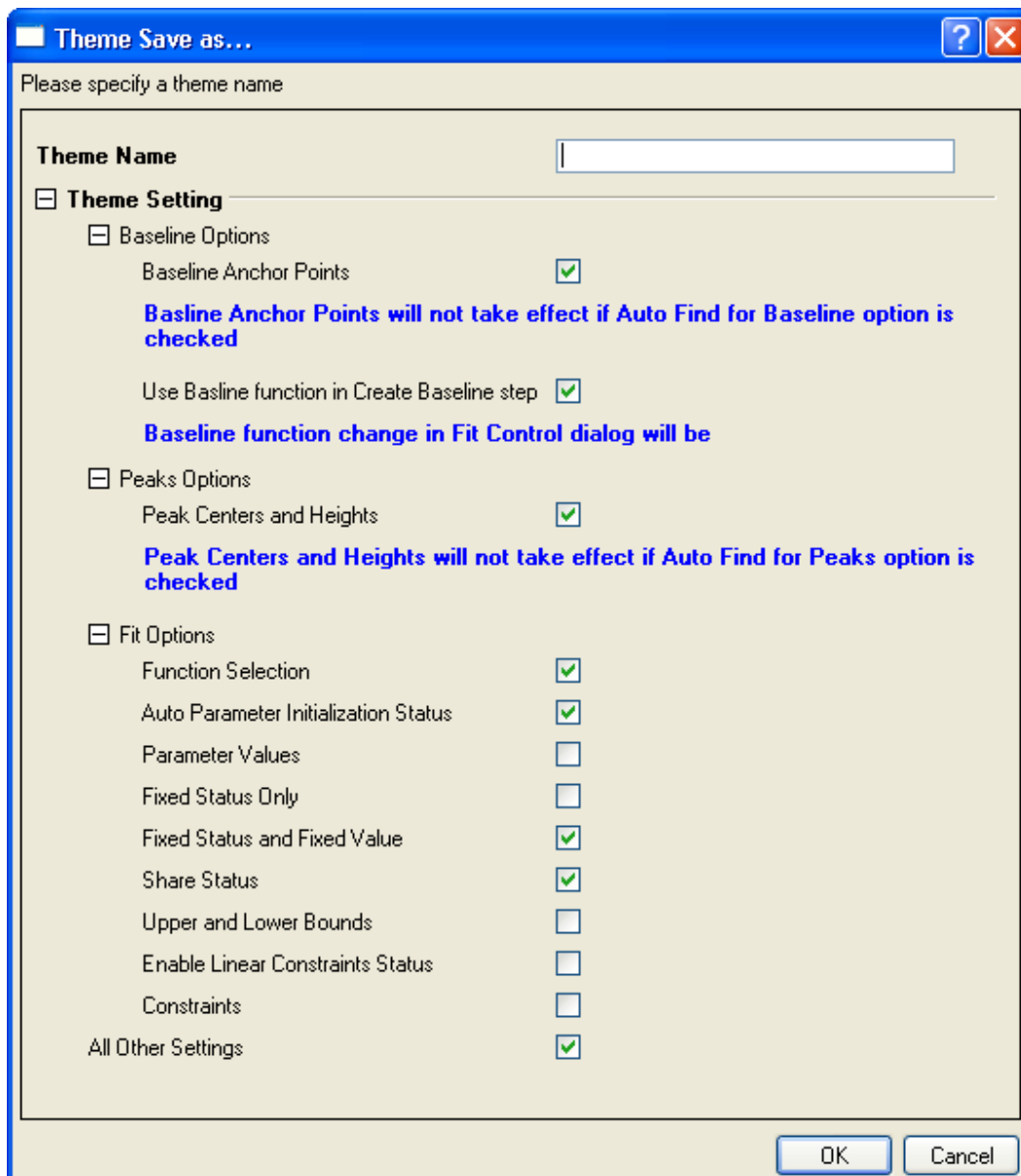
1. Start a new worksheet and import the file <Origin Program Folder>\Samples\Spectroscopy\HiddenPeaks.dat.
2. Highlight the second column and select **Analysis: Peaks and Baseline: Peak Analyzer** from the Origin menu to open the **Peak Analyzer**.
3. On the first page, select the **Fit Peaks** radio button in the **Goal** group. Click **Next** to go to the **Baseline Mode** page.
4. In the **Baseline Mode** page, select **None** with the **Baseline Mode** drop-down list. Click **Next** to go to the **Find Peaks** page.
5. In the **Find Peaks** page:
 - a. Clear the **Enable Auto Find** check box, because we want to find the peaks manually. Click the **Peaks Info** button to open the **Peak Info** dialog.
 - b. In the **Peak Info** dialog, click the **Add** button seven times to add 7 peaks. Enter the peak centers and heights as follows:



Click **OK** to return to the Peak Analyzer.

- c. Click **Next** to go to the **Fit Peaks** page.

6. In the **Fit Peaks** page:
 - a. Click **Fit Control** to open the **Peak Fit Parameters** dialog.
 - b. In the **Peak Fit Parameters** dialog, click the **Fix or release all peak centers** button. Then click the **Fit Until Converge** button. When the fitting is done, click **OK** to return to the **Peak Analyzer** dialog.
 - c. Click the right-sided triangle button to the right of **Dialog Theme** in the upper panel. Select **Save As** from the short-cut menu. The **Theme Save as** dialog opens.
 - d. In the **Theme Save as** dialog, enter **MyFitting** after **Theme Name**. Clear and select the check boxes as the screenshot below:



- e. Click the **Finish** button in the **Peak Analyzer** to complete the analysis.

Reuse the theme

1. Start another new workbook and import the file <Origin Program Folder>\Samples\Spectroscopy\HiddenPeaks.dat.
2. Highlight the second column
3. Select **Analysis: Peaks and Baseline: Peak Analyzer** from the Origin menu to open the **Peak Analyzer** dialog.
4. On the first page of the **Peak Analyzer**, click the right-sided triangle button to the right of **Dialog Theme**. From the short-cut menu, pick **MyFitting**.
5. Click **Next** to check if the settings in every step are correct. Note that in the **Find Peaks** page, you can see the peak centers and heights are same as last time.
6. When you reach the last page, click the **Fit Control** button to open the **Peak Fit Parameters** dialog. Make sure that all peak centers are fixed and the values are the same as last time. Click **OK** to return to the **Peak Analyzer**.
7. Click **Finish** to complete the analysis. Check the results to see whether they are the same as the results we got last time.

Setting the Fix, Share Status or Bounds for Multiple Peak Parameters Simultaneously

Summary

When performing peak analysis, one often wants to fix parameter values, or share parameters between multiple peaks, or specify bounds. If your data has a few peaks, you can simply perform these settings for each peak. But if your data has many, it may be time consuming to set individually. To make the process more efficient, the **Peak Analyzer** offers context menus which can allow you to set the fix, share status or bounds for multiple peak parameters simultaneously. For more details about these settings, please refer to the Origin Help File.

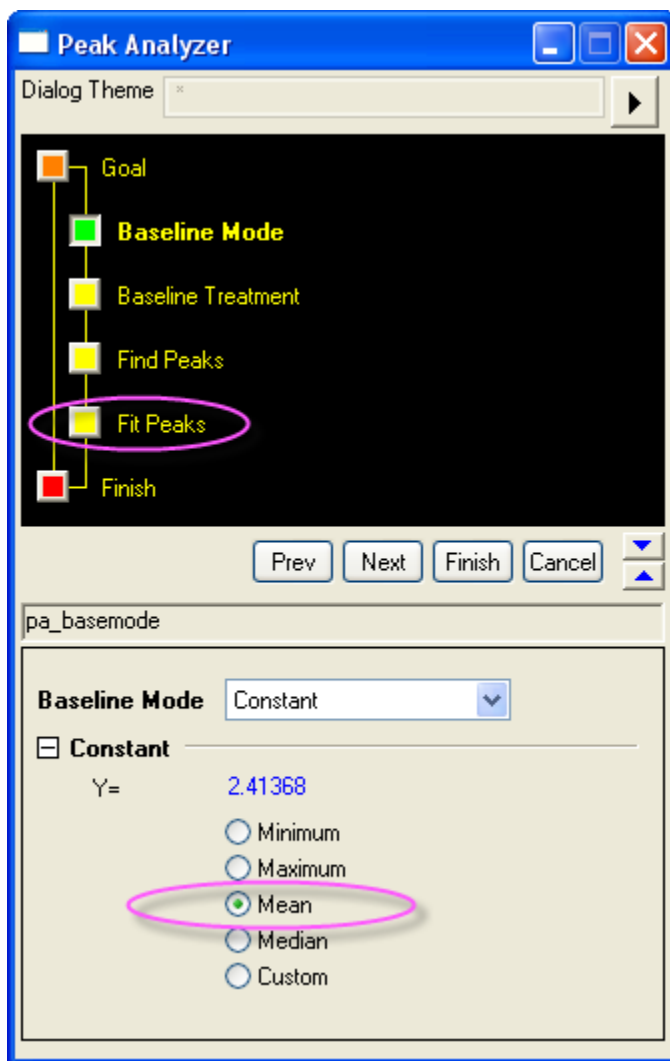
Minimum Origin Version Required: OriginPro 8.0 SR6

What You Will Learn

- How to set share status of multiple peak parameters simultaneously.
- How to set upper bounds and upper bound values to multiple peak parameters simultaneously.

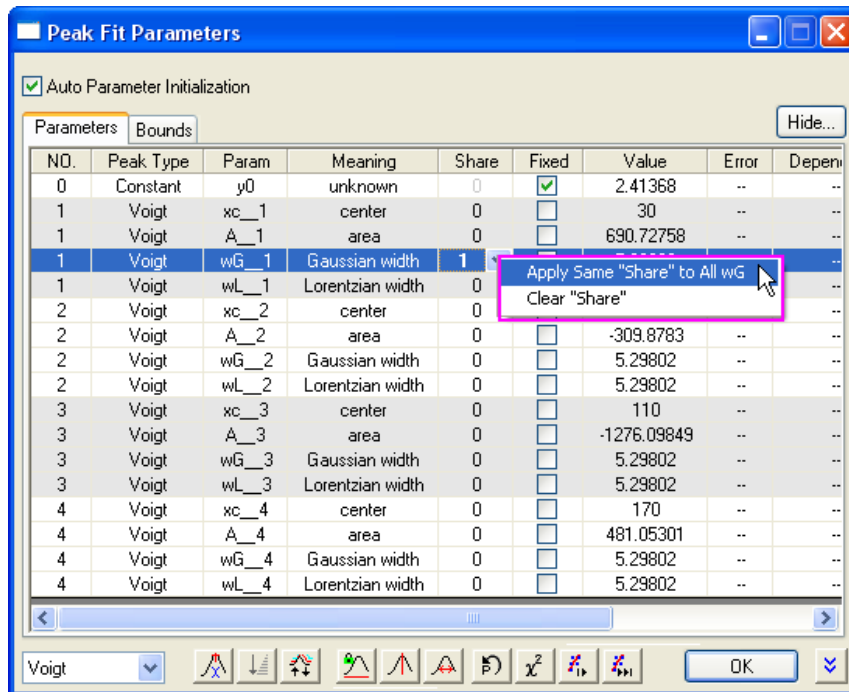
Steps

1. Start a new workbook and import the file *<Origin Program Folder>\Samples\Spectroscopy\Positive & Negative Peaks.dat*.
2. Highlight the second column and select **Analysis: Peaks and Baseline: Peak Analyzer** to open the **Peak Analyzer** dialog. In the first page (the **Start** page), select the **Fit Peaks** radio button in the **Goal** group. Then press the **Next** button to go to the next page.
3. In the **Baseline Mode** page, select **Constant** with the **Baseline Mode** drop-down list and choose **Mean** in the **Constant** group. Then click **Fit Peaks** in the wizard map to directly go to the **Fit Peaks** page.

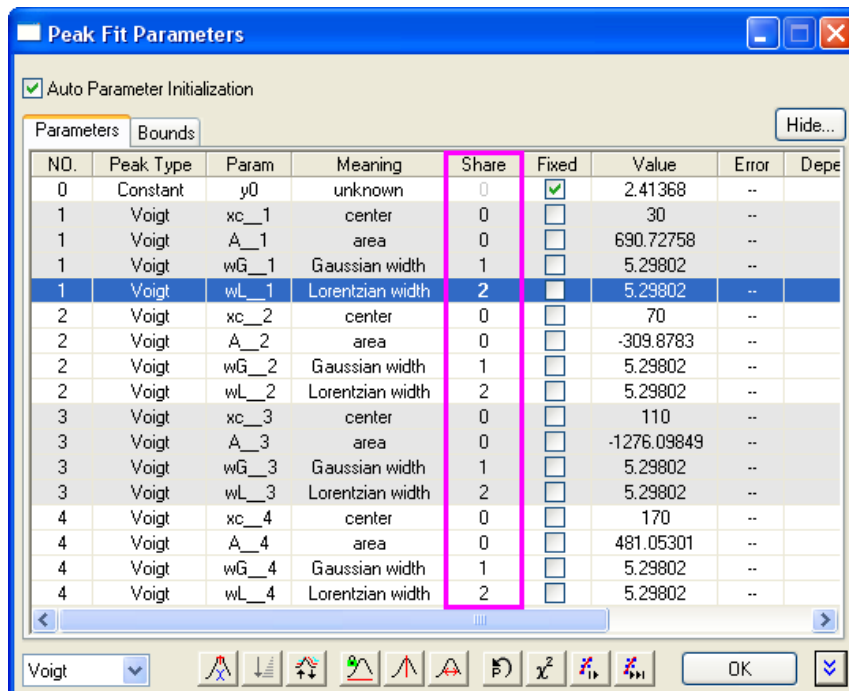


4. In the **Fit Peaks** page:
 - o click **Fit Control** button to open the **Peak Fit Parameters** dialog. In the lower left corner of the dialog, set the fitting function to **Voigt**.

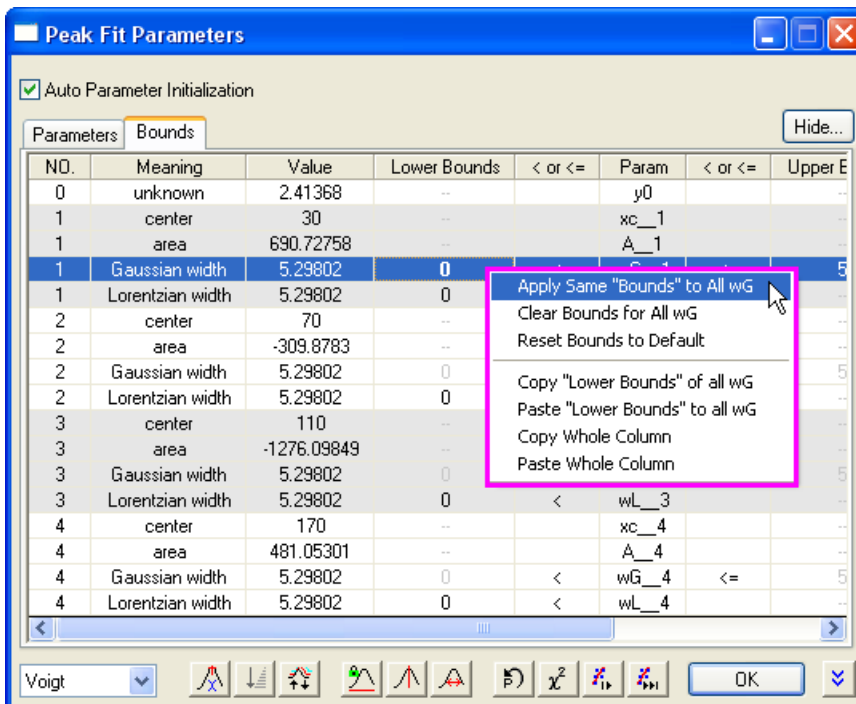
- o Make sure the **Parameters** tab is active and then select **1** in the **Share** column of the **wG_1** row. Then right click on it and select **Apply Same "Share" to All wG**. Then you will find that all the parameters with the **wG** prefix are shared in the same group.



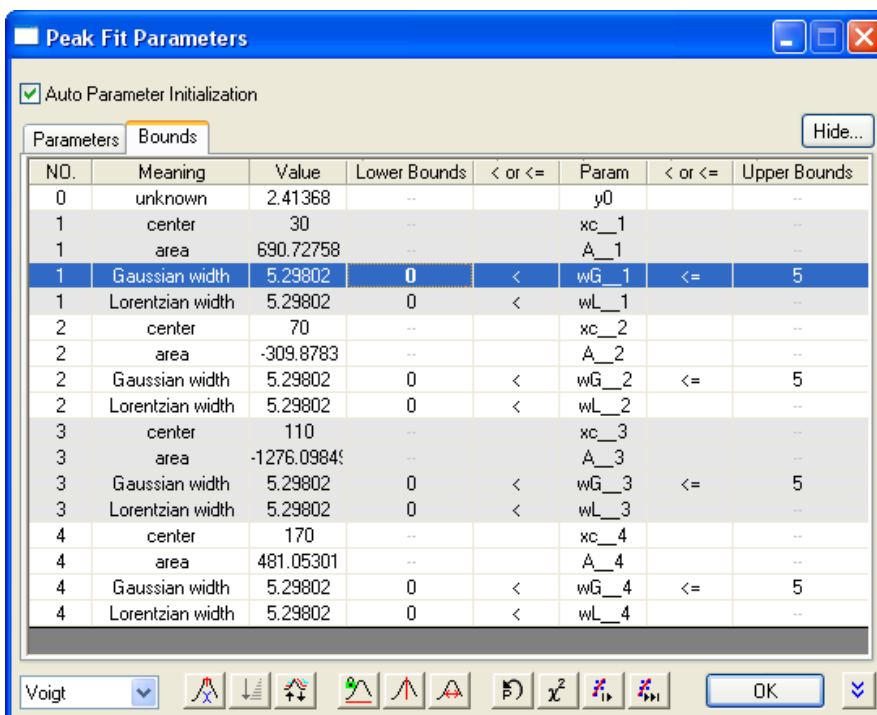
- o Then select **2** in the **Share** column of the **wL_1** row. Then right click on it and select **Apply Same "Share" to All wL**. Then you will find that all the parameters with the **wL** prefix are shared in the same group. After this, the **Parameters** tab should look like below:



- o Activate the **Bounds** tab. Double-click in the cell in the first **Gaussian width** row and the second **< or <=** column. And you will find **<=** is shown in this cell (the **<** will be shown if you double-click in the cell for one more time). Then type **5** into the **Upper Bounds** column in the same row.
- o Then right click on it and select **Apply Same "Bounds" to All wG**.



After this, the **Bounds** tab should look like:



5. Click the **Fit Until Converged** button. When the fitting is done, click **OK** to close the dialog.
6. Back in the **Fit Peaks** page, click **Finish** to complete the analysis. See the results in the source workbook and the graph report.

1.3 Data Manipulation

1.3.1 Setting Column Values

Summary

Origin provides several ways to fill a worksheet column with values. Use **Auto Fill** or script commands to fill a series of values. Use the **Set Values** dialog box to define a mathematical formula to generate or transform a data set. Refer to values in other columns from the same sheet or from other sheets and books. Select from a large collection of built-in functions to compute values. Create variables from metadata stored in worksheets or column headers, and use these variables in your column formula.

This tutorial will show you how to compute column values by:

- Filling a Column with an Arithmetic Series
- Using Built-in Functions
- Using Other Columns
- Using Cell Values
- Using Variables from Workbook Metadata

Filling a Column with Arithmetic Series

Origin provides multiple methods to fill a column with arithmetic series.

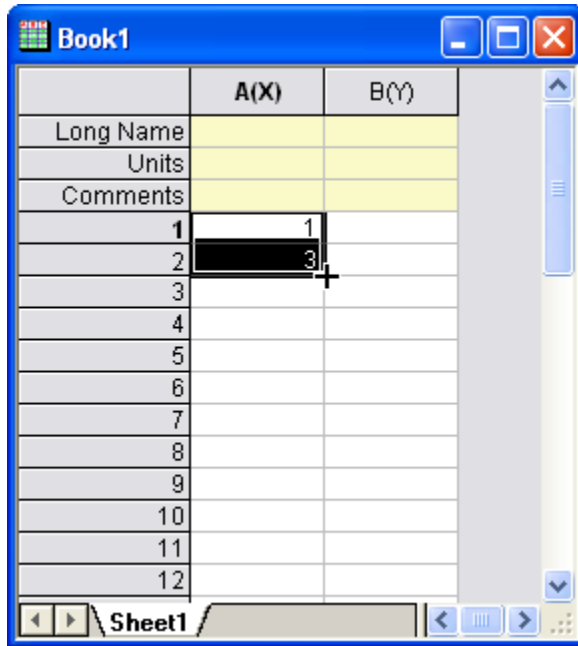
Using Auto Fill

Enter a few starting values in cells.

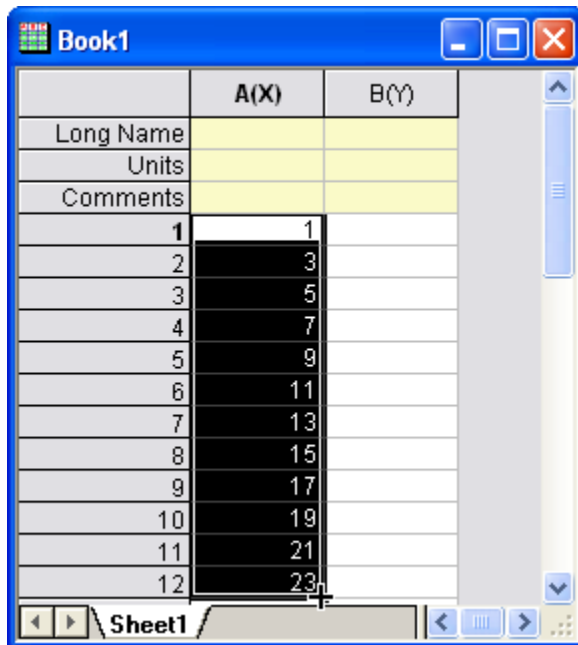
	A(X)	B(Y)
Long Name		
Units		
Comments		
1	1	
2	3	
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		

1. Select the two cells.

2. Move the mouse to the bottom right-hand corner of the second cell. The cursor will change to display "+".



3. Hold down the Alt key and drag the mouse toward the bottom of the column. The column will be filled with 1, 3, 5, 7,



Note that a row can also be auto filled by dragging towards the right. To repeatedly copy values instead of generating new values, hold down the CTRL key instead of the ALT key.

Using Data List

Type the following script in the Command window.


```
col(B) = {1:2:23};
```

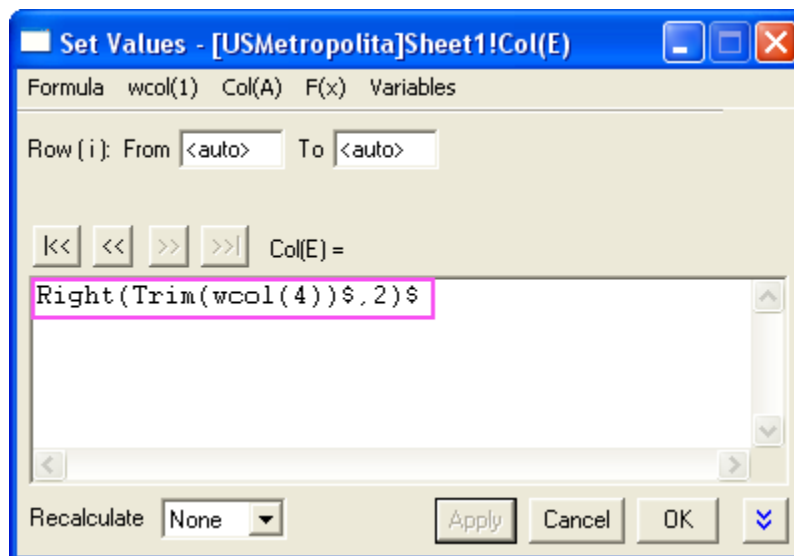
Column B will be filled with values: 1, 3, 5, 7, ..., 23



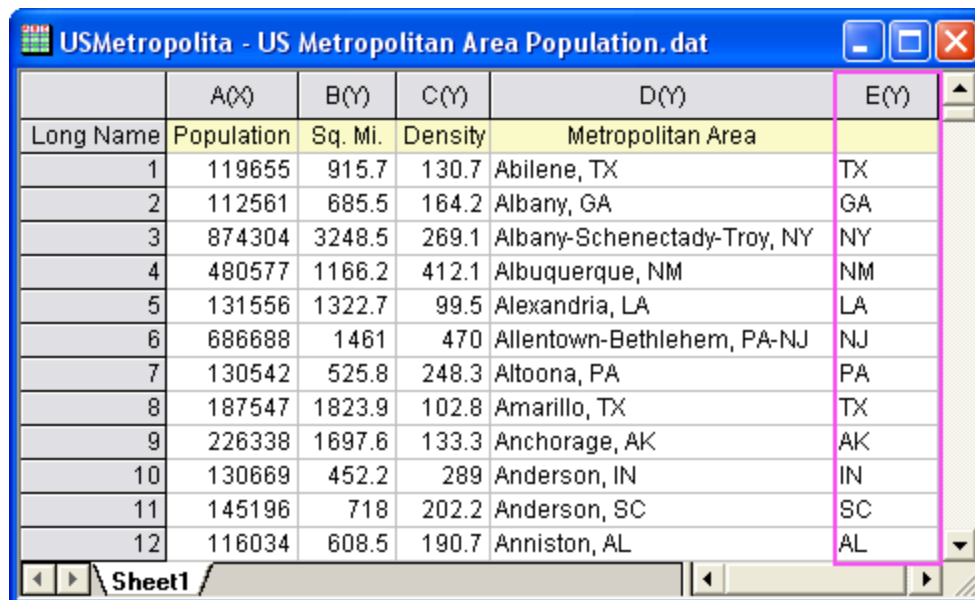
{v1:vstep:vn} produces the same result as the function data(v1,vn,vstep).

Using Built-in Functions

1. Create a new workbook. Import the *US Metropolitan Area Population.dat* file from the `\Samples\Data Manipulation\` folder.
2. Click the **Add New Columns** button  on the **Standard** toolbar to add a new column **E**. Highlight this column and right-click on it to select **Set Column Values** from the context menu. The **Set Values** dialog opens.
3. Select **F(x): String: Right(str\$,n)\$** to add **Right(,)\$** into the **Column Formula** panel.
4. Click in the position between the left parenthesis and the comma, then insert the **Trim** function by selecting **F(x): String: Trim(str\$[,n])\$**. The formula should look like: **Right(Trim(),)\$**.
5. Select **wcol(1):wcol(4)** to insert **wcol(4)** as the input of the **Trim** function. Then input **2** for the **Right** function and the expression should look like:



6. Click the **OK** button and the last column will get filled with States from column 4.



The screenshot shows the spreadsheet 'USMetropolita - US Metropolitan Area Population.dat'. The data is as follows:

	A()	B()	C()	D()	E()
Long Name	Population	Sq. Mi.	Density	Metropolitan Area	
1	119655	915.7	130.7	Abilene, TX	TX
2	112561	685.5	164.2	Albany, GA	GA
3	874304	3248.5	269.1	Albany-Schenectady-Troy, NY	NY
4	480577	1166.2	412.1	Albuquerque, NM	NM
5	131556	1322.7	99.5	Alexandria, LA	LA
6	686688	1461	470	Allentown-Bethlehem, PA-NJ	NJ
7	130542	525.8	248.3	Altoona, PA	PA
8	187547	1823.9	102.8	Amarillo, TX	TX
9	226338	1697.6	133.3	Anchorage, AK	AK
10	130669	452.2	289	Anderson, IN	IN
11	145196	718	202.2	Anderson, SC	SC
12	116034	608.5	190.7	Anniston, AL	AL

Note that some columns had two states at the end of the Metropolitan Area name, so to get both names change the formula to:

Right(Col(Metropolitan Area),Len(Col(Metropolitan Area))-Find(Col(Metropolitan Area),",")-1)\$



When referring to another column in the same worksheet, you can use index, short name, or long name to identify the column.

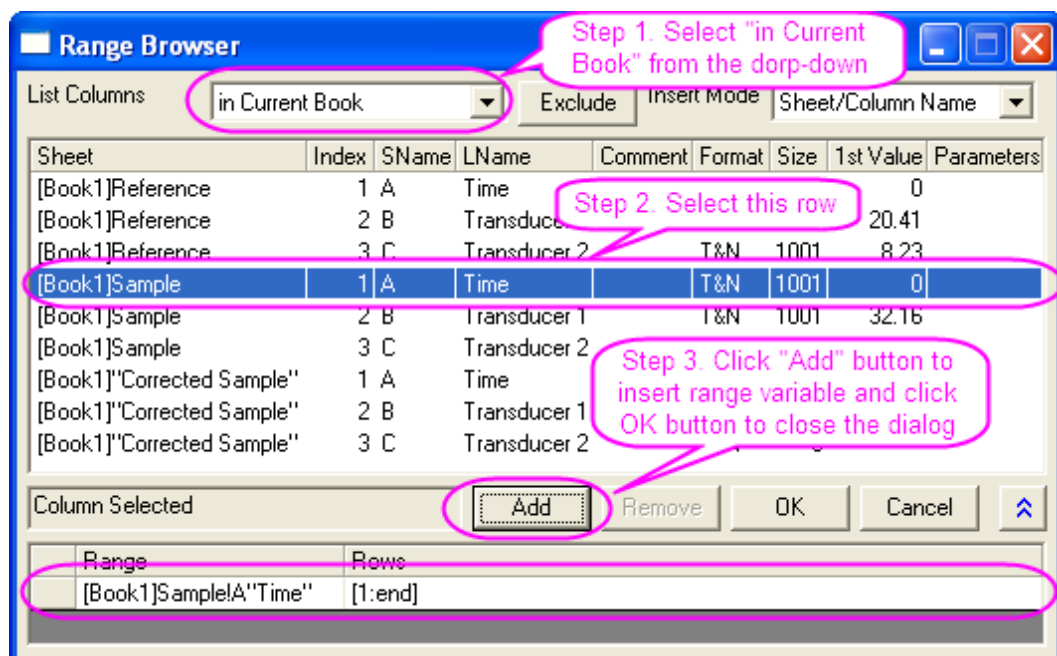
Using Other Columns

1. We will continue with the steps from above to show you how to use other columns in the Set Values dialog. Add a new column to the worksheet (right-click to the right of the last column in the worksheet and select **Add New Column** from the context menu). Change the Long Name of the column to "Population/Sq. Mi."
2. Highlight this column and right-click on it. Select **Set Column Values** to bring up the dialog. Click the Col(A) menu and choose **Col(A):Population** and then enter the / character. Click the Col(A) menu again and choose **Col(B):Sq. Mi.**. The formula should look like:
Col(Population)/Col(Sq. Mi.)
3. Click OK and the column will get computed using data from the other two columns.

Using Columns from Other Sheets

The **Set Values** dialog provides an **Insert** menu to easily insert range variables that point to columns in other books/sheets, which can then be used to compute column values for the current column.

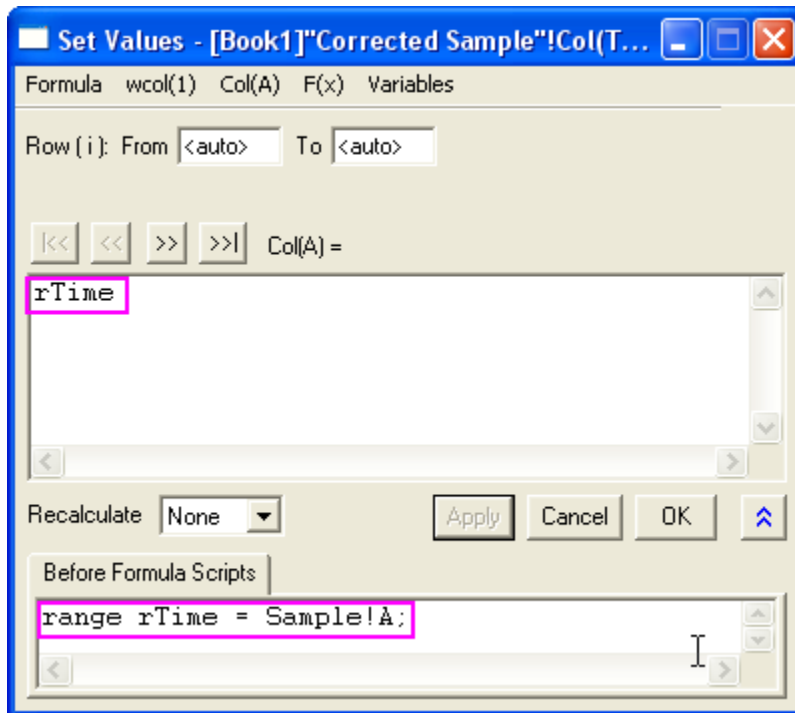
1. Open the project **Samples\Data Manipulation\Setting Column Values.OPJ** and switch to the **Columns from Other Sheets** subfolder.
2. Right-click on the **Sample** sheet and select **Duplicate Without Data**. Rename (by double-clicking on the current name) the new sheet as: **Corrected Sample**.
3. Now you will fill these three columns with data based on formulas that reference columns in the other sheets. Highlight the first column and right-click on it to select **Set Columns Values** to open the dialog. Select **Variables: Insert Range Variables** to open the **Range Browser** dialog. You will use this dialog to add a range variable to the **Before Formula Scripts** panel, according to the instructions in the image below:



Click **OK** to close the dialog. **range r1 = Sample!A** will be automatically inserted into the **Before Formula Scripts** panel. Please rename it as:

```
range rTime = Sample!A;
```

4. Then enter **rTime** in the **Column Formula** and click the **Apply** button to generate data for the first column.



5. Click the **>>** button to go to the next column. Then select **Variables: Insert Range Variables** to open the **Range Browser** dialog. You will use this dialog to insert two range variables to the **Before Formula Script** panel. Sort the data sets by long name (Click the **LName** heading to sort it). Insert two range variables that refer to **Transducer1** columns in both the **Reference** worksheet and the **Sample** worksheet. Rename them as:

```
range rRef = Reference!B;
range rSample = Sample!B;
```

6. Then input the following expression into the **Column Formula**:

```
rSample - (rSample[1] - rRef[1])
```

- Click the **Apply** button to generate data for the second column of the **Corrected Sample** worksheet. Don't click the **OK** button yet.

	A(X)	B(Y)	C(Y)
Long Name	Time	Transducer 1	Transducer 2
Units	sec	mV	mv
Sparklines			
13	12	19.99	
14	13	19.78	
15	14	19.84	
16	15	20.16	
17	16	20.01	
18	17	19.51	
19	18	19.04	
20	19	19.49	
21	20	18.91	
22	21	18.52	
23	22	19.34	
24	23	19.19	
25	24	19.03	




You reference a particular cell value with square brackets, so [1] in the formula above means the first element.

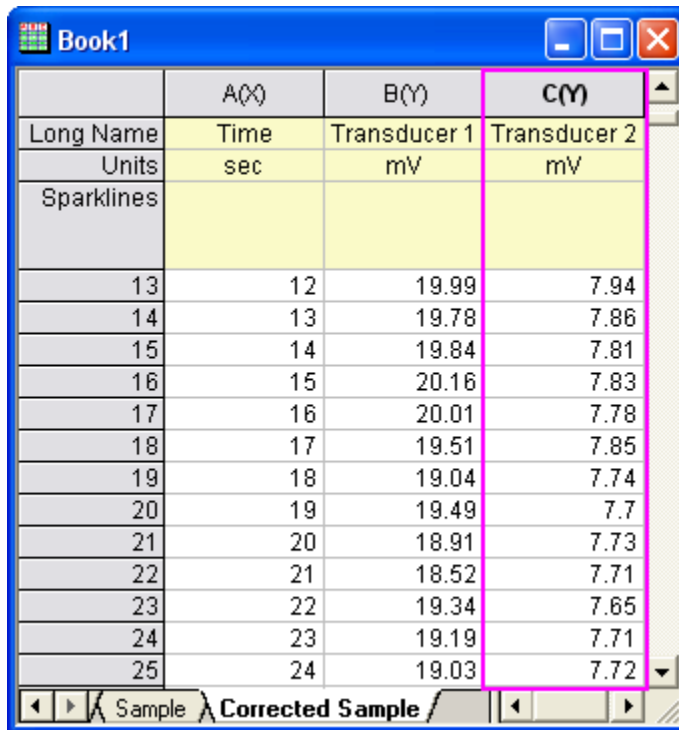
Your formulas can be saved and reloaded into other columns to generate new data.

- Now we will edit the range variables in the **Before Formula Scripts** panel and use another expression to get the same results. Remove the column names **B"Transducer 1"** of the two range variables and select **F(x): Variables and Constants: wcol(_ThisNumCol)** in both lines so it looks as follows:

```
range rRef = Reference!WCol(_ThisColNum);
range rSample = Sample!WCol(_ThisColNum);
```

- Leave the expressions in the **Column Formula** panel unchanged and click **Apply** to generate data. You will find that it gives you the same results, but the formula can now be applied to any column in the **Corrected Sample** worksheet, and the range variables will point to the same column, by index, in the **Reference** and **Sample** worksheets.
- Select **Formula: Save** to open the **Save** dialog and name it "My Correction". Click the **OK** button to save it.

4. Click the  button to go to the next column. Select **Formula: Load: My Correction** and click the **Apply** button to generate data for the third column.



	A(X)	B(Y)	C(Y)
Long Name	Time	Transducer 1	Transducer 2
Units	sec	mV	mV
Sparklines			
13	12	19.99	7.94
14	13	19.78	7.86
15	14	19.84	7.81
16	15	20.16	7.83
17	16	20.01	7.78
18	17	19.51	7.85
19	18	19.04	7.74
20	19	19.49	7.7
21	20	18.91	7.73
22	21	18.52	7.71
23	22	19.34	7.65
24	23	19.19	7.71
25	24	19.03	7.72

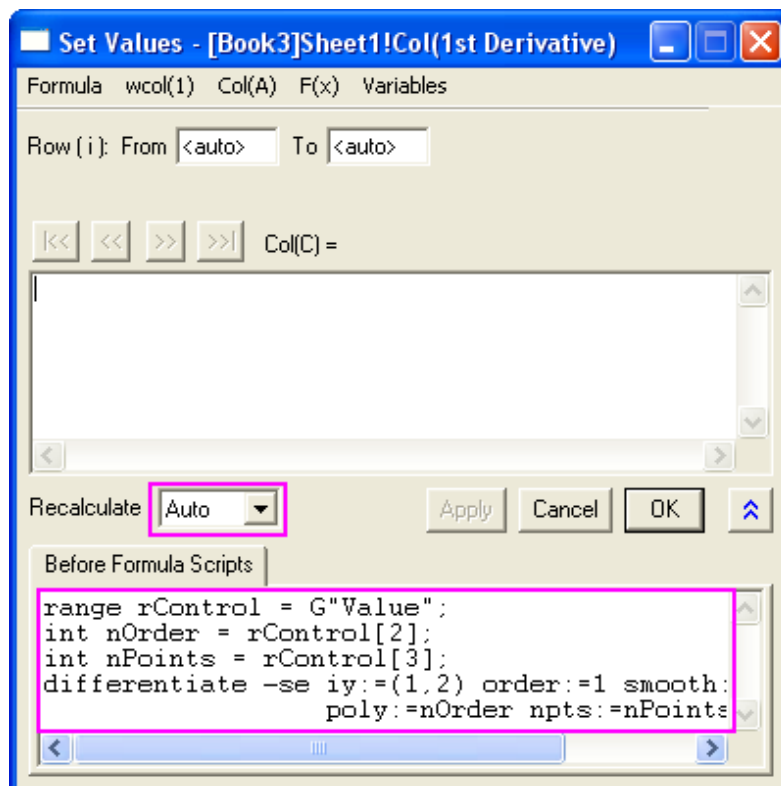
Using Cell Values

Values contained in specific worksheet cells can be referenced and used to compute the formula for setting column values. This provides an easy way to use worksheet cells as control cells for updating values in a column.

1. Open the project `\Samples\Data Manipulation\Setting Column Values.opj` and switch to the **Cells in a Worksheet** subfolder in Project Explorer.
2. Right-click on column C and select the **Set Column Values...** context menu to bring up the Set Values dialog.
3. Use the **Variables: Insert Range Variable...** menu item to open the Range Browser. Then select the column with the long name (LName) **Value**. Press the **Add** button to insert a variable. Press the **OK** button to close the dialog.
4. In the **Before Formula Scripts** panel, change the name of the range variable to be **rControl** and add these additional lines so that the script looks like below

```
range rControl = G"Value";
int nOrder = rControl[2];
int nPoints = rControl[3];
differentiate -se iy:=(1,2) order:=1 smooth:=1
poly:=nOrder npts:=nPoints
oy:=(1,3);
```

5. The script calls the **differentiate** X-Function and passes the cell values from column G as arguments for polynomial order and number of points, which controls the Savitzky-Golay smoothing performed during the differentiation.
6. Set the **Recalculate** drop-down to **Auto** and press OK to close the dialog.



7. Now you can try to change the values in column G, to change the output.

Note: Allowed values of polynomial order are 1 to 9.

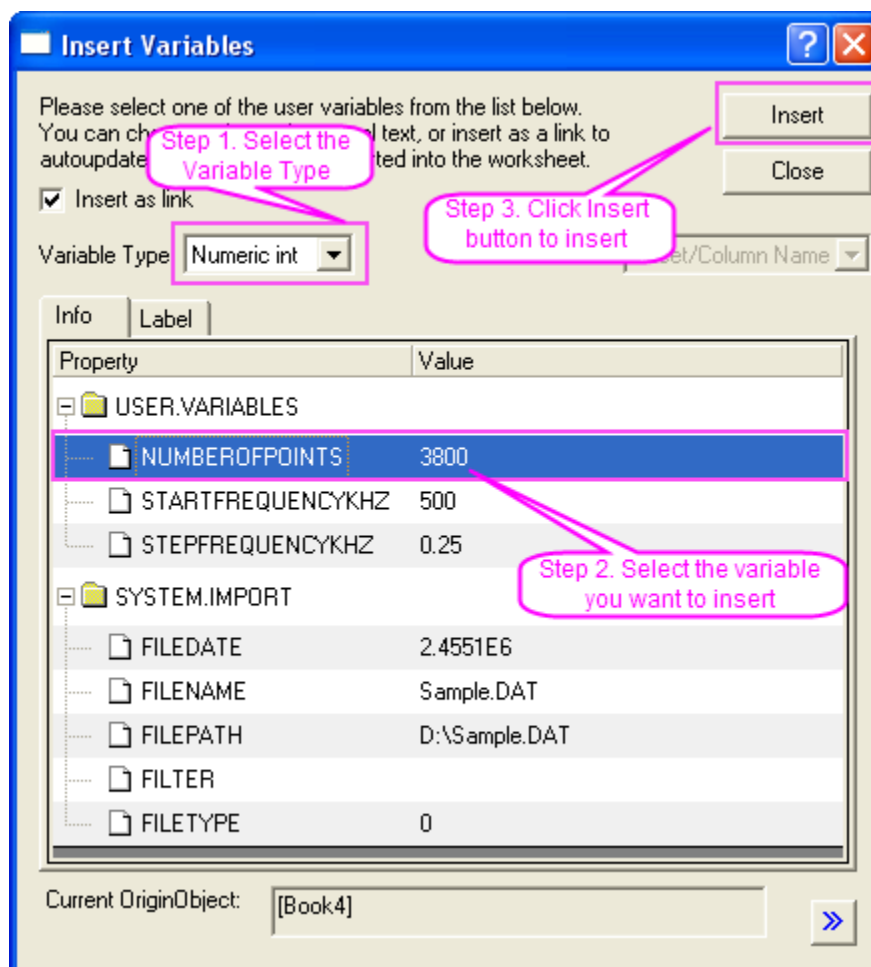


The graph shown in the worksheet was first created and then embedded into the worksheet by merging a group of cells.

Using Variables from Workbook Metadata

Metadata stored in the workbook, such as variables saved when importing data using the Import Wizard, can be referenced and used for computing column values.

1. Open or continue working with `\Samples\Data Manipulation\Setting Column Values.OPJ`, and switch to the **Worksheet Metadata** subfolder from the Project Explorer window.
2. Select column A and right-click to select the **Insert** menu option. A new column is inserted to the left of column A.
3. Select the first column (this newly inserted column) and right-click on it. Then select the **Set Column Values** menu item to open the Set Values dialog.
4. Select the **Variables: Insert Info Variable** menu item to open the **Insert Variables** dialog. Select **Numeric int** from the **Variable Type** drop-down list. Then select **NumberOfPoints** and press the **Insert** button to insert this variable into the **Before Formula Scripts** panel.



5. Next, set **Variable Type** to **Numeric double**. Hold the **Shift** key down to select both **StartFrequencyKHz** and **StepFrequencyKHz**, and then press **Insert** to insert these two variables. Press the **Close** button to close the dialog.
6. In the upper **Column Formula** panel, input $\{d1:d2:d1+(n1-1)*d2\}$ and then press the **OK** button to generate data and close the dialog. The column will be filled with frequency values.

7. Highlight the first and second columns, right-click on them and select **Set As: XYY** to change the plotting designations to X and Y. After you change the long name of the first column to **Frequency**, the worksheet should look like:

	C1(X)	A(Y)	B(Y)
Long Name	Frequency	Real	Imaginary
Sparklines			
1	500	6.35	-4.39
2	500.25	5.98	-4.27
3	500.5	5.86	-3.91
4	500.75	6.23	-3.66
5	501	6.47	-3.42
6	501.25	6.47	-3.3
7	501.5	6.71	-2.69
8	501.75	6.1	-3.05
9	502	5.74	-2.32
10	502.25	6.23	-2.2
11	502.5	6.1	-1.71
12	502.75	5.86	-2.08
13	503	5.74	-2.08
14	503.25	5.27	2.2

1.3.2 Worksheet Data Operations

Worksheet Query

Summary

This tutorial will show you how to use the **Worksheet Query** dialog.

Minimum Origin Version Required: Origin 8.5.1 SRO

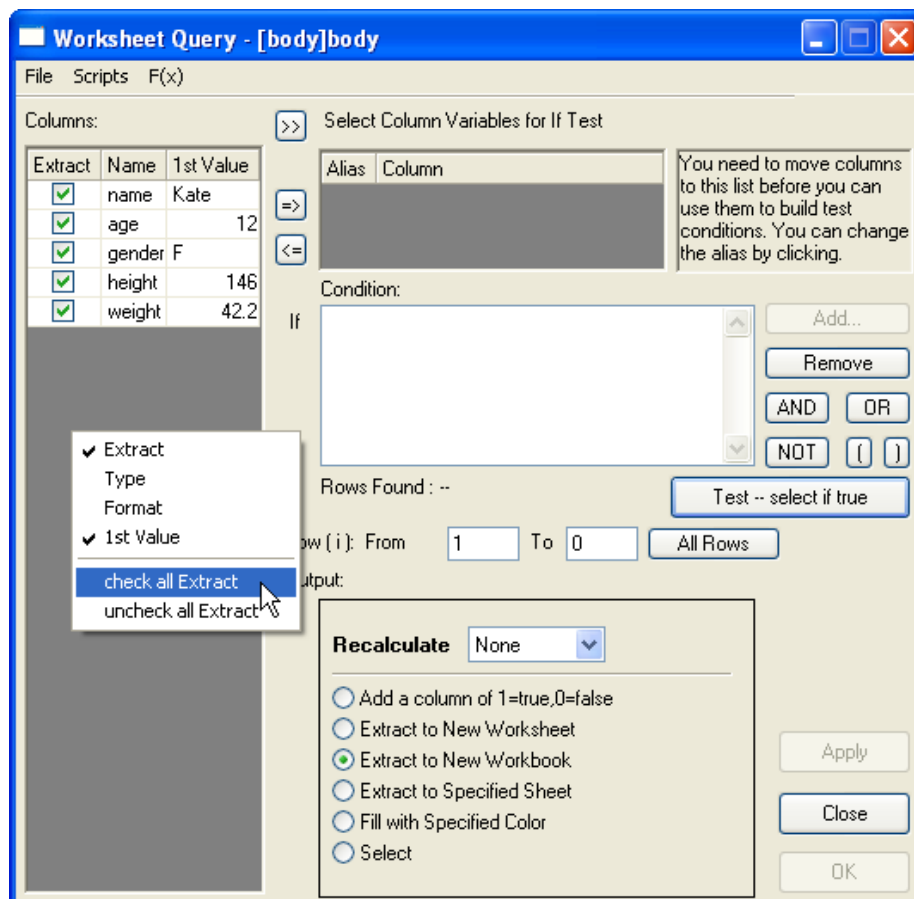
What you will learn

This tutorial will show you how to:

- Extract numeric and time data
- Use an alias in an extraction condition
- Use LabTalk functions in an extraction condition


Dialog overview

Start with a new workbook and import the file `\Samples\Statistics\body.dat`, click the menu item **Worksheet: Worksheet Query** to open the dialog as follows:



Basically, there are two main panels in the **Worksheet Query** dialog. The left panel lists all the columns in the active worksheet, you can right-click and select some column properties you want to see, such as *Format*, *1st Value*, etc.

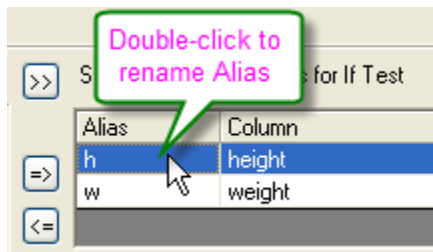
Note the **Extract** column in this panel, only data selected in the Extract checkbox will be extracted.

The right panel is where you set and test extract conditions. For example, you can select the column you want to use in the extract condition, and then click the  button to move it into the **Select Column Variable for If Test** group.

Set the conditions

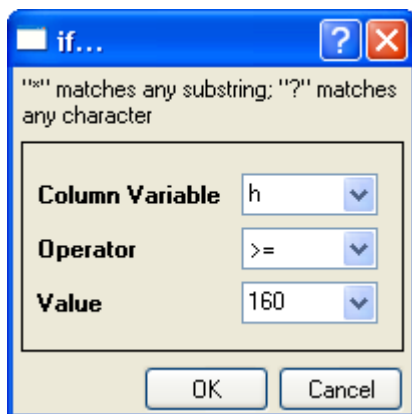
Extract Numerical Data

When there are available columns in the **Select Column Variable for If Test** group, the **Condition** edit box becomes editable for you to set conditions. For example, select *height* and *weight* to the group, Origin will automatically set an alias for each column. You can click into the **Alias** cell and rename the alias:

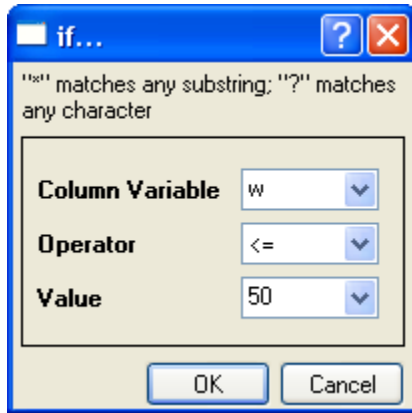


These alias can be used directly in the extract condition. Let's keep the default alias, *h* and *w* in this example.

The buttons on the right side of **Condition** edit box can be help to establish extract conditions. For example, to extract data that height is greater and equal to 160cm, highlight the column on **Select Column Variable for If Test** and click **Add** and build the first condition as follow:



When there are multiple conditions, you can also combine these conditions by logical operation. Click **AND** button to add one more condition. Then highlight *w* on **Select Column Variable for If Test** group and click **ADD** again, this time, we are looking for weight less than or equal to 50kg:



When the condition is done, click **All Rows** button to select all rows, then click the **Test -- select if true** button and Origin will return 5 found records. Of course, if you familiar to logical operation syntax, you can type the condition on the edit box directly:

```
h>=160 AND w<=50
```

Accept other default settings and click the **OK** button. A new workbook is created with these 5 records.

Extract Strings

When extracting strings, you need to enclose the string by double quotation marks ". For example, select the *gender* column into the **Select Column Variables for If Test** group. Using the alias *g*, you can extract all female data by:

```
g == "F"
```

Extract Time Data

Date and Time data are internally saved as numeric values in Origin. Date is the integer part of the numeric value, while Time is the fractional part. In Origin, you can use the *int()* and *frac()* functions to return the integer and fractional part of a number, and use the *Date(MM/DD/YY)* and *Time(HH:mm:ss)* functions to transfer string to time data. We can combine these functions to extract time data.

For example, using data from Import Time Data tutorial, you can extract data within time period 10:00 ~ 11:00 by:

```
frac(B) > Time(10:00:00) AND frac(B) < Time(11:00:00)
```

You can see Origin found 120 records. Similarly, if you want to extract *Date* data, you can try some condition like:

```
int(A) > Date(01/24/2004)
```

1.4 Analysis Templates

1.4.1 Creating and Using Analysis Templates

Summary

Routine tasks can be simplified by creating an Analysis Template. Such templates can contain multiple analysis results and also custom report sheets. A new instance of the template can then be opened any time and source data can be changed to update all analysis results and custom reports.

Minimum Origin Version Required: Origin 8.0 SR6

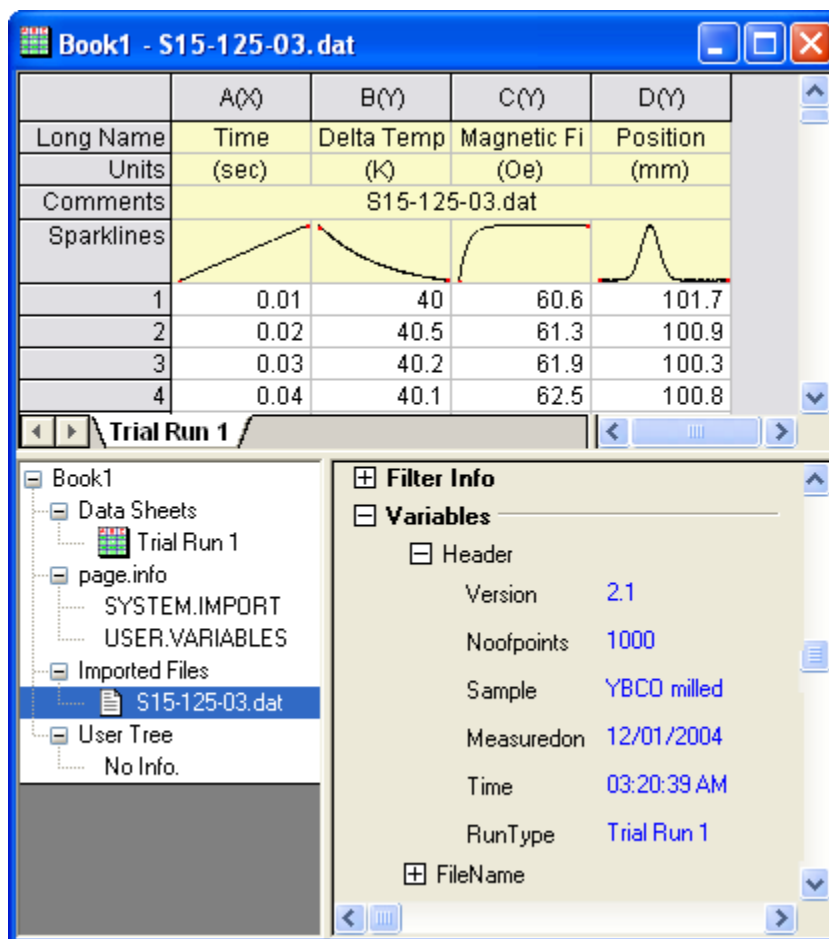
What you will learn

- How to create analysis template (OGW)
- How to re-use analysis template with new data

Steps

Importing Data

1. Start with a new workbook.
2. Click the **Import Wizard** button on the Standard Toolbar, or invoke the **Import: Import Wizard...** menu item. the Import Wizard dialog will open.
3. Click the ... button next to the **File** edit box and select the file <Origin EXE folder>\Samples\Import and Export\S15-125-03.dat.
4. Verify that the **Import Filters for Current Data Type** drop-down shows **Data Folder: VarFromFileNameAndHeader**.
5. Change the **Import Mode** drop-down to **Replace Existing Data**.
6. This is an import filter shipped with the sample file, that specifies how to import the file and what header and file name strings to parse to create import variables. Walk through the wizard pages to view the settings (Optional) and then click Finish button to import the file.
7. Right-click on workbook title bar and select **Show Organizer** to turn on organizer panel. Expand branches and verify that variables have been created and saved, as in the picture below:



Performing Analysis

1. Highlight column D and use the **Analysis: Fitting: Nonlinear Curve Fit...** menu item to open the **NLFit** dialog.
2. Fit the data with **Gauss** function. This will add a hierarchical report sheet to the book, with result tables and embedded graph with data and fit curve.
3. Go to the **FitNL1** report sheet and double-click to open the graph containing data and fit curve. Perform some customization of the graph such as adding grid lines, changing font size etc. Click the **X** button on the graph window to put the modified graph back into the report.

Saving the Analysis Template

1. Go to the source data sheet of the workbook, which should be the first sheet. Select the **Worksheet: Clear Worksheet...** menu item and press **OK** in the dialog that opens. This will clear all the data from the sheet. The analysis report sheet will now be empty. Clearing the data is optional, and it makes the size of the analysis template file to be smaller.
2. Use the **File: Save Window As...** menu item and save the book as an **OGW** file under your **User Files Folder** with a suitable name such as **Analysis Template**. This OGW file can now serve as an Analysis Template for future analysis of similar data.

Re-using the Analysis Template

1. Start a new project and then select the menu item **File: Recent Books** and from the fly-out options select the Analysis Template saved earlier.
2. Make the data sheet active, and select **File: Import Wizard...** and select the file <Origin EXE path>\Samples\Import and Export\S21-235-07.dat.
3. Make sure the filter drop-down shows **VarsFromFileNameAndHeader** and change the **Import Mode** drop-down to **Replace Existing Data** and click **Finish**.
4. Press the **Recalculate** button, which is the last button on the **Standard** toolbar. Origin will recalculate the analysis results and update the custom report sheet links, and at this point you can view and print the custom report sheet.

1.4.2 Creating Analysis Templates using Set Column Value

Summary

This tutorial will demonstrate how to add a column, set up **Before Formula Script** and have that script run whenever data changes in other columns. This technique can be used to create an **Analysis Template** for repeated analysis of similar data.

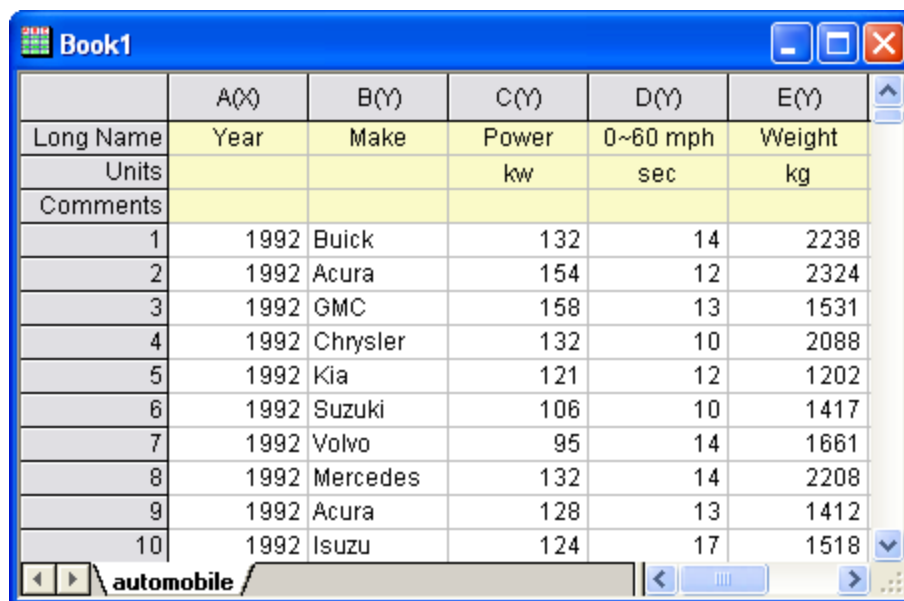
Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

- How to use Set Column Values to create an analysis template
- How to select rows via the *Go to* function

Steps

1. Import the data from `\Samples\Statistics\automobile.dat` into a newly created workbook, as below. In this example, we will extract data, according to the *Make* column, into different worksheets.



	A(Y)	B(Y)	C(Y)	D(Y)	E(Y)
Long Name	Year	Make	Power	0~60 mph	Weight
Units			kw	sec	kg
Comments					
1	1992	Buick	132	14	2238
2	1992	Acura	154	12	2324
3	1992	GMC	158	13	1531
4	1992	Chrysler	132	10	2088
5	1992	Kia	121	12	1202
6	1992	Suzuki	106	10	1417
7	1992	Volvo	95	14	1661
8	1992	Mercedes	132	14	2208
9	1992	Acura	128	13	1412
10	1992	Isuzu	124	17	1518

2. Add an empty column to the worksheet and bring up the **Set Column Values** dialog of the column. In the **Before Formula Script** box, enter the script below.

```
// Data range on which to perform discrete frequency
count
range makeCol = !col(make);
// Worksheet to be extracted
range sourceWks = !;

// Clear worksheets
int sheetNum = page.nlayers;
```

```

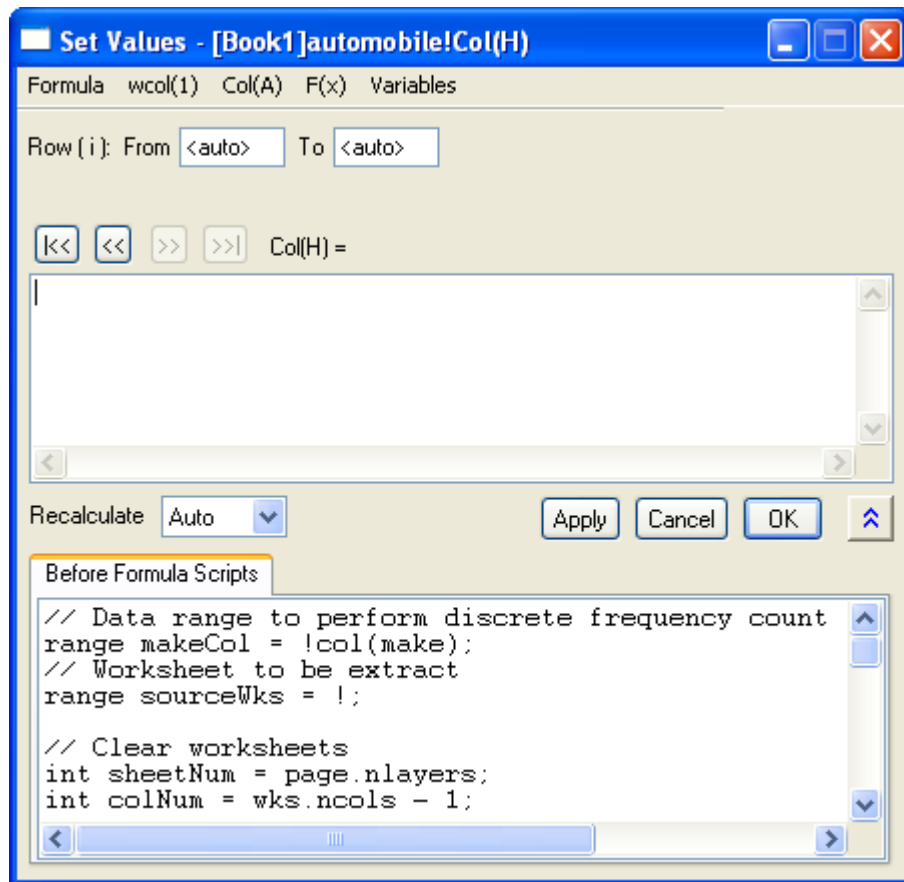
int colNum = wks.ncols - 1;
if (sheetNum>1)
{
    for (jj=2; jj<=sheetNum; jj++)
    {
        layer -d 2;
    }
}

// Tree variable to hold discfreqs outputs
tree tr;
// Perform discrete frequency count
discfreqs irng:=makeCol rd:=tr;
// String array to get result from tree
StringArray sa;
sa.append(tr.FreqCount1.Data1);

if( sa.GetSize() != NANUM )
{
    // Loop to extract data
    for (ii=1; ii<=sa.GetSize(); ii++)
    {
        string sn$ = sa.GetAt(ii)$;
        // Extract condition string
        string cond$ = "makeCol$ = " + sn$;
        // Create worksheet with different Make name
        newsheet name:=sn$ cols:=colNum outname:=on$
active:=0;
        // Extract data
        wxt test:=cond$ iw:=sourceWks c2:=colNum
ow:=on$;
    }
}

```

3. This script will first perform a discrete frequency count on the *Make* column to get distinct values for *Make*. It will then create a new worksheet for each brand and extract data into these sheets.



4. Make sure the recalculate mode is set to **Auto** and click **OK**. The data will be separated into different worksheets. Then the empty column (H) with a green lock icon indicates that this procedure can be updated automatically.

	E(Y)	F(Y)	G(Y)	H(Y)
Long Name	Weight	Gas Mileage	Engine Displacement	
Units	kg	mpg	cc	
Comments				
1	2238	11	5736.5	
2	2324	11	5212	
3	1531	10	5900.4	
4	2088	12	6277.4	
5	1202	12	5736.5	
6	1417	14	5736.5	
7	1661	13	5031.7	
8	2208	12	5736.5	
9	1412	12	5736.5	
10	1518	13	5900.4	

5. There are 18 makes of cars in the source data, so 18 new worksheets were created. Now we can check whether auto-update works.

Go to the **Honda** tab. Notice that there are 19 rows.

6. Switch to the first sheet. Scroll down and delete the last row.

	A(X)	B(Y)	C(Y)	D(Y)	E(Y)	F(Y)	G(Y)	H(Y)
Long Name	Year	Make	Power	0~60 mph	Weight	Gas Milea	Engine Di	
Units			kw	sec	kg	mpg	cc	
Comments								
Sparklines								
338	2004	Infiniti	38	12	867	35	1491.5	
339	2004	Isuzu	49	15	848	36	1589.8	
340	2004	Honda	62	10	1210	25	1458.7	
341								
342								
343								
344								
345								
346								
347								

Then auto-update will be triggered. Go to the **Honda** tab. You will see there are only 18 rows.

	A(Y)	B(Y)	C(Y)	D(Y)	E(Y)	F(Y)	G(Y)
Long Name	Year	Make	Power	0~60 mph	Weight	Gas Mileag	Engine Dis
Units			kw	sec	kg	mpg	cc
Comments							
Sparklines							
13	1999	Honda	70	18	1067	24	2196.3
14	2000	Honda	44	13	1210	24	1606.2
15	2001	Honda	64	17	746	27	1589.8
16	2001	Honda	66	14	1847	26	2474.9
17	2002	Honda	51	17	1240	29	1475.1
18	2003	Honda	38	19	1024	31	1245.6
19							
20							
21							
22							
23							

Navigation tabs: Volvo, Toyota, Saturn, Nissan, Chrysler, **Honda**

1.4.3 Creating a Custom Report Sheet

Summary

Worksheets in Origin can be customized by merging cells and placing various objects such as graphs, external images, links to variables and tables/cells in other sheets, in order to create custom reports. Such custom reports can be part of an analysis template, thus allowing user to open the analysis template, change data, and simply print their updated custom report.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

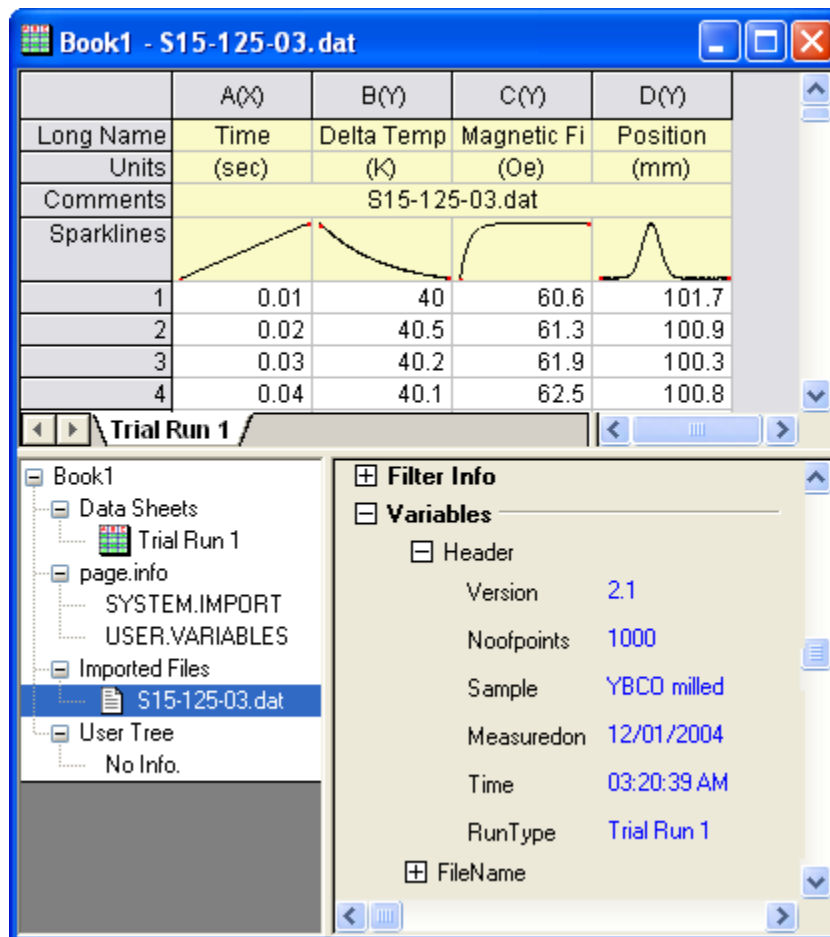
- How to create a custom report sheet
- How to save custom report as part of Analysis Template (OGW) and re-use with new data

Steps

Note: First finish the previous tutorial named "Creating and Using Analysis Templates" where an analysis template named **Analysis Template.OGW** is created.

Importing Data

1. Use the **File :Open** menu item and open the Analysis Template **Analysis Template.OGW**. This analysis template already has a nonlinear fitting analysis operation set up for data from column D of the first sheet.
2. Click the **Import Wizard** button on the Standard Toolbar, or invoke the **Import: Import Wizard...** menu item. the Import Wizard dialog will open.
3. Click the ... button next to the **File** edit box and select the file **<Origin EXE folder>\Samples\Import and Export\S15-125-03.dat**.
4. Verify that the **Import Filters for Current Data Type** drop-down shows "Data Folder: VarFromFileNameAndHeader".
5. Verify that the **Import Mode** drop-down is set to **Replace Existing Data**.
6. This is an import filter shipped with the sample file, that specifies how to import the file and what header and file name strings to parse to create import variables. Walk thru the wizard pages to view the settings (Optional) and then click Finish button to import the file.
7. Right-click on workbook title bar and select **Show Organizer** to turn on organizer panel. Expand branches and verify that variables have been created and saved, as in the picture below:

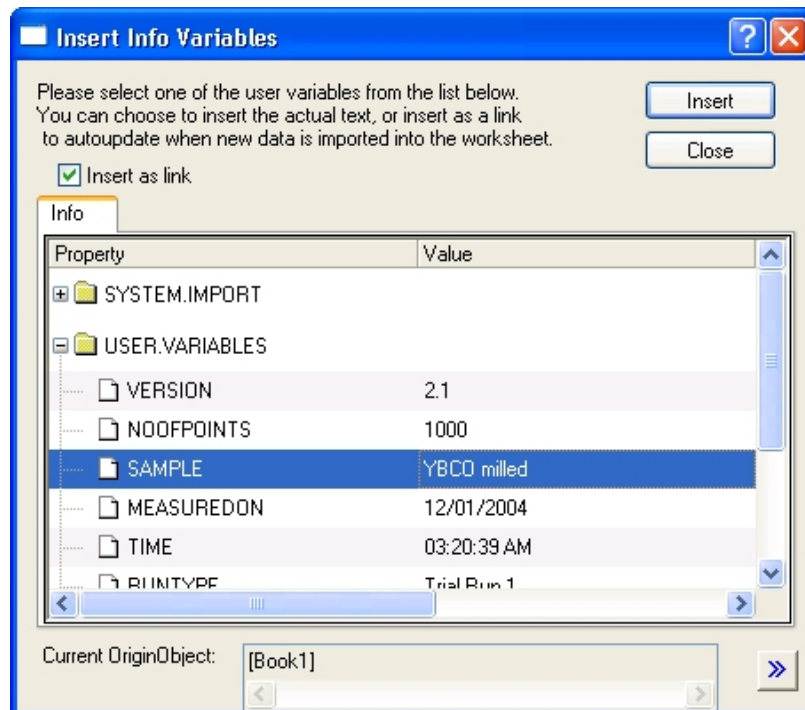


8. Press the **Recalculate** button on the **Standard** toolbar to update the analysis result sheet. Verify that the analysis was updated and the embedded graph shows the new raw data and fit curve.

Creating Custom Report Sheet

1. Right click on one of the worksheet tabs and select **Add** to add a new worksheet. Rename this worksheet as **Custom Report**.
2. Make the **Custom Report** sheet active and add multiple blank columns.
3. Go to **FitNL1** worksheet and right click on the graph with data and fit curve and select **Copy** from the context menu.
4. Go to **Custom Report** and right click in 1st column in a middle row and select **Paste Link**. This will paste a link to the embedded graph. Click and select a group of cells with this pasted cell at the top-left. Then click the **Merge Cells** button, which is the last button on the **Styles** toolbar. This will merge the group of cells and the graph image will be shown larger in size.
5. Go to **FitNL1** report sheet and right click on the **Parameters** node and select **Copy Table** from the context menu.
6. Go to the **Custom Report** sheet and right click in a cell to the right of the graph and select **Paste Link** from the context menu. This will place links to all values of the parameter table entries in the custom report.

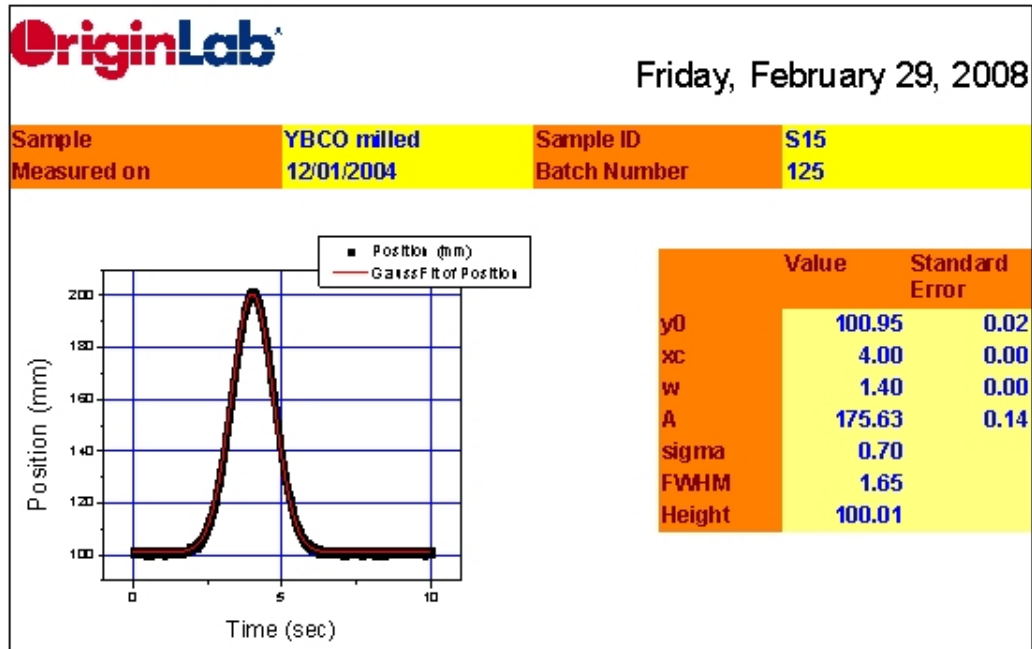
7. Select the numeric value cells and right-click and select **Format Cells...** to bring up format dialog. Change the **Digits** drop-down to **Set Decimal Places=** and enter **2** in the **Decimal Number** edit box and click **OK** to format the numbers.
8. Select various cells in the table and use the **Style** toolbar controls to change foreground and background color, and use the **Standard** toolbar to change font size etc.
9. Right-click on top-left cell in the custom report sheet and select **Insert Images from Files...** context menu and select some image such as a company logo image. Click and drag to cover more cells and then click the **Merge Cells** button to increase the size of the logo display.
10. Click inside a cell on top-right and type in the string **var://@D** and press **Enter**. Right click on cell and select **Format Cells...**, set the **Format as Date**, and then elect a suitable format from the **Display** drop-down. This will place the current date, pointed to by @D LabTalk variable, into the worksheet cell. Click and expand the selection to multiple cells and press the **Merge Cells** button to show the date with larger font size.
11. Right-click on a cell below the logo and date, and select **Insert Variables** context menu. In the dialog that opens, select **User.Variables** branch and select **Sample**. Check the **Insert as Link** check box on top and press **Insert** to insert variable as link into the report sheet.



Click on a cell to the left of the inserted variable, and enter the static text **Sample**.

12. Insert more variables and format the cells for color and font.
13. Invoke the **Format: Worksheet...** menu item to open the **Worksheet Properties** dialog. Under the **View** tab, expand **Show Headers** and uncheck the column and row header check boxes. Expand the **show Grid Lines** branch and uncheck the column and row grid check boxes. Select the **Format** tab and check the **Show Missing as Blank** check box. Click **OK** to close this dialog.

- Right-click on worksheet title bar and select **View: Long Name** to turn off long name row. Also turn off **Units** and **Comments**.
- Select the **File: Print** menu item to open the print dialog, and press **Options** button, and uncheck the Horizontal/Vertical grid lines. Select **File: Print Preview**. Your custom report sheet should look like the image below:



Saving the Analysis Template

- Go to the source data sheet of the workbook, which should be the first sheet. Select the **Worksheet: Clear Worksheet...** menu item and press **OK** in the dialog that opens. This will clear all the data from the sheet. The analysis report sheet and the custom report sheet will now be empty. Clearing the data is optional, and it makes the size of the analysis template file to be smaller.
- Use the **File: Save Window As...** menu item and save the book as an **OGW** file under your **User Files Folder** with a suitable name such as **My Custom Analysis**. This OGW file can now serve as an Analysis Template for future analysis of similar data.

Re-using the Analysis Template

- Start a new project and then select the menu item **File: Recent Books** and from the fly-out options select the Analysis Template saved earlier.
- Make the data sheet active, and select **File: Import Wizard...** and select the file **<Origin EXE path>\Samples\Import and Export\S21-235-07.dat**.

3. Make sure the filter drop-down shows **VarsFromFileNameAndHeader** and change the **Import Mode** drop-down to **Replace Existing Data** and click **Finish**.
4. Press the **Recalculate** button, which is the last button on the **Standard** toolbar. Origin will recalculate the analysis results and update the custom report sheet links, and at this point you can view and print the custom report sheet.

1.5 Analysis Themes

1.5.1 Summary

In Origin 8, analysis procedures can be controlled by Themes. Themes are actually XML files which save settings in the analysis dialog. For example, after performing the analysis, there will now be a <Last Used> theme for this dialog which has saved the most recently used settings. You can assign a proper name for the theme and use it in the future.

For this tutorial, the Statistics on Columns dialog will be used to demonstrate how to create and use an analysis theme. This analysis provides descriptive statistics about the data such as mean, standard deviation, minimum, maximum, and more. For visualization, a histogram or box chart can also be created in the Analysis Result Sheet.

Minimum Origin Version Required: Origin 8.0 SR6

1.5.2 What you will learn

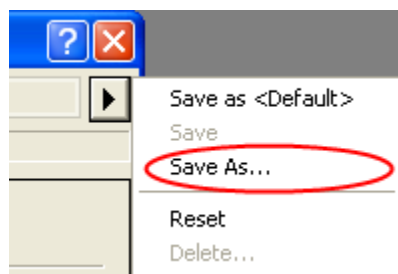
This tutorial will show you how to:

- Perform simple descriptive statistics
- Create an Analysis Theme
- Use the theme

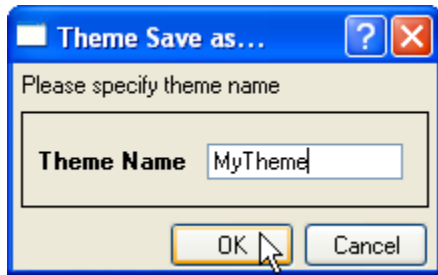
1.5.3 Steps

Save the analysis procedure as Theme

1. Start with a new workbook and import the file `\Samples\Statistics\automobile.dat`.
2. Highlight column C and use the menu item **Statistics: Descriptive Statistics: Statistics on Columns** to open the dialog.
3. Expand the **Moments** tree node, and check the *N Total*, *Mean*, *Standard Deviation*, *SE of Mean* and *Sum* box.
4. Expand the **Plots** tree node, and check the *Histograms* and *Box Charts* check boxes. You will then get the corresponding histogram as well as box chart graphs.
5. Your selections in this analysis dialog can be saved as your theme, so that you may easily repeat the procedure. Click the "Save Theme as..." button:



to bring up this dialog:



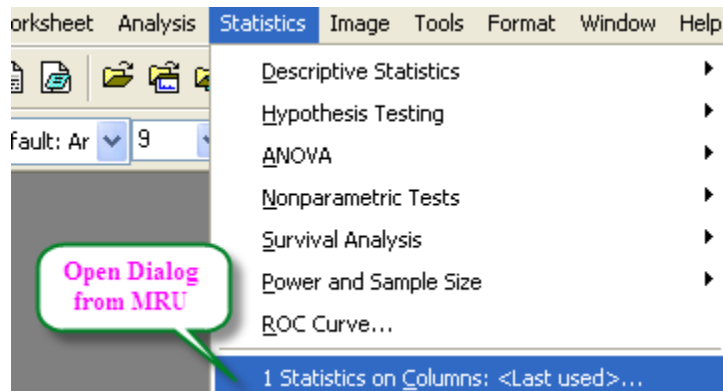
6. Type a proper theme name, such as "*MyTheme*" and click **OK** button.
7. Click the **OK** button in the **Statistics on Columns** dialog. You will see the result in a new worksheet named DescStatsOnCols1.

Descriptive Statistics					
	N total	Mean	Standard Deviation	SE of mean	Sum
Power	340	79.85	28.07561	1.52261	27149

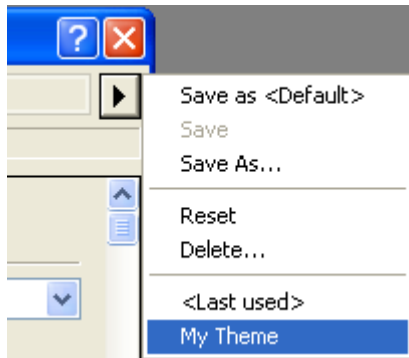
Repeat the analysis procedure by Theme

Once you save a theme, there are many ways to use it. For example, you can highlight column E and perform the same statistics on it.

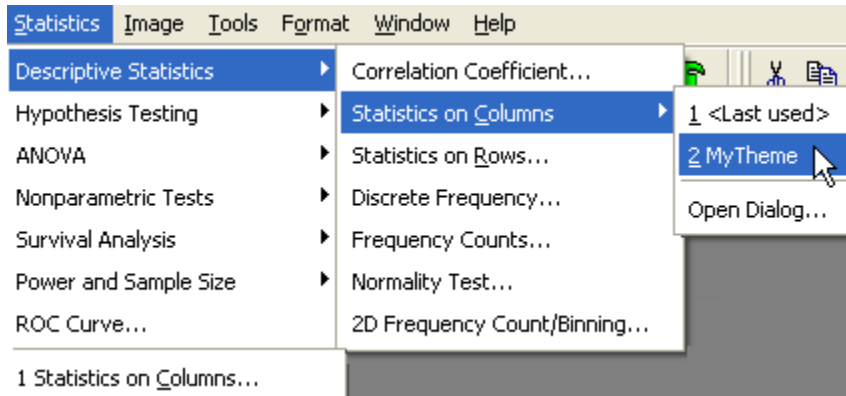
- Open the **Statistics on Columns** dialog from the **Most Recently Used** menu. Most of the menu accessible dialogs can be found from MRU.



When you open the dialog from MRU, the default theme is <Last Used>. To use the **MyTheme** theme, select *MyTheme* from the **Dialog Theme** drop-down list. The settings from that theme will then be displayed in the dialog. Click the **OK** button to do the analysis.



- Another way to apply the analysis theme is to use the cascaded menu item. Once you use an analysis dialog, or save a theme for a dialog, there will be one more menu level added. You can choose the *MyTheme* menu.



When selecting, *Open Dialog...*, the dialog will open to the <Factory Default> theme. To change the settings of your theme, you can select your theme name from the **Dialog Theme** drop-down list inside the dialog, make changes, and resave the theme. Alternatively, to open a dialog with a saved theme without performing the analysis, hold down the *shift* key while selecting the theme from the menu. This will bring up the dialog with your theme applied so you can make changes as needed.

1.6 Batch Processing

1.6.1 Analysis Template and Batch Processing

Summary

Origin version 8.5 allows users to easily perform batch processing of multiple files or data sets, using an Analysis Template. This tutorial will show you how to use this new functionality to batch process multiple files.

This tutorial will show you how to:

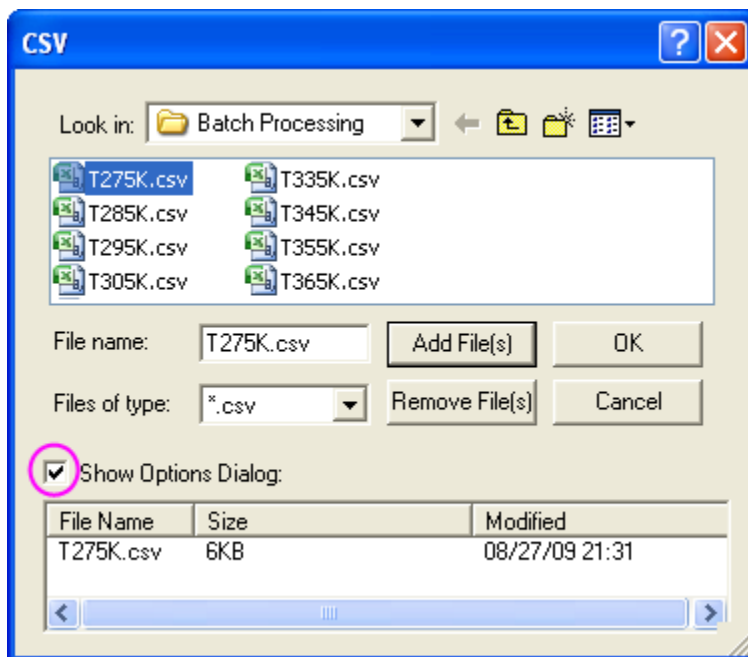
- Create an Analysis Template
- Use the Analysis Template to batch process multiple data files

Creating an Analysis Template

The first task is to create an Analysis Template for processing a single file/data set. Once the Analysis Template has been created, it can be used by the Batch Processing tool to process multiple files/data sets.

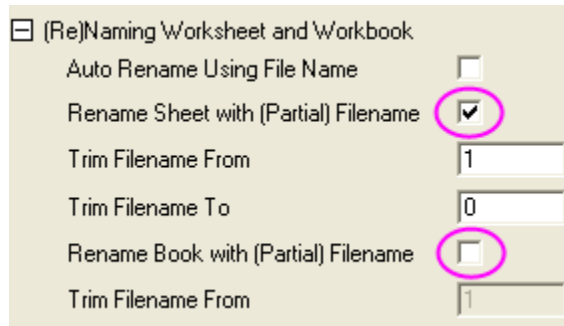
The following steps demonstrate how to create an Analysis Template:

1. Start with a new workbook.
2. Use the **File: Import: Comma Delimited (CSV)** menu item, and in the file dialog that opens, navigate to, and select, the file `\Samples\Batch Processing\t275k.csv`, and press the **Add File(s)** button. Then check the **Show Options Dialog** check box and press **OK**.

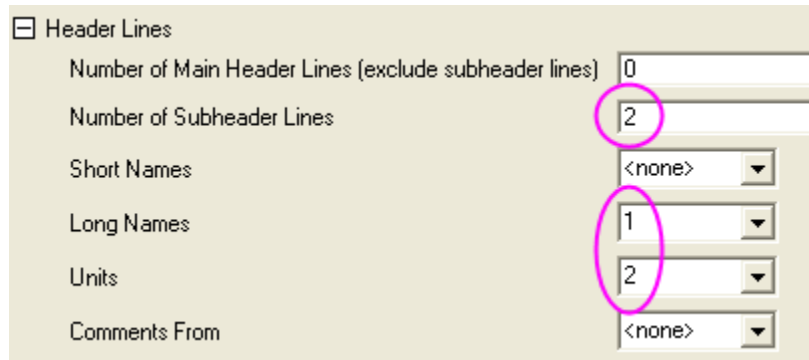


The **impCSV** dialog now opens, which can be used to specify custom settings for importing the selected file.

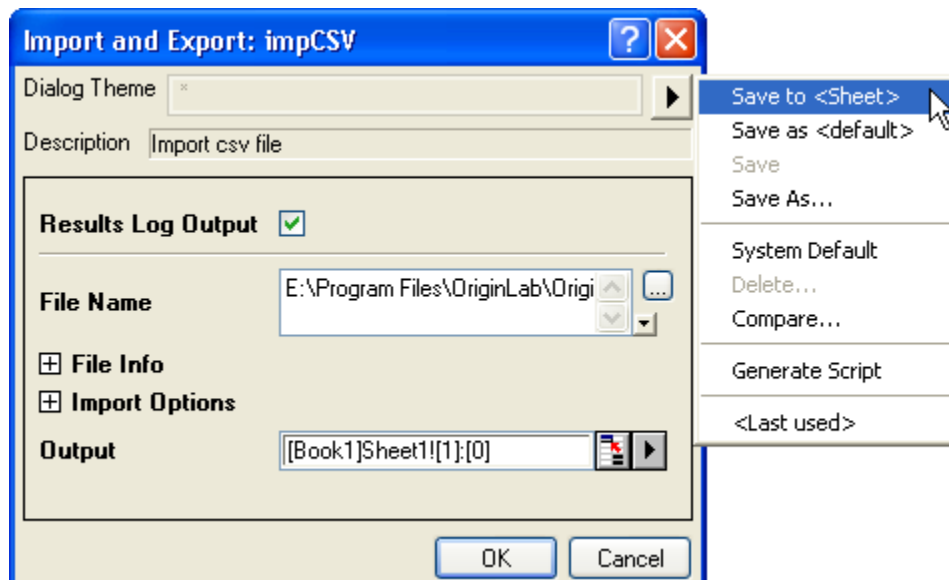
- Expand the **Import Options** node and then the **(Re)Naming Worksheet and Workbook** subnode, and change the settings as below, so that only the sheet gets renamed by file name, not the book.



- Our data file has two sub header lines suitable for Long Name and Units. Expand the **Header Lines** node and set Number of Subheader Lines to 2, and set Long Names as 1 and Units as 2.



- Go to the Dialog Theme control in the top of the dialog, press the arrow button, and select **Save to <Sheet>** from the pop-up menu. This saves the custom settings right into the worksheet, so that the settings will become part of the Analysis Template and will be used for importing similar files when the template is used later by the Batch Processing tool.



6. Press **OK** and the file gets imported

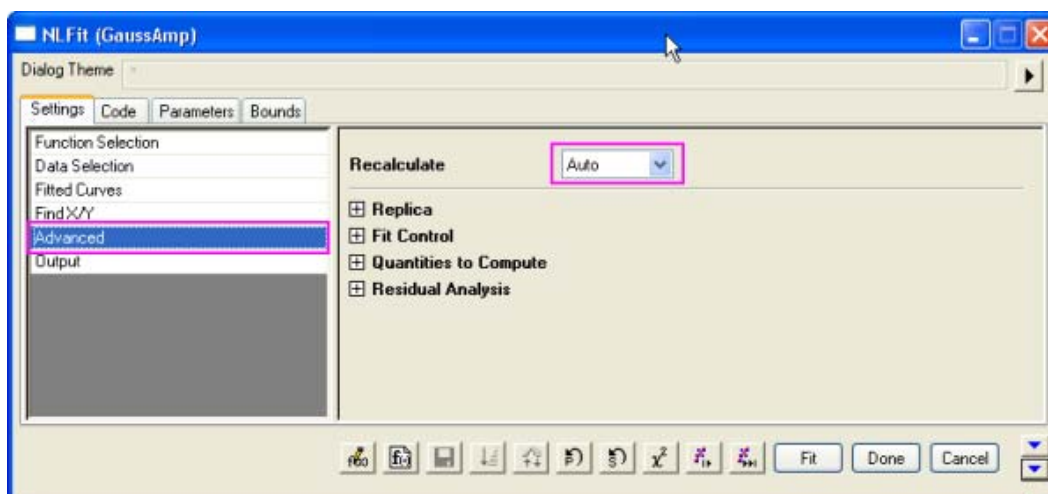
	A(X)	B(Y)
Long Name	Wavelength	Amplitude
Units	nm	mV
Comments		
1	835	2.406
2	836	-5.717
3	837	6.53
4	838	-8.223
5	839	0.794
6	840	-9.049
7	841	6.663
8	842	4.172



Import dialogs, such as ASCII import (impASC), CSV Import (impCSV), or any of the third-party imports, including ETAS INCA MDF (impMDF), all allow saving custom settings to the <sheet>. If you use the Import Wizard for importing the file, the custom settings can also be saved into the worksheet as a filter, in the Save Filters page of the wizard. Saving the import settings or the import wizard filter into the worksheet allows for all information to be contained in the analysis template itself, making it portable. There is then no need for additional files, in case you are sharing the analysis template with another user.

The next step in creating the Analysis Template is to add desired analysis operations based on the imported data. In our example we will perform curve fitting on the imported data.

7. Highlight Column **B** and select **Analysis: Fitting: Nonlinear Curve Fit** to bring up the dialog. Select **GaussAmp** from the **Function** drop-down list. Click **Advanced** to the left and then select **Auto** from the **Recalculate** drop-down list, as shown in the following image. Click the **Fit** button to complete the fitting process.



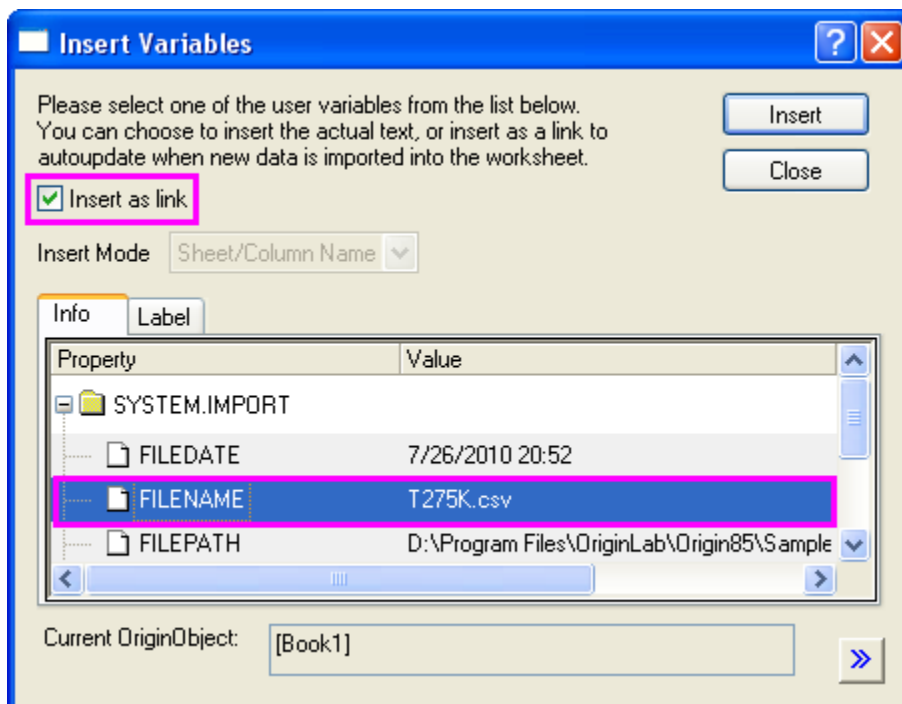
8. Rename the first worksheet **Raw Data**, so that the template has a suitable and recognizable sheet name.

The next important step in creating the Analysis Template is to add a **Result Sheet**, into which you can place desired information from the import and the analysis. This sheet will then be used by the Batch Processing tool to create a summary report from multiple files/data sets. Note that this step is optional; you must add such a sheet only if you need to obtain a summary report at the end of the batch processing operation.

9. Add a new worksheet named **My Peaks** into the current workbook, and add three columns. Set the long names of the columns as shown in the image below. This sheet will be set up by paste-linking desired quantities from the current analysis. The Batch Processing dialog will then create a new report book with this information appended for each data file.



10. Right-click on the **first cell** of Column **A** of the worksheet **My Peaks** and select **Insert Variables**. Select **FILENAME** to insert. Make sure the **Insert as link** check box is checked.



11. Go to the **FitNL1** sheet to copy the values of parameter **xc** and then go to the **My Peaks** worksheet. Right-click on the first cell of Column **B** and select **Paste Link**. Then **Copy** relevant values from the **Parameters** table (parameters **A** and **W**) and from the **Statistics** table (**Adj. R²**), to **Paste Link** them to Columns **C**, **D** and **E**.

T275K - T275K.csv

1 Nonlinear Curve Fit (GaussAmp) (2009-8-7 15:54:59)

Notes

Input Data

Parameters

	Value	Standard Error
y0	-0.32072	0.24543
xc	915.09771	0.02384
w	20.52677	0.02489
A	994.6433	1.01537
FWHM	48.33685	
Area	51177.36939	

Reduced Chi-sqr = 24.2737263993
 COD(R²) = 0.99960082457751
 Iterations Performed = 4
 Total Iterations in Session = 4
 Fit converged - tolerance criterion satisfied.
 FWHM, Area are derived parameter(s).

Statistics

	Amplitude
Number of Points	512
Degrees of Freedom	508
Reduced Chi-Sqr	24.27373
Residual Sum of Squares	12331.05301
Adj. R-Square	0.9996
Fit Status	Succeeded(100)

Fit Status Code :
 100 : Fit converged

Raw Data | **FitNL1** | FitNLCurve1 | My Peaks

12. Select **File: Save Workbook as Analysis Template** and name it **My Peak Analysis**. The raw data and related results will be cleared and the book will be saved to the disk as an **OGW** file with the specified name.

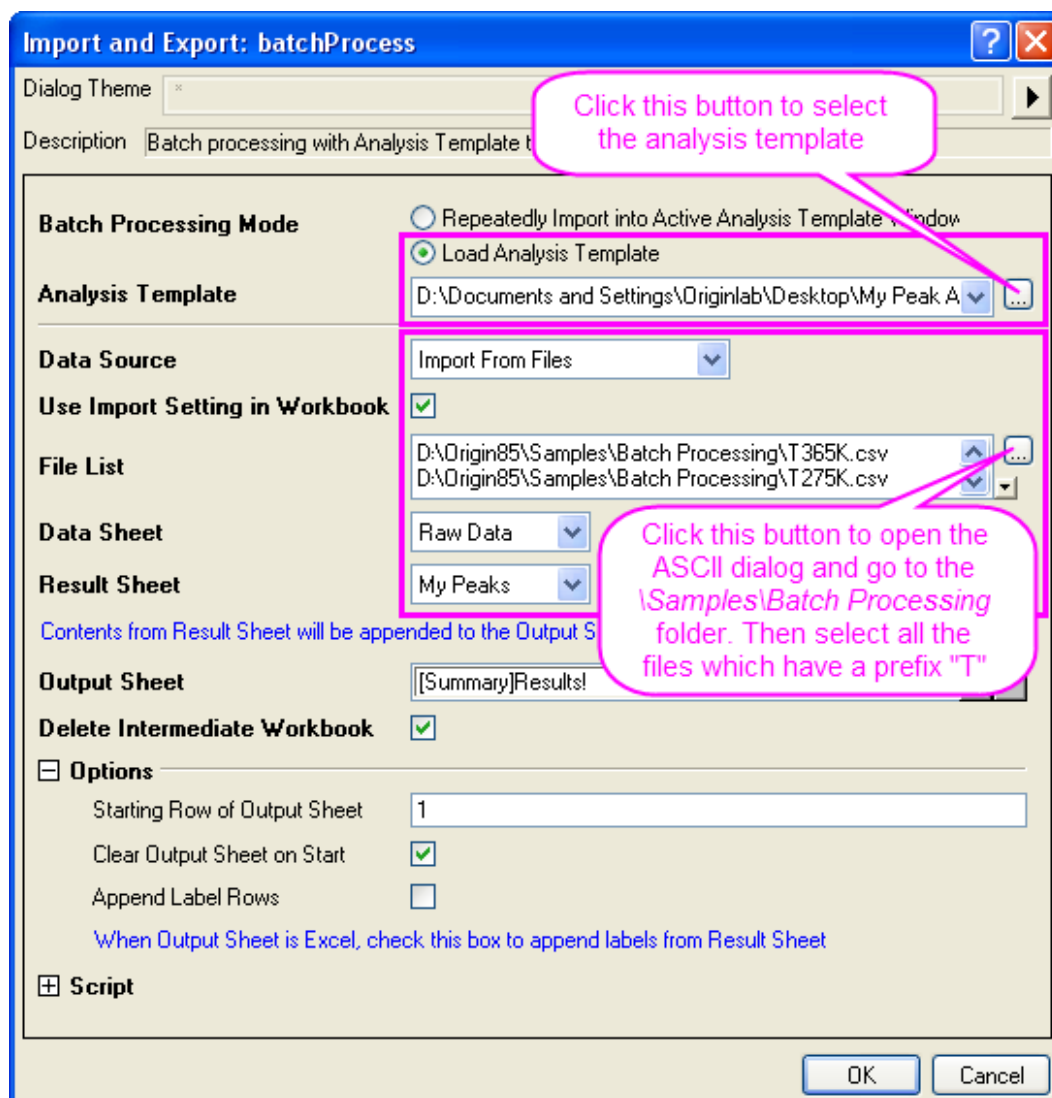


Analysis Templates can also include custom sheets other than a summary sheet. For instance, the user can create a custom sheet which combines multiple elements such as graphs and result tables in a format suitable for presentation and reporting. A graph can be inserted into a cell and then multiple cells in a sheet can be merged with this cell, to create a larger display of the graph image, for instance. Please view other tutorials available from the Help menu for information on how to create custom reports with merged cells.

Performing Batch Processing

We will now use the Analysis Template created in the steps above to perform batch processing of data from ten similar data files.

1. Create a new project and select **File: Batch Processing** from the main menu. Set the controls in the dialog as shown in the image below. Please note that the data files are under the *Origin Installation Folder\Samples\Batch Processing* folder, and all of them have the same prefix, T. Please remember to select **All Files(*.*)** from the **Files of type** drop-down list after you've opened the ASCII dialog by clicking the browser button, which is next to the **File list**.



2. Click **OK** to perform Batch Processing. The summary report should look like the image below. Sparklines were turned on for the data columns to visually compare the fit results from the different data files.

	A(X)	B(Y)	C(Y)	D(Y)	E(Y)
Long Name	File Name	Peak Center	Peak Heigh	Peak Width	Adj. R^2
Sparklines					
1	T275K.csv	915.09771	994.6433	20.52677	0.9996
2	T285K.csv	945.89011	980.77809	21.67798	0.99965
3	T295K.csv	977.5276	959.81779	22.98456	0.99963
4	T305K.csv	1009.93406	937.70088	24.51295	0.99962
5	T315K.csv	1043.1428	908.6266	26.08323	0.99957
6	T325K.csv	1077.12324	871.31437	27.94735	0.99958
7	T335K.csv	1111.88461	834.49884	29.87287	0.99956
8	T345K.csv	1147.4659	791.40866	31.9675	0.99954
9	T355K.csv	1183.85578	741.19166	34.28623	0.99951
10	T365K.csv	1221.05614	689.68533	36.70036	0.99943

1.6.2 Batch Processing with Summary Report in External Excel File

Summary


This tutorial is associated with the Sample Origin Project: **\Samples\Batch Processing\Batch Processing with Summary Report in External Excel File.OPJ**

Minimum Origin Version Required: Origin 8.1 SR2

What you will learn

- How to perform batch processing of multiple data files
- How to send results to an external Excel file and save that file

Steps

1. Make the 'Raw Data' Worksheet in the 'Book1' Workbook active.
2. Select **File : Batch Processing...** from the menu or click the **Batch Processing** button  on the Standard Toolbar.
3. Select the **Repeatedly Import into Active Analysis Template Window** radio.
4. Set **Data Source** as **Import From Files**.
5. Check the **Use Import Setting in Workbook** check box.
6. Click the ... browse button next to **File List**.
7. Select **All Files (*.*)** in Files of Type and browse to Origin's \Samples\Batch Processing folder.
8. Select all 10 .csv files in the folder, click **Add File(s)** then click **OK**.
9. **Data Sheet** should be set to 'Raw Data'. Note: 'Raw Data' is the first sheet of 'Analysis Template', it will be renamed as the file name of last import file by default if this template has already been used and re-saved.
10. **Result Sheet** should be set to 'My Results'.
11. Click the button to the right of the "Output Sheet" edit box. *This will minimize the main dialog.* Then click on the title bar of the Excel book (Book2), and press the button to the right of the scrolled up dialog to expand it again.
12. Expand the Options branch and check the **Clear Output Sheet on Start** check box and enter **7** as the **Starting Row of Output Sheet**.
13. Check the **Append Label Rows** check box.
14. Click OK

2 Statistics

2.1 Descriptive Statistics

Summary

Origin provides comprehensive Descriptive Statistics support including basic statistics (mean, median, variance, etc.), frequency counts, and correlation coefficients of data you select. In addition to strong plotting features, Origin's statistical tools help you summarize and analyze your data.


This tutorial will show you how to:

- Use the **Statistics on Column** Dialog to calculate descriptive statistics for grouped data.
- Copy statistical results to a new worksheet for further processing.
- Unstack Columns to a graph.
- Analyze data sets with the **Correlation Coefficient** Tool.

Finding Frequency Information for Groups

Start with some data. We can use the **Discrete Frequency** Tool to quickly obtain frequency information for groups of data.

1. Start with a new project or a new workbook. Import the data file

`\\Samples\Statistics\automobile.dat` by using **Import Single ASCII** 



2. Highlight the first two columns. Select **Statistics: Descriptive Statistics: Discrete Frequency** to open a dialog. Column A and Column B are automatically picked as Input Data. Click **OK**

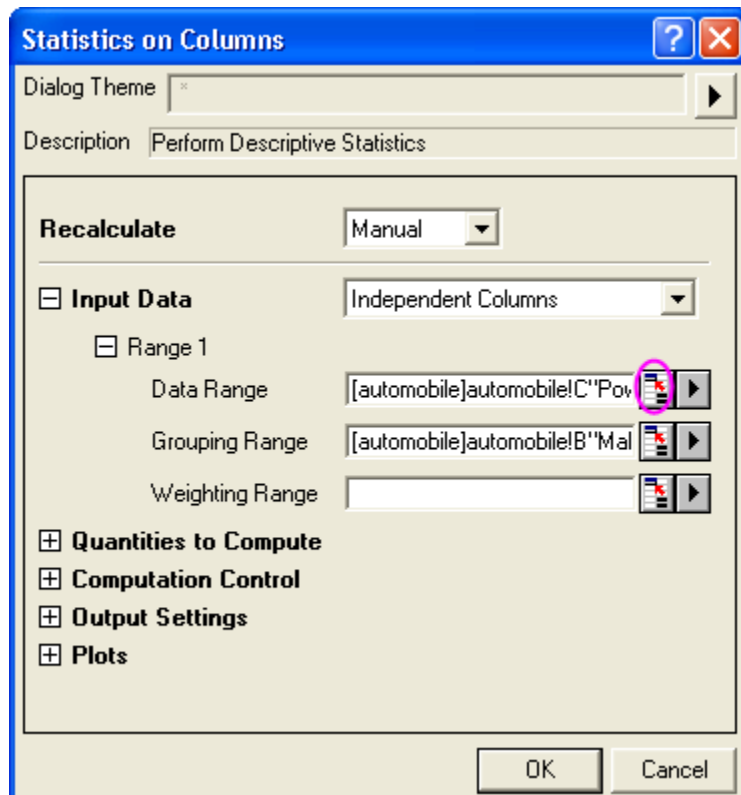


Results of discrete frequency are sorted in descending order of **Count**; the most frequently occurring data will appear first. You can rearrange the results by sorting worksheets even though there are locks on the columns.

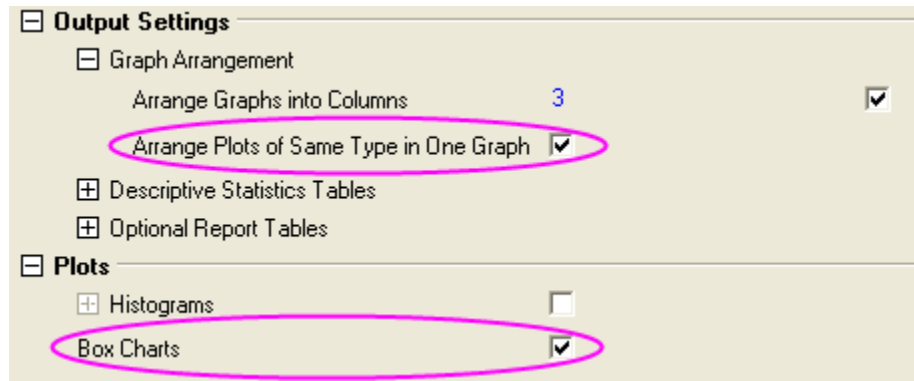
Calculating Descriptive Statistics on Grouped Data

Using the **Statistics on Columns** tool, we can find basic statistics for each group of data.

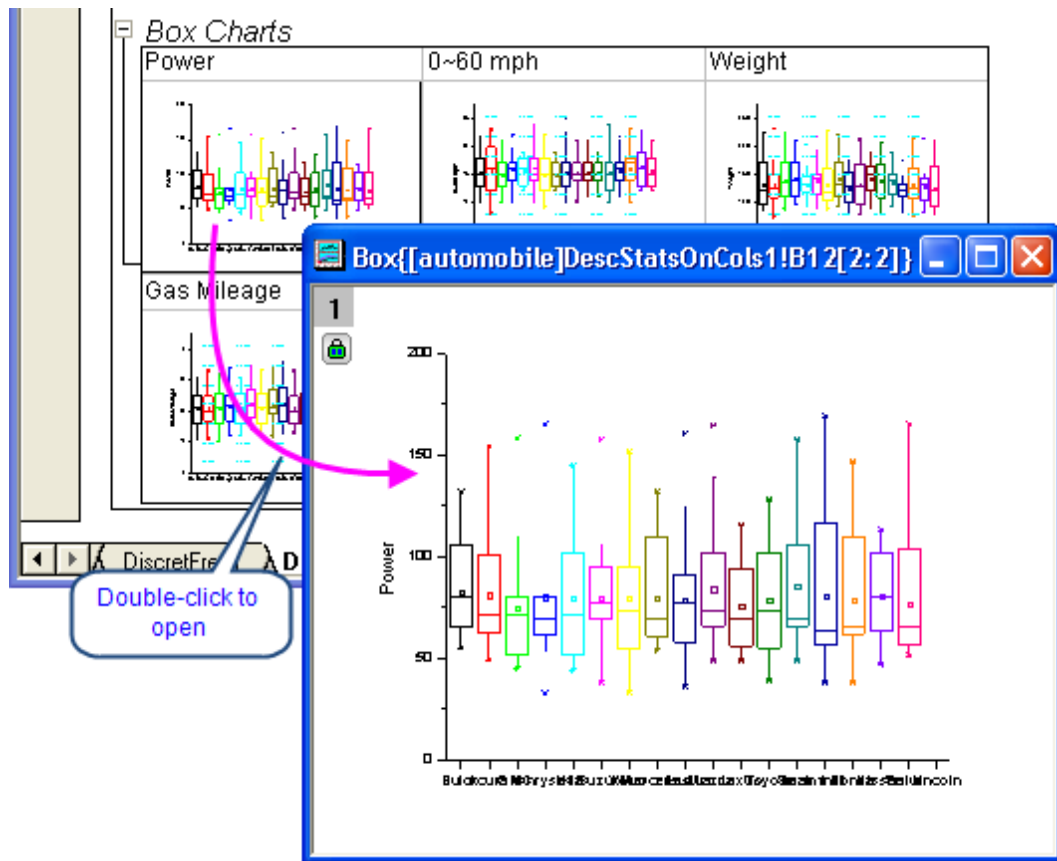
1. Switch back to the first sheet.
2. Select **Statistics: Descriptive Statistics: Statistics on Columns** to open the **Statistics on Columns** dialog.
3. Open the **Range 1** branch and click the **interactive button** . The dialog will "roll up" and you can set **Data Range** as Column C ~ Column G by selecting C(Y) and dragging to G(Y) in the Worksheet. Click the button in the rolled up dialog to restore the dialog. To set **Group Range** to **B(Y): Make**, click the **triangle button**  next to **Grouping Range** and select **B(Y) : Make**.



4. Here, we will show how to make a box plot for the grouped data and put all groups in a graph for a quick comparison. Do the following: 1) Expand the **Output Settings** branch and the **Graph Arrangement** sub-branch. Select the **Arrange Plots of Same Type in One Graph** check box. 2) Expand the **Plots** branch, and select the **Box Charts** check box.



5. Click the **OK** button to get the results in a report sheet.



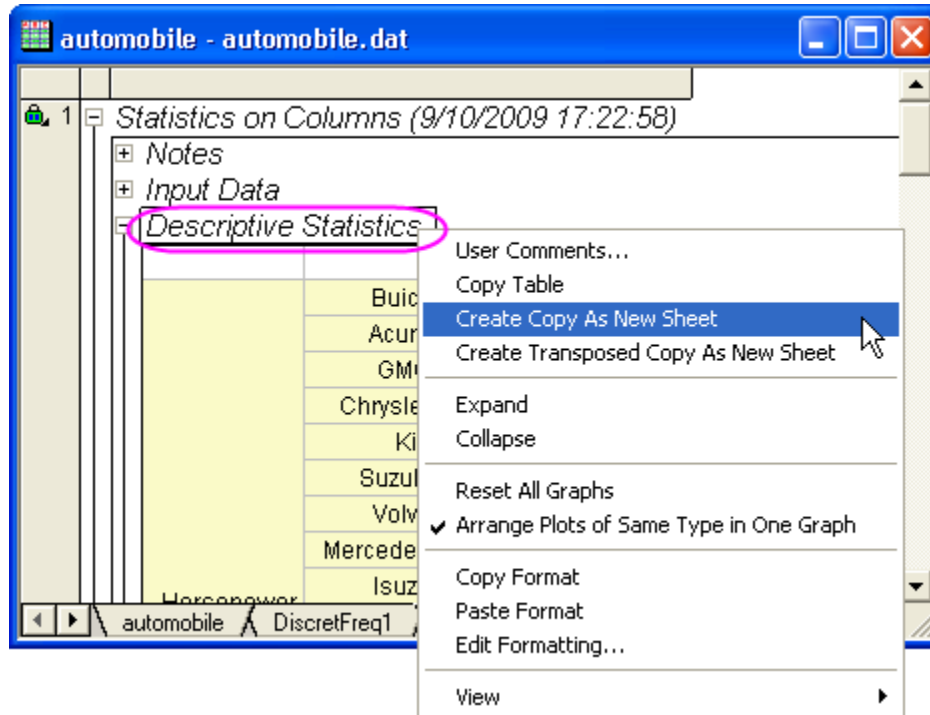
You can double-click to open the graph containing the box plot and customize the graph. Click the **Close** button on the graph to restore the modified graph to the Report Worksheet.

Using Statistical Results for Further Operations

After using the **Statistics on Columns** dialog to produce a report tree, you may wish to do further analysis and plotting on the statistical results.

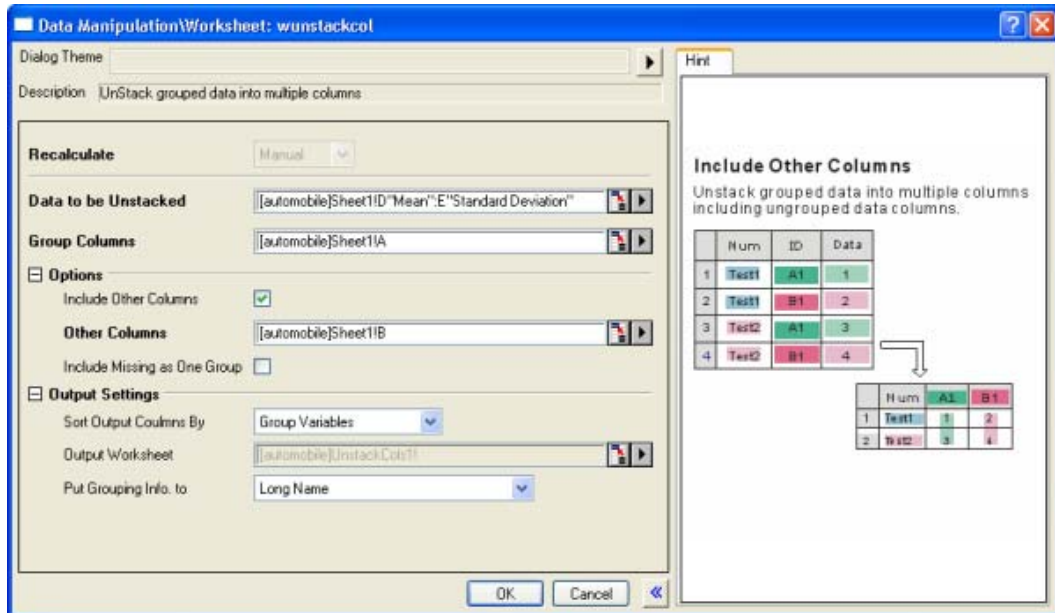
For example, to get average attribute values (i.e. horsepower, 0-60 mph time, weight, mileage) by vehicle Make from 1992 to 2004, perform the following:

1. In the report sheet, right-click on the title of the Descriptive Statistics table and select **Create Copy as New Sheet** from the short-cut menu.



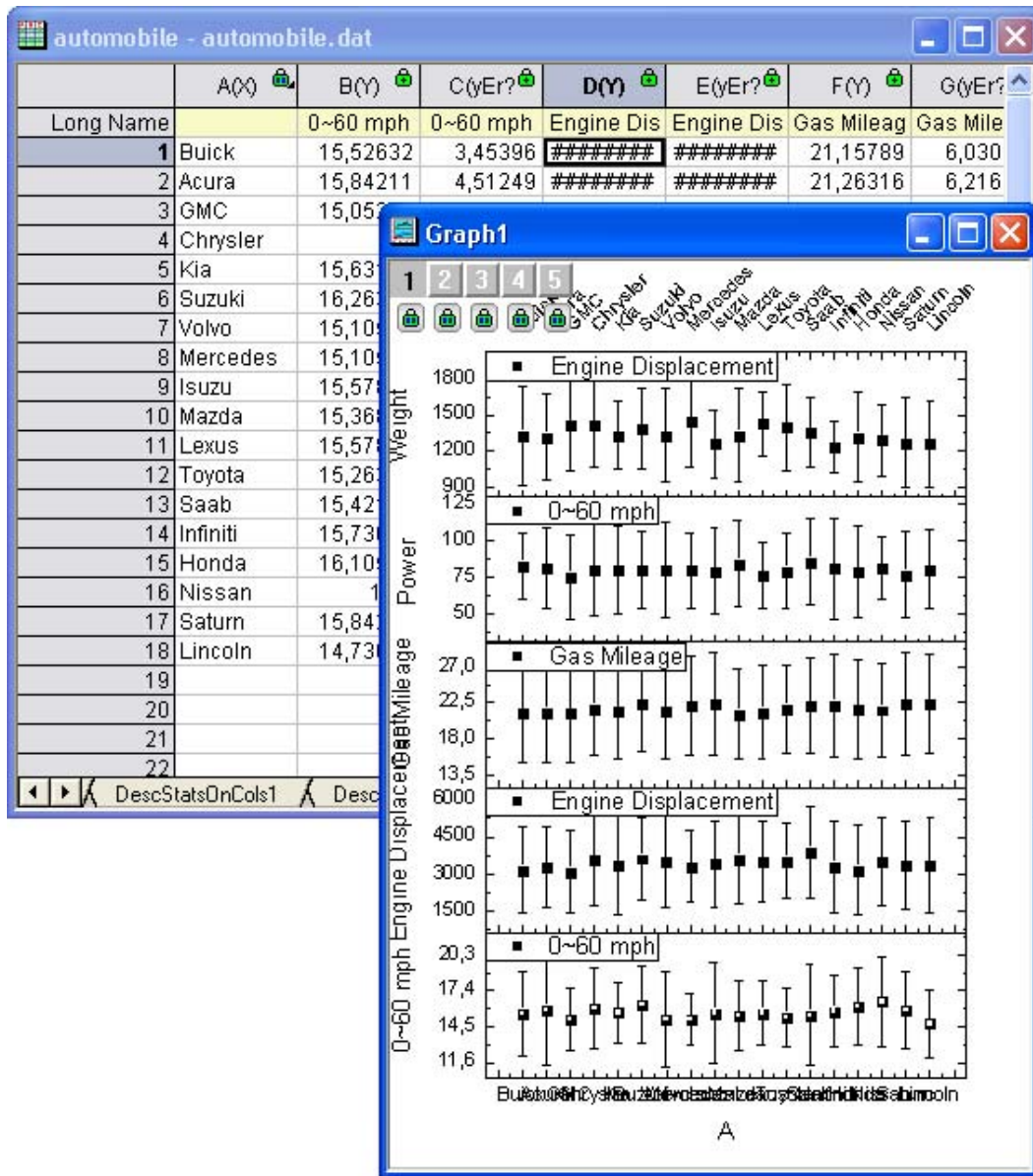
2. When the new sheet is active, select **Worksheet: Unstack Columns**.
3. In the dialog that comes up, set columns D and E as **Data to be Unstacked**. Since the triangle button fly-out menu supports only one selection, you need to use the interactive button.
4. Set **column A** as **Group Variables**.
5. Set **Put Group Info.** to **Long Name**.

6. Select the **Include Other Columns** check box and set **Other Columns** to column B. Click the **OK** button.



7. In the result of **Unstack Columns**, we get the mean and standard deviation of Power, 0~60 mph time, Weight, Gas Mileage and Engine Displacement for the 18 different car makes.
8. To compare these indicators, we can plot a graph of the results. Set user parameter row **A** as **Long Name** by right-clicking on this user parameter row and selecting **Set As Longname** from the context menu.
9. Highlight the whole worksheet. Select **Plot: Multi-Curve: Stack** from the main menu.

- In the pop-up dialog, all columns in the worksheet are automatically set as **Input**. Set **Plot Type** to **Scatter** and click the **OK** button.

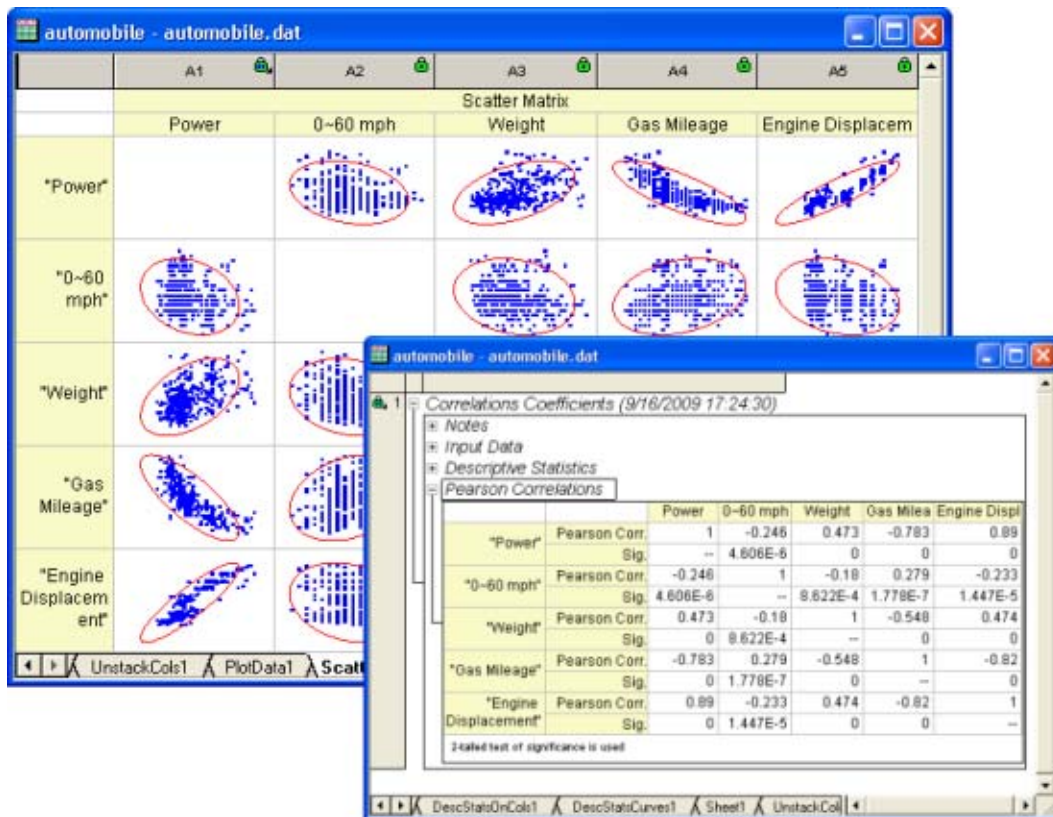


In the above screenshot, the X-Axis Tick Labels have been rotated 45 degrees for clarity. To do this, double-click on the tick labels to open the **X-Axis** dialog. Set the **Rotation** on the **Custom Tick Labels** tab.

Analyzing the Relationship between different Indicators

We can use a correlation coefficient to explore the relationship between columns of our automobile data. In addition, we can plot a scatter matrix with a confidence ellipse to get a graphical representation of the correlation.

1. Go to the original worksheet with the source data. Highlight the last five columns.
2. Select **Statistics: Descriptive Statistics: Correlation Coefficient** from the Origin menu to open the **Correlation Coefficient** tool. Note that **Pearson** is the default selection. This method is suitable for quantitative data.
3. Under the **Plots** branch, select the **Add Confidence Ellipse** check box. The **Scatter Plot** check box should then be automatically selected. This means that the tool will create a scatter matrix with a confidence ellipse added to each scatter plot. Click **OK**.



Note the high positive correlation between **Engine Displacement** and **Power** and the high negative correlation between **Gas Mileage** and **Engine Displacement**.

2.2 ANOVA

2.2.1 One Way ANOVA

Summary

There are two main modes of datasets in Statistics - indexed and raw. When you perform an analysis, you do not need to use the whole dataset, so Origin provides several ways to select data. For example, you can use the interactive Regional Data Selector button to graphically select the data or you can use the **Column Browser** dialog to make your selection.

In this tutorial, you'll use the Analysis of Variance (ANOVA) statistical test, to learn how to use these two different modes of data to perform analysis and how to select data by using the **Column Browser** dialog.

ANOVA is a kind of parametric method for means comparison and is an extension of t-test. When there are more than two groups to be compared, pairwise t-test is not appropriate and ANOVA should be used. ANOVA requires normality and equal variance. Otherwise, non-parametric analysis should be used.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Use different input data mode on statistical analysis dialog
- Test normality for special part of dataset
- Perform one-way ANOVA
- Select data by Column Browser

Steps

Origin can calculate ANOVA in indexed as well as raw data mode. For One-Way ANOVA, when using indexed mode, data should be organized in two columns : one for Factor and the other for data.

	A(X)	B(Y)
Long Name	plant	nitrogen
Comments	Factor	Data
1	PLANT3	18.15473
2	PLANT3	12.90409
3	PLANT2	18.61197
4	PLANT1	17.7111
5	PLANT4	11.81661
6	PLANT3	11.68327
7	PLANT2	23.43165
8	PLANT2	14.01454

When using Raw data mode, the different levels are in different columns.

	A()	B()	C()	D()
Long Name	Plant1	Plant2	Plant3	Plant4
Comments	Level1	Level2	Level3	Level4
1	17.7111	18.61197	18.15473	11.81661
2	32.15046	23.43165	12.90409	2.39438
3	17.70871	14.01454	11.68327	1.09914
4	28.07729	12.17685	23.52293	16.00756
5	7.83567	4.86902	16.00594	13.85077
6	2.06008	18.93963	3.04056	9.22245
7	22.81923	29.92086	14.29516	14.86523

Indexed data mode

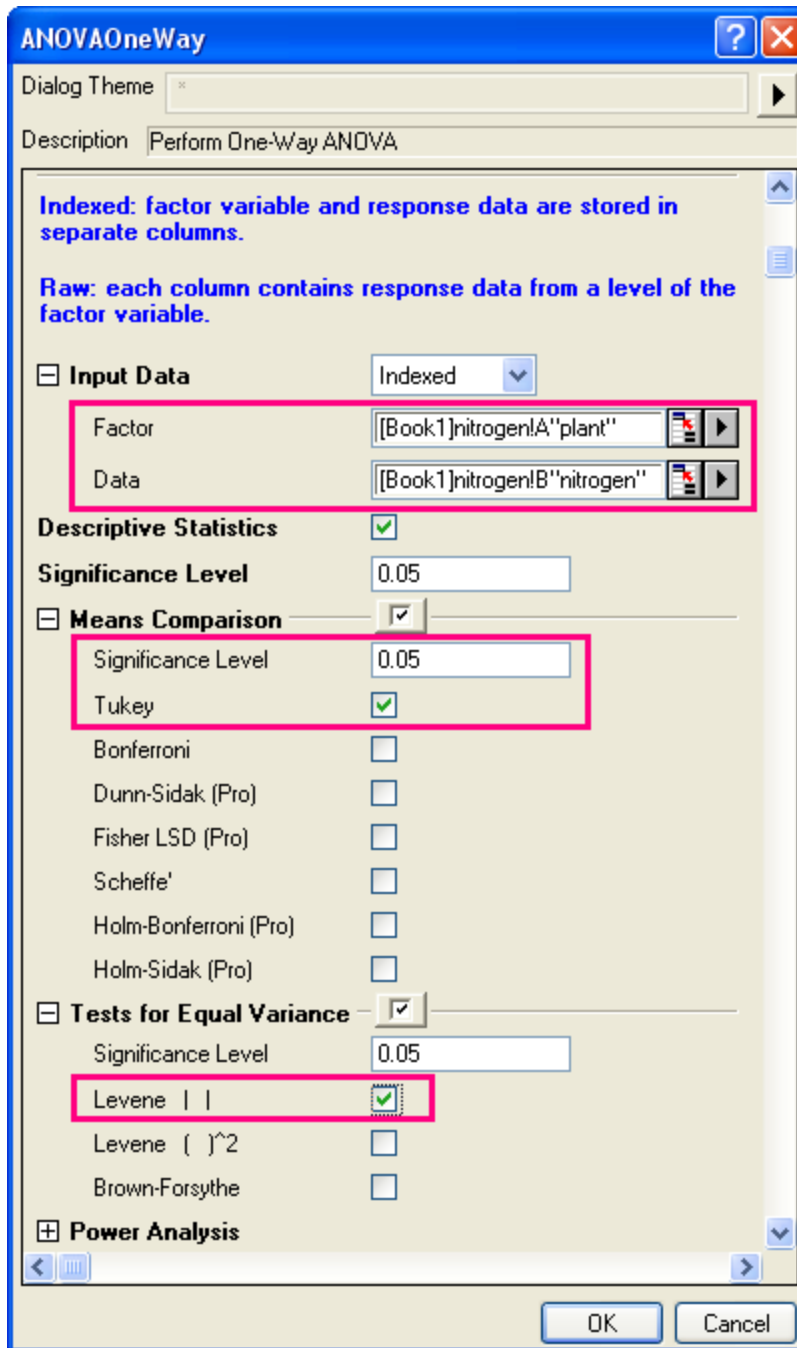
Nitrogen content has been recorded in milligrams for 4 kinds of plant, and we are interested in whether different plants have different nitrogen content. We will perform One-Way ANOVA using index data mode for this example.

1. Start with a new workbook and import the file `\Samples\Statistics\nitrogen.txt`. Make sure you select `.txt` from the drop-down menu Files of type. First, we should perform a normality test on each group of data to determine if they are from a normal distribution.
2. Highlight the first column, right-click and select **Sort Worksheet** from the Worksheet menu and choose **Ascending**.
3. Highlight the second column from row 1 to row 20 - which belongs to "PLANT1" - and open the **Normality Test** dialog by choosing the menu item **Statistics: Descriptive Statistics:**

Normality Test.

	A(X)	B(Y)
Long Name	plant	nitrogen
Units		
Comments		
Sparklines		
1	PLANT1	12.25362
2	PLANT1	18.62515
3	PLANT1	19.41474
4	PLANT1	17.7111
5	PLANT1	6.22726
6	PLANT1	18.20339
7	PLANT1	22.81923
8	PLANT1	2.06008
9	PLANT1	32.15046
10	PLANT1	28.07729
11	PLANT1	17.70871
12	PLANT1	20.60228
13	PLANT1	25.23966
14	PLANT1	16.73526
15	PLANT1	20.75954
16	PLANT1	30.35227
17	PLANT1	31.14475
18	PLANT1	17.58991
19	PLANT1	12.39297
20	PLANT1	7.83567
21	PLANT2	18.93963
22	PLANT2	29.92086
23	PLANT2	12.17685
24	PLANT2	18.61197
25	PLANT2	21.06720

- Use the default setting of the dialog and click **OK**. From the p-value of result, we can see "PLANT1" follows a normal distribution.
- In a similar way, you can highlight the range of data "PLANT2", "PLANT3" and "PLANT4" and test for Normality. Our sample data has normal distribution for all plants.
- With our nitrogen data worksheet active, open the **ANOVAOneWay** dialog by using the menu item **Statistics: ANOVA: One-Way ANOVA**. Set the **Input Data** mode as **Indexed**, assign the "plant" and "nitrogen" column as **Factor** and **Data** respectively using the right-arrow buttons. Click the + to expand the **Means Comparison** node, set **Significance Level** as 0.05 and check the **Tukey** Means Comparison method. Check **Levene** | | from **Tests for Equal Variance** branch. Click the **OK** button to perform One-Way ANOVA.



Explaining the result:

- From the "Homogeneity of Variance Test" table of one-way ANOVA result, we can see that the four groups have equal variance, since the p-value is bigger than 0.05.

Homogeneity of Variance Test

Levene's Test(Absolute Deviations)

	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	3	18.06843	6.02281	0.34578	0.79229
Error	76	1323.76846	17.41801		

At the 0.05 level, the population variances are not significantly different.

- From the result of Overall ANOVA we can conclude that at least two groups of the four have significant different means, since the p-value is smaller than 0.05.

Overall ANOVA

	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	3	1996.36652	665.45551	12.86214	6.99338E-7
Error	76	3932.05317	51.73754		
Total	79	5928.41969			

Null Hypothesis: The means of all levels are equal.
 Alternative Hypothesis: The means of one or more levels are different.
 At the 0.05 level, the population means are significantly different.

- To research further, we expand the results of "Means Comparisons".

Means Comparisons

Tukey Test

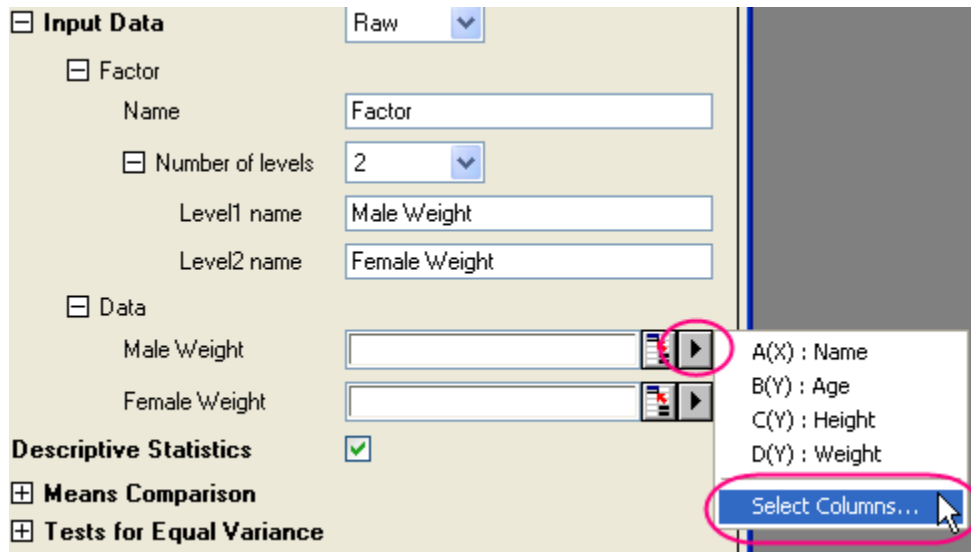
	MeanDiff	SEM	q Value	Prob	Alpha	Sig	LCL	UCL
PLANT2 PLANT1	2.26308	2.27459	1.40706	0.75274	0.05	0	-3.71181	8.23796
PLANT3 PLANT1	-2.46538	2.27459	1.53284	0.70039	0.05	0	-8.44027	3.5095
PLANT3 PLANT2	-4.72846	2.27459	2.93989	0.16935	0.05	0	-10.70334	1.24643
PLANT4 PLANT1	-10.93833	2.27459	6.80085	4.38499E-5	0.05	1	-16.91322	-4.96345
PLANT4 PLANT2	-13.20141	2.27459	8.20791	8.24355E-7	0.05	1	-19.1763	-7.22653
PLANT4 PLANT3	-8.47295	2.27459	5.26801	0.00207	0.05	1	-14.44784	-2.49807

Sig equals 1 indicates that the means difference is significant at the 0.05 level.
 Sig equals 0 indicates that the means difference is not significant at the 0.05 level.

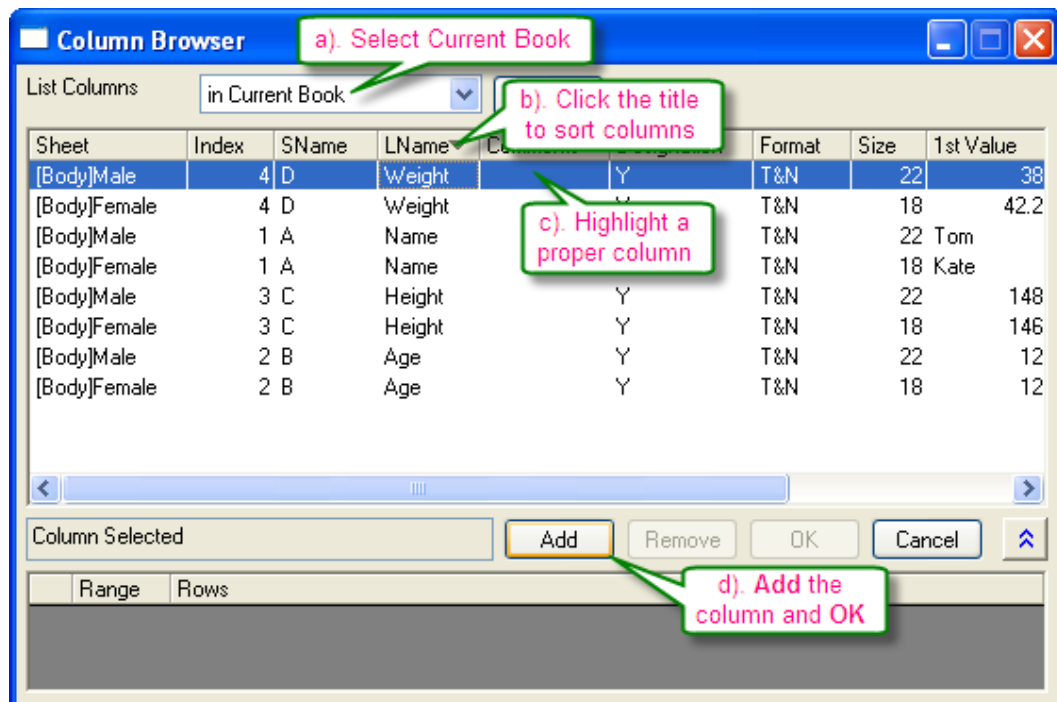
Here we see that PLANT4 has significantly different means when compared to each of the other three groups.

raw data mode

- Select **File : Open** and choose **WorkBooks** from **Files of type** drop-down list, and browse to `\Samples\Statistics` folder and open the file `Body.ogw`
- Select menu item **Statistics : ANOVA : One-Way ANOVA** to bring up the **ANOVAOneWay** dialog. Choose **Raw** as **Input Data** mode. Enter the **Level1 Name** and **Level2 Name** as "Male Weight" and "Female Weight" respectively.
- Now we will use the **Data Browser** to select data in the **Data** branch. Click the triangle icon beside **Male Weight** edit box, in the fly-out menu, select **Select Columns...** to open the **Column Browser** dialog.



In the Column Browser dialog, you can select **in Current Book** from **List Columns** drop-down list to see all available worksheet columns in the current book. Select *Weight* in the sheet *[Body]Male* and click **Add** and **OK** to add it to **Male Weight** edit box. Similarly, assign *Weight* from *[Body]Female* to **Female Weight** edit box.



- Accept other default settings in the ANOVAOneWay dialog and click **OK**. From the output report footnote, we can conclude that at the 0.05 level, the population weight means between male and female are not significantly different.

3 Graphing

3.1 Graphing Data From Multiple Sheets

3.1.1 Summary

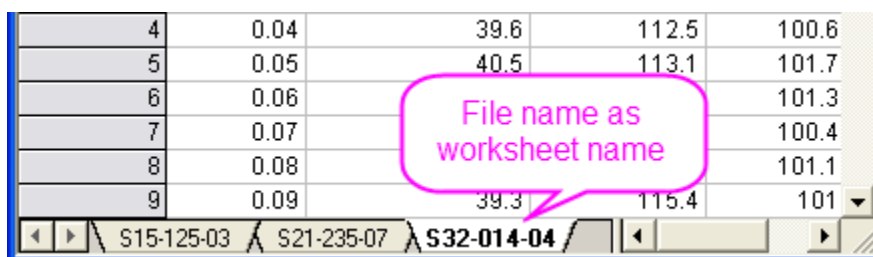
Origin provides close to 150 built-in graph templates that can be used to create a wide variety of plots. You can modify these templates or create your own to add to the collection. Creating a graph in Origin is as simple as selecting the desired data and then selecting a template from a menu or from the Graphing toolbars. The Plot Setup dialog offers more flexibility in creating plots, such as plotting data from multiple books or sheets.

This tutorial will show you how to:

- Select data in a worksheet and quickly create a plot
- Add data to an existing graph with drag-and-drop plotting
- Use Plot Setup to plot data from multiple sheets
- Create and save a custom graph template
- Plot groups of plots by label

3.1.2 Create a Plot Quickly by Selecting Data

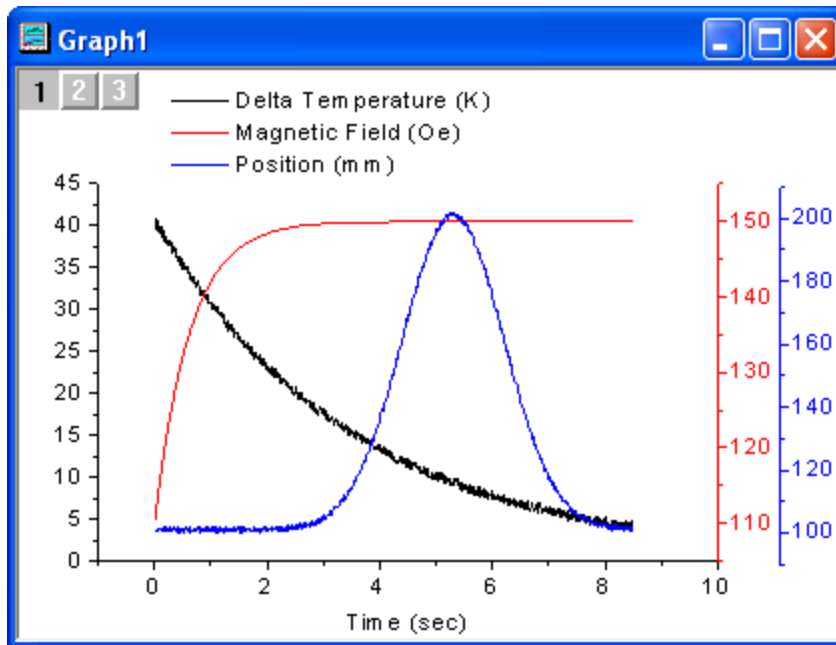
1. Select **File: Recent Imports: impASC: My Multifile Import** from the main menu. (**My Multifile Import** is a theme saved in another tutorial. If you have never saved it, please perform the steps in the tutorial to create it.)
2. Import the files *S15-125-03.dat*, *S21-235-07.dat* and *S32-014-04.dat* from **\Samples\Import and Export**. Note that the settings in the theme are used to import the files, so Origin will import each file to a different worksheet of the same workbook. The file names are used as the worksheet names.



4	0.04	39.6	112.5	100.6
5	0.05	40.5	113.1	101.7
6	0.06			101.3
7	0.07			100.4
8	0.08			101.1
9	0.09	39.3	115.4	101


3. Make the third worksheet, *S32-014-4*, active. To create a three layer graph, highlight the three Y columns, Delta Temperature, Magnetic Field, and Position, and then select **Plot: Multi-Curve: 3Ys Y-YY**. Note: There is no need to highlight the Time column, as Origin will automatically plot

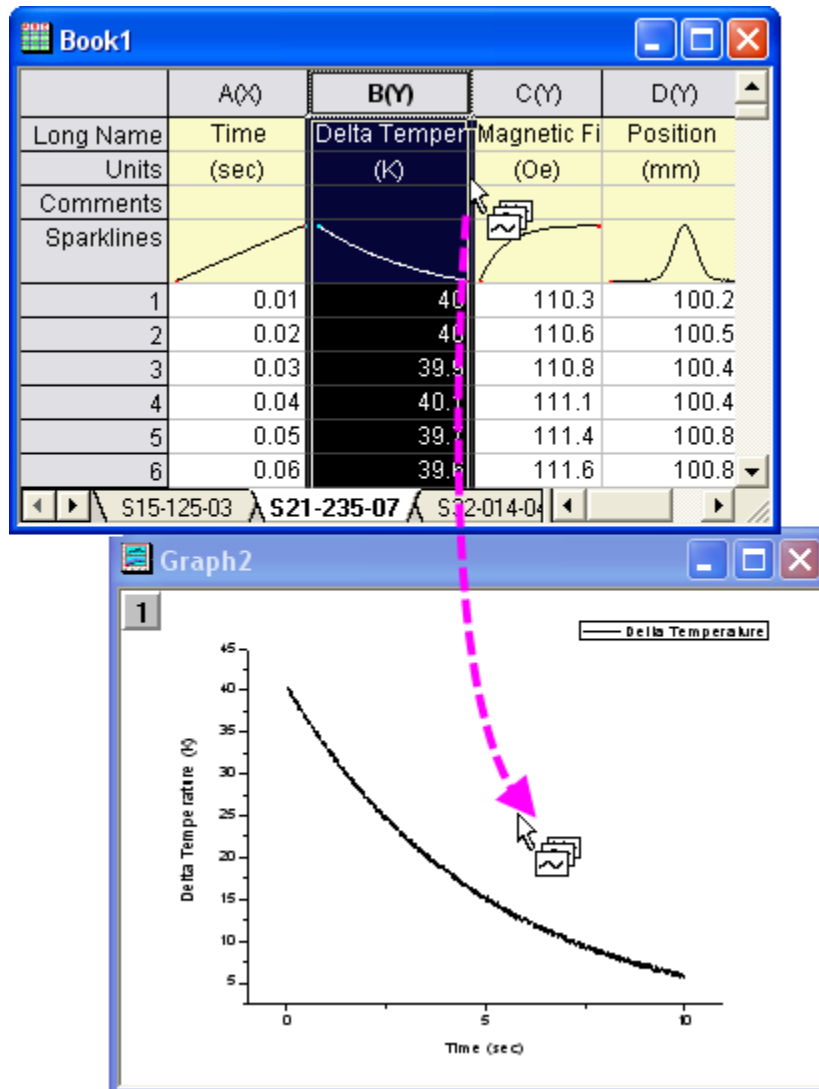
the Y columns against the associated X column in the worksheet.



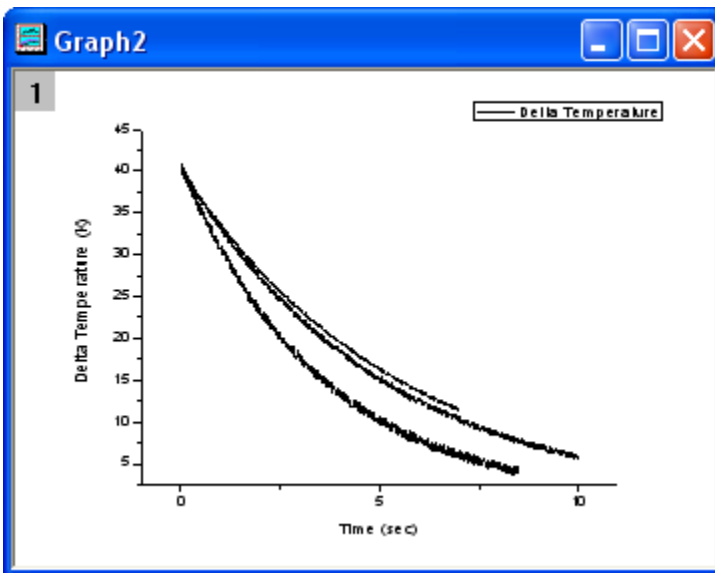
3.1.3 Add Data to an Existing Graph with Drag and Drop Plotting

1. Go back to the book with the three sheets of data from the above example.
2. Highlight the **Delta Temperature** column (Column B) from the first sheet and then select **Plot: Line: Line** to create a line plot.
3. Go back to the workbook, and for the other two sheets, one at a time, select the **Delta Temperature** column, and position the cursor at the edge of the column, until you see the

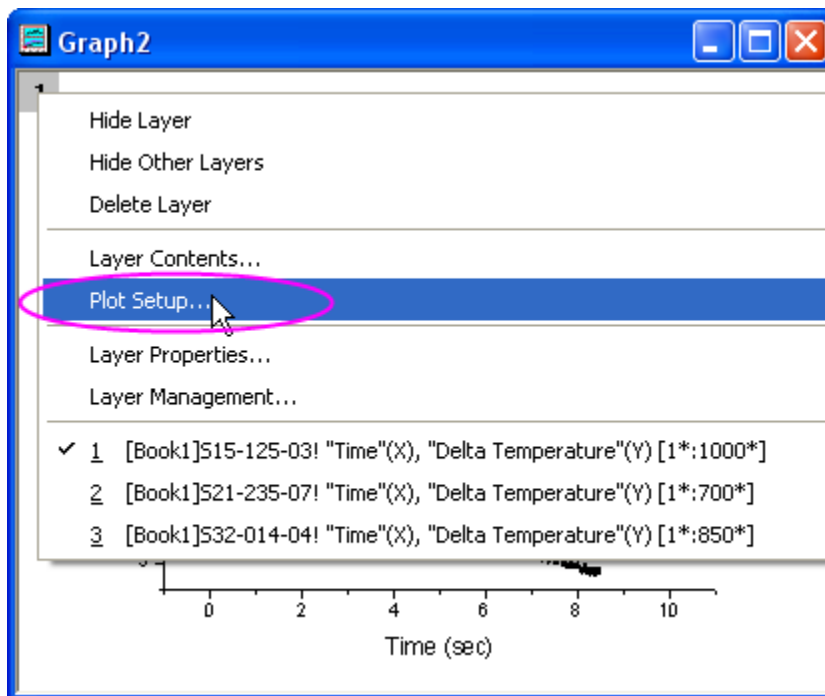
cursor turn into .




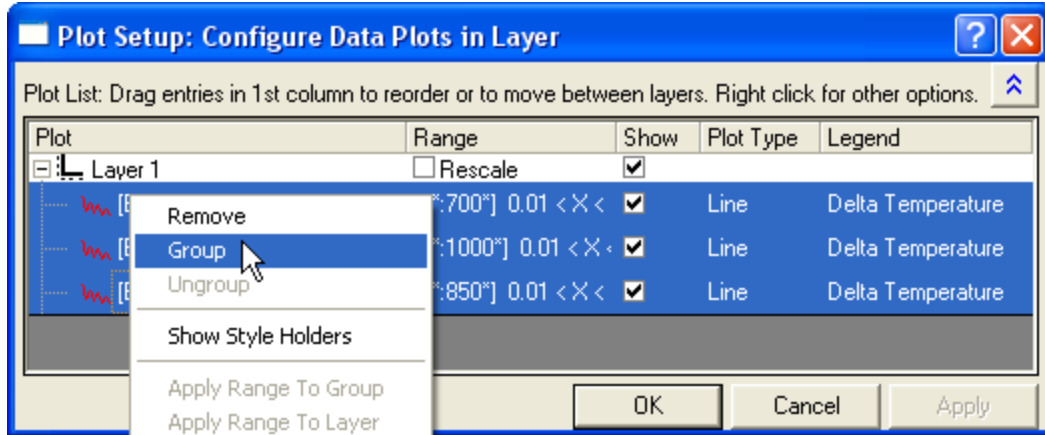
Then you can drag-and-drop the column into the graph page. Another curve will be added to the current layer.



4. After adding the other curves, right click on the layer icon and select **Plot Setup** from the context menu.

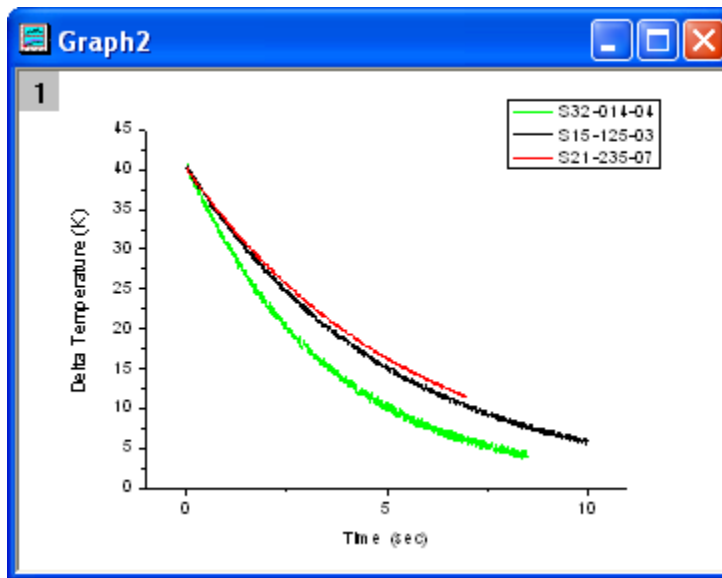


5. The **Plot Setup** dialog opens with the bottom panel visible. (Note: The **Plot List** panel is the only one you need, but if you wish, you can click the  button to expand and show the other two panels of the dialog.) Hold the **Shift** key while selecting all three data plot entries, and then right-click and select **Group**. Click **OK** to close the dialog.




Grouping the plots allows for quick creation of presentation-ready graphs, because each selection in the group is assigned a differentiating set of plot attributes (line color = black, red, green; symbol shape = square, circle, triangle; etc...).

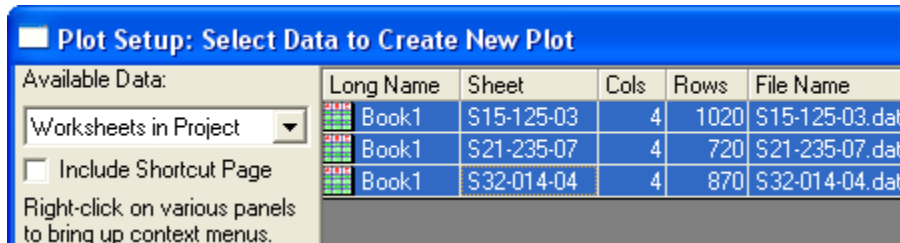
6. Select **Graph: Update Legend** to open the *legendupdate* X-Function dialog. Set the **Auto Legend Translation Mode** to *Custom*. Enter **@WS** for **Legend Custom Format**. Click **OK** and the legend will now contain the worksheet name for each data plot.




3.1.4 Create a Plot using Plot Setup

The Plot Setup dialog box is useful for a variety of plotting tasks, including creating graphs, modifying the plot type, adding plots to or removing plots from the graph, grouping or ungrouping plots, and editing the plot range.

1. Using the same workbook as in the above examples, select no data. It doesn't matter what worksheet is active; what is important is that no columns are highlighted.
2. Select **Plot: Multi-Curve: 4 Panel** from the menu. Without any data selected, Origin will open the **Plot Setup** dialog, allowing you to choose the data you wish to plot.
3. Expand the top panel by clicking the  button. Hold down the **Shift** key to highlight the three worksheets. Alternatively, you can just click and drag your selection so that all three worksheets are highlighted.



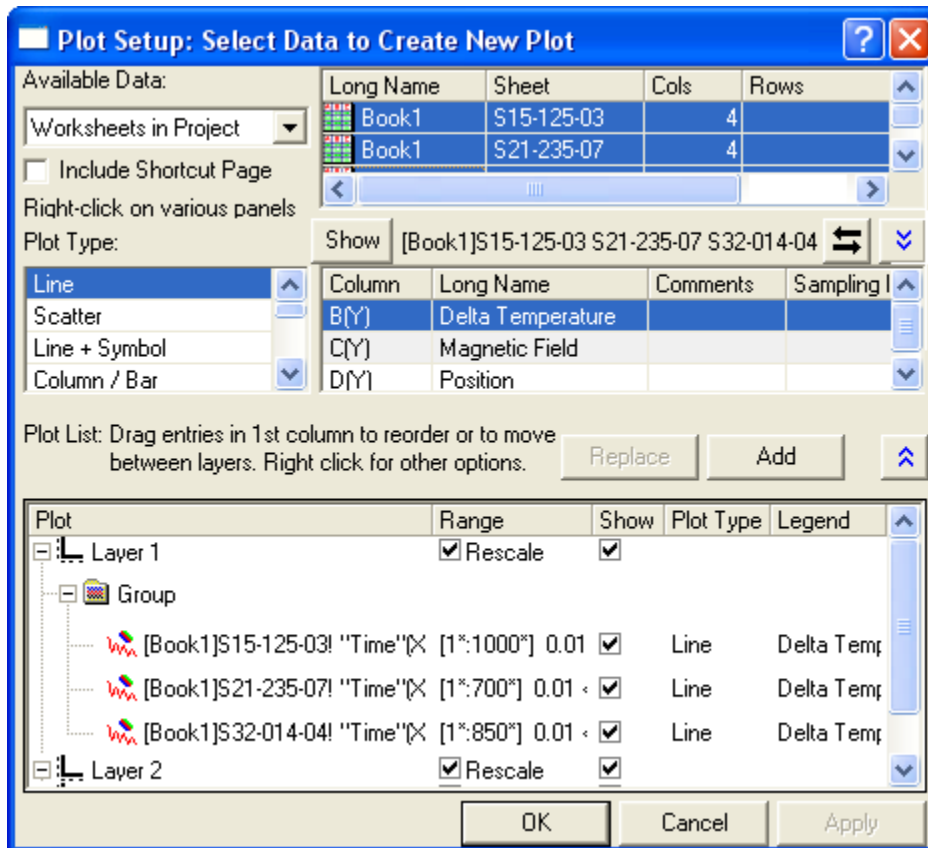
4. In the middle panel, common columns in all three sheets are displayed. In this case, all three sheets have similar data with matching column names. Click the double arrow button  to show the plot column list. This mode is easier because you don't have to check the X and Y designation check boxes.

X	Y	yEr	L	Column	Long Name	Comments
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<autoX>	From/Step=	
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A	Time	
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	B	Delta Temperature	
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	C	Magnetic Field	
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	D	Position	

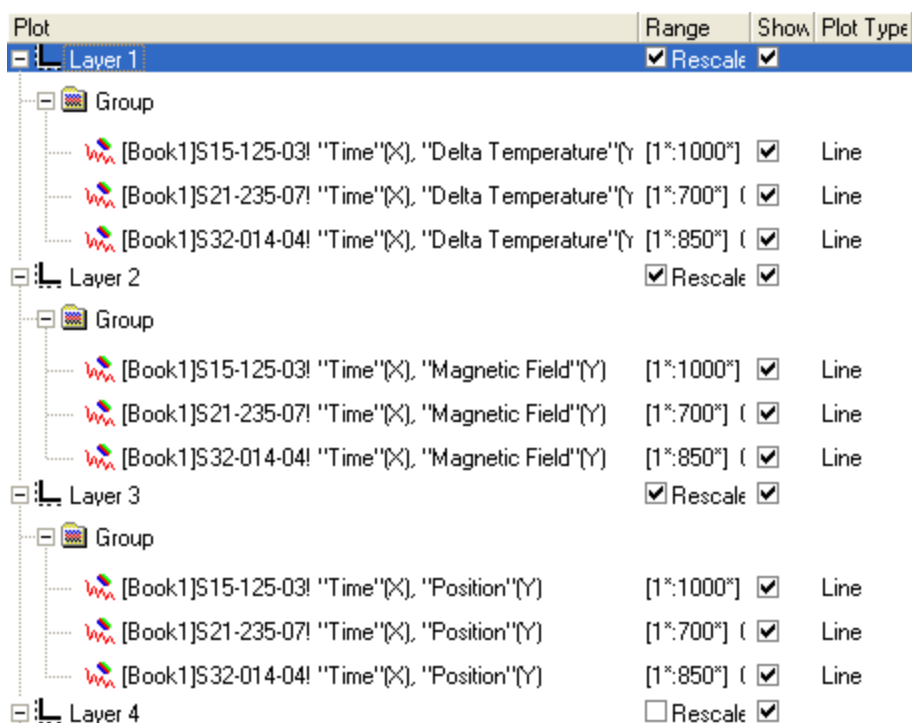


Column	Long Name	Comments	Sampling Interval
B(Y)	Delta Temperature		
C(Y)	Magnetic Field		
D(Y)	Position		

5. Open the bottom panel of the dialog, the Plot List, if it is not already visible. **Layer 1** is highlighted. Select **Delta Temperature** in the middle panel, and then click the **Add** button. Since you already selected the three worksheets in Step 3, this will add the **Delta Temperature** column from each of the three worksheets to layer 1.

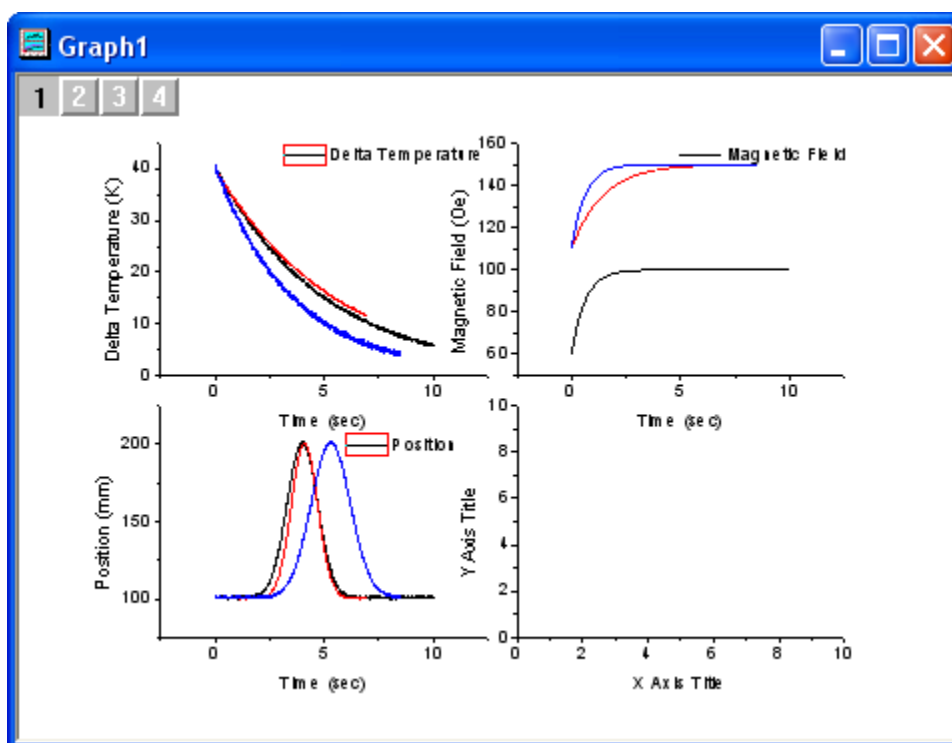


6. Repeat these steps to add **Magnetic Field** and **Position** into layer 2 and layer 3, respectively.



Note that in each layer, the three data plots are automatically grouped.

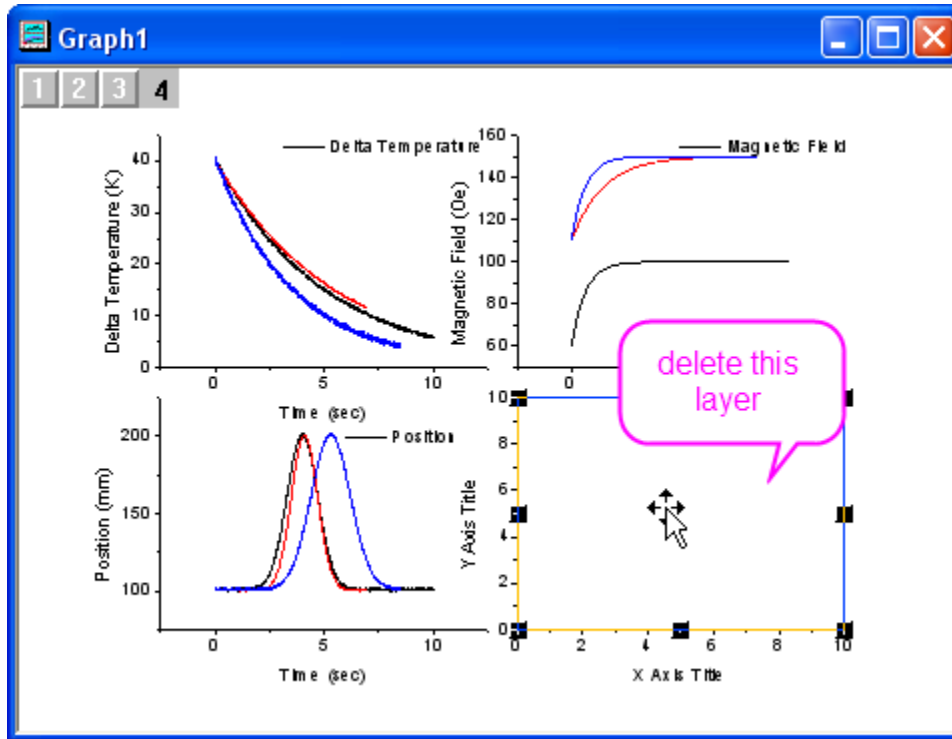
7. Click the **OK** button to create the graph.



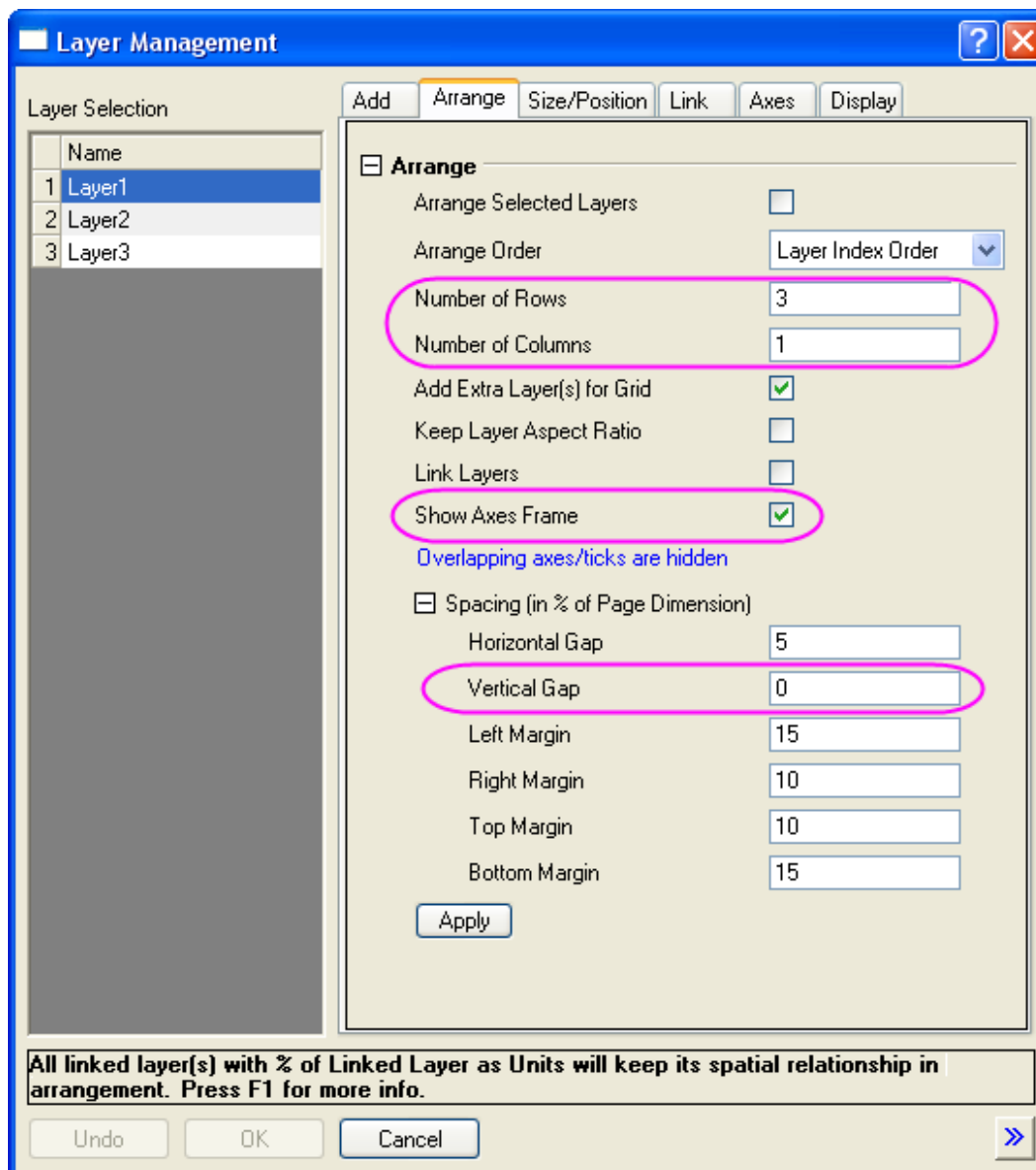
3.1.5 Customize and Save a Graph Template

In the above examples, the 3Ys Y-YY, Line and 4 Panel plots are each created from a different, specific built-in plot template. If you don't see the exact graph you need, you can often create it and save it as your own template to reuse later with similar data.

1. Continue with the 4 panel plot from above. Select the empty 4th layer and press the **Delete** key on the keyboard to get rid of it.

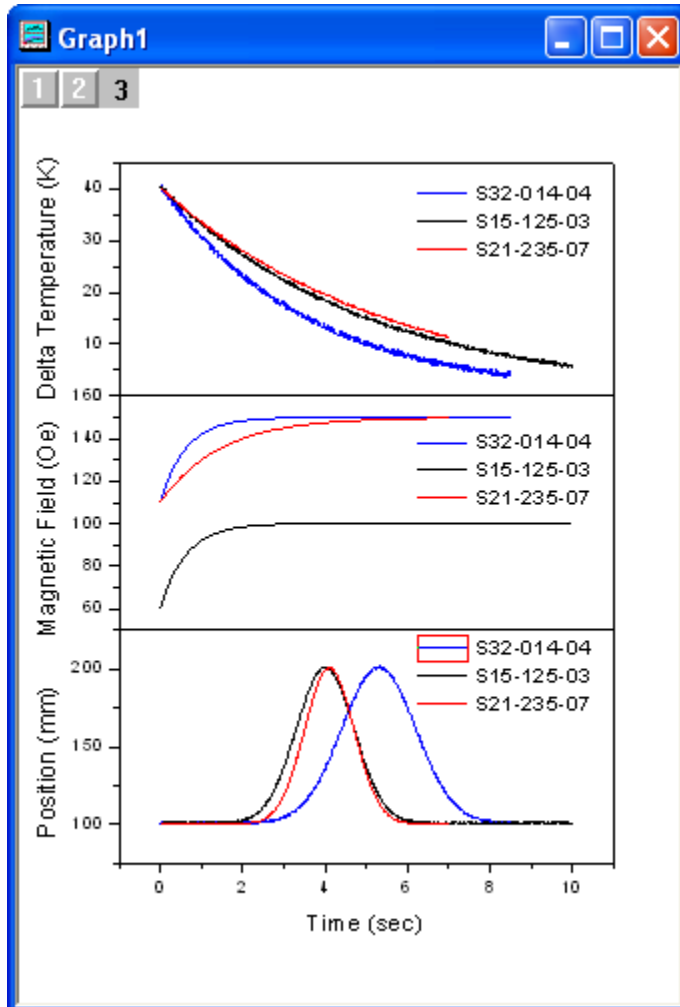


2. Select **Graph: Layer Management** from the menu to bring up the Layer Management dialog. On the **Arrange** tab, set column=1, row=3; check the **Show Axes Frame** check box; expand the Spacing(% of Page) branch and set the **Vertical Gap** to 0. Click **Apply** and then click **OK**.



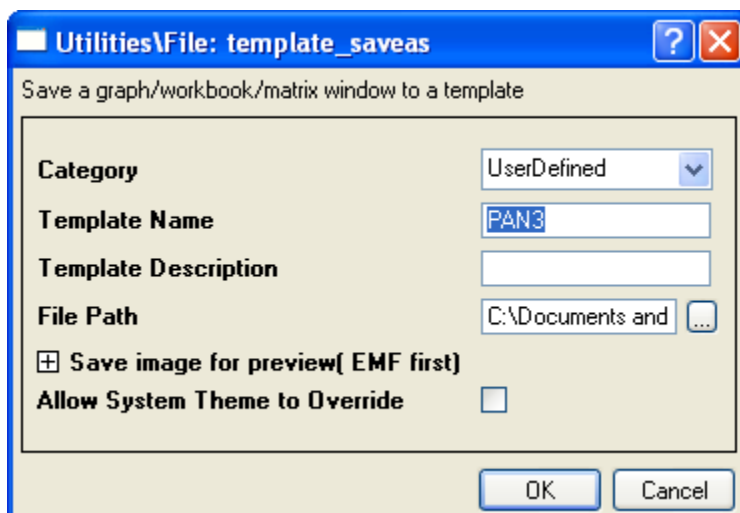
3. Select **File: Page Setup** from the menu and change the **Orientation** to **Portrait**.

4. Select **Graph: Update Legend: Open Dialog**. Leave the destination as Whole Page to update the legend in all layers on the graph page. Set the **Auto Legend Translation Mode** to *Custom*. Enter **@WS** for **Legend Custom Format**, and click OK. Your graph will look as follows:



To align the left Y-axis titles, you can hold down Shift and select them, and then use the Left Align button on the Object Edit toolbar.

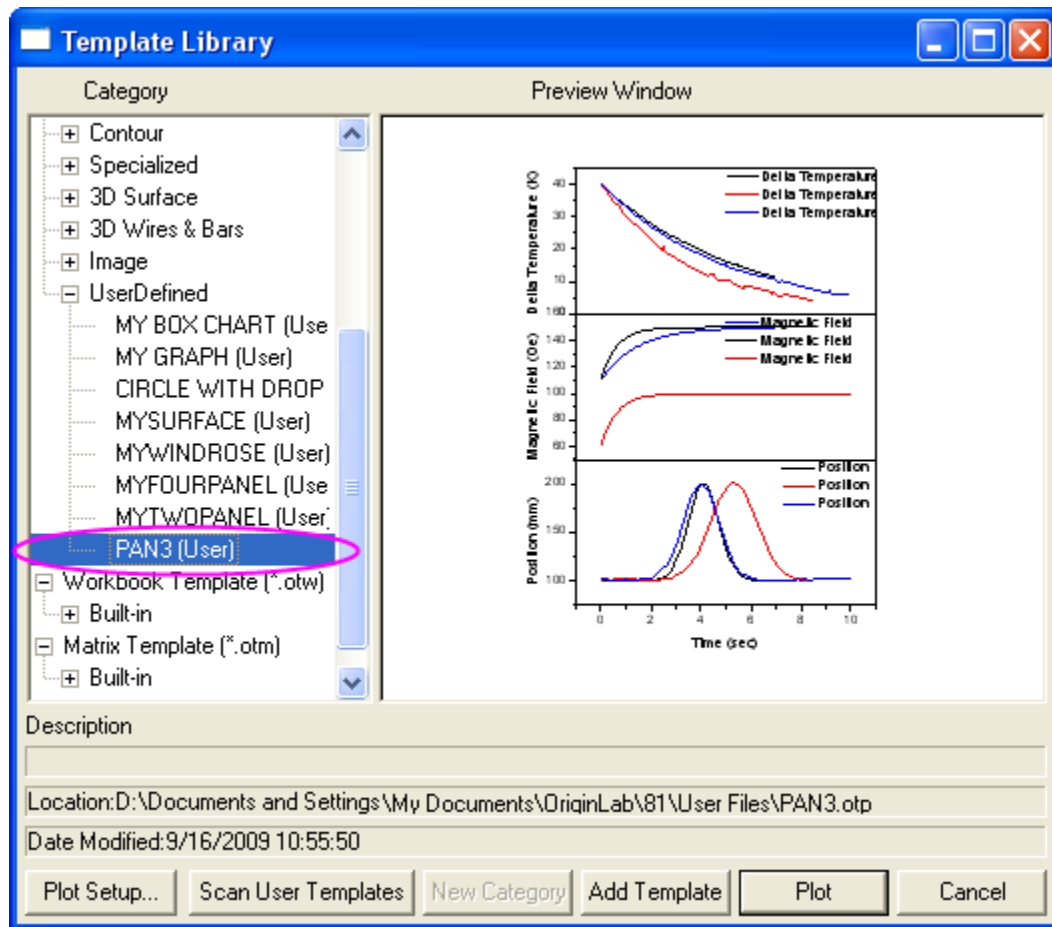
5. Now that you have customized the graph, select **File: Save Template As**. In the dialog that opens, save as a new template with a new name such as **PAN3** under the **UserDefined** category.



Now you can reuse this template in the next example.

3.1.6 Plot into a Saved Custom Template

1. Go back to the workbook, and in the menu, select **Plot: Template Library** to bring up the Template Library dialog. All of the Origin templates, including Graph, Workbook, and Matrix templates, are listed here. Expand the **UserDefined** category under **Graph Template**, and you'll see the one that you just saved in the last example.



2. Click the **Plot Setup** button. Then you can use the same steps from the **Create a Plot using Plot Setup** section to create a three-panel graph directly.

3.1.7 Plot Groups

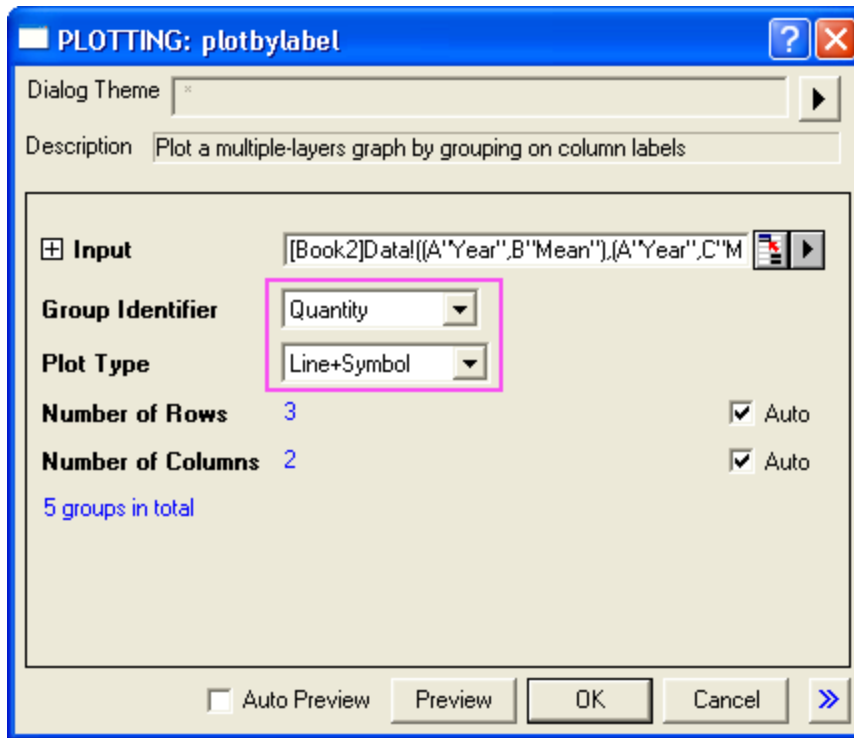
Origin offers a plotting option, **Multiple Panels by Label**, that allows you to create a multilayer graph, each layer of which contains multiple plots, identified by the same label.

1. Open the file **Samples\Graphing\Automobile Data.ogw**.

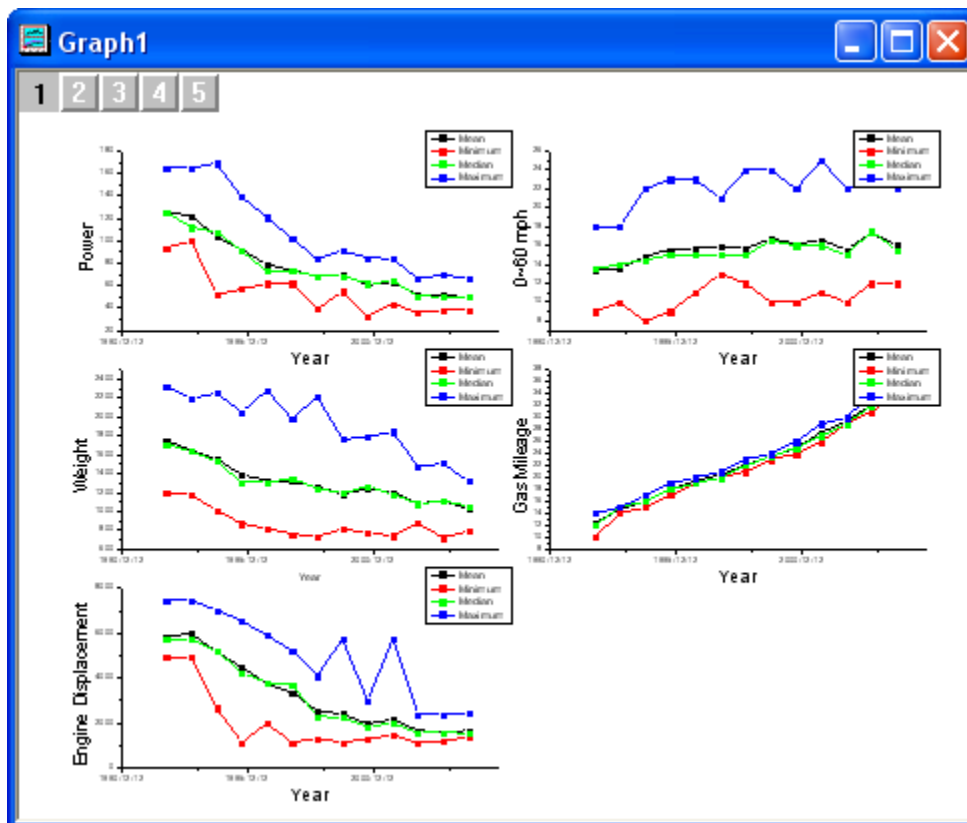
	A(X1)	B(Y1)	C(Y1)	D(Y1)	E(Y1)	F(X2)	G(Y2)	
Long Name	Year	Mean	Minimum	Median	Maximum	Year	Mean	M
Quantity	Power	Power	Power	Power	Power	0~60 mph	0~60 mph	0~
1	1992	125.46	94	126	165	1992	13.35	
2	1993	122.32	100	112	165	1993	13.52	
3	1994	103.57	52	108	169	1994	14.86	
4	1995	92.19	57	91	139	1995	15.57	
5	1996	78.96	62	73	121	1996	15.73	
6	1997	73.48	62	73	102	1997	15.86	
7	1998	68.42	39	69	84	1998	15.70	
8	1999	69.96	55	69	91	1999	16.75	
9	2000	60.72	33	62	85	2000	16.16	
10	2001	63.10	44	64	84	2001	16.59	
11	2002	52.24	36	51	66	2002	15.57	
12	2003	51.27	38	50	70	2003	17.32	
13	2004	49.88	38	50.5	66	2004	16.00	

2. Click on top left corner of the worksheet to select the entire sheet. Select **Plot: Multi-Curve: Multiple Panels by Label** to open the **plotbylabel** dialog.

- Set the dialog options as follows, and click the OK button to create the graph.



The graph should look like this:



3.2 Line Symbol

3.2.1 Basic 2D Plotting

Summary

Origin provides flexible ways to create 2D plots. You can easily customize plot attributes, arrange layers, and select different datasets for each layer. This tutorial will teach you the basic plotting skills.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

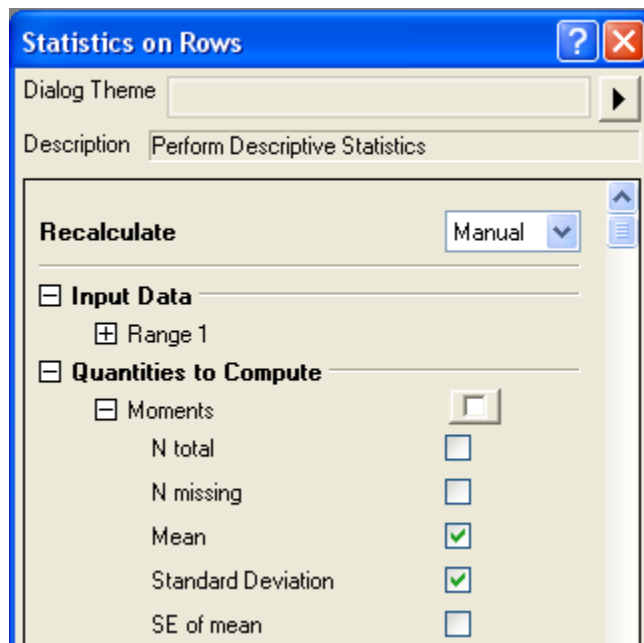
This tutorial will show you how to:

- Perform simple row statistics
- Create a graph and save as a template
- Plot into your template
- Use the Plot Setup dialog

Steps

Simple row statistics

1. Start with an empty worksheet, select **File: Import: Single ASCII...** to open the *Import Single ASCII* dialog, browse to the `\Samples\Curve Fitting` subfolder of the Origin program folder, and import the file *Dose Response - No Inhibitor.dat*.
2. Highlight columns 2 through 4 and select **Statistics: Descriptive Statistics: Statistics on Rows**. Make sure to check the **Mean** and **Standard Deviation** check boxes on *Quantities to Compute* branch to output these results.

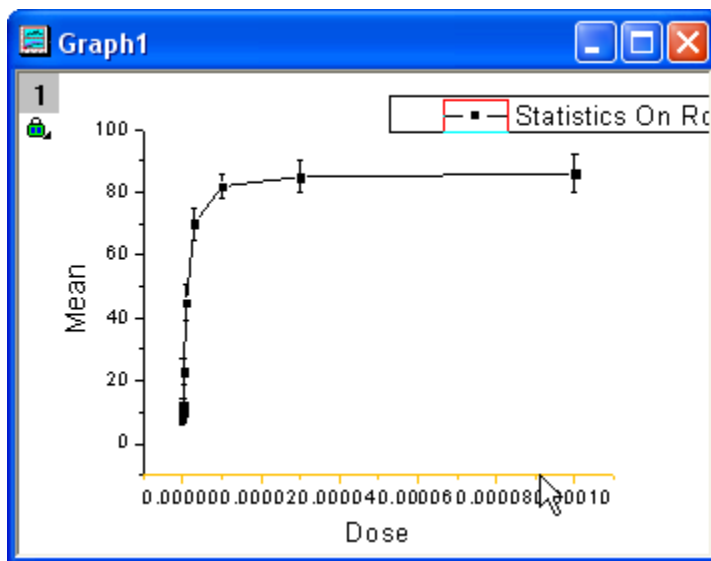


- After you click the **OK** button, two new columns, **Mean(Y)** and **SD(yErr)** are added to the source worksheet. Here, **yErr** means that this is an error column and the data in this column can be used to plot error bars.

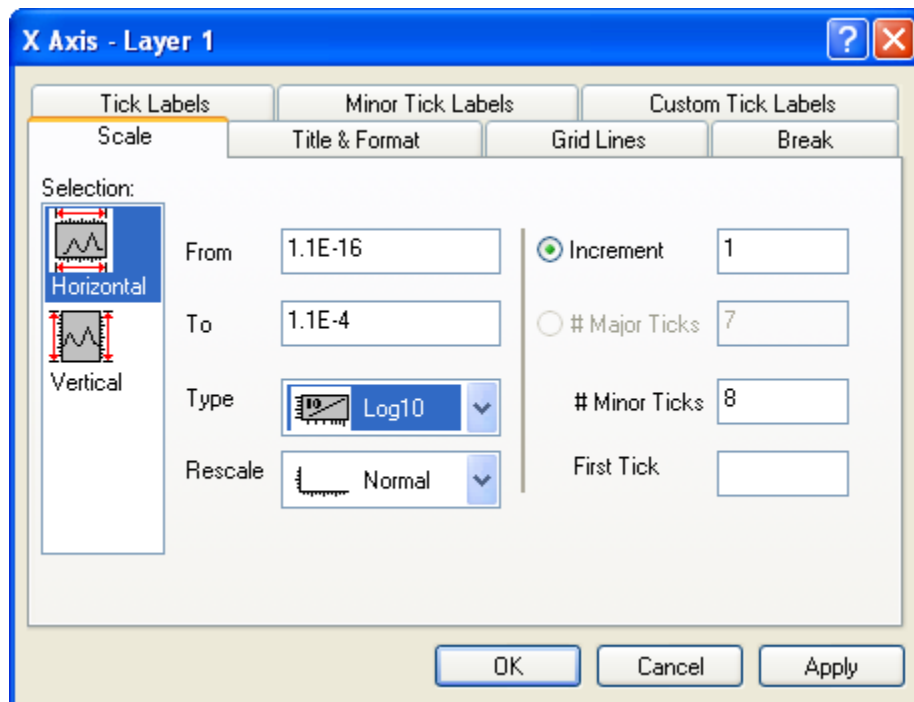
Note: To simplify plotting, each column in an Origin worksheet has a plotting designation. To change a column's plotting designation, select the column and click on the **Column** menu. Alternatively, right-click on the column and choose **Set As** from the context menu.

Create a graph and save as a template

- Highlight the Mean(Y) and SD(yErr-) columns and select **Plot: Line+Symbol: Line+Symbol** to create the plot:

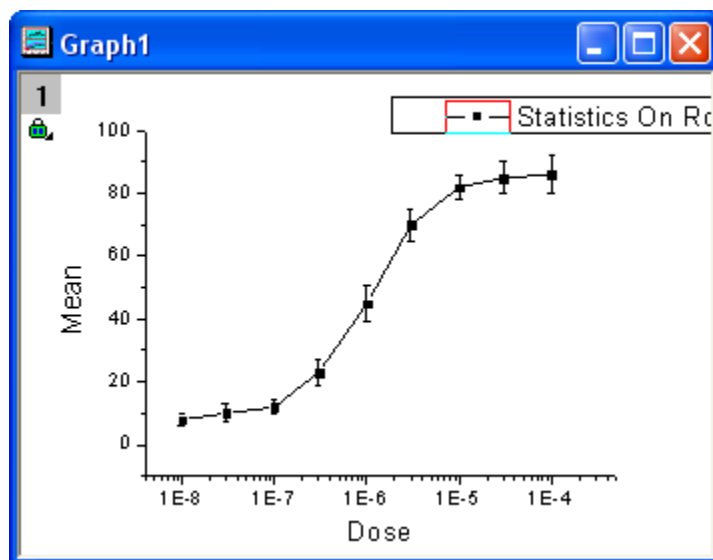


- To change the X scale to Log, double-click on the **X** axis to bring up the **X Axis** dialog. On the **Scale** tab, change the axis **Type** to **Log10**:

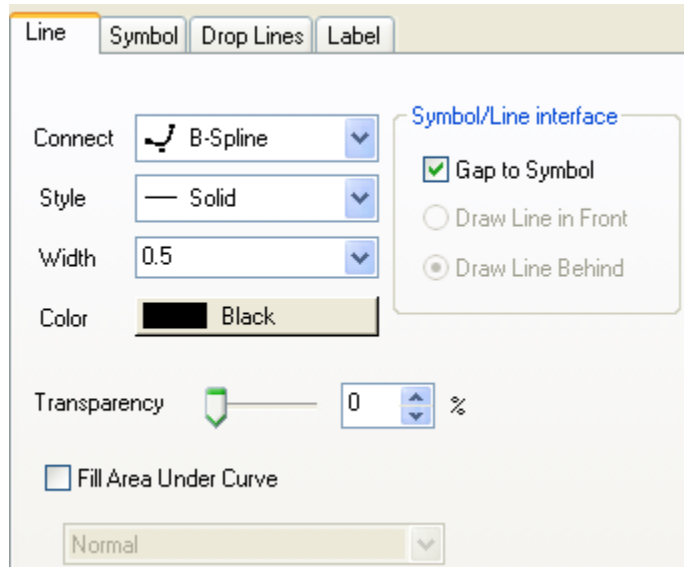


Click the **OK** button to close the dialog.

3. Select **Graph: Rescale to Show All** from the menu, which will rescale the X and Y axes of the graph. :

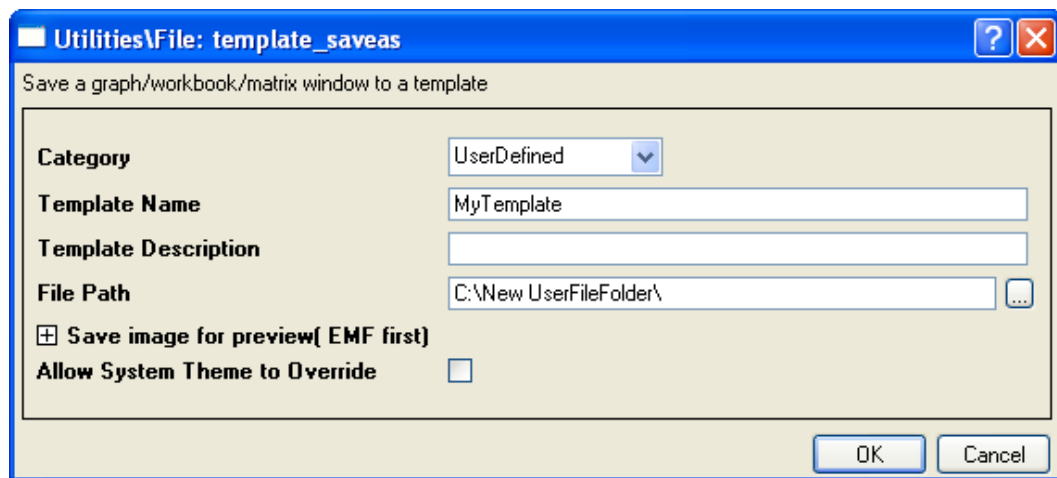


4. To edit the curve, double-click on any plot symbol to bring up the **Plot Details** dialog. Alternatively, right-click inside the graph and choose **Plot Details** from the context menu. On the **Line** tab of right panel, select **B-Spline** as connect line to get a smoother curve.




Click the **OK** button to close the dialog.

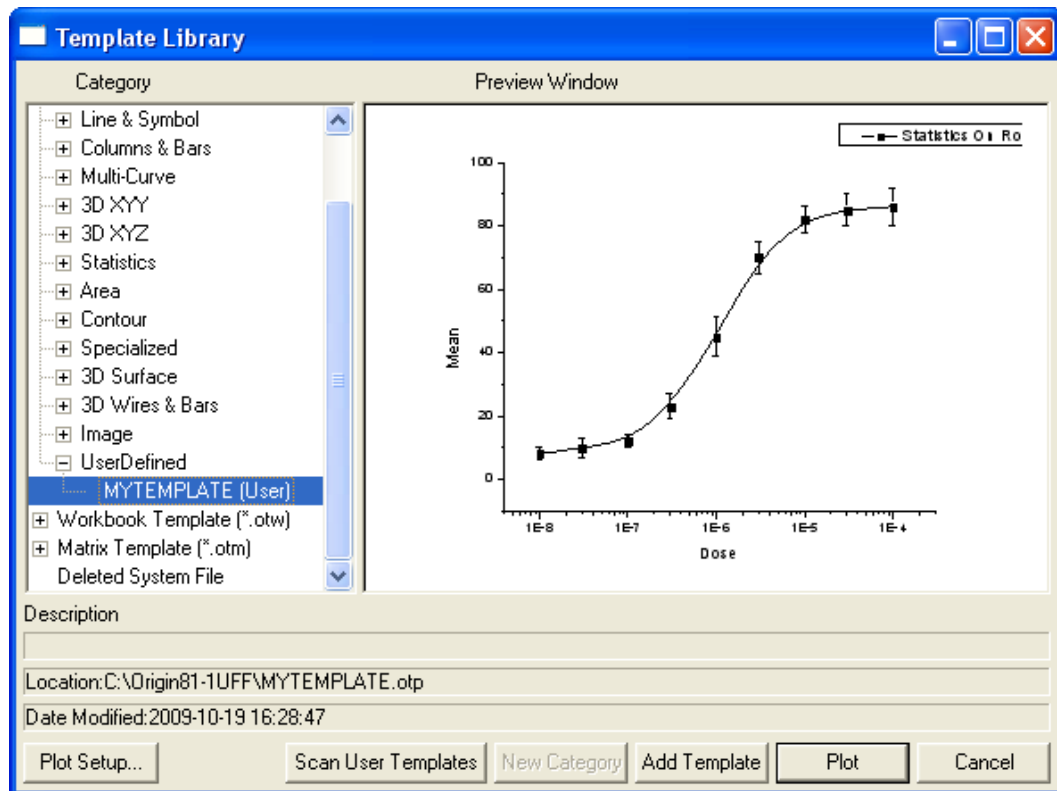
- When all modifications have been made and the graph looks the way you want it, you can use this graph to create a template, to be used in the future with similar data. Select **File: Save Template as** to open the *Save Template* dialog. In the **Category** drop-down list, select *UserDefined*; and then type a proper **Template Name**. In this example, we use *MyTemplate*. Click **OK** to save the template.





Plot into graph template with the Plot Setup dialog

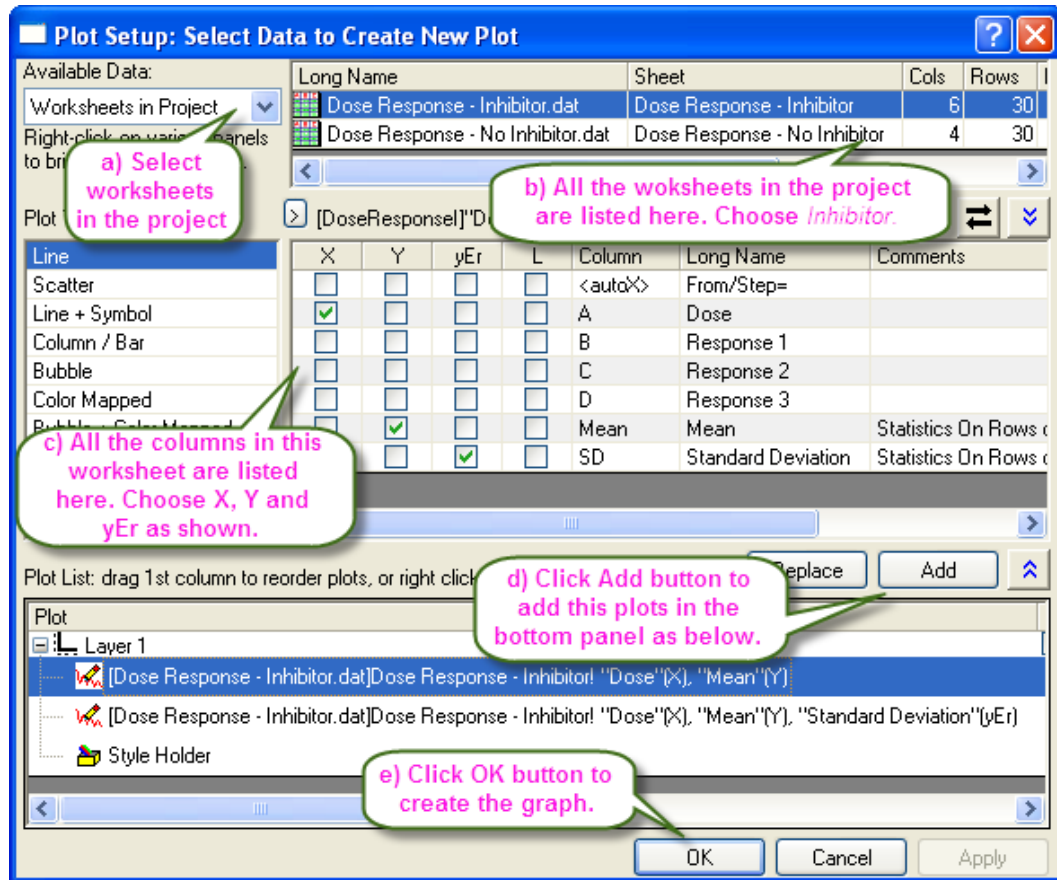
- Click the  button to open a new workbook, and import the file *\Samples\Curve Fitting\Dose Response - Inhibitor.dat* as above. Perform **Statistics on Rows**, calculating the Mean and SD of this worksheet as you did above and by following the same steps.

2. Select **Plot: Template Library** to open the *Template Library*. Select *MyTemplate* from the *UserDefined* category.



Click the **Plot Setup** button to select the data from which to create the plot. If you click the Plot button, Origin will plot whatever data is highlighted in the worksheet.

3. In the *Plot Setup* dialog, you can choose which columns are to be plotted. (There are three panels in *Plot Setup* dialog, click the  or  button to expand them) To finish creating the plot from your template, please follow the steps a - e outlined on the picture below.



Plot Setup: Select Data to Create New Plot

Available Data:

Long Name	Sheet	Cols	Rows
Dose Response - Inhibitor.dat	Dose Response - Inhibitor	6	30
Dose Response - No Inhibitor.dat	Dose Response - No Inhibitor	4	30

Plot List: drag 1st column to reorder plots, or right click

Plot:

Line	X	Y	yEr	L	Column	Long Name	Comments
Scatter	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<autoX>	From/Step=	
Line + Symbol	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A	Dose	
Column / Bar	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	B	Response 1	
Bubble	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	C	Response 2	
Color Mapped	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	D	Response 3	
Button Color Mapped	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Mean	Mean	Statistics On Rows c
	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SD	Standard Deviation	Statistics On Rows c

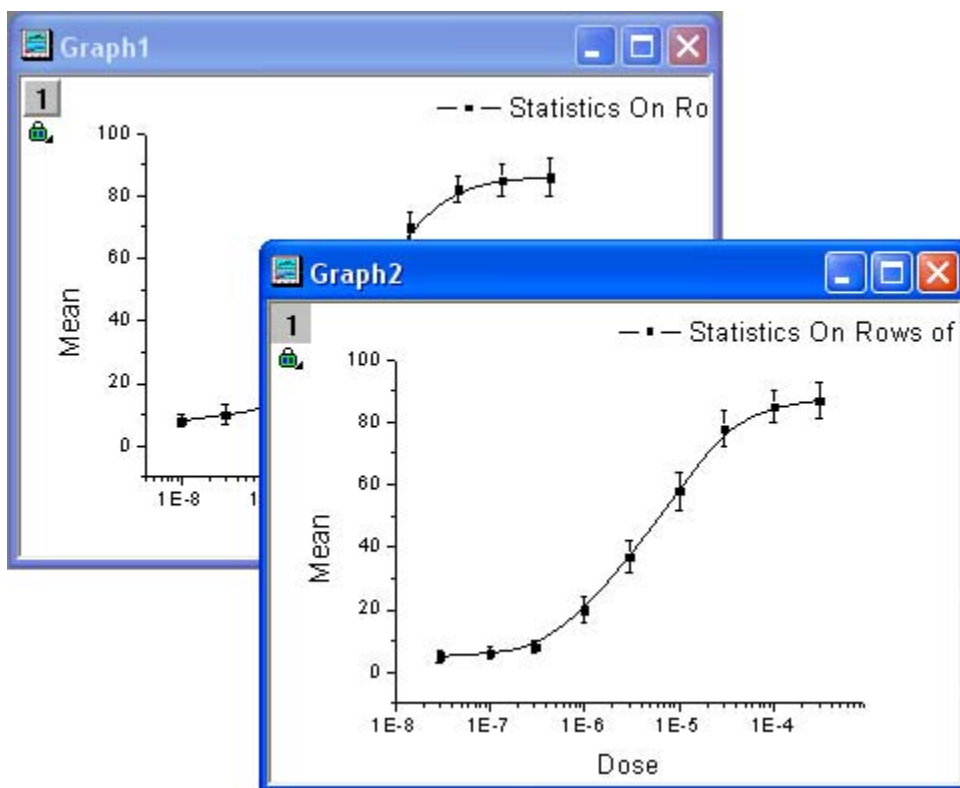
Plot List:

- [Dose Response - Inhibitor.dat]Dose Response - Inhibitor! "Dose"(X), "Mean"(Y)
- [Dose Response - Inhibitor.dat]Dose Response - Inhibitor! "Dose"(X), "Mean"(Y), "Standard Deviation"(yEr)
- Style Holder

Callout boxes:

- a) Select worksheets in the project
- b) All the worksheets in the project are listed here. Choose *Inhibitor*.
- c) All the columns in this worksheet are listed here. Choose X, Y and yEr as shown.
- d) Click Add button to add this plots in the bottom panel as below.
- e) Click OK button to create the graph.

And then you will have:



3.2.2 Line Graph with Recession Bars

Summary

This tutorial will show how to create a line graph with vertical bars spanning across the layer from top to bottom. This type of graph is common when plotting economics data with periods of recession marked as vertical bars.



Minimum Origin Version Required: Origin 8.0 SR6

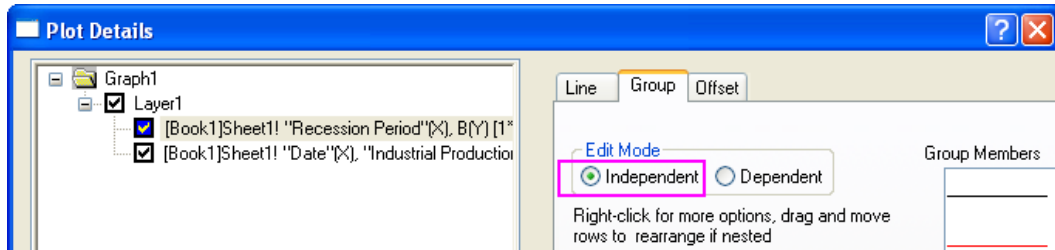
What you will learn

- Plot a line graph
- Use **Span Vertical Bars between Missing Values** to show recession bars
- Set date display format in the **Axes** dialog

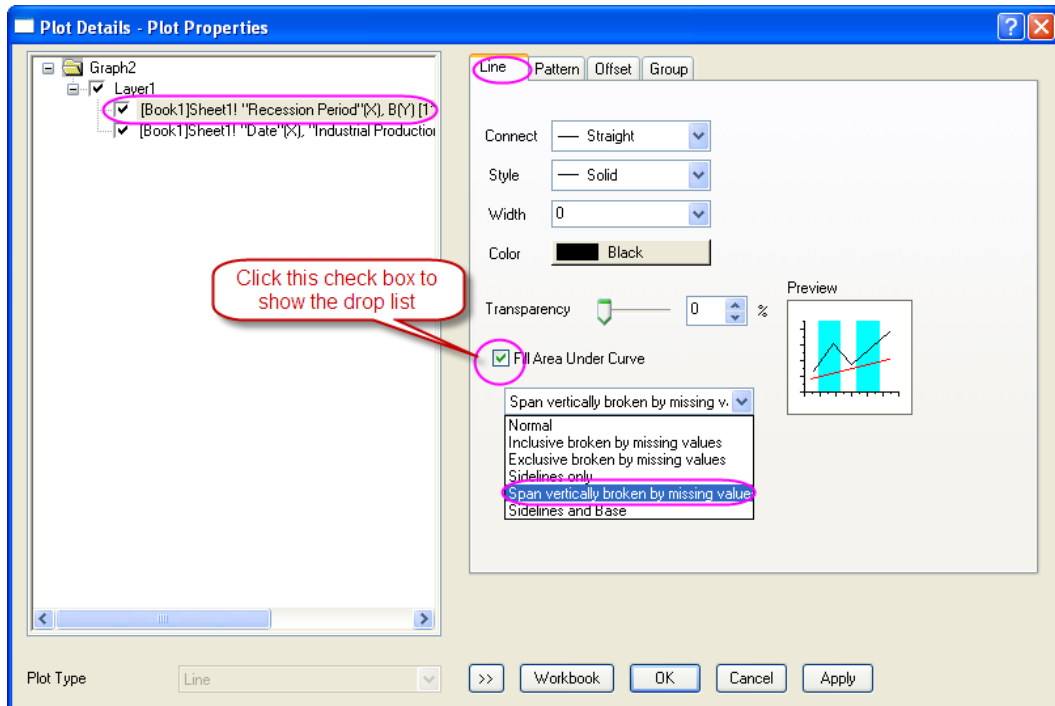
Steps

This tutorial is associated with the **Graphing: Line and Symbol: Recession Bars** folder in the Sample project (\Samples\Samples.opj).

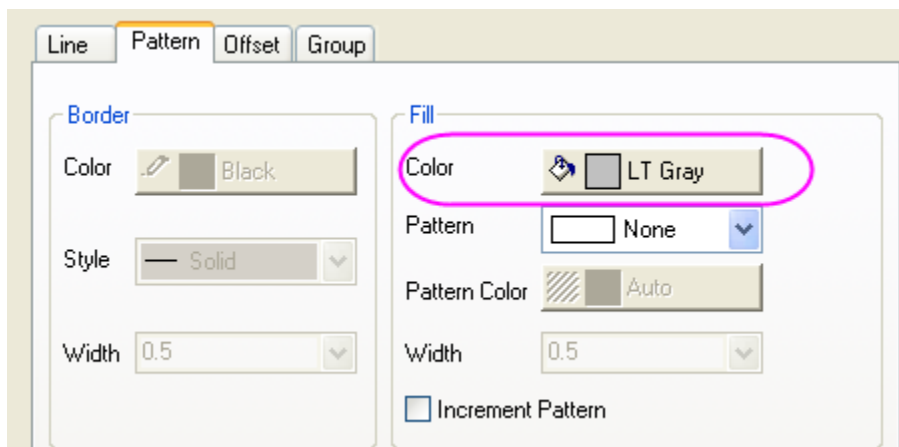
1. Open the Recession Bar folder in Sample.opj and active the workbook **Book1G**.
2. Highlight all four columns in the worksheet and then right-click to select **Set as: XY XY** from the context menu. After that, choose "**Plot: Line**" menu to plot a line graph.
3. Double-click the plot to show the **Plot Details** dialog box. Select the **Group** tab, and select **Independent**.



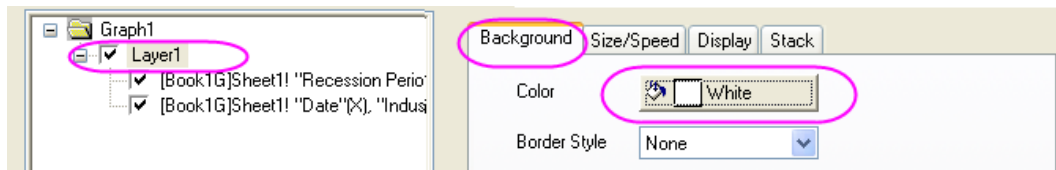
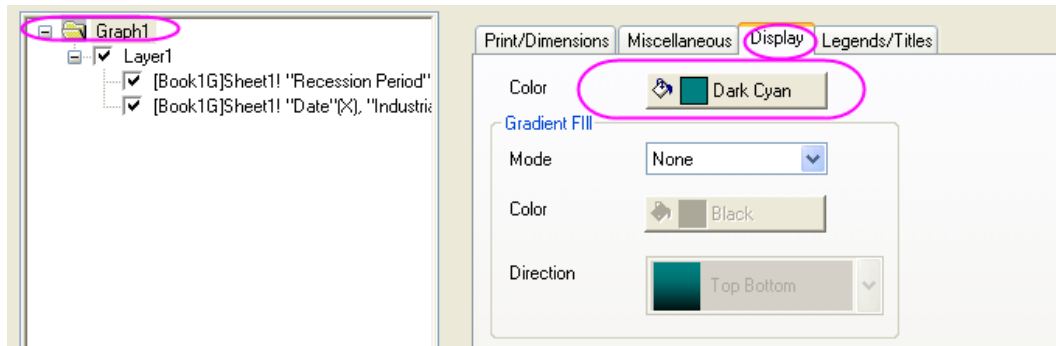
4. In the left panel of the dialog box, choose the first plot node of recession data (A(X):B(Y)). In the **Line** tab, choose **Span Vertical Bars between Missing Values** from **Fill Area Under Curve** drop down list.



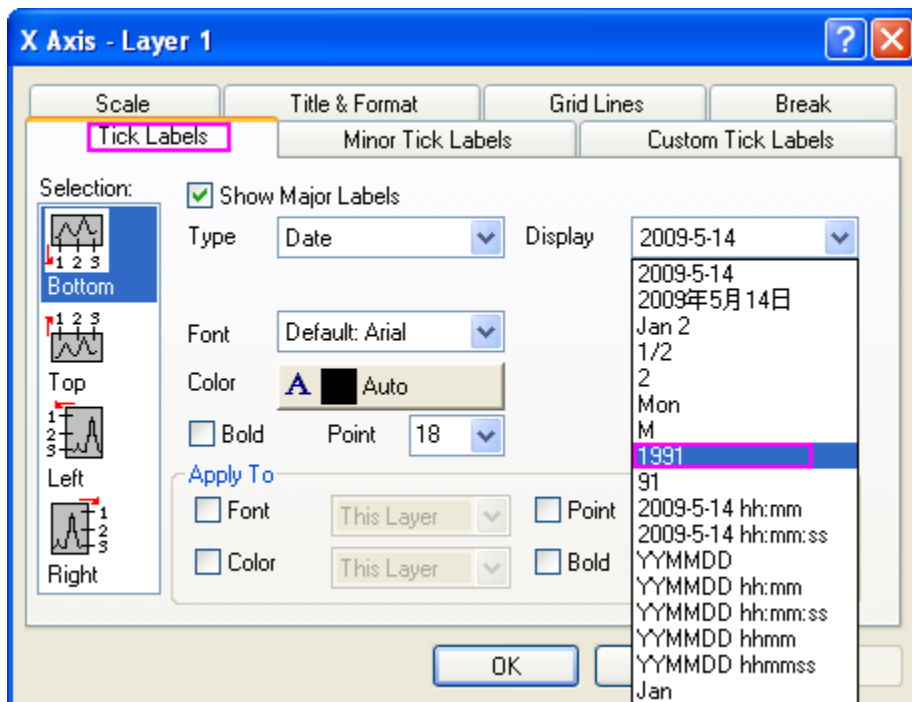
5. Go to the **Pattern** tab, set **Color** in the **Fill** group as **LT Gray**. Click **Apply** button to apply these settings.



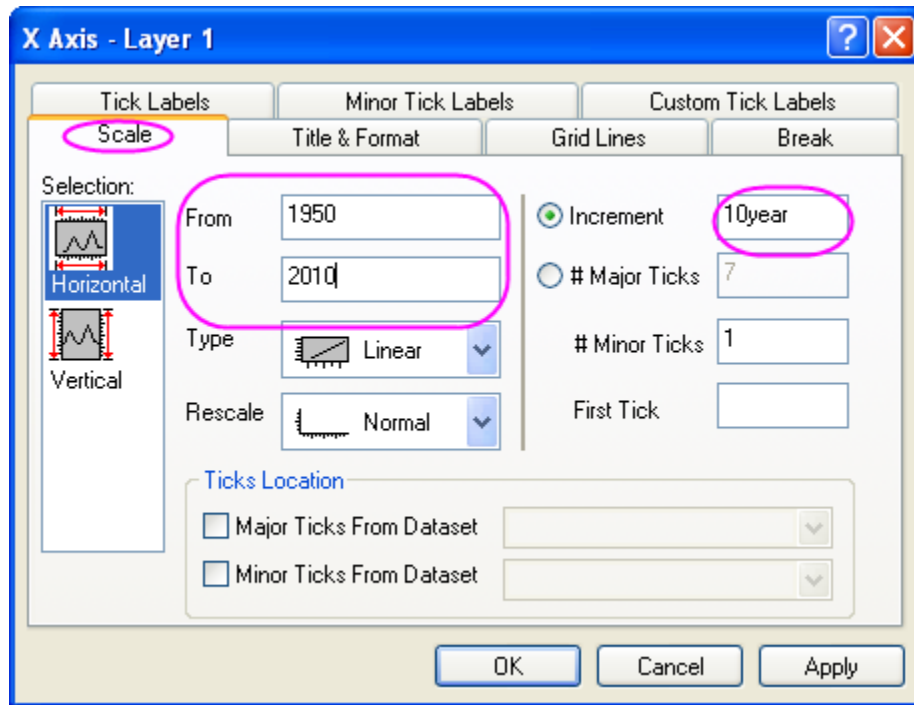
- After that, go to the Graph level, and set **Color** in the **Display** tab as **Dark Cyan**. Then go to the Layer level, set the **Color** in **Background** tab as **White**. Click **OK** button to close this dialog.



- Double-click the X axis to open the **Axes** dialog. Go to **Tick Labels** tab to change the **Display** item as 1991.



- Then go to Scale tab to set the **From**, **To** and **Increment** item as 1950, 2010 and 10. Click **OK** button.

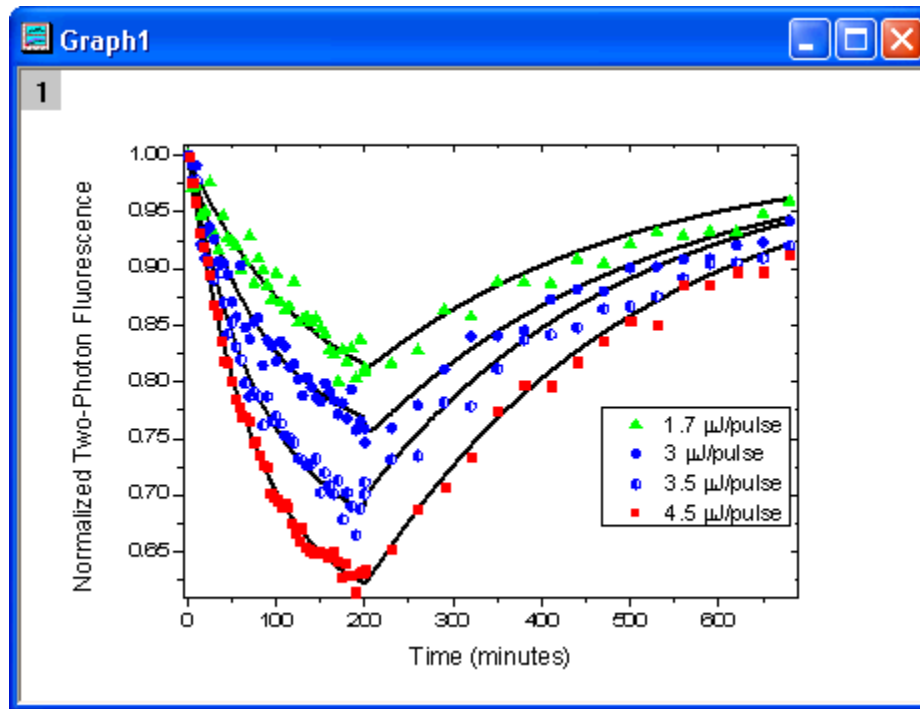


9. Delete the Legend for this graph and add the the title for Y axis. After all, you will get the final graph.

3.2.3 Scatter Plot of Decay and Recovery Curves

Summary

The scatter plot below depicts 3 decay and recovery curves obtained after taking two-photon fluorescence measurements of reversible photodegradation in a dye-doped polymer. To learn more about the graph, please read the case study.




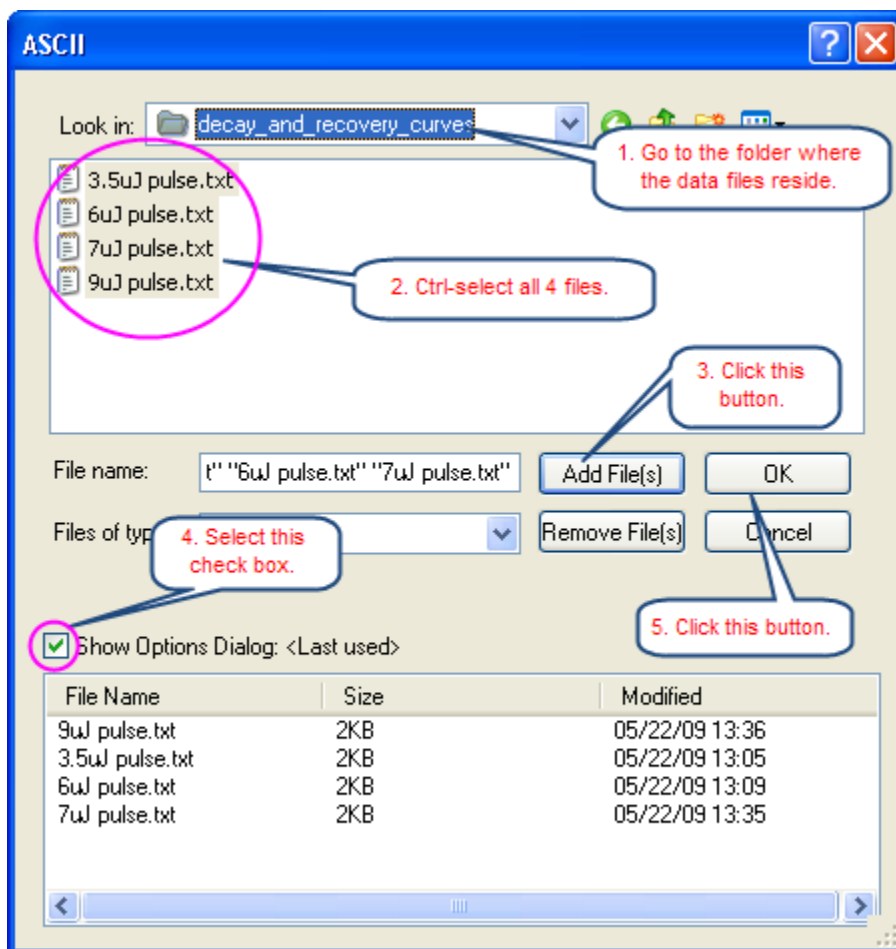
Minimum Origin Version Required: Origin 8.5.1 SRO

What you will learn

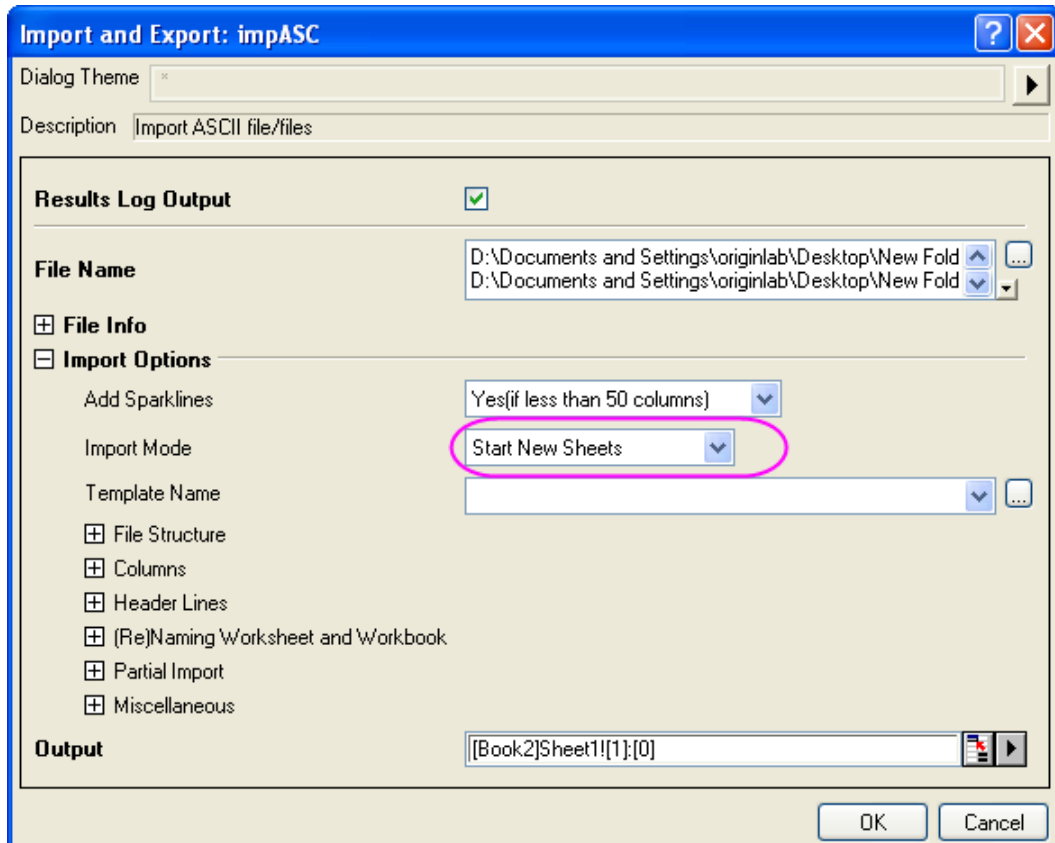
- How to use the **Plot Setup** dialog to arrange plots in a layer
- How to customize the symbols in your graph




Steps

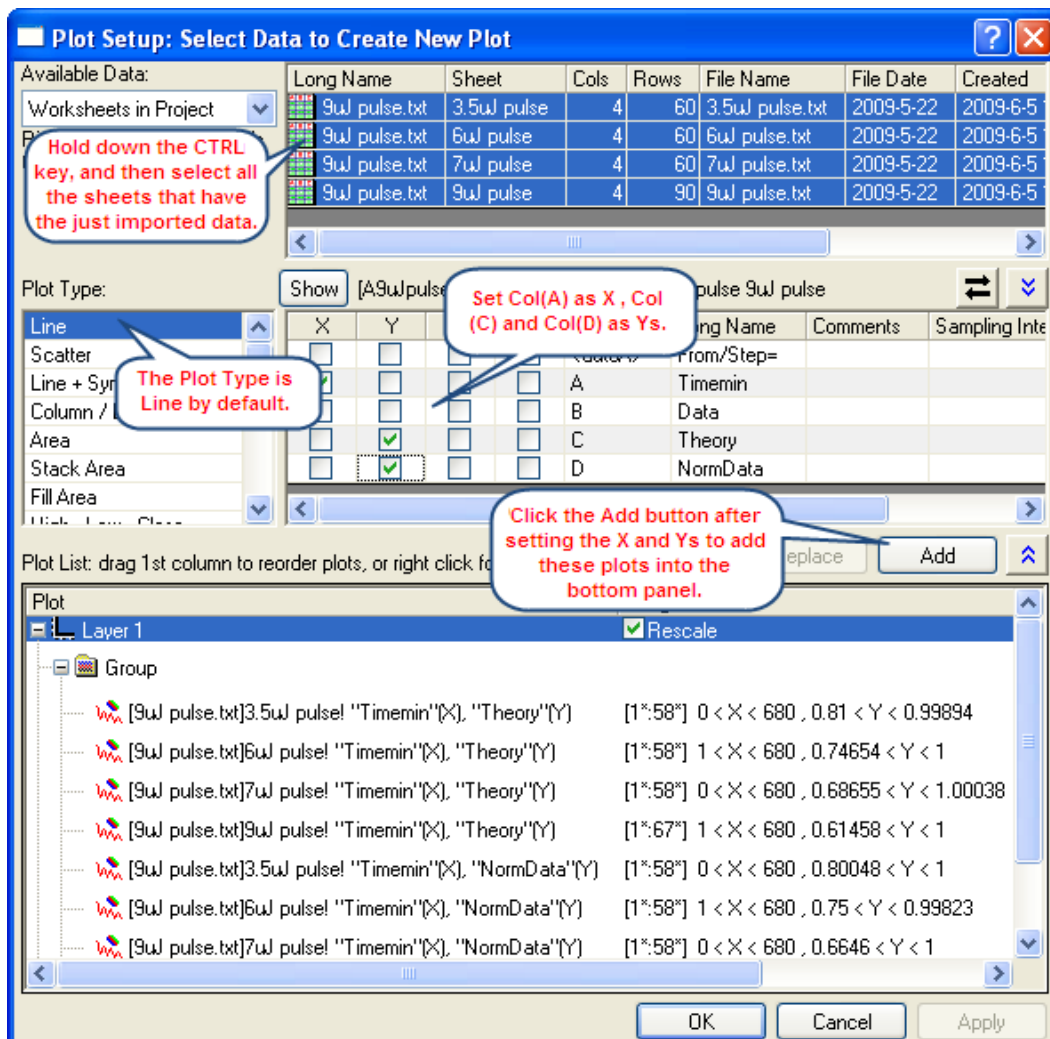
1. Download the zip file from here and extract the text files .
2. Open Origin and click the **Import Multiple ASCII** button  on the **Standard** toolbar to open the **ASCII** dialog and then import the text files.



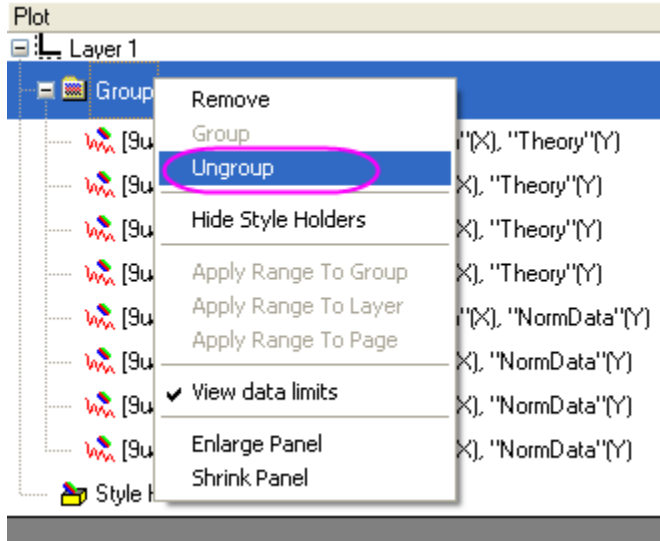
- In the **impASCII** dialog, set **Import Mode** as **Start New Sheets**. Click **OK** to finish importing.



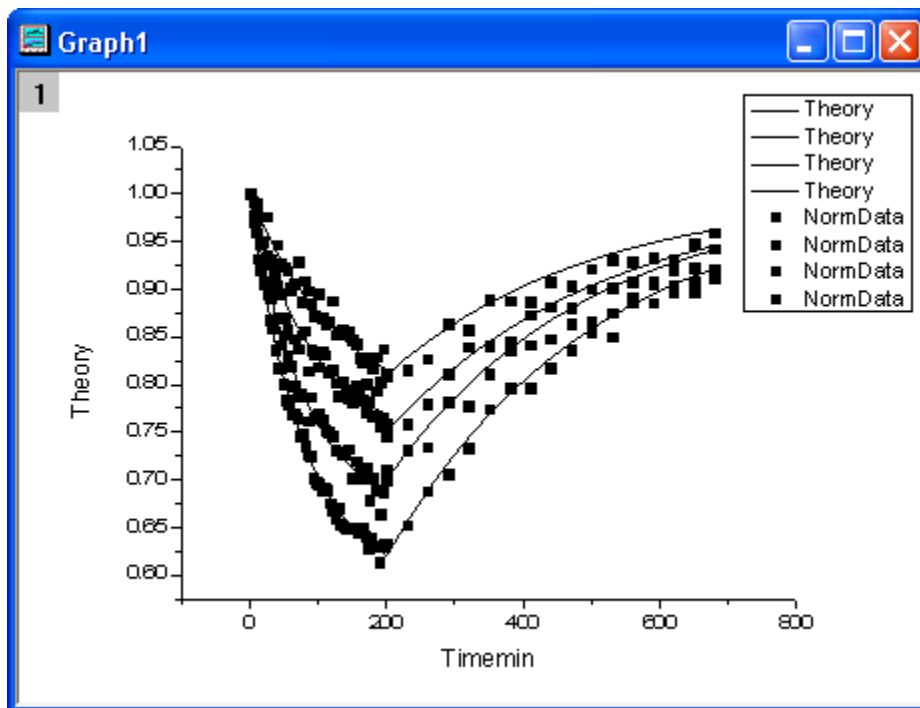
4. We will use the Plot Setup dialog to create a graph with 8 plots. Active the workbook, and make sure that no datasets are now selected. Click the  button on the **2D Graphs** toolbar to open the **Plot Setup** dialog.
 - Show all of the three panels of **Plot Setup** dialog (if not all of them are shown) by clicking the  and  buttons.
 - First, we will add 4 line plots into a graph by using the Plot Details dialog. Highlight all dataset in the top panel, and then select column **Timemin** as X, column **Theory** as Ys in the middle panel. Then add them into the bottom panel.
 - Then we will add 4 scatter plots in to the same graph. Select **Scatter** from the **Plot Type** drop-down list, make sure all dataset in the top panel has been selected, and then select column **Timemin** as X, column **NormData** as Ys in the middle panel. Then add them into the bottom panel.



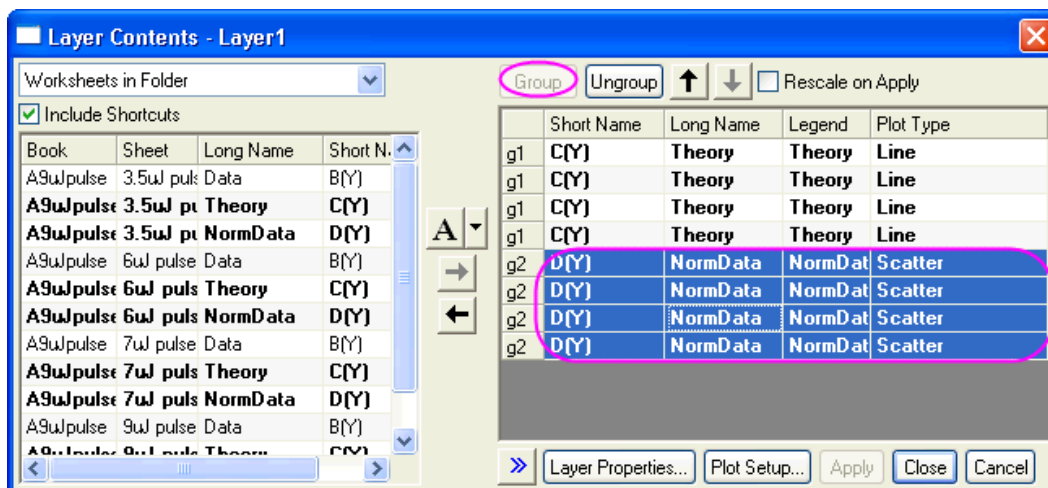
- o In bottom panel, if there is a **Group** branch under **Layer1**, right-click on it to select **Ungroup** from the short-cut menu to ungroup these plots.



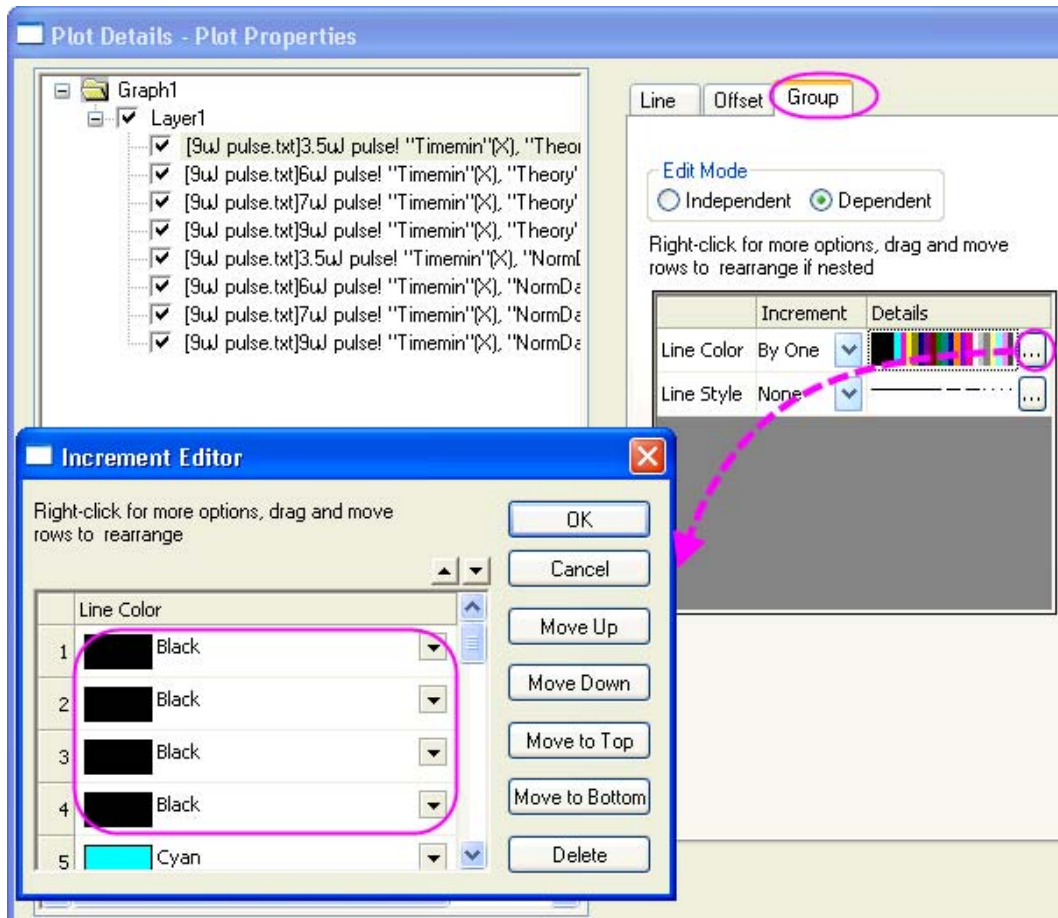
- o Click **OK** to generate the graph which look like the following image shows.






5. Double-click the layer icon at the top-left corner of graph window to open the **Layer Contents** dialog. Then group the **Theory** plots and **NormData** Plots as Group 1 and Group 2 using the **Group** button after highlight the 4 Theory/Norma Data plots separately by using mouse and the **Shift** key,

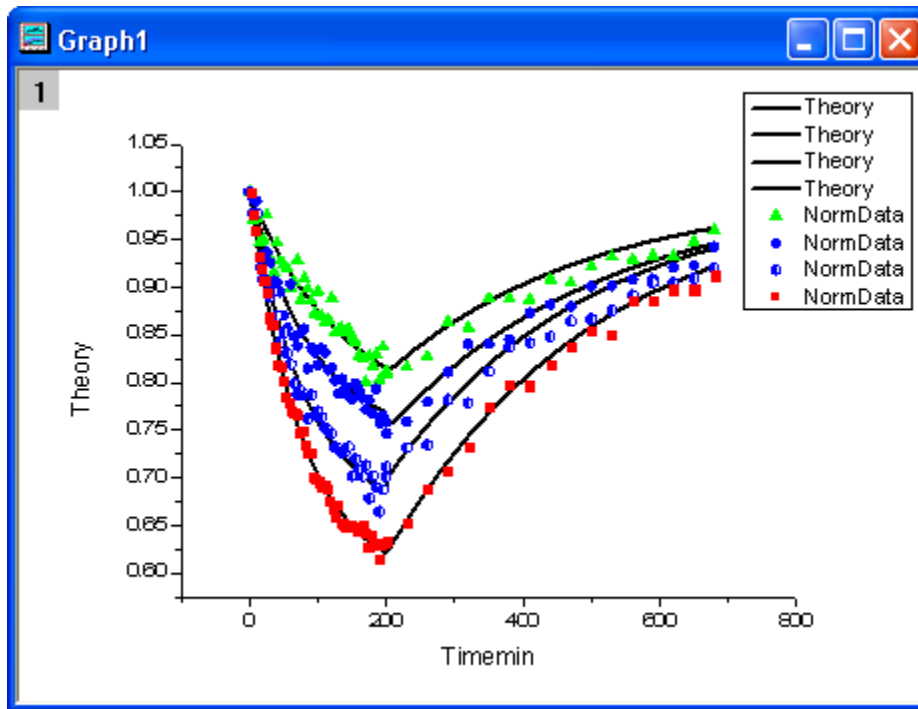


- Then we will customize the 8 plots in the Plot Details dialog. Select **Format: Plot Properties** to open the Plot Details dialog. In the left panel of the dialog, you could see there are 8 plots: the first 4 plots contain a word *Theory* are line plots, the other 4 contain *NormData* are scatter plots.
- We will first customize the 4 line plots. Select the first line plot, in the **Line** tab, select **B-Spline** from the **Connect** drop down list and set the **Width** to **3**. Then click the **Apply** button. In the **Group** tab, click the button that next to the **Line Color** as the screenshot shows to change the color of the 4 lines to **Black**. Click OK to apply these settings.



8. Then customize the 4 scatter plots. Select the first Scatter plot, in the **Symbol** tab, change the **Size** to 8.
 - o In the **Group** tab, we will mainly customize the symbols in the list box that in the middle of the tab. In the **Symbol Type** row, select **By one** in the **Increment** column. Click the  button to open the **Increment Editor** dialog, in the dialog select **UpTriangle**, **Circle**, **Hexagon** and **Square** for the first 4 rows.
 - o In the **Symbol Edge Color** row, select **By one** in the **Increment** column. Click the  button to open the **Increment Editor** dialog, in the dialog select **Green**, **Blue**, **Blue** and **Red** for the first 4 rows.
 - o In the **Symbol Interior** row, select **By one** in the **Increment** column. Click the  button to open the **Increment Editor** dialog, in the dialog select **Solid**, **Solid**, **Half Left** and **Solid** for the first 4 rows.

Click OK button to close the Plot Details dialog, then the graph will look like.



9. Then customize the Axes of the graph. Double click on the X axis, in the **Scale** tab, change **From** to **-5**, **To** to **690**. Then click **Vertical** icon in the **Selection** box, change **From** to 0.61 and **To** to **1.01**. In the **Title&Format** tab, select **Top** in the Selection box, enable the **Show Axis&Tick** check box, and select **In** from both **Major Ticks** and **Minor Ticks** drop-down lists. Then select **Right** from the **Selection** box, and do the same thing as we did for the Top.

10. Then we will customize the titles and the legend. Change the titles as the following images shows. Right click on the legend and select **Properties** from the context menu, and input the following strings into it.

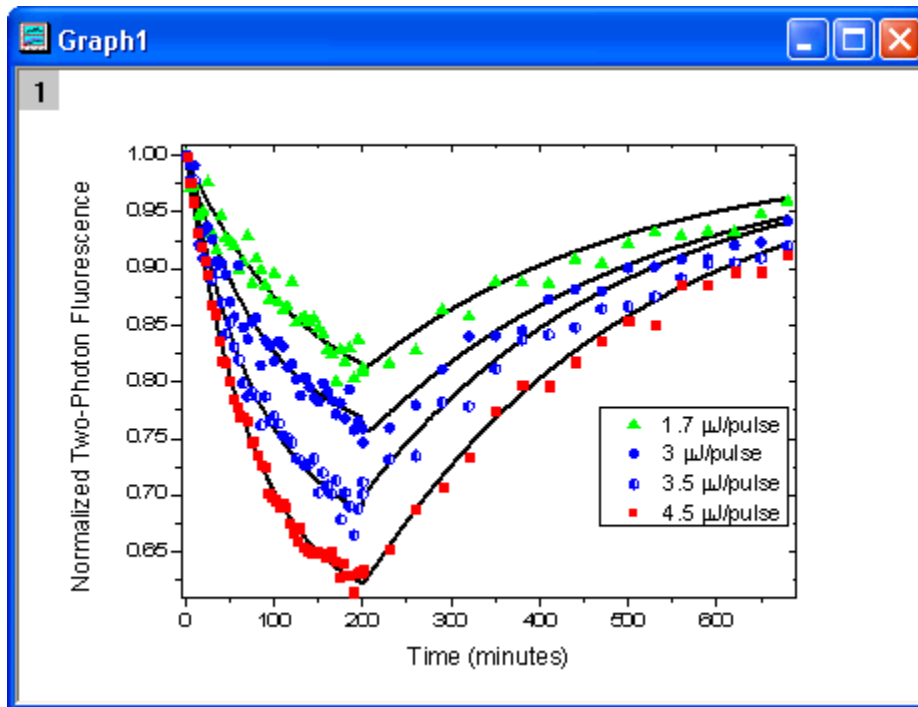
\(5) 1.7 \text{ g(m)J/pulse}

\(6) 3 \text{ g(m)J/pulse}

\(7) 3.5 \text{ g(m)J/pulse}

\(8) 4.5 \text{ g(m)J/pulse}

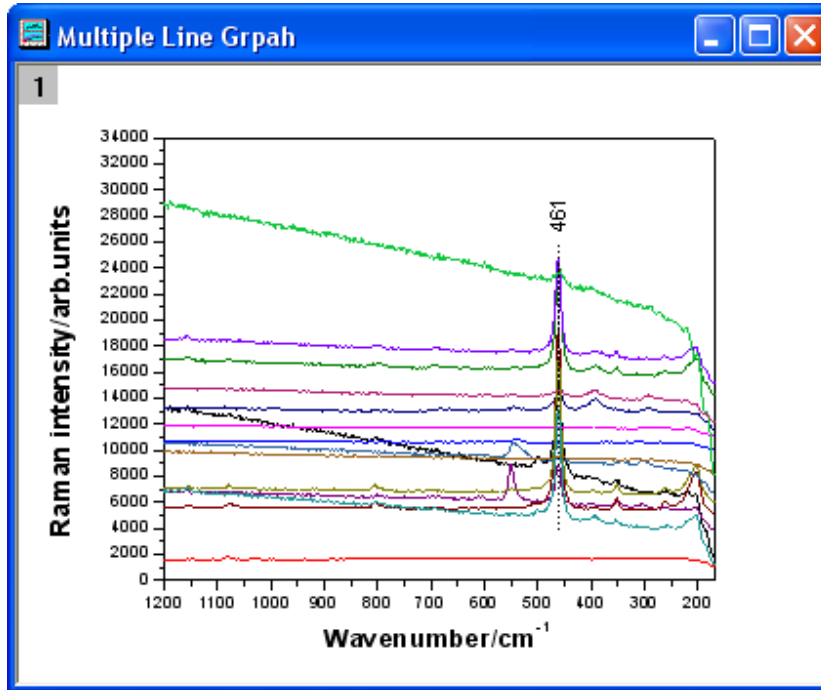
The final graph will look like



3.2.4 Micro-Raman spectroscopy of complex nanostructured mineral systems

Summary

This tutorial will show you how to create a multiple line plot and how to customize it.



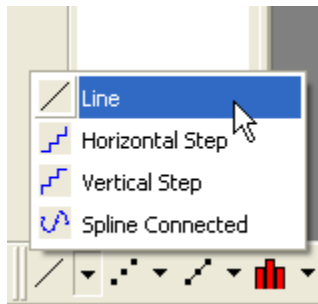
Minimum Origin Version Required: Origin 8.5.1

Steps

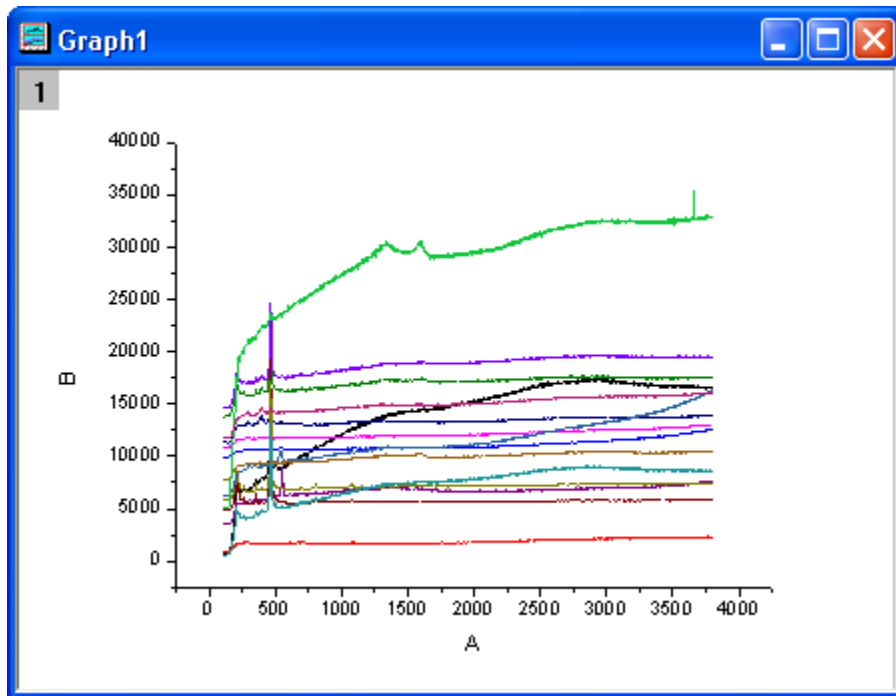
1. Create a new worksheet. Import the binned data.

	A(X)	B(Y)	C1(Y)	C2(Y)	C3(Y)	C4(Y)	C5(Y)	C6
Long Name								
Units								
1	100.263	762.24	100.263	947.9676	100.263	9966.063	100.263	10919
2	101.997	765.94	101.997	951.6754	101.997	9940.0309	101.997	10926
3	103.728	771.5	103.728	944.2353	103.728	9954.8876	103.728	10911
4	105.461	765.92	105.461	957.2229	105.461	9936.3143	105.461	10926
5	107.194	769.61	107.194	936.8001	107.194	9951.1572	107.194	10919
6	108.927	767.74	108.927	944.2147	108.927	9956.7125	108.927	10926
7	110.657	767.73	110.657	959.035	110.657	9943.7279	110.657	10931
8	112.389	784.39	112.389	955.3154	112.389	9969.6578	112.389	10921
9	114.121	760.3	114.121	955.3027	114.121	9945.5715	114.121	10921
10	115.85	762.14	115.85	951.589	115.85	9952.9684	115.85	10931
11	117.581	765.83	117.581	940.4814	117.581	9945.5616	117.581	10926
12	119.312	773.21	119.312	947.8704	119.312	9928.9203	119.312	10936
13	121.041	763.95	121.041	962.6417	121.041	9952.9419	121.041	1091
14	122.771	767.63	122.771	934.9271	122.771	9945.5469	122.771	10921
15	124.501	771.31	124.501	942.3073	124.501	9951.0786	124.501	10921
16	126.228	784.21	126.228	938.6123	126.228	9947.3817	126.228	10921
17	127.957	765.75	127.957	945.9827	127.957	9938.1579	127.957	10921
18	129.686	774.95	129.686	955.1878	129.686	9963.9533	129.686	10926
19	131.413	762.04	131.413	947.8086	131.413	9945.5224	131.413	10931

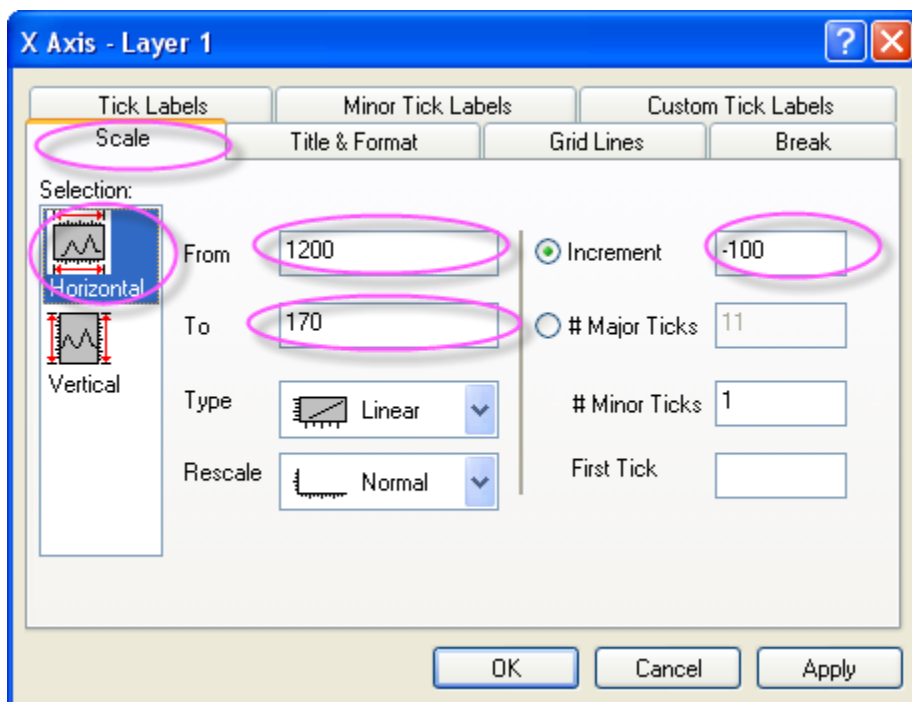
- Highlight all the columns in the worksheet. Right-click and select **Set As: XY XY** from the context menu. Then click the **Line** button on the **2D Graphs** toolbar.



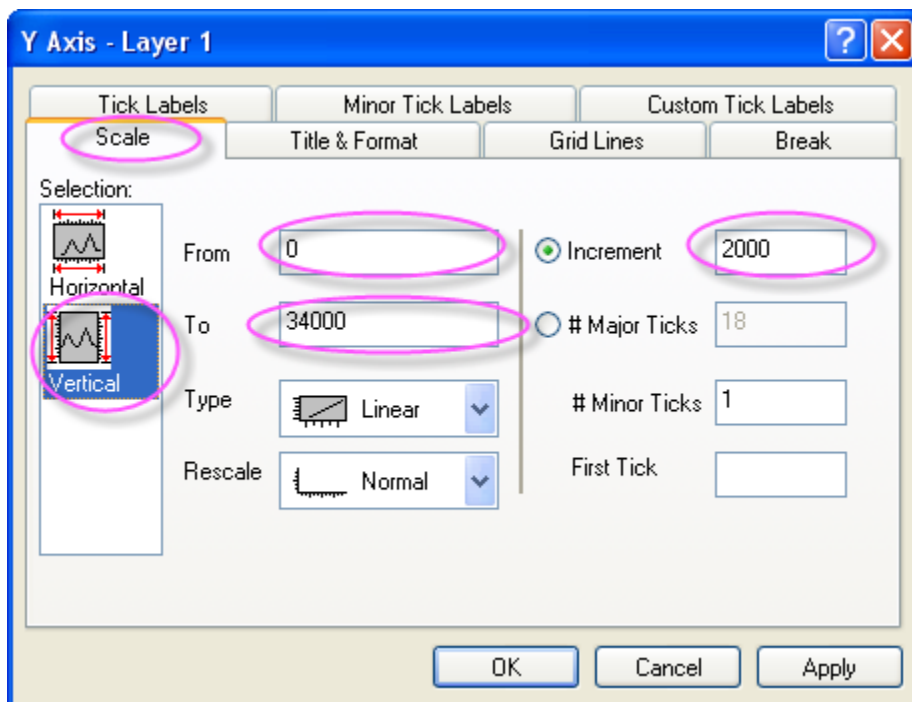
Delete the legend. The graph should look like:



- Double-click on the **X** axis to bring up the **X Axis** dialog box. In the **Scale** tab, select **Horizontal** in the **Selection** list box. Then set the options as the screenshots below.

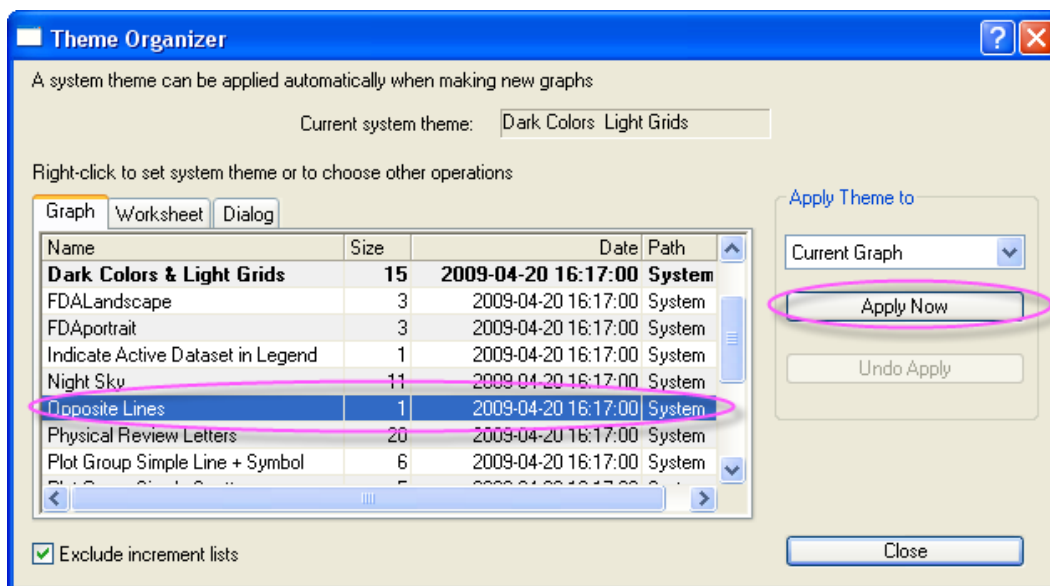


Select **Vertical** in the **Selection** list box. Then set the options as the screenshots below.



Click **OK**.

- Then we apply a graph theme to add a top X axis and a right Y axis. Select **Tool: Theme Organizer** to open the **Theme Organizer** dialog. Activate the **Graph** tab and select **Opposite Lines** from the table. Then click the **Apply Now** button. Click the **Close** button to close the dialog.



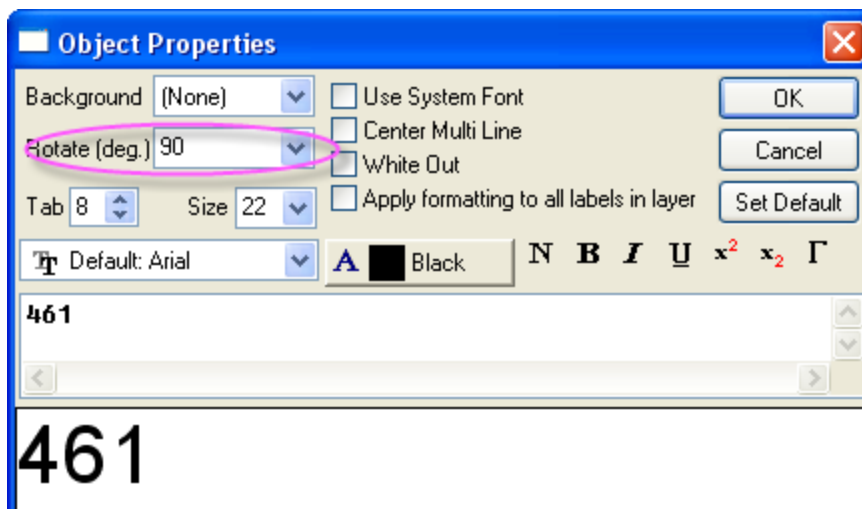
- Click the **Line** button on the **Tools** toolbar and draw a line across the peaks' centers as the sample image shows. Please hold down the **SHIFT** key while drawing the line to force it to be a vertical line. Double-click on the line. In the **Line** tab, select **Dash** with the **Type** drop-down list. Click **OK**.



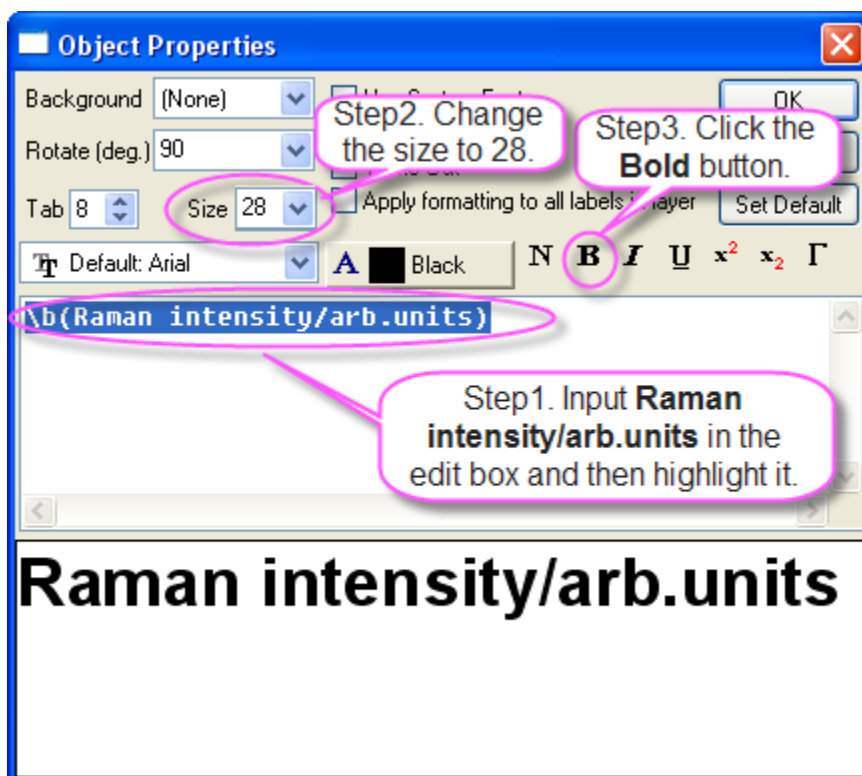
- Click the **Text** button in the **Tools** toolbar. Add a text object near the line object and enter **461** in it.



Right-click on the label and select **Properties** from the short-cut menu. Then set the dialog options as the screenshot below. Click **OK**.



- Right-click on the title of the Y axis and select **Properties** from the short-cut menu. Then set the dialog options as the screenshot below. Click **OK**.



- Right-click on the title of the X axis and select **Properties** from the short-cut menu. Then set the dialog options as the screenshot below. Click **OK**.

Object Properties

Background: [] Use []
 Rotate (deg.): [] Center [] White []
 Tab: [8] Size: [28] Apply formatting to all labels in layer [] Set Default []

Font: Default: Arial [v] A [] Black [] N [] **B** [] *I* [] U [] x^2 [] x_2 [] Γ []

Step 2. Set the Size to 28.

Step 4. Just select -1 and click **SuperScript** button.

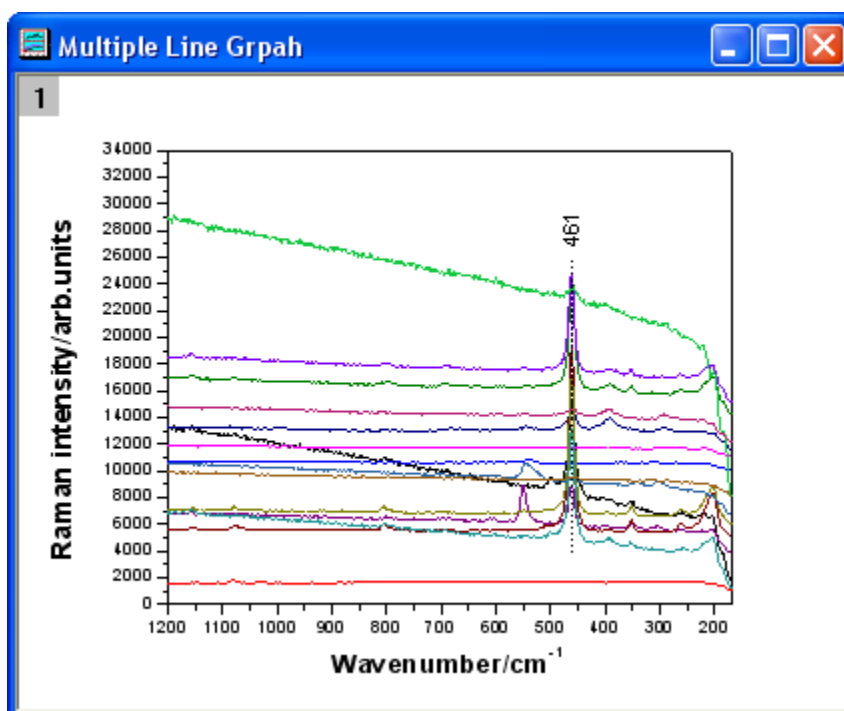
Step 3. Click this button.

Step 1. Input Wavenumber/cm-1 in the edit box and highlight it.

$\backslash b (Wavenumber/cm \backslash + (-1))$

Wavenumber/cm⁻¹

The graph should look like



Sample Data

Download the **Micro_Raman_Spectroscopy.txt** file from http://www.originlab.com/ftp/graph_gallery/data/Micro_Raman_Spectroscopy.txt. Click the **Import**

Single ASCII button  and select the file to import it into Origin.

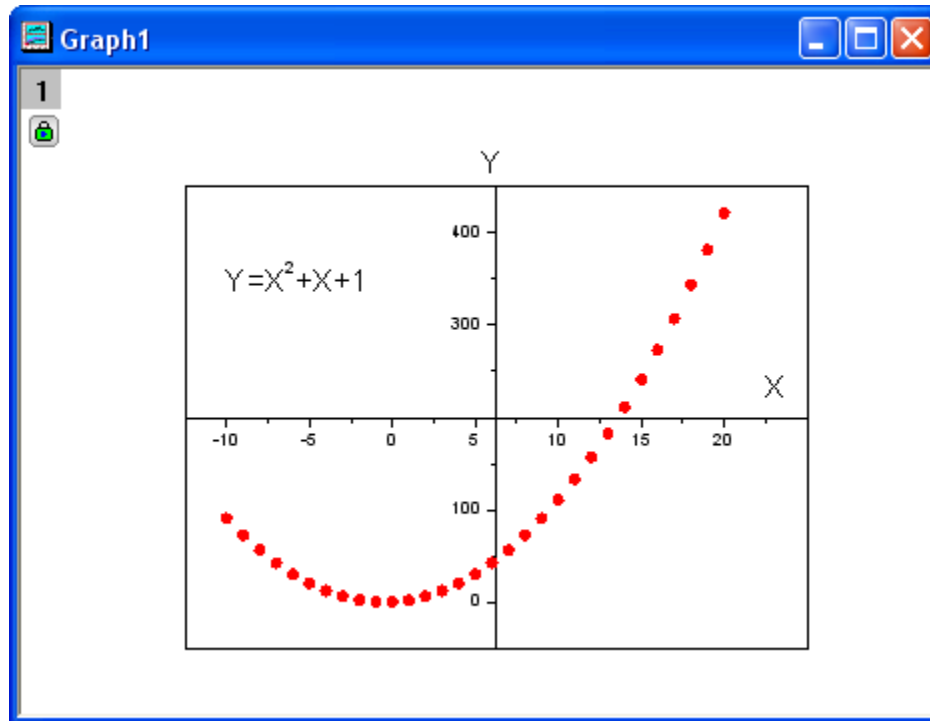
The following table contains part of the sample data.

A(X1)	B(Y1)	A(X2)	B(Y2)	A(X3)	B(Y3)
100.263	762.24	100.263	947.9676	100.263	9966.063
101.997	765.94	101.997	951.6754	101.997	9940.0309
103.728	771.5	103.728	944.2353	103.728	9954.8876
105.461	765.92	105.461	957.2229	105.461	9936.3143
107.194	769.61	107.194	936.8001	107.194	9951.1572
108.927	767.74	108.927	944.2147	108.927	9956.7125
110.657	767.73	110.657	959.035	110.657	9943.7279
112.389	784.39	112.389	955.3154	112.389	9969.6578
114.121	760.3	114.121	955.3027	114.121	9945.5715
115.85	762.14	115.85	951.589	115.85	9952.9684

3.2.5 Scatter Central Plot

Summary

Scatter central plot is a graph with the X and Y axes located in the middle of the layer. In this tutorial, a scatter central plot will be created, then the symbol and the axes will be customized.



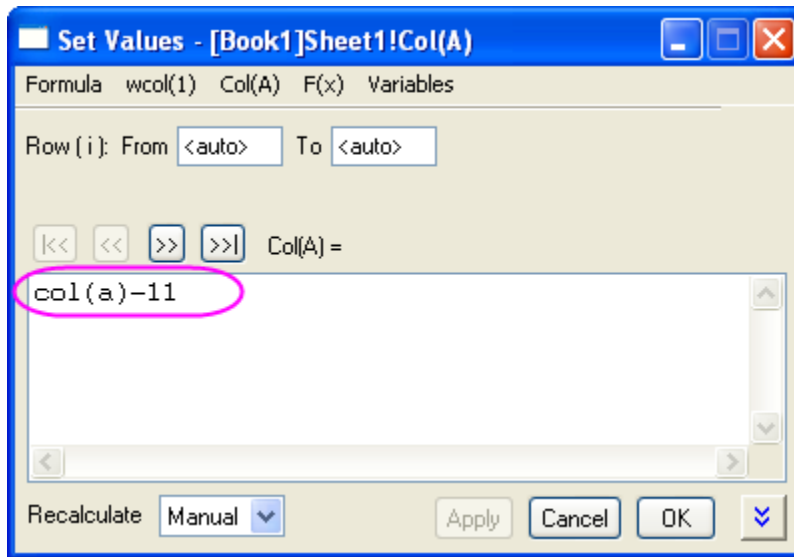
Minimum Origin Version Required: Origin 8.1 SRO

What you will learn

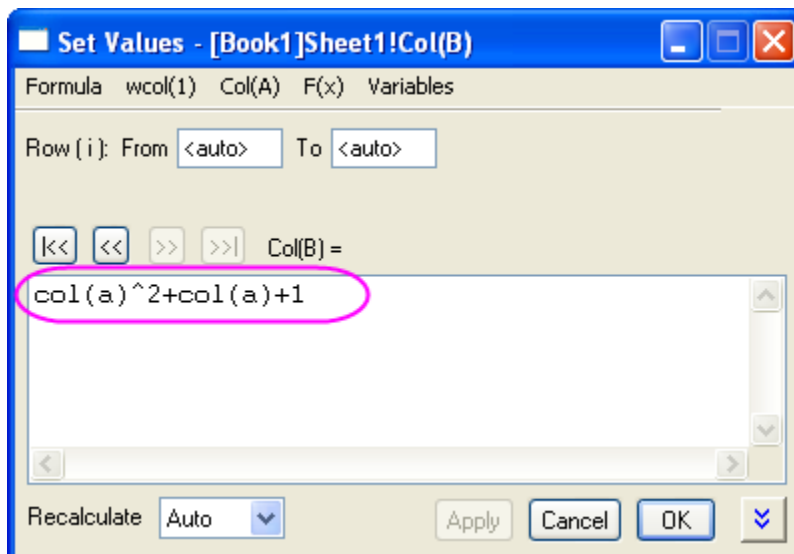
- Set column values by using **Set Values** dialog
- Create a scatter central plot
- Change the color and shape of the scatter points
- Customize the tick label of axis

Steps

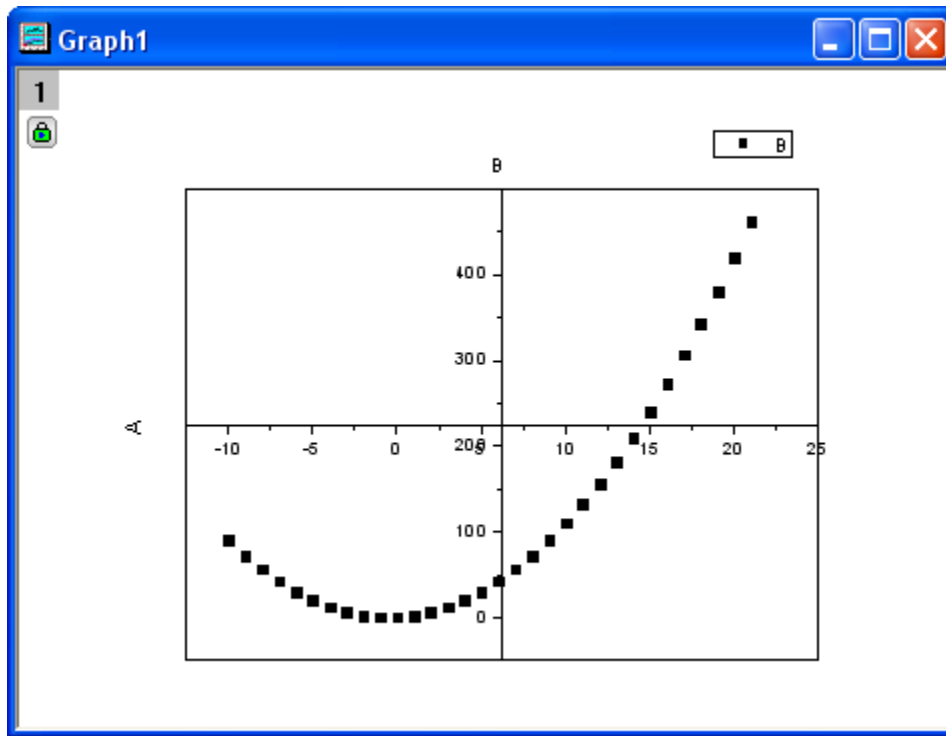
1. Set column values by using **Set Values** dialog. Highlight column A, right-click and select **Fill Column with: Row Numbers** from the context menu. Right-click column A again and select **Set Column Values** to open the **Set Values** dialog. Type **Col(a) - 11** in the middle text box and then click the **OK** button to finish setting values for column A.



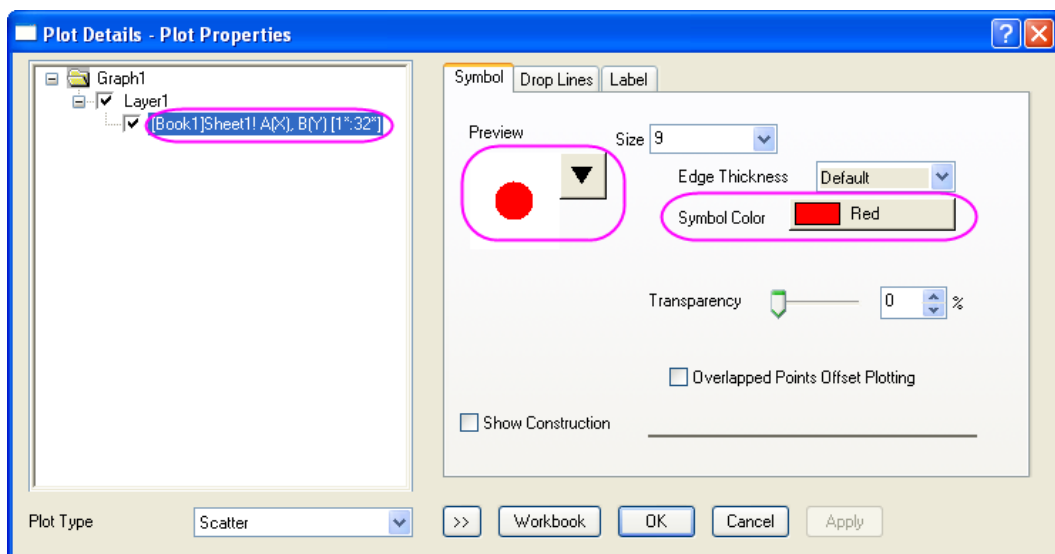
- Do the same thing for column B, highlight it and select **Set Column Values** dialog from the right-clicking menu to open the **Set Values** dialog again. Type $\text{Col(a)}^2 + \text{Col(a)} + 1$ in the middle text box this time. Change the **Recalculate** to **Auto**, then click the **OK** button to finish setting values for column B.



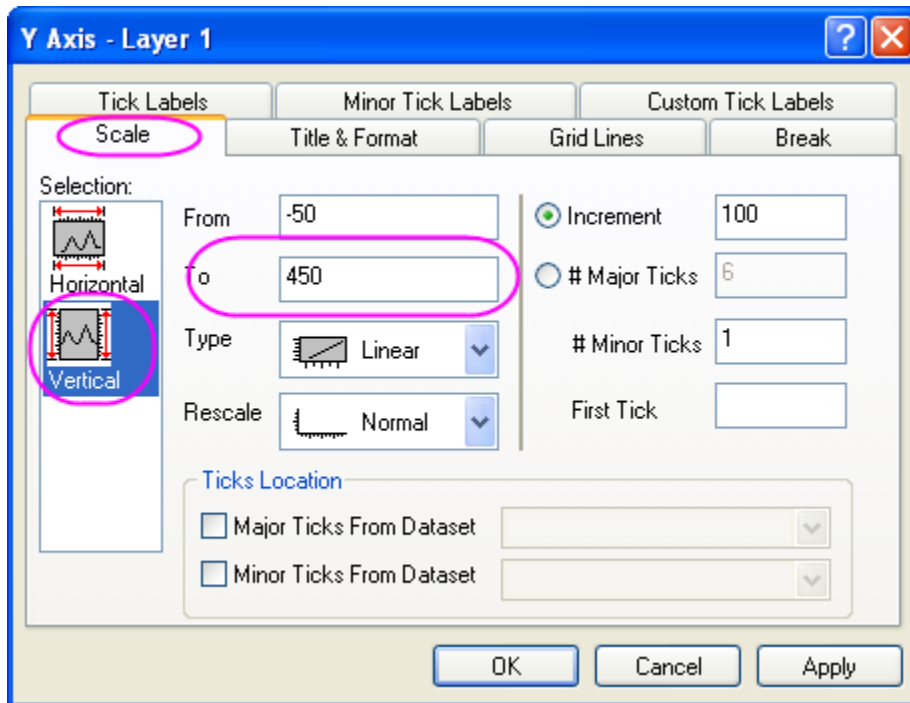
- Create a scatter central plot. highlight column B and select **Plot: Symbol: Scatter Central** from the Origin main menu to create a scatter central plot.



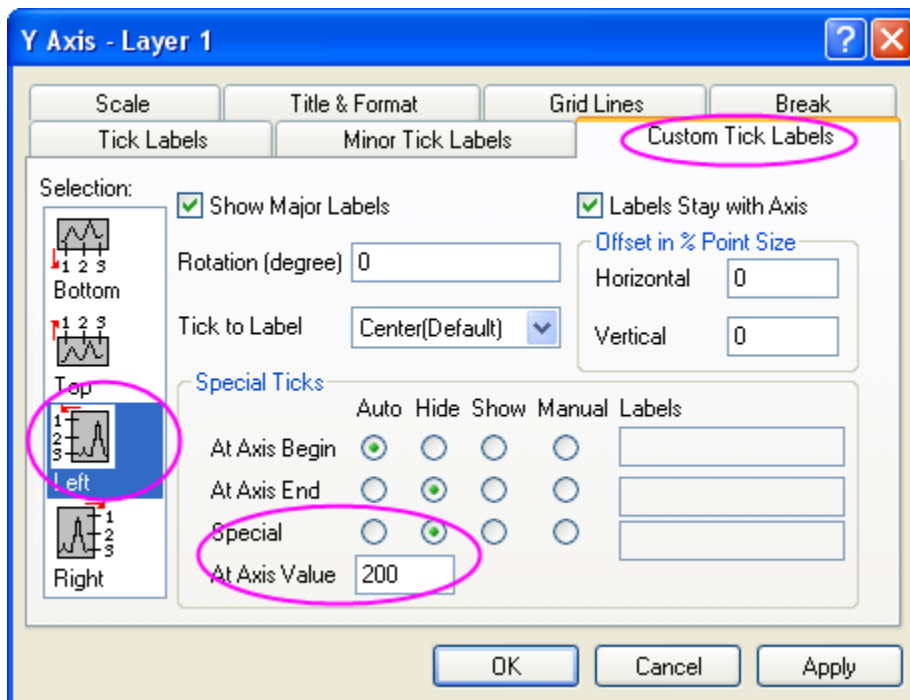
- Change the color and shape of the scatter points. Double click on the scatter plot to open the **Plot Details** dialog. In the **Symbol** tab of right panel, click the button beside **Preview** and choose the solid circle symbol. And then click the button beside **Symbol Color** and choose **Individual Color: Red**. Click the **OK** button to close the **Plot Details** dialog.



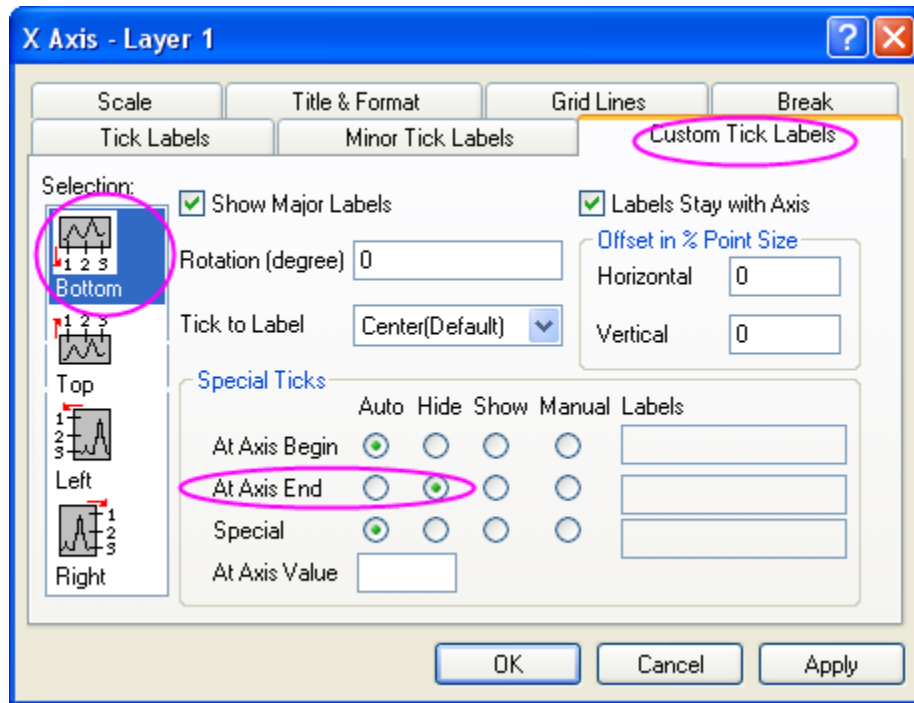
- Customize the tick label of axis. Double click on the vertical axis to open the dialog for axes settings. Change the **To** value to 450.



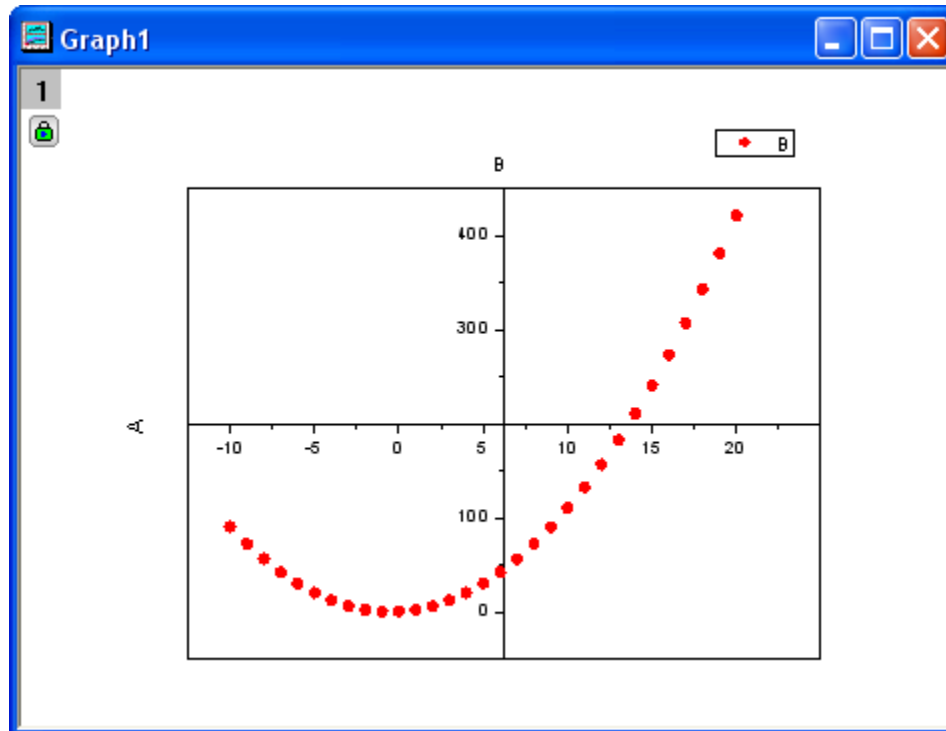
6. Switch to the **Custom Tick Labels** tab, select the **Hide** radio on the **Special** row and then type **200** in the **At Axis Value** box.



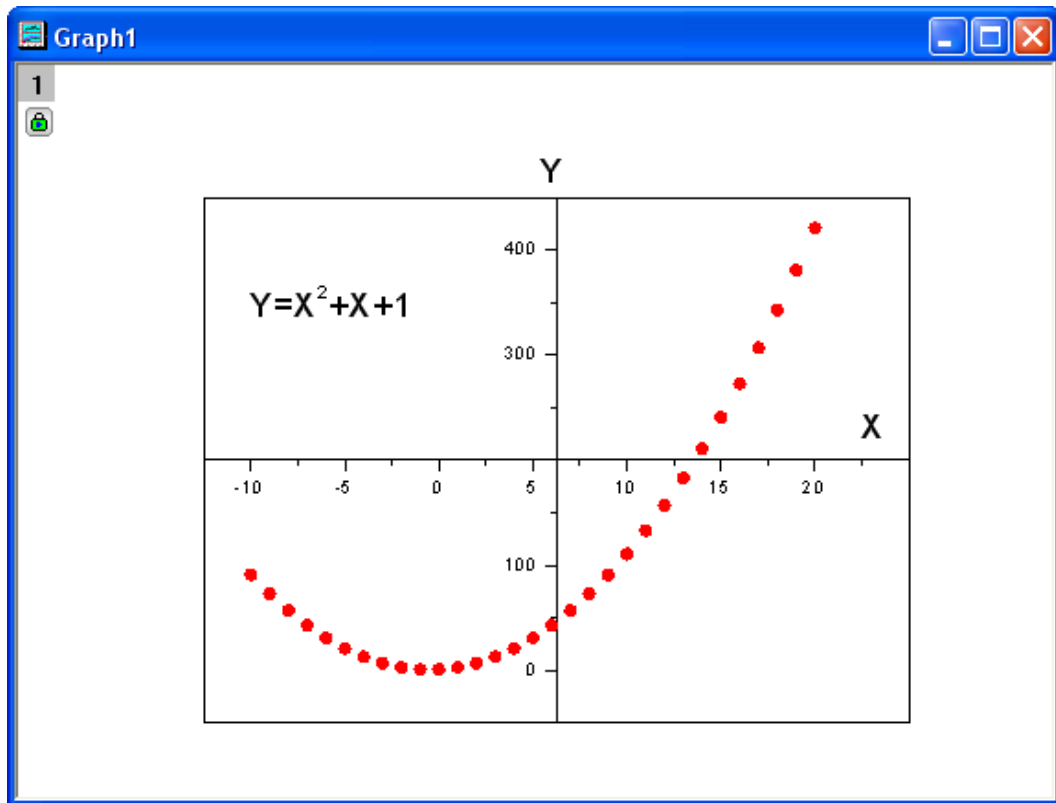
7. Click the **Bottom** option in the **Selection** panel, and then select the **Hide** radio on the **At Axis End** row.



8. Click the **OK** button to finish axes settings and the scatter central graph looks like below.



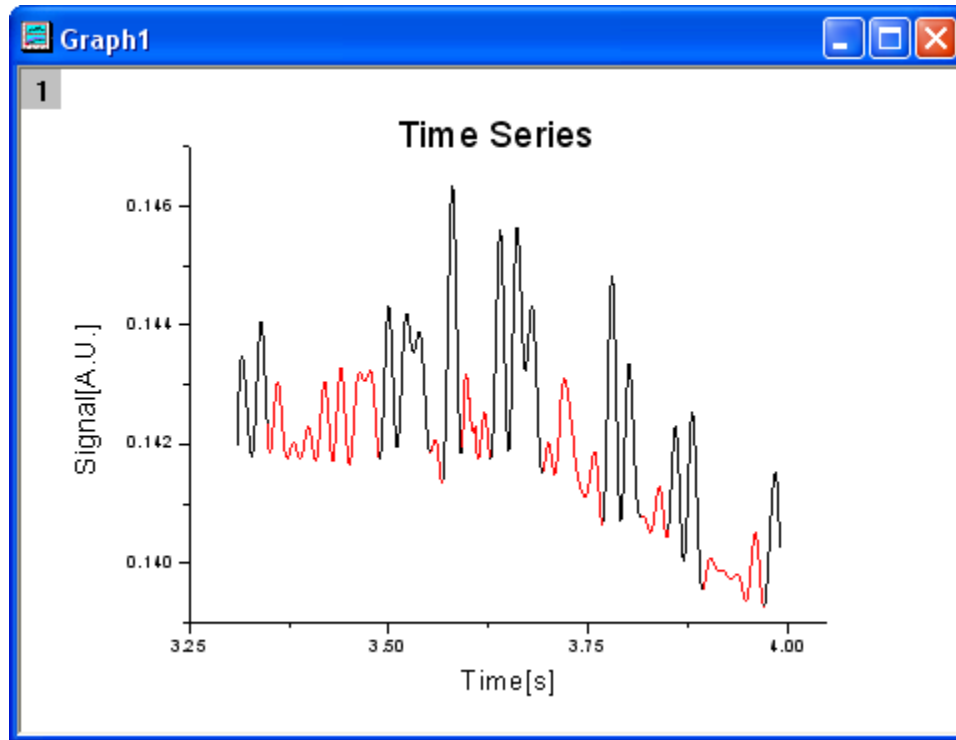
9. Now, delete the legend and change the label of axes, **A** to **X** and **B** to **Y**. Move the **X** to the end of X axis. Add a text object to the graph, the text says $Y = X^2 + X + 1$. The final graph looks like below.



3.2.6 Line Graph with Masked Data

Summary

In a line graph, masking data is supported. In this tutorial, a line graph will be created first, then the masking tool is used to mask the low peaks on the graph.




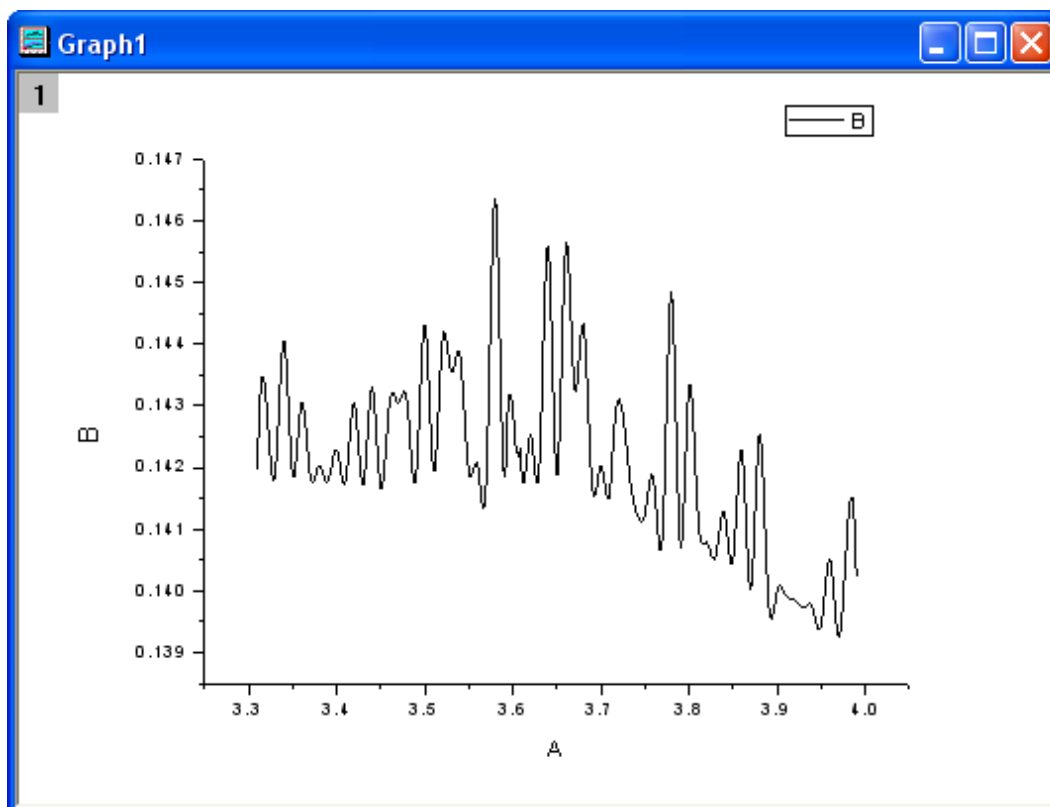
Minimum Origin Version Required: Origin 8.1 SRO

What you will learn

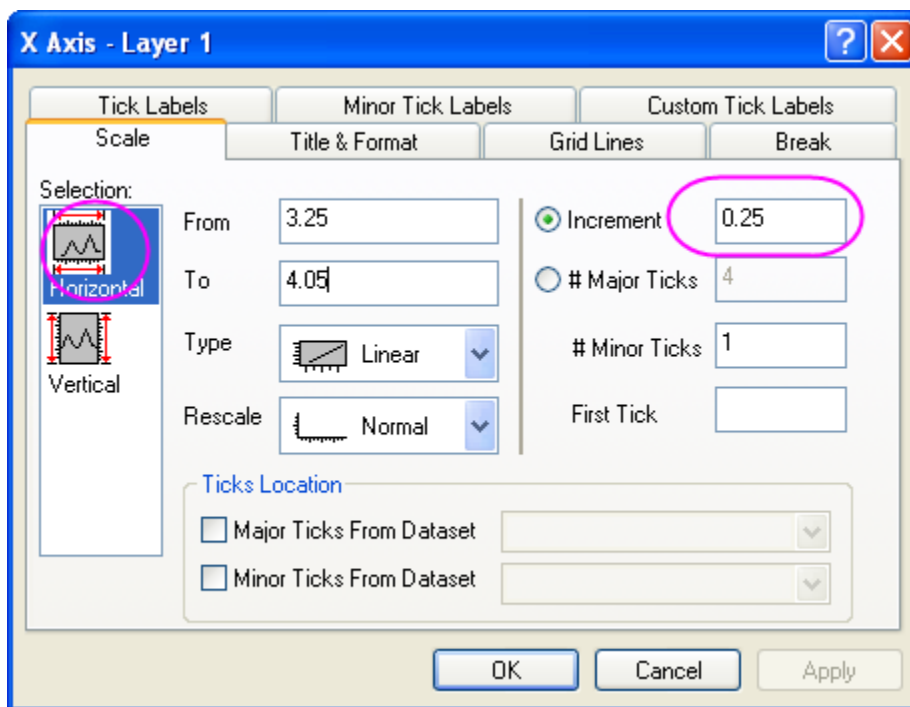
- Plot a line graph
- Set the scales of axes
- Mask data on the line graph

Steps

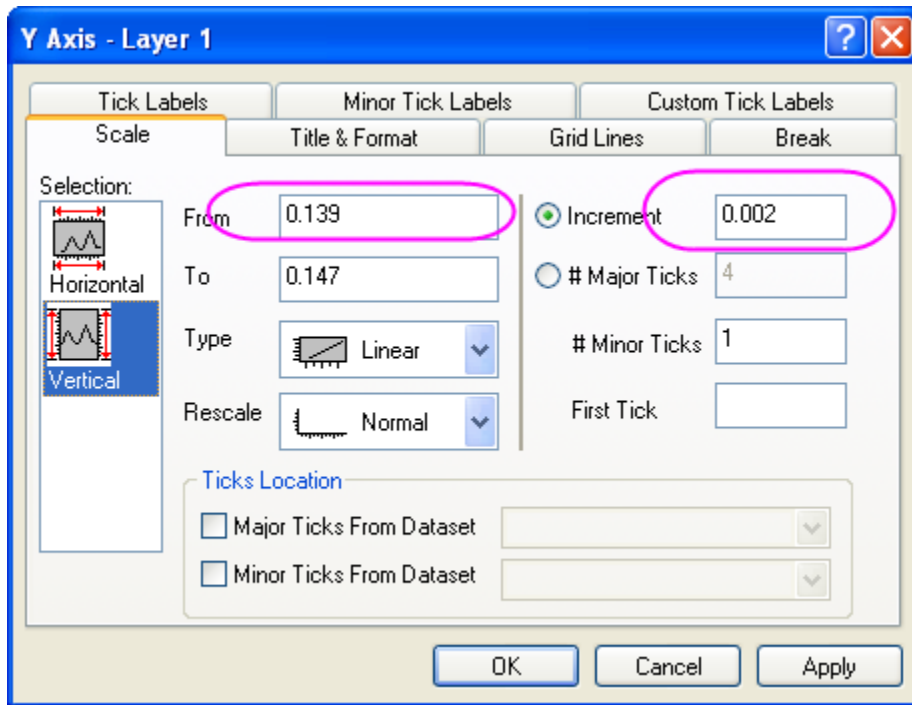
1. To start this tutorial, please download the data file from ftp.
2. Click the **Import Single ASCII**  button to open the file browser, then select the file just downloaded and keep the default settings to import this file.
3. Highlight both columns in the worksheet and select **Plot: Line: Line** from the Origin main menu to plot a line graph.



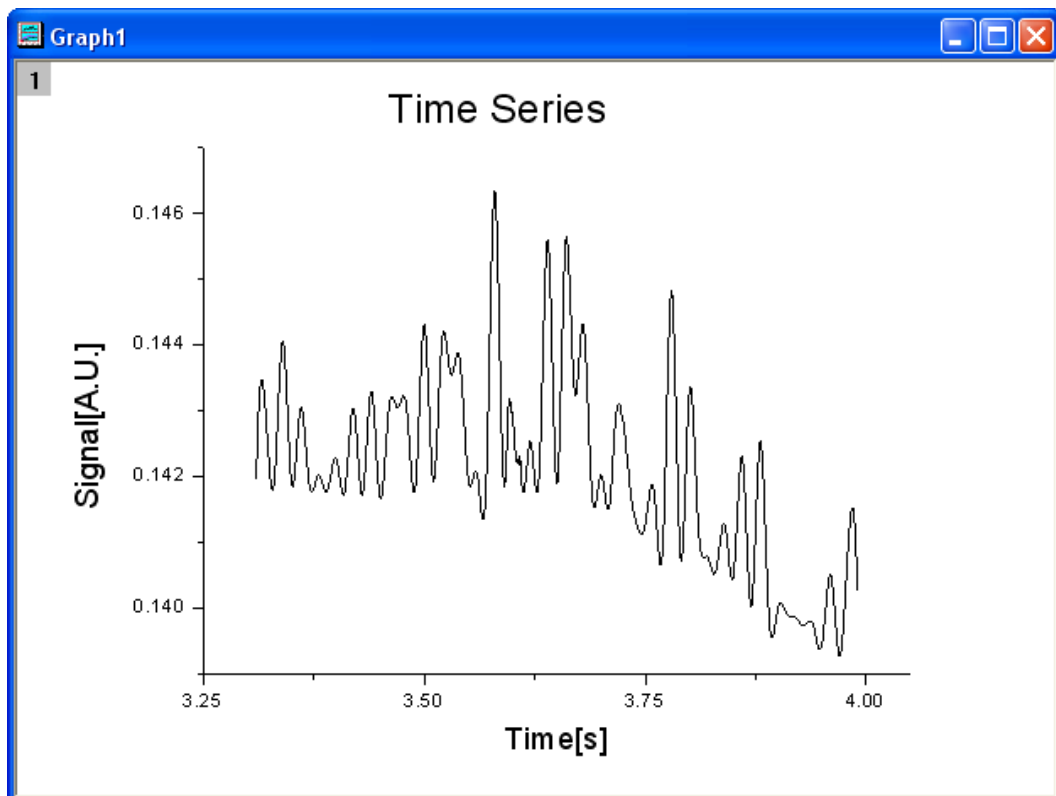
4. Double click on the X axis to open the axes setting dialog, then change the **Increment** to 0.25.



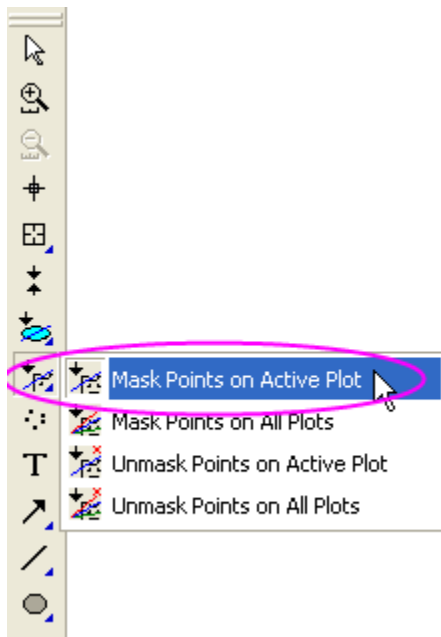
5. Select the **Vertical** in the **Selection** panel, then change the **From** to 0.139 and the **Increment** to 0.002. Click the **OK** button to finish axes settings.



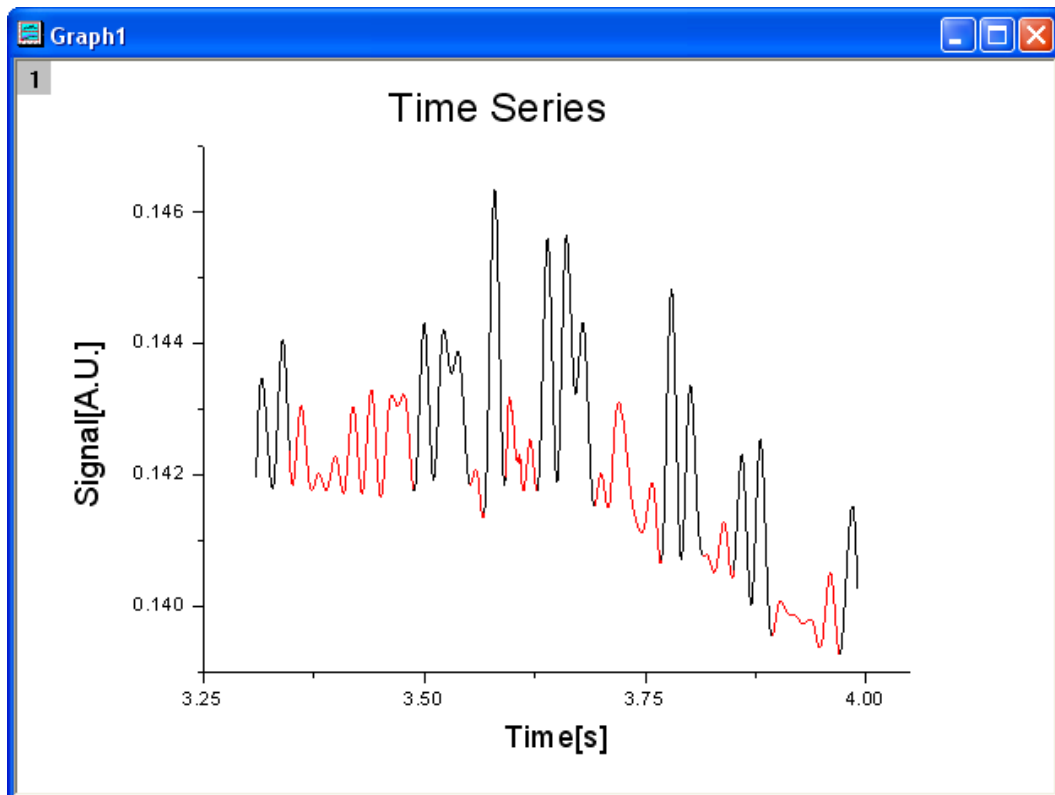
6. Delete the legend on the graph. Then change the X label to **Time[s]** and the Y label to **Signal[A.U.]**. And then add a text object as the title, say **Time Series**. Set font size of both axes labels to 28 and the title to 36.



7. Activate this graph, then click down on the **Regional Mask Tool** button on the Tool toolbar until a context menu appears. Then select **Mask Points on Active Plot** from this menu to get in the masking mode.



8. Click the proper place on the graph and drag a regional to select the low peaks, at the same time, mask them. Repeat it until all low peaks are masked. Then click the **Pointer** button on the Tool toolbar to end the masking mode.



3.2.7 Mark out a segment of plot with different plot style

Summary


In Origin, you can mark out a segment of plot with different plot style, such as a segment of dashed in a solid line plot.

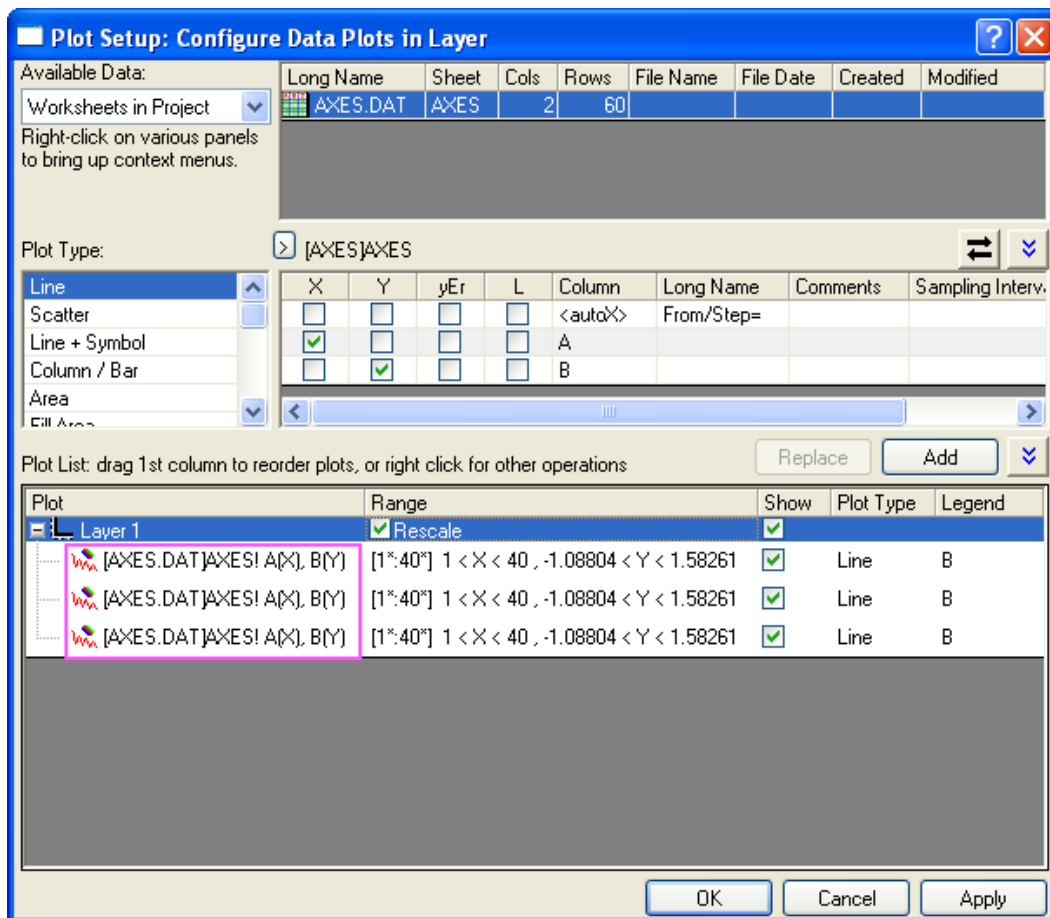
Minimum Origin Version Required: Origin 8.0 SR6


What you will learn

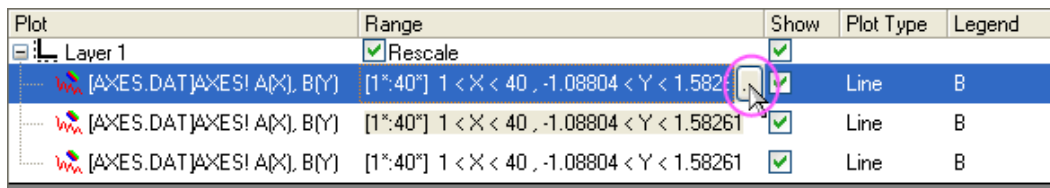
- Use the **Plot Setup** dialog to create a graph
- Mark out the special segment of a plot

Steps

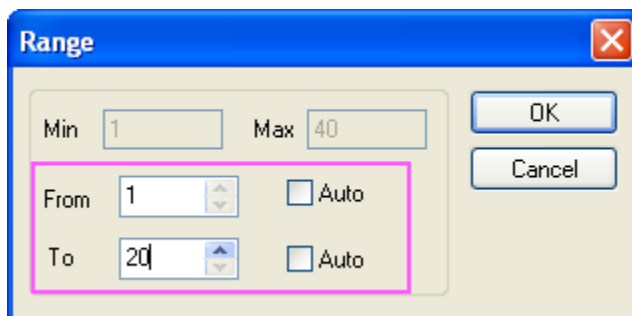
1. Start with an empty worksheet. Select **File: Import: Single ASCII...** from the Origin menu to open the **Import Single ASCII** dialog. Browse to the `\Samples\Graphing` subfolder of the Origin program folder, and import the file `AXES.DAT`.
2. Click the  button on the **Standard** toolbar to create a new graph window and then select **Graph: Plot Setup** from the main menu to bring up the **Plot Setup** dialog.
3. Show all of the three panels of the **Plot Setup** dialog. Select the **AXES** worksheet in the top panel. Then go to the middle panel to select **A** as X and **B** as Y. After that, click **Add** to add this data plot to the bottom panel. Repeat this step three times. Three data plots should be listed in the bottom panel.



4. In the lower panel, click in the **Range** column that corresponds to the first data plot. The  button should be activated. Then click this button to open the **Range** dialog box.



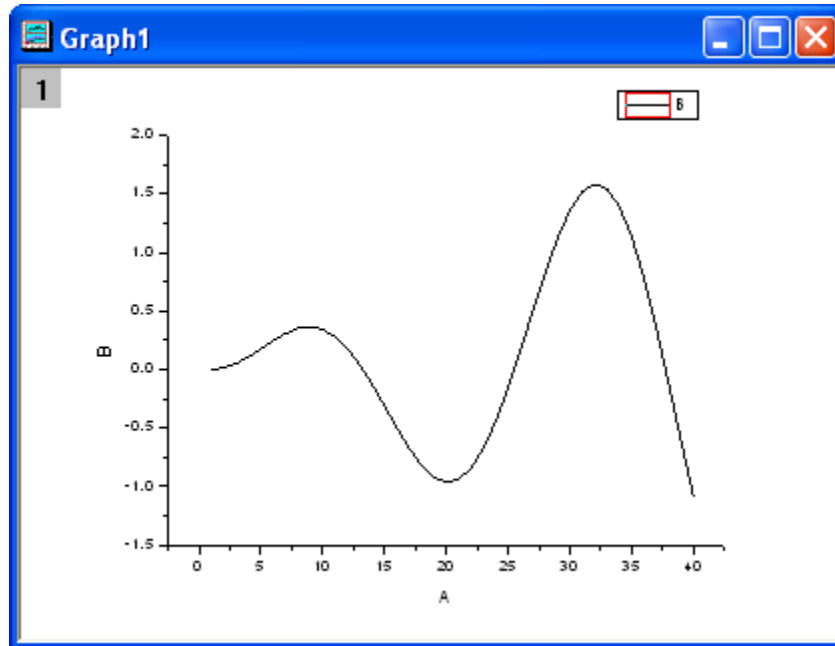
5. Clear the **Auto** check boxes (if they are selected) and then set **From** to **1** and **To** to **20**. Click **OK** to close the dialog.



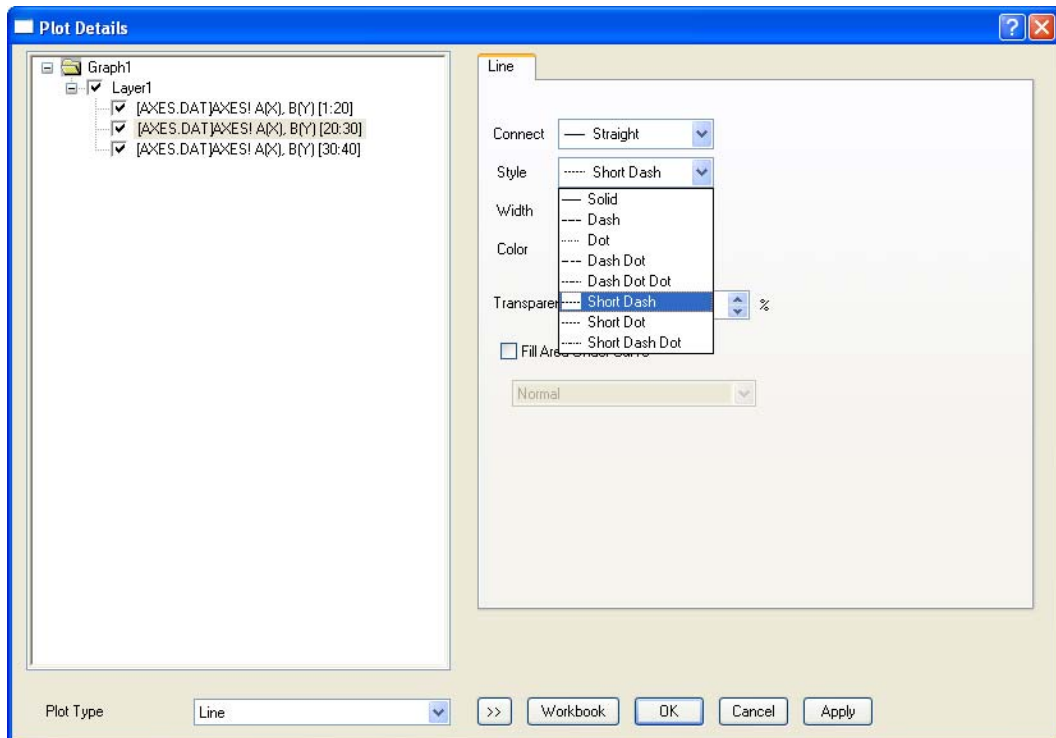
6. Similarly, set the ranges for other two data plots to "20 to 30" and "30 to 40" respectively.

Plot	Range	Show	Plot Type	Legend
Layer 1	<input type="checkbox"/> Rescale	<input checked="" type="checkbox"/>		
[AXES.DAT]AXES! A(X), B(Y)	[1:20] $< X < 20, -0.96201 < Y < 0.3638$	<input checked="" type="checkbox"/>	Line	B
[AXES.DAT]AXES! A(X), B(Y)	[20:30] $20 < X < 30, -0.96201 < Y < 1.35909$	<input checked="" type="checkbox"/>	Line	B
[AXES.DAT]AXES! A(X), B(Y)	[30:40] $30 < X < 40, -1.08804 < Y < 1.58261$	<input checked="" type="checkbox"/>	Line	B

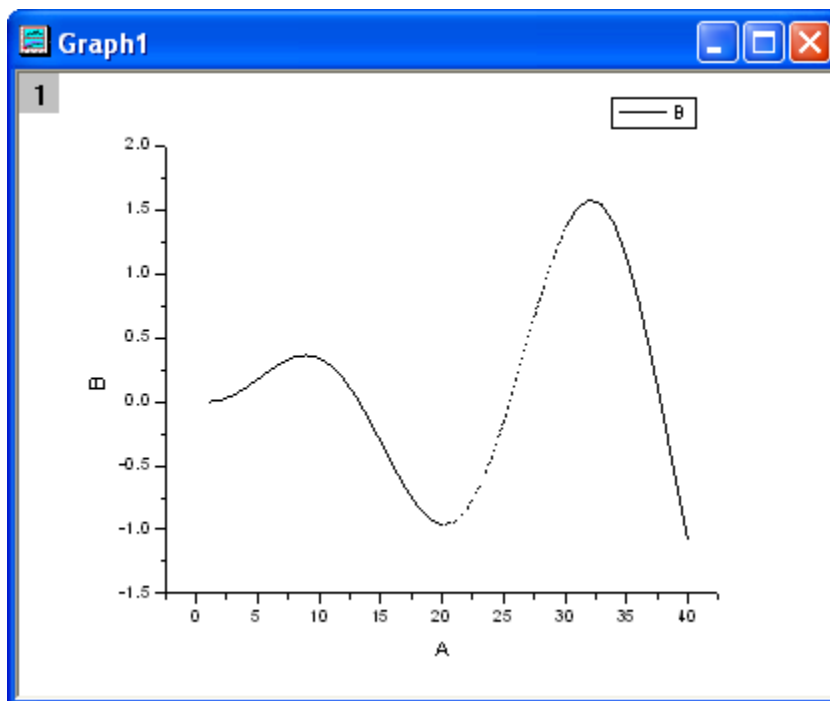
7. Click the **OK** button to close the **Plot Setup** dialog. You should get a graph like this:



8. Double-click on the curve in the graph window to open the **Plot Details** dialog. Select the second data plot from the left panel. In the right panel, change the **Style** to **Short Dash**, and then click the **Ok** button.



9. Finally, we get the plot with a range marked out.



3.2.8 Adding a Data Plot to Existing Graph

Summary

The Plot Setup dialog can be used to add/reorder/arrange data plots in an existing graph. This dialog provides flexibility in selecting the desired data sheet and then selecting data to be plotted using column meta data such as long name.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Use the Plot Setup Dialog Top Panel to find your dataset
- Add the chosen dataset to existing graph

Steps

Choosing the Data Source

1. Click the **New Project** button on the Standard toolbar, to begin with a new project.
2. Click the **Import Wizard** button on the Standard toolbar. The Import Wizard opens. (Note that if this is the first time that you have started the Import Wizard, you will experience a slight delay as Origin compiles the necessary files.)
3. Verify that the **ASCII** radio button is selected in the **Data Type** group.
4. Click the browse button to the right of the File text box. Navigate to the Origin folder; browse to the **Samples** folder and then the **Import and Export** folder.
5. Double-click to select S15-125-03.dat from the list of files. Repeat for files, S21-235-07.dat and S32-014-04.dat.
6. Click **OK**.
7. Leave the **Import Filters for current Data Type as Data Folder:**
VarsFromFileNameAndHeader. (This filter has the settings to use when importing the file.)
8. Set the **Import Mode** as **Start New Sheets**.
9. Click the **Finish** button. The three data files import into the workbook, each as a new sheet. You will have a book with three sheets: **Trial Run 1**, **Trial Run 2**, and **Trial Run 3**.

Plotting the Data

1. Select **Trial Run 1** sheet.
2. Highlight the D(Y) column.
3. Click the **Line** button on the **2D Graphs** toolbar. A new graph is created.

Adding data to the graph

1. Double-click on the layer 1 icon in the upper-left hand corner of the graph. The Plot Setup dialog opens.
2. Select **Layer 1** in the Plot List.

3. Click the blue arrows in the upper right corner of the dialog to **Show Plot Designations**.
4. Again click the blue arrows in the upper right corner of the dialog to **Show Available Data**.
5. Select **Trial Run 2** from the Available Data list.
6. Check **Time** as X and **Position** as Y.
7. Click **Add**.
8. Check the **Rescale** checkbox.
9. Click **OK**.

Updating the Legend and Formatting the Plot

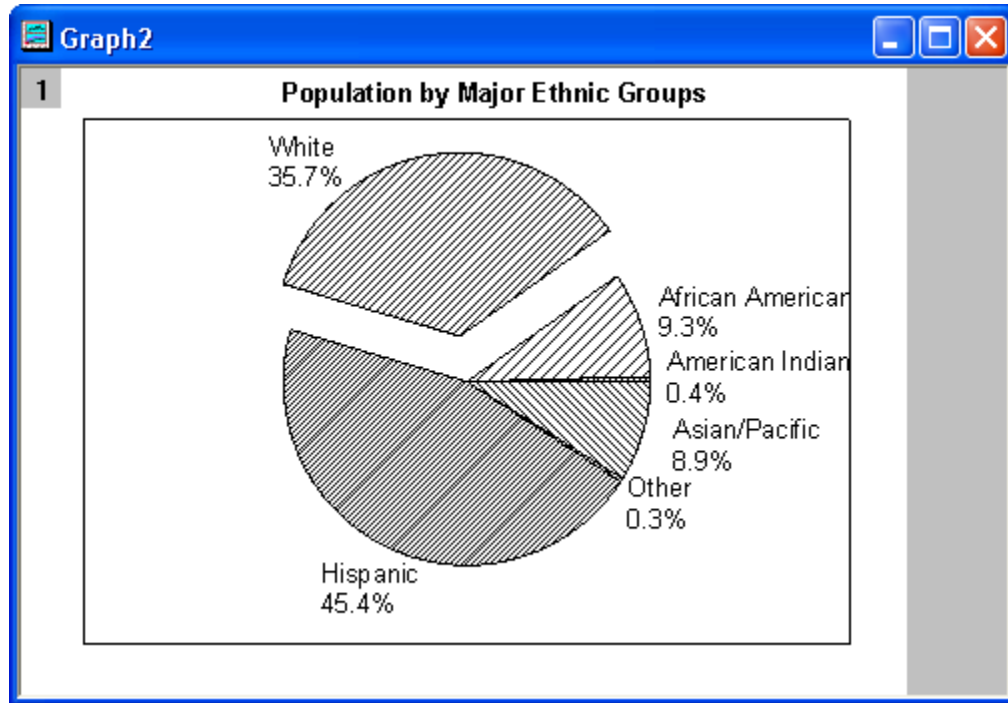
1. Select **Graph:New Legend**.
2. Double-click on the line symbol for the second data plot in the legend. The Plot Details dialog opens.
3. Change the **Color** from Black to Red.
4. Click **OK**.

3.3 Column Bar Pie

3.3.1 2D Pie Chart of a Population Study

Summary

Pie charts created by the default template has a 3D view. However, you can easily convert it to a 2D chart. In addition, you can set the size and rotation of the chart to further customize it. This tutorial will show you how to create and customize a **2D Pie** chart.



Minimum Origin Version Required: Origin 8.5.1 SRO

What will you learn

This tutorial will show you how to

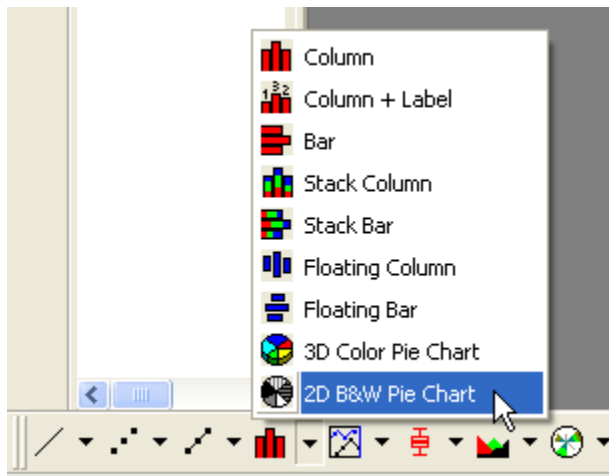
- Create a 2D Pie chart
- Customize the Pie chart

Steps

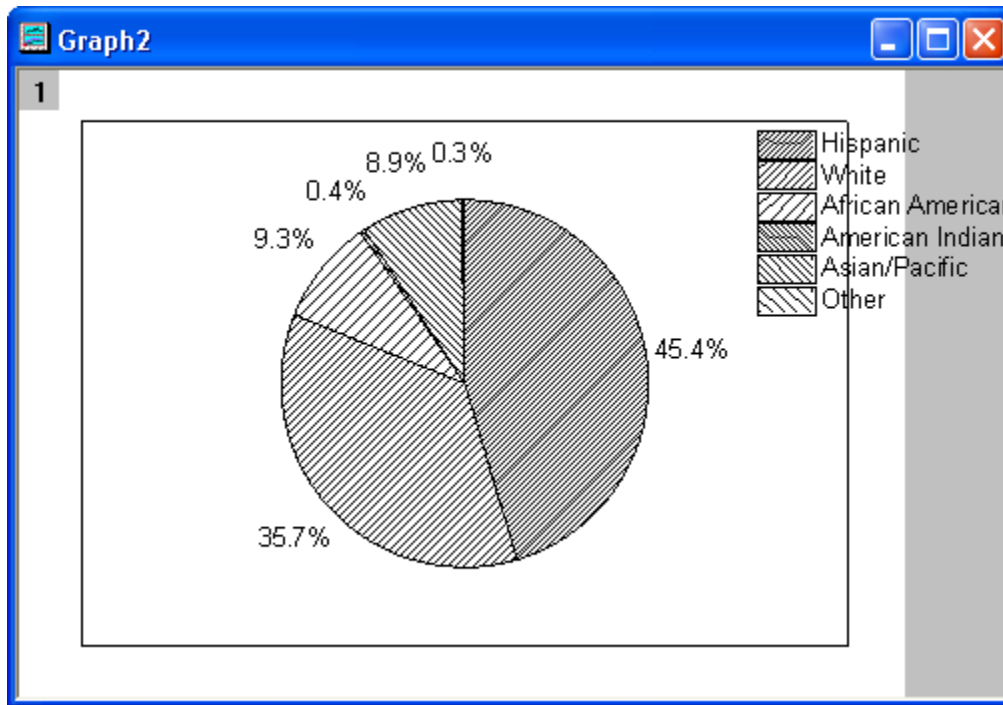
1. Create a new worksheet. Import the sample data (See the sample data here).

	A(X)	B(Y)	C(Z)
Long Name			
Units			
1	Hispanic	45.4	1
2	White	35.7	18
3	African American	9.3	19
4	American Indian	0.4	20
5	Asian/Pacific	8.9	24
6	Other	0.3	18
7			
8			
9			

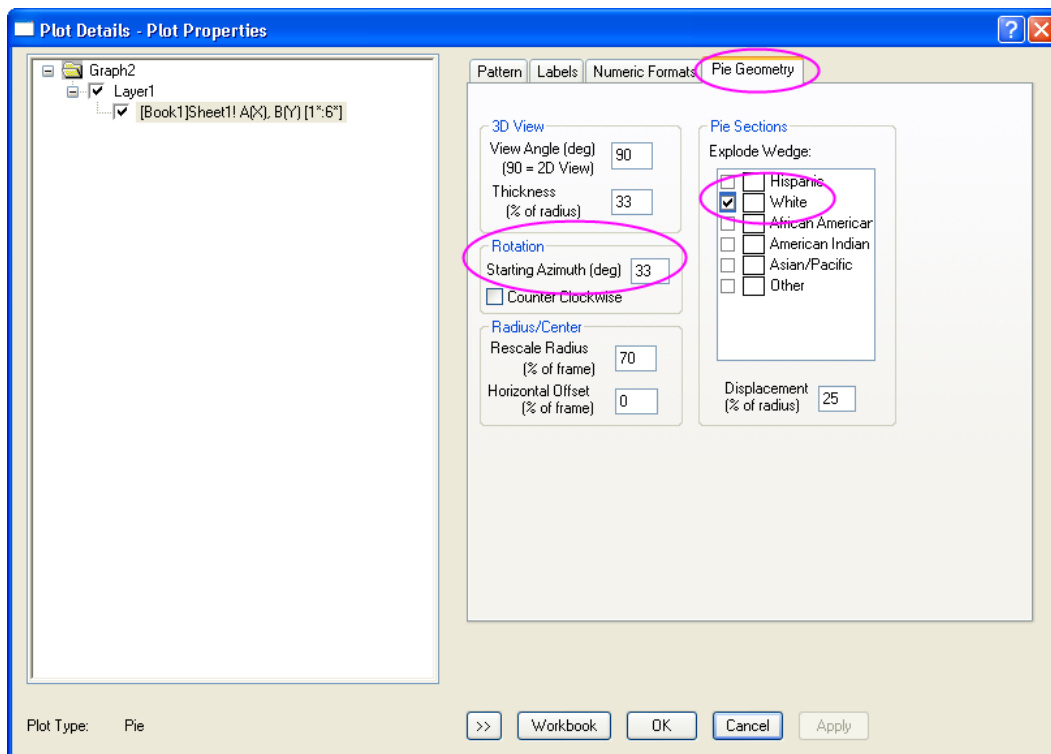
- Highlight column **B** and select the **Pie Chart** button on the **2D Graphs** toolbar.



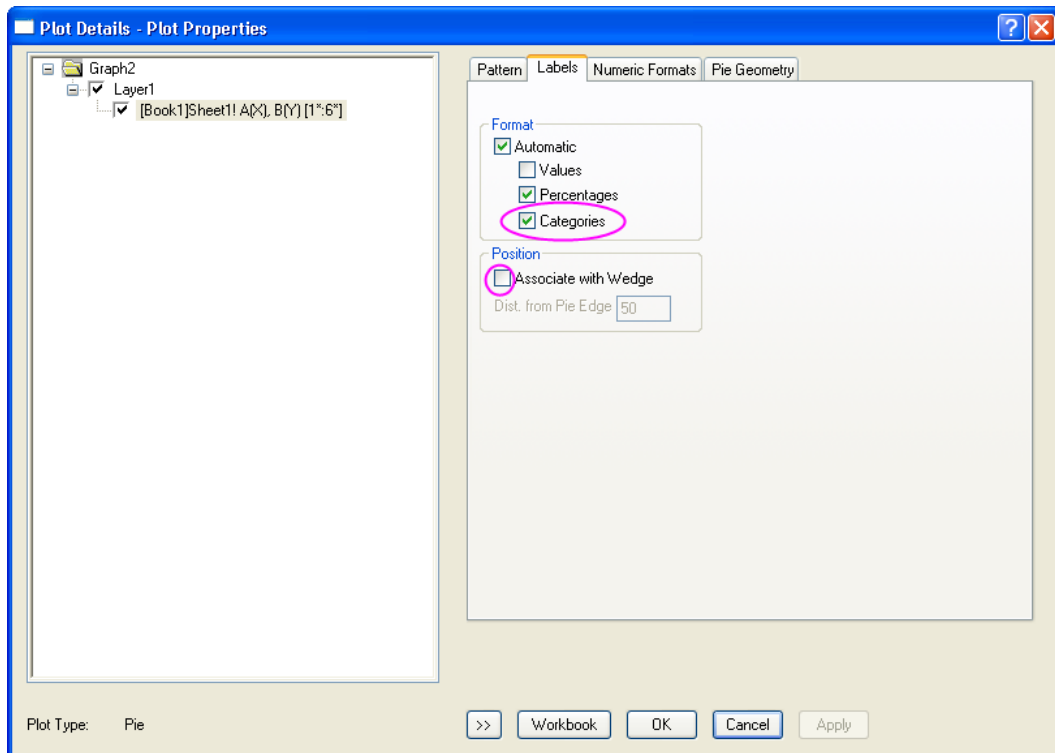
The graph should look like this:



- Now we further customize the chart. Right-click on it and select **Plot Details** from the short-cut menu to open the **Plot Details** dialog. In the **Pie Geometry** tab, set the options as the screenshot below.

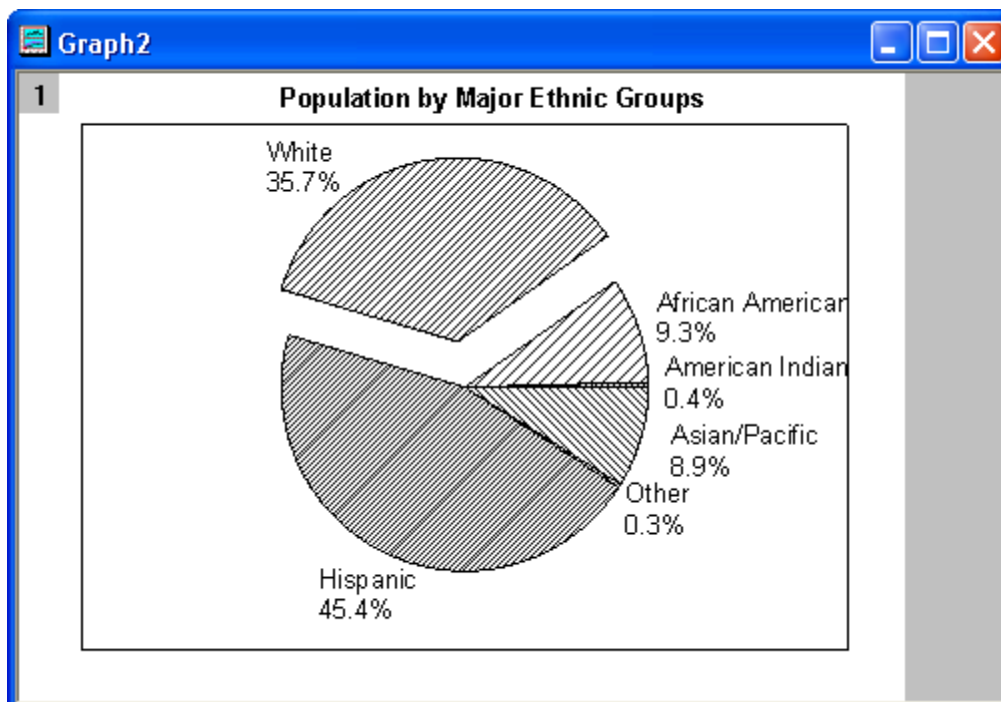


- In the **Labels** tab, set the dialog as the screenshot below.




Click **OK** to close the **Plot Details** Dialog. Rearrangethe text label of each pie wedge.

5. Right click on the **Legend** and select **Delete** from the short-cut menu.
6. Select the **Text** button from the **Tools** toolbar. Then click near the top of the layer. Enter **Population by Major Ethnic Groups** to add a title for the graph. Then you can use the tools in the **Format** toolbar to customize the text. The graph should look like:



Sample Data

Download the **2D_Pie_Chart.txt** file from http://www.originlab.com/ftp/graph_gallery/data/2D_Pie_Chart.txt. Click the **Import Single ASCII**

button . Select the file to import it into Origin.

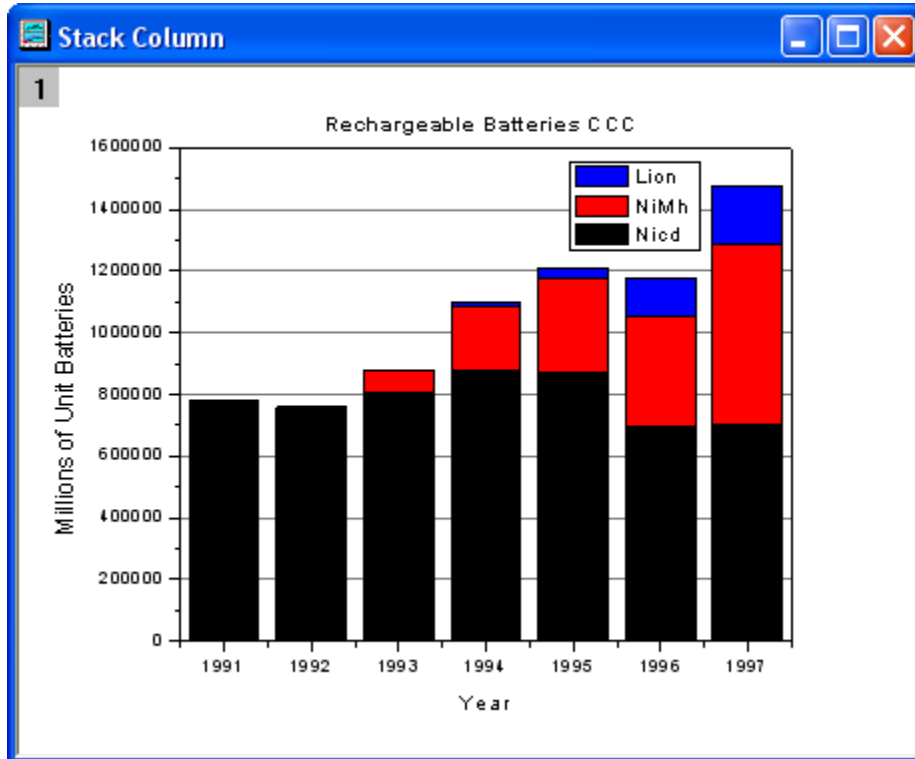
The following table contains part of the sample data.

A(X)	B(Y)	C(Y)
Hispanic	45.4	1
White	35.7	18
African American	9.3	19
American Indian	0.4	20
Asian/Pacific	8.9	24
Other	0.3	18

3.3.2 Stack Column and Bar Charts of Rechargeable Battery Statistics

Summary

This tutorial will show you how to create a Stack Column bar graph.



Minimum Origin Version Required: Origin 8.0 SR6

What will you learn

This tutorial will show you how to

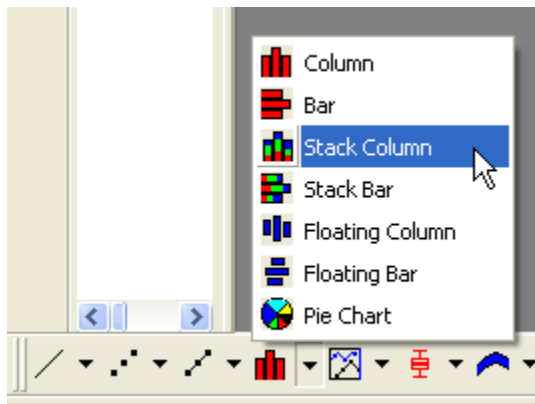
- Create a Stack Column bar graph.
- Customize the graph.

Steps

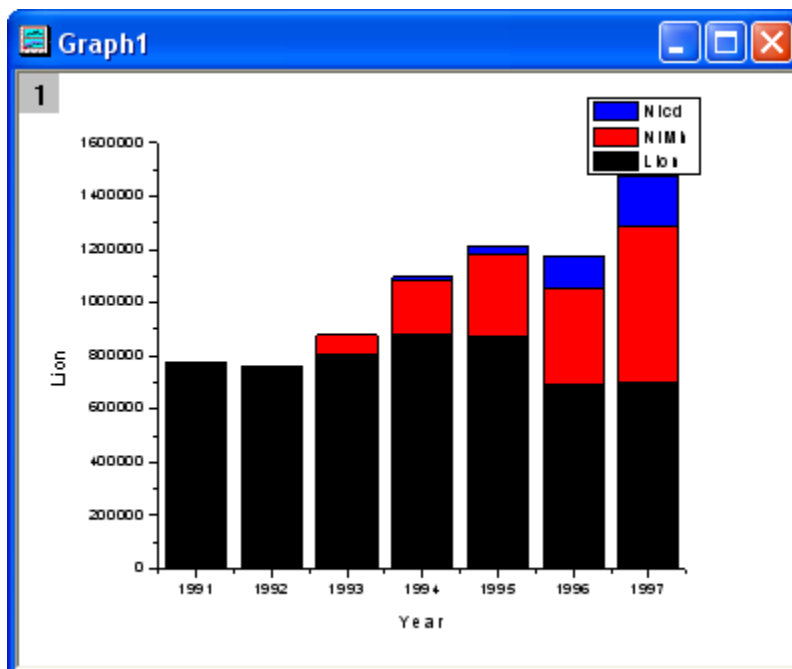
1. Create a new worksheet. Import the sample data.

	A(Y)	B(Y)	C(Y)	D(Y)
Long Name	Year	Lion	NiMh	Nicd
Units				
1	1991	777719	--	--
2	1992	759629	--	--
3	1993	807048	72204	--
4	1994	879264	204666	12000
5	1995	872698	306106	32029
6	1996	694384	356218	125388
7	1997	703296	580930	193496
8				

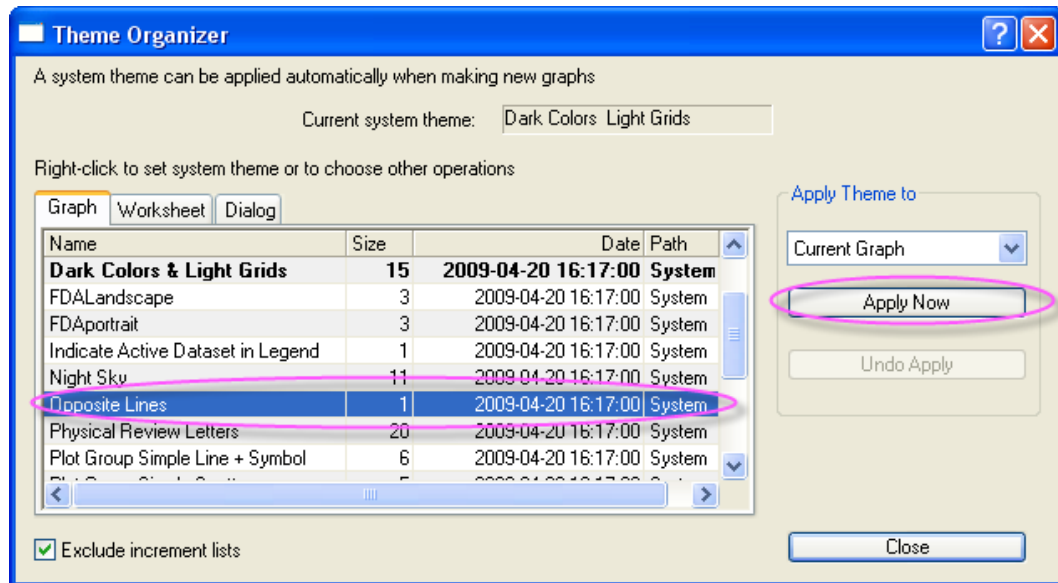
- Highlight all the columns and click the **Stack Columns** button on the **2D Graphs** toolbar.



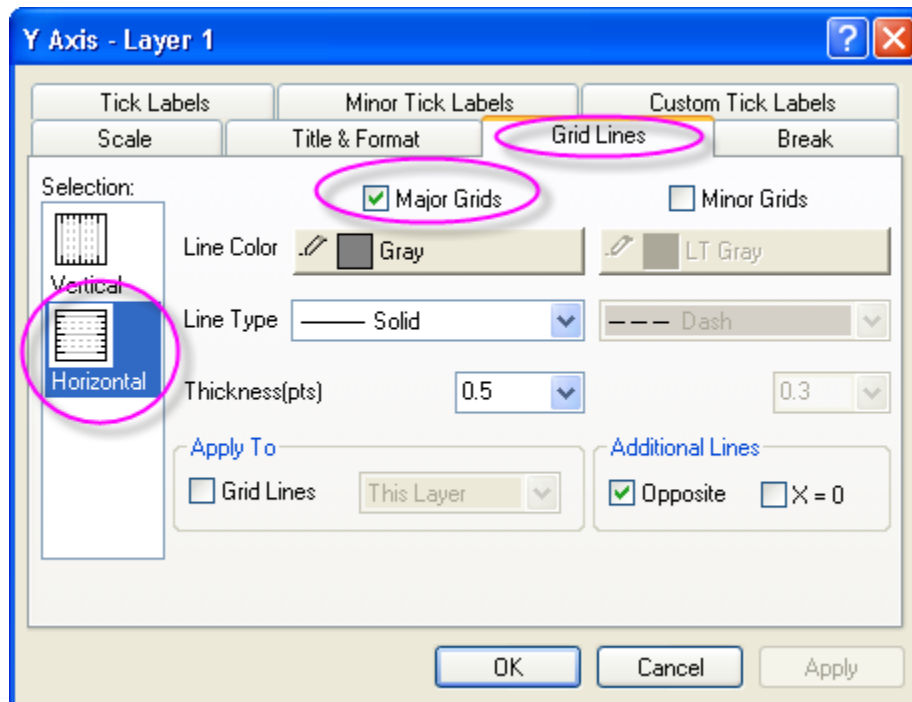
The graph should look like



- Then apply a graph theme to add a top X axis and a right Y axis. Select **Tool: Theme Organizer** to open the **Theme Organizer** dialog. Activate the **Graph** tab and select **Opposite Lines** from the table. Then click the **Apply Now** button. Click the **Close** button to close the dialog.

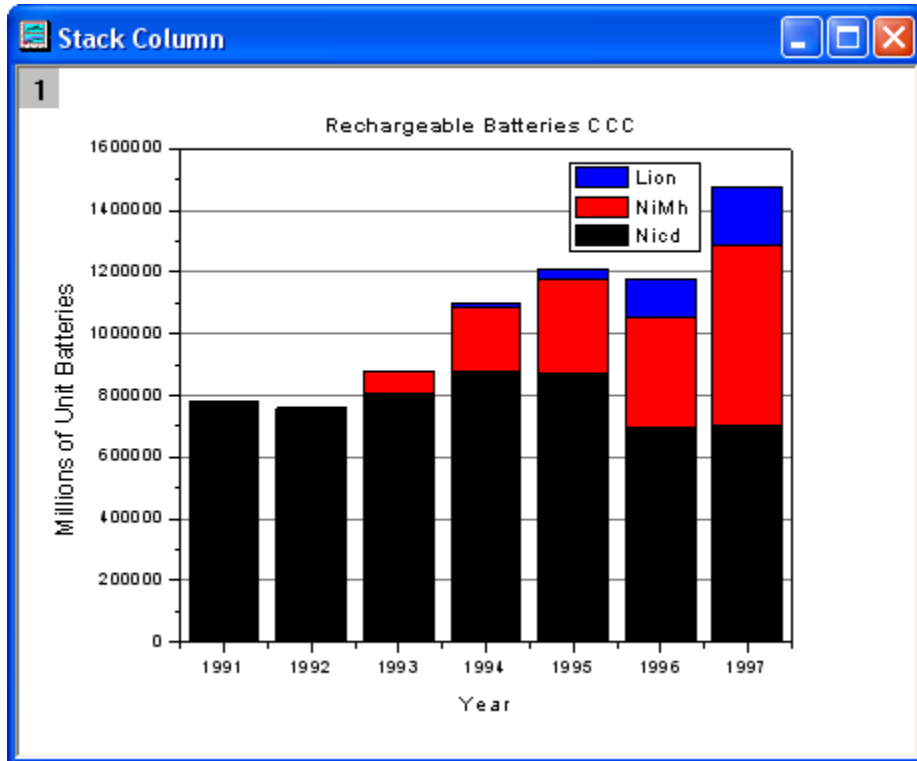


- Double-click on the Y axis to open the **Axis Properties** dialog. Activate the **Grid Lines** tab. Set the options as the screenshot below. Click the **OK** button to add grid lines to the graph.




- Change the title of the Y axis to **Millions of Unit Batteries** and add a graph title, **Rechargeable Batteries CCC**.

The graph should look like



Sample Data

Download the **Stack_Column_and_Bar_Charts.txt** file from http://www.originlab.com/ftp/graph_gallery/data/Stack_Column_and_Bar_Charts.txt. Click the

Import Single ASCII button  and select the file to import it into Origin.

The following table contains part of the sample data.

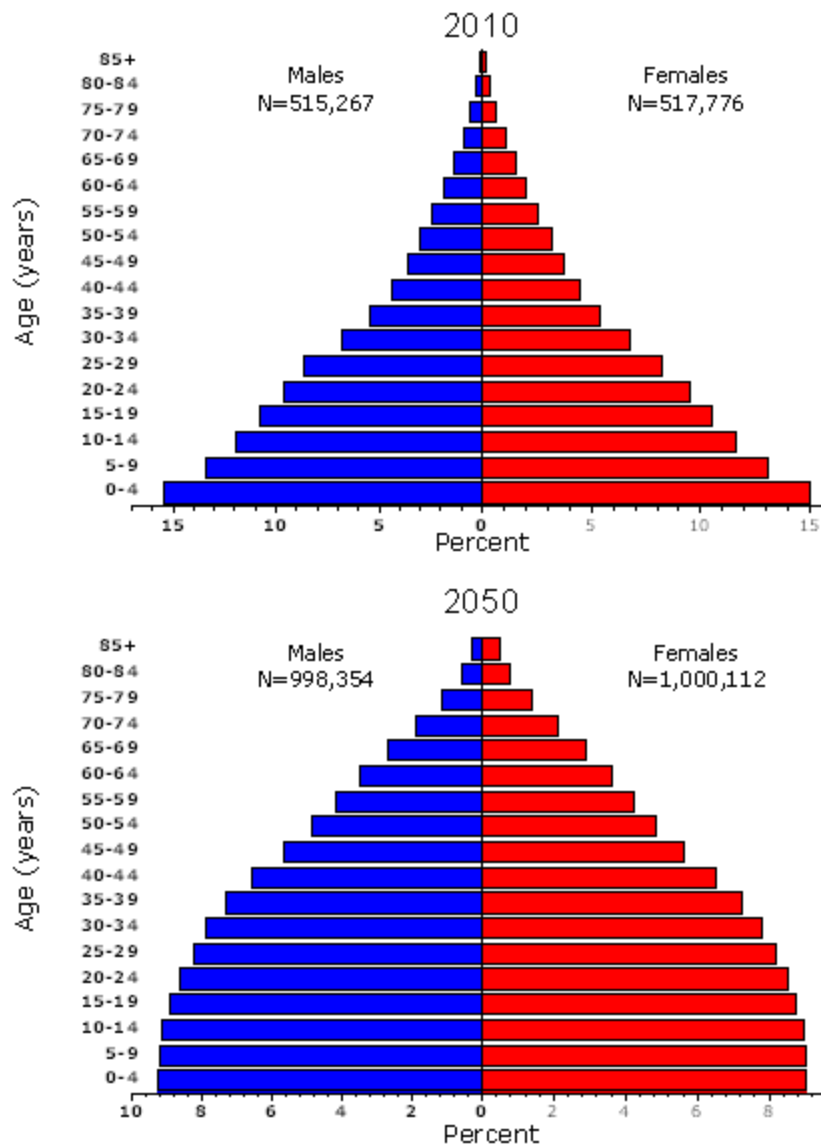
year(X)	Nicd(Y)	NiMh(Y)	Lion(Y)
1991	777719	--	--
1992	759629	--	--
1993	807048	72204	--
1994	879264	204666	12000
1995	872698	306106	32029
1996	694384	356218	125388
1997	703296	580930	193496

3.3.3 Bar Graph of African Population

Summary

In this tutorial, a graph will be created to show the African population distribution, 2010 versus 2050.

Figure 1. African population by five-year age groups and sex, 2010 versus 2050



Source: Population Division of the Department of Economic and Social Affairs of the United Nations Secretariat, World Population Prospects: The 2008 Revision, <http://esa.un.org/unpp>, Monday, April 27, 2010; 3:19:17PM.

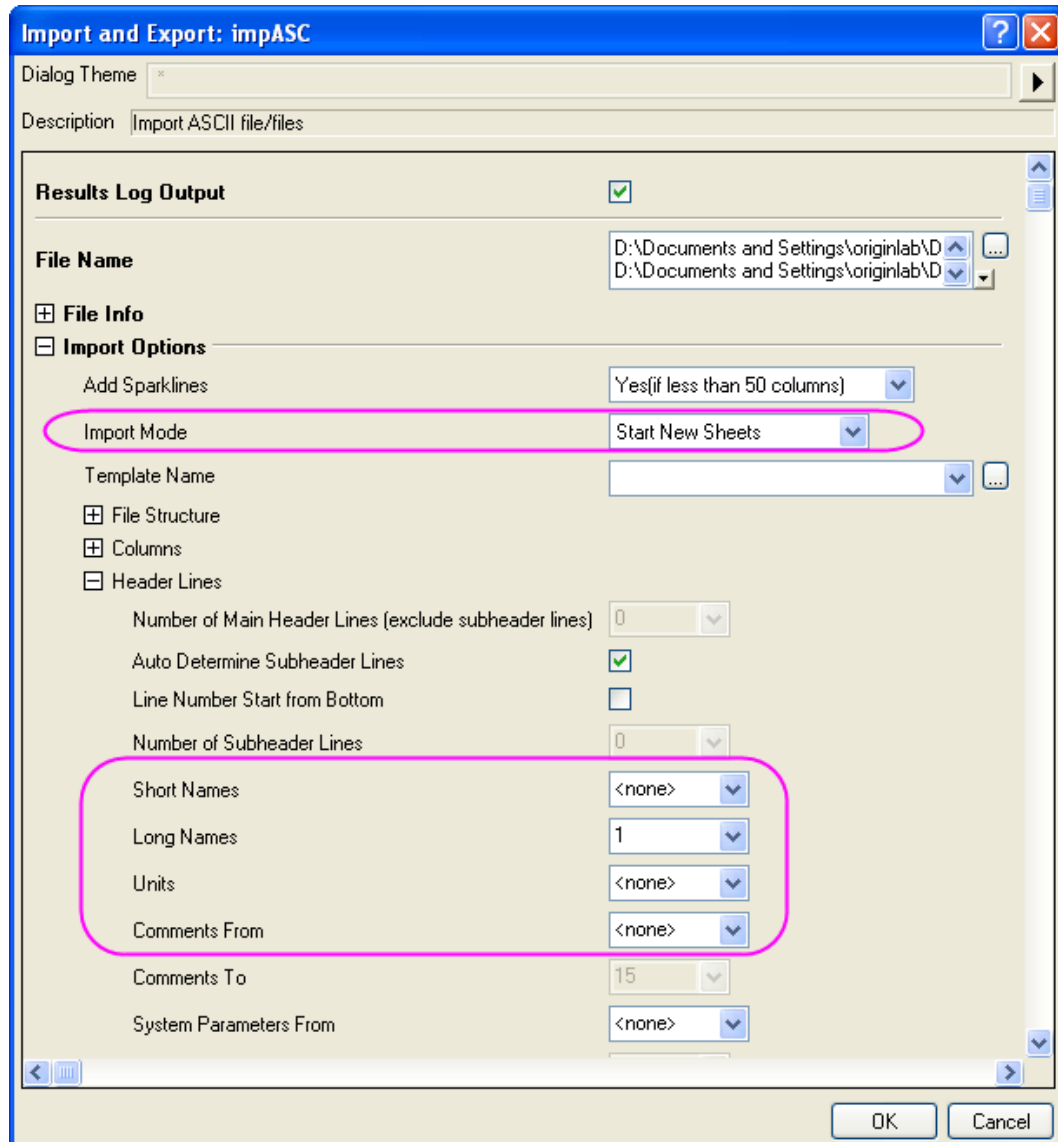
Minimum Origin Version Required: Origin 8 SR0

What you will learn

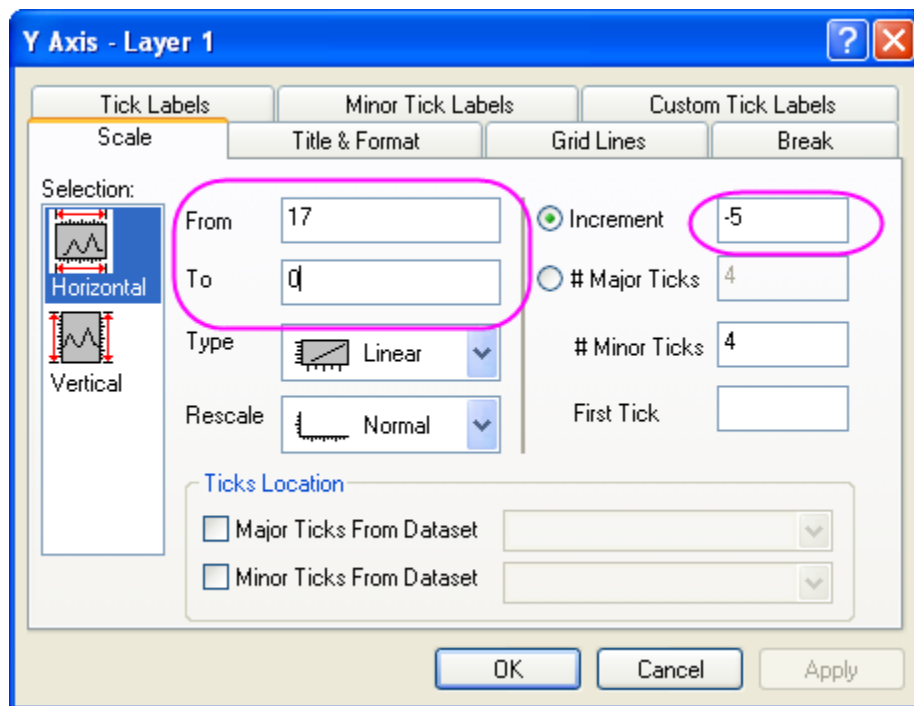
- Set axis scale to reverse bar chart
- Merge graphs

Steps

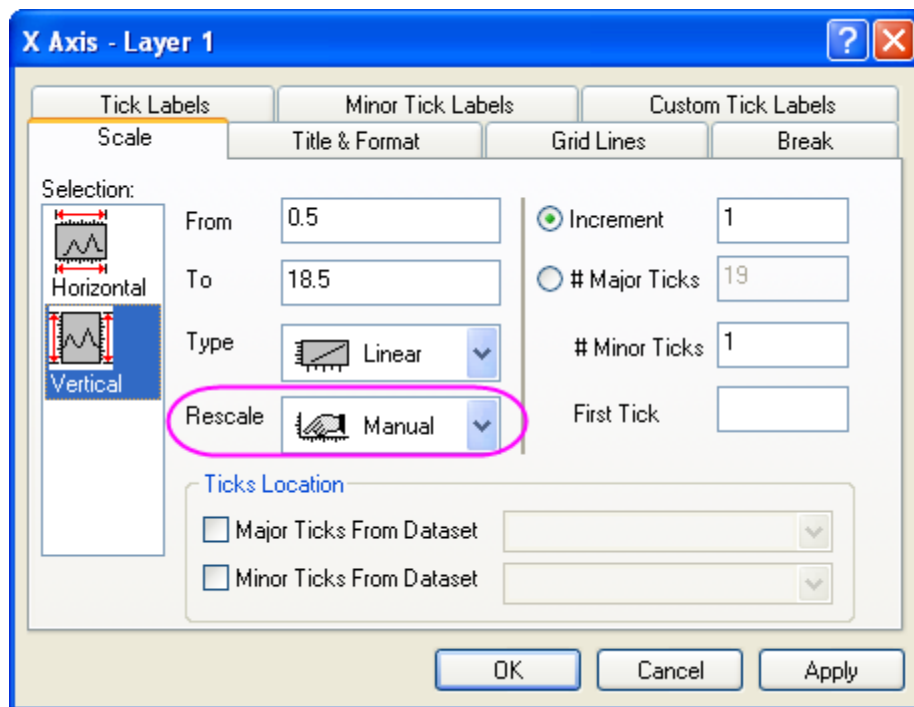
1. Download the source data files `bar_graph_of_African_population_01.txt` and `bar_graph_of_African_population_02.txt` from our ftp. Import these two files into an Origin workbook with two worksheets, each for one .txt file. The **Import Multiple ASCII** dialog can be set as follow:



2. Highlight the "male" column in the "bar_graph_of_African_population1" sheet and click the Bar button on the 2D Graphs toolbar to plot a bar graph.
3. In the bar graph you just created, double-click X-axis to open the **X-Axis** dialog box. Go to the **Grid Lines** tab. Select **Horizontal** in the **Selection** box and then uncheck **Minor Grids** to hide the grid lines.
4. Go to the Scale tab. Select **Horizontal** in the **Selection** box. Change From and To to 17 to 0, respectively. Set increment to -5.

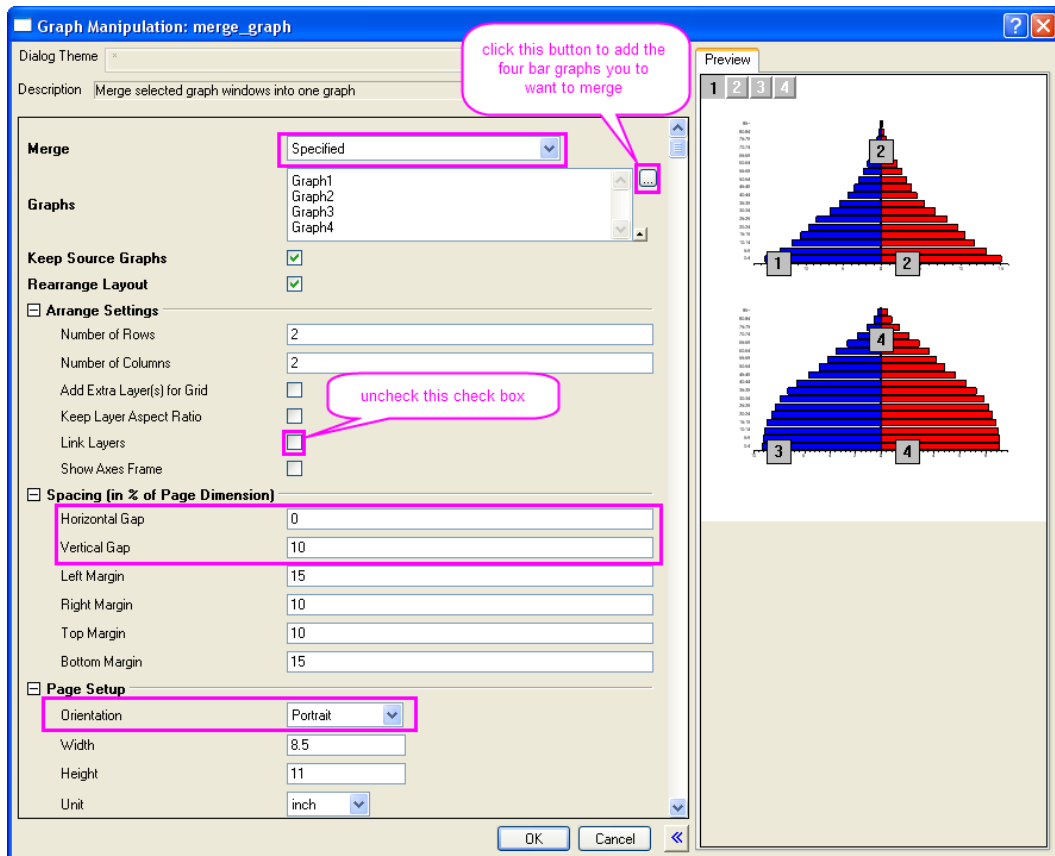


Then select **Vertical** in the **Selection** box and select **Manual** from the **Rescale** drop-down list. Click **OK** to apply.



- Double-click on the bar graph to open the **Plot Details** dialog. Go to the **Pattern** tab and change **Color** (in the Fill box) to Blue. Click **OK** to change the bar fill color to blue.

6. Delete the XY axis titles and the Legend.
7. Highlight the ?female? column in the "bar_graph_of_African_population1" worksheet and click the Bar button to plot another bar graph.
8. Repeat step 3 to turn off the grid lines. Then do the same things as above to reset the axis scale. However this time, change the Horizontal axis? From and To values to 0 to 16, respectively, and set Increment to 5.
9. Delete the XY axis Tick Labels, Title and Legend.
10. Repeat steps 2 through 9 to plot two more bar graphs from the worksheet "bar_graph_of_African_population_". At this time, set the X-Axis scale in the male bar graph to go From 10.0116 To 0, by an Increment of -2. Leave the Axis scale in the female bar graph as the default and change only the Y-axis to Manual.
11. Now you have four bar graphs. To merge these four graphs, activate one of the four graph windows and select **Graph: Merge Graph Windows...** from the main menu. Doing so opens the merge_graph dialog box. Change the settings as follows to merge these four bar graphs into one graph window.

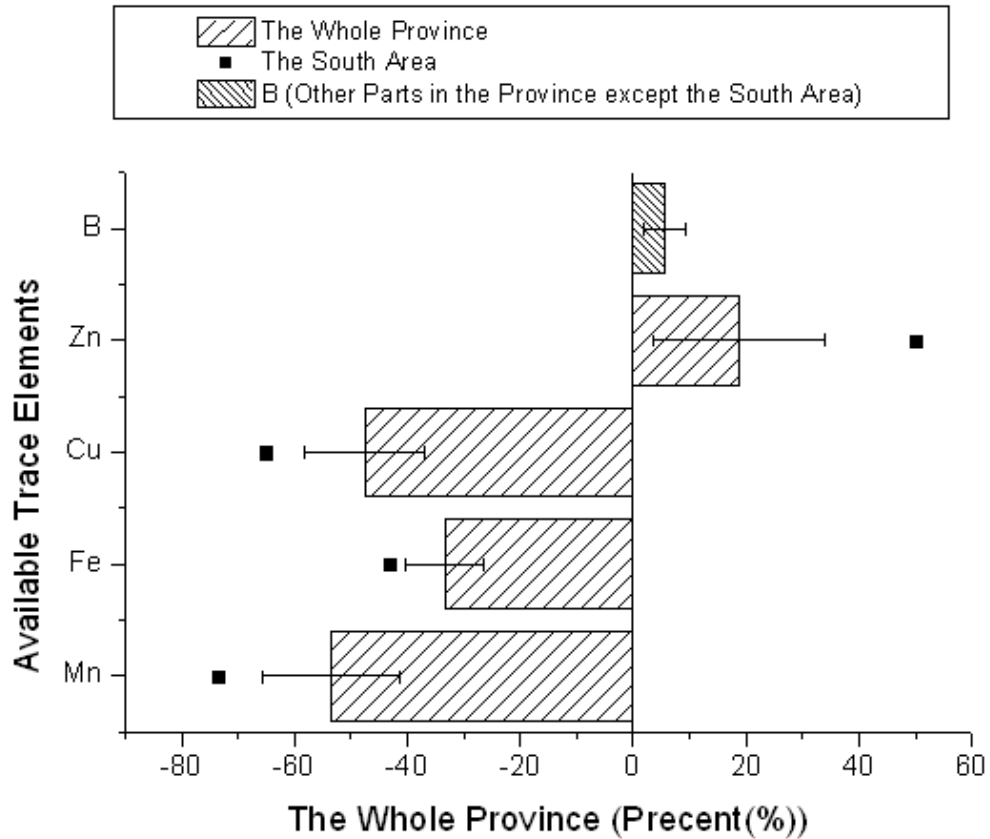


12. In the merged graph, add text boxes: ?Percent? and ?Age (years)?, as axis titles of the upper and lower graphs. Also add some other text boxes, such as a graph title: ?Figure 1. African population by five ? year age groups and sex, 2010 versus 2050?, to better describe this graph.

3.3.4 Bar and Scatter Plot with Error Bar

Summary

Below graph is made of bar chart and scatter plot.



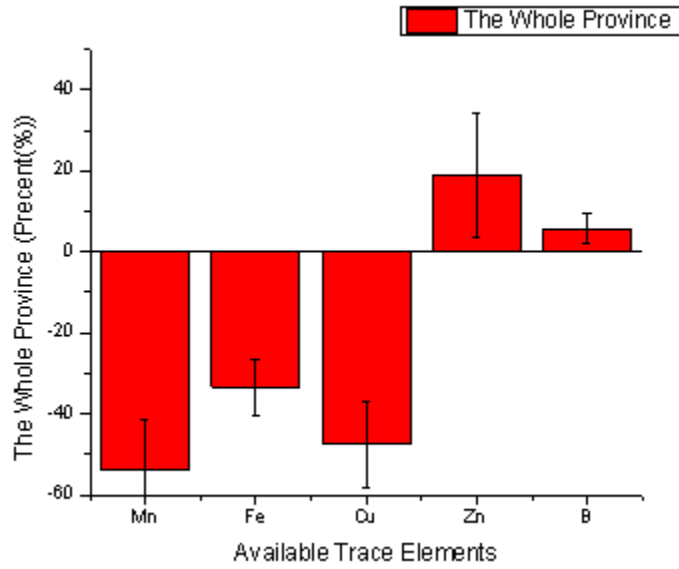
Minimum Origin Version Required: Origin 8.0 SR0

What will you learn

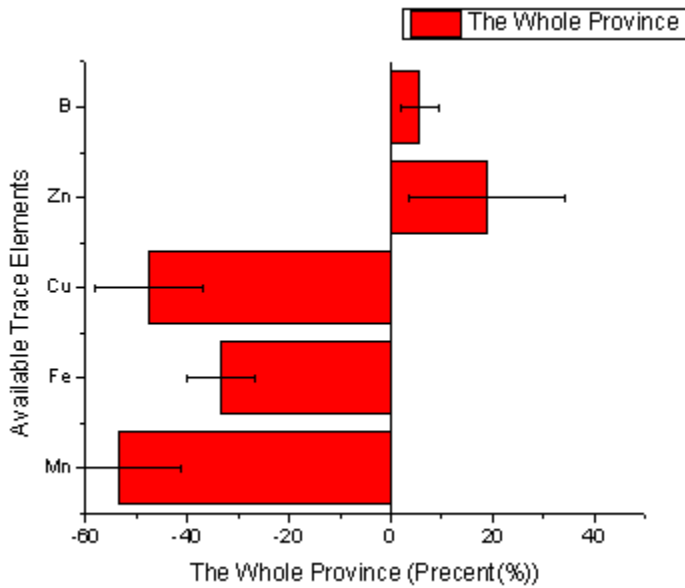
- How to add a scatter plot to a bar graph
- How to set the Plus and Minus error bar

Steps

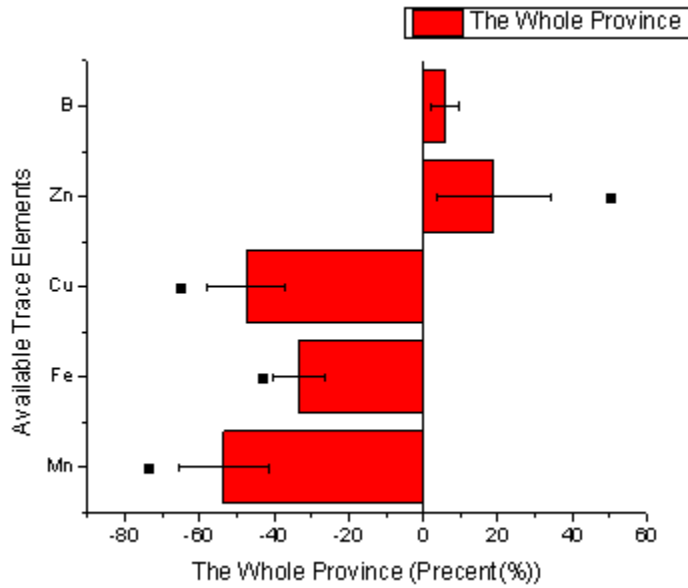
1. Download the source data file from our ftp and import it into Origin.
2. Right-click the col(C) to select **Set as: Y Error** from the context menu.
3. Highlight Col(A), col(B) and col(C), select **Plot: Column/Bar/Pie: Column** from the main menu to plot a column graph with Y error bar.



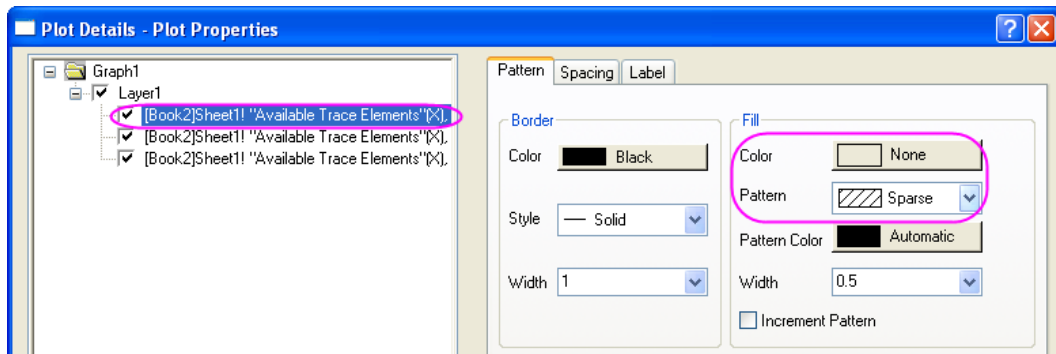
- Active the graph window, and then select **Graph: Exchange X-Y Axes**.



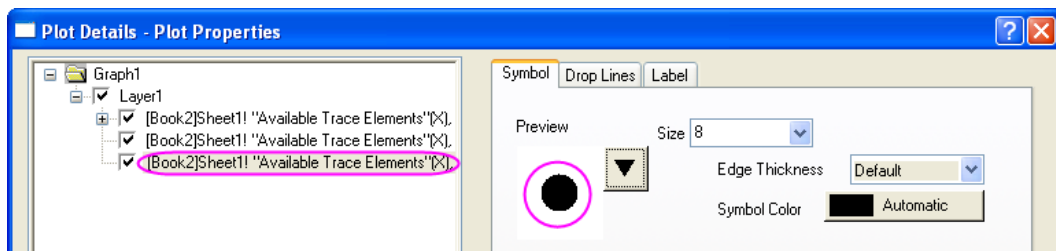
- Highlight the col(D) in the worksheet and then active the graph window again. Select Graph: Add Plot to Layer: Scatter from the main menu to add the col(D) as scatter to the column graph.



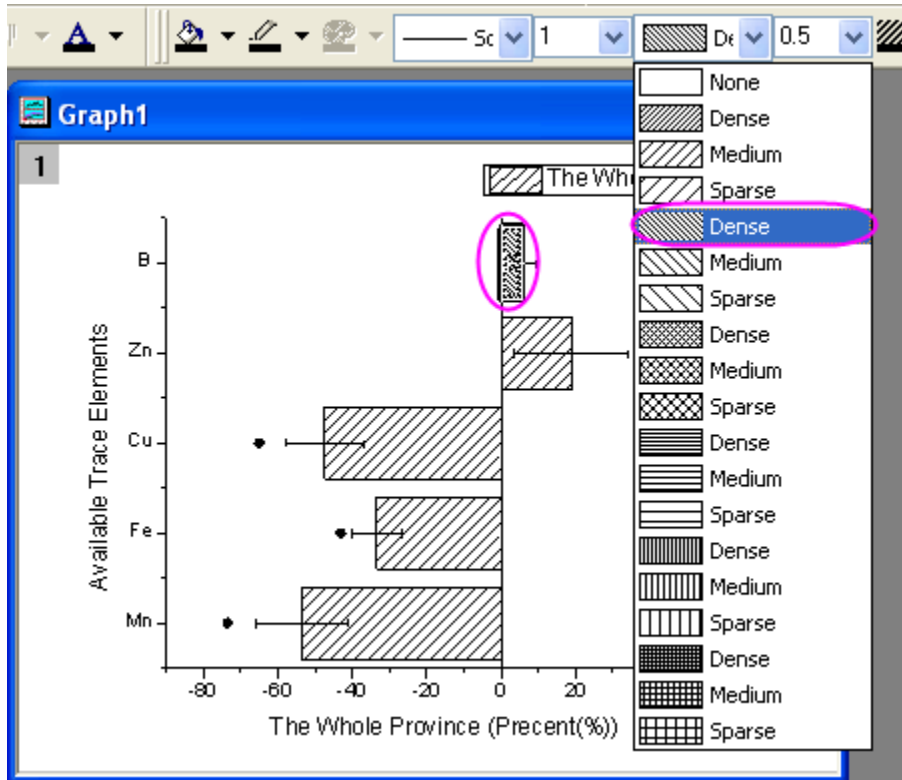
- Double-click the column graph to open the Plot Details dialog. Set the **Color** and **Pattern** options in the **Fill** group of **Pattern** Tab as below.



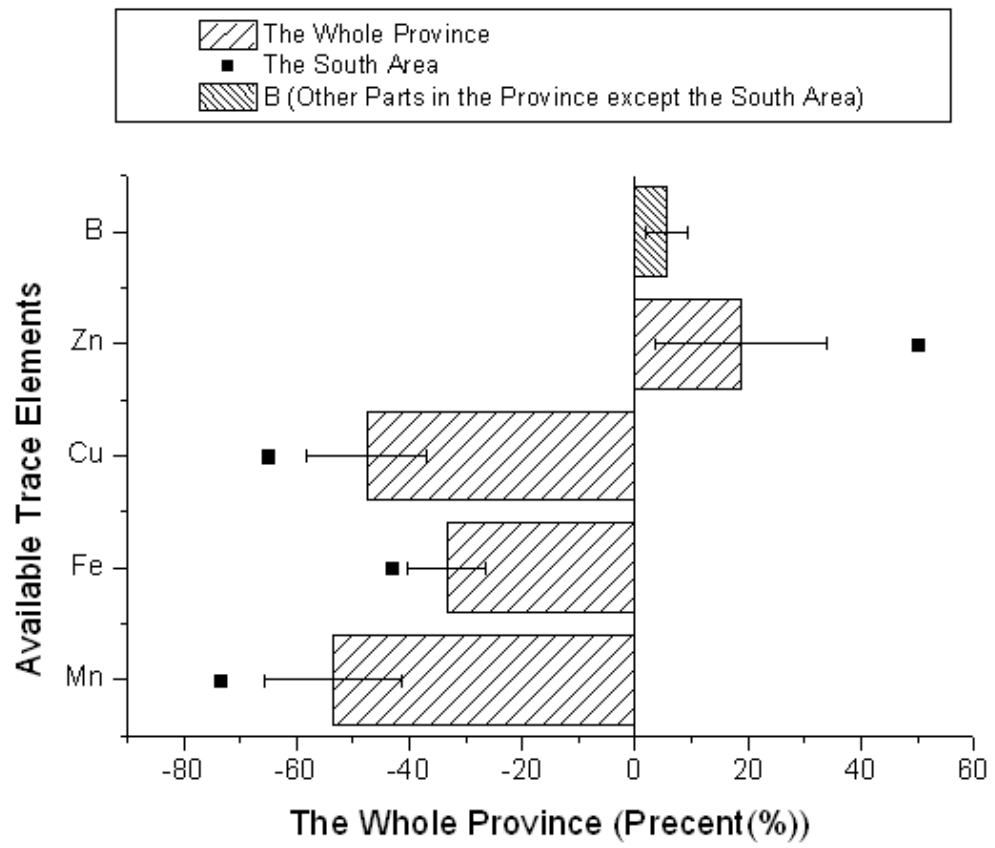
- Select the scatter plot in the left panel, then set the Symbol as below. Then click **OK** button to close the dialog.



- Active the graph window, hold the **Ctrl** key and select the first bar to set its pattern as **Dense** as below from the **Style** toolbar.



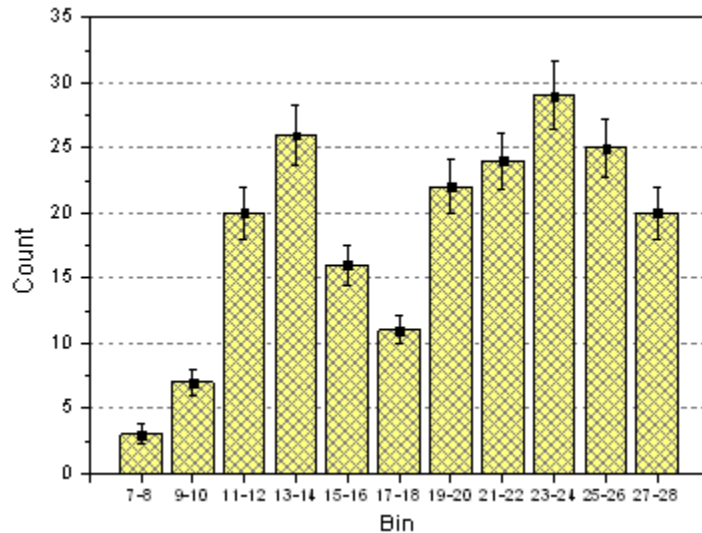
9. Delete the legend, then select Graph: New Legend from the main menu to add a new legend for graph.
10. Right-click the legend to select Properties from the context menu to open the **Objects Properties** dialog. Set **Background** as **Black Line**. Then edit and move the legend as below.



3.3.5 Column Graph with Error Bars

Summary

This custom graph illustrates how Origin can include error bars along with its data plots. The same Y data is plotted twice, once as a scatter plot and again as a column plot. The error bars are attached to the scatter plot in this case. Error bars can be included in the graph in both the Y and X directions.



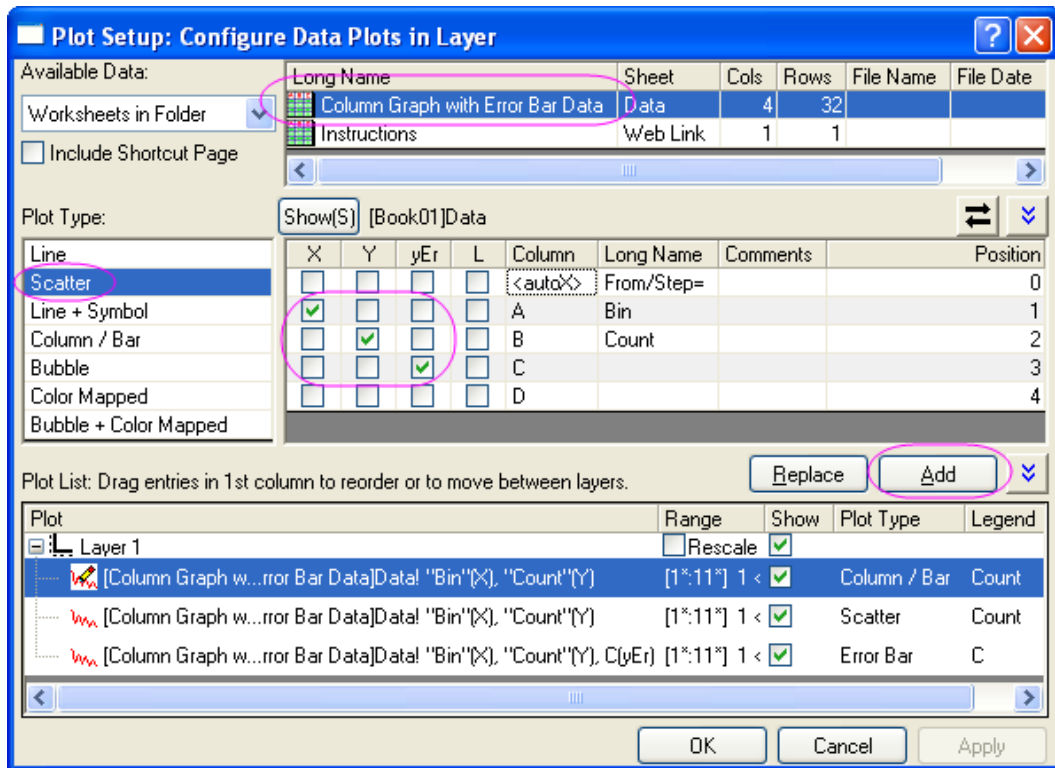
What you will learn

- Create and customize column graph
- Use Plot Setup dialog to add a new data plot into your graph

Steps

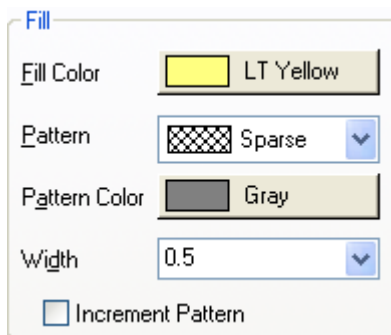
This tutorial is associated with the sample project: `\Samples\Samples.opj`. (If you don't have the Project file, please download the data file from [here](#))

1. Open the Project file, and browse to the folder *Graphing: Column and Pie: Column with Error Bar*. Active the worksheet and make sure the column type as **X, Y, Y Error** and **Label** accordingly.
2. Highlight column 2 and select **Plot: Column/Bar/Pie: Column** to create a column graph.
3. With the graph window active, select **Graph: Plot Setup** to bring up the Plot Setup dialog. We will add the scatter and error bars from this dialog as below:

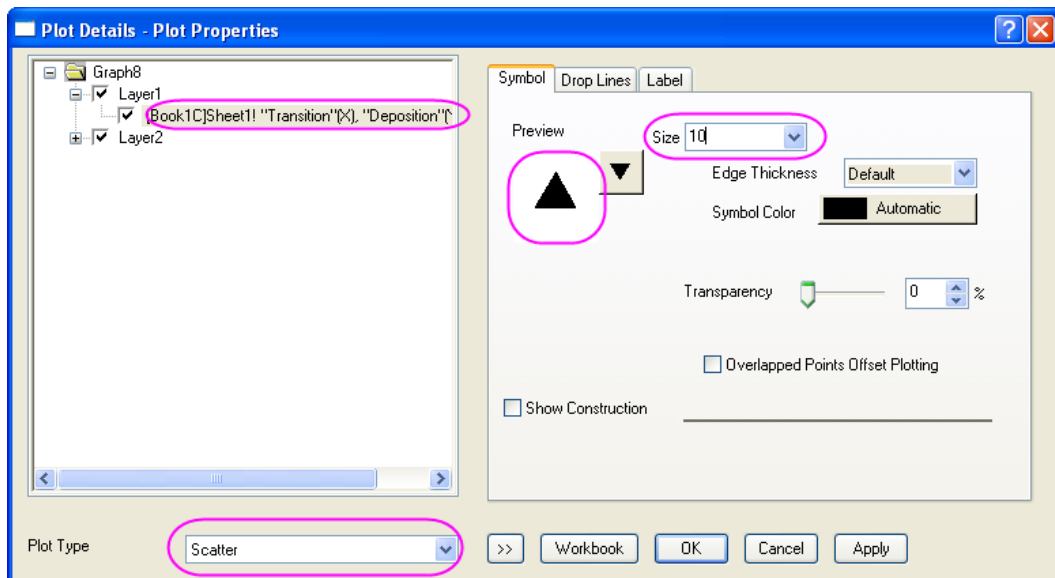


Click the **Add** button to add scatter data to column plot. Then click **OK** to go back to column graph window.

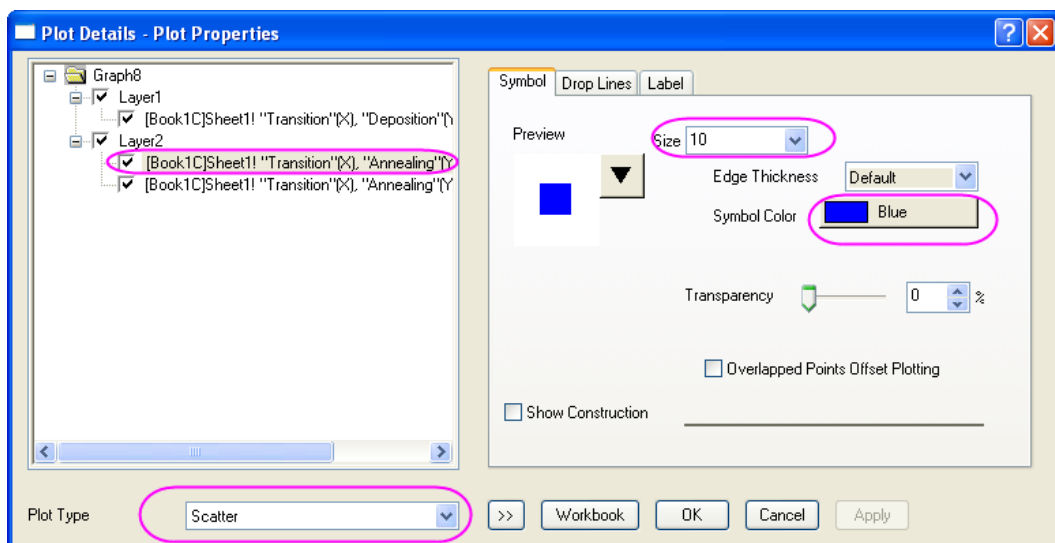
4. Double-click the columns to bring up the **Plot Details** dialog to customize the graph in **Pattern** tab as below:



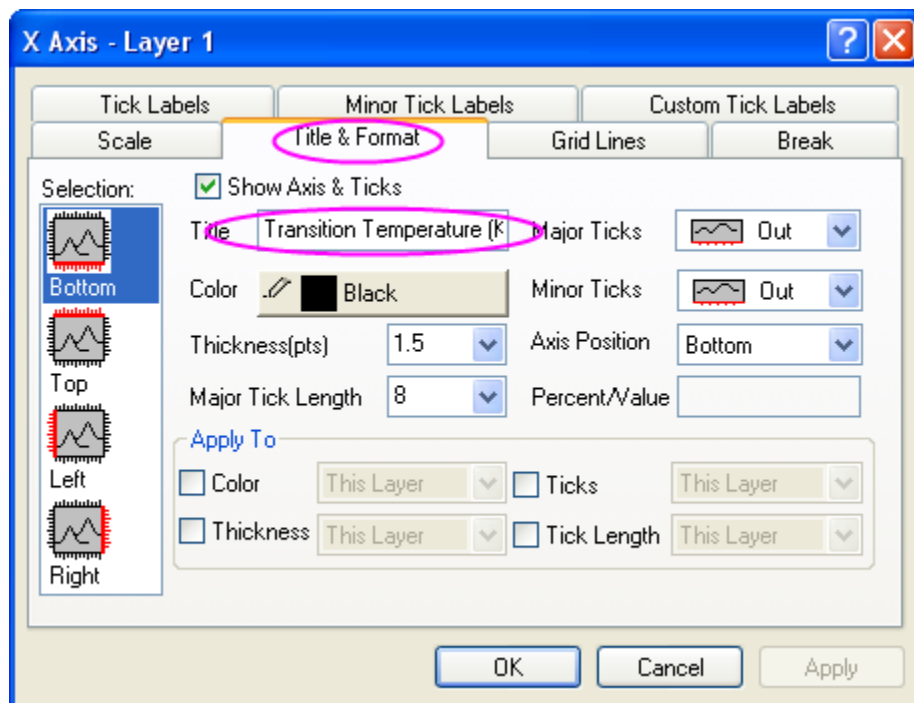
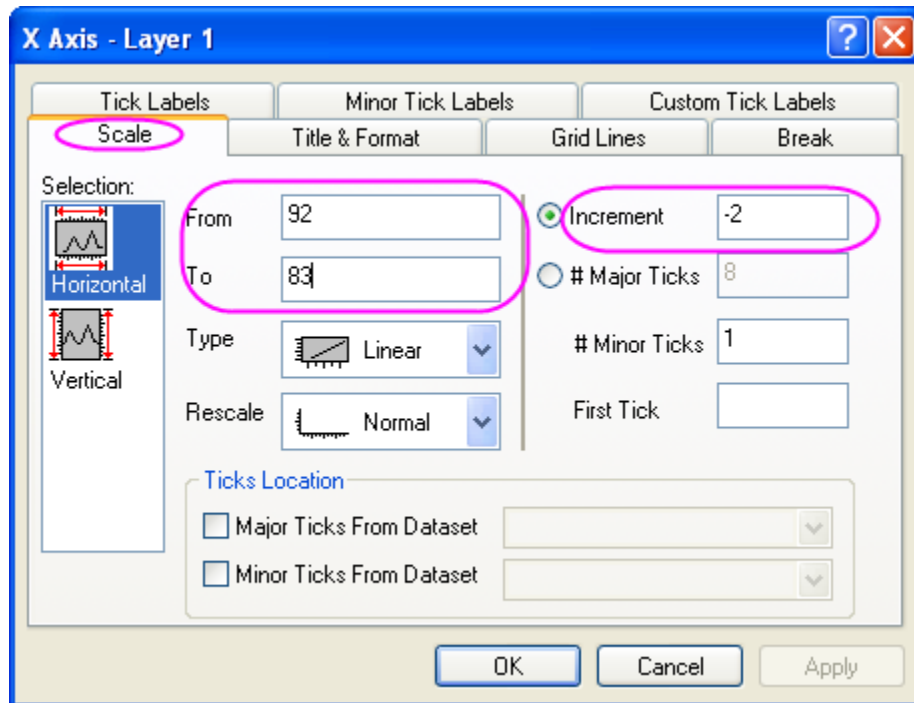
5. Double-click the Y axis, and set the Vertical axis scale **From 0 To 35**. Then active the **Grid Lines** tab, enable **Horizontal Major Grid** with **Dash** line. Then check the **Opposite** checkbox for both Horizontal and Vertical axis. Go to the the **Tick Labels** tab, choose the **Bottom** icon from the left panel. Choose **Text from dataset** from the **Type** drop-down list and choose **[Book01]Data!D** from the Dataset drop-down list. Click OK to apply these settings.
6. Delete or customize the legend and axis title as you need.



- o Choose the Annealing plot in Layer2, then change the **Plot Type, Size and Color** as below.



- o Choose the Err plot in Layer2, then change **Color to Gray** in the **Error Bar** tab.
 - o Click the **OK** button to finish scatter settings.
4. Double click on X axis, in the opened dialog, select the **Scale** tab and change **From 92 To 83** with **Increment** of -2. Then choose the **Title & Format** tab and set **Title** to *Transition Temperature (K)*. Click the **OK** button.

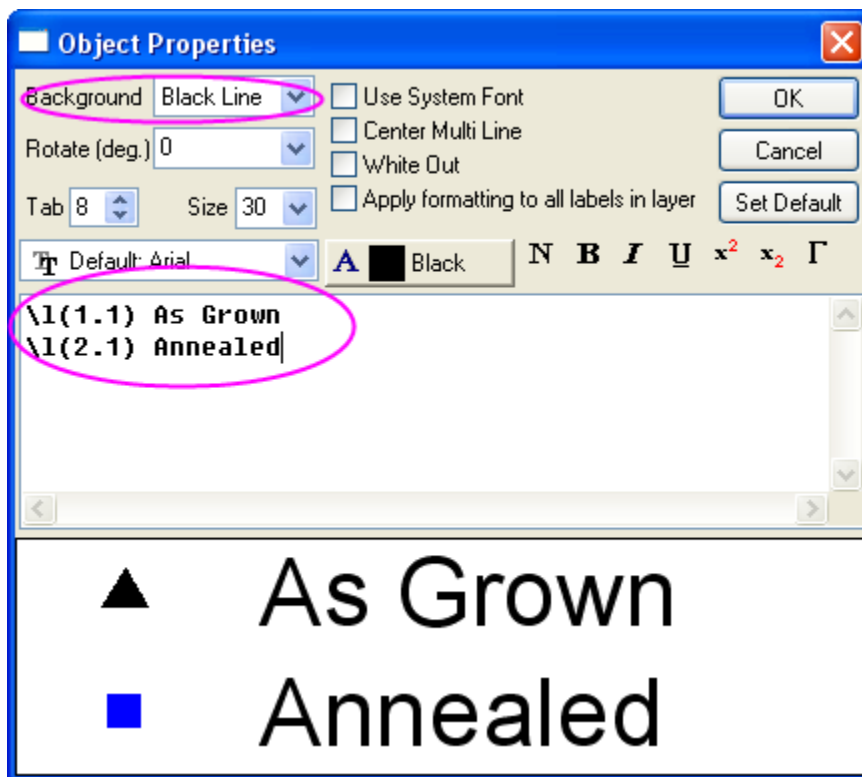


5. Double click on the left Y axis, in the opened dialog, do the same thing as last step (step 4), but with scale type of *Log10*, scale **From** $1E-4$ **To** 100 , **Increment** of 1 , and **Title** to *Deposition Pressure (Torr)*.
6. Double click on the right Y axis, in the opened dialog, do the same thing as step 4, but with scale **From** 765 **To** 795 , **Increment** of 10 , **Title** to *Annealing Temperature ($\backslash+(0)C$)*, and both **Major Ticks** and **Minor Ticks** to *In*.

- Click the legend to select it and then choose the **Properties** from the right-click menu to open the **Object Properties** dialog. Change **Background** to *Black Line*, and in the text field, enter the following text.

\(1.1) As Grown

\(2.1) Annealed

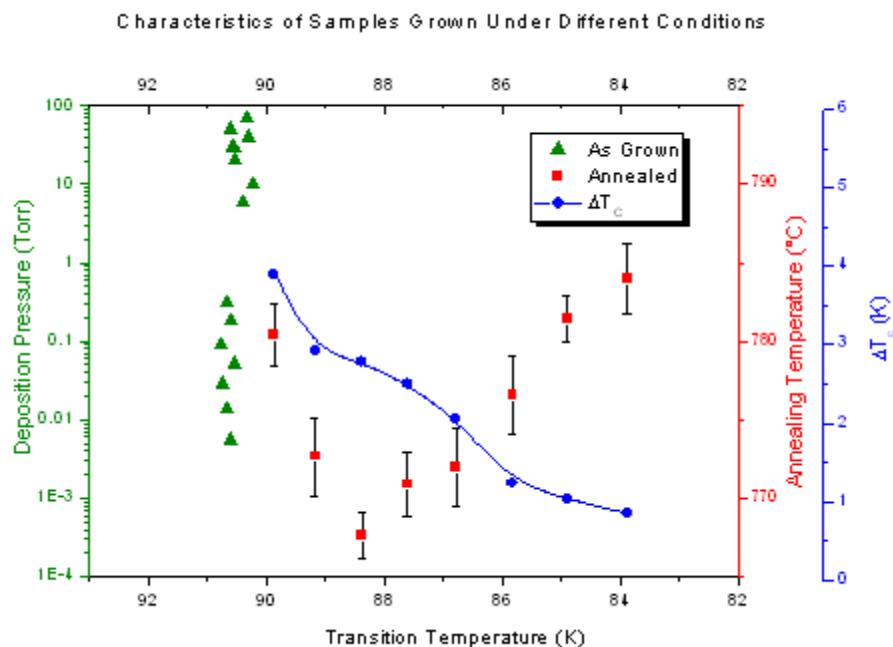


- Move the legend to the proper position.

3.4.2 3Ys Y-YY

Summary

This tutorial will show how to create a graph with three Y axis, one left Y axes and double right Y axis.



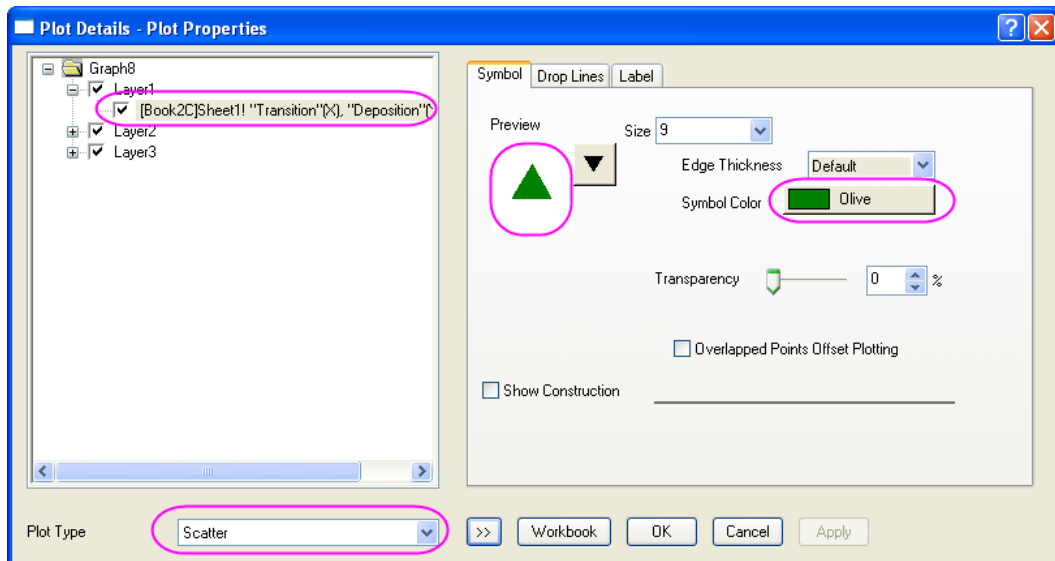
What you will learn

- Create a graph with three Y axis, one left and double right
- Customize settings of data scatters
- Change axis scale, type, title, etc.
- Show a axis
- Update legend

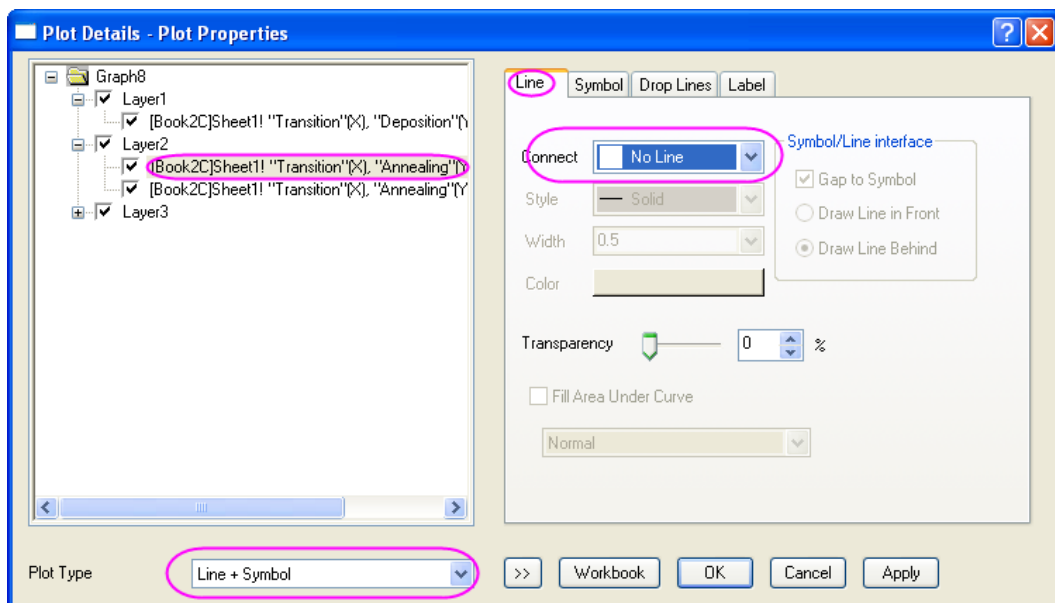
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

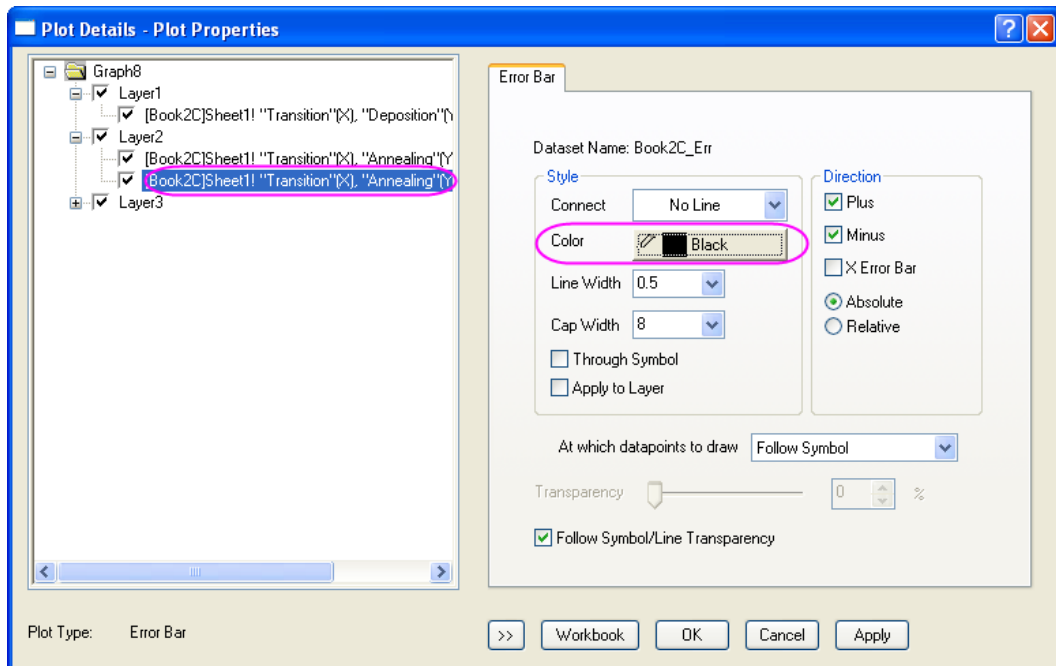
1. Open the sample project `<Origin Installation Directory>\Samples\Graphing\Sample Graphs.opj`, and then browse to the **Graphing: Multi-Curve: 3Ys Y-YY** folder in Project Explorer.
2. Highlight all columns in the data worksheet, then choose menu **Plot: Multi-Curve: 3Ys Y-YY** to create a graph with three Y axes, one left Y and two right Y.
3. Customize data plots.
 - Double click on the plot to open the **Plot Details** dialog. In the left panel, select the plot in Layer1, then change the plot type, symbol and color as below.



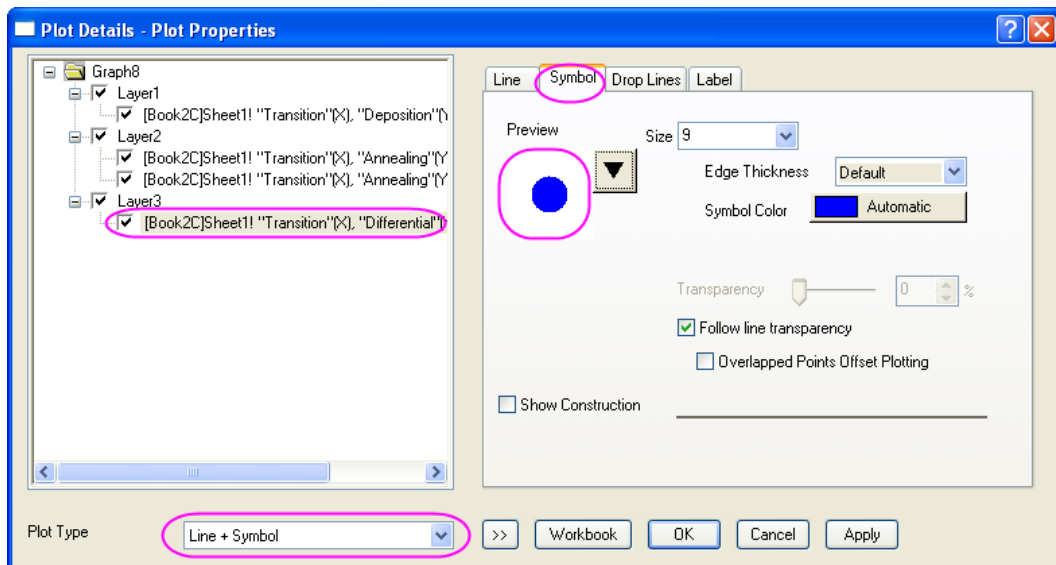
- Choose the Annealing plot in Layer2, then change the plot type to **Line + Symbol**. In the right panel, go to the **Line** tab, set **Connect** to **No Line**.



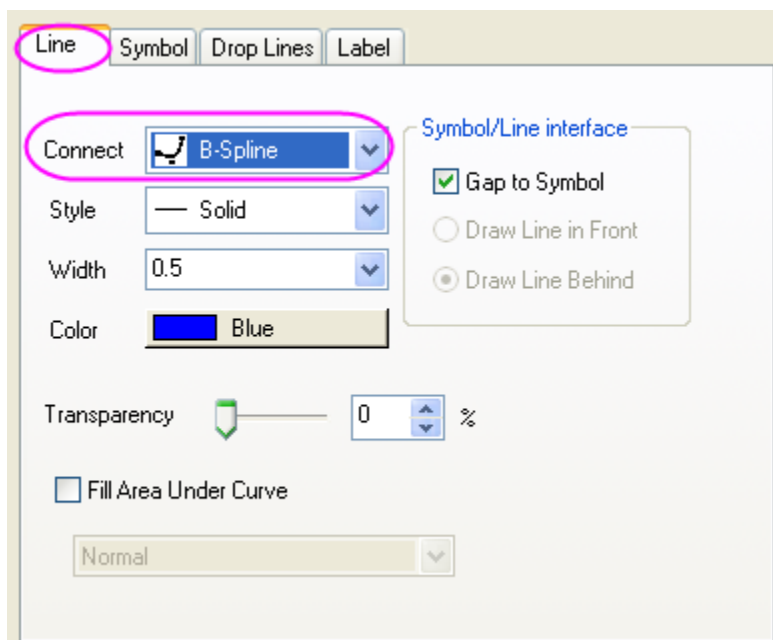
- Choose the Err plot in Layer2, then change color to Black.



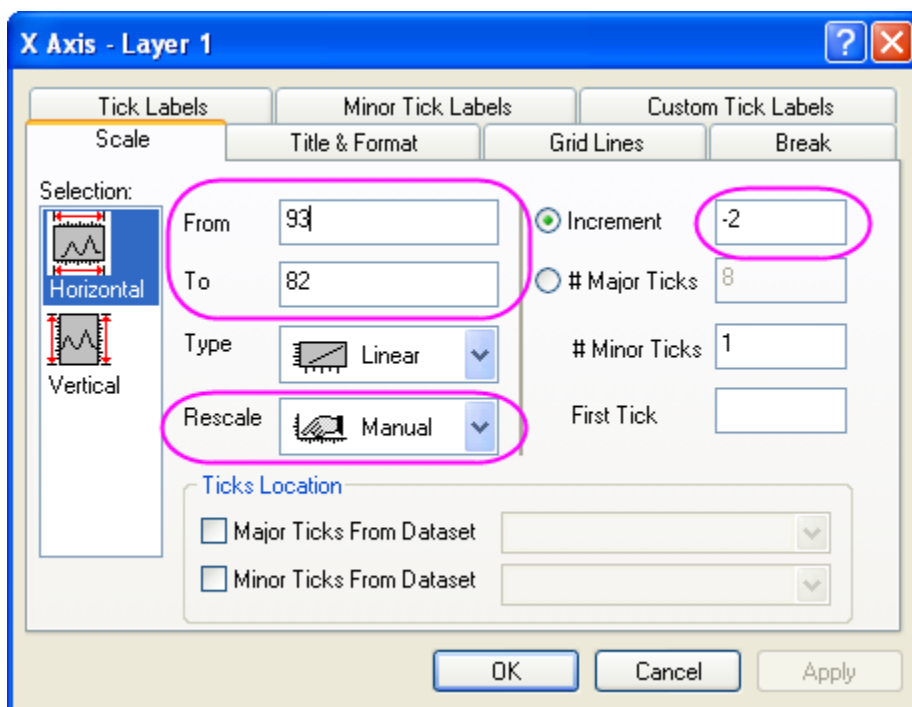
- Choose the plot in Layer3, then in the **Symbol** tab of the right panel, change the plot type to **Line + Symbol** and symbol to solid circle.

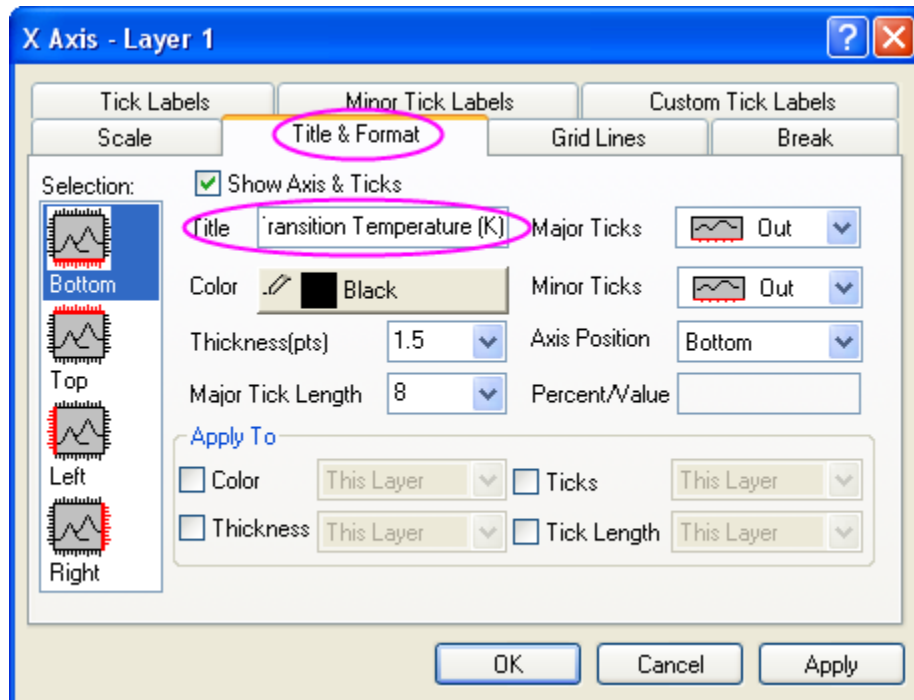


- Select the **Line** tab when the plot in Layer 3 is selected and change **Connect** to **B-Spline**.

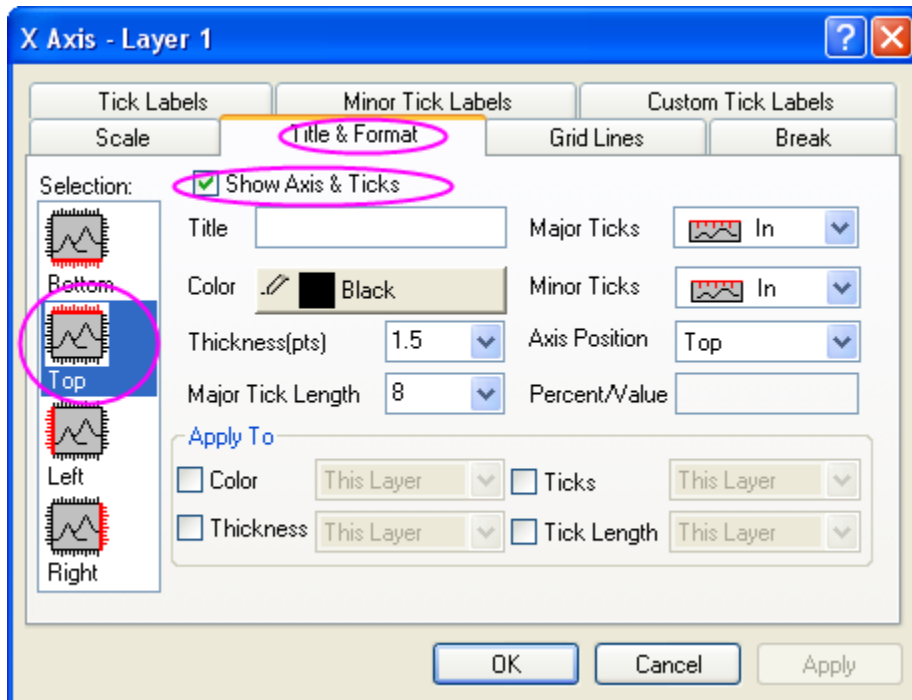


4. Double click on X axis, in the opened dialog, select the **Scale** tab and change **From** 93 To 82 with **Increment** of -2, and select **Manual** for **Rescale**. Then choose the **Title & Format** tab and set **Title** to *Transition Temperature (K)*. Click the **OK** button.

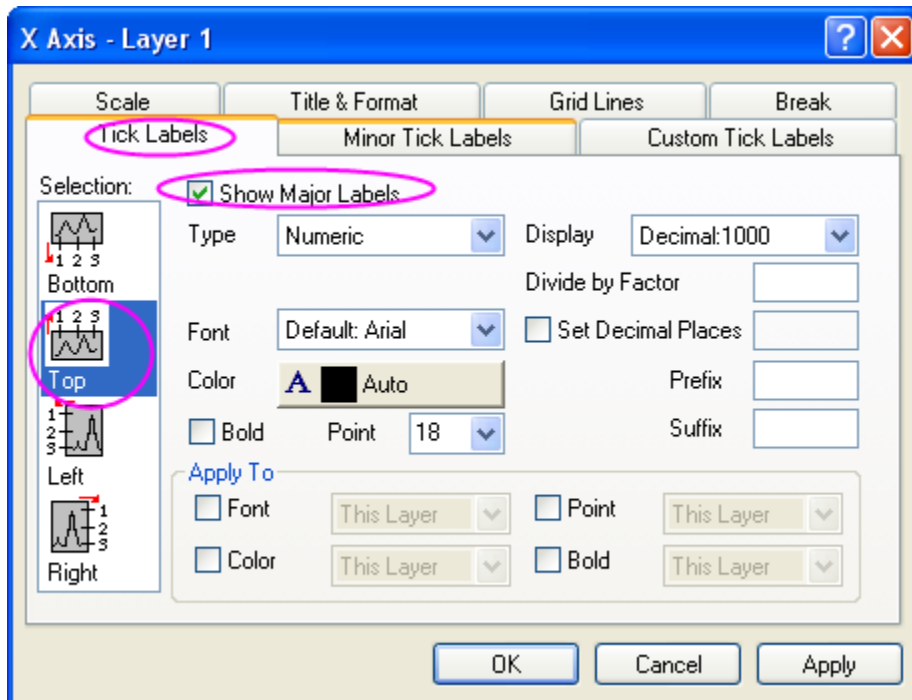




5. Double click on the left Y axis, in the opened dialog, do the same thing as last step (step 4), but with scale type of **Log10**, scale **From $1E-4$ To 100**, **Increment of 1**, and **Title to Deposition Pressure (Torr)**.
6. Double click on the right red Y axis, in the opened dialog, do the same thing as step 4, but with scale **From 765 To 795**, **Increment of 10**, **Title to Annealing Temperature ($\backslash+(0)C$)**.
7. Double click on the right blue Y axis, in the opened dialog, do the same thing as step 4, but with scale **From 0 To 6**, **Increment of 1**, **Title to $\backslashg(D)T\backslash(c)$ (K)**.
8. Show the top axis.
 - o Double click on the bottom X axis, in the opened dialog, select the **Title & Format** tab and then choose **Top** from the **Selection** box. Check the **Show Axis & Ticks** check-box.



- Switch to the **Tick Labels** tab, check the **Show Major Labels** check-box.

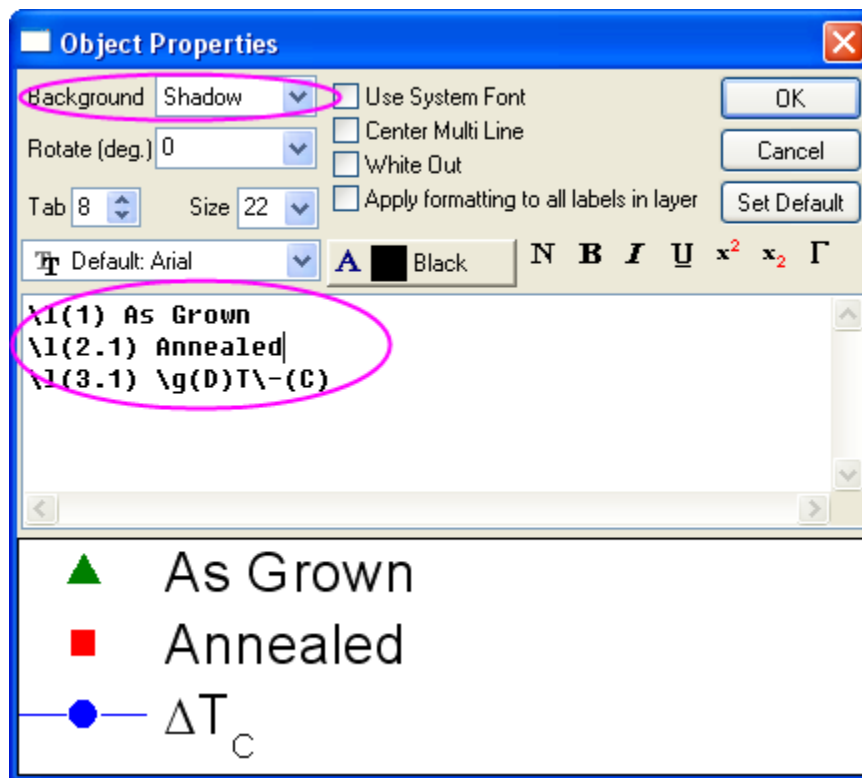


9. Click the legend to select it and then choose the **Properties** from the right-click menu to open the **Object Properties** dialog. Change **Background** to **Shadow**, and in the text field, enter the following text. Then move the legend to the proper position.

\\(1) As Grown

\\(2.1) Annealed

$\backslash(3.1) \backslash g(D)T\backslash-(C)$

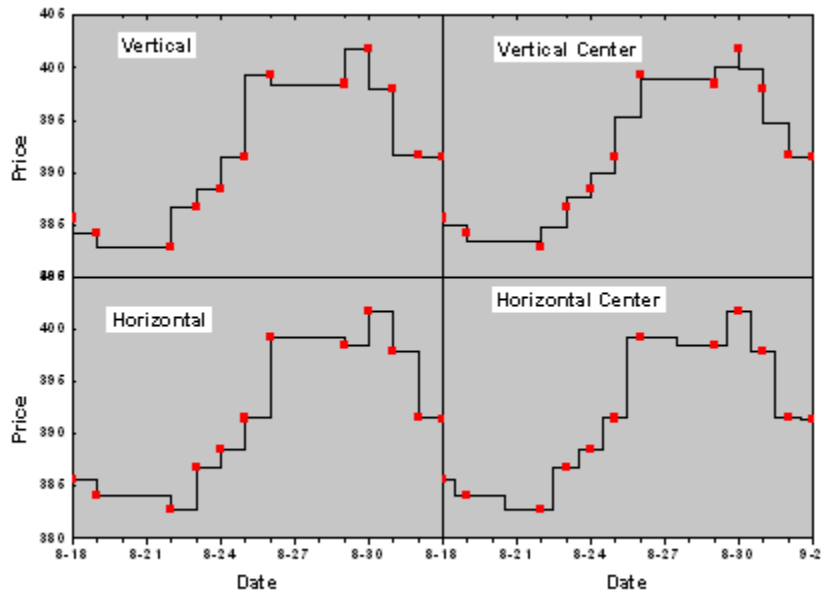


10. Add a text with *Characteristics of Samples Grown Under Different Conditions* as the graph title.

3.4.3 Multiple Layers with Linked Axis

Summary

This tutorial will show how to merge four graphs to one graph, which contains multiple layers with linked axis.



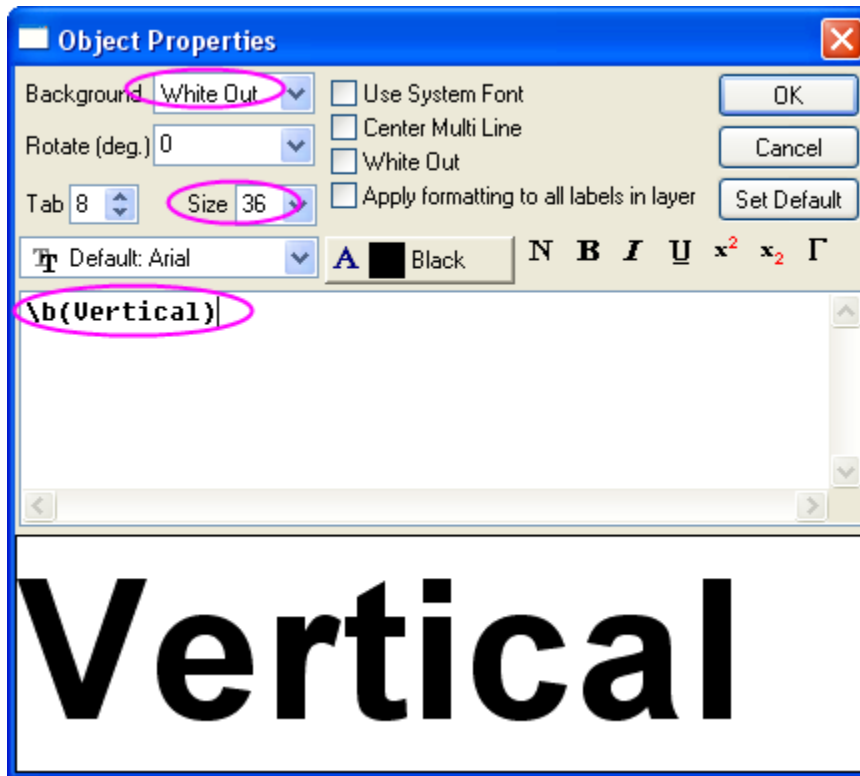
What you will learn

- How to make a Line + Symbol plot
- How to merge graphs
- How to link axis of layers
- How to customize axis

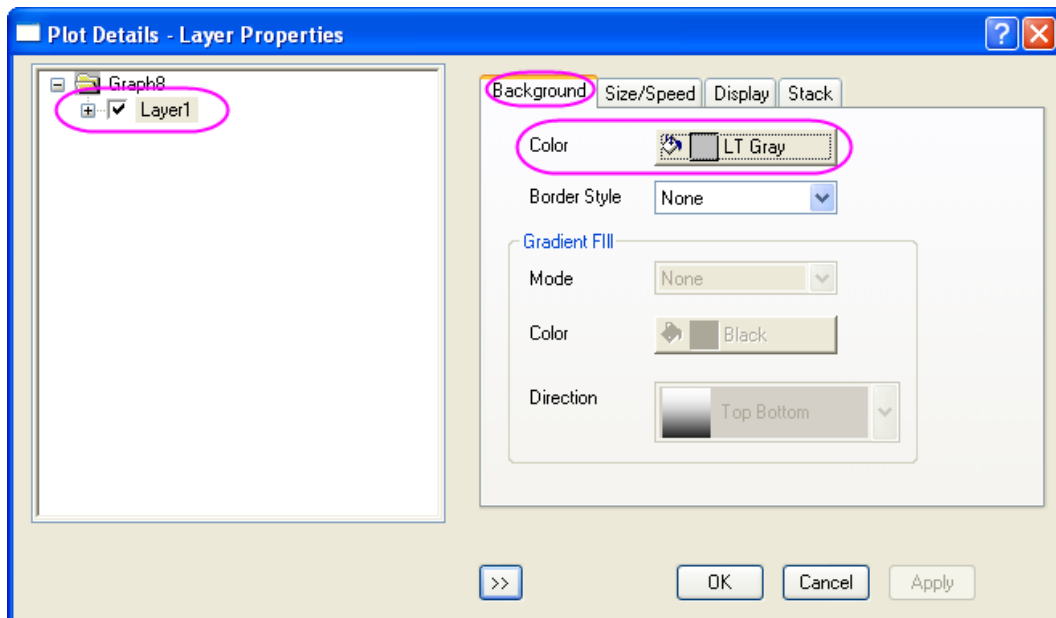
Steps

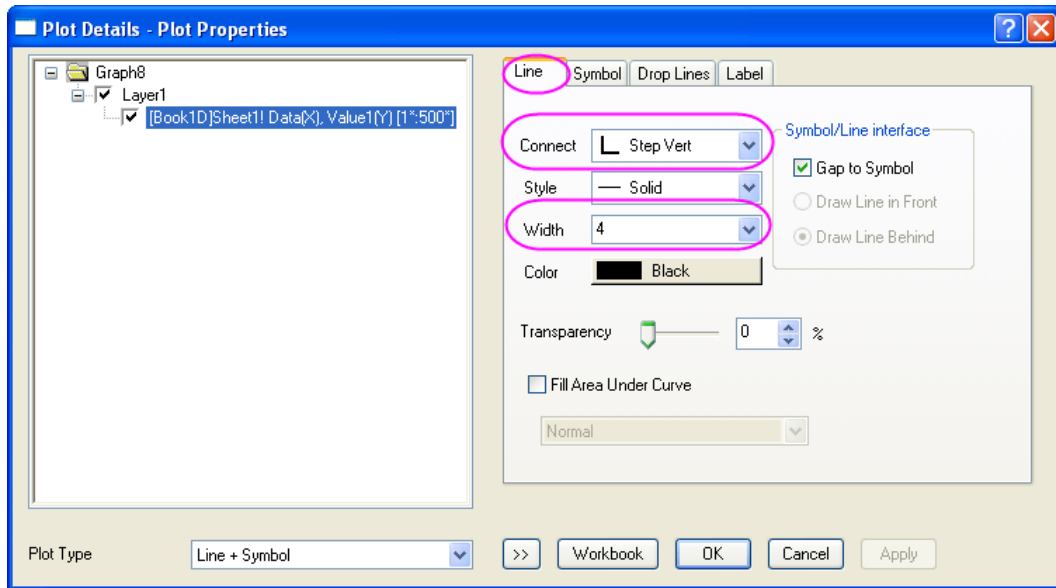
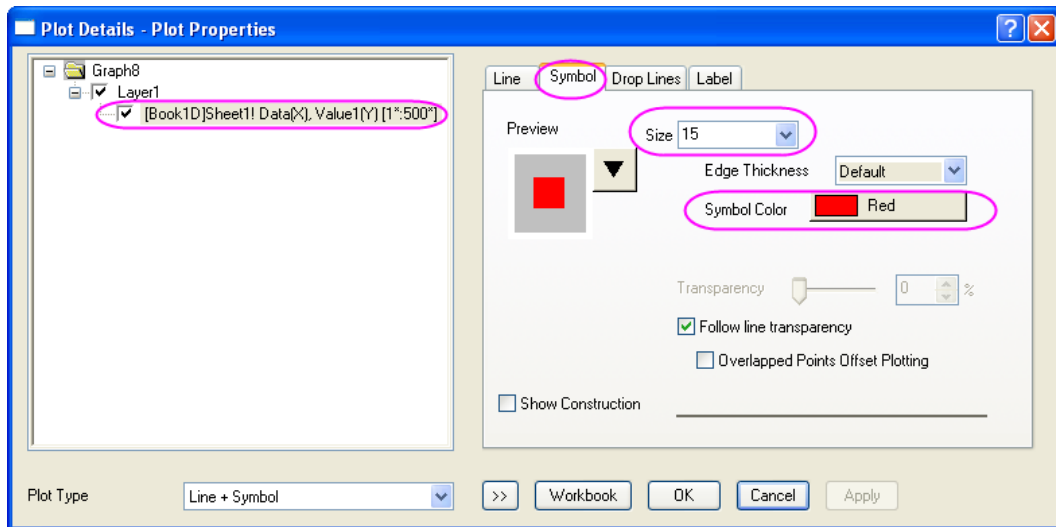
This tutorial is associated with the sample project: `\Samples\Samples.opj`.

1. Open the sample project, open the folder *Graphing: Multiple Layers* in the **Project Explorer**.
2. Highlight column Value1 and make a graph by selecting Origin menu **Plot: Line + Symbol: Line + Symbol**.
3. Activate the graph, select the legend and choose **Properties** from the context menu. In the **Object Properties** dialog, change the **Background** to **White Out**, **Size** to 36 and put the following text to the text field. Click the **OK** button. Move the legend to the right position.
`\b(Vertical)`



4. Double click on the open white space in layer of the graph to open the **Plot Details** dialog. In the left panel, select **Layer1**. Activate the **Background** tab in the right panel, choose *LT Gray* for the **Color**. Then select the plot in **Layer1** from the left panel, under the **Symbol** tab in the right panel, set **Size** to 15 and **Symbol Color** to red. Switch to the **Line** tab, choose **Step Vert** from the **Connect** drop-down list and set **Width** to 4.





5. Repeat step 2 to step 4 for column Value2, Value3 and Value4 respectively. For each column, the legend text and **Connect** drop-down list will be different, listed below.

For column **Value2**:

legend text = \b(Vertical Center)

Connect = Step V Center

For column **Value3**:

legend text = \b(Horizontal)

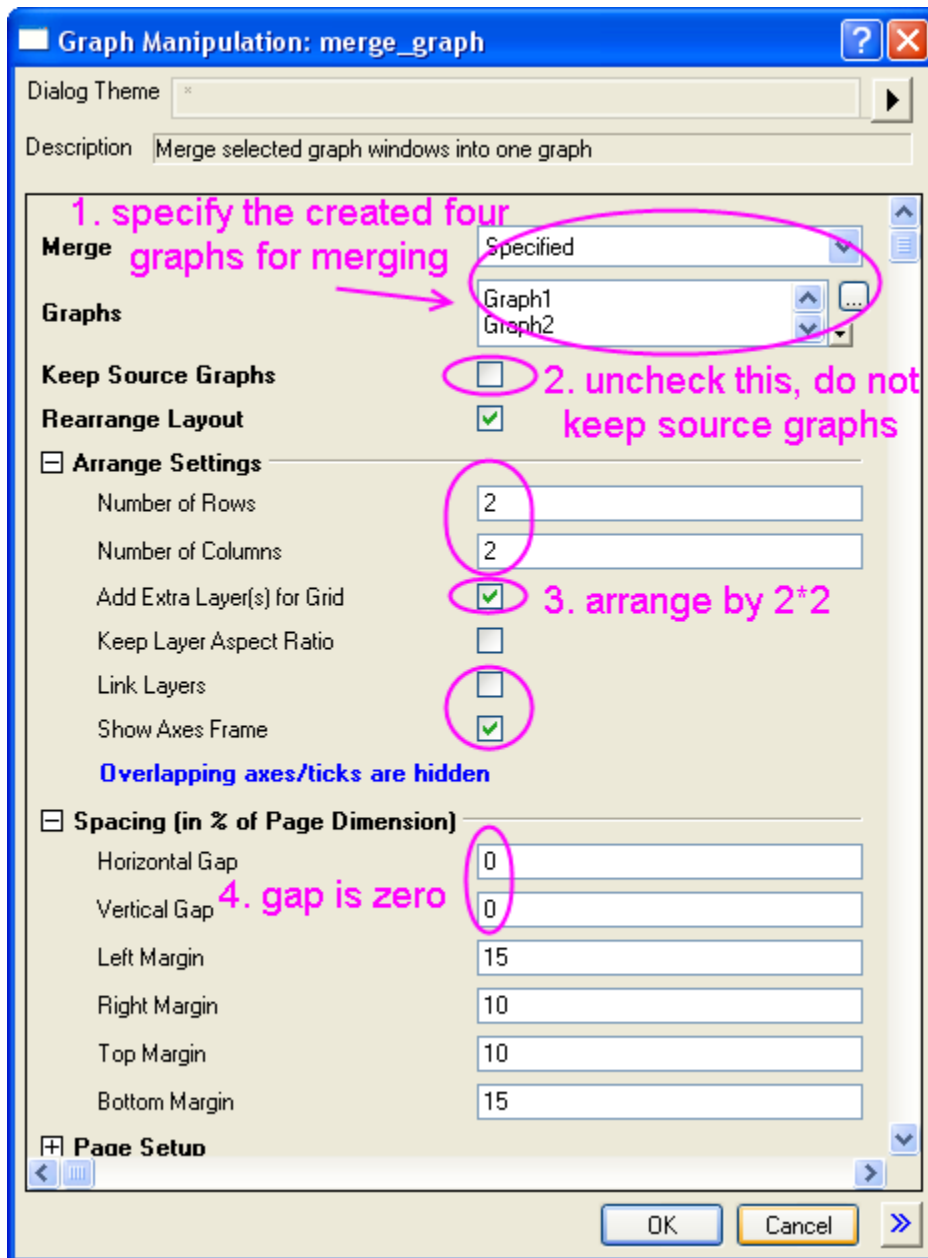
Connect = Step Horz

For column **Value4**:

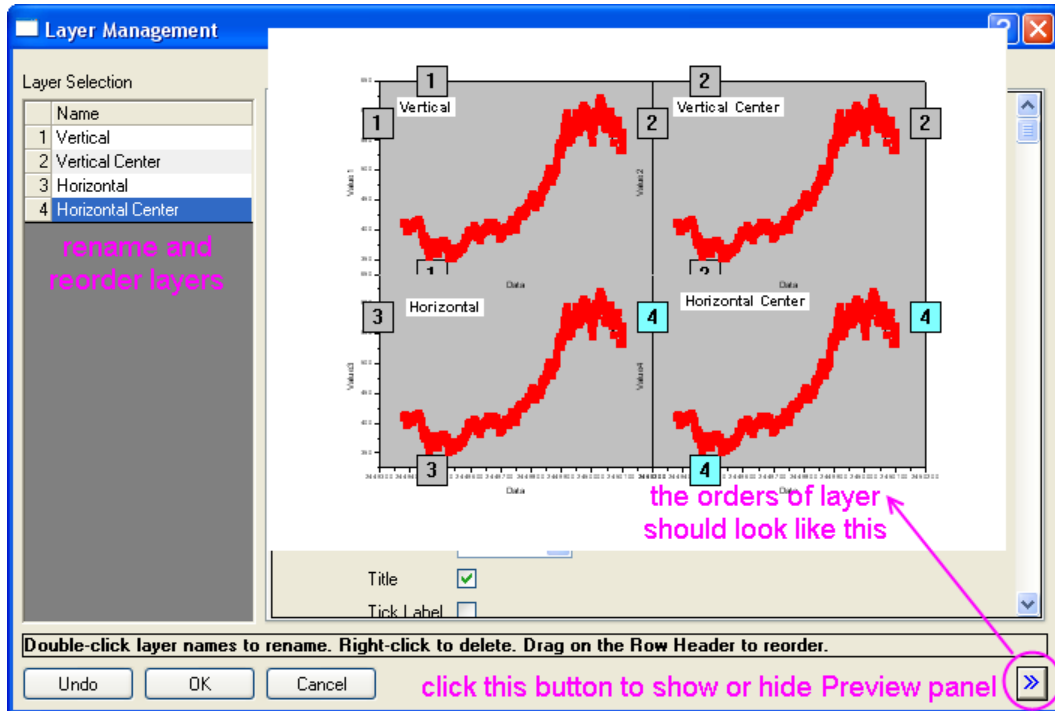
legend text = \b(Horizontal Center)

Connect = Step H Center

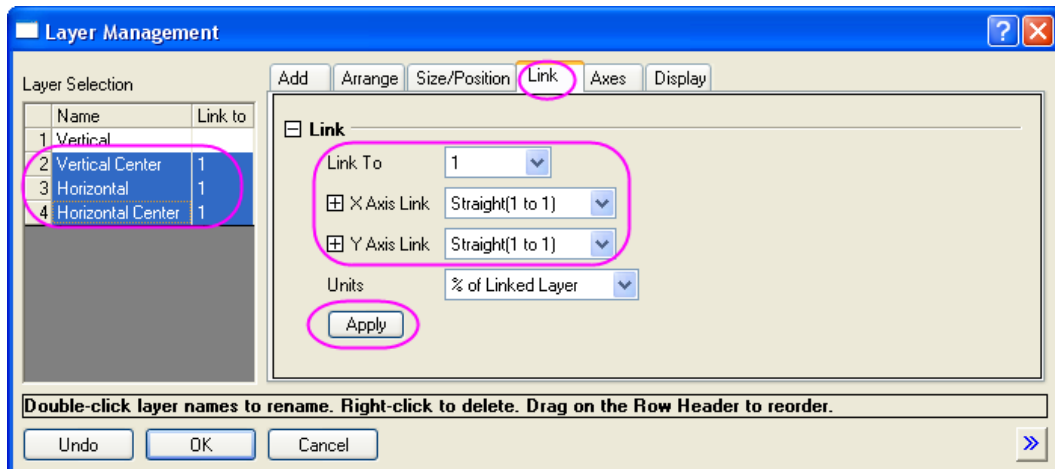
6. Activate a graph and then select Origin menu **Graph: Merge Graph Windows** to open the **merge_graph** dialog. Change the settings as the following image shows. Then click the **OK** button to merge these graphs.



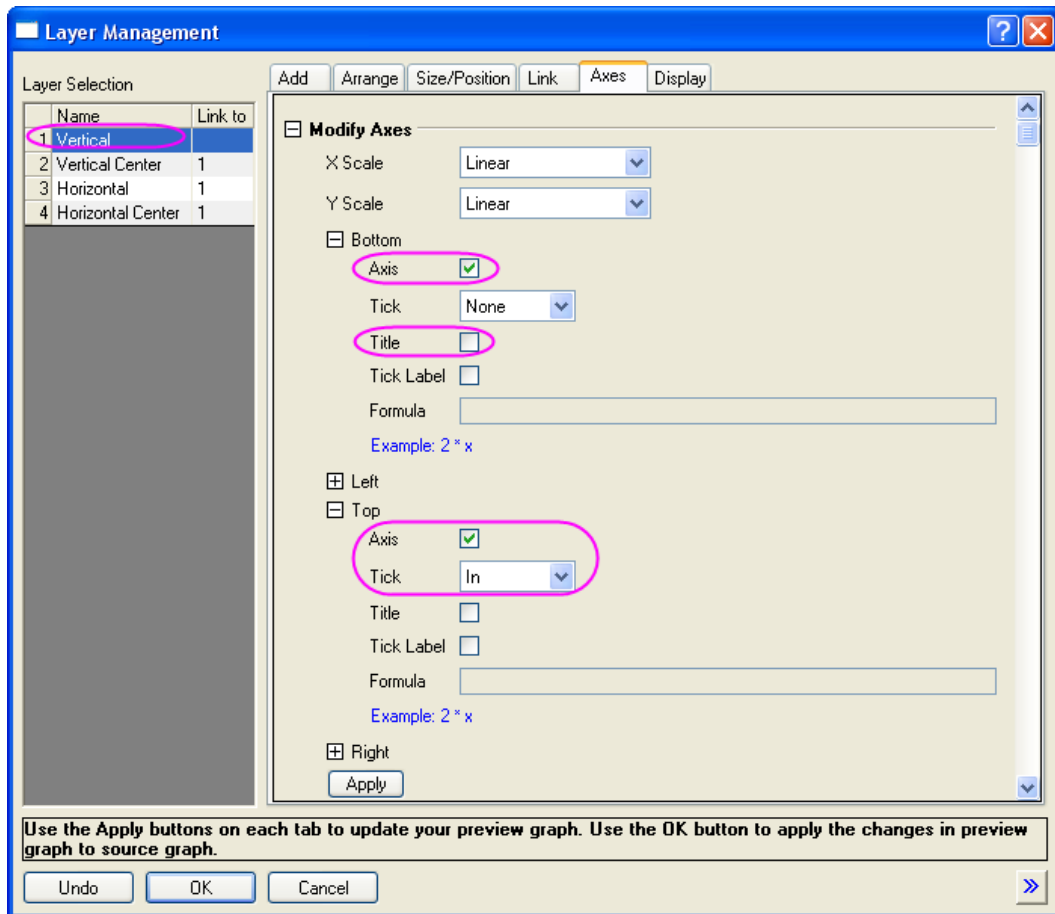
7. Activate the new merged graph, choose Origin menu **Graph: Layer Management** to open the **Layer Management** dialog. In the **Layer Selection** panel, rename the layers by double clicking on the name and reorder the layers by clicking on the layer index and dragging up and down. Make sure the final layers' names and order are the same as the image shows below (In the **Preview** panel, the layers' indices and positions can be shown).



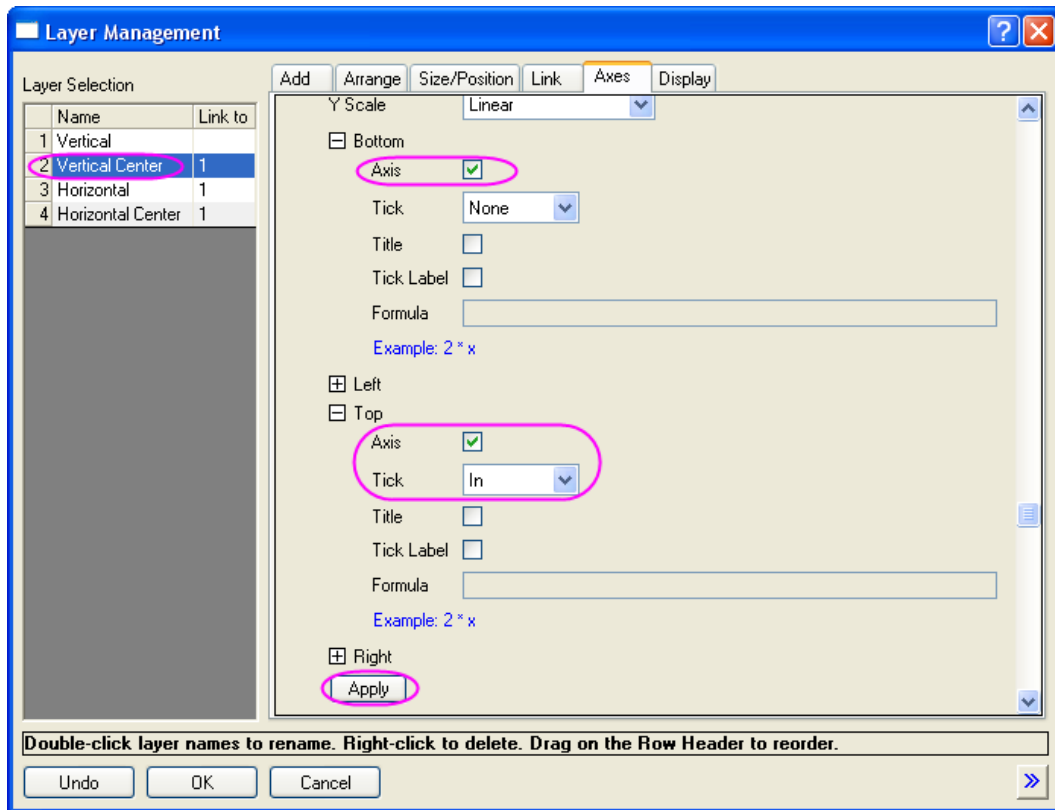
8. In the **Layer Selection** panel, press **Ctrl** key on the keyboard to select the following layers: *Vertical Center*, *Horizontal* and *Horizontal Center*. Go to the **Link** tab, select **1** from the **Link To** drop-down list. Then both **X Axis Link** and **Y Axis Link** are set to **Straight(1 to 1)**. Then click the **Apply** button.



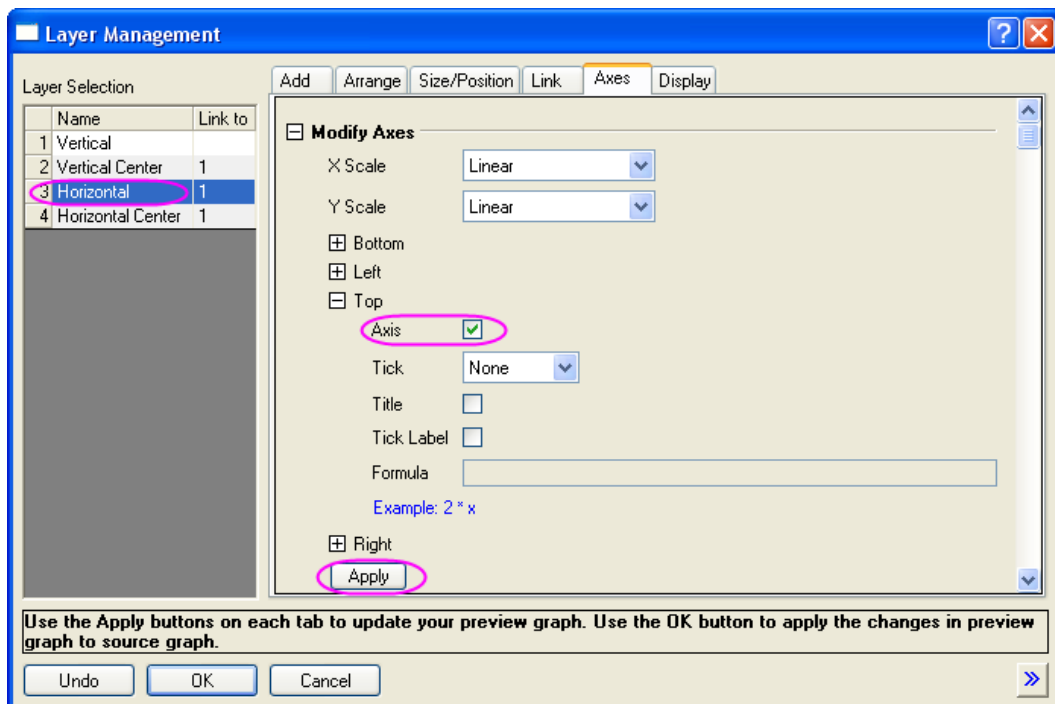
9. Go to the **Axes** tab, select *Vertical* layer from the left panel, then check all **Axis** check-boxes under **Bottom** branch, **Left** branch, **Top** branch and **Right** branch. Also, uncheck the **Title** check-box under **Bottom** branch and select *In* for **Tick** drop-down list under **Top** branch. Click the **Apply** button.



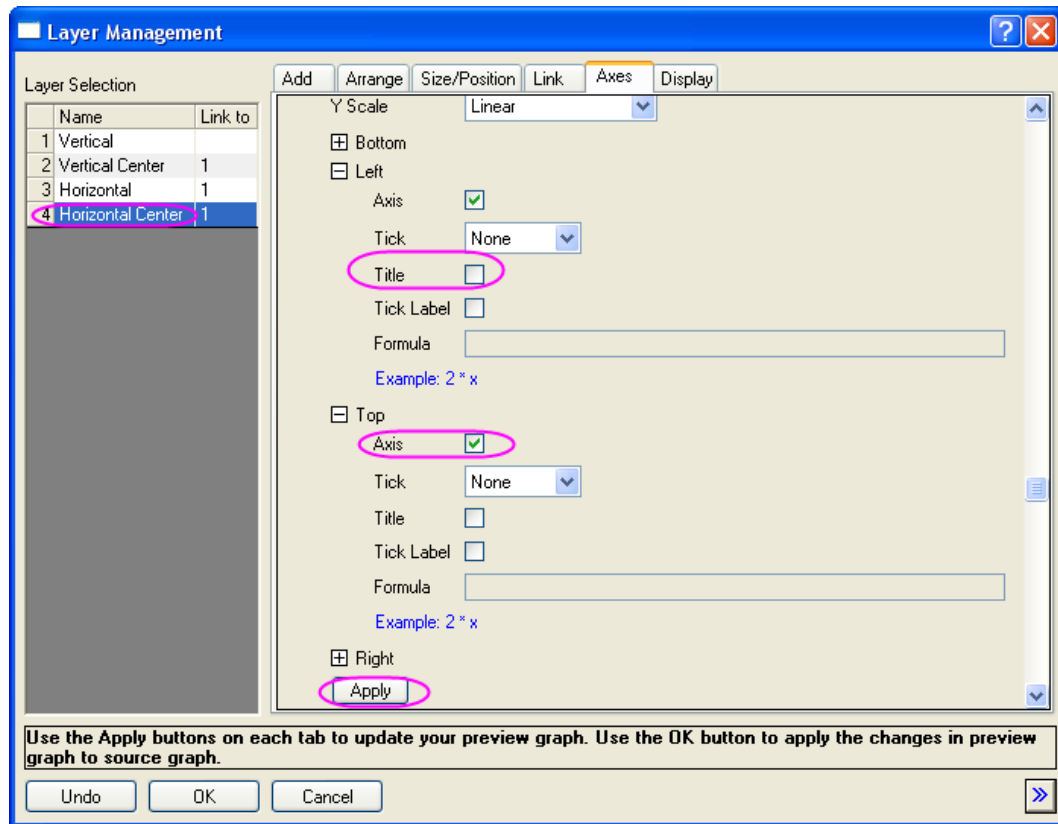
10. Select *Vertical Center* layer, in the **Axes** tab, **Bottom** branch, **Left** branch and **Right** branch share the same settings. And settings for **Top** branch shows in the image below. Click the **Apply** button.



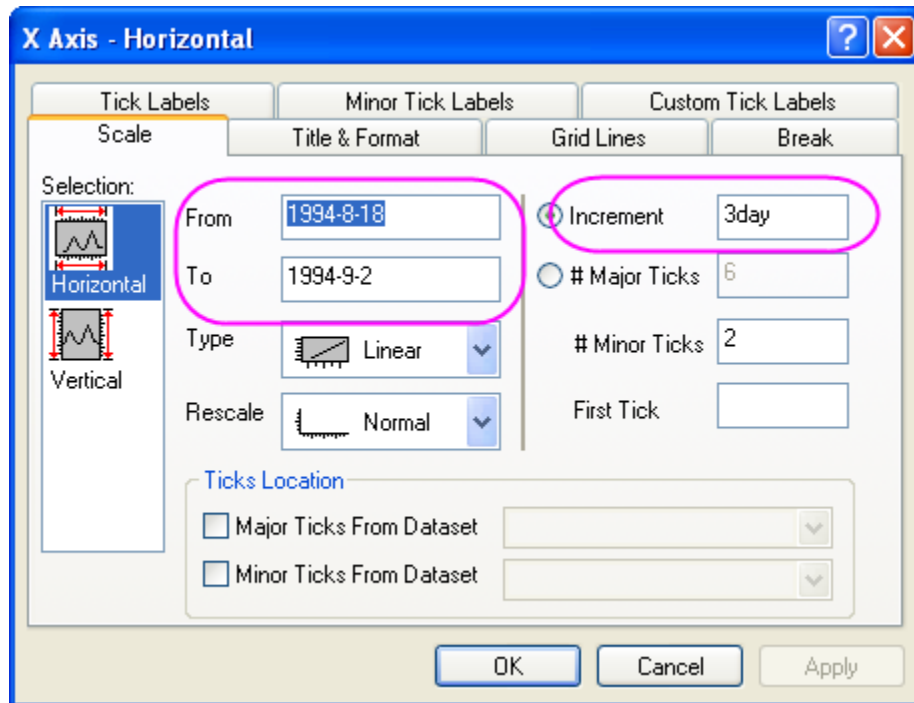
11. Select *Horizontal* layer, in the **Axes** tab, check the **Axis** check-boxes under both **Top** branch and **Right** branch. Click the **Apply** button.



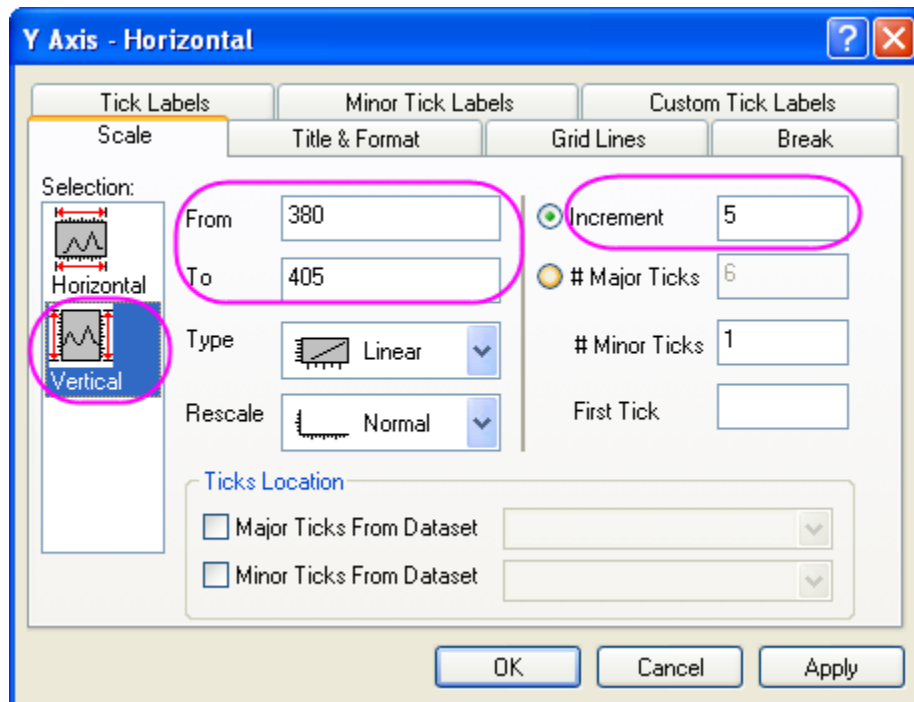
12. Select *Horizontal Center* layer, in the **Axes** tab, uncheck the **Title** check-box under **Left** branch and check the **Axis** check-box under **Top** branch. Click the **Apply** button.



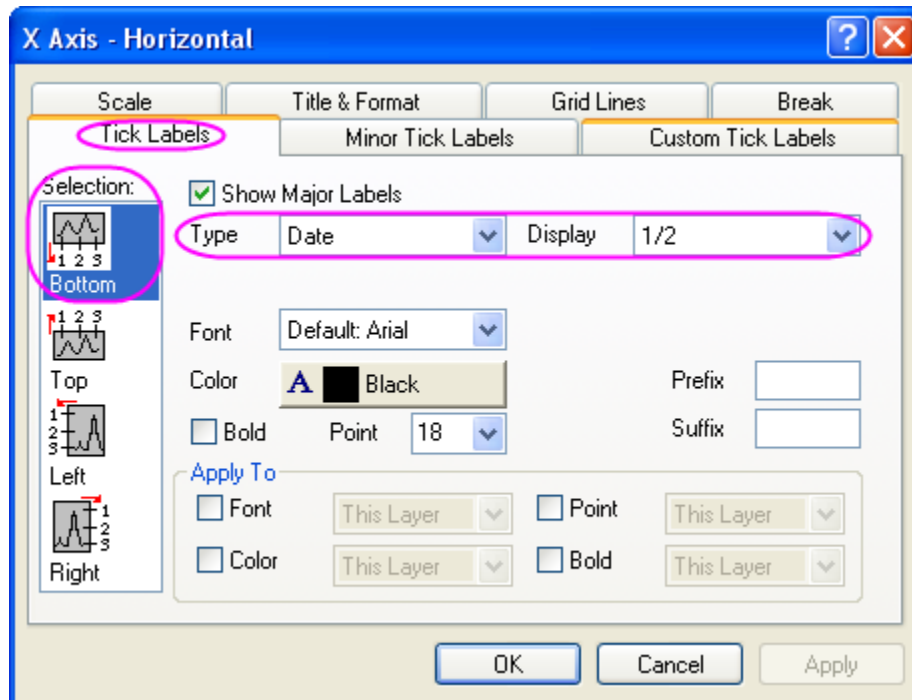
13. Click the **OK** button to close the **Layer Management** dialog.
14. Double click on the X axis of the lower-left layer to open the **X Axis** dialog. In the **Scale** tab, select **Horizontal** from the **Selection** panel, change **From**, **To**, **Increment** and **Minor Ticks** as the following image shows.



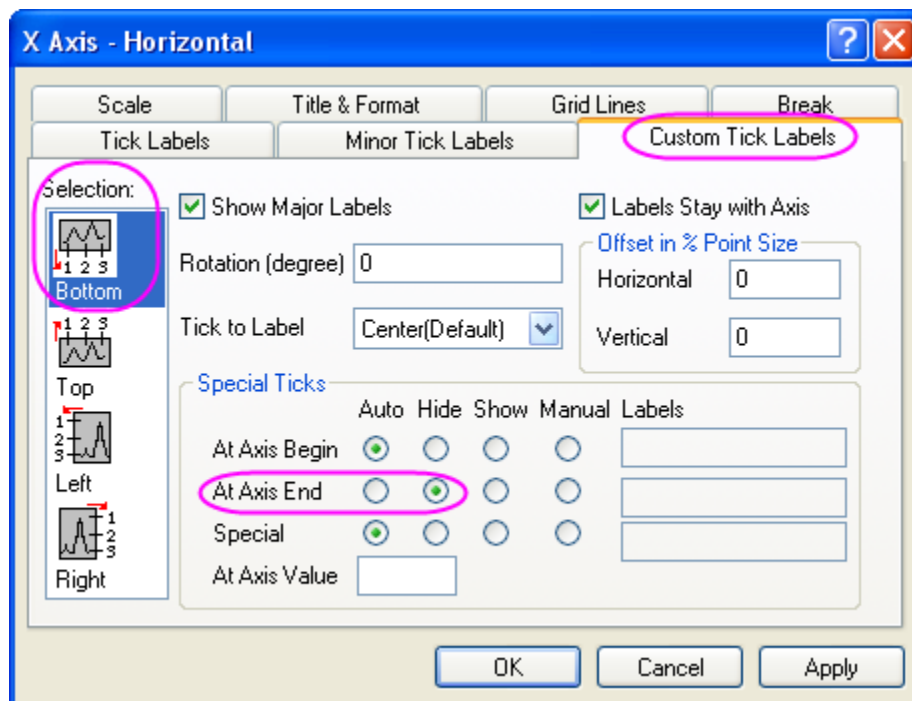
15. Select **Vertical** from **Selection** panel, change the settings as the following image shows.



16. Go to the **Tick Labels** tab, select **Bottom** from the **Selection** panel, set **Type** to *Date* and **Display** to $1/2$.



17. Go to the **Custom Tick Labels** tab, select **Bottom** from the **Selection** panel, choose **Hide** for the **At Axis End**.

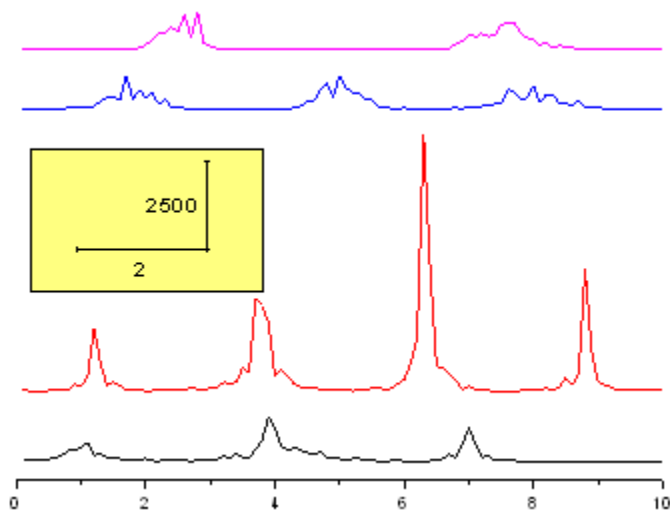


18. Repeat step 14 to step 17 for the X axis of the lower-right layer.
 19. Select the labels of X axis and Y axis, then in the **Format** toolbar, change the size to 30 and click down the **Bold** button.
 20. Change the Y title to **Price** and then set size of X and Y title to 36.

3.4.4 Stack Lines by Y Offsets

Summary

This tutorial will show how to create a graph with stack lines by Y offsets.



What you will learn

- How to create a graph with stack lines by Y offsets
- How to customize a rectangle object
- How to add new XY scaler

Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

1. Open the sample project `<Origin Installation Directory>\Samples\Samples.opj`, and then browse to the **Graphing: Multi-Curve: Stack Lines by Y Offsets** folder in Project Explorer.
2. Highlight all columns in the data worksheet of **Book6A**, then choose menu **Plot: Multi-Curve: Stack Lines by Y Offsets** to create a graph with stack lines by Y offsets.
3. Delete the following objects by selecting them and then pressing the **Delete** key on your keyboard:

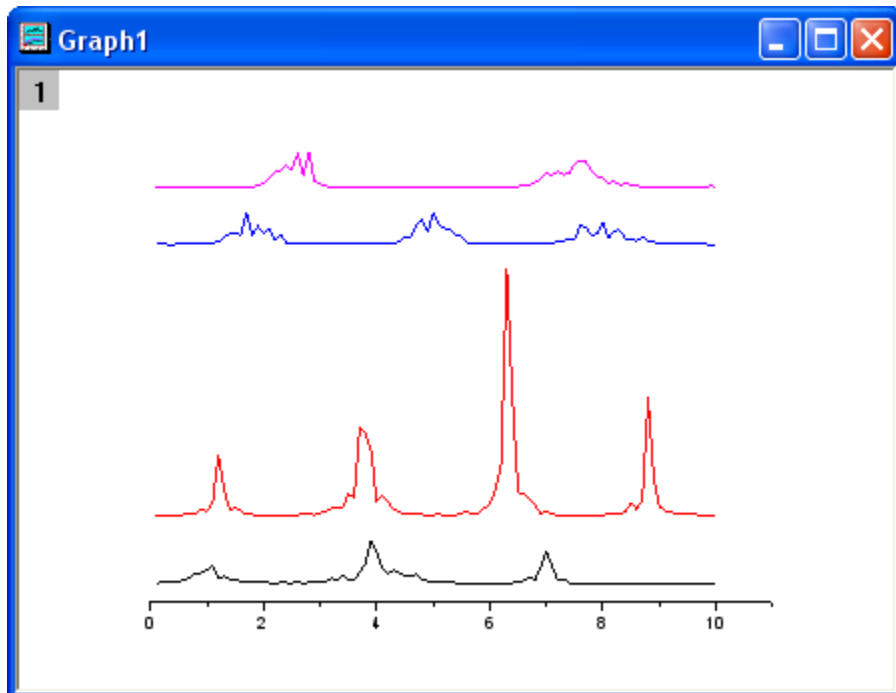
Legend

Y axis

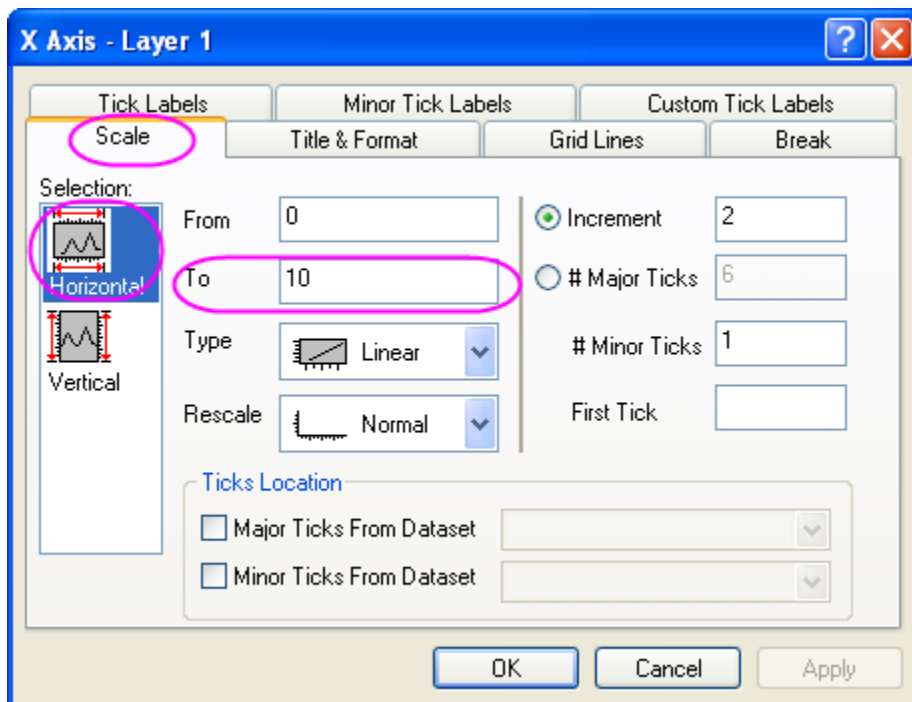
Y axis labels

Y axis title

X axis title



- Double click on the X axis to open the Axis Properties dialog. In the **Scale** tab, make sure that **Horizontal** is selected in the **Selection** list. Change **To** to *10*. Click **OK** to close the dialog.

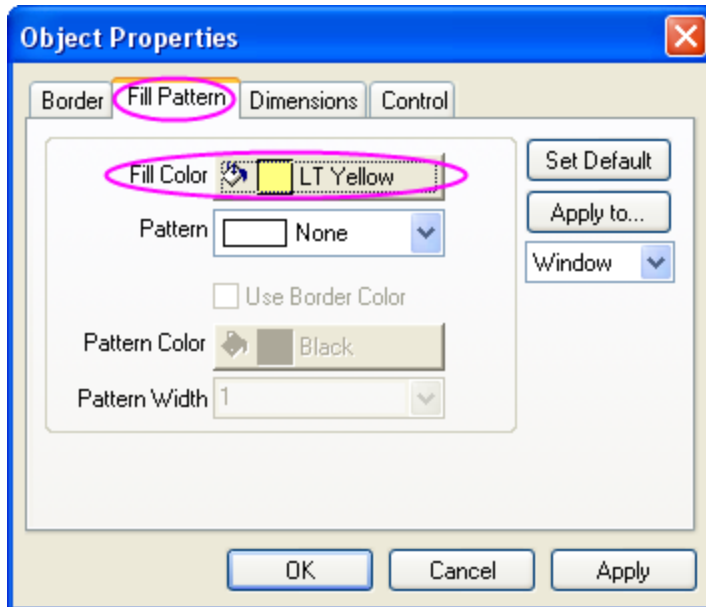


- Click the **Rectangle** button on the **Tools** toolbar and then create a rectangle on the graph.

6. Double click on the rectangle to open its properties dialog. Change the following settings.

Fill Pattern tab

Fill Color = LT Yellow



Dimensions tab

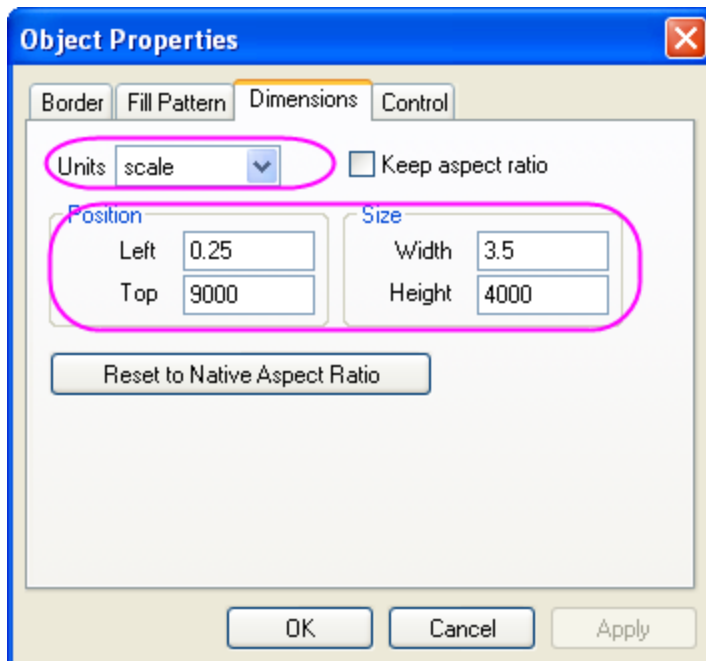
Units = scale

Left = .25

Top = 9000

Width = 3.5

Height = 4000



7. Click OK to close the **Object Properties** dialog.
8. Select **Graph: New XY Scaler** from the Origin menu to create a new XY scaler.
9. Double click on the scaler to open the **Scaler Properties** dialog. Set the following properties in the dialog:

Font Settings branch

Font Size = 24

X branch

Length(Scale) = 2

Title = 2

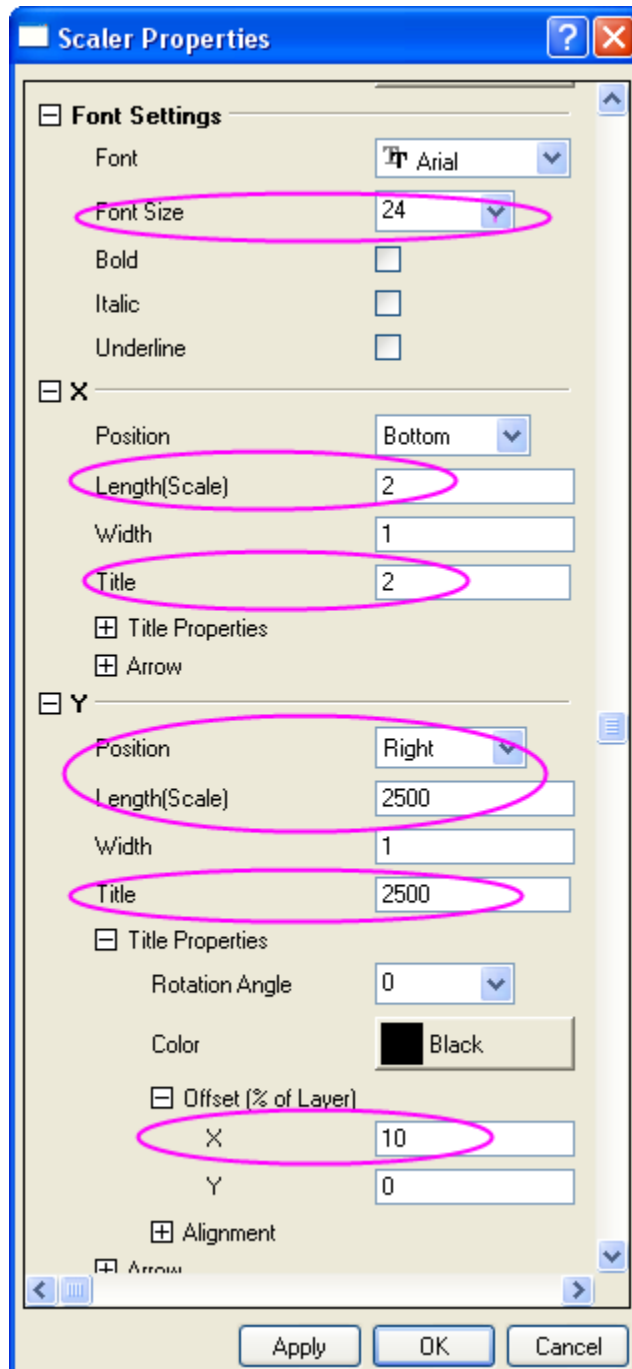
Y branch

Position = Right

Length(Scale) = 2500

Title = 2500

X (in the **Offset** branch under the **Title Properties** branch) = 10

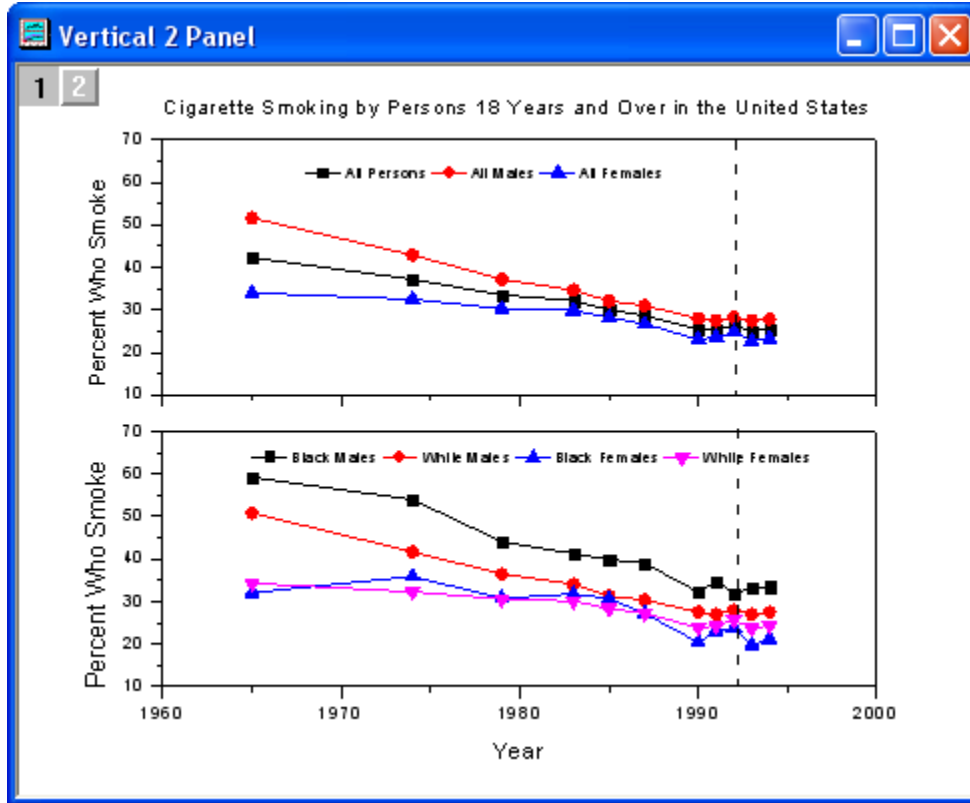


10. Click OK to close the **Scaler Properties** dialog.
11. Move the Scaler object onto the rectangle you created.

3.4.5 Vertical 2 Panel Line

Summary

This tutorial will show you how to merge two graphs and how to customize them.



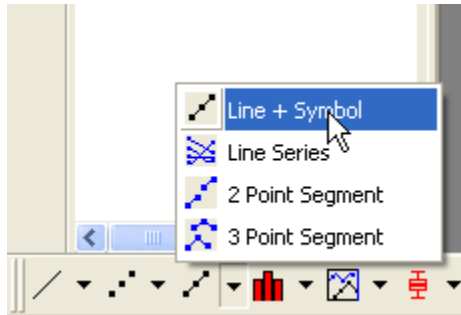
Minimum Origin Version Required: Origin 8.0 SR6

Steps

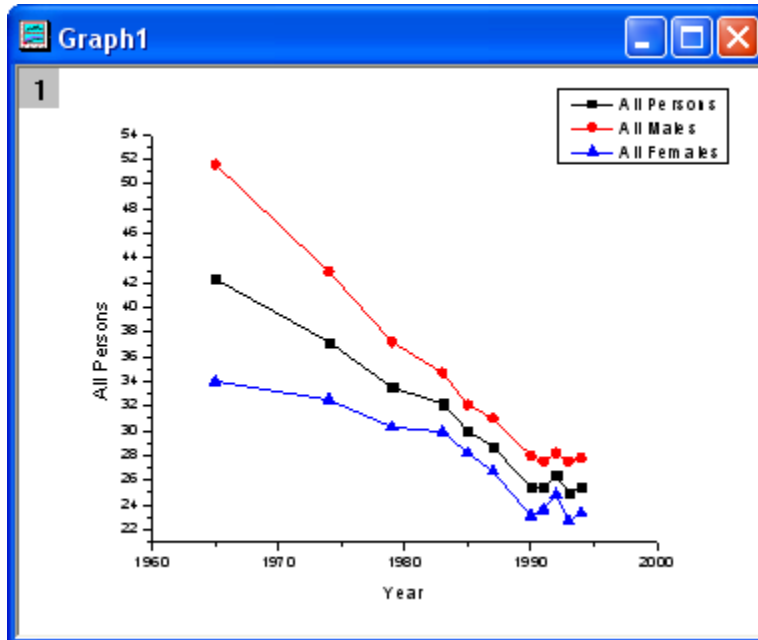
1. Create a new worksheet. Import the data.

	A(Y)	B(Y)	C(Y)	D(Y)	E(Y)	F(Y)	G(Y)	H(Y)
Long Name	Year	All Persons	All Males	All Females	Black Males	White Males	Black Females	White Females
Units								
1	1965	42.3	51.6	34	59.2	50.8	32.1	34.3
2	1974	37.2	42.9	32.5	54	41.7	35.9	32.3
3	1979	33.5	37.2	30.3	44.1	36.5	30.8	30.6
4	1983	32.2	34.7	29.9	41.3	34.1	31.8	30.1
5	1985	30	32.1	28.2	39.9	31.3	30.7	28.3
6	1987	28.7	31	26.7	39	30.4	27.2	27.2
7	1990	25.4	28	23.1	32.2	27.6	20.4	23.9
8	1991	25.4	27.5	23.6	34.7	27	23.1	24.2
9	1992	26.4	28.2	24.8	32	28	23.9	25.7
10	1993	25	27.5	22.7	33.2	27	19.8	23.7
11	1994	25.5	27.8	23.3	33.5	27.5	21.1	24.3
12								

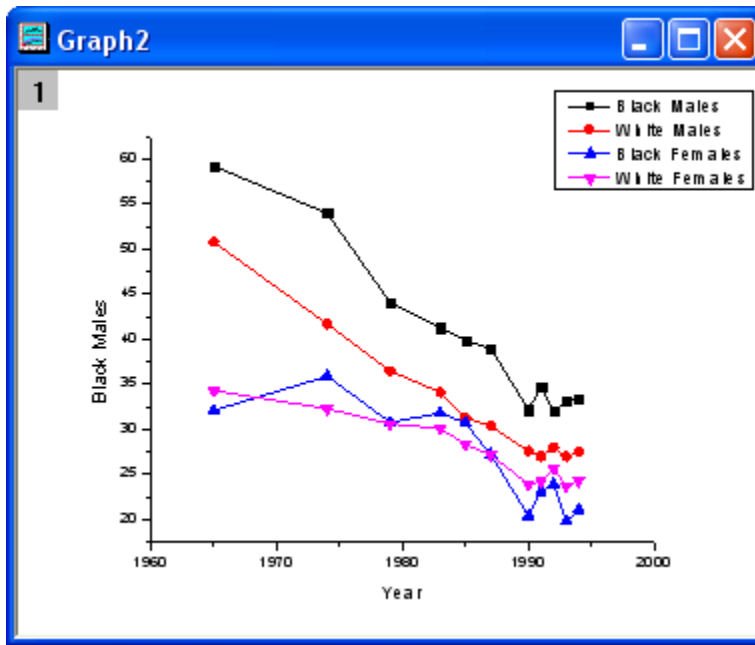
2. Select column 2 to column 4. Click the **Line + Symbol** button on the **2D Graphs** toolbar.



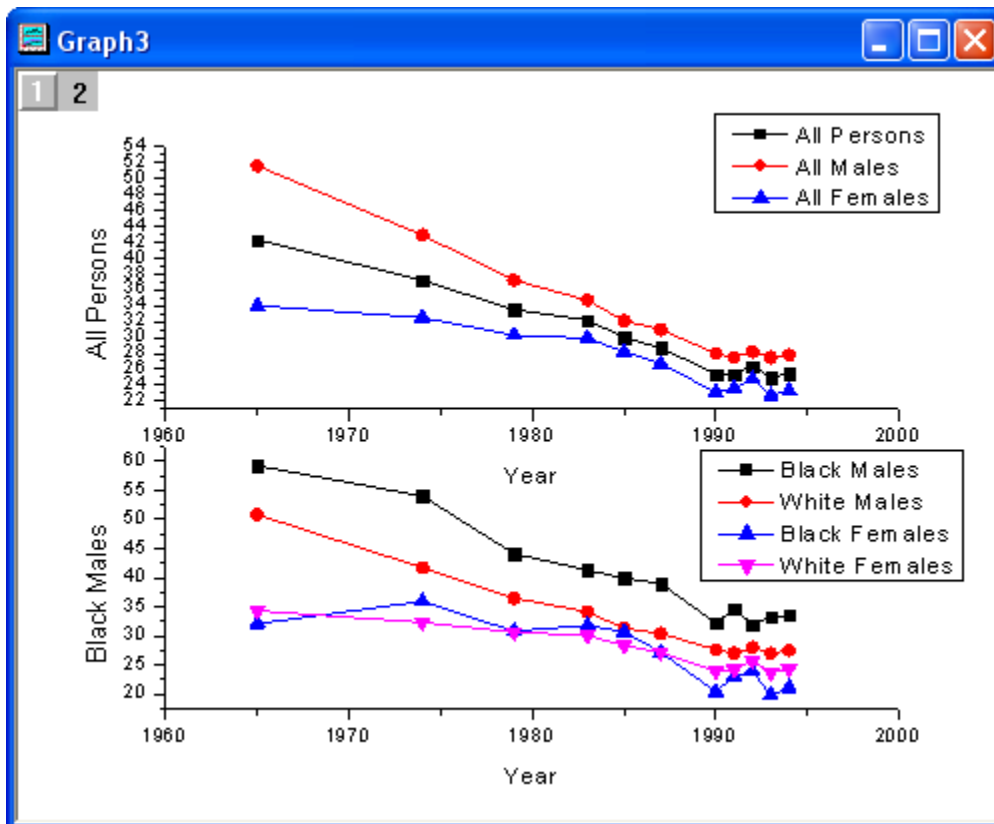
The first graph should look like



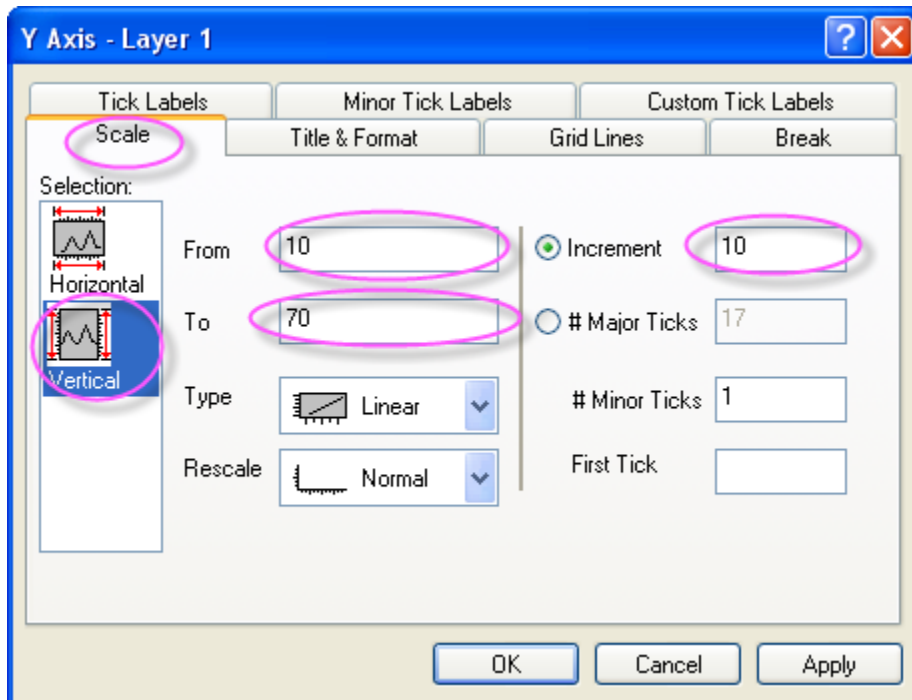
Go back to the worksheet. Select column 5 to column 8 and then click the **Line + Symbol** button on the **2D Graphs** toolbar to create a new graph. The second graph should look like



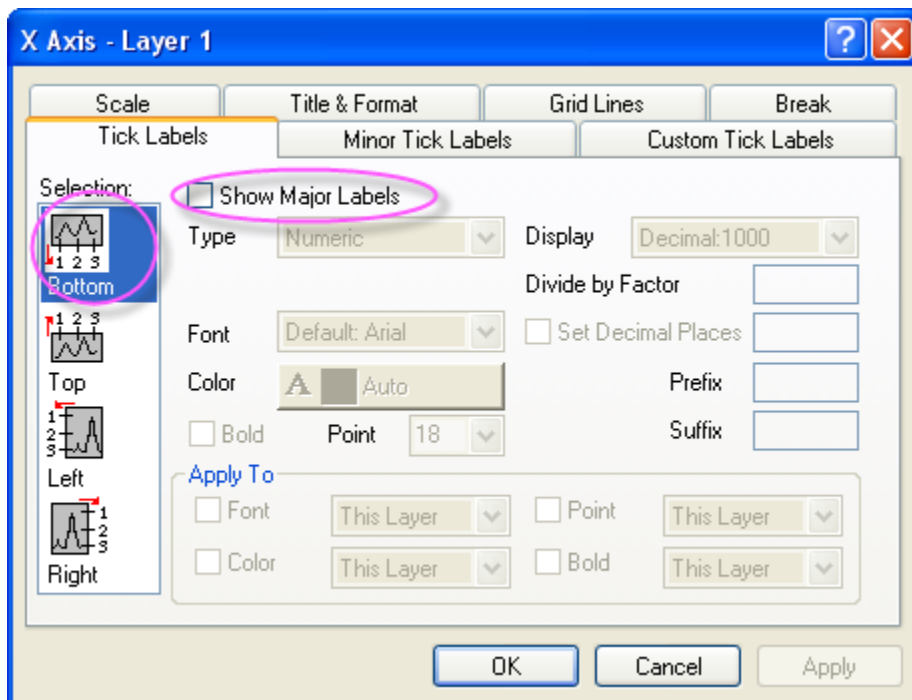
- To merge two graphs, select **Graph: Merge Graph Windows** from the menu. Accept the default settings. Click **OK** to merge two graphs. The merged graph should look like



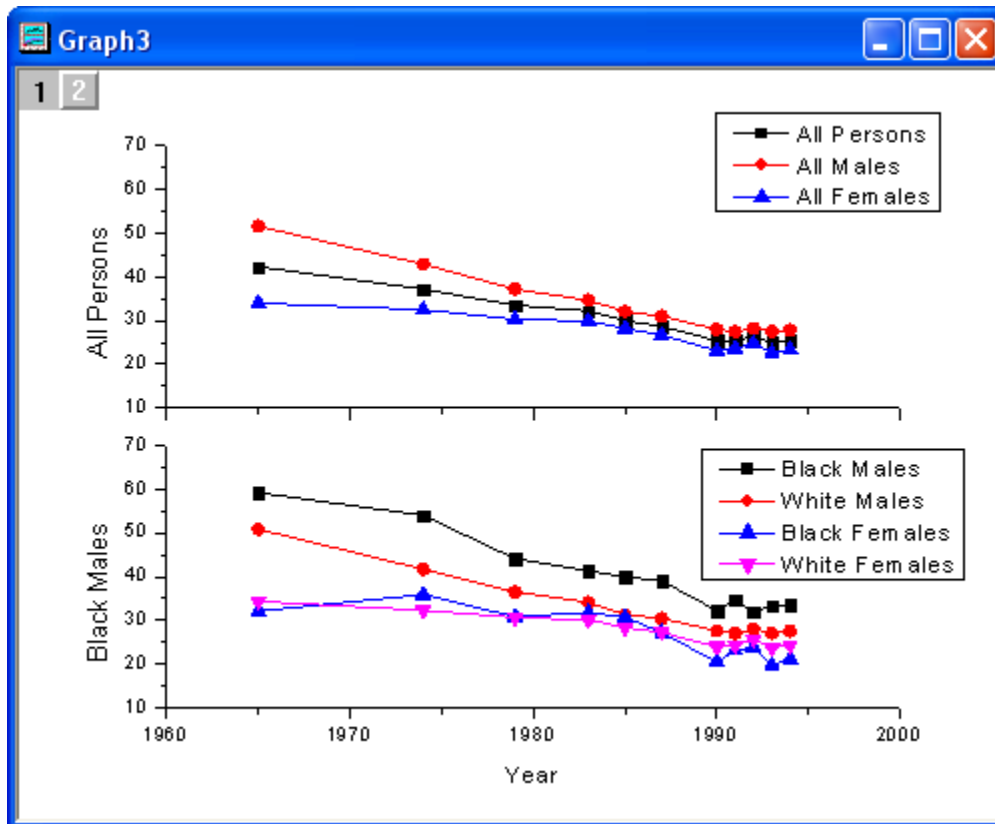
- Double-click on the **Y** axis of Layer 1 to open the **Axis Properties** dialog. In the **Scale** tab, set the dialog options as the screenshot below.



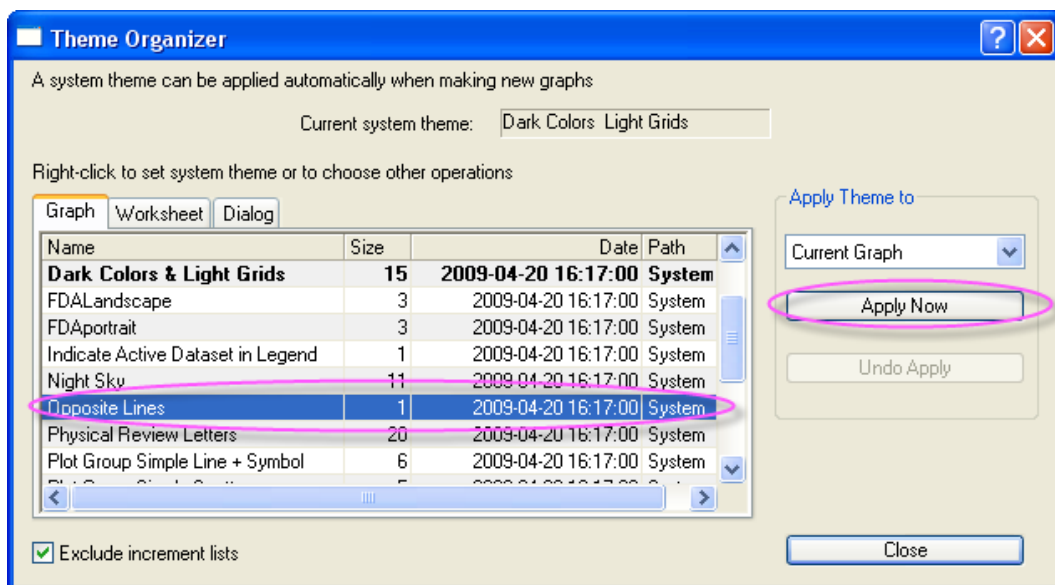
- In **Tick Labels** tab, set the dialog options as the screenshot below. Click OK to finish.
5. Delete the **X** title in Layer 1.



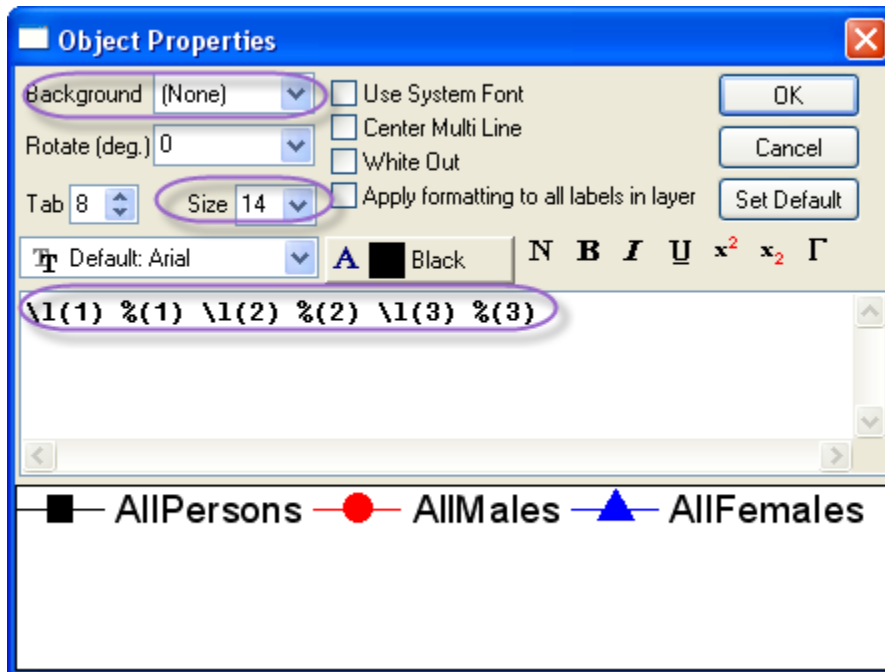
Set the same scale for the Y axis of Layer 2. After all of these operations, the graph should look like



- Now, apply a theme to add a top X axis and a right Y axis. Select **Tool: Theme Organizer** from the main menu to open the **Theme Organizer** dialog. Activate the **Graph** tab and select the **Opposite Lines** from the table. Then click the **Apply Now** button. Click the **Close** button to close the dialog.



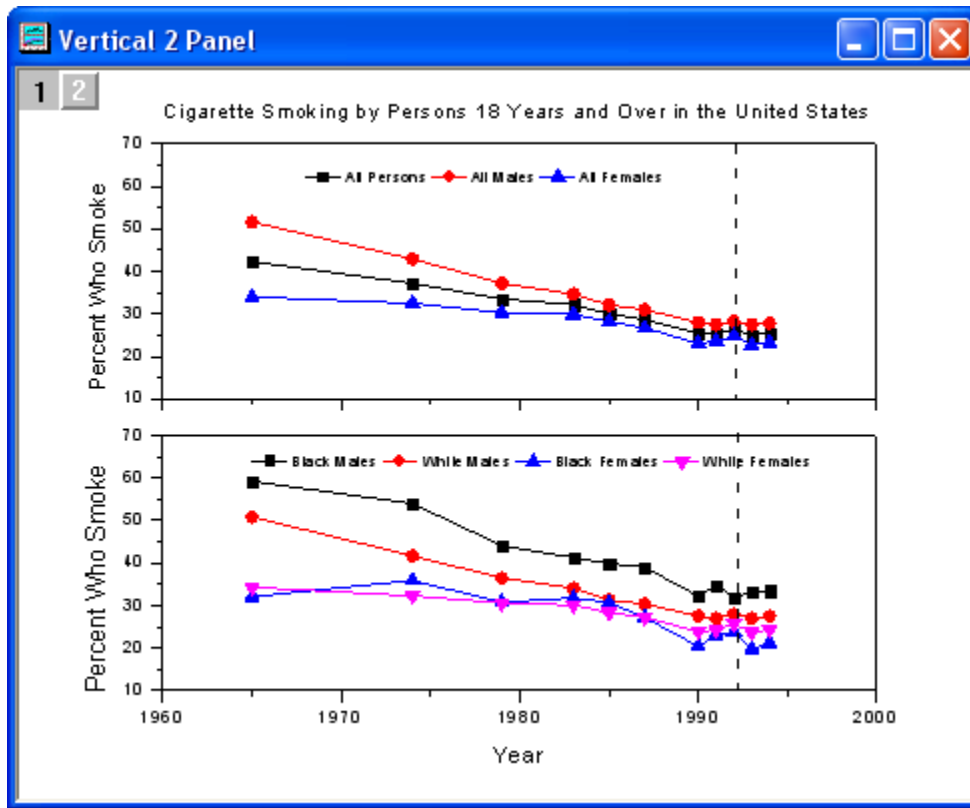
- Select the **Legend** of Layer1 and right-click on it to select **Properties** from the short-cut menu. Then set the dialog options as following image shows:



Move the legend to the position as the sample image shows.


8. Click the **Line** button on the **Tools** toolbar and draw a line on the Layer 1 as the sample image shows. Hold down the **SHIFT** key while drawing to ensure that the line is vertical. Double-click on the line to open the properties dialog of the line object. In the **Line** tab, select **Dash** with the **Type** drop-down list. And enter **2** in the **Size** edit box. Activate the **Control** tab. Select both the **Horizontal Movement** and the **Vertical Movement** check boxes. Then click the **OK** button.
9. Repeat steps 6-7 for Layer 2.
10. Click the **Text** button in the **Tools** toolbar. Then click in the middle of the graph and enter **Cigarette Smoking by Persons 18 Years and Over in the United States** to add a title for the graph. Double-click on the **Y** axis title of Layer 1 and Layer 2, then enter **Percent Who Smoke** in them.

The graph should look like



Sample Data

Download the **Vertical_2_Panel_Line.txt** file from http://www.originlab.com/ftp/graph_gallery/data/Vertical_2_Panel_Line.txt. Click the **Import Single**

ASCII button  and select the file to import it into Origin.

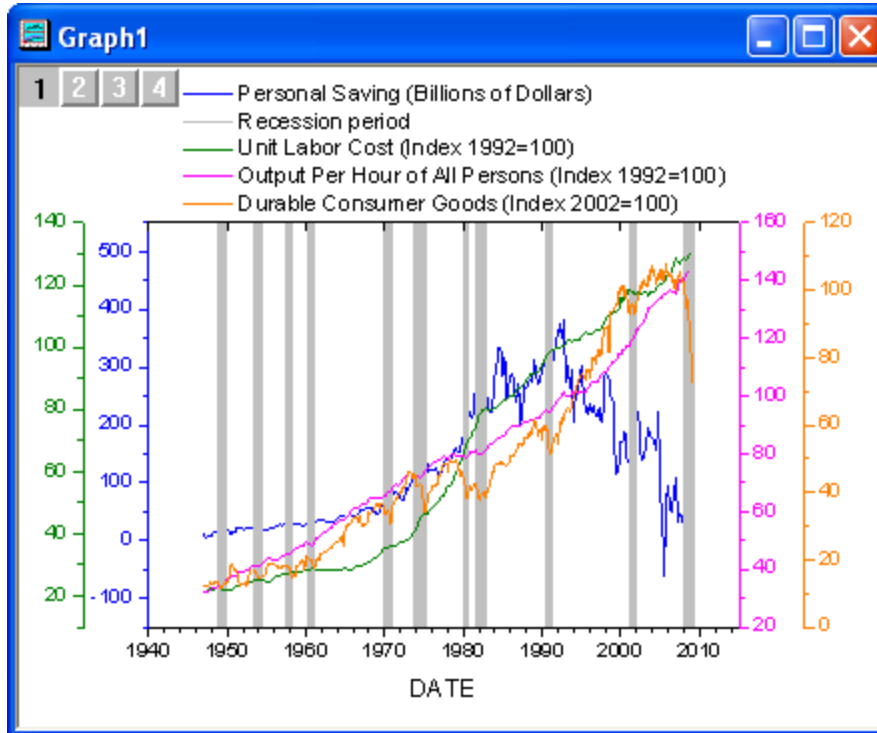
The following table contains part of the sample data.

Year(X)	All Persons(Y)	All Males(Y)	All Females(Y)	Black Males(Y)	White Males(Y)	Black Females(Y)	White Females(Y)
1965	42.3	51.6	34	59.2	50.8	32.1	34.3
1974	37.2	42.9	32.5	54	41.7	35.9	32.3
1979	33.5	37.2	30.3	44.1	36.5	30.8	30.6
1983	32.2	34.7	29.9	41.3	34.1	31.8	30.1
1985	30	32.1	28.2	39.9	31.3	30.7	28.3
1987	28.7	31	26.7	39	30.4	27.2	27.2
1990	25.4	28	23.1	32.2	27.6	20.4	23.9
1991	25.4	27.5	23.6	34.7	27	23.1	24.2
1992	26.4	28.2	24.8	32	28	23.9	25.7
1993	25	27.5	22.7	33.2	27	19.8	23.7
1994	25.5	27.8	23.3	33.5	27.5	21.1	24.3

3.4.6 Multiple Axis Plot

Summary

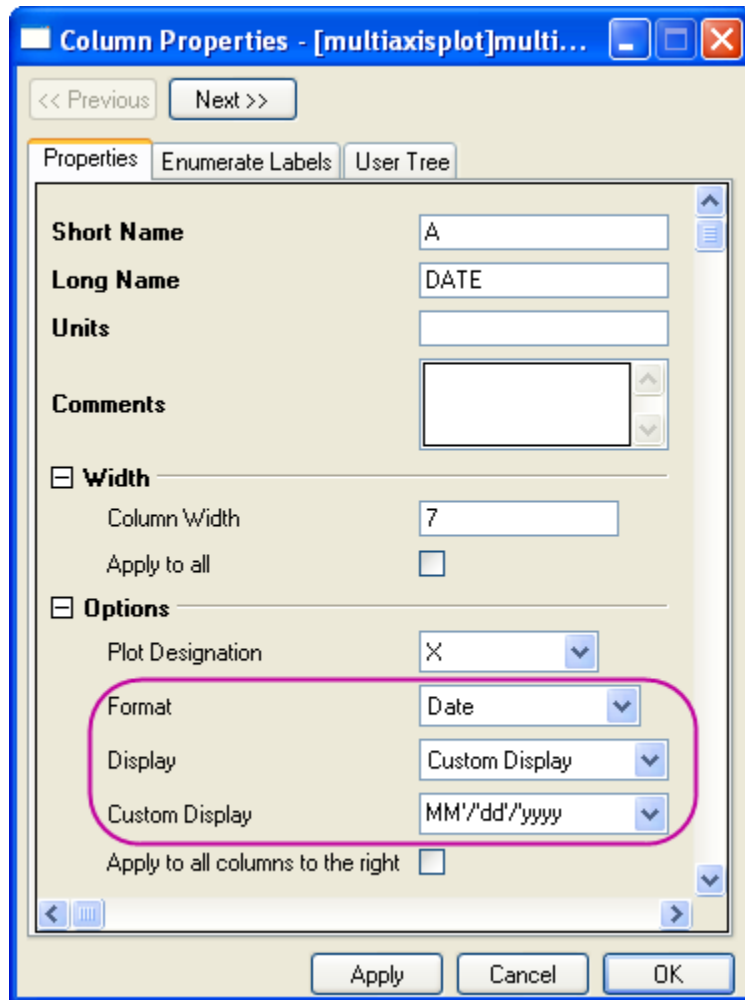
This tutorial will show you how to create a 4-Y plot and add recession bars.



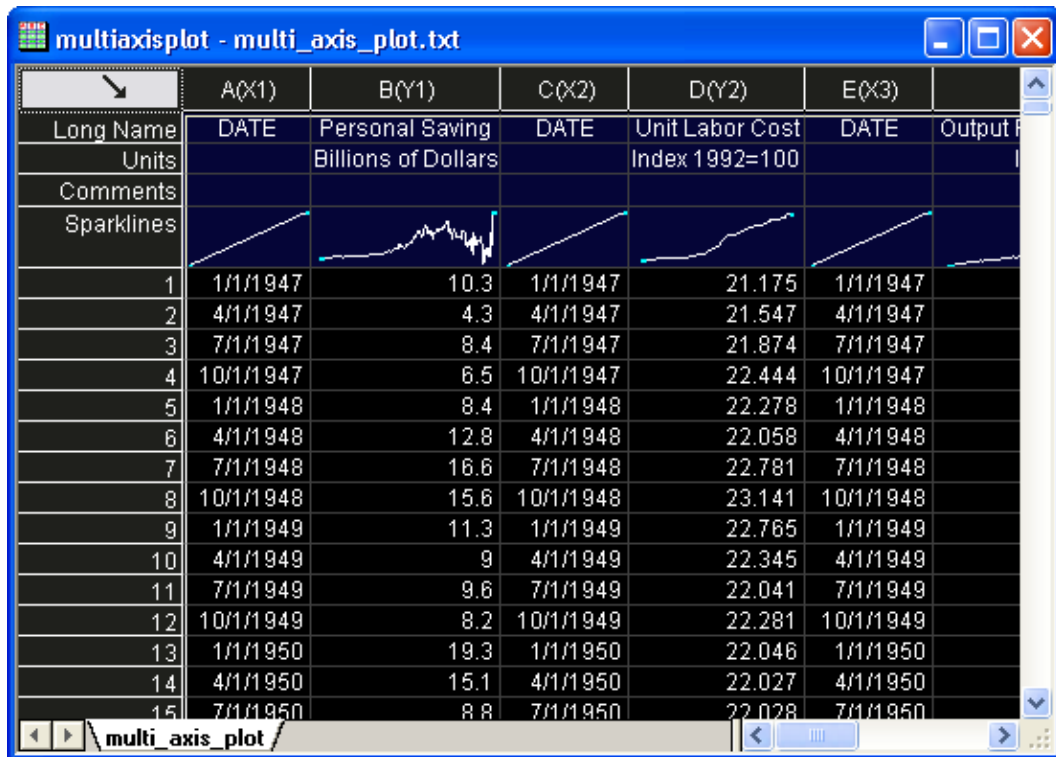
Minimum Origin Version Required: Origin 8.5.1

Steps

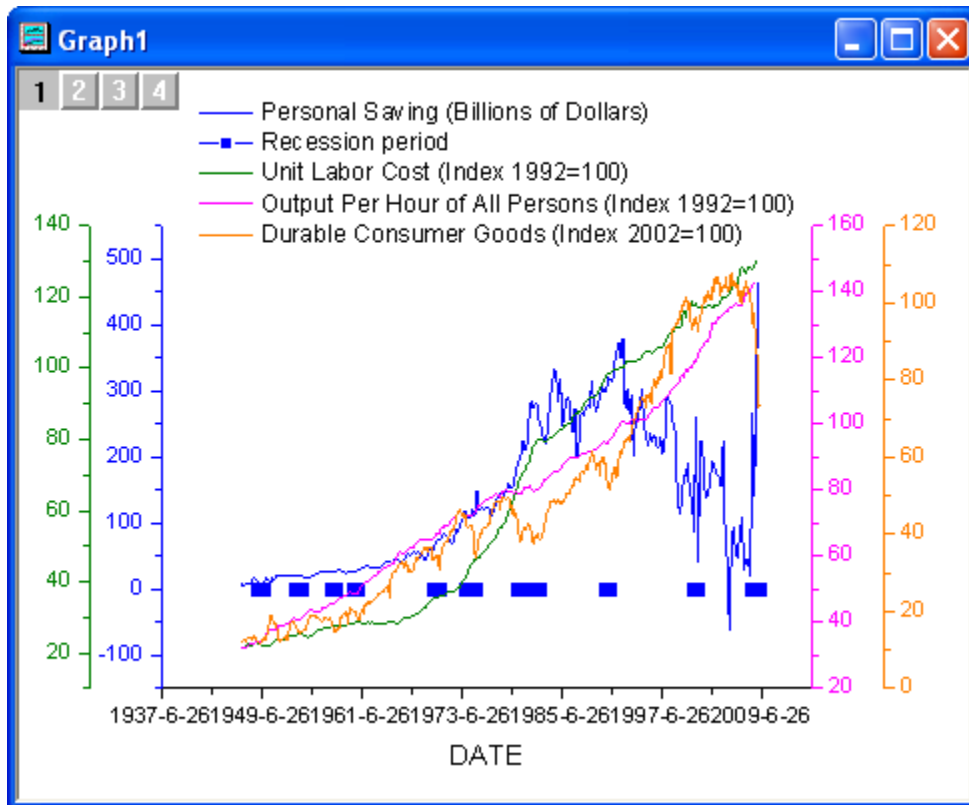
1. Create a new workbook and import the sample data into it.
2. Set the Format of col(A), col(C), col(E), col(G) and col(I) as **Date**, display as **MM'/'dd'/'yyyy** in the **Column Properties** dialog.



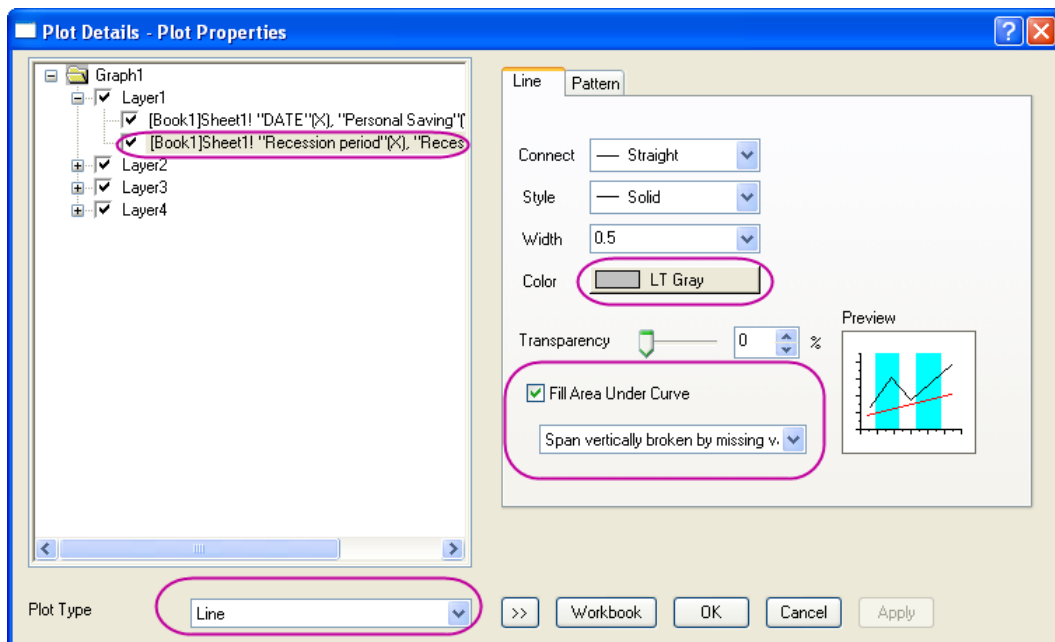
3. Place the mouse cursor near the top left corner of the worksheet. When the cursor changes to a down-right arrow, click the left button to select the whole worksheet.



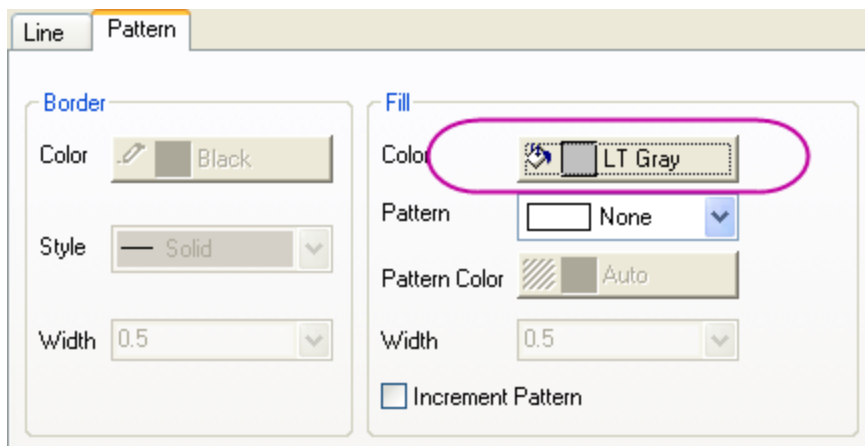
4. Right-click and select **Set as: XY XY** from the short-cut menu to set proper plotting designations for the worksheet columns.
5. Highlight all columns. From the main menu, select **Plot: Multi-curve: 4Ys YY-YY** to create a plot.



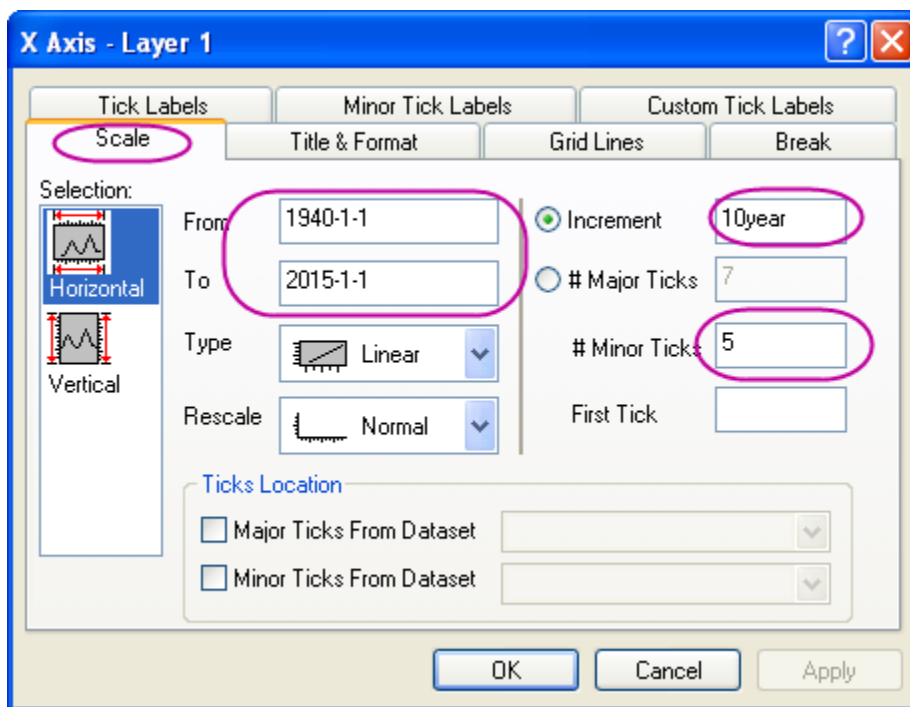
- Double-click on the graph to open the **Plot Details** dialog. In the left panel, select the second plot in Layer 1 and set the plot type as line and click Apply button. Then go to the **Line** tab in the right panel as the following screenshot and then click **Apply**.



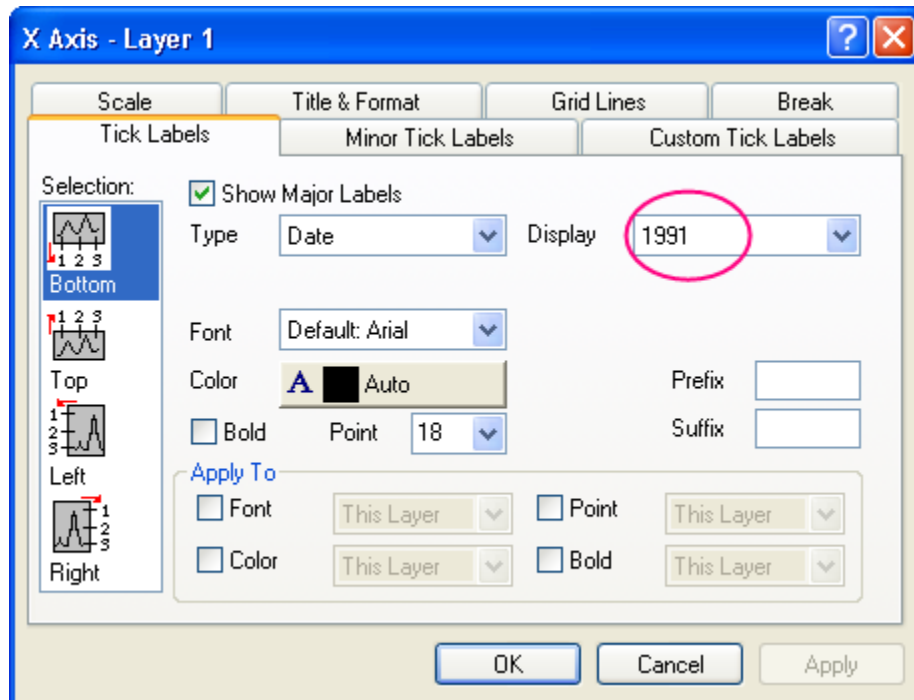
- Go to the **Pattern** tab. Set the dialog options as follows and then click **OK**.



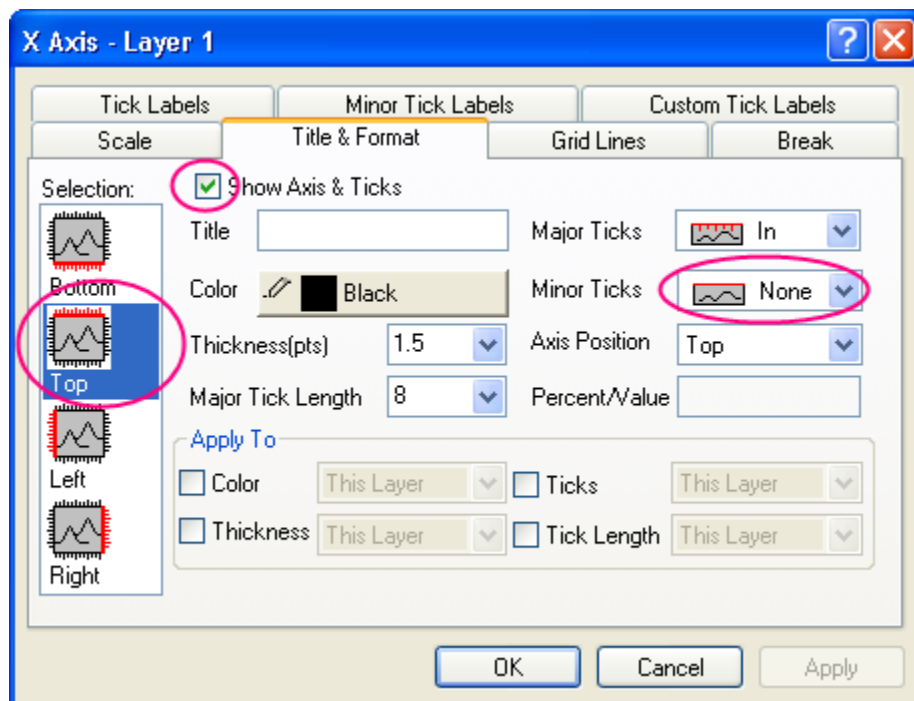
8. Double-click on the bottom axis to open the **Axis Properties** dialog. In the **Scale** tab, change the dialog options as follows and then click **Apply**.



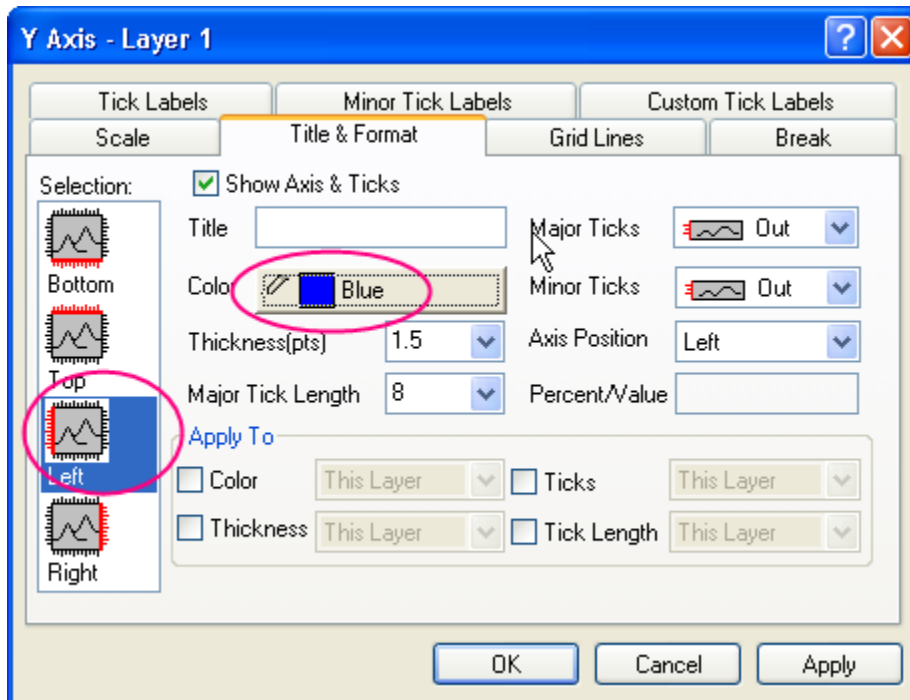
9. Activate the **Tick Labels** tab. Change the dialog options as follows and then click **Apply**.



10. Activate the **Title & Format** tab. Select **Top** in the **Selection** list. Then change the dialog options as follows and then click **Apply**.

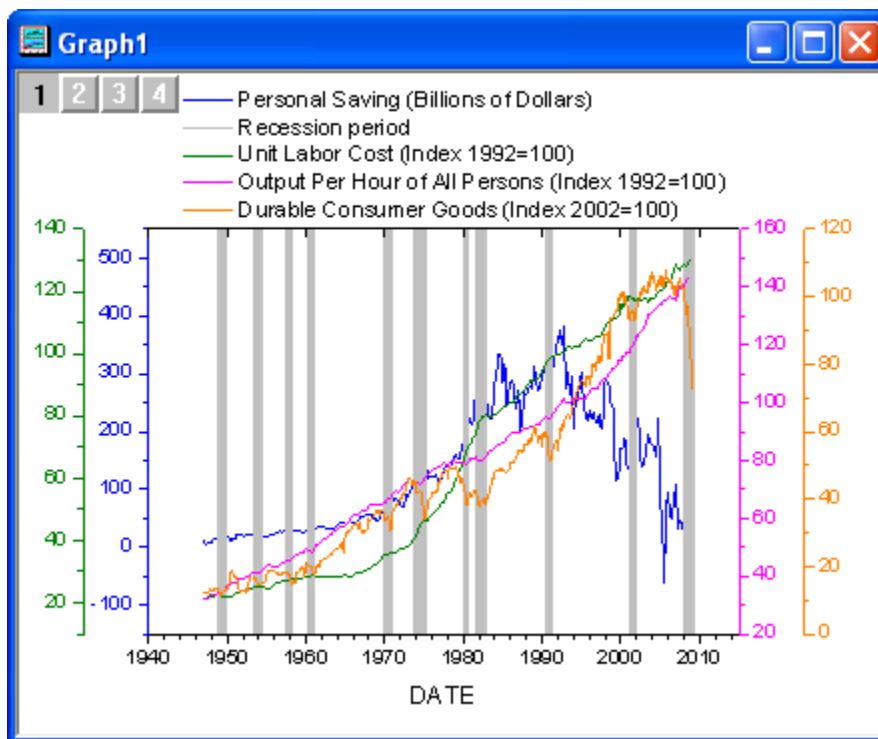


11. Still in the **Title & Format** tab, select **Left** in the **Selection** list. Then change the dialog options as follows and then click **OK**.




12. From the Origin menu, select **Graph: New Legend** to update the legend. Click on the legend to select it and then use the mouse to drag it to a better position.

The final graph should be like:



Sample Data

Download the sample data file from http://www.originlab.com/ftp/graph_gallery/data/multi_axis_plot.txt. Create a new worksheet. Click the **Import Single ASCII** button  and select the file to import it into Origin.

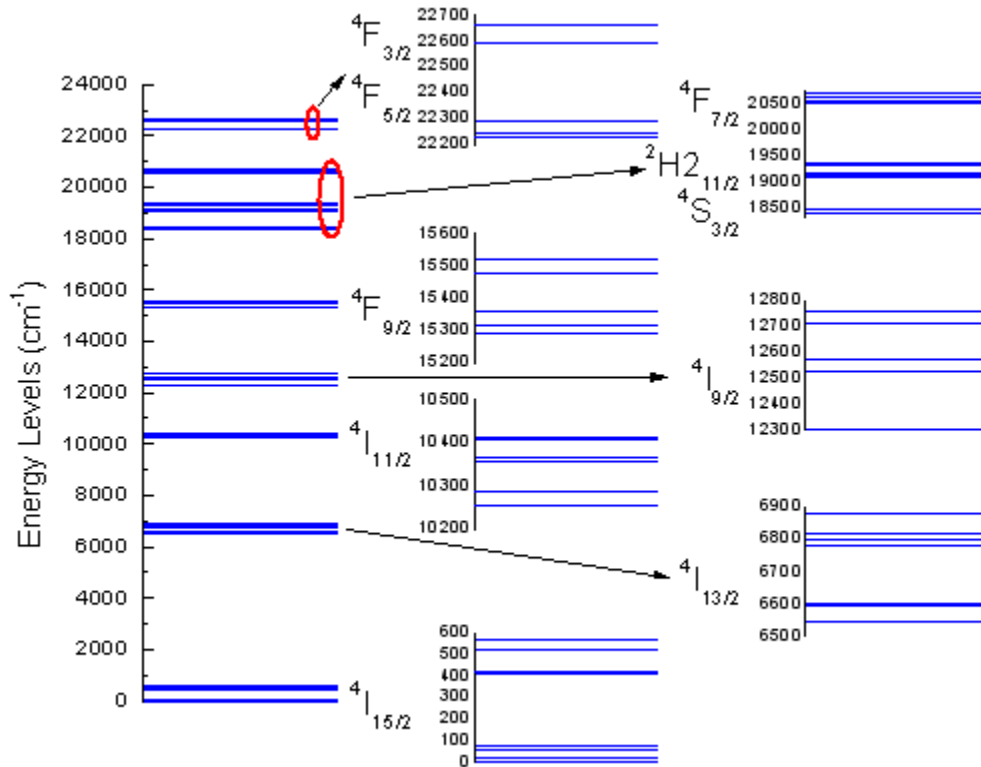
The following table contains part of the sample data.

DATE	Personal Saving	DATE	Unit Labor Cost	DATE	Output Per Hour of All Persons	DATE	Durable Consumer Goods Recession period	Recession Period	Recession Period
1/1/1947	10.3	1/1/1947	21.175	1/1/1947	32.11	1/1/1947	11.8583	11/1/1948	1
4/1/1947	4.3	4/1/1947	21.547	4/1/1947	32.304	2/1/1947	12.222	10/1/1949	1
7/1/1947	8.4	7/1/1947	21.874	7/1/1947	32.069	3/1/1947	12.4888	--	1

3.4.7 Energy-Level Structure of the Er³⁺ Ion in A YAG Crystal

Summary

In this tutorial, a graph will be created to show the energy-level structure of the Er³⁺ ion in a YAG crystal.




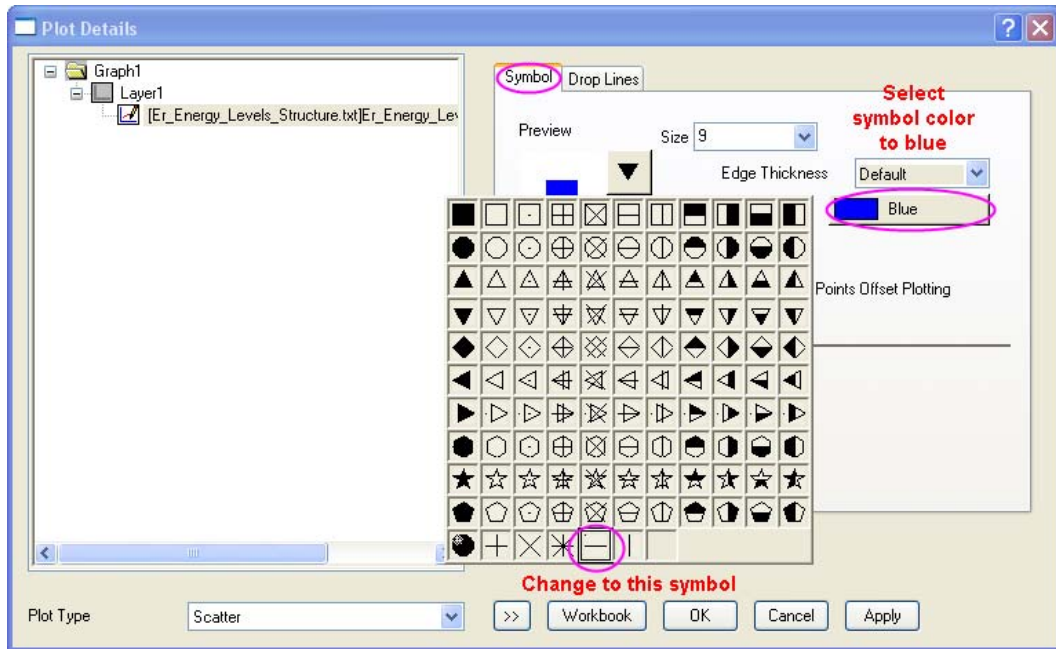
Minimum Origin Version Required: Origin 8 SR0

What you will learn

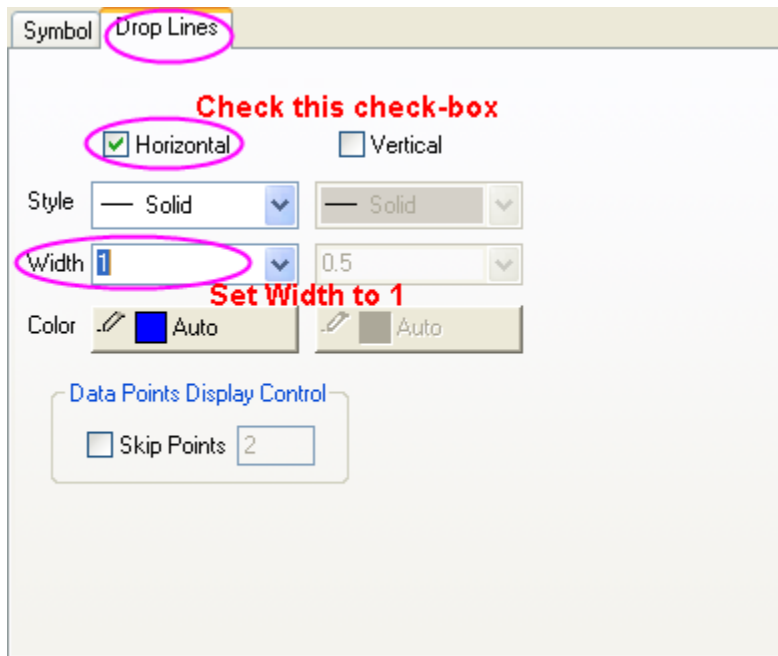
- Create a graph by selecting a part of data from a column
- Set symbol properties
- Merge graphs
- Change the size and position of a layer

Steps

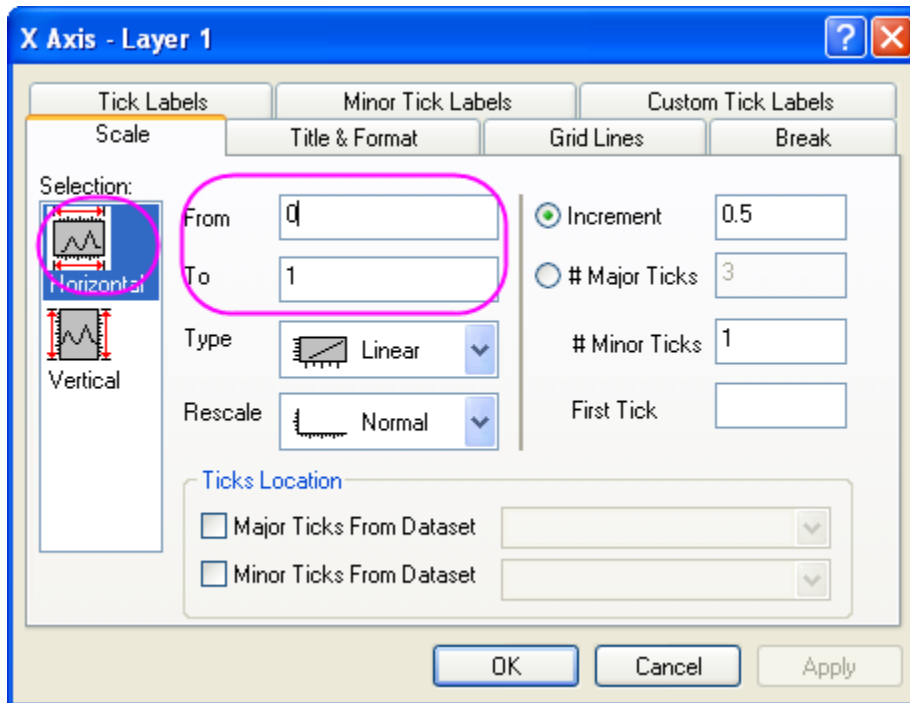
1. To start this tutorial, please download the data file from ftp.
2. Create an empty workbook, click the **Single Import ASCII**  to import the data file downloaded in step 1.
3. Highlight column B and create a symbol plot by selecting menu **Plot: Symbol: Scatter**.
4. Double click on the scatter to open the **Plot Details** dialog. In the **Symbol** tab of the right panel, change the symbol and **Symbol Color** as following image shows.



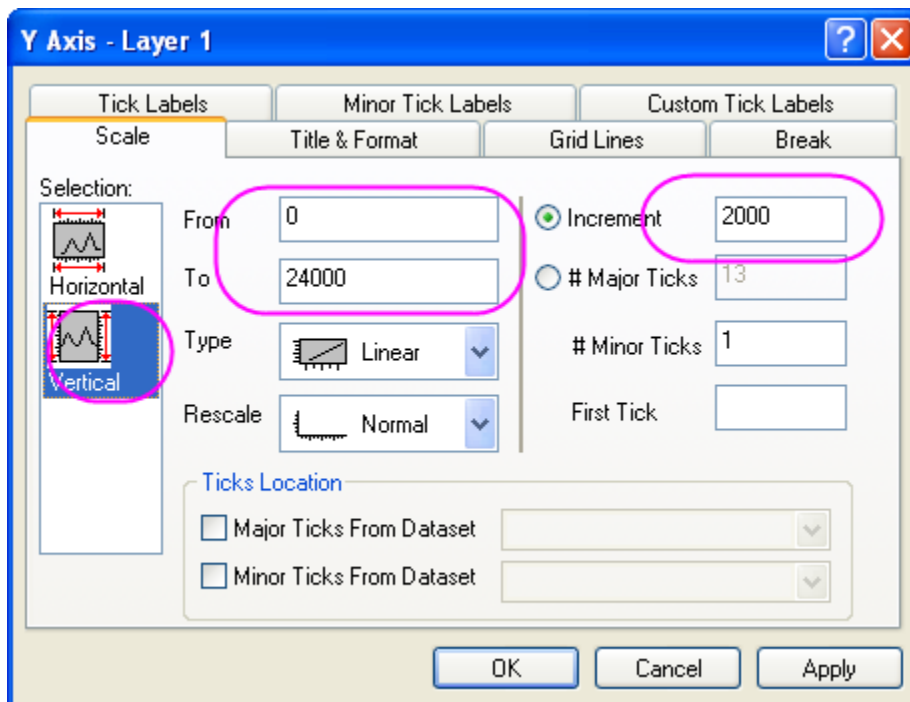
5. In the **Drop Lines** tab, check the **Horizontal** check-box and set **Width** to 1. Then click the **OK** button.



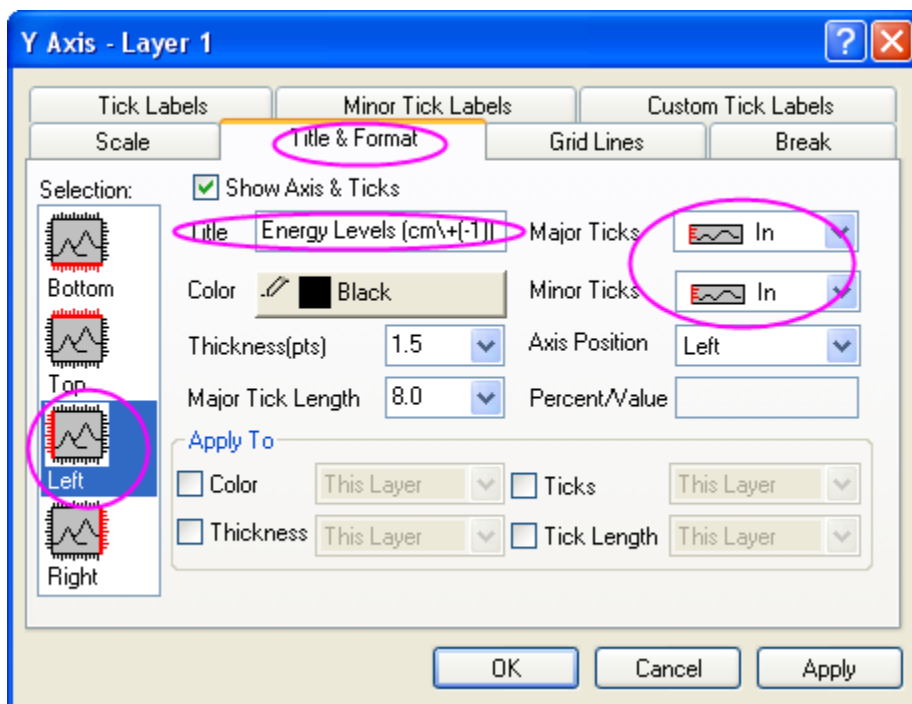
6. Double click on the X axis to open the axis setting dialog. Set X axis **From 0 To 1**.



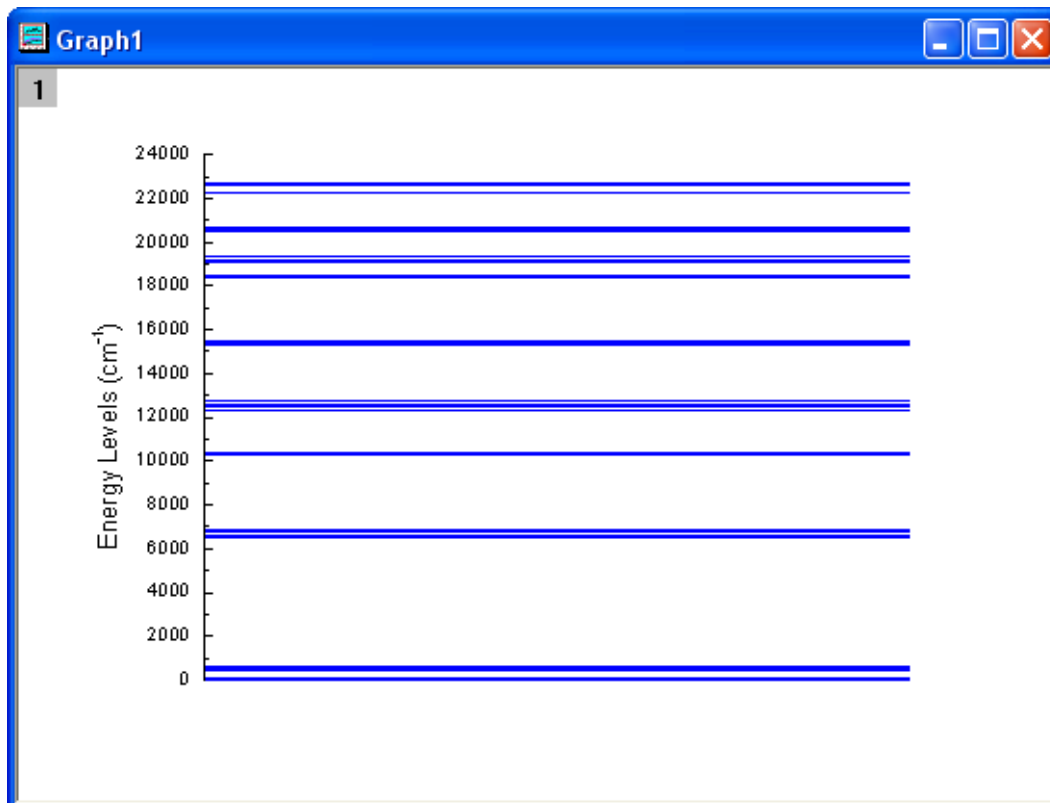
7. Select **Vertical** from the left **Selection** panel. Set Y axis **From 0 To 24000**, and **Increment** is 2000.



8. Switch to the **Title&Format** tab, make sure **Left** is selected in the left **Selection** panel. Change the **Title** to *Energy Levels (cm)⁺⁽⁻¹⁾*, **Major Ticks** and **Minor Ticks** both are *In*. Then click the **OK** button.



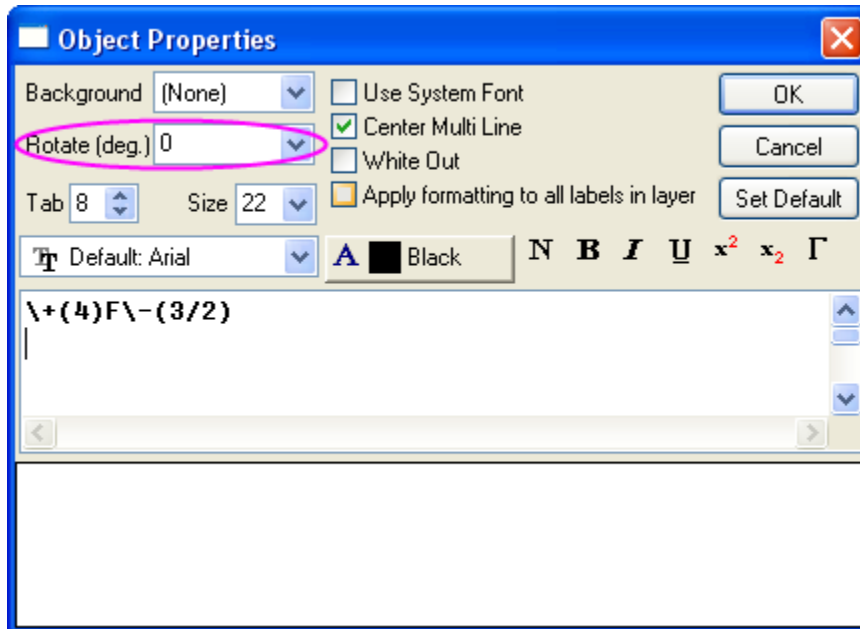
9. Delete the legend, X axis, title of X axis and tick labels of X axis.



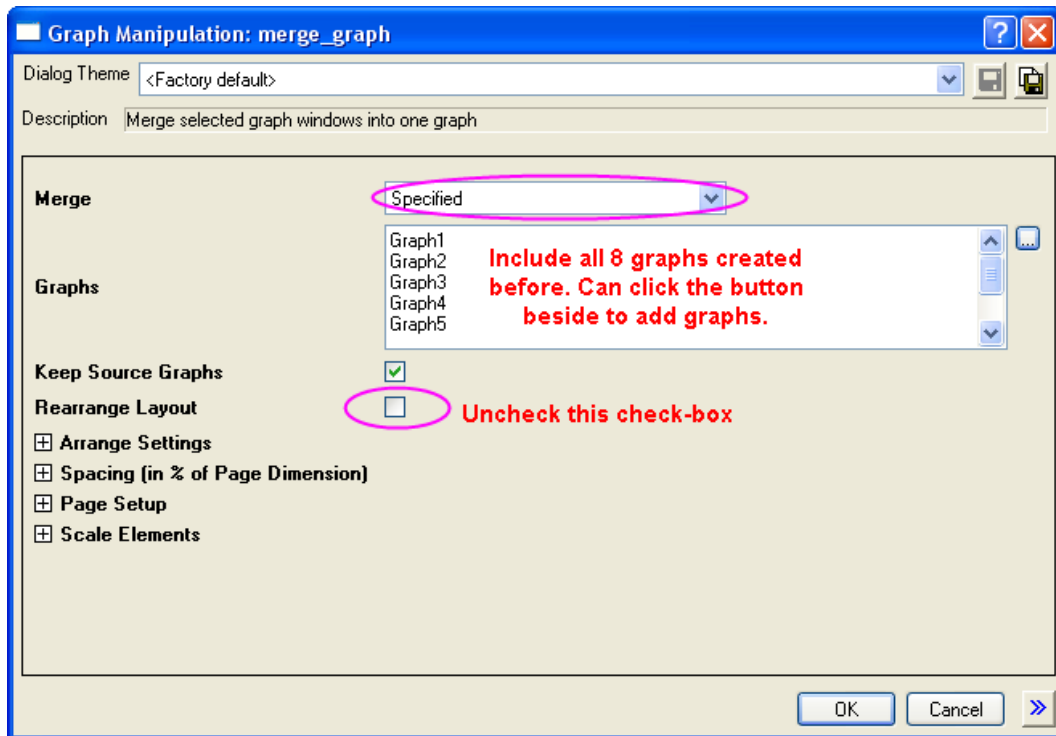
10. Highlight cell 1 to cell 8, cell 9 to cell 15, cell 16 to cell 21, cell 22 to cell 26, cell 27 to cell 31, cell 32 to cell 43, and cell 44 to 48 respectively, and then repeat step 3 through step 9 to make another 7 graphs. The **Increment** of Y axis (step 7) all are set to 100, and **From** and **To** are 0 to

600, 6500 to 6900, 10200 to 10500, 12300 to 12800, 15200 to 15600, 18300 to 20800, 22200 to 22700, respectively. Also, the **Title** of Y axis (step 8) are different, as $\sqrt[4]{I}^{-15/2}$, $\sqrt[4]{I}^{-13/2}$, $\sqrt[4]{I}^{-11/2}$, $\sqrt[4]{I}^{-9/2}$, $\sqrt[4]{F}^{-9/2}$, $\sqrt[4]{F}^{-7/2}$, $\sqrt[2]{H2}^{-11/2}$, $\sqrt[4]{S}^{-3/2}$, and $\sqrt[4]{F}^{-3/2}$, $\sqrt[4]{F}^{-5/2}$, respectively.

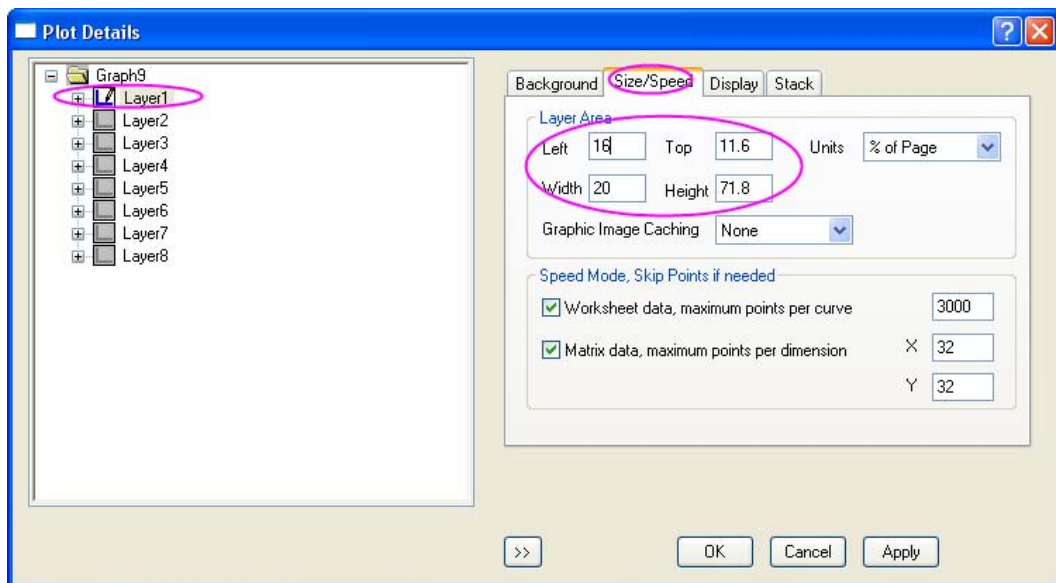
- Change the Y title's rotate angle of graphs created in step 10. Just select the Y title and choose **Properties** from the right-click menu and set **Rotate(deg.)** to 0 in the open **Object Properties** dialog.



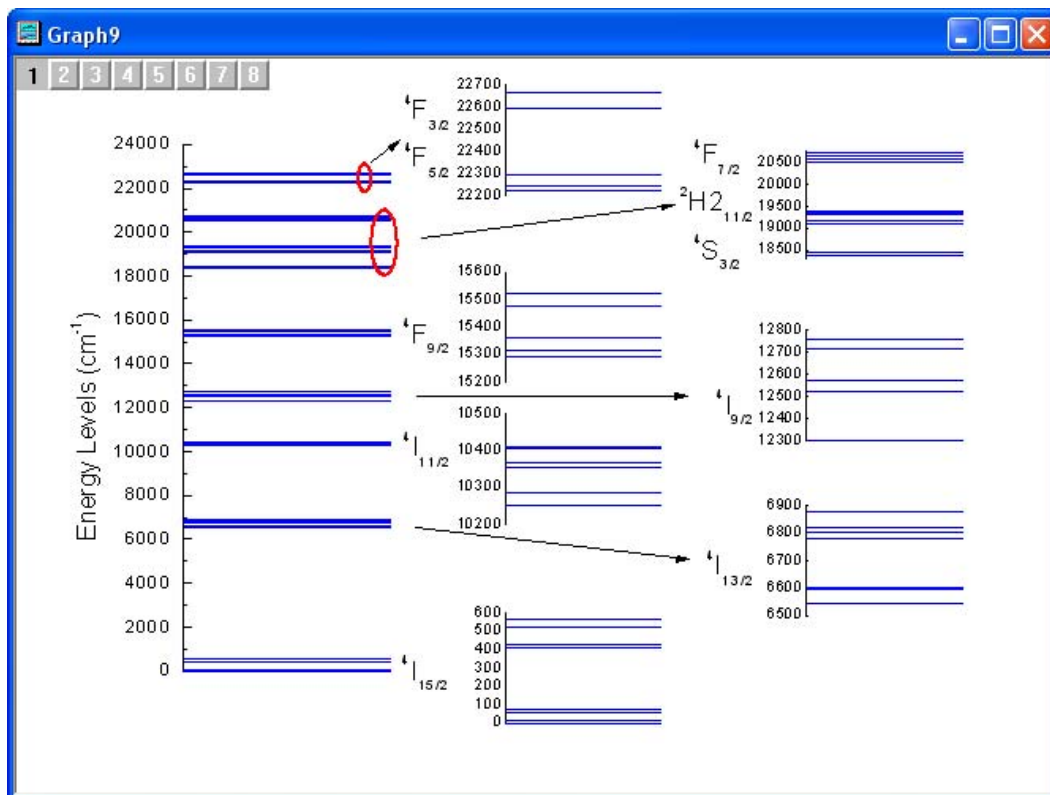
- Activate one of the graphs, select menu **Graph: Merge Graph Window**. Change the settings as following and click the **OK** button to merge 8 graphs created before.



13. Double click on the merged graph to open the **Plot Details** dialog. In the **Size/Speed** tab of layer 1, change the **Layer Area** as following.



14. Change other layers' size as 15 by 15 (Width and Height in step 13), then move them to the proper position, adjust the size of Y axis label and add the corresponding arrows and circle, finally get the graph as the following image shows.



3.4.8 Create an 8 layer multi-panel plot template

Summary

All child windows in Origin, with the exception of the Notes window, are created from template files. These template files describe how to construct the window. For a graph window, the template file determines all page and layer characteristics, such as page size, number of layers, inclusion of text labels, data plot style information, etc.

The template library lists all built-in as well as user-created templates.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

- How to create an 8 layer multi-panel plot
- How to save the formatting as a template
- How to reuse the template with similar data

Steps

Choosing the Data Source

1. Click the **New Project** button on the Standard toolbar, to begin with a new project.
2. Click the **Import Wizard** button on the Standard toolbar. The Import Wizard opens. (Note that if this is the first time that you have started the Import Wizard, you will experience a slight delay as Origin compiles the necessary files.)
3. Verify that the **ASCII** radio button is selected in the **Data Type** group.
4. Click the browse button to the right of the File text box. Navigate to the Origin folder; browse to the Samples folder and then the Curve Fitting folder. Select Step01.dat from the list of files.
5. Click the **Add File(s)** button.
6. Click **OK**.
7. Leave the **Import Filters for current Data Type** as **Data Folder: step**. (This filter has the settings to use when importing the file.)
8. Click the **Finish** button. The data file imports into the worksheet.

Plotting the Data

1. Highlight the entire worksheet of data. (Note that you can select an entire worksheet by placing your cursor in the blank area in the upper left corner of the worksheet. When the cursor becomes a downward pointing arrow, click once to select the entire worksheet.)
2. Select **Plot:Multi-Curve:9 Panel**. A new 9 layer graph is created.
3. Select layer 9 by clicking inside it.
4. Press the **Delete** key on the keyboard. This will delete layer 9, leaving you with 8 layers.
5. Select **Graph:Layer Management**. The **Layer Management** dialog opens.
6. Select the **Arrange** tab.
7. Set **Column** to 2 and **Row** to 4.

8. Click the **Apply** button. The preview in the dialog redraws to show you a 2x4 arrangement.
9. Click **OK**.

Editing the Graph

The goal is to save this 8 panel graph as a template; i.e. a new plot type, so that it can be used again with new similar data. Since the template will also save plot style information, let's customize the graph a bit further.

1. Double-click on the X axis in layer 1. The **X-Axis** dialog opens.
2. Select the **Grid Lines** tab.
3. Check the **Major Grids** and **Minor Grids** checkboxes.
4. Set the **Line Color** for both Major and Minor grids to **LT Gray**.
5. Check the **Apply To Grid Lines** checkbox for **This Layer**.
6. Click **OK**.
7. Select layer 1 by clicking inside it.
8. Right-click inside the layer and select **Copy Format:All Style Formats**. This will copy the style formats of layer 1.
9. To apply formatting to all layers, right-click outside of any layer (make sure that no layer is selected...one easy way to do that is to right-click in the gray area of the window, outside the white printable part of page), and select **Paste Format**.

Saving as a new graph template

1. Select **File:Save Template As**. A dialog opens allowing you to choose the category that the template will be saved in as well as the name given to the new template.
2. Change the **Category** to **UserDefined**.
3. Change the **Template Name** from **PAN9** to **PAN8**. (Note that the **Template Name** that appears when the dialog opens is the name of the original template that was used to create the graph.)
4. Expand the **Option** node and note that the template will be saved to your **User Files Folder**.
5. In the **Description** edit box, enter the following: My new 8 panel graph
6. Click **OK**.

Plotting into your new template

1. Click the **New Folder** button on the Standard toolbar.
2. Click the **Import Wizard** button on the Standard toolbar. The Import Wizard opens.
3. Click the browse button to the right of the File text box. Navigate to the Origin folder; browse to the Samples folder and then the Curve Fitting folder. Select Step02.dat from the list of files.
4. Click the **Add File(s)** button.
5. Click **OK**.
6. Leave the **Import Filters for current Data Type** as **Data Folder: step**. (This filter has the settings to use when importing the file.)
7. Click the **Finish** button. The data file imports into the worksheet.

8. Highlight the entire worksheet of data. (Note that you can select an entire worksheet by placing your cursor in the blank area in the upper left corner of the worksheet. When the cursor becomes a downward pointing arrow, click once to select the entire worksheet.)
9. Select **Plot:Template Library** or click the **Template Library** button on the **2D Graphs** toolbar.
10. Scroll down to the **UserDefined** category under Graph Template.
11. Select **PAN8**. (Note that the Preview window is not a preview of the new data that you are plotting. It is an image of the graph when you saved your template.)
12. Click **Plot**.

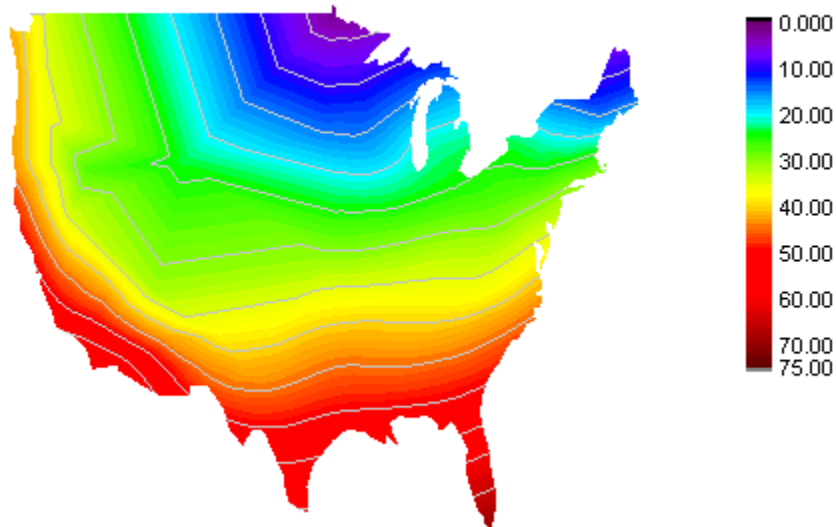
3.5 Contour

3.5.1 XYZ Contour

Summary

This tutorial will show you how to create a contour from XYZ data.

30-Year Mean Temperature for the Month of January



What will you learn

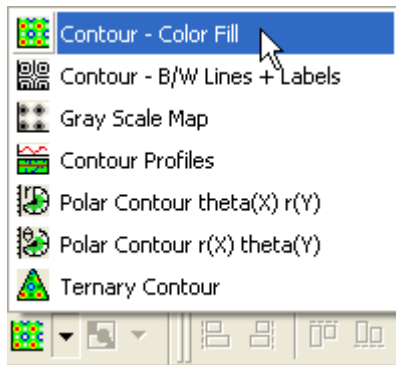
This tutorial will show you how to

- Create a Contour Plot from XYZ data
- Customize levels, lines, and color mapping
- Use a Custom Boundary
- Customize the Color Scale
- Customize the axes of the plot

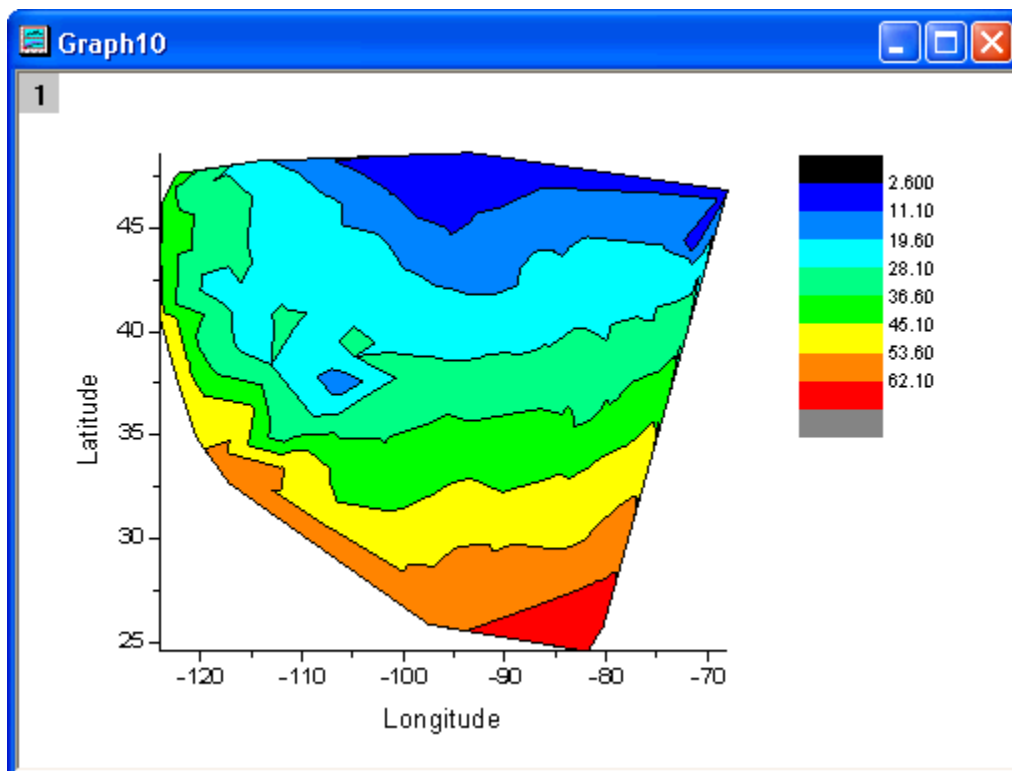
Steps

This tutorial is associated with the sample project: *\Samples\Samples.opj*.

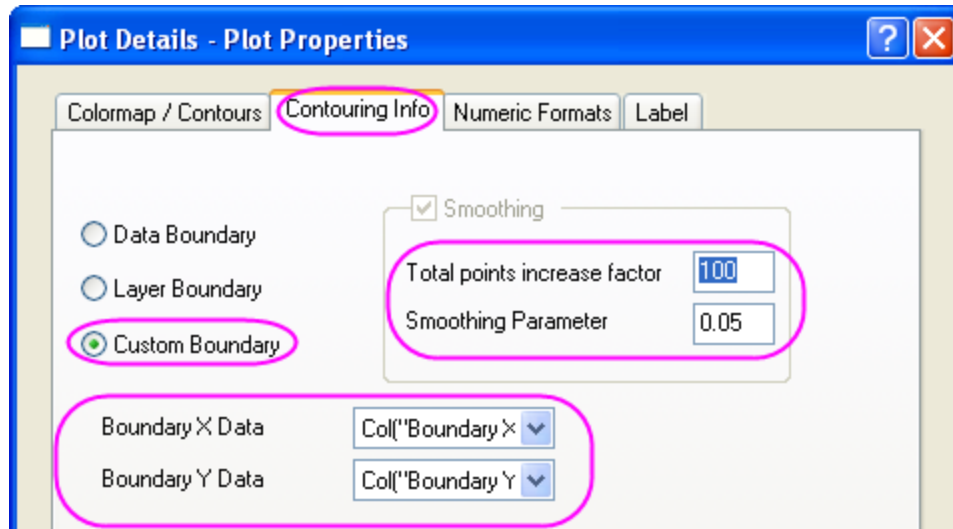
1. Open the *Graphing: Contour: XYZ Contour* folder in the **Project Explorer**. Activate **Book1** and highlight column **D** and then click **Contour-Color Fill** button from the **3D and Contour Graphs toolbar**.



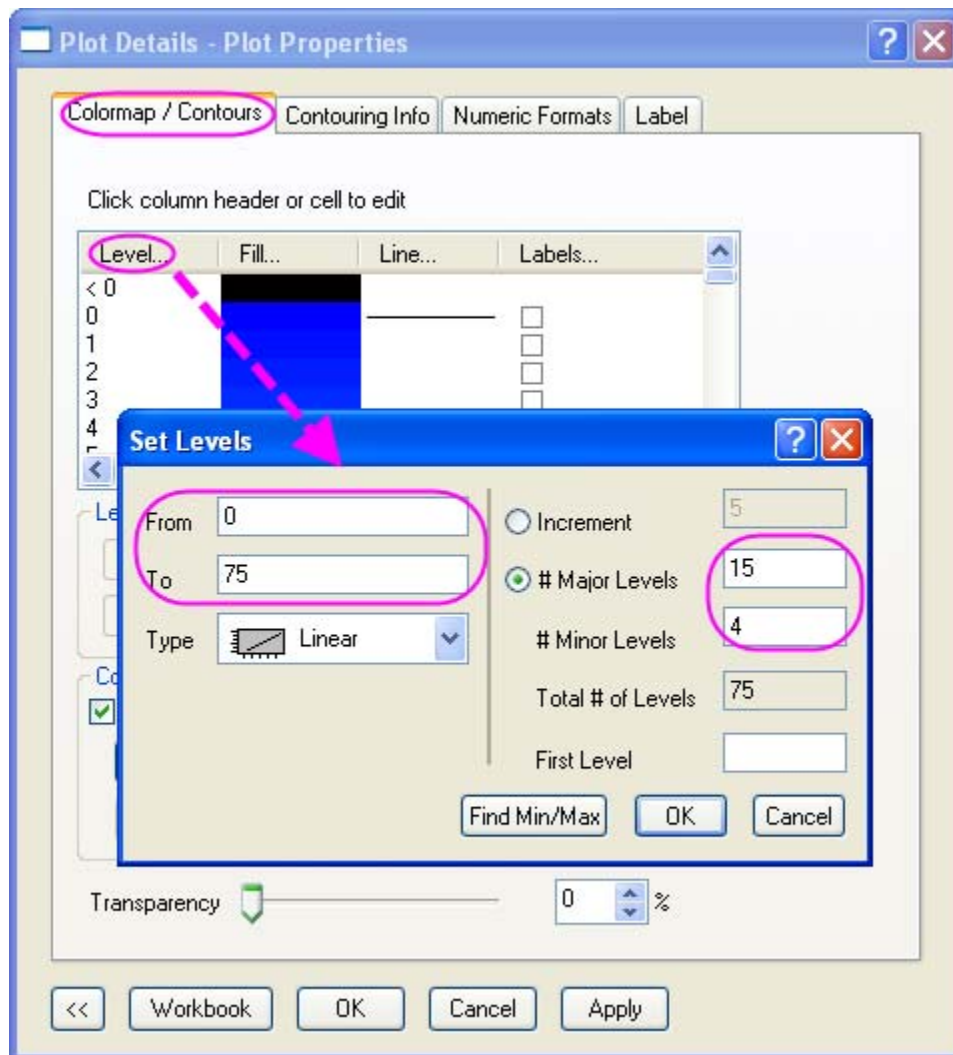
The graph should look like:



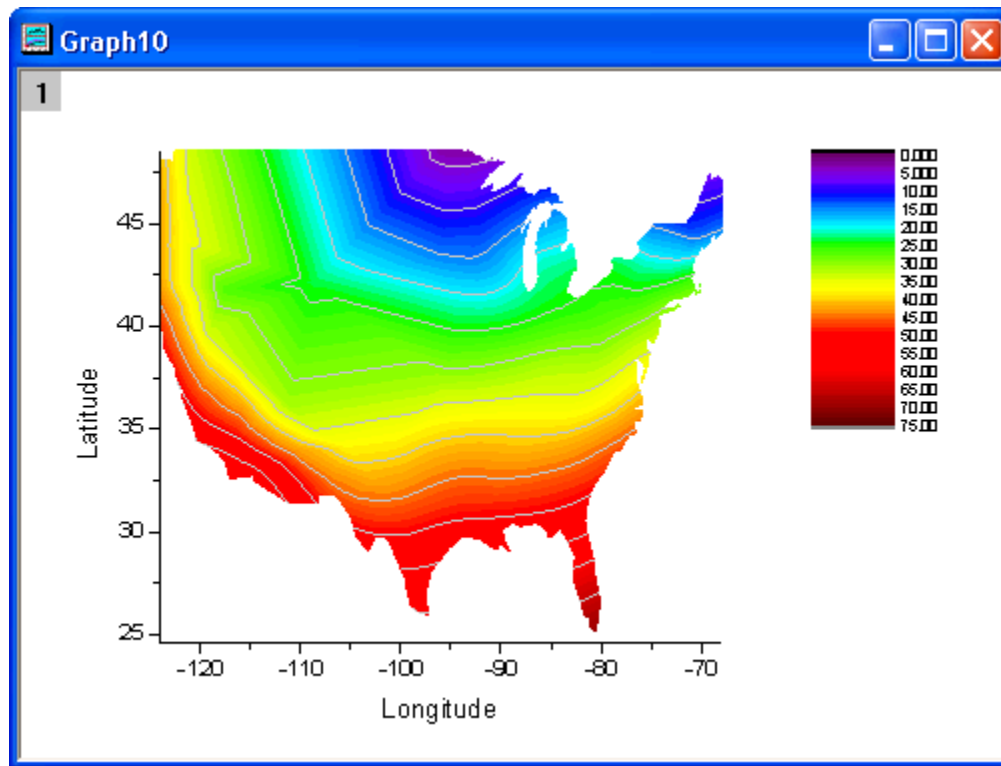
- Double-click on the contour plot to bring up **Plot Details** dialog. Select the **Contouring Info** tab and set the options in the dialog as the screenshot below:



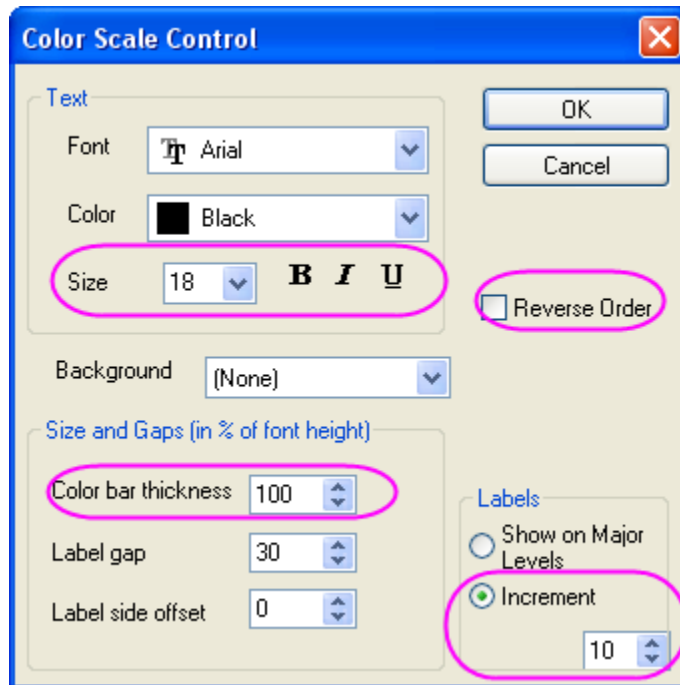
3. Select the **Color Map/Contours** tab, click the **Level** heading and set the dialog as the following screenshot shows:



- Click the **Fill** heading, select **Load Palette** and then select **Rainbow** from the Palette list.
 - Click the **Lines** heading, select **Show on Major Levels** only. Enable **Apply to All** checkbox and choose **LT Gray** for the **Color** dropdown list.
4. Click OK to close the Plot Details dialog, then the graph should look like

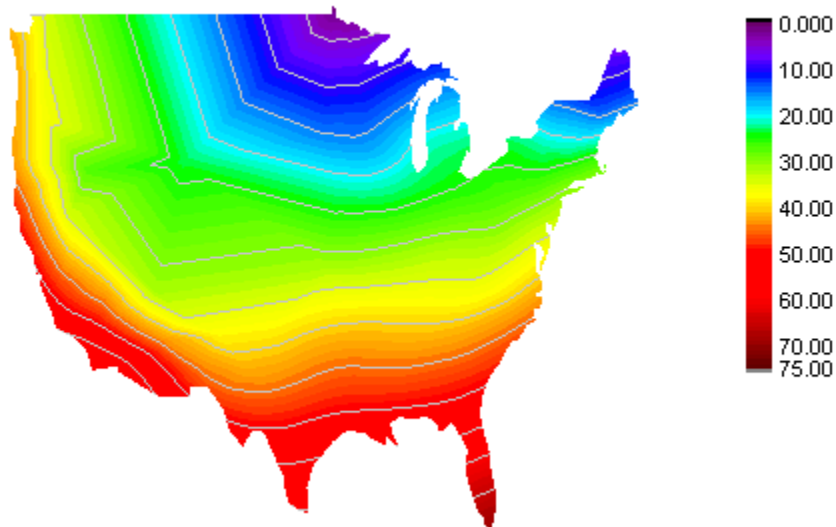


5. Select **Format: Axes: X Axes** to open the **Axis** dialog. Then do the following things.
- Select the **Scale** tab and then select **Horizontal** icon in the **Selection** list box, set **X From**, **To** and **Increment** equal to **-127**, **-65** and **10** individually.
 - Select the **Vertical** icon in the **Selection** list box, set **Y From**=**23**, **To**=**50**, and **Increment**=**5**.
 - Select the **Title and Format** tab and then select **Left** icon in the **Selection** list box, disable the **Show Axis & Ticks** check box.
 - Select the **Bottom** icon in the **Selection** list box, disable the **Show Axis & Ticks** check box.
 - Select the **Tick Labels** tab and then select **Bottom** icon in the **Selection** list box, disable the **Show Major Labels** check box.
 - Select the **Left** icon in the **Selection** list box, disable the **Show Major Labels** check box.
6. Click OK to close the dialog. Double-click on the color scale legend to bring up its properties dialog. And set the dialog as the following screenshot shows:



- Right-click above the contour plot and select **Add Text** to add a graph title. Use the **Format** toolbar and/or double-click on the completed text label to update the font, font size, etc. The graph should look like

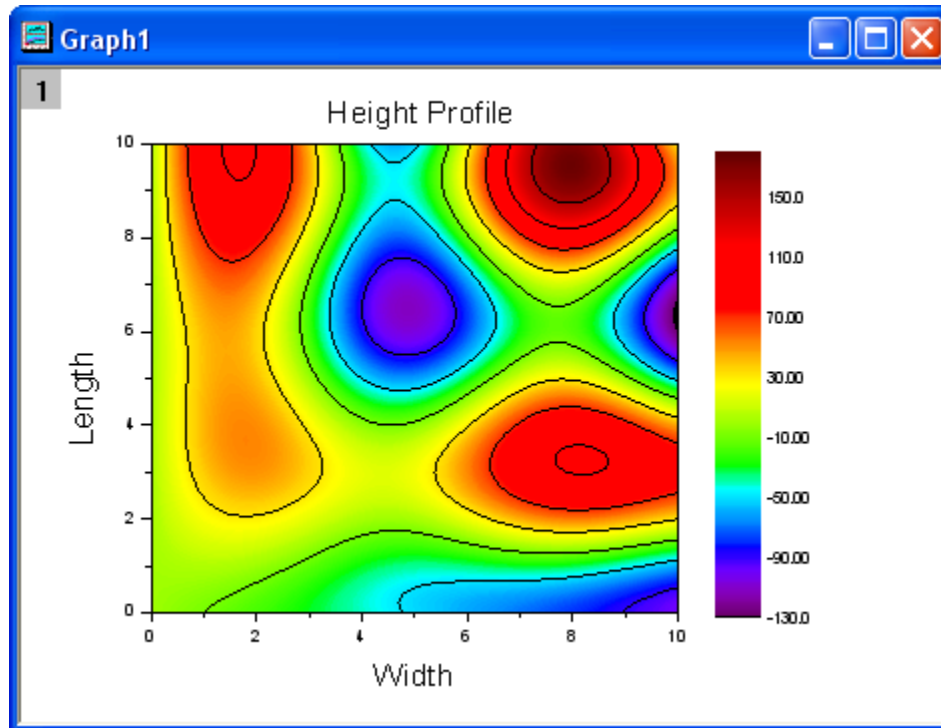
30-Year Mean Temperature for the Month of January



3.5.2 Contour Plot with Major and Minor Levels Filled by Using Color Palette

Summary

The levels in a contour plot can be divided into many levels from the minimum to the maximum, including major levels and minor levels. Each level can then be assigned a specific color, or a color palette could be used to assign colors to the levels. Contour line for each level can be customized as well. In this tutorial, a contour plot is created from a matrix, and then Rainbow palette will be used to assign colors.




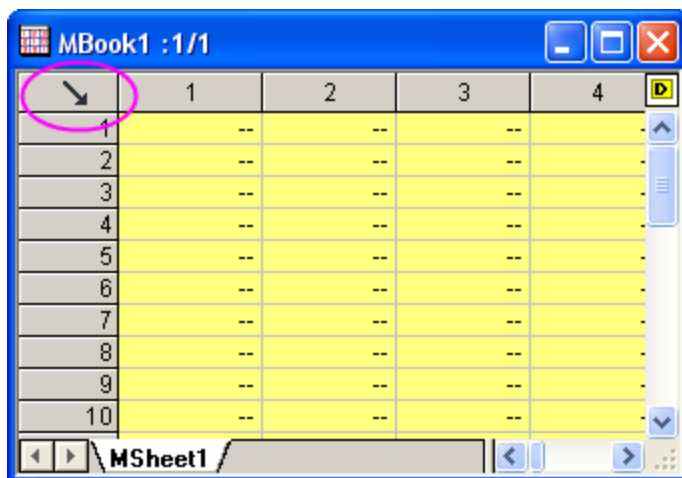
Minimum Origin Version Required: Origin 8.1 SR0

What you will learn

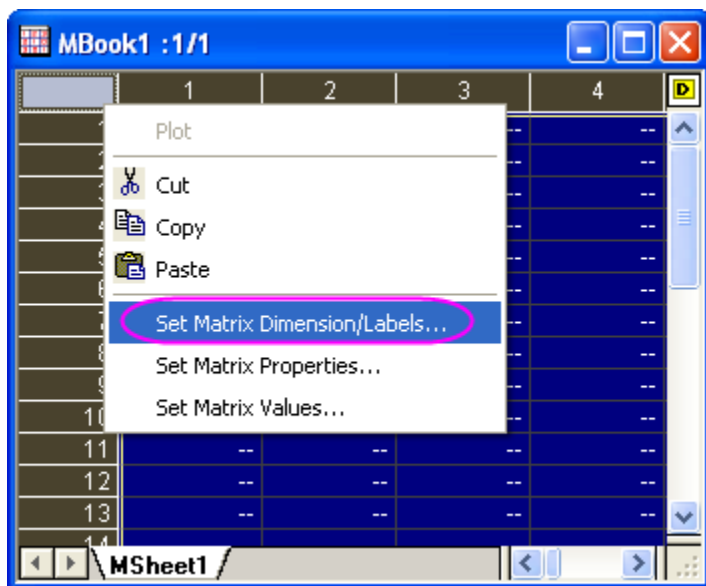
- Set dimensions for a matrix
- Set values for a matrix
- Turn on/off the **Speed Mode**
- Create a contour plot from a matrix
- Customize levels, contour lines and filled color for a contour plot
- Change the settings of the color scale

Steps

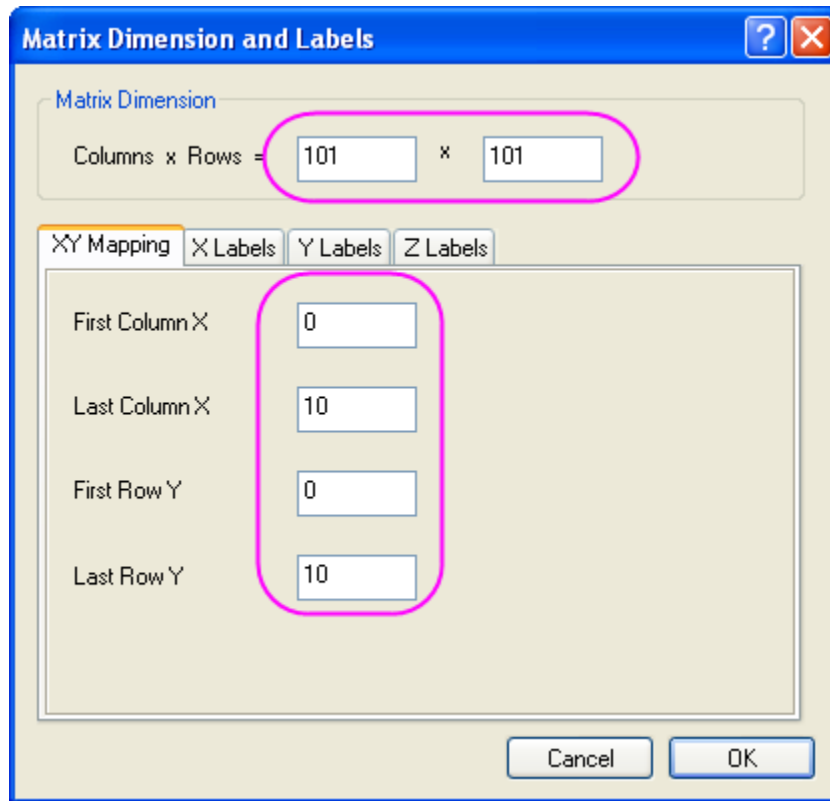
1. New a project, then click the **New Matrix**  button to create a new matrix. Then move the mouse to the upper left corner of the matrix until the shape of the pointer has been changed as the following image shows and click it to select the entire matrix.



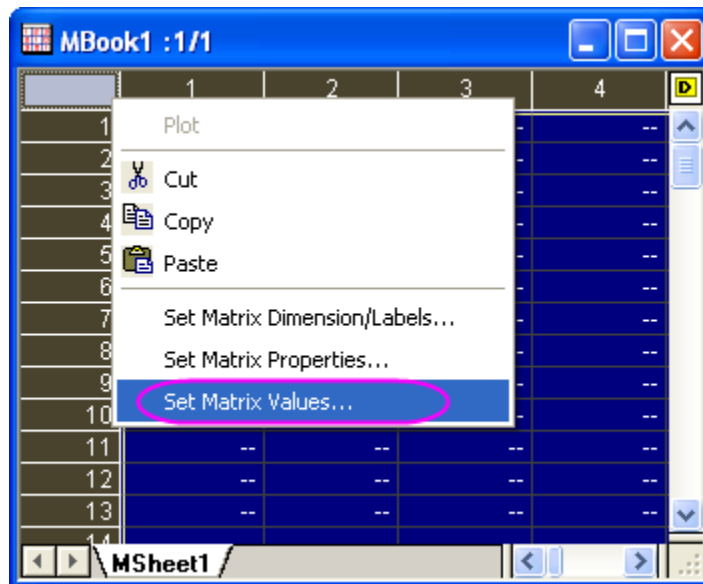
2. Right-click and select **Set Matrix Dimensions** from the context menu.



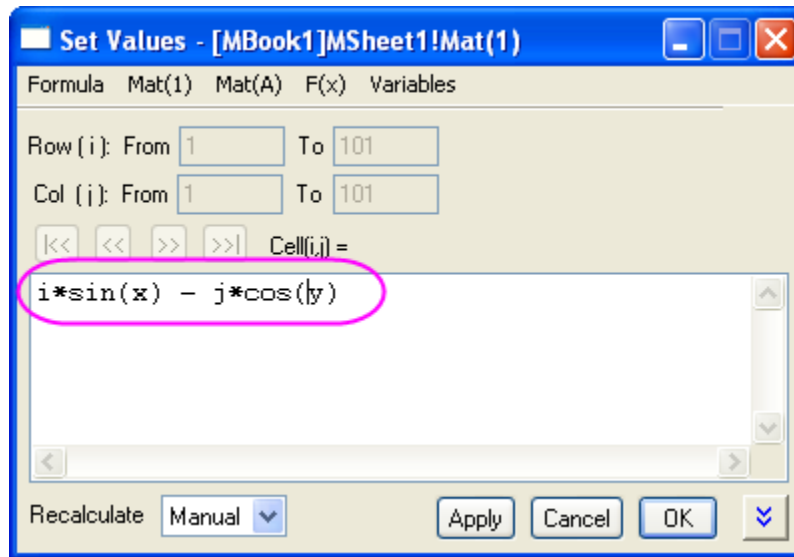
3. Set the pop-up dialog as the following image shows. Then click the **OK** button to finish setting dimensions.



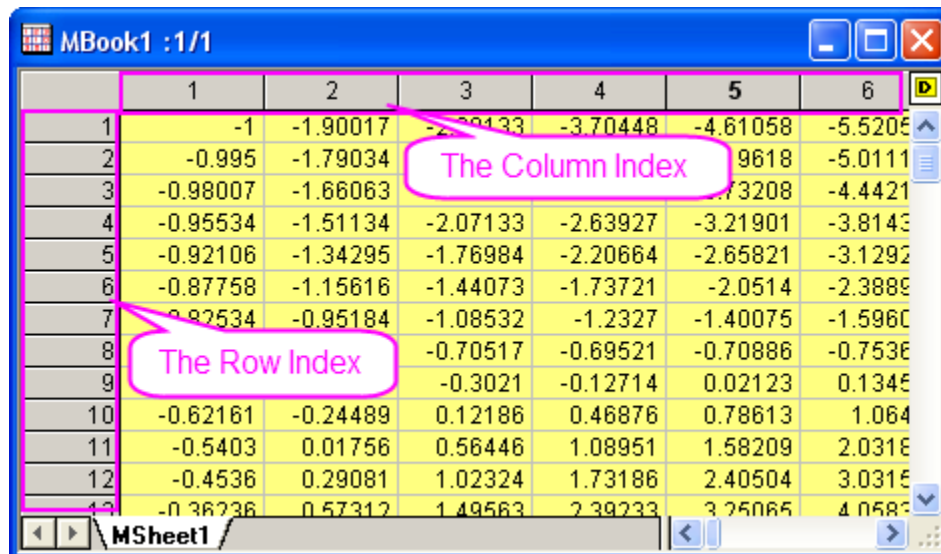
4. Highlight the matrix and right-click on it. Select **Set Matrix Values** to open the **Set Values** dialog.



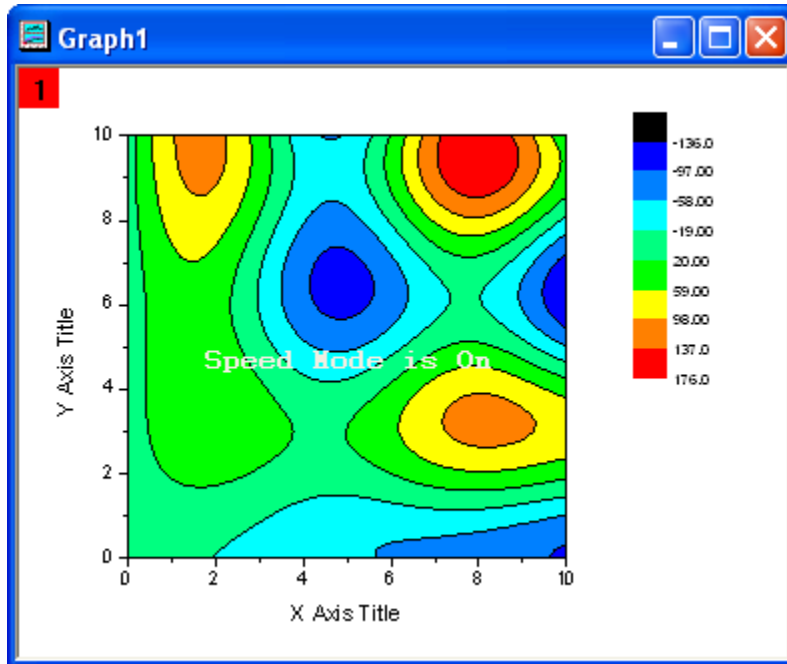
5. In the open dialog, input $i \cdot \sin(x) - j \cdot \cos(y)$ in the **Formula** edit box.



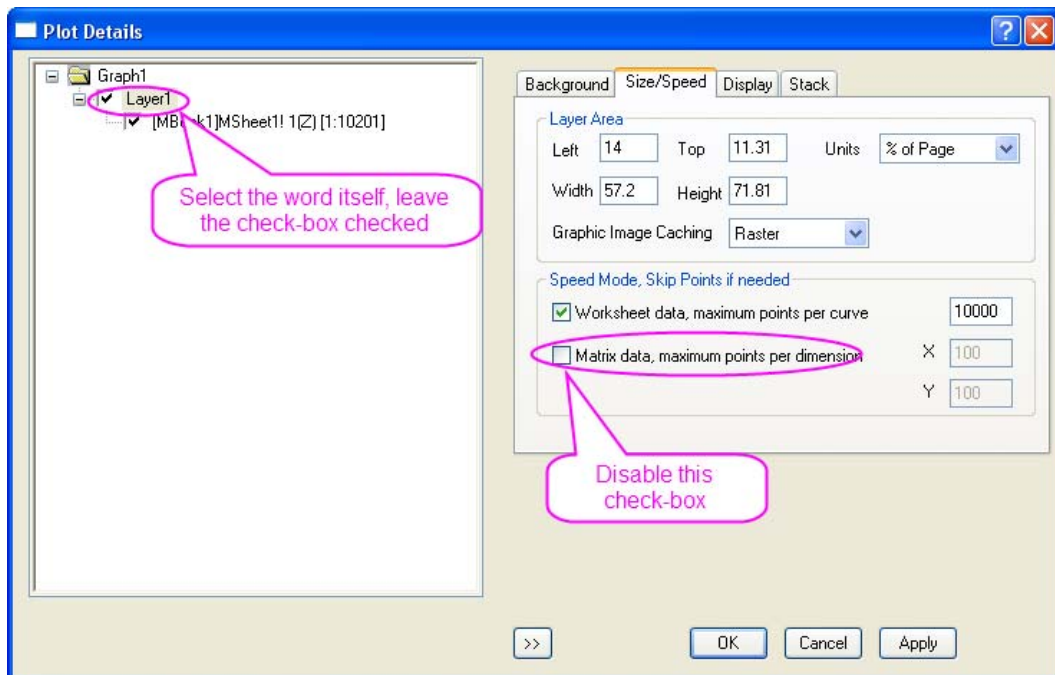
6. Click the **OK** button to generate data and the matrix look like this:



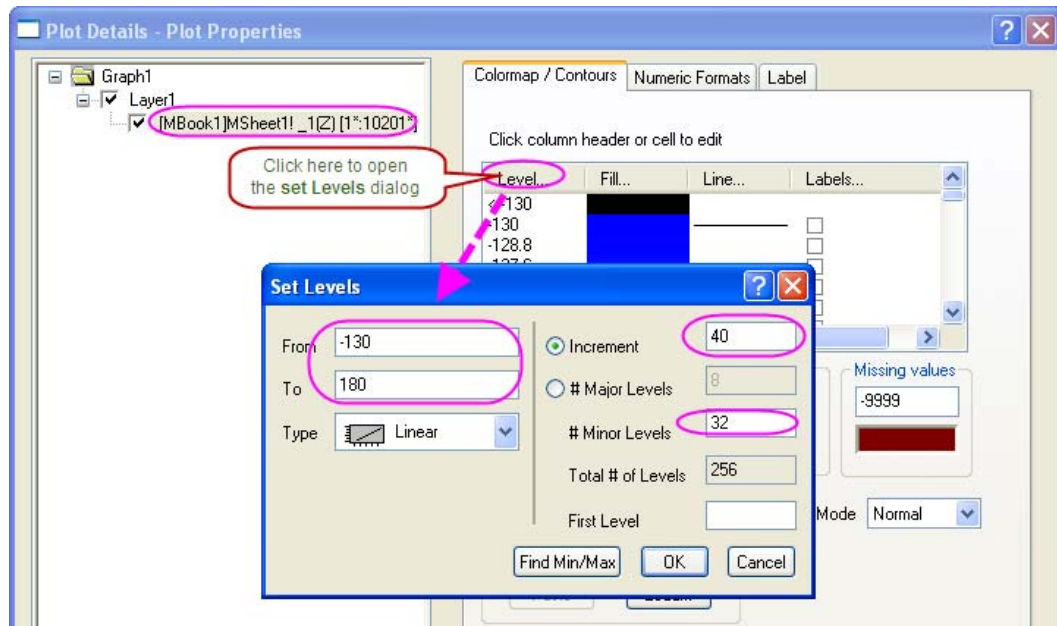
7. Activate this matrix and select **Plot: Contour: Contour - Color Fill** from the Origin main menu to create a contour plot. The graph looks like this:



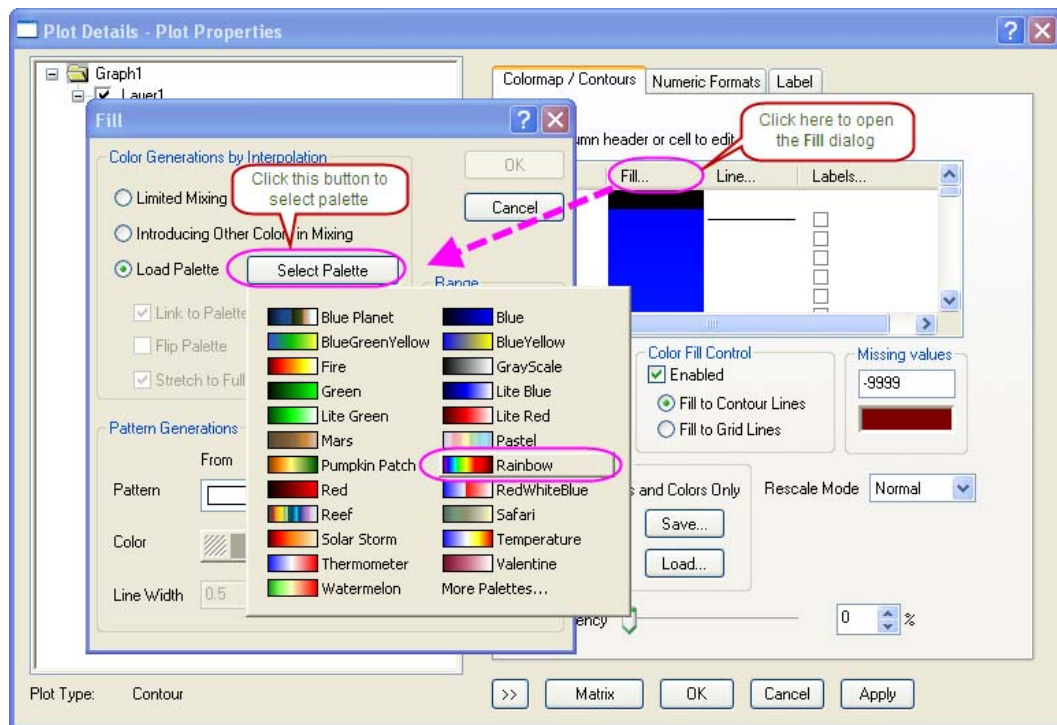
- Double-click on the contour plot to open the **Plot Details** dialog. Select **Layer1** in the left panel and then go to the **Size/Speed** tab in the right panel, disable the **Matrix data**, **maximum points per dimension** to turn off the **Speed Mode**.



- Select **[MBook1]MSheet1!1(Z)[1:10201]** to go to the matrix level. Click on the **Level** heading in the right panel to open the **Set Levels** dialog. Set values as the following image shows.

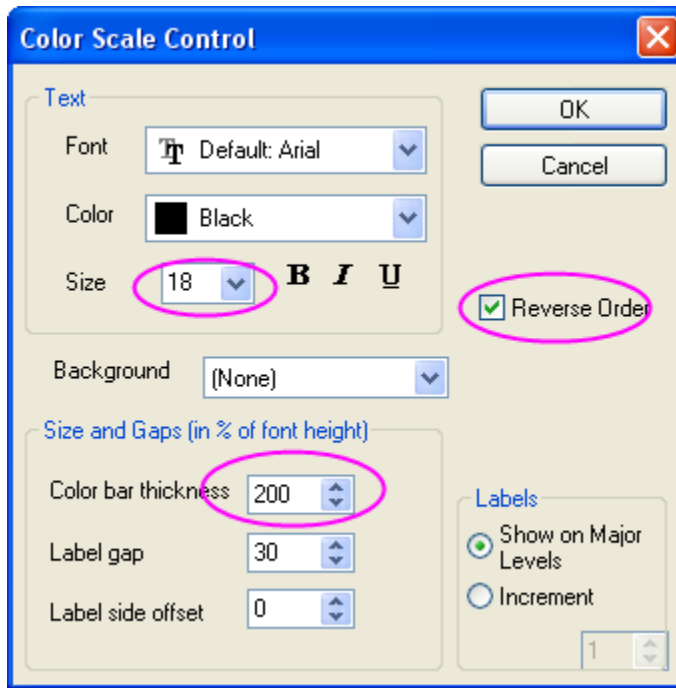


10. Click on the **Fill** heading to open the **Fill** dialog. Select **Load Palette** radio box and then click the **Select Palette** button to select the **Rainbow** palette. Click the **OK** button to close this dialog.

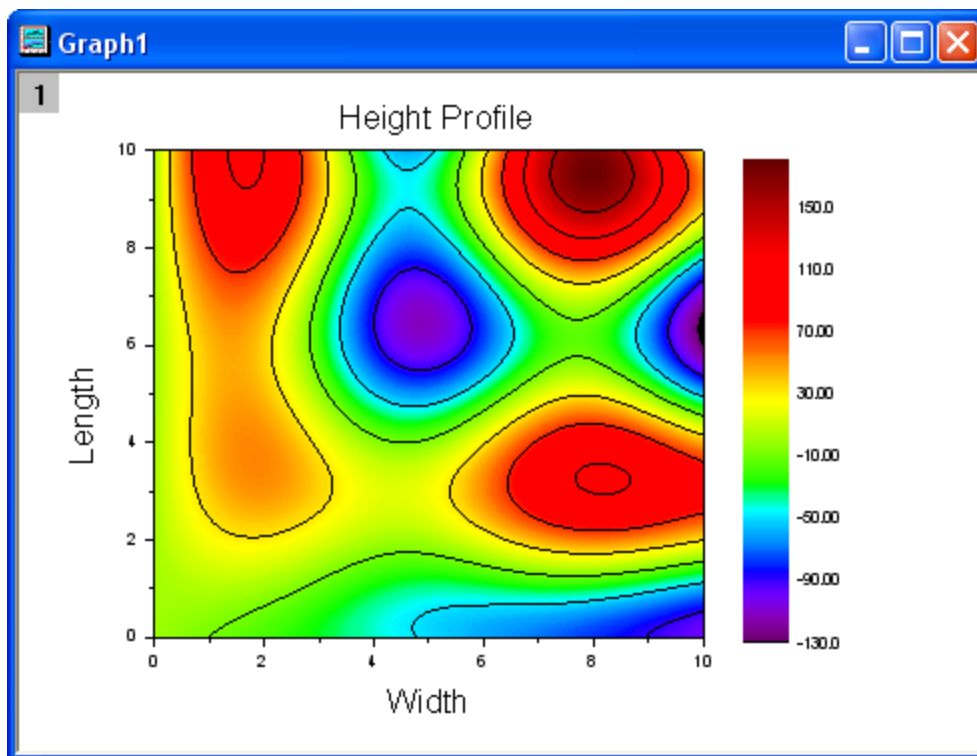


11. Click the **OK** button to close the **Plot Details** dialog.
12. Double click on the color scale to open the **Color Scale Control** dialog. Set **Size** to 18 and **Color bar thickness** to 200. Check the check-box before **Reverse Order**. Then click the **OK**

button to apply the settings to the color scale. Select the color scale and drag the handles around it to get a appropriate size.



- Change the labels of axes, label of X axis is **Width** and label of Y axis is **Length**, both font size is 31. Add a text object, say **Height Profile**, to the top of layer, font size is 31. Finally get a contour graph as follow:



3.5.3 Contour Plots and Color Mapping

Summary

Origin offers rectangular, polar, and ternary contour plots. For **rectangular contour plots**, the data can be either in a matrix or in a worksheet in XYZ format. **Polar contour plots** can be generated from three columns of data in a worksheet, organized either as R Θ Z or Θ R Z. **Ternary contour plots** can be generated from worksheet data organized in X Y Z Z format where the 2nd Z-column contains the 4th parameter which is the height value at a given XYZ point in the ternary space.

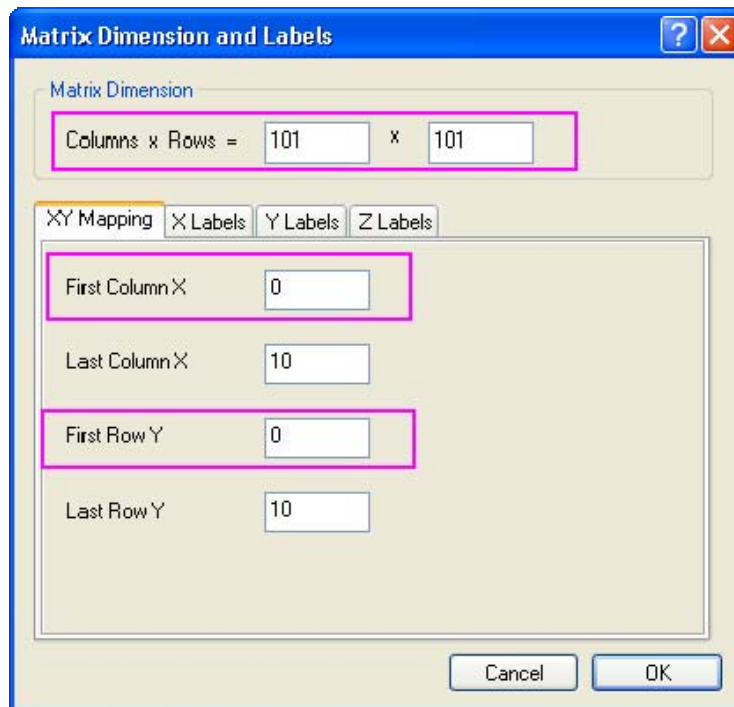
Many options are available for customizing contour plots, such as setting different major and minor contour levels, displaying contour lines only at major levels, applying color palettes, and there is also a control for a custom boundary in the case of contour plots created directly from the worksheet.

This tutorial will show you how to:

- Set values in a Matrix and create a Contour Plot
- Customize levels, lines, and color mapping
- Extract data from contour lines
- Create a Contour Plot directly from XYZ data
- Use a Custom Boundary

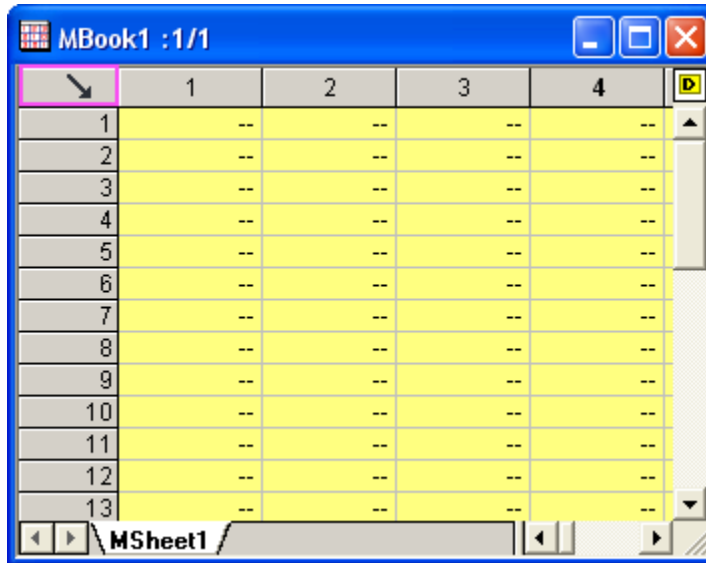
Create Contour Plot from Matrix

1. Click the **New Matrix button**  to create a new matrix.
2. Select **Matrix: Set Matrix Dimensions/Labels** from the main menu and set the dialog as shown in the following image:



Click **OK**.

3. Then move the mouse to the upper left corner of the matrix until the shape of the pointer has been changed, as the following image shows, and click it to select the entire matrix.



4. Input data into the matrix by using the **Set Values** dialog. Right-click on the matrix. Select **Set Matrix Values** to open the **Set Values** dialog. Input $i*\sin(x) - j*\cos(y)$ in the **Formula** edit box and click **OK** to generate data. The matrix should look like this:

	1	2	3	4	5
1	-1	-1.90017	-2.80133	-3.70448	-4.61058
2	-0.995	-1.79034	-2.58767	-3.38898	-4.19618
3	-0.98007	-1.66063	-2.34419	-3.03371	-3.73208
4	-0.95534	-1.51134	-2.07133	-2.63927	-3.21901
5	-0.92106	-1.34295	-1.76984	-2.20664	-2.65821
6	-0.87758	-1.15616	-1.44073	-1.73721	-2.0514
7	-0.82534	-0.95184	-1.08532	-1.2327	-1.40075
8	-0.76484	-0.73102	-0.70517	-0.69521	-0.70886
9	-0.69671	-0.49491	-0.3021	-0.12714	0.02123
10	-0.62161	-0.24489	0.12186	0.46876	0.78613
11	-0.5403	0.01756	0.56446	1.08951	1.58209
12	-0.4536	0.29081	1.02324	1.73186	2.40504
13	-0.36236	0.57312	1.49563	2.39233	3.25065
14	-0.2675	0.86267	1.97887	3.06729	4.11436

5. Each data point in the matrix corresponds to two different kinds of indices. One is the column and row indices. The other is the X and Y coordinates. You can select **View: Show X/Y** to see the X, Y coordinates.

The screenshot shows a window titled 'MBook1 :1/1'. A callout box labeled 'The X coordinates' points to the column headers (0, 0.1, 0.2, 0.3, 0.4). Another callout box labeled 'The Y coordinates' points to the row headers (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3). The matrix data is as follows:

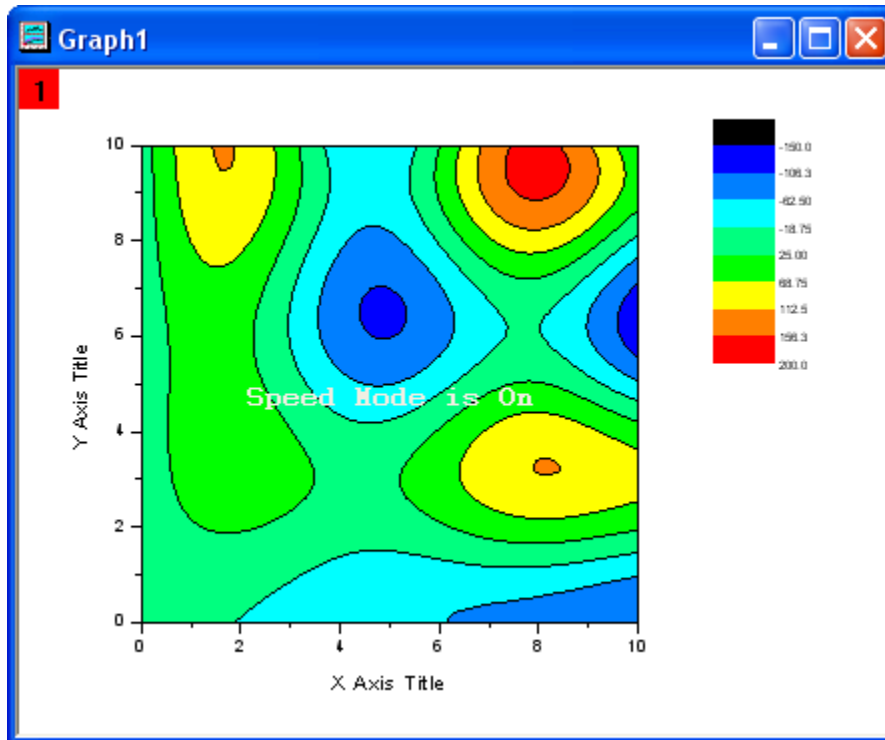
	0	0.1	0.2	0.3	0.4
0	-1	-1.90017	-2.80133	-3.70448	-4.61058
0.1	-0.995	-1.79034	-2.58767	-3.38898	-4.19618
0.2	-0.98007	-1.66063	-2.34419	-3.03371	-3.73208
0.3	-0.95534	-1.51134	-2.07133	-2.63927	-3.21901
0.4	-0.92106	-1.34295	-1.76984	-2.20664	-2.65821
0.5	-0.87758	-1.15616	-1.44073	-1.73721	-2.0514
0.6	-0.82534	-0.95184	-1.08532	-1.2327	-1.40075
0.7	-0.76484	-0.73102	-0.70517	-0.69521	-0.70886
0.8	-0.69671	-0.49491	-0.3021	-0.12714	0.02123
0.9	-0.62161	-0.24489	0.12186	0.46876	0.78613
1	-0.5403	0.01756	0.56446	1.08951	1.58209
1.1	-0.4536	0.29081	1.02324	1.73186	2.40504
1.2	-0.36236	0.57312	1.49563	2.39233	3.25065
1.3	-0.2775	0.86267	1.97887	3.06729	4.11436

6. Select **View: Show Matrix Thumbnails** to turn on thumbnails.

The screenshot shows a window titled 'MBook1 :1/1'. A thumbnail view of the matrix is shown in the top left corner, labeled '1'. The main window displays the same matrix data as in the previous screenshot.

	0	0.1	0.2	0.3	0.4
0	-1	-1.90017	-2.80133	-3.70448	-4.61058
0.1	-0.995	-1.79034	-2.58767	-3.38898	-4.19618
0.2	-0.98007	-1.66063	-2.34419	-3.03371	-3.73208
0.3	-0.95534	-1.51134	-2.07133	-2.63927	-3.21901
0.4	-0.92106	-1.34295	-1.76984	-2.20664	-2.65821
0.5	-0.87758	-1.15616	-1.44073	-1.73721	-2.0514
0.6	-0.82534	-0.95184	-1.08532	-1.2327	-1.40075
0.7	-0.76484	-0.73102	-0.70517	-0.69521	-0.70886

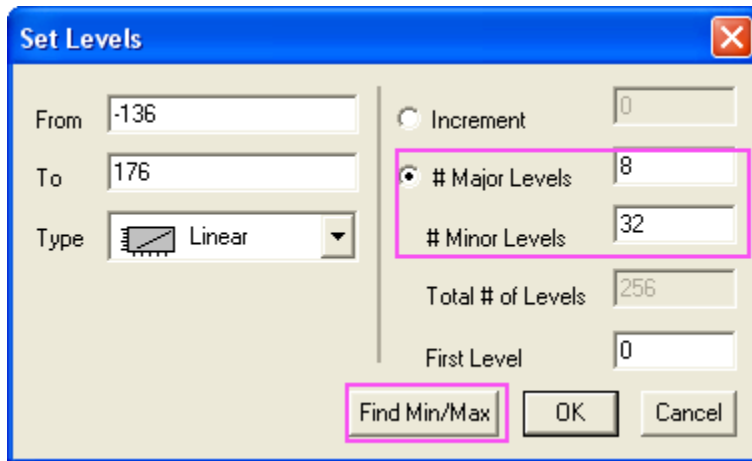
- Now, create a contour plot. Activate the matrix and select **Plot: Contour: Contour - Color Fill** to create a contour. Right-click on the contour and select **Display Caching: None**. The graph should look like this:



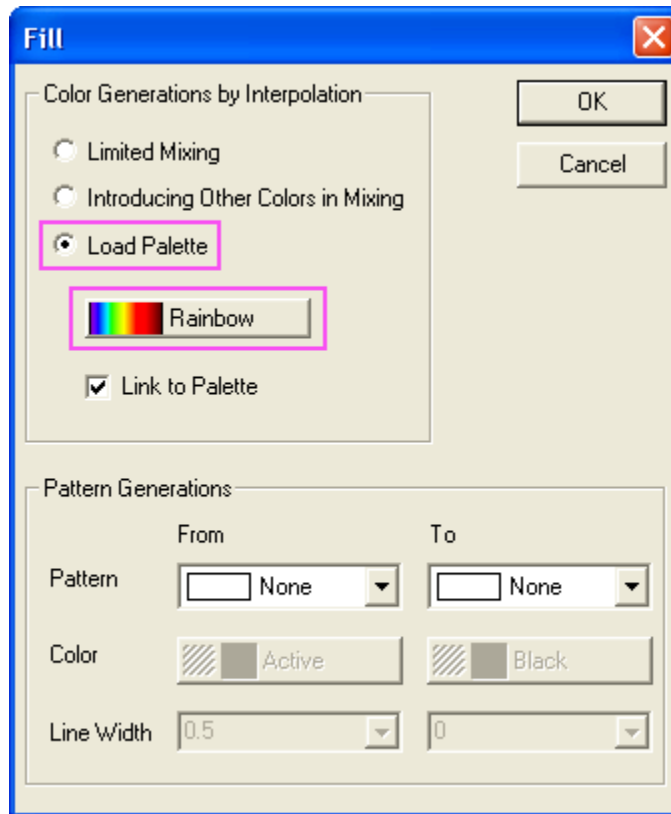
Customize Levels, Lines and Color Map

Origin makes it easy to customize a contour plot, including changing the color scale and adding labels. The following steps show you how.

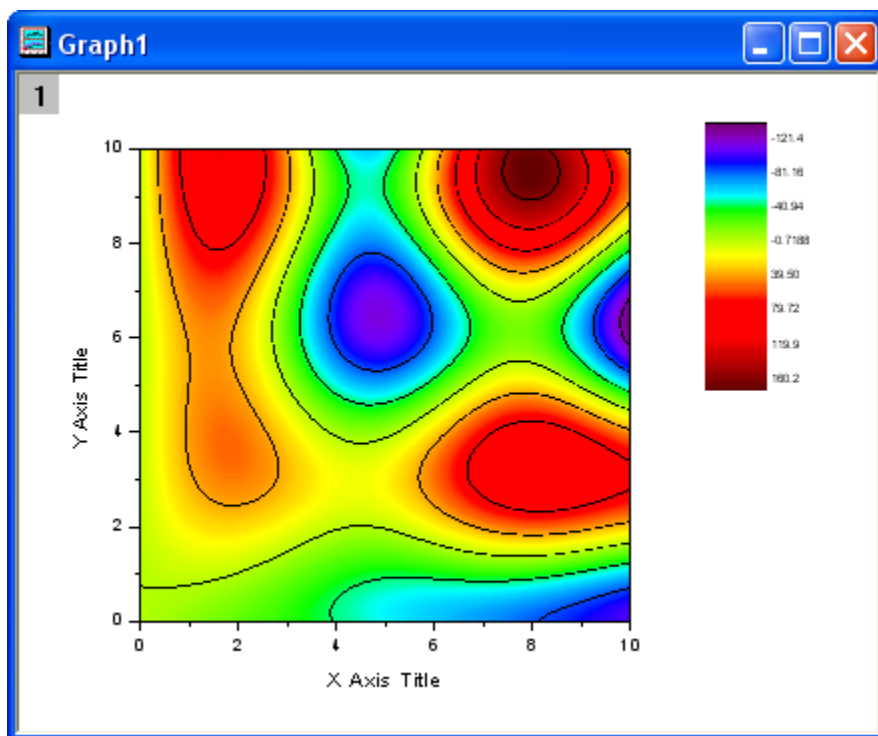
1. Double-click on the contour plot to open the **Plot Details** dialog. Click on the word **Layer1** in the left panel (leave the check box checked, and click on the word itself), and then select the **Size/Speed** tab in the right panel, and disable **Matrix data, maximum points per dimension** to turn off Speed Mode.
2. Then select **[MBook1]MSheet1!1(Z)(1:10201)** to go to the matrix level. Click on the **Level** heading in the right panel to open the **Set Levels** dialog. Click **Find Min/Max** button and then set **Major Levels** to **8** and **Minor Levels** to **32**, and then click OK to close it.



3. Now load a palette for the contour. Click on the **Fill** heading to open the **Fill** dialog, select the **Load Palette** radio box and then click the **Select Palette** button to select the **Rainbow** palette. Click Ok to close the dialog.



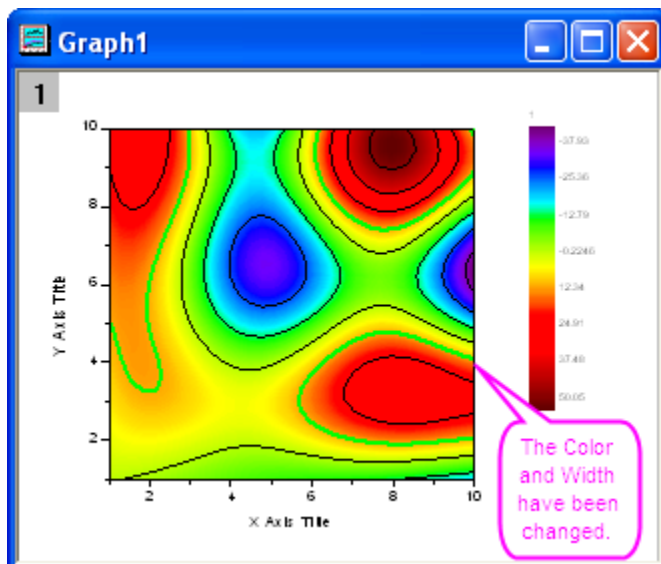
4. Click the **Line** heading to open the **Contour Lines** dialog. Then select **Show on Major Levels only** and click OK to close the dialog. Then click OK to close the Plot Details dialog.
5. Set the properties for the color scale. Right-click on the color scale and select **Properties** to open the **Color Scales Control** dialog. Select the **Show on Major Levels** check box and click OK to apply it to the color scale. Then the contour should look like



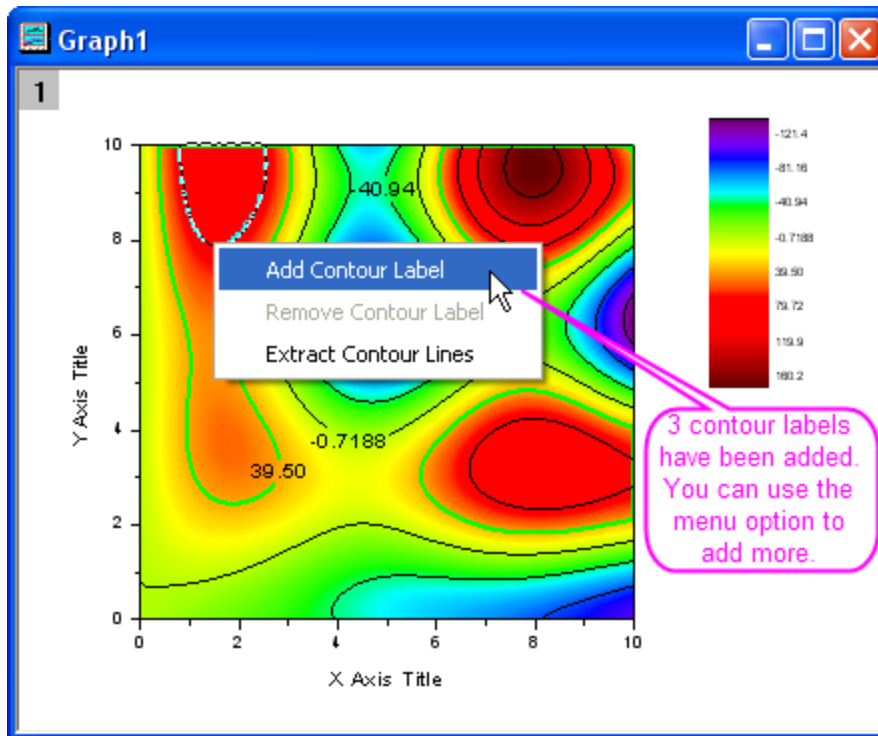
6. Click twice on a contour line (two single clicks, the first click selects the layer denoted by a red square at the plot center, the second click selects the contour line; if you mistakenly double-click, the Plot Details dialog will open) and then click again (a third time) to select all lines at that level.

Change the color to **Green** by using the Line/Border Color button  in the **Font** toolbar. Also

change the width to **3** by using the **Line/Border Width** button  in the same toolbar.



Right-click on the selected contour line and select **Add Contour Label** to add a label.




- Click twice on a contour line to select it, then right-click and select **Extract Contour Lines**. Origin will extract the data for that contour line to a worksheet. The following image shows part of the data in the worksheet.

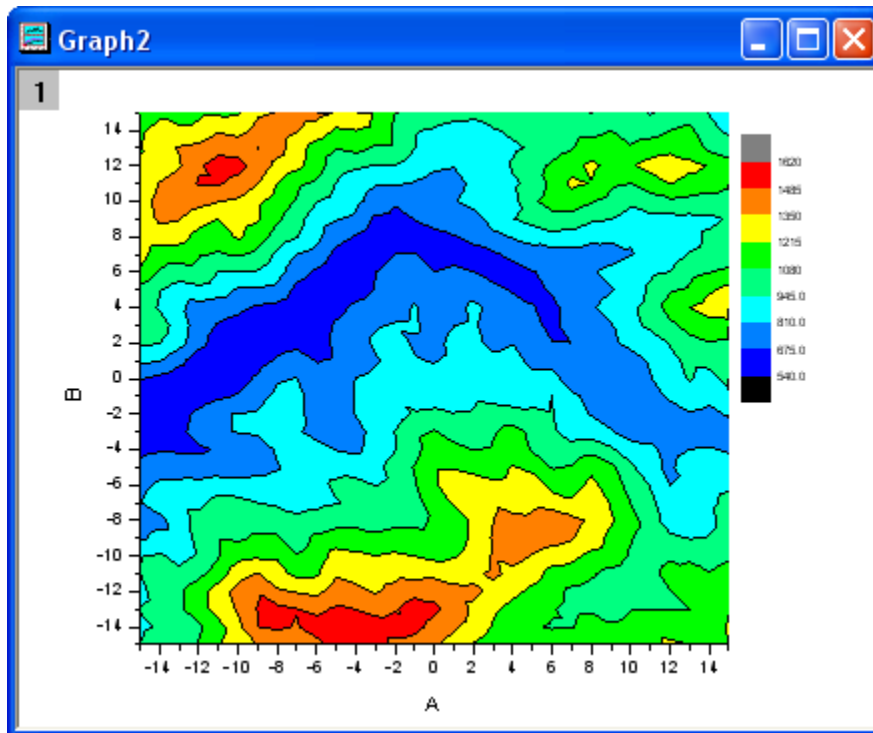
The figure shows a spreadsheet window titled "Book2" with a table of data. The table has columns for "Long Name", "Units", "Comments", "Z-Level", "A(X)", and "B(Y)". The "Z-Level" column contains the value 39.500000. The "A(X)" and "B(Y)" columns contain a series of numerical values corresponding to the contour line data.

	A(X)	B(Y)
Long Name		
Units		
Comments		
Z-Level	39.500000	39.500000
1	0.36099	10
2	0.36218	9.9
3	0.36377	9.8
4	0.36581	9.7
5	0.36832	9.6
6	0.37133	9.5
7	0.37487	9.4
8	0.37895	9.3
9	0.38362	9.2
10	0.38889	9.1
11	0.39479	9


Create a Contour Plot from XYZ Data

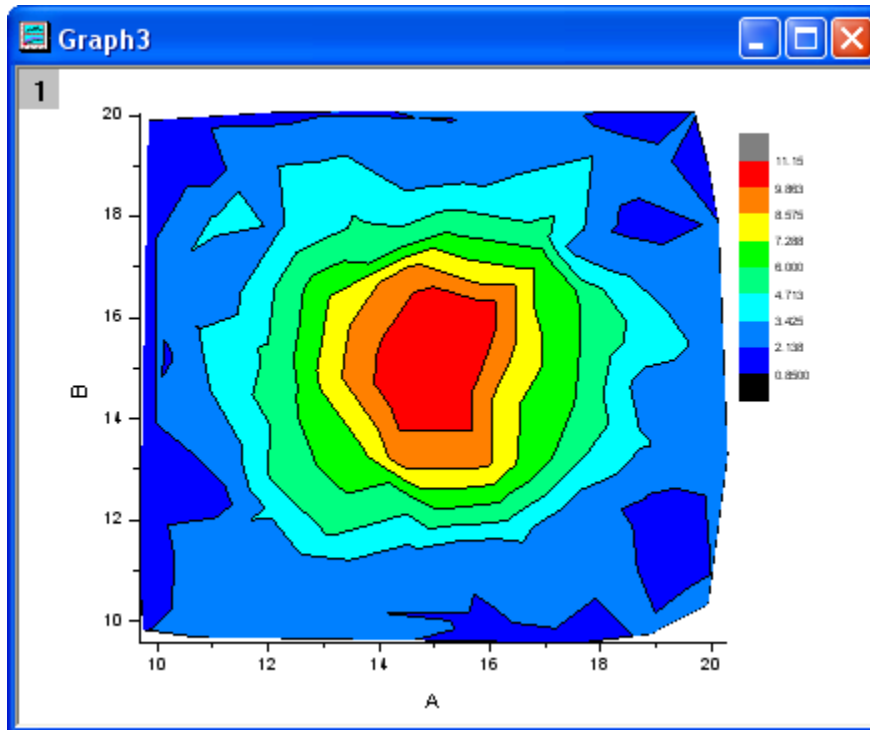
Origin can create contour plots directly from XYZ data in a worksheet without the need for an intermediate matrix. Delaunay Triangulation is used to compute and draw the contour lines.

1. Create a new worksheet and import the **3D XYZ.dat** from the **\Samples\Matrix Conversion and Gridding** folder by using the **Import Single ASCII** button .
2. Highlight the third column and right-click on it to select **Set As: Z**. Then select **Plot: Contour: Color Fill** to create a graph, as the following image shows.

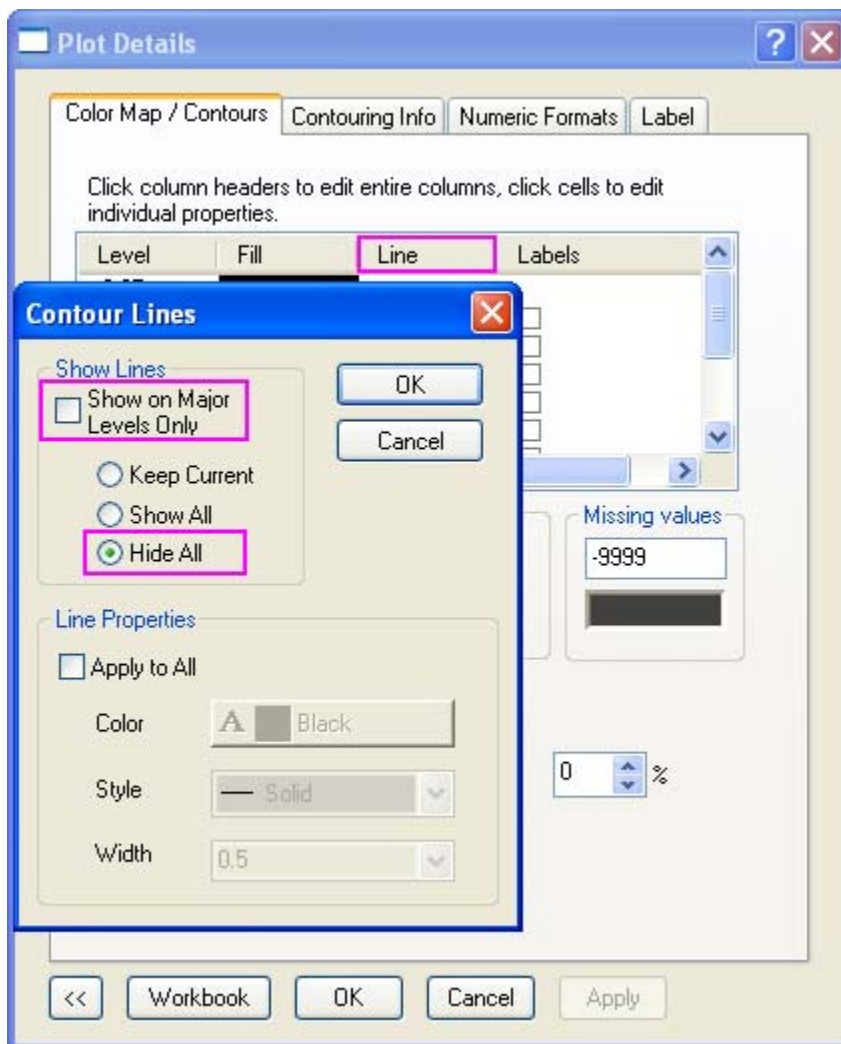


Show the Triangulation Grid

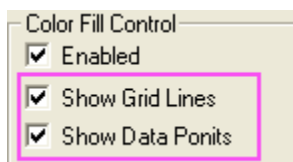
1. Show the triangulation grid that is used to create a contour plot from the data. The individual data points are located at the nodes or vertices of the grid. First, create a new worksheet and import the **XYZ Random Gaussian.dat** file from the **\Samples\Matrix Conversion and Gridding** folder by using the **Import Single ASCII** button .
2. Highlight the third column and right-click on it to select **Set As: Z**. Then select **Plot: Contour: Color Fill** to create a graph.



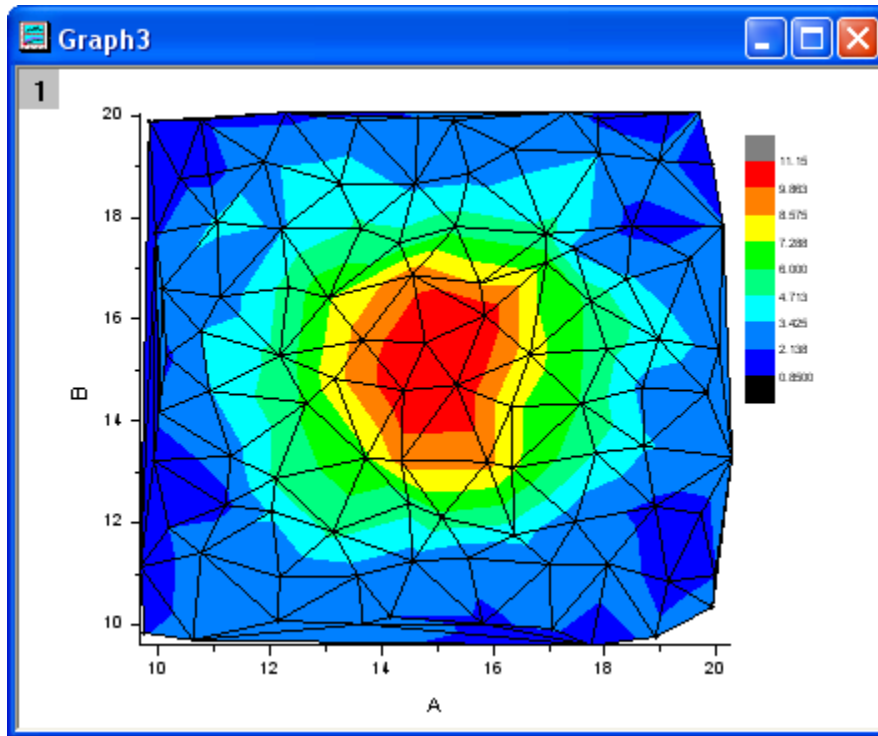
- Double-click on the contour plot to open the **Plot Details** dialog. Go to the **Color Map/Contour** tab, click on the **Line** heading to open the **Contour Lines** dialog, and select **Hide All** to hide all the contour lines. Click OK to close this dialog.



- Then select the **Show Grid Lines** and **Show Data Points** check boxes in the **Color Map/Contours** tab. Click OK to close the Plot Details dialog.




The contour plot with its superimposed grid should look like this:



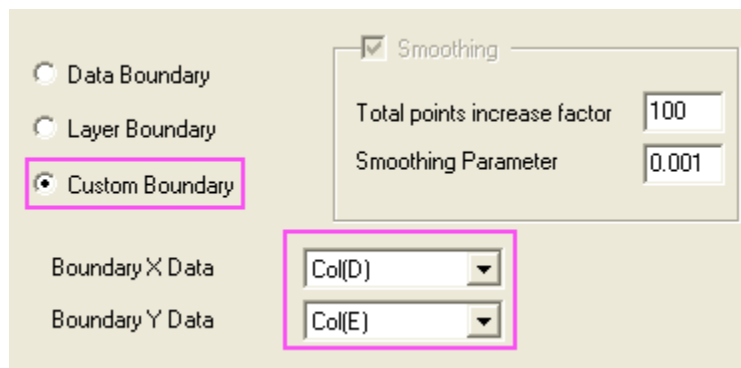
Apply Custom Boundary

A custom boundary can be applied to contour plots created directly from XYZ data, where the user provides the data points for the boundary in additional columns on the same worksheet. This feature is useful when you have specific boundary data that defines the outline of an object, such as, say, the profile of an engine, and the contour shows engine temperature.

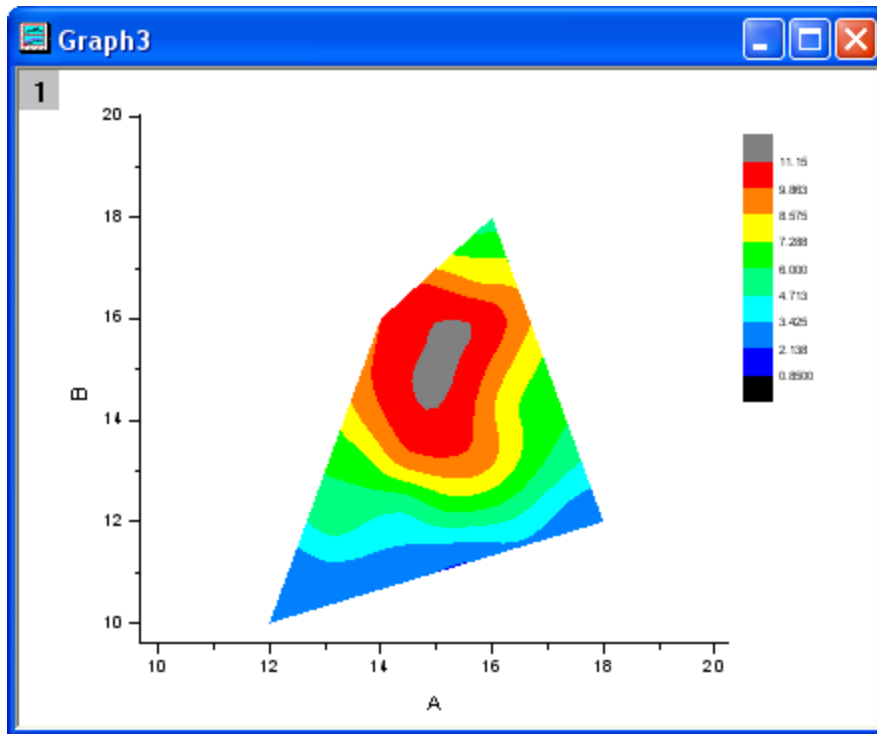
1. We will continue to use the worksheet and the contour plot created with **XYZ Random Gaussian.dat** in the previous example.
2. Go to the **XYZRandomGaus** workbook and click the **Add New Columns** button  twice to add two columns. Highlight the two columns and right-click to select **Set As: XY XY**. Enter four rows of data, like you see below:

12	10
18	12
16	18
14	16

3. Double-click on the contour plot to open the **Plot Details** dialog. Select the **Contour Info** tab in the right panel, and set the dialog like the following image shows, to customize the boundary of the contour plot. Remember to uncheck the **Show Grid Lines** and **Show Data Points** check boxes in the **Color Map/Contours** tab. Click OK to close the Plot Details dialog.



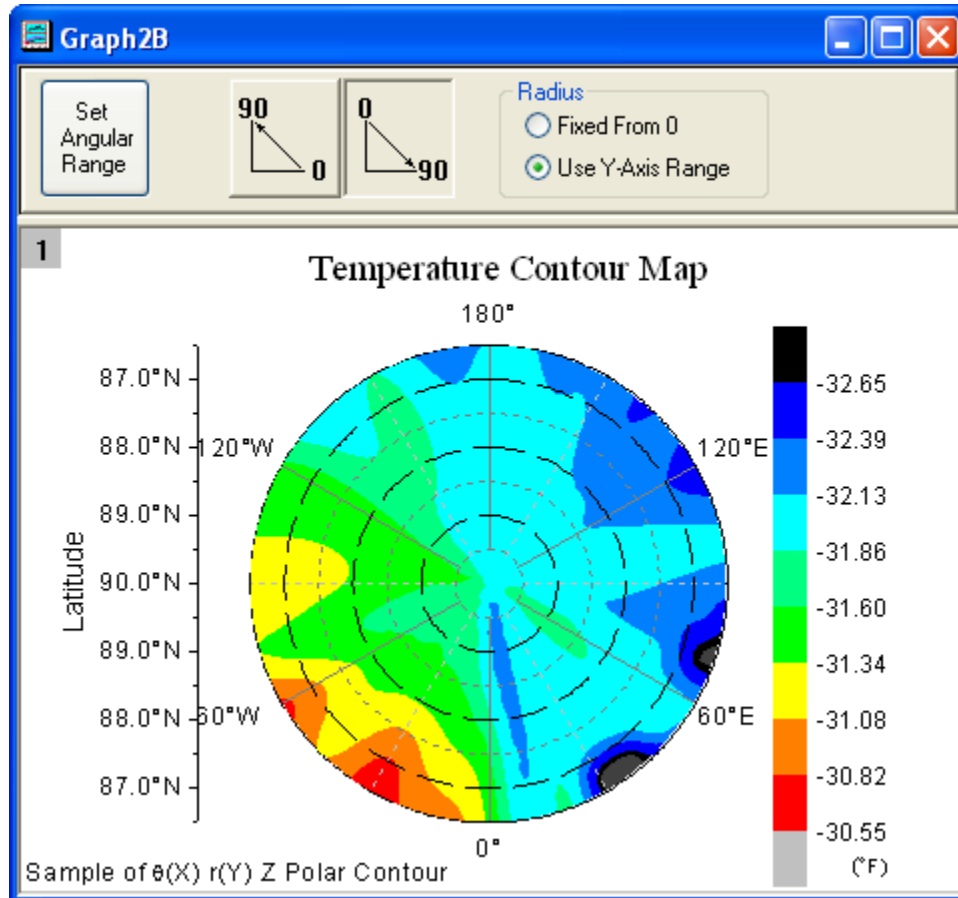
The graph should look like:



3.5.4 Polar Contour

Summary

This tutorial will show you how to create a Polar Contour graph.



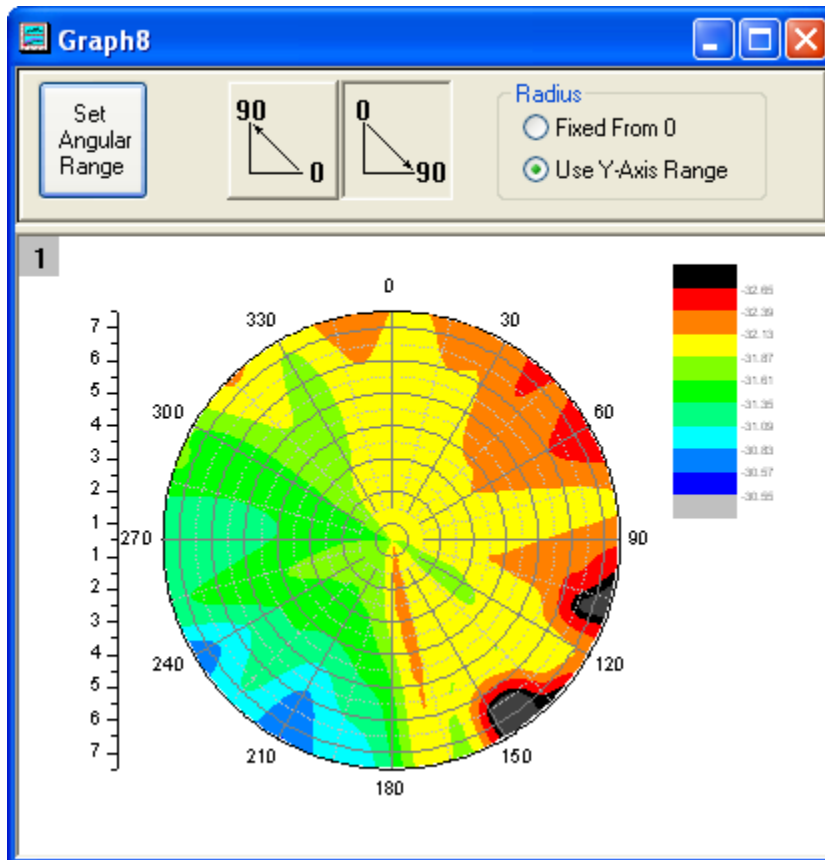
What will you learn

- Create a Polar Contour graph
- Customize the graph by using the Plot Details dialog

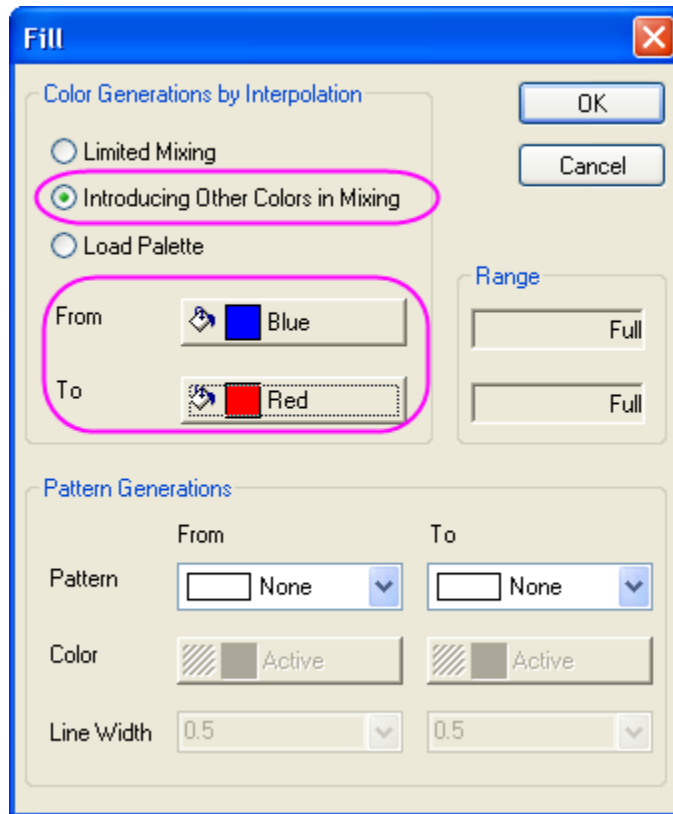
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

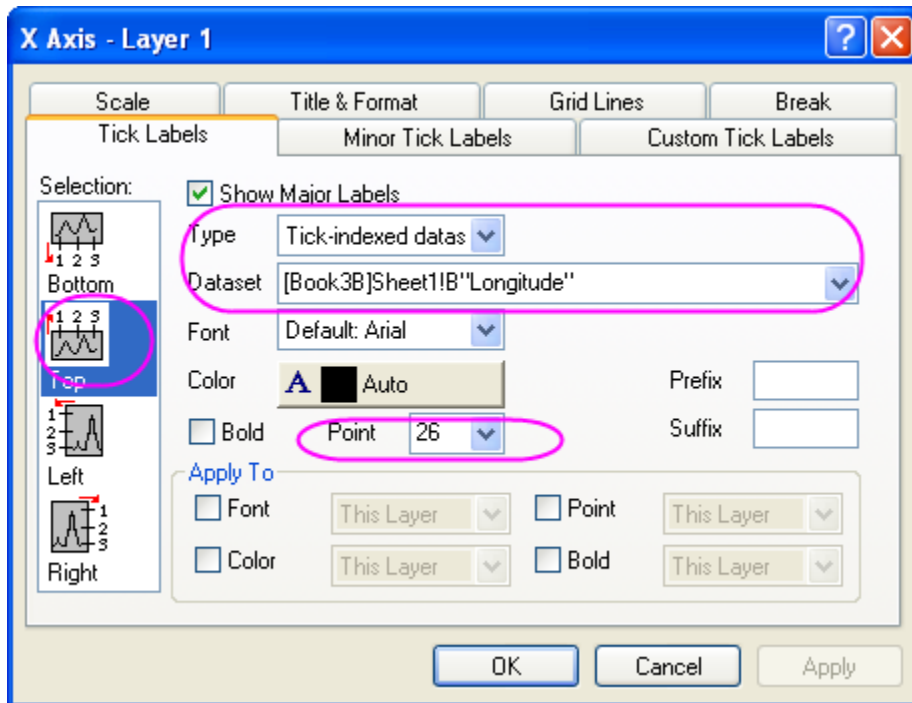
1. Open the *Graphing: Contour: Polar Contour* folder in the **Project Explorer**. Activate **Book2B** with the temperature and location data and select column **C**. Select **Plot: Contour: Polar Contour Theta(X) r(Y)** from the menu to create a polar contour graph. The graph should look like:



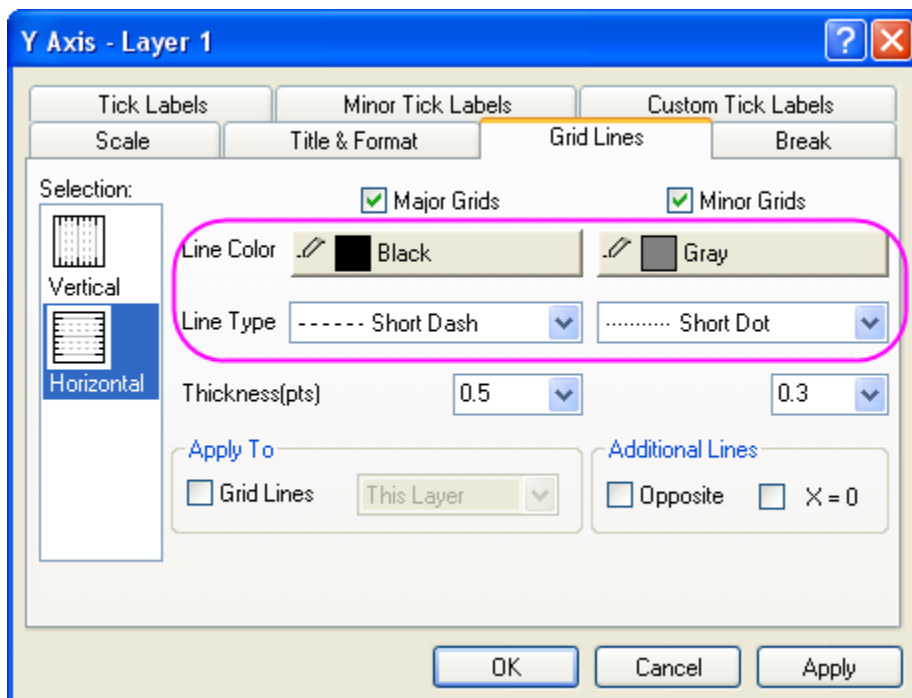
- Then we will customize the contour graph by using the **Plot Details** dialog. Double-click on the contour plot to bring up Plot Details dialog, select the **Color Map/Contours** tab and click the **Fill** heading, and then set the dialog as the following graph shows.



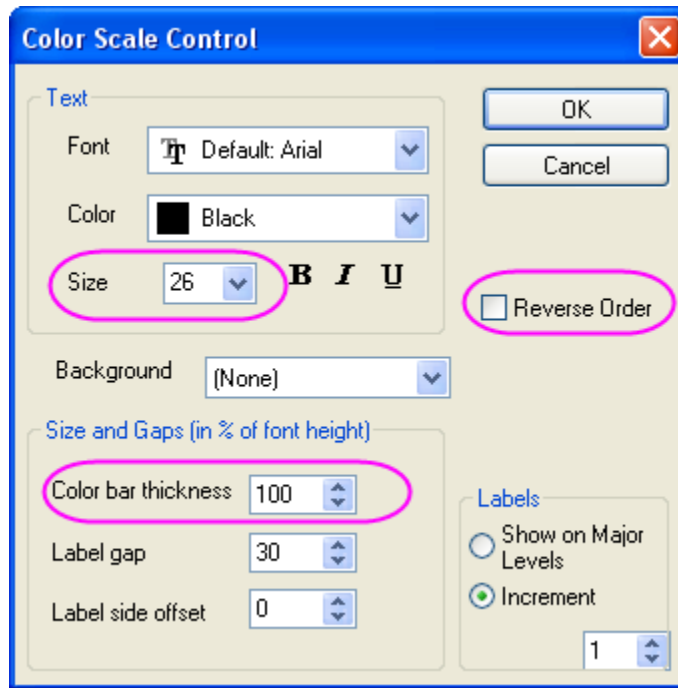
3. Click the **Contouring Info** Tab, select the **Layer Boundary** radio button.
4. Click OK button to close the **Plot Details** dialog.
5. Then we will customize the axes of the graph. Double-click on any tick label to open the **Axis** dialog. Then do the following things
 - o Select the **Scale** tab, choose **Horizontal** in the **Selection** list, set **Increment** to 60.
 - o Select **Vertical** in the **Selection** list, set **From** to 0, set **To** to 7 and set **Increment** to 2.
 - o Select **Tick Labels** tab and select **Top** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!B** for **Dataset**. Change **Point** to 26.



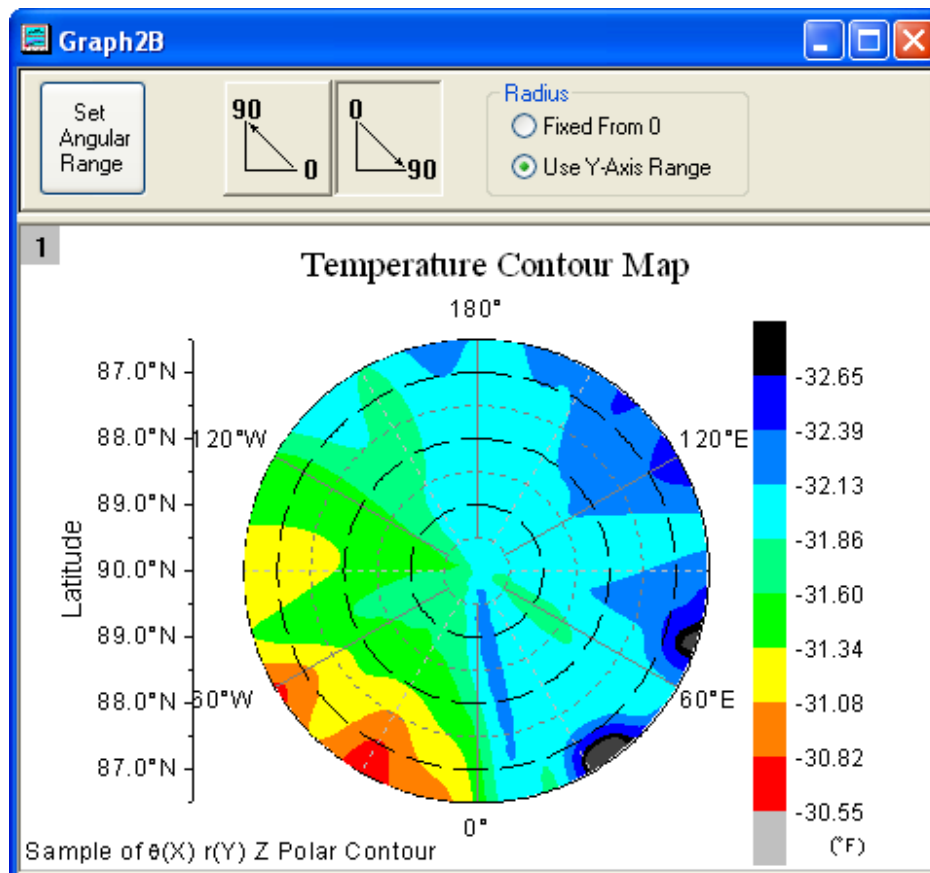
- Select **Left** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!A** for **Dataset**. Change **Point** to **26**.
- Select **Right** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!A** for **Dataset**. Change **Point** to **26**.
- Go to the **Grid Lines** tab and make sure **Horizontal** has been selected in the **Selection** list. Then set the dialog as the following image shows.



6. Double click on the Color Scale to open the **Color Scale Control** dialog and set the dialog as the following image shows.



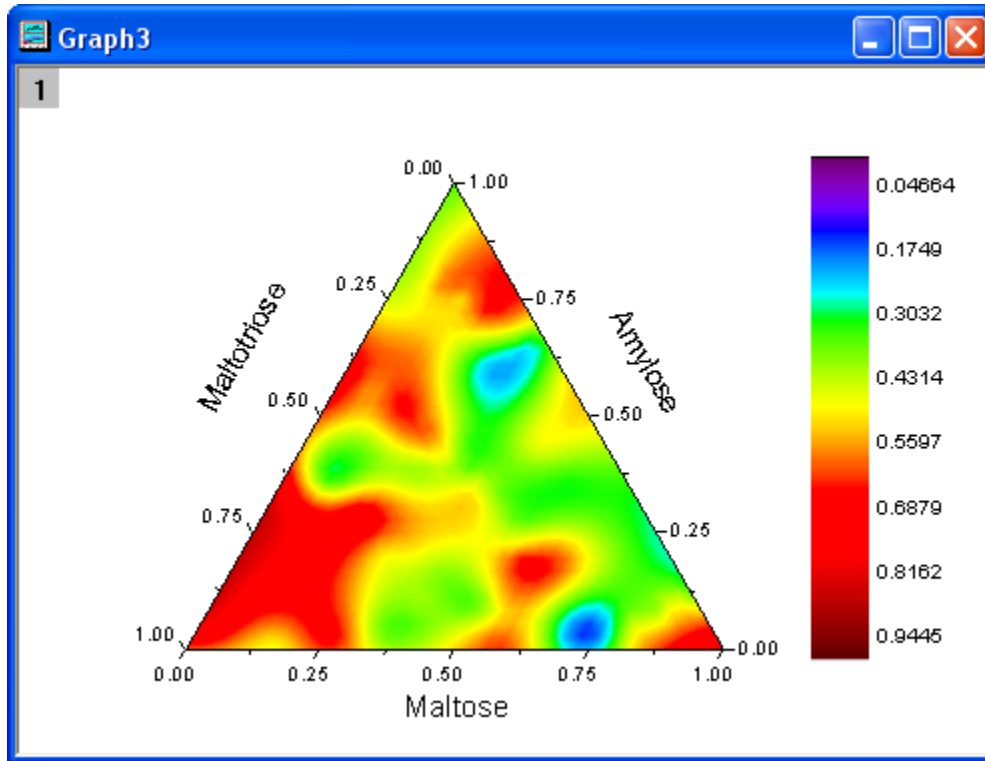
7. The graph should look like



3.5.5 Ternary Contour

Summary

This tutorial will show you how to create a Ternary Contour graph.



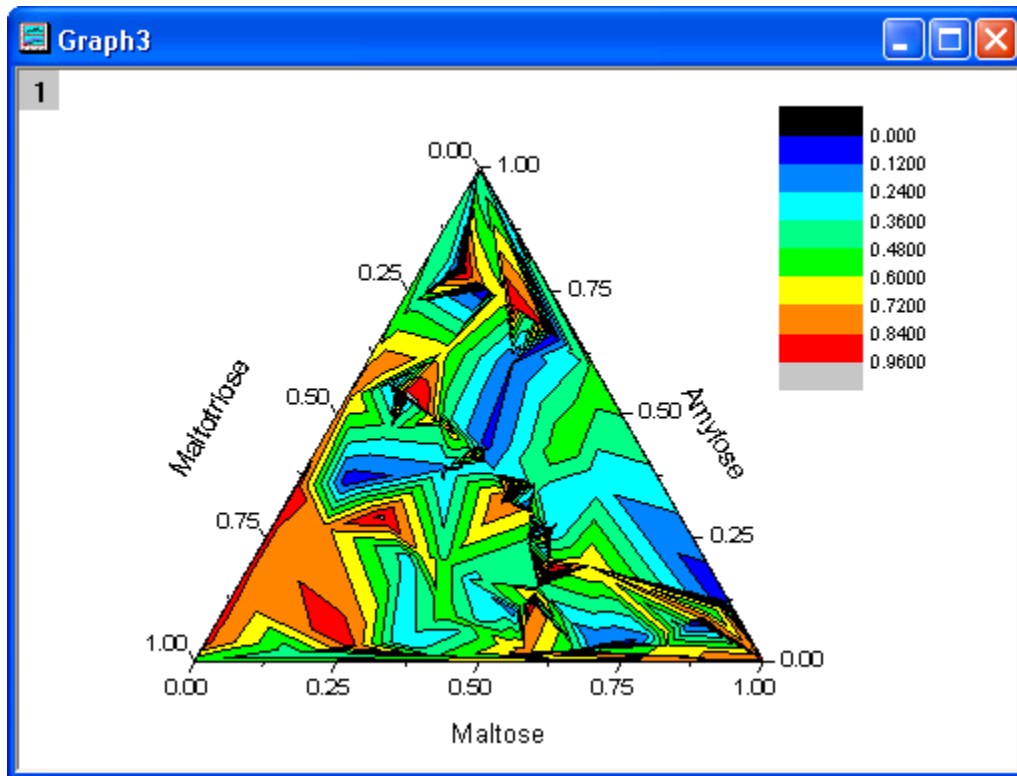
What will you learn

- Create the Ternary Contour graph
- Customize the graph by using the Plot Details dialog

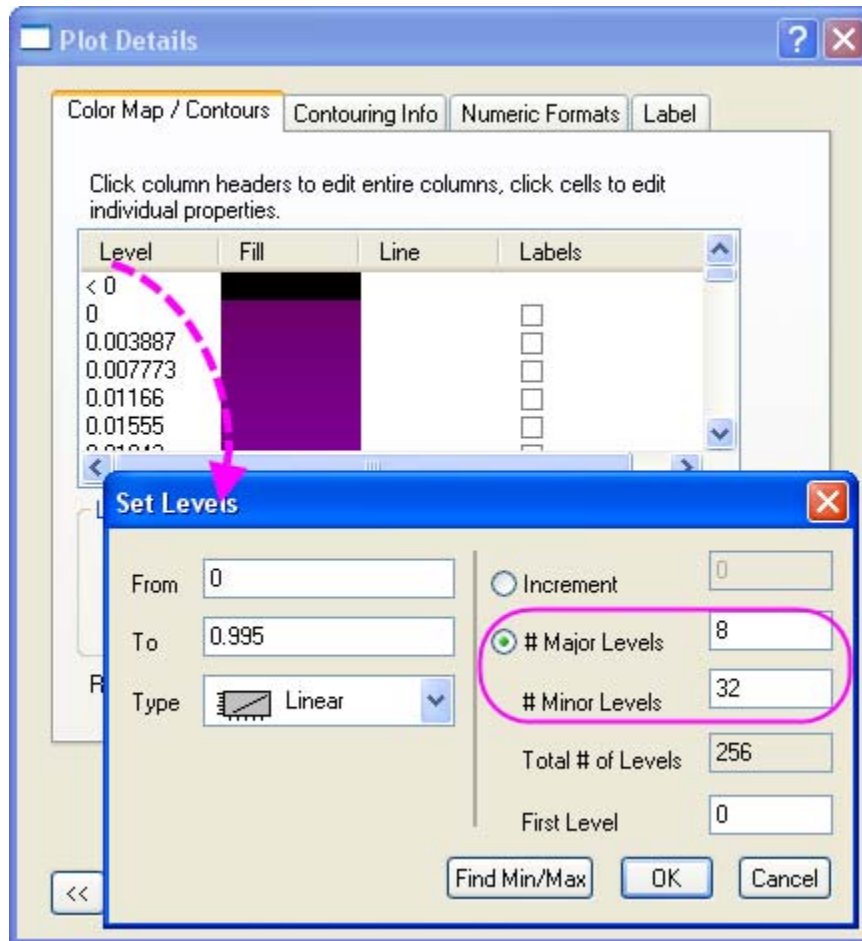
Steps

This tutorial is associated with the sample project: *\Samples\Samples.opj*.

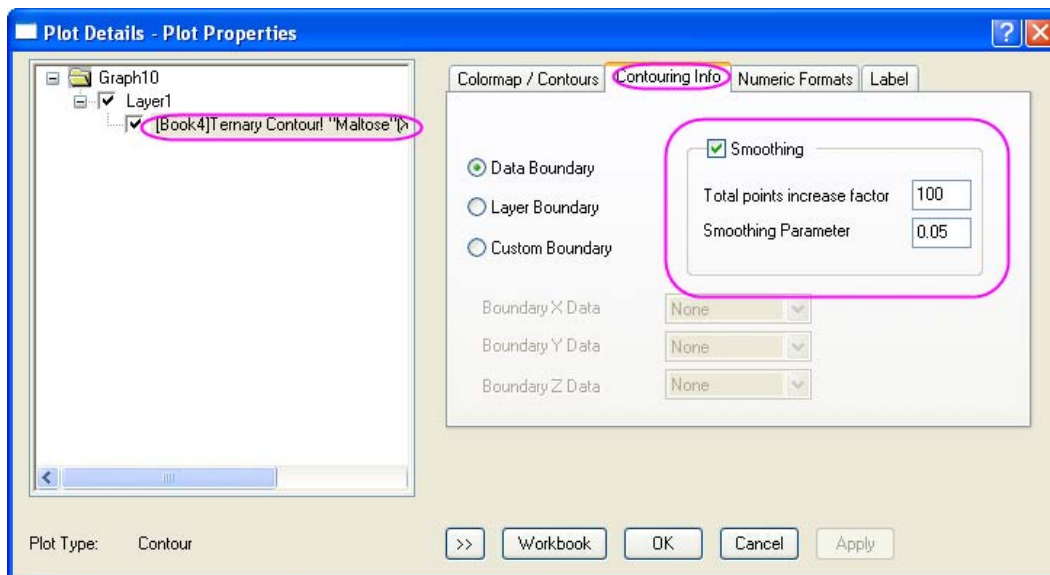
1. Open the *Graphing: Contour: Ternary Contour* folder in the **Project Explorer**. Activate **Book4**, highlight entire worksheet and select **Graph: Contour: Ternary Contour** to create a Ternary Contour graph. The graph should look like:



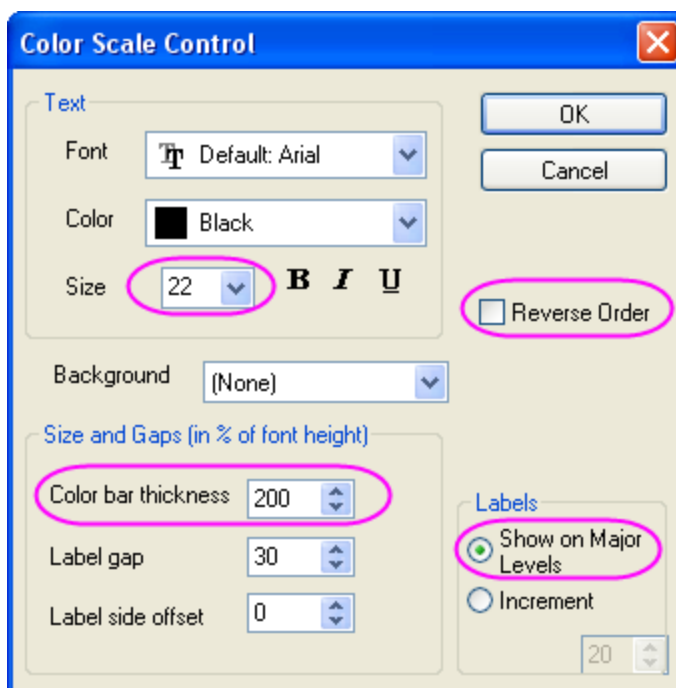
2. Now we will customize the contour graph with the Plot Details dialog. Double-click on the contour plot to bring up **Plot Details** dialog.
 - select the **Color Map/Contours** tab and click the **Level** heading, then set the dialog as the following graph shows.



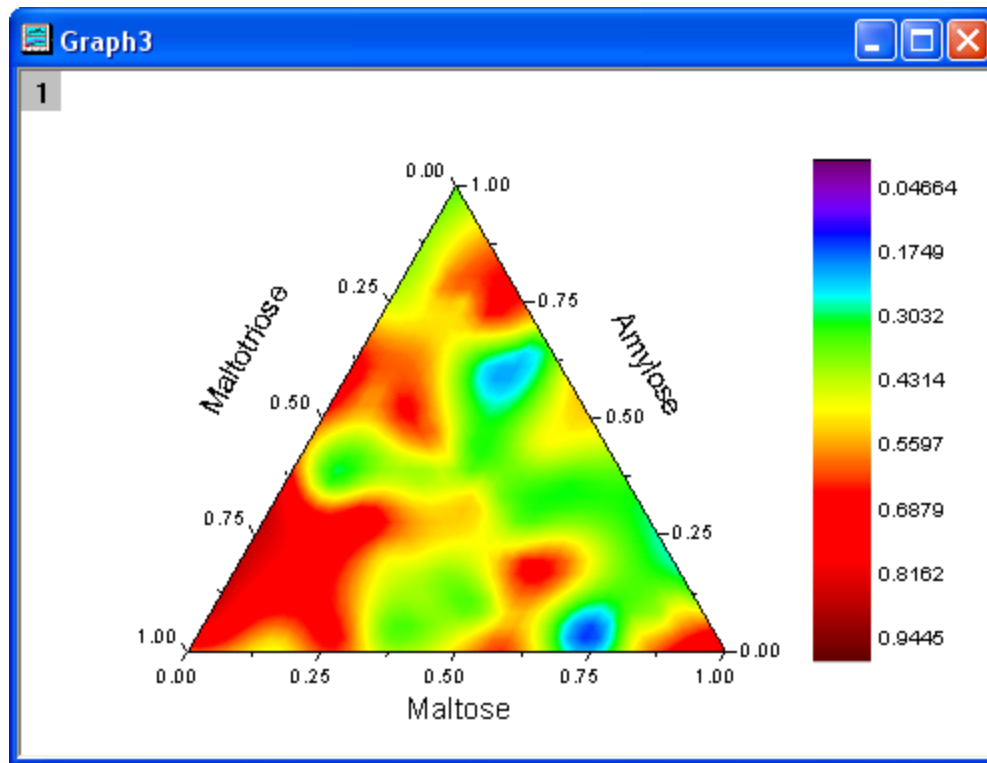
- Click OK to close the **Set Levels** dialog then click the **Fill** heading to open the **Fill** dialog. Select **Load Palette** and then select **Rainbow** from the **Palette** list and click OK.
- Click the **Lines** heading, uncheck the **Show on Major Level only** check box and then select the **Hide All** option and click OK.
- Go to the Contouring Info tab, check the **Smoothing** check box.



- o Click OK to close the **Plot Details** dialog.
- o Double click on the **Color Scale** to open the **Color Scale Control** dialog and set the dialog as the following graph shows.



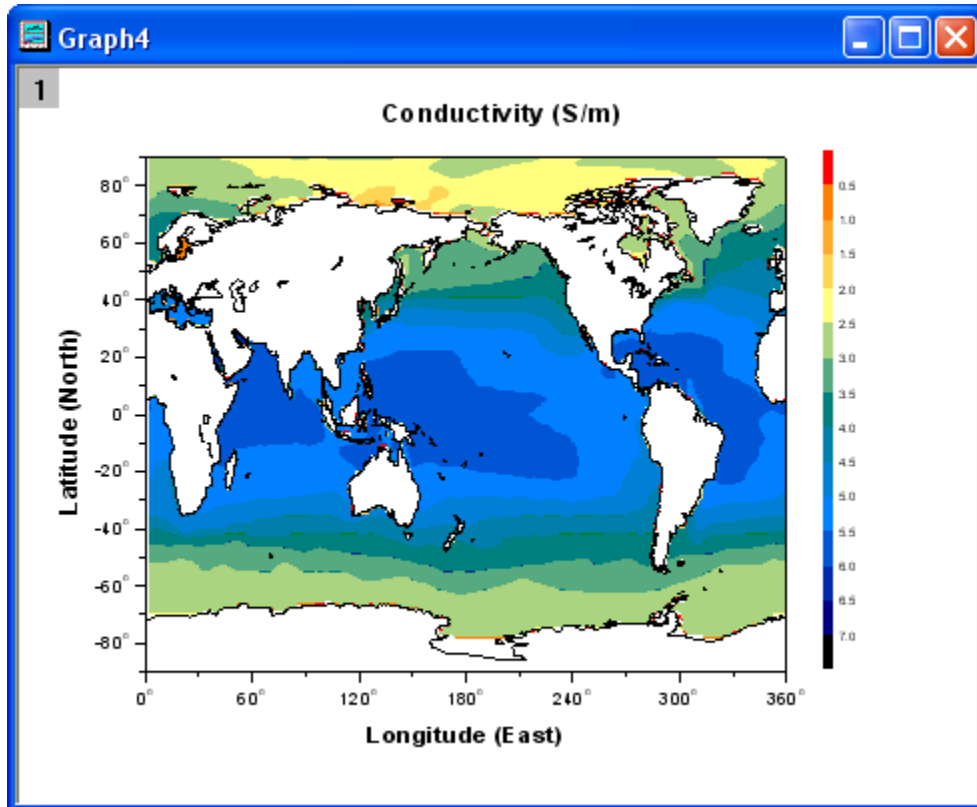
The graph should look like



3.5.6 Combining Line and Contour Plots

Summary

This tutorial will show you how to create a Word Map by combining Line Plot and Contour graph.



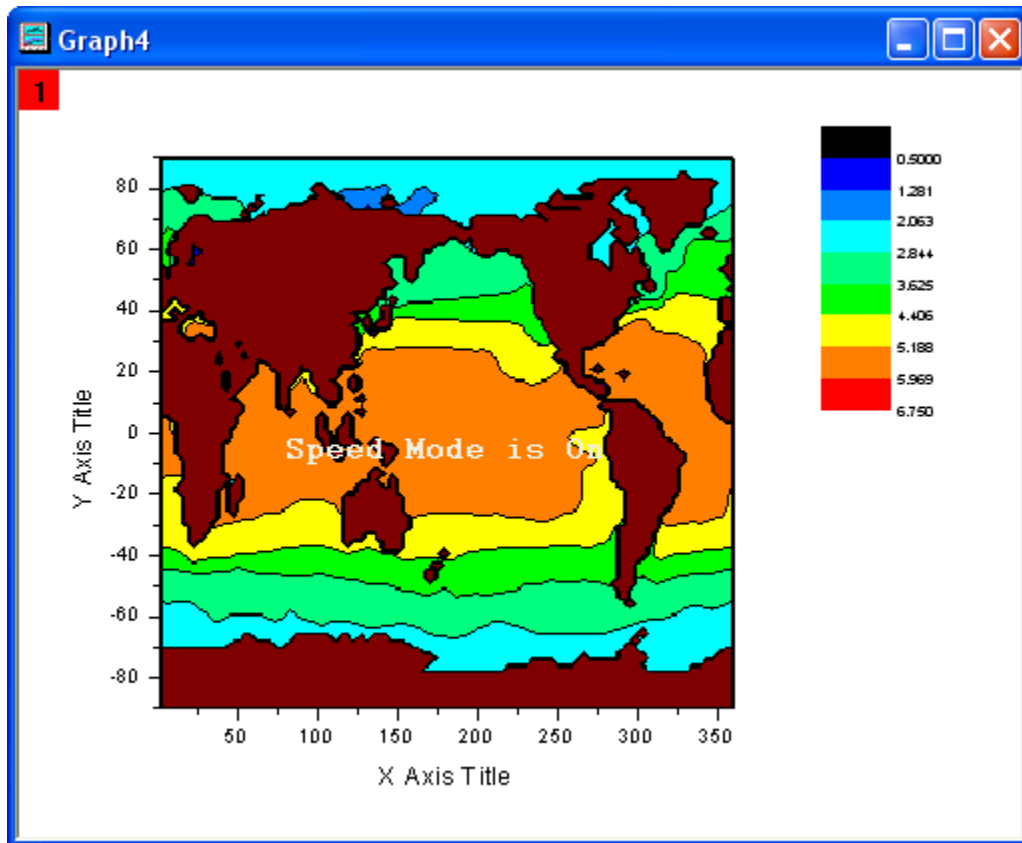
What will you learn

- Create a contour graph
- Combine a Line Plot and a Contour graph
- Customize the contour graph

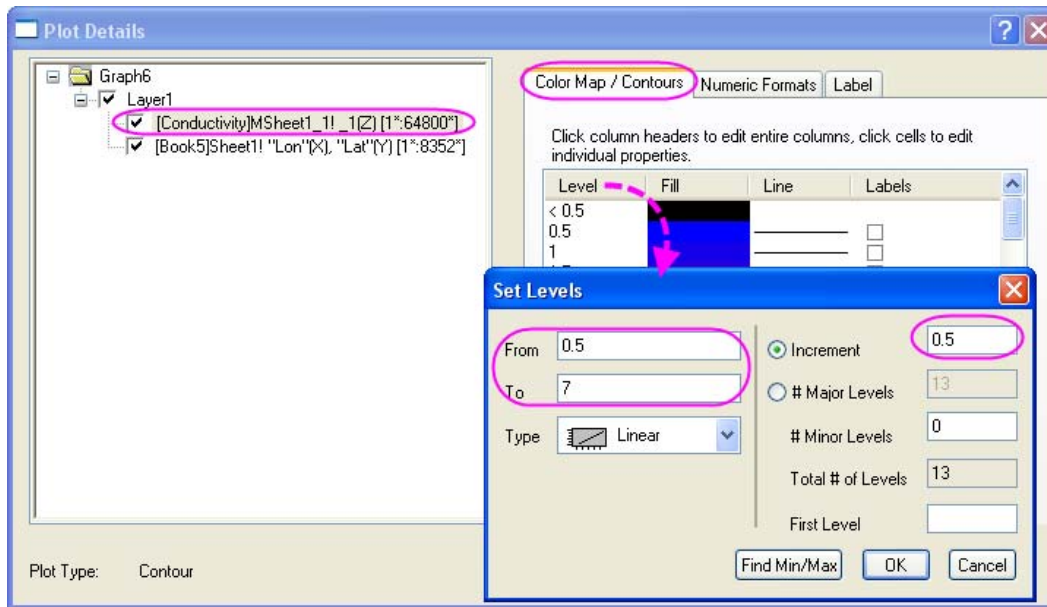
Steps

This tutorial is associated with the sample project: `\Samples\Sample.opj`.

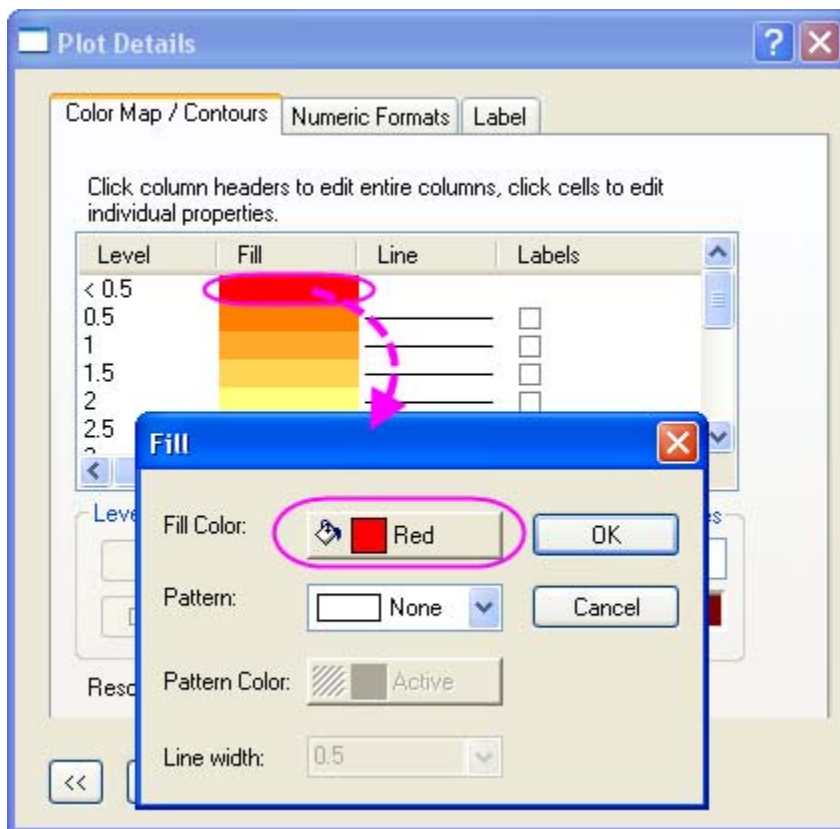
1. Open the *Graphing: Contour: Map Combining Line and Contour Plots* folder in the **Project Explorer**. Activate **Conductivity** matrix and highlight entire matrix. Select **Plot: Contour: Contour - Color Fill** to create a contour graph.



2. At this step, we will add a Line plot to the contour graph. Select all columns of **Book5** and then active the new created contour graph. Then select **Graph: Add Plot to Layer: Line** to create a new line plot in the same layer. Delete the legend for the added line plot.
3. In the following steps, we will customize the graph. Select **Format: Layer Properties** to open the Plot Details dialog. Select the **Size/Speed** tab, disable two checkboxes that in the **Speed Mode, Skip Points if needed** group.
4. Expand the **Layer1** branch and select contour plot in the left panel of the Plot Details dialog. Then do the following things:
 - o Go to the **Color Map/ Contours** tab, click the **Level** heading to open the **Set Levels** dialog. Set the dialog as the following image shows:

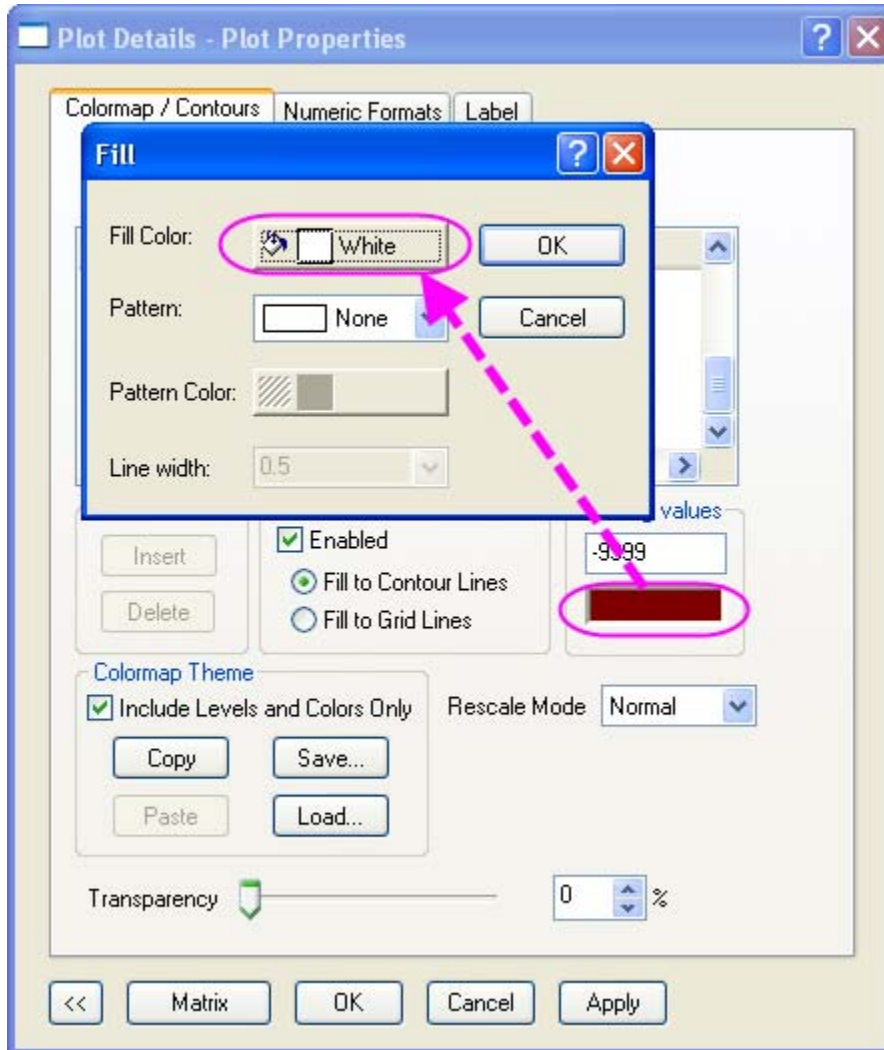


- Click the **Fill** head to open the **Fill** dialog, select **Introduce Other Colors in Mixing** radio box. Then set **From** to **Orange** and **To** to **Navy**.
- Click the cell the in the **Fill** column and **<0.5** row to set fill color to **Red**.

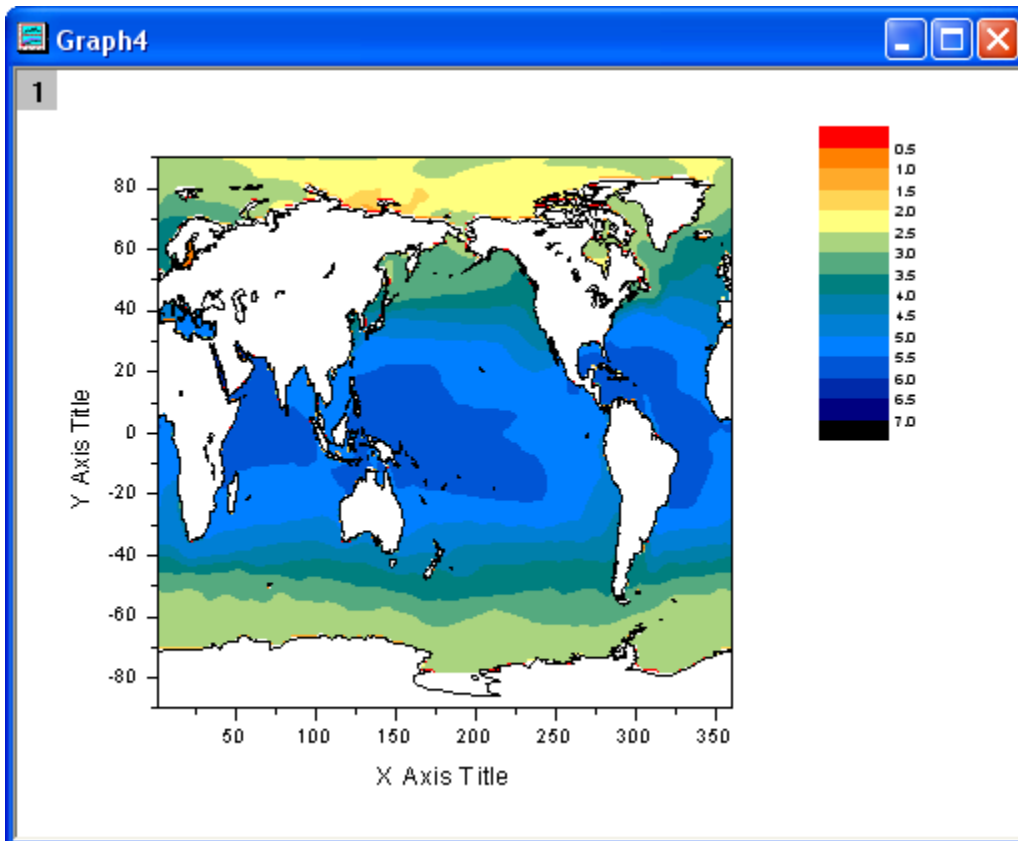


- Click the cell the in the **Fill** column and **>7** row to set fill color to **Black**.

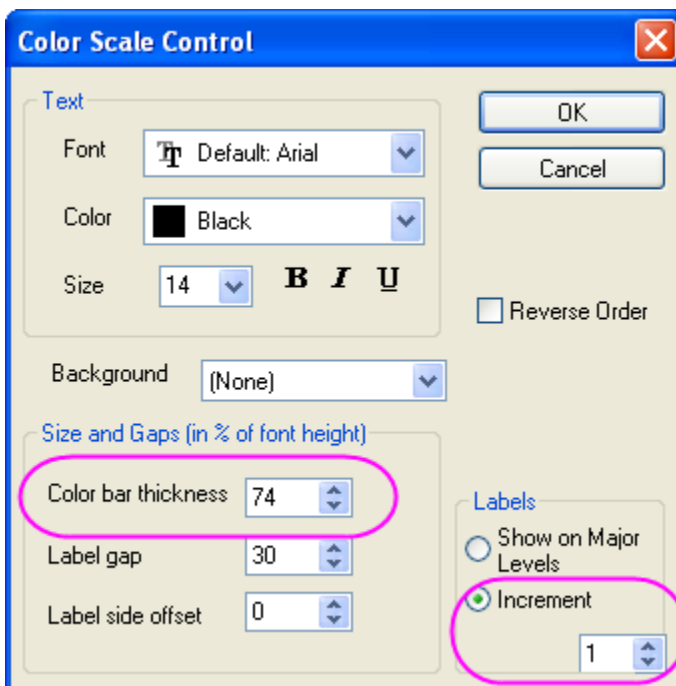
- Click the **Lines** heading to open the Contour Lines dialog, uncheck the **Show on Major Levels only** check box and then select the **Hide All** option.
- Click the **Color** box that in the **Missing Value** group, set fill color to **White**. Click OK button.



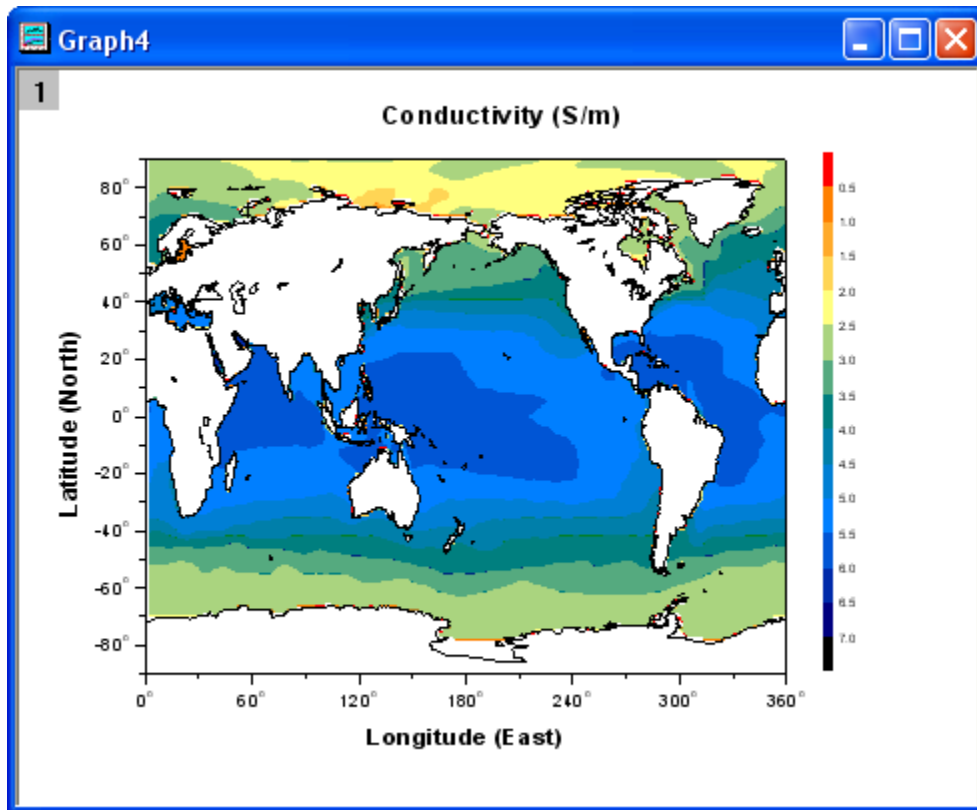
5. Go to **Numeric Formats** tab of **Plot Details** dialog, select **Decimal Places** radio box and keep the default value 1.
6. Click **Ok** button to apply the settings to the graph. The graph should look like:



Double click on the **Color Scale** to open the **Color Scale Control** dialog and set the dialog as the following graph shows:



7. Resize and reposition of the color scale. Then modify the X, Y titles and add the graph titles as the example graph shows. The graph should look like

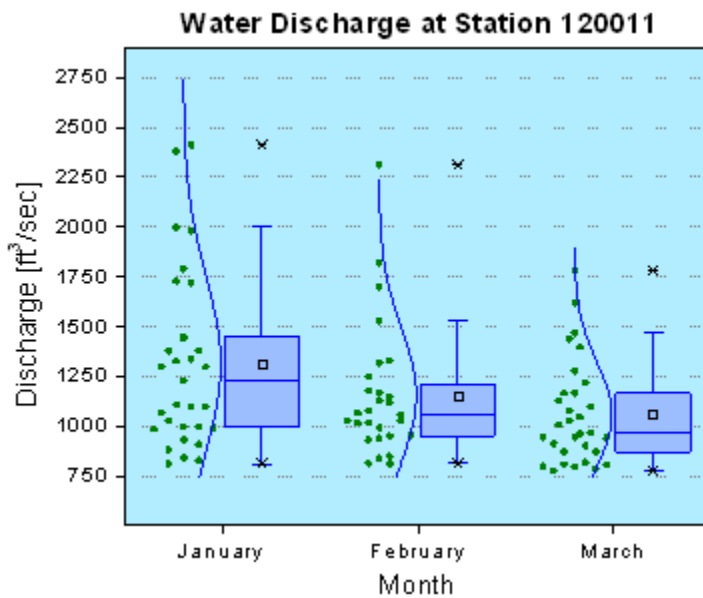


3.6 Statistical

3.6.1 Box Plot

Summary

This tutorial shows how to create the following box plot in the sample OPJ \Samples\Samples.opj.



What you will learn

- Create a data overlapped box plot
- Customize box plot settings
- Change layer background

Steps

This tutorial is associated with the sample project: \Samples\Samples.opj.

1. Open the sample project and browse to folder *Graphing: Statistical: Box Chart*.
2. Select columns January(Y), February(Y), and March(Y) and plot them to the Box Chart from **Plot: Statistics: Box Chart**.
3. Double-click on one of the box charts to bring up **Plot Details** dialog. Select **None** from **Border Color** drop-down list and click **Apply** button, so no data plot properties are set to automatically increment. Then, update the properties for the group as follows (if a property is not mentioned, leave it at its default):

Box Tab:

Type: Box[Right]+Data[Left]

4. **Data Tab:**

Curve Type: Normal

Automatic Binning: Disabled/Unchecked

Bin Size: 100

Begin: 750

End: 3600

Bin Height 100

5. **Line Tab:**

Style: Solid

Width: 2

Color: Blue

6. **Symbol Tab:**

Size: 5

Symbol Color: Olive

Shape: Circle

Interior: Solid

7. **Pattern Tab:**

Border Color: Blue

Border Style:	Solid
Border Width:	2
Fill Color:	Custom (R:156, G:190, B:254)
Pattern:	None

8. **Percentile Tab:**

Size:	5
Edge Color:	Black
Fill Color:	Black

9. In the **Plot Details** dialog, select the layer in the left panel to activate the tabs of layer level. Go to **Background** tab, set the color as a the color (R:177, G:237, B:254).
10. Double-click on the Y Axis and select the Scale tab to set axis scale as follow:

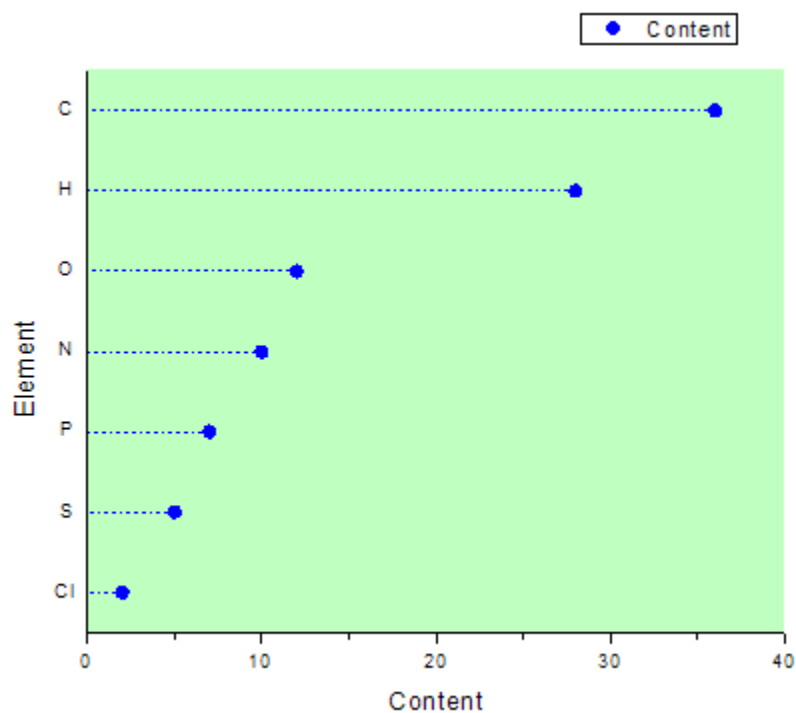
From:	510
To:	2900
Increment:	250
Num. Minor Ticks:	0

11. Activate the **Grid Lines** tab. Make sure the **Horizontal** icon is selected in the **Selection** list box so you can edit the grid lines for the Y axis. Enable the **Major Grids** check box and set the **Line Color** = *Light Gray* and **Line Type** = *Dot*.
12. Enable the **Opposite Line** check box on the **Grid Lines** tab for both the *Vertical* and *Horizontal* selections in the **Selection** List box.
13. Delete the legend and update the X and Y axis titles if desired.
14. Right-click the layer and select **Add/Modify Layer Title**, and add title as you want.

3.6.2 Simple Dot Chart

Summary

Dot chart is a statistical chart which consist of data points plotted on a simple scale. It is often used as a substitute for the pie chart, as it allows for quantities to be compared easily. This tutorial will teach you how to create a simple dot chart plot.



Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a scatter graph
- Change the X-Y Axis
- Use Plot Setup dialog to customize your graph

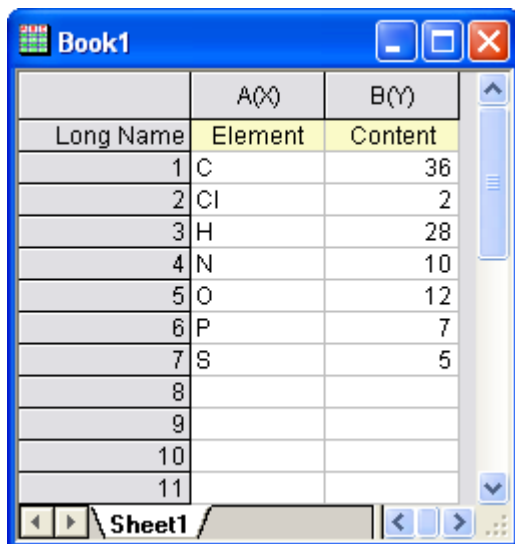
Steps

Let us start with the following data which represents various elements in a compound:

Element	Content
C	36
Cl	2
H	28
N	10
O	12

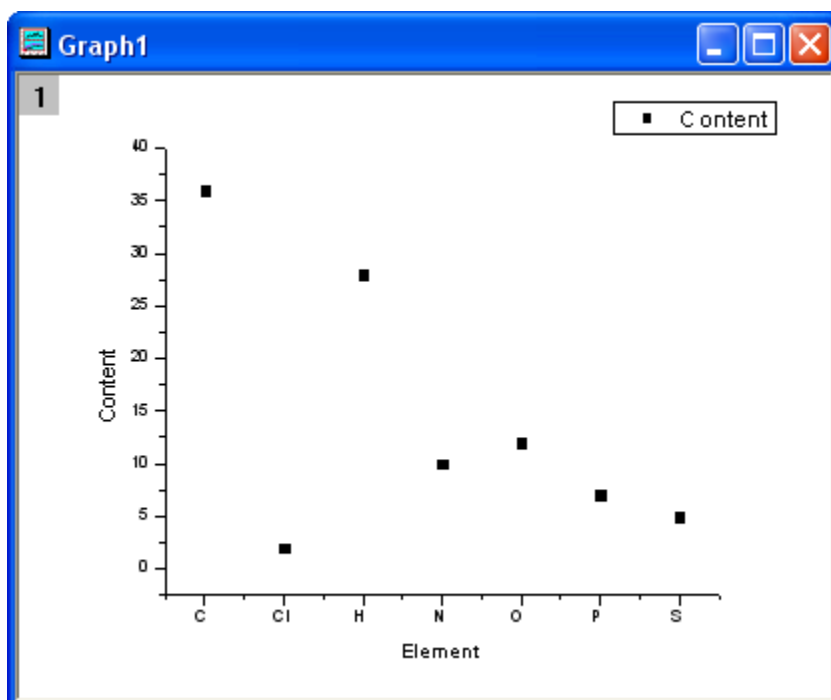
P	7
S	5

1. Create a new workbook, and input the data.

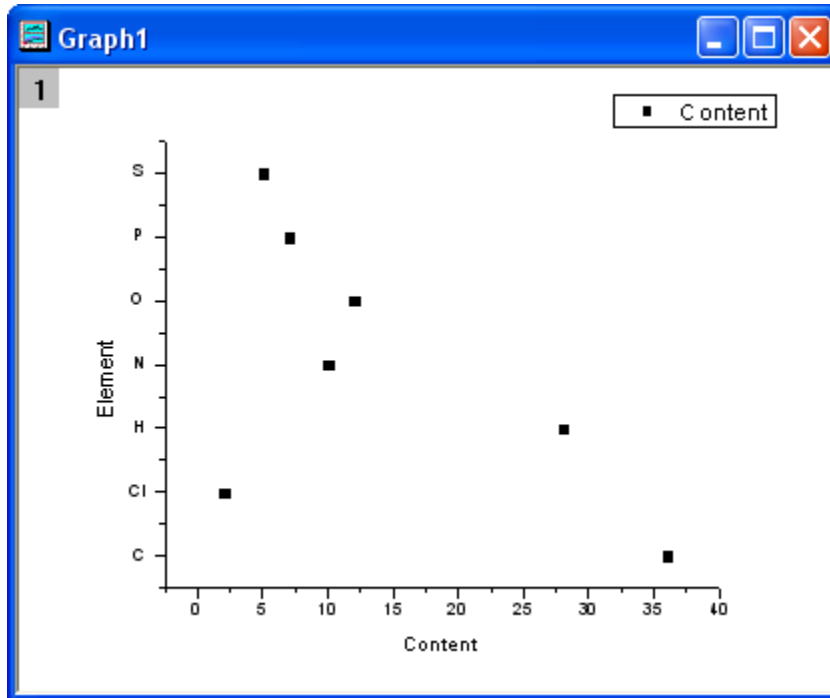


	A(X)	B(Y)
Long Name	Element	Content
1	C	36
2	Cl	2
3	H	28
4	N	10
5	O	12
6	P	7
7	S	5
8		
9		
10		
11		

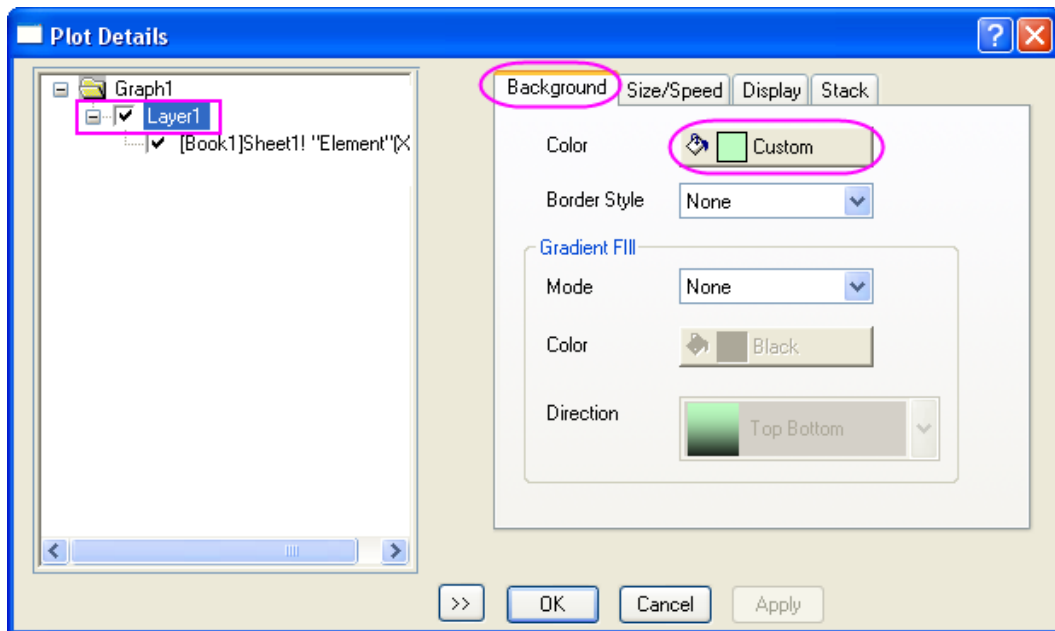
2. Highlight col(A) and col(B), and then select the **Plot: Symbol: Scatter** menu item from the Origin menu to create a scatter plot.

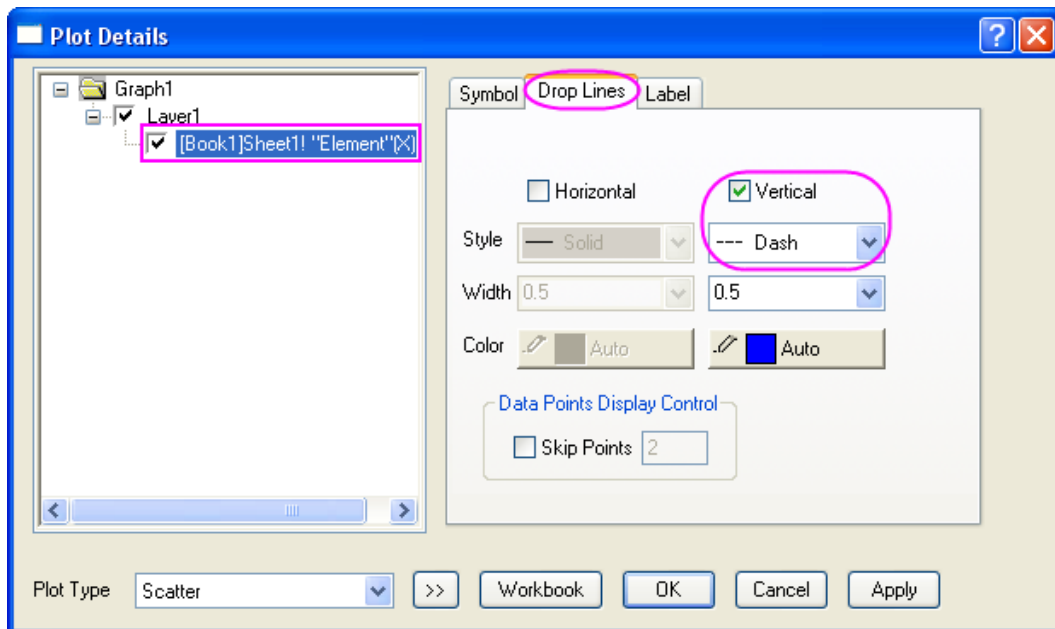
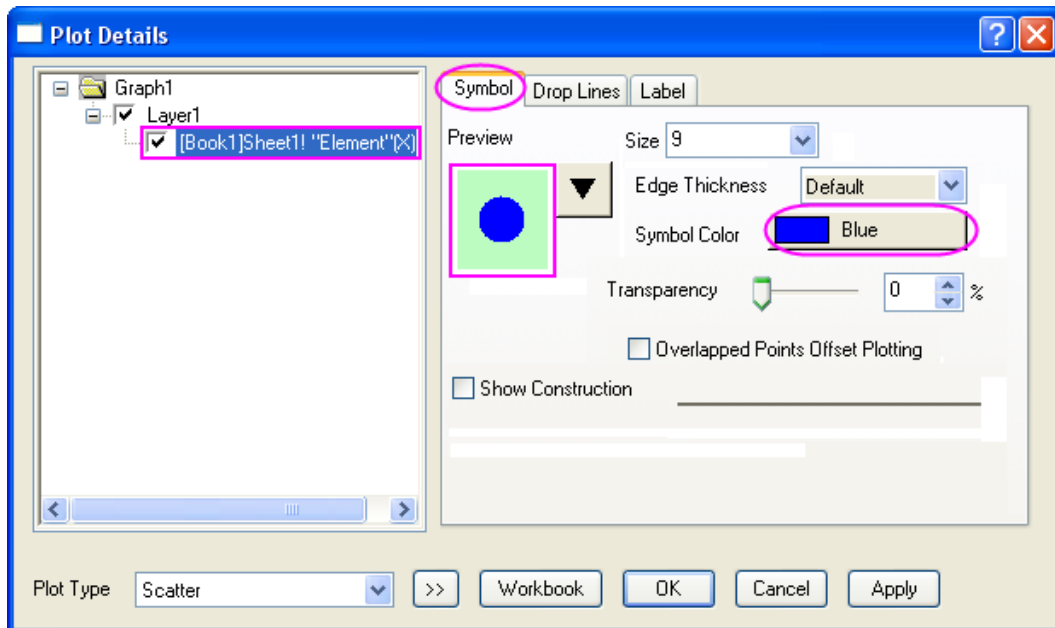


3. Select **Graph: Exchange X-Y Axis** from the menu.

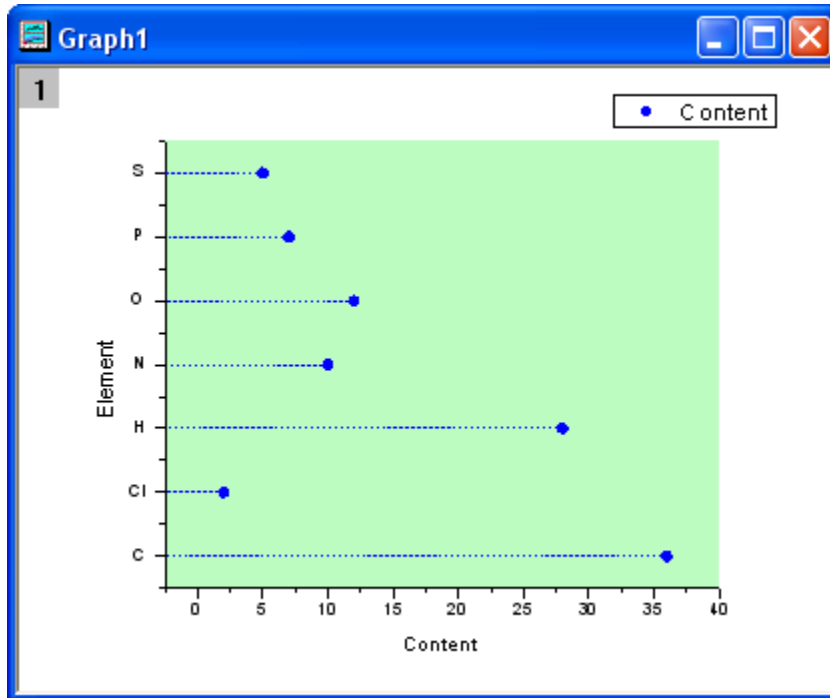


4. Double-click on the graph to bring up the **Plot Details** dialog, change the symbols and the symbol color as in the following image:

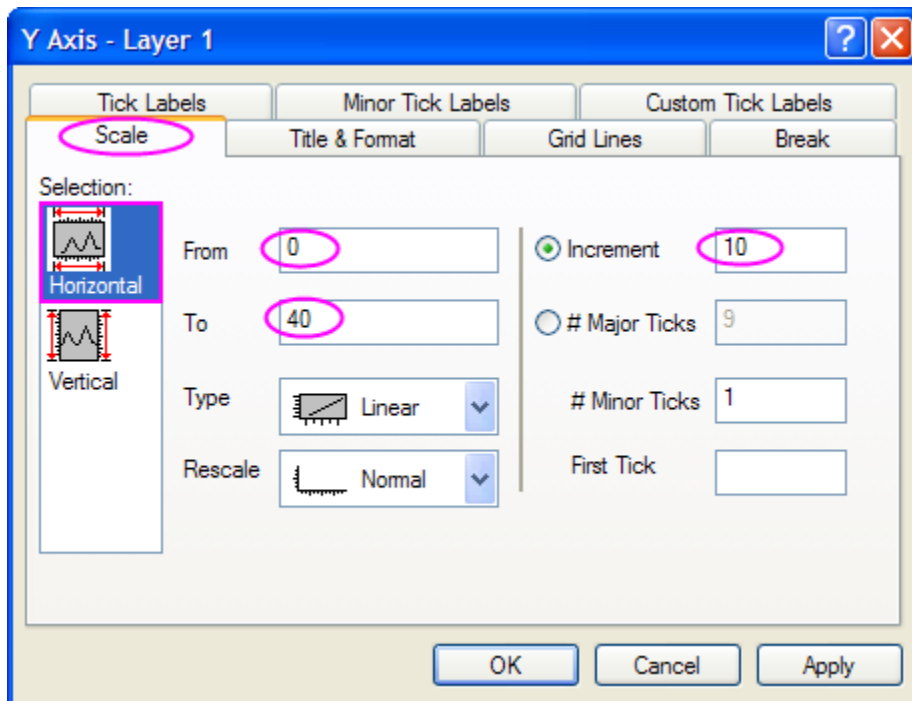




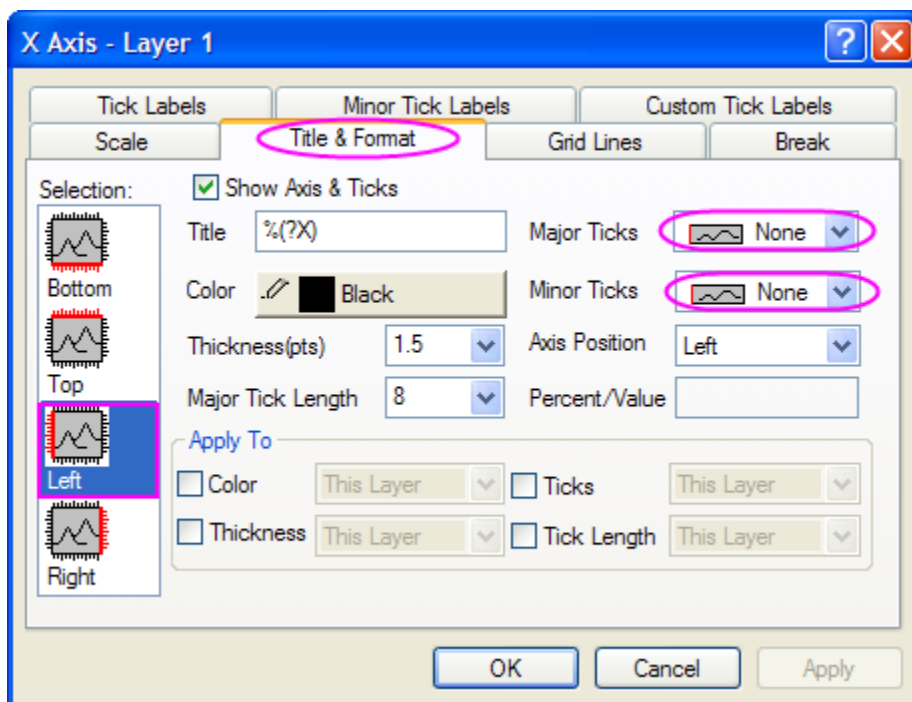
5. Click the **OK** button to close the dialog. Your graph should look like the image below:



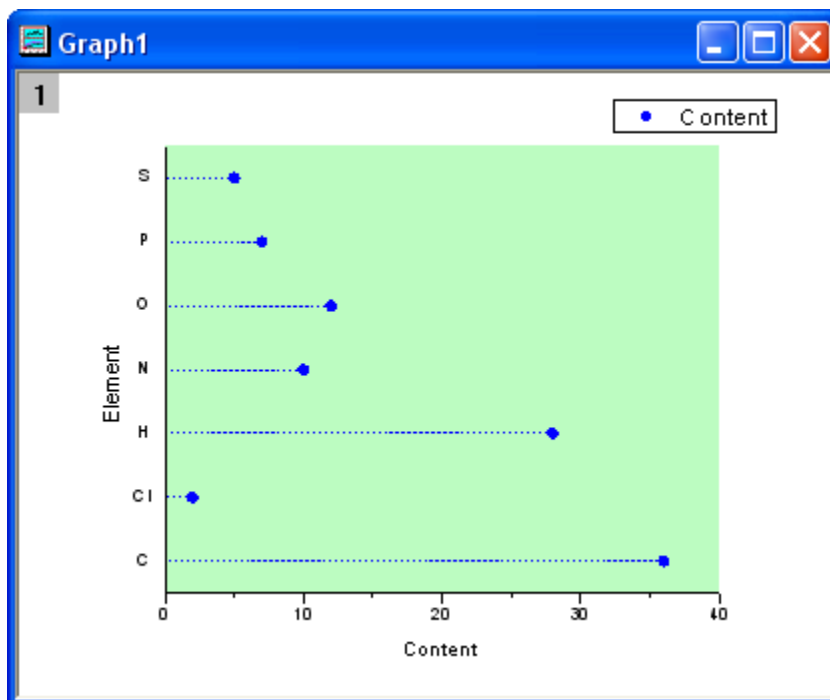
6. Let us reset the X and Y Axes. Double-click the x axis to open the **Axis Properties** dialog. In the **Scale** tab, set **From** as 0 and **To** as 40. Set the **Increment** as 10.



7. In the **Title and Format**, make sure **left** is selected in the **Selection** list, and then set **Major Ticks** and **Minor Ticks** as **None**. Finally, click **OK** button.



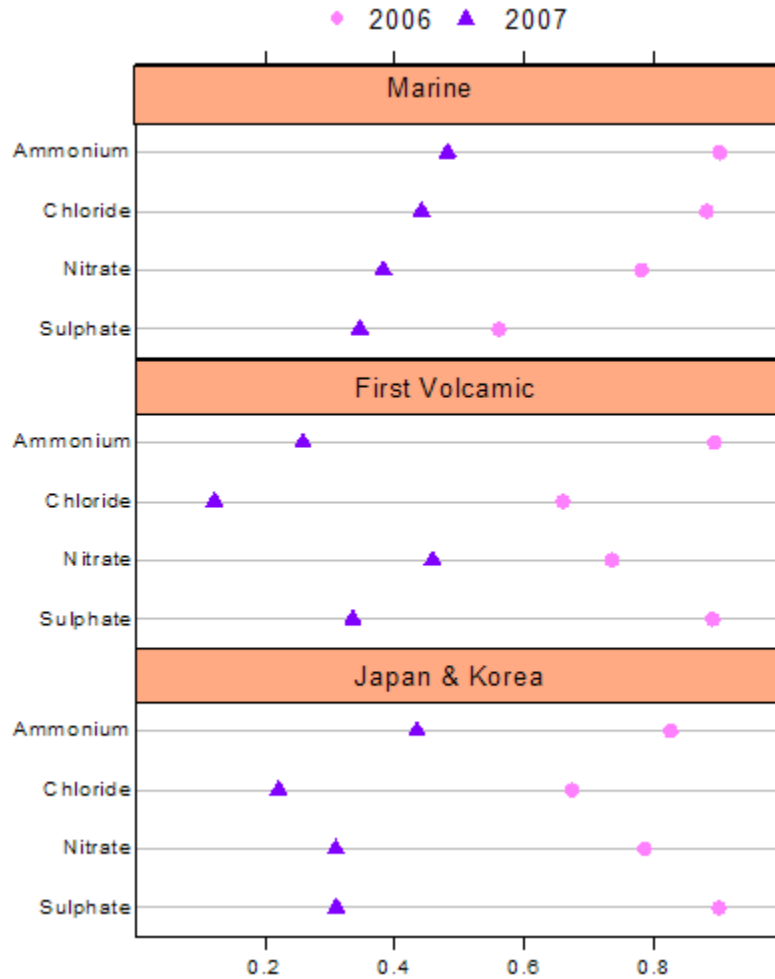
Now the Dot chart is plotted successfully and should look like below:



3.6.3 Multi-Data Dot Chart

Summary

A **Dot Chart** is a statistical chart which consists of data points plotted on a simple scale. It is often used as a substitute for the pie chart because it can make the comparing of quantities easy. This tutorial will teach you how to create the Multi-Data Dot Chart.



Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Create a scatter graph
- Change the X-Y Axis
- Use the Plot Setup dialog to customize your graph
- Use Layer Management
- Customize the axis
- Add objects on the graph

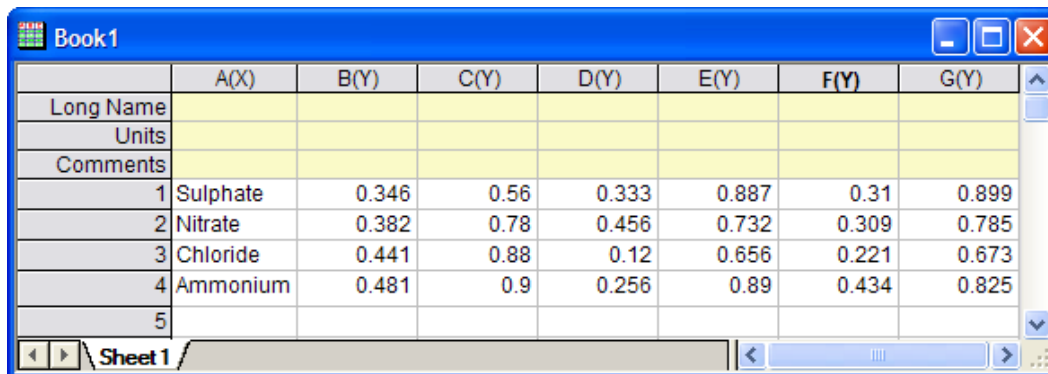
Steps

Let us learn how to create a multi-data dot chart. Here is some data about the element content of several areas in different time. We can use it to create a dot chart.

Sulphate	0.346	0.560	0.333	0.887	0.310	0.899
Nitrate	0.382	0.780	0.456	0.732	0.456	0.732
Chloride	0.441	0.880	0.120	0.656	0.221	0.673
Ammonium	0.481	0.900	0.256	0.890	0.434	0.825

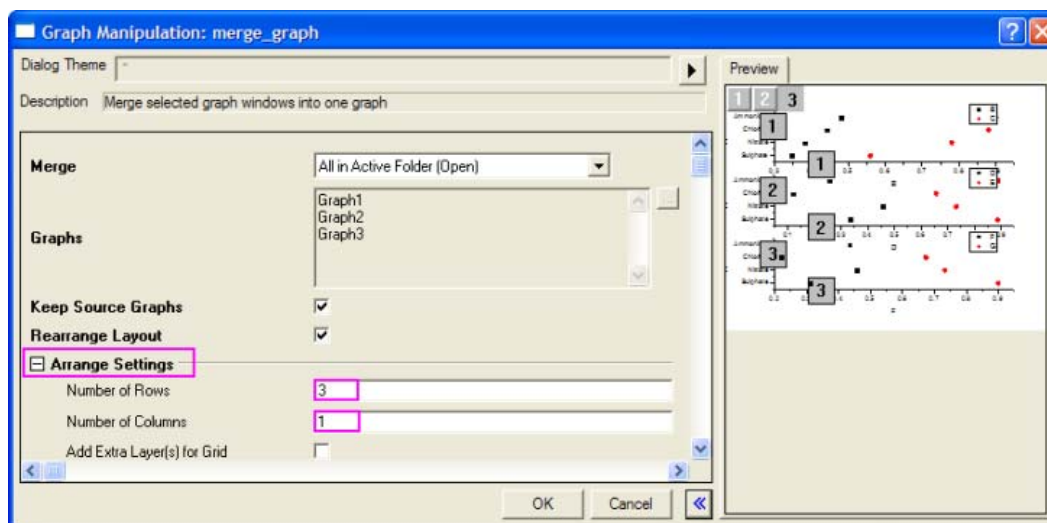
Now, let us begin.

1. Create a new workbook and input the data.

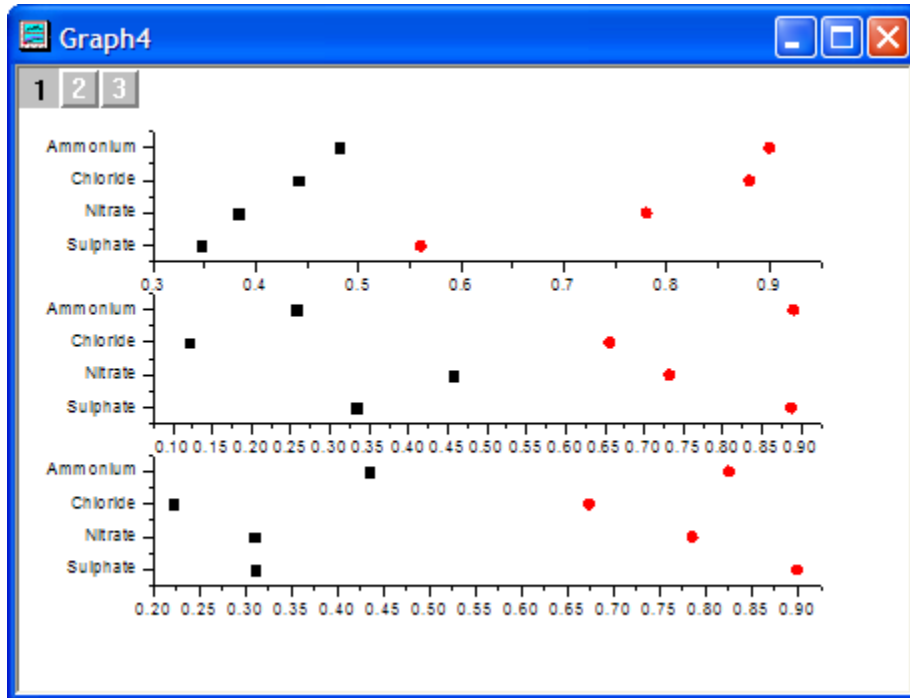


	A(X)	B(Y)	C(Y)	D(Y)	E(Y)	F(Y)	G(Y)
Long Name							
Units							
Comments							
1	Sulphate	0.346	0.56	0.333	0.887	0.31	0.899
2	Nitrate	0.382	0.78	0.456	0.732	0.309	0.785
3	Chloride	0.441	0.88	0.12	0.656	0.221	0.673
4	Ammonium	0.481	0.9	0.256	0.89	0.434	0.825
5							

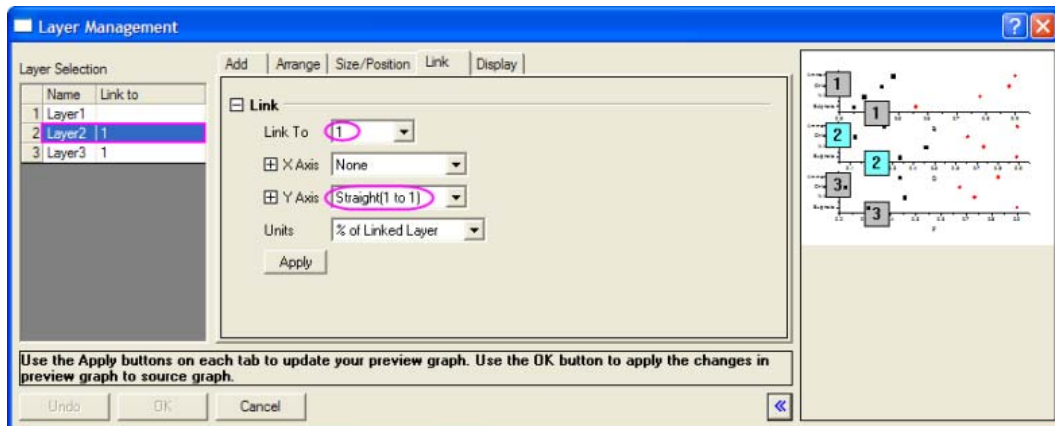
2. Highlight col(B) and col(C), select **Plot: Symbol: Scatter** in the main menu to draw a graph, then select **Graph: Exchange X-Y Axis**.
3. Repeat step 2 to create one graph with col(D) and col(E), and another graph with col(F) and col(G).
4. Merge these three graphs. Select **Graph: Merge Graph Windows: Open dialog**. Expand **Arrange Settings**, set the **Number of Rows** as 3 and **Number of Columns** as 1. Click the **OK** button. Now you get a new graph which contains three layers.



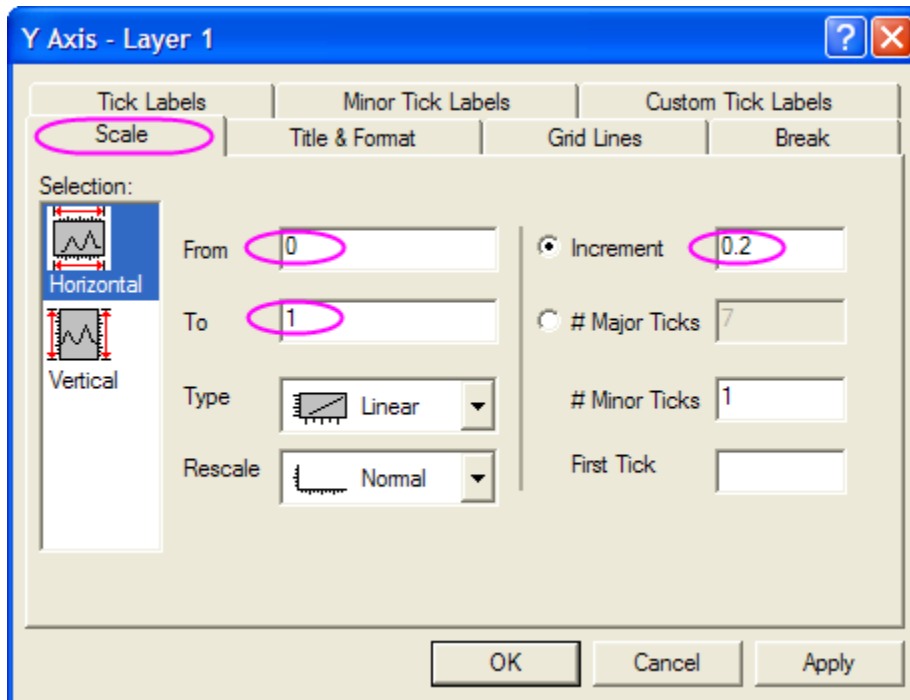
5. You can delete the legend and the XY axis labels in the graphs to clean up the graph.



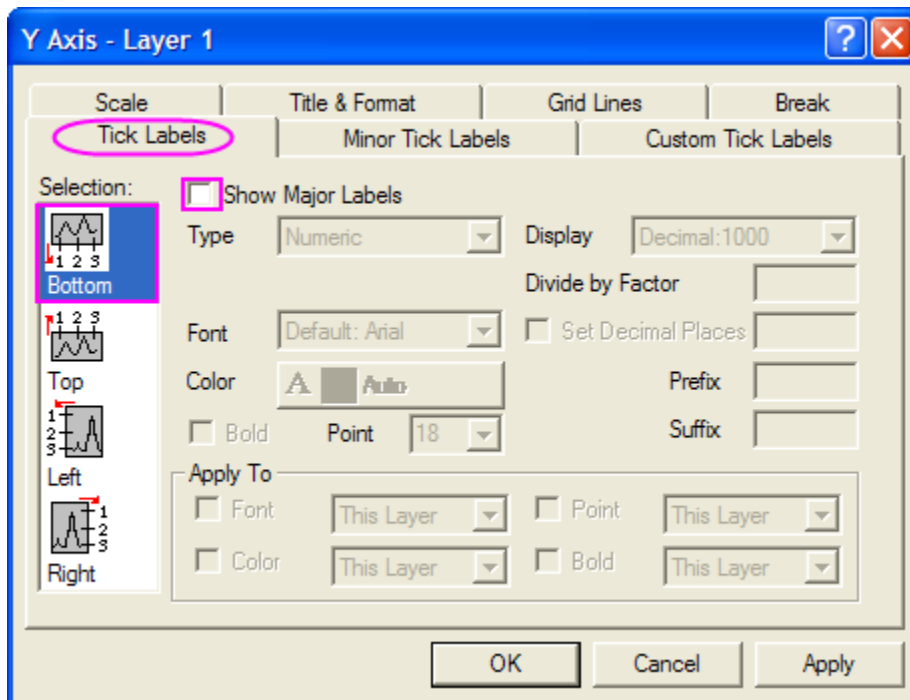
6. Select **Graph: Layer Management** from the main menu. Select Layer 2 on the left. Then on the right panel, make sure that the **Link** tab is active. In this tab, set **Link to** as **1** and **Y Axis** as **Straight (1 to 1)**. Click the **Apply** button.



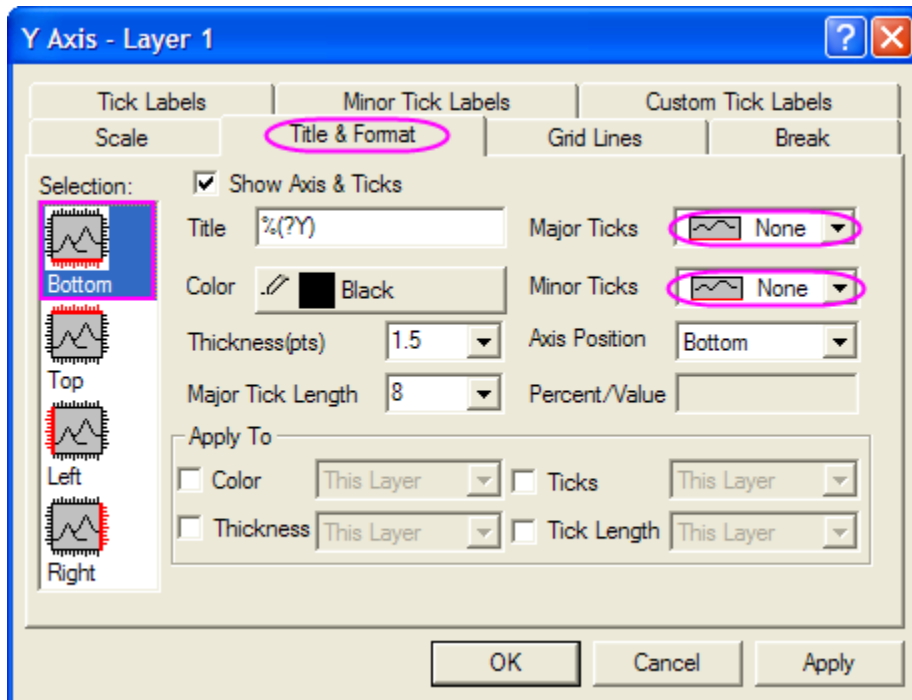
7. Select Layer 3 on the left. Also in the **Link** tab, set **Link to** as **1** and **Y Axis** as **Straight (1 to 1)**. Then click the **OK** button.
8. Reset X and Y Axes. Make sure **Layer 1** is active, double-click the X axis of Layer 1. Set the scale of X **From 0 To 1**, and the **Increment** as **0.2**.



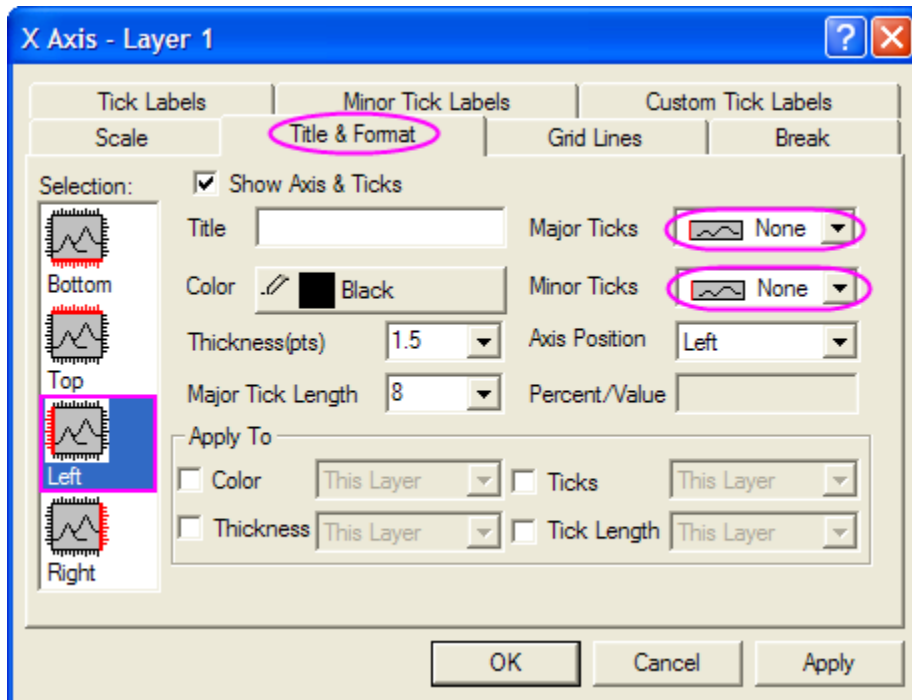
9. In the **Tick Labels** tab, uncheck the **Show Major Label** box.



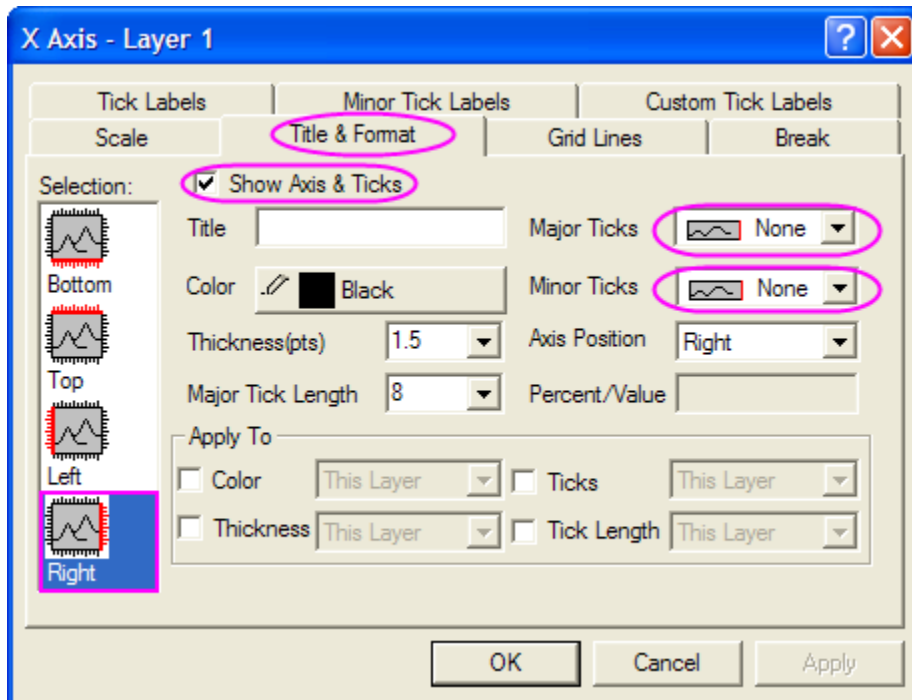
10. In the **Title and Format** tab, set **Major Ticks** and **Minor Ticks** as **None**.



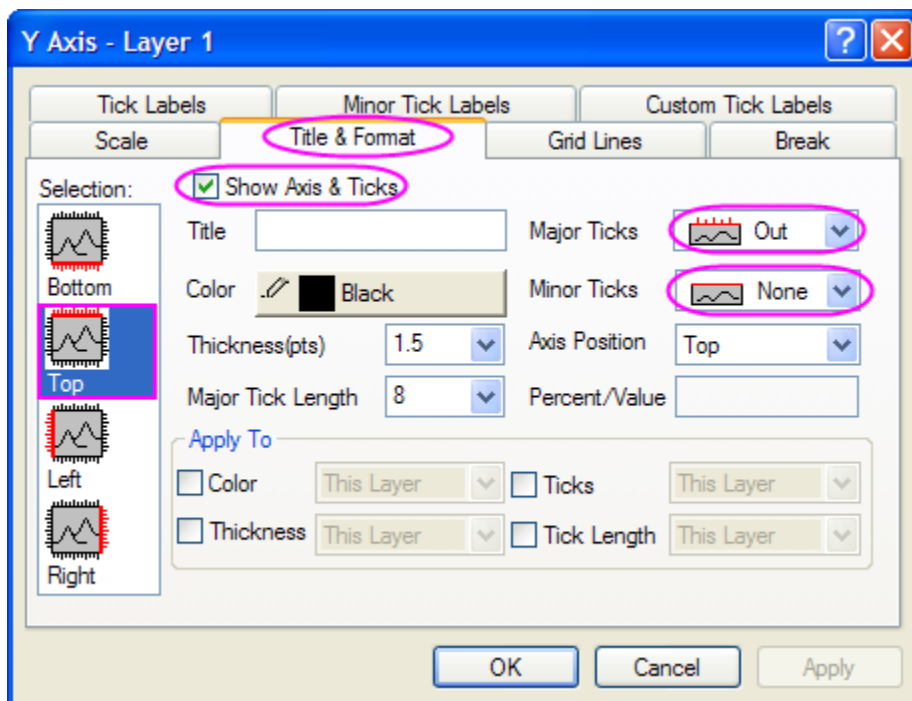
11. Click **Left** in the **Selection** list, select the **Show Axis & Tick** check box and set **Major Ticks** and **Minor Ticks** as **None**.



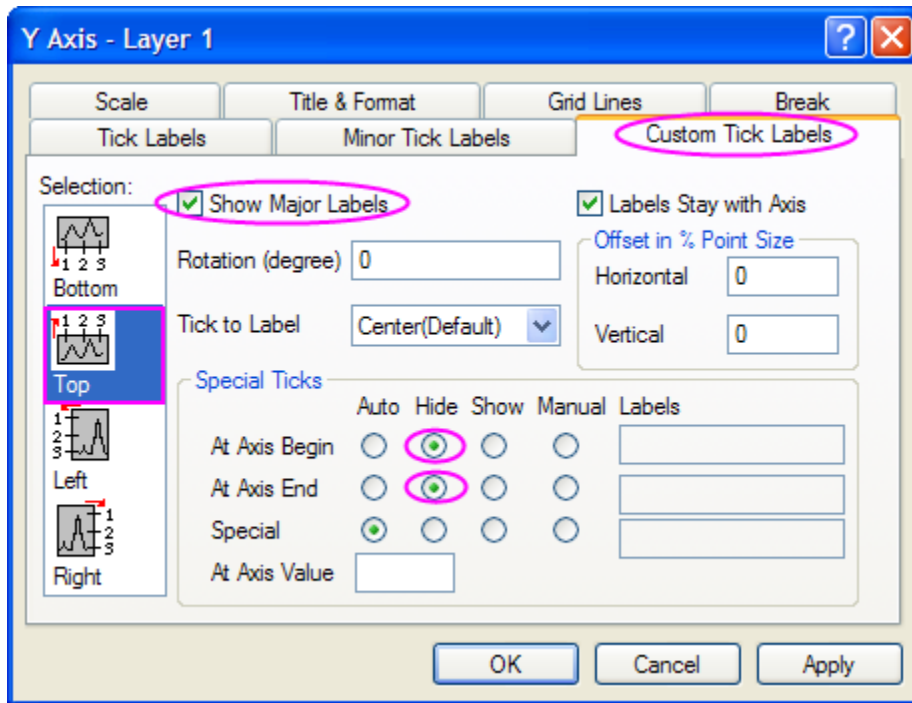
12. Click **Right** in the **Selection** list, select the **Show Axis & Tick** check box and set **Major Ticks** and **Minor Ticks** as **None**.



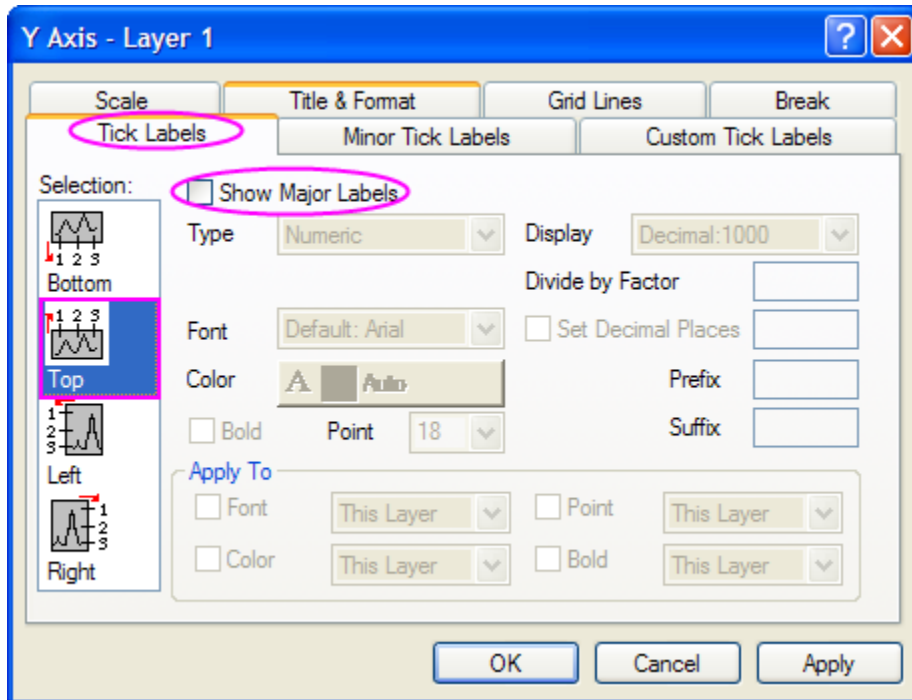
13. Select **Top** in the **Selection** list, check the **Show Axis & Tick** box. set **Major** as **Out** and **Minor** as **None**.



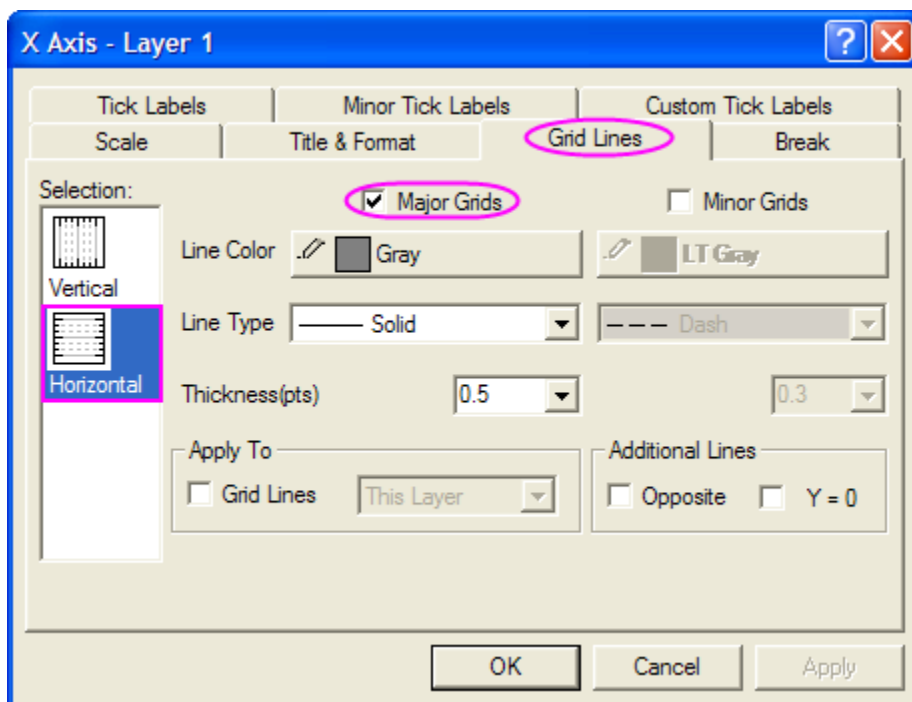
14. Select the **Custom Tick Labels** tab, highlight **Top** in the Selection, choose the **Hide** radio button both with **At Axis Begin** and **At Axis End**.



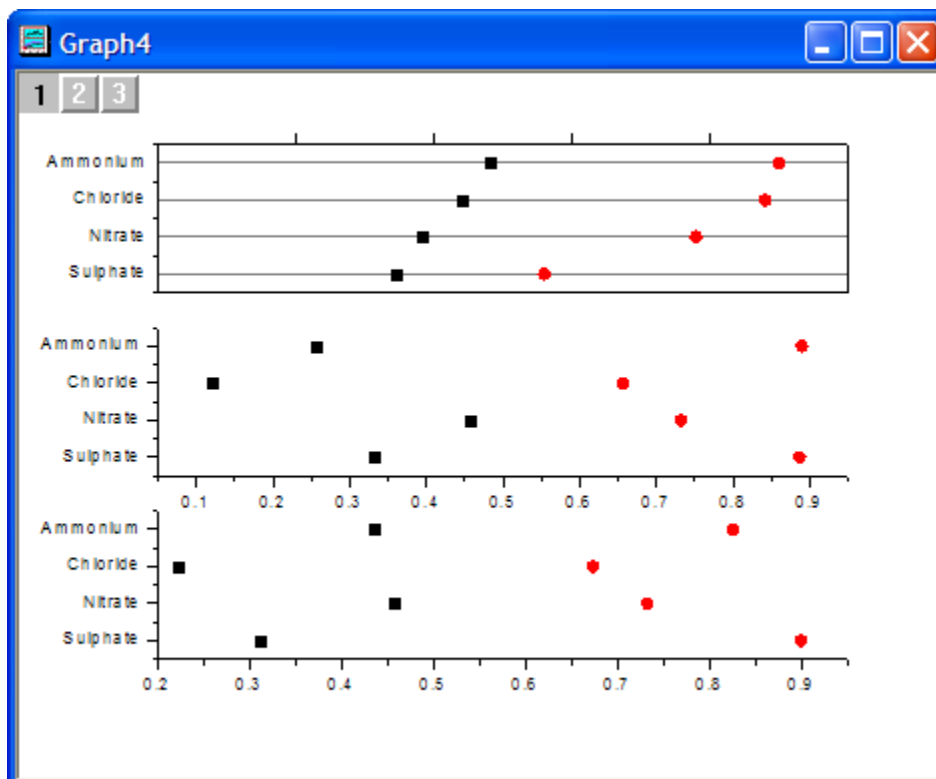
15. In the **Tick Labels** tab, uncheck the **Show Major Label** box.



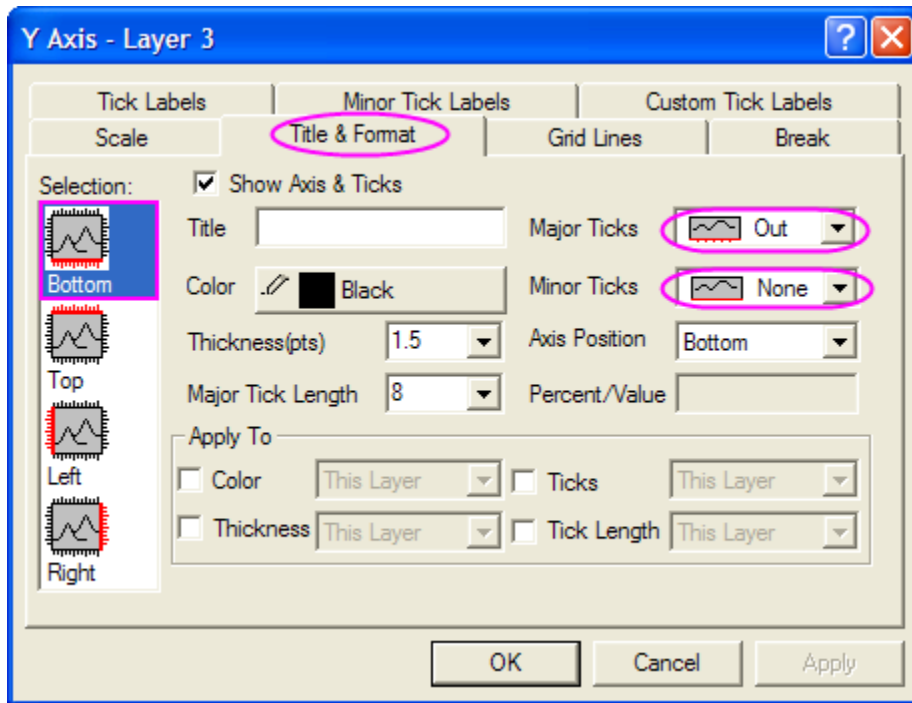
16. In the **Grid Lines** tab, select **Horizontal** in the **Selection** list and then select the **Major Grid** checkbox. Also, choose the color and the style for the grid lines. Click the **OK** button.



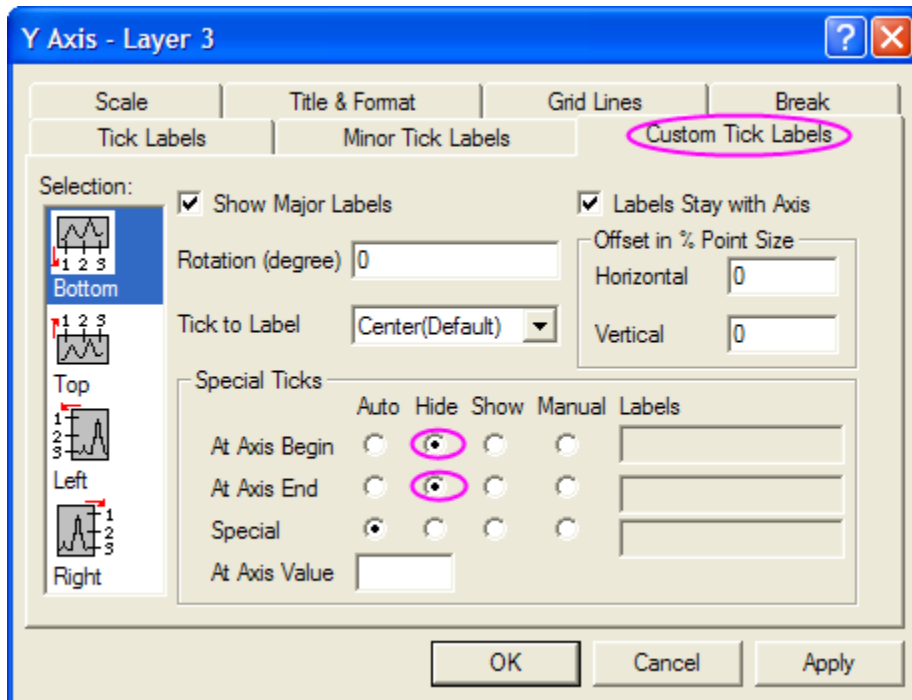
17. Activate Layer 2, repeat the steps from 9 to 12 and step 16.



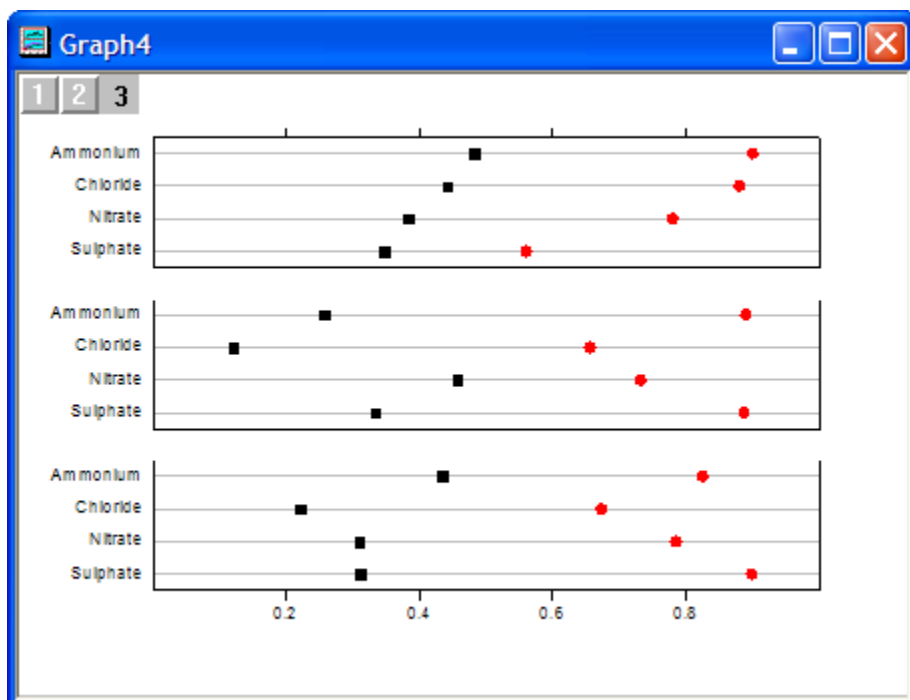
18. Activate Layer 3, also repeat steps from 9 to 12 and step 16 except step 10. In the **Title and Format** tab, set **Major Ticks** as **out** and **Minor Ticks** as **None**.



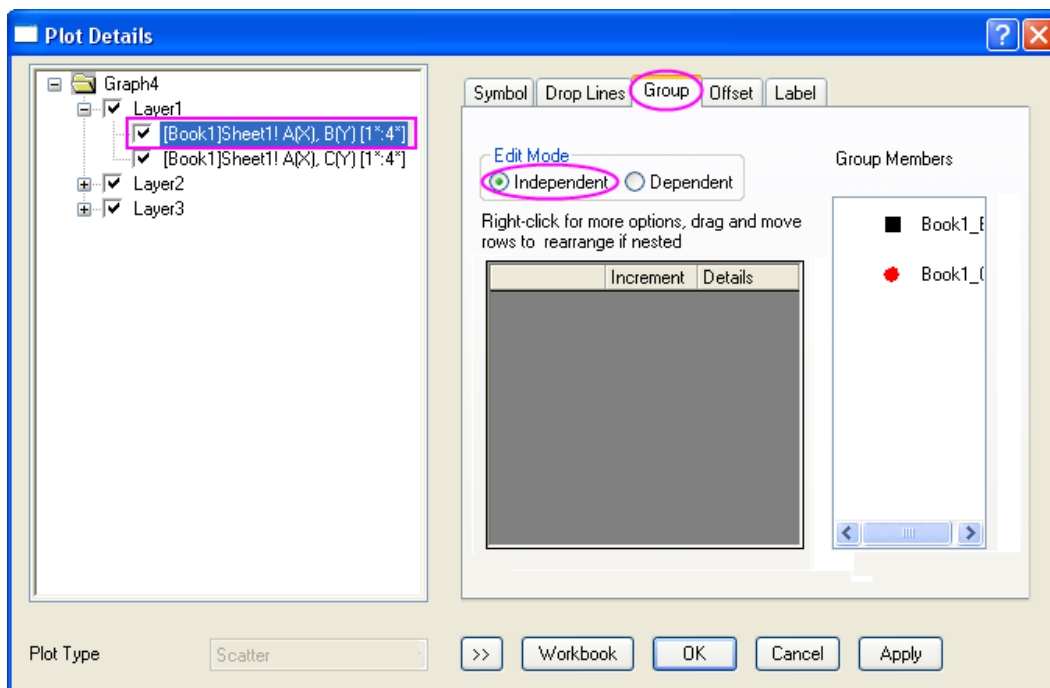
19. Select the **Custom Tick Labels** tab, highlight **Bottom** in the **Selection**, choose the **Hide** radio button both with **At Axis Begin** and **At Axis End**.

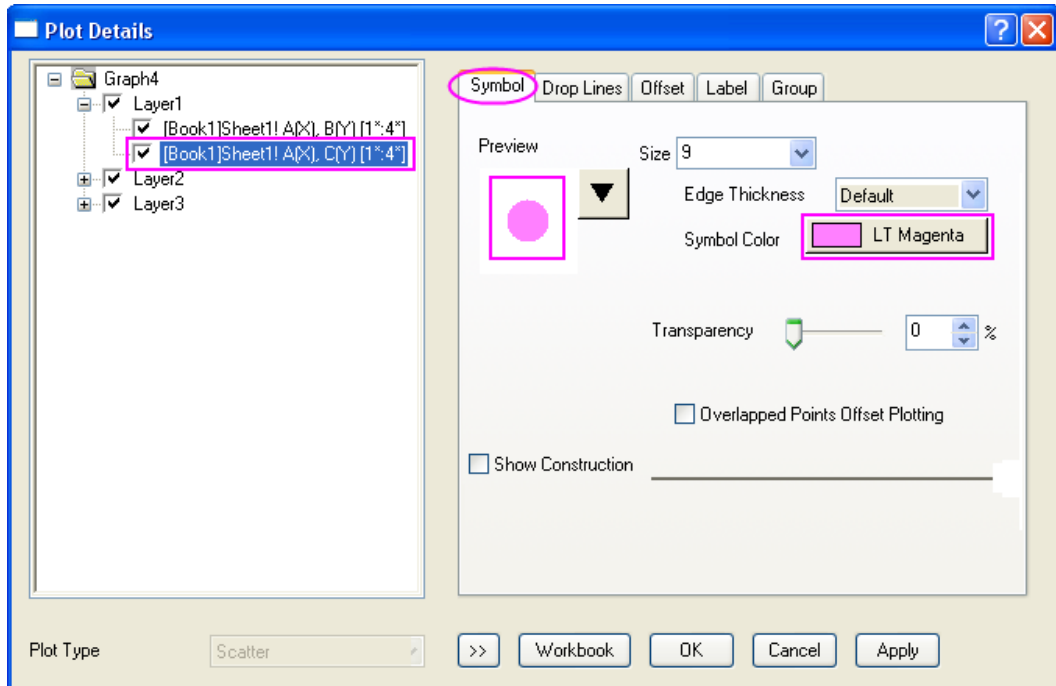
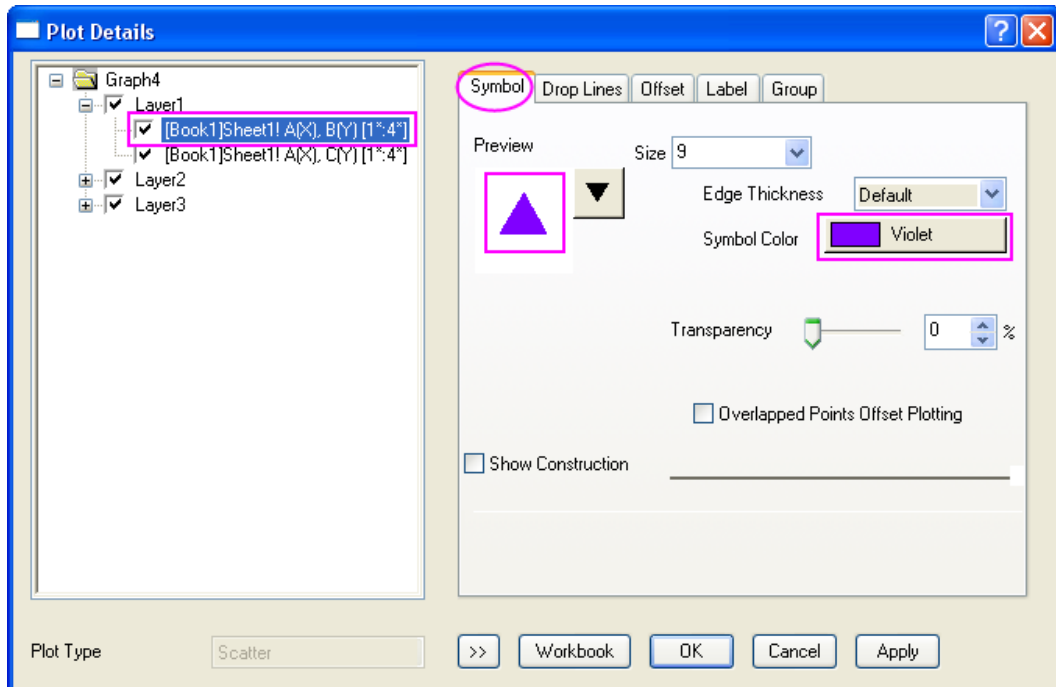


20. Click **OK**. Now you will see the graph below.

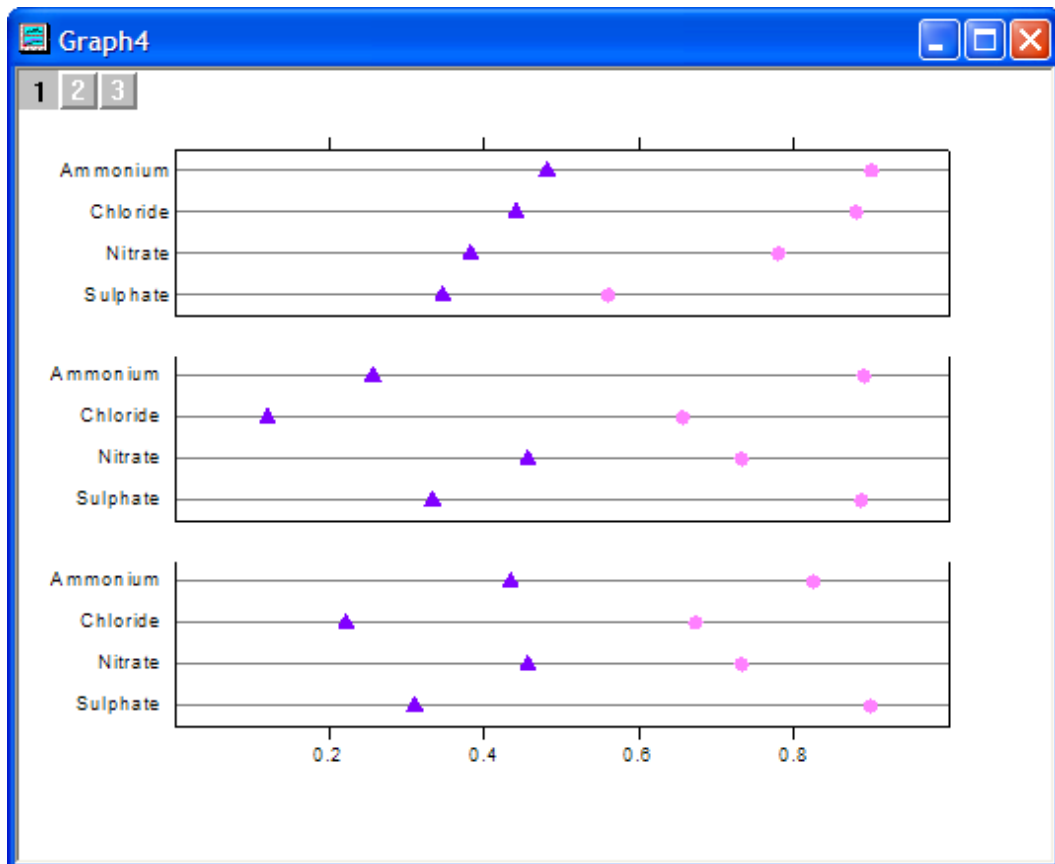



21. Double-click the graph to bring up the **Plot Details** dialog. Change the options as the following screenshot.



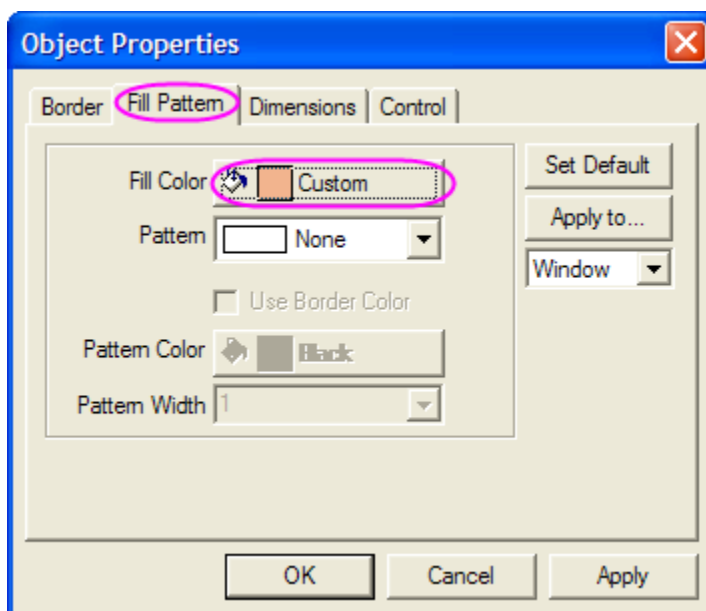


22. Repeat the steps of 21 for the Layer 2 and Layer 3. Click the **OK** button and you will see this graph.

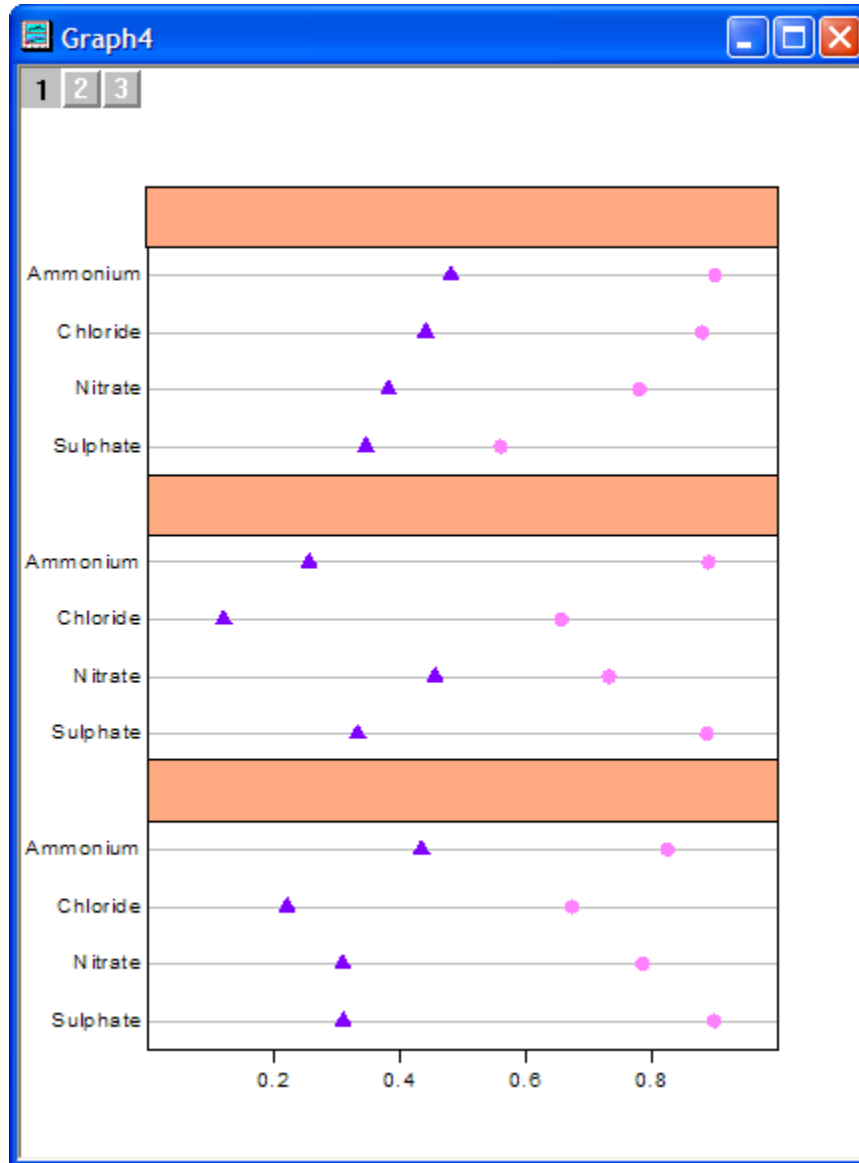


23. Select  in the **Tools** toolbar to draw three rectangles on the graphs. For each rectangle:

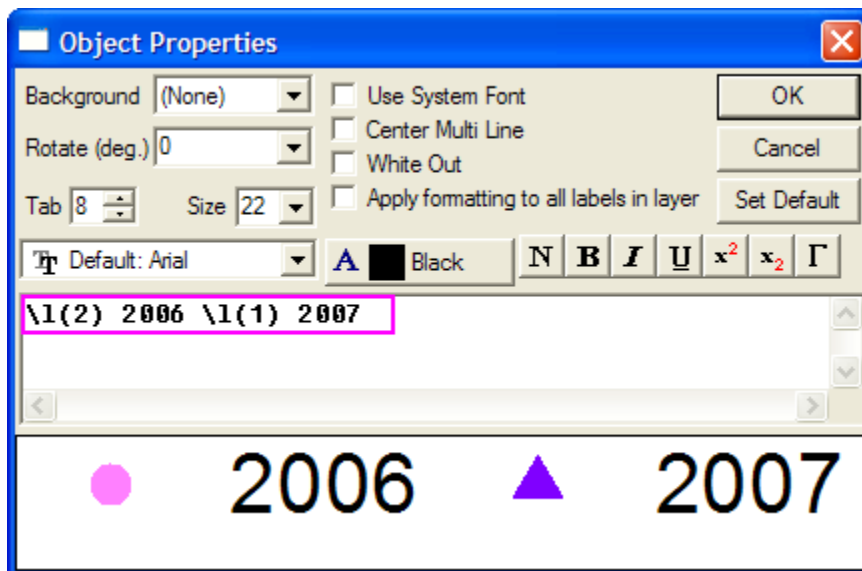
- Double-click on the rectangle to bring up the **Object Properties** dialog.
- In the **Fill Pattern** tab, set the desired **Fill Color**.



- In the **Dimensions** tab, adjust the size and position of the rectangle if so desired.

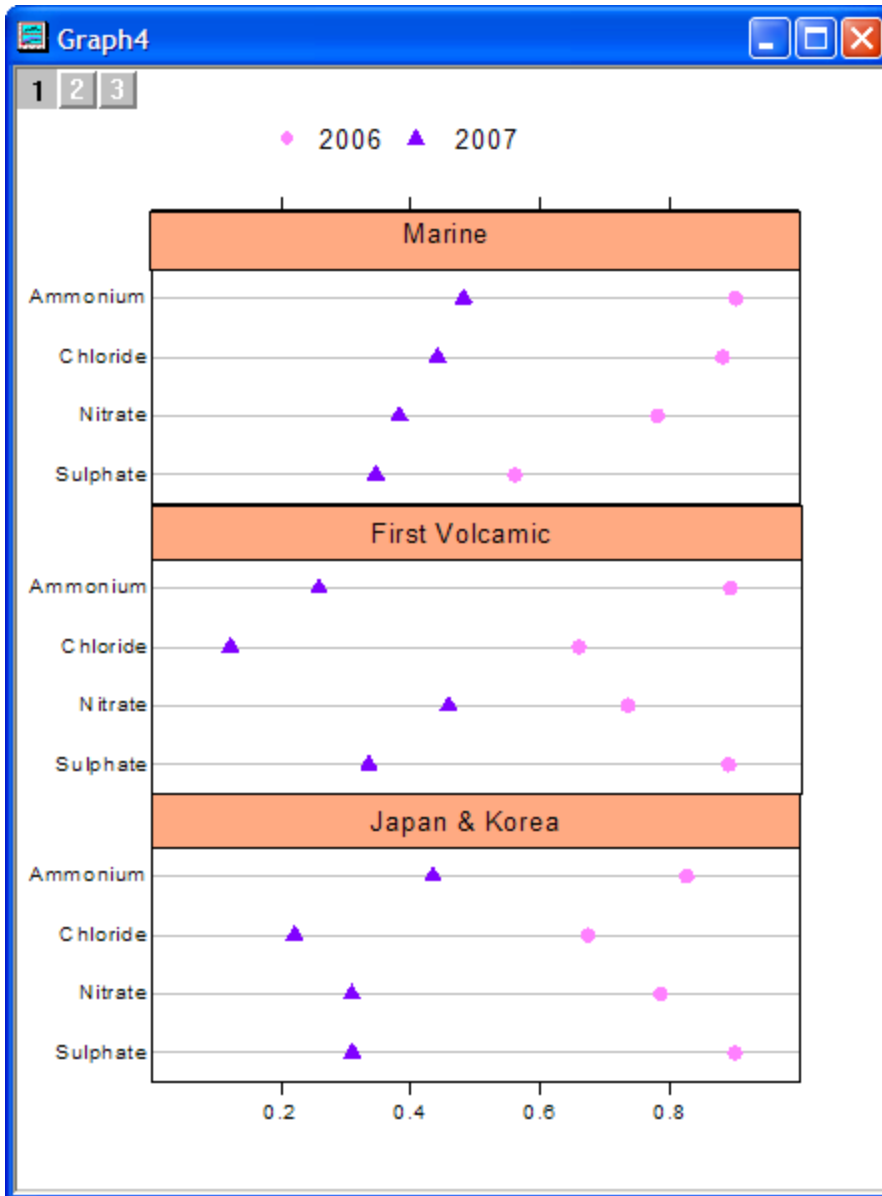


24. Select the Text Tool **T** in the Tools toolbar and click inside the rectangles to add the text that you want.
25. Select **Graph: New Legend** from the main menu. Move the legend to a suitable place, then right-click on it and select **Properties**. Change the settings as below:



26. Activate the top x axis of Layer 1 to move it to a suitable place.

Now the multi-data dot chart is finished. You can see the graph below.



3.6.4 Add multiple fitted curves in a Histogram

Summary

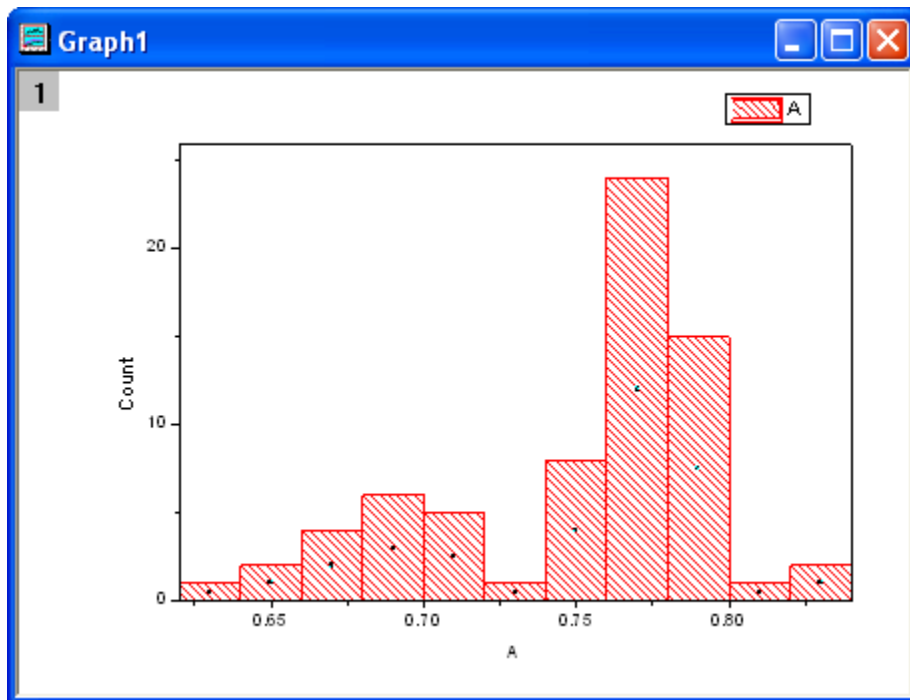
After you plot a Histogram, Origin allows you to overlay a distribution curve on the binned data by selecting **Normal**, **Lognormal**, **Poisson**, **Exponential**, **Laplace**, or **Lorentz** from the **Type** drop-down list in the Data tab of the Plot Details dialog. If you want to add multiple distribution curves to the Histogram, the procedure involves a few more steps.

What you will learn

- Plotting the Histogram
- Using Frequency Count to do statistics
- Using Peak Analyzer to find peaks and do fitting
- Adding new layers

Steps

Copy and paste the sample data into Origin and set the column as Y (Highlight the column and choose **Column: Set As Y** from the Origin menu). Plot this data as a Histogram by clicking **Plot: Statistics: Histogram** from the menu.



Frequency Count

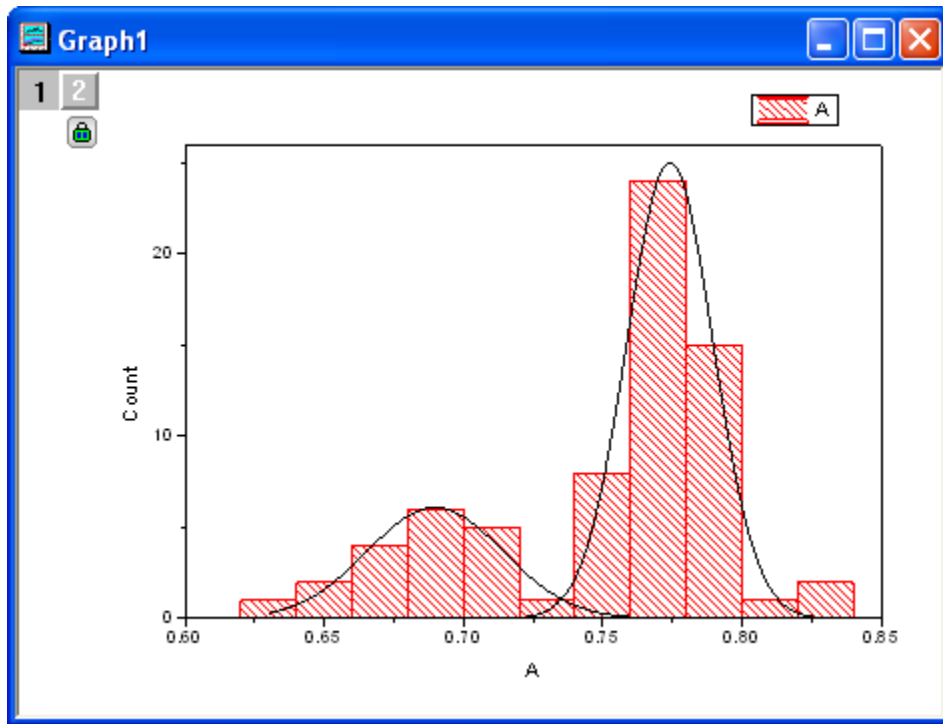
1. Highlight the sample data, then open the **Frequency Count** dialog by selecting **Statistics: Descriptive Statistics: Frequency Count**.
2. Click **OK** to finish. A new result sheet will be generated.

Fit peaks

1. Select Col(Counts) and then open the **Peak Analyzer** dialog from **Analysis: Peak and Baseline: Peak Analyzer**.
2. In the start page, select **Fit Peaks** as **Goal**, then click **Next**.
3. In the **Baseline Mode** page, set the baseline as **Y=0** when the **Custom** radio box is checked.
4. Click the **Next** button twice to go to the **Find Peaks** page. Click the **Find** button under the **Enable Auto Find** check box to find two peaks.
5. Click the **Next** button again to open the **Fit Peaks** page. Click **Fit Control** at the bottom of this page to open the **Peak Fit Parameters** dialog.
6. In this dialog, the default fitting function is **Gaussian**, which is the right function for normalizing the data. Close the **Peak Fit Parameters** dialog and go back to the **Peak Analyzer** dialog. Click **Finish** to complete the fitting.

Add the fitted curves

1. Activate the Histogram graph and add a layer by selecting **Graph: New Layer(Axes): Right-Y** from the main menu.
2. Right-click the **Layer2** icon and select **Plot Setup** from the short menu to open the **Plot Setup** dialog.
3. Select the sheet **FitPeakCurve1** from the top panel, then set col(A3) as X and col(A4) as Y, and add them into the Layer **RightY** in the bottom panel.
4. Do the same things for col(A5) and col(A6). After that, both fitted peaks have been added into the Layer **RightY**.
5. Click OK. Two fitted curves had been added to the Histogram.
6. Double-click the graph to open the **Plot Details** dialog. Select **RightY** from the left panel, then open the **Link Axes Scales** tab in the right panel and select **Straight(1 to 1)** for both **X Axis Link** and **Y axis Link**. Click OK to close the dialog.
7. The fitted curves are added into the Histogram with the proper scale. The following is the result graph, with the right Y-axis removed.



Sample Data

0.631
0.642
0.652
0.662
0.669
0.676
0.677
0.69
0.691
0.696
0.697
0.699
0.699
0.7
0.7
0.708
0.712
0.718
0.731
0.744
0.749
0.751

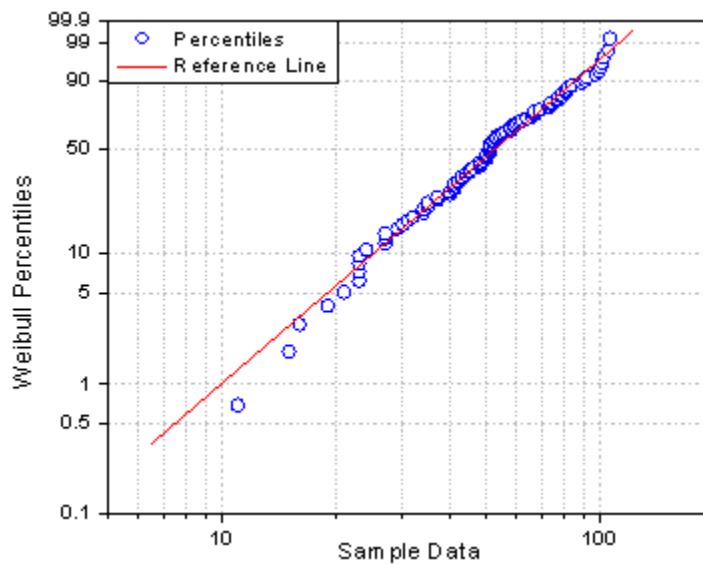
0.752
0.753
0.758
0.758
0.759
0.761
0.761
0.763
0.763
0.763
0.765
0.767
0.768
0.768
0.769
0.769
0.77
0.771
0.771
0.772
0.774
0.775
0.775
0.776
0.776
0.776
0.777
0.778
0.779
0.78
0.78
0.781
0.784
0.784
0.785
0.785
0.789
0.789
0.791
0.794
0.795
0.796
0.798

0.798
0.803
0.82
0.831

3.6.5 Weibull Probability Plot

Summary

The probability plot shows a graph with observed cumulative percentage on X axis and expected cumulative percentage on Y axis. The Weibull probability plot is used to test whether or not a dataset follows Weibull distribution. And its X scale type and Y scale type are Log10 and Double Log Reciprocal respectively. If all the scatter points are close to the reference line, we can say that the dataset follows the Weibull distribution.



Minimum Origin Version Required: Origin 8.1 SR0

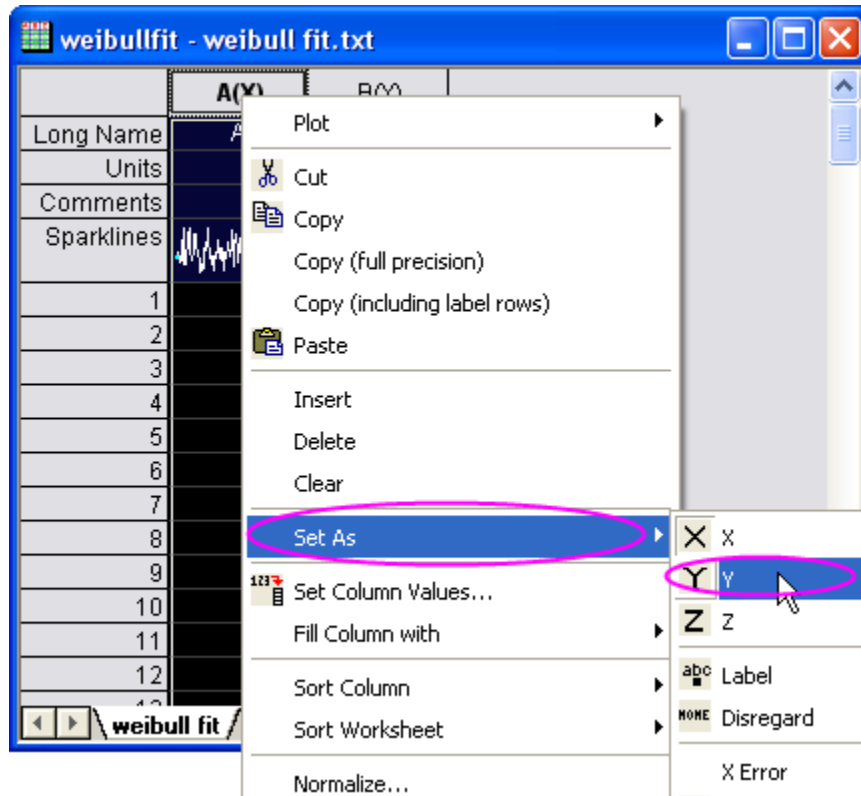
What you will learn

- How to create a Weibull probability plot
- How to change the shape of the symbol
- How to change the width of the line

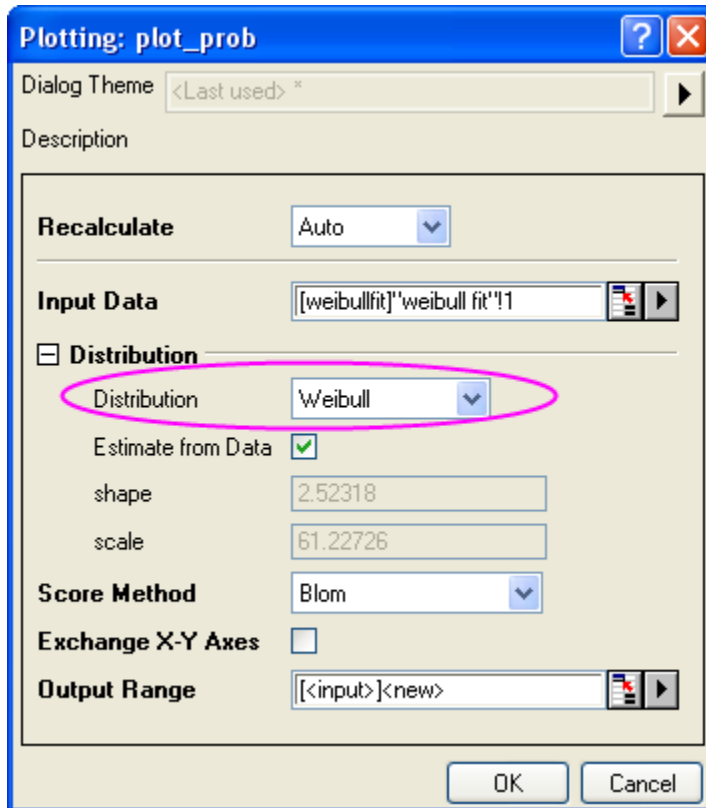
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`. (If you don't have the OPJ, you can also download the data from here)

1. Open the project, browse to the folder *Graphing: Statistical: Probability, QQ Plot*. Active the worksheet *Probability Plot Data*. Highlight column A and right-click to select **Set As: Y** from the context menu.



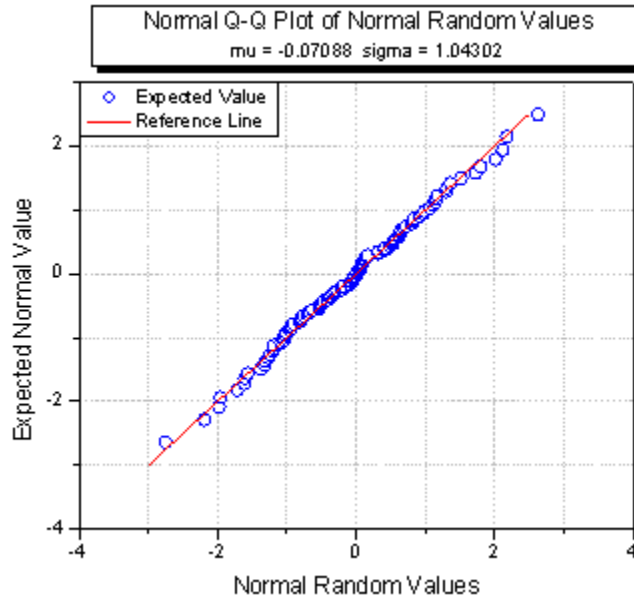
- Keep this column highlighted, select **Plot: Statistics: Probability Plot** from the Origin main menu to open the **plot_prob** dialog. In this dialog, change **Distribution** to **Weibull** and then click the **OK** button to create a Weibull probability plot with column A.



3.6.6 Q-Q Plot

Summary

Q-Q plot is a graphic method to test whether or not a dataset follows a given distribution. In a Q-Q plot, observed values on X axis and expected values on Y axis are shown. If all the scatter points are close to the reference line, we can conclude that the dataset follows the given distribution.



Minimum Origin Version Required: Origin 8.1 SR0

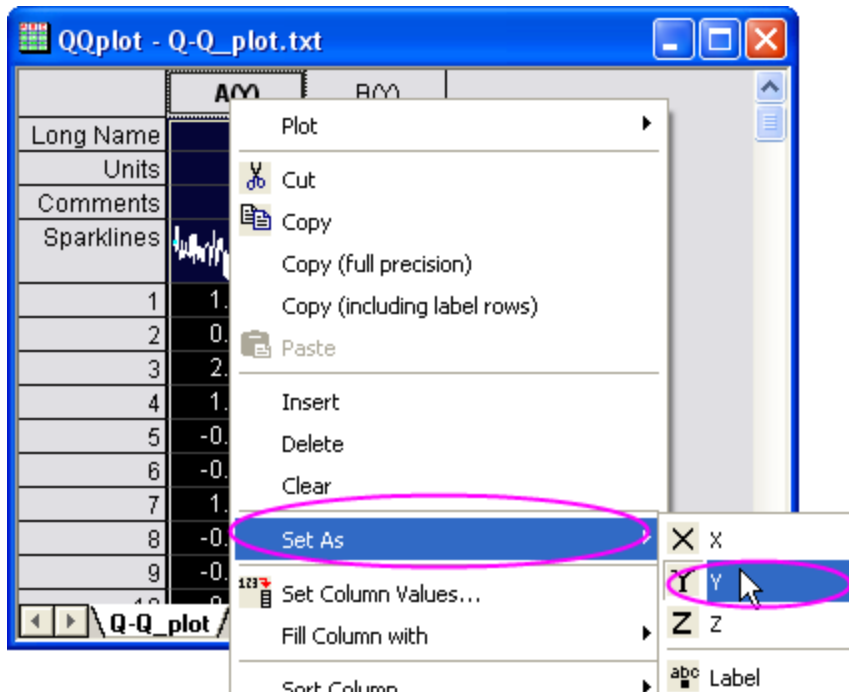
What you will learn

- How to create a normal Q-Q plot
- How to change the color and shape of the symbol
- How to change the color of the line

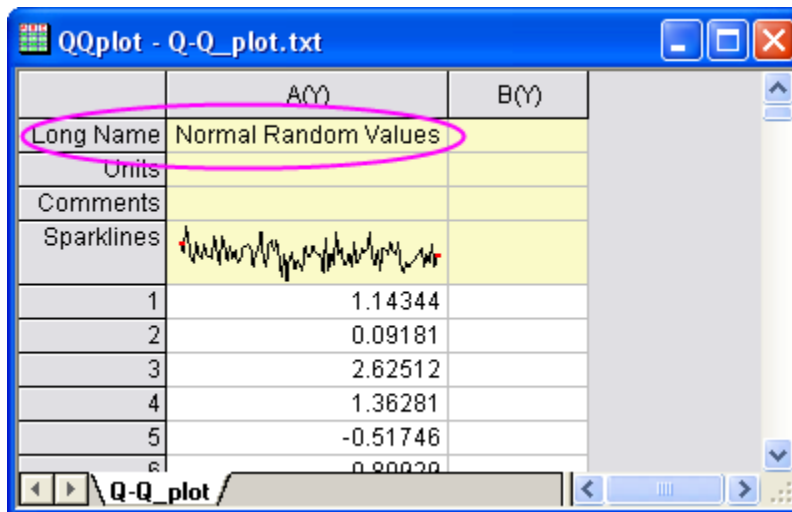
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`. (If you don't have the OPJ, you can also download the data from here)

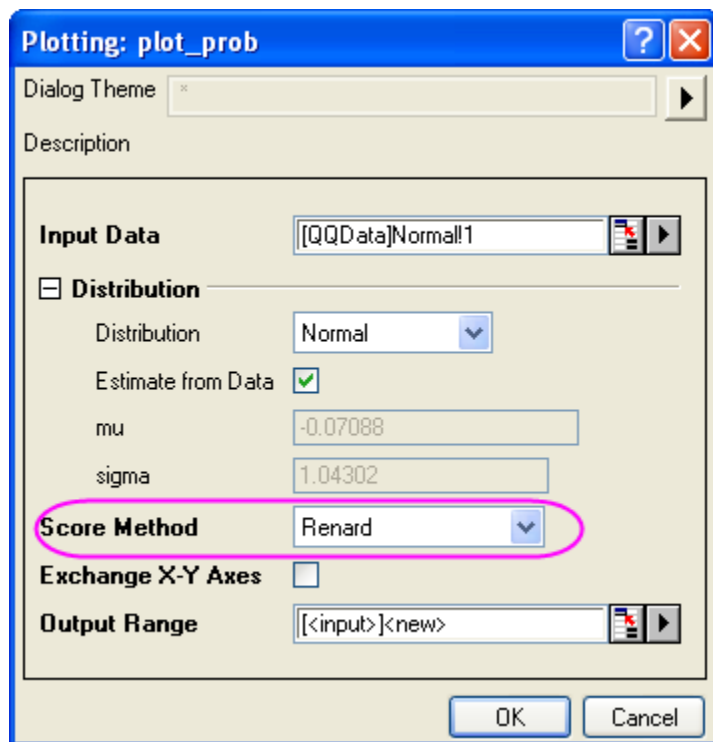
1. Open the sample project, open the folder *Graphing: Statistical: Probability, QQ Plot* in the **Project Explorer**. Highlight column A and right-click to select **Set As: Y** from the context menu.



2. Add a long name to this column, say **Normal Random Values**.



3. Highlight this column and select **Plot: Statistics: Q-Q Plot** from the Origin main menu to open the **plot_prob** dialog. In this dialog, change **Score Method** to **Renard**.



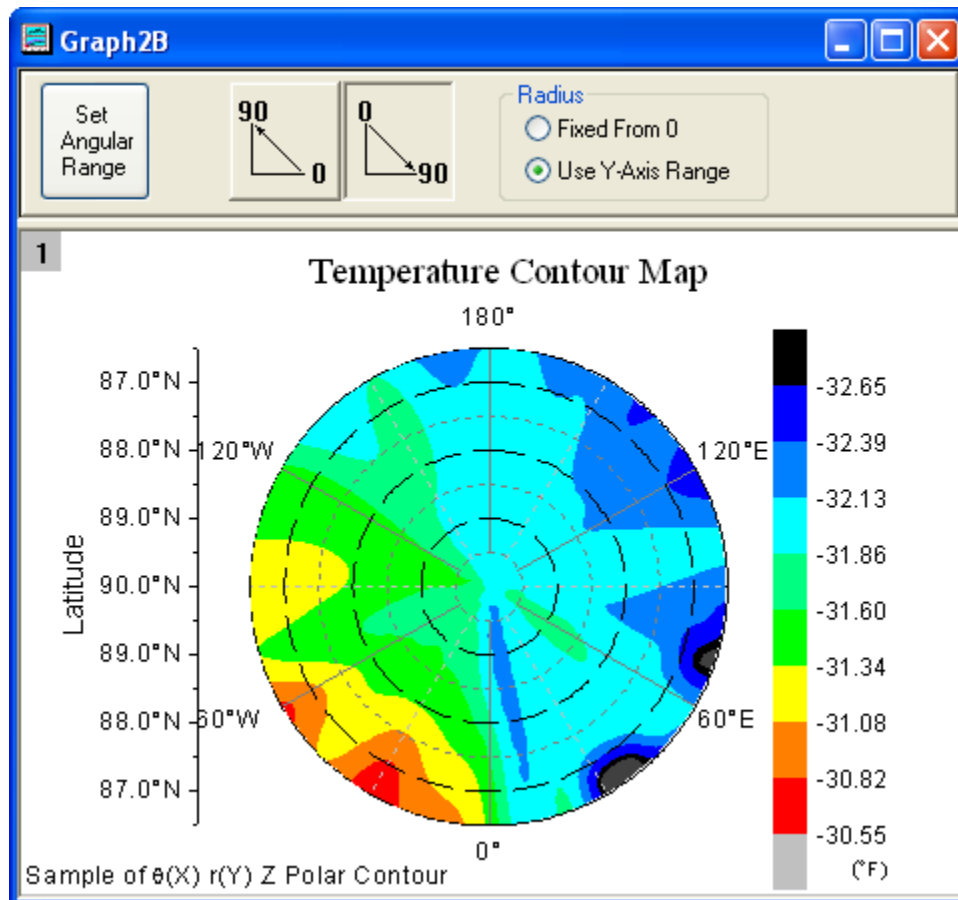
4. Click the **OK** button to create a normal Q-Q plot with column A.

3.7 Polar

3.7.1 Polar Contour

Summary

This tutorial will show you how to create a Polar Contour graph.



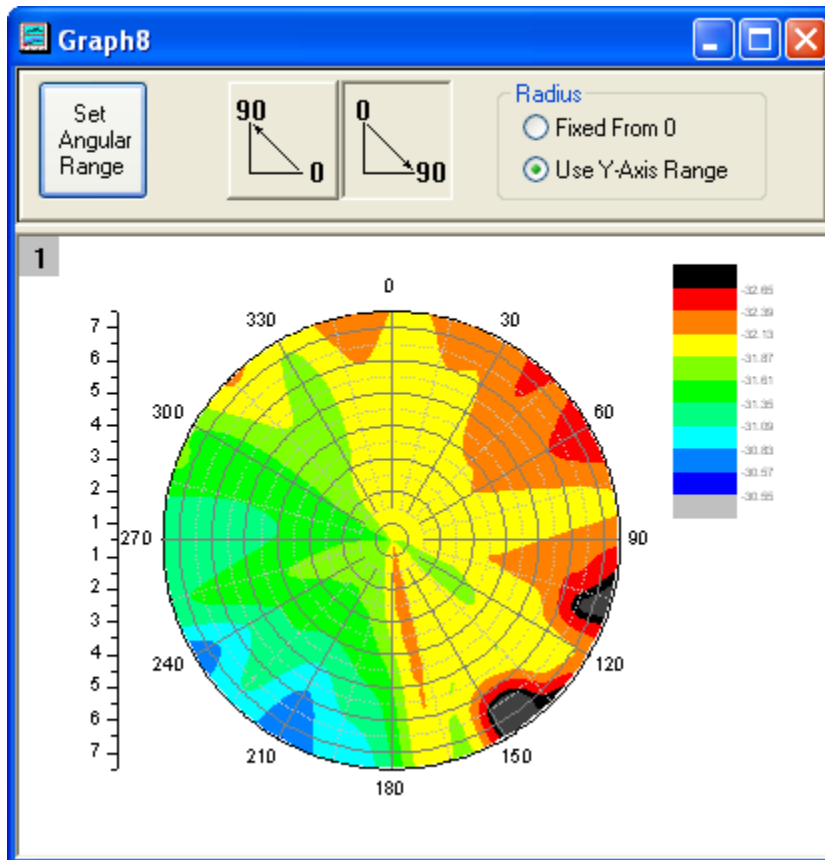
What will you learn

- Create a Polar Contour graph
- Customize the graph by using the Plot Details dialog

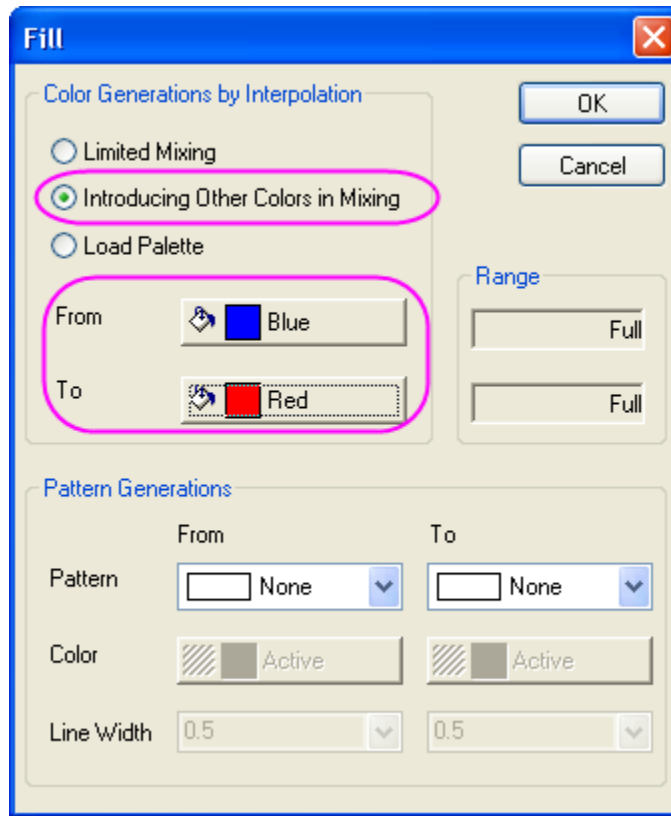
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

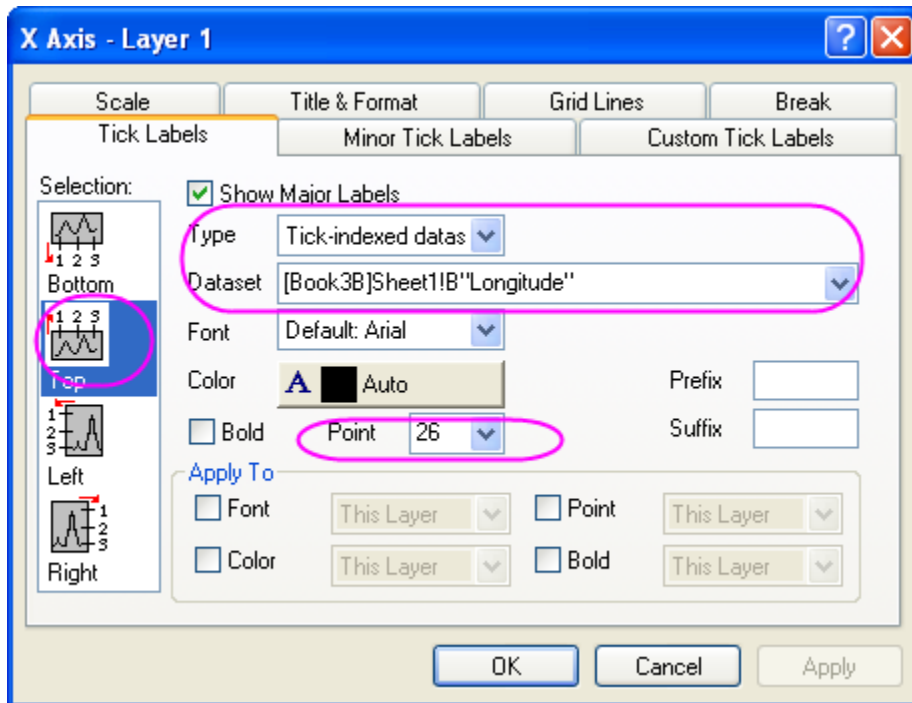
1. Open the *Graphing: Contour: Polar Contour* folder in the **Project Explorer**. Activate **Book2B** with the temperature and location data and select column C. Select **Plot: Contour: Polar Contour Theta(X) r(Y)** from the menu to create a polar contour graph. The graph should look like:



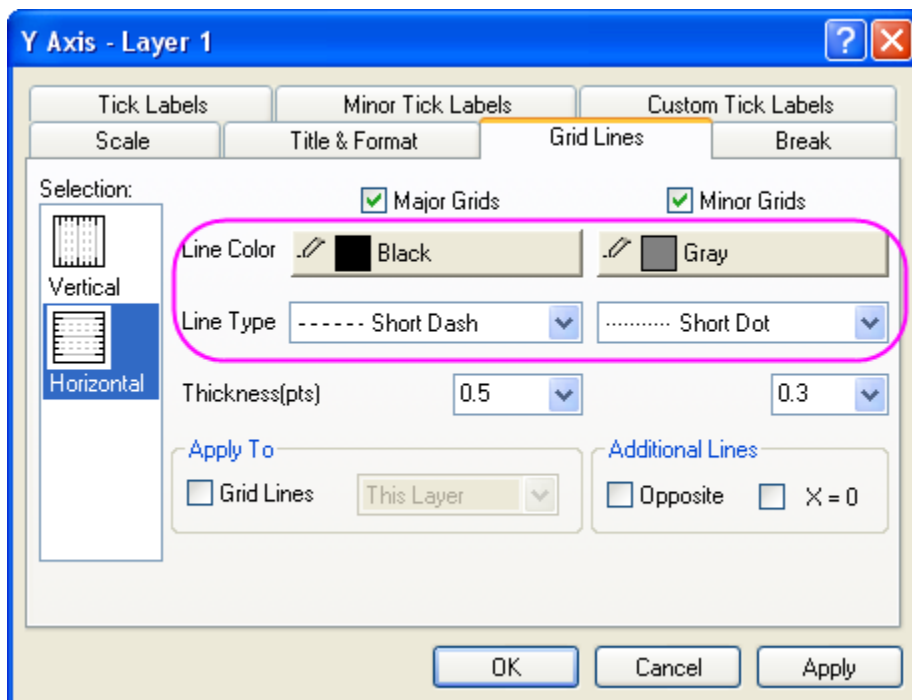
- Then we will customize the contour graph by using the **Plot Details** dialog. Double-click on the contour plot to bring up Plot Details dialog, select the **Color Map/Contours** tab and click the **Fill** heading, and then set the dialog as the following graph shows.



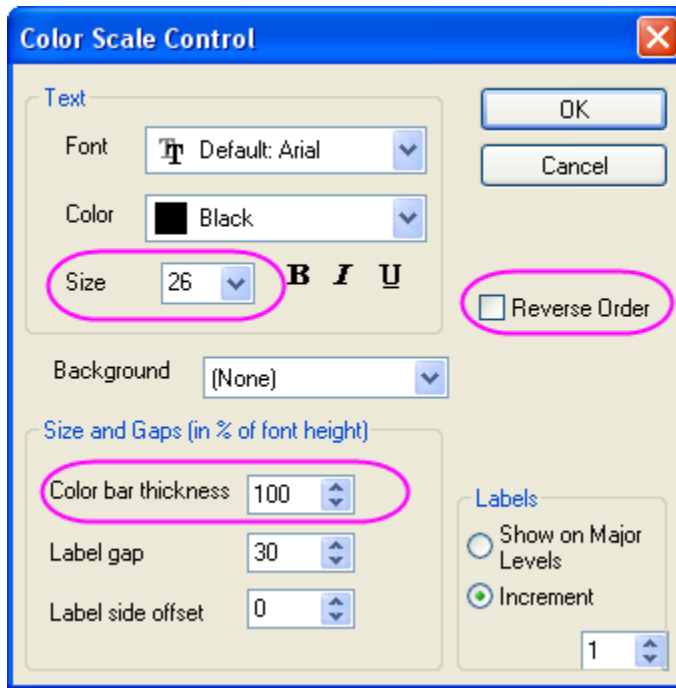
3. Click the **Contouring Info** Tab, select the **Layer Boundary** radio button.
4. Click OK button to close the **Plot Details** dialog.
5. Then we will customize the axes of the graph. Double-click on any tick label to open the **Axis** dialog. Then do the following things
 - o Select the **Scale** tab, choose **Horizontal** in the **Selection** list, set **Increment** to 60.
 - o Select **Vertical** in the **Selection** list, set **From** to 0, set **To** to 7 and set **Increment** to 2.
 - o Select **Tick Labels** tab and select **Top** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!B** for **Dataset**. Change **Point** to 26.



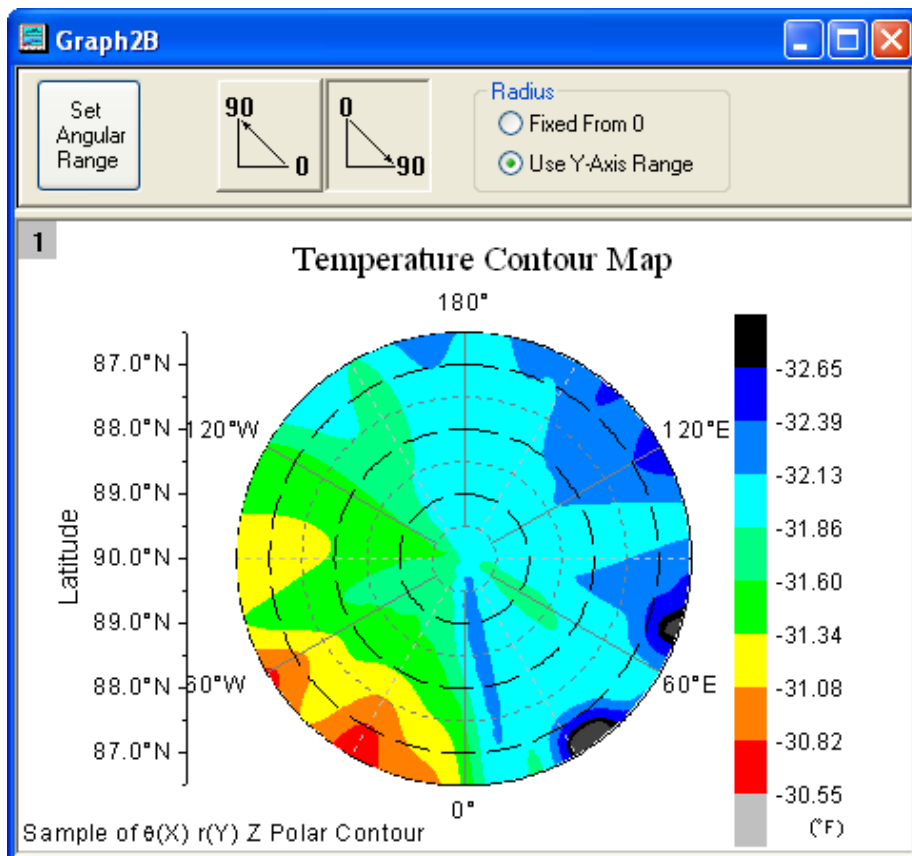
- Select **Left** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!A** for **Dataset**. Change **Point** to **26**.
- Select **Right** in the **Selection** list. Then select **Tick-Indexed Dataset** with the **Type** drop-down list. Choose **[Book3B]Sheet1!A** for **Dataset**. Change **Point** to **26**.
- Go to the **Grid Lines** tab and make sure **Horizontal** has been selected in the **Selection** list. Then set the dialog as the following image shows.



- Double click on the Color Scale to open the **Color Scale Control** dialog and set the dialog as the following image shows.



- The graph should look like



3.8 3D

3.8.1 Basic 3D Plotting

Summary

In Origin, most 3D plots -- including 3D surface, wire frame/wire surface, 3D bar plot and 2D contour -- are created from an Origin matrix. In most cases, the raw data is XYZ data and you should convert it to a matrix first, using one of Origin's built-in gridding routines.

Minimum Origin Version Required: Origin 8.0 SR6

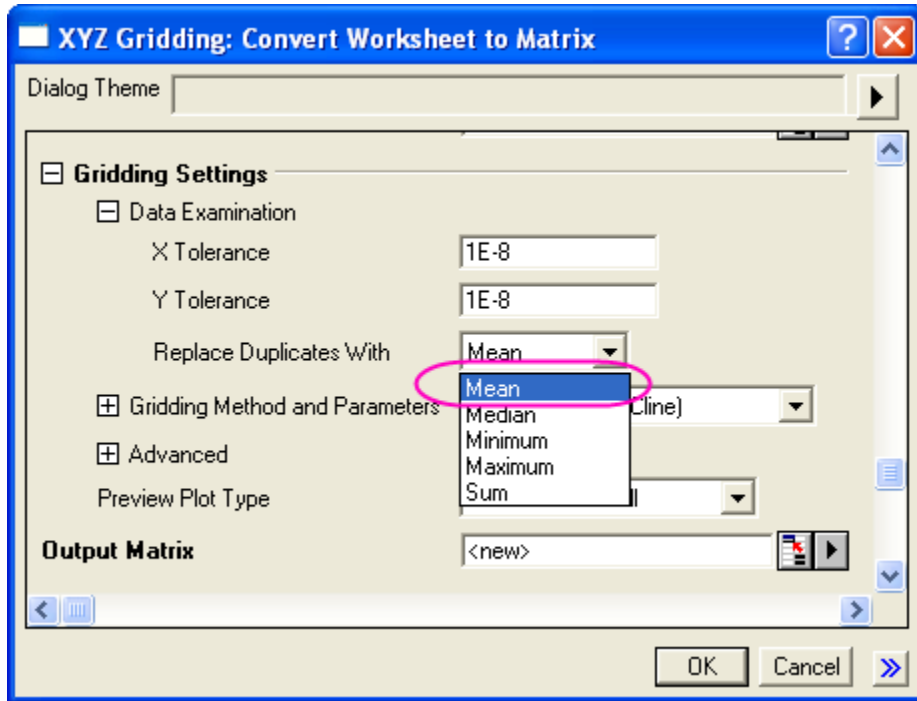
What you will learn

This tutorial will show you how to:

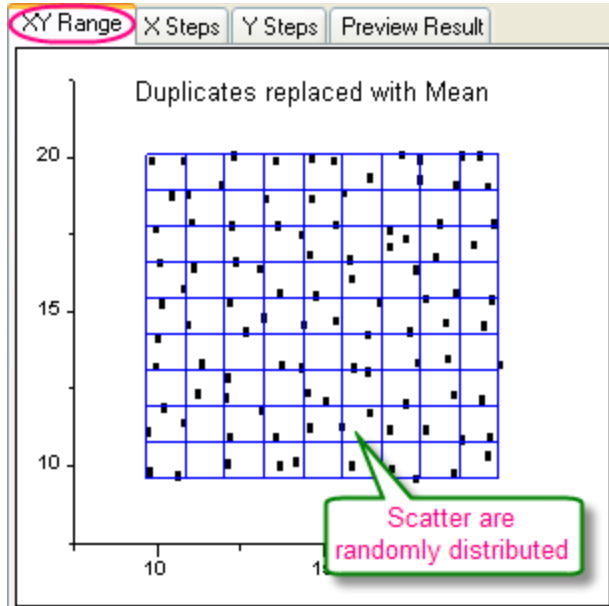
- Create a 3D graph in Origin
- Convert Worksheet data to a Matrix
- Use the layer contents dialog to add/remove dataset
- Use the Plot Details dialog to modify graph

Steps

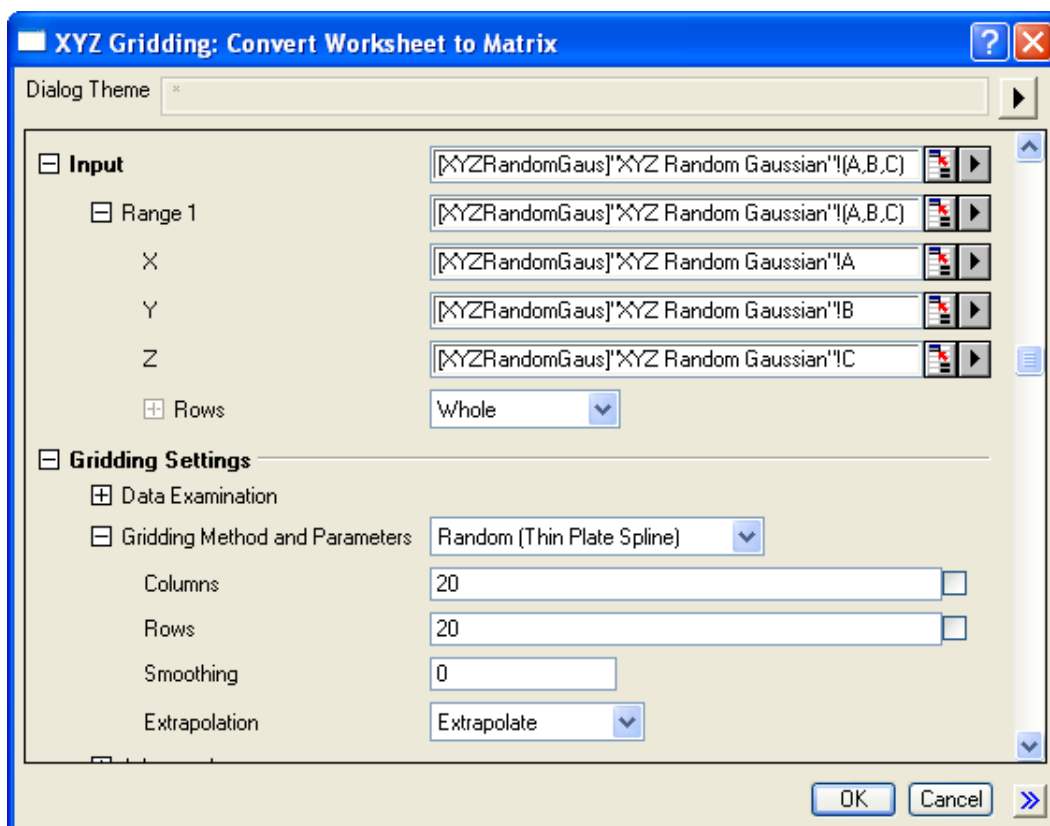
1. Import the file *\Samples\Matrix Conversion and Gridding\XYZ Random Gaussian.dat*.
2. Highlight the 3rd column, right-click, and select **Set As: Z** from the context fly-out menu.
3. To convert the worksheet XYZ data into a matrix, select **Worksheet: Convert to Matrix: XYZ Gridding** to bring up the **XYZ Gridding** dialog. Select **Mean** from the drop-down list of the **Replace Duplicates with** item as below:



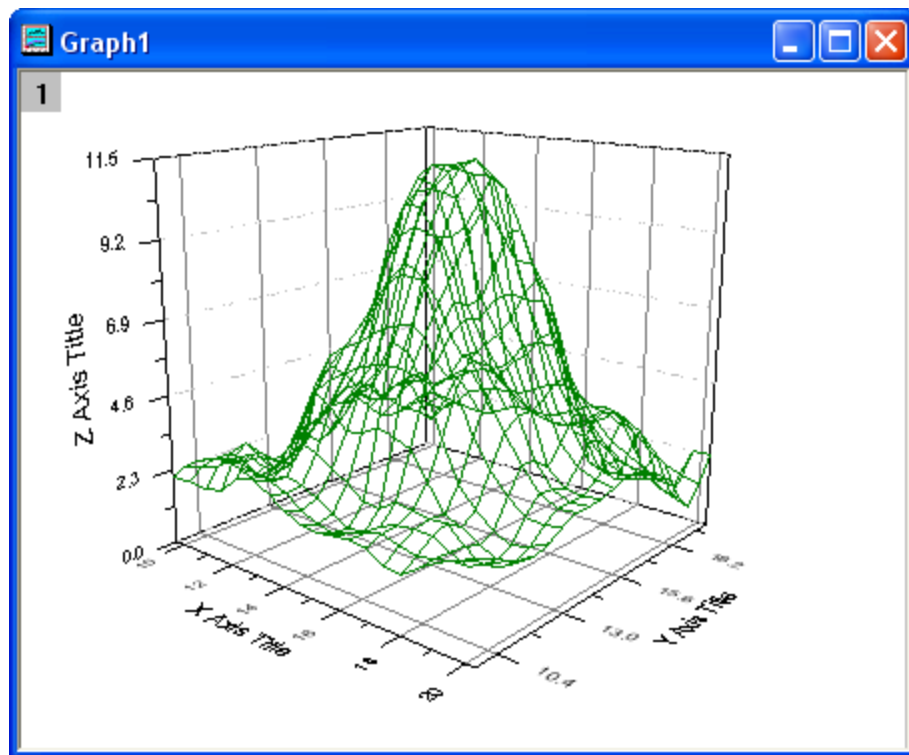
4. After that, you can see the right preview panel as below. As you can see, the XY data are randomly distributed, so a random gridding method should be used.



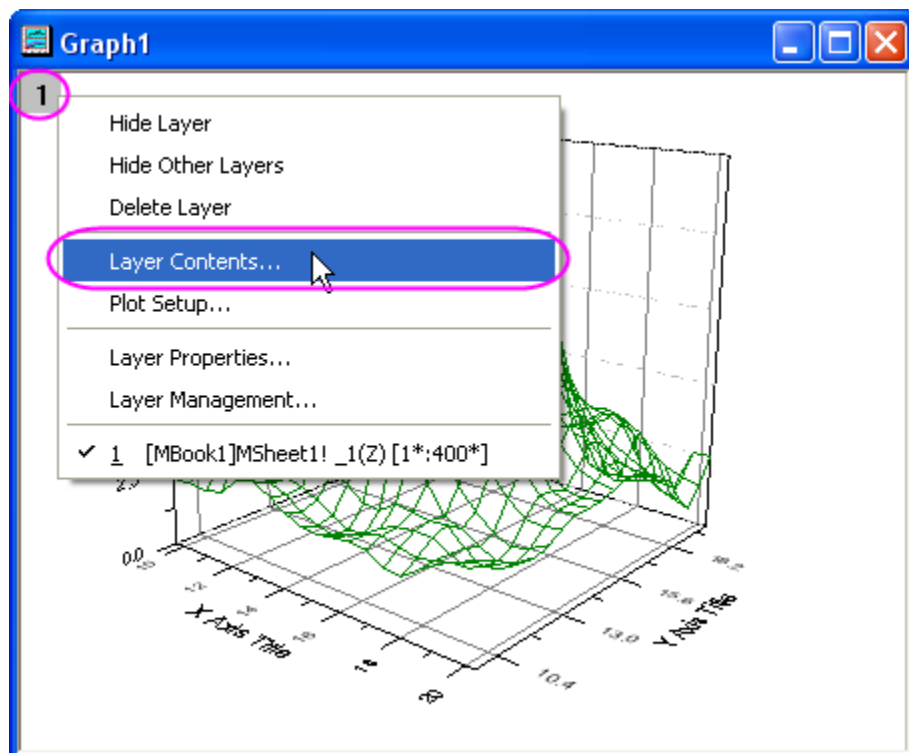
5. Use the following settings and click **OK** to convert the XYZ columns of data into a matrix of data. The TPS gridding method will generate a smooth surface.



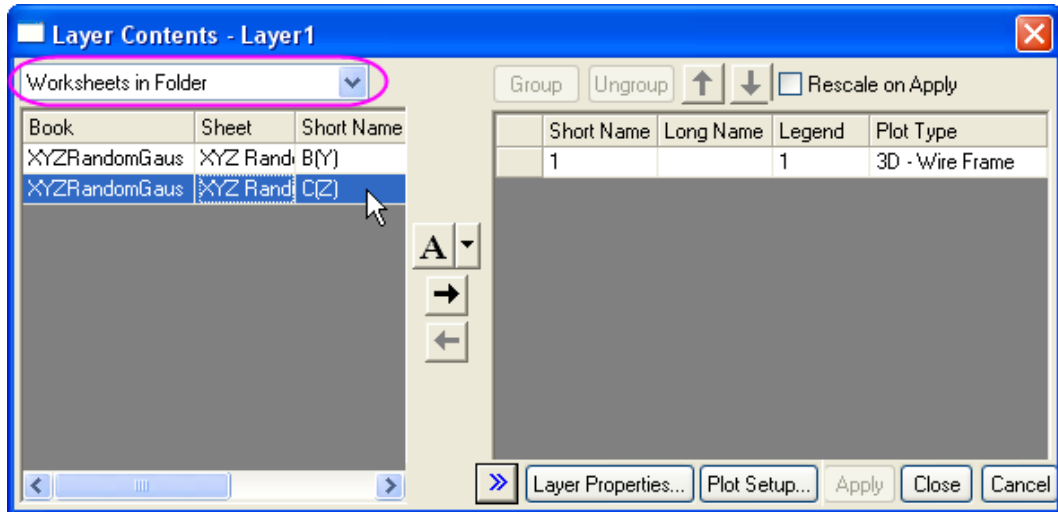
6. Set the new generated matrix as the active window and select **Plot: 3D Wire/Bar/Symbol: Wire Frame** from the menu to plot a 3D mesh:



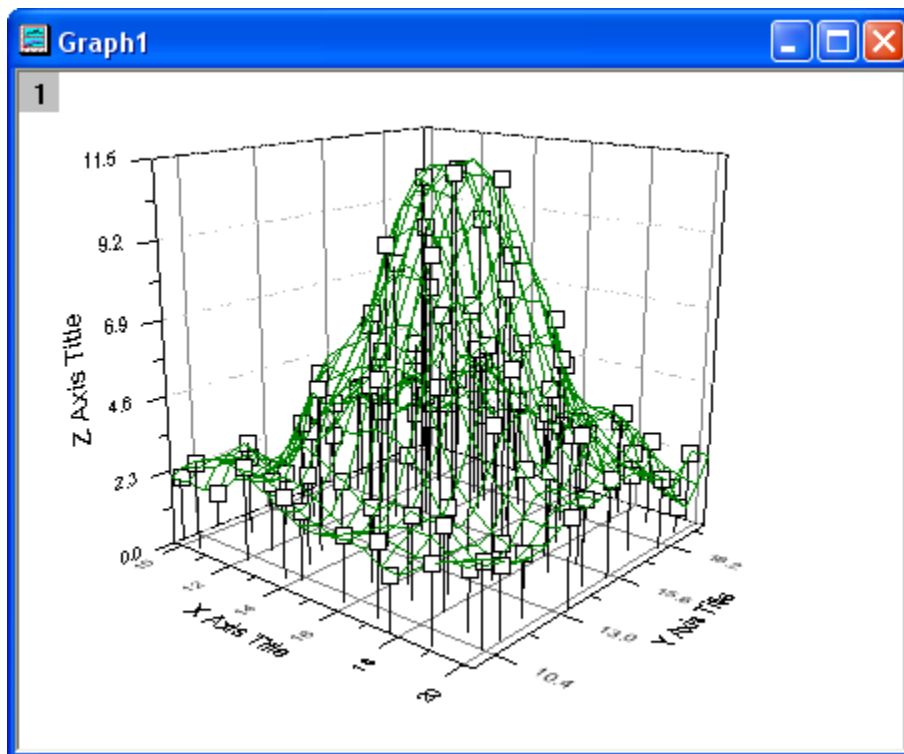
7. In order to plot the original data points on the graph, you can use the **Layer Contents**. Right-click on the layer icon and select **Layer Contents**.



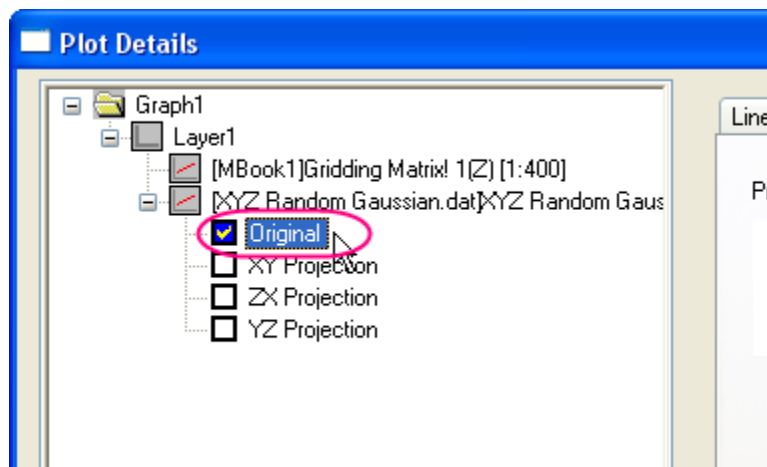
In the Layer Contents dialog, select **Worksheets in Folder** from the available data box (in the upper left corner of the dialog). Then select the worksheet Z column and click -> button to add it into the right panel list.



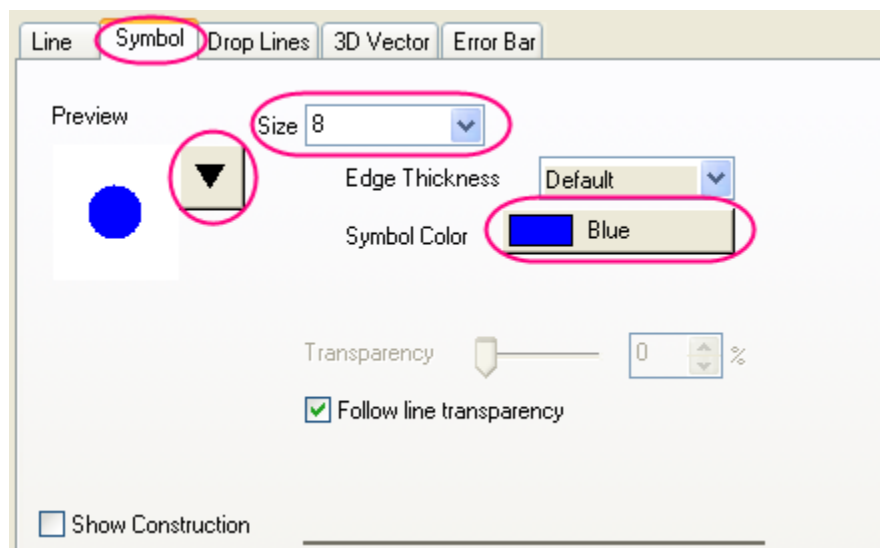
After you click **Apply** button, the source data will be added to the layer.



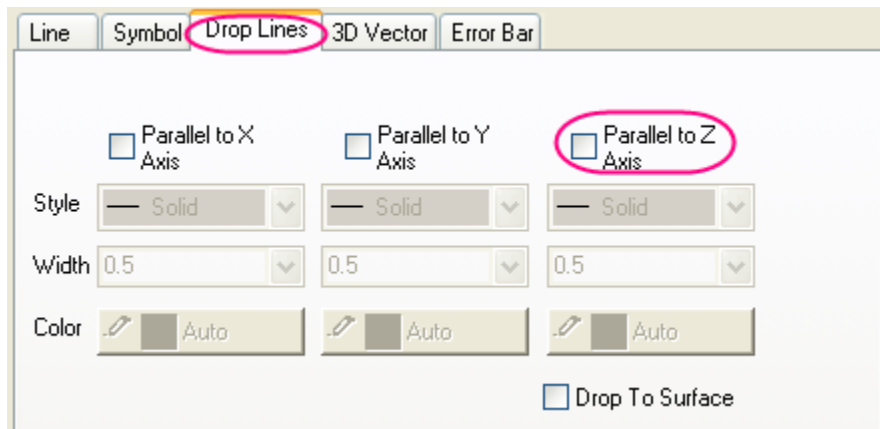
- You can now use the **Plot Details** dialog to modify the appearance. Double-click on the graph to bring up the **Plot Details** dialog. On the left panel, select the 3D scatter data:



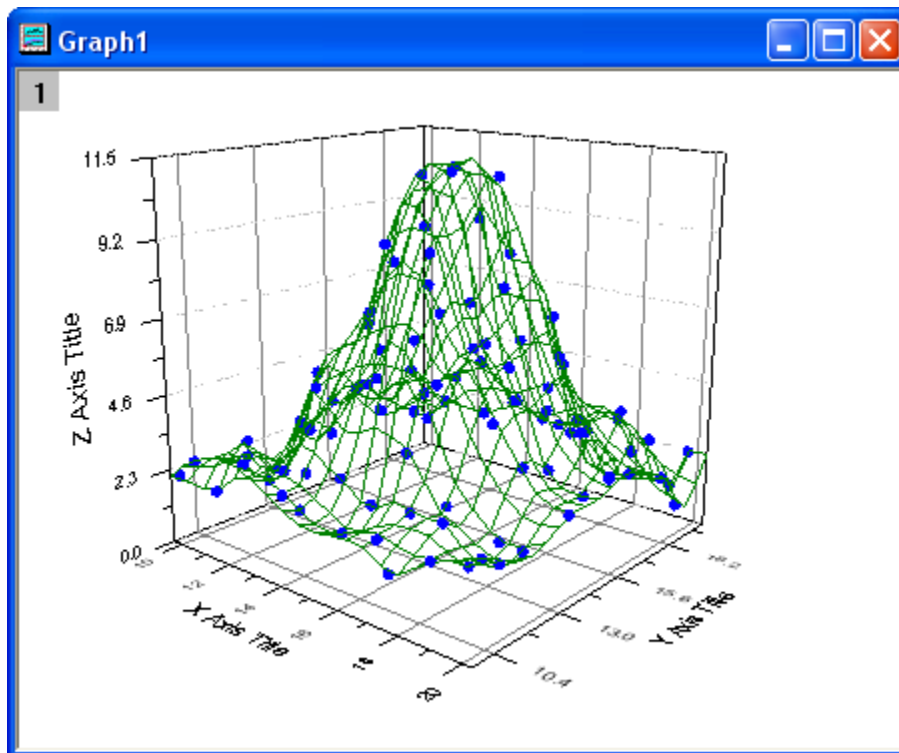
And then go to the **Symbol** tab on the right panel, adjust the symbol type, size and color, etc.



Remove the drop lines on the **Drop Lines** panel:



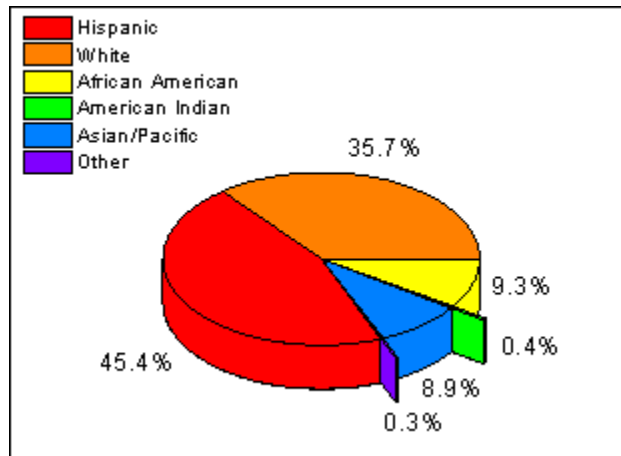
9. When done, click **OK** to accept the modifications:



3.8.2 3D Pie Chart

Summary

Origin's 3D Pie Chart gives you complete control over the look of the plot. Set the thickness of the pie slice, displacement, view angle, and size and rotation of the chart. You can even choose to explode the view of one or more pie slices.



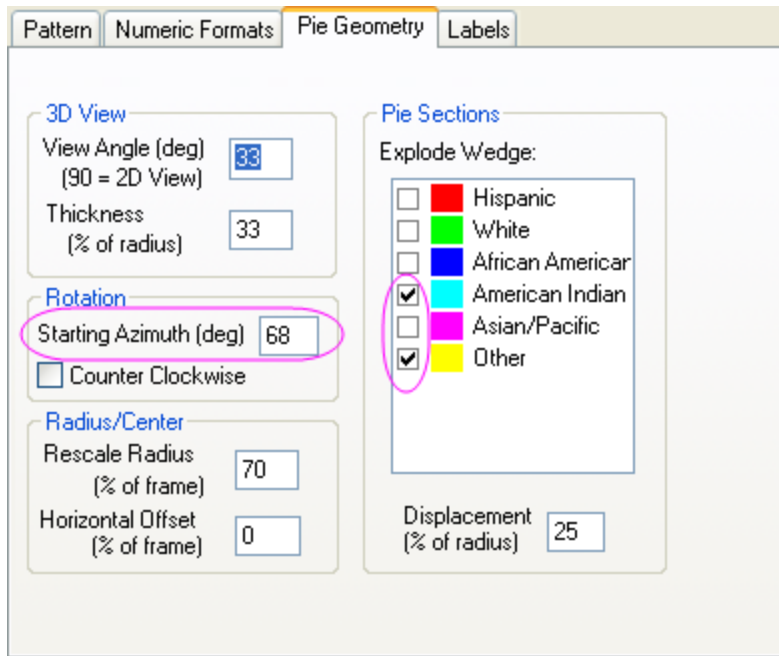
What you will learn

- Create 3D pie chart and change the view angle.
- Explode the view of pie slices.
- Customize pie slices individually.

Steps

This tutorial is associated with the folder **Column and Pie** of the sample project:
 \Samples\Samples.opj.

1. Import the data \Samples\Graphing\3D Pie Chart.dat into worksheet, and rename the column long name as *Demographic* and *Percent*.
2. Highlight the 2nd column and create a 3D Pie Chart from **Plot: Column/Bar/Pie: 3D Color Pie Chart**.
3. Double-click the pie chart to bring up the **Plot Details** dialog. Switch to **Pie Geometry** tab, and change the **Rotation** degree to 68. Then check the checkbox for *American Indian* and *Other* in **Explode Wedge** group as below and click **OK**.

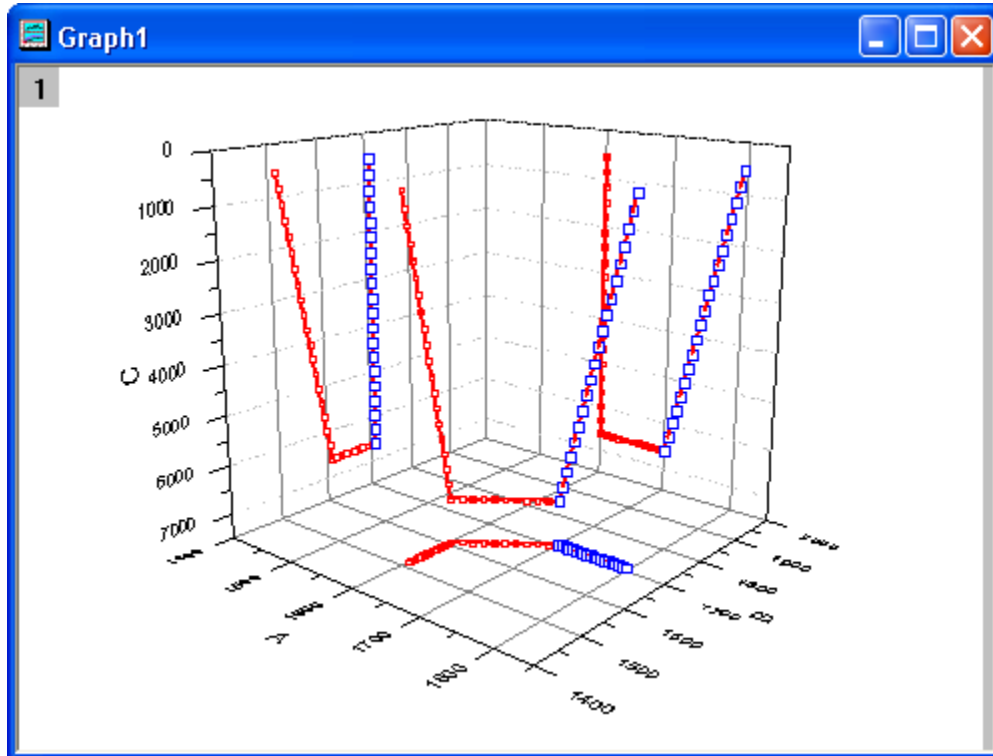


4. Then hold the **Ctrl** key and double click each pie slices to open **Plot Details** dialog, and change **Fill Color** in **Pattern** tab, respectively.

3.8.3 3D Scatter Plot with Line Projections of Core Drilling Locations

Summary

This tutorial will show you how to create a 3D scatter plot and how to show the projections of the plots.



Minimum Origin Version Required: Origin 8.5.1

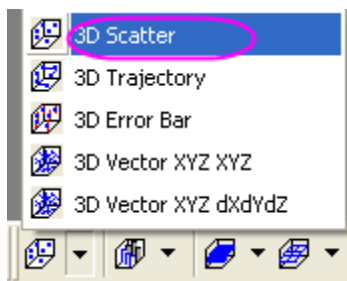
Steps


1. Create two new workbooks. Import the data into the workbooks

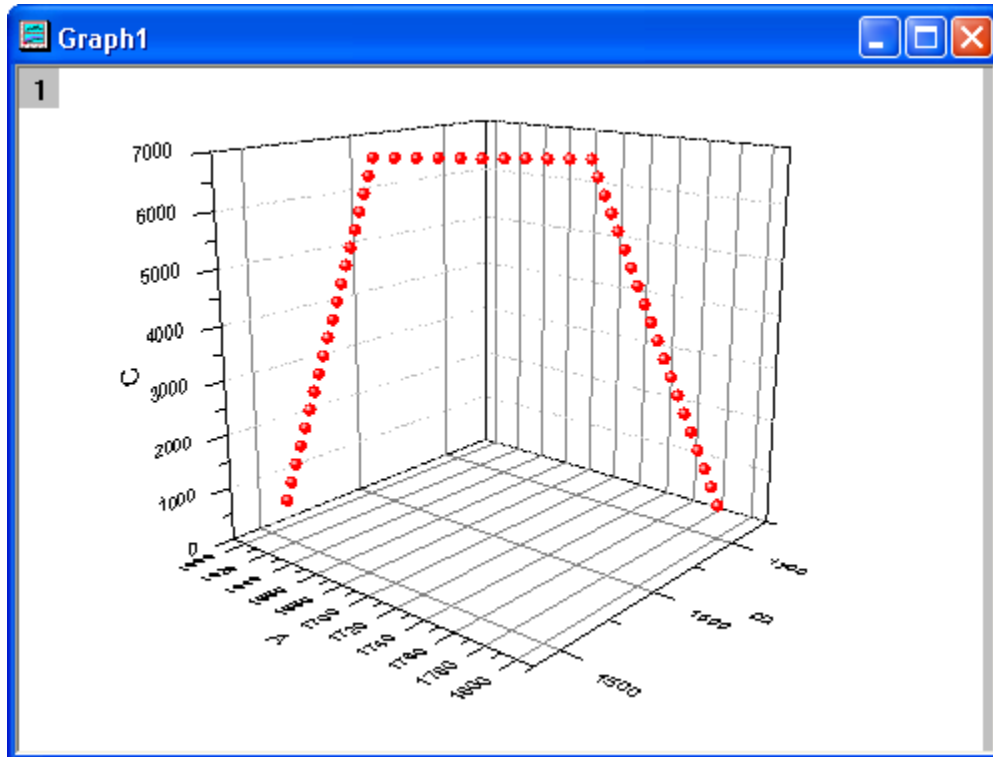
	A(X)	B(Y)	C(Z)
Long Name			
Units			
1	1600	1610	6600
2	1600	1605	6280
3	1600	1600	5960
4	1600	1595	5640
5	1600	1590	5320
6	1600	1585	5000
7	1600	1580	4680
8	1600	1575	4360
9	1600	1570	4040
10	1600	1565	3720
11	1600	1560	3400
12	1600	1555	3080
13	1600	1550	2760
14	1600	1545	2440

	A(X)	B(Y)	C(Z)
Long Name			
Units			
1	1700	1710	6600
2	1705	1710	6280
3	1710	1710	5960
4	1715	1710	5640
5	1720	1710	5320
6	1725	1710	5000
7	1730	1710	4680
8	1735	1710	4360
9	1740	1710	4040
10	1745	1710	3720
11	1750	1710	3400
12	1755	1710	3080
13	1760	1710	2760
14	1765	1710	2440

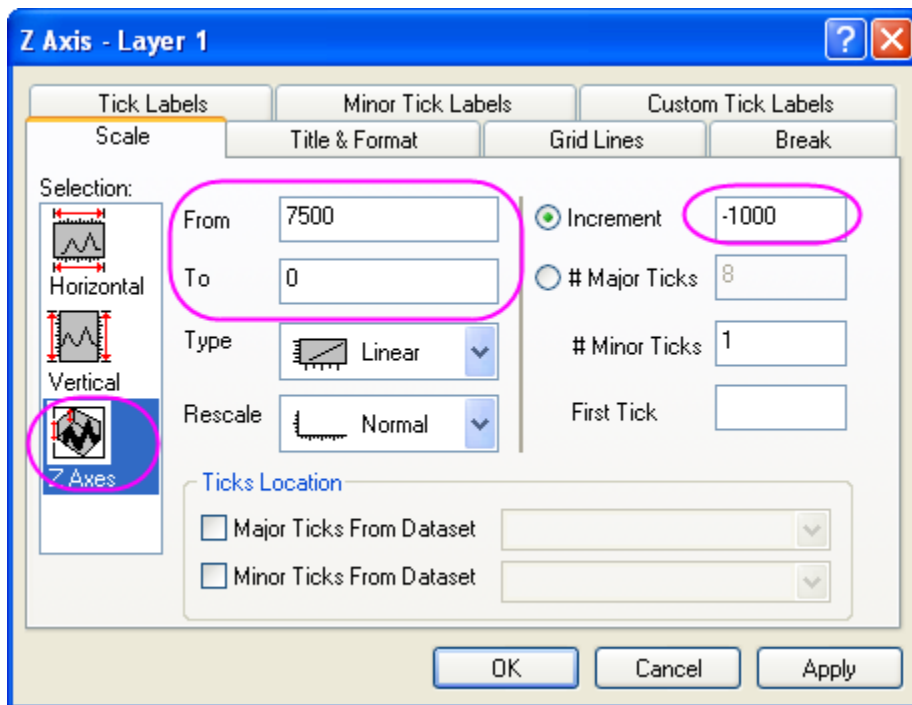
2. Activate **The First Curve of 3D Scatter** worksheet. Highlight column C. Right-click on it and select **Set As: Z** from the context menu. Then click the **3D Scatter** button on the **3D and Contour Graphs** toolbar as the screenshot below.



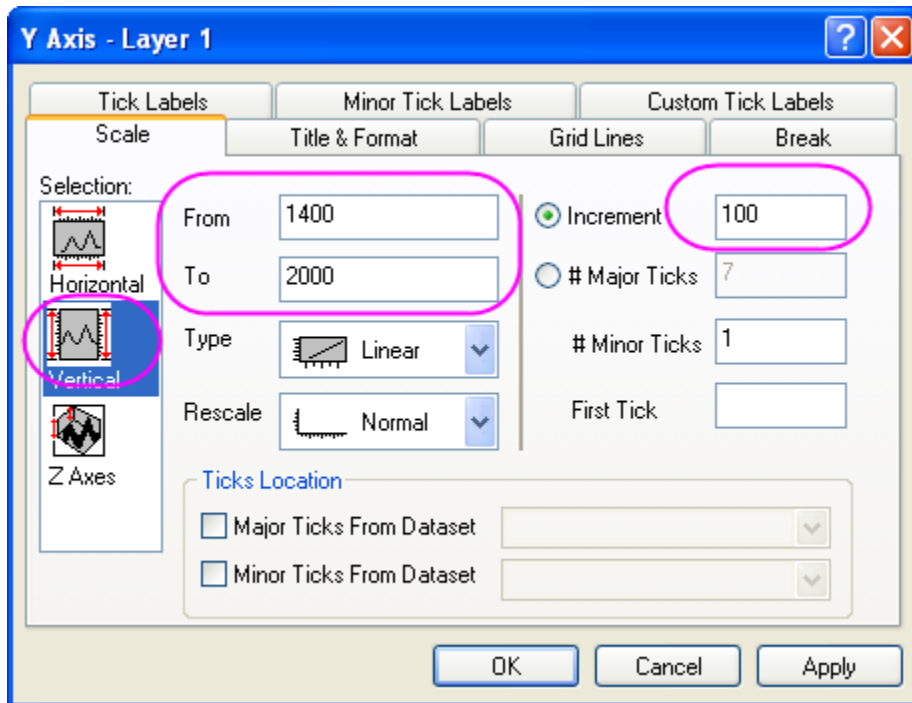
3. Activate **The Second Curve of 3D Scatter** worksheet. Highlight column C. Right-click on it and select **Set As: Z** from the context menu. Then move the mouse to the right edge of the selection area until the shape of the mouse change to . Then hold down the left mouse button and drag the highlighted data into the new created graph window. The graph should look like



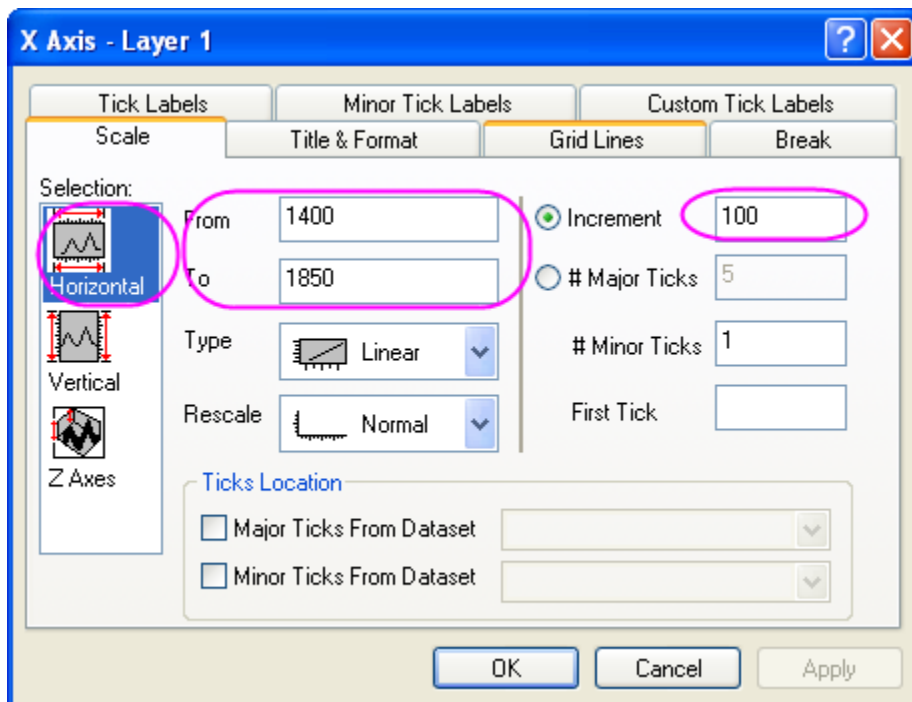
4. Double-click on the Z axis to open the **Axis Properties** dialog. In **Scale** tab, set the dialog options as the screenshot below.



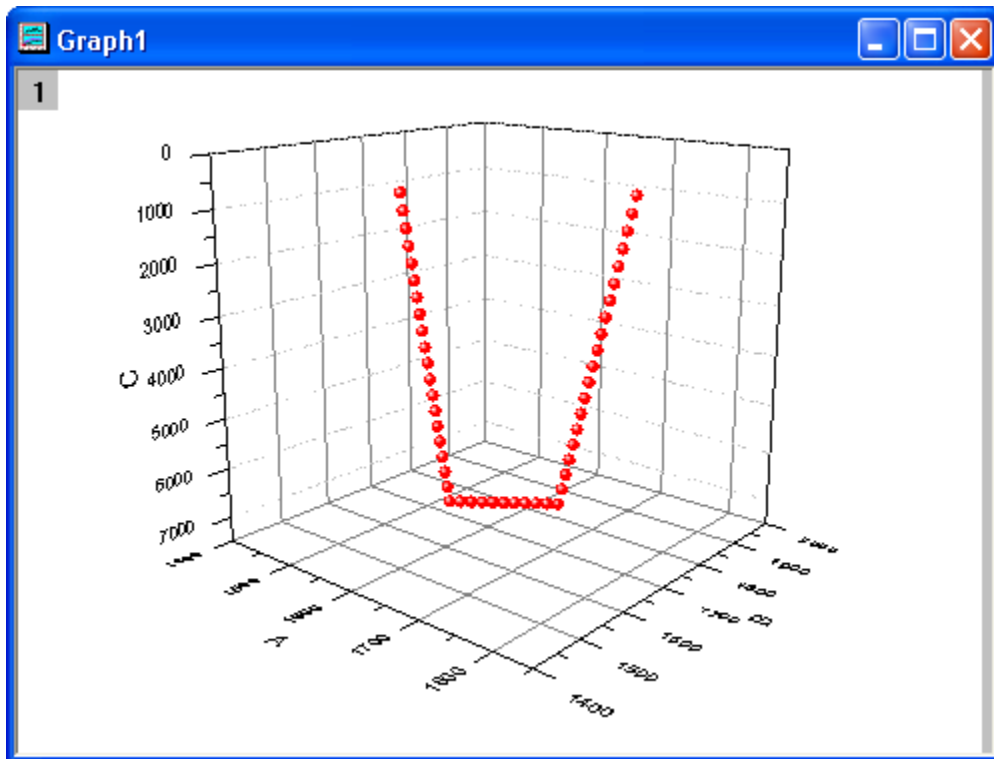
Select **Vertical** in the **Selection** list box. In **Scale** tab, set the dialog options as the screenshot below.



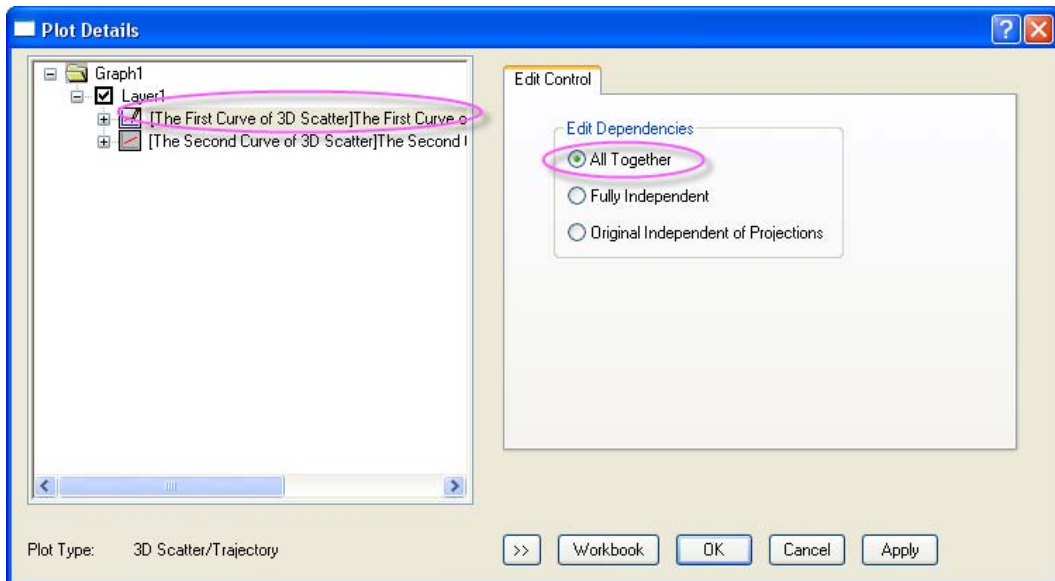
Select **Horizontal** in the **Selection** list box. In **Scale** tab, set the dialog options as the screenshot below.



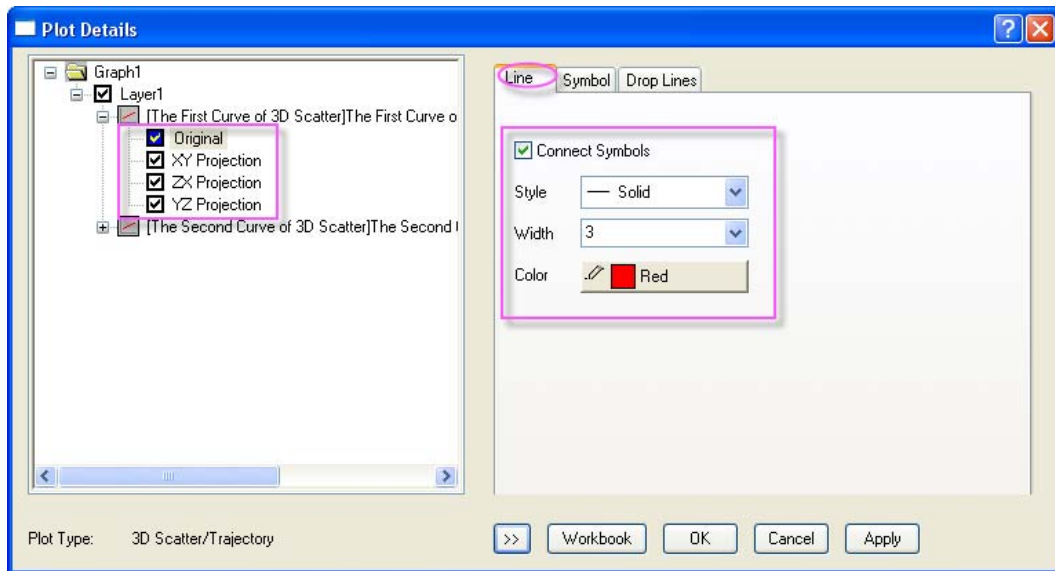
The graph should look like



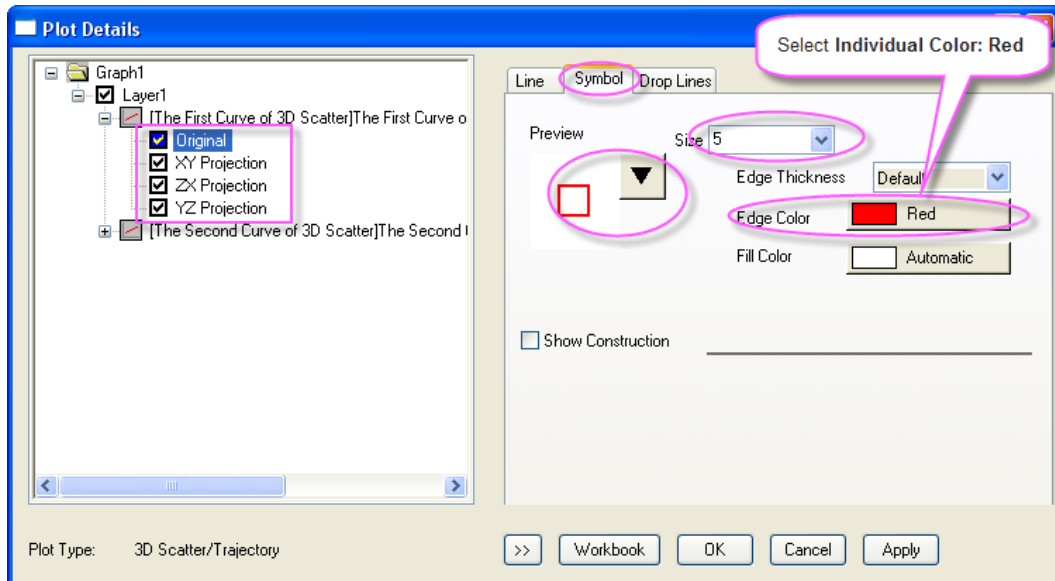
5. Select **Format: Layer** from the main menu to open the **Plot Details** dialog. Expand the **Layer1** node. Select the **first** plot. Set the dialog options as the screenshot below.



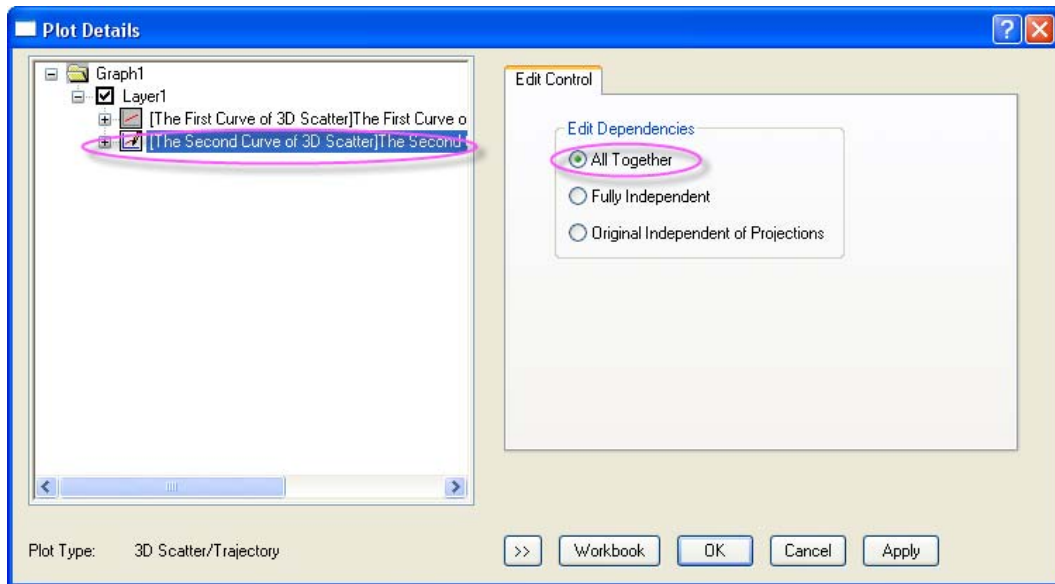
Expand the first plot and select the **Original** plot. Then set the dialog options as the screenshot below.



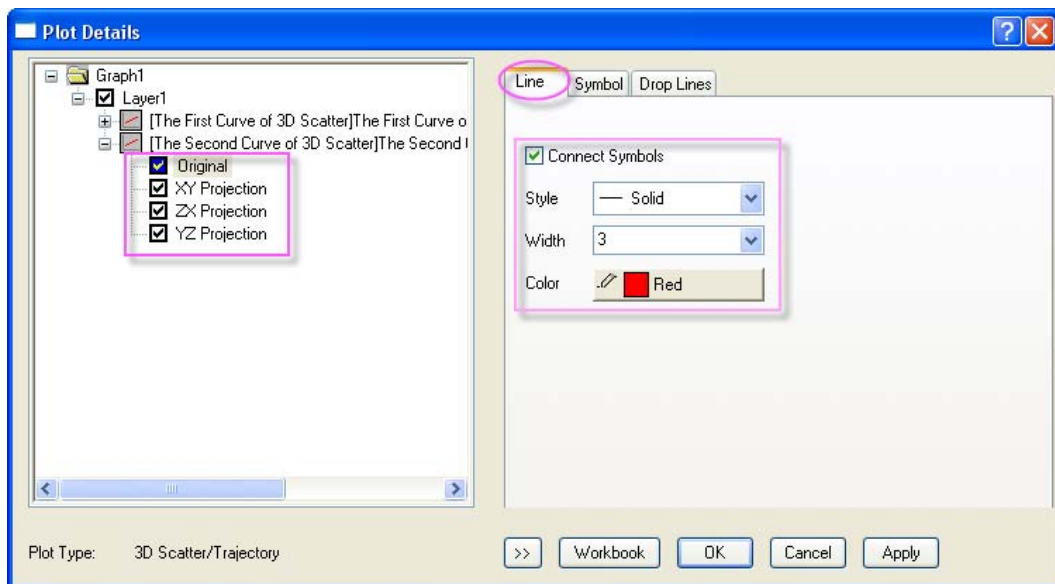
6. In the **Symbol** tab, set the dialog options as the screenshot below.



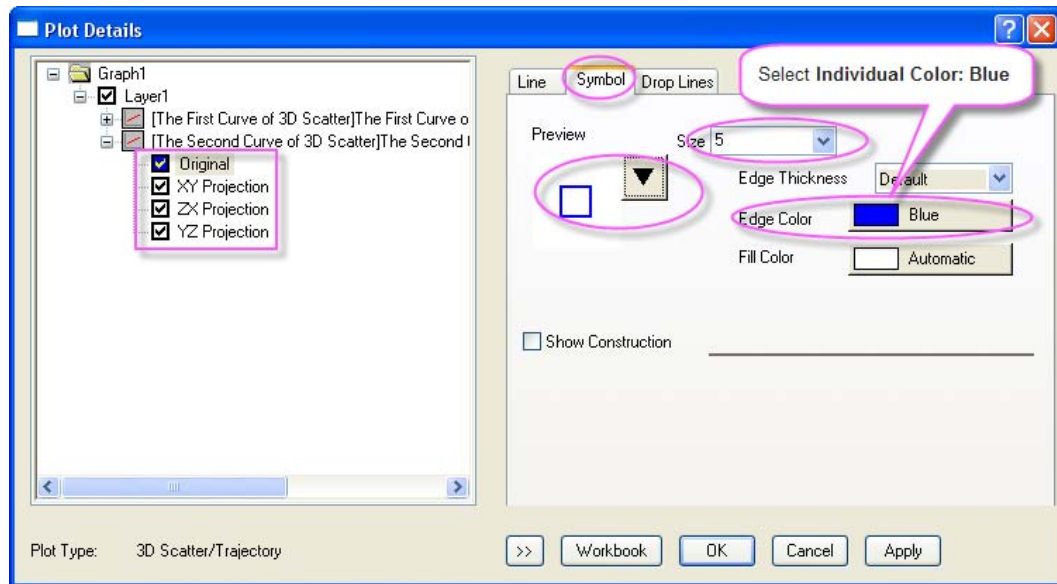
7. Select the second plot. Set the dialog options as the screenshot below.



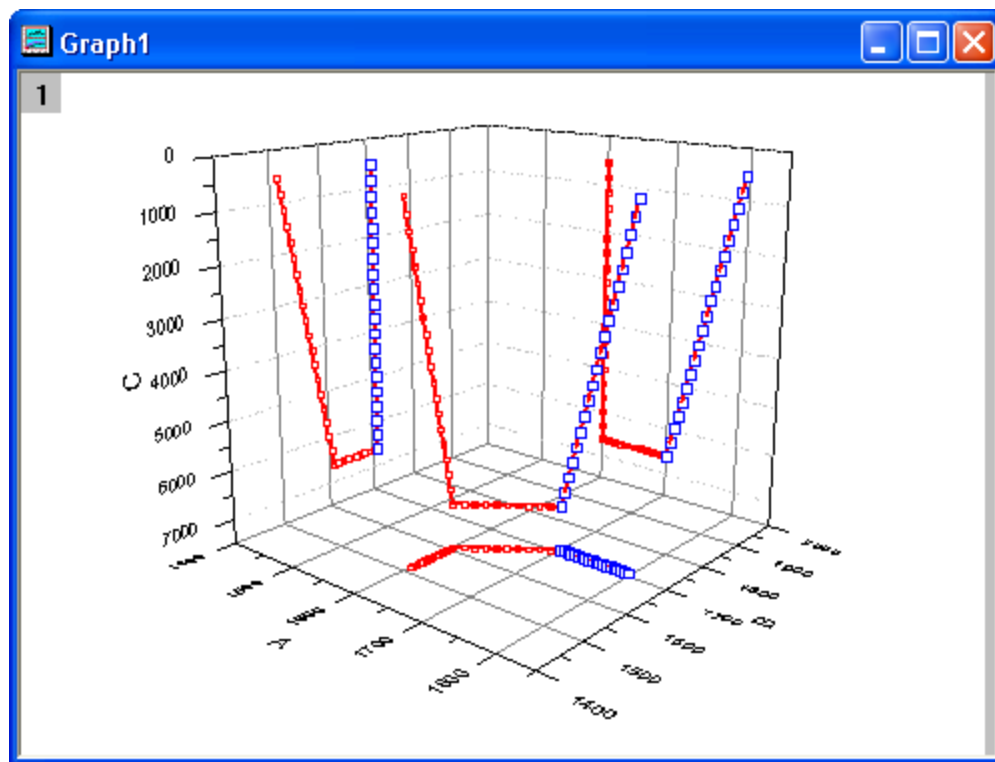
Expand the second plot and select the **Original** plot. Then set the dialog options as the screenshot below.



8. In the **Symbol** tab, set the dialog options as the screenshot below.




The graph should look like



Sample Data

Download the [The_First_Curve_of_3D_Scatter.txt](http://www.originlab.com/ftp/graph_gallery/data/The_First_Curve_of_3D_Scatter.txt) file and [The_Second_Curve_of_3D_Scatter](http://www.originlab.com/ftp/graph_gallery/data/The_Second_Curve_of_3D_Scatter.txt) file from http://www.originlab.com/ftp/graph_gallery/data/The_First_Curve_of_3D_Scatter.txt and http://www.originlab.com/ftp/graph_gallery/data/The_Second_Curve_of_3D_Scatter.txt. Create a

new worksheet. Click the **Import Single ASCII** button  and select the **The First Curve of 3D Scatter.dat** to import it into Origin. Then select **File: New: Workbook** from the main menu to create another empty workbook and import the **The Second Curve of 3D Scatter.dat** with the **Import Single ASCII** button.

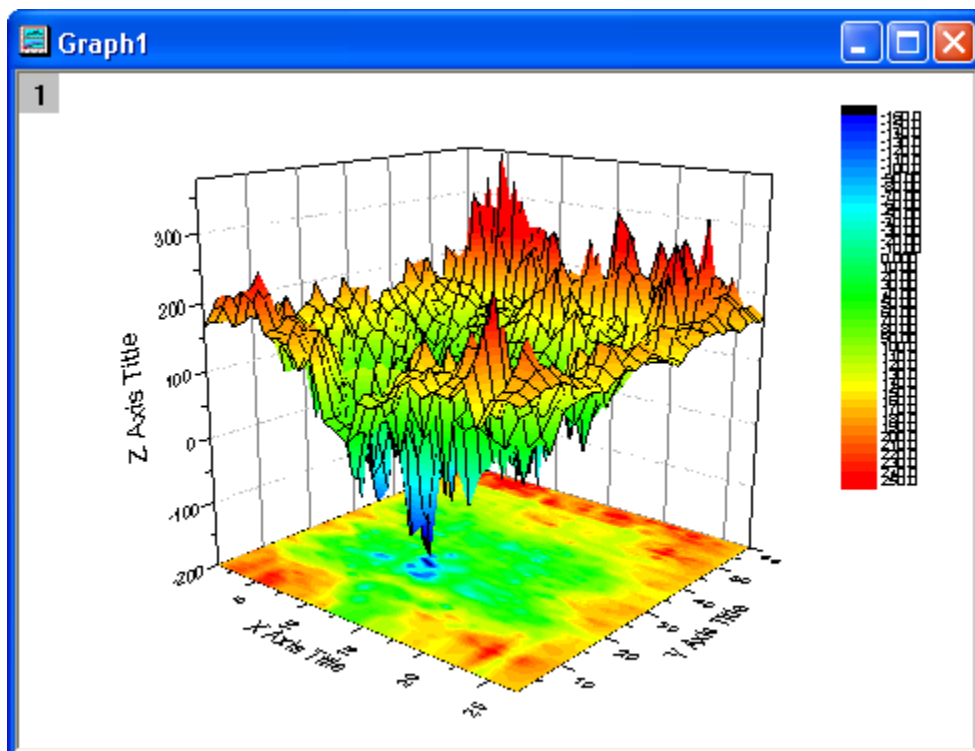
The following table contains part of the sample data.

A(X)	B(Y)	C(Z)
1600	1610	6600
1600	1605	6280
1600	1600	5960
1600	1595	5640
1600	1590	5320
1600	1585	5000
1600	1580	4680
1600	1575	4360
1600	1570	4040
1600	1565	3720

3.8.4 Emission Intensity of Single-walled Nanotubes Plotted as a surface Graph

Summary

This tutorial will show you how to create a 3D surface graph and how to customize it.



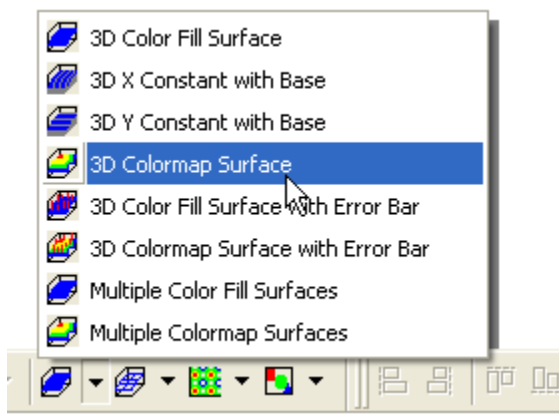
Minimum Origin Version Required: Origin 8.5.1

Steps

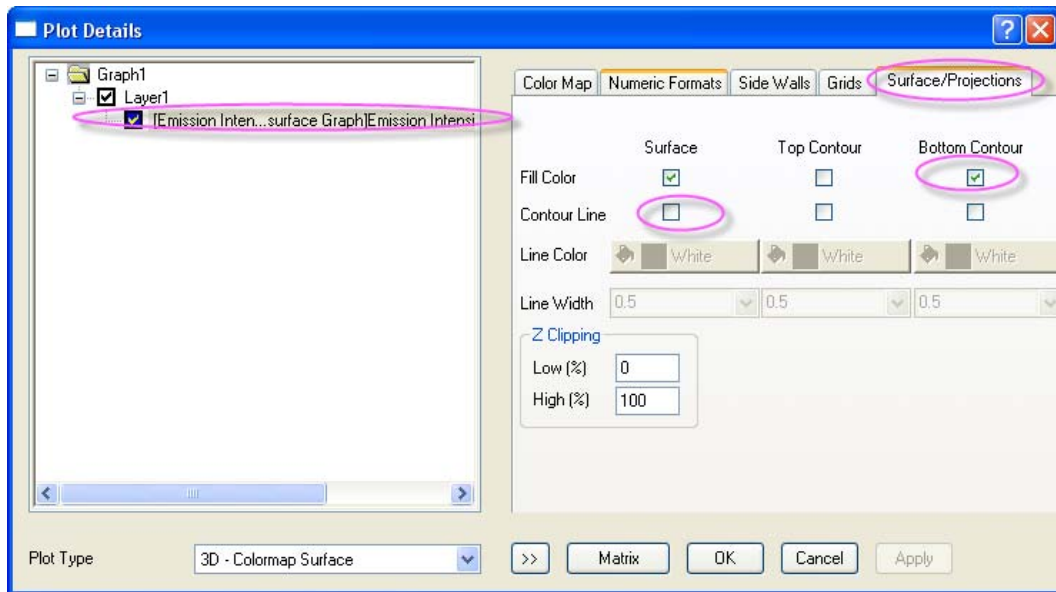
1. Create a new matrix. Import the data.

	1	2	3	4	5	6	7
1	166	195.3	203	174.9	229.8	217.4	17
2	171.5	211.3	217.5	179.5	228.6	233.6	18
3	169.6	198	209.5	181.4	238	220.1	17
4	167.6	201.5	215.4	187.3	252.5	225.8	18
5	171.2	175.4	202.9	181.5	229.1	182.1	19
6	177.9	178.6	206	181	213.5	179.5	19
7	178.1	176.7	212.9	174.9	181.7	218.6	1
8	206.8	174.2	199.7	155.2	176.1	172	17
9	182.8	193.2	180.1	163.1	182.7	174	17
10	165.7	184.2	134.8	133.1	133	141.6	17
11	167.4	160.6	137.7	114.5	115.3	122.7	1
12	144.7	129.6	86.6	92.5	101.8	111.2	10
13	143.9	129.9	110	95.9	110.6	74.4	1
14	152.6	139.1	115.1	102	18.1	57.7	8
15	179.5	170	144.7	122.5	72.8	71.3	2
16	168.1	123.4	123.5	130.6	86.6	67.6	2
17	166.8	120.6	100.7	56.8	64.9	59.6	5
18	143.4	105.5	63.1	59.7	37.2	-34	1
19	154.9	141.1	85.2	17.5	78.6	13.8	-
20	161.8	143	108.1	66.5	99.4	88.2	3

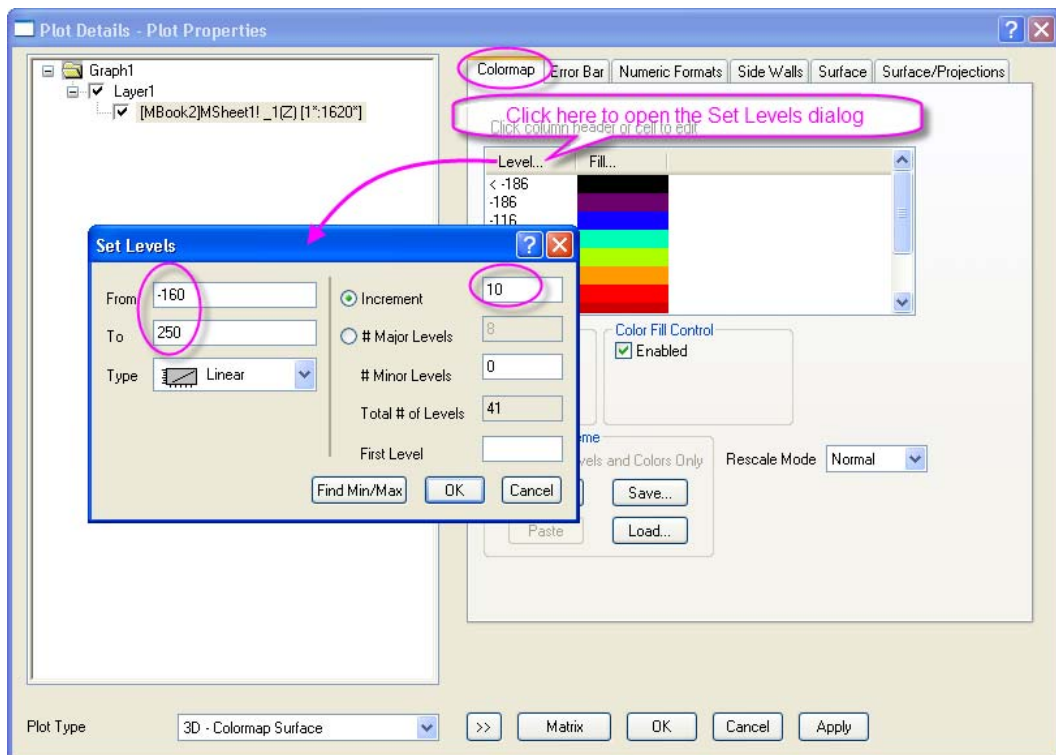
- Highlight the matrix and click the **3D Color Map** button in the **3D and Contour Graphs** toolbar to create a graph.



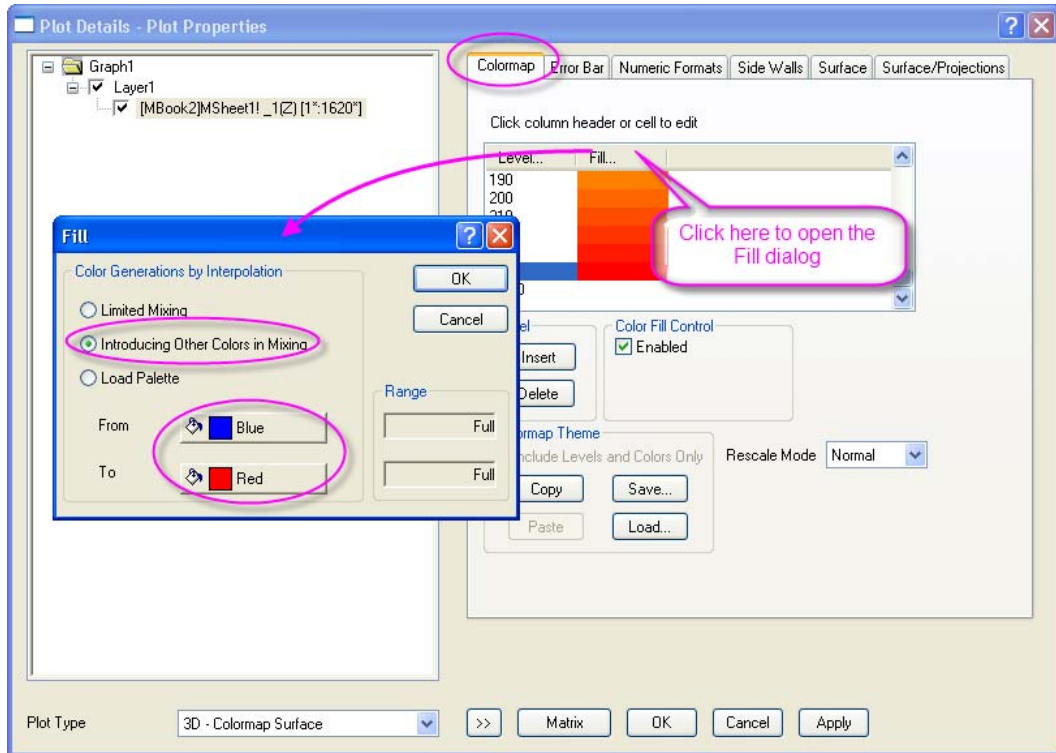
- Select **Format: Plot Properties** from the main menu to bring up the **Plot Details** dialog. Do the following in the **Plot Details** dialog:
 - In the **Surface/ Projections** tab, set the dialog options as the screenshot below.



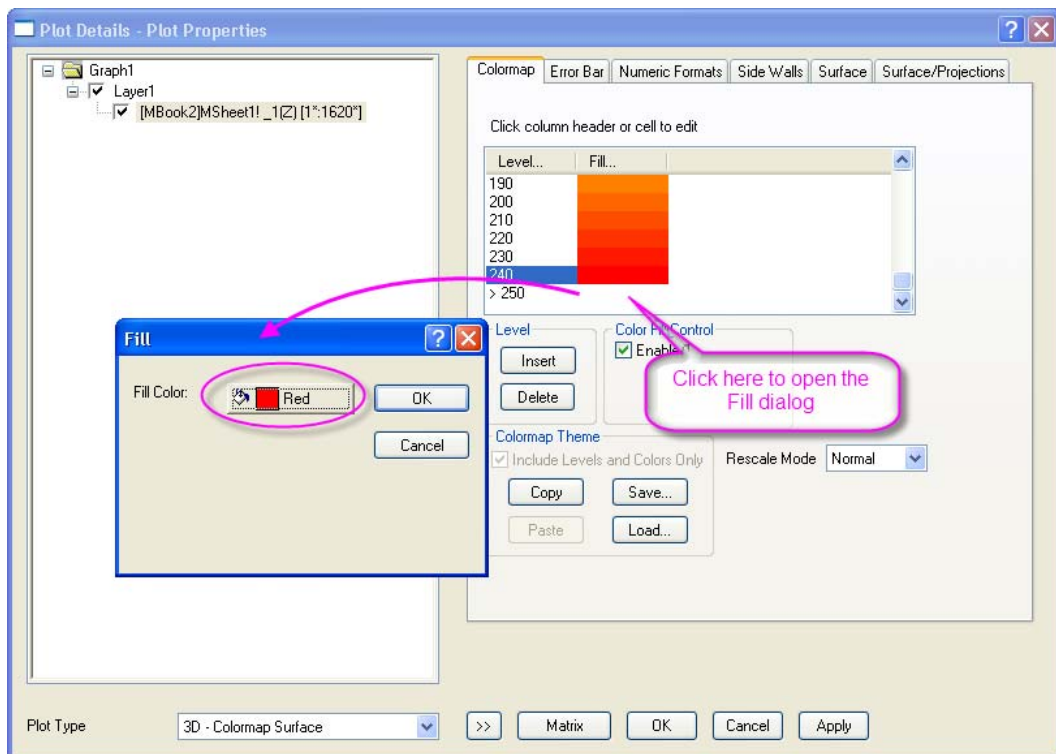
- o In the **Color Map** tab, click on **Level** heading to open the **Set Levels** dialog. Set the dialog options as the screenshot below. Then click **OK** to close the **Set Levels** dialog.



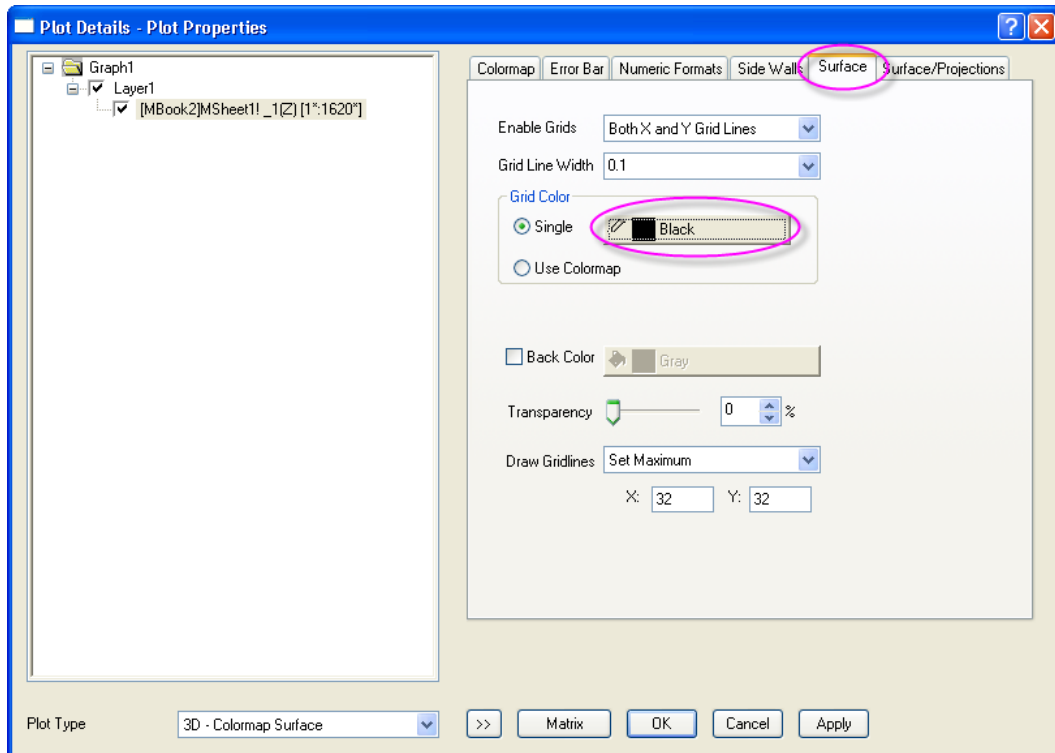
- o Click on **Fill** heading to open the **Fill** dialog. Set the dialog options as the screenshot below the Click OK



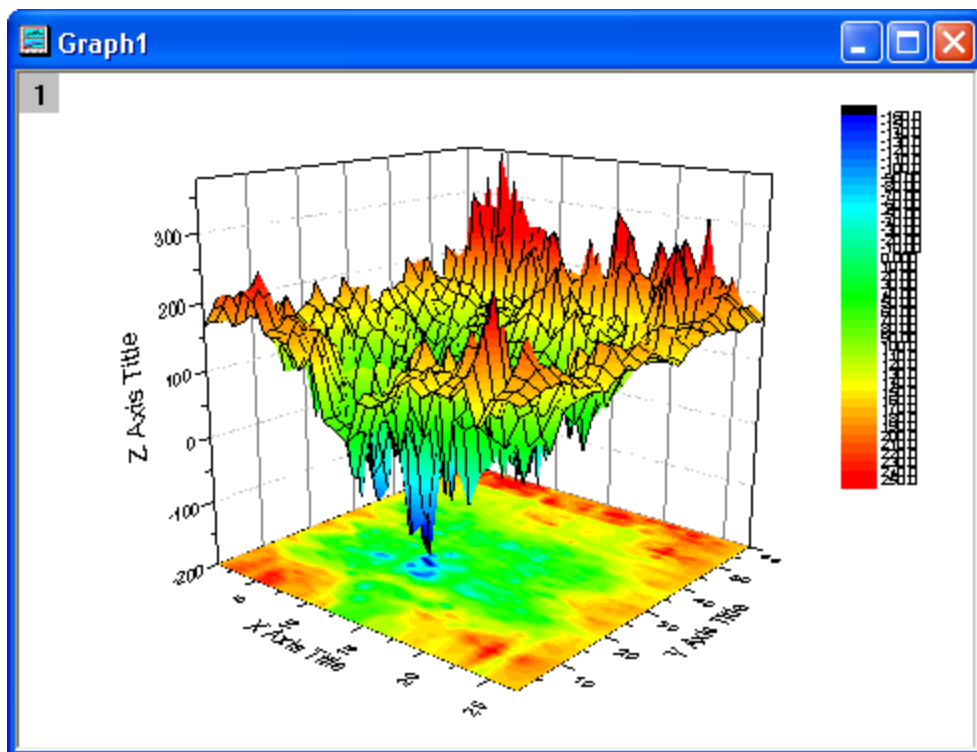
- o Click on the last row of the **Fill** column which is next to > 250 row to pop up the **Fill** dialog. Set the dialog options as the screenshot below.




- o In the **Grids** tab, set the dialog options as the screenshot below.



Click **OK** to close this dialog. The graph should look like



Sample Data

Download the **Surface_Graph.txt** file from http://www.originlab.com/ftp/graph_gallery/data/Surface_Graph.txt. Select **File: New: Matrix** from the main menu to create an empty matrix. Click the **Import Single ASCII** button  and select the file to import it.

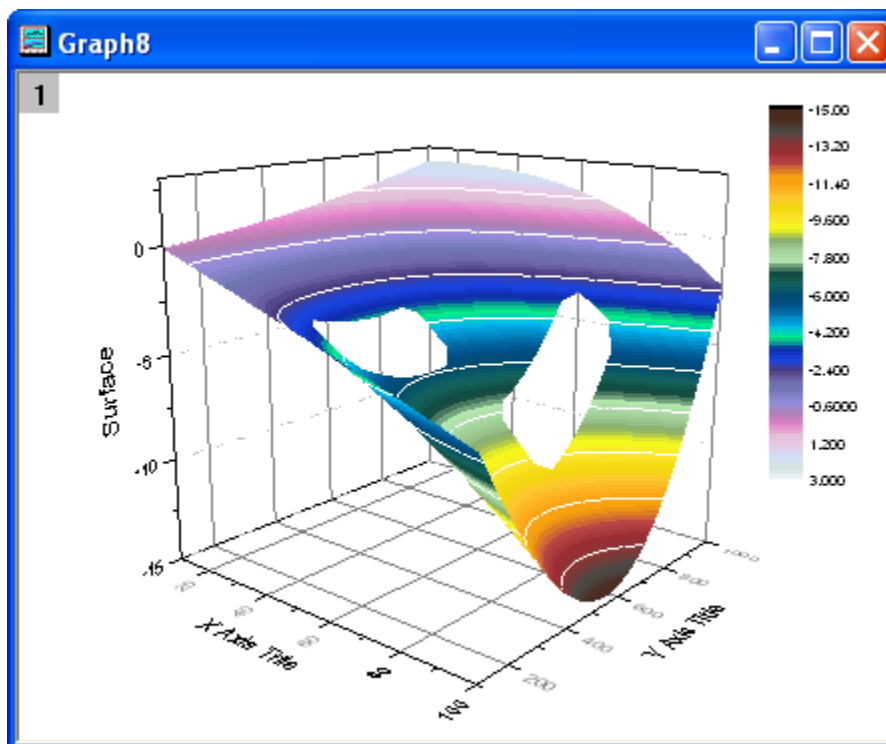
The following is part of the matrix.

166	195.3	203	174.9	229.8	217.4	174.5
171.5	211.3	217.5	179.5	228.6	233.6	181.7
169.6	198	209.5	181.4	238	220.1	178.8
167.6	201.5	215.4	187.3	252.5	225.8	185.5
171.2	175.4	202.9	181.5	229.1	182.1	195.5
177.9	178.6	206	181	213.5	179.5	193.7
178.1	176.7	212.9	174.9	181.7	218.6	190
206.8	174.2	199.7	155.2	176.1	172	178.2

3.8.5 Surface with Missing Values

Summary

In this tutorial, a 3D color map surface graph will be created from a matrix with missing values. Also, gridlines can be skipped by customization.



Minimum Origin Version Required: Origin 8.5 SR1

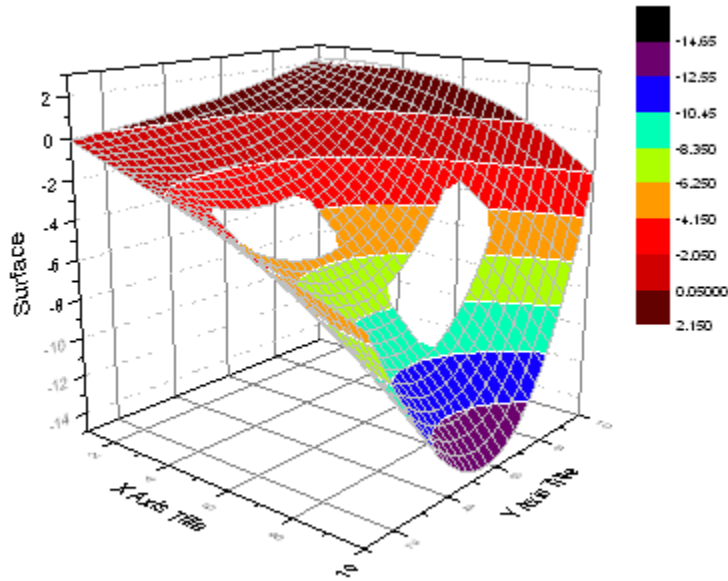
What you will learn

- Create a 3D surface plot from a matrix by ignoring missing values
- Skip gridlines of a 3D surface plot
- Hide contour lines
- Set contour levels and fill colors
- Disable axes ticks and labels

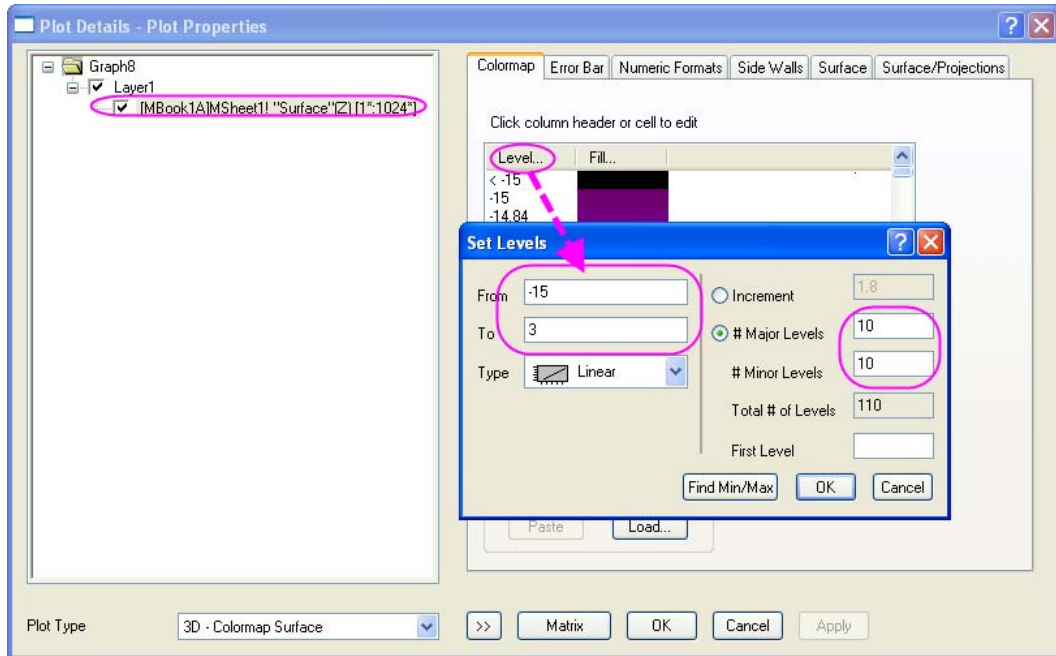
Steps

This tutorial is associated with the folder **Graphing: 3D Surface: Surface with Missing Values** of sample project: `\Samples\Samples.opj`.

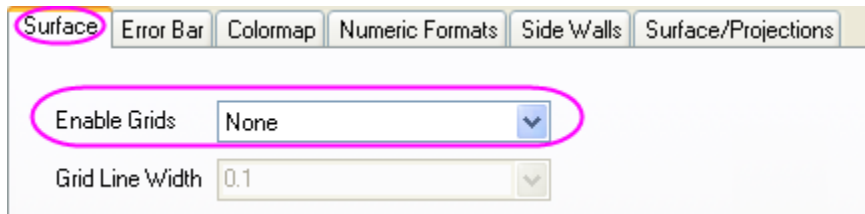
1. Open the **Graphing: 3D Surface: Surface with Missing Values** folder in the **Project Explorer**. Activate the **MBook1A** and select **Plot: 3D Surface: Color Map Surface** from the Origin main menu to create a graph. The graph should look like



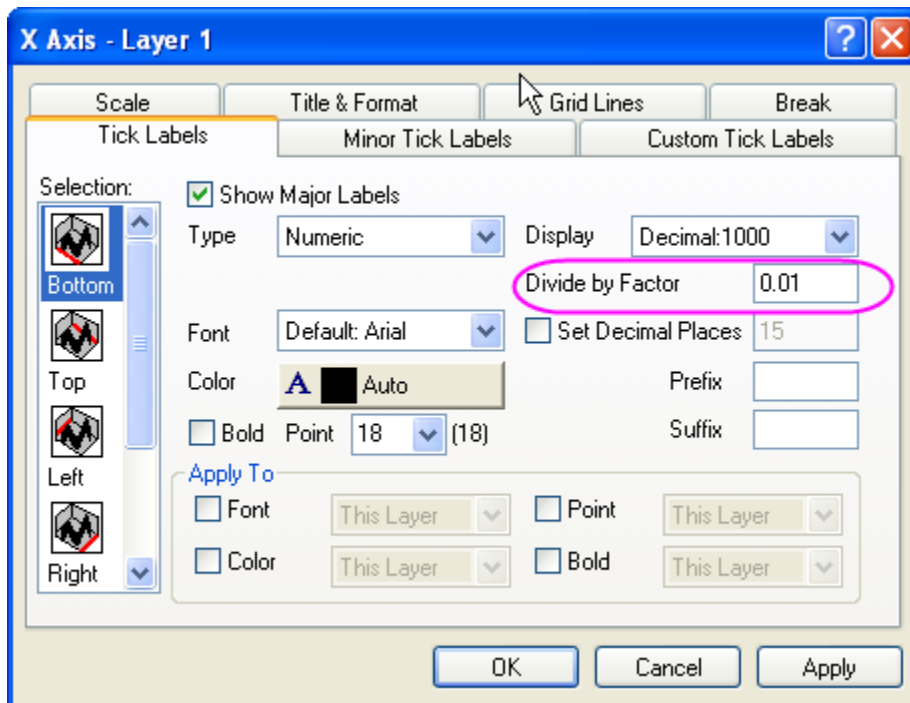
2. Double click on the graph to open the **Plot Details** dialog. Expand the tree node on the left panel to make sure that the last node is selected. Activate the **Color Map** tab on the right panel. Click the **Level** heading to open the **Set Levels** dialog and set the dialog as the following screenshots shows:



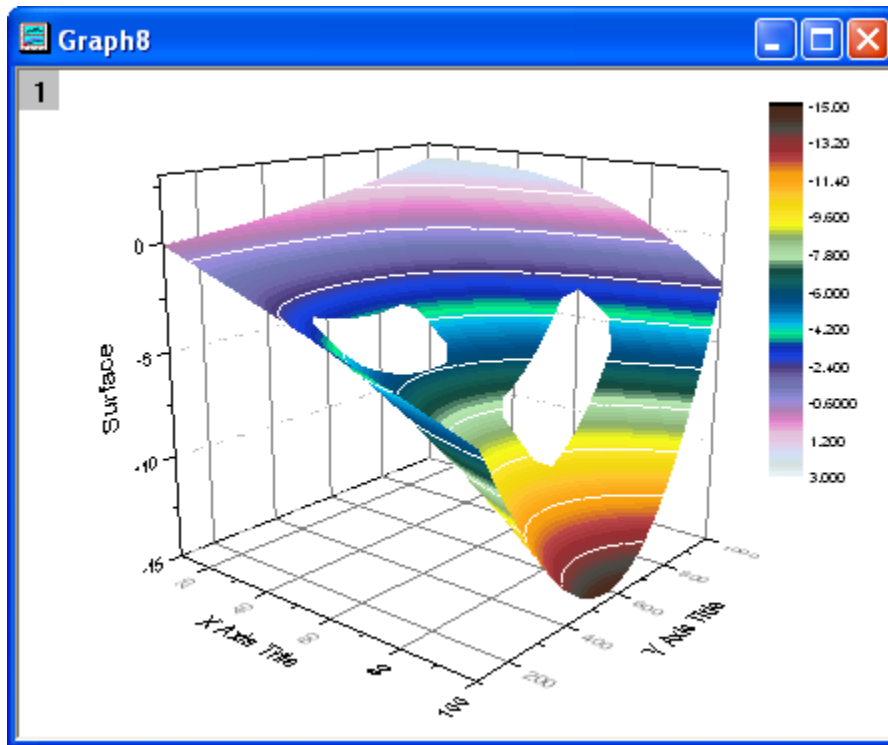
3. Click on the **Fill** heading to open the **Fill** dialog. In this dialog, choose the **Load Palette** radio and then click the **Select Palette** button to choose the **Reef** palette. And make sure the **Link to Palette** check box has been enabled. Click the **OK** button to return the **Plot Details** dialog.
4. Activate the **Surface** tab on the right panel. Set the dialog as the following screenshot shows.



5. Double-click the X axis to open the **Axis** dialog. In this dialog, go to the **Tick Labels** tab, set the **Divide by Factor** as **0.1**, then click Apply button.



6. Go to the Scale tab, set **From**, **To**, **Increment** as **10**, **100**, **20** respectively.
7. Do the same thing for Y axis: Set the **Divide by Factor** as **0.01** in **Tick Labels** tab and set **From**, **To**, **Increment** as **100**, **1000**, **200** respectively in the **Scale** tab.
8. Set **From**, **To**, **Increment** as **-15**, **3**, **5** respectively in the **Scale** tab of **Axis** dialog for Z axis, then click **OK** button. The final graph should look like



3.8.6 Creating Intersecting Surface Plots from Worksheets

Summary


Many plot types, such as 3D Surface, 3D Bars, and Contour, can be created from data contained in a matrix object or from data arranged in a block of cells in a worksheet. The latter arrangement is referred to as a **Virtual Matrix**. Whereas a typical matrix object only supports a linear mapping of X and Y coordinates, a virtual matrix supports nonlinear mapping as well. In this tutorial, we show you how to create intersecting color map surface plots from virtual matrix data.

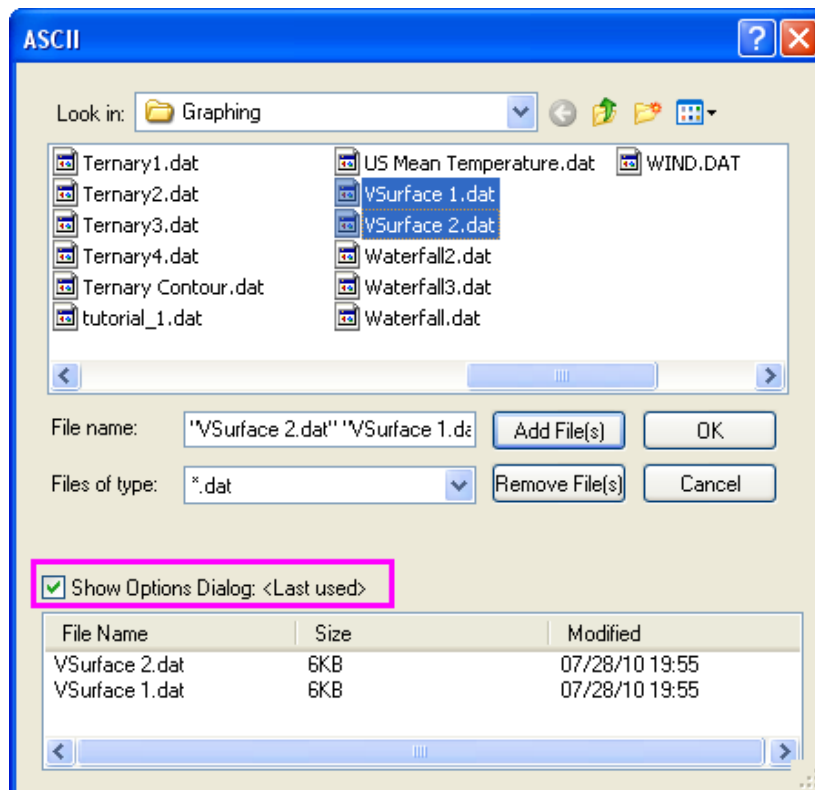
This tutorial will show you how to:

- Create a Color Map Surface plot using virtual matrix data from a worksheet
- Add one surface graph to another to create intersecting surface plots
- Customize the Color Map of one surface plot and copy and paste the settings to another plot
- Set transparency

Create Surface Plots from Virtual Matrix Data

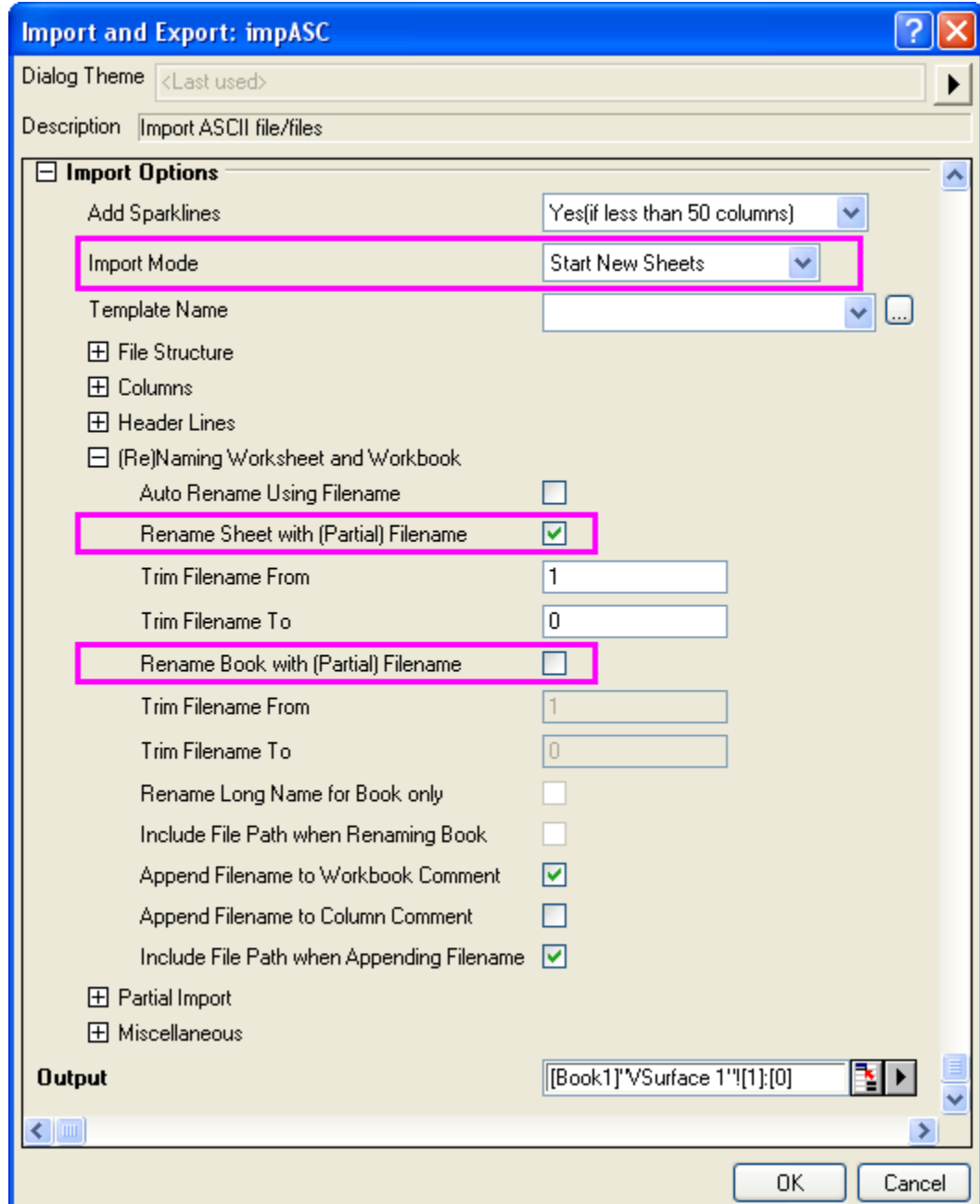
In this section, we will show you how to create surface plots from data in a worksheet.

1. Start a new project. Click the **Import Multiple ASCII** button  on the **Standard** toolbar. In the dialog that opens, navigate to the `\Samples\Graphing` folder and select `VSurface 1.dat` and `VSurface 2.dat`. Select the **Show Options Dialog** check box.

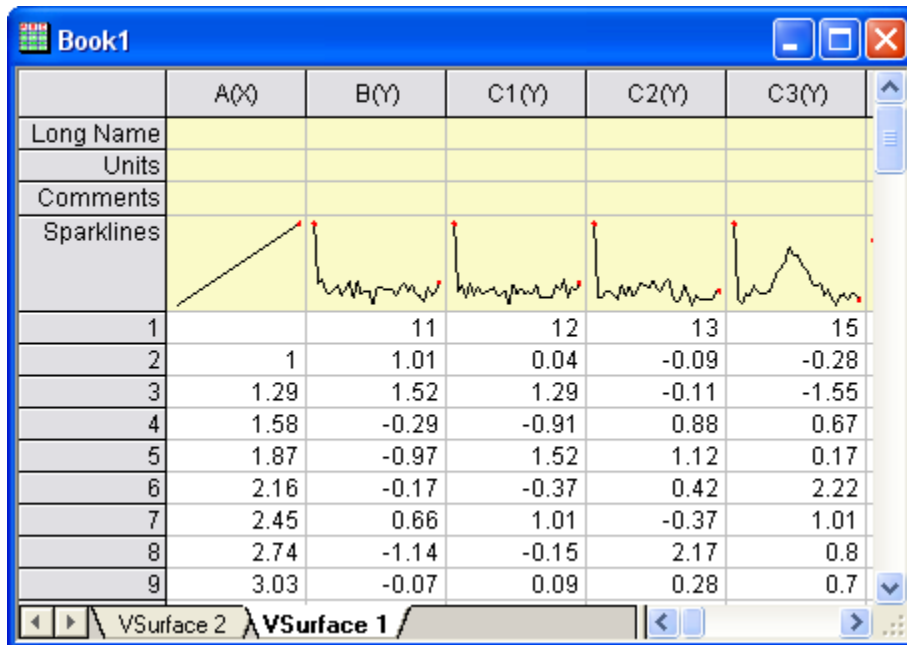


Press **OK** button.

- In the **impASC** dialog, select **Start New Sheets** from the **Import Mode** drop-down list, to import data files to different worksheets of the same workbook. Expand the **(Re)Naming Worksheet and Workbook** branch of **Import Options**. Check **Rename Sheet with (Partial) Filename** and uncheck **Rename Book with (Partial) Filename**, to rename the worksheet with the file name.

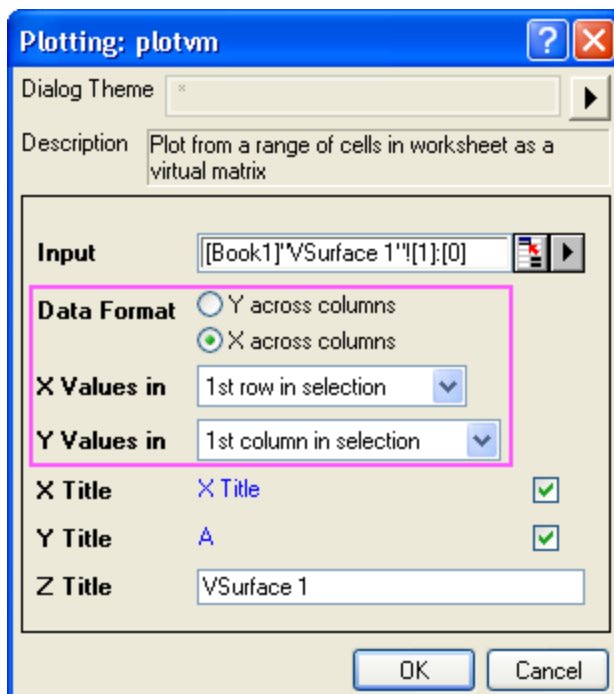


Click **OK** to import these two data files.

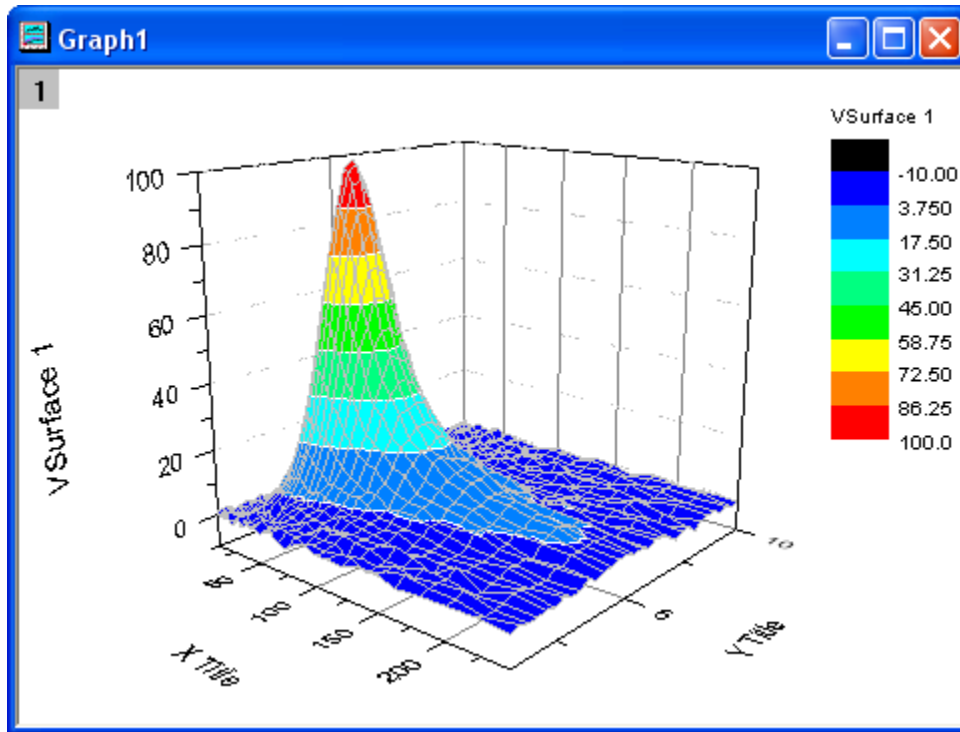


The data has X coordinate values in the top row and Y coordinate values in the first column. Note that the X coordinates have nonlinear spacing.

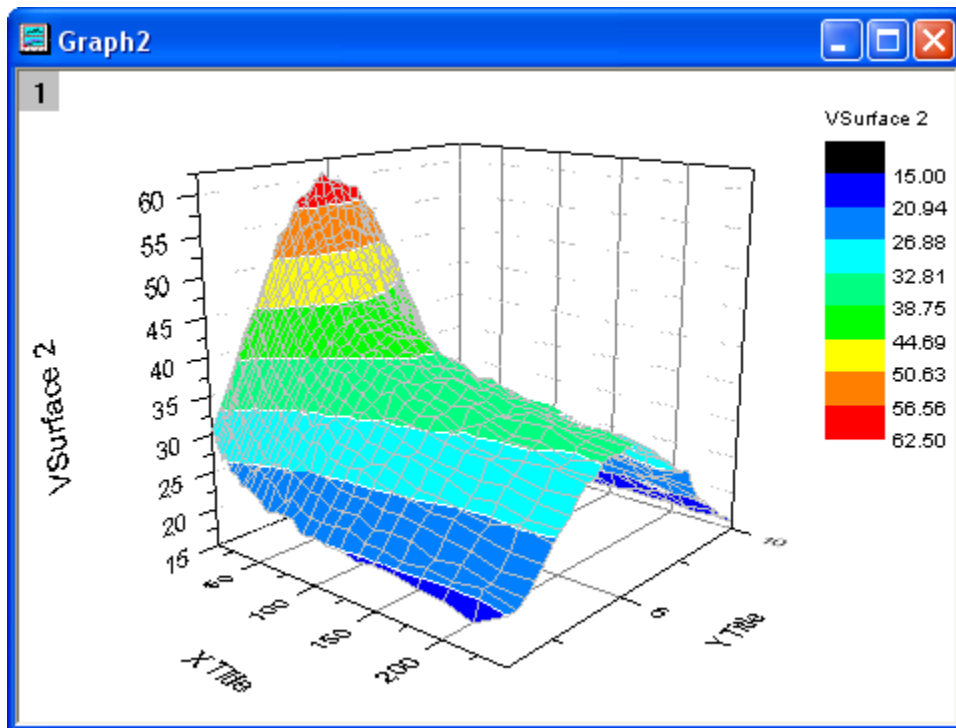
- With the "VSurface 1" sheet active, click on top left corner of the workbook to select the entire sheet. Then select the menu item **Plot: 3D Surface: Color Map Surface**. This will open the **plot_vm** dialog. This dialog opens any time a 3D or Contour plot menu item is invoked with a group of worksheet cells (a virtual matrix) selected. The dialog allows you to define where the X and the Y coordinate values are located. In the dialog, select **X across columns** from the **Data Format** drop-down list. Set the **X Values in** drop-down list to **1st row in selection** and the **Y Values in** drop-down list to **1st column in selection**:



Click **OK** to plot a Color Map Surface plot.



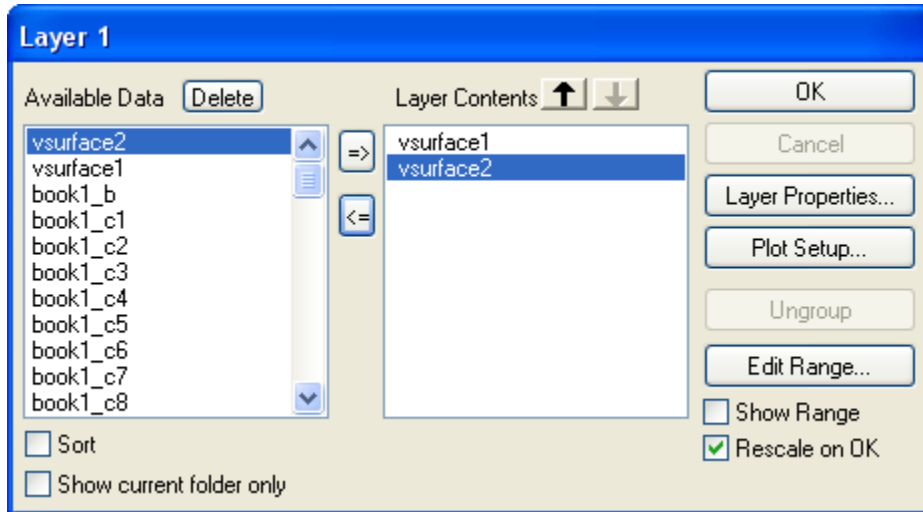
- Repeat the last step to plot another Color Map Surface plot using data from the VSurface 2 worksheet.



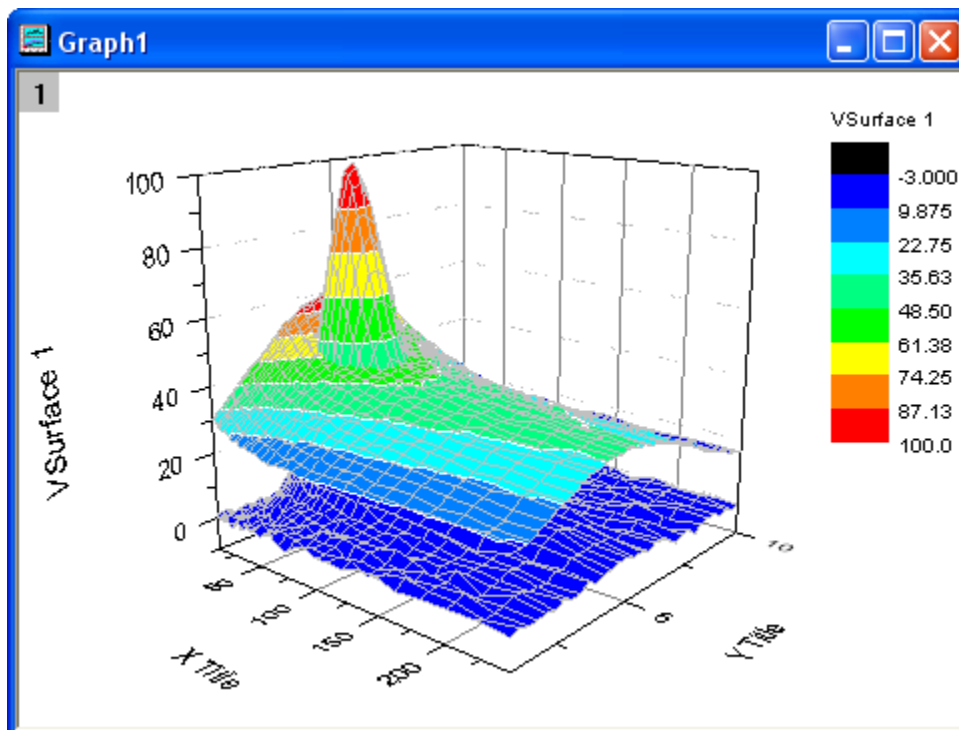
Add another Surface Plot to Layer

In this section, we will show you how to add a surface plot to a graph layer already containing another surface plot.

5. Activate Graph1. Right-click on the Layer 1 icon on the top left corner of the graph and select **Layer Contents** from the context menu. In the **Layer** dialog that opens, select vsurface2 in the Available Data box and add it to the Layer Contents box.

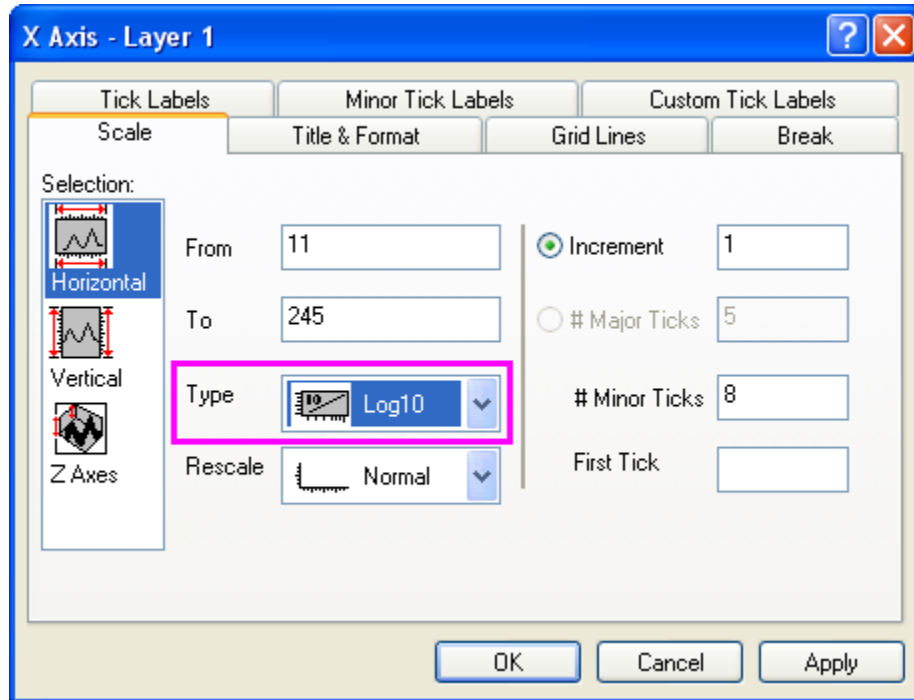


Click **OK** to add the second surface.



Note that Origin displays the two surfaces with proper intersection.

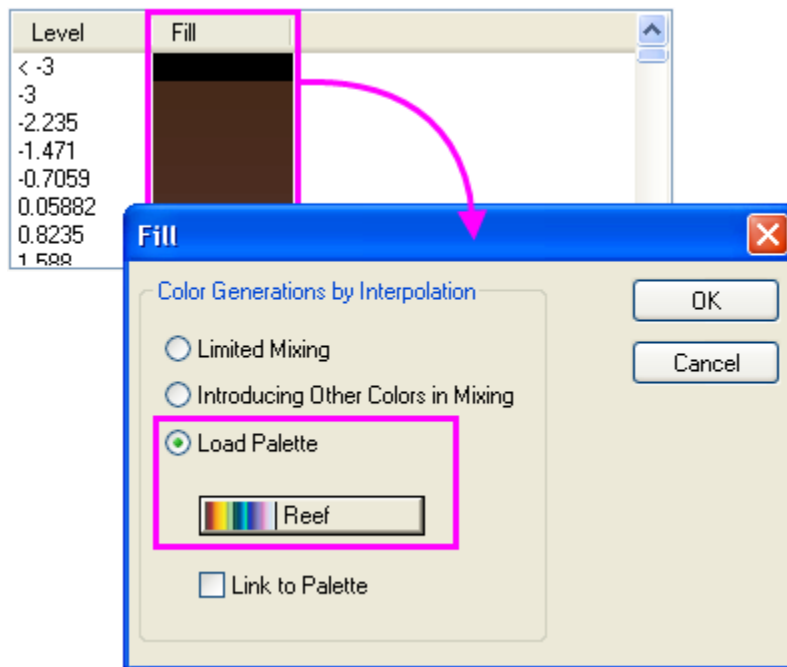
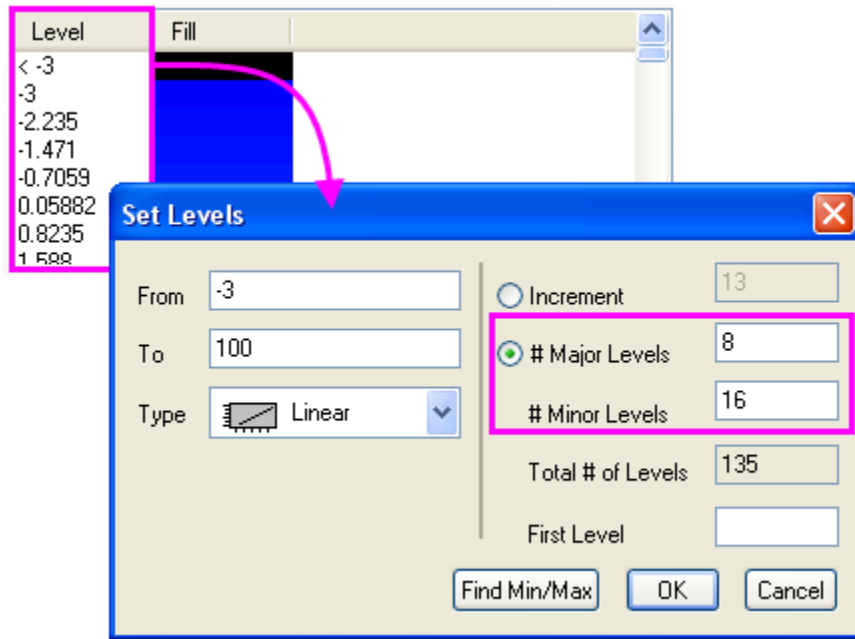
- Double-click on X axis to open the **X Axis** dialog. On the **Scale** tab, select **Log10** from the **Type** drop-down list.



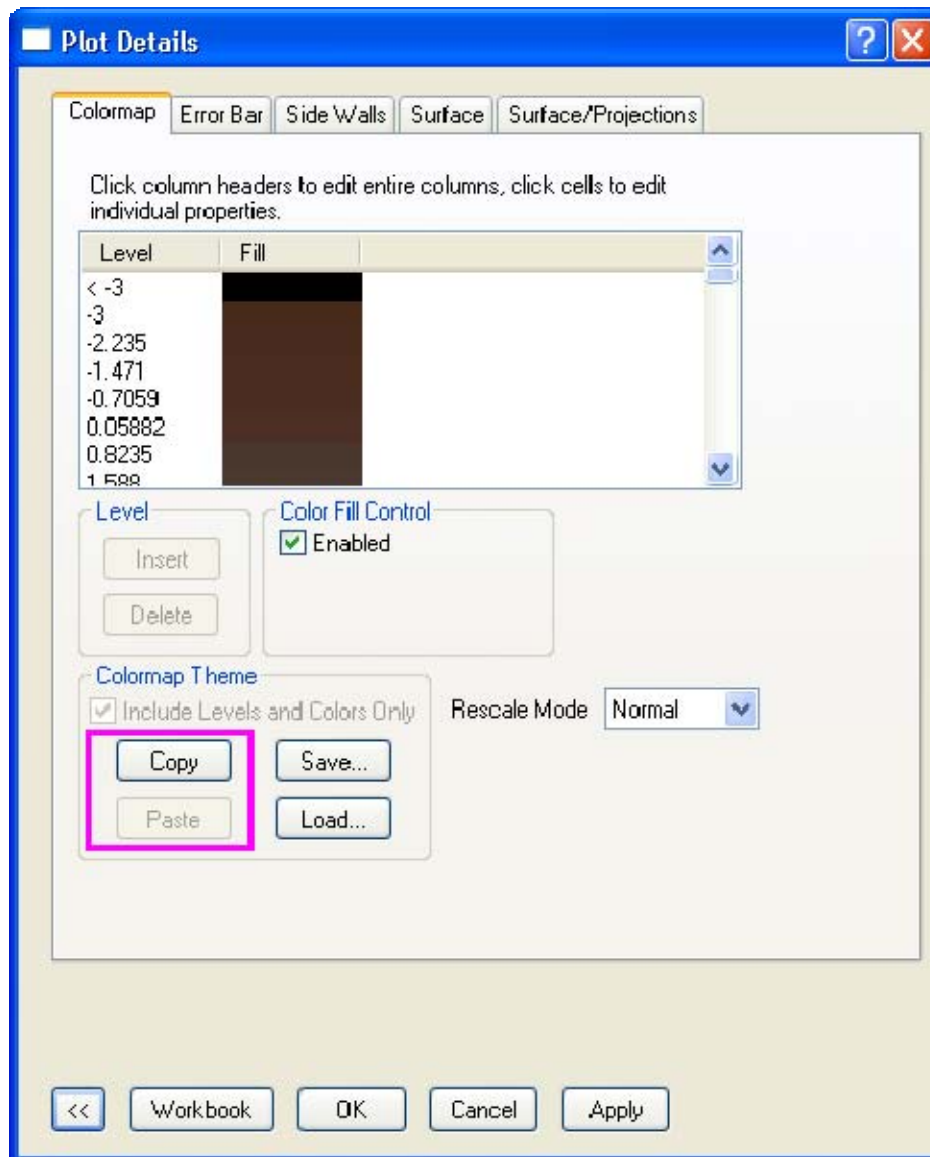
Customize Color Map and Copy-Paste to another Plot

In this section, we will customize the Color Map settings of one plot, and then copy and paste those settings to the other plot.

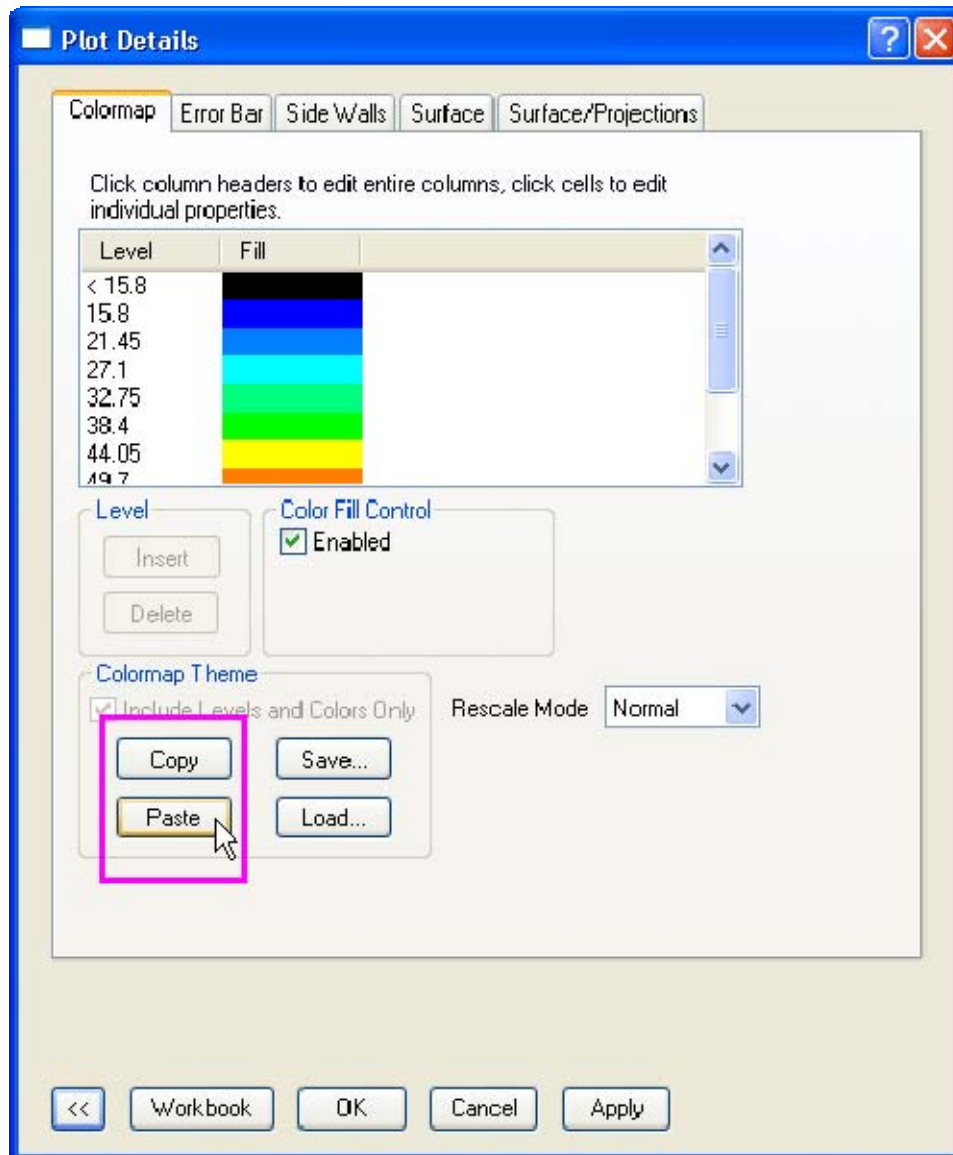
- Double-click on the graph to open the **Plot Details** dialog. On the left panel, expand the **Layer1** node and select **VSurface 1**. Then switch to the **Color Map** tab and set **Level** and **Fill** as below:



8. Now press the **Copy** button under the **Color Map Theme** group on this tab to copy the level and color settings to the clipboard.



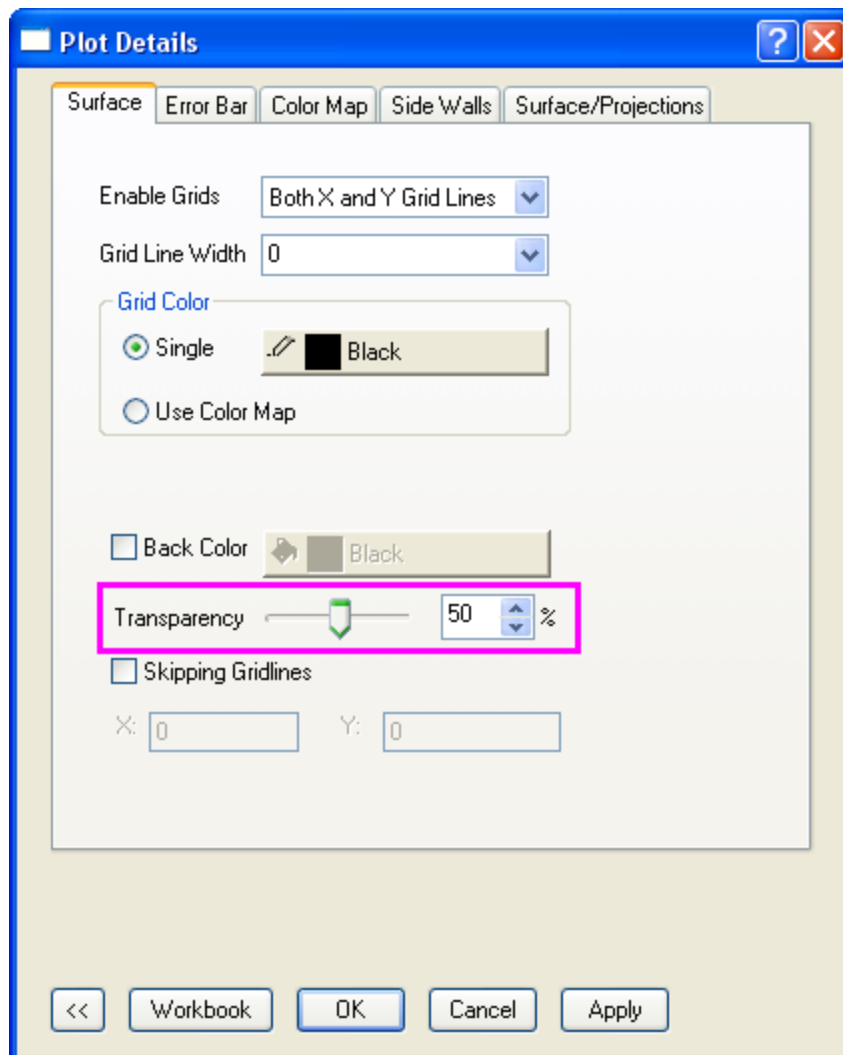
- Switch to the **VSurface 2** plot in the left panel of Plot Details, and on the right panel, select the corresponding Color Map tab. Then press the **Paste** button to paste the settings from the clipboard to the second plot.



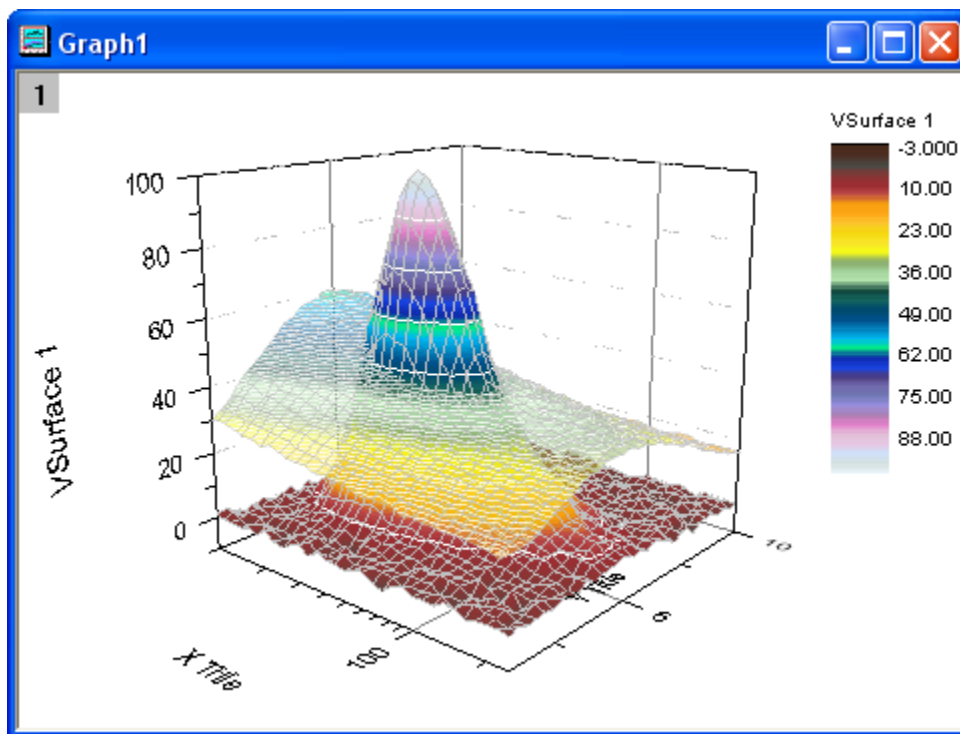
Set Surface Transparency

We will now set the transparency of the second surface so that parts of the first surface are visible in this intersection plot.

10. Switch to the **Surface** tab and move the **Transparency** slider to 50%.



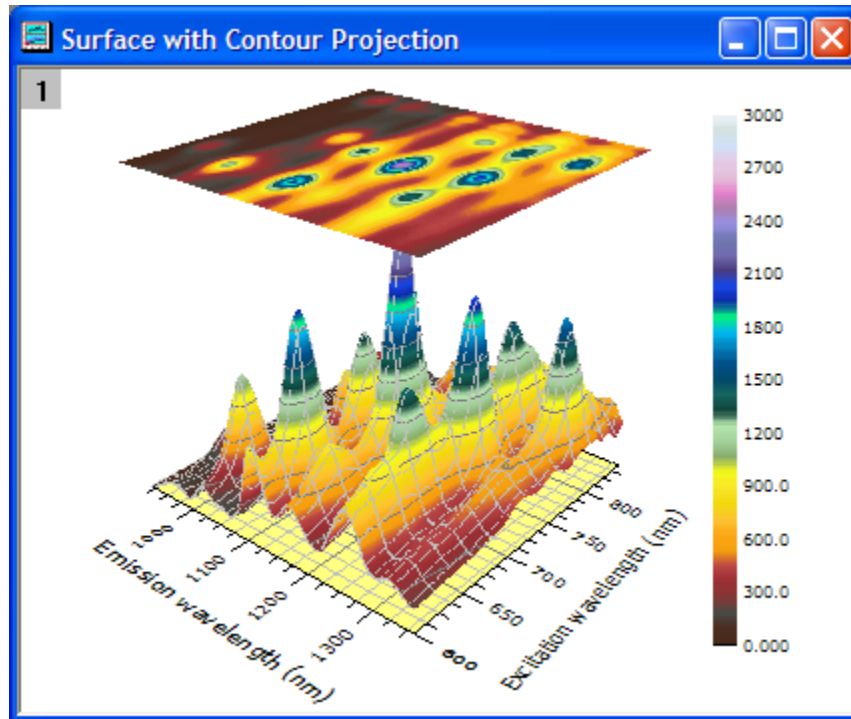
The final graph should look like the image below:



3.8.7 Surface with Contour Projection

Summary

Origin's Colormap Surface provides full controls for customizing its appearance, including surface color, projection contour, grid lines, and plane color.



Minimum Origin Version Required: Origin 8.1 SR6

What you will learn


- Create Colormap Surface from matrix.
- Change color mapping level and color palette.
- Show contour projection.
- Customize planes and axes.

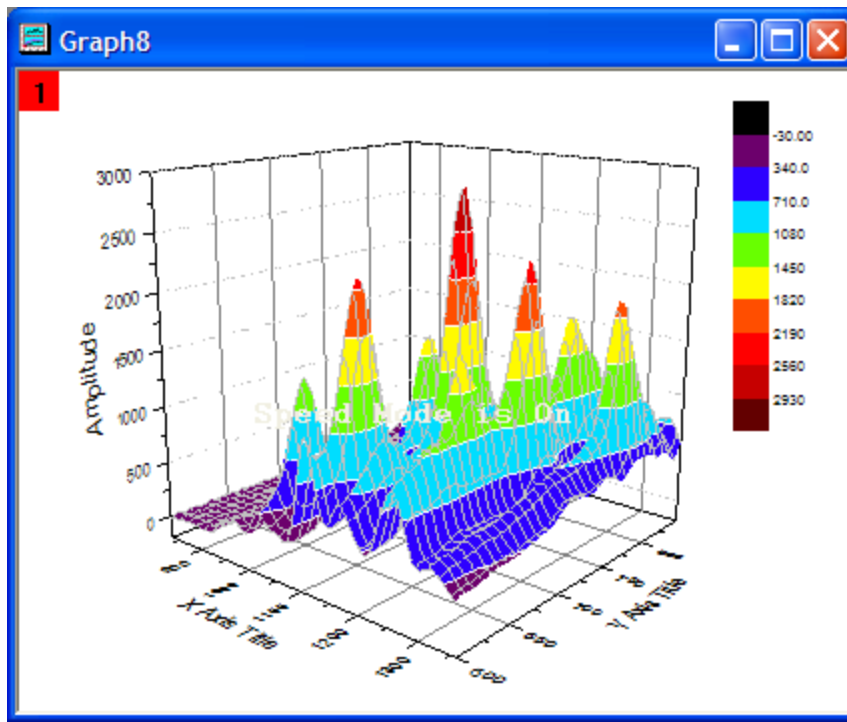
Steps

This tutorial is associated with the folder **Surface with Contour Projection** in the **Sample Graphs** project (\Samples\Samples.opj).

1. In the Project Browser, browse to the \Samples\Graphing\3D Surface\Surface with **Contour Projection** folder. Select the **Mbook2G** window to make active. This is a matrix book containing data in Image Mode. To view this data in numerical form, click **View** on the main menu, and then click **Data Mode**.

- On the main menu, click **Plot**, point to **3D Surface**, and then click **Colormap Surface**.

(Alternatively, you may click the Colormap Surface button  on the **3D and Contour Graphs** toolbar.) This will generate a colormap surface plot.

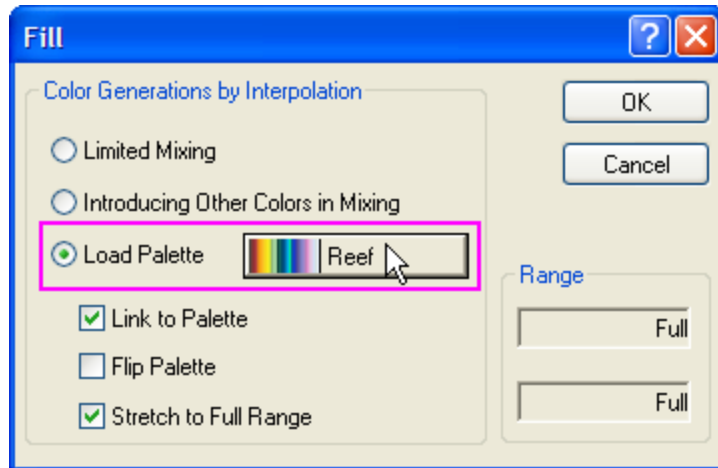


Note that speed mode is on.

- First we need to customize our colormap surface. Double-click anywhere on the surface plot to open the **Plot Details** dialog box. Select the **Colormap** tab. Click the column header **Level?** to open the **Set Levels** dialog box. Set the levels as shown below:

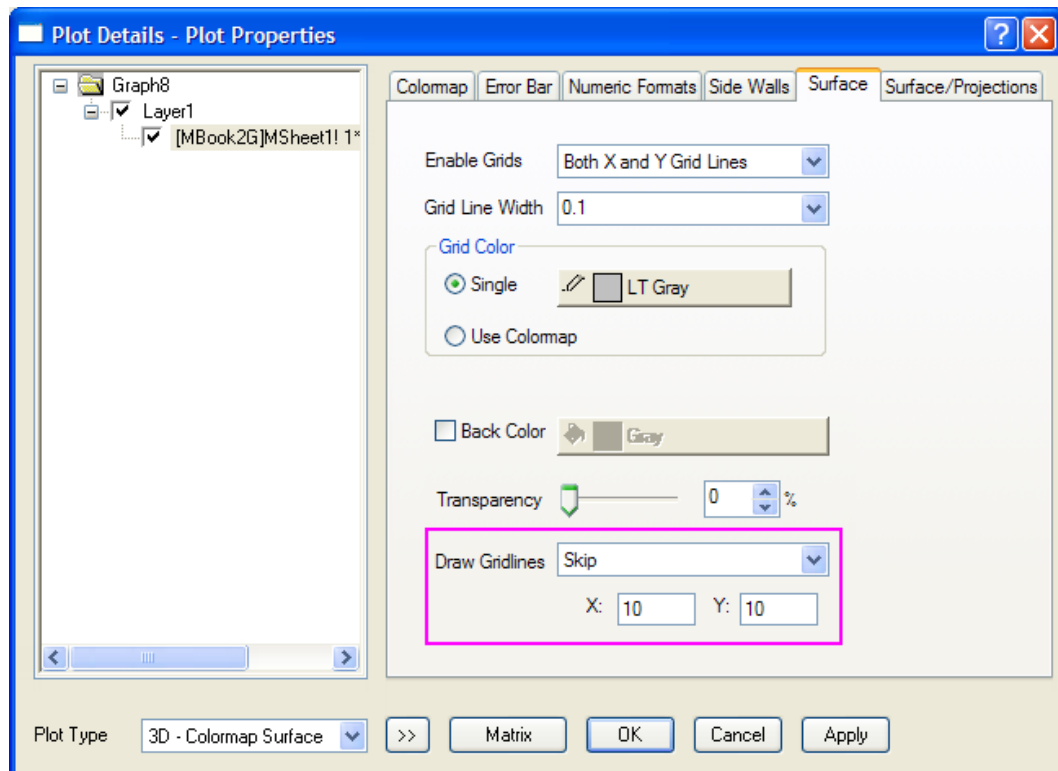
Click **OK** to close the **Set Levels** dialog box.

Click the column header **Fill?** to open the **Fill** dialog box. Change the color palette from **Rainbow** to **Reef**.

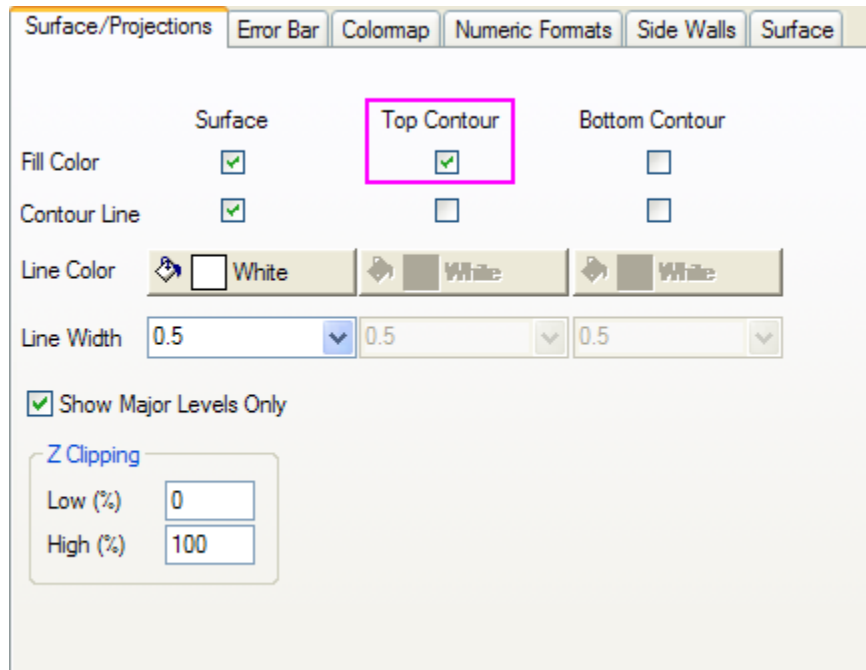


Click **OK** to close the **Fill** dialog box.

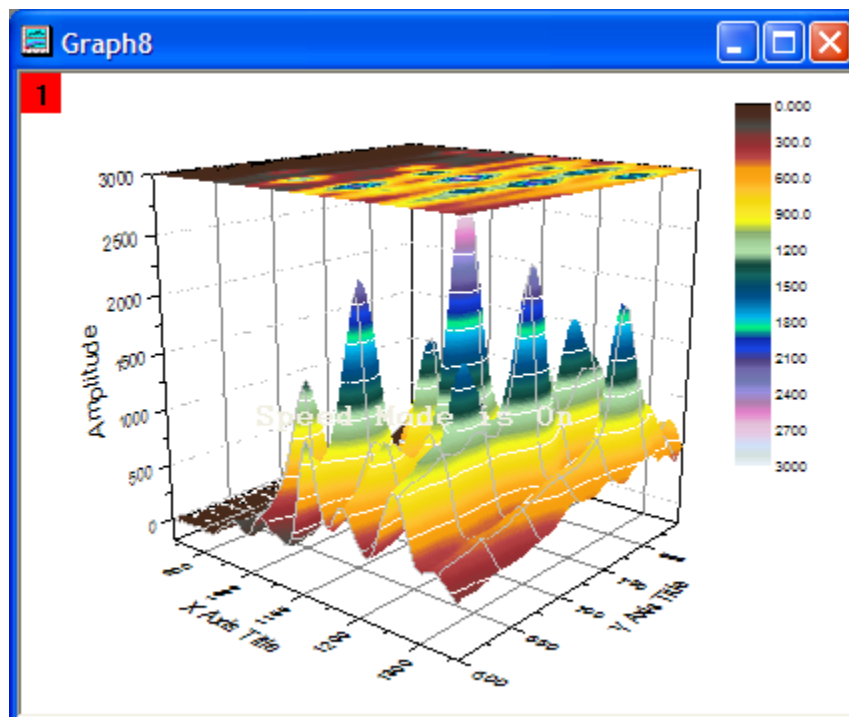
4. Select the **Surface** tab. In the **Draw Guidelines** menu, select **Skip**. Type **10** in both the **X** and **Y** fields. This will draw one gridline for every ten X or Y values.



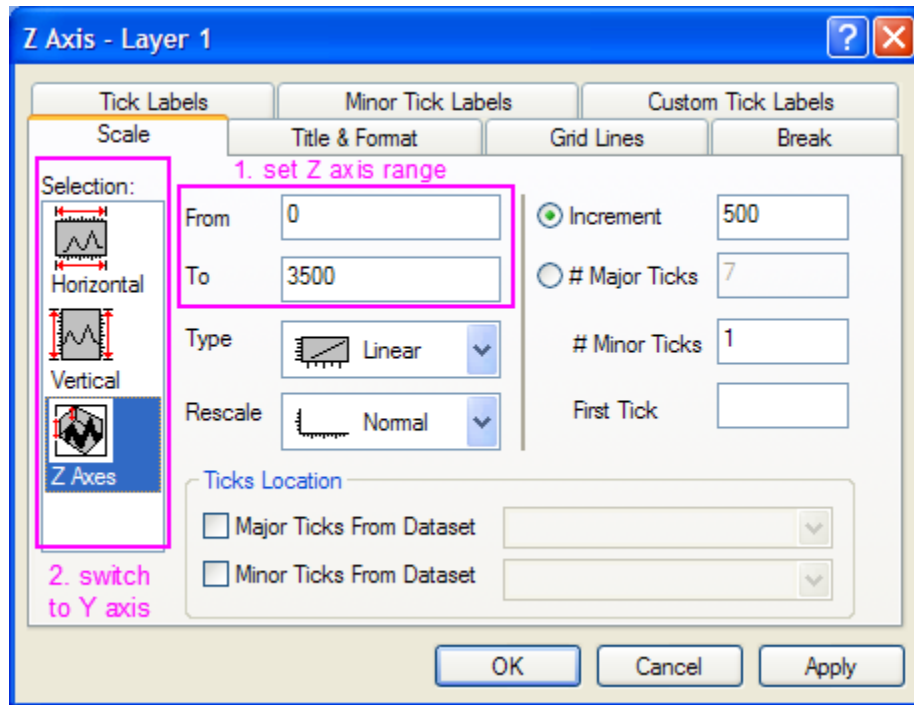
5. Next we need to show the contour projection on the surface. Select the **Surface/Projections** tab. Select the check box in the **Fill Color** row and **Top Contour** column.



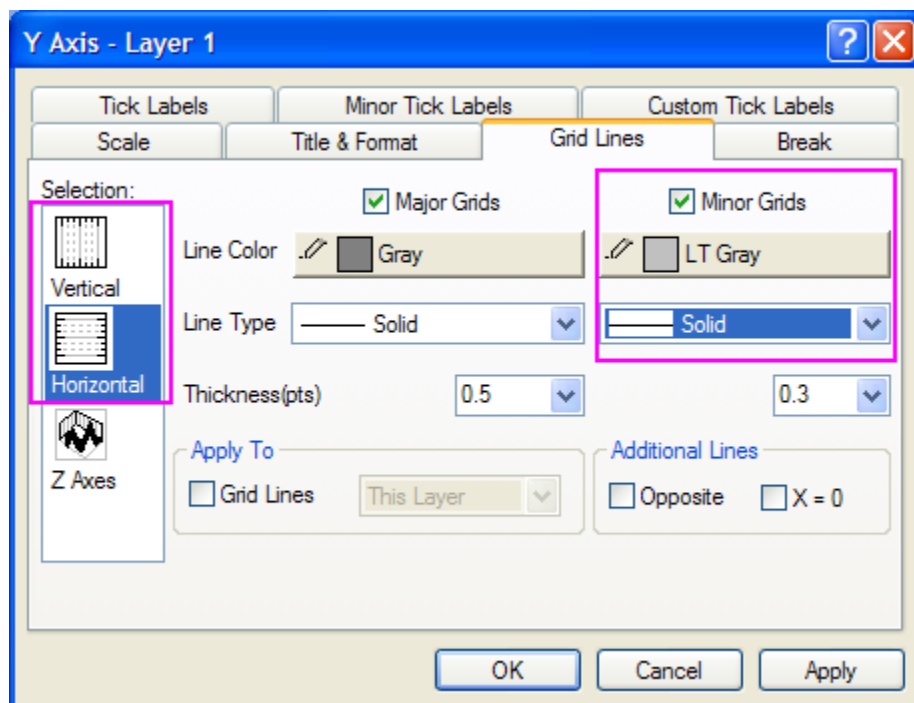
Click **OK** to close the dialog box.



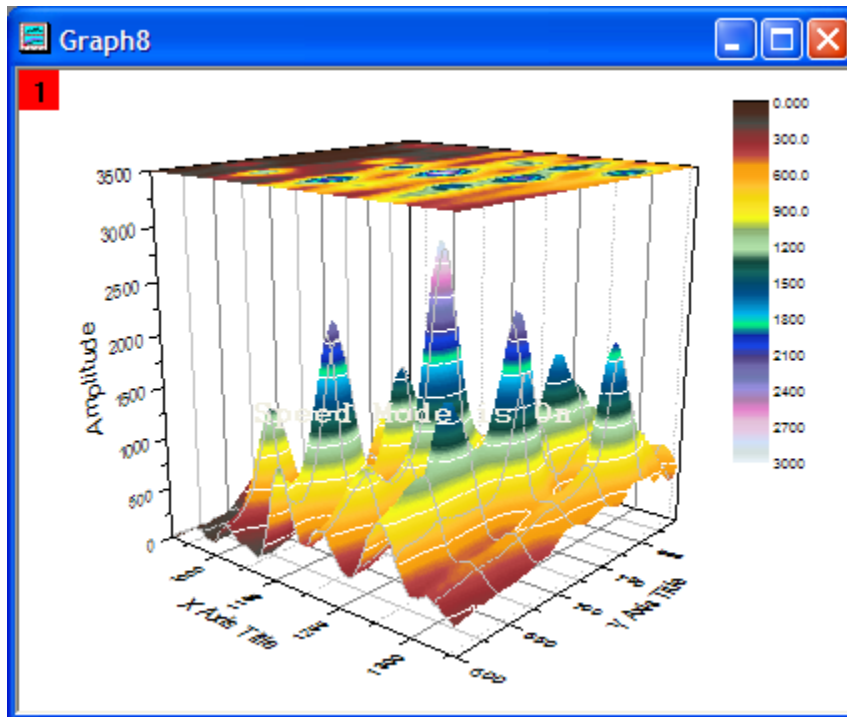
- The following steps will show you how to customize planes and axes. On the graph, double-click the Z-axis to open the **Z Axis** dialog box. Select the **Scale** tab. Type **0** in the **From** field, and **3500** in the **To** field. Select **Horizontal** from the **Selection** column on the left side of the dialog box. Type **920** in the **From** field.



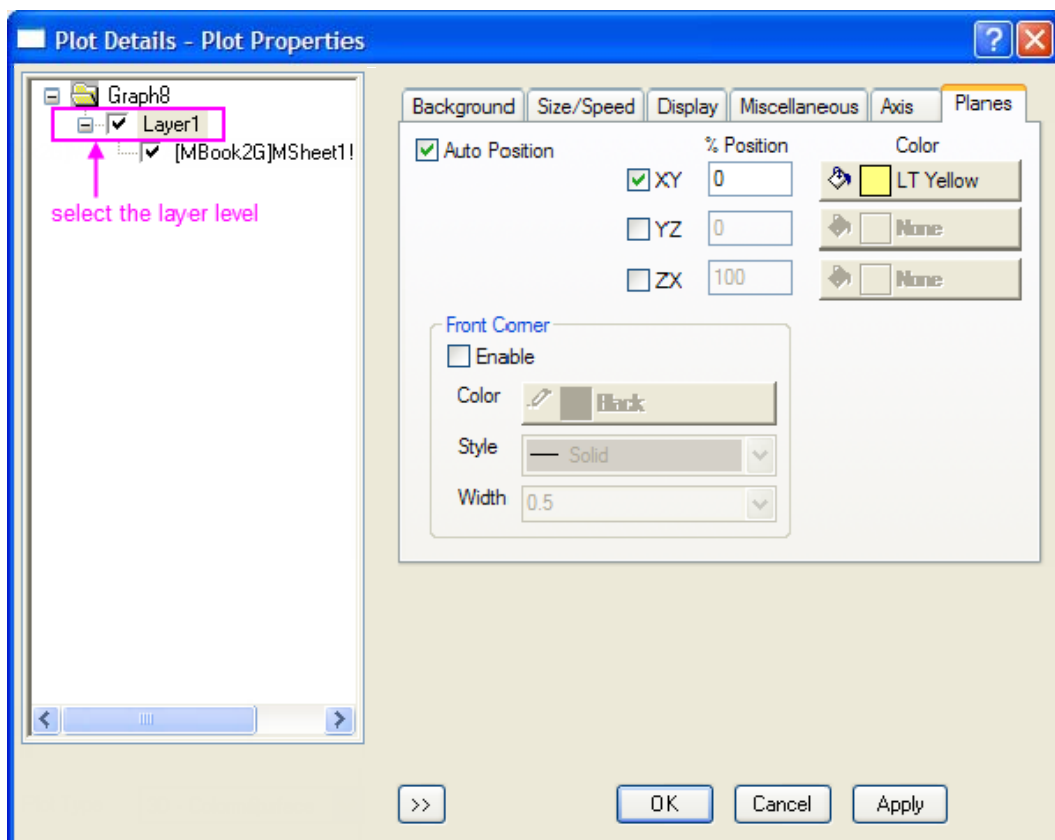
7. Select the **Grid Lines** tab. Select **Vertical** from the **Selection** menu on the left side of the dialog box. Select the **Minor Grids** check box. Select **Horizontal** from the **Selection** menu and select the **Minor Grids** check box. Change the minor grids **Line Type** to **Solid**.



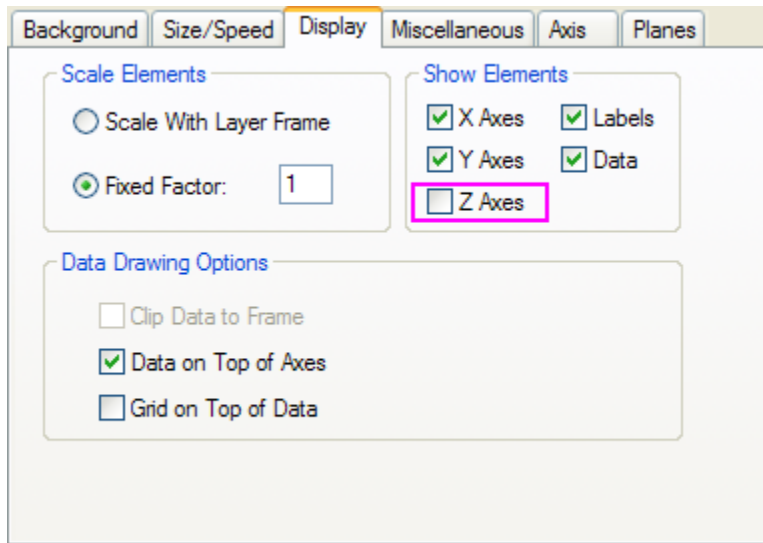
Select **Z Axes** from the **Selection** menu. Clear the **Major Grids** check box. Click **OK** to close the dialog box.



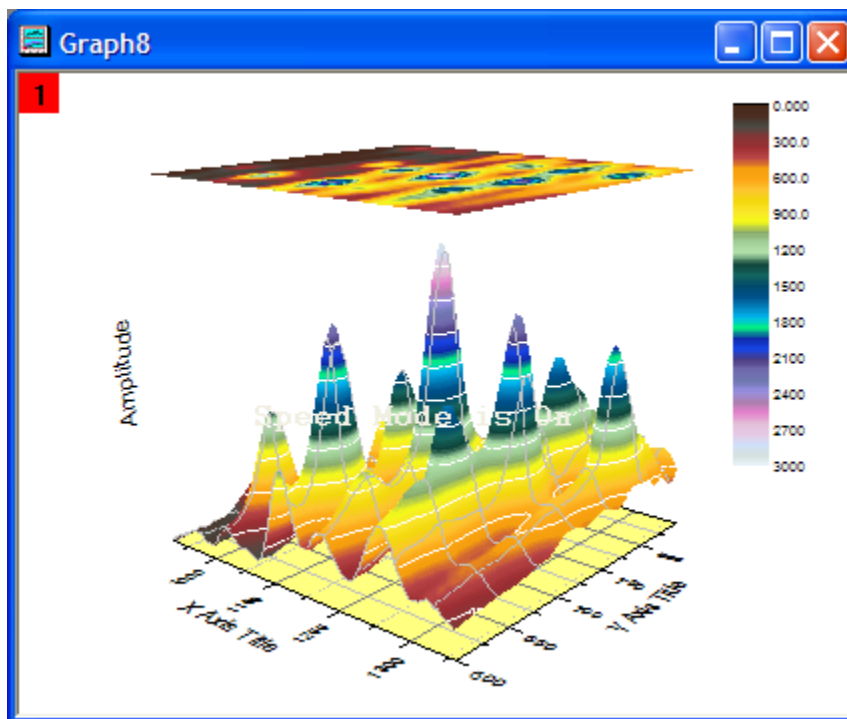
8. Double-click the surface plot to open the **Plot Details** dialog box. Select **Layer 1** from the browser in the left panel. Select the **Planes** tab in the right panel. Click the color select button for the **XY** plane, and select **LT Yellow**. Clear the **YZ** and **ZX** check boxes.



9. Select the **Display** tab. In the **Show Elements** group, clear the **Z Axes** check box.

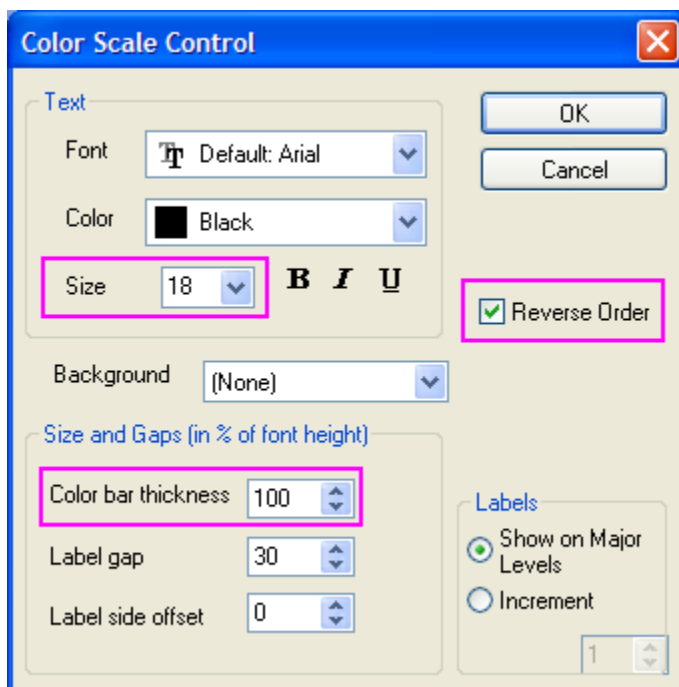


Click **OK** to close the **Plot Details** dialog box.

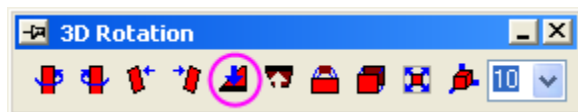


In the graph window, select the Z axis title ("Amplitude" in our example). Press **DELETE** to delete the Z axis title.

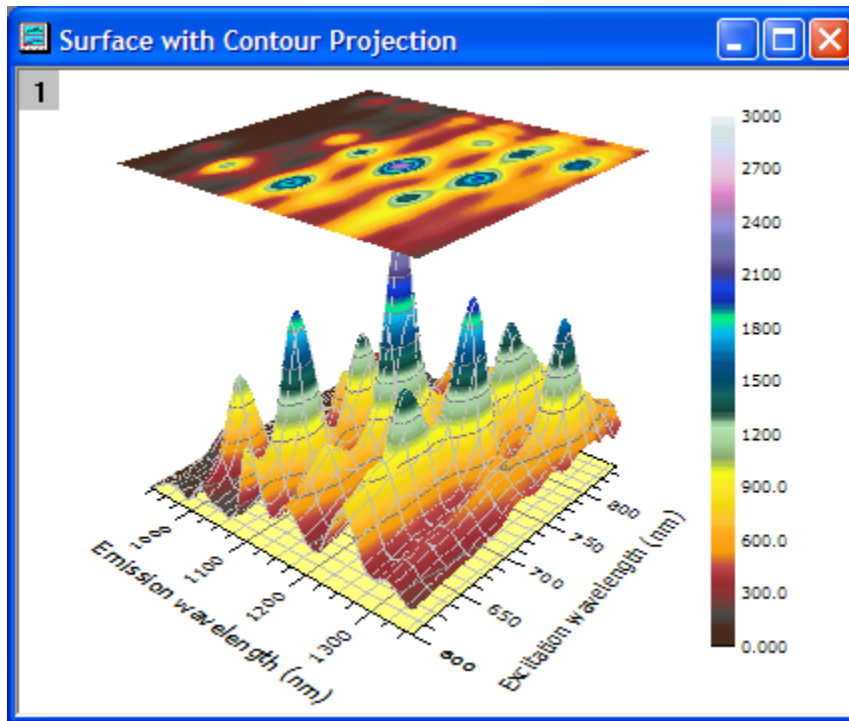
10. Double-click anywhere on the color scale to open the **Color Scale Control** dialog box. Select the **Reverse Order** check box. Change text **Size** to **18** and **Color bar thickness** to **100**.



11. Select the **MBook2G** window to make it active. On the main menu, click **Matrix**, then click **Set Dimension/Labels....** In the **Matrix Dimension and Labels** dialog box, select the **X Labels** tab and type "Emission Wavelength" in the **Long Name** field and "nm" in the **Units** field. Select the **Y Labels** tab and type "Excitation Wavelength" in the **Long Name** field and "nm" in the **Units** field. Click **OK** to close the dialog box. Select the graph window to make it active. On the main menu, select **Window**, then click **Refresh** to refresh your graph image.
12. Use the **Tilt Down** button and other rotation buttons on the **3D Rotation** toolbar to improve your graph's perspective.



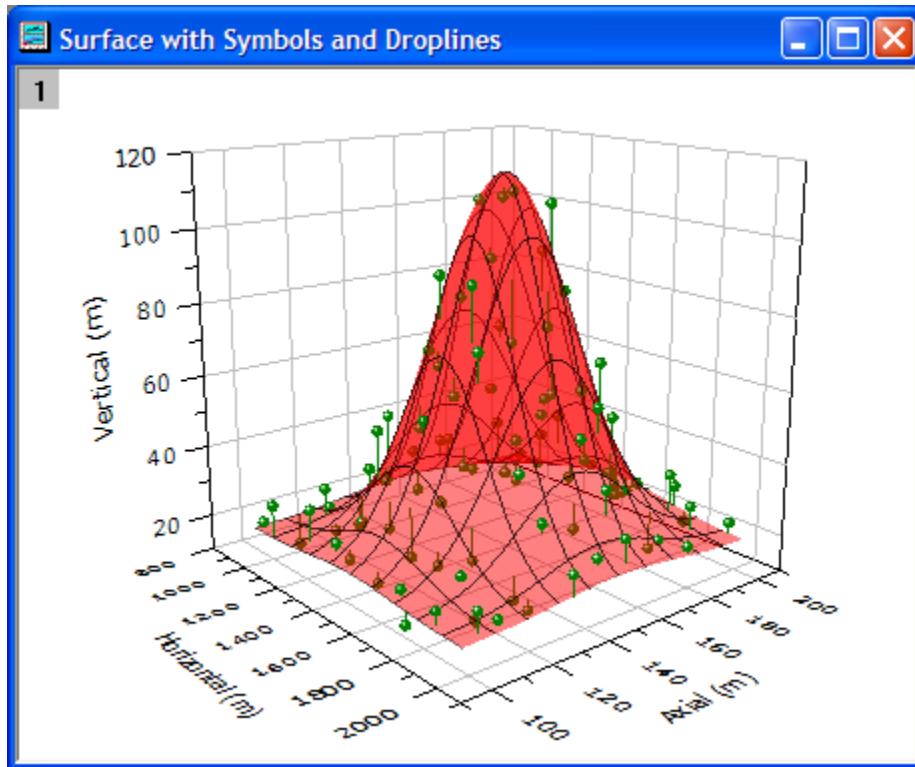
Your final graph should look like this:



3.8.8 Surface with Symbols and Droplines

Summary

Origin supports transparency in most graph types. This feature permits the visibility of parts of a graph that would otherwise be obscured by overlapping plots. In this tutorial, we will add a fitted surface to a scatter plot, drop lines from data points to the surface, and set transparency to make data points behind the surface visible.



Minimum Origin Version Required: Origin 8.5 SR0

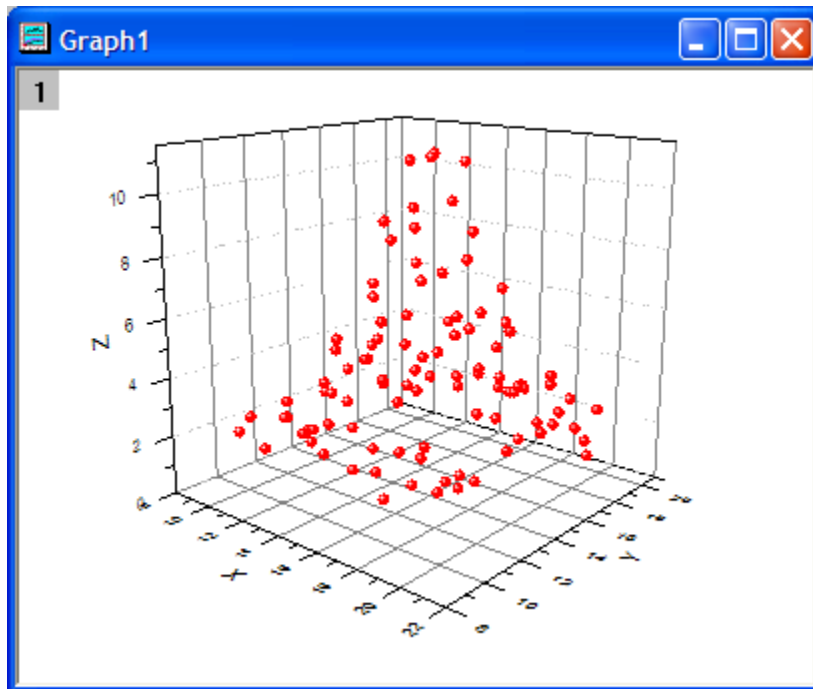
What you will learn

- Create a 3D scatter plot from worksheet data and add a color fill surface to it.
- Draw droplines from data points to the surface.
- Set surface transparency.

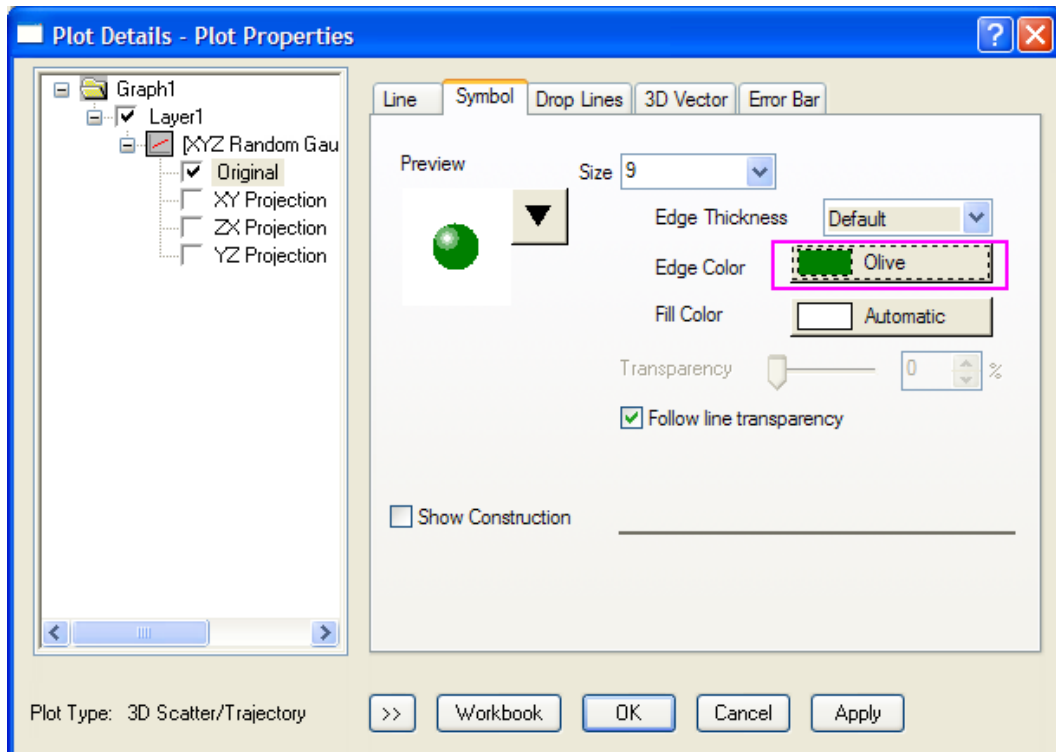
Steps

This tutorial is associated with the folder `/Samples/Graphing/3D Surface/Surface with Symbols and Droplines/` in the **Sample Graphs** project (`..\Samples\Samples.opj`).

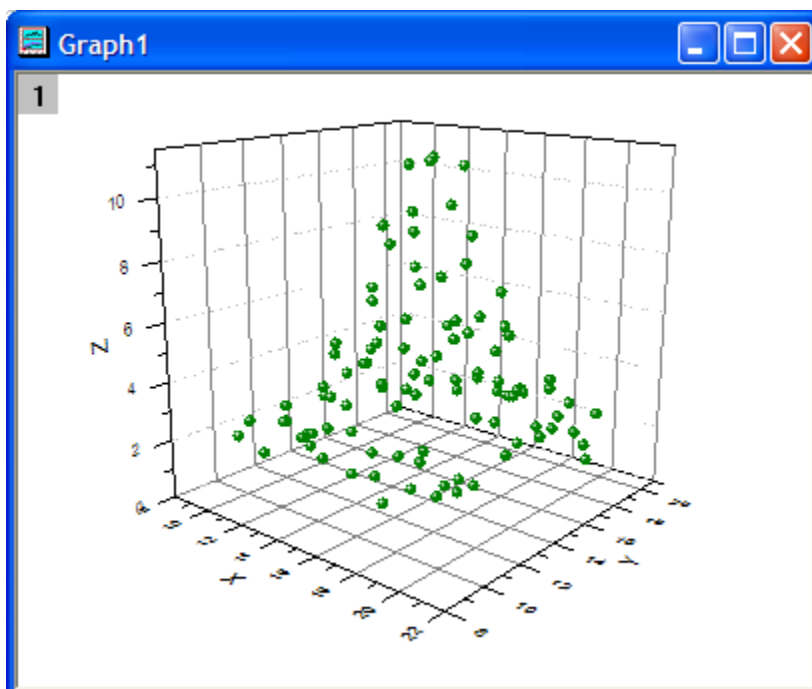
1. In Project Explorer, browse to the `\Samples\Graphing\3D Surface\Surface with Symbols and Droplines` folder. (Or select `Window : Folders : /Samples/Graphing/3D Surface/Surface with Symbols and Droplines/` from the Origin menu.)
2. Select the **XYZRandomGaus** worksheet window to make it active. Click on **C(Y)** to select all of column C. Right-click the selected column and choose **Set As : Z**. On the main menu, click **Plot** and choose **3D XYZ : 3D Scatter**. This will generate a 3D scatter plot.



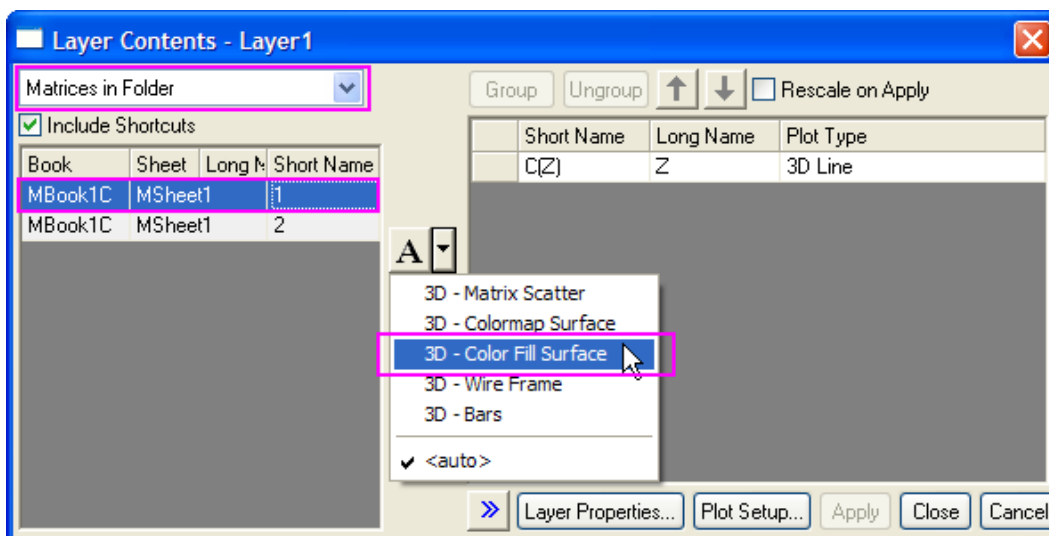
- Double-click the scatter plot to open the **Plot Details** dialog box. In the tree in the left panel, expand all items and select **Original**. (Or choose Format : Plot Properties to go directly to Original.) In the right panel, select the **Symbol** tab. Click on the **Edge Color** button, point to **Individual Color**, and select **Olive**.




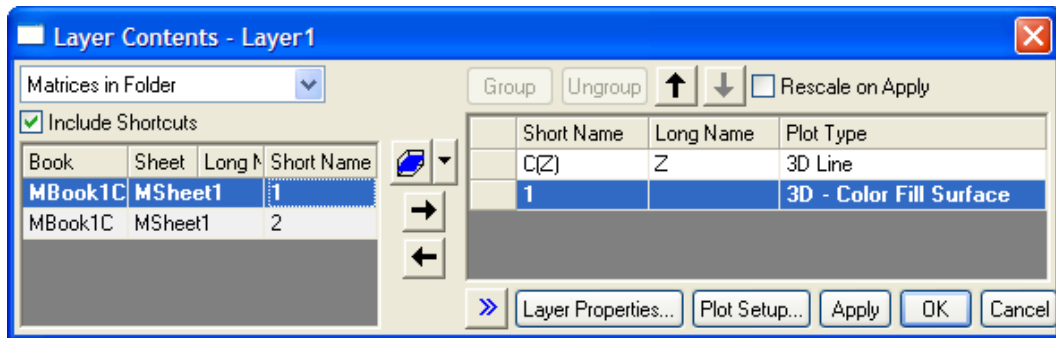
Click **OK** to close the dialog box.



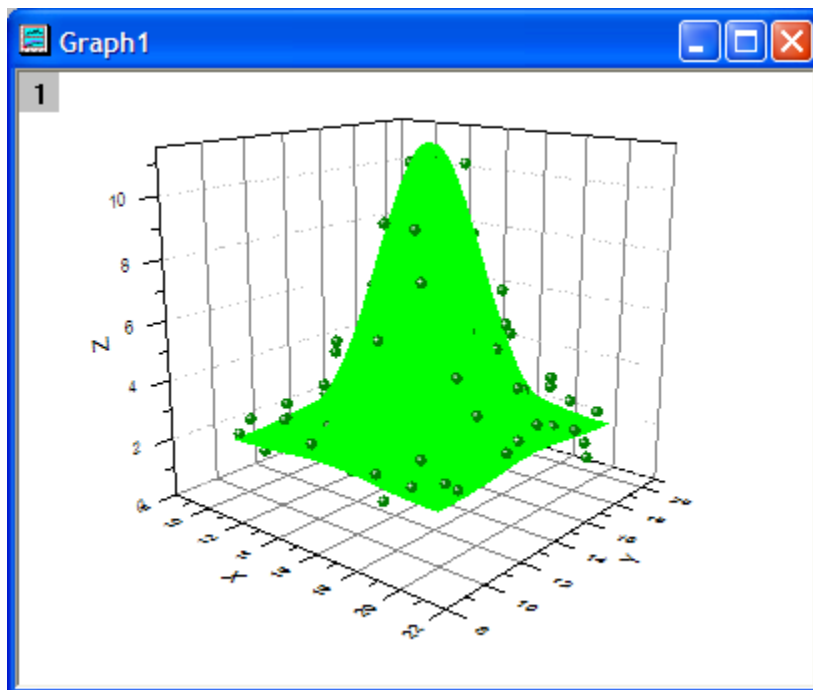
4. Now we need to add a 3D Color Fill Surface plot to this scatter graph. In the top left corner of the graph window, right-click the layer 1 icon, and then select **Layer Contents?** from the menu. In the **Layer Contents** dialog box, select **Matrices in Folder** from the dropdown menu in the upper left. In the left panel, select matrix object 1. Click the **triangle button** next to **A** and select **3D ? Color Fill Surface**.



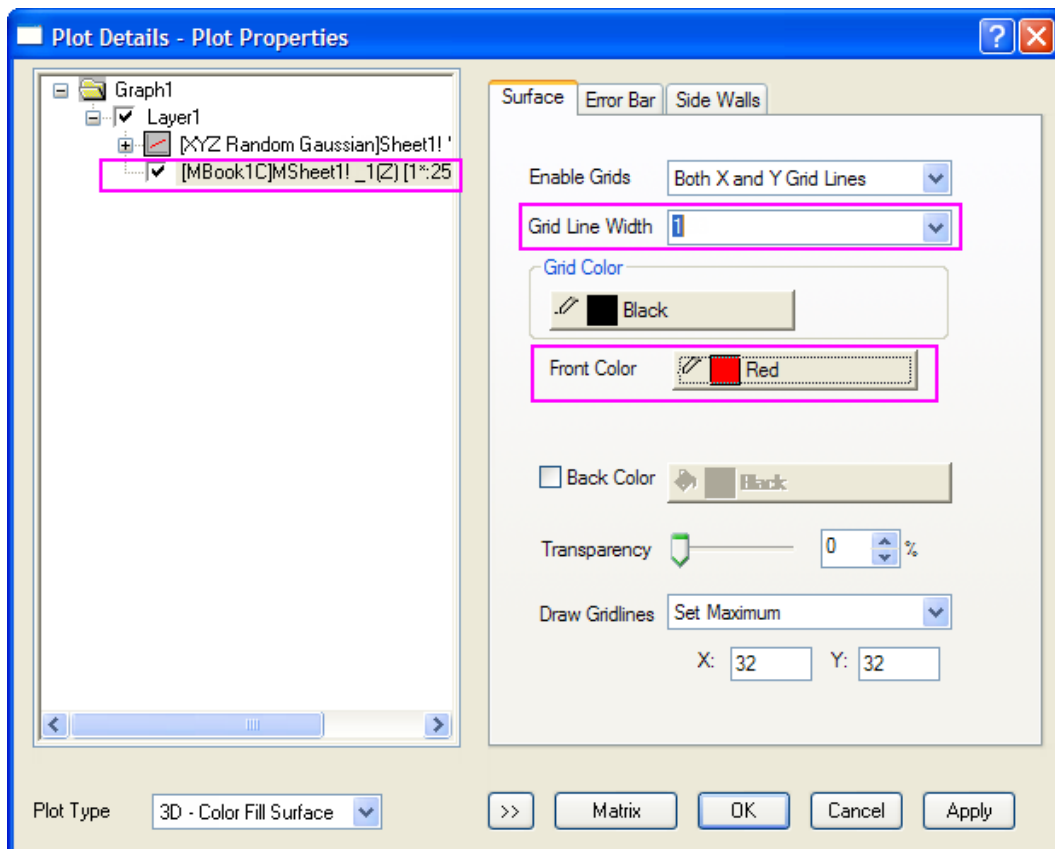
Click  to add it to the right panel.



Click **OK** to close the dialog box.



- Double-click the surface plot to open the **Plot Details** dialog box. In the left panel, select your surface plot. ([MBook3]MSheet1!_1(Z) [1*.2500*]) In the right panel, select the **Surface** tab. Change the **Grid Line Width** to **1** and the **Front Color** to **Red**.



In the **Draw Gridlines** menu, select **Skip**. Type 25? in both the **X** and **Y** fields.

Surface Error Bar Side Walls

Enable Grids Both X and Y Grid Lines

Grid Line Width 1

Grid Color

Black

Front Color Red

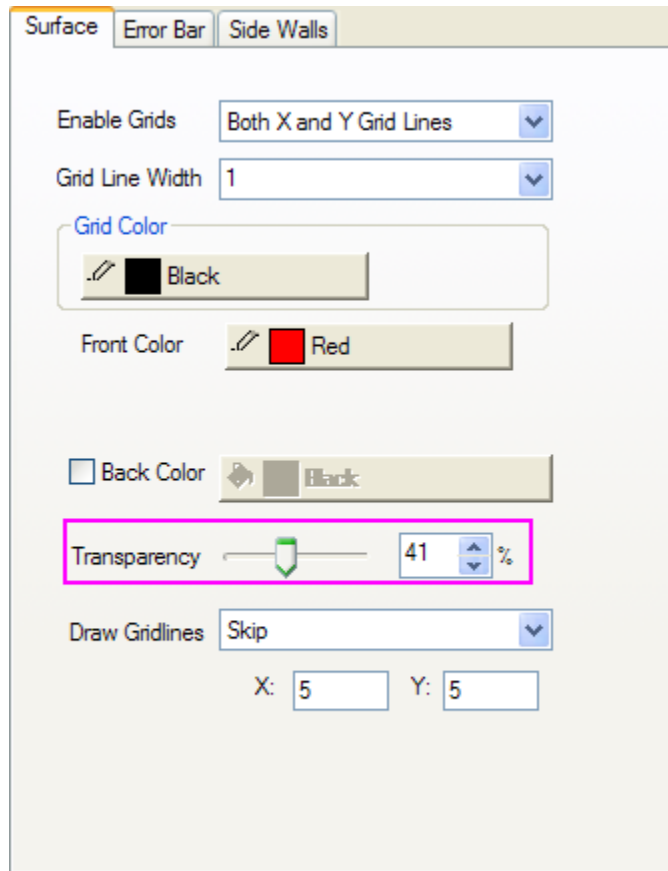
Back Color Black

Transparency 0 %

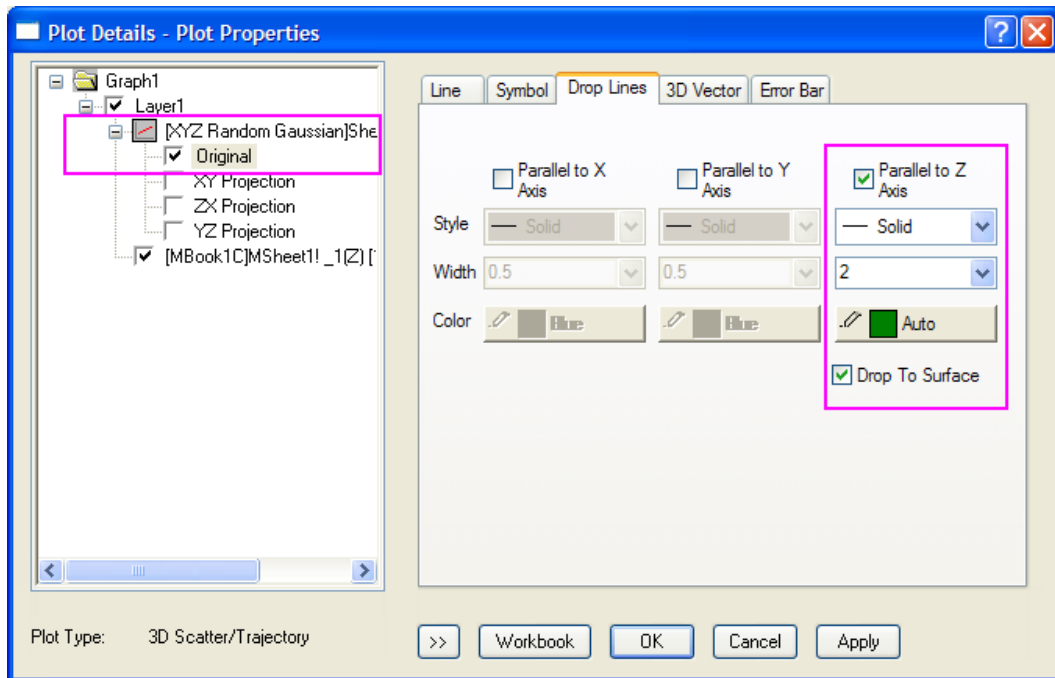
Draw Gridlines Skip

X: 5 Y: 5

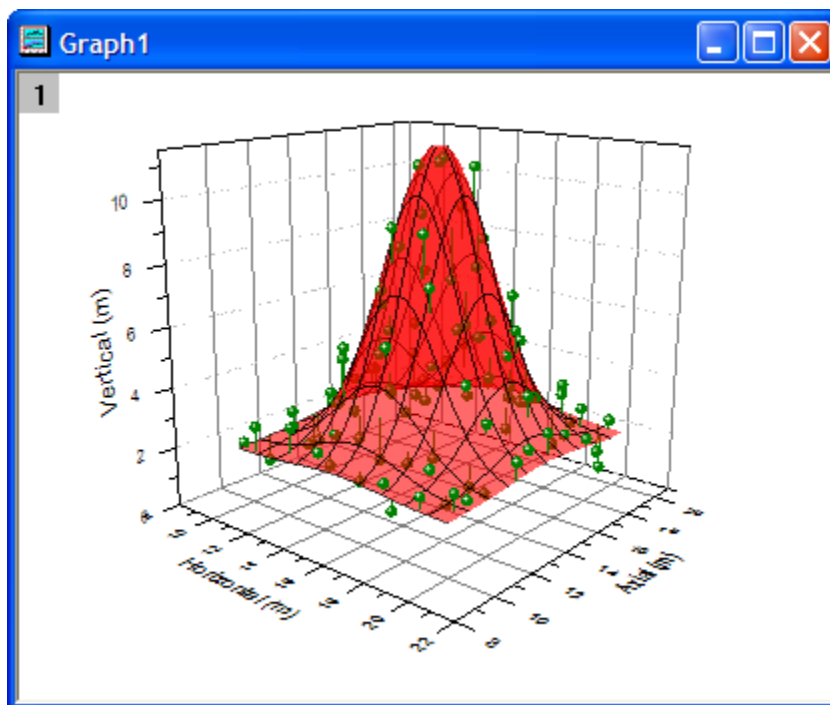
Set Transparency to 41.



- Now we will draw the drop lines. In the tree in the left panel, expand all items and then select the scatter plot. (Original) In the right panel, select the **Drop Lines** tab. Check the **Parallel to Z Axis** check box. Check the **Drop to Surface** check box. Change the drop line **Width** to **2**, and the color to **Auto**.

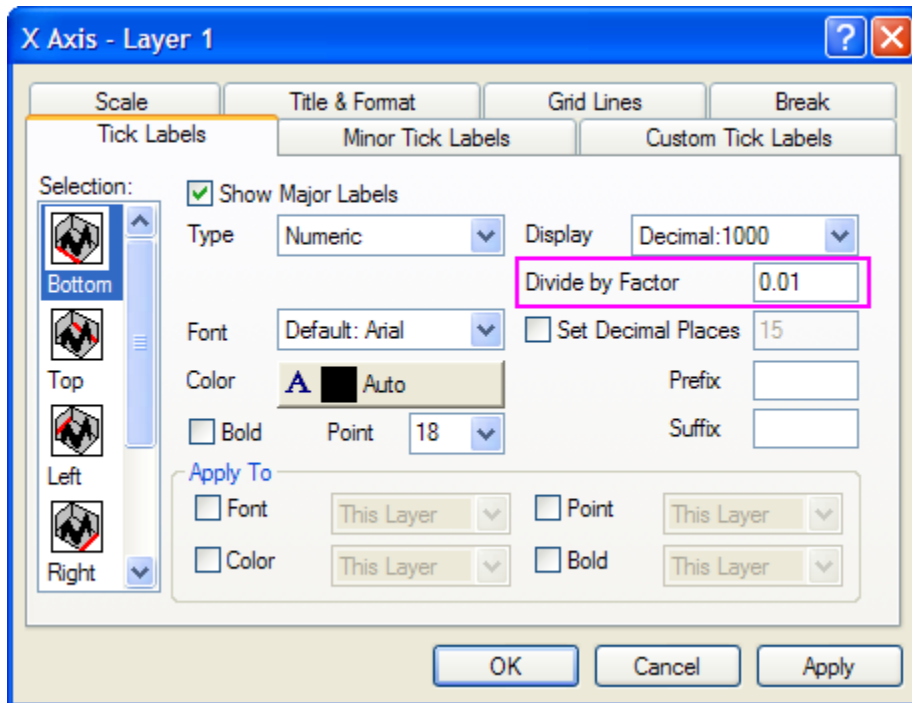


Click **OK** to close the dialog box.

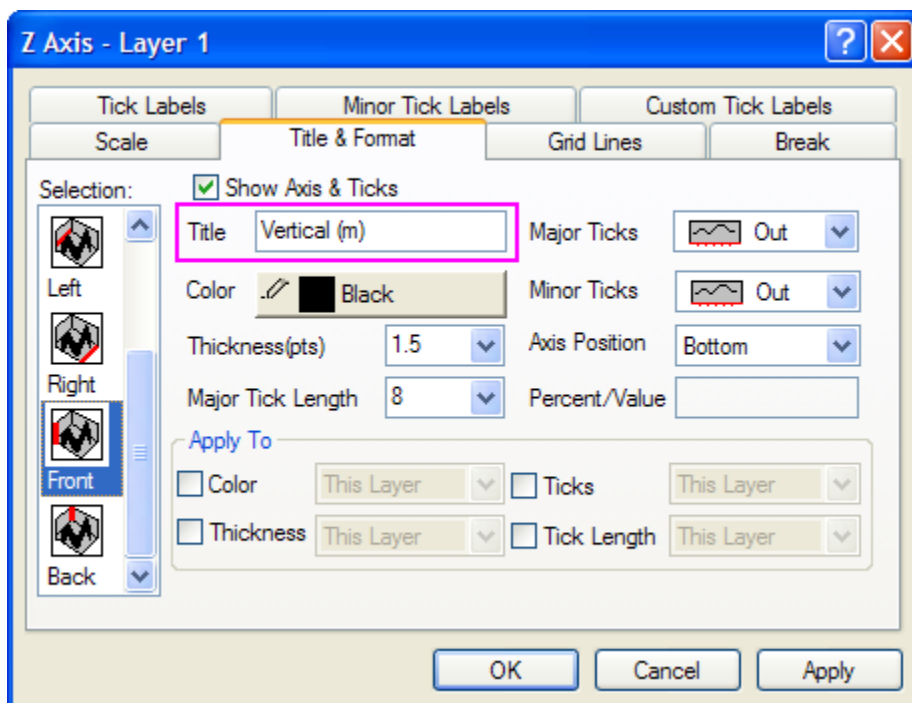



- In the graph window, double-click the X-axis to open the **Axis dialog box**.

Select the **Tick Labels** tab. In the **Divide by Factor** field, type ?0.01?. In the **Selection** list on the left, select **Front**. In the **Divide by Factor** field, type ?0.1?.

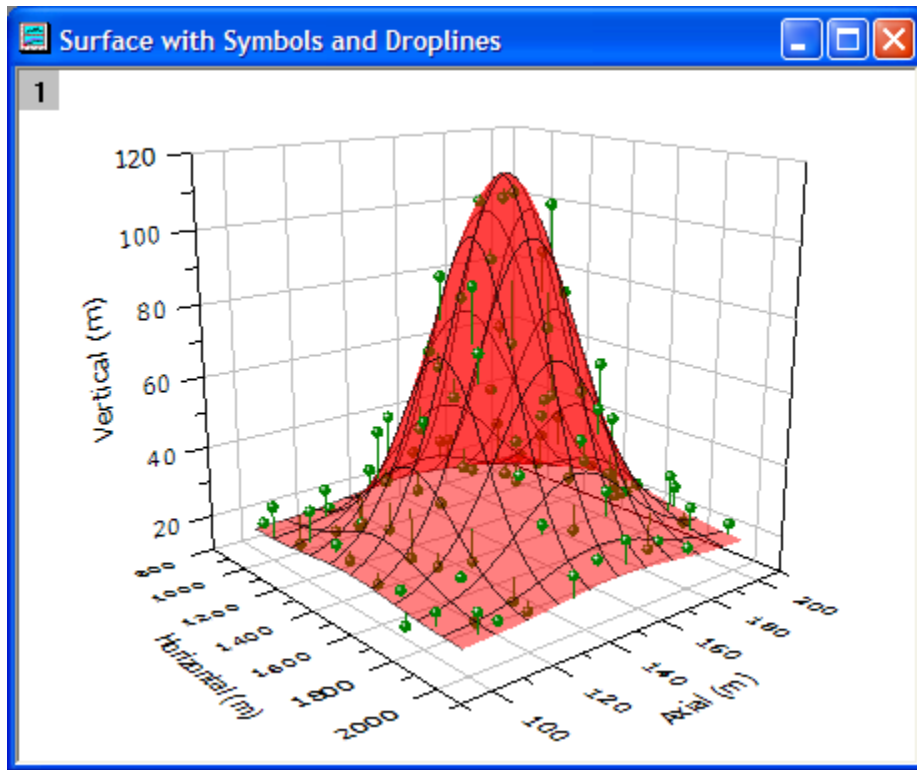


Select the **Title & Format** tab. In the **Selection** list, select **Bottom**. In the **Title** field, type ?Horizontal (m)?. In the **Selection** list, select **Right**. In the **Title** field, type ?Axial (m)?. In the **Selection** menu, select **Front**. In the **Title** field, type ?Vertical (m)?.



Click **OK** to close the dialog box. On the main menu, click the **Rescale** button .

Your final graph should look like this:



3.8.9 Intersecting Color Surfaces

Summary

Origin supports multiple intersecting surfaces. Transparency can be used to enhance visualization.

Minimum Origin Version Required: Origin 8.5 SR0 - however, this tutorial uses a project file only available starting with 8.5.1

What you will learn

- The means and requirements for displaying multiple intersecting surfaces.

Steps

This tutorial is associated with the **Intersecting Surfaces** folder in the **Sample Graphs** project (\Samples\Samples.opj).

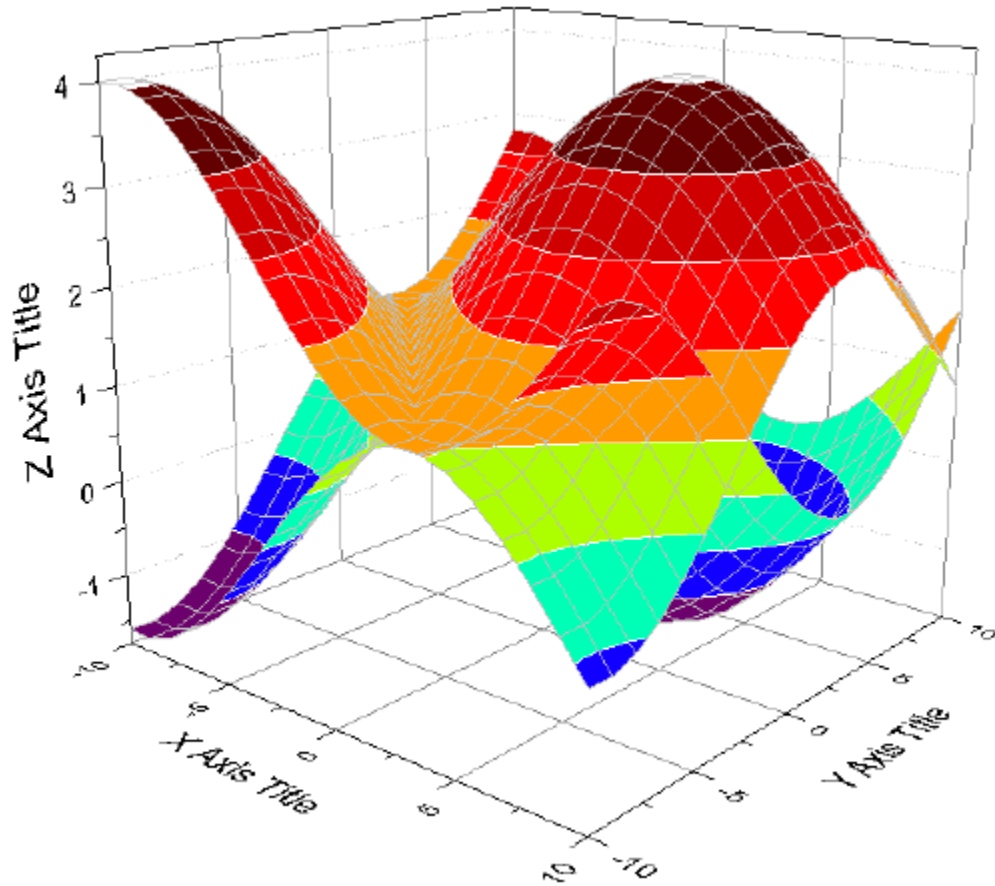
1. In the Project Explorer (usually at the left of the screen), browse to **\Samples\Graphing\3D Surface\Intersecting Surfaces**. Select the **MBook36** window to make active. Note that the matrix displays two thumbnail images above the data, one for each matrix object in the matrix.



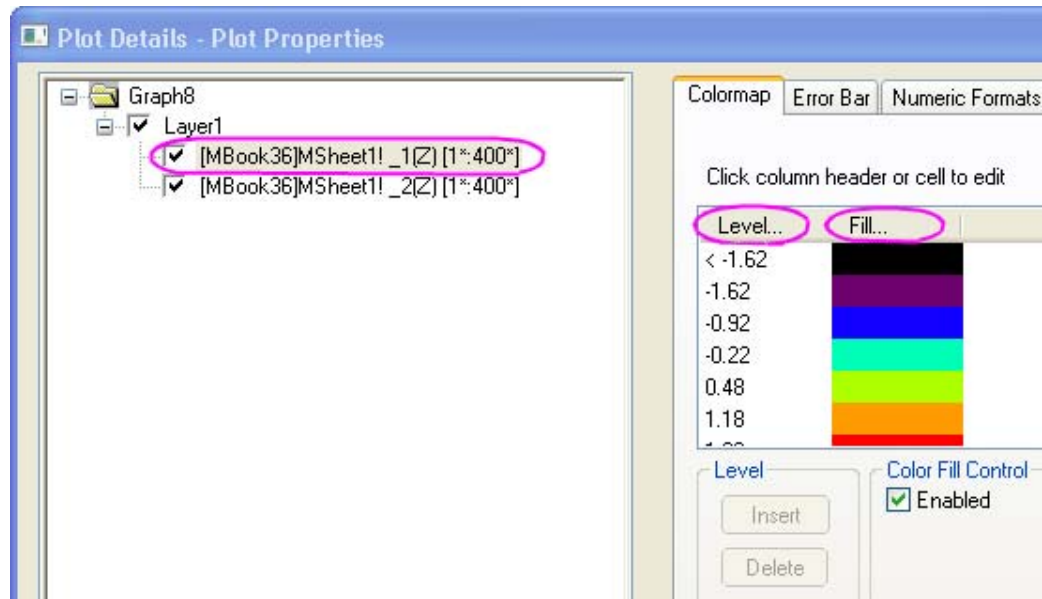
To plot intersecting surfaces, each matrix object must have identical dimensions and XY Mapping. This requirement is automatically fulfilled when the two objects are contained in the same matrix.

See Hierarchy of Origin Objects for additional information.

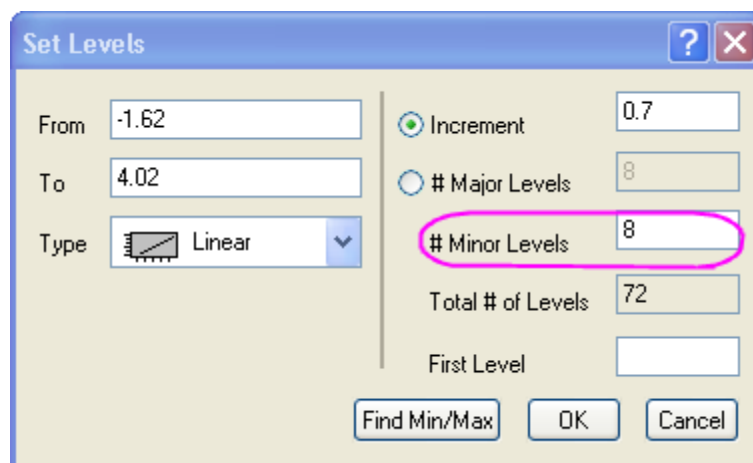
2. Select either thumbnail image. From the main menu, select **Plot**, then point to **3D Surface**, and then click **Multiple Colormap Surfaces**. This will generate an intersecting surface plot of every matrix object in the matrix sheet.



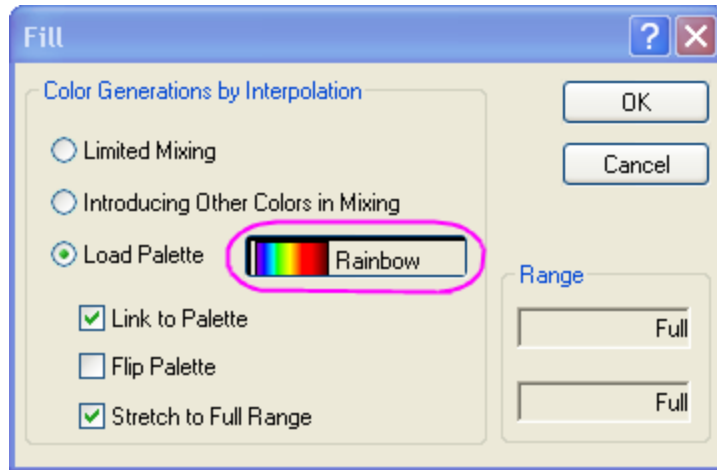
3. Open the Plot Properties dialog box by double-clicking on a plot surface, or by selecting **Format** from the main menu, and then clicking **Plot Properties....** In the left panel, select **[MBook36]MSheet1!_1(Z)[1*:400*]**. In the right panel, click the **Level...** column header to open the **Set Levels** dialog box.



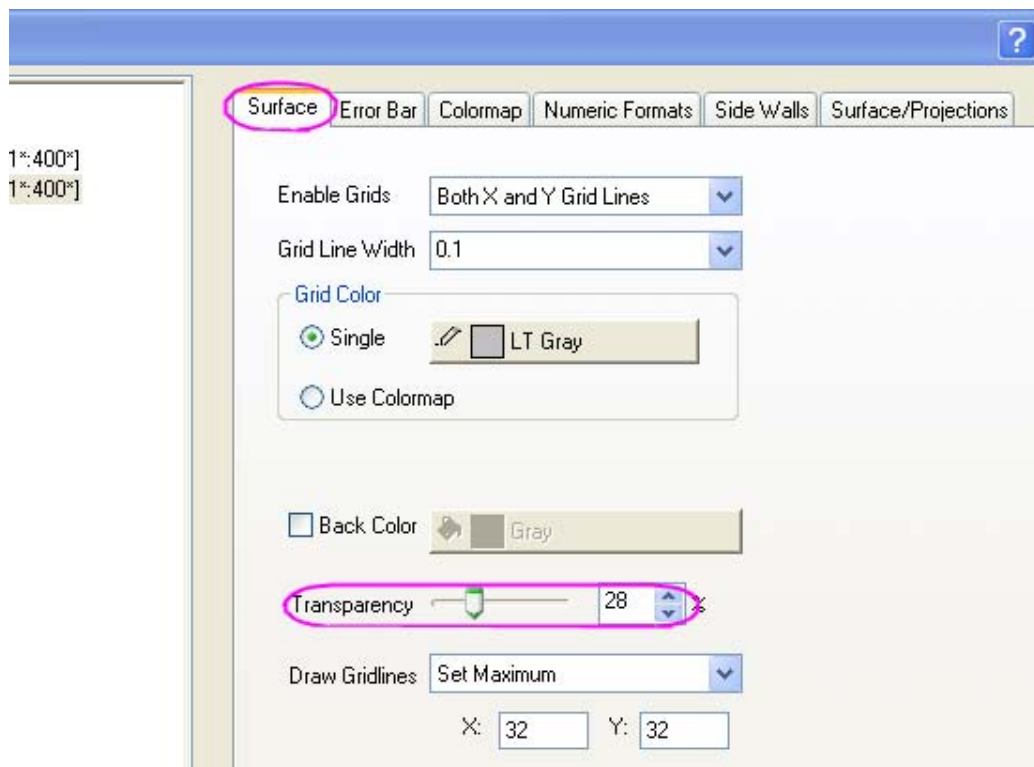
4. Set # **Minor Levels** to **8** and click **OK** to close the Set Levels dialog box.



5. Click the **Fill...** column header to open the Fill dialog box. Click the default **Rainbow** palette to open the palette menu. On the palette menu, select **Pastel**. Click **OK** to close the Fill dialog box.

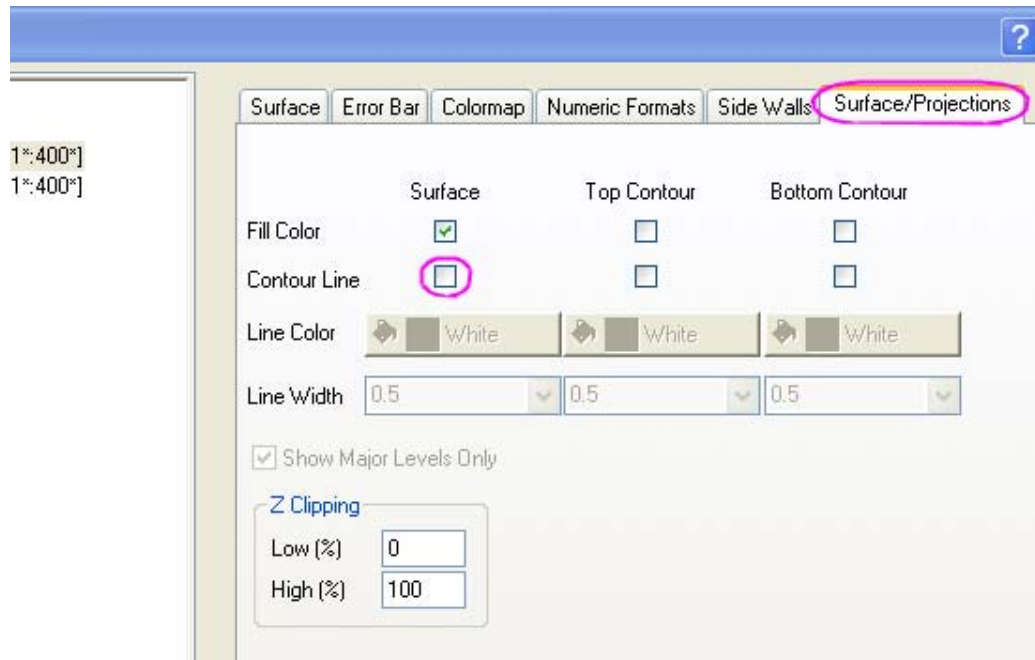


6. Select **[MBook36]MSheet1!_2(Z)[1*:400*]** in the left panel. Click the **Level...** column header and set # **Minor Levels** to **8**. Click **OK** to close the Set Levels dialog box. Click **Apply** to see the modifications so far.
7. In the left panel, select **[MBook36]MSheet1!_1(Z)[1*:400*]**. Select the **Surface** tab and set **Transparency** to a value between **20** and **30%**. Select **[MBook36]MSheet1!_2(Z)[1*:400*]**, select the **Surface** tab, and then set **Transparency** to a value between **20** and **30%**.



8. With **[MBook36]MSheet1!_2(Z)[1*:400*]** still selected, select the **Surface/Projections** tab and clear the **Contour Line check box** in the Surface column. Select

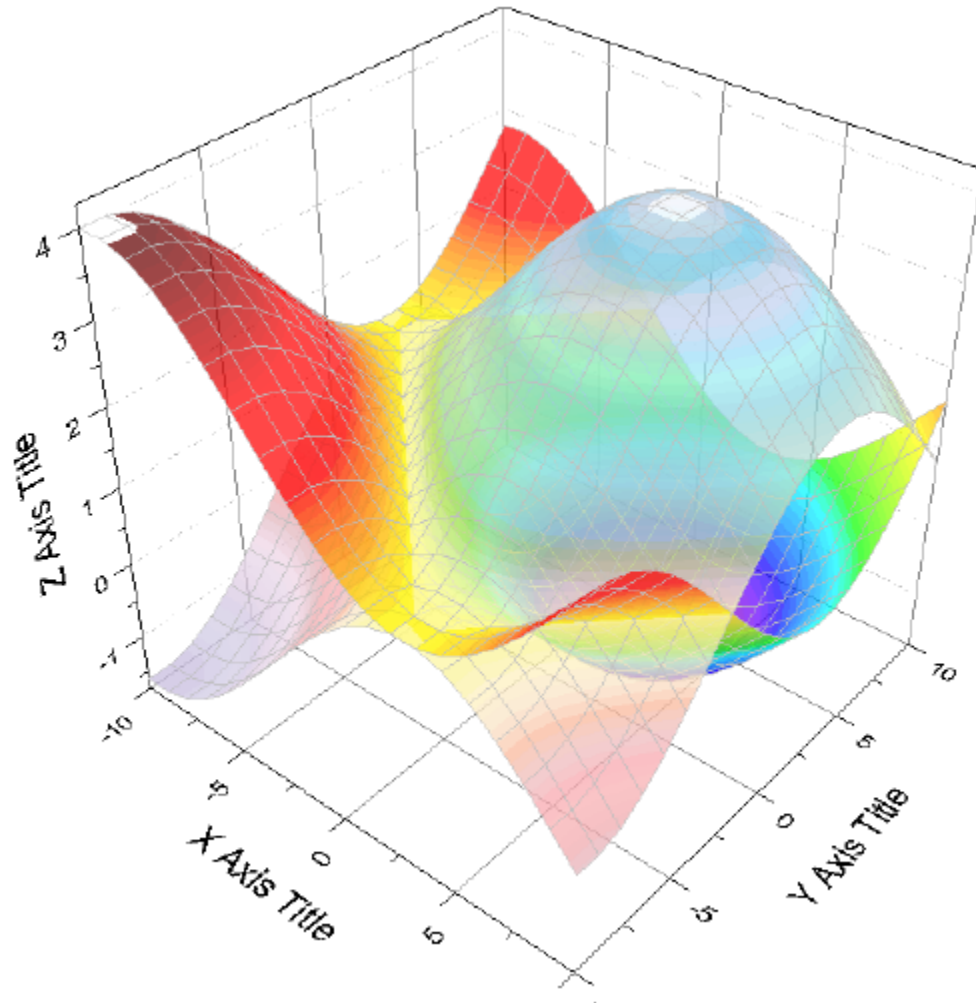
[MBook36]MSheet1!_1(Z)[1*:400*] and clear the **Contour Line** check box in the Surface column. Click **OK** to close the Plot Details dialog box and update the graph.



9. You can use the **Tilt Down** button on the 3D Rotation toolbar to gain an overhead perspective.



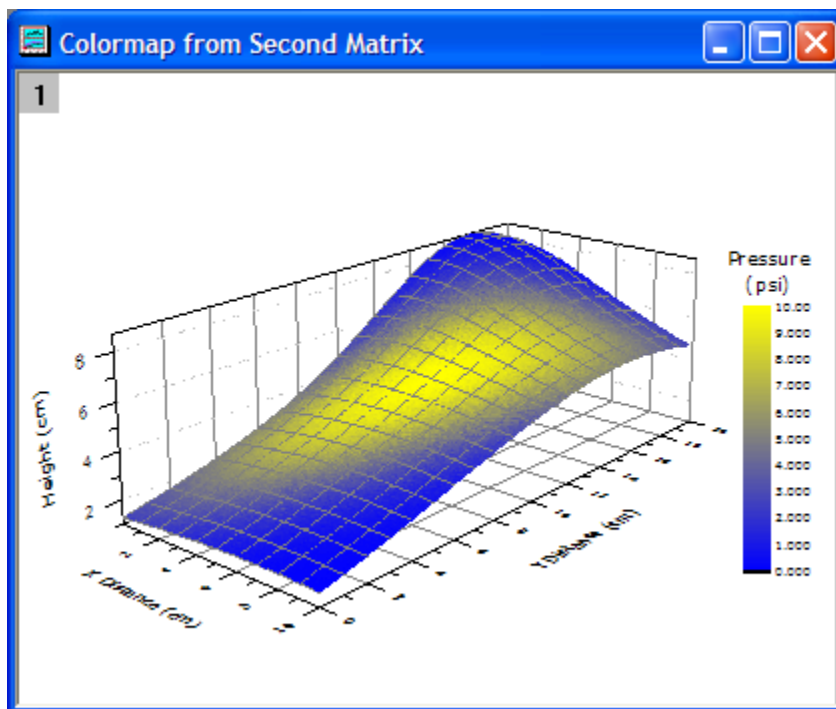
Your final graph should look something like this:



3.8.10 Colormap from Second Matrix

Summary

Origin can represent four-dimensional data by color-mapping a surface plot using a second matrix.



Minimum Origin Version Required: Origin 8.5 SR0

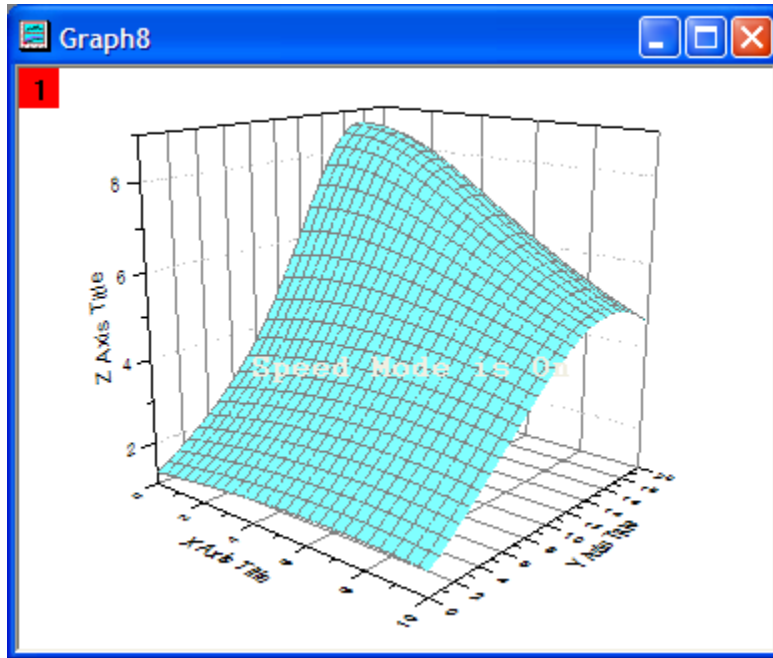
What you will learn

- Create a Color Fill Surface from a matrix.
- Color map a surface plot using a second matrix.
- Customize color map levels and palette.

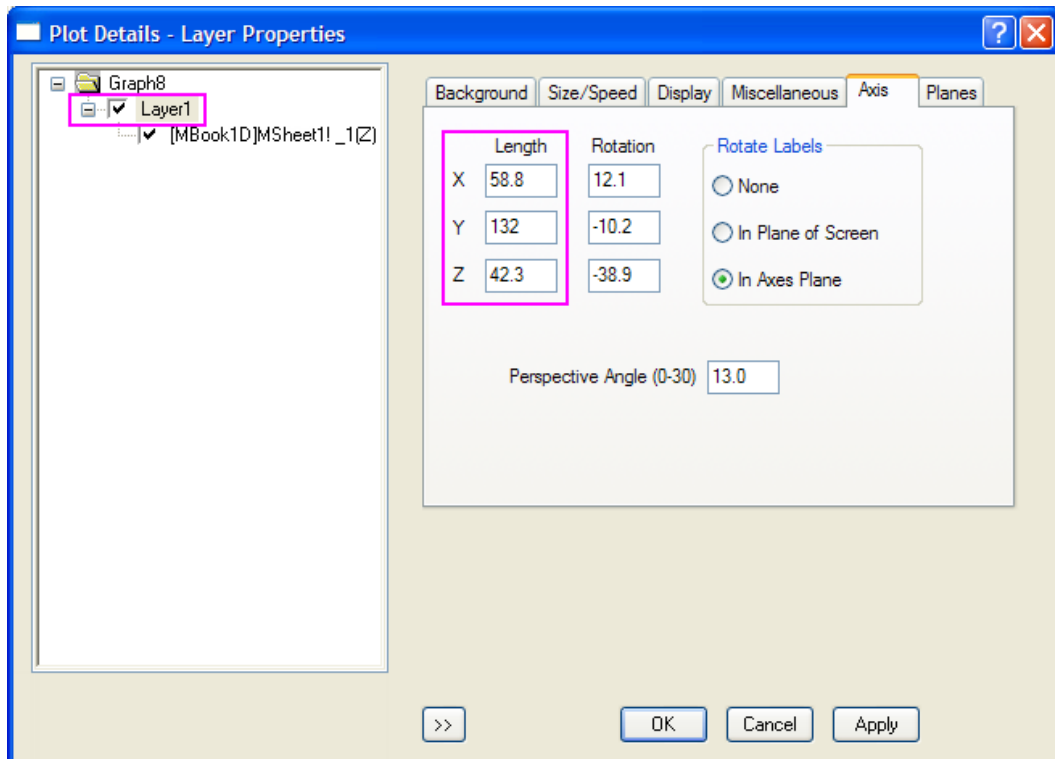
Steps

This tutorial is associated with the **Colormap from Second Matrix** folder in the **Sample Graphs** project (\Samples\Samples.opj).

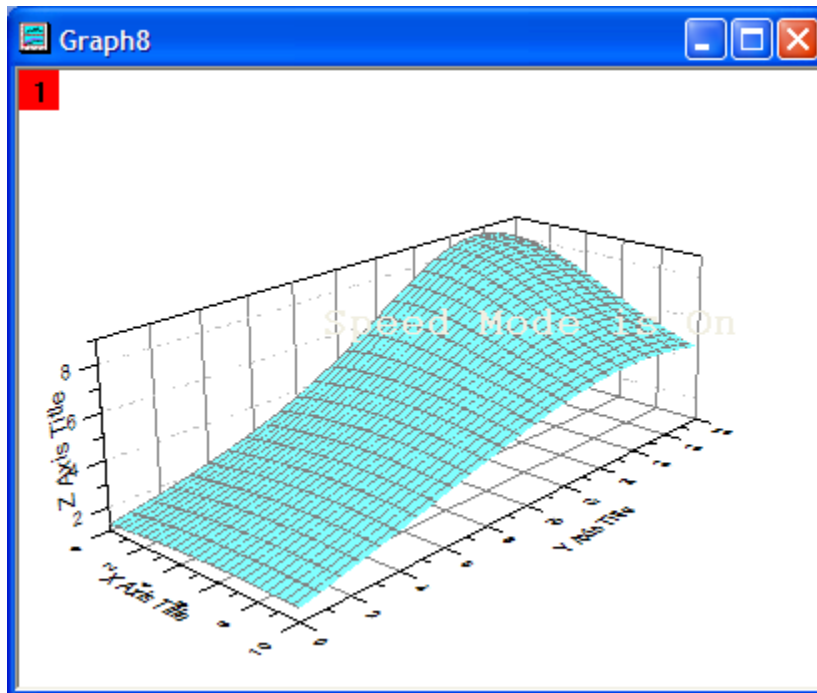
1. In the **Project Explorer** (usually at the left of the screen), browse to **\Samples\Graphing\3D Surface\Colormap from Second Matrix**. Open the matrix book **MBook1D**. You should see two image thumbnails above the matrix data, just under the title bar. (If you do not see image thumbnails, right-click on the matrix title bar and select **Show Image Thumbnails**.) Select image thumbnail **1**.
2. On the main menu, click **Plot**, point to **3D Surface**, then click **Color Fill Surface** to generate a surface plot.



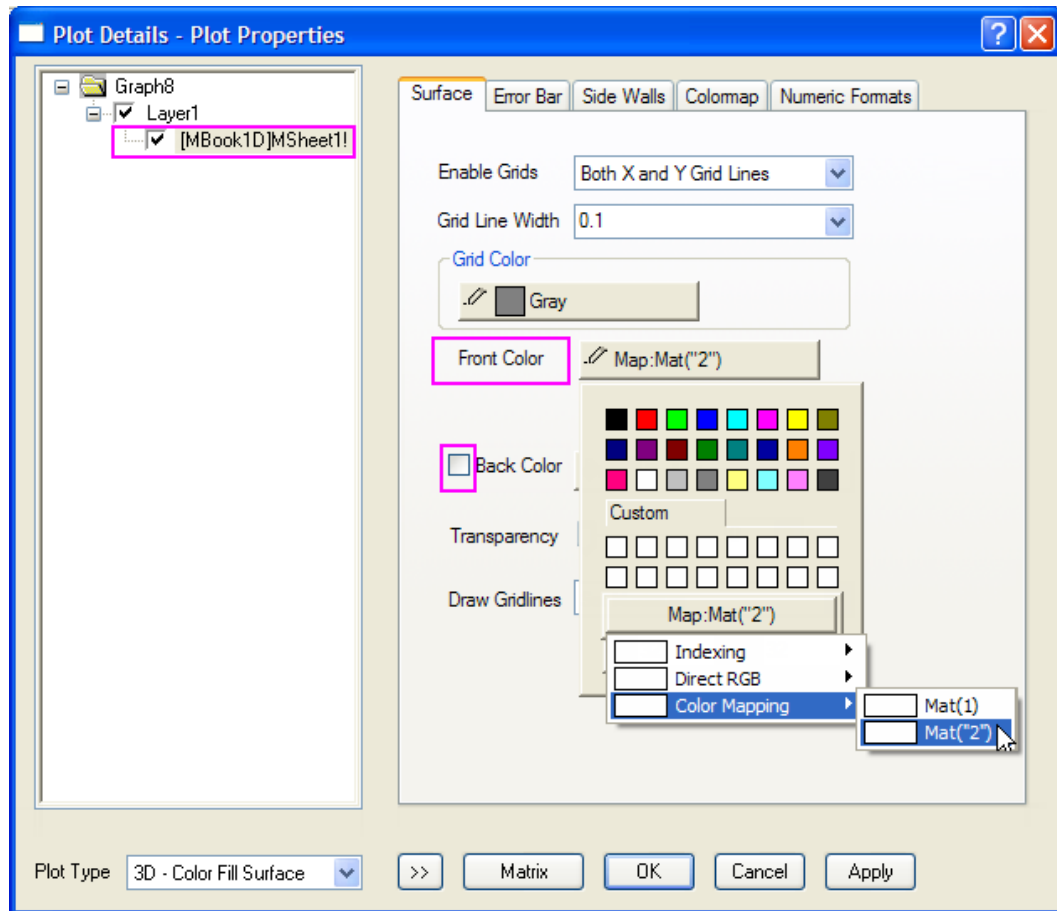
- Double-click anywhere on the plot to open the **Plot Details** dialog box. In the browser in the left panel, select **Layer1**. In the right panel, select the **Axis** tab and type values for the lengths of X, Y, and Z as shown below:



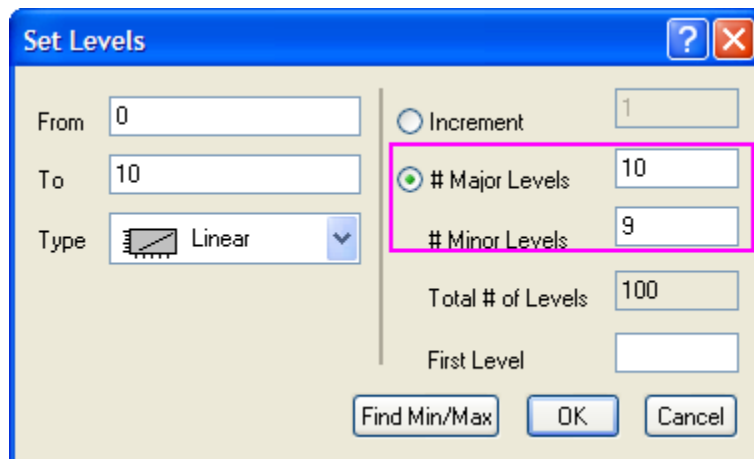
Click **Apply**.



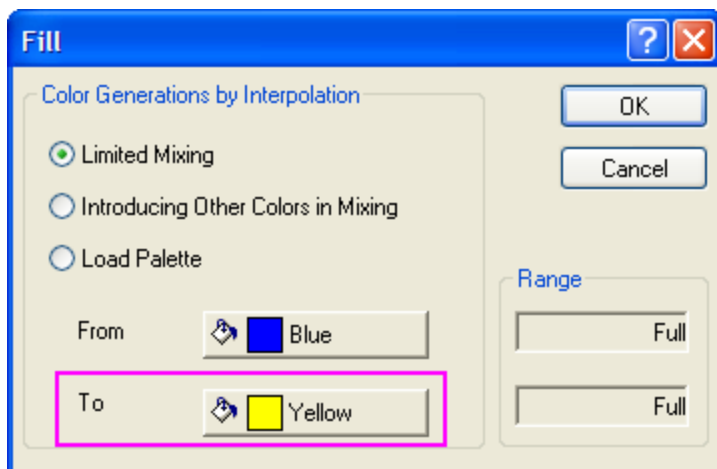
4. In the left panel, select **(MBook1D)MSheet1**. In the right panel, select the **Surface** tab. Click on the **Front Color** button to reveal a menu containing a color palette.
5. In the color palette menu, click **Use Matrix Values**, point to **Color Mapping**, and click **Mat(???)**. (Note: **Use Matrix Values** only appears when **Front Color** is set to **Custom**.) Clear the **Back Color** check box.



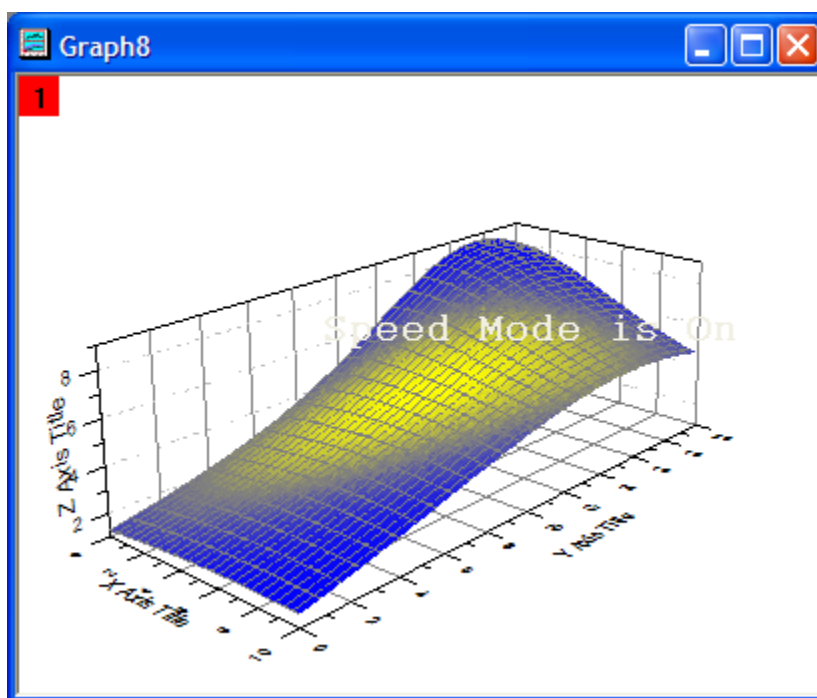
6. Select the **Colormap** tab in the right panel. Select the column header **Level?** to open the **Set Levels** dialog box. Click the **Find Min/Max** button to set the **From** and **To** values. Set **Increment** and **Major/Minor Levels** as shown below:




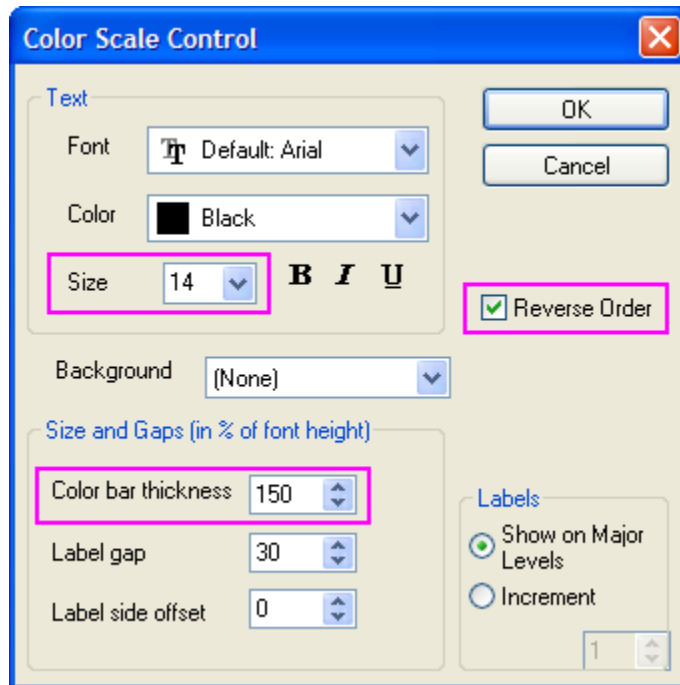
Click **OK** to close the **Set Levels** dialog box. Click the column header **Fill?** to open the **Fill** dialog box. Set the **From** color to **blue**, and the **To** color to **yellow**. Click **OK** to close the **Fill** dialog box.





Click **OK** to close the **Plot Details** dialog box.

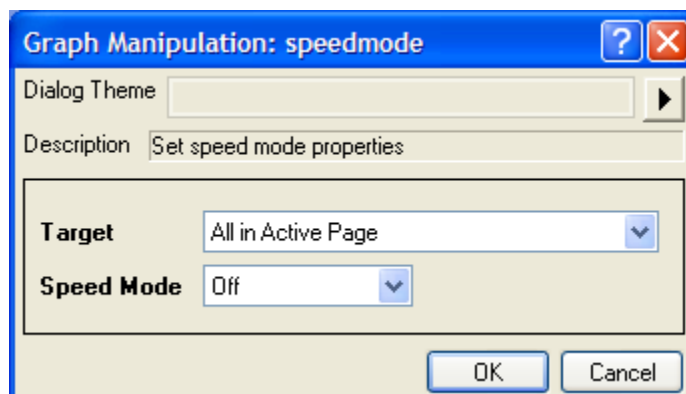


7. Keep the graph window active. On the standard toolbar (usually at the top of the screen), click the **Add Color Scale** button . This will add a color scale to the current graph.
8. Double-click anywhere on the color scale to open the **Color Scale Control** dialog box. Change text **Size** to **14**, and **Color bar thickness** to **150**. Select the **Reverse Order** check box.

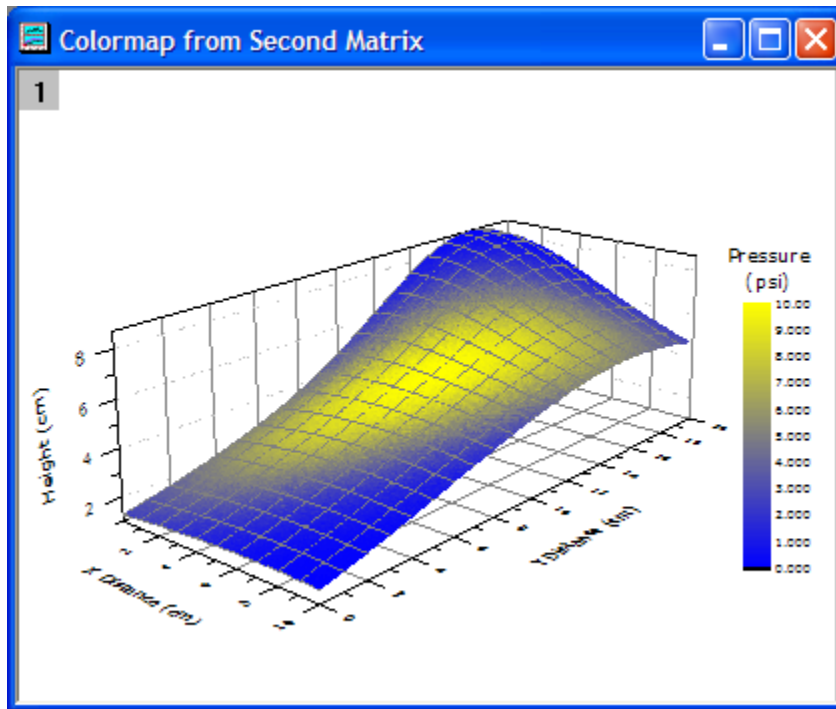


Click **OK** to close the **Color Scale Control** dialog box.

9. On the Tools toolbar (usually on the left side of the screen), select the **Text Tool** button . Click in the graph window above the color scale and type some text to add a title. Hold the **Shift**-**key** down while selecting the color scale and its title text. With both items selected, click the **Group** button  on the Object Edit toolbar (usually near the bottom of the screen).
10. On the main menu, click **Graph**, then **Speed Mode**, then **Open Dialog?** In the **Speed Mode** dialog box, select **All in Active Page** from the **Target** menu, and **Off** from the **Speed Mode** menu.



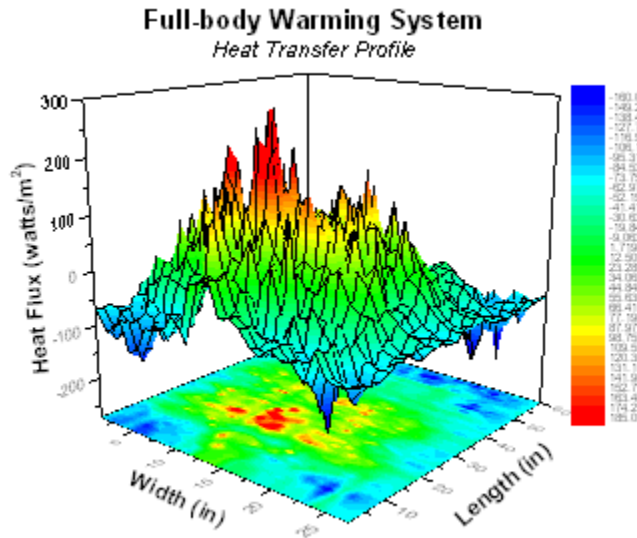
Click **OK** to close the **Speed Mode** dialog box. Your final graph should look like this:



3.8.11 Color Map Surface Graph

Summary

This tutorial will show you how to create a 3D color map surface.



What will you learn

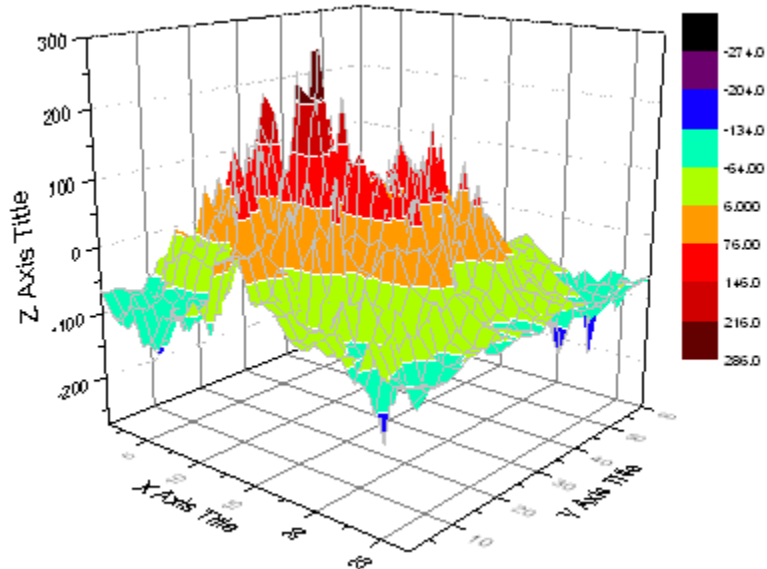
This tutorial will show you how to

- Create a 3D color map surface graph
- Customize the 3D surface graph

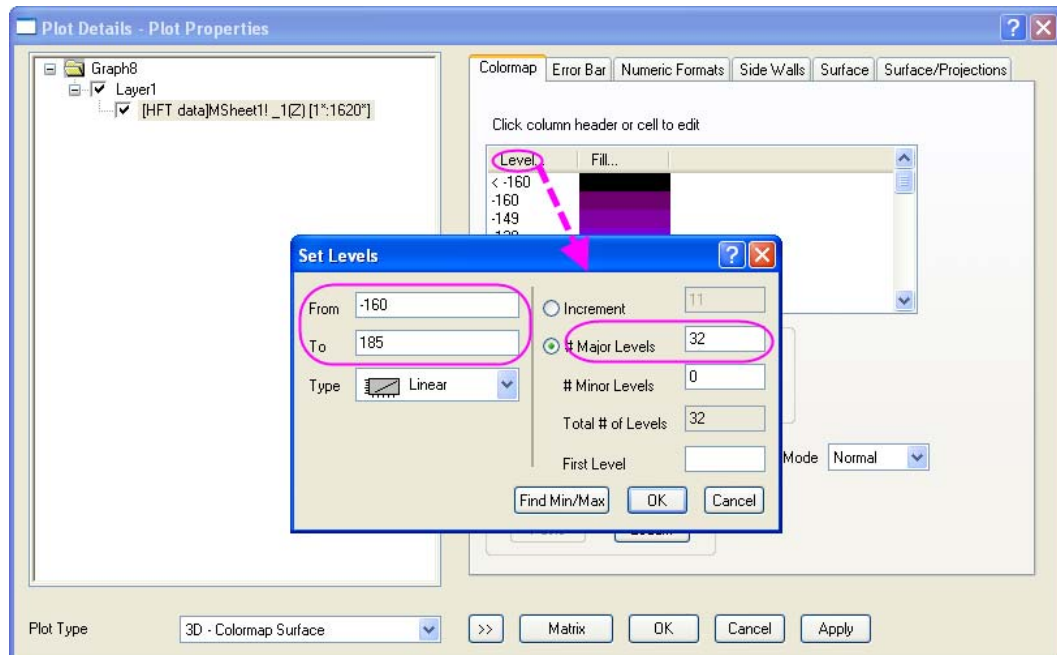
Steps

This tutorial is associated with this Graph Galley page.

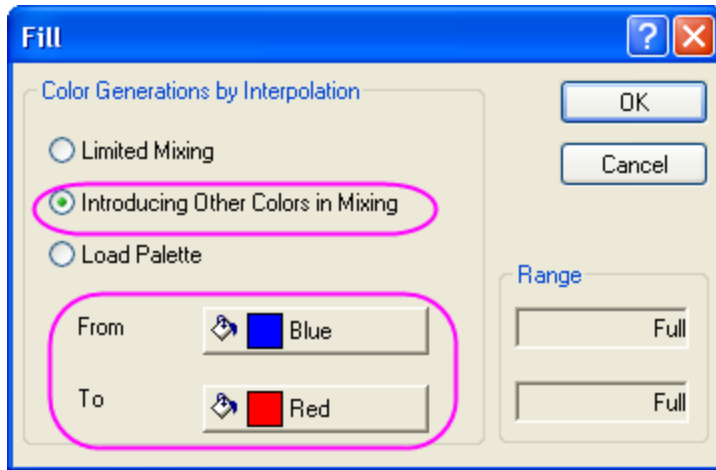
1. Download the project by clicking the Download Project link at the top of this page.
2. Open the project in the zipped file. Highlight **HFT data** matrix and select **Plot: 3D Surface: Color Map Surface** to create a 3D graph.



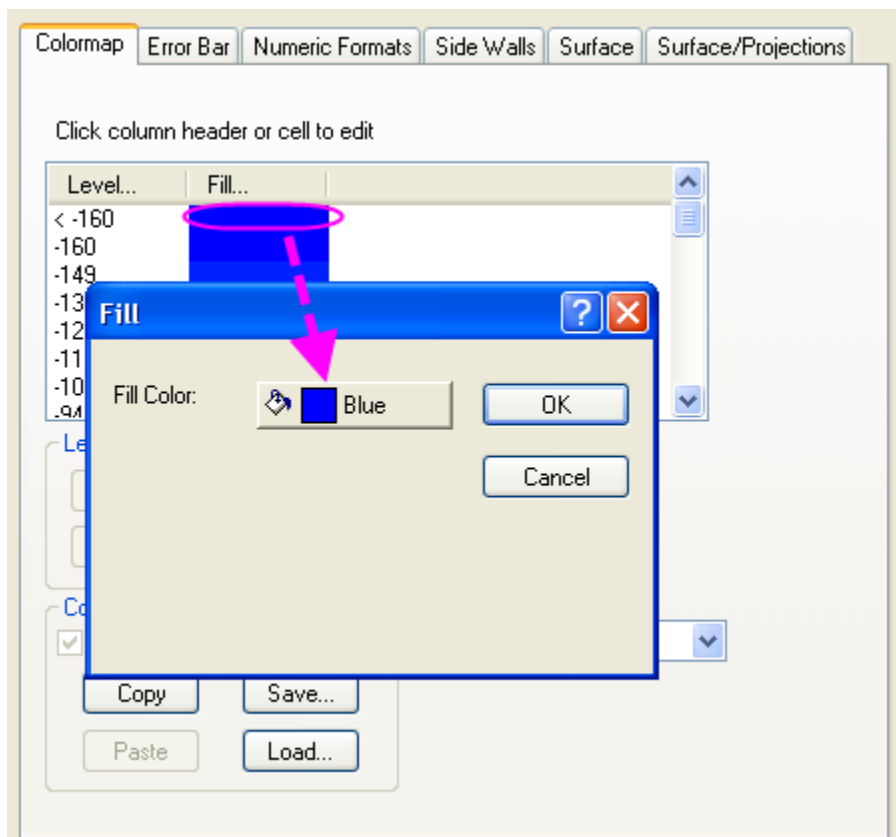
3. In the following steps we will show you how to customize the graph. Choose **Format: Layer Properties** to go to the layer level of the **Plot Details** dialog. Go to the **Planes** tab and uncheck the **XY**, **YZ**, and **ZX** planes check boxes.
4. Expand the **Layer1** branch and select the 3D Surface plot. Go to the **ColorMap** tab, click on the **Level** heading to open the **Set Levels** dialog as the following graph shows:



Click on the **Fill** heading to open the **Fill** dialog and select the **Introducing Other Colors in Mixing** radio button. Set the **From** and **To** color as Blue and Red.

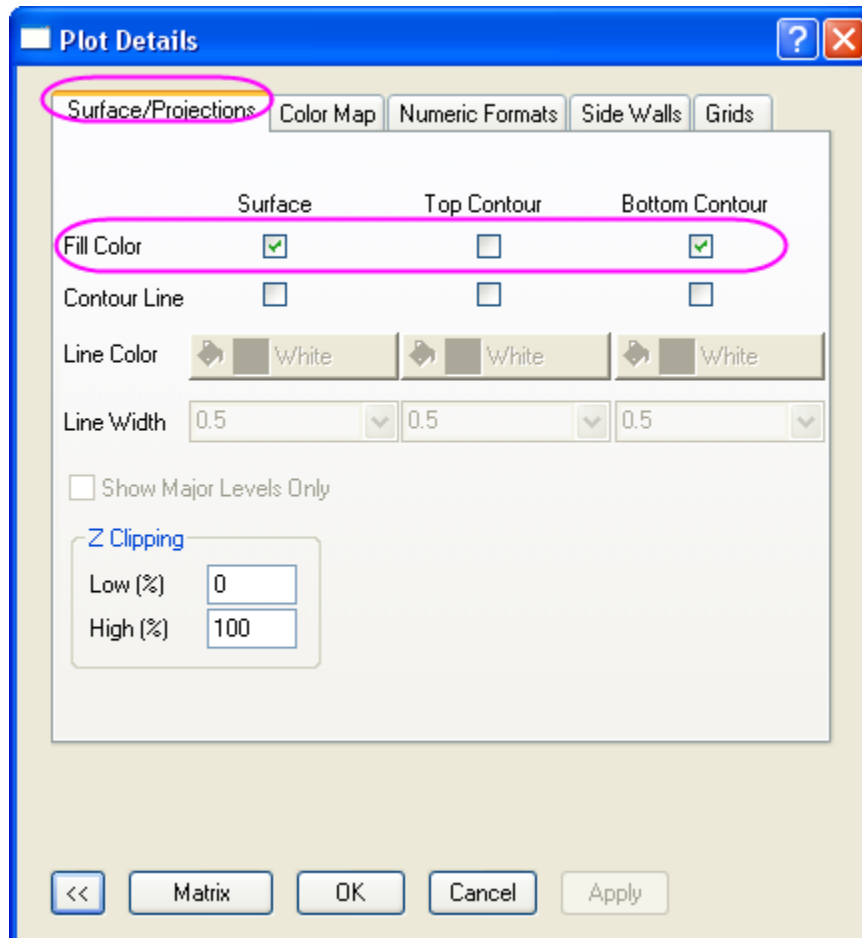


Click the cell the in the **Fill** column and <-160 row to set **Fill Color** to **Blue**.

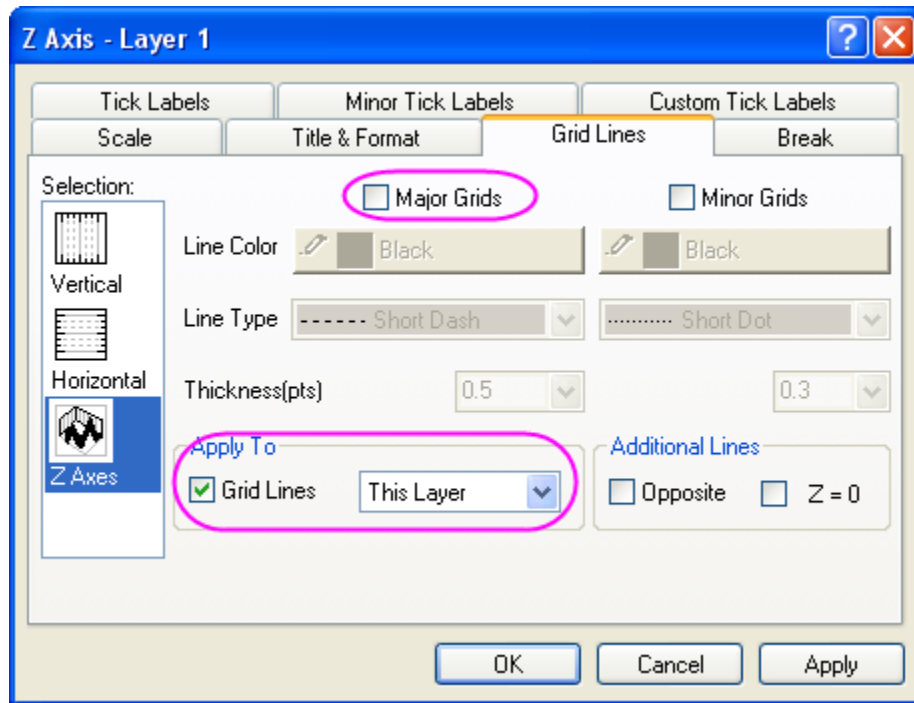


Click the cell the in the **Fill** column and >185 row to set **Fill Color** to **Red**.

5. Go to the **Surface/Projections** tab. Set the dialog as the following image shows:

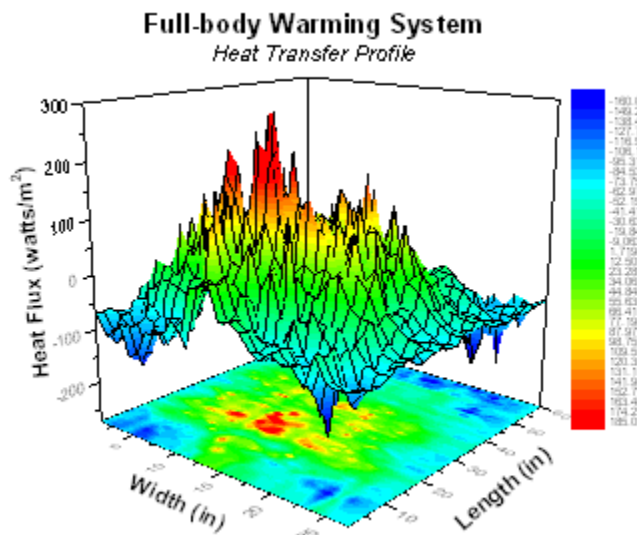


6. Go to the **Surface** tab, select **Black** color from the color list in the **Grid Color** group. Then click OK button to close the Plot Details dialog.
7. Double-click on the one of the axes of the graph to open the Axes dialog. Go to the **Grid Lines** tab and set the dialog as the following graph shows:



Click **OK** button to apply this change.

8. Finally, change axis titles and add a graph title as the following graph.



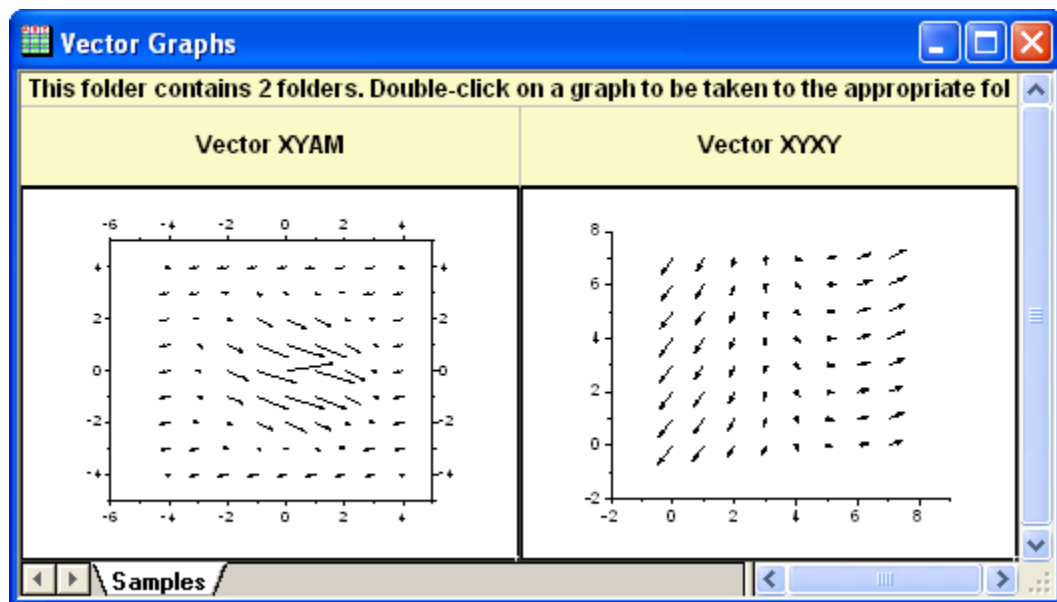
3.9 Vector

3.9.1 Vector Graph

Summary

A vector plot is a multidimensional graph used in industries such as meteorology, aviation, research, and construction that illustrate flow patterns (e.g. of wind, water, magnetic fields, etc.). Both direction and magnitude are represented in a vector graph. Origin includes two types of vector plots:

1. **Vector XYAM** - takes a starting XY location for the vector tail (by default), an angle and a magnitude.
2. **Vector XYXY** - takes two XY positions and connects them with a vector.



What you will learn

- Two data organizing mode to plot a vector graph.
- Use Plot Setup to assign plot data

Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

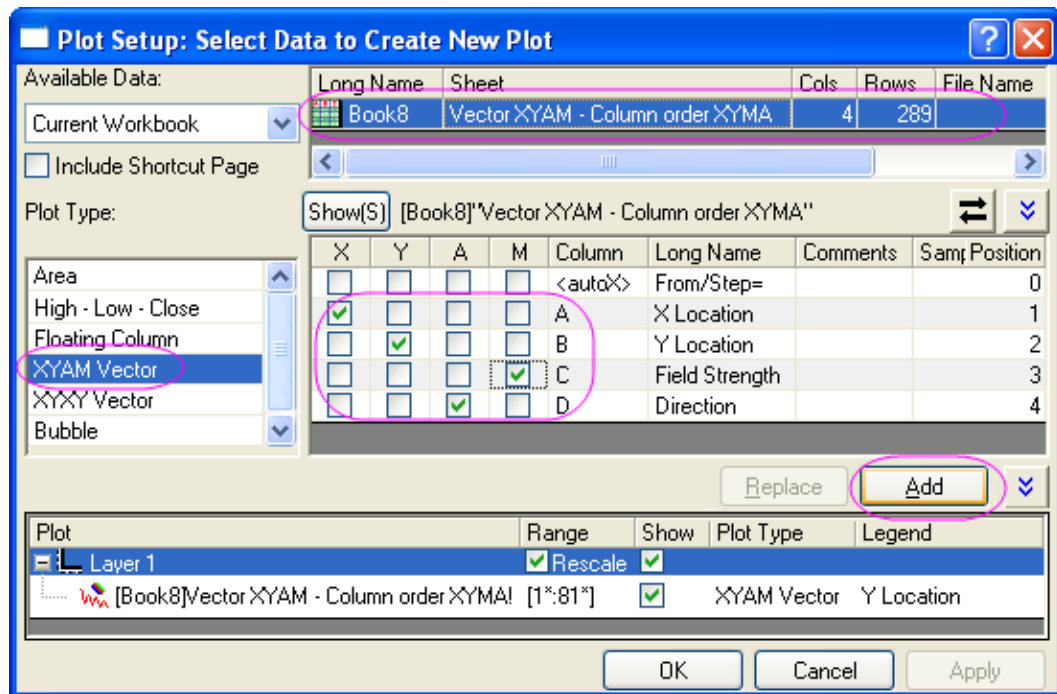
To create a vector plot, you need four data columns, and organize your data into two modes:

Vector XYAM

XYAM means X, Y, Angle and Magnitude, respectively. Vectors will start from point (X, Y), and rotate an angle you specified. The following example uses XYAM data mode to create a vector graph.

1. Browse the folder *Graphing: Vector: 2D Vector* in the **Project Explorer**. Active the data worksheet **Book8E**. Without highlighting any columns, select **Plot: Specialized: Vector XYAM**

to open the **Plot Setup** dialog. And assign column A, B, D, C to X, Y, A, M as below, then click **OK** to plot the graph.



2. Select **Format:Plot** to go to the plot level of the **Plot Details** dialog. Go to the **Vector** tab and change the **Magnitude Multiplier** in **Vector Data** group to 75. Click **OK** button to close **Plot Details** dialog.
3. Double-click X or Y axis to open the **Axis** dialog. Activate the **Title & Format** tab, and check/enable the **Show Axis & Ticks** option for the **Right** axis. And set the **Major Ticks** and **Minor Ticks** to **Out**. Show the **Top** axis either in this tab.
Go to **Tick Labels** tab, and check the **Show Major Labels** box for both **Right** and **Top** axis.

Vector XYXY

Another data organizing mode is XYXY, where the first X, Y is the vector starting point, while the last X, Y is the ending point.

To create such vector graph, active the *Graphing: Vector: 2D Vector* folder in the OPJ, highlight all of the four data columns of **Book9E**, and select **Plot: Specialized: Vector XYXY** from menu. Delete the axis titles if necessary.

3.9.2 3D Vector

Summary

Origin can plot 3D Vectors from two different data structures: **XYZ XYZ** and **XYZ dXdYdZ**.

XYZ XYZ defines the tail and head data for the vector, while XYZ dXdYdZ provides the vector's tail data and the distance between tail and head projections on X, Y, and Z axes.

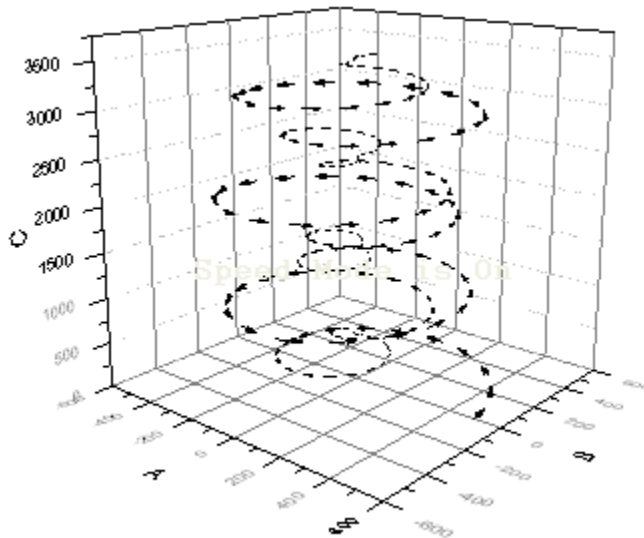
What you will learn

- Create a 3D Vector from XYZ XYZ worksheet data.
- Customize the 3D Vector

Steps

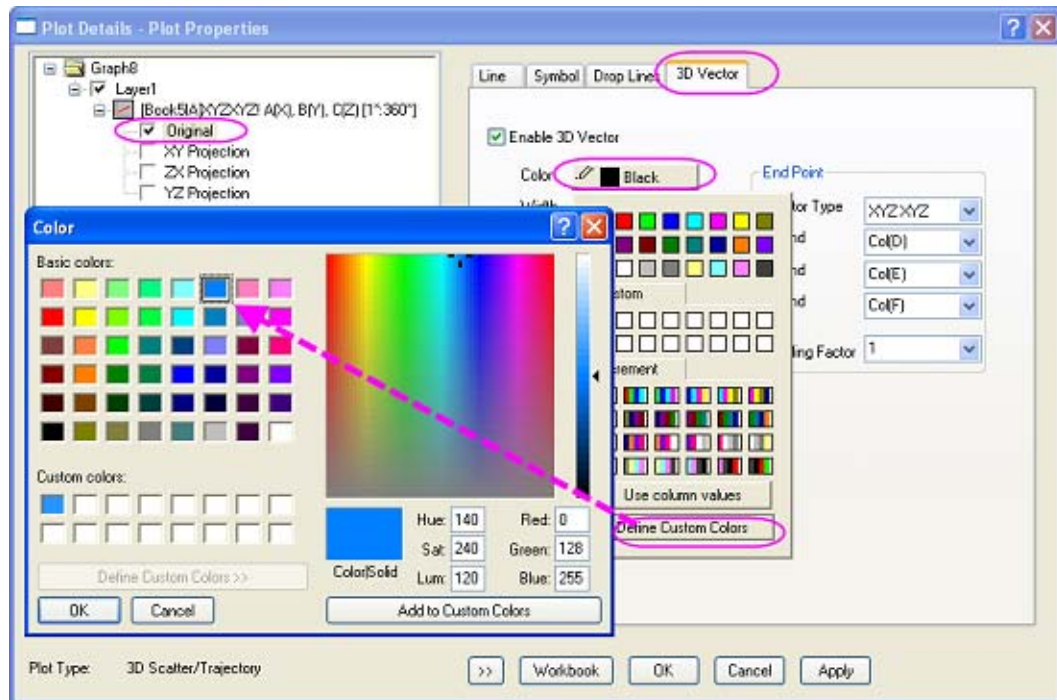
This tutorial is associated with the **3D Vector** folder in the **Sample Graphs** project (\Samples\Samples.opj).

1. In the Project Explorer (usually at the left of the screen), browse to **\Samples\Graphing\Vector\3D Vector**. Select workbook **Book51A**.
2. Click in the upper left corner of the worksheet to select all data.
3. On the main menu, select **Plot**, then point to **3D XYZ**, and then click **3D Vector XYZ XYZ**. This will create a 3D Vector graph with our built-in template.

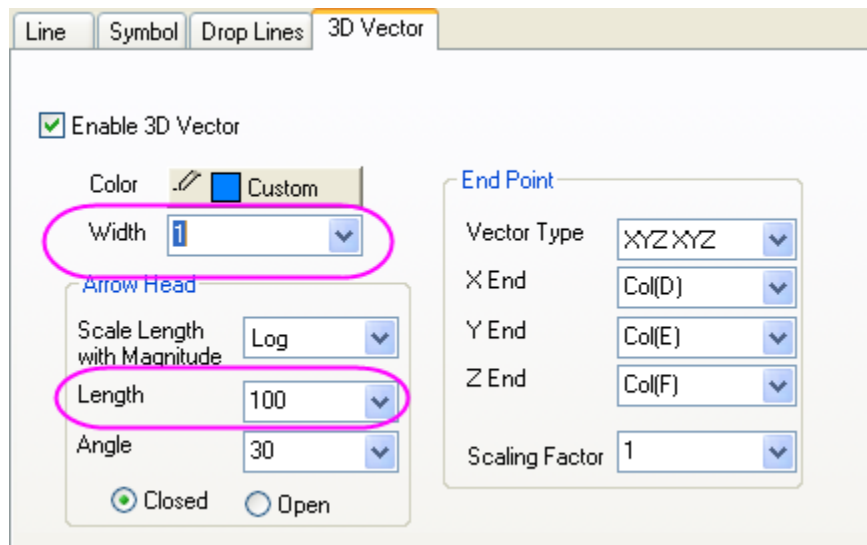


Note that speed mode is on.

4. Double-click on the vector plot to open the Plot Details dialog box. Ensure that the **Original** vector plot has been selected in the left panel of the dialog box. Select the **3D Vector** tab.
 - Click the **Color** button and select **Define Custom Colors** from the menu. Define a custom color in the Color dialog box. (We chose a custom shade of blue.)

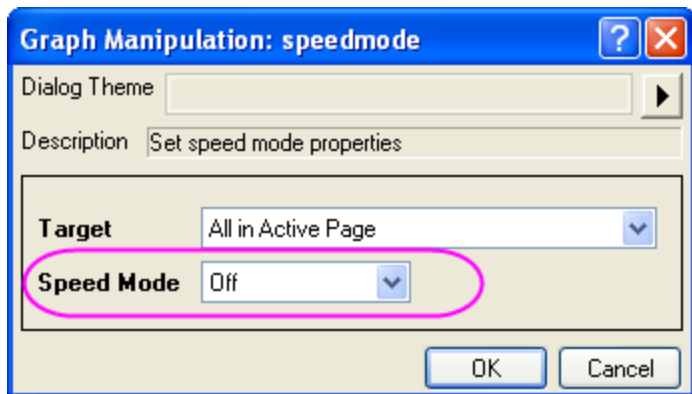


- In the **3D Vector** tab of **Plot Details** dialog box, change the vector **Width** from 0.5 to **1**. In the Arrow Head group, change the **Length** from 63 to **100**.

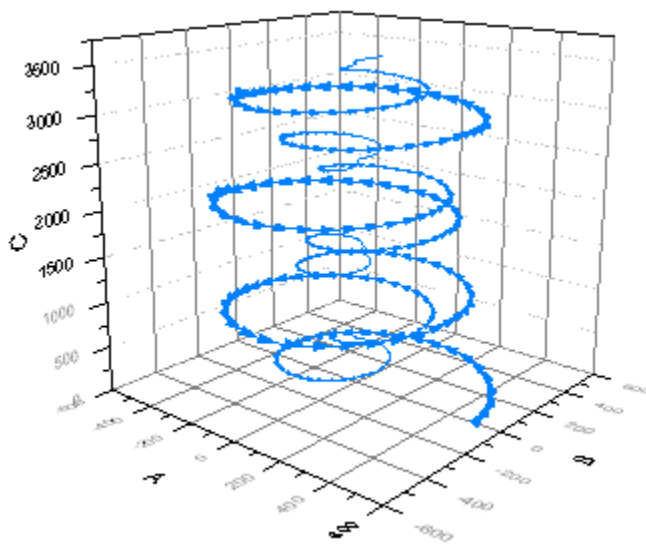


Click **OK** to apply these settings and close the dialog box.

5. Click the **Rotate counterclockwise** button several times.
6. On the main menu, select **Graph**, point to **Speed Mode...**, then click **Open Dialog...** to open the **speedmode** dialog box. Select **Off** from the Speed Mode dropdown menu. Click **OK** to turn off speed mode and close the dialog box.



The final graph should look something like this:

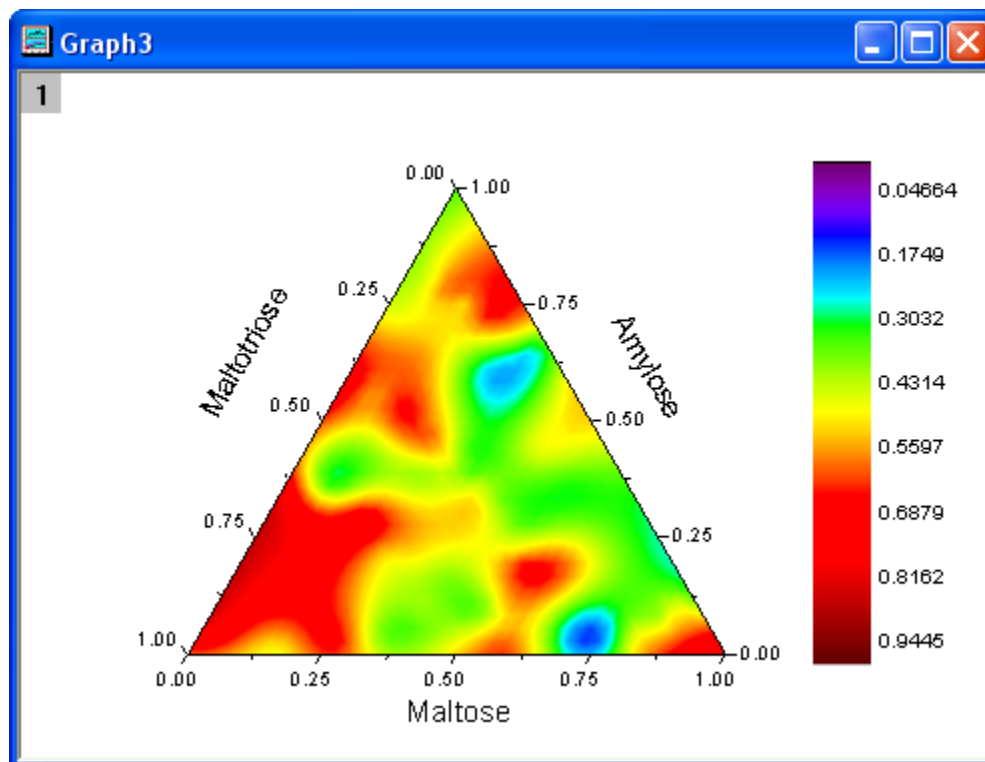


3.10 Ternary

3.10.1 Ternary Contour

Summary

This tutorial will show you how to create a Ternary Contour graph.



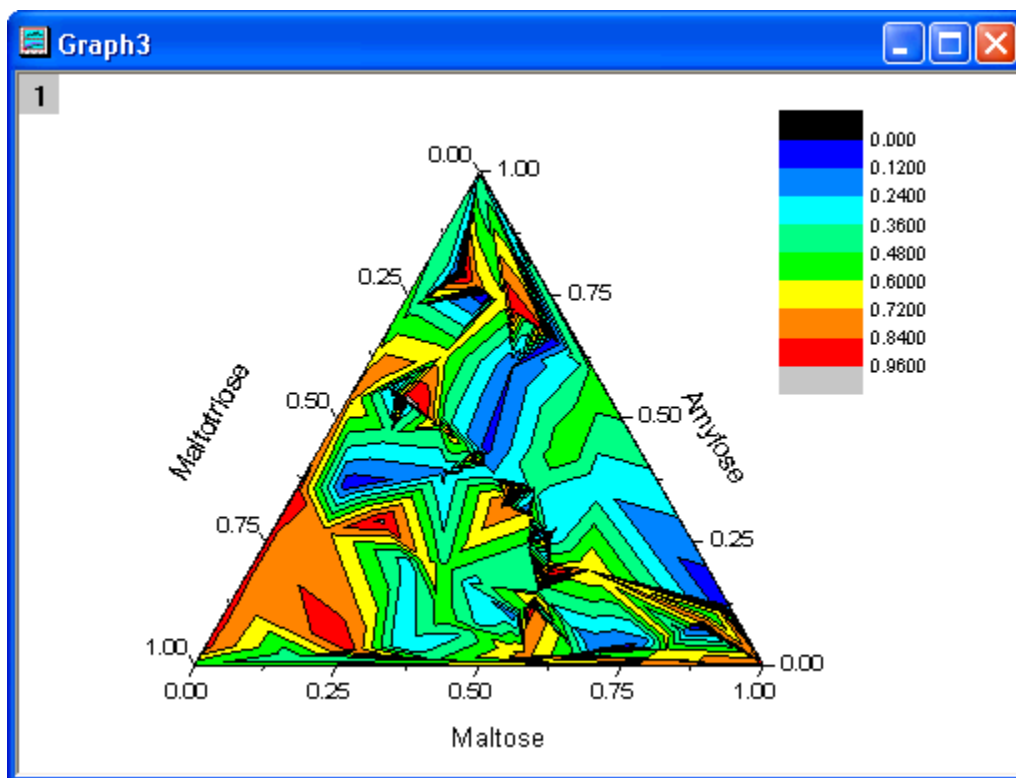
What will you learn

- Create the Ternary Contour graph
- Customize the graph by using the Plot Details dialog

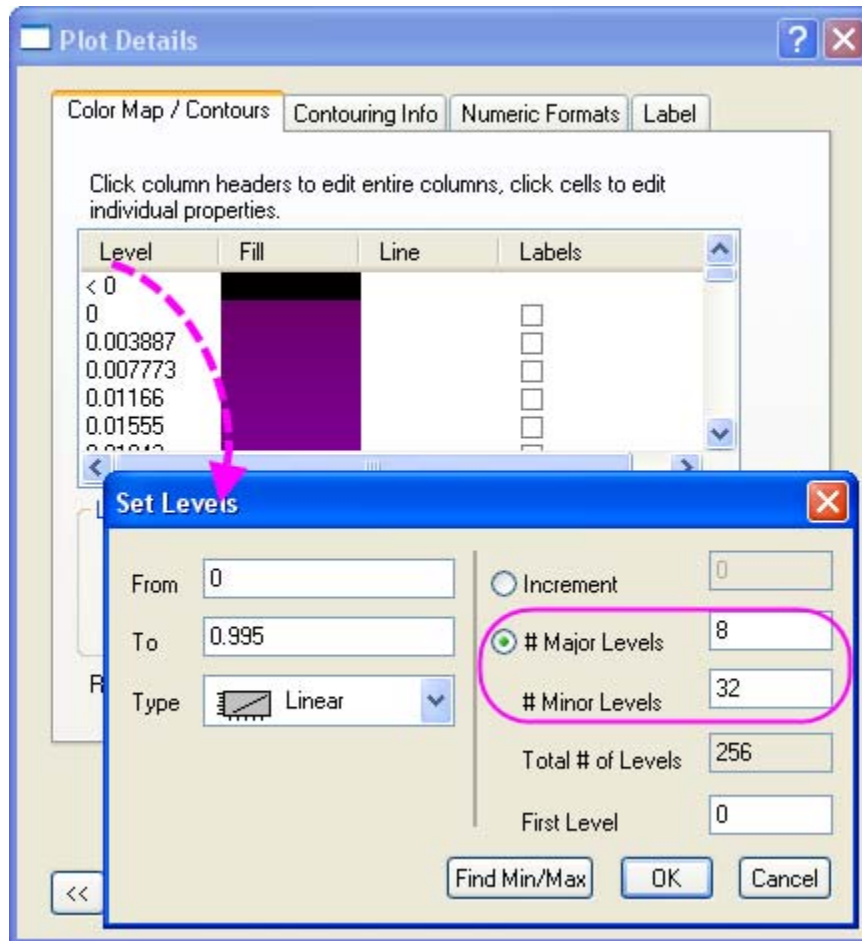
Steps

This tutorial is associated with the sample project: `\Samples\Samples.opj`.

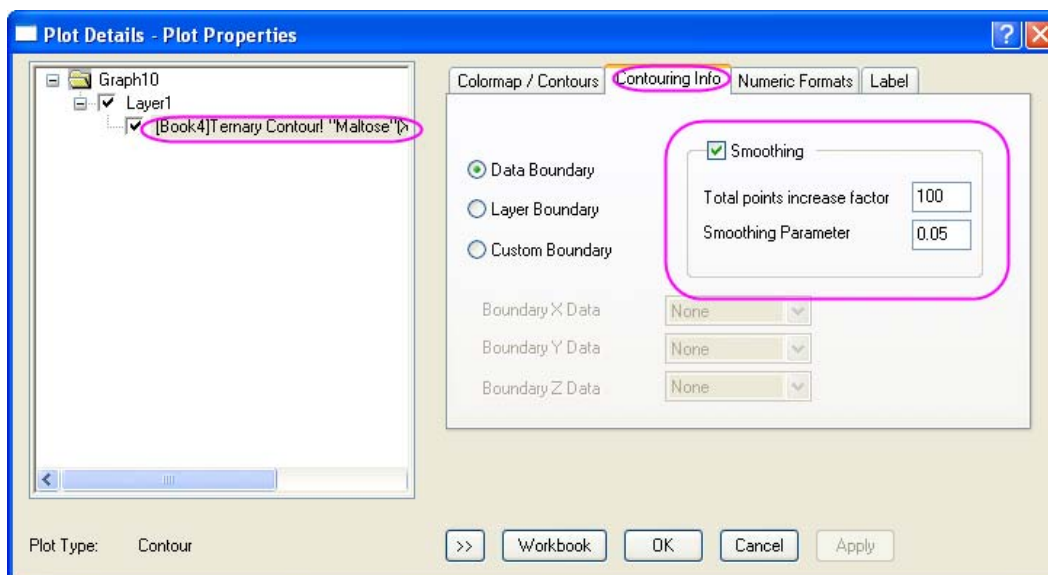
1. Open the *Graphing: Contour: Ternary Contour* folder in the **Project Explorer**. Activate **Book4**, highlight entire worksheet and select **Graph: Contour: Ternary Contour** to create a Ternary Contour graph. The graph should look like:



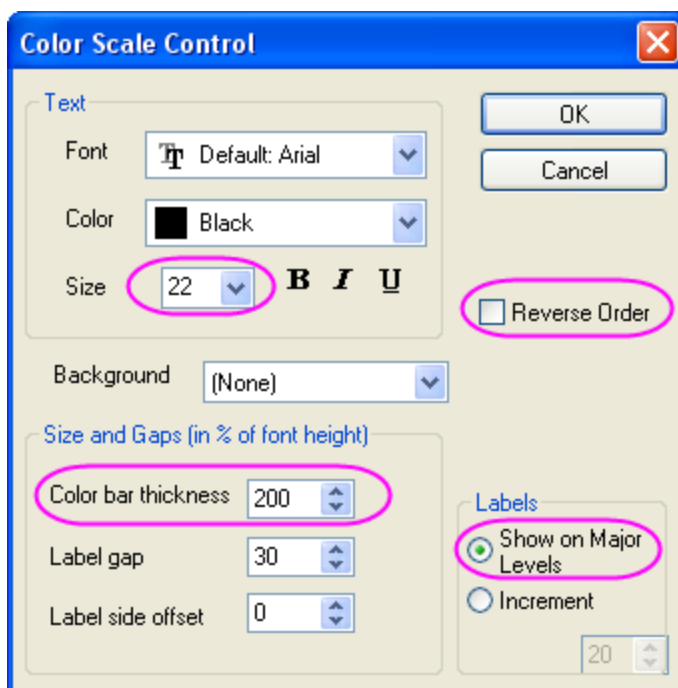
2. Now we will customize the contour graph with the Plot Details dialog. Double-click on the contour plot to bring up **Plot Details** dialog.
 - select the **Color Map/Contours** tab and click the **Level** heading, then set the dialog as the following graph shows.



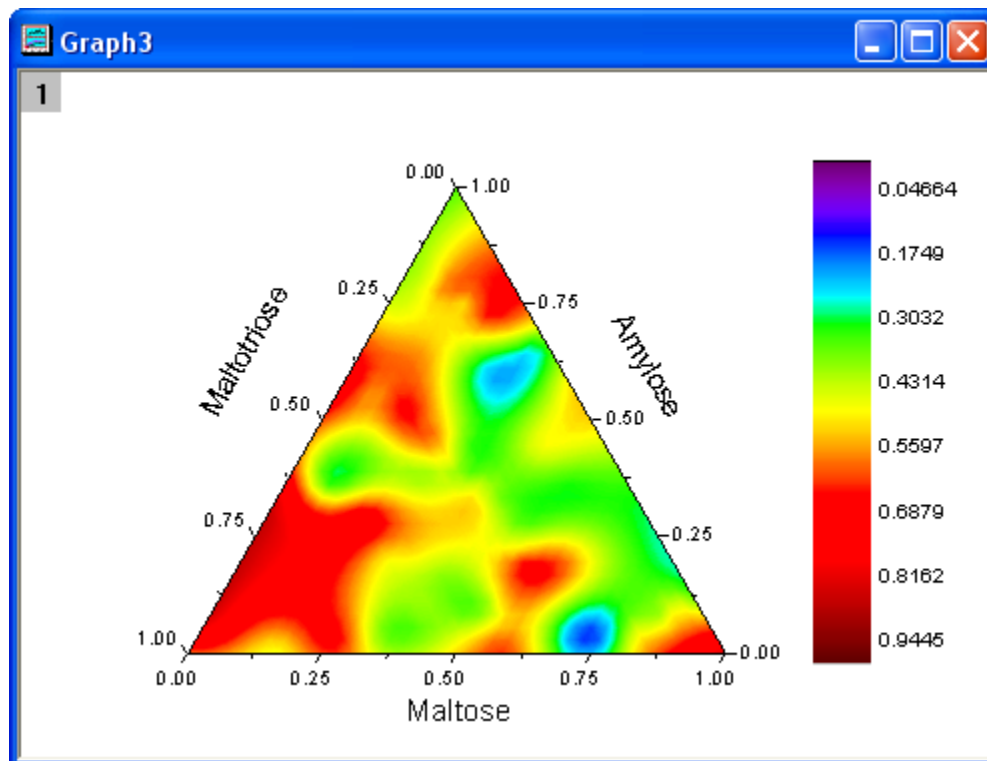
- Click OK to close the **Set Levels** dialog then click the **Fill** heading to open the **Fill** dialog. Select **Load Palette** and then select **Rainbow** from the **Palette** list and click OK.
- Click the **Lines** heading, uncheck the **Show on Major Level only** check box and then select the **Hide All** option and click OK.
- Go to the Contouring Info tab, check the **Smoothing** check box.



- Click OK to close the **Plot Details** dialog.
- Double click on the **Color Scale** to open the **Color Scale Control** dialog and set the dialog as the following graph shows.



The graph should look like



3.11 Waterfall

3.11.1 Waterfall with Y and Z Color Mapping

Summary

Origin's **Waterfall** plot can use a dataset stored in a parameter row to set the Z offset and color map with Y or Z values.

What you will learn

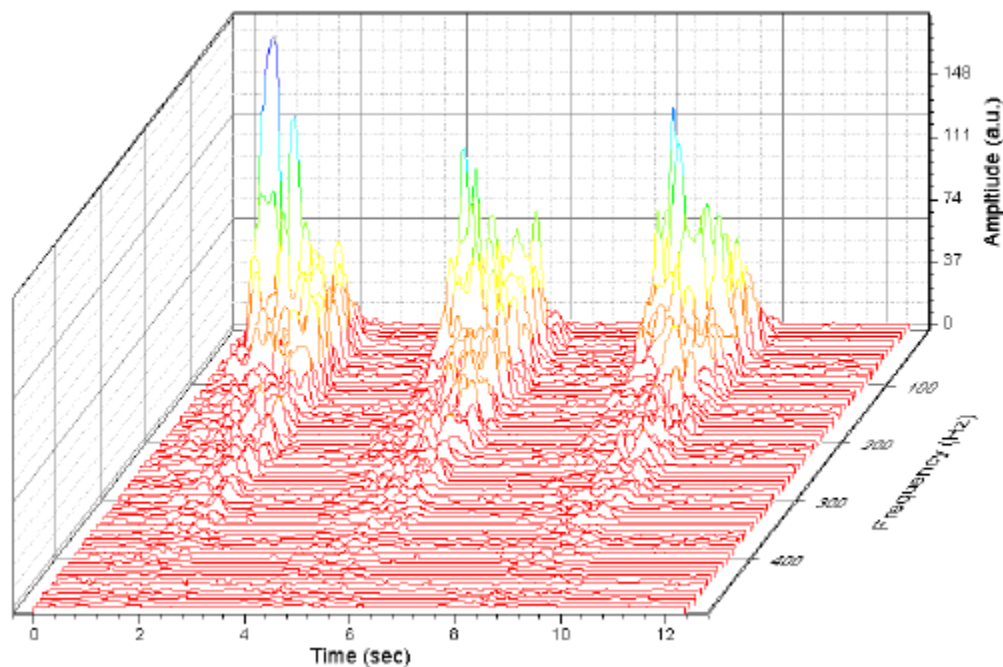
- Create a Waterfall color map with Y or Z values
- Customize colormap levels and palette

Steps

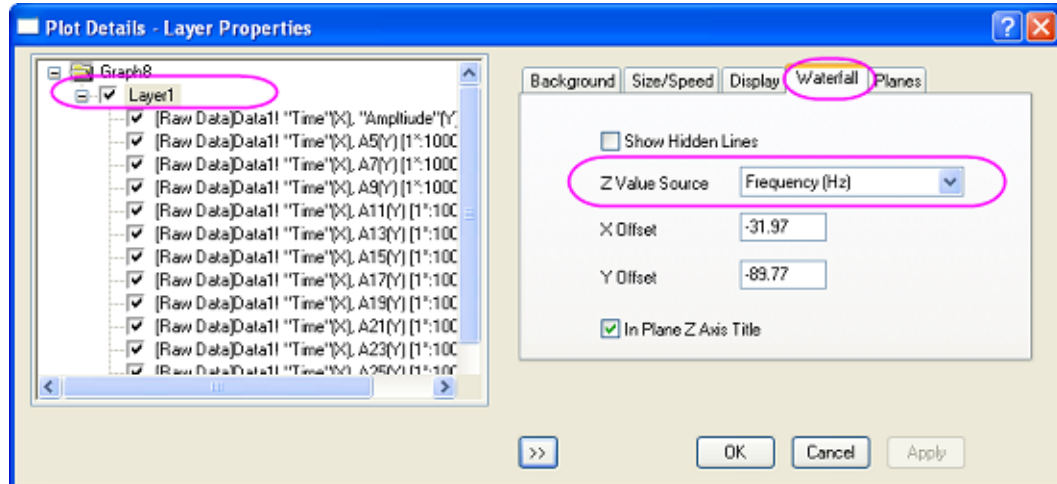
This tutorial is associated with the **Waterfall** folder in the **Sample Graphs** project (\Samples\Samples.opj).

Create Waterfall with Y Colormap

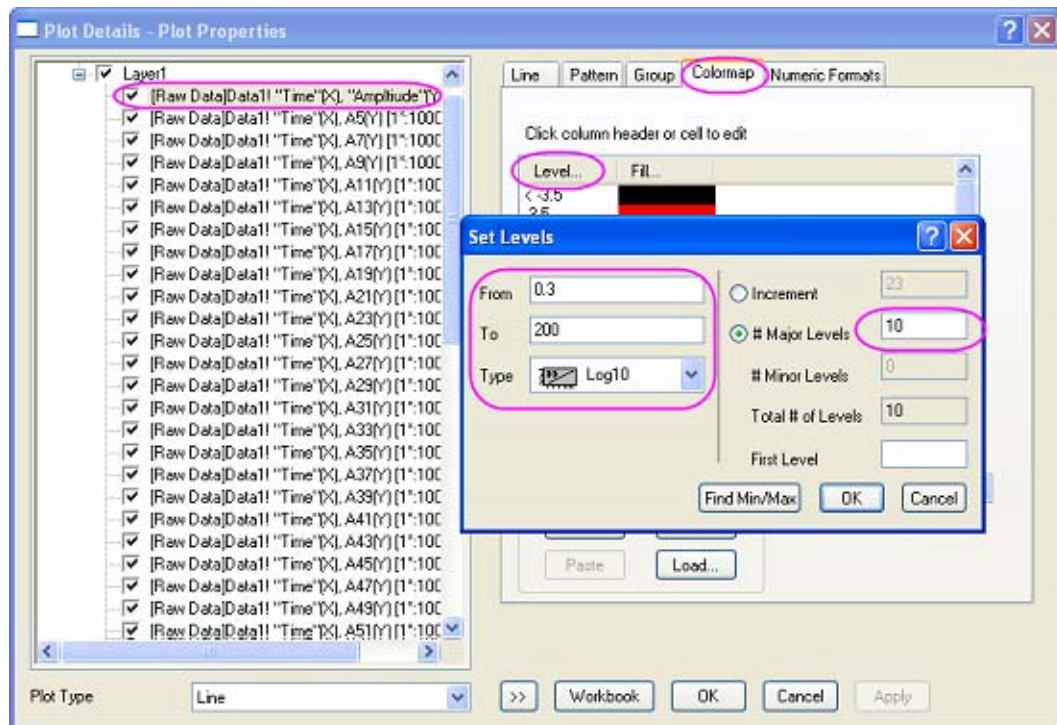
1. In the Project Explorer (usually at the left of the screen), browse to **\Samples\Graphing\Waterfall**. Select the **Book4G** window to activate. Click in the upper left corner of the worksheet to select all data.
2. Right-click and select **Plot** from the menu, then point to **Multi-Curve**, and then select **Waterfall Y:Color Mapping** to create a Waterfall with Y color mapping. (Alternatively, select the **Waterfall Y:Color Mapping** button from the 2D Graphs toolbar.)



Note that the values stored in the user-defined parameter row **Frequency (Hz)** have been automatically picked up as the Z Value Source. Confirm this by double-clicking the plot to open the **Plot Details** dialog box and selecting **Layer1** in the left panel. Select the **Waterfall** tab and observe that the **Z Value Source** is set to **Frequency (Hz)**.

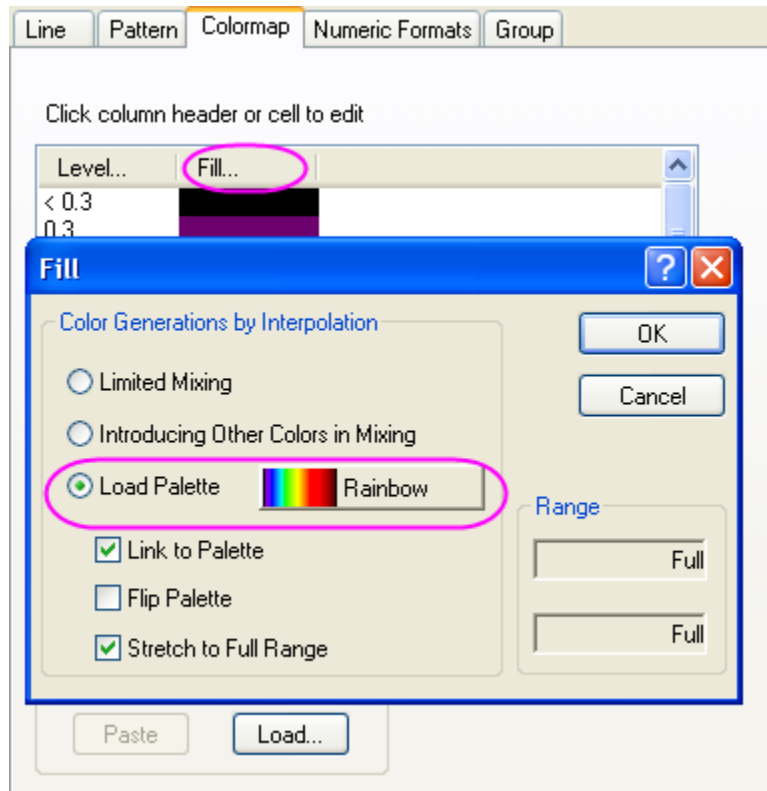


- With the Plot Details dialog box still open, in the left panel, select the first line plot under Layer1. Select the **Colormap** tab in the right panel. Click on the **Level?** column header to open the **Set Levels** dialog box. Set levels as shown below:



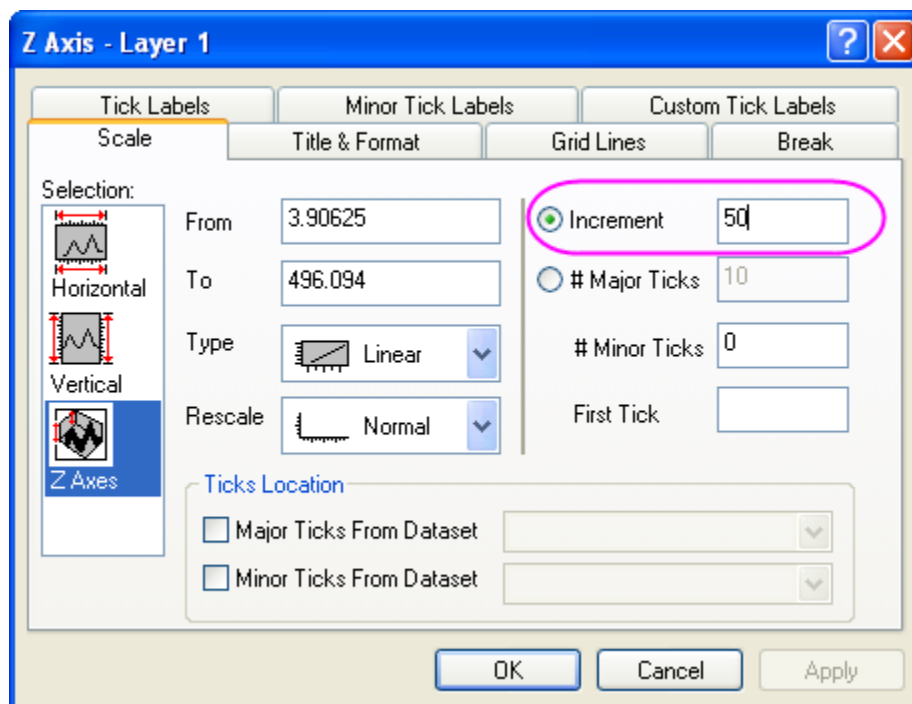
Click **OK** to close the Set Levels dialog box.

- Click on the **Fill?** column header to open the **Fill** dialog box. Select **Load Palette**, and choose the **Rainbow** palette from the palette list. Click **OK** button to close the **Fill** dialog box.

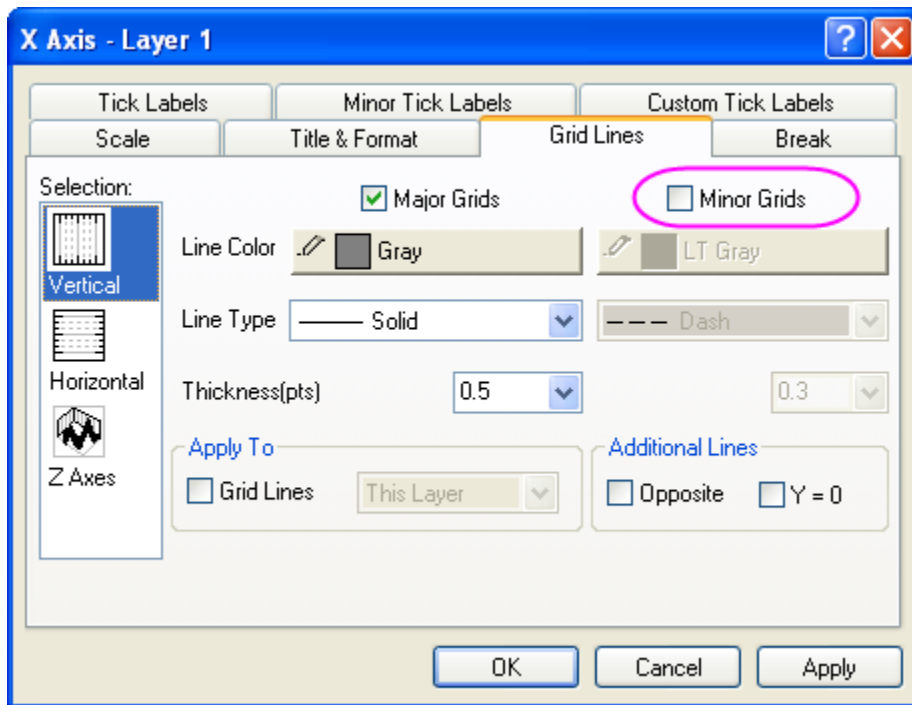


Click **OK** to apply these settings and close the **Plot Details** dialog box.

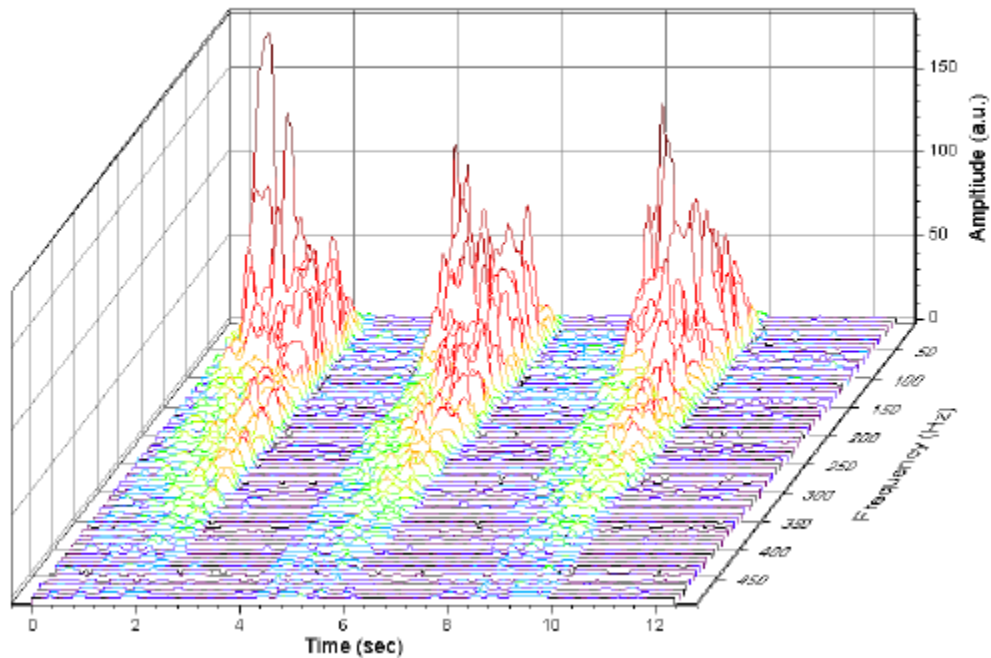
- Double-click the plot's Z axis to open the Axis dialog box. Set the **Increment** to **50**. Select **Vertical** in the **Selection** box and set the vertical **Increment** to **50**.



Select the **Grid Lines** tab. Select **Horizontal** from the Selection box, and clear the Minor Grids check box. Select **Vertical** from the **Selection** box, and clear the **Minor Grids** check box.

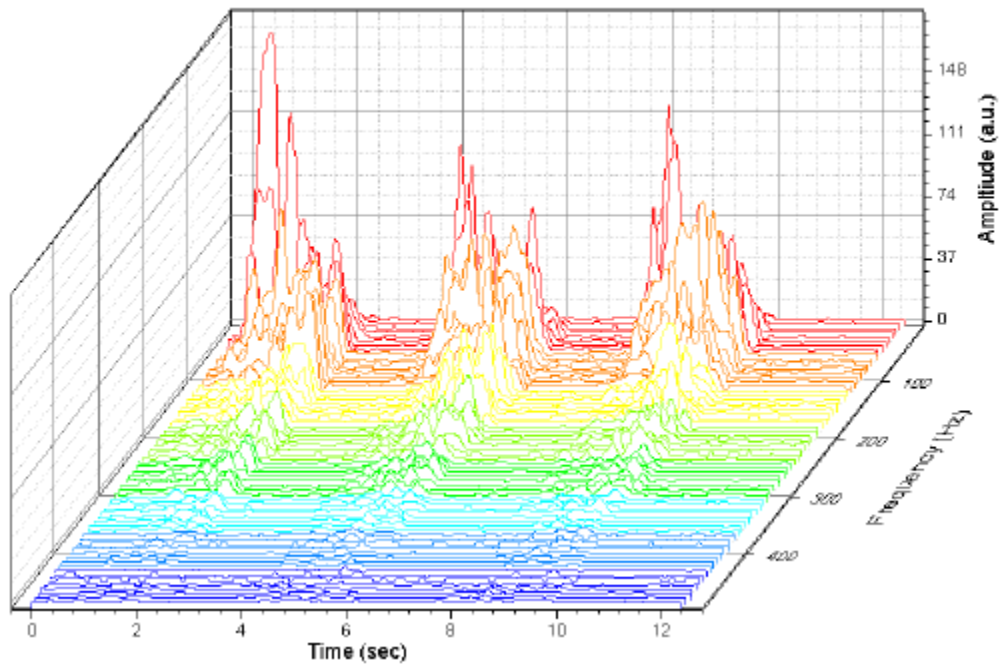


Click **OK** to apply changes and close the dialog box. The final graph should look like this:

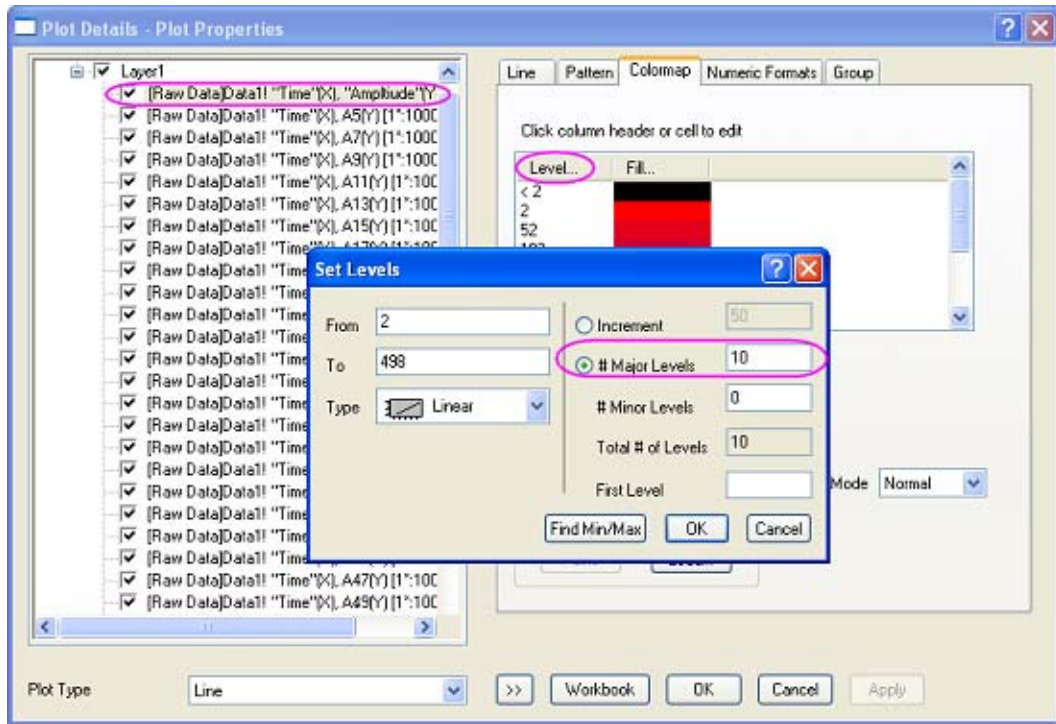


Create Waterfall with Z Colormap

1. In the Project Explorer (usually at the left of the screen), browse to **\Samples\Graphing\Waterfall**. Select the **Book4G** window to activate. Click in the upper left corner of the worksheet to select all data.
2. On the main menu, click **Plot**, then point to **Multi-Curve**, and then select **Waterfall Z:Color Mapping** to create a Waterfall with Z color mapping. (Alternatively, select the **Waterfall Z:Color Mapping** button from the 2D Graphs toolbar.)

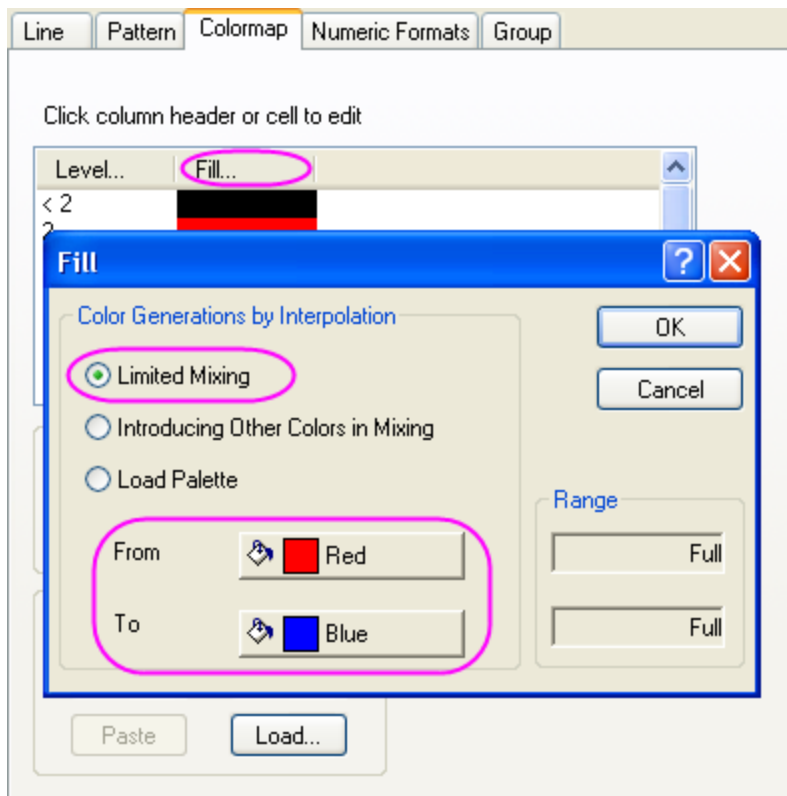


3. Double-click the plot to open the Plot Details dialog box. In the left panel, select the first line plot under Layer1. Select the **Colormap** tab in the right panel. Click the **Level?** column header to open the Set Levels dialog box. Set **Major Levels** to **10**.



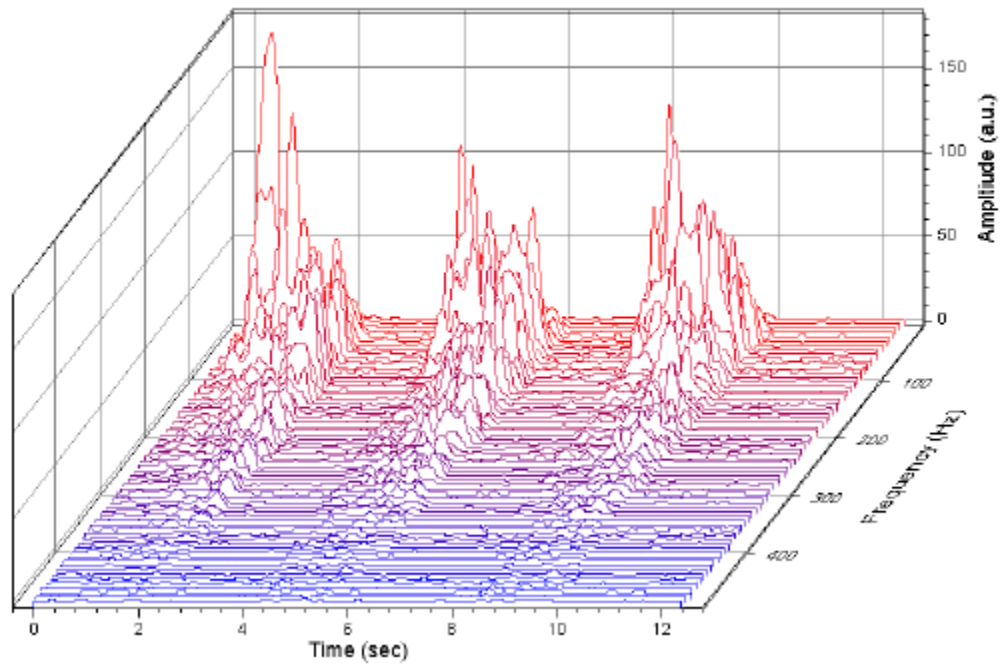
Click **OK** to close the Set Levels dialog box.

- Click the **Fill?** column header to open the Fill dialog box. Select **Limited Mixing** and set the **From** color to **red** and the **To** color to **blue**. Click **OK** to apply these settings and close the dialog box. Click **OK** to close the Plot Details dialog box.



5. Double-click the Y axis to open the **Axis** dialog box. Set the vertical **Increment** to **50**. Select the **Grid Lines** tab. Select **Horizontal** from the **Selection** box, and clear the **Minor Grids** check box. Select **Vertical** from the **Selection** box and clear the **Minor Grids** check box. Click **OK** to close the Axis dialog box.

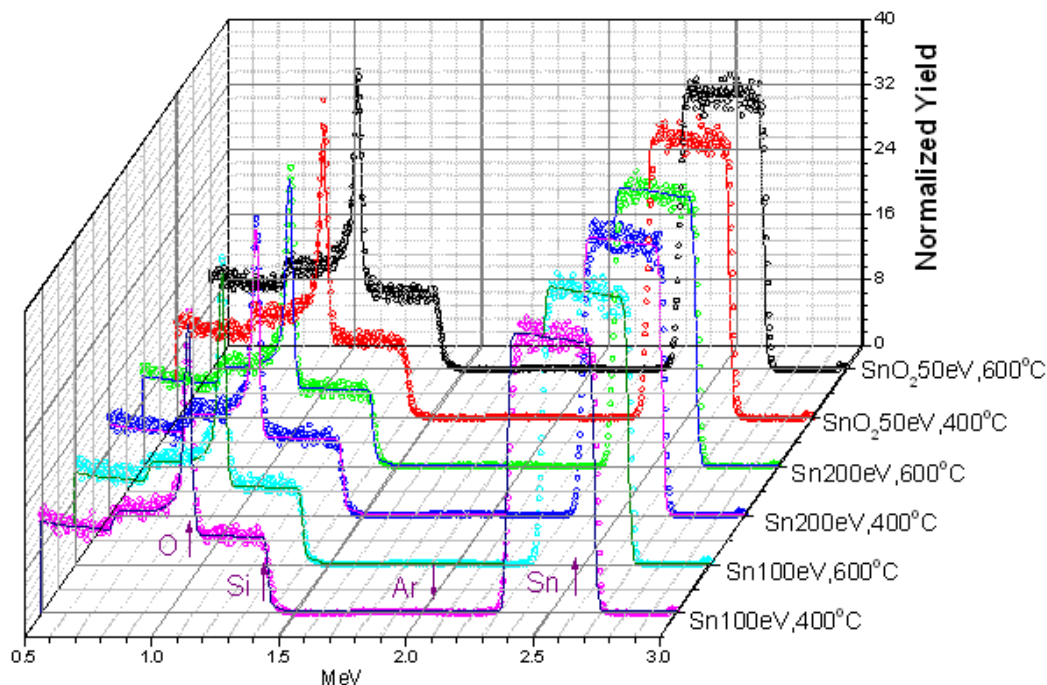
Your final graph should look like this:



3.11.2 3D Waterfall Graph

Summary

The Origin 2D Waterfall graph plots one or more Y columns, or a range from one or more Y columns, as a series of line plots that "recede" into the page. Such graphs are effective for comparing datasets collected under conditions where some parameter is varied incrementally.



Sample of Waterfall graph in Multi-Curve category

What you will learn

- Create waterfall plot
- Change symbol/color for grouped plots
- Merge graphs

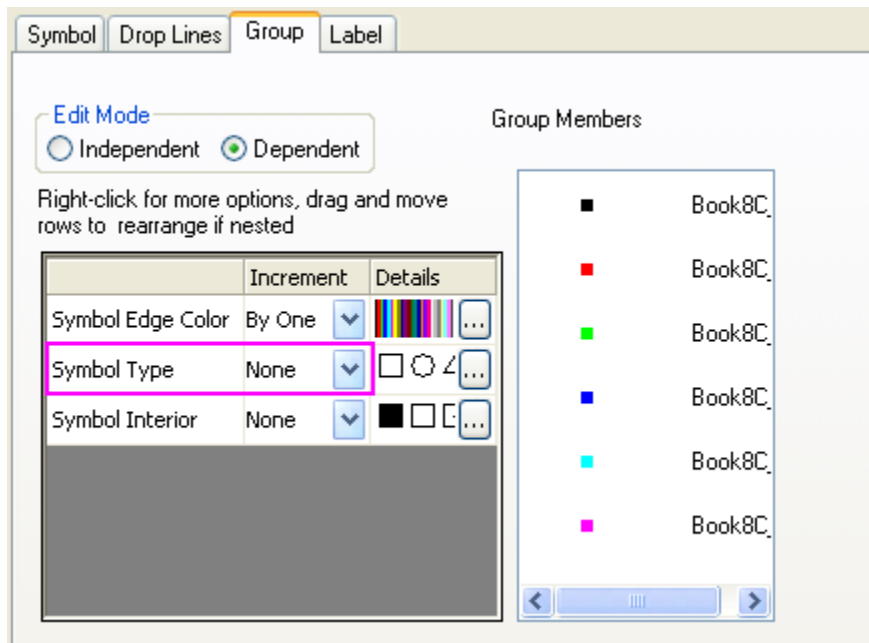
Steps

This tutorial is associated with this graph in Graph Galley. Download the project file in this graph galley page.

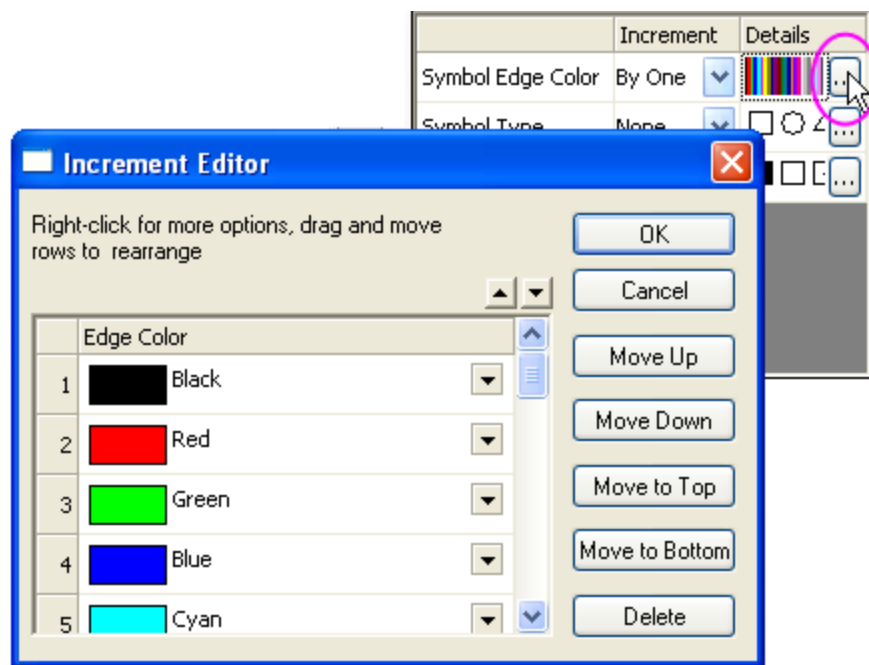
There are two worksheets and one graph in this folder. This example illustrates how to create such a waterfall plot.

1. For each worksheet, highlight all columns and plot as Waterfall graph from **Plot: Multi-Curve: Waterfall**, and use *Measured* and *Simulated* as graph long name, respectively.
2. Double-click the X or Y axis of each plot and adjust axes scales (X Axis From = 0.5, To = 3.0 and Y Axis From = 0, To = 40).
3. Double-click any plot in the *Measured* plot to bring up **Plot Details** dialog at the data level, and change the **Plot Type** drop-down in the bottom left corner to **Scatter** and click **Apply**.

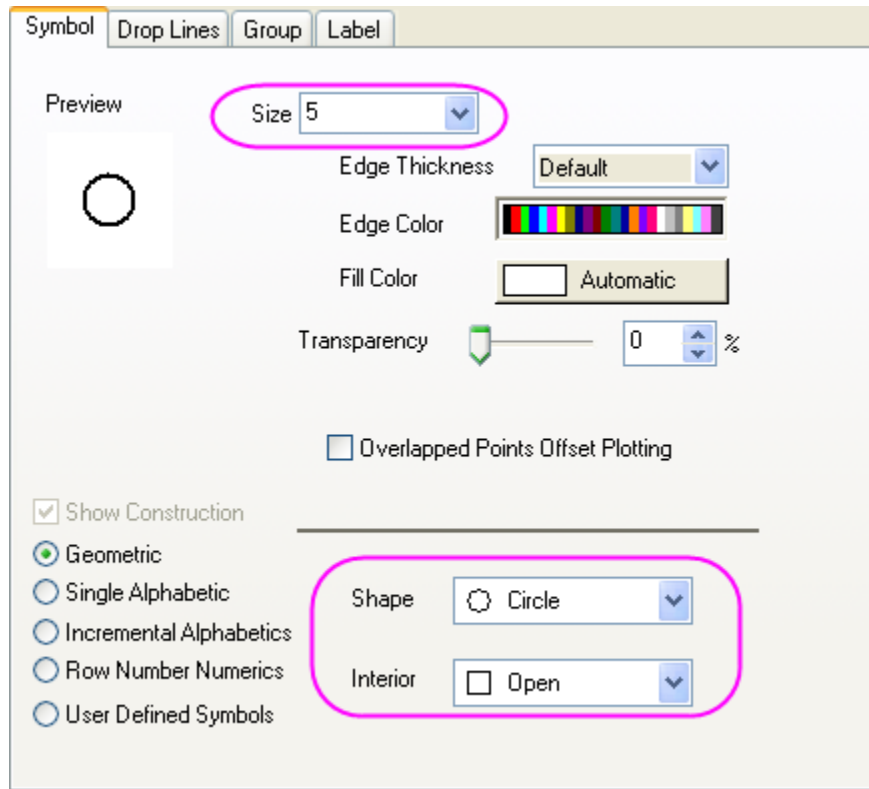
In the **Group** tab of the **Plot Details** dialog, change the **Symbol Type** to **None** and click **Apply**.



Click in Details next to **Symbol Edge Color** and click the Browse button that appears. Note how the color increment is Black, Red, Green, Blue, Cyan and Magenta.

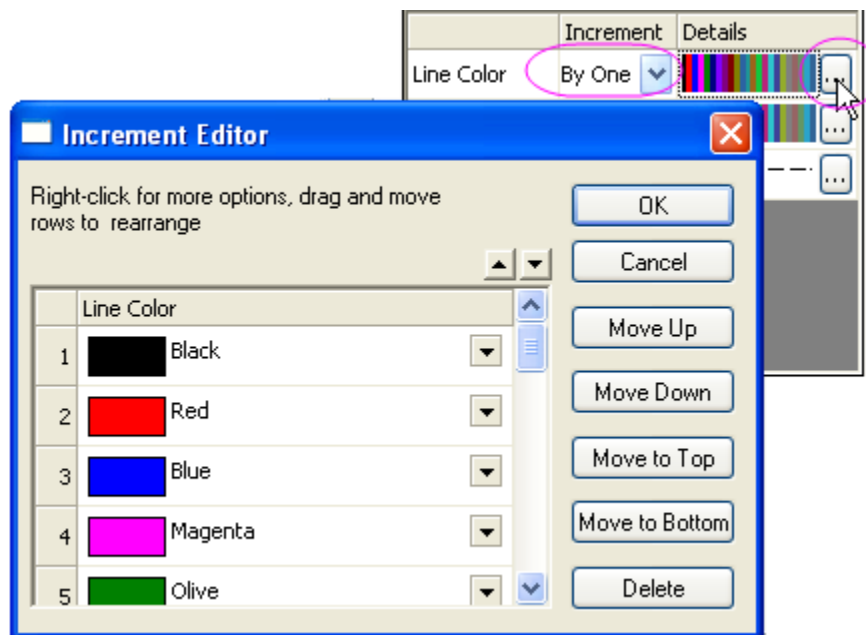


Switch to the **Symbol** tab. Set symbol **Size** to 5, and use **Open** and **Circle** symbol.



Click **OK** to close **Plot Details** dialog.

4. Double-click on any plot in the *Simulated* graph to bring up **Plot Detail** dialog at the data level for this plot. In **Group** tab, change the **Line Color** increment type to **By One**, and change the color Details for **Line Color** to Black, Red, Green, Blue, Cyan and Magenta.



- Click **OK** to close **Plot Details** dialog.
- Now, we will merge these two graphs. Select **Graph: Merge Graph Windows**. Choose **Specified** from **Merge** drop-down list, and select the two graphs you just created in **Graphs** edit box. Arrange the graph by using the following settings:

Arrange Settings:

Number of Rows:	1
Number of Columns:	1
Linked Layers:	Check/Enable

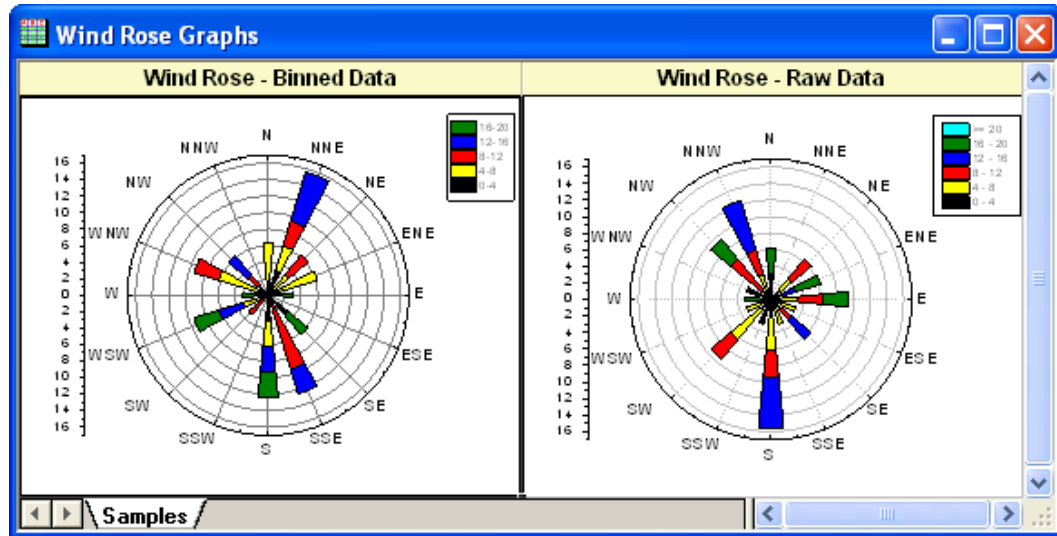
- Click the **OK** button to merge the graphs.
- In the sample we have added labels and changed axis titles.

3.12 Specialized

3.12.1 Windrose Graph

Summary

Windrose graphs are used to present wind speed data and wind direction data collected over some time at a particular location. This tutorial will show you how to create a windrose graph with Origin 8 (post-SR5 builds are required. To create windrose graph with *Raw Data* mode, Origin 8.1 is required).



Minimum Origin Version Required: Origin 8.5.1

What will you learn

This tutorial will show you how to:

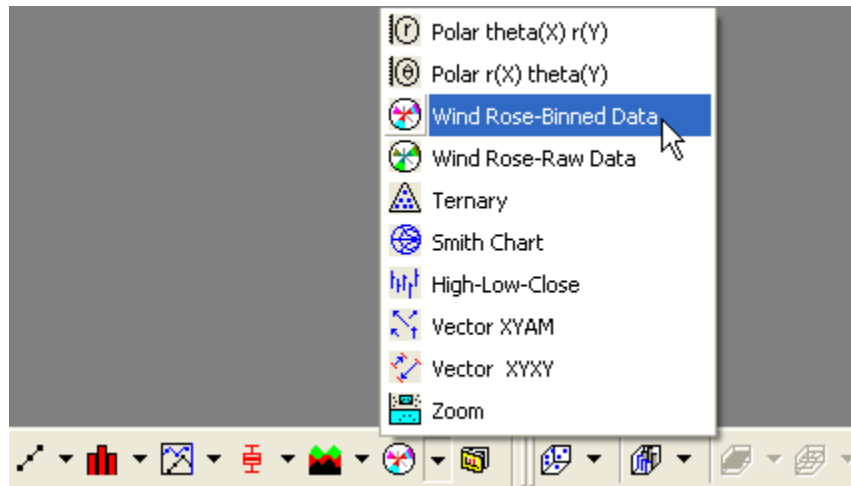
- Create a Windrose graph from binned data.
- Customize the direction tick labels of the Windrose graph.
- Create a Windrose graph from raw data.

Steps

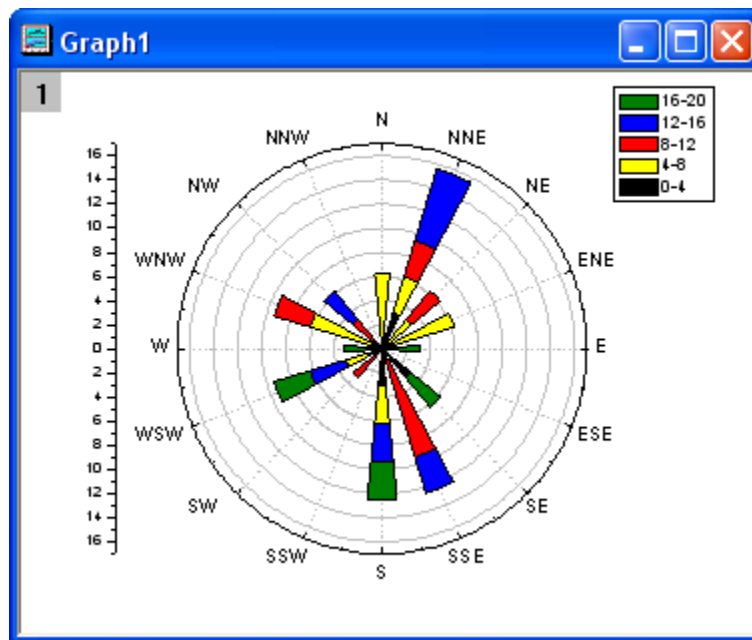
This tutorial is associated with the sample project: `\Samples\Samples.opj`.

Part 1: Create a Windrose Graph from Binned Data

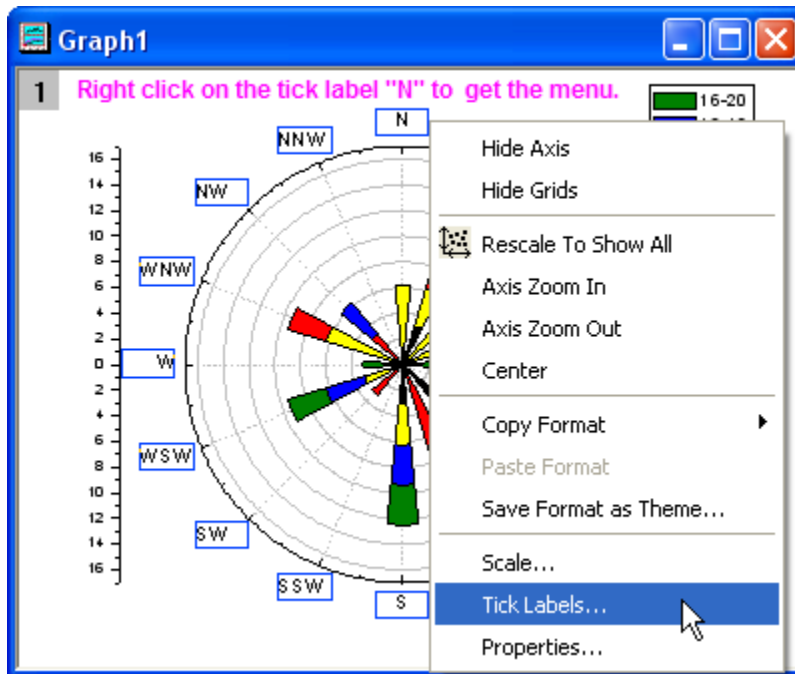
1. Go to the project folder *Graphing: Specialized: Wind Rose* and activate the *Windrose Binned Data* workbook. (If you don't have the project file, please import the sample data from here).
2. Highlight all the columns and click the **Wind Rose - Binned Data** button from the **2D Graphs** toolbar.



The Windrose graph should look like the following:



- Now, change the wind direction to show only N,E,S,W. Right-click on the tick label **N** and then select **Tick Labels** from the short-cut menu. This opens the **X Axis** dialog.



Change the options as the following screenshot.

X Axis - Layer 1

Scale Title & Format Grid Lines Break

Tick Labels Minor Tick Labels Custom Tick Labels

Selection: Show Major Labels

Type: Tick-indexed string

N NNE NE ENE E ESE SE SSE S SSW SW WSW W WNW

Font: Arial

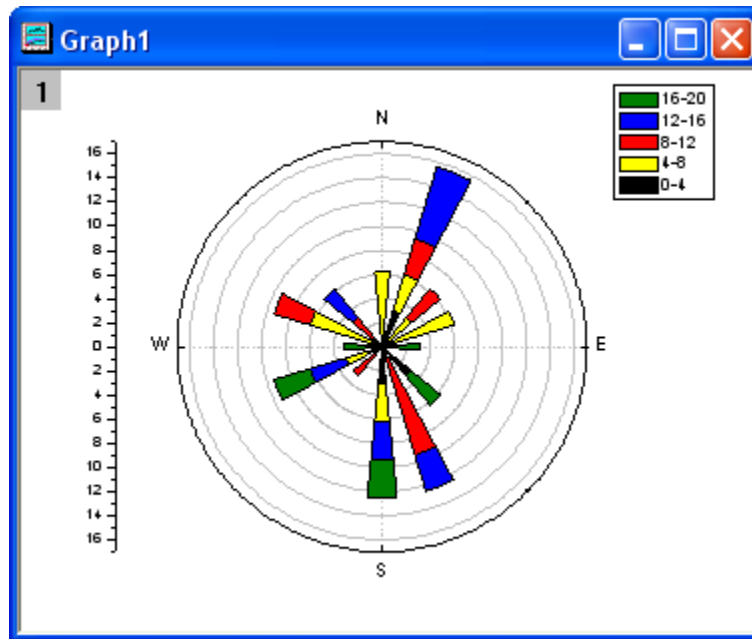
Color: Auto

Point: 18

Apply To: Font This Layer, Point This Layer, Color This Layer

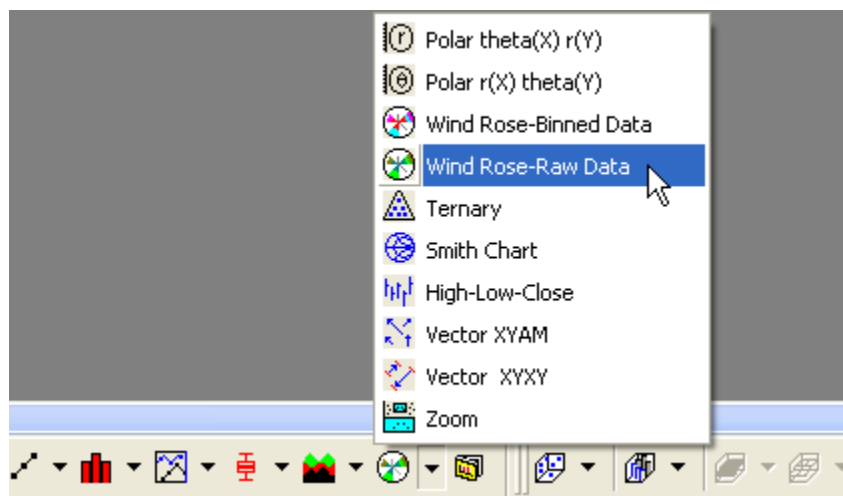
OK Cancel Apply

Activate the **Scale** tab. Enter **90** in the **Increment** edit box and then click the **OK** button. The graph should look like:

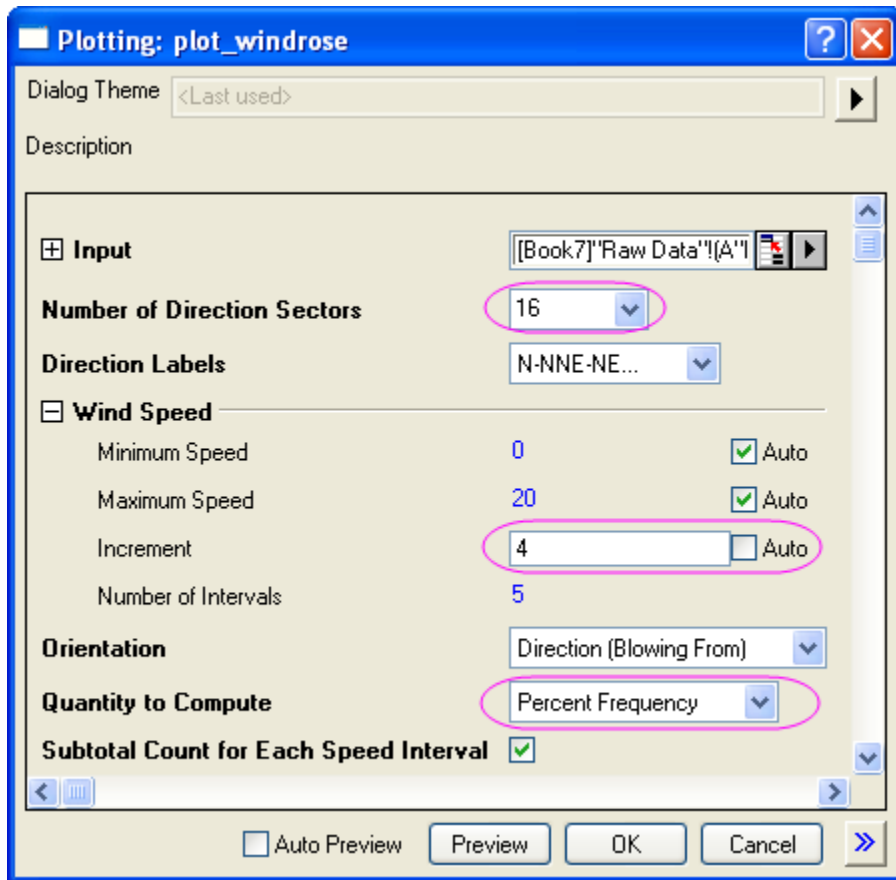


Part 2: Create a Windrose from Raw Data

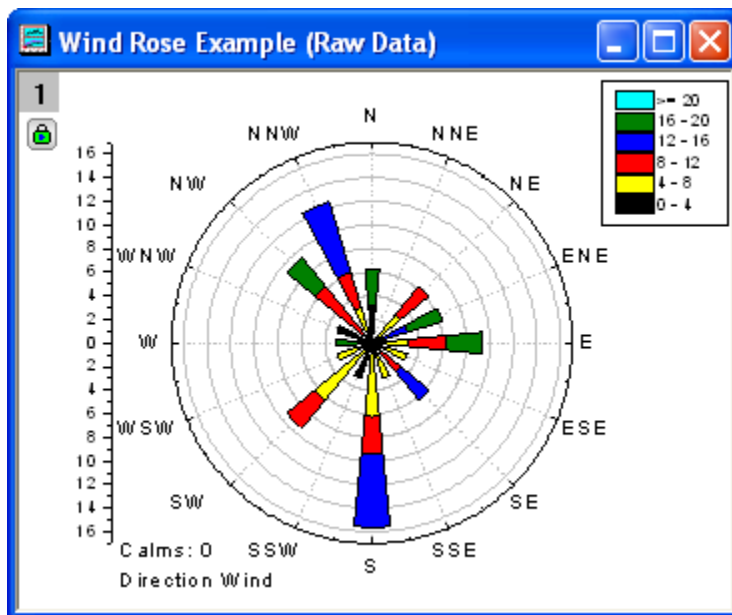
1. Go to the project folder *Specialized: Wind Rose* and activate the *Windrose Raw Data* workbook. Go to the first sheet (the Raw Data worksheet). (If you don't have the project file, please import the sample data from here).
2. Highlight columns A & B. Then click the **Wind Rose - Raw Data** button on the 2D Graphs toolbar.



- Use the settings below to create the Wind Rose graph:



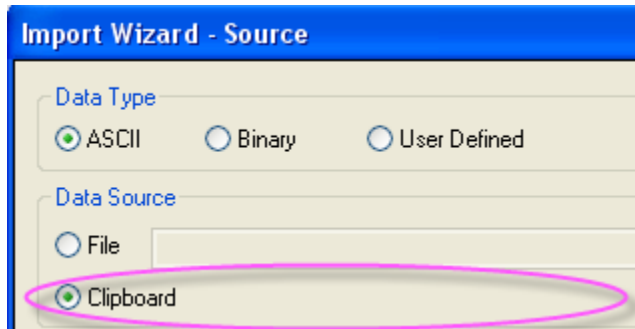
The final graph should look like:



Sample Data

Binned Data

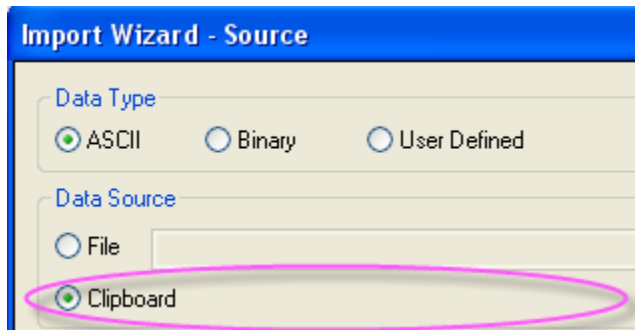
To import **Binned Data** to worksheet, you can copy the **Binned Data 1** (including the heading) and then select **File: Import Wizard**. Select the **Clipboard** checkbox in **DataSource** group and then click **Finish** button to import the data into Origin.



Direction	0-4	4-8	8-12	12-16	16-20
22.5	3.125	3.125	3.125	6.25	0
45	0	3.125	3.125	0	0
67.5	0	6.25	0	0	0
90	0	0	0	0	3.125
112.5	0	0	0	0	0
135	3.125	0	0	0	3.125
157.5	0	0	9.375	3.125	0
180	3.125	3.125	0	3.125	3.125
202.5	0	0	0	0	0
225	0	0	3.125	0	0
247.5	0	3.125	0	3.125	3.125
270	0	0	0	0	3.125
292.5	0	6.25	3.125	0	0
315	0	0	3.125	3.125	0
337.5	0	0	0	0	0
360	0	6.25	0	0	0
382.5	0	0	0	0	0

Raw Data

To import **Raw Data** to worksheet, you can copy the **Raw Data** (including the heading) and then select **File: Import Wizard**. Select the **Clipboard** checkbox in **DataSource** group and then click **Finish** button to import the data into Origin.

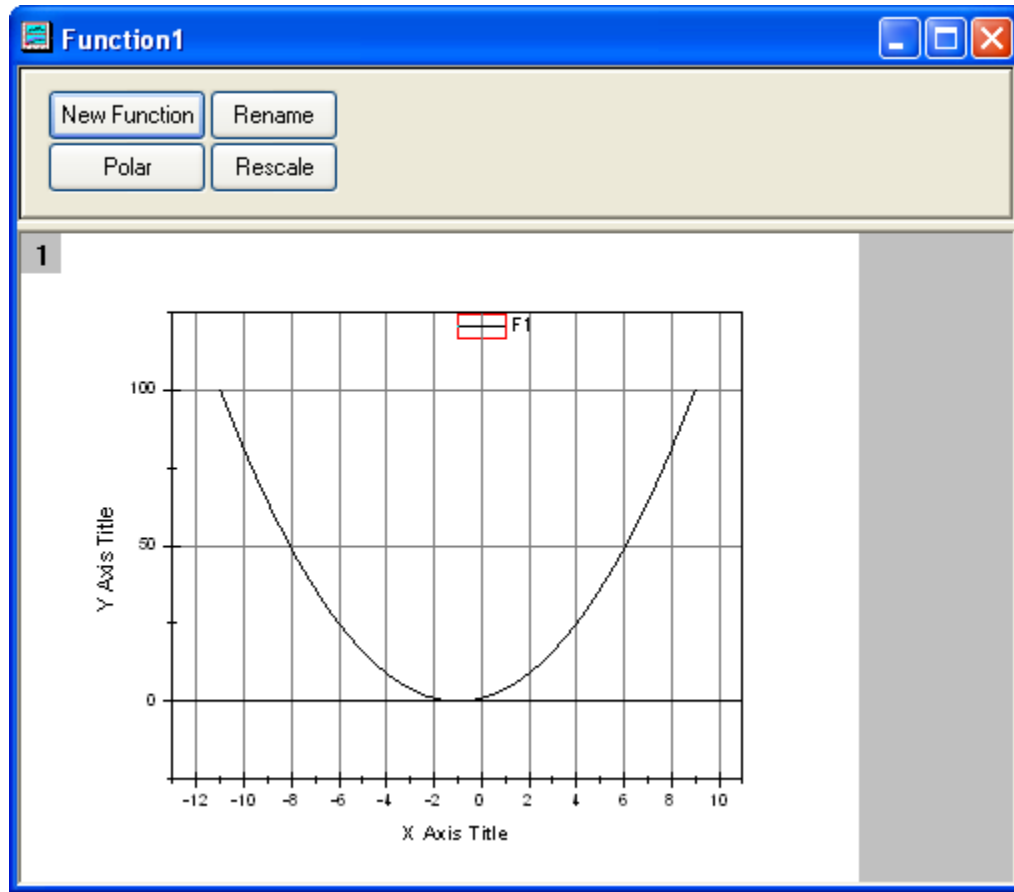


Direction	Speed
311.5	12.75
142.7	11.18
161.6	5.9
277.3	8.24
155.3	13.46
40.8	8.57
43.4	4.38
1.3	10.91
78.8	18.72
237.8	16.22
114.6	0.88
2.1	12.05
290.5	4.6
174.1	3.29
267.6	16.64
8	5.6
213.9	8.04
134.8	17.26
137.6	11.87
46.1	5.48
4.5	13.47
311.2	10.17
154.4	11.17
176.2	18.91
348.1	4.11
225.2	6.65
236.4	12.87
11.7	1.07
278.9	4.36
356.5	5.01
58.9	7.3
161.8	15.6

3.12.2 Plot Functions with Parameters Defined in a Worksheet

Summary

Origin can plot functions. It also can plot functions with parameters defined in a worksheet. The function graph can be updated automatically as the parameters in the worksheet change.



Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:


- Define variables from a worksheet in the **Set Values** dialog box.
- Plot a function graph with parameters.
- Update a graph automatically when parameters are changed.

Steps

Let us use this function as an example: $y=p_0+p_1*x+p_2*x^2$

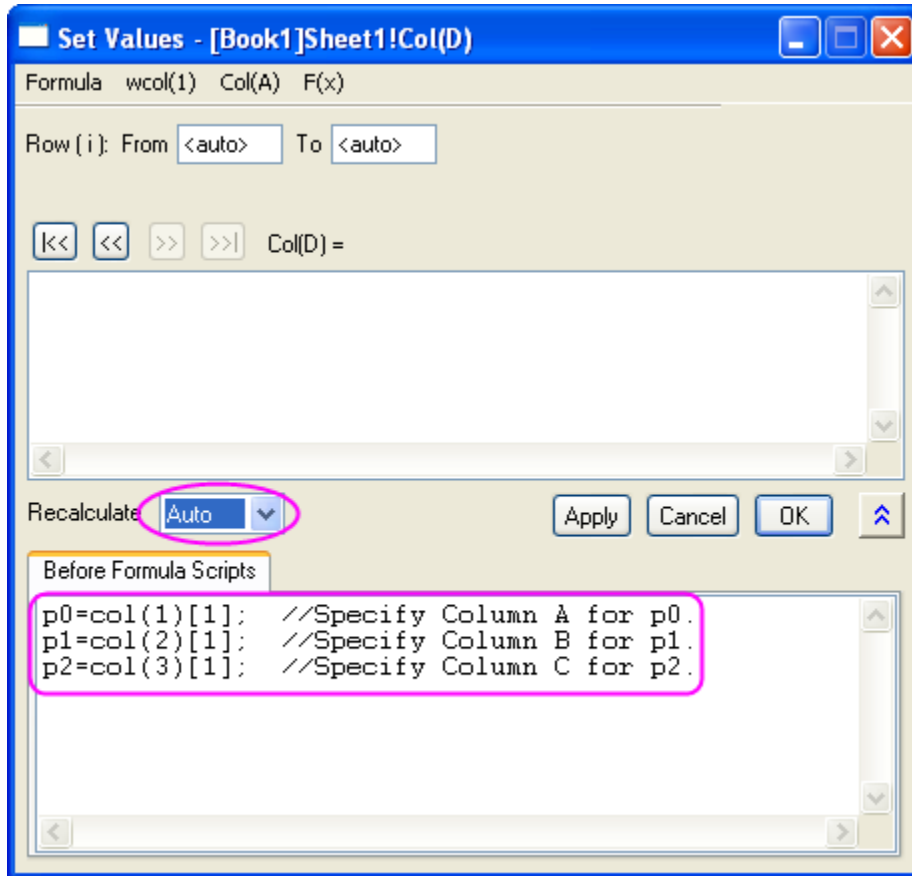
1. Set up a worksheet with three parameters p_0 , p_1 , p_2 stored in Column A, Column B, Column C as shown below.


	A(Y)	B(Y)	C(Y)
Long Name	p0	p1	p2
Units			
Comments			
1	1	2	1
2			

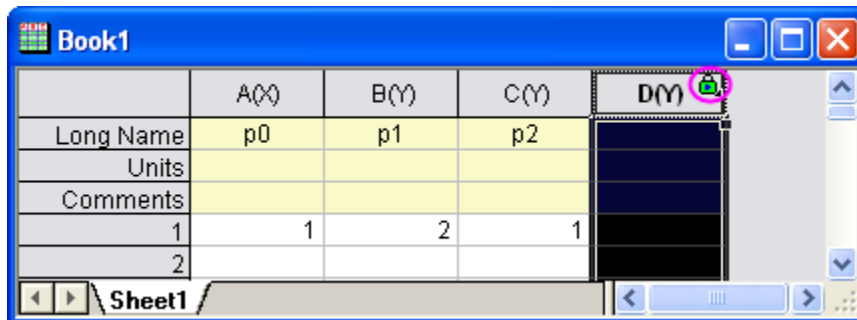
1. Click on the **Add New Columns** button  on the **Standard** toolbar to add a new column to the worksheet.


	A(Y)	B(Y)	C(Y)	D(Y)
Long Name	p0	p1	p2	
Units				
Comments				
1	1	2	1	
2				

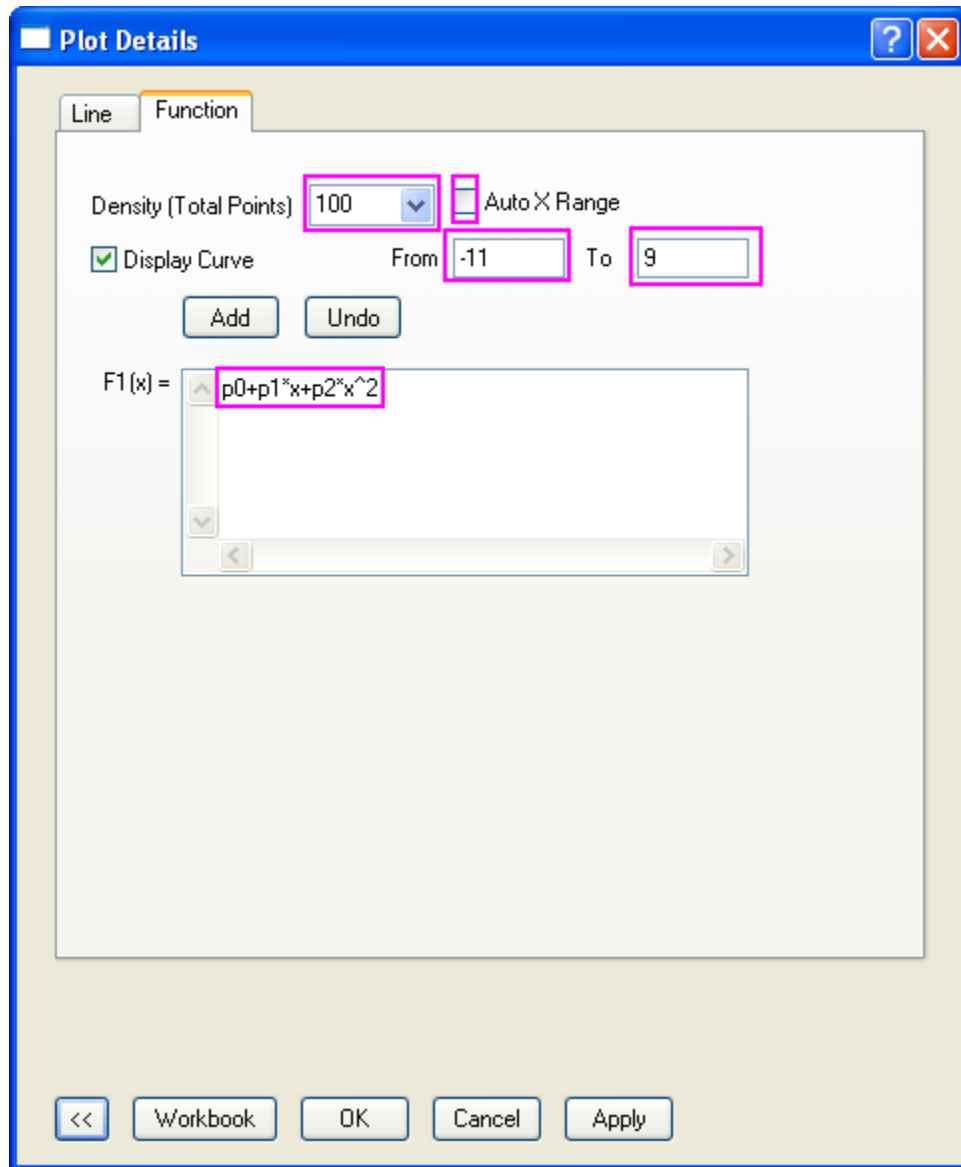
2. Highlight Column D and then select **Column: Set Column Values**. Select **Auto** from the **Recalculate** drop-down. Type the script shown below to define the parameters in the **Before Formula Scripts** edit box. Click the **OK** button to close the dialog box.



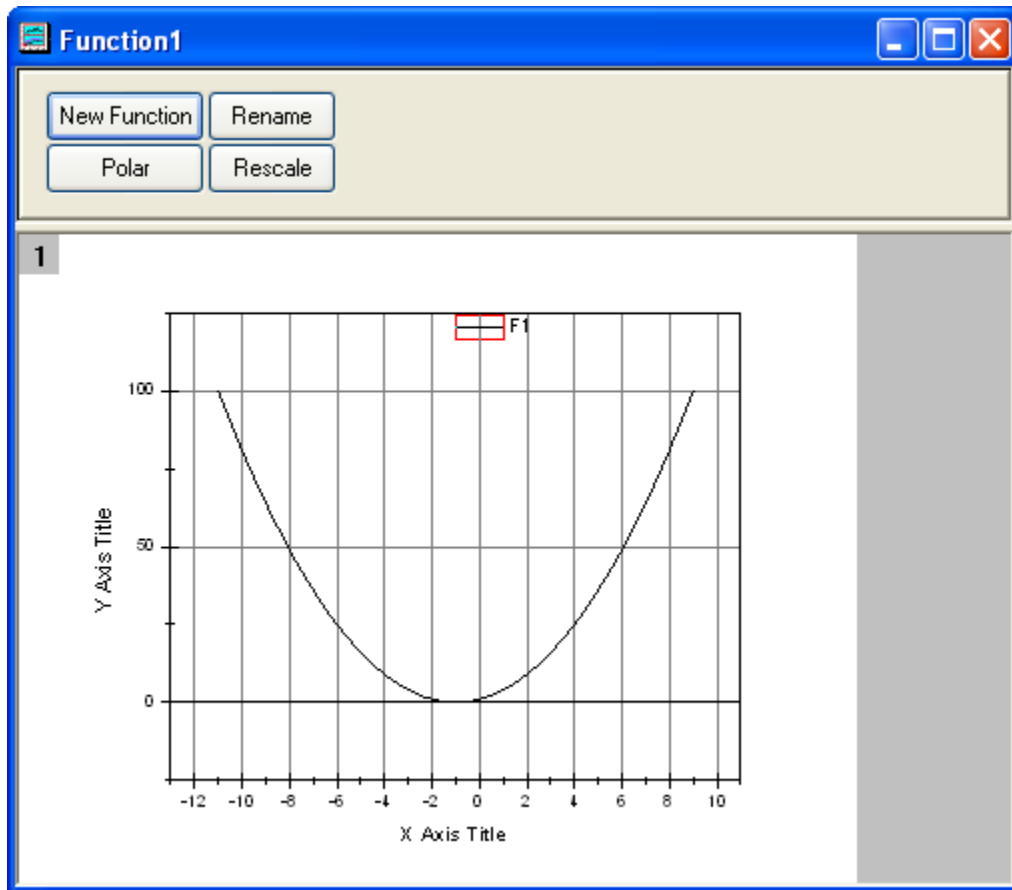
Note that there is a green lock icon  on the top right corner of Column D which indicates that the Recalculate Mode is Auto.



3. Click on the **New Function** button  on the **Standard** toolbar. The **Plot Details** dialog opens.
4. In the **Plot Details** dialog, set the options as follows and click the **OK** button to close the dialog box.



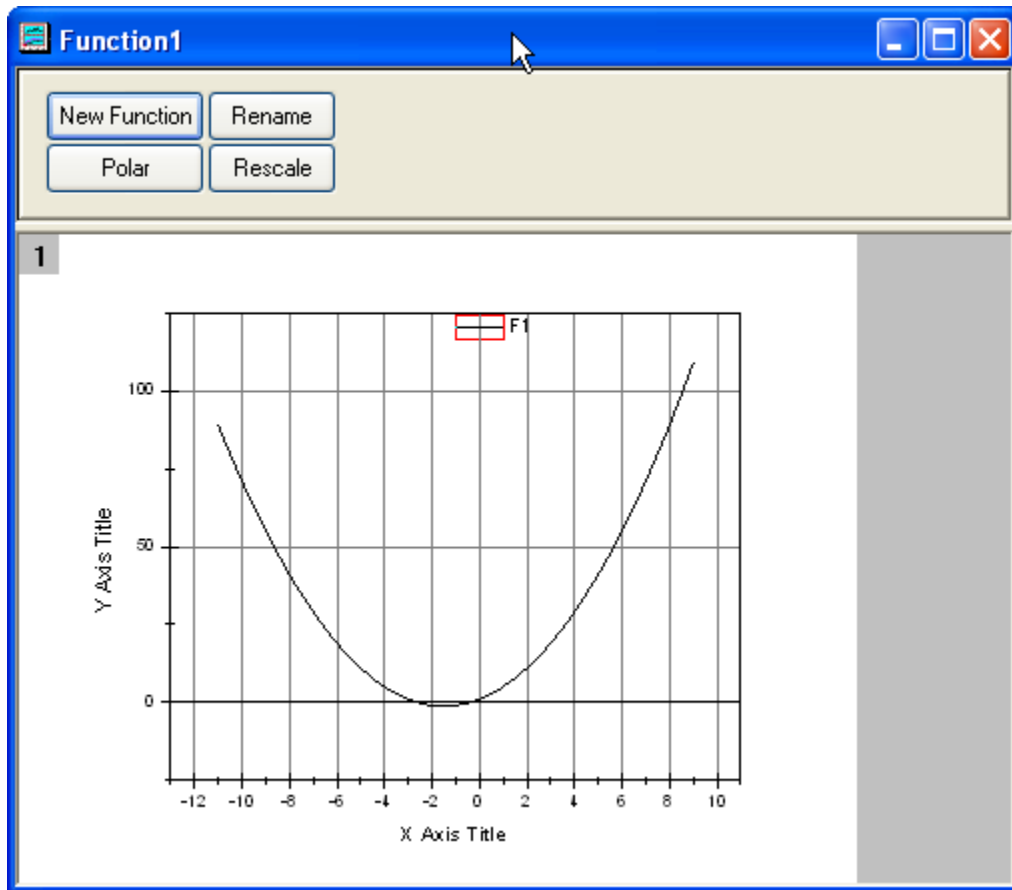
Click on the **Rescale** button  on the **Graph** toolbar to adjust the graph's scales.



5. The function graph can be updated automatically when the parameters in the first row of the worksheet change. Change the value in the first row of Column 2 from 2 to 3. Click outside this cell to finish editing.

	A(Y)	B(Y)	C(Y)	D(Y)
Long Name	p0	p1	p2	
Units				
Comments				
1	1	2	3	1
2				

The function graph updates to reflect this change.



Scripts

The script used in the **Before Formula Scripts** edit box of the **Set Values** dialog box is:

```
p0=col(1)[1]; //Specify Column A for p0.
p1=col(2)[1]; //Specify Column B for p1.
p2=col(3)[1]; //Specify Column C for p2.
```

The function in the **F1(x)** edit box of the **Plot Details** dialog box is as follows:

```
p0+p1*x+p2*x^2
```

3.12.3 Open-High-Low-Close-Volume Stock Chart

Summary

This tutorial will show you how to create a stock chart to display open, high, low, and close prices, together with trading volume.



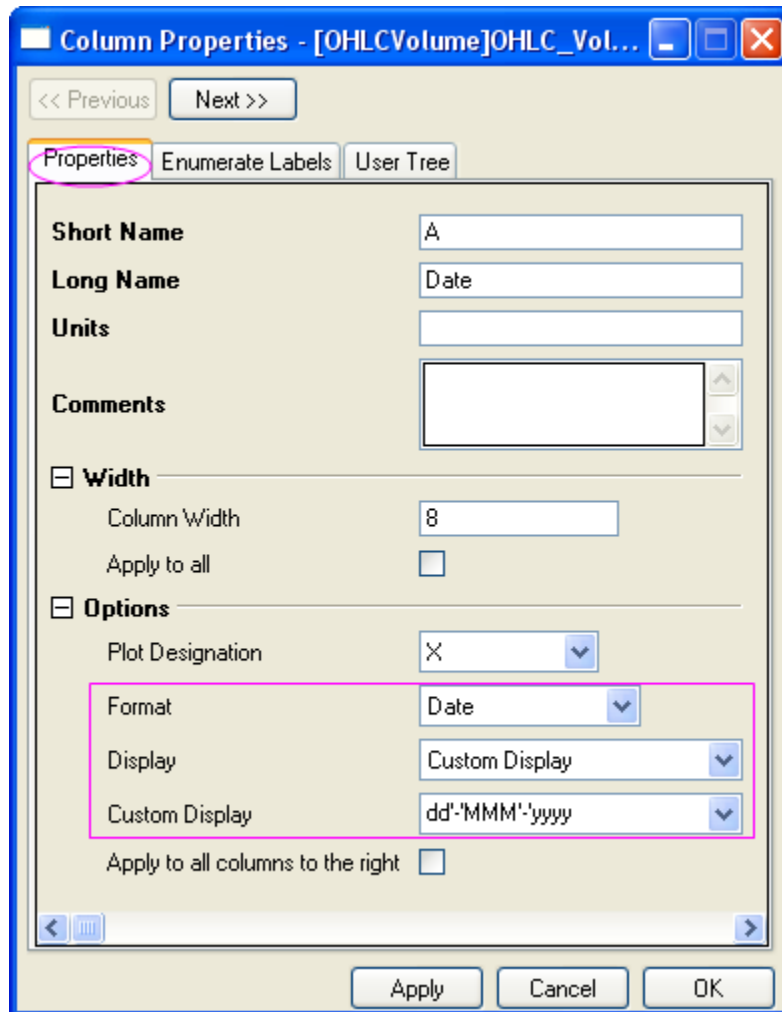
Minimum Origin Version Required: Origin 8.5 SR0

What you will learn

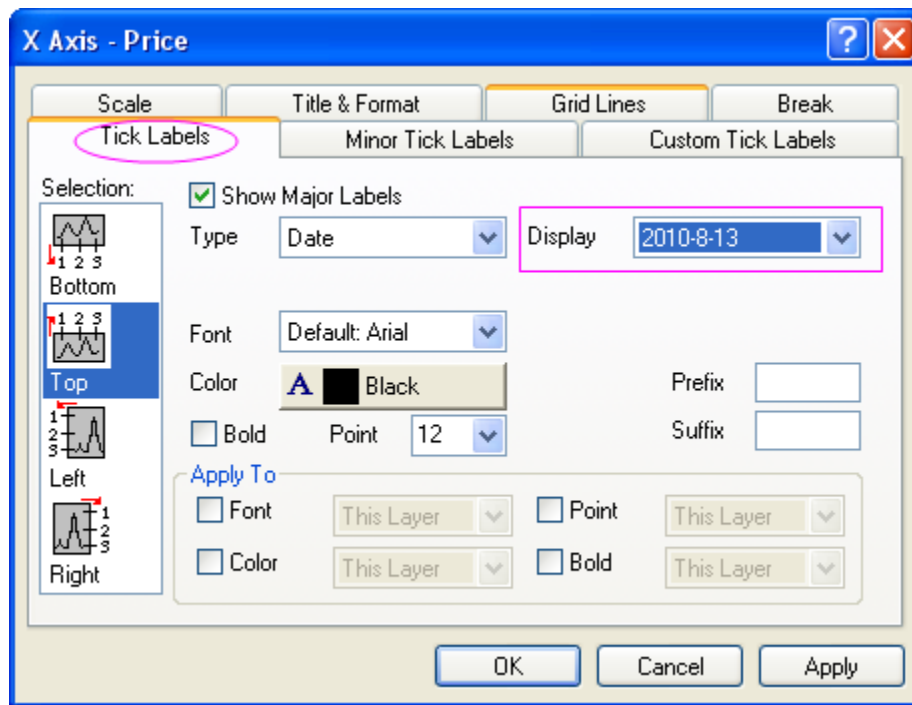
- Set a column format
- Create a stock chart with open, high, low, and close prices, and trading volume
- Set date format for display label of axis

Steps

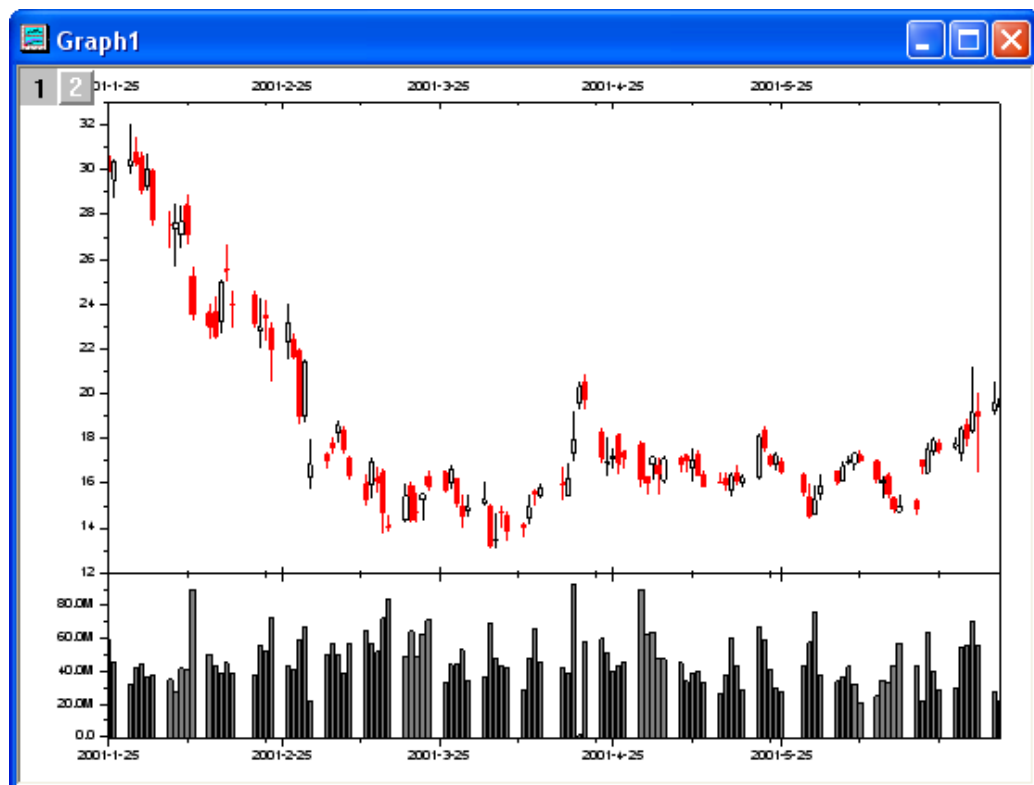
1. Download the source data file from our ftp and import it into Origin. When import this file, keep the system default settings.
2. Double click on the header of column A to bring up the **Column Properties** dialog. Activate the **Properties** tab, under the **Options** branch, Select *Date* from the **Format** drop-down list, and then select *Custom Display* from the **Display** drop-down list. Type "dd'-'MMM'-'yyyy" in the **Custom Display** drop-down list. Click the **OK** button to finish setting column A as Date column.



3. Highlight all columns in the worksheet, and then choose menu item **Plot: Stock: OHLC-Volume** to create a graph.
4. Double click on the top axis of the newly created graph to open the **X Axis** setting dialog. Switch to the **Tick Labels** tab, and then select the first data format in the **Display** drop-down list like the following image shows.



- Click the **OK** button to finish the axis setting and get the stock chart of open, high, low, and close prices, and trading volume.



4 Customizing Graphs

4.1 Customizing a Graph

4.1.1 Summary

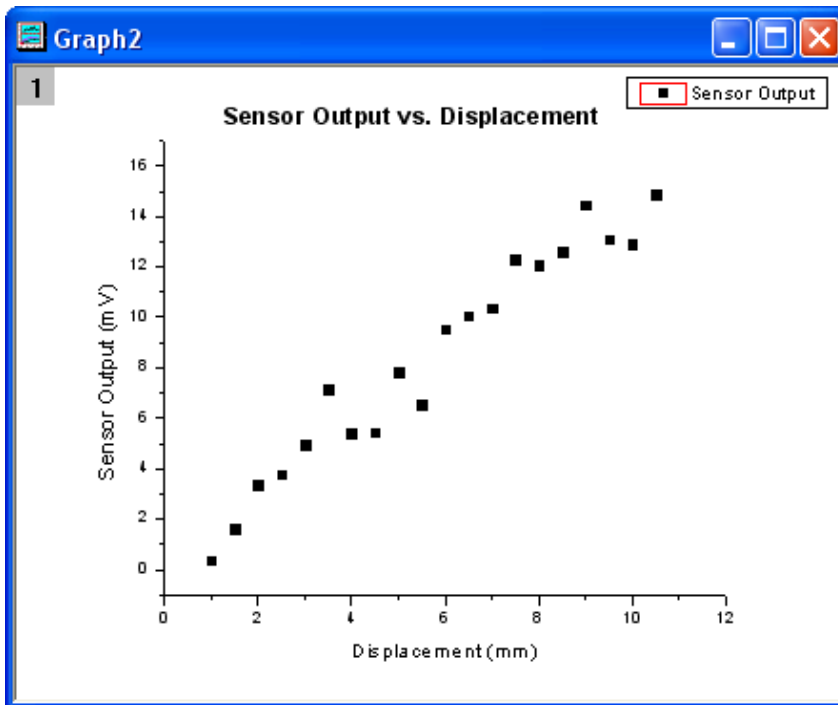
Editing your Origin graph is very easy. Any graph element can be selected and with its associated dialog, the properties can be modified. In fact, you can customize your graph right down to a single data point.

This tutorial will show you how to:

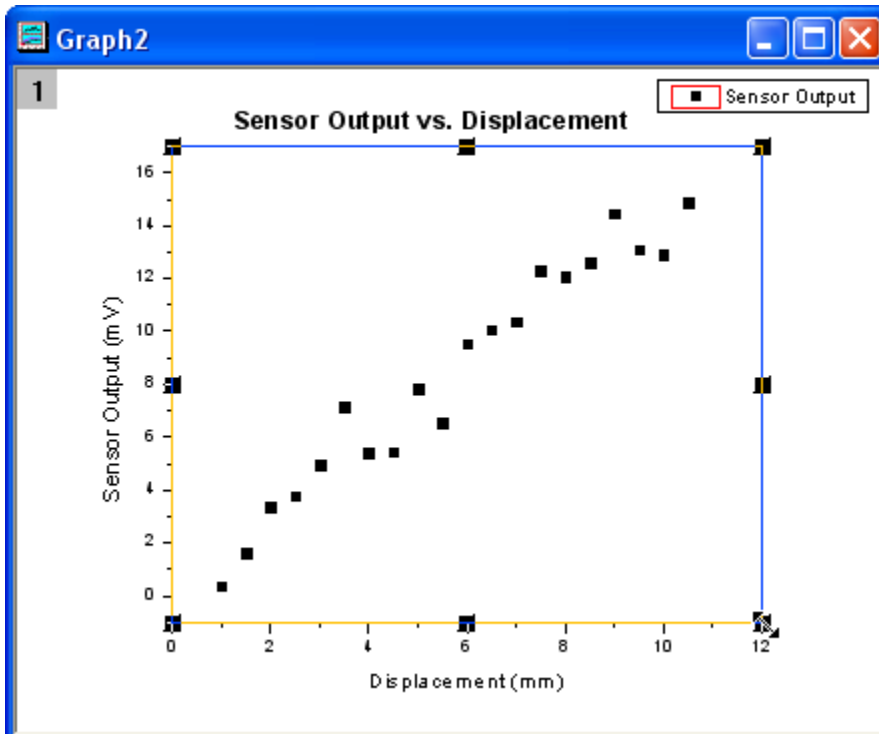
- Resize a Layer
- Add a Layer Title
- Customize and Save a Template
- Customize Axes
- Apply a Graph Theme
- Change Plotting Order in a Layer
- Customize a Point
- Customize a Grouped Plot
- Create a Color-mapped Waterfall Plot

4.1.2 Resize Layer

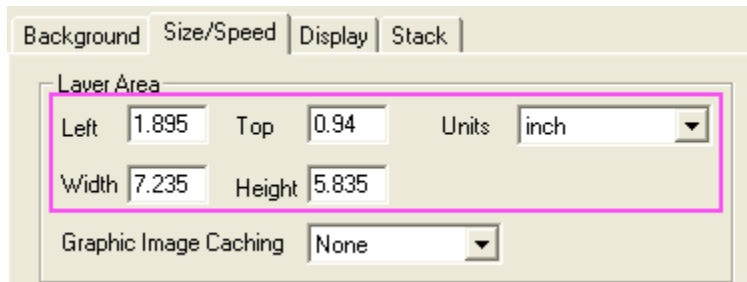
1. Open **Customizing Graphs.OPJ** from the **\Samples\Graphing** folder and select the **Resize Graph and Customize Symbol** folder from the **Project Explorer** window.
2. Make **Graph2** active and right-click inside the layer, above the data points to select **Add/Modify Layer Title** from the context menu that opens. Add a title, as the following graph shows.



The layer can easily be resized graphically by dragging the sizing handles. To resize it, single click inside the layer, but not on a data point. You can left-click in the same space you right-clicked when adding the Layer Title. The layer will become selected as seen below, and then you can drag one of the 8 anchor points to resize the layer. Note: If you hold the Ctrl key down while dragging, the aspect ratio will be maintained.




- You can also use the **Plot Details** dialog to input the size of the layer, in order to resize it precisely. Double-click inside the layer (in the same spot you left-clicked above) to open the **Plot Details** dialog. Go to the **Size/Speed** tab, and set the layer area to the values shown in the image below:

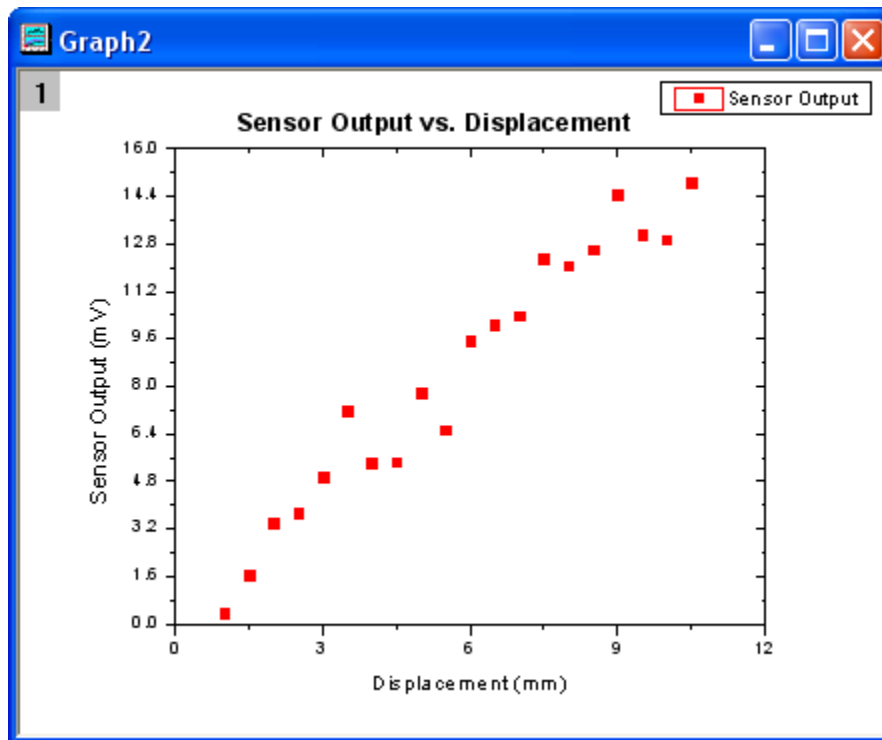


4.1.3 Customize Data Plot and Axes

In this section, we will show you how to change the color of a data plot, and how to change the properties of the axes.

- Click on one of the data points of **Graph2** to select the entire data plot, and then change the color of the data points to **red** by using the **Line/Border Color** button  in the **Style** toolbar.
- Next we will use the **Axis** dialog to customize the axes. Double-click on the X-axis to open the dialog and set the controls as follows:
 - On the **Scale** tab, choose the **# Major Ticks** radio button and input **5** into it. Then select **Vertical** from the **Selection** list box. The name of the dialog changes to **Y Axis** when you do this. Input **0, 16** and **11** into the **From, To** and **# Major Ticks** edit boxes, respectively.
 - On the **Title&Format** tab, select **Top** from the **Selection** list and check the **Show Axis&Ticks** check box to add the top X-axis. Then select **Right** from the **Selection** list and again check the **Show Axis&Ticks** check box to add the right Y-axis. Now the

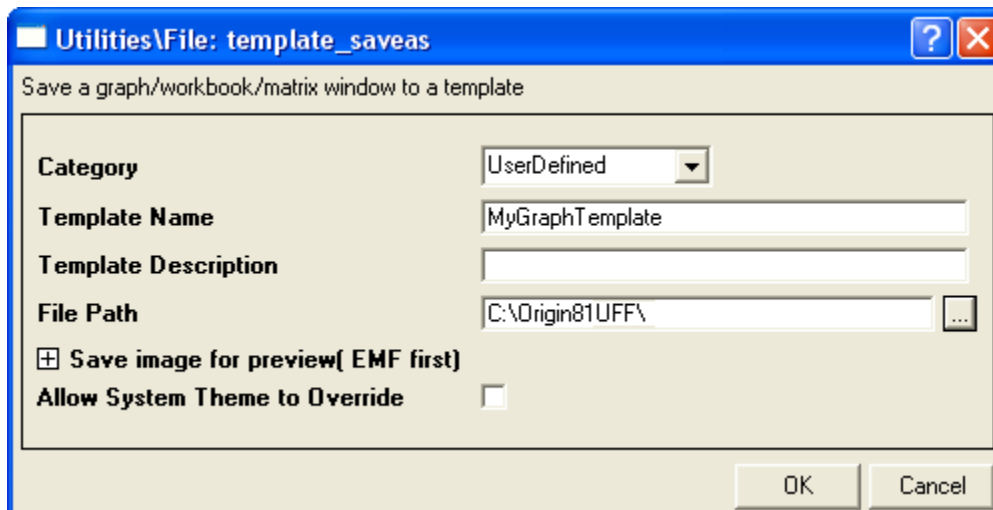
graph should look like:



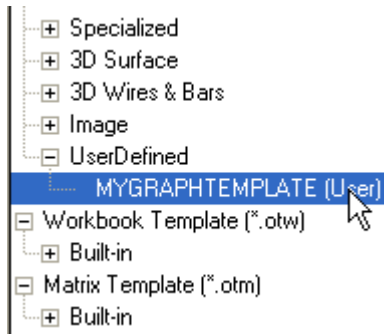
4.1.4 Save and Reuse a Template

In this section, we will show you how to save the graph above as a template and reuse it.

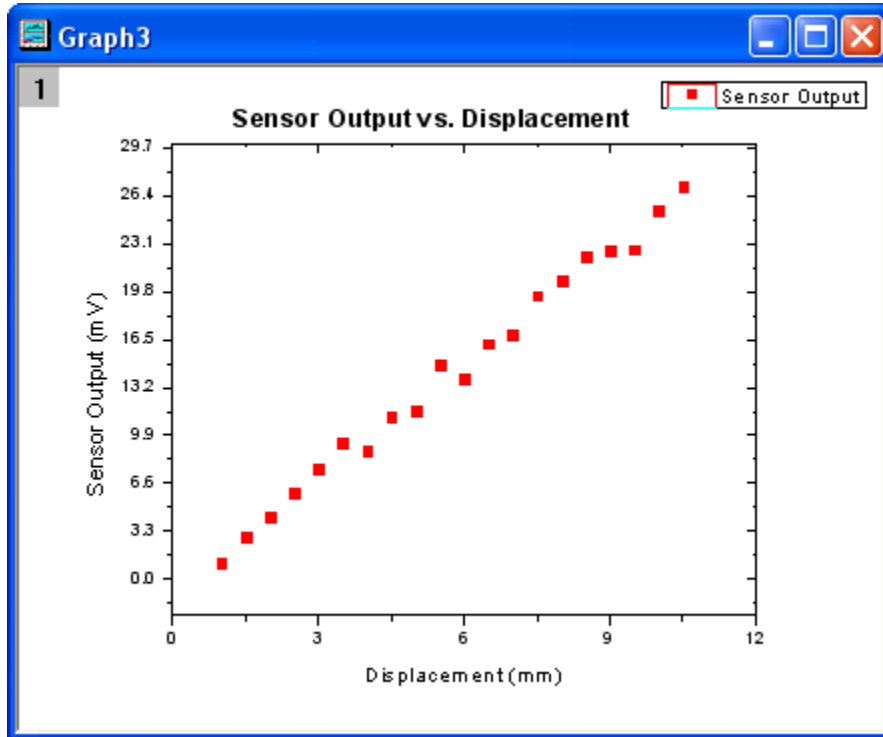
1. In the menu, select **File: Save Template As** (alternatively, right-click on the graph window title and choose **Save Template As** from the context menu) and rename the template as **MyGraphTemplate**. Then click the **OK** button to save it.



2. Create a new workbook and import the data file `\Samples\Curve Fitting\Sensor2.dat`, by using **File: Import: Single ASCII**. Highlight column B and select **Plot: Template Library**. Then select **MyGraphTemplate** and click the **Plot** button to create a graph.



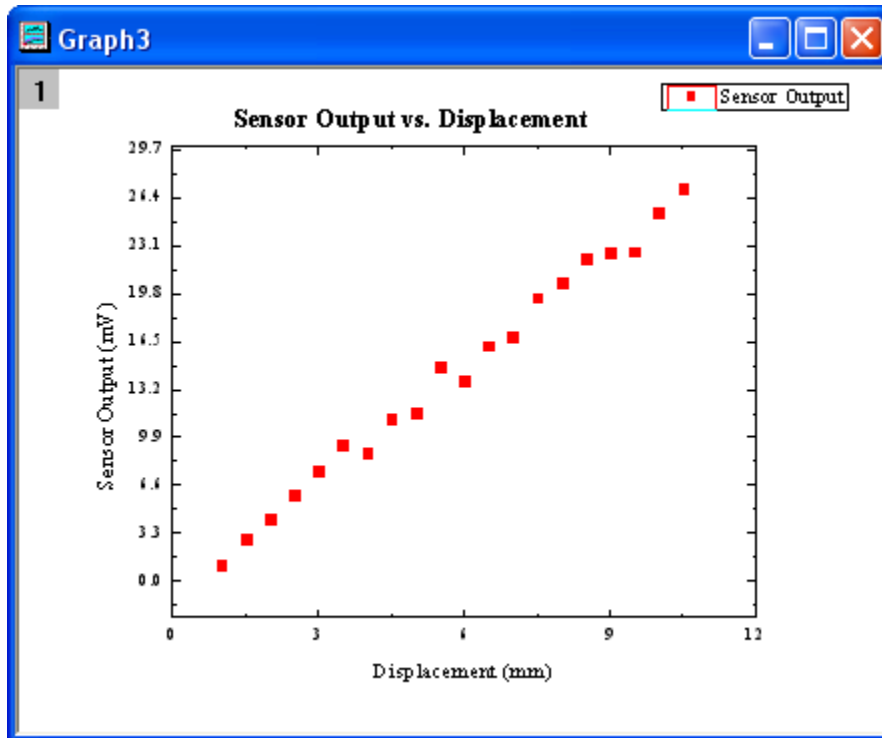
The graph should look like:



4.1.5 Changing graph using theme

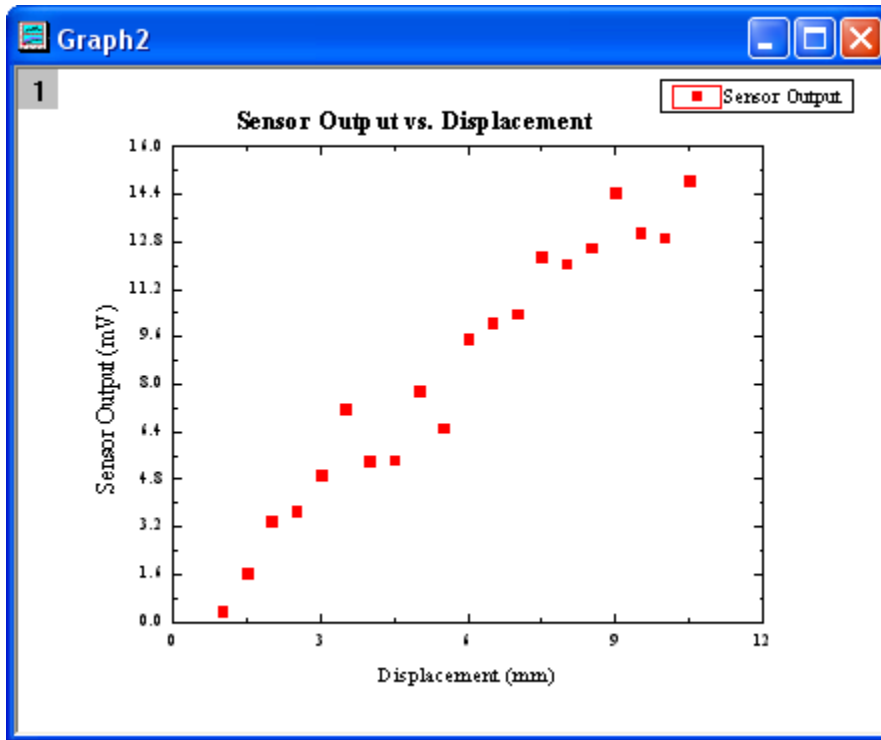
Origin stores the properties of a graph in a theme file. In this section we will show you how to customize a graph by using a theme.

1. With **Graph3** active, select **Tools: Theme Organizer** to open a dialog. Apply the themes **Ticks All In** and **Times New Roman Font**. Then click the **Close** button to close the dialog. The graph should look like:



2. Then we will copy the formatting of the current graph and paste it to **Graph2**. Right-click to the right of the layer, anywhere on an empty white space, or even on the gray area, and select **Copy Format: All Style Formats**. Then activate **Graph2**, right-click in a similar spot and select **Paste**

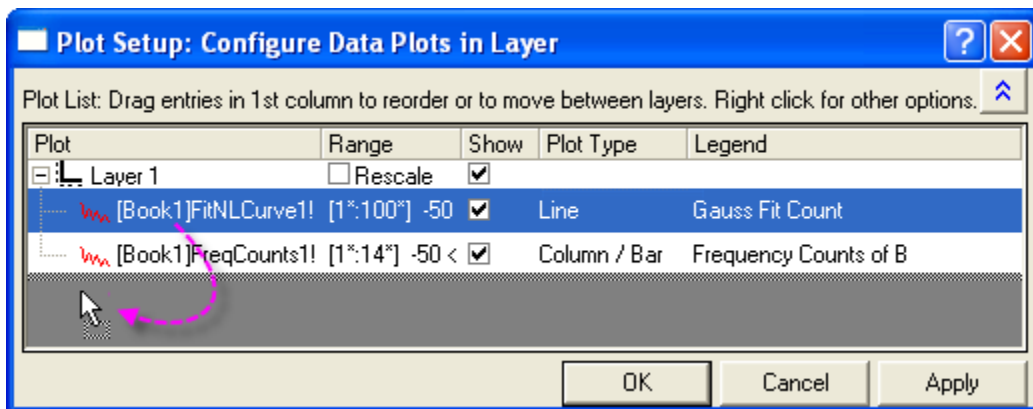
Format. Then **Graph2** should look like the image below:



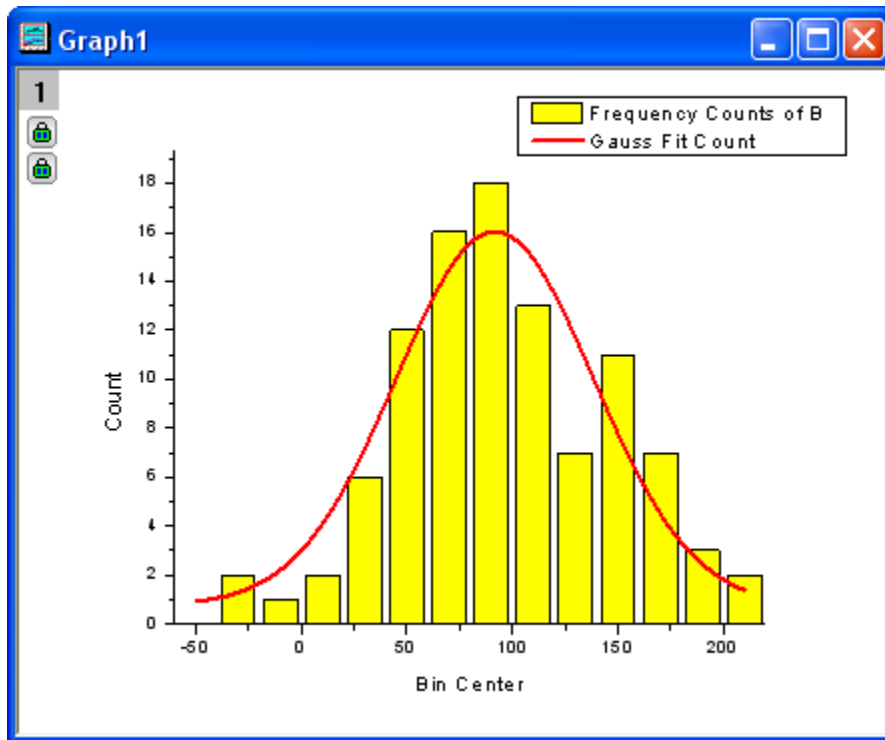
4.1.6 Plotting Order

In this section, we will show you how to change the plotting order by using the **Plot Setup** dialog.

1. Go to the **Plotting Order** folder and activate **Graph 1**. Select **Graph: Plot Setup** from the main menu to open the **Plot Setup** dialog. (Alternatively, double-click on the layer 1 icon in the upper left-hand corner of the graph, which also will open the **Plot Setup** dialog.)
2. In the **Plot List** panel, drag the line plot and drop it beneath Column/Bar Plot Type.




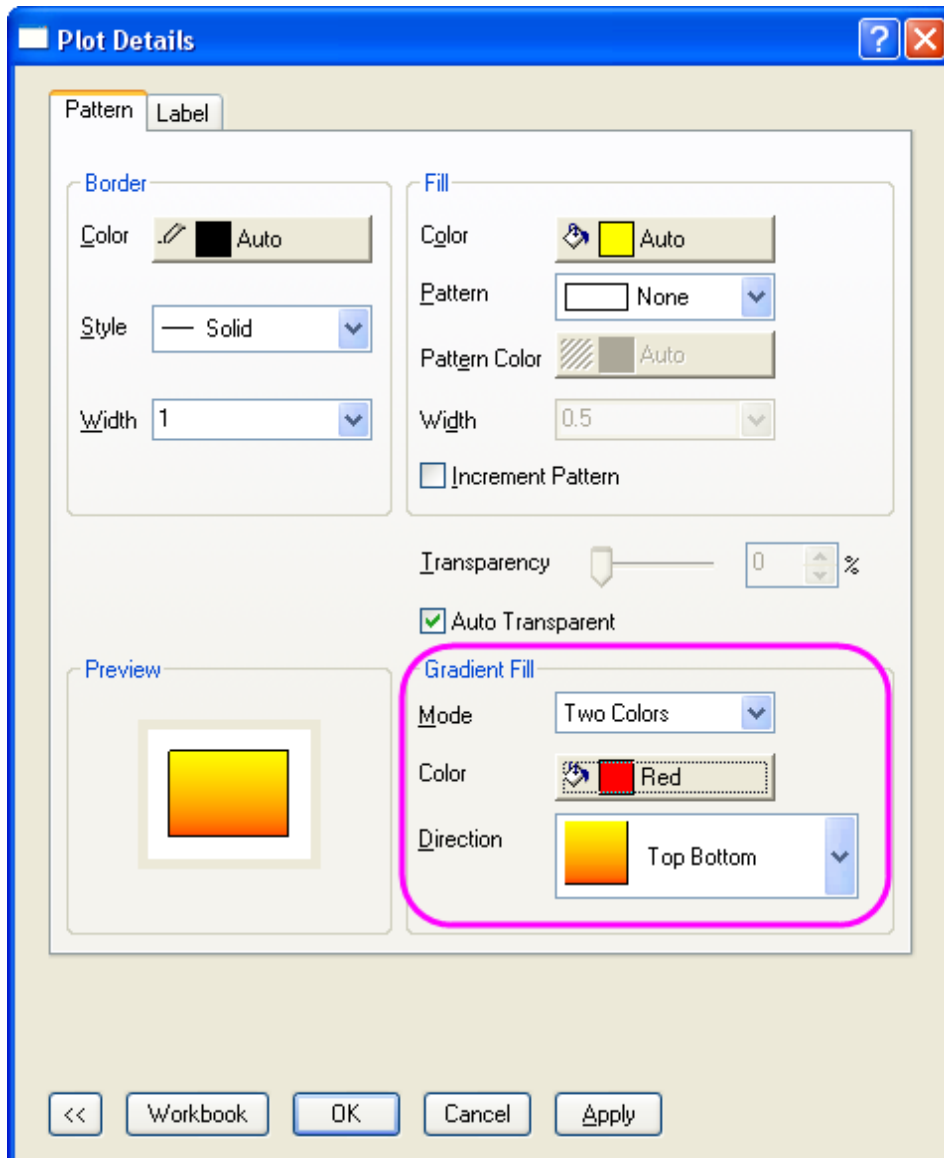
- Click the **OK** button and you will see that the red curve is now drawn on top. Notice the legend reflects the new plotting order as well.



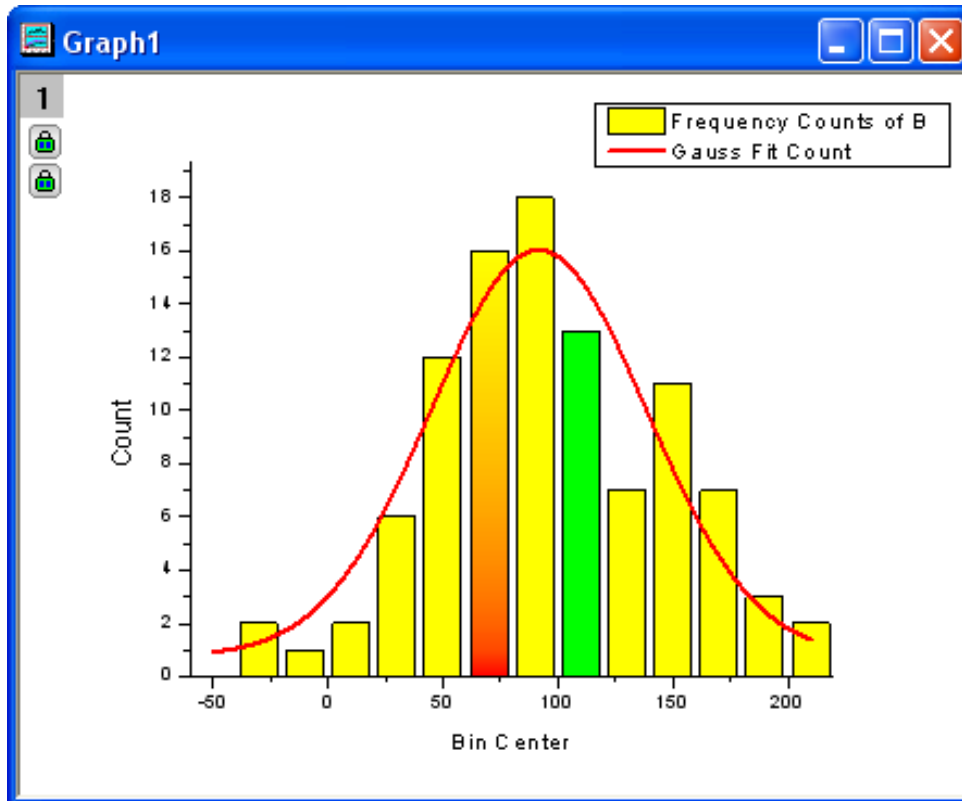
4.1.7 Customize points

In this section, we will show you how to customize a single data point.

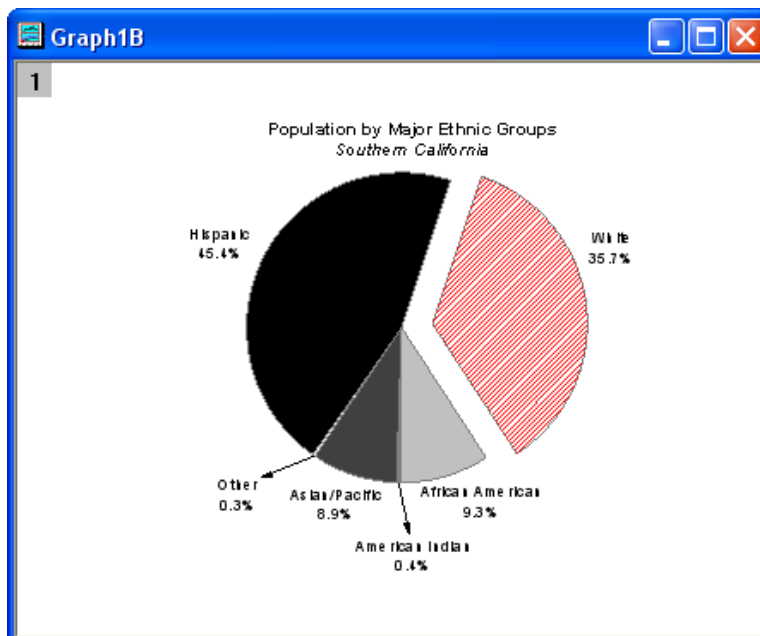
1. Continue viewing the contents of the **Plotting Order** folder and make sure **Graph 1** is active. Click to select one of the columns and all columns become selected. Click again on one of the columns to select just that one column. Then change the color to **green** by using the **Fill Color** button  on the **Style** toolbar.
2. You can also use the **Plot Details** dialog to customize a single column. Hold down the Ctrl key while double-clicking on one of the columns to open the **Plot Details** dialog. You'll be editing the properties of just that single data point. Notice the index number for that point in the Plot Details dialog. Set the **Gradient Fill** group as follows to change the column color from yellow to red, gradually.



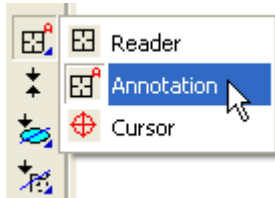
Then the graph should look like:



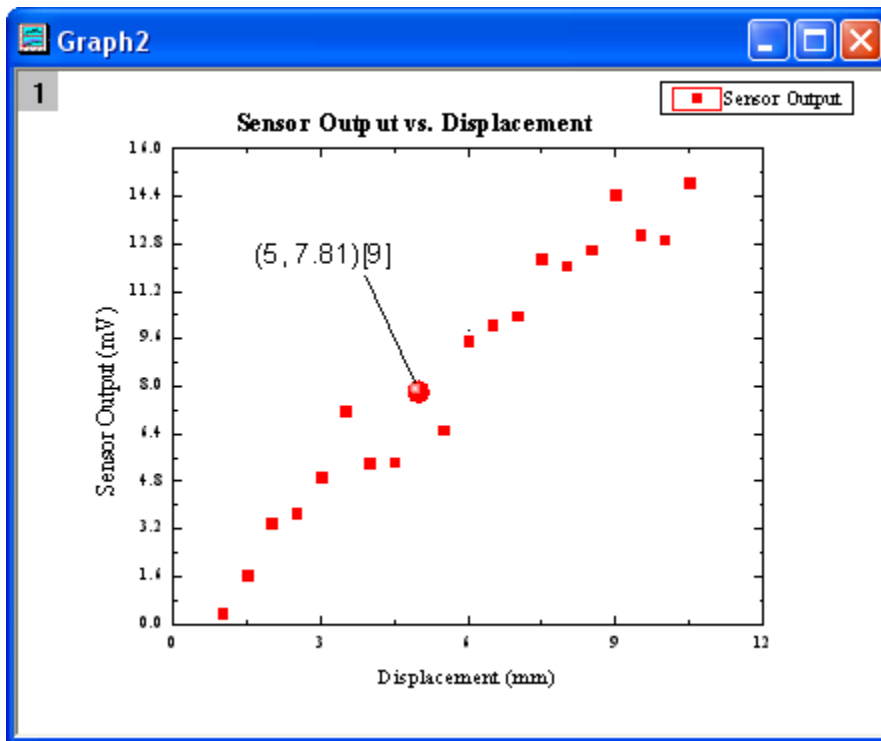
3. You can customize a slice of a pie graph in the same way. Go to the **Edit Single Data Point** folder and activate the pie graph. Hold down the Ctrl key and double-click on one of the slices to open the **Plot Details** dialog. Under the **Pattern** tab, set the **Fill Pattern** to **Dense**, from the drop-down list, and set the **Pattern Color** to **Red**. (Alternatively, you could have clicked once and then clicked again on a slice to select just that one point, and then used the Style toolbar to customize that point.) The graph should look like:



4. You can customize a single data point and add an annotation to it. Go back to the **Resize Graph and Customize Symbol** folder. Hold down the Ctrl key and double-click on one of the scatter points to open the **Plot Details** dialog. Under the **Symbol** tab, click the triangular **Preview** button to open the symbol gallery, and then select **Sphere** for the symbol. Increase the **Size** to **18**, and click the **Ok** button to close the dialog.
5. Now you can use the annotation tool to add the X and Y values that correspond to your customized data point. Select the **Annotation** button from the **Tools** toolbar.




Then move the annotation cursor to your customized data point and double-click on it. Origin will automatically add a text object. Hit the ESC key or click on the Pointer button to stop annotating. You can customize the text label by double-clicking on it to enter Edit mode. You can also change its position by clicking and dragging - a line will automatically be drawn connecting the point and the label.

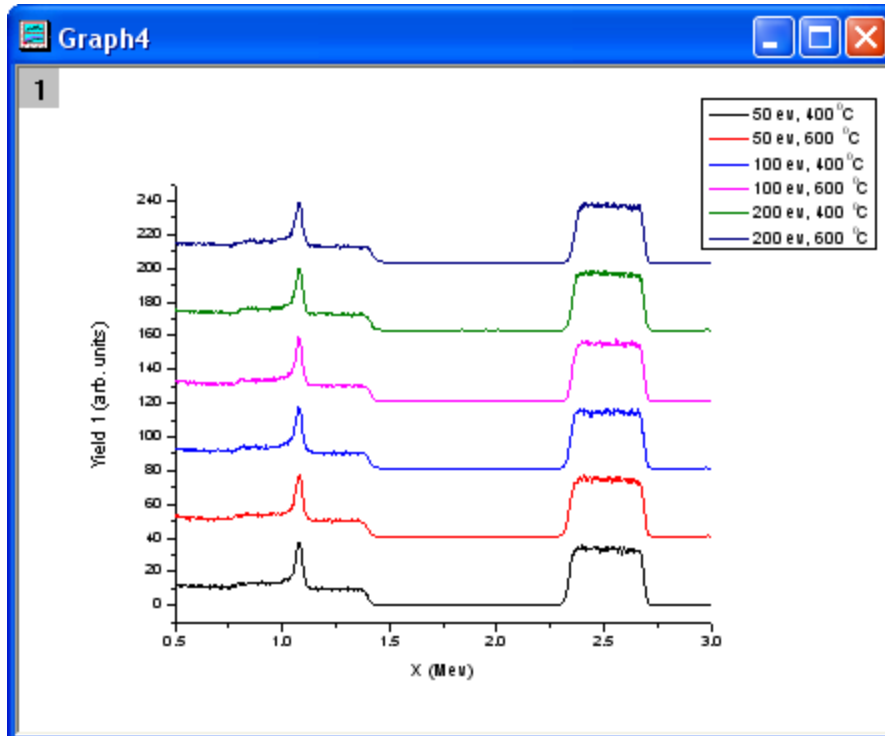




4.1.8 Group plots

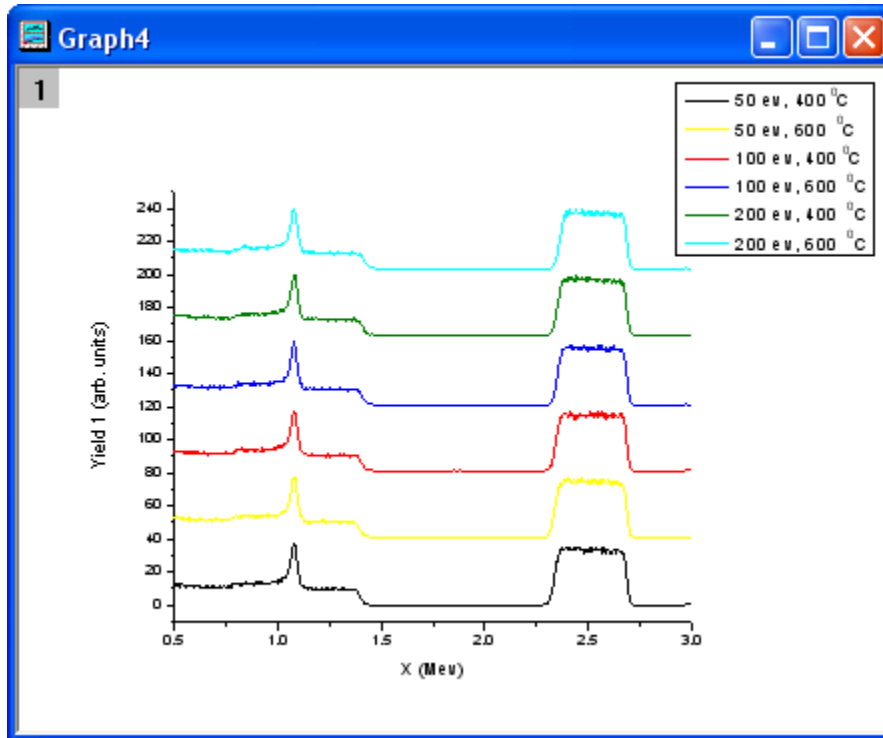
In this section, we will show you how to customize a grouped data plot.


1. Select the **Grouped Data** folder. **Book3** will be active. Highlight the entire worksheet and click the line button  on the **2D Graphs** toolbar to create a line graph.

2. Double-click on the X-axis to open the **Axis** dialog. Input **0.5** into the **From** edit box and **3.0** into the **To** edit box. Select **Manual** from the **Rescale** drop-down list (you will need to scroll up in the list). This prevents the **From** and **To** values from changing while rescaling. Click **OK** to apply these settings.
3. Double-click somewhere inside the layer, on a blank place above the line plots, to open the **Plot Details** dialog. On the **Stack** tab, select **Auto** in the **Offset** group. Click **OK** to close the dialog.
4. Select **Graph: Rescale to Show All**. The Y scale of the graph automatically rescales, while the X scale doesn't change because the option was set to manual. Resize the layer and move the legend as you see fit.



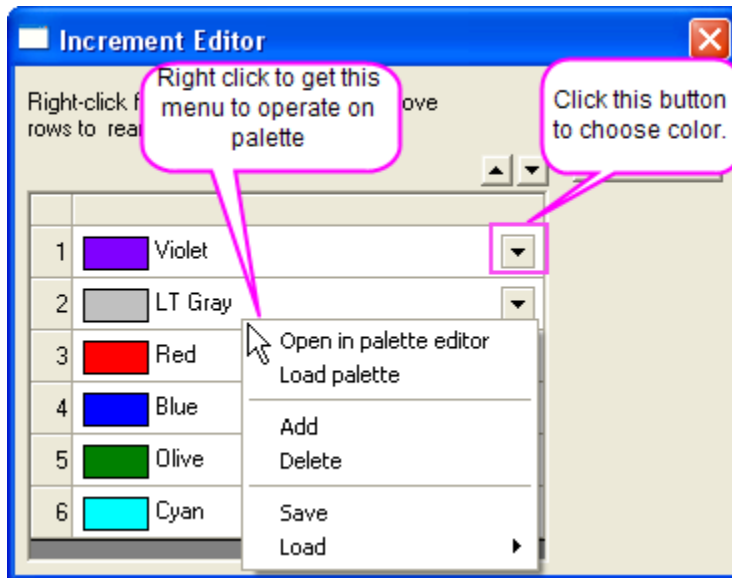
- Click on one of the data plots to select the entire group, and then use the **Line/Border Color** button  on the **Style** toolbar to change the colors. In **Incr. List**, you can select the second-to-last increment list . The graph should look like:



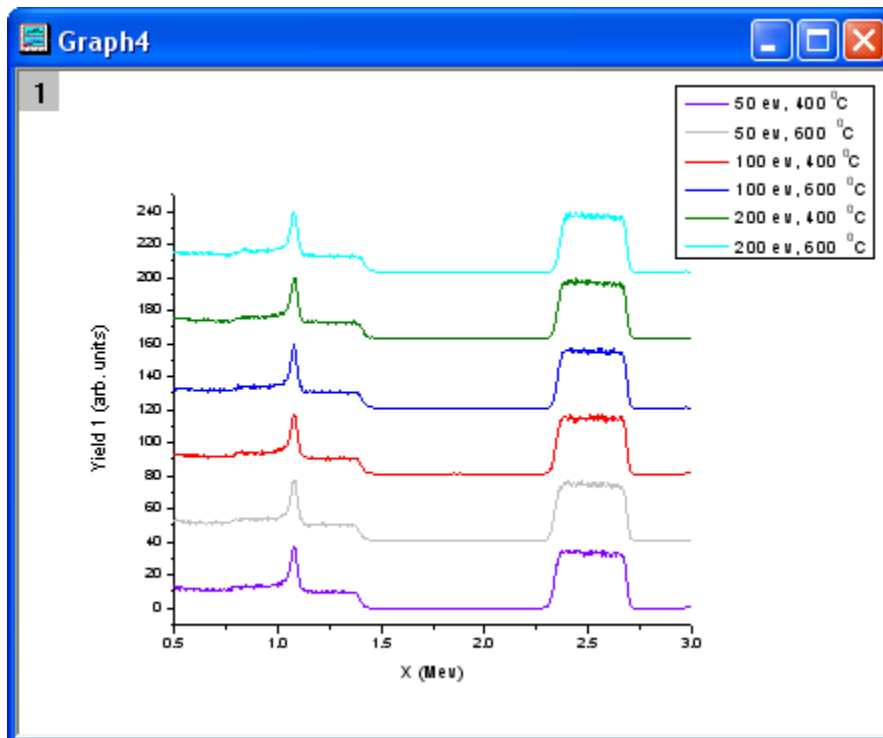
- Though these data plots have been grouped, you can also customize each of them individually by clicking on the data plot twice. For example, you can click the yellow data plot twice (click once, pause and click again) and then click the **Line/Border Color** button  on the **Style** toolbar to change the color to another color, such as **Olive**.
- You can also customize the graph by double-clicking on one of the data plots to open the **Plot Details** dialog. Under the **Group** tab, you can click the browser button



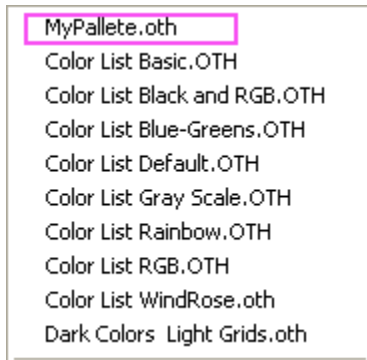
to open the **Increment Editor** dialog. In this dialog, customize the graph as shown in the following image. Please note that you can also drag the index of a row and move it to change the order of the color list.



The graph should look like:

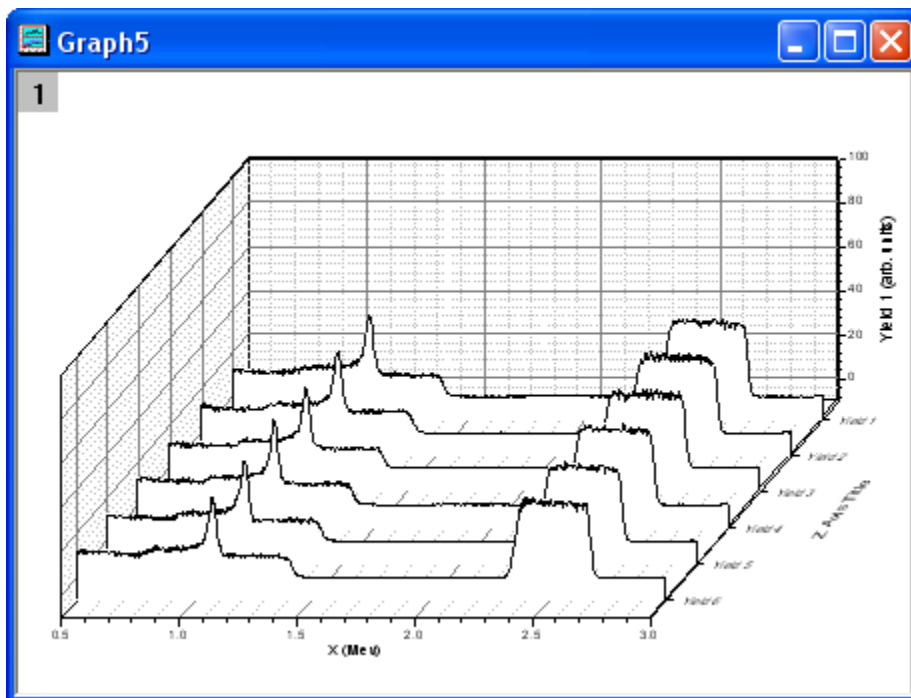




- Right-click on the inner position of this dialog and select **Save** to save it as **MyPalette** for future use. Then you can right-click to get the context menu and select the **Load** option. You will find that **MyPalette** appears as the first entry in the fly-out menu, as the following image shows.



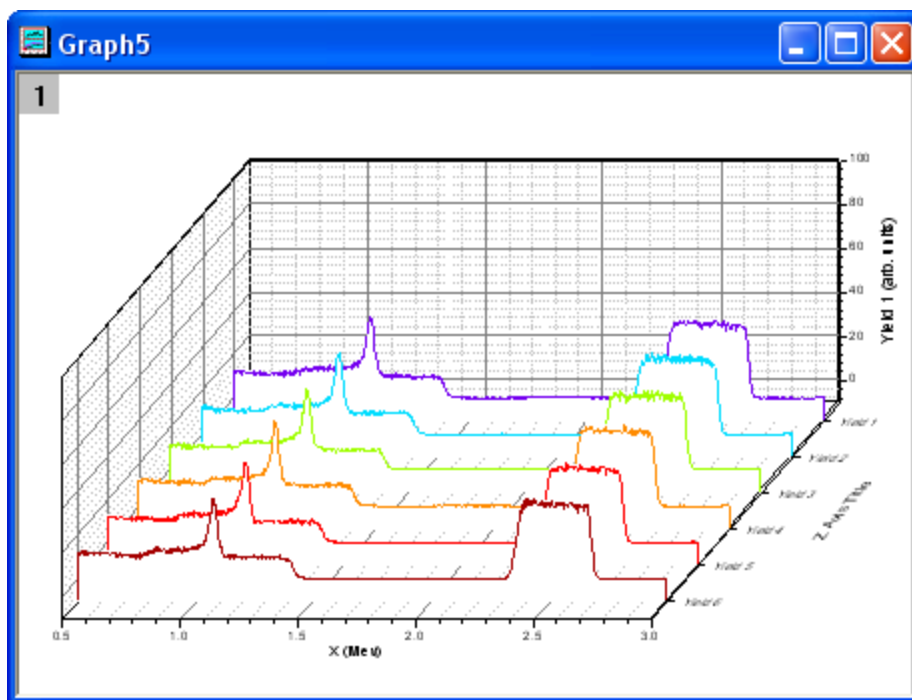
In the following section, we will show you how to use **Palettes** to set colors for a group of data plots.

- Activate **Book3** and highlight all the columns. Select **Plot: Multi-Curve: Waterfall** to create a graph. Double-click on the X-axis to open the **Axis** dialog, and set **From** as **0.5** and **To** as **3**. Select **Vertical** from the **Selection** list, and set the **From**, **To**, and **Increment** edit boxes to **10**, **100** and **20**, respectively.

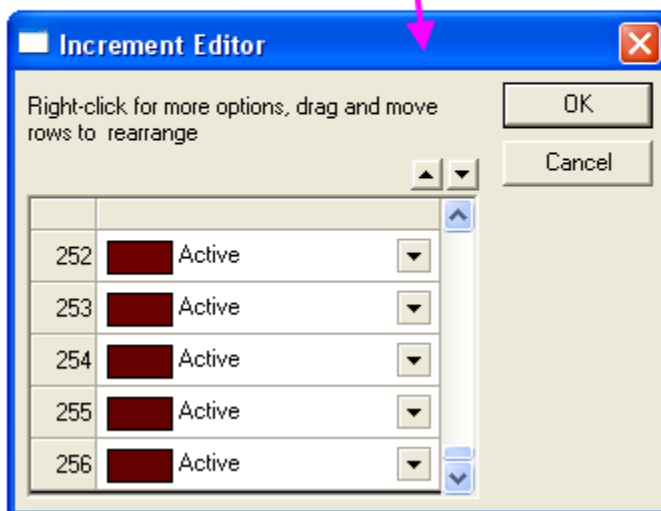
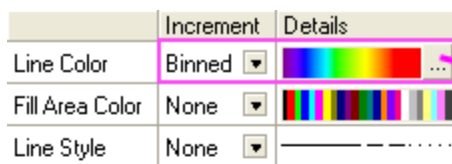


- Click on one of the data plots and then use the **Line/Border Color** button  on the **Style** toolbar to change the colors. You can select the **Rainbow** palette  under **Palettes**. The

graph should look like:






3. You can double-click on one of the plots to get the color list, as the following image shows:

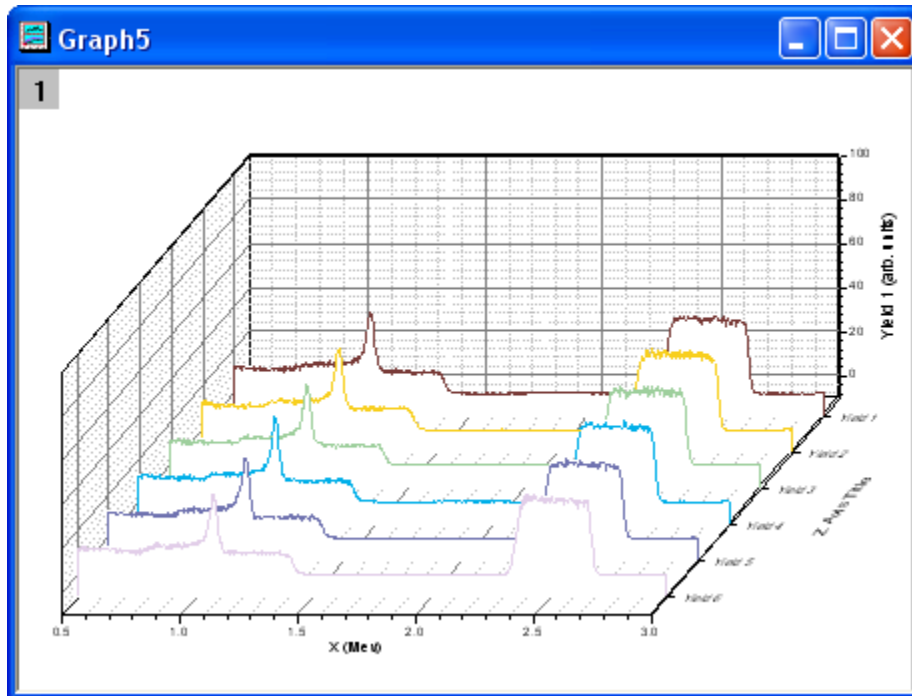


You can find that the **Increment** has been set to **Binned**, so colors are picked from the 256 available in the **Rainbow** palette.

4. We will change the palette to **Reef** by clicking on the **Browser** button to open the **Increment Editor** dialog. Then right-click on the inner position of this dialog and select **Load: Reef.PAL**. The color list in the **Plot Details** dialog should be:

	Increment	Details
Line Color	Binned	
Fill Area Color	None	
Line Style	None	

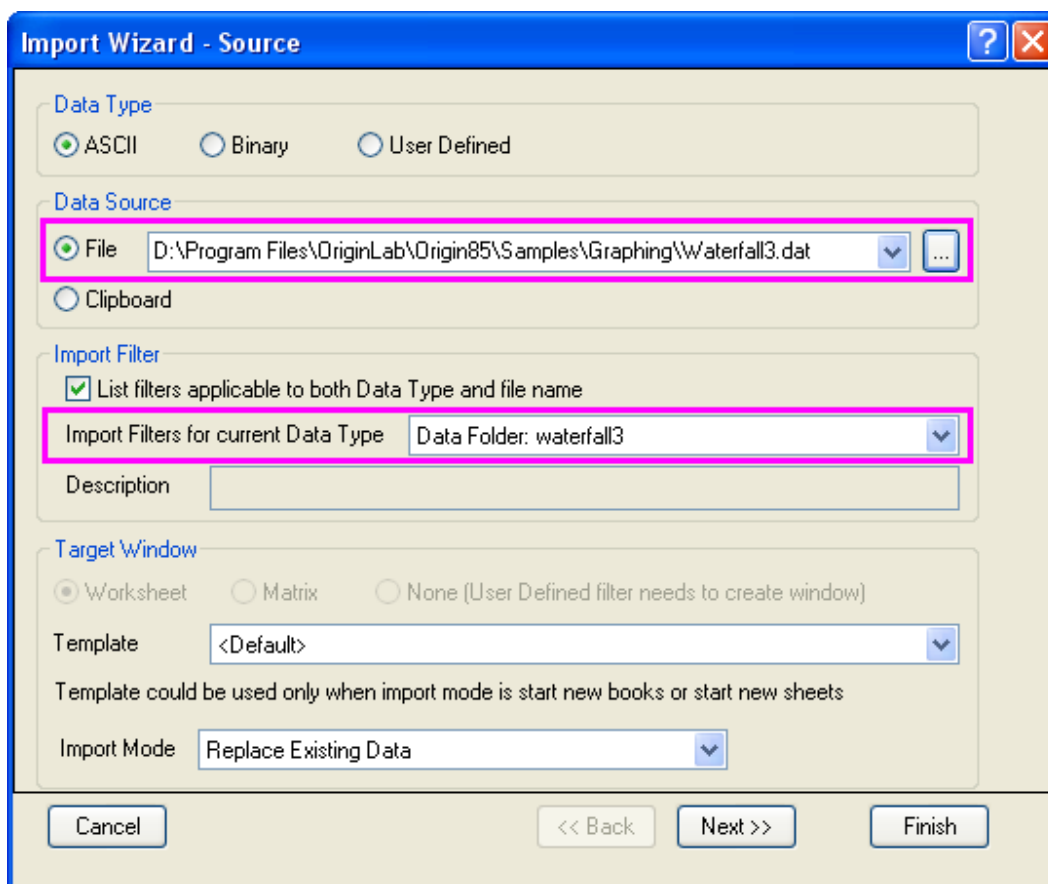
And the graph should look like:



4.1.9 Plot a Color-mapped Waterfall

In this section, we will show you how to create a color-mapped waterfall and use the desired column label row as the Z-axis.

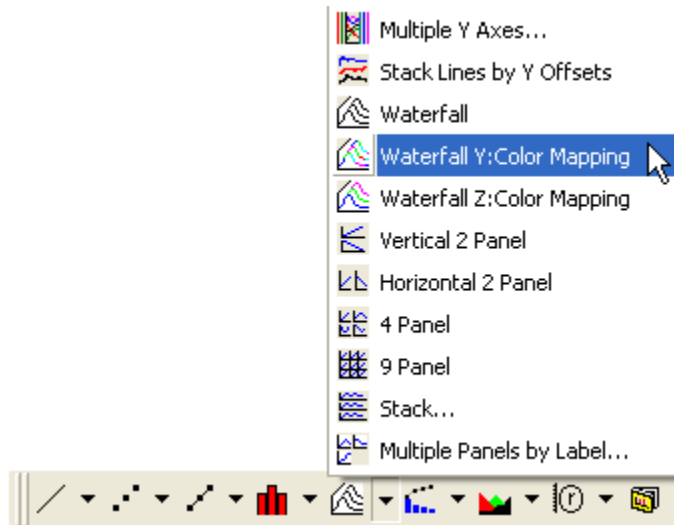
1. Click the **New Workbook** button  on the **Standard** toolbar to create a new workbook.
2. Click the **Import Wizard** button  on the **Standard** toolbar. This opens the **Import Wizard** dialog. Click the browser button to the right of **File** and select *Waterfall3.dat* from the */Samples/Graphing* folder. Make sure *Waterfall3.oif* is picked up automatically for **Import Filters for current Data Type**.



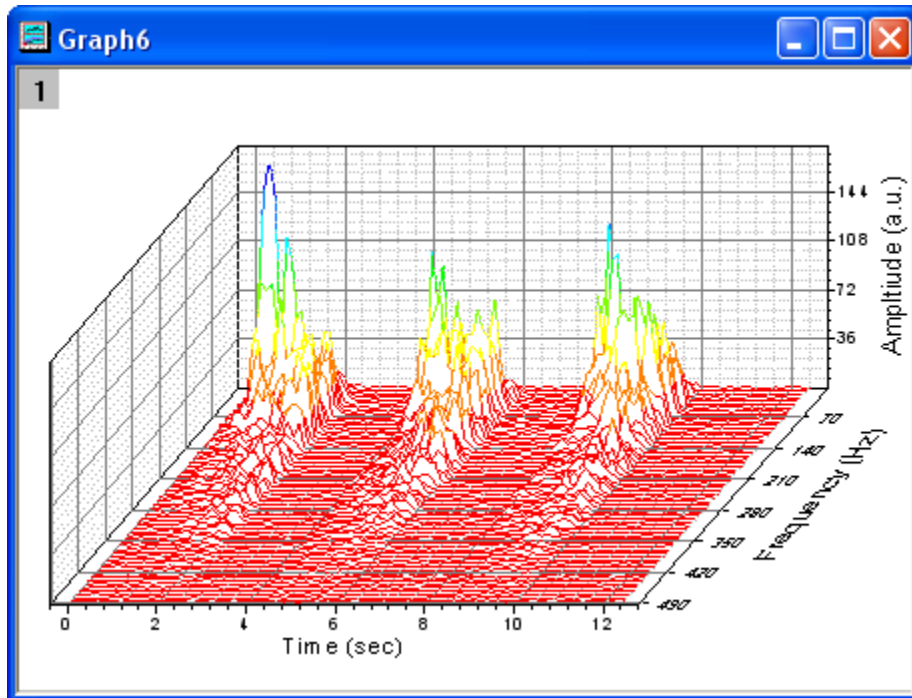
Click **Finish** to import the *Waterfall3.dat* file.

	A(Y)	B(Y)	C1(Y)	C2(Y)
Long Name	Time	Amplitude		
Units	sec	(a.u.)		
Comments				
Frequency (Hz)	--	3.91	11.72	19.53
1	0	0.766	0.697	0.406
2	0.012	0.413	0.097	0.03
3	0.025	0.14	-0.34	-0.26
4	0.037	-0.059	-0.628	-0.465
5	0.05	-0.191	-0.784	-0.588
6	0.062	-0.264	-0.823	-0.632
7	0.075	-0.284	-0.76	-0.6
8	0.087	-0.257	-0.61	-0.493
9	0.099	-0.191	-0.39	-0.315

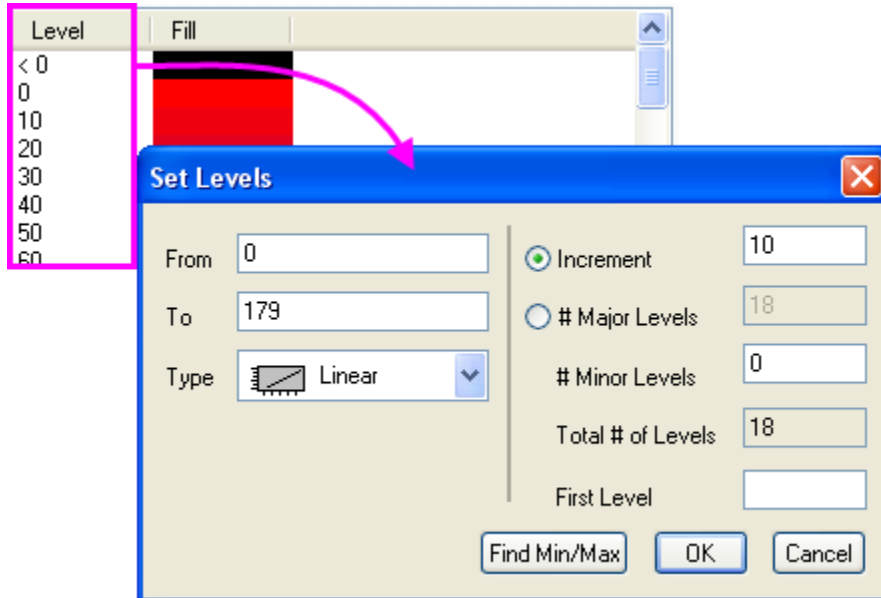
- To plot a waterfall color-mapped using each line's Y value, highlight the whole worksheet and select **Waterfall Y: Color Mapping** from the **2D Graphs** toolbar (Alternatively, Select **Plot: Multi-Curve: Waterfall Y:Color Mapping** from the main menu).



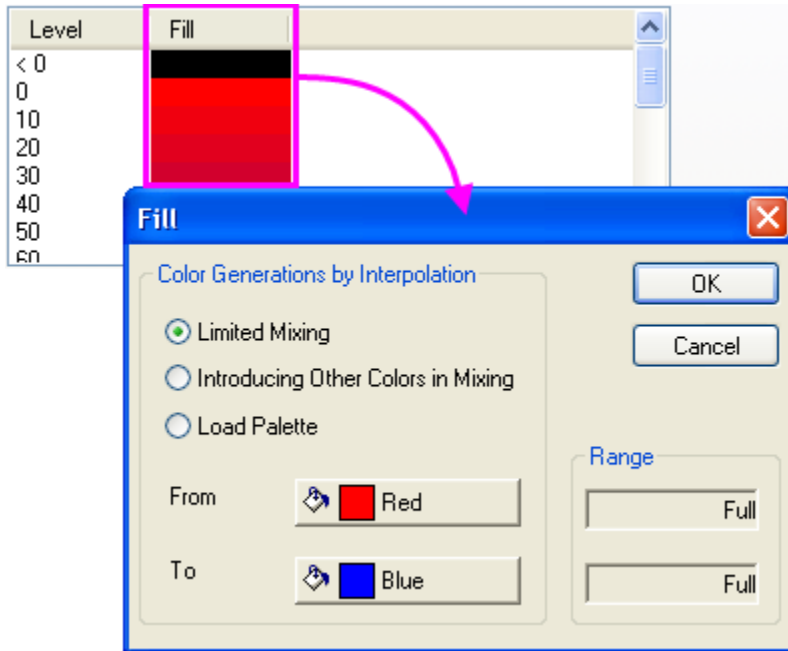
The graph should look like the picture below:



4. Double-click on the waterfall plot to open the **Plot Details** dialog. Go to the **Color Map** tab. On this tab, you can:
 - o Click on the **Level** column header to change the color levels

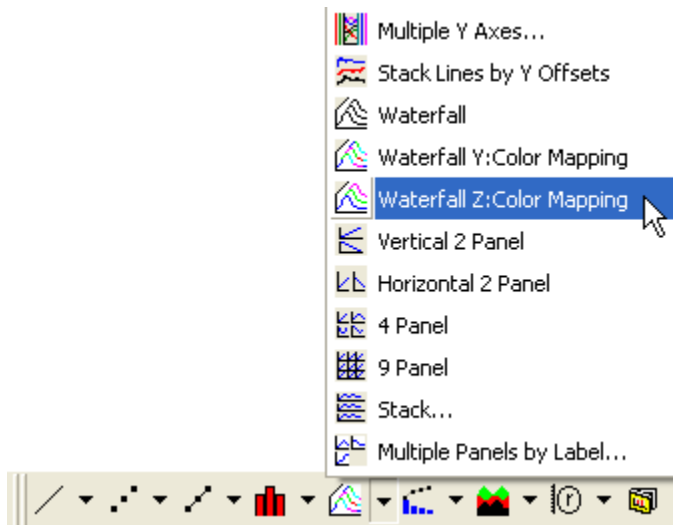


- o Click on the **Fill** column header to load a color palette or change the filled color list:

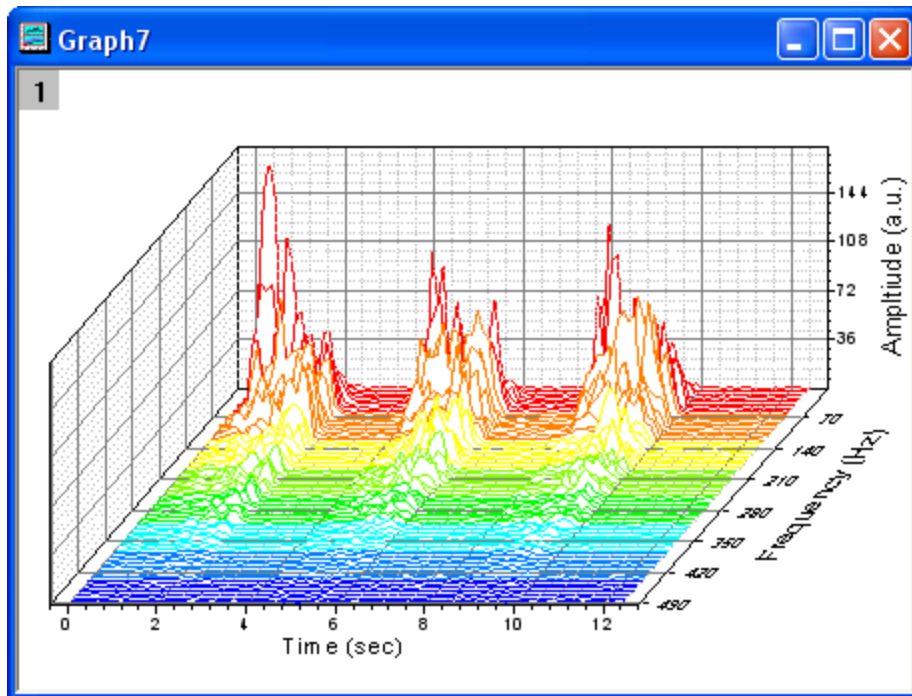


Please note that you can also click inside one single cell to change that level individually.

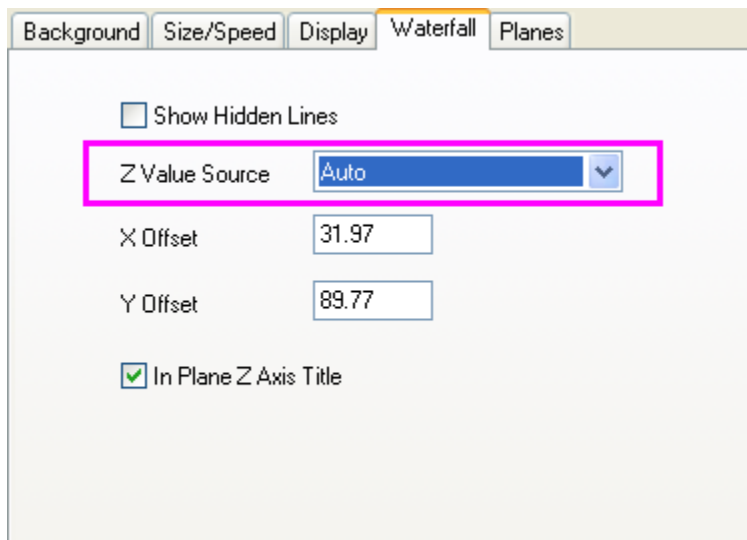
5. To plot a waterfall color-mapped with Z values, highlight the **waterfall** worksheet in **Book 2** and select **Waterfall Z: Color Mapping** from the **2D Graphs** toolbar (Alternatively, Select **Plot: Multi-Curve: Waterfall Z:Color Mapping** from the main menu).



The graph should look like the picture below:

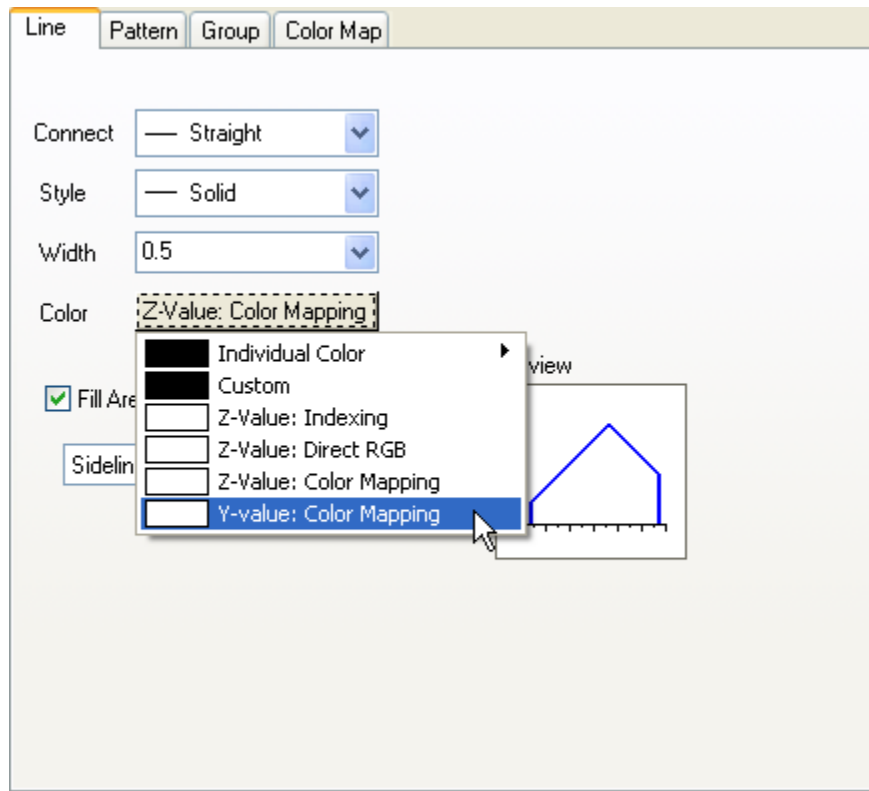


6. Note that the user-defined parameter "Frequency (Hz)" is used as the Z axis automatically. To use another column label as the Z axis, Short Name for example, double-click inside the layer and away from the waterfall lines to open the **Plot Details** dialog at the **Layer** level. Go to the **Waterfall** tab, and select **Auto** from the **Z Value Source** drop-down list.



7. To switch between Y and Z color mapping, select the first plot on the left panel of the **Plot Details** dialog and activate the **Line** tab. Select **Y-value: Color Mapping** from the **Color** drop-down list, to switch to Y color mapping. You can also select **Z-Value: Indexing** or **Z-Value:**

Direct RGB from this drop-down list.



4.2 Themes

4.2.1 Copy and Apply a Plot Format to Another Plot

Summary

It is possible to copy and paste formatting from one plot to another, so there is no need to spend time recreating identical customizations such as size and color of symbols and lines.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to:

- Copy a plot format (color, size, etc. of the symbol or the line), and apply it to other plot.

Steps

1. Click the **New Project** button on the Standard toolbar to begin with a new project.
2. Select **File: Import: Single ASCII** menu, and import **exponential_decay.dat** in the **\Samples\Curve Fitting** subfolder in your Origin program directory.
3. Highlight column B, C and D, and select **Plot: Line+Symbol: Line+Symbol" menu to plot these three datasets.**
4. Double-click on the plot to show the **Plot Details** dialog box.
5. Choose **Group** tab in the dialog, and select **Independent** for the **Edit Mode** -- this makes it easier to customize individual plots.
6. Make sure that the top data plot (Time(X) Decay 1(Y)) is selected in the left panel of the **Plot Details** dialog. If not, select this plot branch in the left panel.
7. Select the **Symbol** tab, and change the Size to "5". (You can also change the shape or the color to others of your choice.)
8. Select the **Line** tab, and change the width to "0.2". (You can also change the style or the color to others of your choice.) Click OK. You will see that the **Decay 1** plot has been customized.
9. Click on **Decay 1** plots to select. Right-click on it and select **Copy format: ALL**. This will copy the plot format of Decay 1 to the clipboard.
10. On the graph, click on the **Decay 2** data plot to select it, right-click and select **Paste Format**. You will see the plot format of Decay 1 copied to Decay 2.

4.3 Color

OriginLab: Plotting Overlapping Data and Setting Transparency

4.3.1 Plotting Overlapping Data and Setting Transparency


Summary

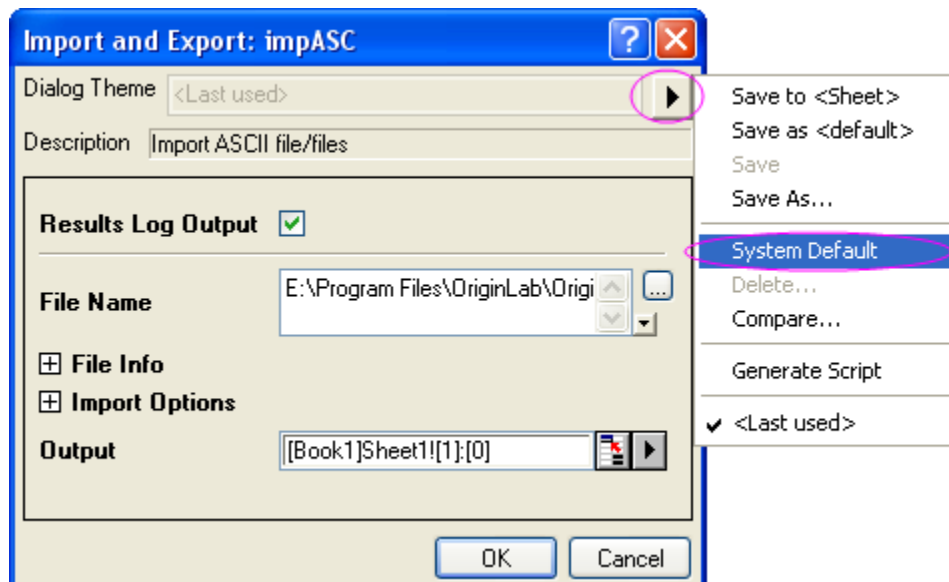
In this tutorial, we will show you how to create overlapping column plots, and then set transparency to make the overlapping parts visible.

This tutorial will show you how to:

- Customize the grouped column plots
- Set Transparency

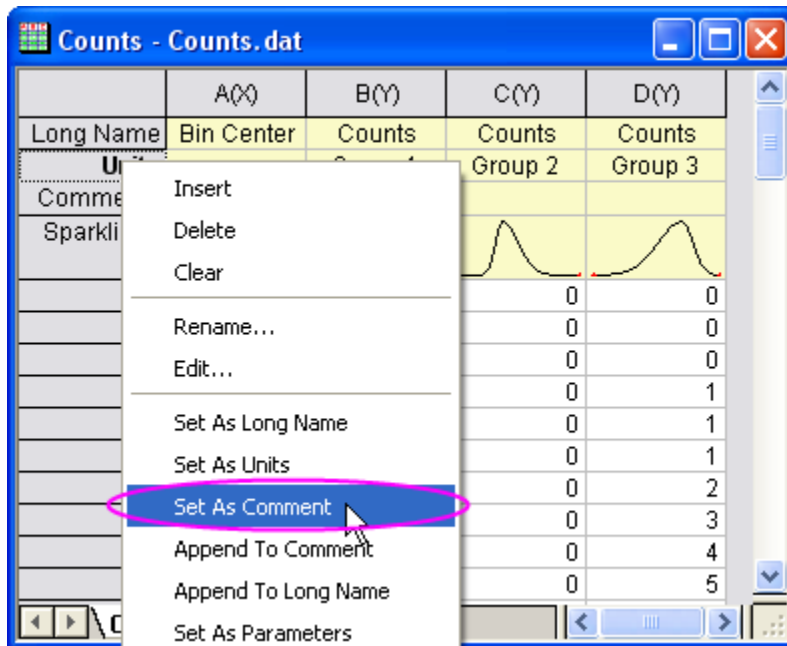
Plot Overlapping Data and Set Transparency

1. Start with a an empty worksheet and from the menu, select **File: Import: Single ASCII** or click the **Import Single ASCII** button  to open the file browser.
2. Choose the file `<Origin Installation Directory>\Samples\Graphing\Counts.dat`, check the **Show Options Dialog** check box, and click the **Open** button to bring up the **impASC** dialog.
3. Click the triangle button in the upper right corner, and then select **System Default** from the context menu to apply the default settings to this dialog.

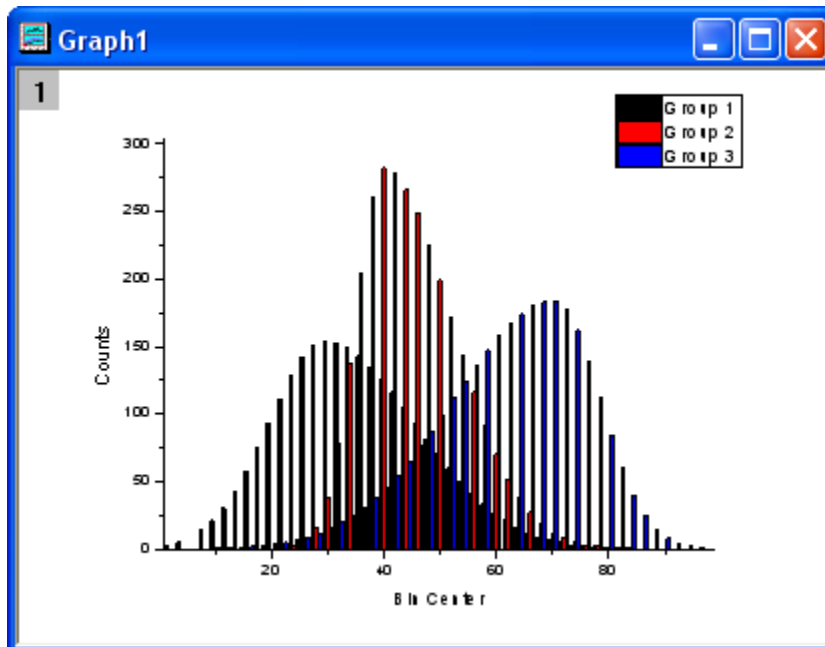


4. Click the **OK** button to import the data to the worksheet.

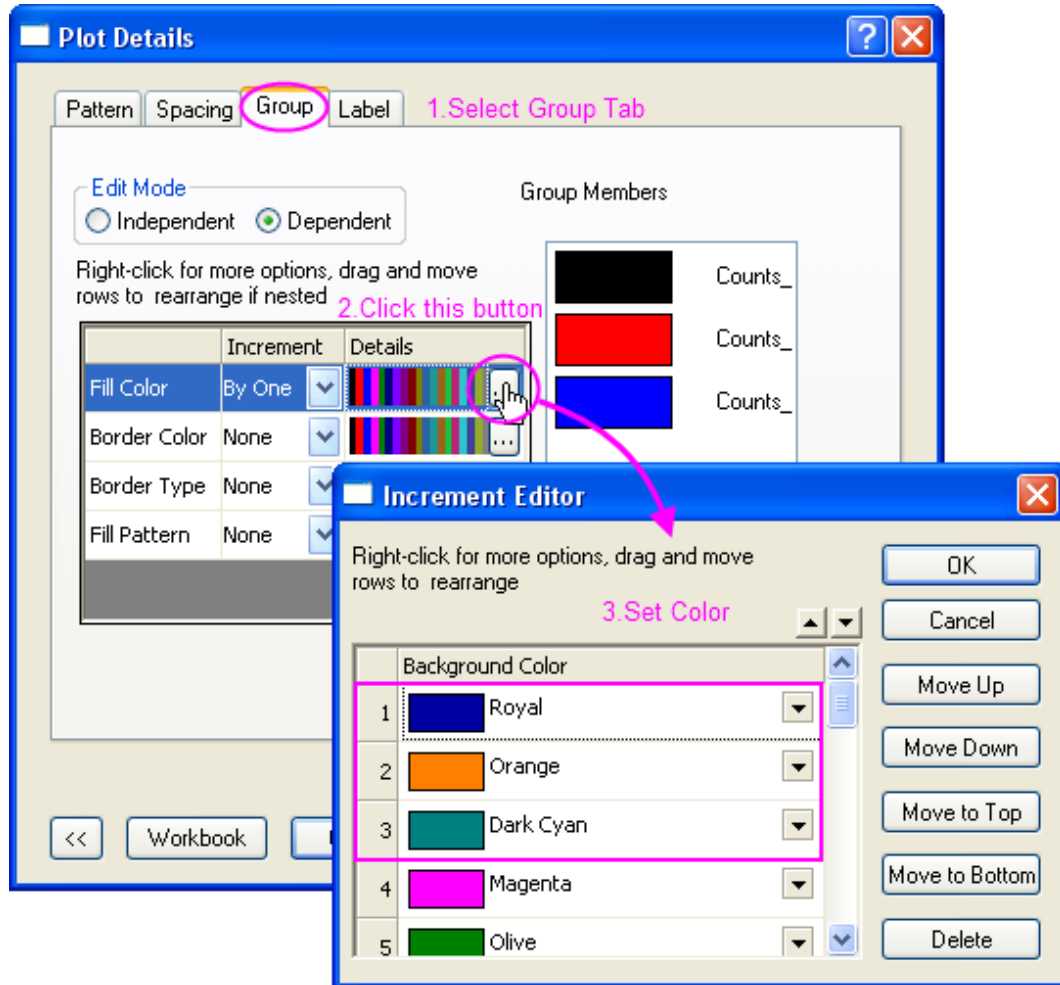
- Right-click on the row header of **Units**, and then select **Set As Comment** from the pop-up menu to set the contents to be comments instead of units.



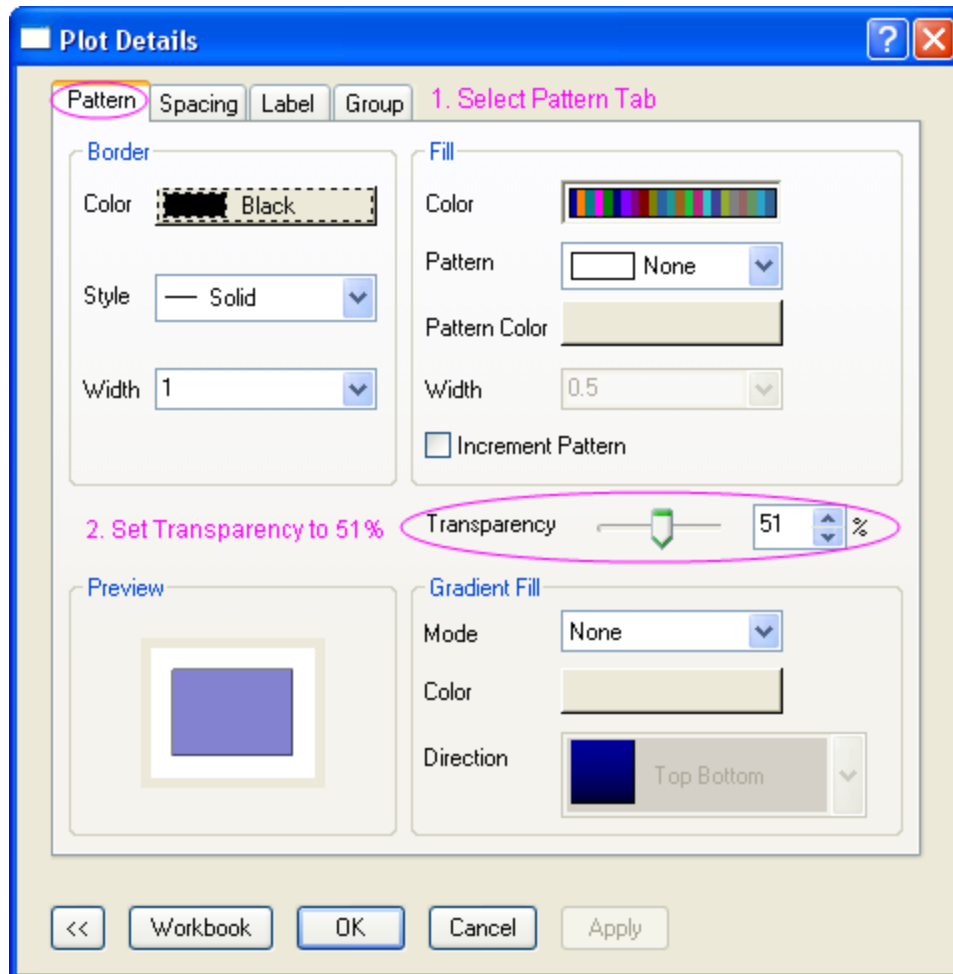
- Highlight all columns in the worksheet, and then from the menu, select **Plot: Column/Bar/Pie: Column** to create a column graph.



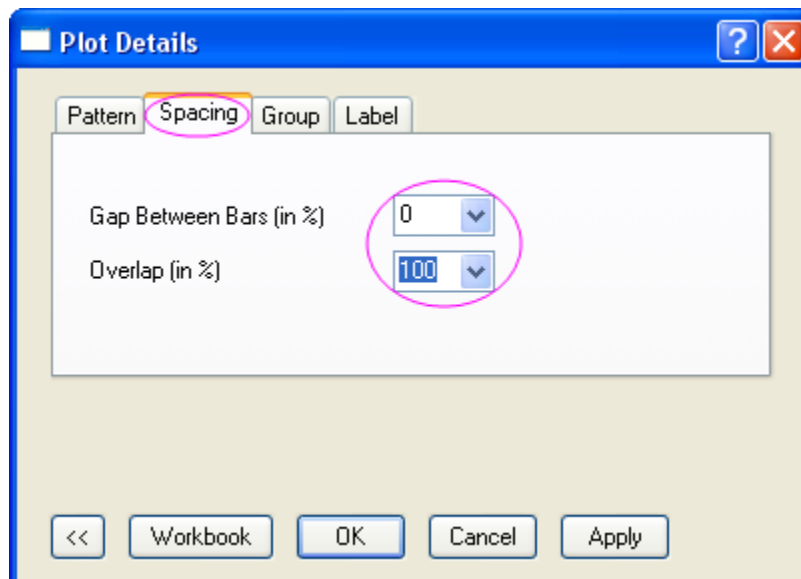
- Double-click on the plot to open the **Plot Details** dialog. In the dialog, choose the **Group** tab. Click on the color bar under **Details** and click in the **Fill Color** row. Then click on the button that appears to open the **Increment Editor** dialog. In this dialog, you can set the first three colors to be some specific colors such as Royal Blue, Orange, and Dark Cyan. Click OK to close the dialog.



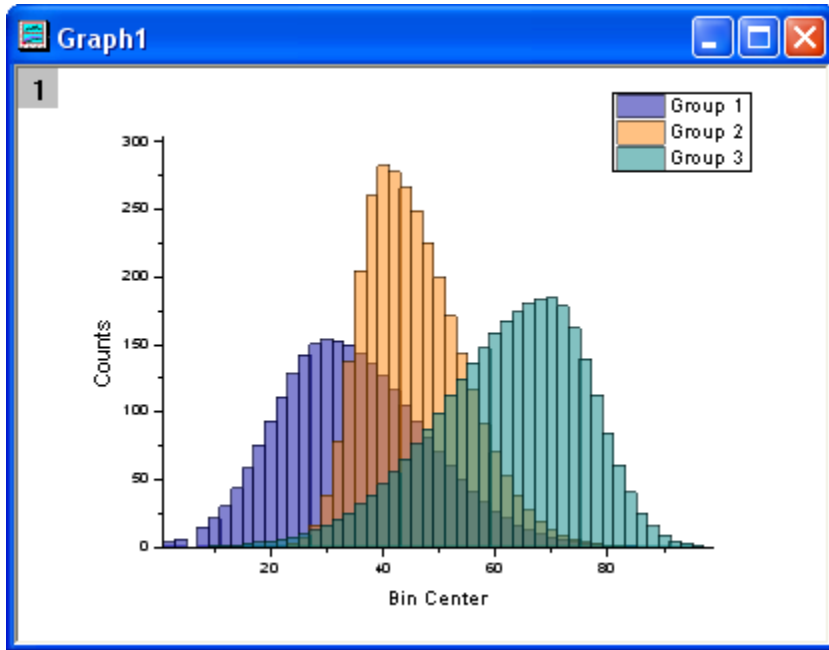
8. Select the **Pattern** tab in Plot Details, and set the **Transparency** control to 50%.



9. Select the **Spacing** tab, and set **Gap Between Bars** to zero and **Overlap** to 100.



10. Click the **OK** button to close the Plot Details dialog. The resulting graph should look like the image below:



4.4 Layers

4.4.1 Adding and Arranging Layers

Summary


A typical graph page generally includes three elements: a set of X, Y (and Z) coordinate axes, one or more data plots, and associated text and graphic labels. Origin combines these three elements into a movable, resizable unit called a layer. While a page can contain as many as 121 layers, only one layer can be active at any one time.

What you will learn



- How to tell how many layers a graph has?
- How to tell what data is in what layer?
- How to swap layers

Steps

Importing data

1. Click the **Import Single ASCII** button, , on the Standard toolbar. The ASCII dialog opens.
2. In the Origin folder, browse to the Samples folder and then the Graphing folder. Select Wind.dat from the list of files.
3. Click **Open**. The data file imports into the worksheet.

Plotting the data

1. Highlight the Speed and Power columns.
2. Click the **Line** button, , on the **2D Graphs** toolbar. A line plot is created. It appears that this data would be better plotted on a double-Y graph, a graph with two controlling Y axes.
3. Click the X to **Close** this window. You will be asked if you want to hide or delete the window. Click the **Hide** button. (If you Delete, you will not be able to Undo, and you will need to re-create the graph. Hiding closes the window from view, but you can later make it visible using the Project Explorer window.)
4. The Speed and Power columns should still be highlighted. Click the **Double-Y Axis** button, , on the 2D Graphs toolbar. This new graph contains 2 layers.


How to tell what data is plotted in a layer?

One way is with the legend:

1. Double-click on Graph1 from the Project Explorer window. The graph opens and becomes the active child window.
2. Select **Format:Page** and go to the **Legends/Titles** tab.

3. Set the **Auto Legend Translation Mode** to *Data Range*.
4. Click **OK**.

A second way is with the status bar:

1. Double-click on Graph2 from the Project Explorer window. Graph2 is now in front and becomes the active child window.
2. Click the Layer 1 icon,  to make layer 1 active.
3. Look in the lower right of Origin's status bar and you'll see [WIND]WIND!Col(Speed)[1:12].
4. Repeat for Layer 2 and you'll see [WIND]WIND!Col(Power)[1:12].

A third way is from the menu:

1. Make Graph2 active.
2. Right-click on the Layer 1 icon. At the bottom of the context menu, you see the data plot list. The one checked is the active data plot.
3. Right-click on Layer 2 to view the list of data that is plotted in it.
4. Note that you can also select the **Data** menu to view the data plot list.

A fourth way is with the **Plot Setup** dialog.

1. With Graph2 still active, double-click on the layer 1 icon.
2. In the opened Plot Setup dialog, expand the Plot List to see also the data in Layer 2. You can see the advantage here is that you can see data in all layers at once.
3. Uncheck the Show check box for the **Speed** data plot.
4. Click **OK**. This data plot is only hidden which is why the legend still indicates its presence in the layer.

The last way is from the **Plot Details** dialog.

1. Double-click on any of the line+symbol plots in Graph2.
2. Expand the tree so you see contents of both **Layer1** and **Layer2**.
3. Uncheck the **Speed** and **Power** data plots.
4. Click **OK**.

4.4.2 Merging and Arranging Graphs

Summary

The **Merge Graph Windows** dialog allows you to select which graphs you wish to combine, choosing from any graph in the project. It also has controls to specify how you want the individual graphs arranged on the new page.

The **Object Edit** toolbar allows you to quickly align and size multiple layers.

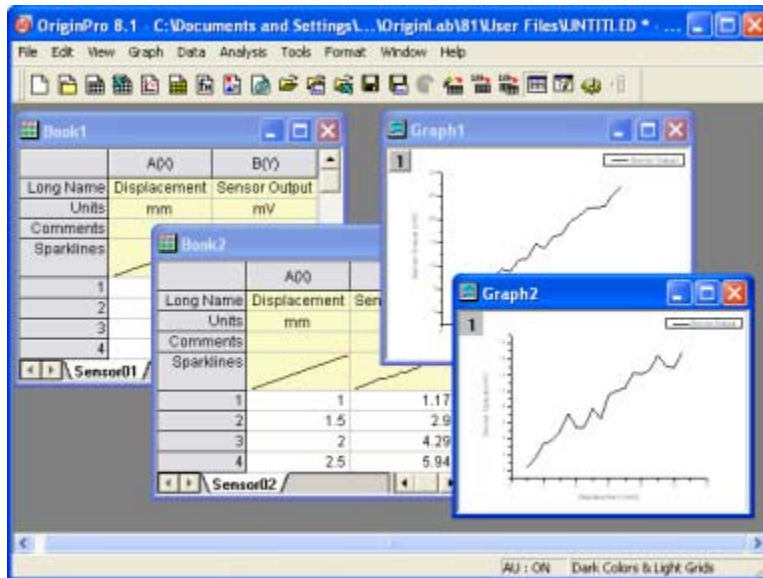
The **Layer Management** dialog lets you add, arrange and link layers on a single graph page.

This tutorial will show you how to:

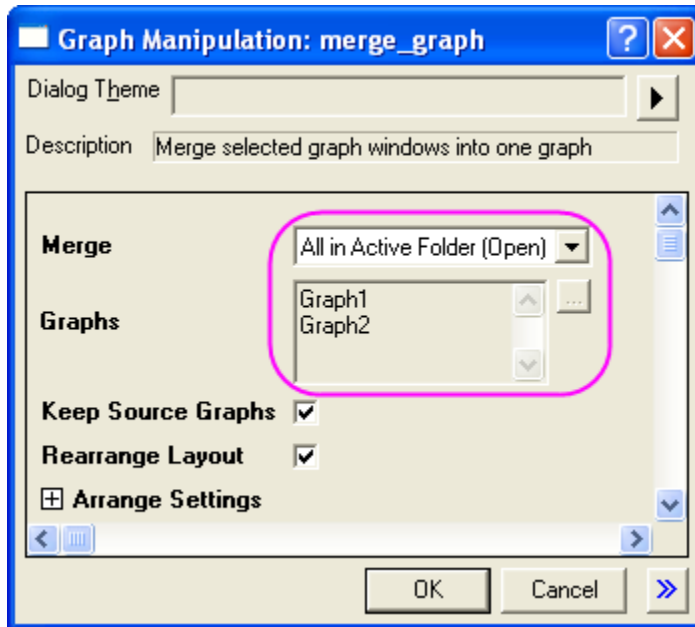
- Merge multiple graphs into one graph
- Resize and align layers quickly
- Add a second axis using a nonlinear relationship with the primary axis
- Use Layer Management for more complex layer positioning and linking

Creating a Merged Graph from Multiple Graphs

1. Import the two files **sensor01.dat** and **sensor02.dat** from **\Samples\Curve Fitting**, as separate sheets in separate books, using the file names as the sheet names. (Please read the Importing Data tutorial for how to import files.)
2. Select the Y column of one sheet and create a line plot. Repeat with the Y column of the other sheet and create a second line plot. You will have two separate graph windows at this point.



- Now, to merge the two graphs into one page, bring up the merge graphs dialog from **Graph: Merge Graph Windows**. In this dialog, the default setting for merging graphs is **All in Active Folder (Open)**, so the two graph windows are already listed in the **Graphs** box.



4. Specify the following settings to merge the two graphs: 1) Uncheck **Keep Source Graphs**. This will remove source graphs after merging. 2) Arrange the layers as one column and two rows. 3) Check **Show Axes Frame**. In this example, the two layers share the same X axis range, so we can hide the overlapped X axis. 4) Set the **Vertical Gap** to 0 for the two layers. 5) Change page orientation to **Portrait**. Then the source plots will be added as layers in the new graph page.

Keep Source Graphs

Rearrange Layout

Arrange Settings

Number of Rows

Number of Columns

Add Extra Layer(s) for Grid

Keep Layer Aspect Ratio

Link Layers

Show Axes Frame

Overlapping axes/ticks are hidden

Spacing (in % of Page Dimension)

Horizontal Gap

Vertical Gap

Left Margin

Right Margin

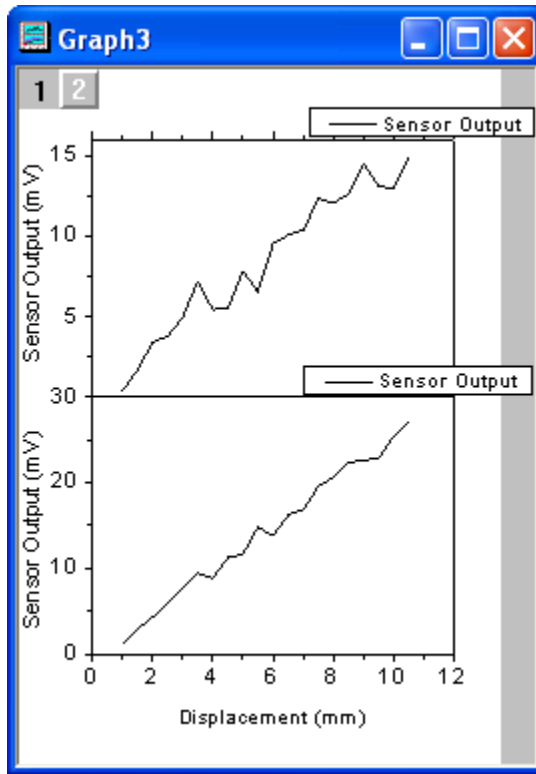
Top Margin

Bottom Margin

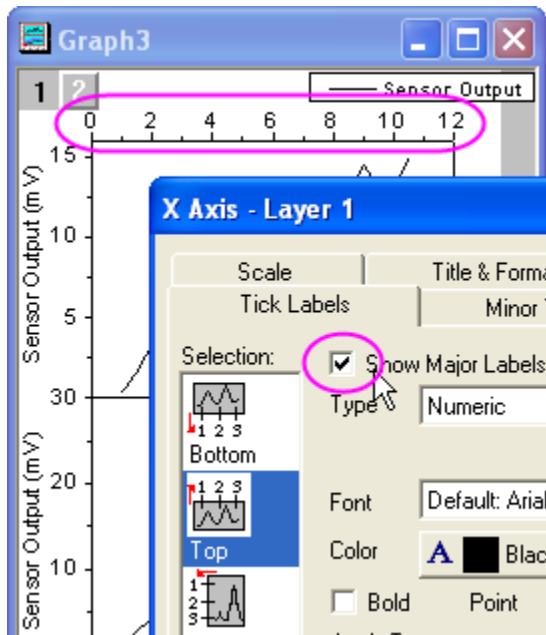
Page Setup

Orientation

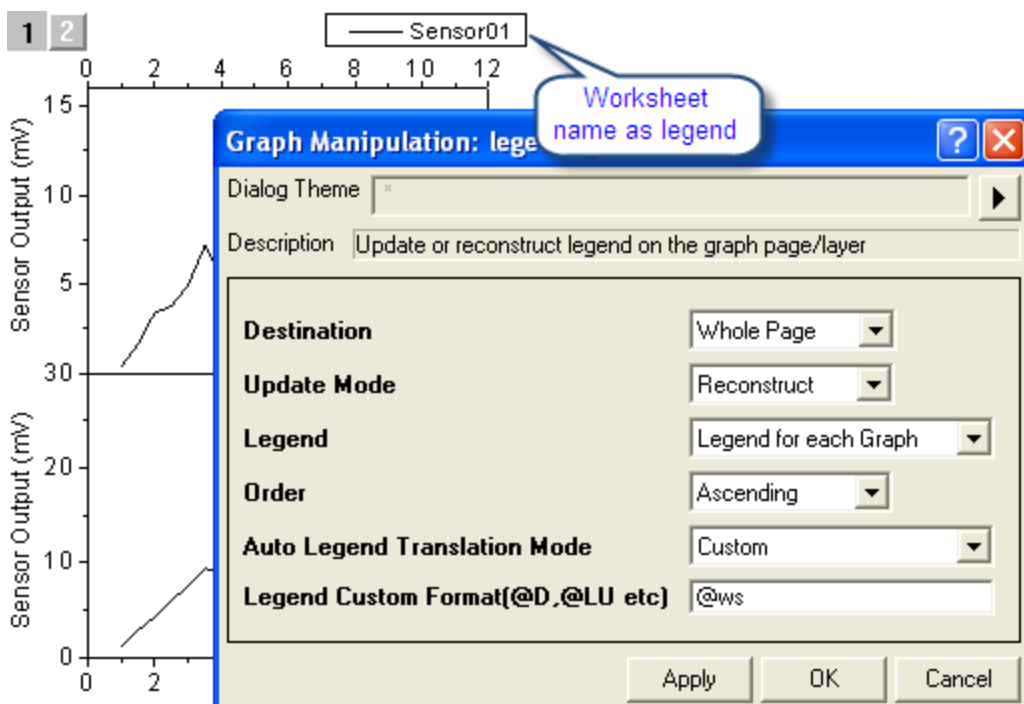
- Click **OK** to close the dialog, and a new merged graph page is created:



- Double-click the top X-axis and add tick labels as below:

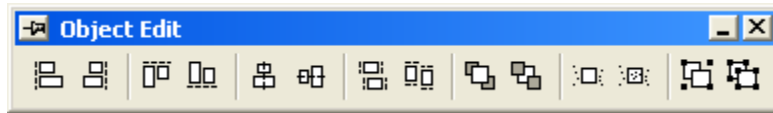






7. The two graph legends are the same in the output page. For the graph legend, Origin uses the worksheet column's **Comments** label row, or if that's empty it will use the **Long Name** if there is one, and if not, the Column name. To modify the legends, select **Graph: Update Legend** from the menu to open the legend update dialog. You will change the legends for the **Whole Page**, and **Reconstruct** them. Select **Custom** from **Auto Legend Translation Mode** and enter **@ws** in the **Legend Custom Format** edit box. Here, @ws means Origin will use the worksheet name as the graph legend.



Aligning Layers using the Object Edit Toolbar

1. Select **File:Open** and open the `OPJ \Samples\Graphing\Layer Management.opj` and go to the subfolder **Arranging Layers**. (If you don't see the subfolders, click **View: Project Explorer** to open the Origin Project Explorer window.)
2. Now we want to use the **Object Edit** tools to rearrange the graph. Make sure you already opened this toolbar, or you can open it from the **View: Toolbars** dialog.



3. On the graph, hold the **Shift** key down and click on all four layers to select them as a group. Then press the **Uniform Width**  and **Uniform Height**  buttons on the **Object Edit** toolbar to make them the same height and width.
4. Click and select the bottom two layers and click the **Bottom**  button in the Object Edit toolbar to align them. Do the same thing for the top two layers.
5. Now click and select the top and bottom on the left column, and do **Left align** , then repeat for the other two in the right side column.



The object edit toolbar provides a quick way to align and set the size of layers. The first layer you select is the reference layer and all others will adjust according to that one. The **Layer Management** tool provides many more options, such as reordering and linking layers, in addition to setting size and alignment.

6. Do not save changes to your project, as the same project is used later to demonstrate Layer Management.

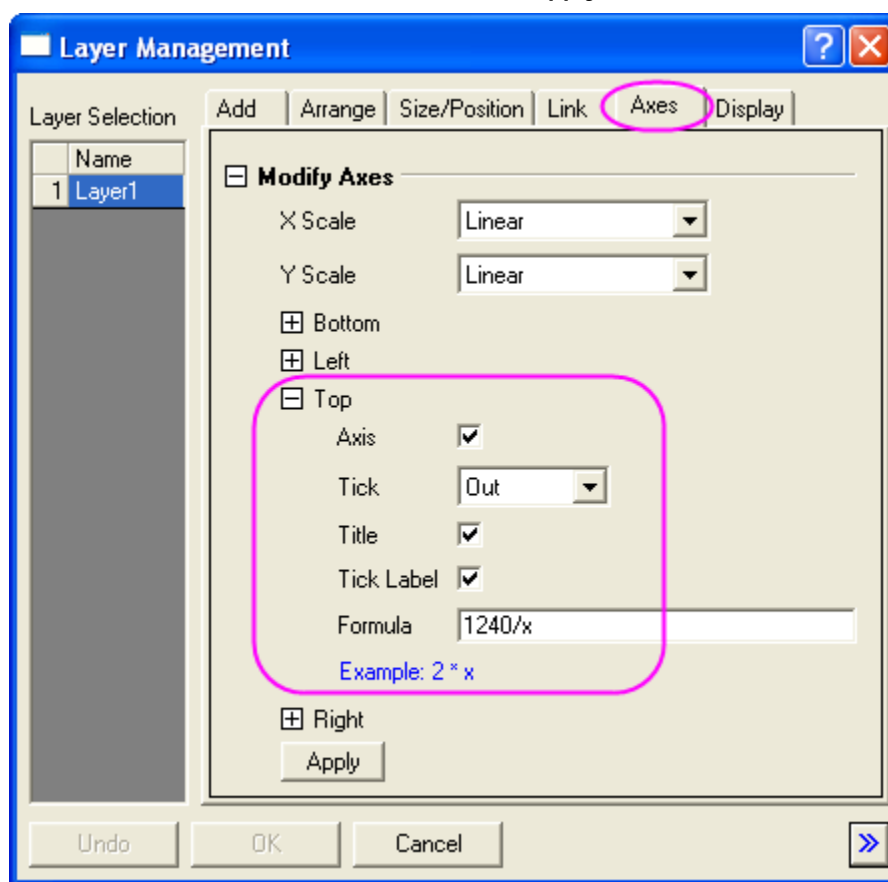
Displaying Opposite Axes with a Nonlinear Formula

An Origin 2D graph layer is a set of X, Y axes, and opposite axes can be turned on for both X and Y. In addition, the opposite axes can also display labels using any user-specified nonlinear formula with respect to the primary axis.

1. Using the Project Explorer window, switch to the subfolder named **Nonlinear Axis**.
2. With the graph active, bring up the Layer Management tool by selecting **Graph: Layer Management**.
3. Switch to the **Axes** tab and expand the **Top** branch and check the **Axis**, **Title**, and **Tick Label** check boxes.
4. We want to display the labels on the top axis in units of Energy and the relationship between wavelength and energy is:

$$\text{Energy (eV)} = 1240/\text{Wavelength (nm)}$$

So in the **Formula** box, enter: 1240/x and click **Apply**, then click **OK** to close the dialog.



5. Double-click and edit the top X-axis title and change it to: Energy (eV)

- Double-click the top axis labels or axis, to open the X-Axis dialog. Switch to the Custom Tick Labels tab, and then click Hide radio for At Axis Begin, to hide the missing value label for energy that corresponds to zero wavelength.

1

The image shows a graph with the x-axis labeled "Energy (eV)". The x-axis has major ticks at 2.48, 1.24, 0.83, 0.62, 0.50, 0.41, and 0.35. There is a missing value label "..." at the far left of the axis. A dialog box titled "X Axis - Layer 1" is open, showing the "Custom Tick Labels" tab. The "Special Ticks" section has the "Hide" radio button selected for "At Axis Begin".

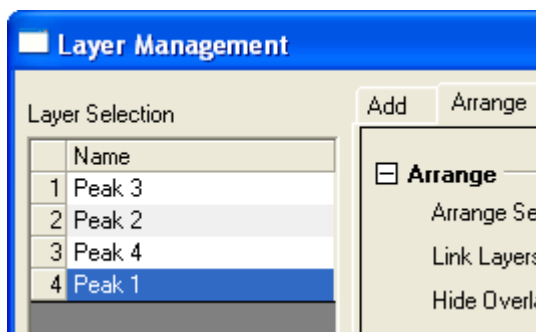


If you are setting up a nonlinear relationship for the right Y-axis, in the **Formula** you still need to use **x** instead of **y**.

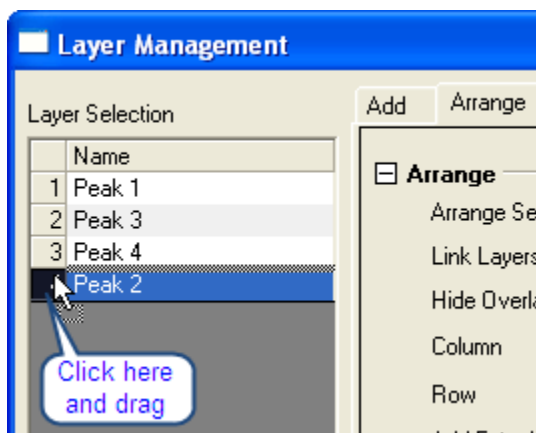
Using Layer Management to Link and Position Layers

The Layer Management tool can be used to position, resize, swap, and link layers in order to establish the desired relationship between layers, as demonstrated in the following steps:

1. Reopen the project `\Samples\Graphing\Layer Management.opj` and then switch to the subfolder named *Arranging Layers*. You can reopen the project by selecting **File: Recent Projects: Layer Management.opj**. Do not save changes to the project.
2. Bring up the Layer Management tool from **Graph: Layer Management**. Then go to left panel and rename the layers so that they are, top to bottom: Peak 3, Peak 2, Peak 4, Peak 1. The layer names now correspond to the legend for each layer. Note: to rename a layer, double-click on the name, as the hint text in the dialog says.



3. Now drag and arrange the list on the left so that they are ordered Peak 1, Peak 2, Peak 3, Peak 4. The layer number and names now match.

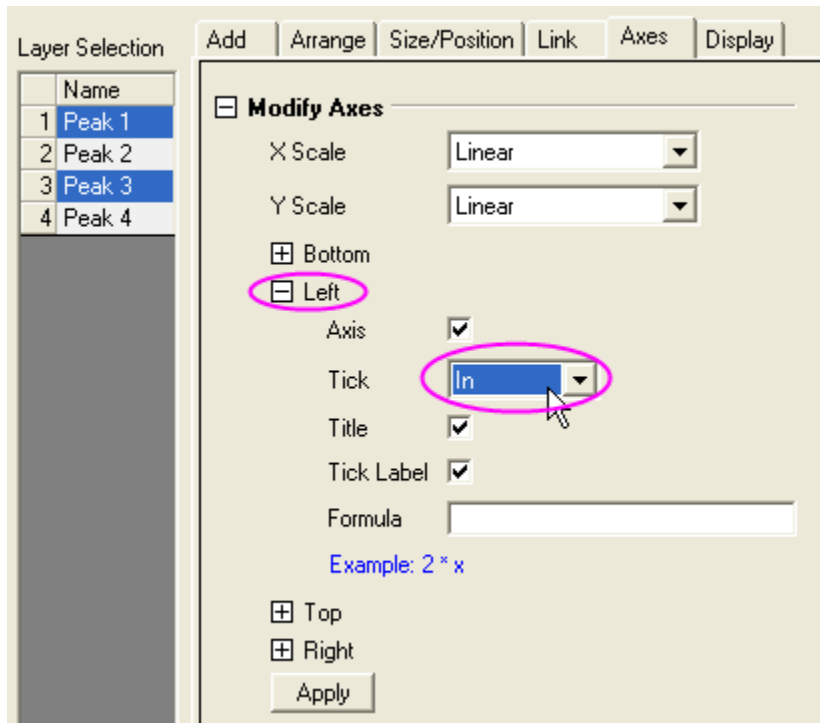


4. Now switch to the **Arrange** tab and do the following steps: 1) Uncheck the **Link Layers** check box. 2) Check **Show Axes Frame**. 3) Leave the **Number of Columns** and **Number of Rows** edit boxes as 2 by 2. 4) Set **Horizontal Gap** and **Vertical Gap** to 0.

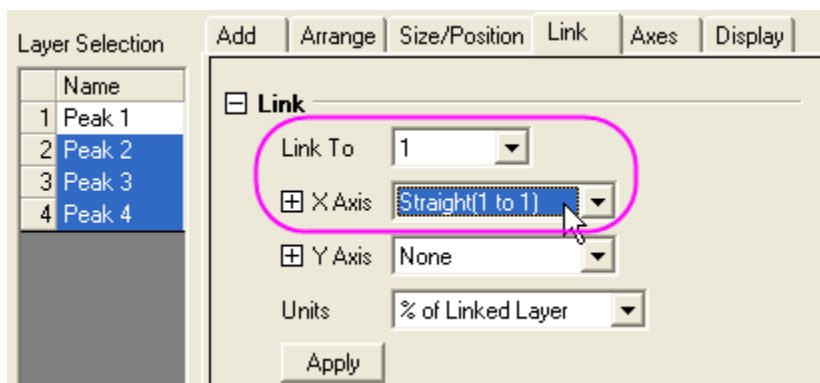
Add	Arrange	Size/Position	Link	Axes	Display
Arrange					
Arrange Selected Layers <input type="checkbox"/>					
Number of Rows <input type="text" value="2"/>					
Number of Columns <input type="text" value="2"/>					
Add Extra Layer(s) for Grid <input checked="" type="checkbox"/>					
Keep Layer Aspect Ratio <input type="checkbox"/>					
Link Layers <input type="checkbox"/>					
Show Axes Frame <input checked="" type="checkbox"/>					
Overlapping axes/ticks are hidden					
Spacing (in % of Page Dimension)					
Horizontal Gap <input type="text" value="0"/>					
Vertical Gap <input type="text" value="0"/>					
Left Margin <input type="text" value="15"/>					
Right Margin <input type="text" value="10"/>					
Top Margin <input type="text" value="10"/>					
Bottom Margin <input type="text" value="15"/>					
<input type="button" value="Apply"/>					

Click **Apply**. This resizes and repositions the layers so they are aligned, and hides ticks and labels where layers overlap.

5. Now go to the Axes tab, hold down the **Ctrl** key and select Peak 1 and 3 on left list, then expand the **Left** branch and set tick direction to **In**, and click **Apply**.

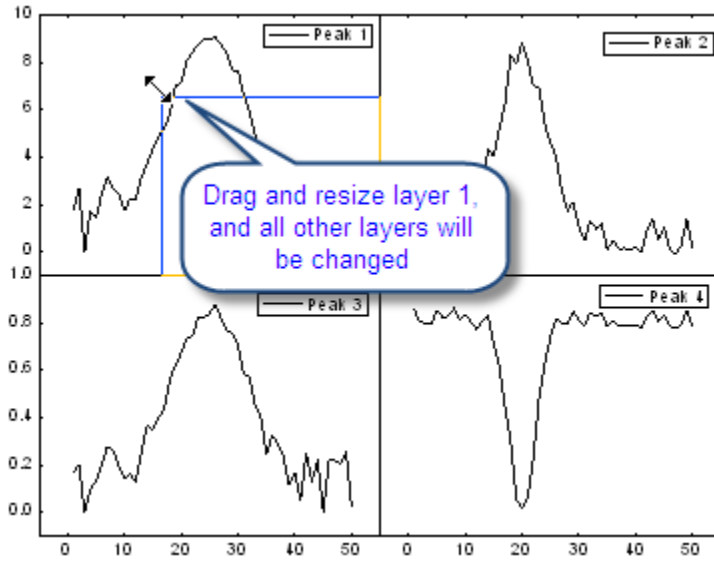


6. Now select Peaks 3 and 4, set the **Bottom** tick direction to **In** and click **Apply**. Then select Peaks 1 and 2, set the **Top** ticks to **In** and click **Apply**.
7. Now go to the Link tab and select Peaks 2, 3, and 4 in the left list and link them to layer 1, making sure you have the X-Axes linked **Straight(1 to 1)**, and click **Apply**.

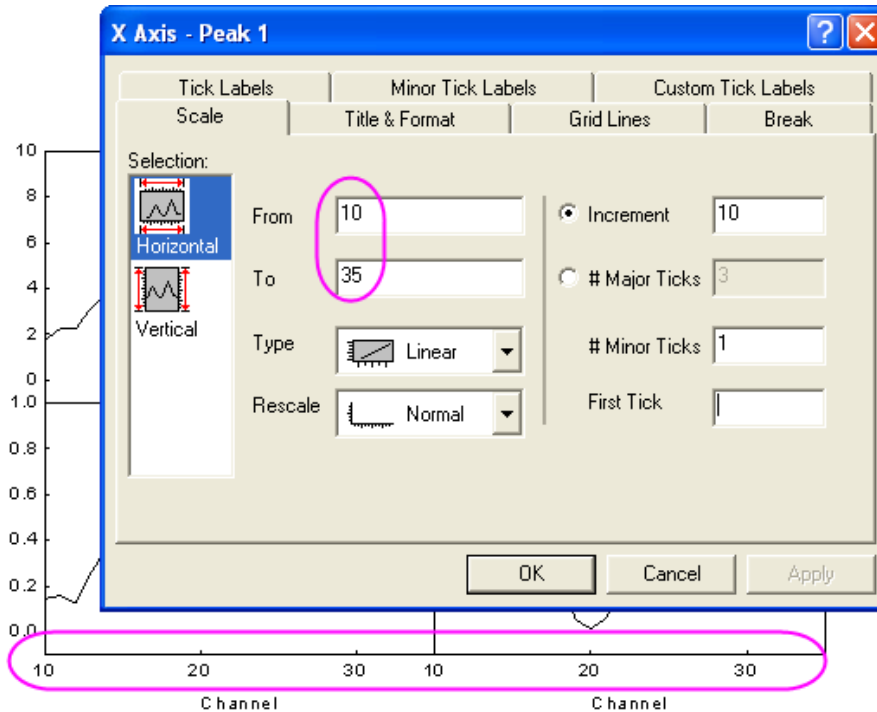


Then click **OK** to close the dialog.

- Click and select the top left layer, layer 1, then drag and resize the layer and move the layer. Note that the other layers resize and move too, since they are linked by dimension to layer 1.



- Go to the top left layer (layer 1), and double-click the top X-axis. In the X-Axis dialog, change the X-Axis scale From 10 To 35, and click **OK**. You'll see that all the other layers now show the same new x range, as they are all linked in X.



When arranging linked layers, they are treated as one unit, so you should first unlink them if you need to rearrange and then you can link them again after arranging.

5 Programming

5.1 LabTalk

5.1.1 Advanced Scripting using LabTalk

Summary

This tutorial demonstrates how to use some advanced LabTalk scripting commands and methods to organize your script files. To learn more about all the commands and methods supported in LabTalk, please refer to **Help: Programming: LabTalk**.

Minimum Origin Version Required: Origin 8.1 SR1

What you will learn

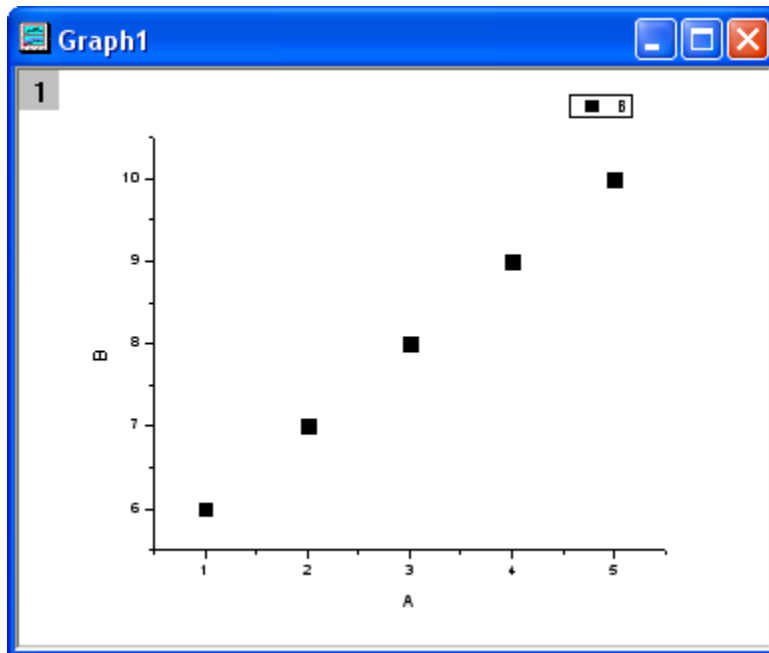
This tutorial will show you how to:

- Modify Plot Attributes via Script
- Define a LabTalk Script Macro Command
- Define a Macro
- See Origin's Predefined System Macros
- Load and Compile your Origin C Function using LabTalk script
- Use .OGS Files to Store Script.

Modifying Plot Attributes via Script

This section demonstrates how to use script commands to change the attributes of a data plot.

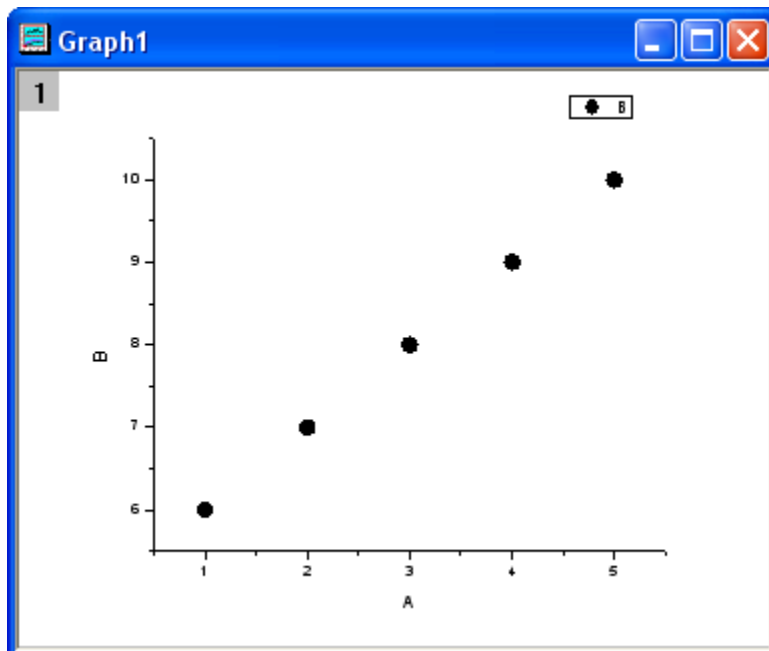
1. Start a new project, enter the numbers 1 through 5 in column A of the Data1 worksheet and numbers 6 through 10 in column B. If you have not already done so, open the Script Window by selecting **Window: Script Window** from the Origin program menu.
2. Using the worksheet data from the previous exercise, create a scatter plot. Note that the scatter plot symbol is a black filled square (symbol size is increased for clarity).



3. To change the symbol shape, type the following:

```
set %C -k 2
```

4. Press **ENTER**.

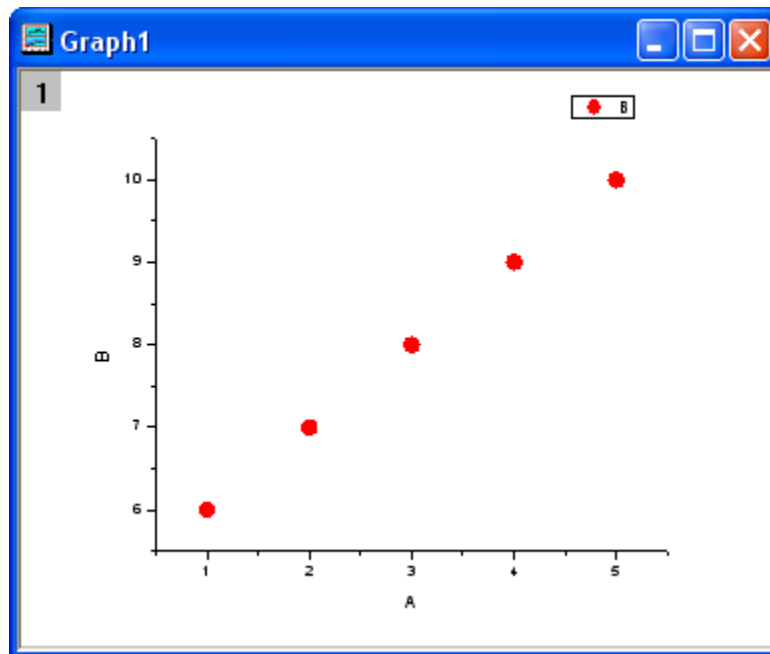


The data plot symbol changes from a filled square to a filled circle (the numbers correspond to Origin's symbol list; 1 = square, 2 = circle, etc.).

5. To change the symbol color, type the following:

```
set %C -c 2
```

6. Press **ENTER**.



The data plot symbol color changes from black to red.

To modify the axis scale values

7. Type the following:

```
x1=0;x2=20;y1=0;y2=10
```

8. Press **ENTER**.

Your X-axis scale now reads from 0-20, and the Y-axis reads from 0-10.

Notes: As this example illustrates, you can type multiple lines of script in a single line by separating commands with a semi-colon.

You can also use the set command to specify the data display range.

9. Type the following:

```
Set %C -b 2
```

10. Press **ENTER**.

The graph's display range now begins with the second data point in the data set.

11. Type the following:

```
Set %C -e 4
```

12. Press **ENTER**.

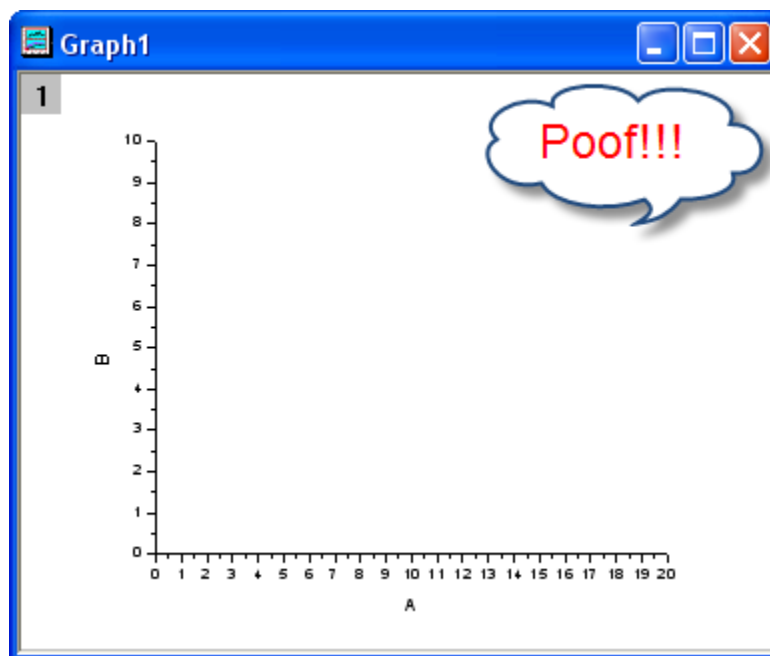
The graph's display range now ends with the fourth data point in the data set.

You can also hide or show a data plot using the set command's `-s` switch.

13. Type the following:

```
Set %C -s 0
```

14. Press **ENTER**.



The active data set is now hidden.

15. To show the hidden data set, type:

```
Set %C -s 1
```

16. Press **ENTER**.

Defining a LabTalk Script Macro Command

A macro is a convenient method of aliasing a LabTalk script. When you define a macro you are associating an entire script with a specific name. This name becomes a command that invokes the associated script.

When developing scripts, macros can provide several advantages.

- Modular code can streamline a script by replacing repetitive or similar blocks of code with multiple calls of the same macro.
- Modifications to your code become easier to implement because you only have to redefine your macro as opposed to modifying repeated blocks of code that are scattered throughout your application.
- There is a limit to the number of tokens that can be included between a set of curly braces that enclose script. Macros provide a means to shorten the code between braces by calling on a pre-defined macro.
- You can modify the behavior of a LabTalk command by creating a macro of the same name. The functionality of the LabTalk command is restored when the macro is deleted.

Defining a Macro

A macro is defined using the define command. The general syntax is:

```
define macroName {
    script
}
```

where *macroName* and *script* are the name of the macro and the body of the macro, respectively.

To define a macro using LabTalk's define command:

1. From the Origin menu, select **Window: Script Window**.
2. Type the following:

```
def hello {
    type -b "Hello World!!!";
}
```

This script defines a macro named hello that will type "Hello World!!!".

<p>Notes: The define command can be abbreviated as def</p>

We will now use the Script Window to call our hello macro.

3. Type the following into the Script window:

```
hello
```

4. Highlight all codes in the Script Window and press **ENTER**.

An attention dialog opens to say "Hello World!!!".



Origin's Predefined System Macros

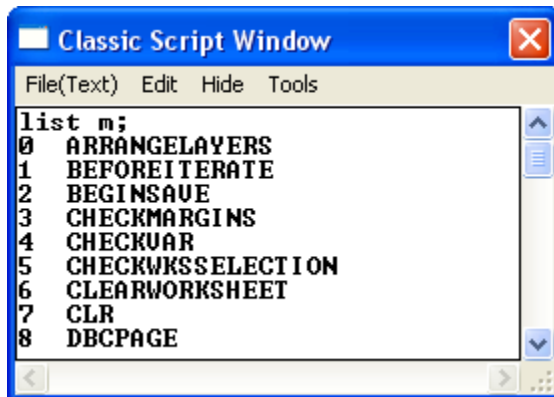
Let's look at Origin's predefined system macros, some of which take in arguments. Macros can take up to five arguments. Use the %1, %2, ? %5 notation within the script definition to indicate that the macro expects one or more arguments (%1= 1st argument, %2= 2nd argument, ?%5 = 5th argument).

1. In the Script Window, type the following:

```
list m
```

2. Press **ENTER**.

Origin responds by typing the names of predefined macros into the Script Window.



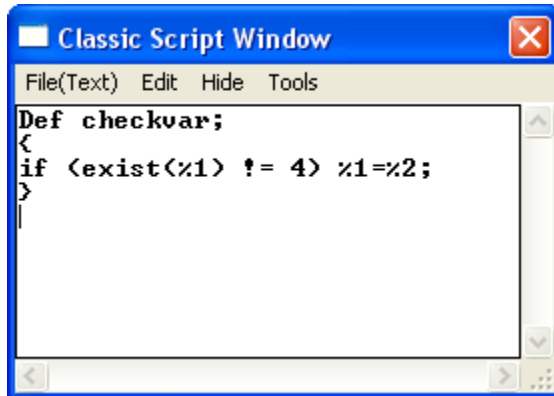
To see the definition of any system macro:

3. To see how a system macro is defined, type **def *macroname***. For example, type the following in the Script Window:

```
def checkvar
```

4. Press **ENTER**.

Origin responds to the Script Window as:



The %1 notation in the macro definition indicates that this macro takes one argument.

To define a new macro as a system macro:

5. Type the following in the Script Window:

```


def graph {
    set %1 -s 1;layer -i %1
}

```


6. Press **ENTER**.

Notes: We used the set data set ?s 1 command to show a data plot. The layer -i data set command adds (plots) the named data set onto the active layer.

To call a macro:

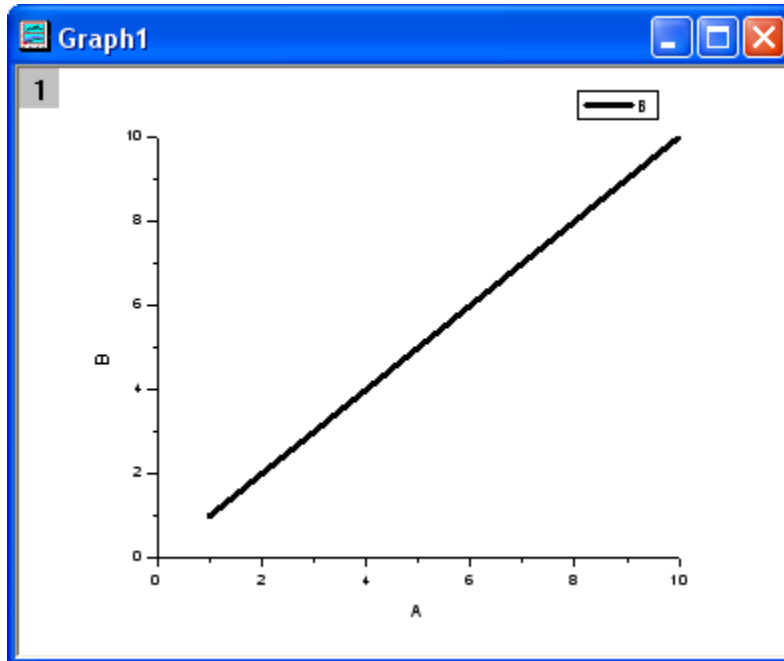
7. Click the **New Worksheet** button 
8. Create a worksheet named *Book1* and type in the following data:

	A(X)	B(Y)
Long Name		
Units		
Comments		
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7

9. Click the **New Graph** button 
10. In the Script Window, type the following:


```
graph book1_b
```
11. Press **ENTER**.

A line plot of the data set book1_b is included in the graph window.



We will modify the macro definition so that it creates a scatter plot using a red, ?up triangle? as the symbol.

You could hard code the appropriate values for scatter plot, red, and up triangle in the macro definition, but it is more efficient to pass the value of a variable as an argument. This way, the macro may be used in other instances when you want to set the color and symbol shape to something other than red and upward-pointing triangle .

Notes: LabTalk often uses integer values to specify plot details. If you look at the color palette, for instance (from the menu, **Format: Color Palette**), you will see (assuming that you have not modified the default color palette) that black =1, red = 2, green = 3, blue = 4, etc. For more information, see documentation on the **Set** command in the LabTalk Language Reference section of the Programming Help file.

12. Type the following in the Script Window to redefine the graph macro:

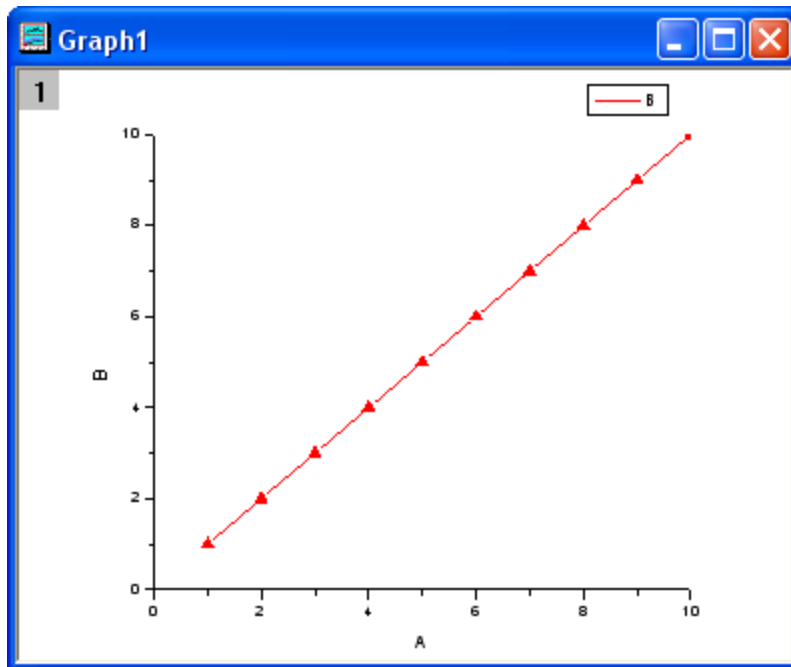
```
def graph {
  set %1 -s 1;
  layer -i %1;
  set %1 -c %2;
  set %1 -k %3;
}
```

13. Type the following:

```
graph book1_b 2 3
```

14. Press **ENTER**.

This plots the data set book1_b as red, upward-pointing triangles.



To better understand what we actually did, let's examine our macro line by line.

First, we used the **def** command to tell Origin that we are defining a macro.

```
def graph
```

Secondly, we used the LabTalk **set** command with the **?s** option.

The **set ?s** command syntax is:

```
set dataset -s value
```

where *dataset* is the name of a data set, and *value* is either 1 (show plot) or 0 (hide plot),

```
set %1 ?s 1;
```

The layer **?i#** command syntax is:

```
layer ?igraphType dataset;
```

Note that the data set name has been assigned to %1.

The **set ?c** command syntax is:

```
set ?c color#
```

This is used to specify the plot symbol color. Note that color will be assigned to %2.

The **set ?k** command syntax is:

```
set ?k shape#
```

This is used to specify symbol shape. Note that symbol shape is assigned to %3.

When we execute our macro by typing

```
graph book1_b 2 3
```

we are passing three arguments to the macro:

- book1_b, which is substituted for %1.
- 2, which is substituted for %2.
- 3, which is substituted for %3.

Note that it is merely coincidence that we chose to substitute a value of 2 for %2, and a value of 3 for %3. We could have chosen any allowed value for symbol color or shape.

Remember that any macro that you define is only available for the duration of your Origin session. If you restart Origin, you cannot execute your macro until you define it again. If you want your macro to be defined automatically when you start Origin, you can save your macro definition to Origin's MACROS.CNF file. Each time Origin starts, it reads MACROS.CNF, and your macro is defined.

Notes: MACROS.CNF is located in the Origin software folder. Because of a turf battle over the .CNF file extension, it will probably only be listed as MACROS and will display a terminal icon. In reality, this file is a text file and can be opened in any text editor, such as Notepad and Origin Code Builder.

For more information on macros, see **Help: Programming: Labtalk** in the Help menu.

Loading and Compiling your Origin C Function using LabTalk script


Before an Origin C function can be used, it must be compiled and linked in the current Origin session. Origin provides the following method to programmatically compile and link a source file, or to programmatically build a workspace, from LabTalk.

```
err = run.LoadOC("myFile",[option]);
```

Notes: For more on the LabTalk run object in **Help: Programming:Labtalk Help**.

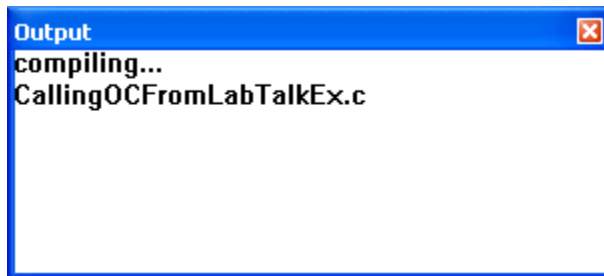
The following example demonstrates how to programmatically load and compile an Origin C source file.

To begin this tutorial:

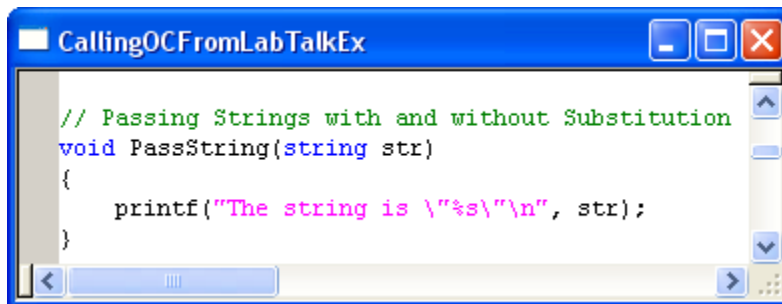
1. On the **Standard** Toolbar, click the **Code Builder** button .
2. Return to the Origin workspace and open a New Project (**File: New?Project**).
3. Open the Script Window (**Window: Script Window**) and type the following:

```
string fld$="Samples\Origin C Examples\Programming
Guide\Calling Functions\";
string fname$=System.path.program$ + fld$ +
"CallingOCFromLabTalkEx.c";
run.LoadOC(%(fname$), 1);
```

The CallingOCFromLabTalkEx.c function is compiled now and can be found under the User folder in Code Builder's Workspace view. The Output Window reports as follows in the Code Builder workspace:



The Origin C functions in CallingOCFromLabTalkEx.c are now accessible. You can call the following section from the C file. Note the comments (the green text with leading //).



To execute the PassString function:

4. Return to the Origin workspace, open the Script Window (**Window: Script Window**) and type in the following:

```
PassString abc
```

5. Press **ENTER**. Origin returns:

The string is "abc"

Using .OGS Files to Store Script

As an alternative to associating your LabTalk script or Origin C function with a button, you could save your script to .OGS files. The advantage is that these .OGS files are self-contained and can be called from many buttons.

These .OGS files are organized by sections. Sections are identified by a name surrounded by square brackets, as in this example:

```
[Main]
```

To execute the code in a portion of a .OGS file, you need only identify the .OGS file and refer to the section containing the code by name, as in this example:

```
run.section(test.ogs, Main)
```

Most of Origin's menu and toolbar commands run LabTalk script in a .OGS file. These files can be opened and edited in Code Builder.

In this tutorial, we will create a new .OGS file, and associate the .OGS file with a new toolbar button.

Notes: This tutorial assumes that you have already created and saved an Origin C file called test.c, as prescribed under **Tutorial: Organizing and Accessing Origin C Functions**.

To create a new .OGS file:

1. From the **Code Builder** menu, select **File: New**.
2. In the **New File** dialog box, select **LabTalk Script File**.
3. In the **File Name** text box, type:

```
Test
```

4. Click **OK**.

You now have an empty document called **test.ogs**.

We will use the run.LoadOC script command to programmatically compile and link the test.C Origin C source file. The advantage of this method is that it allows you to program your buttons or other user-created visual objects to make behind-the-scenes calls to your Origin C functions.

To make a call to an ?uncompiled? Origin C function from Origin:

5. In the blank **test.ogs** window, type the following:

```
[CreateGraph]
run.LoadOC("test.c");
Plot_Data("scatter", "book1_b");
```

6. From the Code Builder menu, select **File: Save As** and save the file to your main Origin software folder.

7. Return to the Origin workspace.

8. From the Origin menu, select **View: Toolbars**. This opens the **Customize Toolbar** dialog.
9. On the **Toolbars** tab, click the **New** button to open the **New Toolbar** dialog.
10. Type in the following name for your new toolbar:

My Toolbar

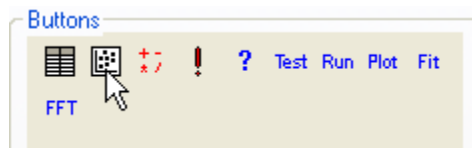
11. Click **OK**.

My Toolbar is added to the **Toolbars** list. A new toolbar is added to the Origin workspace.



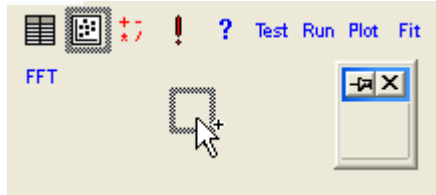
Notes: The toolbar may be a bit hard to spot because it does not contain any buttons.

12. Return to the **Customize Toolbars** dialog and select the **Button Groups** tab.
13. Scroll to the bottom of the **Groups** list and select **User Defined**.
14. Select the second button in this group.



15. Click the **Settings** button. This opens the **Button Settings** dialog box
16. In the File Name text box, type **test.ogs**.
17. In the Section Name text box, type **CreateGraph**. We are not passing arguments to the section, so we can skip the **Argument List** text box.
18. In the **Tool Tip Text** box, type **CreateGraph**. A Tool Tip is the message that displays when you mouse over a toolbar button.
19. In the **Status Bar** text box, type **Example, plotting data from Origin C** as Status Bar Text. When you mouse over a toolbar button, the Status Bar message displays in the lower left corner of your Origin workspace.
20. In the **Context** group, verify that the **Windows** radio button is selected and clear the **Graph**, **Matrix**, **Layout**, and **Excel** check boxes. Leave only **Worksheet** selected. This limits toolbar availability to active worksheets.

21. Click **OK**.
22. Point to the toolbar button, hold down the left mouse button and drag the button to the floating toolbar.



23. Click **Close** to close the **Customize Toolbar** dialog.

To test this method, close and restart Origin. Remember that the second argument to our function is book1_b, so the Origin workspace will need to have a worksheet named book1, a B(Y) column and some data in both the X and Y columns. Note, too, that our toolbar button is grayed out when a graph is the active window.

This concludes the tutorial on **Advanced Scripting Using LabTalk**.

5.2 Origin C

5.2.1 Introduction to Origin C and Code Builder

Summary

Origin C supports a nearly-complete ANSI C language syntax as well as a subset of C++ features including internal and DLL-extended classes. In addition, Origin C is "Origin aware". This means that Origin objects such as worksheets and graphs are mapped to classes in Origin C, allowing direct manipulation of these objects and their properties from Origin C.



Origin C's integrated development environment (IDE) is called Code Builder. Code Builder provides standard tools for writing, compiling, and debugging your Origin C programs. Once an Origin C function is compiled, the function can be called in various ways from the Origin or Code Builder workspaces.

Minimum Origin Version Required: Origin 8.0 SR0

What you will learn


This tutorial introduces you to Origin C and Code Builder by showing you how to write, compile and call a function that types the message "Hello World!!!".

Steps

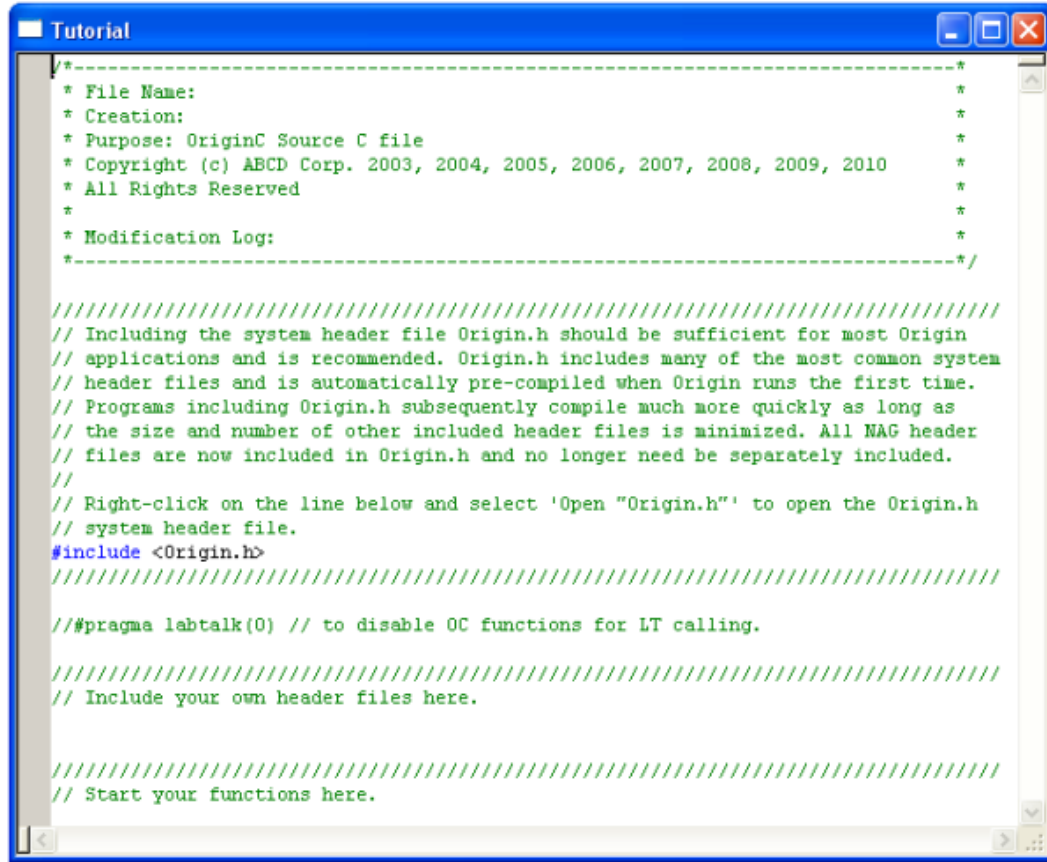
1. On the Origin Standard toolbar, click the **Code Builder** button 
2. On the Code Builder toolbar, click the **New** button 

In the **New File** dialog, select **C File**.

In the **File Name** text box, type **Tutorial**.

In the **Location** text box, select the \Origin C subfolder (Note: The Browse button looks like this: ).

Click **OK**. A file named Tutorial.c opens in the Code Builder workspace.



```

/*-----*/
* File Name:
* Creation:
* Purpose: OriginC Source C file
* Copyright (c) ABCD Corp. 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010
* All Rights Reserved
*
* Modification Log:
*-----*/

////////////////////////////////////
// Including the system header file Origin.h should be sufficient for most Origin
// applications and is recommended. Origin.h includes many of the most common system
// header files and is automatically pre-compiled when Origin runs the first time.
// Programs including Origin.h subsequently compile much more quickly as long as
// the size and number of other included header files is minimized. All NAG header
// files are now included in Origin.h and no longer need be separately included.
//
// Right-click on the line below and select 'Open "Origin.h"' to open the Origin.h
// system header file.
#include <Origin.h>
////////////////////////////////////

#pragma labtalk(0) // to disable OC functions for LT calling.

////////////////////////////////////
// Include your own header files here.

////////////////////////////////////
// Start your functions here.


```

3. Type the following beneath the line that reads // start your functions here:

```

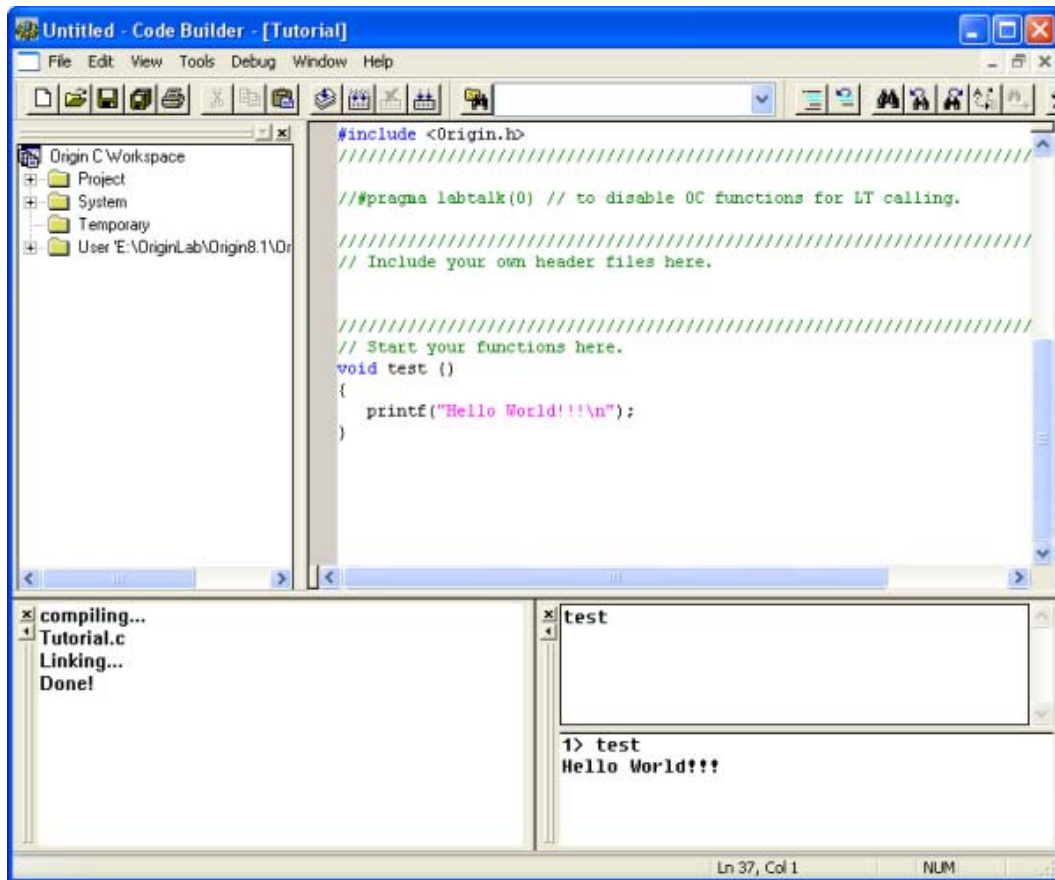
void test ()
{
    printf("Hello World!!!\n");
}

```

4. On the Code Builder workspace Standard toolbar, click the **Build** button . This compiles the test function.
5. To call this function, click in the upper pane of the **LabTalk Console**. This is located in the lower right corner of the **Code Builder** workspace (This is the default location. If the LabTalk console isn't visible, select **View: LabTalk Console** from the Code Builder menu and make sure that the menu item is checked).
6. Type the following LabTalk function call in the **LabTalk Console**:

```
test
```

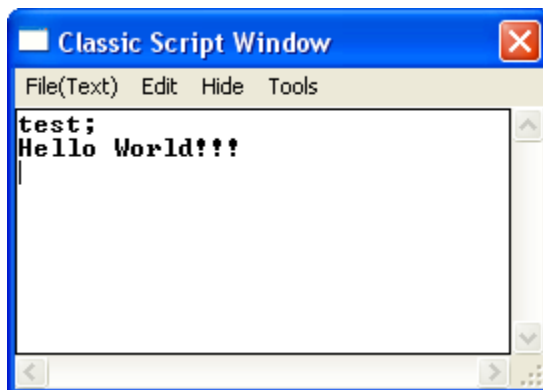
7. Press **ENTER**.



To test this function in the Origin Script Window:

8. Return to the Origin workspace, and select **Window: Script Window**.
9. In the Script window, type the following:


```
test
```
10. Press **ENTER**.



"Hello World!!!" displays in the Script Window.

This concludes the **Introduction to Origin C and Code Builder** tutorial.

5.2.2 The Code Builder Workspace

Summary

In this exercise, we will create a workspace, add a source file with a new function, then build, test and save the workspace file.

Minimum Origin Version Required: Origin 8.0 SR0

What you will learn

This tutorial will show you how to:

- Build a Workspace File
- Build Workspace Folders
- Build on Startup

The Workspace File


A workspace is a collection of files that can be opened by a single menu option (**File: Open Workspace?**) in Origin's Code Builder. Any text file can be a part of the collection. They do not necessarily have to be source code files; they could be notes, for example.

All files opened in the Multiple Document Window by a workspace can be edited and saved individually. In addition to files being opened in the Multiple Document Window, source code files can be added to the Workspace Window with the **File: Add to Workspace** menu option.


By including source code files in the Workspace Window, you can build individual or multiple files with the appropriate menu option or toolbar button. Header files can be referenced within source files and do not need to be loaded in the Workspace Window or even open in the Multiple Document Window.

Since you can save a workspace with a new name, you can have multiple workspace files. However, only one workspace file can be open at a time.

To create a workspace:

1. On the **Standard** toolbar, click the **Code Builder** button .
2. From the Code Builder menu, select **File: New Workspace**. This creates a new workspace with the default name of "Untitled.ocw".
3. From the Code Builder menu, select **File: New**. This opens the **New** File dialog.
4. Choose **C File**, and type **foo** in the **File Name** text box. The **Add to Workspace** and **Fill with Default Contents** check boxes should be selected. You may accept the default Location. Click **OK**.
5. In **FOO.C**, starting below the line that says `?\\start your functions here"`, type the following:

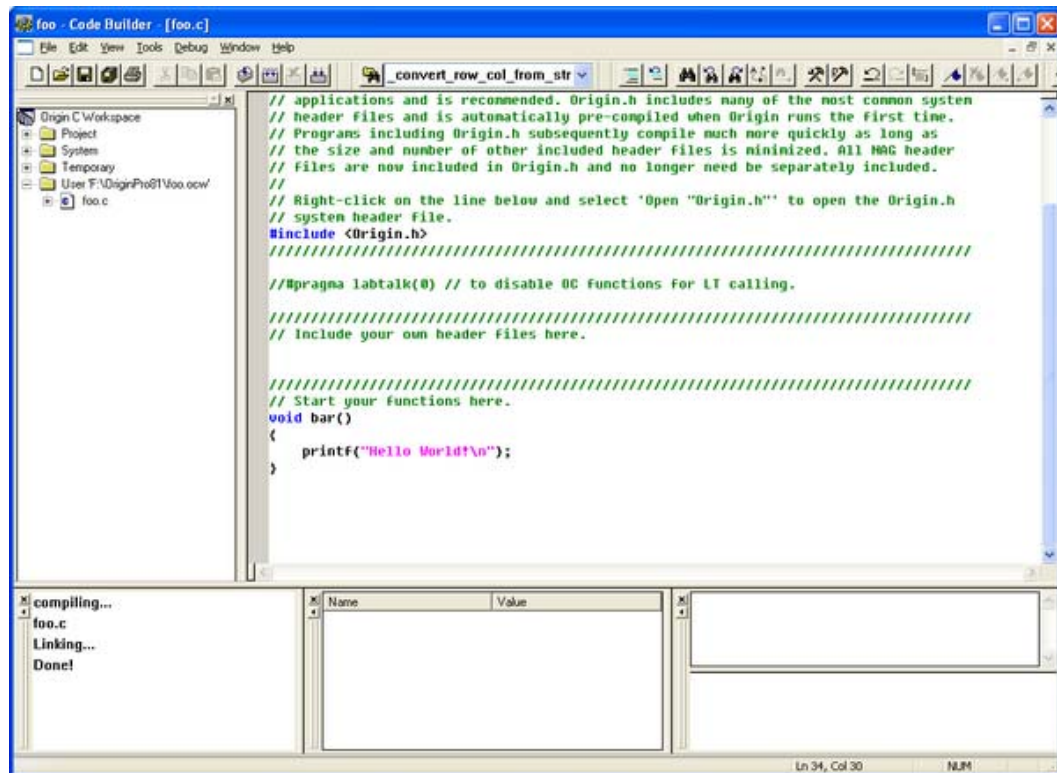
```
void bar()
{
    printf("Hello World!\n");
}
```

6. Click the **Build** button . Origin automatically saves the source file and compiles and links the function.

7. From the Origin menu choose **Window: Script Window**.
8. To test our new function, type:

bar
9. Press **ENTER**
Origin responds by typing **Hello World!!!**
10. From the Code Builder menu, choose **File: Save Workspace As?**.

The figure shows the foo.ocw workspace file containing a single source file, foo.c, in the Multiple Document Window. The file has been added to the Workspace Window. The Output Window shows that the file has been compiled. The source file contains a single function `bar()` which is listed in the tree structure of the workspace.



Workspace Folders

The Code Builder workspace has four subfolders named Project, System, Temporary, and User. Files added by user, such as foo.c that you just added, are placed in the User subfolder. Origin, itself, uses Origin C for many analysis routines. When these routines are accessed, the Origin C source files are loaded into the Workspace into either the System subfolder or the Temporary subfolder. The Project subfolder is reserved for files that are saved and loaded with the Origin project. This aspect of attaching a file to the project is described in a separate tutorial.

Build on Startup

If you right click on "Origin C Workspace", the shortcut menu has a **Build on Startup** option. When this option is checked, the last workspace you saved will be loaded when you restart Origin. All source files in the Workspace will be built and all functions in the source files will be available for immediate use.

For information on building individual source files on startup by including information in the ORIGIN.INI file, please view Build on Startup

This concludes the tutorial on the **Code Builder Workspace**.

5.2.3 Adding New Origin C Functions to Origin

Summary

Functions written in Origin C are accessible from various locations within the Origin interface, such as the Script Window, provided they meet the following criteria: the function should return either void (as in the previous tutorial), double, string, or vectors of type double or string. Variables passed to the function from Origin should be of type double or string, or vectors of these types. Functions that do not meet these criteria are not callable from the Origin interface, but can be called within other Origin C functions. Note that although an Origin C function that accepts and returns type int can be called from the Origin interface, the data may be truncated since the interface only supports type double.

In this tutorial, you will be introduced to writing a math function that returns computed values. We will first create a function that returns type double to Origin, and then we will create a function that returns vectors of type double.

Minimum Origin Version Required: Origin 8.0 SR0

What you will learn

How to add a new function and how to run this function in the Script Window.

Steps

1. Start a new Origin C file in **Code Builder**.
2. Enter the following code:

```
double myfunc1(double x, double a)
{
    return sin( a * x );
}
```

3. Click the **Build** button  to compile the function.

This function can now be called from the Origin interface, in places such as the Script Window.

4. Go to the Script Window, and type in the following lines, pressing **ENTER** at the end of each line:

```
y = myfunc1(2, 3)
y =
```

You can also use worksheet cells instead of absolute numbers:

5. Make a worksheet active, enter a number in the first row of column A. Then type the following into the Script Window and press **ENTER**:


```
col(B)[1] = myfunc1(col(A)[1], 3)
```

Note that a function such as myfunc1, that accepts and returns type double, can also be used to perform vector operations.

6. Fill rows 1 through 10 of Column A with numbers, and type the following into the Script Window:

```
col(B) = myfunc1(col(A), 3)
```

In the above example, Origin calls the myfunc1 function for each row of column A. For performing vector operations as above, it is more efficient to write functions that accept and return vectors.

7. Go back to Code Builder and add the following function to the same file, and compile the file by clicking the **Build** button 

```
vector<double> myfunc2(vector<double>& vecIn, double a)
{
    vector<double> vecOut;
    vecOut = sin( a * vecIn );
    return vecOut;
}
```

8. Go back to the Origin interface, fill Column A with some new numbers, and type the following into the Script Window:

```
col(B) = myfunc2(col(A), 3)
```

The function myfunc2 is called only once for computing the entire column.

Note that you can use such functions in other places such as the "Set Column Values" dialog. The Auto Update feature of "Set Column Values" can be enabled by checking the appropriate check box in this dialog. As long as the Origin C function is compiled and ready to be called from Origin, any changes to the source column will result in an update of the destination column.

5.2.4 Organizing and Accessing Origin C Functions

Summary

Techniques for using your Origin C functions.

Minimum Origin Version Required: Origin 8.0 SRO



What you will learn

This tutorial will show you how to:

- Save your Origin C Function with your Project
- Associate your Programs with Visual Objects
- Load and Compile your Origin C Function from script

Saving your Origin C Function with your Project

One way to load and compile your Origin C function is to save the Origin C file as an attachment to your Origin Project (*.OPJ) file. When a project file is opened, all files attached to it are separated out and stored in a temporary folder. In addition, any Origin C file that was attached is also automatically loaded into the Code Builder workspace, and compiled. The function is then ready to be called upon opening the Origin Project file.

1. Start a new Origin project file by clicking on the **New Project** button  on the **Standard** Toolbar.
2. On the **Standard** Toolbar, click the **Code Builder** button .
3. From the Code Builder menu, select **File: New**. This opens the **New File** dialog box.
4. In the top list-box, select **C File**.
5. In the **File Name** text box, type: **Test**. Keep the **Add to Workspace** check box selected. Click **OK**. The file Test.c is added to the workspace.
6. Select and copy the following function, and paste it into the Test.c file. Be sure to paste the text below the line that reads "//start your functions here."

```
void Plot_Data(string strTemplate, string strData)
{
    // Create a graph window from a Template
    GraphPage gp;
    BOOL bOK = gp.Create(strTemplate, CREATE_VISIBLE);
    if( !bOK )
        return;

    // Attach the first layer (0) to a GraphLayer object
    GraphLayer gl = gp.Layers(0);

    //Attach a dataset to a Curve object
    Curve crv(strData);

    // Add the Curve to the graph layer
    int nPlot = gl.AddPlot(crv);
    if(nPlot >= 0)
```



```

{
    // Set plot color to Green(2)
    gl.DataPlots(nPlot).SetColor(2, TRUE);

    // Rescale this graph layer
    gl.Rescale();
}
}

```

The Plot_Data function takes two arguments: (1) the template name and (2) the name of a Y dataset to include (plot) in the layer.

7. Click the **Build** button  to compile and link the file.
8. Drag-and-drop the file Test.c from the User subfolder branch of the Workspace tree, to the Project subfolder. (**Hint:** You may need to first expand the User subfolder branch to display the Test.c entry prior to dragging the file).
9. Go back to the Origin interface and save the project by clicking the **Save** button  on the Standard Toolbar. Give the project the name Test.OPJ, and save it in a location of your choosing.
10. The Origin C file, Test.c, is now saved with the Project. To verify this, close the project, and go back to Code Builder. You will see that there are no entries under the Project subfolder of the Workspace tree. Now go back to Origin interface and reopen the project. Go to Code builder and verify that Test.c is now listed under the Project subfolder (**Hint:** you may need to expand the Project subfolder branch to see the Test.c entry).

Associating your Programs with Visual Objects

You will now learn how to create a button on a worksheet and program the button to call the Origin C function in the Test.c file that you saved with the project.

1. Open the Test.OPJ project that you saved under step 9 (previous section).
2. Highlight the A(X) and B(Y) columns, right-click and select **Fill Columns With: Row Numbers**.
3. From the menu, select **Format: Worksheet** to open the **Worksheet Properties** dialog box.
4. In the **Size** tab, **Worksheet Measurement** branch, set the **Gap from Top** to **40** and click **OK** to close the dialog.

	A(X)	B(Y)
Long Name		
Units		
Comments		
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6

The worksheet now has sufficient space above the column headings to add a text label.

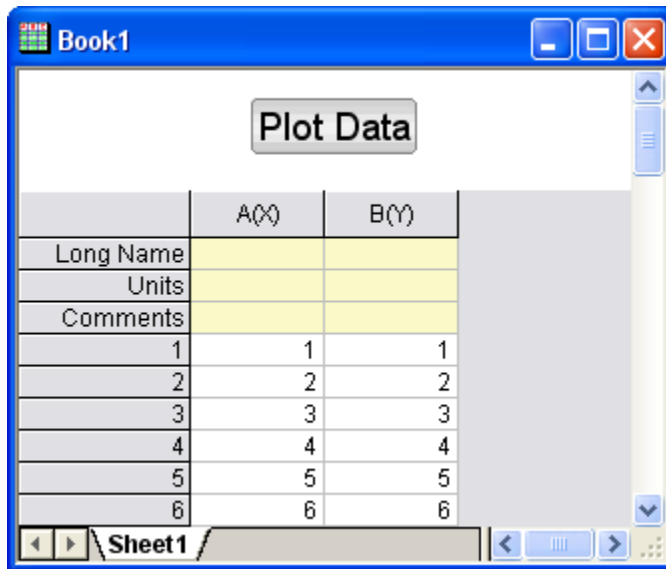
Right-click in the area directly above the two columns and choose **Add Text**.

5. At the cursor, type: **Plot Data**.
6. Click once outside the text label to deselect it.
7. Right-click on the text label and choose **Programming Control** to open the **Programming Control** dialog box. (Hint: Please choose **Label Control** in Origin 7.5)
8. From the **Script, Run After** drop-down list, choose **Button Up**.
9. Type the following script in the text box at the bottom of the dialog box:

```
Plot_Data("scatter", "book1_b");
```

10. Click **OK**.

Your text label will now look like a button.



11. Click the **Plot Data** button on your worksheet.
12. The Plot_Data function in your Test.c file is called, and a scatter plot is created.

Notes: The script behind the button assumes that you have data in column B(Y) of the Data1 worksheet and that there is an associated X data set.

Loading and Compiling your Origin C Function from Script

In this tutorial we learned how to save an Origin C function along with the project file and then access the function from the Origin interface. Saving an Origin C file with a project limits access to functions within that file to only that project. When a new project is opened, the functions are not available any more.

To access functions in an Origin C file that is saved on disk, the file can be programmatically loaded and compiled using LabTalk script. The script command for performing the programmatic load and compile is `run.LoadOC`. Refer to the **LabTalk** Help files (**Help: Programming: Labtalk**) for more information on using this command.

This concludes the **Origin C Functions** tutorial.

5.2.5 Accessing Internal Origin Objects by Origin C

Summary

Internal Origin objects (such as Project Explorer folders, Origin windows (pages), layers, plots, graphic objects, data sets, etc.) are accessed using Origin C classes. To access or programmatically control an internal Origin object, you must attach it to an Origin C object.

To attach something to an internal Origin object you must first "find" it using the properties, methods, and collections of a container class. Common container classes include the Project, Folder, Page, GraphPage, Layer, GraphLayer, Worksheet, MatrixLayer, and Collection classes. Once found, an internal Origin object can easily be attached to an Origin C object of the appropriate type.

The internal Origin object is then programmatically controlled by manipulating the class methods and properties of the attached Origin C object. The objective of this tutorial is to demonstrate how to find particular internal Origin objects, attach things to those objects, and access the objects by manipulating the methods and properties of the attached Origin C objects.

Minimum Origin Version Required: Origin 8.1 SR1

What you will learn





This tutorial will show you how to:

- Access Worksheet Related Objects
- Access Graph Related Objects

Accessing Worksheet Related Objects


Familiarity with the Origin C Project class (Project.h), the Collection class (Collection.h), and the Folder class (Folder.h), is valuable when attempting to understand how to find particular internal Origin Objects. Users may find it helpful to preview these classes in the **Origin C Reference: Classes** book of **Origin C Help** or in the above header files located in the ..\Origin\OriginC\system subfolder.

To begin this tutorial:

1. On the **Standard** toolbar, click the **New Project** button .
2. On the **Standard** toolbar, click the **Code Builder** button .
3. On the **Code Builder** menu, select **File: New Workspace**.
4. On the **Code Builder** menu, click the **Open** button .
5. Browse to the \Samples\Origin C Examples\Programming Guide\Introduction to Accessing Origin Objects folder in the Origin software directory, select AccessWorksheetObjectsTutorial.c, check the **Add to Workspace** check box and click **Open**.
6. On the **Code Builder** toolbar, click the **Rebuild All** button . This compiles and links the file.
7. On the Code Builder **View** menu, verify that the **LabTalk Console** (Command & Results) and the **Local Variables** windows are visible (the corresponding menu items should be checked).
8. From the **Code Builder** menu, select **Tools: Customize**. Select the Toolbars tab and make sure that the **Debug** toolbar check box is selected.
9. In Code Builder, activate AccessWorksheetObjectsTutorial.c.
10. Near the top of the file, locate and click on the line:

```
PageBase pb;
```

You can position the cursor anywhere on the line.

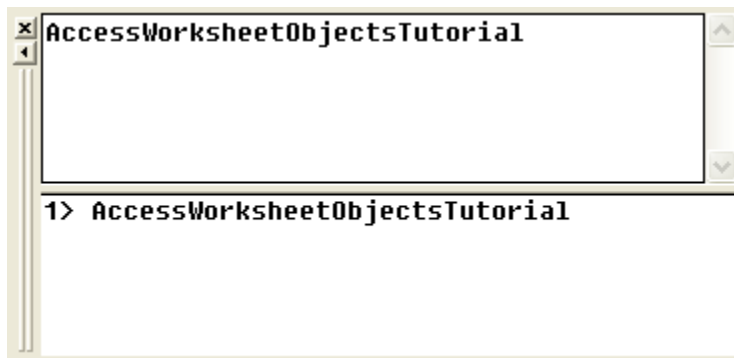
- From the Code Builder menu, select **Debug: Toggle Breakpoints**. Alternately, press **F9** or click the **Toggle Breakpoint** button  on the **Debug** toolbar.



A brown circle is displayed in the gray margin to the left of the above line indicating that a Debug breakpoint has been set for that line.

- In the Code Builder workspace, activate the LabTalk Console (Command & Results window) and type in the following:

```
AccessWorksheetObjectsTutorial
```

- Press **ENTER** to execute the function.



- On the **Debug** toolbar, press the **Step Into** button .
- Press the **Step Into** button  repeatedly, stopping to read the comments for each statement. Periodically stop and resize and/or reposition the Local Variables window to view the current run-time value of each variable.


```

#include <Origin.h> // Includes most Origin C header files (including HAG)

/**
 * Demonstrate accessing worksheet related internal Origin Objects. This function assumes
 * the active window is a worksheet having two columns A and B.
 * RETURN:
 * Returns TRUE on successful exit and FALSE on error.
 */
BOOL AccessWorksheetObjectsTutorial()
(
    PageBase      pb; // Declare PageBase object
    WorksheetPage wp; // Declare WorksheetPage object
    Worksheet     wks; // Declare Worksheet (Layer) object
    Column        col; // Declare Column object
    DataRange     dr; // Declare DataRange object

    string  strWindowName, strColName;
    int     iPageType, iColNum;

    // Note: The Project object is an instance of the Project class that is automatically available
    // to all Origin C functions

    // *** Use Project class Pages collection to get active Page (must use PageBase not Page or ***
    // *** WorksheetPage - don't assume type of active window) ***
    pb = Project.Pages(); // Use no argument to get active PageBase
    if( !pb.IsValid() ) // If not found (not valid) return error
        return FALSE;

```




Name	Value
pb	PageBase[Invalid Obj...
wks	Worksheet[Invalid Ob...
wp	WorksheetPage[Inval...

AccessWorksheetObjectsTutorial

1> AccessWorksheetObjectsTutorial


Ln 22, Col 1 NUM

Accessing Graph Related Objects

- Return to the *Origin* workspace and, on the **Standard** toolbar, click the **Open** button .
- Browse to the \Samples\Origin C Examples\Programming Guide\Introduction to Accessing Origin Objects subfolder, select AccessGraphObjectsTutorial.OPJ, and click **Open**. You may be prompted to save changes to an untitled project. Click **No** and a worksheet and graph should open.
- From the Code Builder menu, select **File: New Workspace**. Click **No** when prompted to save workspace changes.
- In Code Builder, click the **Open** button .
- Browse to the \Samples\Origin C Examples\Programming Guide\Introduction to Accessing Origin Objects subfolder, select AccessGraphObjectsTutorial.c, select the **Add to Workspace** check box, and click **Open**.
- Click the **Rebuild All** button  to compile and link the file.
- On the Code Builder **View** menu, verify that the **LabTalk Console** (Command & Results) and the **Local Variables** windows are visible (the corresponding menu items should be checked).
- From the Code Builder menu, select **Tools: Customize**. Select the Toolbars tab and make sure that the **Debug** toolbar check box is selected.
- In the Code Builder workspace, activate AccessGraphObjectsTutorial.c.
- Near the top of the file locate and click on the line:

```
GraphPage gp;
```



You can position the cursor anywhere on the line.

11. From the Code Builder menu, select **Debug: Toggle Breakpoints**. Alternately, press **F9** or click the **Toggle Breakpoint** button  on the **Debug** toolbar.

A brown circle is displayed in the gray margin to the left of the above line indicating that a Debug breakpoint has been set for that line.

12. Activate the LabTalk Console (Command & Results window) in Code Builder and type the following:

```
AccessGraphObjectsTutorial
```

13. Press **ENTER** to execute the function.
14. On the **Debug** toolbar, press the **Step Into** button .
15. Press the **Step Into** button  repeatedly, stopping to read the comments for each statement. Periodically stop and resize and/or reposition the Local Variables window to view the current run-time value of each variable.

This concludes the **Internal Origin Objects** tutorial.

5.3 X-Functions

Topics covered in this section:

1. Command Window
2. Introduction to X-Functions
3. How to Create a Wizard

5.3.1 Command Window and X-Functions

Summary

Many of Origin's analysis tools and other data processing tools have been implemented using X-Functions. The Command Window provides a convenient way to run these functions.

Another important use for the Command Window is to send LabTalk script commands to Origin. Script commands can range from simple math and data operations, to user-created X-Functions or Origin C functions.

Minimum Origin Version Required: Origin 8.0 SR6

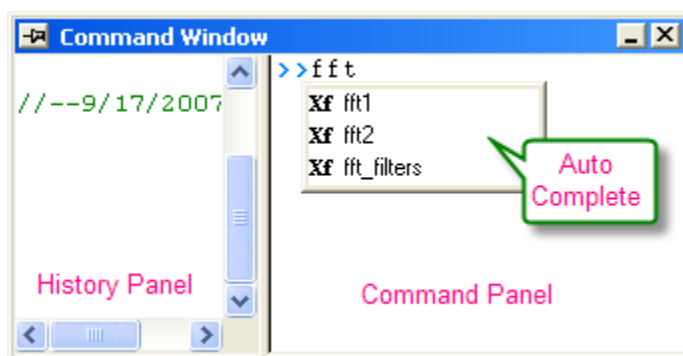
What you will learn

This tutorial will introduce you to the Command Window and show you how to:

- Perform simple calculation
- Access worksheet cells/columns
- Access X-Functions

Command Window

The Command Window consists of two panels: the **Command Panel** and **History Panel**:



The Command Window is normally located at the bottom right corner of the screen, but if it is not visible, you can access it by pressing **Alt + 3** or by selecting **View: Command Window**.

When typing in the Command Panel, the Auto Complete support allows you to choose among X-Function script commands and OGS files in the current working folder. The command and OGS file name will be respectively preceded by Xf and LT. You can move up and down the list using the arrow

keys; pressing Enter selects the item. After your selection, press the space bar and the Auto Complete now shows you the available options for the command.

Examples

Perform Calculations

The Command Window can be used as a calculator or to access any of Origin's mathematical functions. See the examples below.

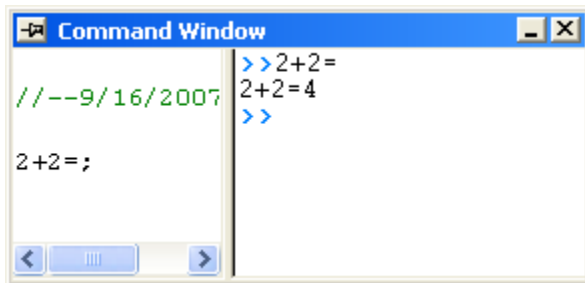
Single Line

One of the more basic uses for the Command Window is as an interface to perform simple calculations. For example, type the following:

```
2+2=
```

Press **ENTER**. Origin returns

```
2+2=4
```



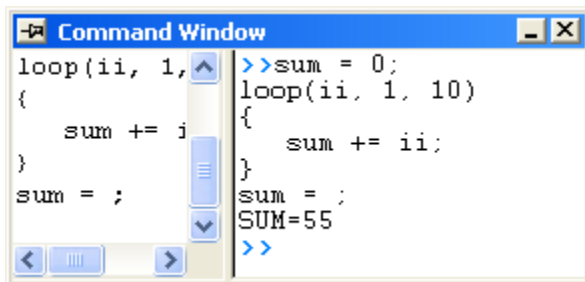
Multiple Lines

If you are typing multiple lines of scripts, first edit it in Code Builder (**View:Code Builder**) or any text editor, such as Windows Notepad, ending each line with a semi-colon, and then Copy + Paste the script in the Command Window, and press **ENTER** to execute. For example, paste the following script in the Command Window and **ENTER**:

```
sum = 0;
loop(ii, 1, 10)
{
    sum += ii;
}
sum = ;
```

Origin returns:

```
SUM=55
```

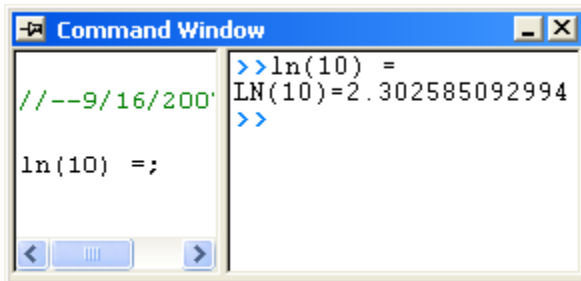


Functions

Any mathematical function, built-in as well as user-created, can be executed from the Command Window. For example, type:

```
ln(10) =
```

Origin returns natural logarithm value of 10.



Access Worksheet Values

You can also use the Script Window to read and write worksheet values, or to perform math operations on datasets.

1. Enter the following data into a fresh worksheet:

The screenshot shows a window titled "Book1" with a blue title bar. It displays a worksheet with a table. The table has three columns: the first column is unlabeled, the second is labeled "A(X)", and the third is labeled "B(Y)". The rows are labeled "Long Name", "Units", "Comments", and then numbered 1 through 5. The data in the numbered rows is as follows:

	A(X)	B(Y)
Long Name		
Units		
Comments		
1	1	6
2	2	7
3	3	8
4	4	9
5	5	10

At the bottom of the window, there are navigation buttons: a left arrow, a button with three vertical bars, and a right arrow, along with a tab labeled "Sheet1".

2. To return the value in the first cell of the second column, type the following:

```
cell(1,2)=
```

3. Press **ENTER**. Origin returns:

```
CELL(1,2)=6
```

You can also use the column name and row number to reference cell values.

4. Type the following:

```
col(B)[1]=
```

5. Press **ENTER**. Origin returns:

```
COL(B)[1]=6
```

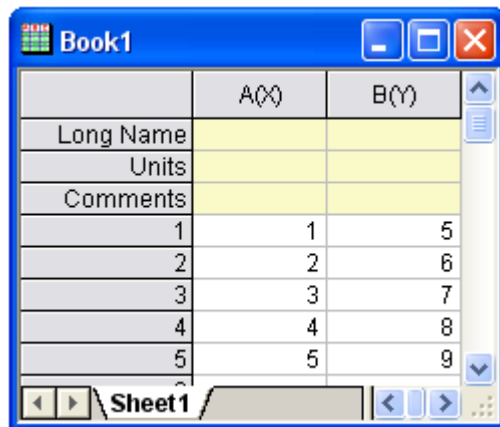
Notes: In addition to using the column name, you can also use the dataset name. In LabTalk, the syntax for naming datasets is *worksheetName_columnName*. So, For example, *Book1_A[1]=* would return the first element of column A in worksheet Data1. Also, if the worksheet that you are referencing is the active window, you can use the LabTalk string variable %H, in place of the worksheet name. For example, %H_A[1].

To subtract the value in row 1 of column A, from all the values in column B?

6. Type the following:

```
col(B)=col(B)-col(A)[1]
```

7. Press **ENTER**. Your worksheet now reads:



	A()	B()
Long Name		
Units		
Comments		
1	1	5
2	2	6
3	3	7
4	4	8
5	5	9

Let's use what we have learned about executing multiple lines of script in the Script window. We will multiply every value in a column of data by some constant b.

8. Type the following:

```
b=3;
```

Press **CTRL+ENTER**. Recall that this gives us a carriage return without executing the command.

9. Now type:

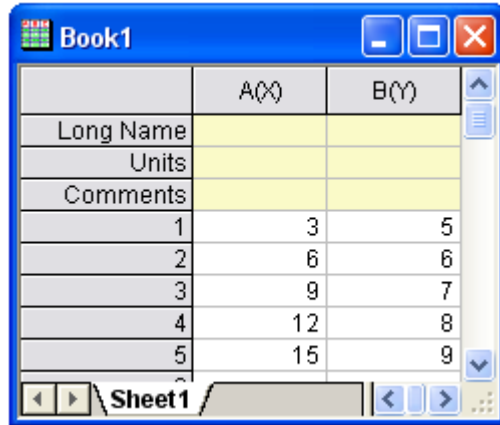
```
col(A)=col(A)*b;
```

Again, Press **CTRL+ENTER**.

10. Choose **Edit** from the Script window's menu bar. **Script Execution** should have a check mark next to it; if not, single-click on the menu item to place a check mark there.

11. Now, select the two lines of script that you just entered into the Script window and press **ENTER**.

Your worksheet now reads:



	A(X)	B(Y)
Long Name		
Units		
Comments		
1	3	5
2	6	6
3	9	7
4	12	8
5	15	9

Notes: The following C notation is also supported:

```
b=3;
col(A)*=b;
```

You can also use linear interpolation or extrapolation on a specified *X dataset* to find the corresponding interpolated or extrapolated value in a *Y dataset*. This requires using a new notation with parentheses () instead of brackets [].

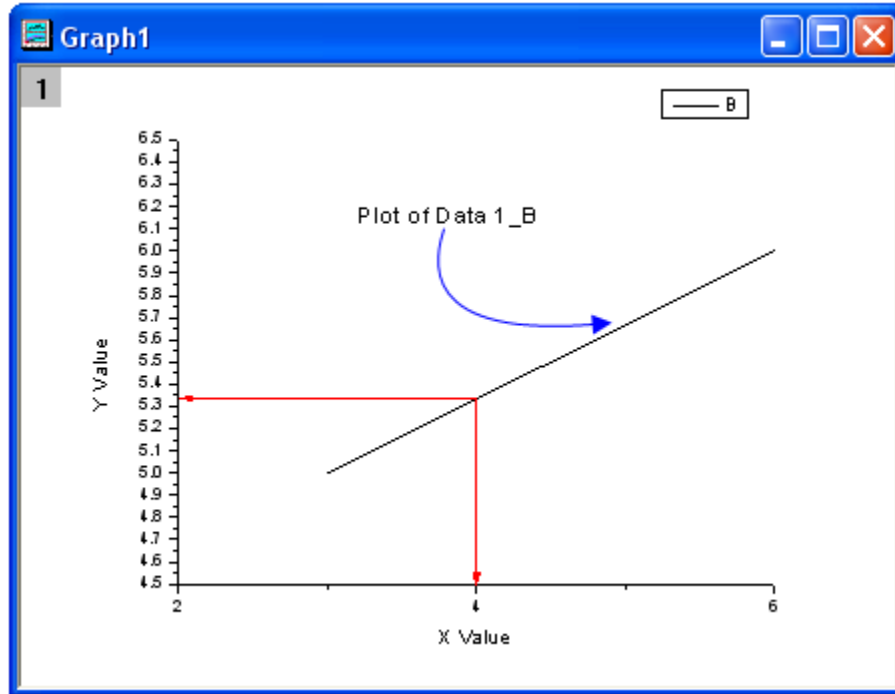
In this example, *book1_b* is a *Y dataset* and (4) is a value in an *X dataset* (*book1_a*) for which you want to find a corresponding, interpolated *Y* value.

12. Type the following:

```
book1_b(4) =
```

13. Press **ENTER**. Origin returns:

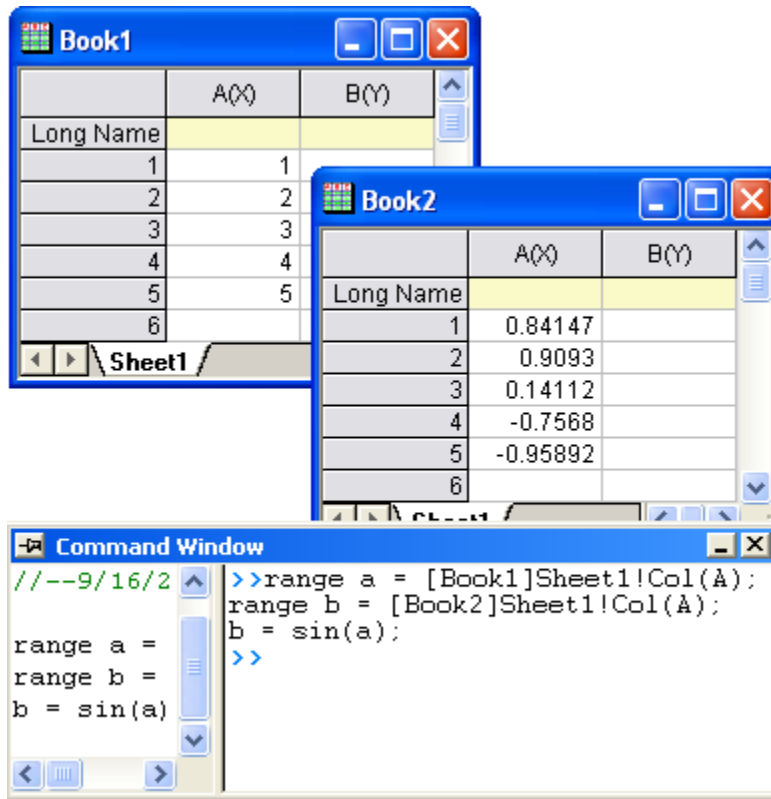
```
BOOK1_B(4)=5.333333
```



This is a line plot of our simple worksheet data. You can see that our interpolated Y value ? the one corresponding to X = 4 ? is 5.333333.

14. If the columns you work with are in different worksheet/workbook, you should use the range variables to represent the worksheet columns. For example, this script calculates the sine value on Book1 column A, and puts the result in Book2 column A (You must have Book2 before hitting Enter):

```
range a = [Book1]Sheet1!Col(A);
range b = [Book2]Sheet1!Col(A);
b = sin(a);
```

Access X-Functions

Origin 8 provides a large collection of X-Functions for performing a wide variety of data processing tasks. Of this collection, many of the X-Functions are accessible from LabTalk script. The functions accessible from script provide a powerful environment for users to create custom script code for their routine tasks.

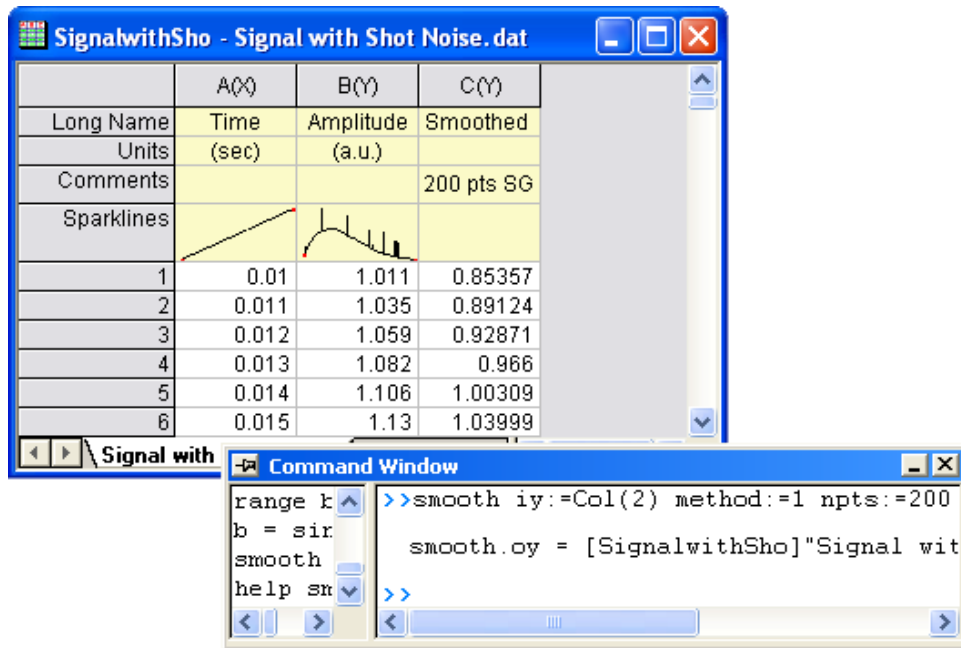
X-Functions that are accessible from script can be listed in the Command Window, and you can also obtain help on the command syntax as well as make use of auto completion of commands for such functions.

X-Functions accept data range string or range variable for specifying source and destination data for the operation. For example, the smooth X-Function under signal processing can be accessed from the Command Window as follows:

1. Import the file `\Samples\Single Processing\Signal with Shot Noise.dat`.
2. In the Command Window, type the following:

```
smooth iy:=Col(2) method:=1 npts:=200
```

3. When you press **ENTER**, the result will append to the source worksheet.



4. For help in using this smooth X-Function, you can type

help smooth

5. to open the corresponding Help.

5.3.2 Introduction to X-Functions

Summary

X-Functions provide a structured programming environment that offers a framework for building Origin tools. Different from the simple GetN box, creating tools by using X-Functions allows the user to focus on the actual data processing code and not have to worry about codes for the user interface.

Most of the dialogs/functions in Origin 8 are X-Functions, and many of them can be run from both menu and command line mode. The flexibility of running X-Functions makes them an attractive approach to customizing Origin

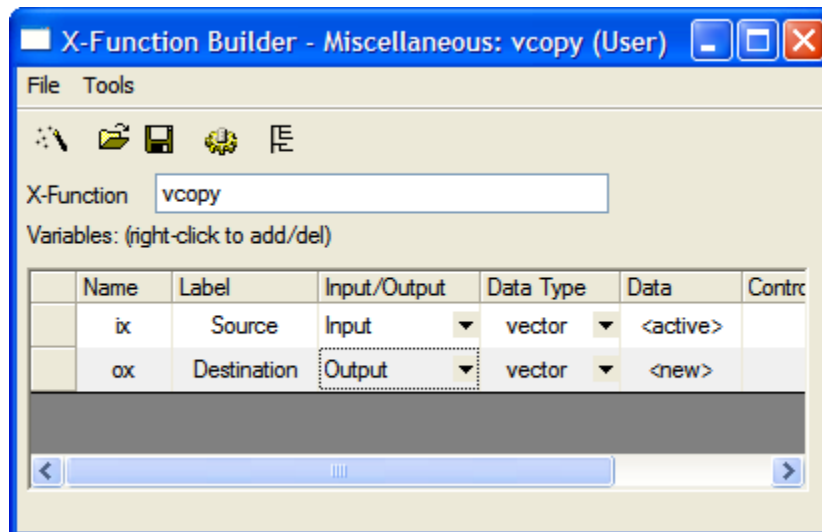
Minimum Origin Version Required: Origin 8.0 SR0


What you will learn

- How to create an X-Function
- How to make the X-Function script accessible
- How to use the X-Function in dialog mode

Create an X-Function

1. Select **Tools: X-Function Builder** or press **F10** to open the **X-Function Builder** dialog
2. Set **Name**, **Label** and **Data** of the 1st variable as **ix**, **Source** and **<active>**
3. Right click in the list panel and select **Add Variables** from the context menu.
4. Set **Name**, **Label**, **Input/Output** and **Data** of the 2nd variable as **ox**, **Destination**, **Output** and **<new>**
5. Select **File:Save** to save the x-function as vcopy



6. Click  to open **Code Builder**
7. Add the following codes in the vcopy function in code builder



```

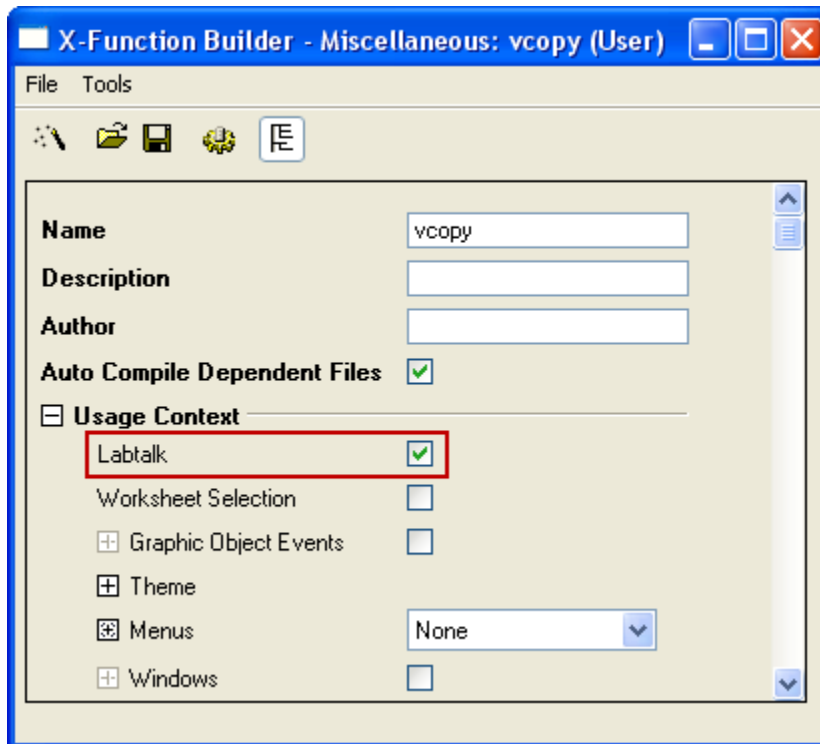
void vcopy(const vector& ix, vector& ox)
{
    if (!ix || !ox)
        XF_THROW(CER_NO_DATA);

    ox = ix;
}

```

Making the X-Function Script Accessible

1. Click the **Return to Dialog** button in **Code Builder**
2. In the **X-Function Builder**, save your changes 
3. Open the X-Function in **Tree View** by clicking 
4. Open the **Usage Context** branch. Make sure the **Labtalk** check box is selected

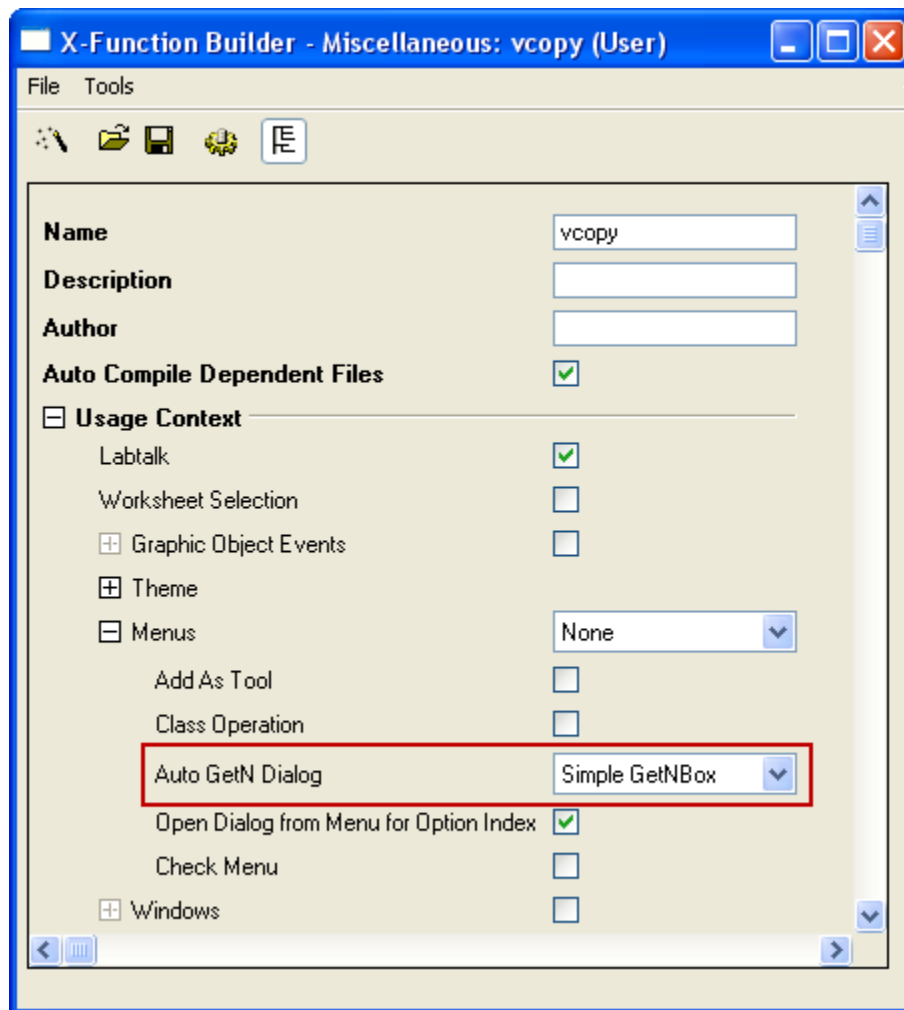


5. Save the x-function and close the **X-Function Dialog**
6. Fill column(A) with row numbers in the active worksheet (Highlight column(A), right-click and select **Fill Column with: Row Numbers**)
7. Type the following script in the command window, Column(A) will be copied to Column(B)
`vcopy col(a) col(b)`

Using the X-Function in Dialog Mode

1. Open the **X-Function Dialog** and open VCOPY.OXF in **Tree View**

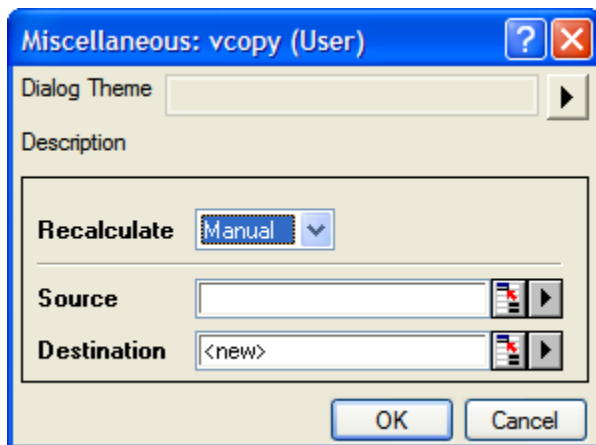
2. Open the **Usage Context** branch
3. Open the **Menus** branch, make sure **Simple GetNBox** is selected from the **Auto GetN Dialog** list box



4. Save the x-function and close the **X-Function Dialog**

5. Type following script in the command window, Dialog of VCOPY.OXF will be opened

`vcopy -d`



5.3.3 How to Create a Wizard

Summary

A wizard is a graphical user interface that includes a series of dialogs to direct a user to complete a mission step by step. A wizard makes a complex task easier to perform. Origin provides several classes in Origin C for users to develop a wizard. The dialog for each step in the wizard can be created using an X-Function.

In this example, the wizard will perform a normality test and then a one-sample t-test for data in a column. The normality test's result can be shared in one-sample t-test.

Note: This tutorial requires the Develop Kit.

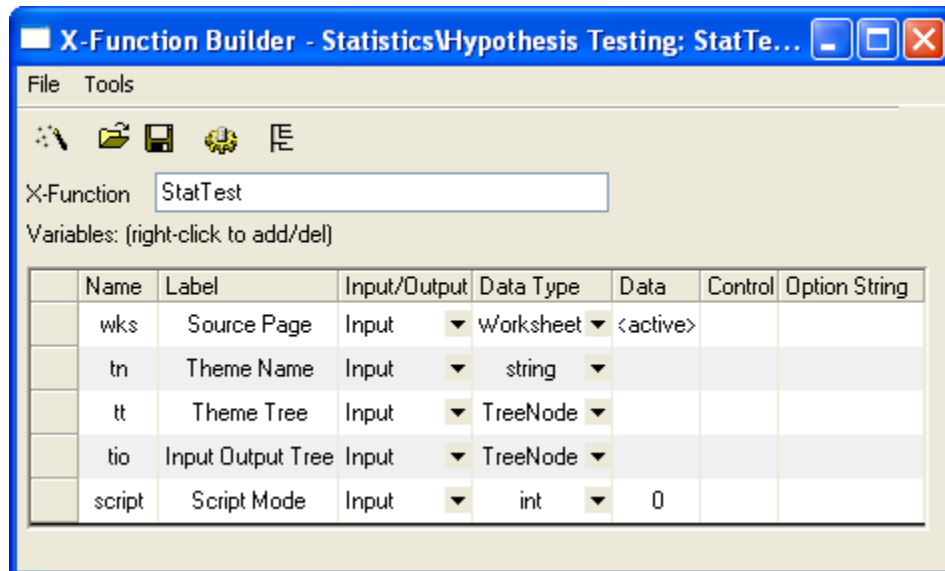
Minimum Origin Version Required: Origin 8.1SR0

What you will learn

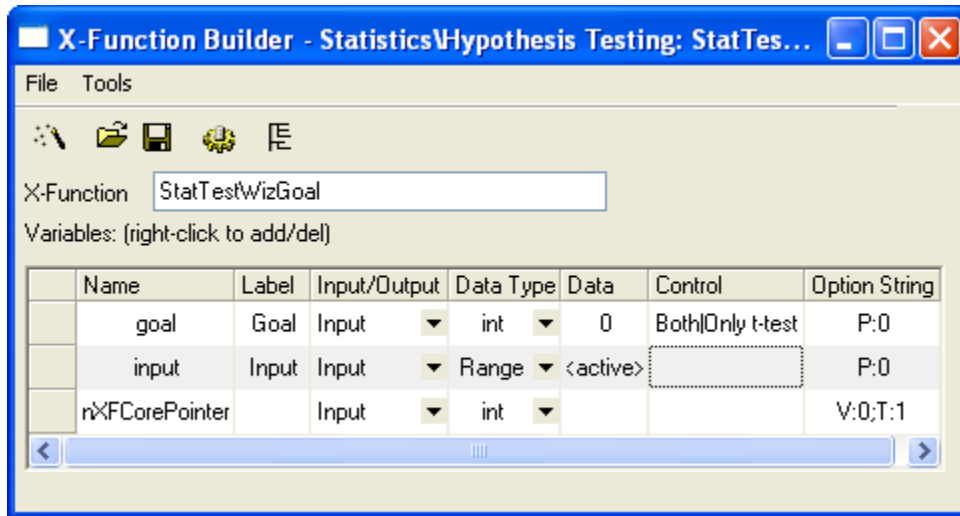
- How to create an X-Function.
- How to share a variable or a DataRange in different steps.
- How to call an X-Function in OriginC.
- How to create a wizard.

Create four X-Functions

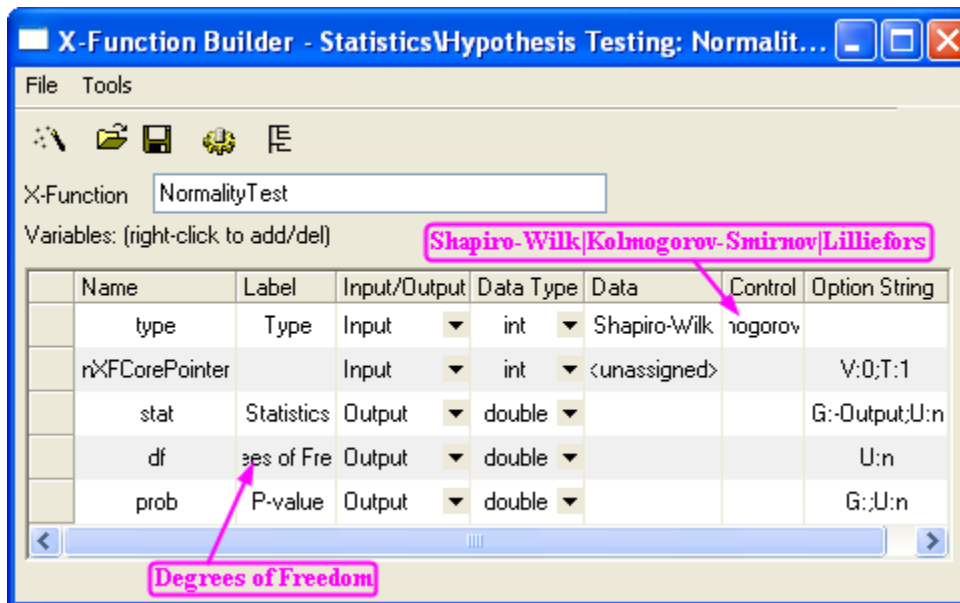
1. Select **Tools: X-Function Builder** or press **F10** to open **X-Function Builder dialog**
2. Add the variables as follows and save the X-Function as "StatTest" in the User Files folder, User Files\X-Functions\Statistics\Hypothesis Testing.



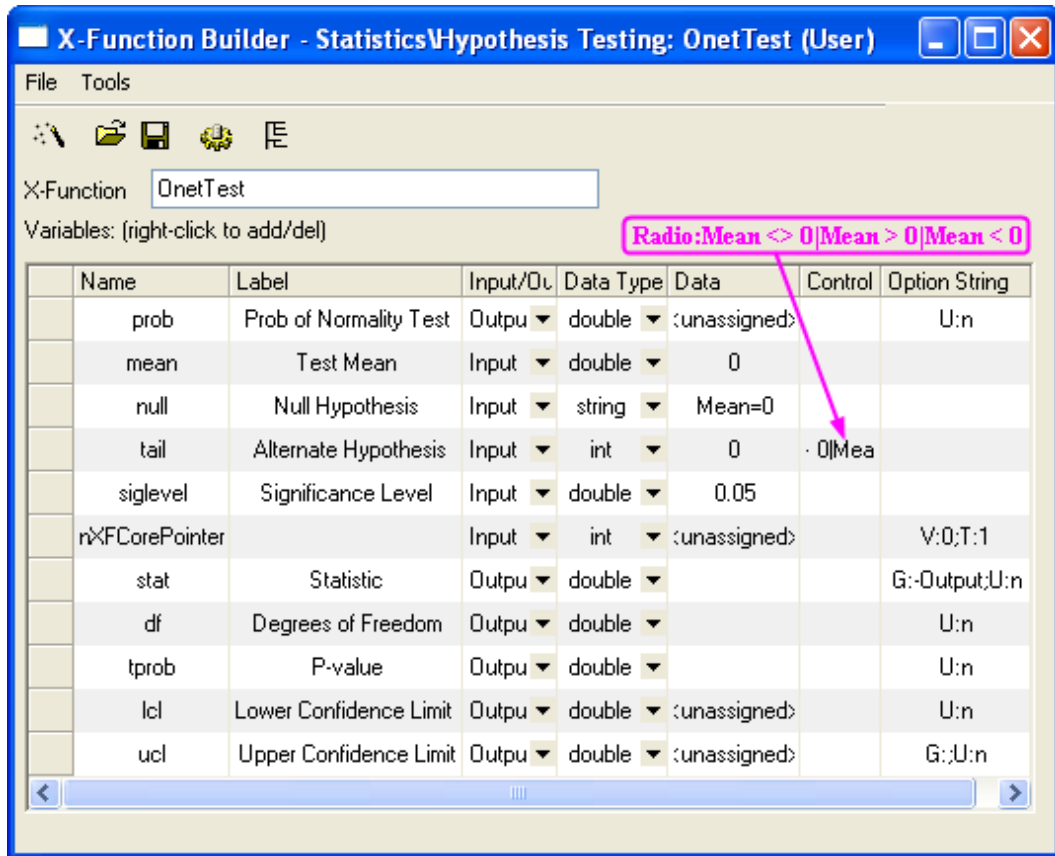
- Click the **New X-Function Wizard** button. Add the variables as follows and save the X-Function as "StatTestWizGoal" in the User Files folder, User Files\X-Functions\Statistics\Hypothesis Testing.



- Click the **New X-Function Wizard** button. Add the variables as follows and save the X-Function as "NormalityTest" in the User Files folder, User Files\X-Functions\Statistics\Hypothesis Testing.



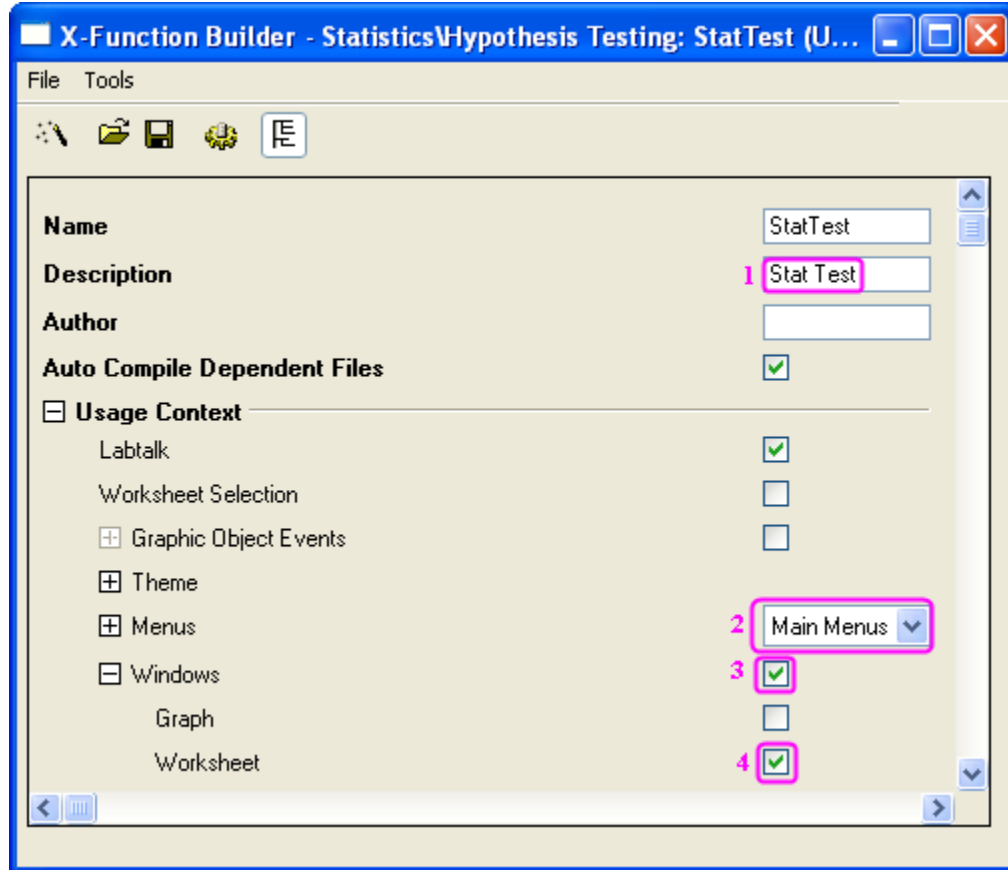
- Click the **New X-Function Wizard** button. Add the variables as follows and save the X-Function as "OnetTest" in the User Files folder, User Files\X-Functions\Statistics\Hypothesis Testing.




Note that the X-Functions **NormalityTest** and **OnetTest** have the same variable "prob", which is a shared variable and will be declared in the source file.

Update X-Function's Property in Tree View

1. Open the X-Function **StatTest**. Click the **TreeView** button  to open the Tree View. Make the following settings in the Tree View.



2. Click the **Save OXF file** button to save the X-Function.
3. Open the X-Function **StatTestWizGoal**, **NormalityTest** and **OnetTest** respectively in X-Function Builder. Click the **TreeView** button  and type "Select Wizard Goal", "Normality Test" and "One-Sample t-test" in the **Description** edit box of each X-Function's Tree View, which will be shown in the dialogs.

Create Files for the Wizard

- Click the **Code Builder** button on the **Standard** toolbar. In Code Builder, click the **New** button. In the **New File** dialog, select **H File**, click the **Browse** button, and select the User Files folder, User Files\OriginC as the new header file's **Location**. Then type StatTestWiz in the **File Name** edit box. Click **OK** to close the dialog.

Add the following script to StatTestWiz.h file.

```
#ifndef __STAT_TEST_WIZ_H__
#define __STAT_TEST_WIZ_H__
```

```

#include <..\OriginLab\XFWiz.h>
#include <..\OriginLab\XFCore.h>
#include <..\OriginLab\XFWizard_utils.h>

class StatTestWizCore : public XFCore
{
public:
    StatTestWizCore();

public:
    void ChangeGoal(int nGoal);
    DataRange GetRange();

    int nStep;
protected:
};

int stat_test_run_wiz_nodlg(LPCSTR lpkszThemeName = NULL,
const XFWizTheme *pXFWizTheme
= NULL, const XFWizInputOutputRange *pXFWizIO = NULL, DWORD
dwOPUID = 0);

int stat_test_open_wiz_dlg(LPCSTR lpkszThemeName = NULL, const
XFWizTheme *pXFWizTheme
= NULL, const XFWizInputOutputRange *pXFWizIO = NULL, DWORD
dwOPUID = 0);

#endif    //__STAT_TEST_WIZ_H__

```

Click the **Save** button to save StatTestWiz.h file.

- Repeat the same operation to create a new **C File**, StatTestWiz.c .

Add the following script to StatTestWiz.c file.

```

////////////////////////////////////
////////////////////////////////////
#include <..\OriginLab\XFWizManager.h>

#include <..\OriginLab\WizOperation.h>
#include <..\OriginLab\XFWizNavigation.h>

#include <..\OriginLab\XFWizScript.h>
#include <..\OriginLab\XFWizDlg.h>

////////////////////////////////////
////////////////////////////////////
// Include your own header files here.
#include "StatTestWiz.h"

enum
{
    GOAL_ALL = 0,
    GOAL_SIMPLE,
};

```

```

//Names of three X-Functions
#define STR_STEP_GOAL      "StatTestWizGoal"
#define STR_STEP_Normal   "NormalityTest"
#define STR_STEP_TTest    "OnetTest"

//Names of steps shown in the wizard.
#define STR_LABEL_STEP_GOAL      "Goal"
#define STR_LABEL_STEP_Normal    "Normality Test"
#define STR_LABEL_STEP_TTest     "One-Sample t-test"

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
//Class StatTestWizTheme
class StatTestWizTheme : public XFWizTheme
{
public:
    StatTestWizTheme();
};

//Name of the variable prob shared by X-Functions
NormalityTest and OnetTest
#define STR_GETN_VAR_SHARED_NProb "prob"

StatTestWizTheme::StatTestWizTheme()
: XFWizTheme()
{
    m_saSharedList.Add(STR_GETN_VAR_SHARED_NProb); //Add the
shared variable
}

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
class StatTestWizInputOutputRange : public
XFWizInputOutputRange
{
};

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
//Class StatTestWizManager

#define STR_CLASS_NAME_TEST      "StatTestWiz"
#define TEST_VERSION_NUMBER      1.0

class StatTestWizManager : public XFWizManager
{
public:
    StatTestWizManager(LPCSTR lpcszThemeName = NULL, const
XFWizTheme *pXFWizTheme
= NULL, const XFWizInputOutputRange *pXFWizIO = NULL, DWORD
dwUIDOp = 0);

protected:
    virtual double GetVersion() { return TEST_VERSION_NUMBER; }
    virtual XFCore* CreateXFCore() { return new
StatTestWizCore; }

```

```

    virtual XFWizTheme* CreateXFWizTheme() { return new
StatTestWizTheme; }
    virtual XFWizInputOutputRange*
CreateXFWizInputOutputRange()
        { return new StatTestWizInputOutputRange; }
    virtual string GetClassName() { return STR_CLASS_NAME_TEST;
}
};

```

```

StatTestWizManager::StatTestWizManager(LPCSTR lpCszThemeName,
const XFWizTheme
*pXFWizTheme, const XFWizInputOutputRange *pXFWizIO, DWORD
dwUIDOp)
: XFWizManager(lpCszThemeName, pXFWizTheme, pXFWizIO, dwUIDOp)
{
    StringArray saMapXFNames = {STR_STEP_GOAL, STR_STEP_Normal,
STR_STEP_TTest};
    StringArray saMapXFLabels = {STR_LABEL_STEP_GOAL,
STR_LABEL_STEP_Normal,
                                STR_LABEL_STEP_TTest};
    m_saMapXFNames = saMapXFNames;
    m_saMapXFLabels = saMapXFLabels;
    ASSERT( m_saMapXFNames.GetSize() ==
m_saMapXFLabels.GetSize() );

    StringArray saDefaultXFNames = {STR_STEP_GOAL,
STR_STEP_Normal, STR_STEP_TTest};
    m_saDefaultXFNames = saDefaultXFNames;

    m_strRunDlgName = _L("Stat Test");
}

```

```

////////////////////////////////////
////////////////////////////////////
//Class StatTestWizCore

```

```

StatTestWizCore::StatTestWizCore()
:XFCore()
{
    StringArray vsXFRecalculateShown = {STR_STEP_GOAL};
    m_vsXFRecalculateShown = vsXFRecalculateShown;
    nStep = GOAL_ALL;
}

```

```

//Select steps in the Goal Step
void StatTestWizCore::ChangeGoal(int nGoal)
{
    XFWizNavigation *pXFWizNavg = (XFWizNavigation
*)GetXFWizNavigation();
    ASSERT(pXFWizNavg);

    nStep = nGoal;

    if ( pXFWizNavg )
    {
        StringArray saXFNames;
        saXFNames.Add(STR_STEP_GOAL);
    }
}

```

```

        switch (nGoal)
        {
            case GOAL_ALL:
                saXFNames.Add(STR_STEP_Normal);
                saXFNames.Add(STR_STEP_TTest);
                break;
            case GOAL_SIMPLE:
                saXFNames.Add(STR_STEP_TTest);
                break;
        }

        pXFWizNavg->SetSteps(saXFNames);

    }
}

//Get input DataRange in the Goal Step.
DataRange StatTestWizCore::GetRange()
{
    XFWizNavigation *pXFWizNavg =
(XFWizNavigation*)GetXFWizNavigation();
    XFWizInputOutputRange* pIORange = pXFWizNavg-
>GetXFWizInputOutputRange();

    DataRange drInput;
    if(!pIORange)
    {
        error_report("Fail to get io ranges!");
        return drInput;
    }

    Array<DataRange&> drs;
    //Get input DataRange.
    if(!pIORange->Get(&drs, STR_STEP_GOAL, true))
    {
        error_report("Fail to get range from WizCore!");
        return drInput;
    }

    drInput = drs.GetAt(0);

    return drInput;
}

////////////////////////////////////
////////////////////////////////////

int stat_test_run_wiz_nodlg(LPCSTR lpcszThemeName, const
XFWizTheme *pXFWizTheme, const
XFWizInputOutputRange *pXFWizIO, DWORD dwOPUID)
{
    TEMPLATE_run_wiz_nodlg(StatTestWizManager, lpcszThemeName,
pXFWizTheme, pXFWizIO, dwOPUID)
}

int stat_test_open_wiz_dlg(LPCSTR lpcszThemeName, const
XFWizTheme *pXFWizTheme, const

```

```

XFWizInputOutputRange *pXFWizIO, DWORD dwOPUID)
{
    TEMPLATE_open_wiz_dlg(StatTestWizManager, lpkszThemeName,
pXFWizTheme, pXFWizIO, dwOPUID)
}

int stat_test_run_wiz(UINT msg, const XFWizTheme *pXFWizTheme,
const
XFWizInputOutputRange *pXFWizIO, DWORD dwOPUID, int nExeMode)
{
    TEMPLATE_run_wiz(StatTestWizManager, msg, pXFWizTheme,
pXFWizIO, dwOPUID, nExeMode)
}

```

Click the **Save** button to save StatTestWiz.c file.

Note that StatTestWiz.c should be compiled after the X-Function **StatTest** is compiled, since the included files in StatTestWiz.c are not yet in the workspace until the X-Function **StatTest** is compiled. In fact StatTestWiz.h is included in X-Function **StatTest**, so StatTestWiz.c will be compiled automatically when X-Function **StatTest** is compiled.

Add Script for X-Functions

Script for X-Function StatTest

In the **X-Function Builder**, click the **Open** button and open the X-Function StatTest. Click the **Edit X-Function in Code Builder** and add the following script.

- Include header files

```

#include <..\OriginLab\XFWiz.h>
#include <..\OriginLab\WizOperation.h>
#include <..\OriginLab\XFCore.h>
#include <..\OriginLab\XFWizNavigation.h>
#include <..\OriginLab\XFWizManager.h>
#include <..\OriginLab\XFWizScript.h>
#include <..\OriginLab\XFWizDlg.h>

#include <..\OriginLab\XFWizard_utils.h>

#include <..\OriginLab\WksOperation.h>

#include <event_utils.h>

#include "StatTestWiz.h"

```

- StatTest()

Add the function body, which specifies the dialog mode.

```

if( script )
    stat_test_run_wiz_nodlg(tn);
else
    stat_test_open_wiz_dlg(tn);

```

- StatTest_before_execute()

Add the function body, which determines not to show this dialog before the wizard is opened.

```
nRet = XFEVT_PROCEED_NO_DLG;
```

Click **Compile** button to compile the file. Then click **Return to Dialog** button to return to **X-Function Builder**. In the **X-Function Builder**, click **Save OXF file** button to save the X-Function.

Script for X-Function StatTestWizGoal

Open the X-Function StatTestWizGoal. Click **Edit X-Function in Code Builder** button, add the following script.

- Include header files

```
#include "StatTestWiz.h"
```

- Add a static function `_check_input()`

This function is used to check whether the input DataRange is a single column.

```
static bool _check_input(const TreeNode trGetN, string&
strErr)
{
    TreeNode trRange = trGetN.input;
    DataRange drInput;

    drInput.Create(trRange.strVal);

    if( drInput.GetNumRanges() == 0 )
    {
        strErr = "Input can't be empty, and it should be a valid
column.";
        return false;
    }
    else
    {
        if( drInput.GetNumRanges() == 1)
        {
            Worksheet wksInput;
            int nC1, nC2;
            drInput.GetRange(wksInput, nC1, nC2);
            if( nC1 == nC2 )
                return true;
        }

        strErr = "Please select one column.";
        return false;
    }
}
```

- StatTestWizGoal_event1()

Add the function body, which updates the dialog.

```
StatTestWizCore* pstatwc =
(StatTestWizCore*)get_xf_core_handler(trGetN);
```



```

ASSERT(pstatwc);

//Update the Wizard page.
if ( 0 == lstrcmp(lpkszNodeName, "goal" ) )
    pstatwc->ChangeGoal(trGetN.goal.nVal);

//Error message is shown at the bottom of the dialog,
//and OK button is disenabled for incorrect choice of
DataRange.
bOKEnable = _check_input(trGetN, strErrMsg);

return false;

```

Click **Compile** button to compile the file. Then click **Return to Dialog** button to return to **X-Function Builder**, and click **Save OXF file** button to save the X-Function.

Script for X-Function NormalityTest

Open the X-Function **NormalityTest**. Click the **Edit X-Function in Code Builder** button and add the following script.

- Include header files

```

#include "StatTestWiz.h"

#include <XFbase.h>

```

- Add a static function `_update_GUI()`

This function is used to update the dialog's edit boxes for normality test result.

```

static void _update_GUI(TreeNode& trGetN)
{
    vector vRes;
    vRes = _norm_test(trGetN.nXFCorePointer.nVal,
trGetN.type.nVal);

    trGetN.stat.dVal = vRes[0];
    trGetN.df.dVal = vRes[1];
    trGetN.prob.dVal = vRes[2];
}

```

- Add a static function `_update_strErr()`

This function is used to update the string shown at the bottom of the dialog.

```

static void _update_strErr(const TreeNode tr, string& strErr)
{
    if(tr.prob.dVal >= 0.05 && tr.prob.dVal <= 1)
        strErr = "At the 0.05 level, the data was significantly
drawn from a
normally distributed population.";
    else if(tr.prob.dVal < 0.05 && tr.prob.dVal >= 0)
        strErr = "At the 0.05 level, the data was not
significantly drawn from a

```

```

        normally distributed population.";
    else
        strErr = "There is not enough information to draw a
conclusion.";
}

```

Note that the string is divided into two lines shown in the page. It should be a command of one line in the script.

- Add a static function `_norm_test()`

This function is used to perform Normality Test using related X-Functions.

```

static vector _norm_test(const int nXFCorePointer, const int
nType)
{
    StatTestWizCore* pstatwc =
(StatTestWizCore*)get_xf_core_handler(nXFCorePointer);
    ASSERT(pstatwc);

    vector vRes(3);
    vRes[2] = -1;
    DataRange drInput;
    drInput = pstatwc->GetRange();
    if( !drInput )
        return vRes;

    vector<string> vsXFName = {"swtest", "kstest", "lillietest"};
    XFBase xfNorm(vsXFName[nType]);
    if( !xfNorm.SetArg("irng", drInput) )
    {
        error_report("Failed to set argument image type");
        return vRes;
    }
    if( !xfNorm.SetArg("stat", vRes[0]) )
    {
        error_report("Failed to set argument image type");
        return vRes;
    }
    if( !xfNorm.SetArg("df", vRes[1]) )
    {
        error_report("Failed to set argument image type");
        return vRes;
    }
    if( !xfNorm.SetArg("prob", vRes[2]) )
    {
        error_report("Failed to set argument image type");
        return vRes;
    }

    if( !xfNorm.Evaluate() )
    {
        error_report("Failed to evaluate the stats X-
Function.");
        return vRes;
    }
}

```

```

    return vRes;
}

```

- NormalityTest()

Update the function body, which exports the result into a worksheet when the **Next** button is pressed.

```

    DataRange drInput;
    StatTestWizCore* pstatwc =
(StatTestWizCore*)get_xf_core_handler(nXFCorePointer);
    ASSERT(pstatwc);
    drInput = pstatwc->GetRange();

    if( !drInput )
        return;
    string strBook, strSheet;
    if(!drInput.GetBookSheet(strBook, strSheet))
    {
        error_report("Workbook and worksheet names can't be
obtained.");
        return;
    }
    WorksheetPage wpData(strBook);

    int nLayer = wpData.AddLayer("Normality Test");

    if(nLayer >= 0)
    {
        Worksheet wksRes = wpData.Layers(nLayer);
        vector<string> vsTypeName = {"Shapiro-Wilk", "Kolmogorov-
Smirnov", "Lilliefors"};
        vector<string> vsNProb = {"Prob<W", "Prob>D", "Prob>D"};
        vector<string> vsParaName = {"Statistic", "DF", ""};
        vsParaName[2] = vsNProb[type];

        vector vRes;
        vRes = _norm_test(nXFCorePointer, type);

        wksRes.Columns(1).SetLongName(vsTypeName[type]);
        for(int ii=0; ii<3; ii++)
        {
            wksRes.SetCell(ii, 0, vsParaName[ii], false);
            wksRes.SetCell(ii, 1, vRes[ii]);
        }
    }
    else
    {
        error_report("New worksheet can't be created.");
    }
}

```

- NormalityTest_event1()

Update the function body, which will update the results in the dialog as the method of normality test changes. Strings shown at the bottom of the dialog will also be updated.

```

_update_GUI(trGetN);
_update_strErr(trGetN, strErrMsg);

return true;

```

- NormalityTest_before_execute()

Update the function body, which will make the edit boxes for results grayed out, and show the result in the dialog.

```

trGetN.stat.Enable = false;
trGetN.df.Enable = false;
trGetN.prob.Enable = false;

```

Click the **Compile** button to compile the file. Then click the **Return to Dialog** button to return to **X-Function Builder**, and click the **Save OXF file** button to save the X-Function.

Script for X-Function OnetTest

Open the X-Function **OnetTest**. Click the **Edit X-Function in Code Builder** button and add the following script.

- Include header files

```

#include "StatTestWiz.h"

#include <XFbase.h>

```

- Define strings

```

const vector<string> vsNull = {"Mean = ", "Mean <= ", "Mean >= "};
const vector<string> vsAlter = {"Mean <> ", "Mean > ", "Mean < "};
const vector<string> vsAcceptNull = {"Not significantly different from", "Not significantly greater than", "Not significantly less than"};
const vector<string> vsRejectNull = {"significantly different from", "significantly greater than", "significantly less than"};
const vector<string> vsProb = {"Prob>|t|", "Prob>t", "Prob<t"};

```

- Add a static function _update_null()

This function is used to update the **Null** edit box.

```

static void _update_null(TreeNode& trGetN, bool bMean = false)
{
    string strNull;

    strNull = vsNull[trGetN.tail.nVal] +
ftoa(trGetN.mean.dVal);
trGetN.null.strVal = strNull;

```

```

        if(bMean)
        {
            string strAlter = vsAlter[0] + ftoa(trGetN.mean.dVal) +
" | ";
            strAlter = strAlter + vsAlter[1] +
ftoa(trGetN.mean.dVal) + " | ";
            strAlter = strAlter + vsAlter[1] +
ftoa(trGetN.mean.dVal);

            trGetN.tail.SetAttribute(STR_COMBO_ATTRIB, strAlter);
        }
    }
}

```

- Add a static function `_check_sig_level()`

This function is used to check the **Significance Level** edit box value.

```

static bool _check_sig_level(TreeNode& trGetN, string& strErr)
{
    if( trGetN.siglevel.dVal > 0 && trGetN.siglevel.dVal < 1 )
    {
        return true;
    }
    else
    {
        strErr = "Significance Level should be between 0 and
1.";
        return false;
    }
}

```

- Add a static function `_update_strErr()`

This function is used to define the string for the conclusion of t-test at the bottom based on P-value.

```

static void _update_strErr(const TreeNode tr, string& strErr)
{
    if(tr.tprob.dVal >= tr.siglevel.dVal && tr.tprob.dVal <= 1)
        strErr.Format("Null Hypothesis is %s%s.\r\nAlternative
Hypothesis is %s%s.
        At the %s level, the population mean is %s the test
mean(%s).",
            vsNull[tr.tail.nVal], ftoa(tr.mean.dVal),
vsAlter[tr.tail.nVal], ftoa(tr.mean.dVal),
            ftoa(tr.siglevel.dVal), vsAcceptNull[tr.tail.nVal],
ftoa(tr.mean.dVal) );
        else if(tr.tprob.dVal < tr.siglevel.dVal && tr.tprob.dVal
>= 0)
            strErr.Format("Null Hypothesis is %s%s.\r\nAlternative
Hypothesis is %s%s.
            At the %s level, the population mean is %s the test
mean(%s).",
                vsNull[tr.tail.nVal], ftoa(tr.mean.dVal),
vsAlter[tr.tail.nVal], ftoa(tr.mean.dVal),
                ftoa(tr.siglevel.dVal), vsRejectNull[tr.tail.nVal],
ftoa(tr.mean.dVal) );
}

```

```

else
    strErr = "There is not enough information to draw a
conclusion.";
}

```

Note that the command is divided into several lines shown in the page. It should be a command of one line in the script.

- Add a static function `_update_GUI()`

This function is used to update edit boxes for results in the dialog.

```

static void _update_GUI(TreeNode& trGetN)
{
    vector vRes;
    vRes = _one_sample_t_test(trGetN.nXFCorePointer.nVal,
trGetN.mean.dVal, trGetN.tail.dVal, trGetN.siglevel.dVal);

    trGetN.stat.dVal = vRes[0];
    trGetN.df.dVal = vRes[1];
    trGetN.tprob.dVal = vRes[2];
    trGetN.lcl.dVal = vRes[4];
    trGetN.ucl.dVal = vRes[5];
}

```

- Add a static function `_one_sample_t_test()`

This function is used to perform One-Sample t-Test using an X-Function.

```

static vector _one_sample_t_test(const int nXFCorePointer,
const double dMean, const int nTail, const double dSiglevel)
{
    DataRange drInput;
    StatTestWizCore* pstatwc =
(StatTestWizCore*)get_xf_core_handler(nXFCorePointer);
    ASSERT(pstatwc);

    vector vRes(6);
    vRes[2] = -1;
    drInput = pstatwc->GetRange();
    if( !drInput )
        return vRes;

    vRes[3] = 100 - 100*dSiglevel;

    XFBase xfTTest("ttest1");

    if( !xfTTest.SetArg("irng", drInput) )
    {
        error_report("Failed to set argument irng");
        return vRes;
    }
    if( !xfTTest.SetArg("mean", dMean) )
    {
        error_report("Failed to set argument mean");
        return vRes;
    }
}

```

```

if( !xfTTest.SetArg("tail", nTail) )
{
    error_report("Failed to set argument tail");
    return vRes;
}
if( !xfTTest.SetArg("alpha", dSiglevel) )
{
    error_report("Failed to set argument alpha");
    return vRes;
}

if( !xfTTest.SetArg("stat", vRes[0]) )
{
    error_report("Failed to set argument stat");
    return vRes;
}
if( !xfTTest.SetArg("df", vRes[1]) )
{
    error_report("Failed to set argument df");
    return vRes;
}
if( !xfTTest.SetArg("prob", vRes[2]) )
{
    error_report("Failed to set argument prob");
    return vRes;
}
if( !xfTTest.SetArg("lcl", vRes[4]) )
{
    error_report("Failed to set argument lcl");
    return vRes;
}
if( !xfTTest.SetArg("ucl", vRes[5]) )
{
    error_report("Failed to set argument ucl");
    return vRes;
}

if( !xfTTest.Evaluate() )
{
    error_report("Failed to evaluate the ttest1 X-
Function.");
    return vRes;
}

return vRes;
}

```

- OnetTest()

Update the function body, which exports the result into a worksheet when the **Finish** button is pressed.

```

DataRange drInput;
StatTestWizCore* pstatwc =
(StatTestWizCore*)get_xf_core_handler(nXFCorePointer);
ASSERT(pstatwc);

```

```

drInput = pstatwc->GetRange();
if( !drInput )
    return ;

string strBook, strSheet;
if(!drInput.GetBookSheet(strBook, strSheet))
{
    error_report("Workbook and worksheet names can't be
obtained.");
    return;
}
WorksheetPage wpData(strBook);

int nLayer = wpData.AddLayer("One-Sample t-test");

if(nLayer >= 0)
{
    Worksheet wksRes = wpData.Layers(nLayer);

    vector<string> vsParaName = {"t Statistic", "DF", "",
"Conf. Levels in %", "Lower Limits", "Lower Limits"};
    vsParaName[2] = vsProb[tail];

    vector vRes;
    vRes = _one_sample_t_test(nXFCorePointer, mean, tail,
siglevel);

    wksRes.SetSize(-1, 4);
    wksRes.Columns(0).SetLongName("Test Statistics");
    string strNull = "Null Hypothesis is " + vsNull[tail] +
ftoa(mean);
    wksRes.Columns(1).SetLongName(strNull);
    wksRes.Columns(3).SetLongName("Confidence Intervals for
Mean");
    for(int ii=0; ii<3; ii++)
    {
        wksRes.SetCell(ii, 0, vsParaName[ii], false);
        wksRes.SetCell(ii, 1, vRes[ii]);

        wksRes.SetCell(ii, 2, vsParaName[ii + 3], false);
        wksRes.SetCell(ii, 3, vRes[ii + 3]);
    }
}
else
{
    error_report("New worksheet can't be created.");
}

```

- OnetTest_event1()

Update the function body, which will update results and show a conclusion at the bottom of the dialog according to the result. As settings change in the dialog, the **Null** edit box will be updated as the mean and hypothesis change, and the **Significance Level** edit box's value is checked.

```

if( 0 == lstrcmp(lpceszNodeName, "mean") )

```



```

    _update_null(trGetN, true);
    if( 0 == lstrcmp(lpVtblNodeName, "tail" )
        _update_null(trGetN);
    if( 0 == lstrcmp(lpVtblNodeName, "siglevel" )
        bOKEnable = _check_sig_level(trGetN, strErrMsg);

    _update_GUI(trGetN);
    _update_strErr(trGetN, strErrMsg);

    return false;

```

- OnetTest_before_execute()

Update the function body, to show/hide or disable the controls in the dialog.

```

    StatTestWizCore* pstatwc =
    (StatTestWizCore*)get_xf_core_handler(trGetN.nXFCorePointer.nV
al);
    ASSERT(pstatwc);
    trGetN.prob.Show = 1 - pstatwc->nStep;
    trGetN.prob.Enable = false;

    trGetN.null.Enable = false;
    trGetN.stat.Enable = false;
    trGetN.df.Enable = false;
    trGetN.tprob.Enable = false;
    trGetN.lcl.Enable = false;
    trGetN.ucl.Enable = false;

```

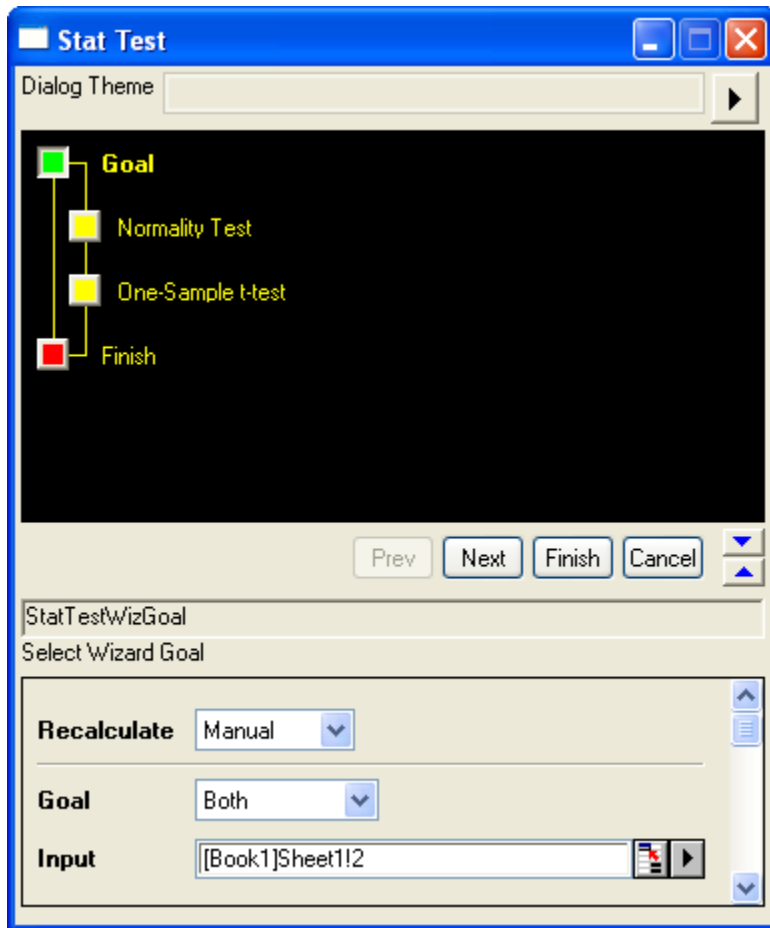
Click the **Compile** button to compile the file. Then click the **Return to Dialog** button to return to the X-Function Builder. Click the **Save OXF file** button to save the X-Function.

Close Origin. Then start Origin and you will notice that a new item **Stat Test** is added to Origin's menu **Statistics: Hypothesis Testing**.

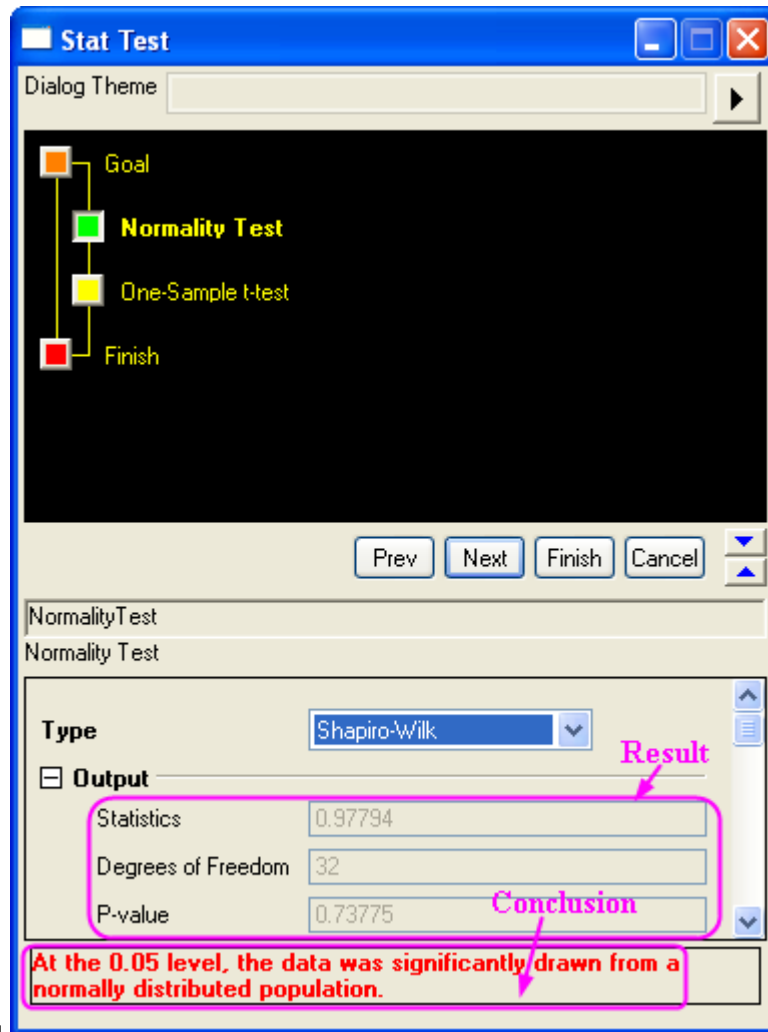
How to Use the Wizard

The following example shows how to use the wizard.

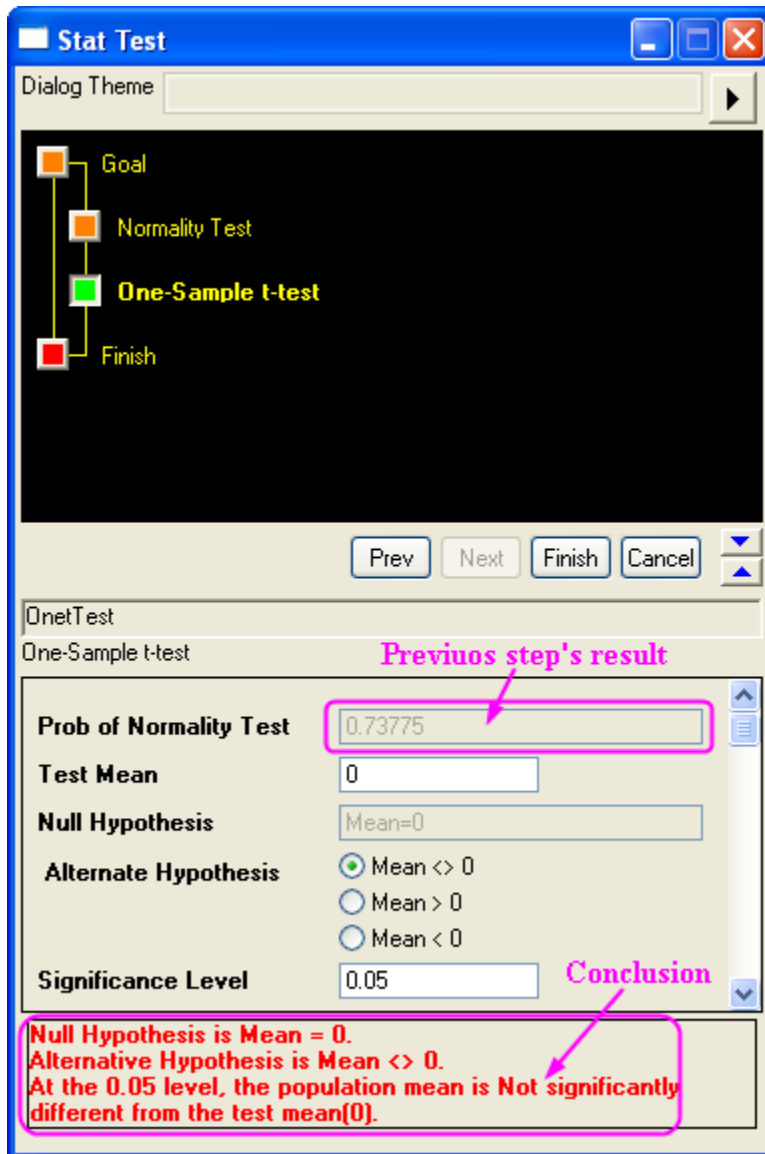
1. Select a column in the worksheet.
2. Select **Statistics: Hypothesis Testing: Stat Test** from the Origin menu or type the command "StatTest -d" in the command window. The **Stat Test** wizard dialog will open.



- Click the **Next** button. The **Normality Test** dialog is opened. The result is shown in the **Output** branch. A conclusion is drawn at the bottom of the



- dialog.
- Click the **Next** button. The **One-Sample t-test** dialog is opened. The result is shown in the **Output** branch. A conclusion is drawn at the bottom of the dialog. Previous step's result of normality test is shown at the top. You can also change the setting in the dialog, and notice the result changes.



Stat Test

Dialog Theme

Goal
Normality Test
One-Sample t-test
Finish

Prev Next Finish Cancel

OnetTest
One-Sample t-test

Output

Statistic	-1.50326
Degrees of Freedom	31
P-value	0.14289
Lower Confidence Limit	-0.6201
Upper Confidence Limit	0.09386

**Null Hypothesis is Mean = 0.
Alternative Hypothesis is Mean <> 0.
At the 0.05 level, the population mean is Not significantly different from the test mean(0).**

5. Click the **Finish** button to end the wizard. Two worksheets for results are created.

	A(X)	B(Y)
Long Name		Shapiro-Wil
Units		
Comments		
1	Statistic	0.97794
2	DF	32
3	Prob<W	0.73775

Sheet1 | Normality Test

	A(X)	B(Y)	C(Y)	D(Y)
Long Name	Test Statisti	Null Hypoth		Confidence
Units				
Comments				
1	t Statistic	-1.50326	Conf. Levels in %	95
2	DF	31	Lower Limits	-0.6201
3	Prob> t	0.14289	Lower Limits	0.09386

Normality Test | One-Sample t-test

6 Importing Data

6.1 ASCII

- Importing Overview
- Simple ASCII
- Import Wizard
- Import Time Data
- Post Processing with Import Filter

6.1.1 Importing Data

Summary

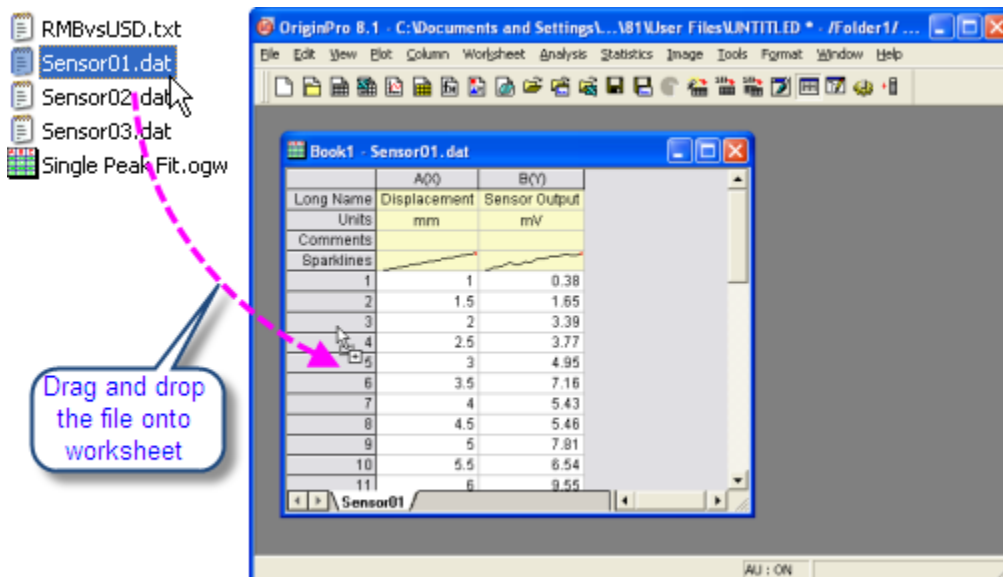
Origin provides flexible ways of importing data including simply dragging and dropping data files, using the ASCII import dialog to customize settings, using the Import Wizard for advanced customization and for extracting variables from header lines, and supporting custom file formats for many third-party files. This tutorial will highlight some of these features.

This tutorial will show you how to:

- Import files by drag-and-drop
- Import multiple ASCII files by customizing settings
- Saving settings for future use
- Import Wizard and import filters

Drag and Drop Importing of ASCII Files

1. Start with an empty worksheet. Open Windows Explorer to the `\Samples\Curve Fitting` subfolder of your Origin installation folder. Drag and drop the file `sensor01.dat` from the Windows Explorer window into the empty Origin worksheet.




2. Data comes into the sheet. Now select two files: **sensor02.dat** and **sensor03.dat** and drop into the same worksheet. You will see that the first file replaces the data already in the worksheet, and the others create new workbooks, as the default setting is to create new books for the 2nd file.

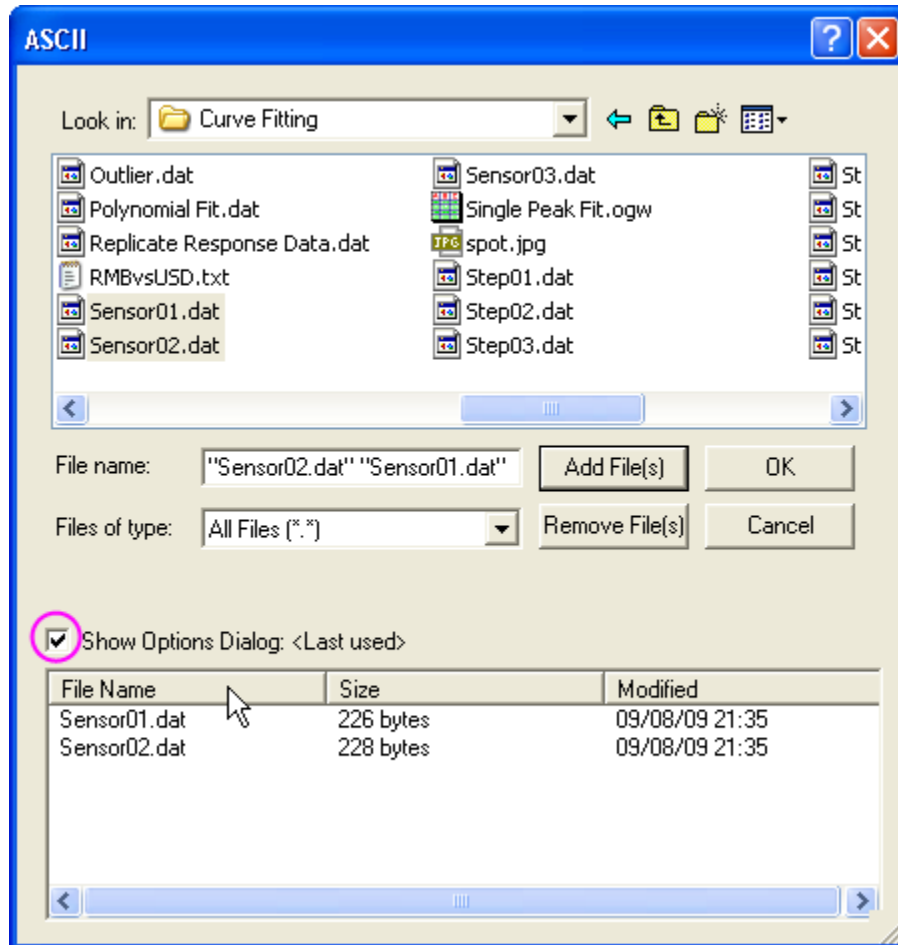


The default setting when dragging and dropping is to replace existing data. If you have some other data already in the sheet, you can drop the file onto the gray area outside of any window, or into a graph window, and Origin will create a new book and import the data.

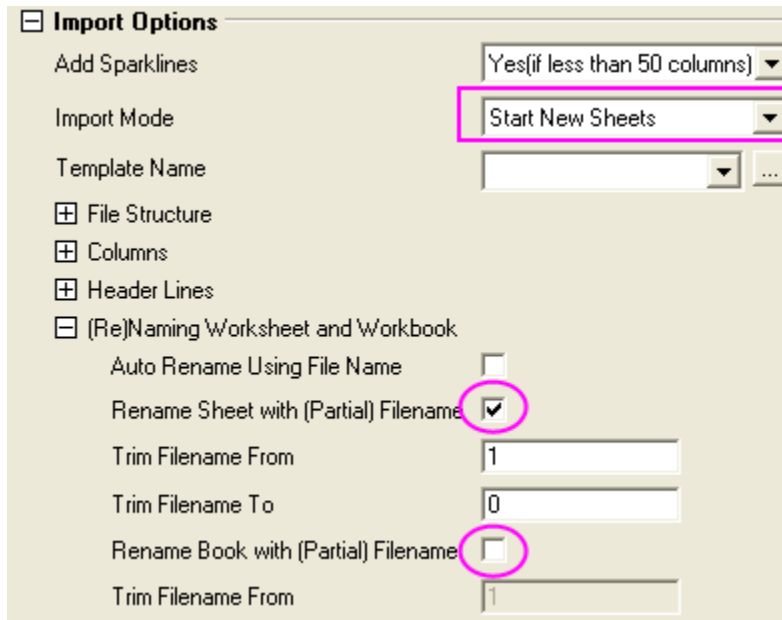
Customizing ASCII Import Dialog Settings and Saving a Theme

ASCII import and custom-file-format import both provide an *options* dialog where a user can customize import settings and then save settings for later use on similar files.

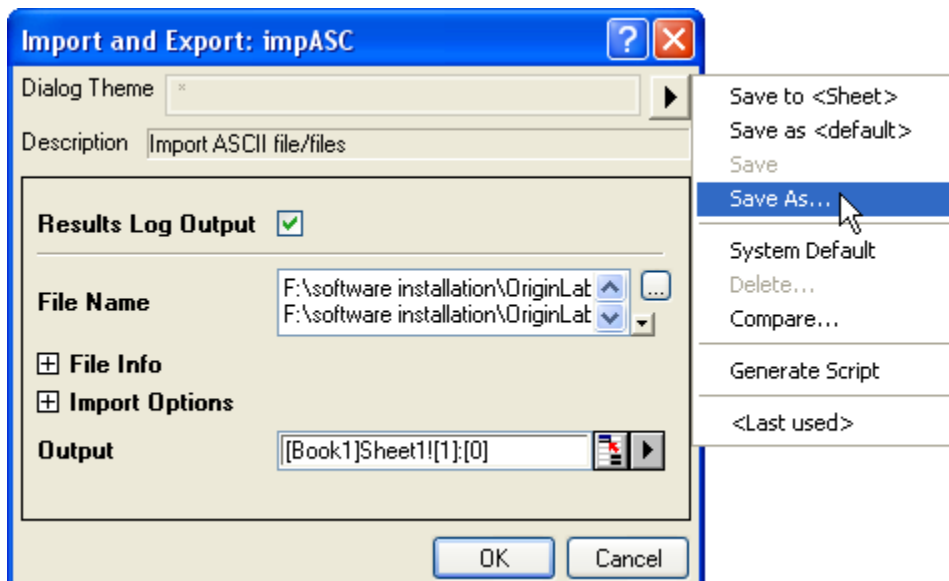
1. Start with a new book and click the **Import Multiple ASCII** button  on standard toolbar.
2. Select the files **sensor01.dat**, **sensor02.dat** from **\Samples\Curve Fitting** and add to lower panel of the file dialog. Click the file name column header in lower panel to sort the files by name. Keep the **Show Options Dialog** box checked and click **OK**. This will open a dialog for import settings.



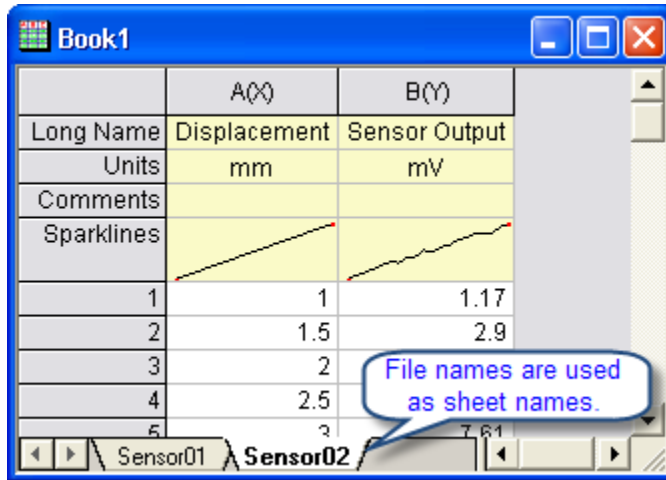
3. Change import mode to **Start New Sheets**. Expand **(Re)Naming Worksheet and Workbook** node and change settings so that only sheet gets renamed.



- Click on right arrow button at the top of the dialog and select **Save As**, then give it a name such as **My Multifile Import** and click **OK**. This saves your settings to a theme file on disk.



- Click **OK** and the 1st file gets imported into the current sheet, and a new sheet is created for the 2nd file. File names are used as sheet names.



- Now start a new book and invoke the menu **File: Recent Imports: impASC: My Multifile Import**. In the file dialog pick the three files: **step01.dat**, **step02.dat**, **step03.dat** from **\Samples\Curve Fitting**. Click the file name column header in lower panel to sort the files by name. Click **OK**. The settings from the theme you saved and selected here are used to perform the importing.

Saving ASCII Import Settings to Worksheet

Custom ASCII import settings can be saved either as a theme file to disk, or can be saved into the worksheet itself.

1. Start with new book and invoke **File: Import: Single ASCII** from menu, and select the file **sensor01.dat** from **\Samples\Curve Fitting** subfolder, and keep the **Show Options Dialog** checked and click **OK**.
2. In the **Import Options** dialog, select **No** with the **Add Sparklines** drop-down list. Expand **(Re)Naming Worksheet and Workbook** node and change settings so that only sheet gets renamed by file name, not book.
3. Click on Top Arrow icon and select **Save to <Sheet>** and click **OK**. Then your custom settings get saved to the sheet and data is imported.
4. Select **File: Save Template As** from the main menu. Enter **SensorImport** as the template name. Click **OK** to save the workbook template. This template contains both the import settings and the worksheet properties settings.
5. Now we create a new workbook from this template. Click the **Open Template** button on the **Standard** toolbar. Select **SensorImport.otw** under your User Files Folder and then click **Open**. A workbook is created from the template.
6. With this book active, drag and drop the file **sensor02.dat**. The data gets imported and only sheet name changes to the new file name. The sparklines were not turned on.

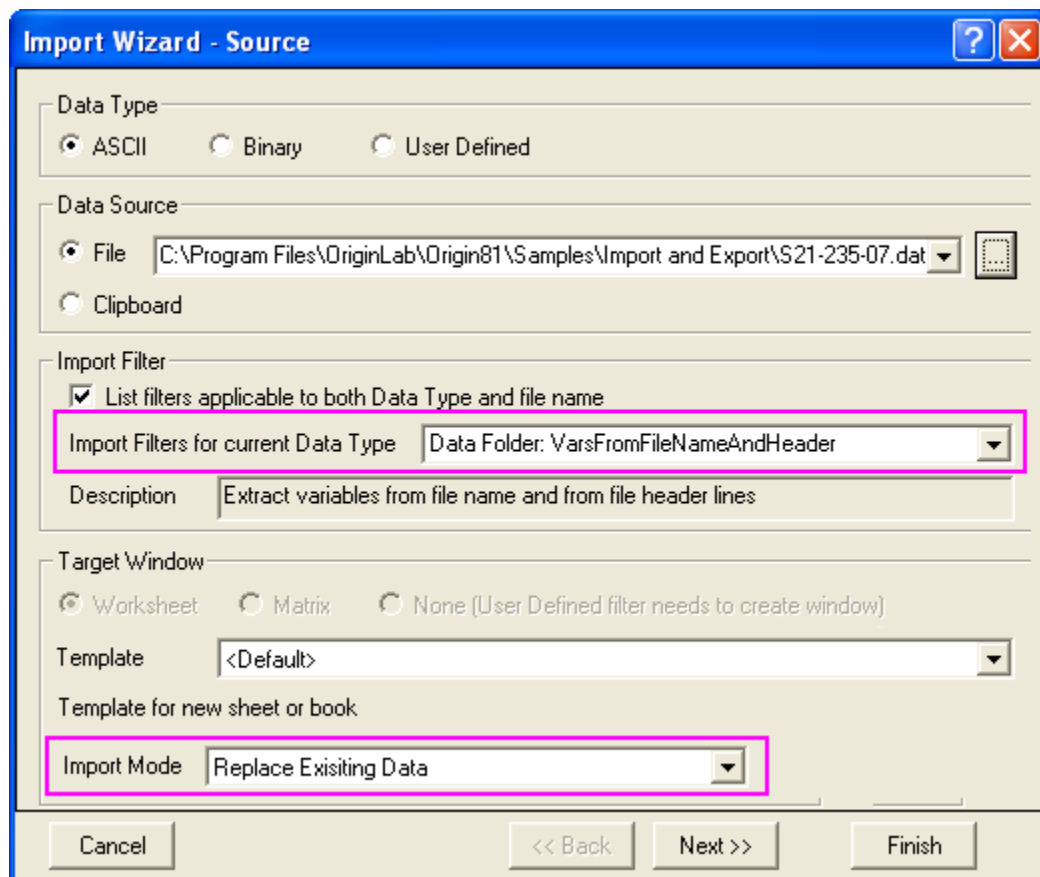


When saving custom settings to a worksheet, it is useful to then save the sheet, along with other desired analysis operations, as an **Analysis Template** for repeat analysis of similar data from multiple files. See the Batch Processing tutorial for details.

Import Wizard and Import Filters

The Import Wizard allows you to step through your data file, customize settings including how to parse header lines to create variables, and then save all of your custom settings as an import filter (.OIF) file for repeat use. The filter file can reside in the data folder, in the **\Filters** sub folder of your **User Files Folder**, or even saved to the worksheet itself for use with **Analysis Templates**. The Wizard is typically useful when the file has header lines that need to be parsed, or the file needs custom settings such as fixed width, or for executing LabTalk script at the end of the import for post processing.

1. Start with a new book. Click on the Import Wizard button  in the Standard toolbar to launch the wizard.
2. Select the file **\Samples\Import and Export\S15-125-03.dat**.
3. Note that the **Import Filter for Current Data Type** drop-down changes to show **Data Folder: VarsFromFileNameAndHeader**, this is a filter already created for this file and shipped with Origin, and is automatically picked up from the same folder as the data file you chose. Then change **Import Mode** to **Replace Existing Data**.



4. Click **Next** and walk through the pages. Notice controls on **Header Lines** pages that allows flexible definition of where the header lines end, where the sub header lines are located, and what gets assigned to long name and units etc.
5. For this file the **Variables Extraction** and **Variables Extraction by Delimiter** page define how to parse the header lines to extract values from them.

6. Click **Next** until you get to the **Save Filters** page. Check the **Save filter** box and change the radio button to **In the Window**. This will save the filter in the active worksheet.

Import Wizard settings can be saved to a filter file for re-use. Filter files can be selected on the first page of this wizard.

Once saved, import filters can also be used to automatically determine import settings when dragging and dropping data files into Origin and when opening data files with the File:Open menu item.

Save filter

In the data file folder C:\Program Files\OriginLab\Origin81\Samples\Import and Exi

In the User Files folder C:\Documents and Settings\Administrator\My Documents\Or

In the Window

Show Filter in File:Open List

Filter Description Extract variables from file name and from file header lines

Filter file name (.DIF extension will be appended) VarsFromFileNameAndHeader

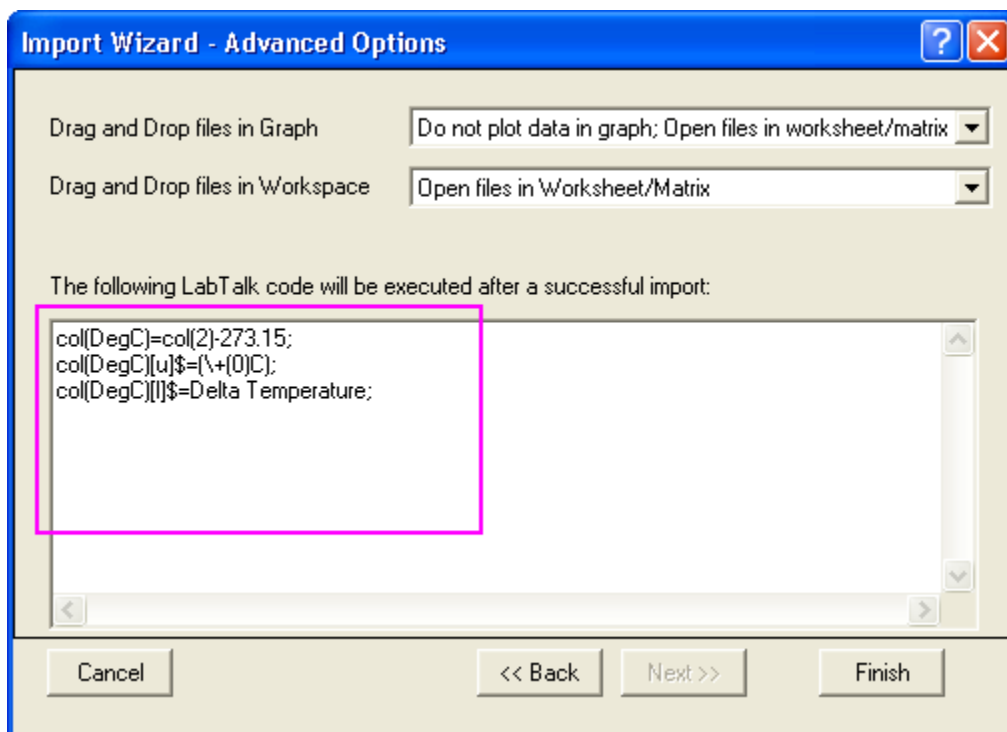
Specify data file names to which this filter will be associated. You can use wild cards, and can specify multiple names separated by ':' S*.*.dat
Ex: *.txt or *.txt; *.dat; mydata????.*

Specify advanced filter options

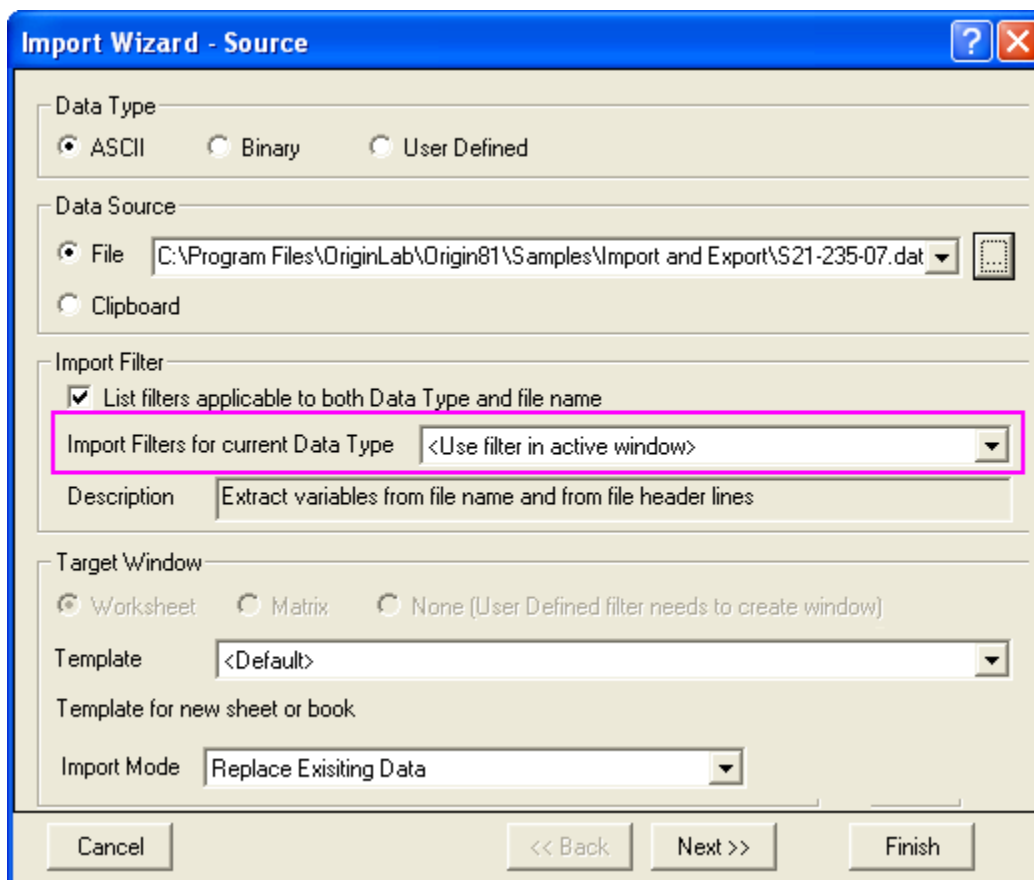
Cancel << Back Next >> Finish

7. Now check the **Specify advanced filter options**. It brings you to a last page where script (to run at the end of the import) can be specified. In the edit box enter:

```
col(DegC)=col(2)-273.15;
col(DegC)[u]$=(\+(0)C);
col(DegC)[l]$=Delta Temperature;
```



8. Click **Finish**, the file gets imported and the import filter is now saved in your worksheet. The fifth column is a column added by the script. It is the Delta Temperature data in Degree Celsius.
9. With the worksheet active, click **Import Wizard** button again and pick file `\S21-235-07.dat`. Note that the **Import Filter for Current Data Type** drop-down shows *<use filter in active window>*, so Origin picks up the filter settings that were saved in the worksheet.



10. Click **Finish** and the file gets imported and the script gets executed (the values in column 5 is updated).



You can save the import settings to the worksheet, perform analysis on the imported data, and save the workbook as an **analysis template** for repeat processing of similar data files. See the tutorial on batch processing for more information.

6.1.2 Single ASCII

Summary

The **File: Import: Single ASCII** menu allows you to automatically import a single ASCII file where the data columns are delimited orderly and it consists of few header lines (maybe just a short description for the file and then names and units for the columns).

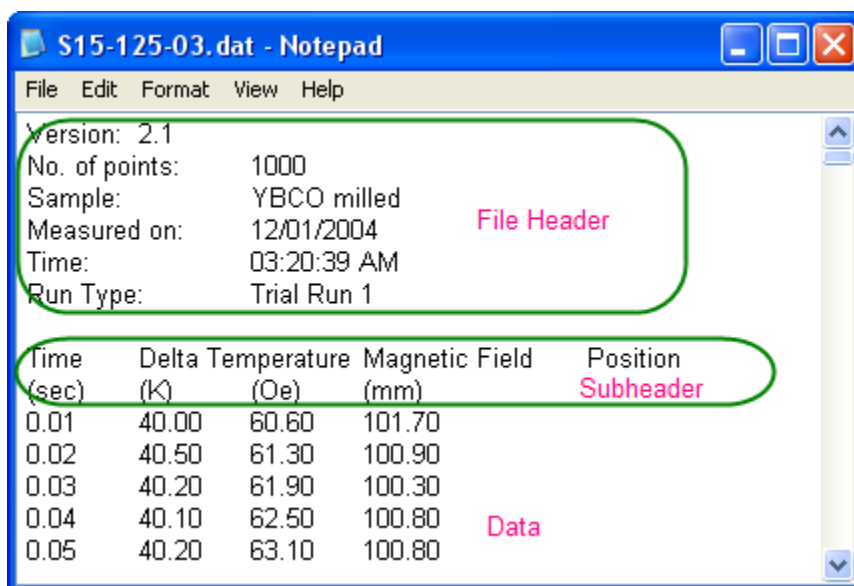
Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

This tutorial will show you how to Import ASCII files.

Steps

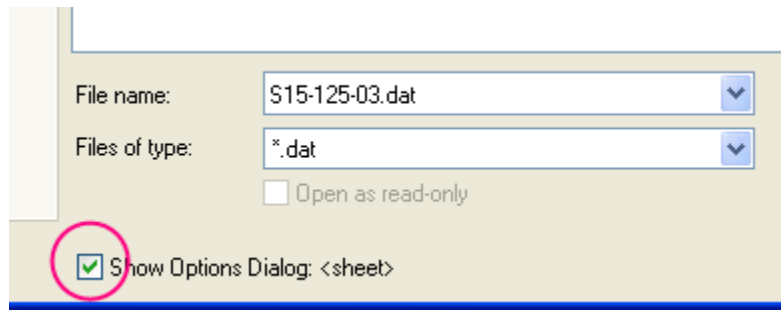
Using Windows Explorer, browse to the `\Samples\Import and Export` subfolder of the Origin program folder (by default installed in the Program Files folder). Open the file `S15-125-03.dat` in Windows Notepad. You can see that this file includes header lines and data lines. For Single ASCII files, Origin can auto detect file header/subheader and extract this information to the worksheet headers, such as *Long Name*, *Units*, etc.



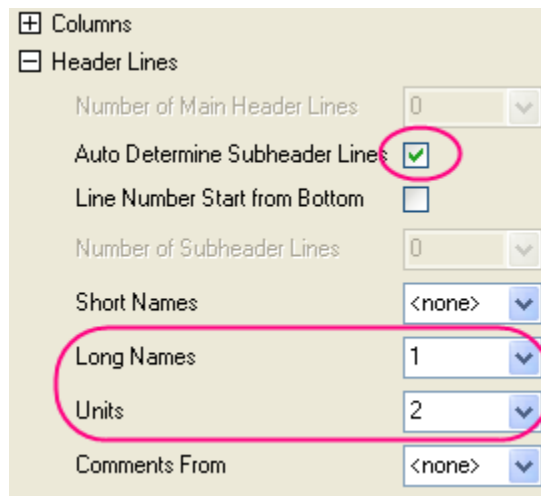
Note: Header lines are lines of text that are not part of the data and do not share the same delimiter formatting as the data. Subheader lines also are not part of the data, but share the same delimiter formatting and therefore correspond to particular columns of data.

To Import this file

1. Select **File: Import: Single ASCII** from the menu to open the File Import dialog. Browse to the `\Samples\Import and Export` subfolder of the Origin program folder. Highlight the file `S15-125-03.dat`.
2. If you double-click this file or click the **Open** button, Origin will import the file automatically. To view the settings for how Origin will import the file, check the **Show Options Dialog** checkbox at the bottom of the dialog and then click **Open**. This will bring up the `impASC` X-Function dialog.



- Expand the **Import Options: Header Lines** tree node.



By default, Origin auto detects the subheader, and data will be imported from subheader. In this example, Origin automatically set the first line, Time Delta Temperature Magnetic Field Position as the worksheet column Long Name and the second line, (sec) (K) (Oe) (mm) as Units.

- Click **OK** to accept these settings and import data into worksheet.

	A(Y)	B(Y)	C(Y)	D(Y)
Long Name	Time	Delta Temp	Magnetic Fi	Position
Units	(sec)	(K)	(Oe)	(mm)
Comments				
Sparklines				
1	0.01	40	60.6	101.7
2	0.02	40.5	61.3	100.9
3	0.03	40.2	61.9	100.3
4	0.04	40.1	62.5	100.8
5	0.05	40.2	63.1	100.8

S15-125-03

6.1.3 Import Wizard

Summary

The **Import Wizard** allows you to preview your file to help with importing of more complicated ASCII files. This is useful for files with many lines of header where you may wish to extract variables from both the import file name and the file header to later use for annotation on your graph.

Minimum Origin Version Required: Origin 8.0 SR6


What you will learn

This tutorial will show you how to use the **Import Wizard**.

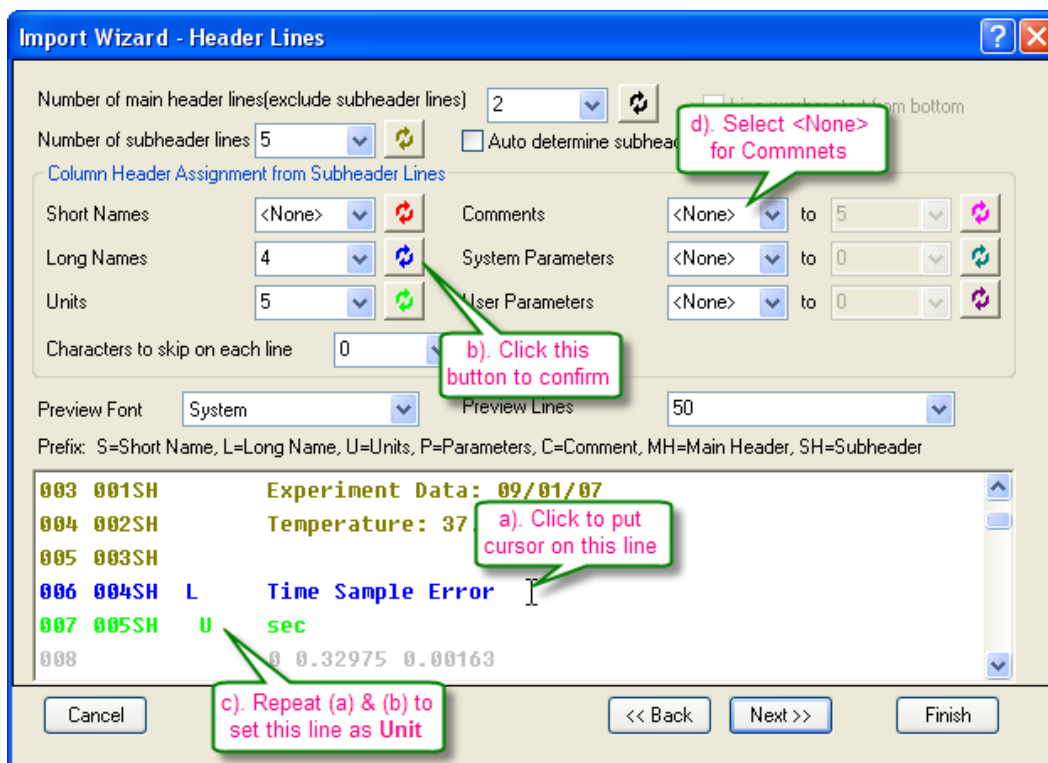
Steps

The **import wizard** allows you to import complicated ASCII files, extract variables from the import file name and header (for reuse in Origin), specify custom delimiters and date formats, or handle post-processing of your imported data using a custom LabTalk script. In addition, another chief advantage of using the Wizard to import your data files is that you can save your custom settings to a filter that can be used repeatedly to import the same or similarly-structured files.

1. Create a new workbook. Select **File: Import: Import Wizard** to open the *Import Wizard* dialog.

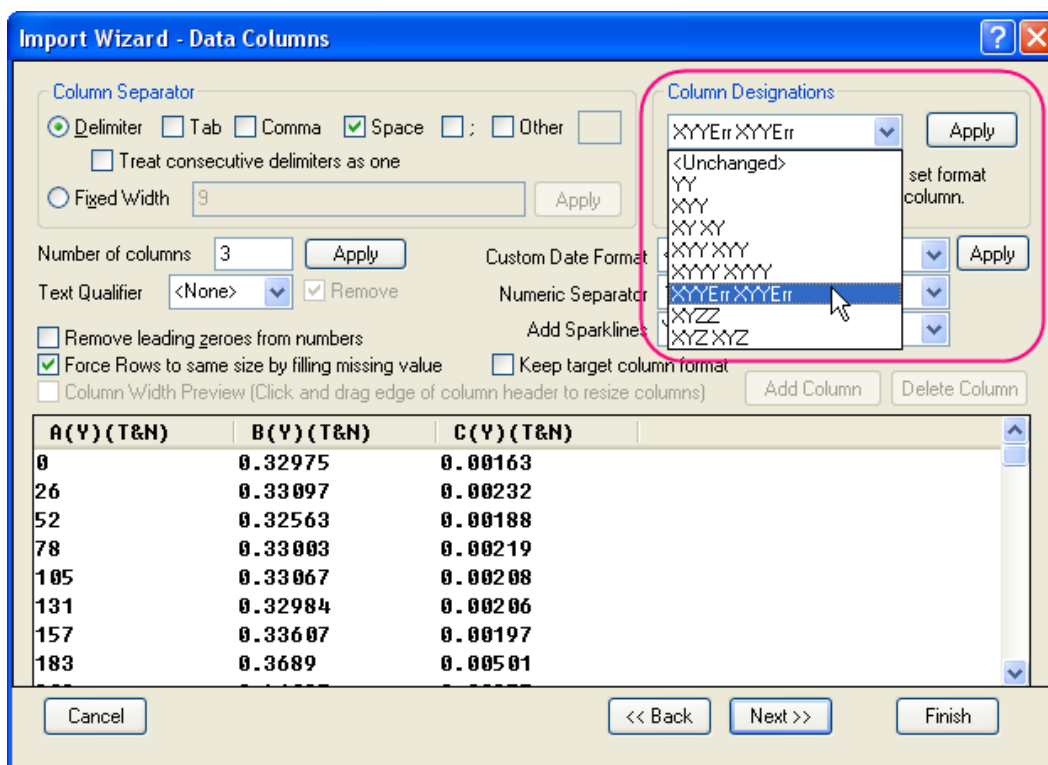
Click the browser button  right beside **File** box. Browse to the *\Samples\Import and Export* folder and open the file *F1.dat*, and then click the **Next** button to navigate to **File Name Options** page.

2. Make sure the **worksheet with file name** box is checked to rename worksheet by imported file name. Click **Next** to navigate to **Header Lines** page.
3. This page enables you to easily customize the worksheet headers. For example, to specify the worksheet long name, put your cursor to the following line, and then click the button beside *Long Names*.



Similarly, specify the *Unit* line and select *<None>* for *Comments*.

4. Skip the next **Variable Extraction** page and go to the **Data Columns** page. Select **XXYErrXXYErr** from the *Column Designations* drop-down and click **Apply**.



5. Skip the **Data Selection** page, and go to the **Save Filters** page. To use these settings again, you can save this import procedure as a filter. Check the *Save filter* box and give a proper filter name in the *Filter file name* edit box (*MyFilter* in this example).

Import Wizard - Save Filters

Import Wizard settings can be saved to a filter file for re-use. Filter files can be selected on the first page of this wizard.

Once saved, import filters can also be used to automatically determine import settings when dragging and dropping data files into Origin and when opening data files with the File:Open menu item.

Save filter

In the data file folder C:\VC\c32\Samples\Curve Fitting\
 In the User Files folder C:\VC\c32\Lary\Filters\
 In the Window

Show Filter in File:Open List

Filter Description: An example to create filter

Filter file name (.OIF extension will be appended): MyFilter

Specify data file names to which this filter will be associated. You can use wild cards, and can specify multiple names separated by ','
*.dat
Ex: *.txt or *.txt; *.dat; mydata????.*

Specify advanced filter options

Cancel << Back Next >> Finish

Click the **Finish** button to import the data.

6.1.4 Import Time Data

Summary

Origin interprets *Dates* based upon the Gregorian Calendar, while *Time* is interpreted in *hours:minutes:seconds*. When working with Date and Time data, Origin displays these data in different formats, but internally uses underlying numeric values for calculations and certainly plotting operations. This tutorial shows you how to import custom date/time data.

Minimum Origin Version Required: Origin 8.0 SR3

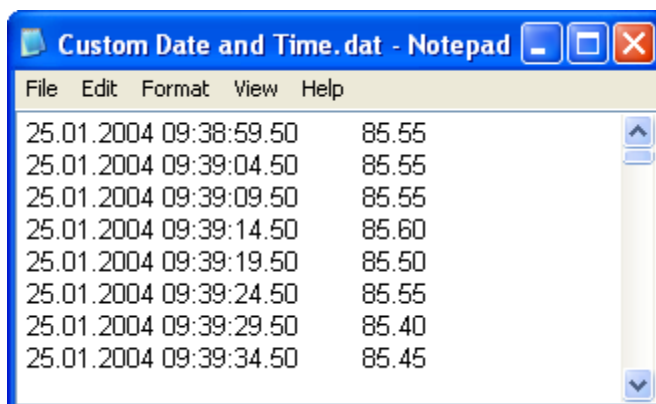
What you will learn

This tutorial covers:

- How to import data using multiple delimiters.
- How to define a custom date/time format
- How to change the display settings for the custom date/time format

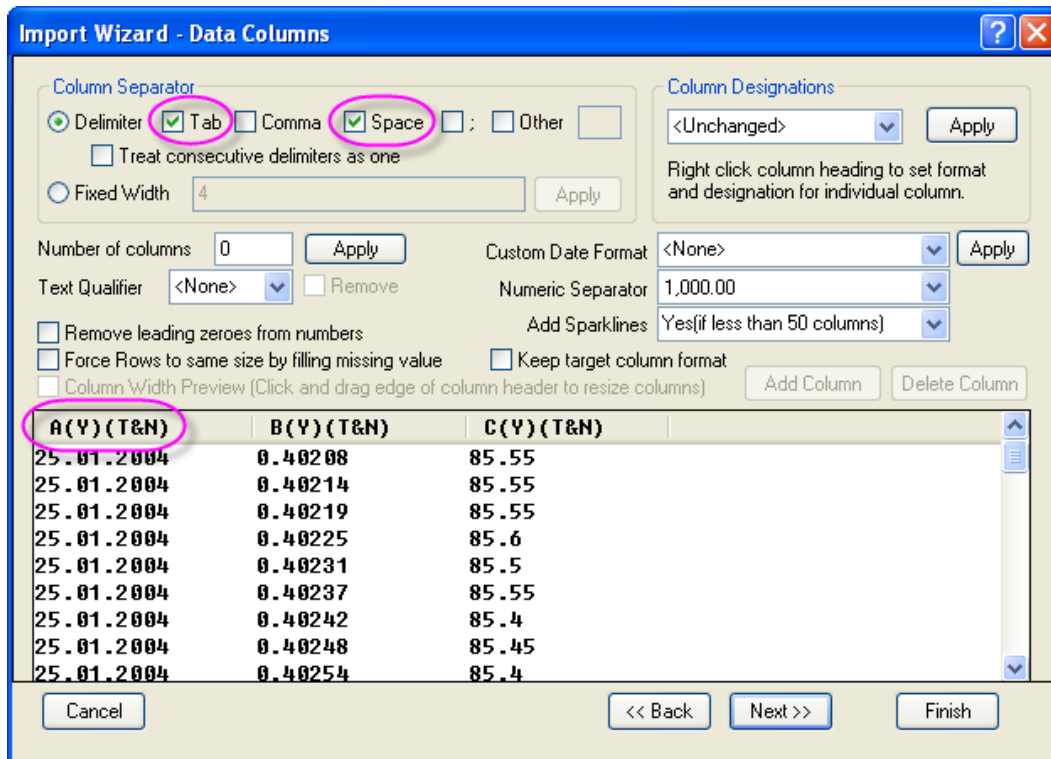
Steps

1. We will import `\Samples\Import and Export\Custom Date and Time.dat` in this tutorial. Before importing the file, let's look at the data structure first.



We can see that there is a space between *Date* and *Time*, and it uses Tab to separate *Time* from the rest of the data. So we will use multiple delimiters to import this data file.

2. Open the file in **Import Wizard**. Accept the default settings on all pages until you get to the **Data Columns** page. Origin will, by default, use Tab to separate the data into two columns. To divide *Date* and *Time*, check the **Tab** and **Space** checkbox in the **Column Separator** group.

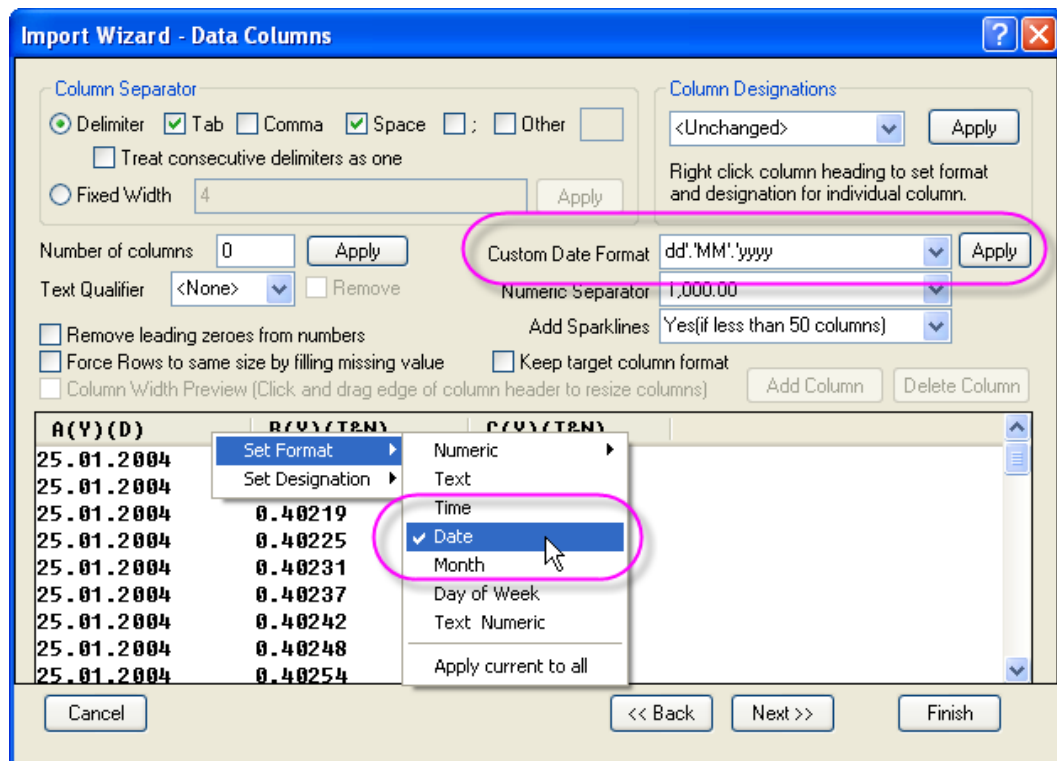


Note in the preview box that the column title is **A(Y)(T&N)**, where **(T&N)** means the data format is *Text & Numeric*. Because the date uses "." to separate day, month and year, Origin by default treats the first column as *Text*. For the second *Time* column, Origin shows the underlying numeric values. To import data correctly, we should change the column properties.

3. In the **Custom Date Format** edit box, enter:

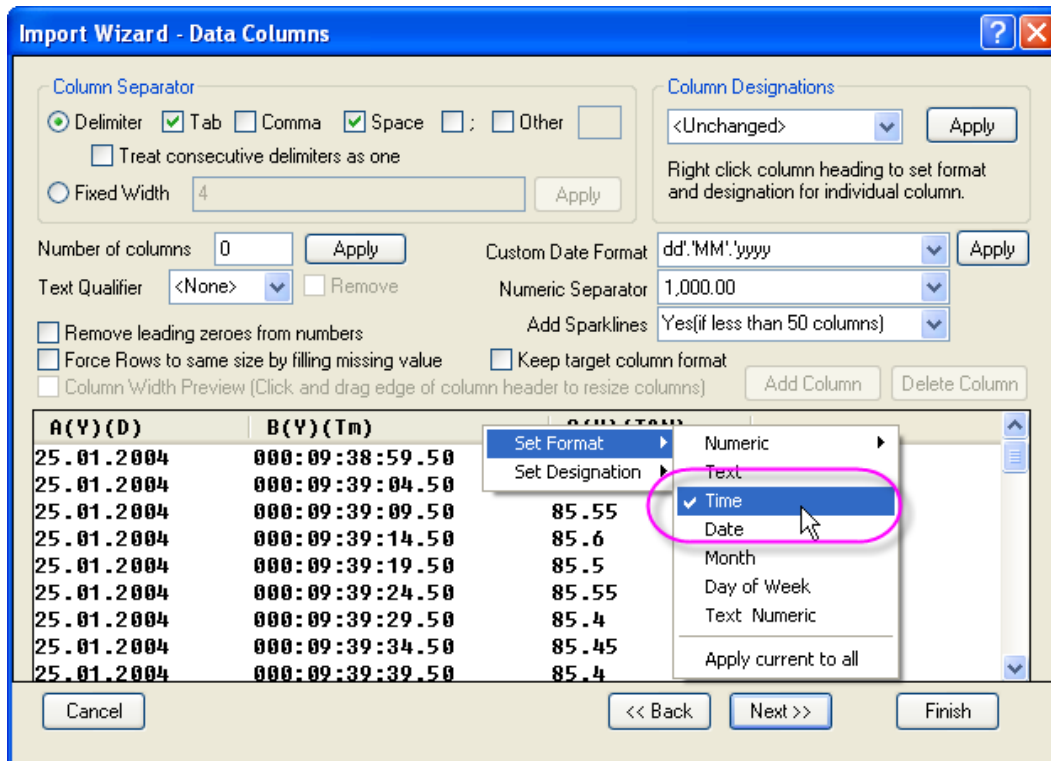
`dd'.'MM'.'yyyy`

where *dd*, *MM* and *yyyy* mean the days, months, and year respectively. Since the "." is used as a separator, we need to put single quotation marks around it in the format specification. After entering the custom format, press the **Apply** button next to the edit box. Then right-click on the header of the first column in the preview and select *Date* from the context menu:



Then the column title will turn into **A(Y)(D)** which means this is now a *Date* data column.

- Similarly, right-click on the header of the second column and select *Time* to set that column as a time column:



You can see the time data display in long format. We can change the display setting after imported.

5. Click **Finish** button to import data. Then double-click the second column title to open the **Column Properties** dialog, and set the Time display as:

HH:mm:ss.##

The final worksheet data after imported will looks like:

	A(Y)	B(Y)	C(Y)
Long Name			
Units			
Comments			
Sparklines			
1	25.01.2004	09:38:59.500	85.55
2	25.01.2004	09:39:04.500	85.55
3	25.01.2004	09:39:09.500	85.55
4	25.01.2004	09:39:14.500	85.6
5	25.01.2004	09:39:19.500	85.5
6	25.01.2004	09:39:24.500	85.55
7	25.01.2004	09:39:29.500	85.4
8	25.01.2004	09:39:34.500	85.45

Note: In the case of this particular data file, the first column simply has the exact same date in every row. So at this point you may want to set this first column as Disregard by right-clicking on the column header and selecting *Disregard* from the context menu. Then you can set the 2nd column as type X, and plot the data in the third column against the time data in the 2nd column.

6.1.5 Post Processing with Import Filter

Summary




The Import Wizard allows defining a custom filter to import ASCII and simple binary files. The filter can then be reused with similar data files once created. The filter mechanism also allows including LabTalk script lines that will then be run at the end of the import. This capability allows user to add post-processing script code to the filter.

Minimum Origin Version Required: Origin 8.0 SR6

What you will learn

- How to add post processing script to existing import wizard filter

Steps

1. Start a new workbook by clicking the **New Workbook** button  on the Standard toolbar.
2. Click the **Import Wizard** button  on the Standard toolbar to open the wizard.
3. In the **Data Source** group, click the  button to the right of **File** and navigate to and add the file **Samples\Import and Export\S15-125-03.dat**.
4. A suitable filter for importing this file already exists in the data folder. Click the **Next** button to proceed thru all the pages of the wizard till you get to the **Save Filters** page.
5. On this page, check the **Save Filter** check box and also the **Specify advanced filter options** check box and then click **Next**. This will bring you to the **Advanced Options** page.
6. On this page, copy and paste the following lines in the edit box:


```
nbegin iy:=(1,4) func:=gauss;
nlfir;
nlend output:=1 autoupdate:=au_auto;
```
7. Press **Finish**. This will save the filter along with these added lines of script, and the file will be imported and the script will run. The workbook will then have three sheets, which will include the custom report sheet and the fitted curve sheet, which are results of gaussian function fit to column 4 of the imported data.
8. Start a new project and open import wizard again and add all three files **S15_125_03.dat**, **S21-235-07**, **S32-014-04** in the file open dialog.
9. Check to see that the **Import mode** drop-down is set to **Start New Books** and click Finish. Your modified filter will be used and after each file is imported, the 4th column will be fit with the gaussian function.

7 Collaboration and Connectivity

7.1 Connectivity

7.1.1 Excel

Working with Excel

Summary

Origin provides flexible ways to interact with Excel. You can either import Excel data into an Origin workbook, or open an Excel book inside Origin. If you require full access to all of Origin's graphing and analysis features, you will probably want to import your Excel data files into Origin. If it is important to maintain a separate Excel workbook file—perhaps so that other colleagues who do not work with Origin have access to that file—you will probably want to open your Excel data files directly. We give a brief introduction to working with Excel in this tutorial.

This tutorial will show you how to:

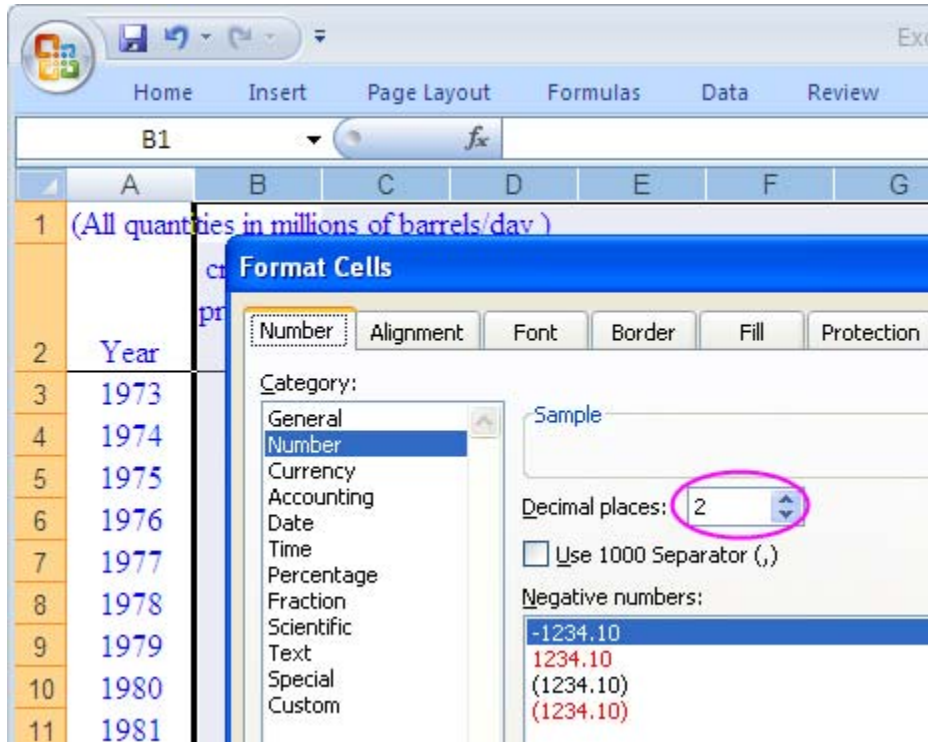
1. Copy and paste data from Excel with full precision
2. Import an Excel file into an Origin workbook
3. Open an Excel file in Origin
4. Save an Excel file with path relative to the Origin Project file

Copying and Pasting Data from Excel

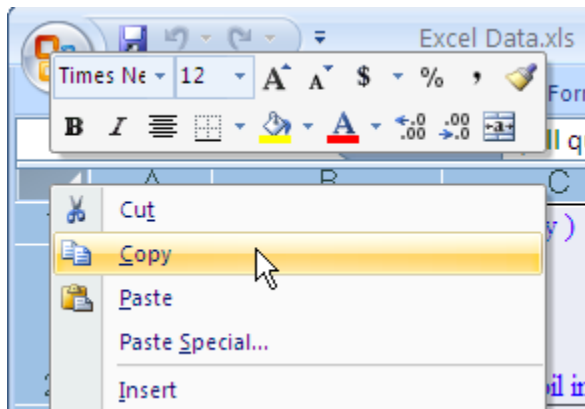
It may be desirable at times to simply copy and paste data from Excel to Origin instead of importing or opening Excel. These steps show that such a copy/paste operation can bring in data with full precision.

1. Launch Excel and Origin separately.
2. Open the file <Origin Installation Folder>\Samples\Graphing\ExcelData.XLS in Excel.

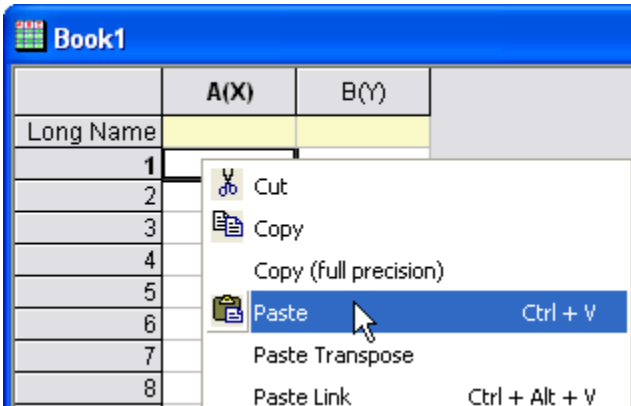
3. Select columns B through L, then right-click and bring up the **Format Cells** dialog, and set the number of decimal places to 2. So now Excel shows fewer decimal places.



4. Click on the top left cell in the Excel sheet to select the entire sheet and right-click and select **Copy**, or use the keyboard shortcut **Ctrl+C** to copy.



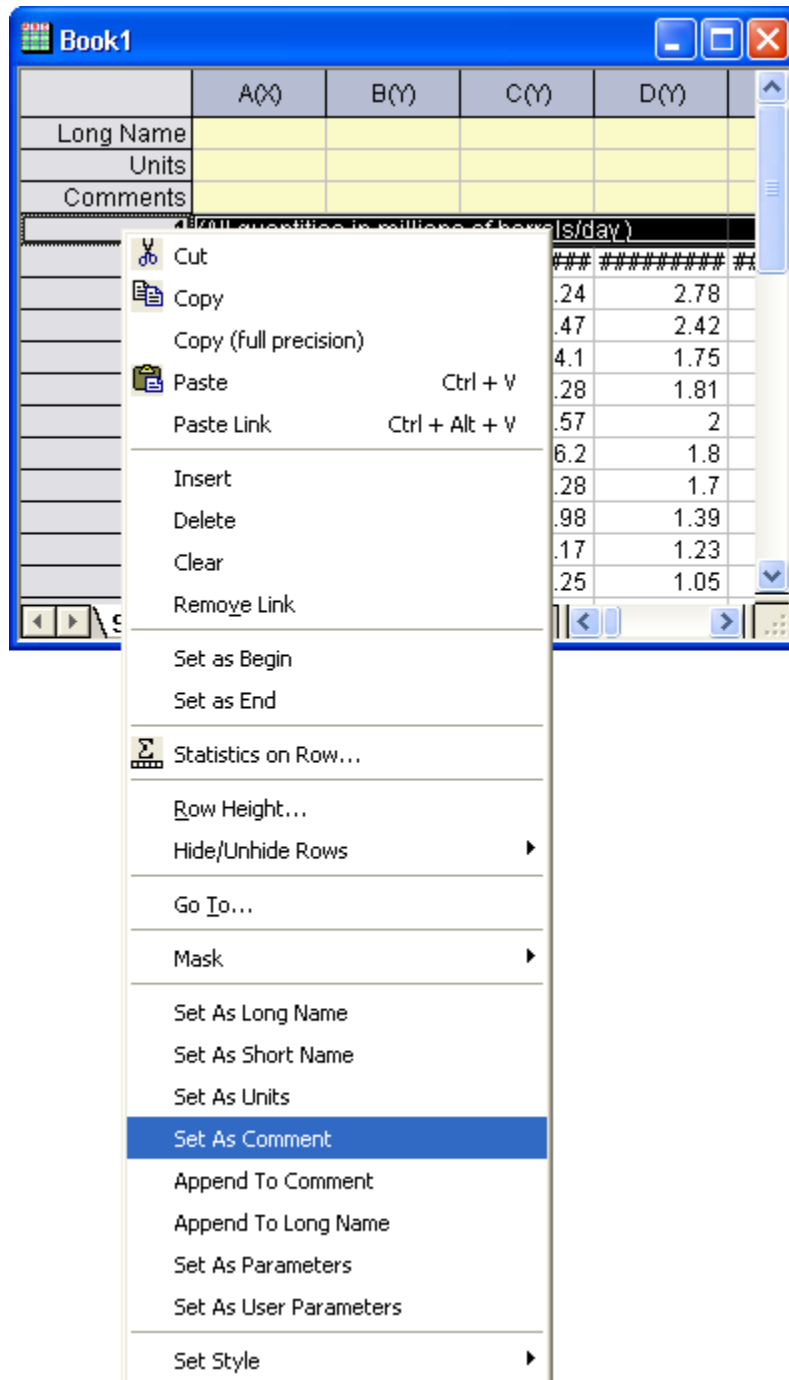
- 5. Go to a new book in Origin, place the cursor in row 1 of column 1 and do **Ctrl+V** or right-click and **Paste**.



- 6. Note that the numbers come into Origin with full precision, not the number of displayed digits (2) in Excel.

C7(Y)	C8(Y)	C9(Y)	C10(Y)	C11(Y)
#####	#####	#####	Transportation petroleum U	
56.39	0.348	0.307	0.915	
55.91				
55.48				
58.74				
61.63	petroleu	imports	petroleu	ation
63.3	m	as % of	m	petroleu
65.17	2	consumpt	U.S.	consumpt m use as
63.07				
60.87	3	56.39	0.35	0.31 0.92
59.5	4	55.91	0.35	0.30 0.94
58.74	5	55.48	0.36	0.29 0.99

7. In Origin, right-click on the row 1 header and select **Set as Comment** to make this row an Origin column comment.



- Right-click on row 1 again and select **Set as Long Name**. Then rows 1 and 2 in Excel become the worksheet header in Origin:

	A(Y)	B(Y)	C1(Y)	C2(Y)
Units				
Comments	(All quanti			
Long Name	Year	Domestic c	Crude oil i	Petroleum
2	1/1/1974	8.77	3.47	2.42
3	1/1/1975	8.37	4.1	1.75
4	1/1/1976	8.13	5.28	1.81
5	1/1/1977	8.25	6.57	2
6	1/1/1978	8.71	6.2	1.8
7	1/1/1979	8.55	6.28	1.7

- You can now double-click column 1 and set it as **Date** and then plot the data.

Column Properties - [Book1]Sheet1...

<< Previous Next >>

Properties Enumerate Labels User Tree

Short Name A

Long Name Year

Units

Comments

Width

Options

Plot Designation X

Format Date

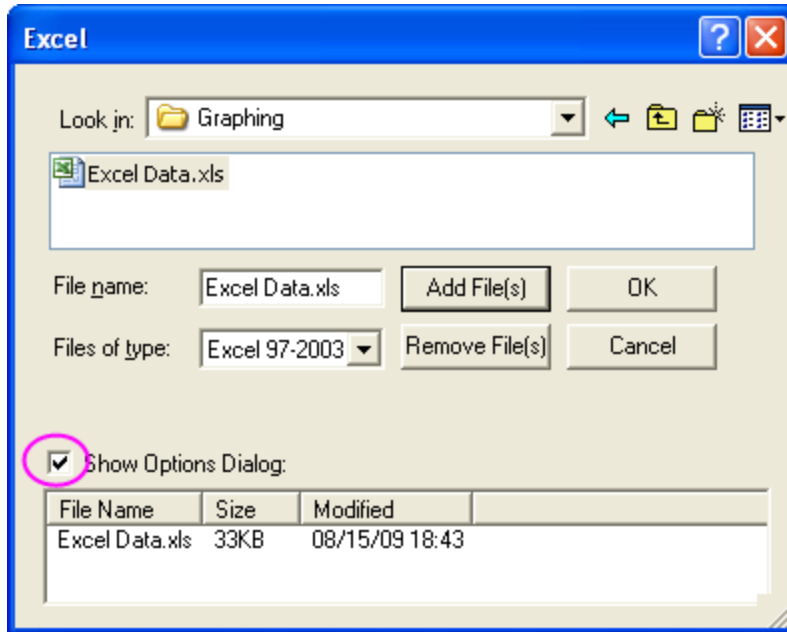
Display 2009

Apply Cancel OK

Importing Excel Files

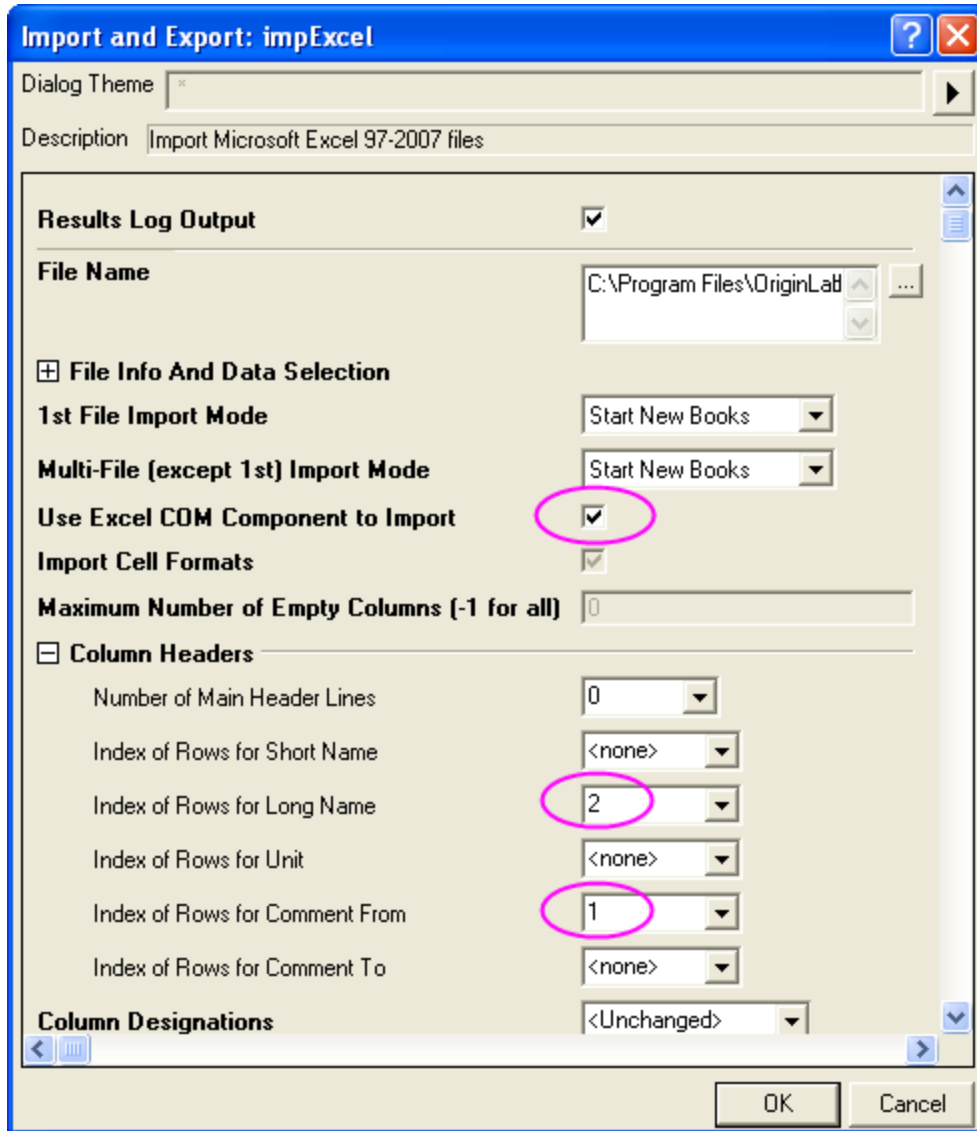
Origin supports importing Excel files directly into Origin workbooks. Multiple sheets are supported and controls are available for setting specific rows in the Excel sheet to be brought into an Origin worksheet as header information, including Long Name and Comments. If you want to perform analysis or data manipulation operations on your Excel data, we recommend importing your data into Origin.

1. With a new book active in Origin, use the menu to select **File: Import: Excel (XLS, XLSX)**.
2. Select the file `\samples\graphing\Excel Data.xls`, and make sure **Show Options Dialog** is checked.



3. In the dialog that comes up, leave the **Use Excel COM Component to Import** check box checked.
4. Set the **Index of Rows for Comments** to 1.

- Set the **Index of Rows for Long Name** drop-down to 2 and click **OK** to Import.

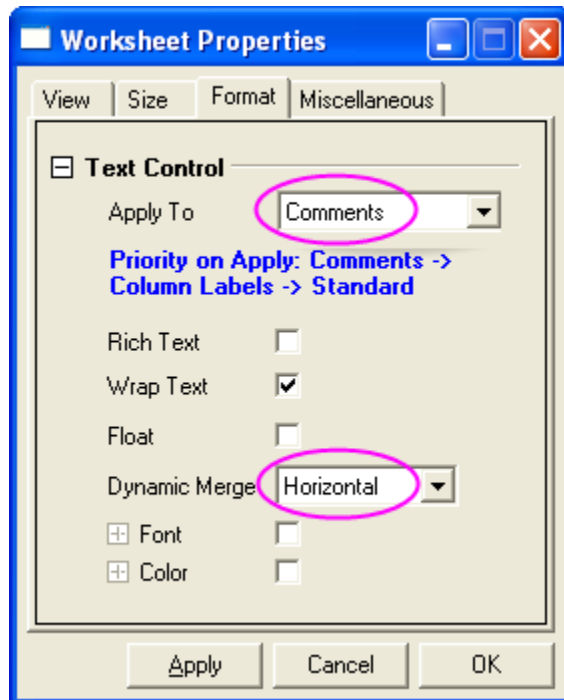


- Click and select the **Comments** cell under column 1, hold down the **Ctrl** key and drag the bottom right point of the selected cell to stretch across all columns with data, copying the same comments to all columns.

	A(X)	B(Y)	C1(Y)
Units			
Long Name	Year		e oil i
Comments	(All quantities)		
1	1/1/1973	9.21	3.24
2	1/1/1974	8.77	3.47
3	1/1/1975	8.37	4.1
4	1/1/1976	8.13	5.28

A callout bubble with the text 'Hold Ctrl key and drag' points to the bottom-right corner of the 'Comments' cell in column A, with a dashed pink arrow indicating the drag direction across the data columns.

- Press **F4** to bring up the format dialog, switch to the **Format** tab, change the **Apply To** drop-down to **Comments** and set **Dynamic Merge** to **Horizontal**, then click **OK**.



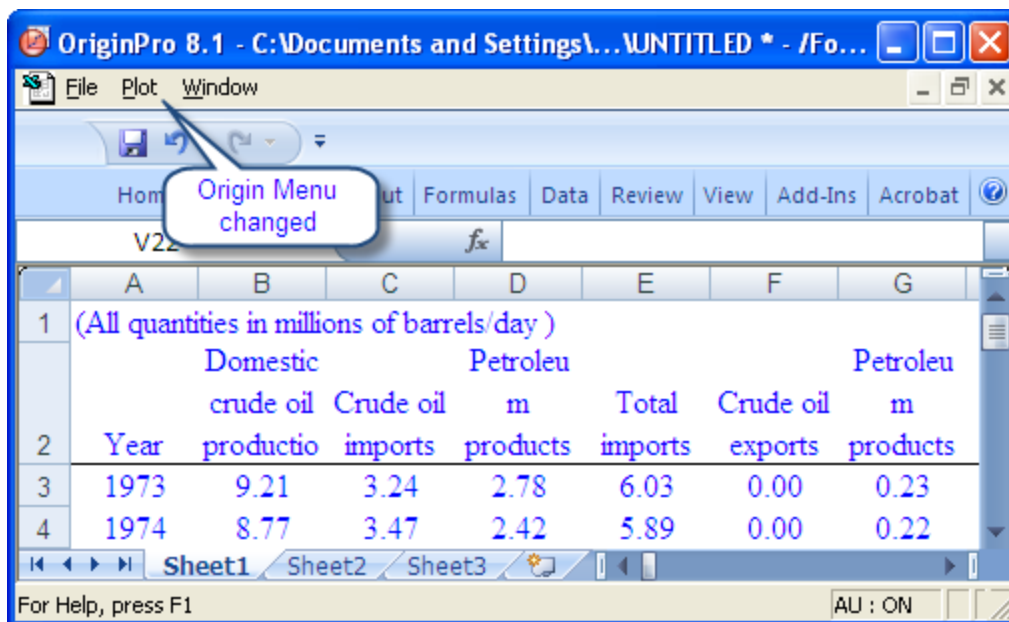
This sets the comments cells to be merged and to show in the center of all data columns.

C3(Y)	C4(Y)	C5(Y)	C6(Y)
Total impor	Crude oil e	Petroleum	U.S. petrole
(All quantities in millions of barrels/day)			
6.03	0	0.23	17.31
5.89	0	0.22	16.65
5.85	0	0.2	16.32
7.09	0	0.22	17.46
8.57	0.05	0.19	18.43

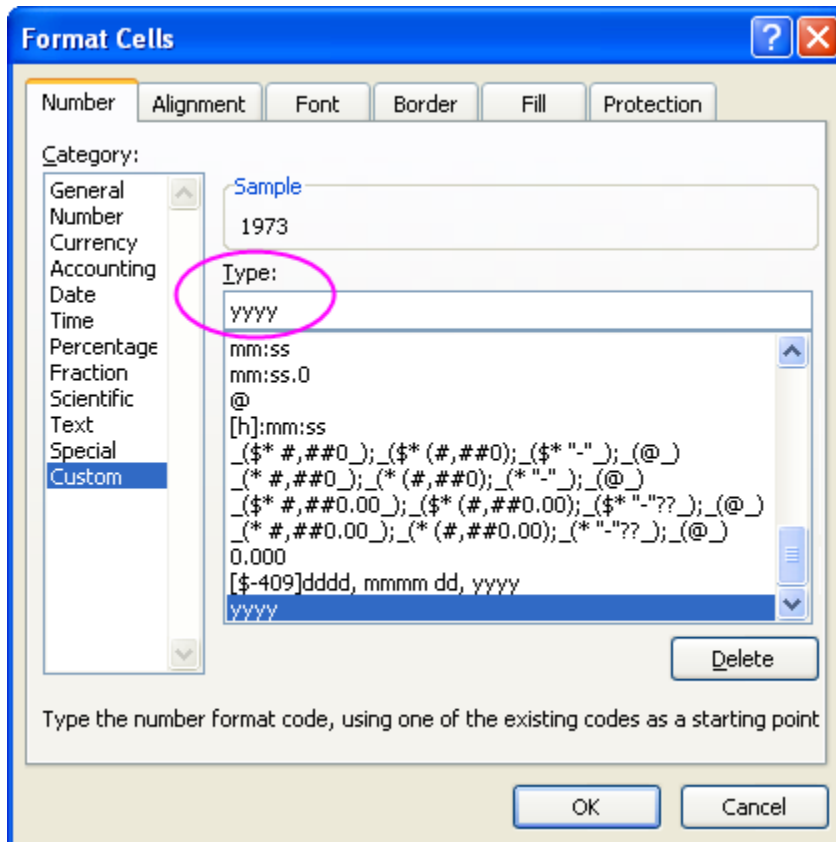
Open Excel File in Origin

At times it may be desirable to keep the data in an external XLS file and simply open the file inside Origin as an Excel window, and then work with the data. When you open Excel (.XLS or .XLSX) files as Excel workbooks in Origin, an OLE instance of Microsoft Excel is launched. You can plot directly using Excel workbook data, but many analysis features, as well as 3D plotting, will be inaccessible.

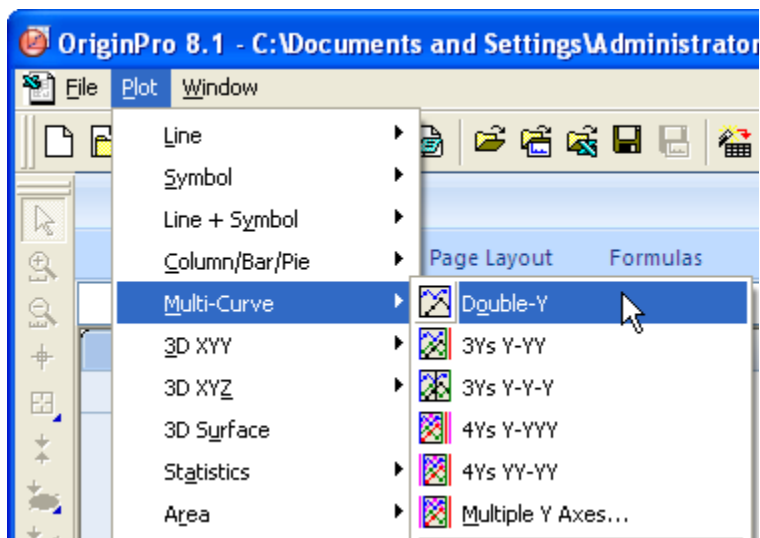
1. Select **File: Open Excel** and select the file `\Samples\Graphing\Excel Data.xls`.
2. A new Excel window opens inside the Origin workspace. When this window is active, the Origin main menu has different entries, some of which are specific to Excel, and the Excel toolbars are available.



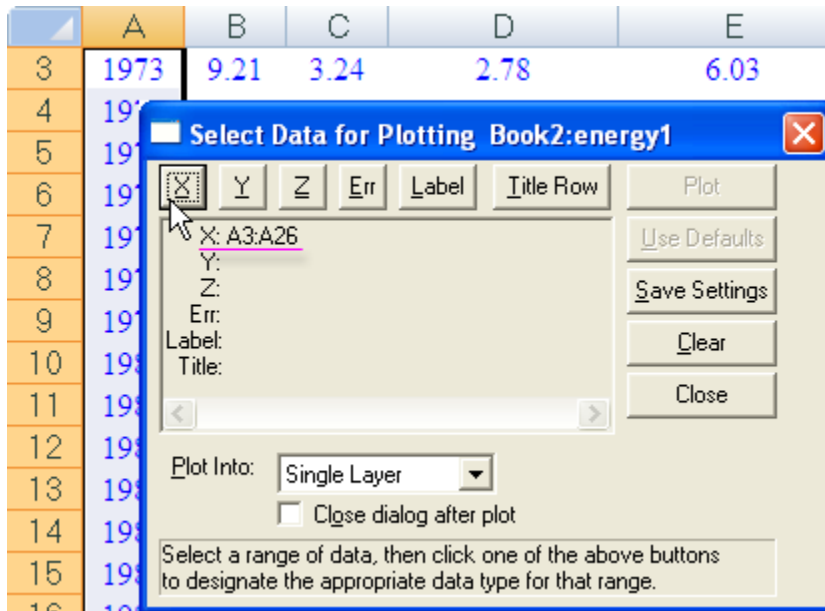
- Highlight the Excel data range **A3:A26** and right-click, then select **Format Cells** to make sure that the data is in **Date** format.



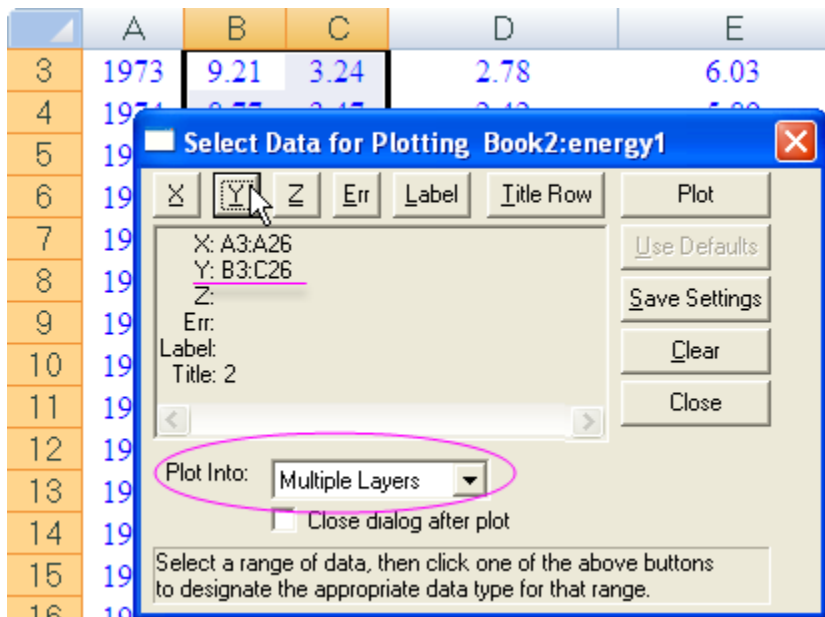
- Now select the **Plot** menu in Origin and select the **Multi-Curve: Double-Y** plot type.



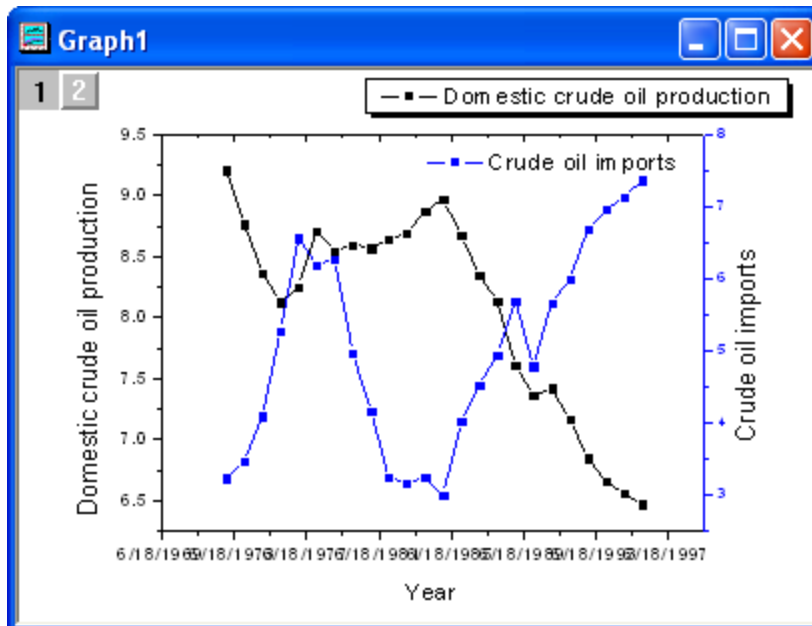
5. Select **A3:A26** in the Excel sheet and click **X** in the **Select Data for Plotting** dialog to assign the X data.



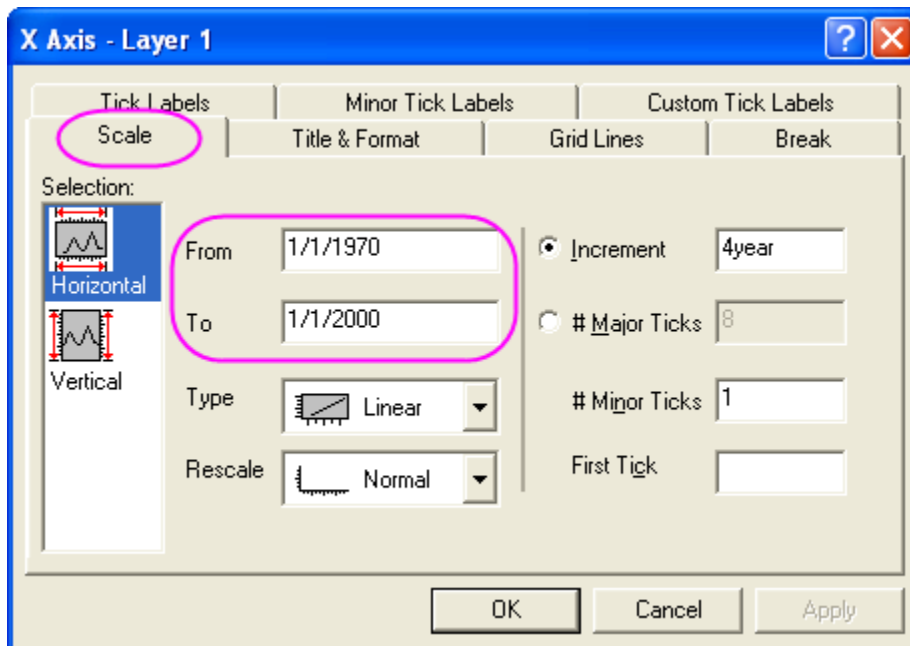
6. Select **B3:C26** in the Excel sheet and click **Y** in the **Select Data for Plotting** dialog to assign the Y data, and then change the **Plot Into** drop-down to be **Multiple Layers**.



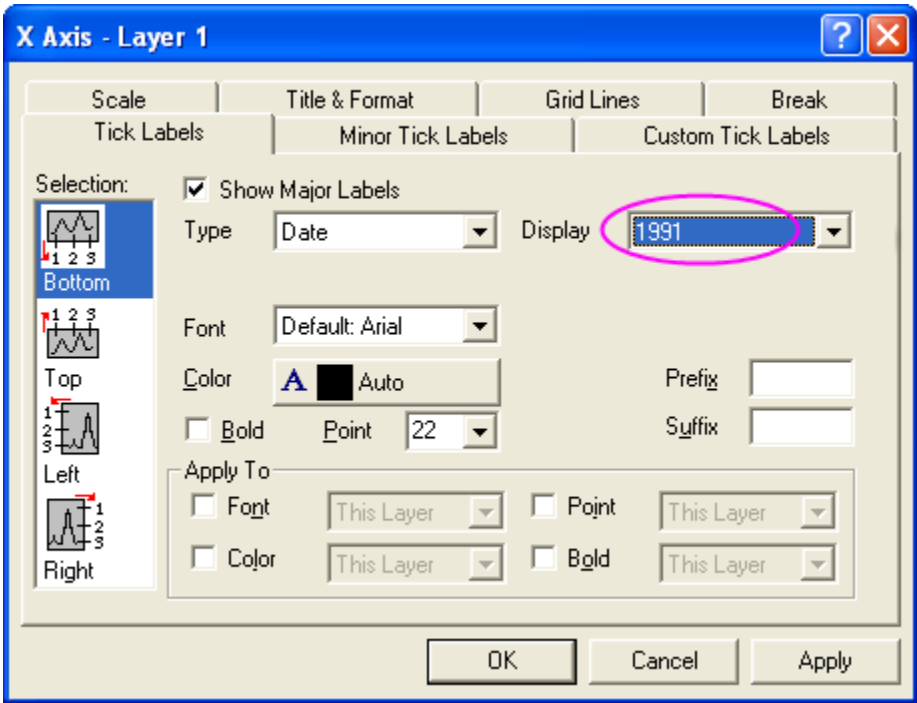
7. Click **Plot** and a double-y plot is created.



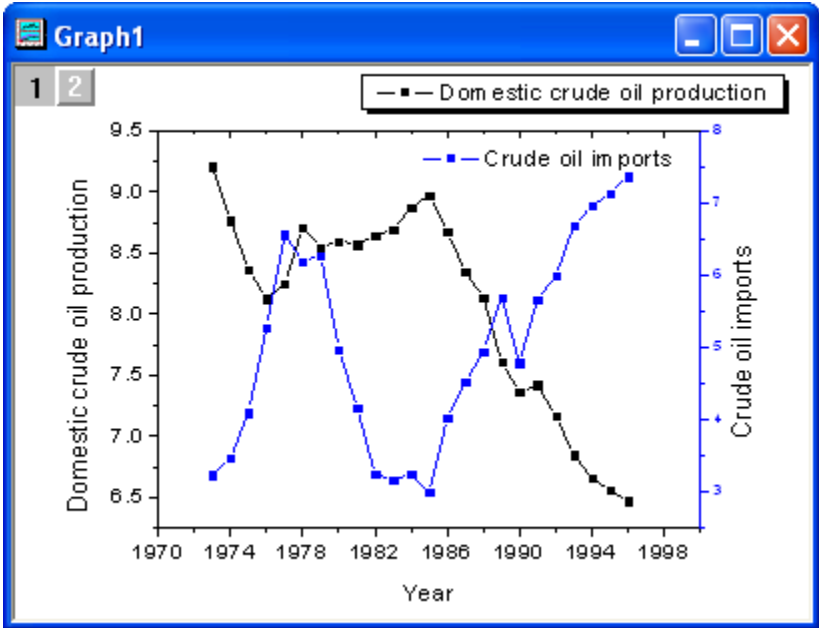
8. By default, Origin displays the tick labels for time on the X-axis in MM/DD/YYYY format. Double-click the X-axis to open an X-Axis Properties dialog box. On the **Scale** tab, change the scale from **1/1/1970** to **1/1/2000**.



On the **Tick Labels** tab, change the **Display** to year only.



Then we get:

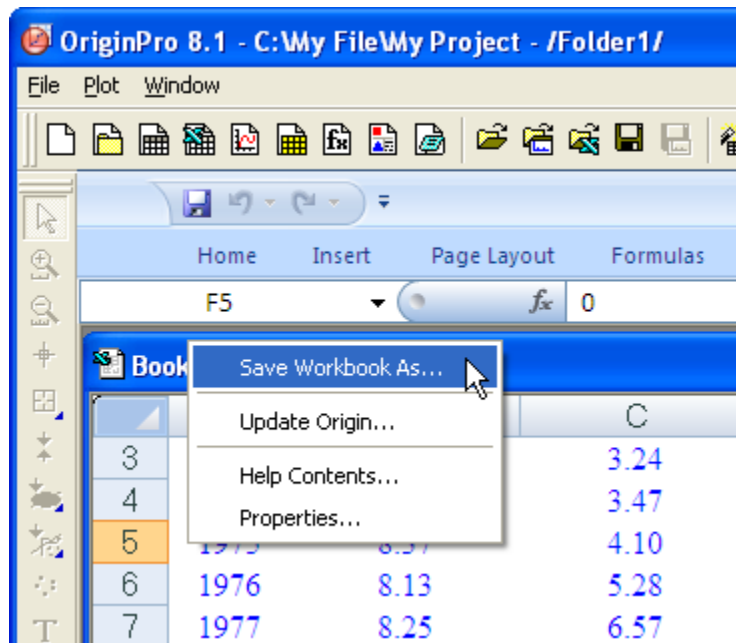


After opening Excel in Origin, if you switch to another Origin window, a toolbar spacer is visible where the Excel menu used to be. Right-click and you can select **Hide Toolbar Spacer** or **Hide Toolbar Spacer Always**.

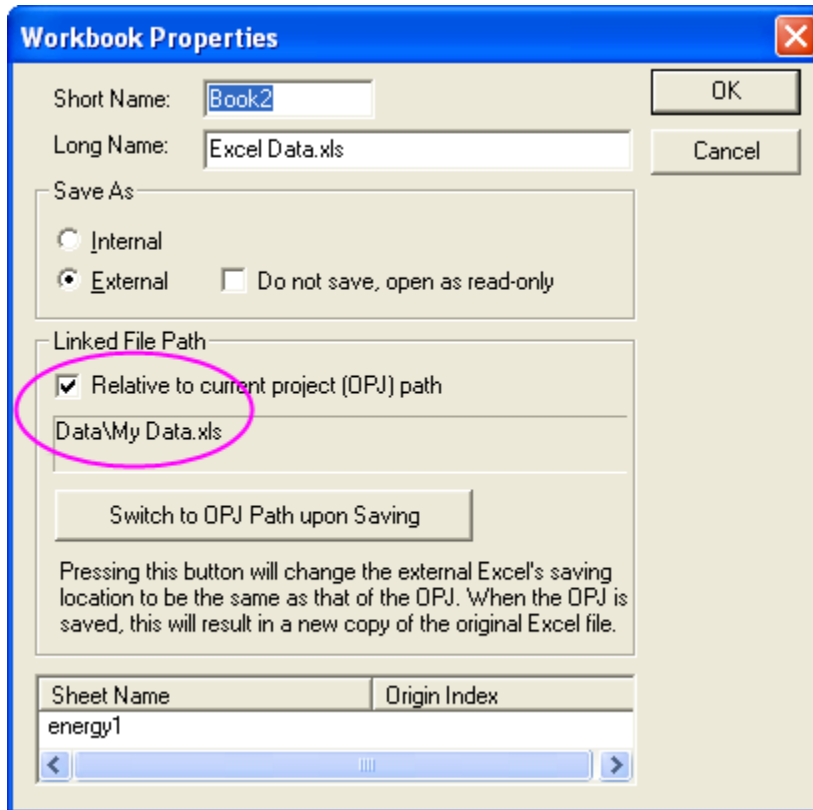
Setting External Excel File Path Relative to OPJ Path

The Origin Project can contain an Excel window which is linked to an external Excel file. It may then be beneficial to save the Excel file in the same folder as the Origin project, or in a subfolder under the Origin project folder, which will then make the two files more portable, as seen in the following steps:

1. Close Excel if it is running.
2. Perform the steps under the **Open Excel File in Origin** section above, and (optionally) create a plot.
3. First save the OPJ to some folder location such as "C:\My Files\My Project.opj".
4. Now right-click on the Excel window and select **Save Workbook As** and save it in a (new) subfolder under the OPJ save location, such as "C:\My Files\Data\My Data.xls".



- Right-click again on the Excel window title and select **Properties**, and then check the box that says **Relative to current project (opj) path**. Note that the Excel file path in the box below changes to the relative path "Data\My Data.xls".



- Save the OPJ again. Now you can copy the entire subfolder structure, starting from where the OPJ is saved, and put it on an external memory device (i.e., a memory stick or similar), or zip the entire folder structure. When taken to another computer and opened, Origin will look relative to the OPJ path to find the Excel file.



If your Excel file is in a different location and you want to save it to the same path as the OPJ, you can open the Excel file in Origin, then right-click the title, select **Properties** and click the **Switch to OPJ path upon Saving** button. On saving the OPJ, the Excel file will be copied from its current location to the same path where the OPJ is saved.