## Human T cell expression data

- The matrix contains 47 expression samples from Lukk et al,
   Nature Biotechnology 2010
- All samples are from <u>T cells in different individuals</u>
- Only the top 3000 genes with the largest variability were used
- The value is log2 of gene's expression level in a given sample as

measured by the microarray technology

a T cell

#### A global map of human gene expression

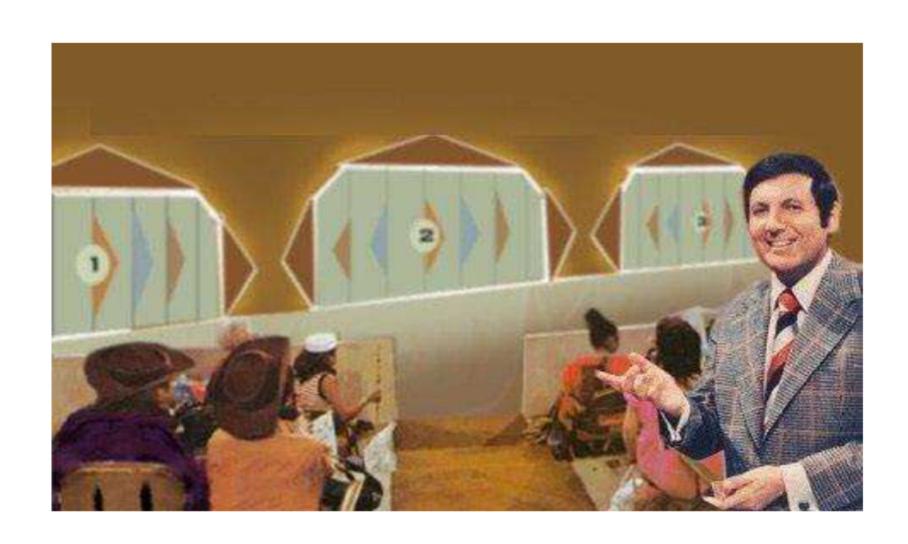
Margus Lukk, Misha Kapushesky, Janne Nikkilä, Helen Parkinson, Angela Goncalves, Wolfgang Huber, Esko Ukkonen & Alvis Brazma

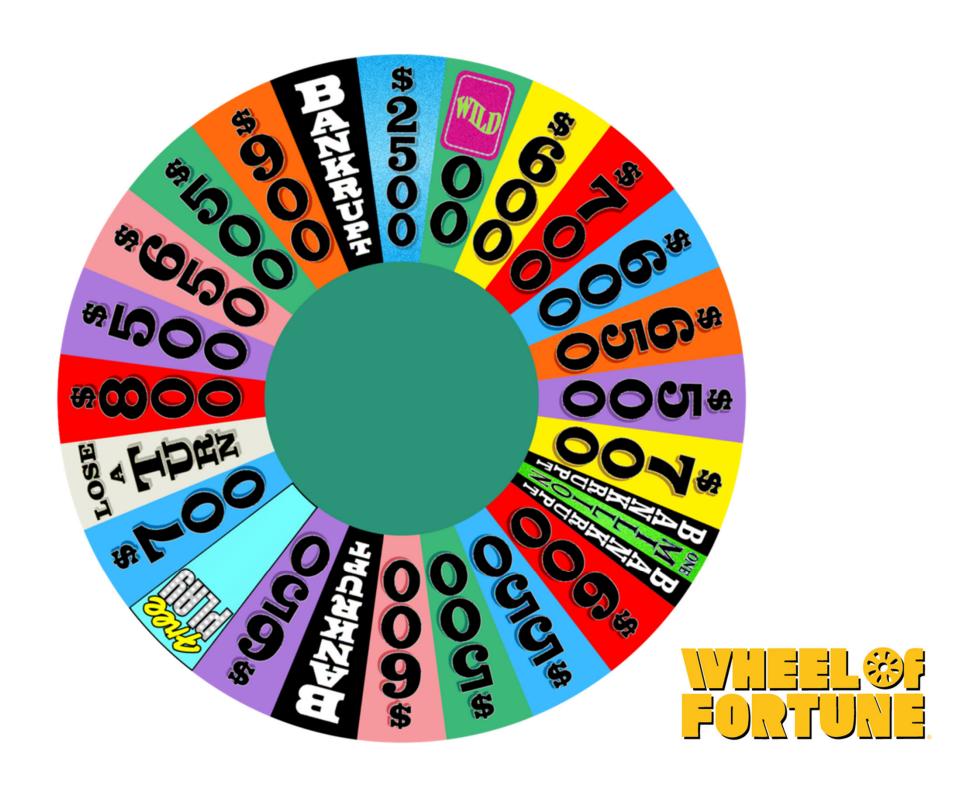
Affiliations | Corresponding author

Nature Biotechnology 28, 322-324 (2010) | doi:10.1038/nbt0410-322

Although there is only one human genome sequence, different genes are expressed in many different cell types and tissues, as well as in different developmental stages or diseases. The structure of this 'expression space' is still largely unknown, as most transcriptomics experiments focus on sampling small regions. We have constructed a global gene expression map by integrating microarray data from 5,372 human samples representing 369 different cell and tissue types, disease states and cell lines. These have been compiled in an online resource (http://www.ebi.ac.uk/gxa/array/U133A) that allows the user to search for a gene of interest and

## "Let's Make a Deal" show with Monty Hall aired on NBC/ABC 1963-1986





#### Matlab exercise #1: "Wheel of Fortune"

- Each group gets a pair of genes that are known to be correlated.
- Each group also gets a random pair of genes selected by the "Wheel of Fortune". They may or may not be correlated
- Download (log-transformed) expression\_table.mat
- Run command fitlm(x,y) on assigned and random pairs
- Record  $\beta_0$ ,  $\beta_1$ ,  $R^2$ , P-value of the slope  $\beta_1$  and write them on the blackboard
- Validate Matlab result for R<sup>2</sup> using your own calculations
- Look up gene names (see gene\_description in your workspace) and write down a brief description of biological functions of genes. Does their correlation make biological sense?

# Correlated pairs plausible biological connection based on short description

```
1, 6 g1=1994; g2=188;
    2, g1=2872; g2=1269;
    3, g1=1321; g2=10;
    4, g1= 886; g2=819;
    5, g1=2138; g2=1364;
no obvious biological common function
g1=1+floor(rand.*3000); g2=1+floor(rand.*3000);
disp([g1, g2])
```

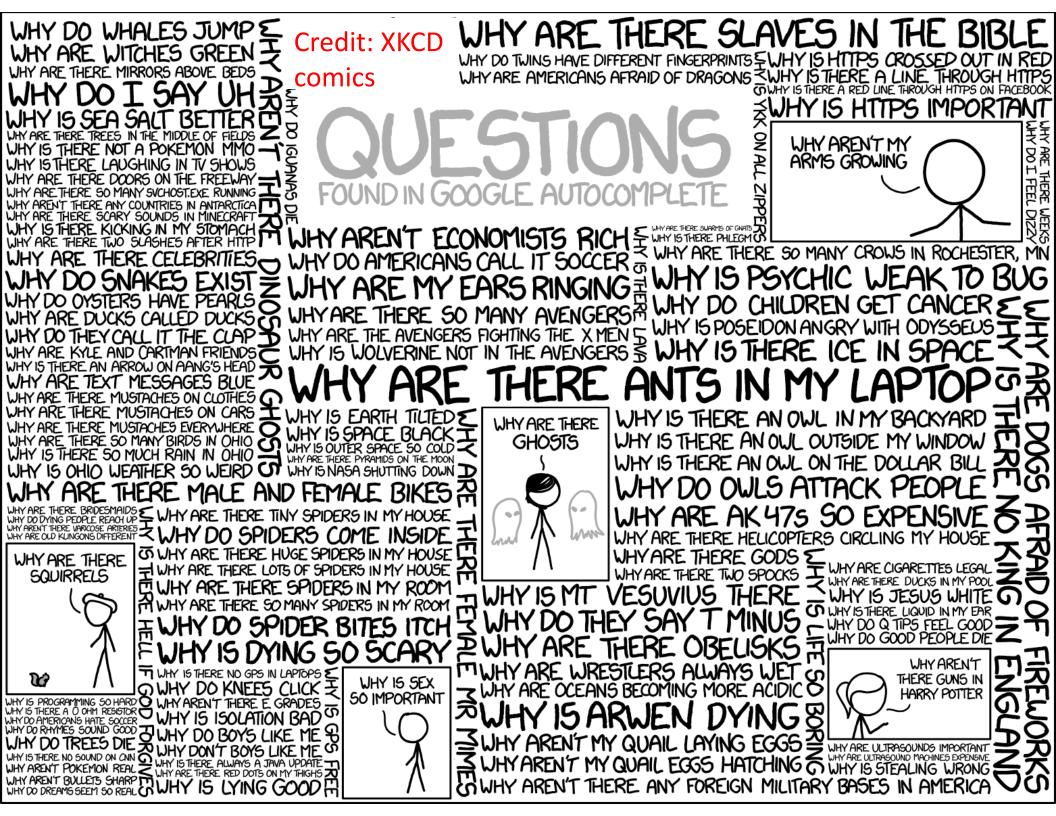
### Random pairs

```
>> g1=floor(3000.*rand)+1; g2=floor(3000.*rand)+1;
disp([g1,g2]);
>> g1=floor(3000.*rand)+1; g2=floor(3000.*rand)+1;
disp([g1,g2]);
>> g1=floor(3000.*rand)+1; g2=floor(3000.*rand)+1;
disp([g1,g2]);
>> g1=floor(3000.*rand)+1; g2=floor(3000.*rand)+1;
disp([g1,g2]);
```

#### Matlab code

 load expression table.mat • g1=2907; g2=288; x=exp\_t(g1,:)'; y=exp\_t(g2,:)'; figure; plot(x,y,'ko'); Im=fitIm(x,y) • y fit=lm.Fitted; hold on; plot(x,lm.Fitted,'r-'); SST=sum((y-mean(y)).^2); SSR=sum((y fit-mean(y)).^2); SSE=sum((y-y fit).^2); R2=SSR./SST disp([gene\_names(g1), gene\_names(g2)]);

disp(gene description(g1)); disp(gene description (g2));



## Multiple Linear Regression

(Chapters 12-13 in Montgomery, Runger)

#### 12-1: Multiple Linear Regression Model

#### 12-1.1 Introduction

- Many applications of regression analysis involve situations in which there are more than one regressor variable X<sub>k</sub> used to predict Y.
- A regression model then is called a multiple regression model.

## Multiple Linear Regression Model

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + ... \beta_k x_k + \varepsilon$$

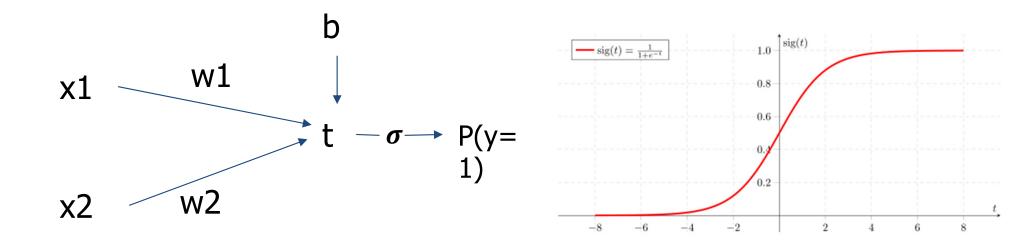
One can also use powers and products of other variables or even non-linear functions like  $exp(x_i)$  or  $log(x_i)$  instead of  $x_3, \ldots, x_k$ .

Example: the general two-variable quadratic regression has 6 constants:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 (x_1)^2 + \beta_4 (x_2)^2 + \beta_5 (x_1 x_2) + \varepsilon$$

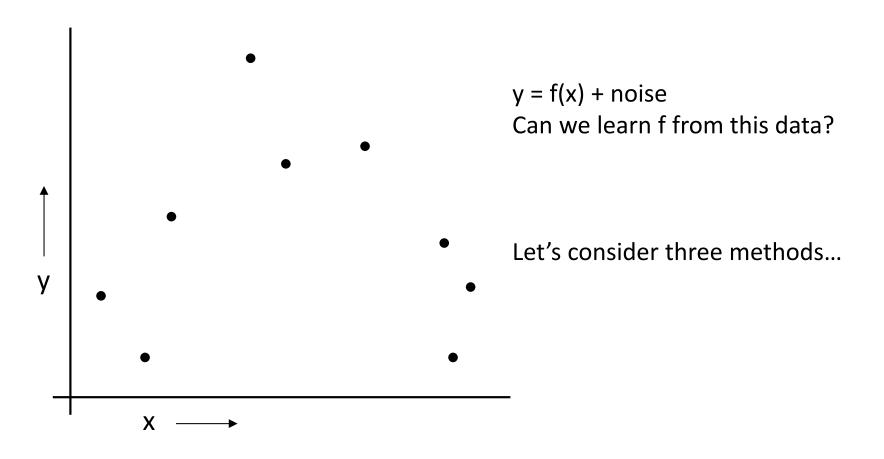
## **Logistic Regression**

$$P(y=1) = \sigma(x1*w1 + x2*w2 + b)$$

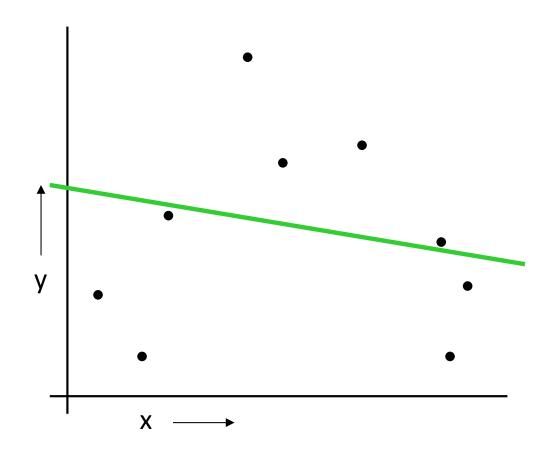


How to know where to stop adding new variables or powers of old variables?

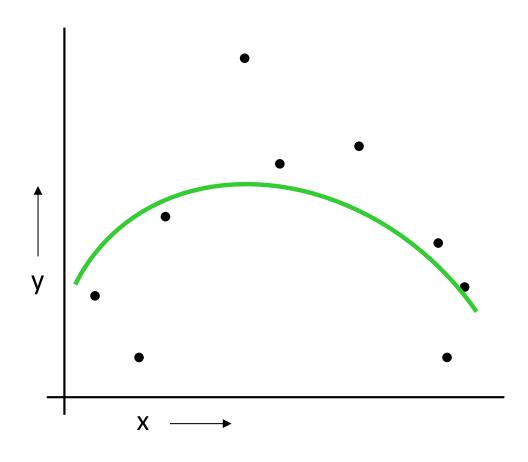
## A Regression Problem



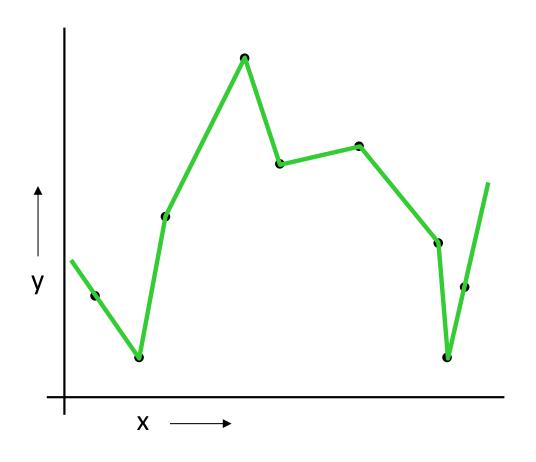
## Linear Regression



## **Quadratic Regression**

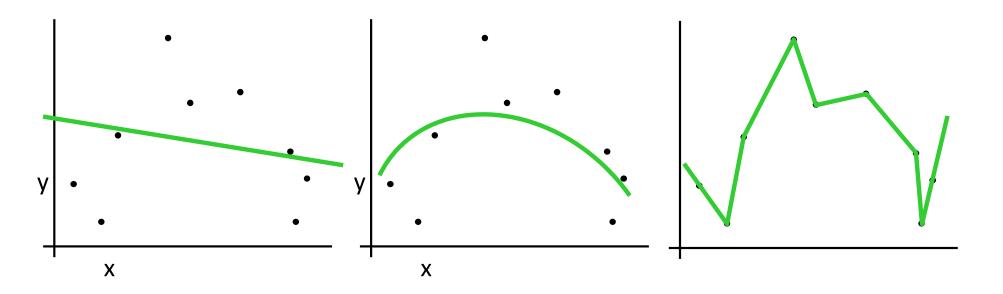


#### Join-the-dots



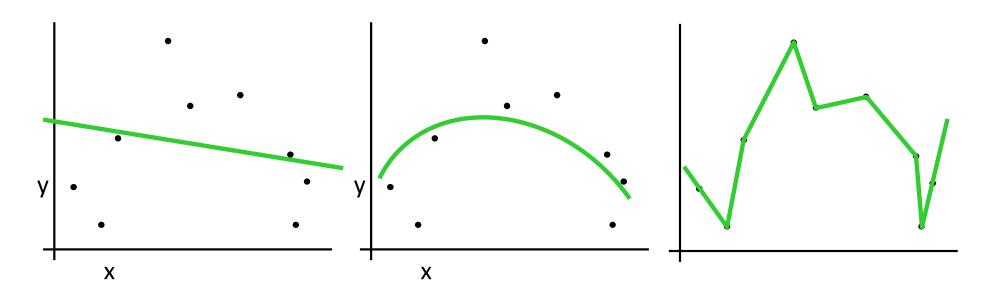
Also known as piecewise linear nonparametric regression if that makes you feel better

## Which is best?



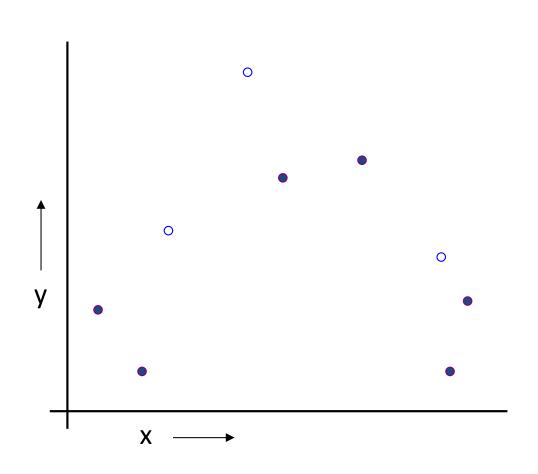
Why not choose the method with the best fit to the data?

## What do we really want?

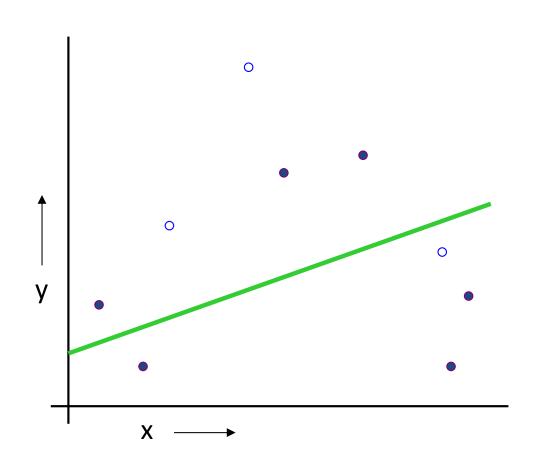


Why not choose the method with the best fit to the data?

"How well are you going to predict future data drawn from the same distribution?"

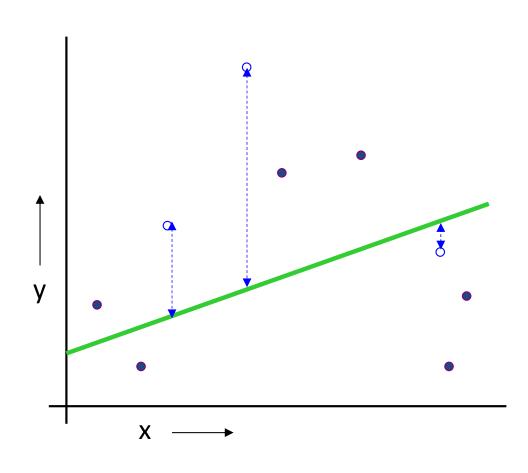


Randomly choose
 of the data to
 in a test set
 The remainder is a training set



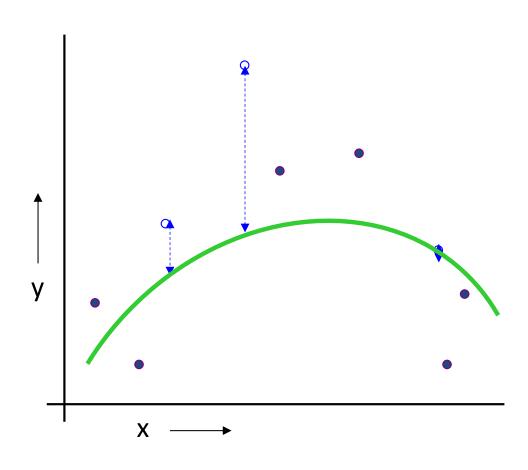
(Linear regression example)

- 1. Randomly choose 30% of the data to be in a test set
- 2. The remainder is a training set
- 3. Perform your regression on the training set



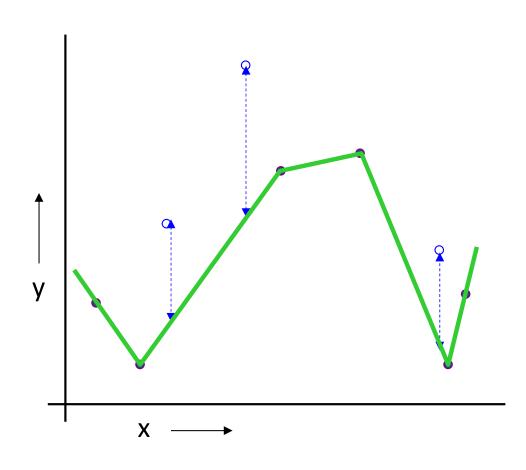
(Linear regression example) Mean Squared Error = 2.4

- 1. Randomly choose 30% of the data to be in a test set
- 2. The remainder is a training set
- 3. Perform your regression on the training set
- 4. Estimate your future performance with the test set



(Quadratic regression example) Mean Squared Error = 0.9

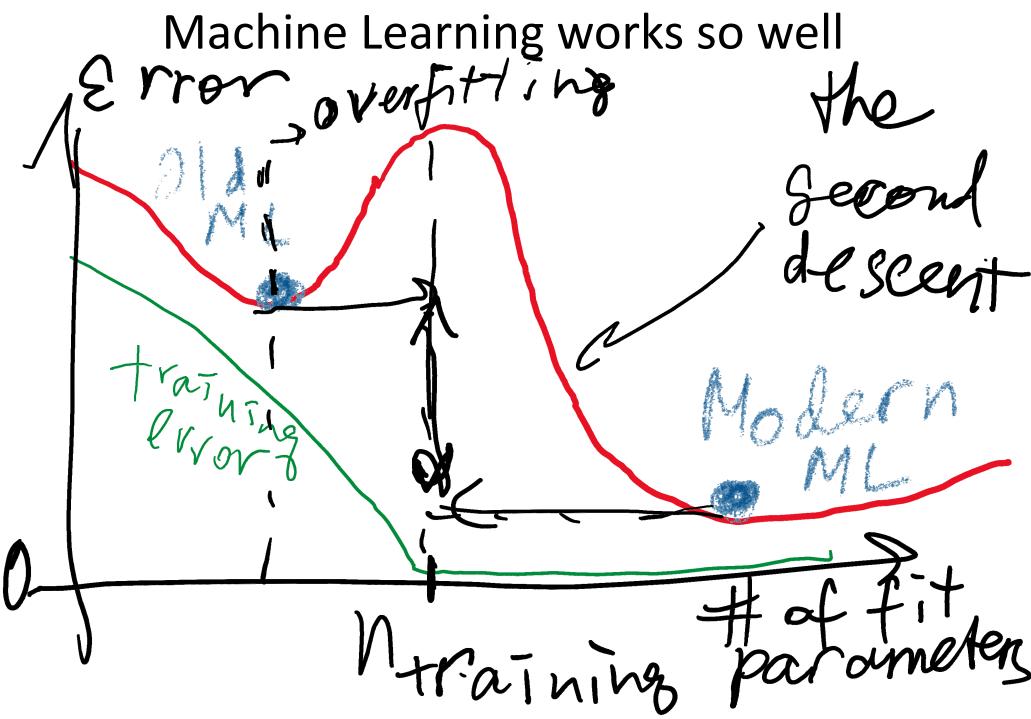
- 1. Randomly choose 30% of the data to be in a test set
- 2. The remainder is a training set
- 3. Perform your regression on the training set
- 4. Estimate your future performance with the test set



(Join the dots example)
Mean Squared Error = 2.2

- 1. Randomly choose 30% of the data to be in a test set
- 2. The remainder is a training set
- 3. Perform your regression on the training set
- 4. Estimate your future performance with the test set

Double descend- the main reason modern



#### 12-1: Multiple Linear Regression Model

#### 12-1.3 Matrix Approach to Multiple Linear Regression

Suppose the model relating the regressors to the response is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_k x_{ik} + \varepsilon_i$$
  $i = 1, 2, \dots, n$ 

In matrix notation this model can be written as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{12-6}$$

#### 12-1: Multiple Linear Regression Model

#### 12-1.3 Matrix Approach to Multiple Linear Regression

#### where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \text{ and } \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

#### 12-1.3 Matrix Approach to Multiple Linear Regression

We wish to find the vector  $\hat{\beta}$  that minimizes the sum of squares of error terms:

$$L = \sum_{i=1}^{n} \epsilon_i^2 = \epsilon' \epsilon = (y - X\beta)' (y - X\beta)$$

$$0 = \frac{\partial L}{2\partial \beta} = -X' (\mathbf{y} - \mathbf{X}\beta) = -X' \mathbf{y} + (X'X)\boldsymbol{\beta}$$

The resulting least squares estimate is

$$\widehat{\beta} = (X'X)^{-1} X'y$$
(12-7)

Analog of Ver(X)

Analog of Cov (X,Y)

Sec 12-1 Multiple Linear Regression Model

## Multiple Linear Regression Model

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

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

$$\hat{y} = Hy, \text{ and } e = (I - H)y.$$

$$H = H^{1} \cdot H^{2} = X(X'X)^{-1} X'X(X'X)^{-1}X^{1} = X(X'X)^{-1}X^{1} + X(X'X)^{-1}X^{1} = X(X'X)^{1}X^{1} = X(X'X)^{-1}X^{1} = X(X'X)^{1}X^{1} = X(X'X)^{-1}X^{1} =$$

#### 12-1: Multiple Linear Regression Models

#### 12-1.4 Properties of the Least Squares Estimators

#### **Unbiased estimators:**

$$E(\hat{\boldsymbol{\beta}}) = E[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}]$$

$$= E[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon})]$$

$$= E[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}\boldsymbol{\beta} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\boldsymbol{\epsilon}]$$

$$= \boldsymbol{\beta}$$

#### Covariance Matrix of Estimators:

$$\mathbf{C} = (\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{10} & C_{11} & C_{12} \\ C_{20} & C_{21} & C_{22} \end{bmatrix}$$

#### 12-1: Multiple Linear Regression Models

#### 12-1.4 Properties of the Least Squares Estimators

Individual variances and covariances:

$$V(\hat{\beta}_j) = \sigma^2 C_{jj}, \qquad j = 0, 1, 2$$
$$cov(\hat{\beta}_i, \hat{\beta}_j) = \sigma^2 C_{ij}, \qquad i \neq j$$

In general,

$$\operatorname{cov}(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} = \sigma^2\mathbf{C}$$

#### 12-1: Multiple Linear Regression Models

#### Estimating error variance $\sigma_{\epsilon}^{2}$

An unbiased estimator of error variance  $\sigma_{\epsilon}^{2}$  is

$$\hat{g}^2 = \frac{\sum_{i=1}^n e_i^2}{n-p} = \frac{SS_E}{n-p}$$
 (12-16)

Here p=k+1 for k-variable multiple linear regression

## R<sup>2</sup> and Adjusted R<sup>2</sup>

#### The coefficient of multiple determination R<sup>2</sup>

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T}$$

The adjusted  $R^2$  is

$$R_{\text{adj}}^2 = 1 - \frac{SS_E/(n-p)}{SS_T/(n-1)}$$

- The adjusted R<sup>2</sup> statistic penalizes adding terms to the MLR model.
- It can help guard against overfitting (including regressors that are not really useful)

# How to know where to stop adding variables?

 Adding new variables x<sub>i</sub> to MLR watch the adjusted R<sup>2</sup>

Once the adjusted R<sup>2</sup>
 no longer increases = stop.
 Now you did the best you can.

