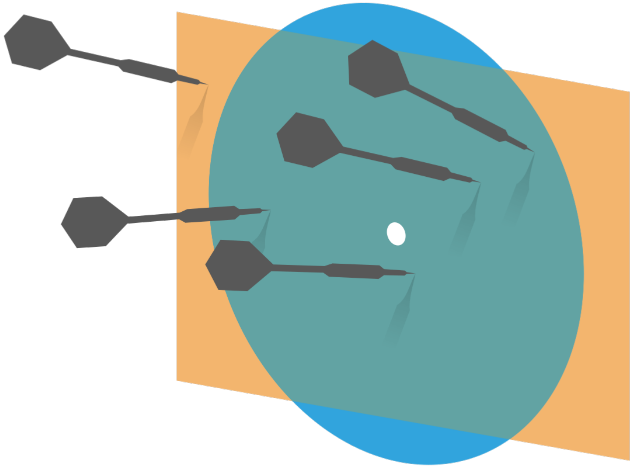
Probability and Statistics for Computer Science



Credit: wikipedia

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

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

Last time

- Linear regression $y = \beta_0 + \beta_1 x^{(1)} + \beta_2 x^{(2)} \cdots + \xi$
  - The problem $y = X\beta + e$
  - The least square solution $\hat{\beta} = (X^T X)^{-1} X^T y$
  - The training and prediction $y^P = X\hat{\beta}$
  - The R-squared for the evaluation of the fit. $y = X\beta + e$

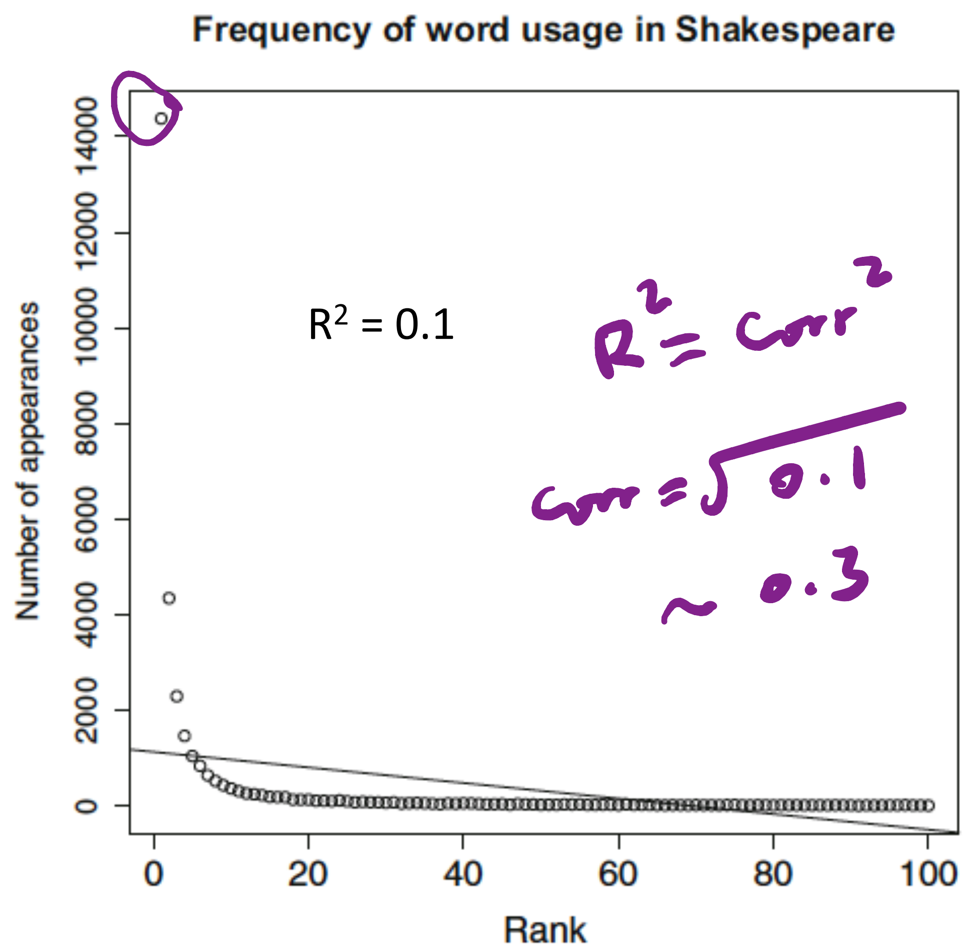
$$R^2 = \frac{var(X\hat{\beta})}{var(y)}$$

Objectives

- Linear regression (cont.)
  - Modeling non-linear relationship with linear regression → bigger scope
  - Outliers and over-fitting issues – better
  - 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



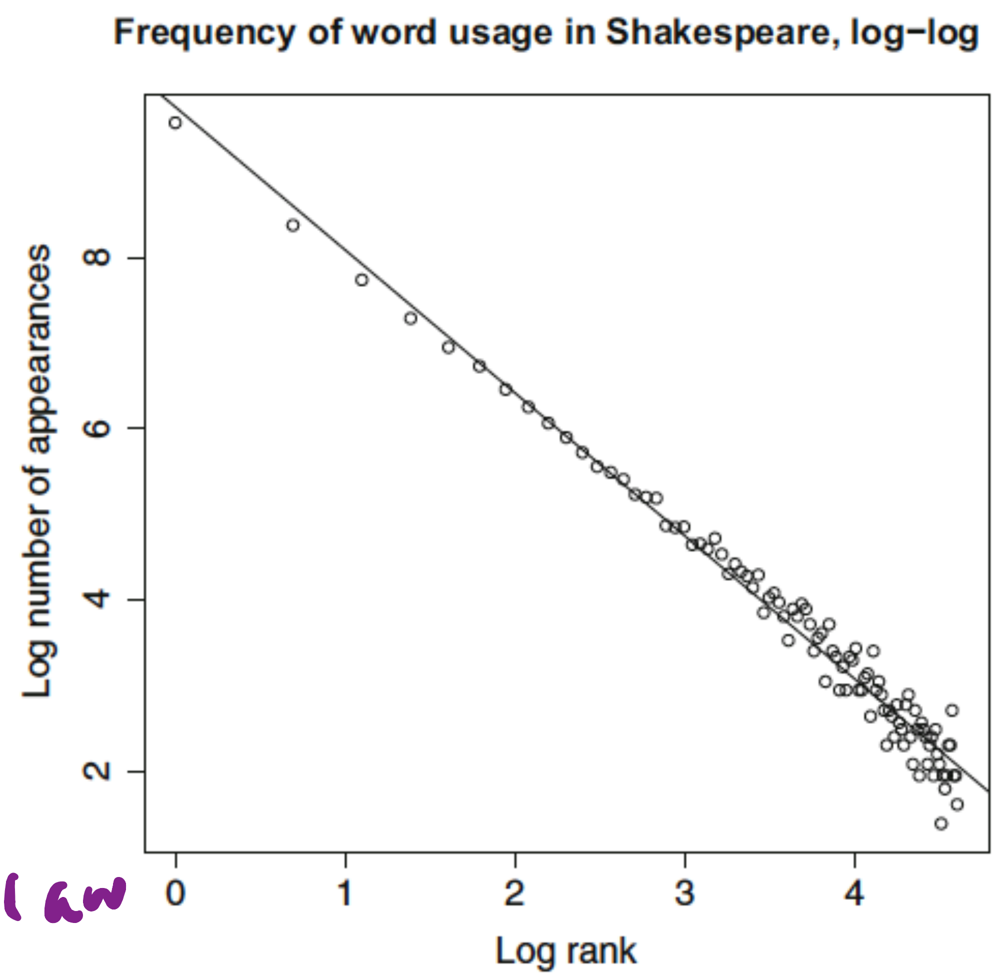
Transforming variables could allow linear model to model non-linear relationship

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

$$\log f = \beta_0 + \beta_1 \log r$$

$$f = c \cdot \left(\frac{1}{r}\right)^s$$

Zipf's law



More example: Data of fish in a Finland lake

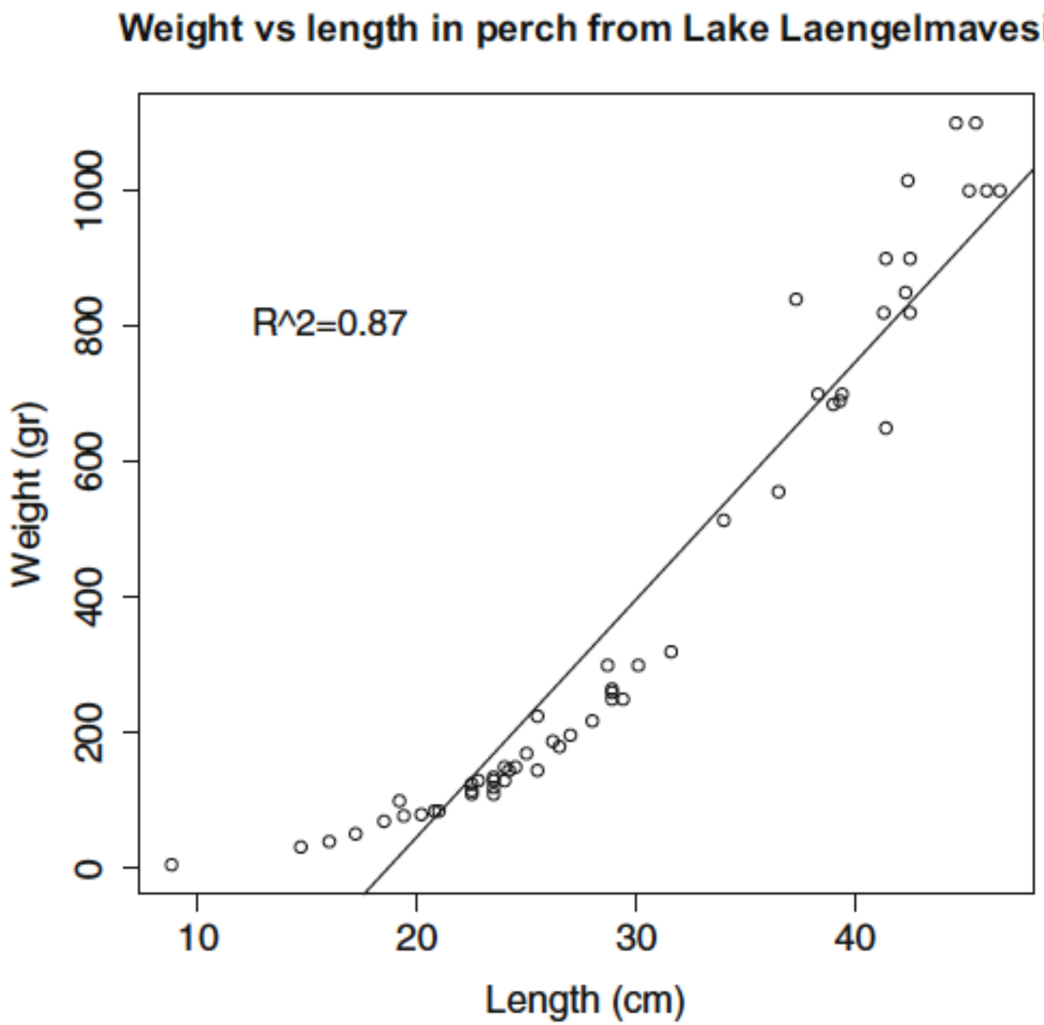
- Perch (a kind of fish) in a lake in Finland, 56 data observations
- Variables include: Weight, Length, Height, Width
- In order to illustrate the point, let's model **Weight** as the dependent variable and the **Length** as the explanatory variable.

Yellow Perch

Is the linear model fine for this data?

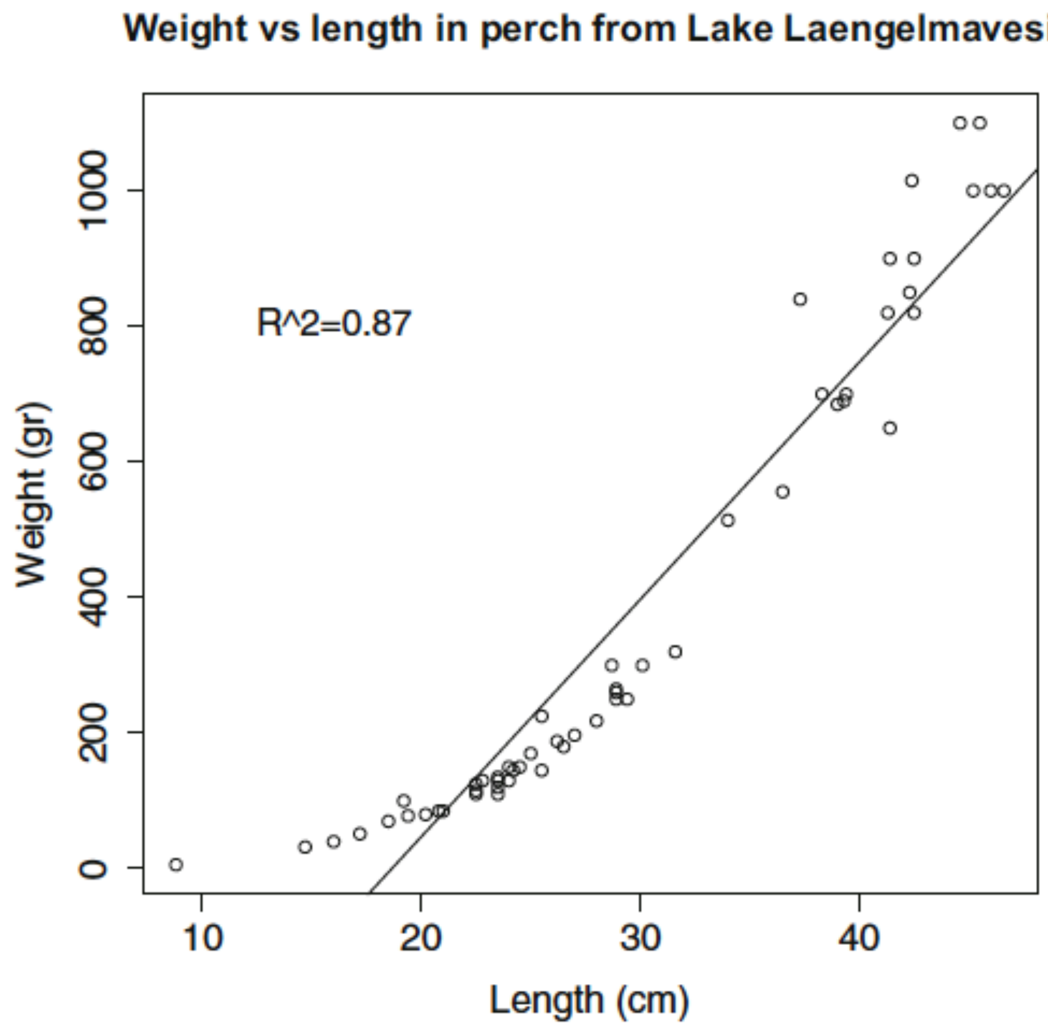
A. YES

B. NO

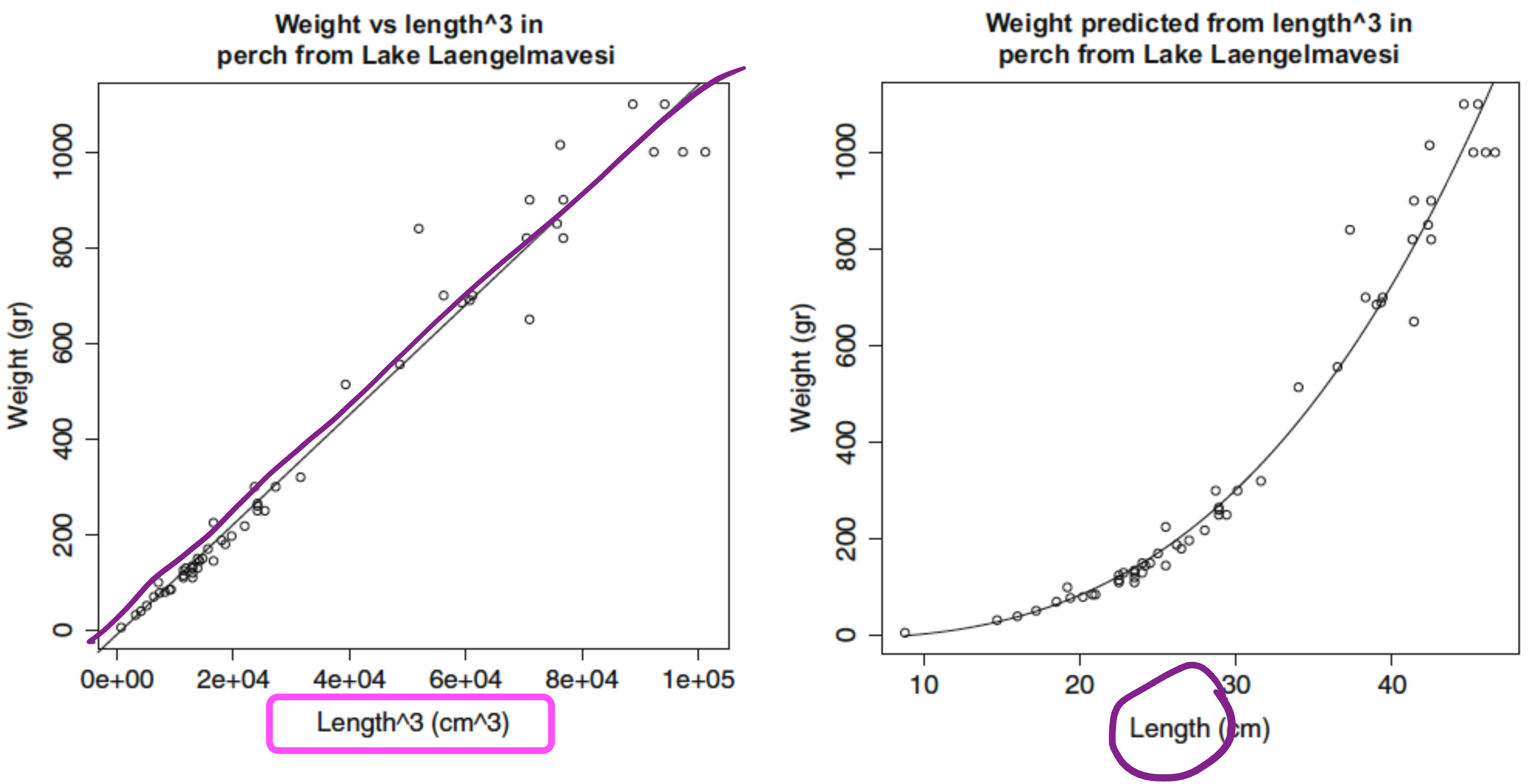


Is the linear model fine for this data?

- 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?

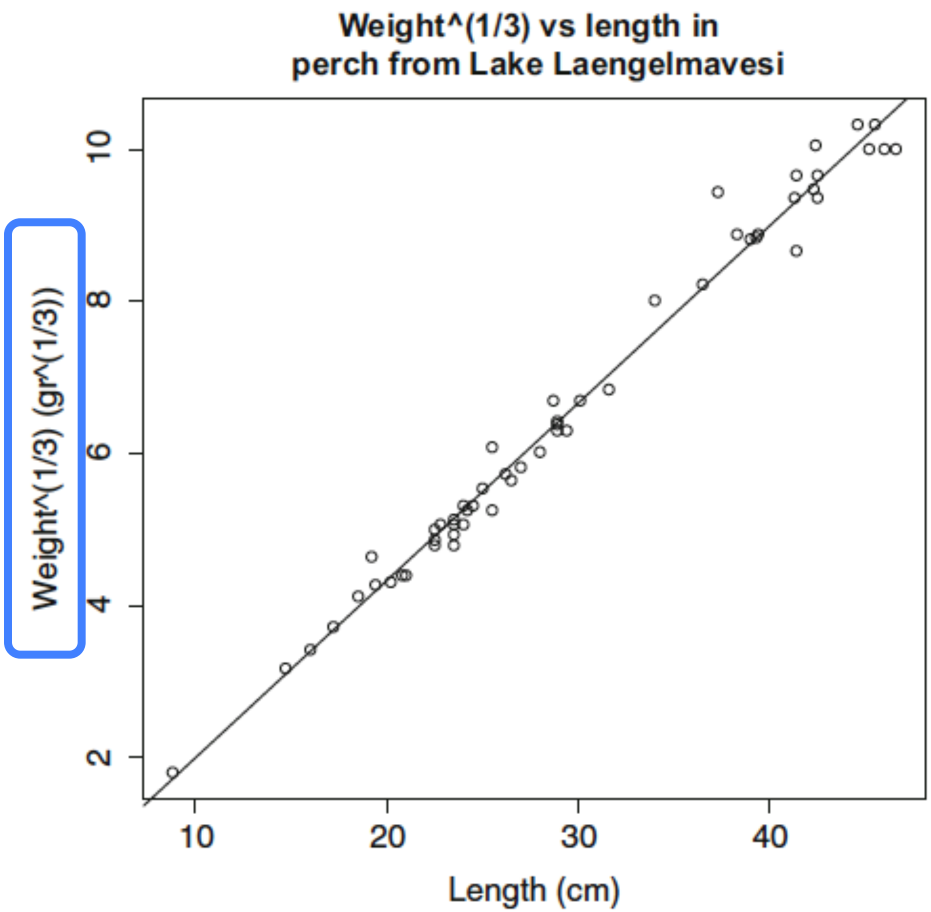
1	Length^3	Weight

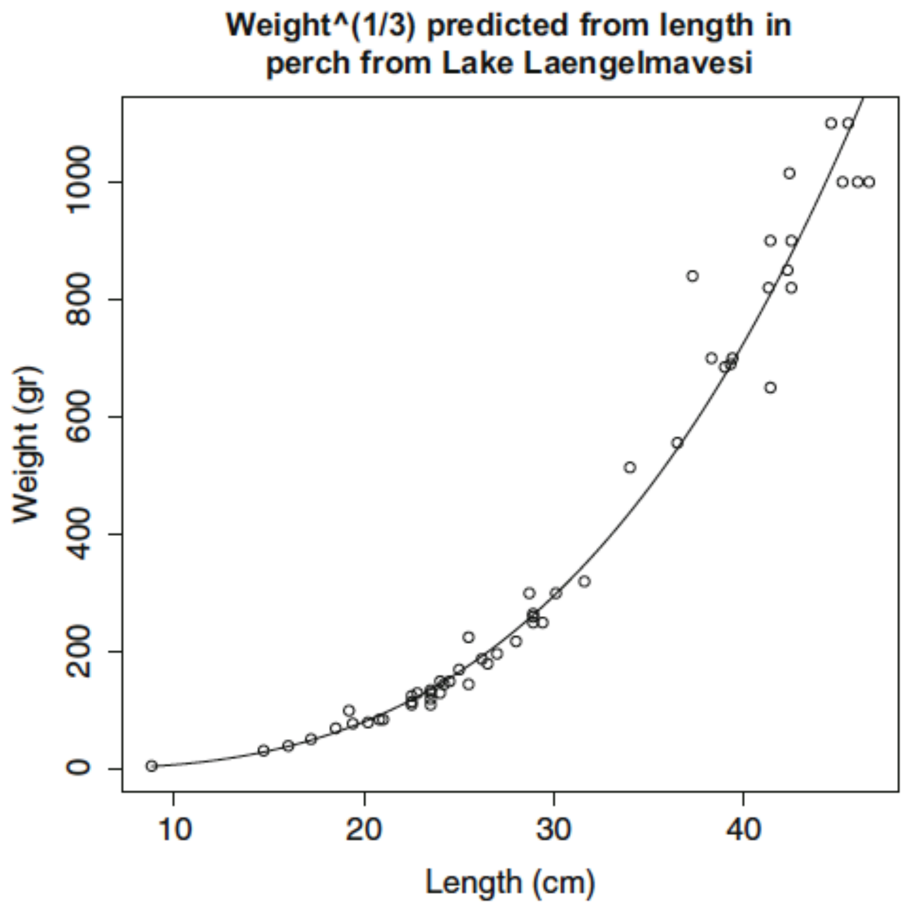
$$\begin{array}{ccc} & & \text{length} \\ X^0 = & \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix} & \end{array}$$

$$\rightarrow X^{new} = \begin{bmatrix} 1 & 8 \\ 1 & 27 \end{bmatrix}$$

$y = y^0$ (as before)

Transforming the dependent variables





What is the model now?

$$\sqrt[3]{y} = \beta_0 + \beta_1 x$$

$$y = \beta_0 + \beta_1 x^3$$

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_n x^n$$

What are the matrix X and y?

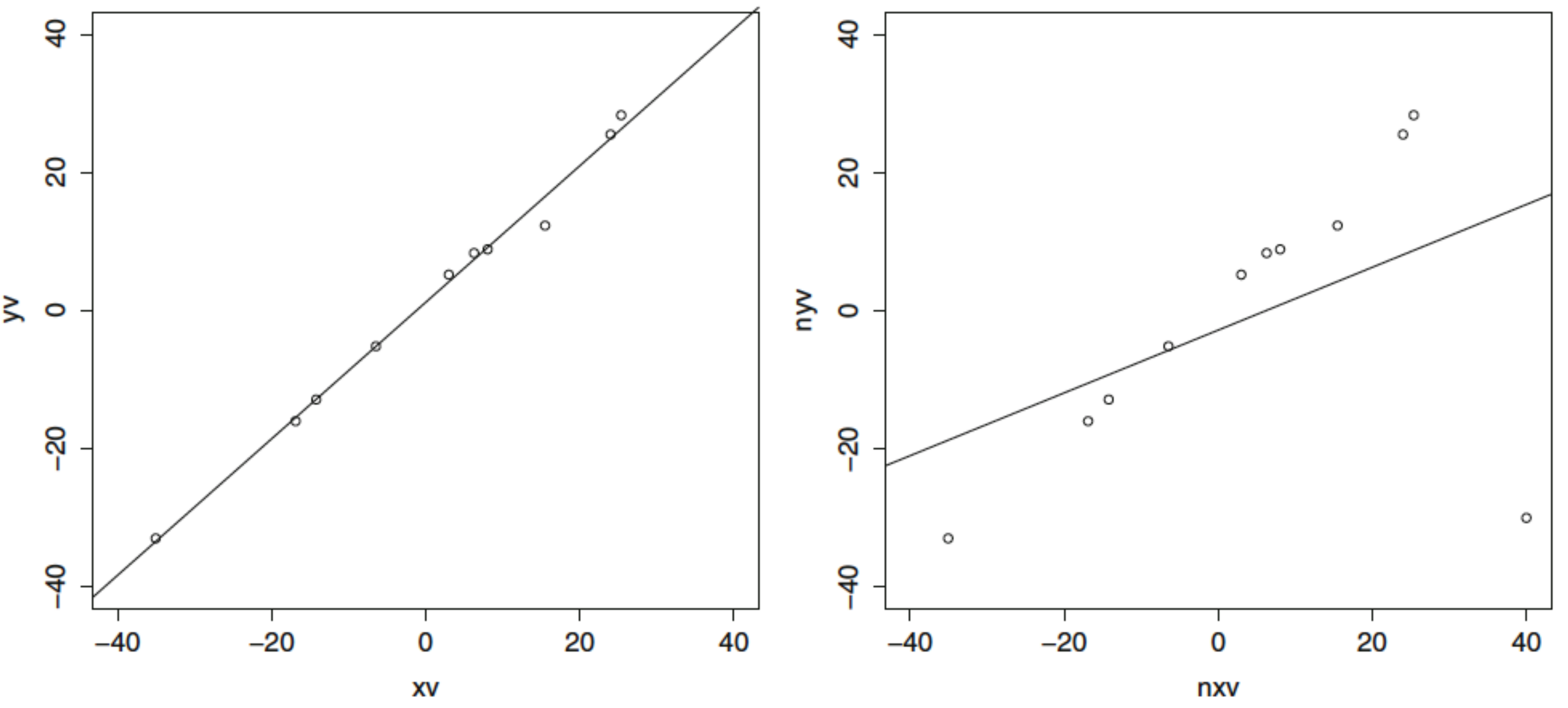
1	Length	$\sqrt[3]{w}$

General form of transformation in Linear Regression

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \cdots$$

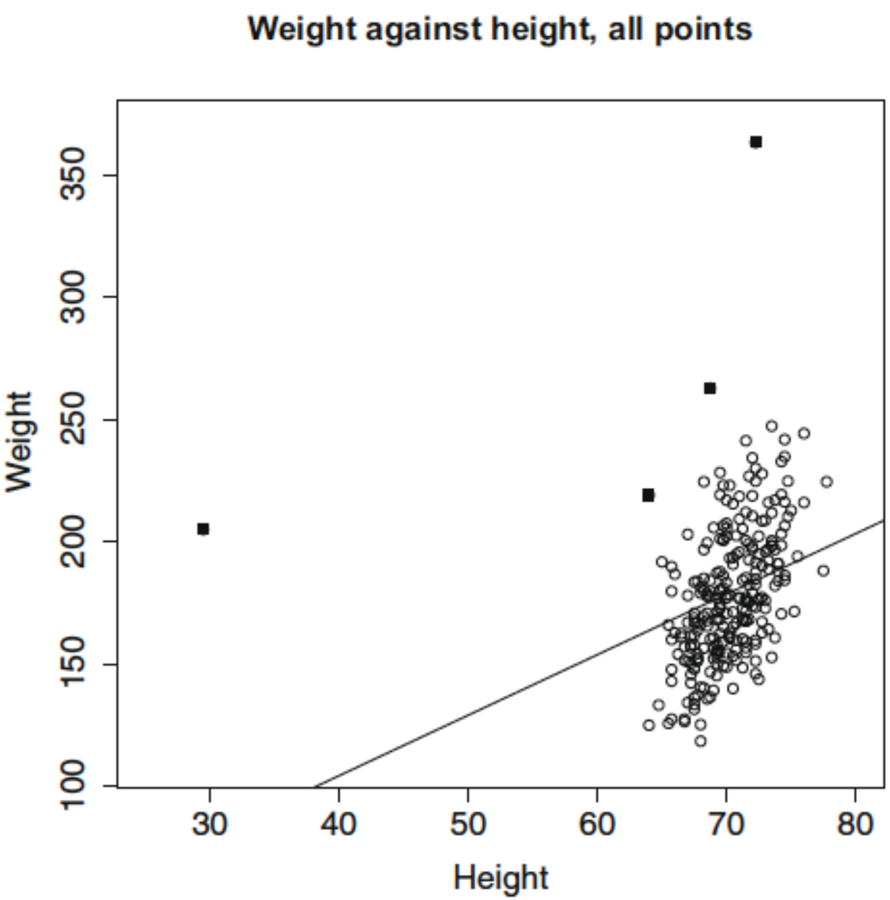
Effect of outliers on linear regression

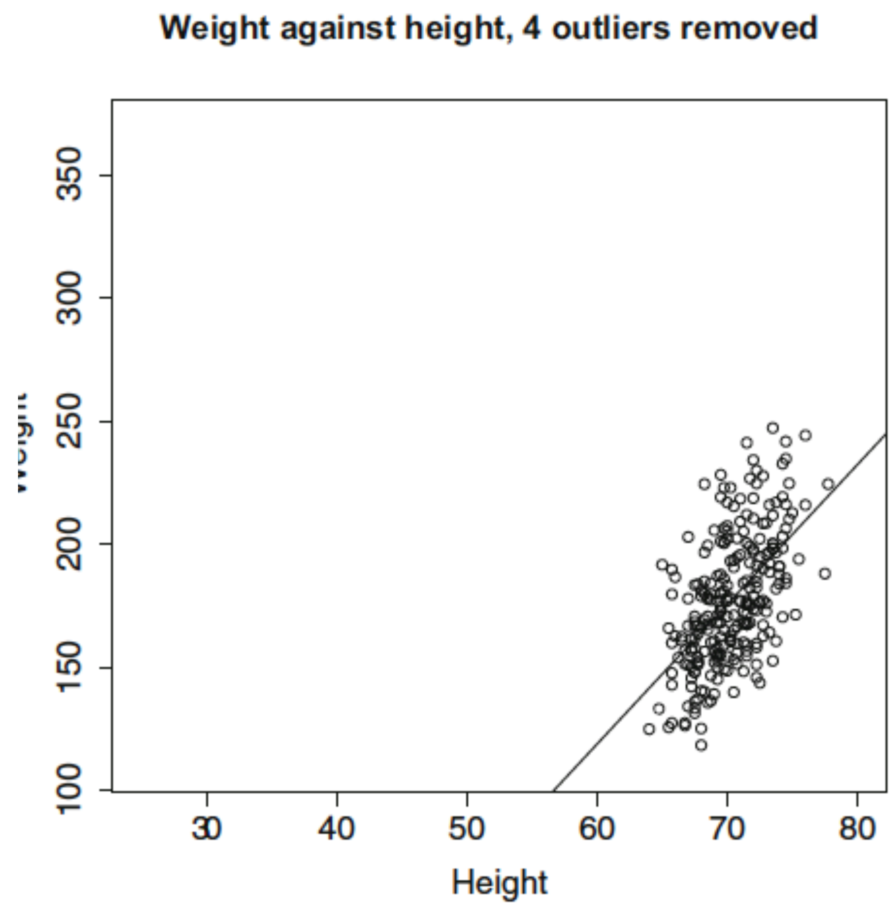
- Linear regression is sensitive to outliers



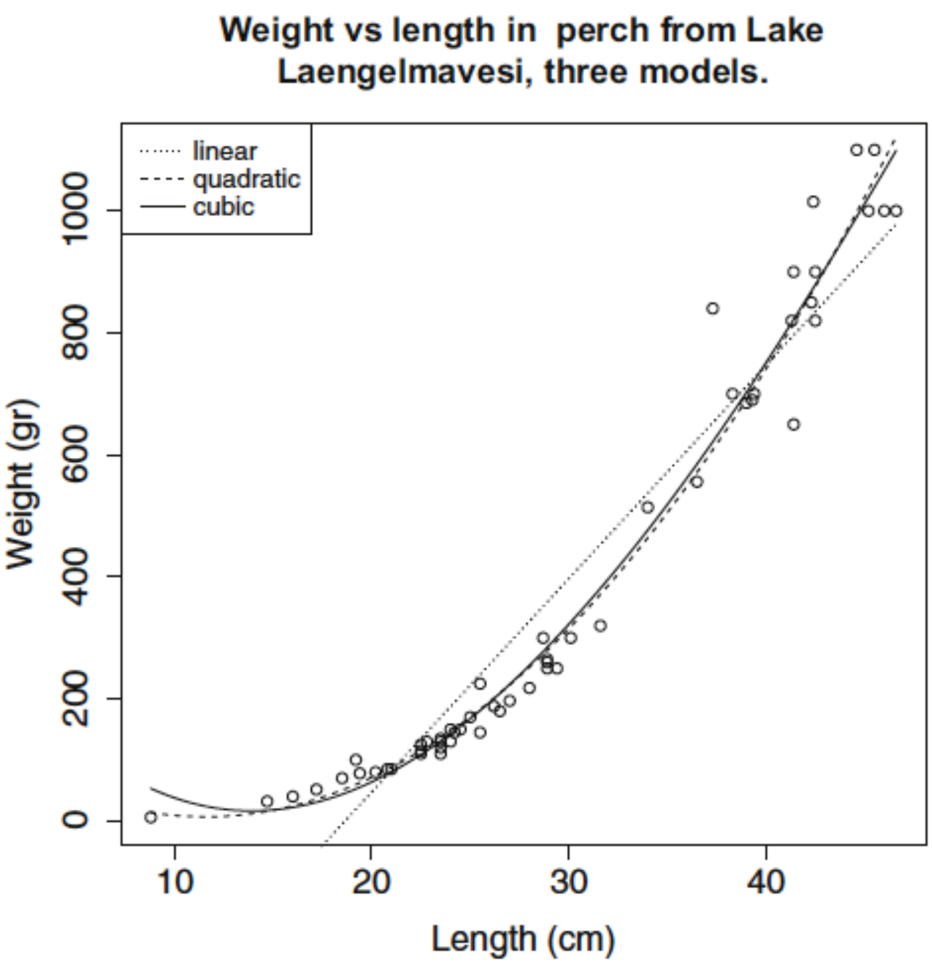
Effect of outliers: body fat example

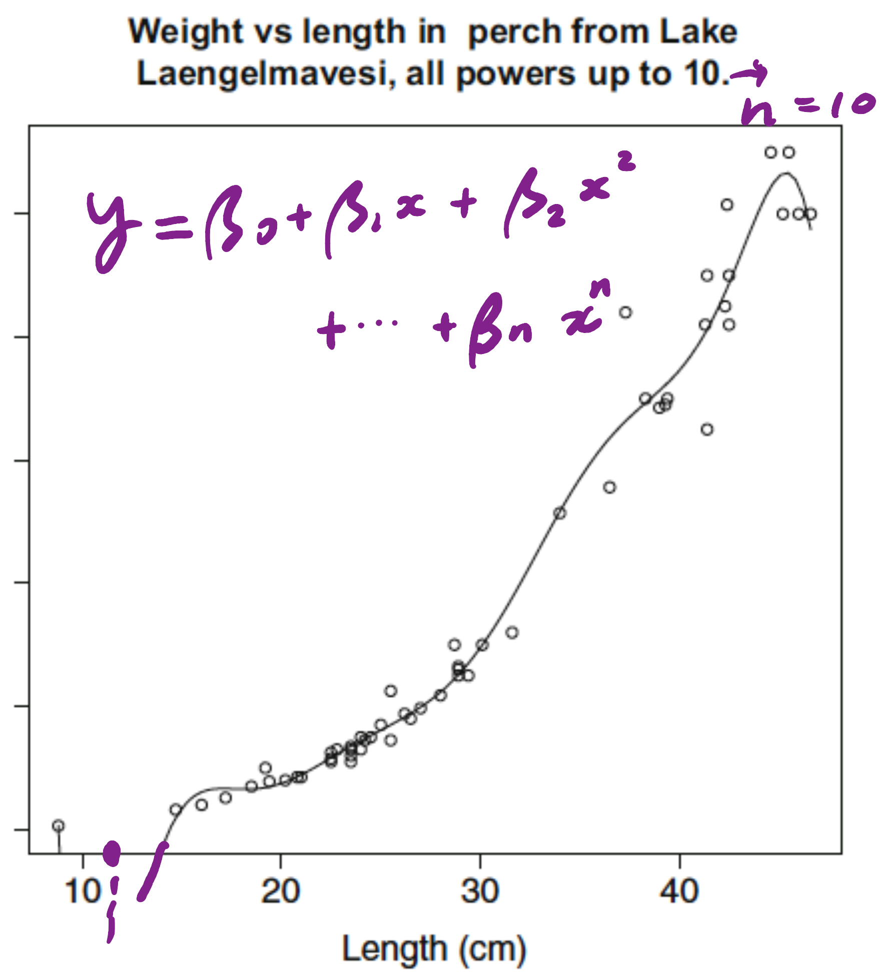
- Linear regression is sensitive to outliers





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





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.

x^n

- **Method 2: regularization**
  - Impose a penalty on complexity of the model during the training
  - Encourage smaller model coefficients

cost $\max(0, 1-y_i a^T x)$

$+$ penalty

$\frac{1}{2}\|a\|^2$

- We can use validation to select regularization parameter λ

Regularized linear regression

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

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

L_2

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

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

Training using regularized least squares

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

$X^T X \hat{\beta} = X^T y$

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

least sq. sol.

$\hat{\beta} = (X^T X)^{-1} X^T y$

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

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

$y^p = X \hat{\beta}$

Why is the regularized version always invertible?

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

positive semi-defi.

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

$\gamma \rightarrow$ eigenvalues

$\gamma_i \geq 0$

positive defi.

$$f^T A f > 0$$

$f \rightarrow$ non-zero vector

$f^T(X^T X + \lambda I) f > 0$

$= f^T X^T X f + f^T \lambda I f$

$= \underbrace{f^T X^T X f}_{\geq 0} + \lambda \underbrace{f^T f}_{> 0}$

$f^T f \rightarrow \|f\|^2 > 0$

$=$

$\lambda > 0$

Why is the regularized version always invertible?

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

Energy based definition of **semi-positive definite**:
Given a matrix A and any nonzero vector **f** , we have

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

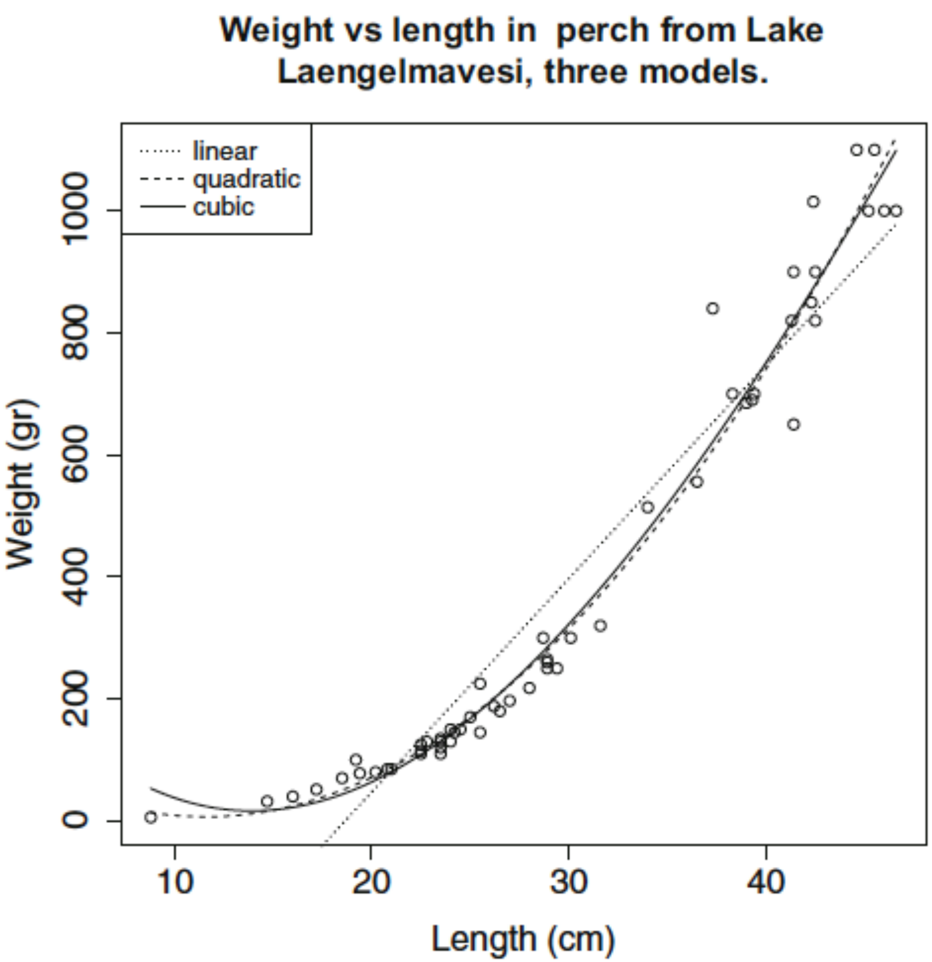
and **positive definite** means

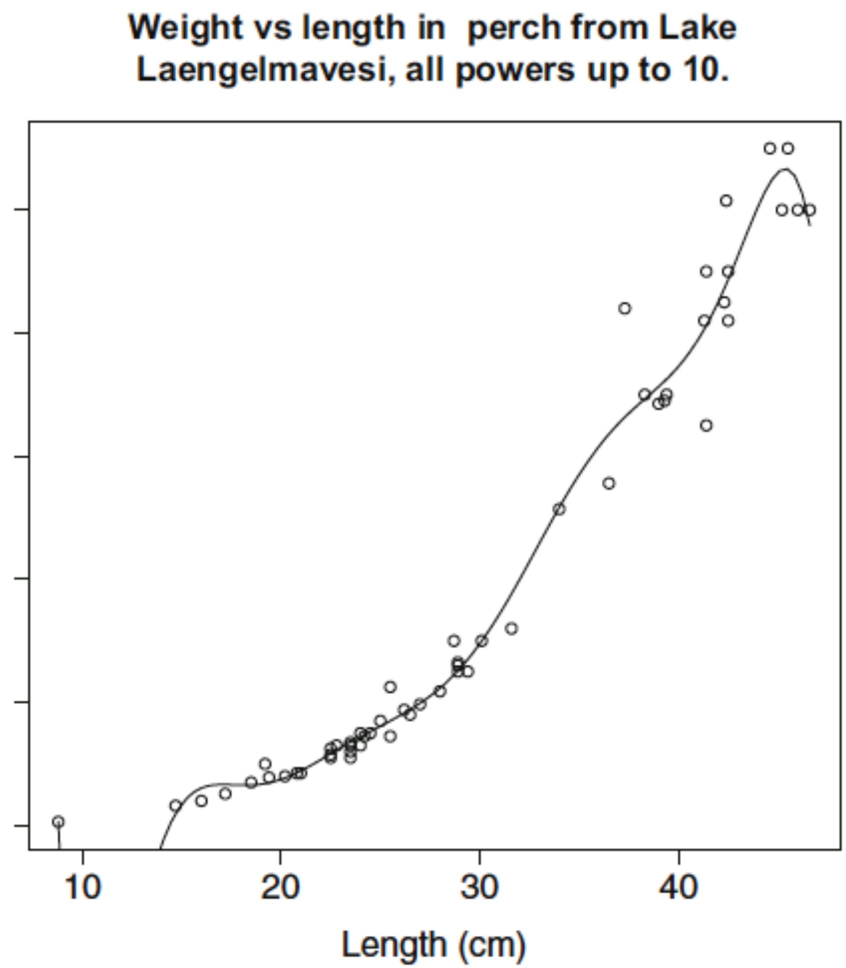
$$f^T A f > 0$$

If A is positive definite, then all eigenvalues of A are positive, then it's invertible

$$
\begin{aligned}
&for\ any\ nonzero\ vector\ f \\
&consider\ f^T(X^T X + \lambda I)f \\
&suppose\ A = X^T X + \lambda I \\
&f^T A f = f^T X^T X f + \lambda f^T f \\
&= f^T X^T X f + \lambda ||f||^2 \\
&given\ X^T X\ is\ semi\ positive\ definite \\
&f^T X^T X f \geq 0 \\
&given\ \lambda > 0 \\
&we\ know\ \lambda ||f||^2 > 0 \\
&\Rightarrow\ f^T A f > 0
\end{aligned}
$$

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

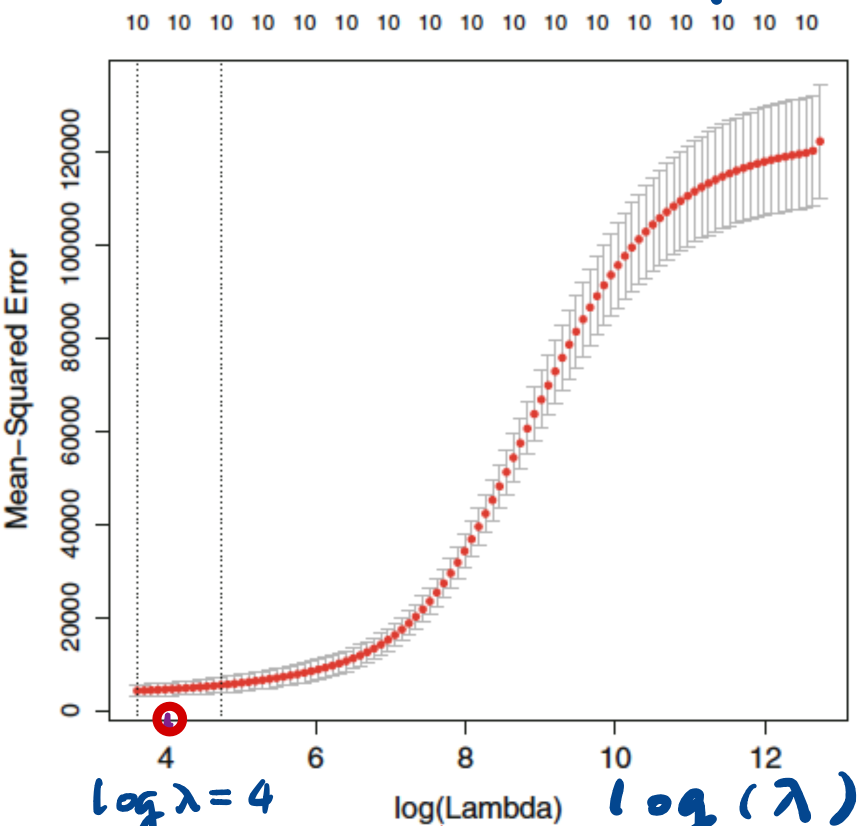




Choosing lambda using cross-validation

||ell

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

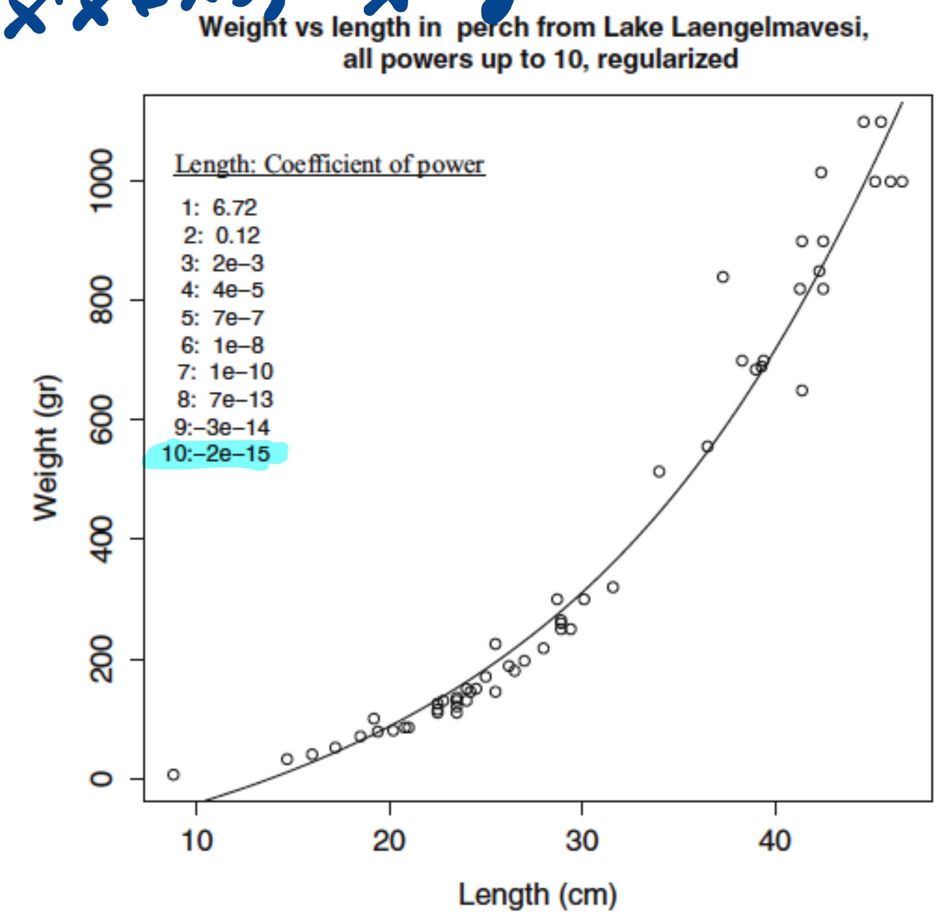


$\log \lambda = 4$

$\lambda = e^4$

$\log(\lambda)$

$\lambda > 0$



Mean Square Error in this model

$$MSE = \frac{e^T e}{N} = \frac{\|y - X\hat{\beta}\|^2}{N}$$

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

$$y^P = X\hat{\beta}$$

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

A. YES

$\boxed{\text{B.}}$ NO

C. YES and NO

?

$$X^T X \beta = X^T y$$

$$X\beta \perp e$$

$$cov(X\beta, e) = 0$$

$$y = X\beta + e$$

$$var(y) = var(X\beta) + var(e) + 2\underbrace{cov(X\beta, e)}_{0}$$

$$R^2 = \frac{var(X\beta)}{var(y)}$$

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

A. YES

B. NO

C. YES and NO

$$X^T X \cdot \beta = X^T y$$

$$e \perp X\beta$$

$$var(y) = var(X\beta) + var(e)$$

$$y = X\beta + e$$

$$var(y) = var(X^T\beta) + var(e) + 2cov(X^T\beta, e)$$

$$(X^T X + \lambda I)\beta = X^T y$$

$$\cancel{e \perp X\beta}$$

Nearest neighbor regression

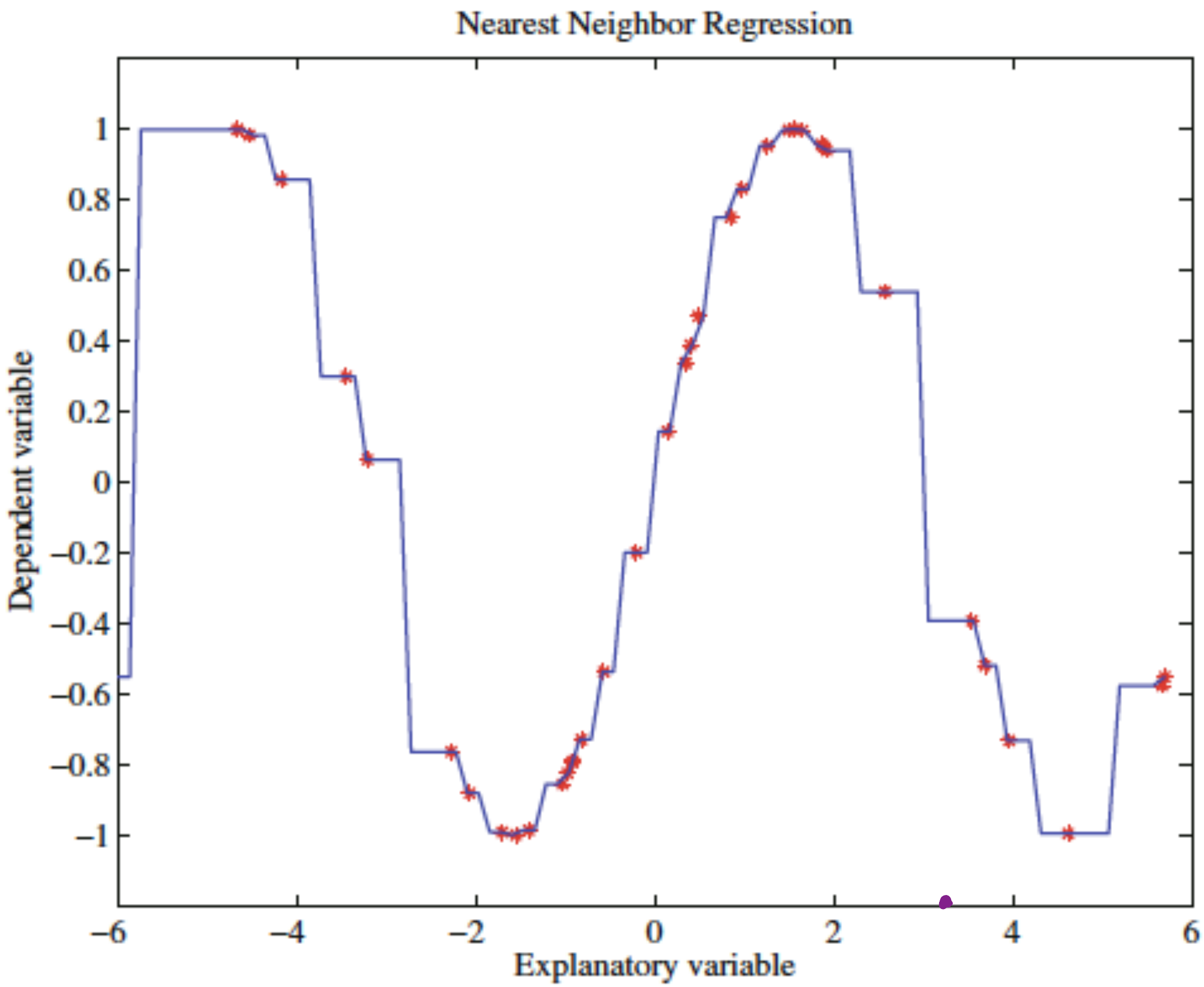
- In addition to linear regression and generalize linear regression models, there are methods such as **Nearest neighbor regression** that do not need much training for the model parameters.
- When there is plenty of data, nearest neighbors regression can be used effectively

y

K nearest neighbor regression with k=1

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

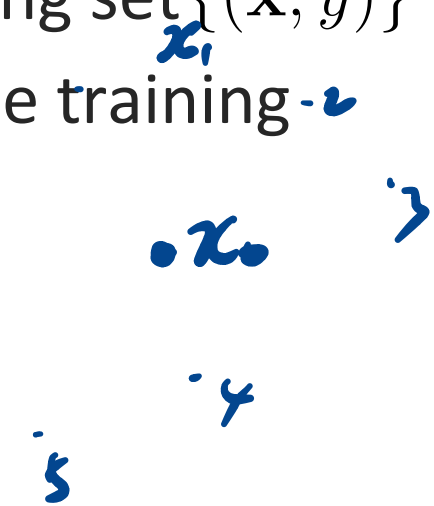
K=1 gives piecewise constant predictions



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:



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

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

$w_j \downarrow$ as dist $\uparrow$

$w_j \uparrow$ as dist $\downarrow$

Choose different weights functions for KNN regression

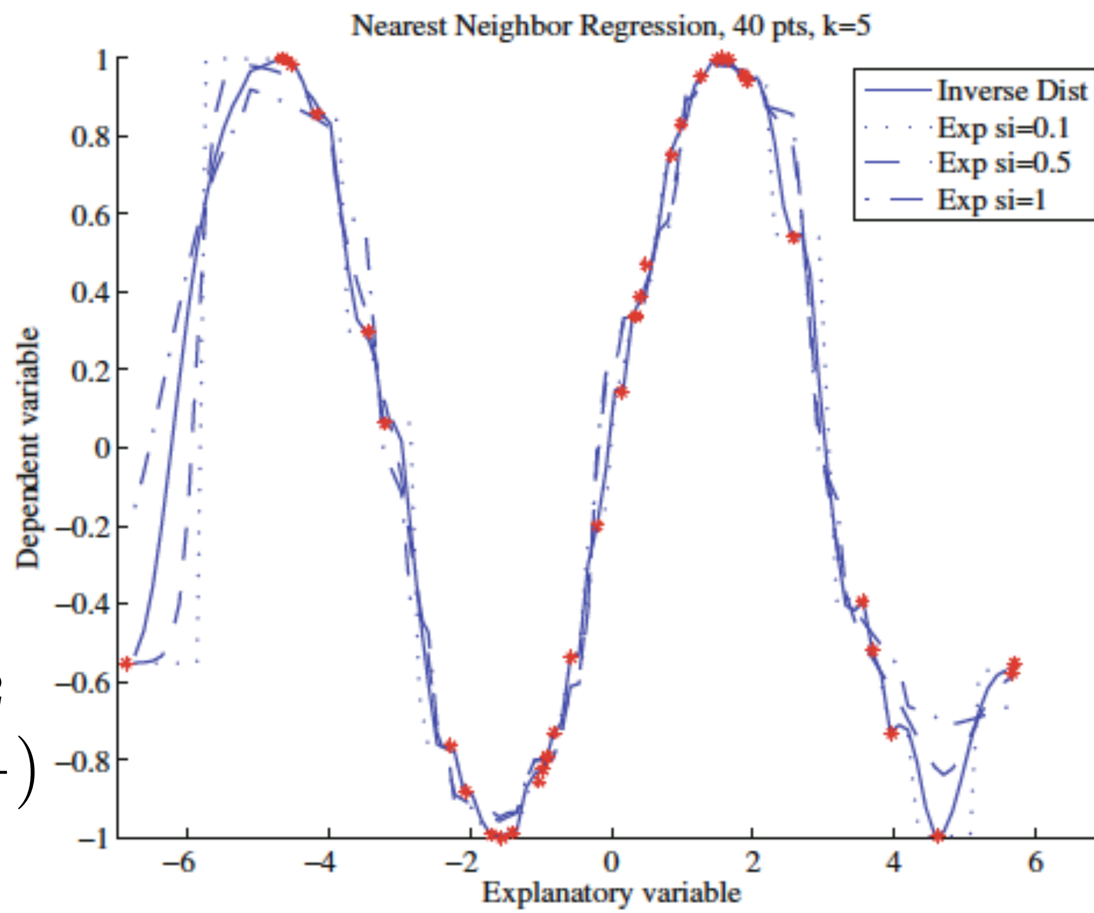
$$\mathbf{y}_0^p = \frac{\sum_j \mathbf{w}_j \mathbf{y}_j}{\sum_j \mathbf{w}_j}$$

- Inverse distance

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

- Exponential function

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



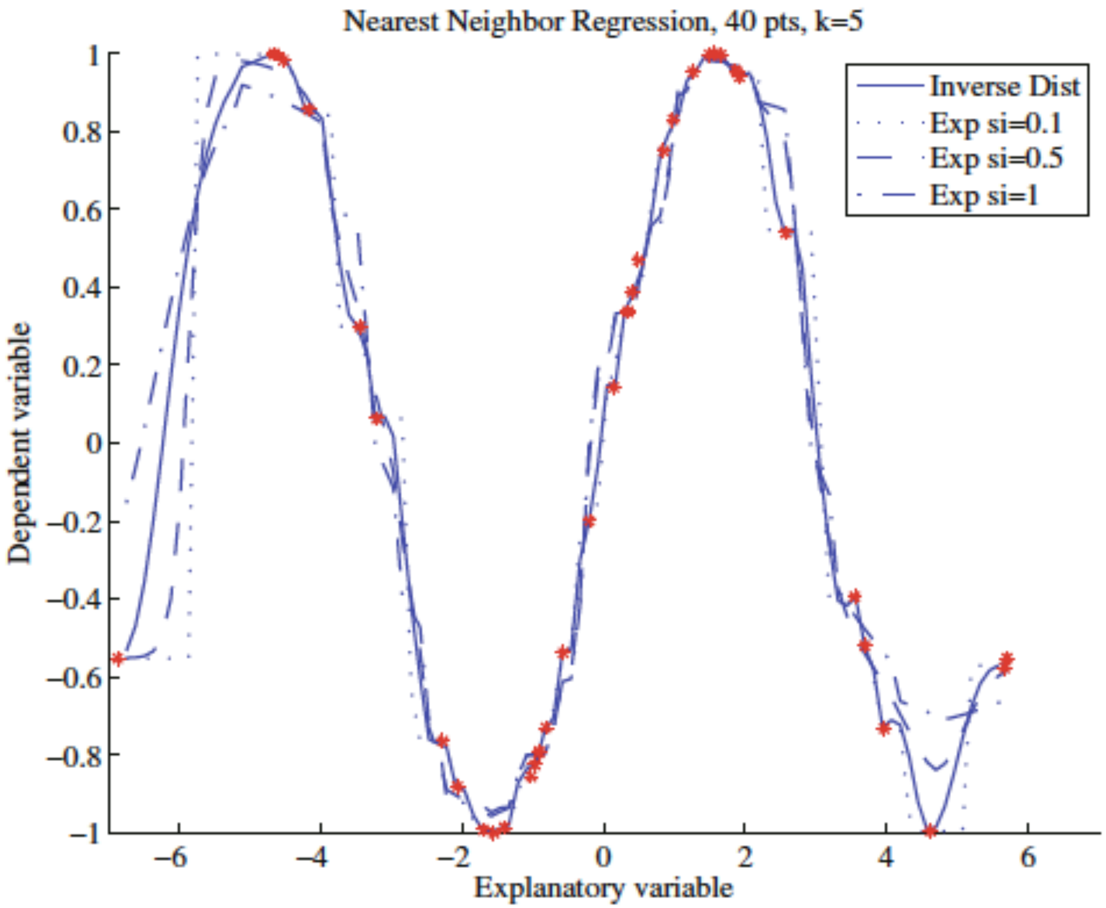
Evaluation of KNN models

- 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.

weight height

_ _

- Cons
  - The method doesn't work well for very high dimensional data
  - The model depends on the scale of the data

Fruit	y
Apple	5
pear	6
banana	5.5

Assignments

- Finish Chapter 13 of the textbook
- Week 13 module including the quiz
- Next time: Curse of Dimension, clustering

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!