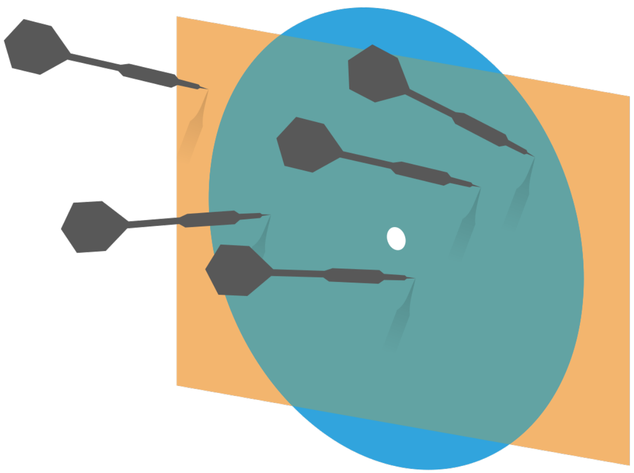
Probability and Statistics for Computer Science



"All models are wrong, but some models are useful"--- George Box

Credit: wikipedia

Hongye Liu, Teaching Assistant Prof, CS361, UIUC, 04.22.2021

Last time

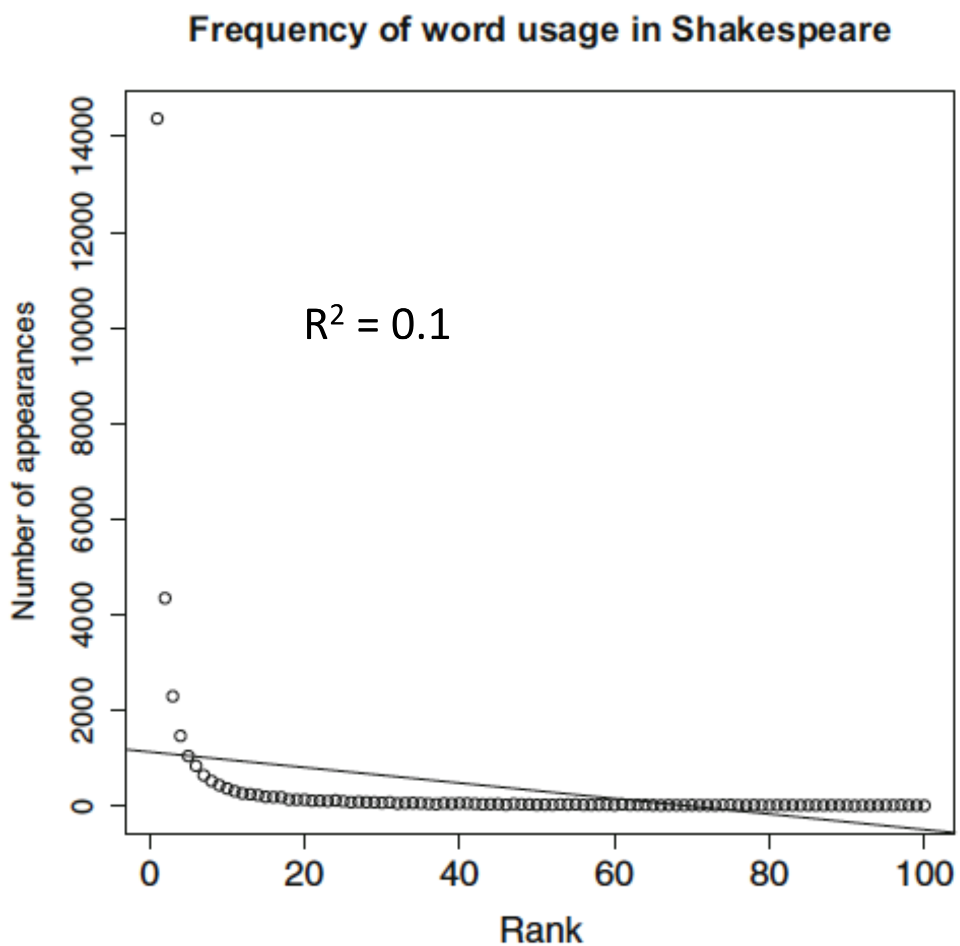
- Linear regression
  - The problem
  - The least square solution
  - The training and prediction
  - The R-squared for the evaluation of the fit.

Objectives

- Linear regression (cont.)
  - Modeling non-linear relationship with linear regression
  - Outliers and over-fitting issues
  - Regularized linear regression/Ridge regression
- Nearest neighbor regression

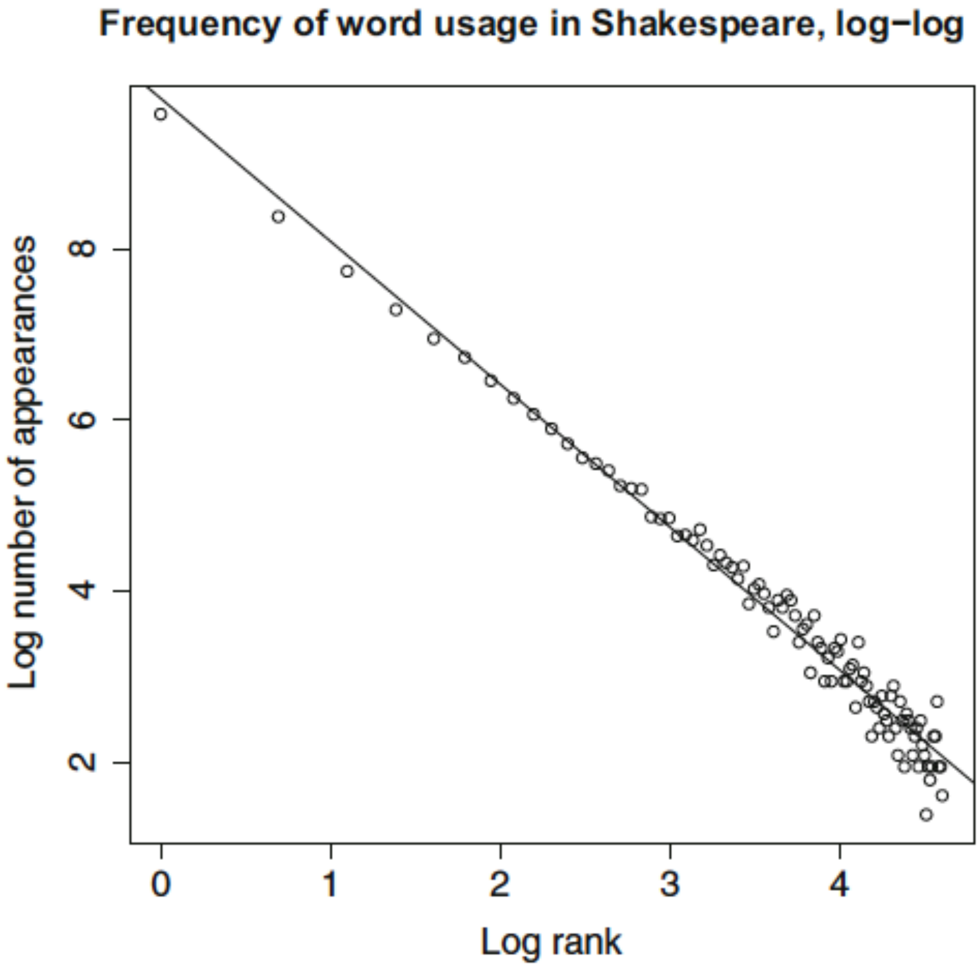
What if the relationship between variables is non-linear?

- A linear model will not produce a good fit if the dependent variable is **not** linear combination of the explanatory variables



Transforming variables could allow linear model to model non-linear relationship

- In the word- frequency example, log-transforming both variables would allow a linear model to fit the data well.



More example: Data of fish in a Finland lake

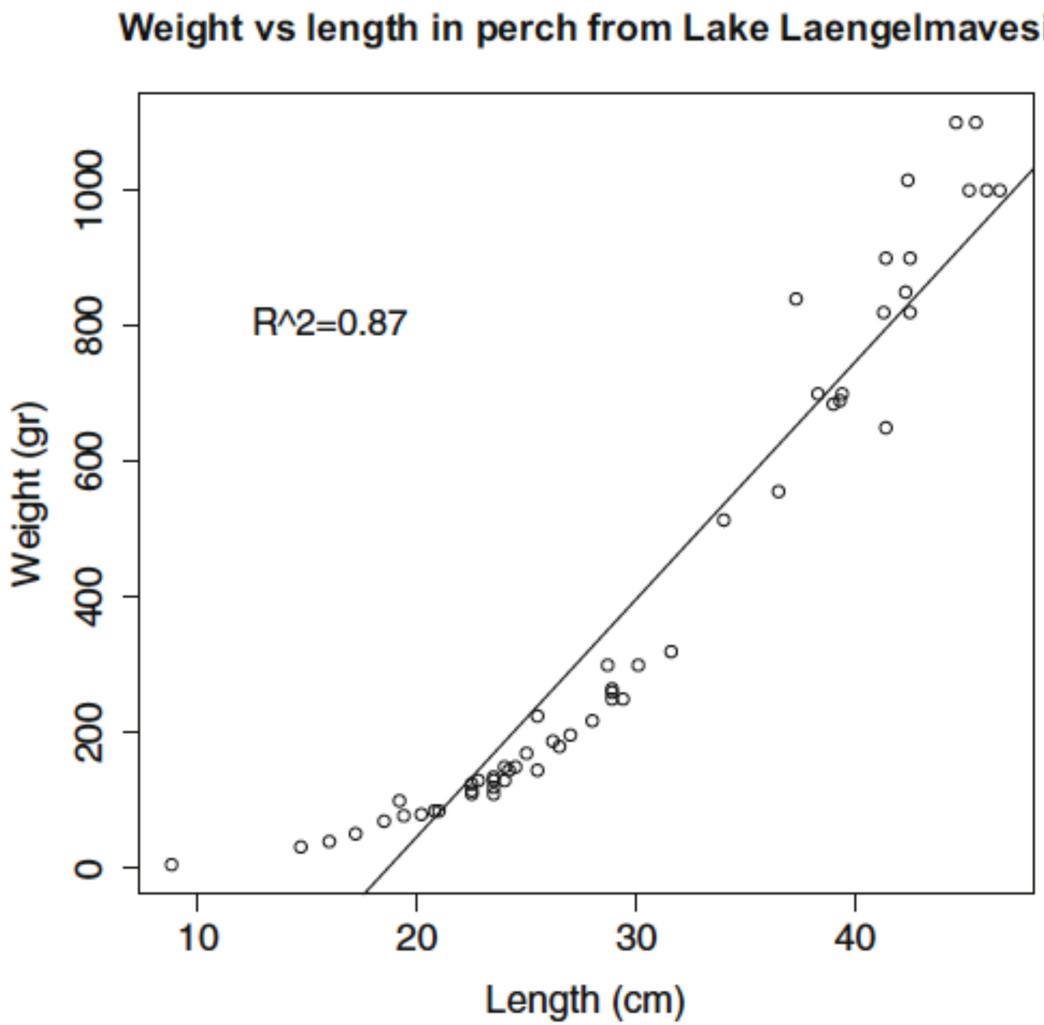
- Perch (a kind of fish) in a lake in Finland, 56 data observations
- Variables include: Weight, Length, Height, Width
- In order to illustrate the point, let's model **Weight** as the dependent variable and the **Length** as the explanatory variable.

Yellow Perch

Is the linear model fine for this data?

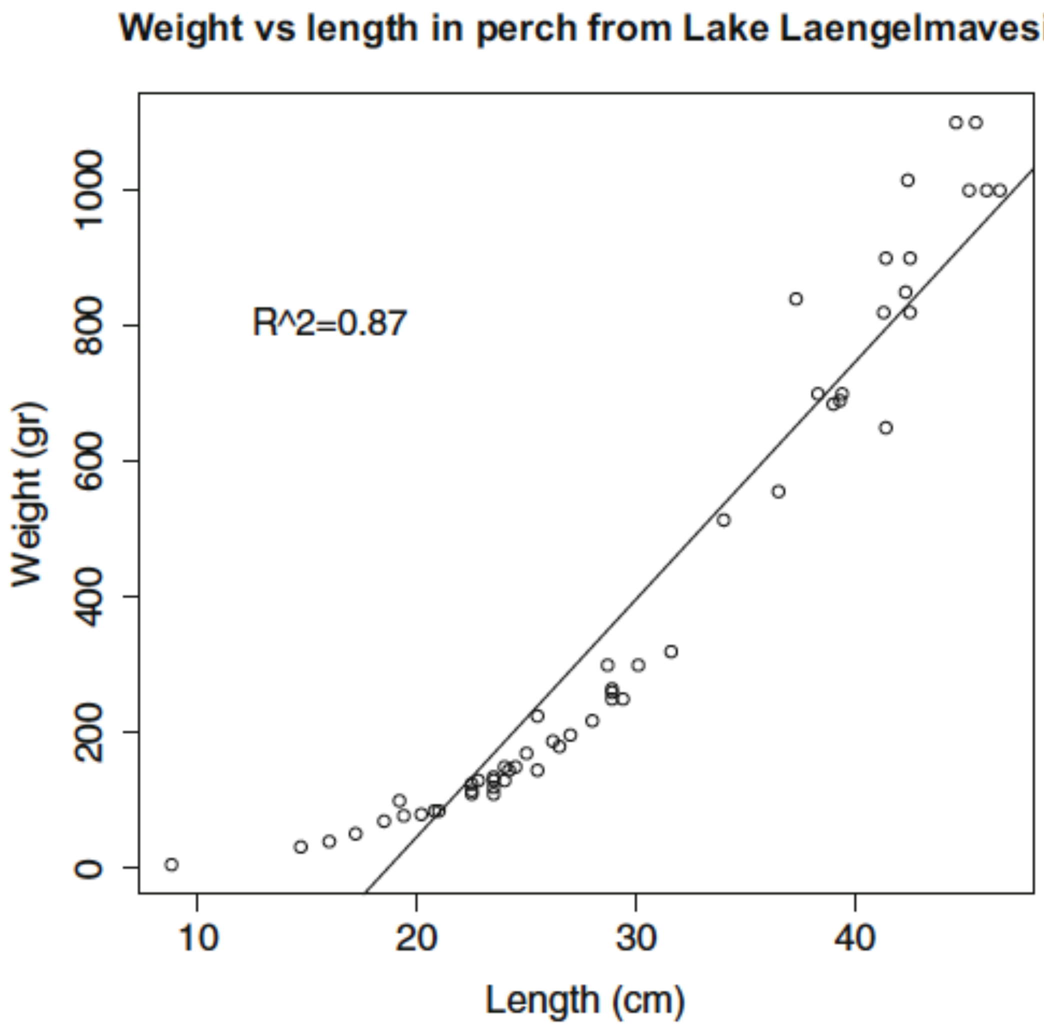
A. YES

B. NO

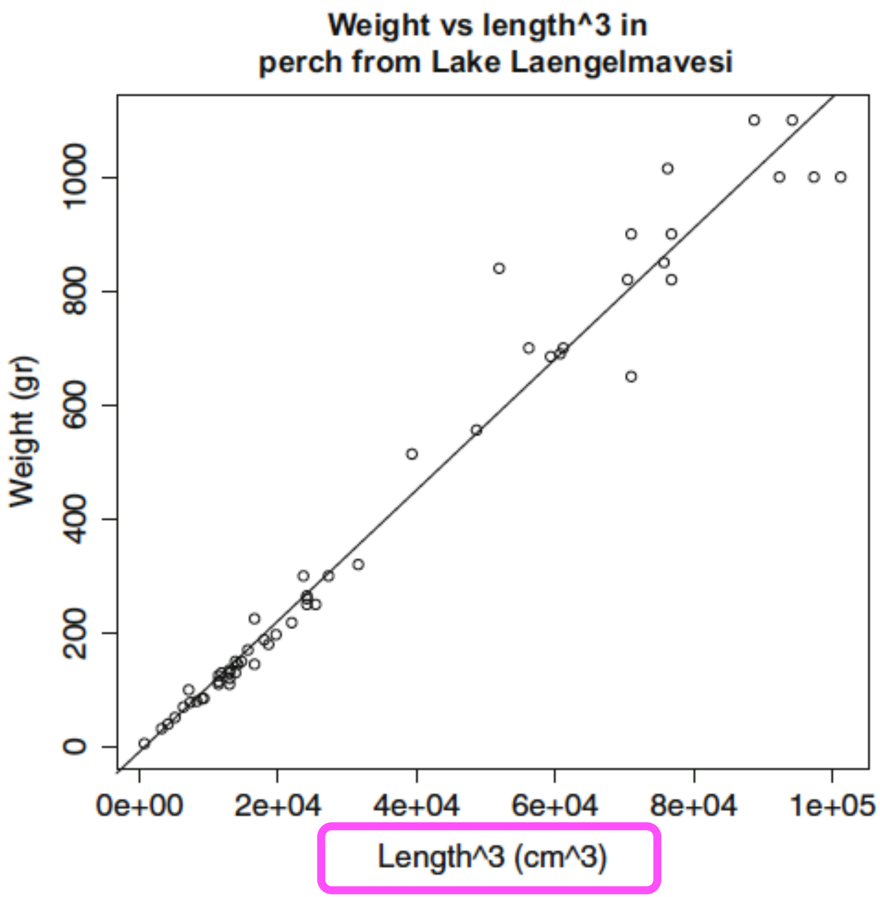


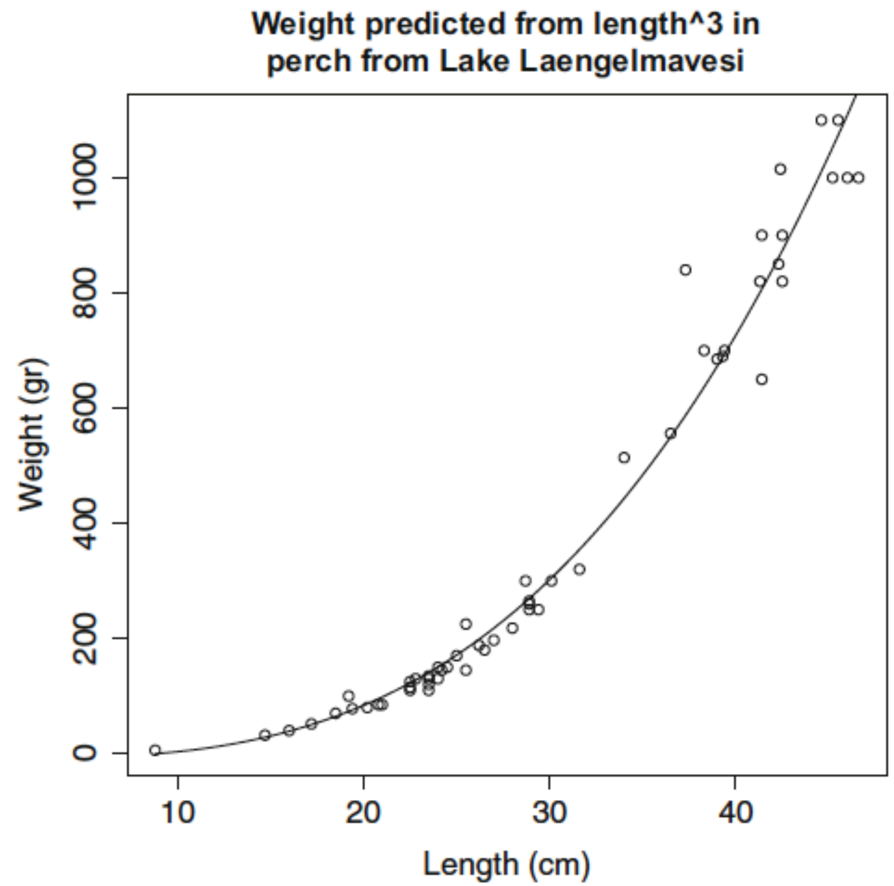
Is the linear model fine for this data?

- R-squared is 0.87 may suggest the model is OK
- But the trend of the data suggests non-linear relationship
- Intuition tells us length is not linear to weight given fish is 3-dimensional
- We can do better!



Transforming the explanatory variables

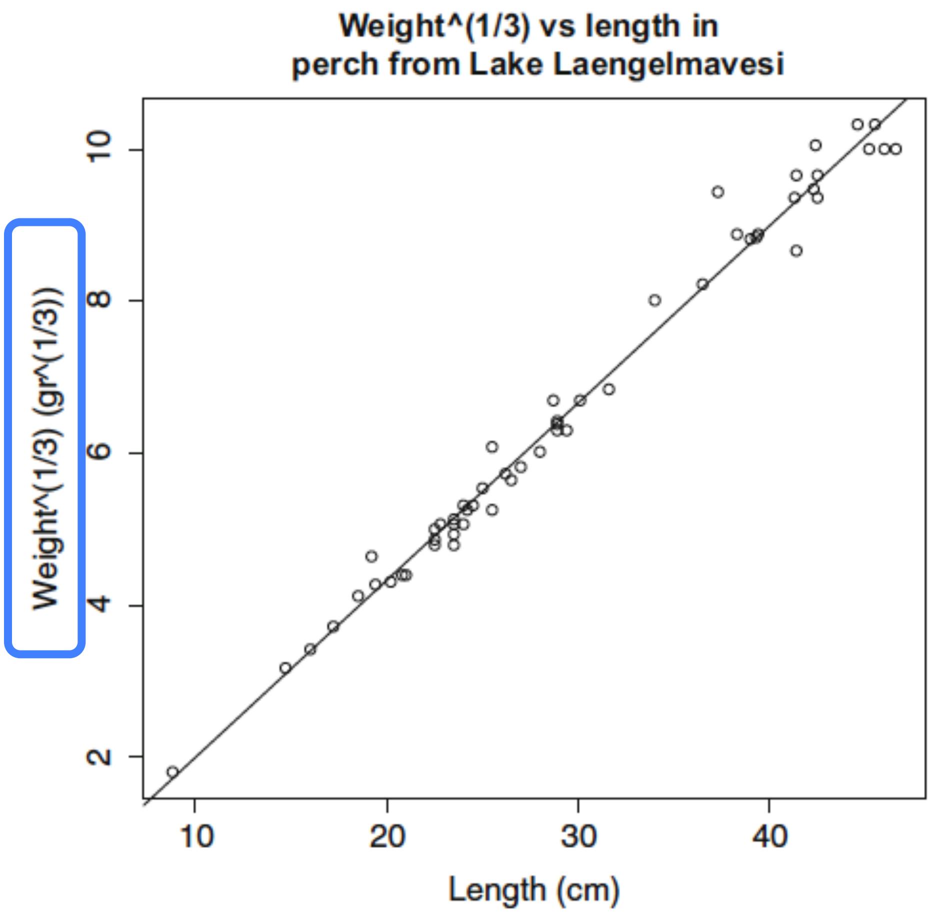


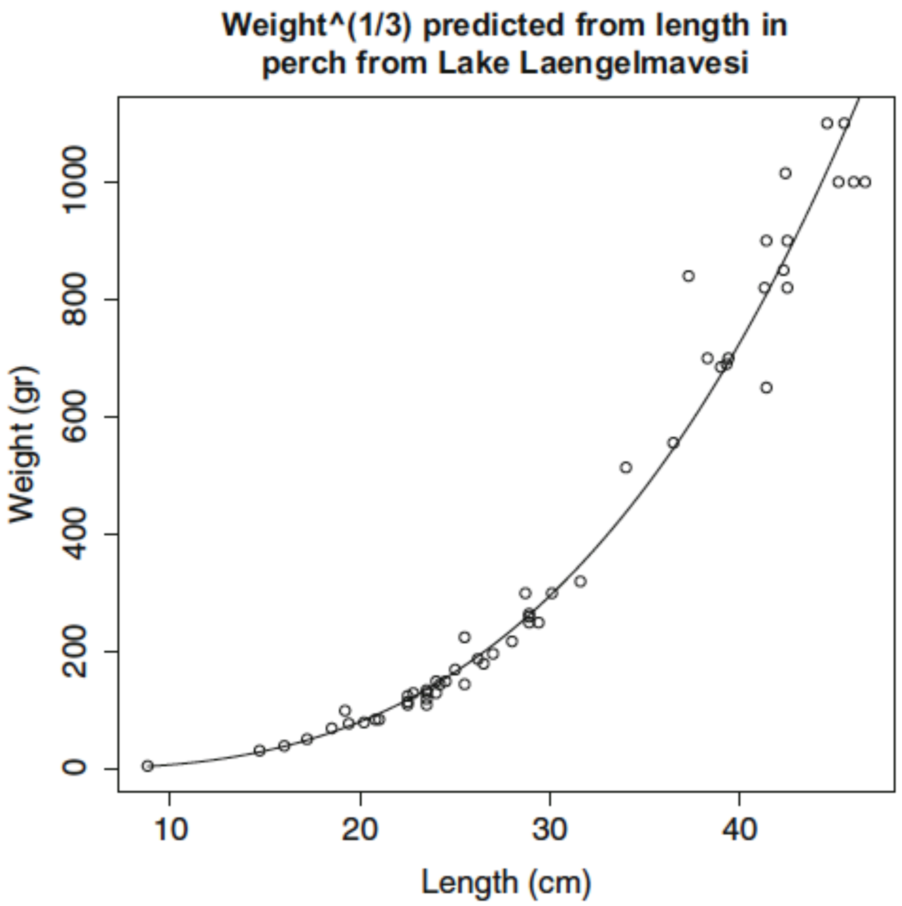


Q. What are the matrix X and y?

1	$Length^3$	Weight

Transforming the dependent variables





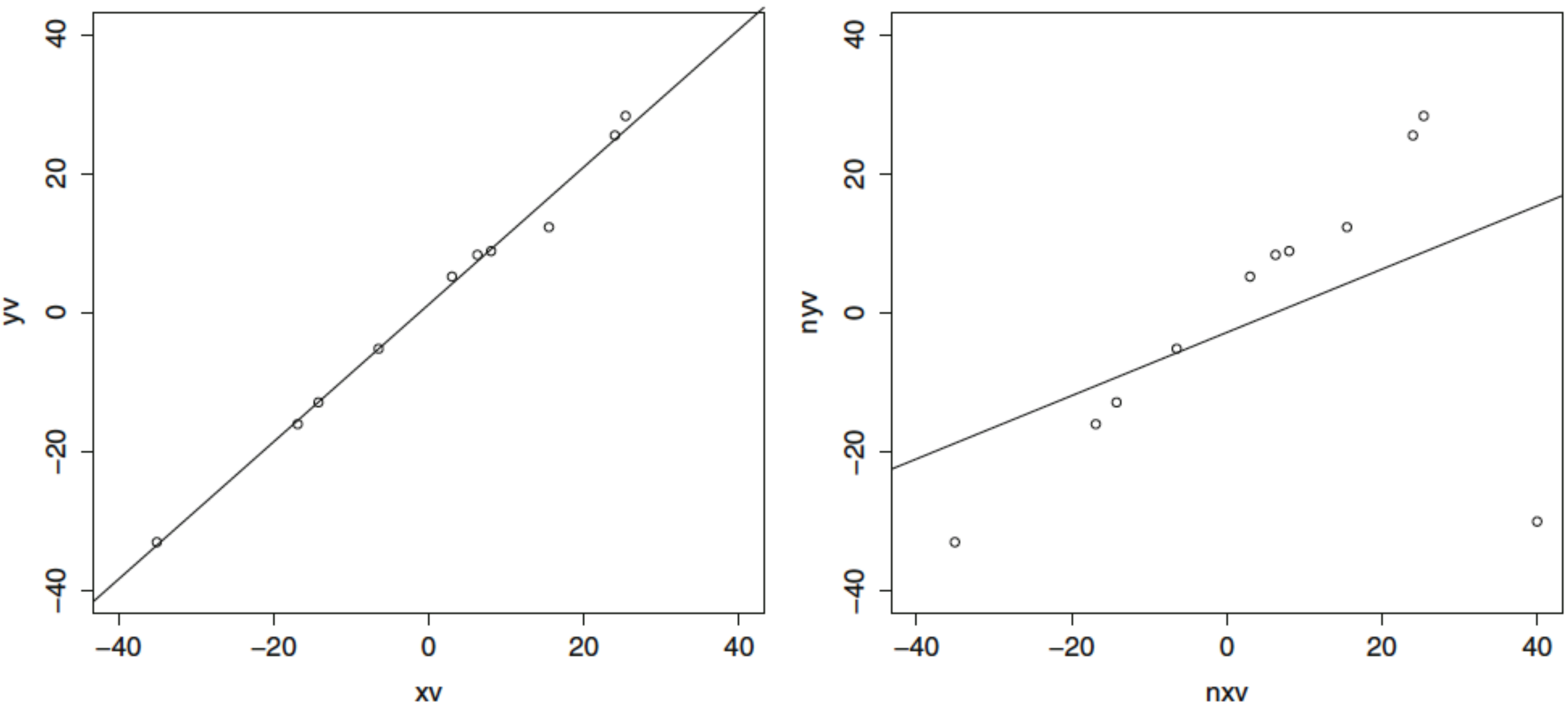
What is the model now?

What are the matrix X and y?

1	Length	$\sqrt[3]{w}$

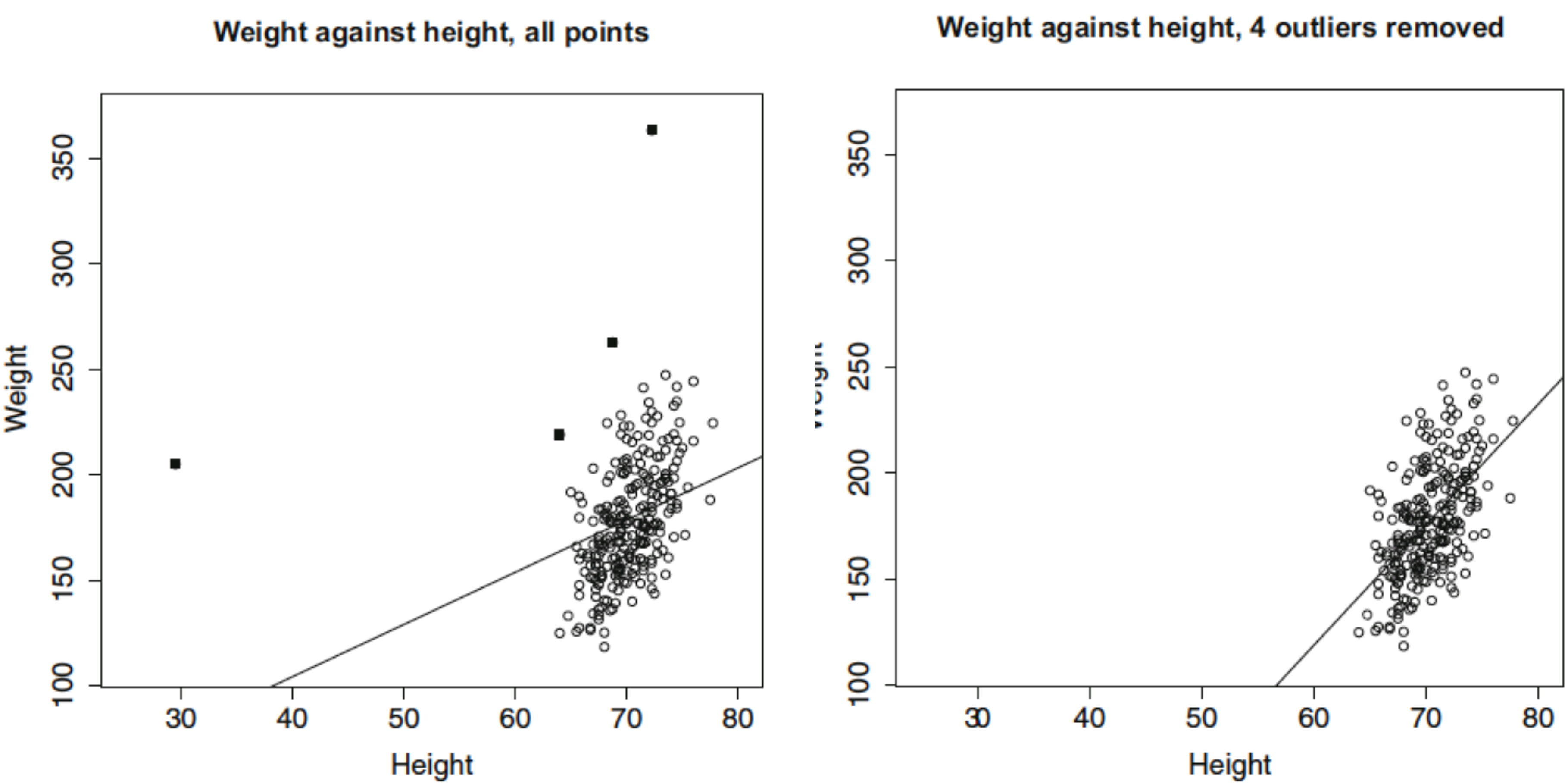
Effect of outliers on linear regression

- Linear regression is sensitive to outliers



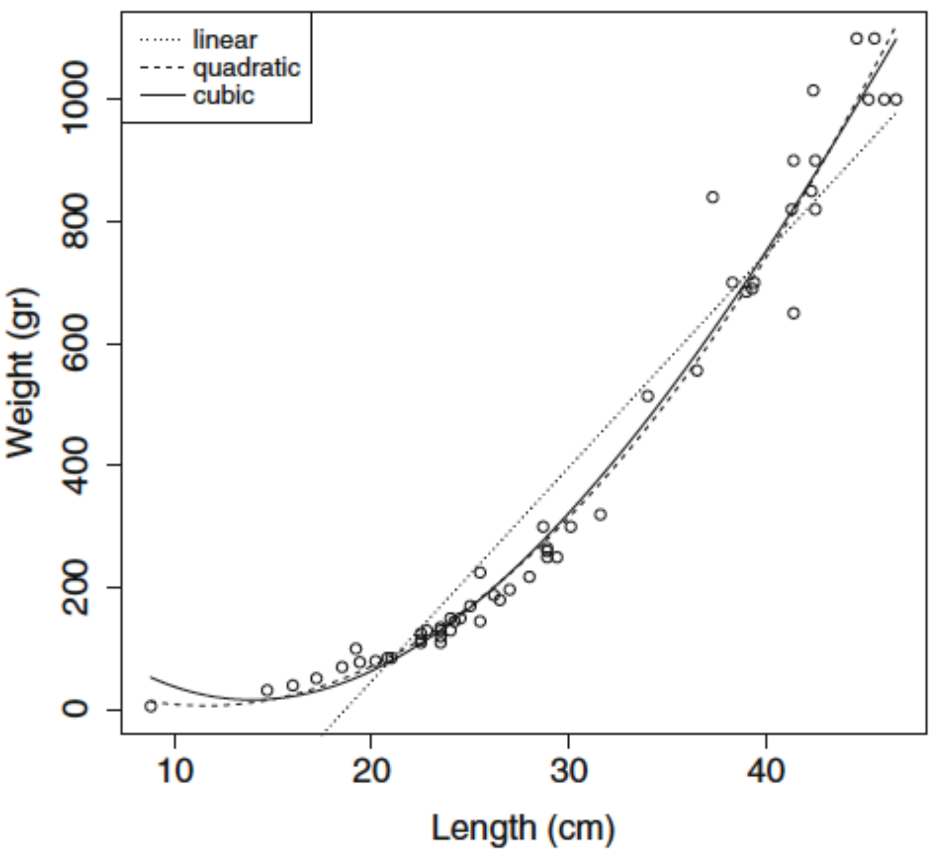
Effect of outliers: body fat example

- Linear regression is sensitive to outliers

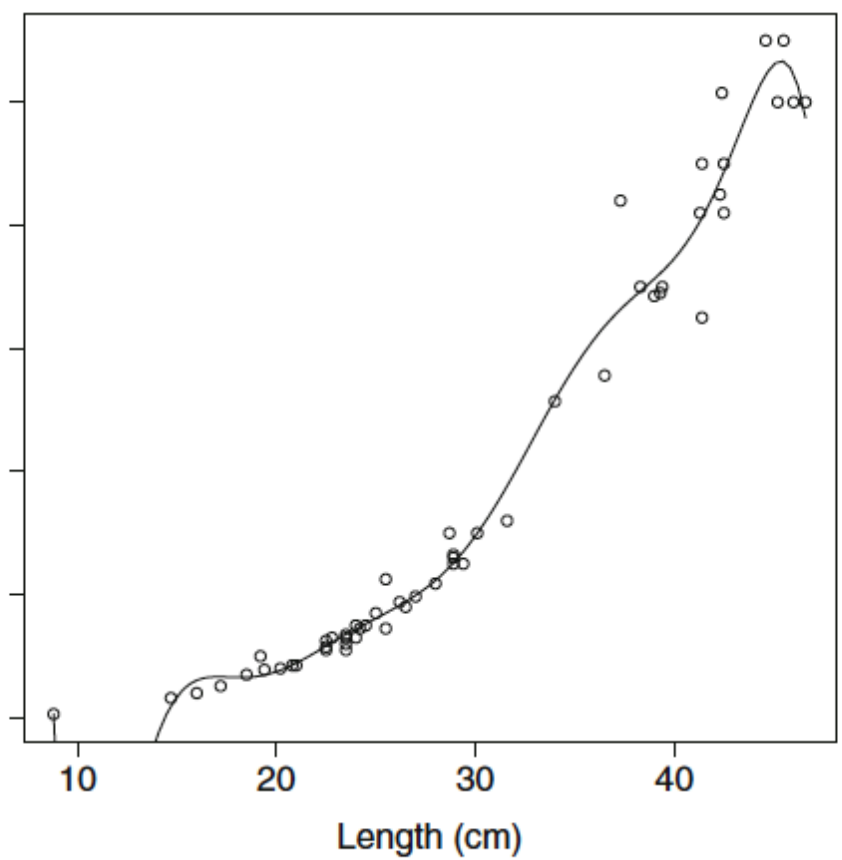


Over-fitting issue: example of using too many power transformations

Weight vs length in perch from Lake Laengelmavesi, three models.



Weight vs length in perch from Lake Laengelmavesi, all powers up to 10.



Avoiding over-fitting

- **Method 1: validation**
  - Use a validation set to choose the transformed explanatory variables
  - The difficulty is the number of combination is exponential in the number of variables.
- **Method 2: regularization**
  - Impose a penalty on complexity of the model during the training
  - Encourage smaller model coefficients
- We can use validation to select regularization parameter λ

Regularized linear regression

- In ordinary least squares, the cost function is $\|\mathbf{e}\|^2$:

$$\|\mathbf{e}\|^2 = \|\mathbf{y} - X\boldsymbol{\beta}\|^2 = (\mathbf{y} - X\boldsymbol{\beta})^T(\mathbf{y} - X\boldsymbol{\beta})$$

- In regularized least squares, we add a penalty with a weight parameter λ (λ>0):

$$\|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda\frac{\|\boldsymbol{\beta}\|^2}{2} = (\mathbf{y} - X\boldsymbol{\beta})^T(\mathbf{y} - X\boldsymbol{\beta}) + \lambda\frac{\boldsymbol{\beta}^T\boldsymbol{\beta}}{2}$$

Training using regularized least squares

- Differentiating the cost function and setting it to zero, one gets:

$$(X^T X + \lambda I)\boldsymbol{\beta} - X^T \mathbf{y} = 0$$

- $(X^T X + \lambda I)$ is always invertible, so the regularized least squares estimation of the coefficients is:

$$\widehat{\boldsymbol{\beta}} = (X^T X + \lambda I)^{-1} X^T \mathbf{y}$$

Why is the regularized version always invertible?

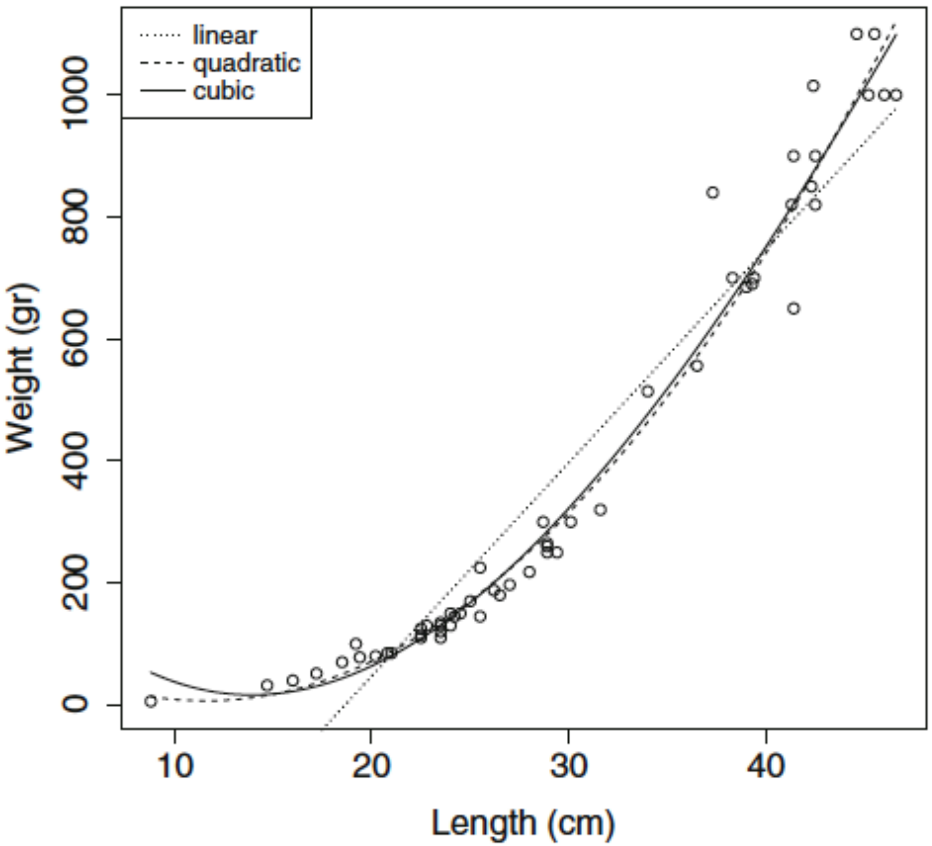
Prove: $(X^T X + \lambda I)$ is invertible (λ>0, λ is not the eigenvalue).

$$f^T A f \geq 0$$

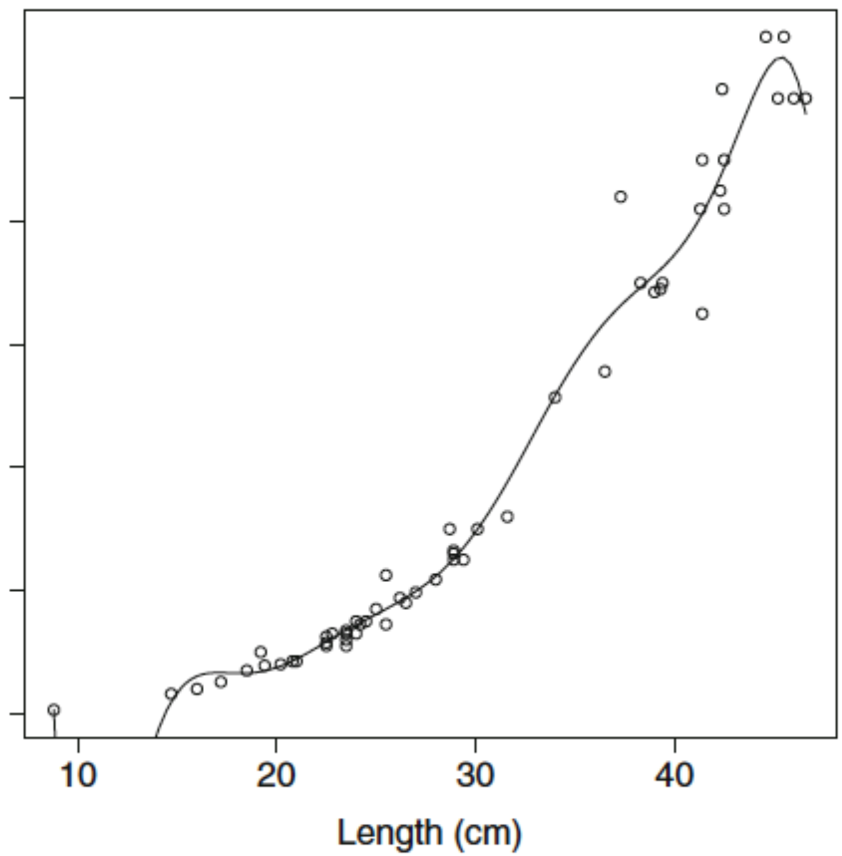
$$f^T A f > 0$$

Over-fitting issue: example from using too many power transformations

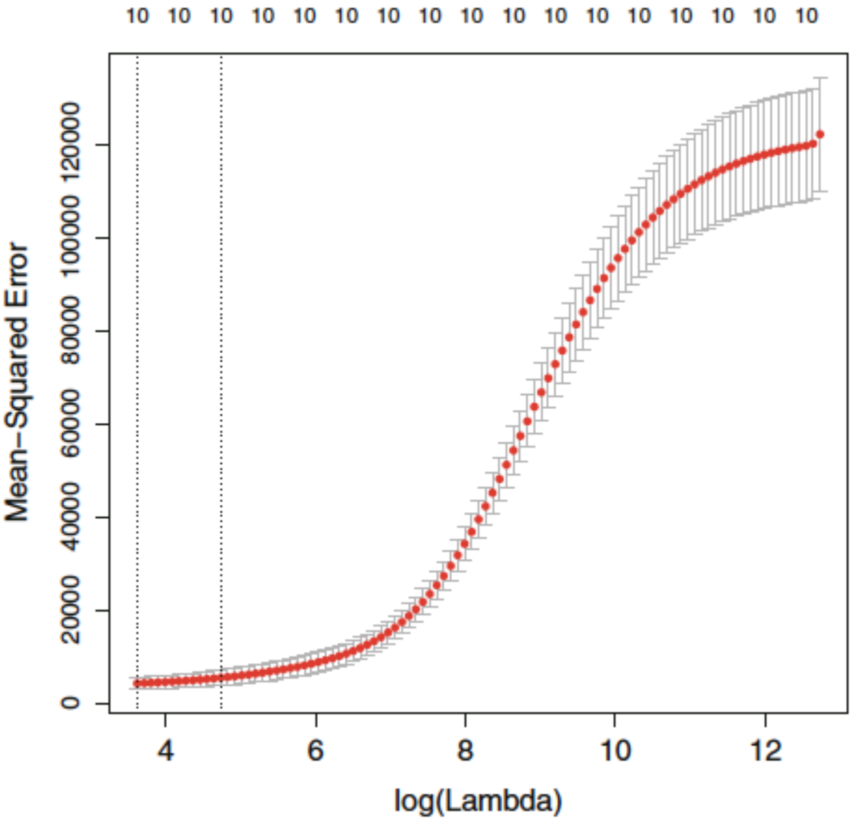
Weight vs length in perch from Lake Laengelmavesi, three models.

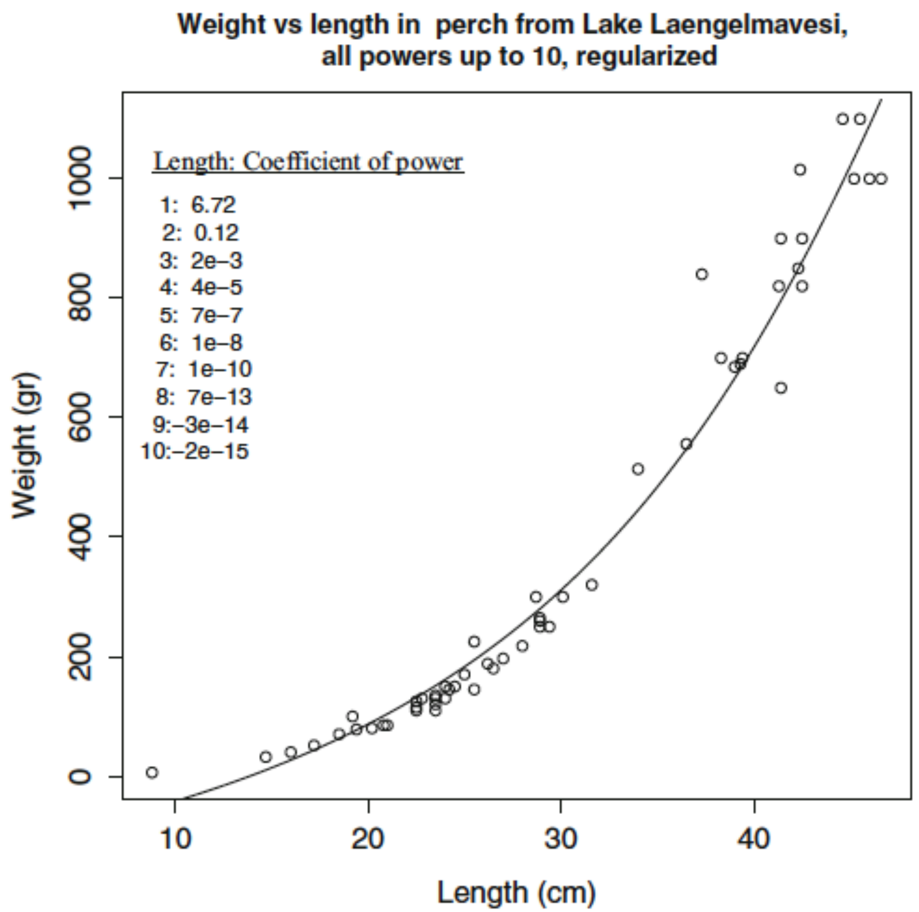


Weight vs length in perch from Lake Laengelmavesi, all powers up to 10.



Choosing lambda using cross-validation





Q. Can we use the R-squared to evaluate the regularized model correctly?

A. YES
B. NO
C. YES and NO

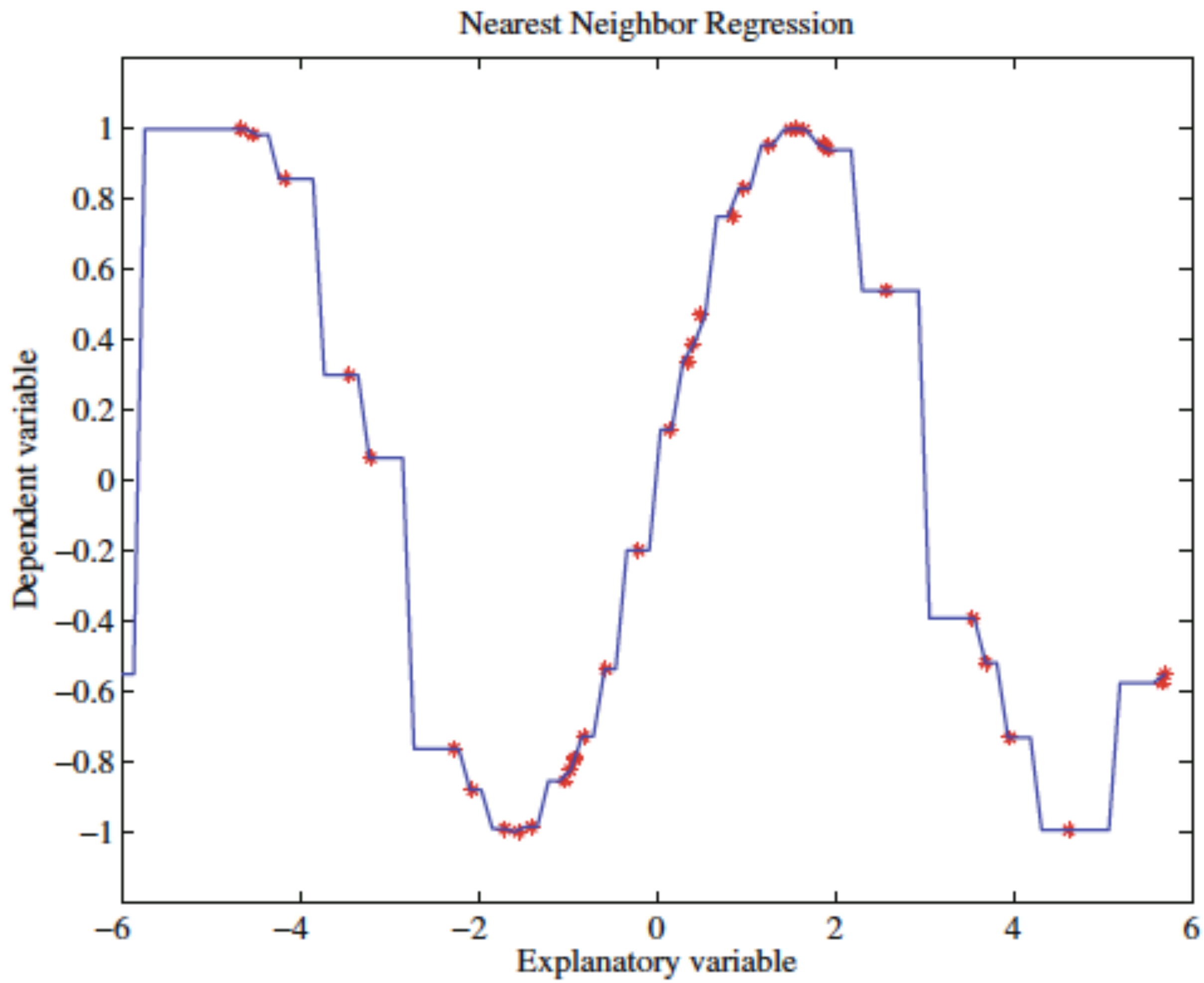
Nearest neighbor regression

- In addition to linear regression and generalize linear regression models, there are methods such as **Nearest neighbor regression** that do not need much training for the model parameters.
- When there is plenty of data, nearest neighbors regression can be used effectively

K nearest neighbor regression with k=1

The idea is very similar to k-nearest neighbor classifier, but the regression model predicts numbers

K=1 gives piecewise constant predictions



K nearest neighbor regression with weights

The goal is to predict y_0^p from $\mathbf{x}_0$ using a training set $\{(\mathbf{x}, y)\}$

- Let $\{(\mathbf{x}_j, \mathbf{y}_j)\}$ be the set of k items in the training data set that are closest to $\mathbf{x}_0$.
- Prediction is the following:

$$\mathbf{y}_0^p = \frac{\sum_j \mathbf{w}_j \mathbf{y}_j}{\sum_j \mathbf{w}_j}$$

Where $\mathbf{w}_j$ are weights that drop off as $\mathbf{x}_j$ gets further away from $\mathbf{x}_0$.

Choose different weights functions for KNN regression

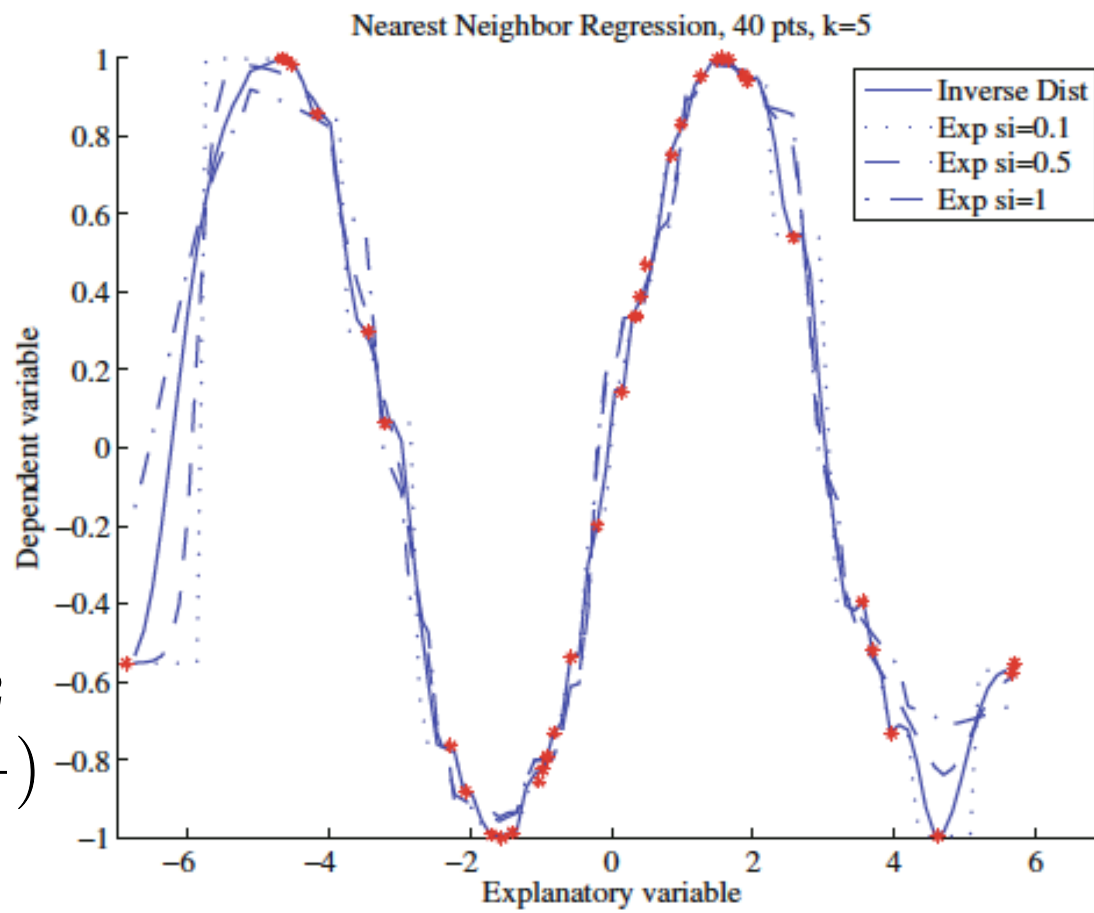
$$\mathbf{y}_0^p = \frac{\sum_j \mathbf{w}_j \mathbf{y}_j}{\sum_j \mathbf{w}_j}$$

- Inverse distance

$$\mathbf{w}_j = \frac{1}{\|\mathbf{x}_0 - \mathbf{x}_j\|}$$

- Exponential function

$$\mathbf{w}_j = exp(-\frac{\|\mathbf{x}_0 - \mathbf{x}_j\|^2}{2\sigma^2})$$



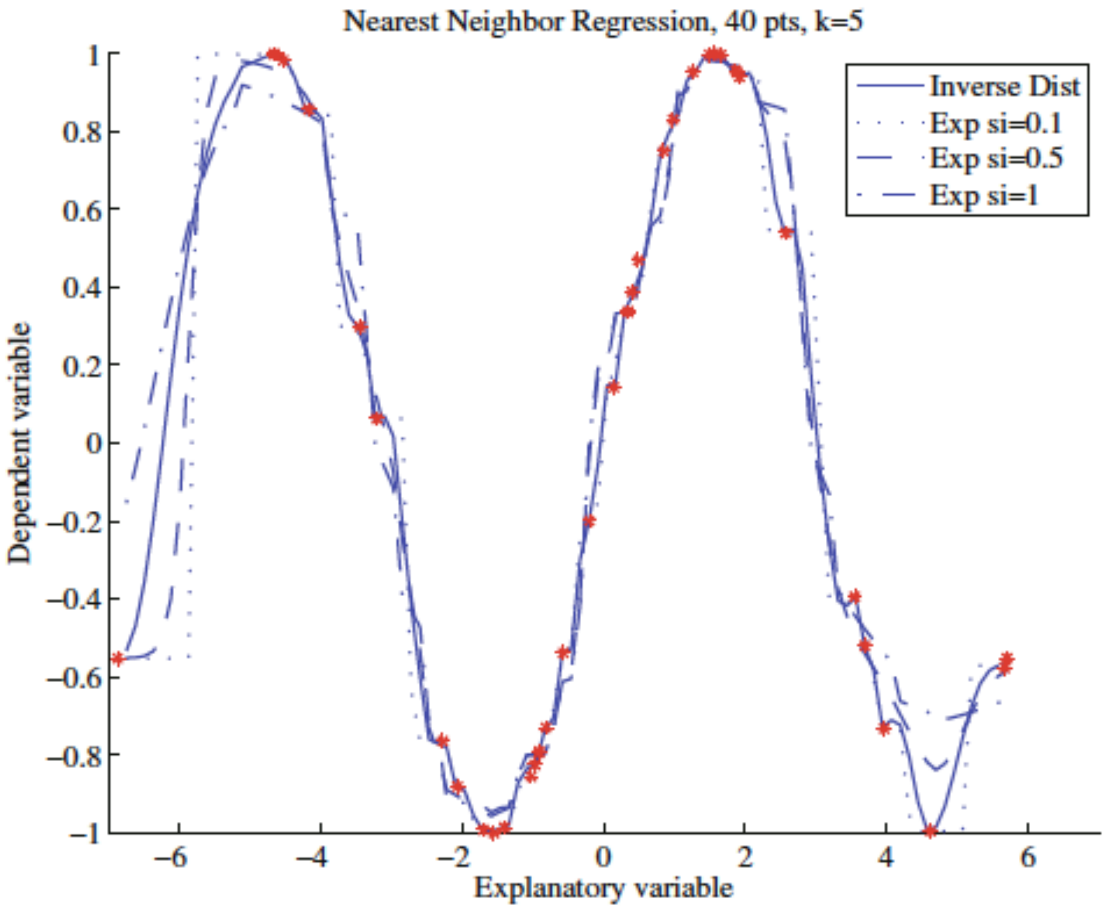
Evaluation of KNN models

- Which methods do you use to choose K and weight functions?

  A. Cross validation

  B. Evaluation of MSE

  C. Both A and B



The Pros and Cons of K nearest neighbor regression

- Pros:
  - The method is very intuitive and simple
  - You can predict more than numbers as long as you can define a similarity measure.
- Cons
  - The method doesn't work well for very high dimensional data
  - The model depends on the scale of the data

Additional References

- Robert V. Hogg, Elliot A. Tanis and Dale L. Zimmerman. "Probability and Statistical Inference"
- Kelvin Murphy, "Machine learning, A Probabilistic perspective"

See you next time

See You!