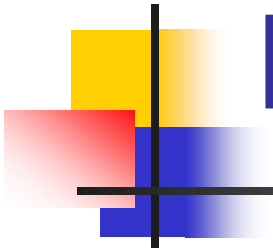
Session 1
An Overview of Machine Learning

An Introduction to Machine Learning for the Behavioural and Social Sciences

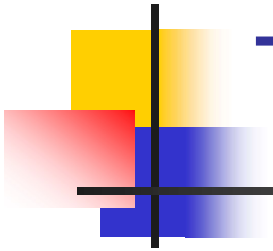
Heungsun Hwang & Gyeongcheol Cho



Machine Learning

- A vast set of statistical tools for explaining and/or predicting data.
  - Supervised learning
    - Both inputs and output(s)
  - Unsupervised learning
    - Only inputs
    - Explain or summarize inherent structure of input data

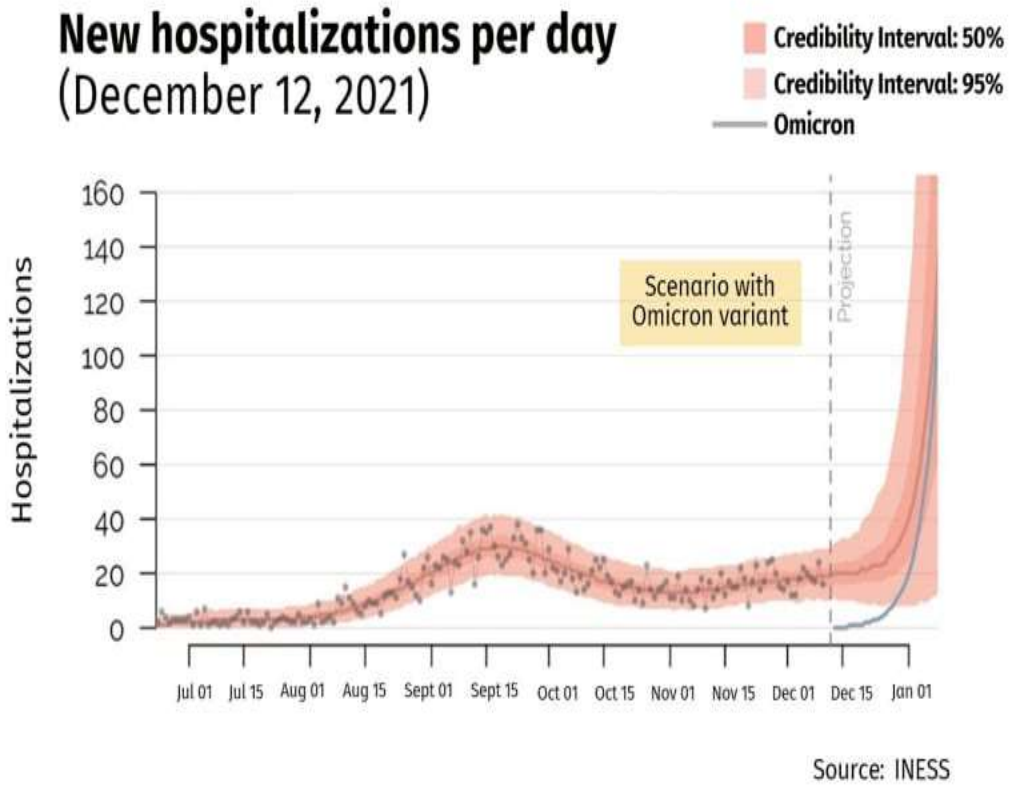
Two Cultures of Statistical Modeling: To Explain or to Predict?



- Explanation: Statistical models/methods are used for understanding or describing associations between variables or testing hypotheses about parameters.
  - Which predictors are associated with the response?
  - What is the relationship between the response and each predictor?
  - How are many different variables associated with one another?

Two Cultures of Statistical Modeling: To Explain or to Predict?

- Prediction: Statistical models/methods are used for predicting new or future observations.



https://www.cbc.ca/news/canada/montreal/quebec-tightens-covid-measures-omicron-1.6288120

Example: Use of Linear Regression for Explanation

- The children's antisocial behaviour data: Part of the National Longitudinal Survey of Youth (NLSY) reported in Curran (1998).
  - Response (output/DV) = The antisocial behaviour of children measured at the first time point (0-12).
  - Predictors (inputs):
    - Gender (female = 0 and male = 1)
    - Cognitive stimulation for children at home (0-14)
    - Emotional support for children at home (0-13)

Example: Use of Linear Regression for Explanation

Model Summary

Model	R	R^2	Adjusted R^2	RMSE
1	0.285	0.082	0.069	1.485

ANOVA

Model		Sum of Squares	df	Mean Square	F	p
1	Regression	42.481	3	14.160	6.418	< .001
	Residual	478.758	217	2.206		
	Total	521.240	220			

Example: Use of Linear Regression for Explanation

Coefficients

Model		Unstandardized	Standard Error	Standardized	t	p
1	(Intercept)	2.448	0.500		4.896	< .001
	GENDER	0.634	0.201	0.206	3.155	0.002
	COGSTM	0.013	0.043	0.020	0.295	0.769
	EMOTSUP	-0.151	0.047	-0.222	-3.216	0.001

Example: Use of Linear Regression for Prediction

As shown below, the Mega Telecom database includes information on current customers of a high speed Internet service. Which of two new customers, Smith or Jones, is more likely to order the high speed Internet service?

Customer	Age	Total # Services Ordered Ever	Monthly Dollars Paid						Outside Questionnaire - Do You Use the Internet?	Non-Basic Services Currently Active on			
			Sep-98	Oct-98	Nov-98	Dec-98	Jan-99	Feb-99		Long Distance	Wireless	Multiple Lines	Toll-Free Number
	31	7	$102	$110	$125	$153	$120	$106	Yes	No	Yes	Yes	No
	27	4	$90	$86	$99	$105	$89	$79	Yes	No	no	No	No

Example: Use of Linear Regression for Prediction

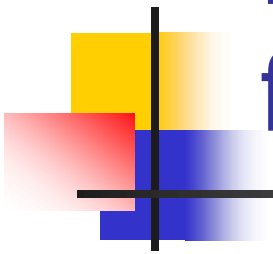
Variable	Definition	Coefficient /Weight
Constant	A constant value	0.151767
X1	= 1, if age between 26 and 30 = 0, otherwise	0.023618
X2	= Total # services ordered ever	0.060634
X3	= 1, if said yes to an outside questionnaire asking if use the Internet = 0, otherwise	0.008761
X4	= Average dollars paid per month in the past 6 months	0.003259
X5	= 1, if currently active on one or more of the following non-basic services: long distance, wireless, multiple lines, toll-free number = 0, otherwise	0.086853

Example: Use of Linear Regression for Prediction

Variable	Smith Status	Smith Score		Jones Status	Jones Score
Constant	yes	0.151767		yes	0.151767
Age between 26 and 30?	no	0		yes	0.023618
Total # services ordered ever	7	7 x .060634 = ..424438		4	4 x .060634 = .242536
Question - use Internet?	yes	0.008761		yes	0.008761
Average $ paid in past 6 months	($102+$110+ $125+$153+ $120+$106)/6 = $119.33	119.33 x .003259 =.388896		($90+$86+$99+$ 105+$89+$79)/6 = $91.33	91.33 x .003259 = .297644
Active on 1+ services?	yes	0.086853		no	0
		1.060715			0.724326

$\hat{Y}_{Smith}$ $\hat{Y}_{Jones}$

Example: Use of Linear Regression for Prediction



- Since Smith has a higher regression score than Jones, Smith is more likely to order the high-speed Internet service.
  - The higher the value, the more likely the customer is to take the response modeled.

Statistical Science
2001, Vol. 16, No. 3, 199–231

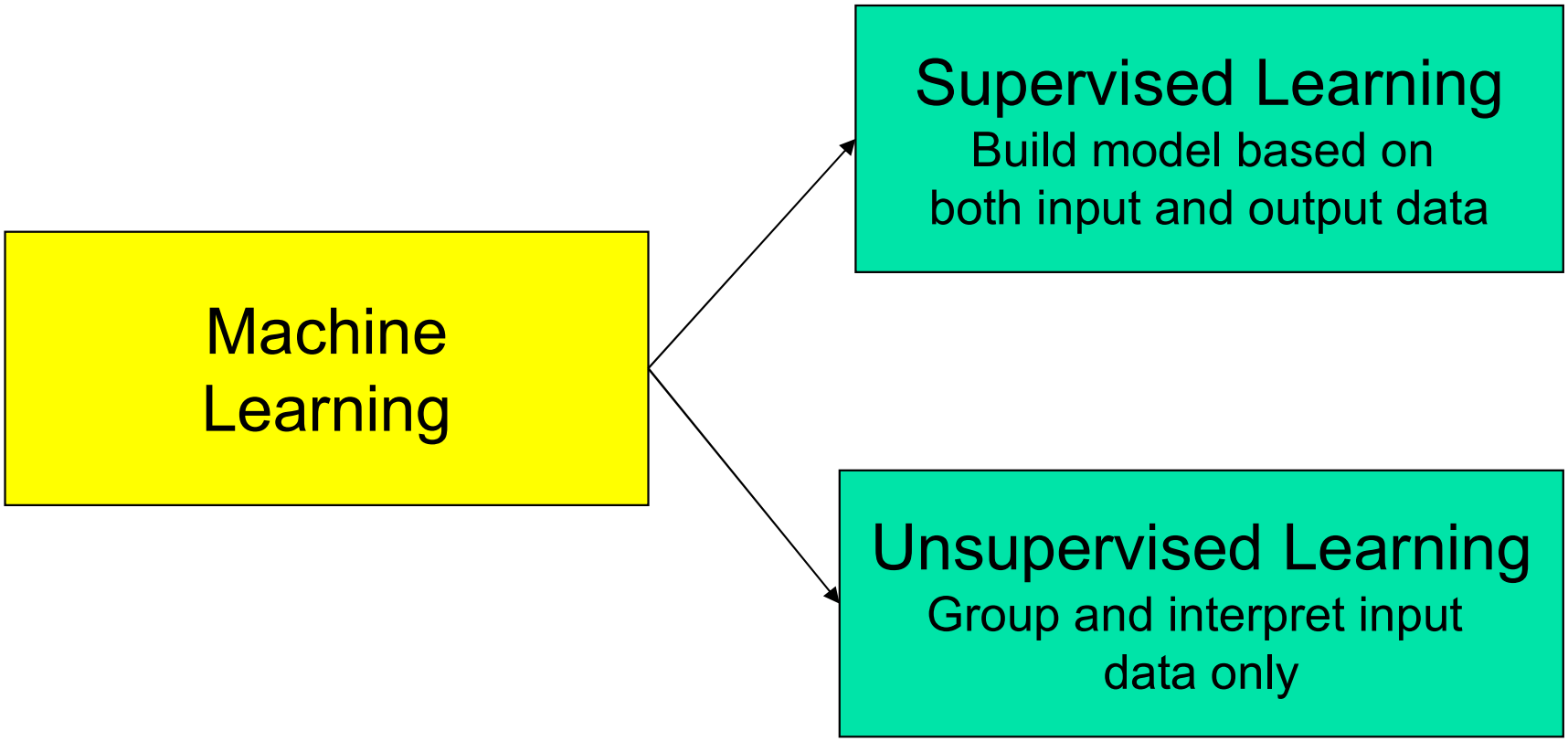
Statistical Modeling: The Two Cultures

Leo Breiman

Abstract. There are two cultures in the use of statistical modeling to reach conclusions from data. One assumes that the data are generated by a given stochastic data model. The other uses algorithmic models and treats the data mechanism as unknown. The statistical community has been committed to the almost exclusive use of data models. This commitment has led to irrelevant theory, questionable conclusions, and has kept statisticians from working on a large range of interesting current problems. Algorithmic modeling, both in theory and practice, has developed rapidly in fields outside statistics. It can be used both on large complex data sets and as a more accurate and informative alternative to data modeling on smaller data sets. If our goal as a field is to use data to solve problems, then we need to move away from exclusive dependence on data models and adopt a more diverse set of tools.

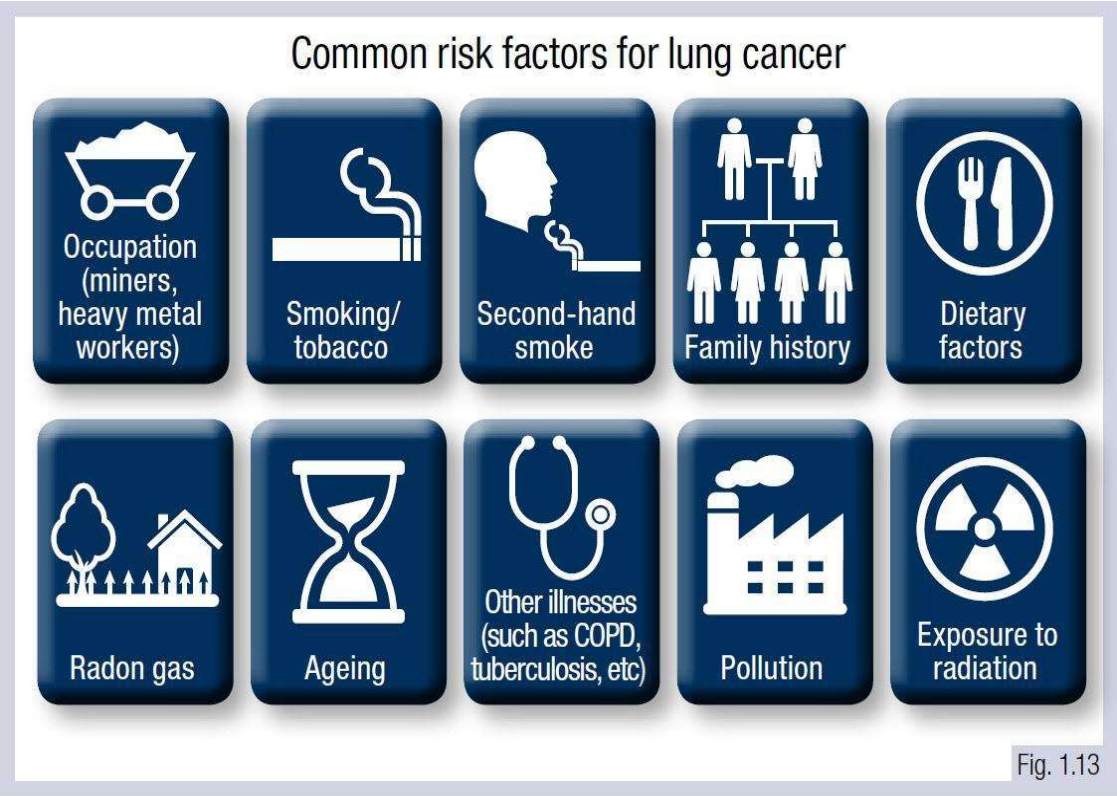
Also see Shmueli, G. (2010). To Explain or to Predict?, *Statistical Science*, 25, 289–310

Machine Learning



Supervised Learning Problems

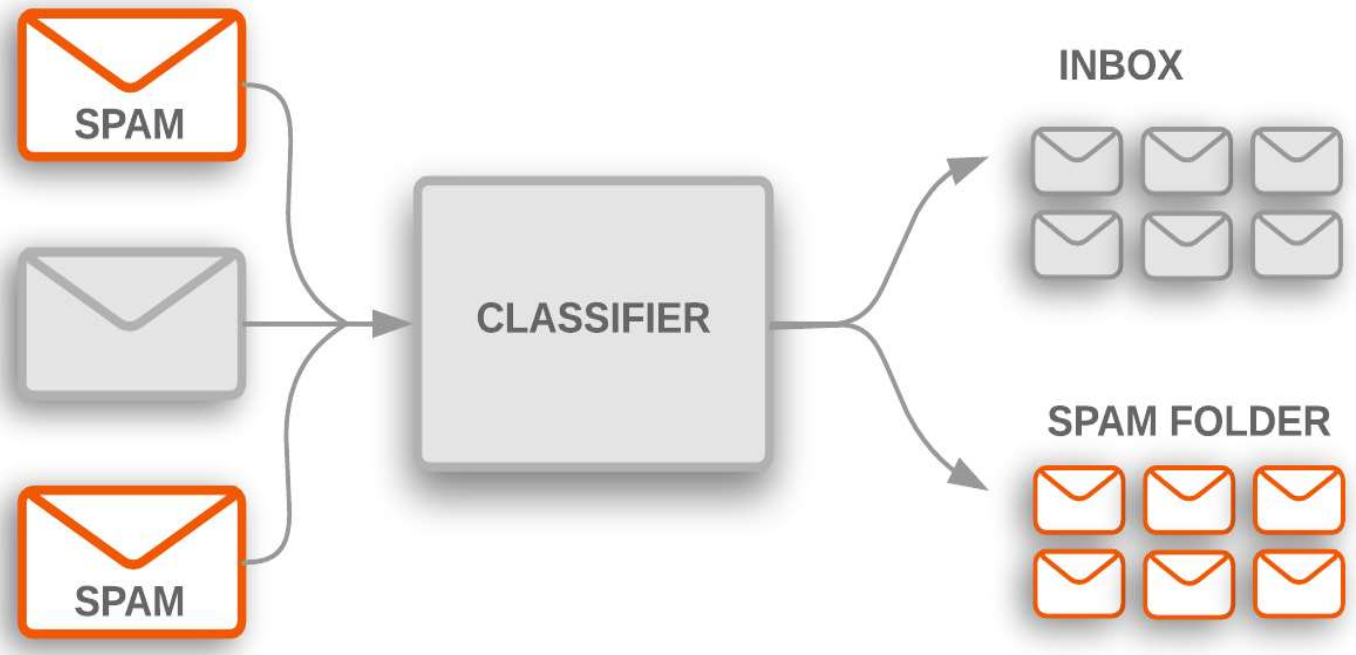
- Identify the risk factors for lung cancer.



COPD, Chronic obstructive pulmonary disease

Supervised Learning Problems

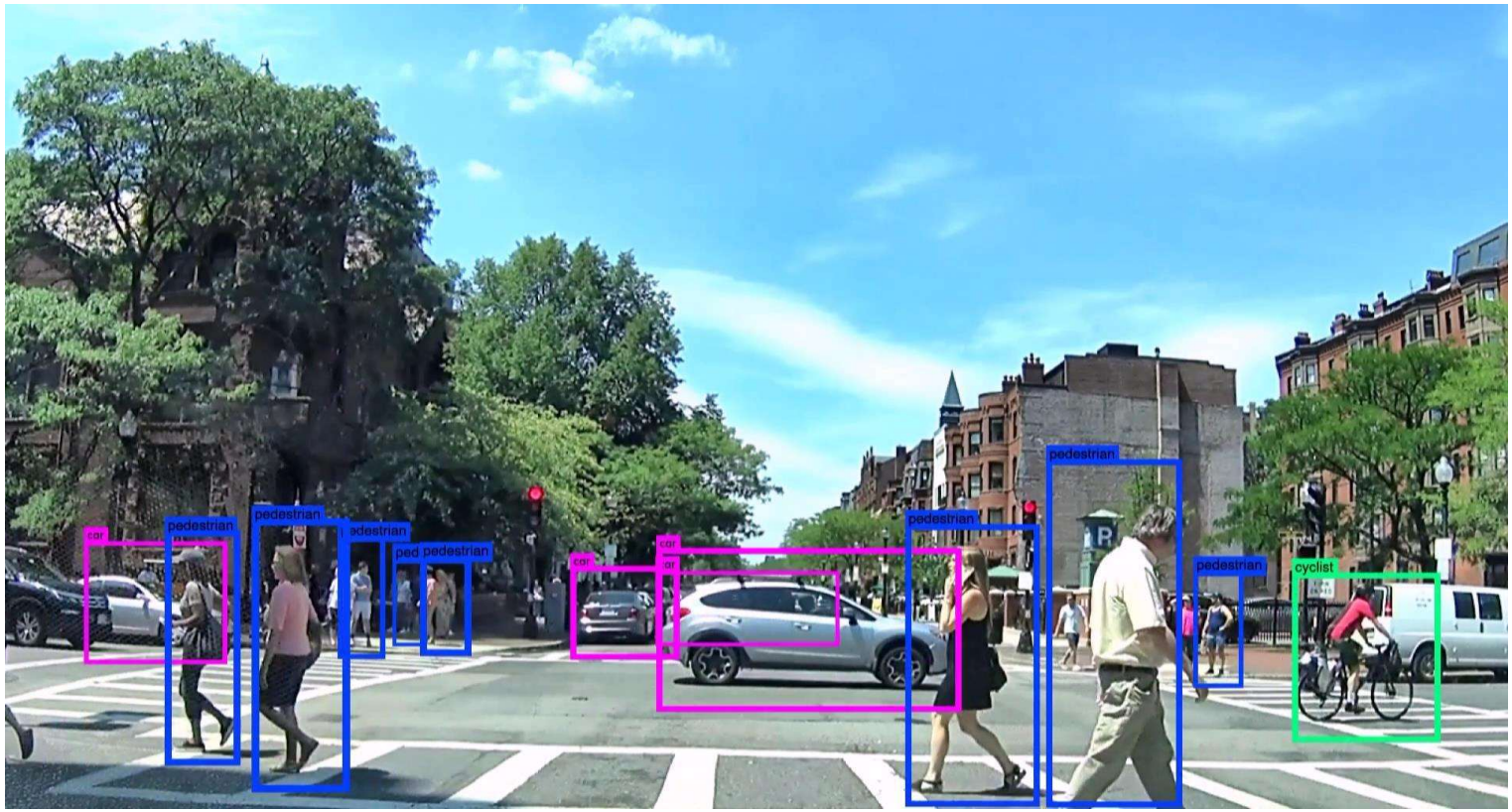
- Distinguish spam emails from important messages.



https://medium.com/@naveeen.kumar.k/naive-bayes-spam-detection-7d087cc96d9d

Supervised Learning Problems

- Train a self-driving car to detect and classify objects based on the data gathered by the vehicle's different sensors.



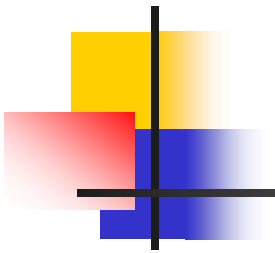
https://www.businesswire.com/news/home/20170118005301/en/Neurala-Announces-14-Million-Series-A-to-Bring-Deep-Learning-Neural-Network-AI-Software-to-Drones-Self-Driving-Cars-Toys-and-Cameras

Supervised Learning

Inputs	Output
Predictors	Response
Independent variables	Dependent variable
X	Y

$$Y = f(X) + e$$

Supervised learning refers to a set of approaches for estimating *f* for explanation and/or prediction



Supervised Learning

$$Y = f(X) + e$$

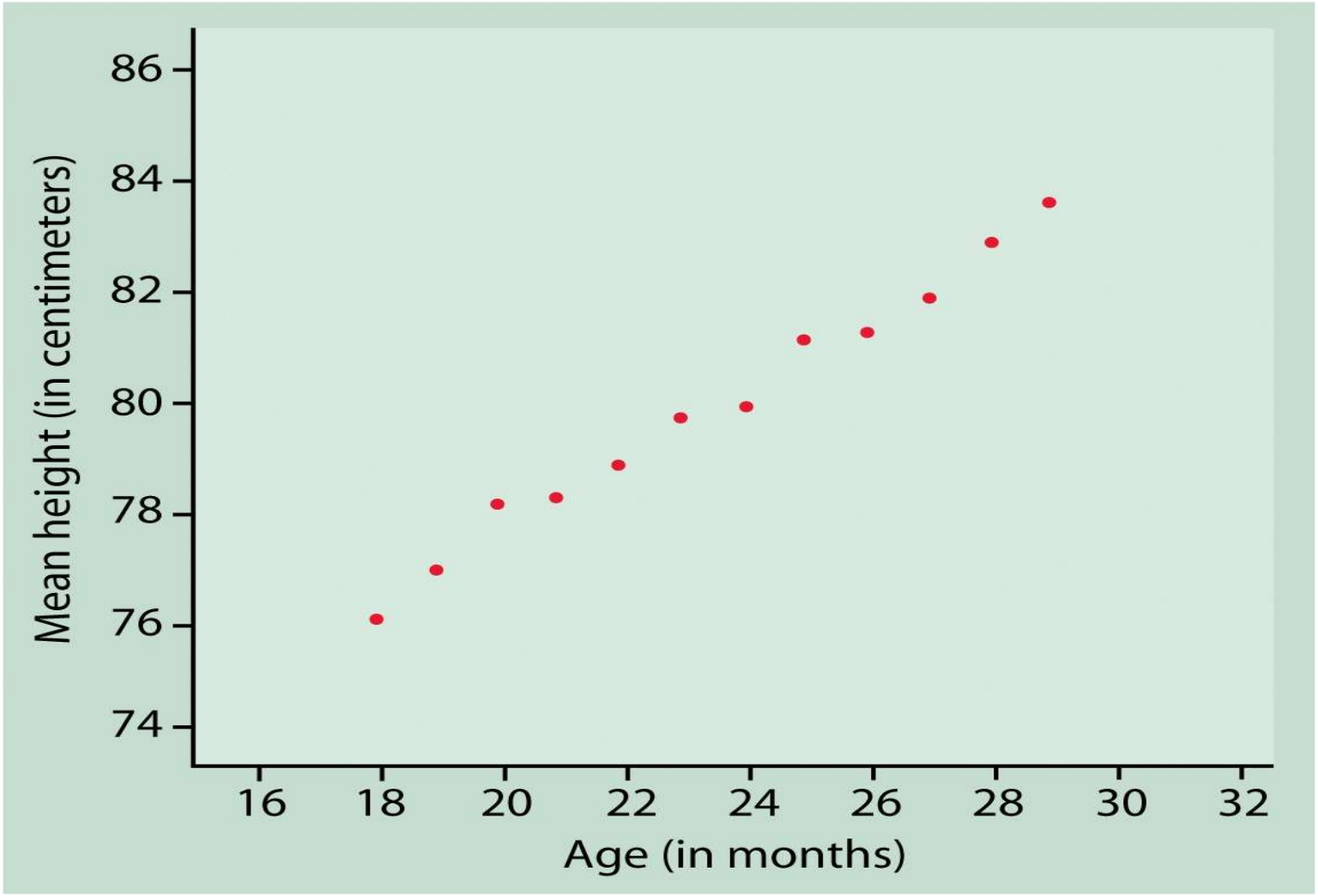
- *f* can be a linear or non-linear function.
- Parametric – we make an assumption about the form of *f*. For example,

$$f(X) = \beta_0 + \beta X$$

- Non-parametric – we do not make explicit assumptions about the form of *f*.

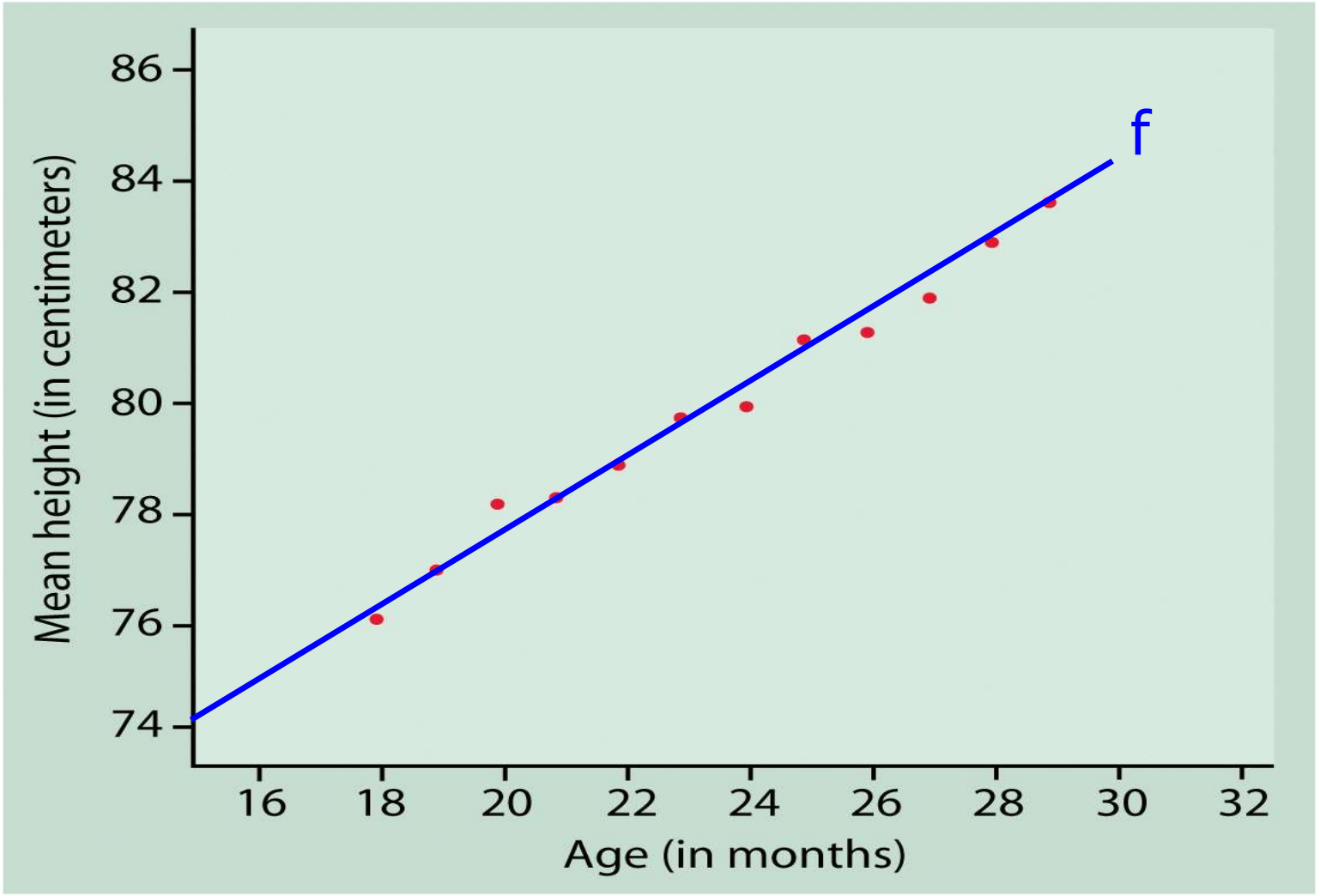
Supervised Learning

Kalama, Egypt



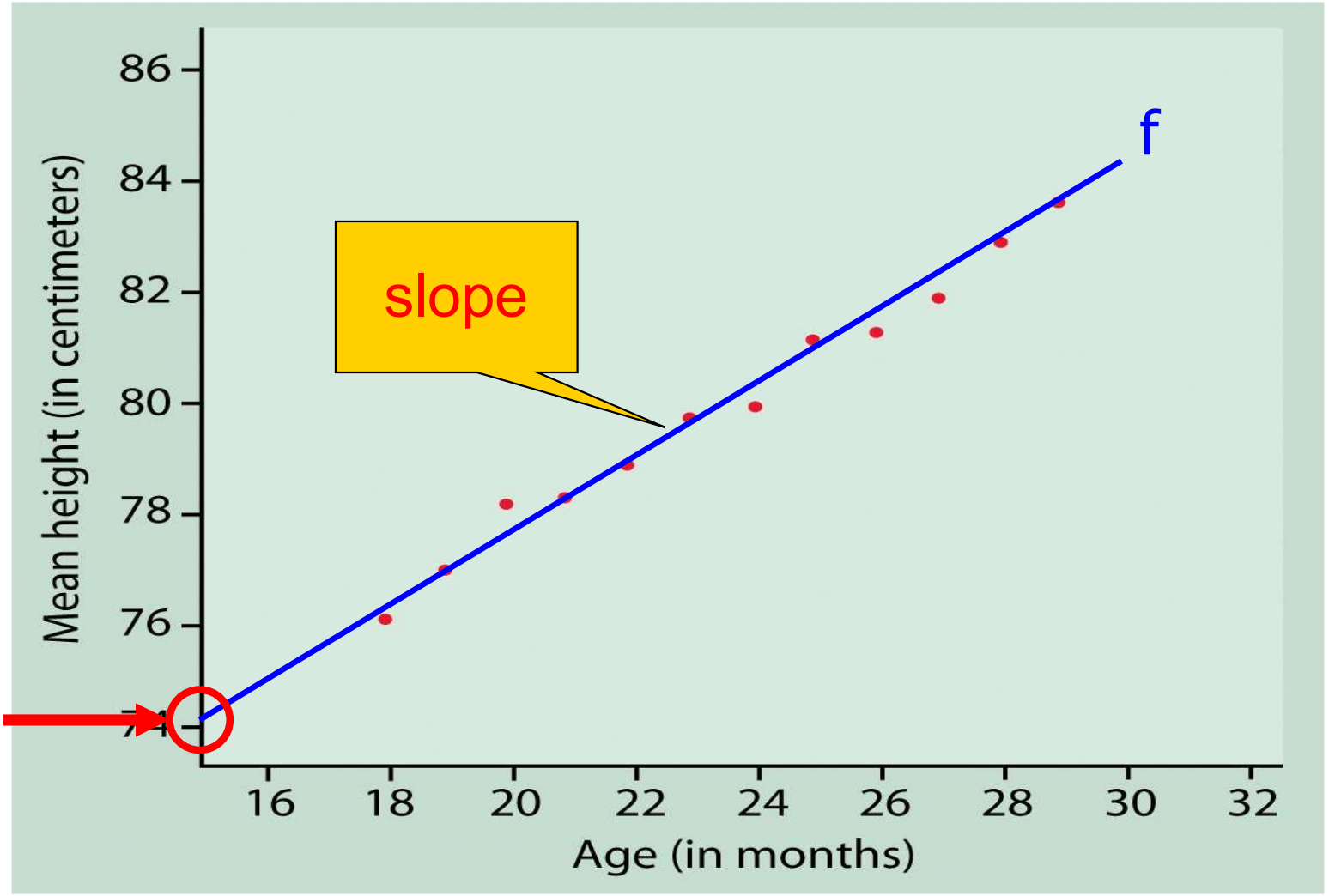
Moore & McCabe (1998, p.136)

Supervised Learning



Moore & McCabe (1998, p.136)

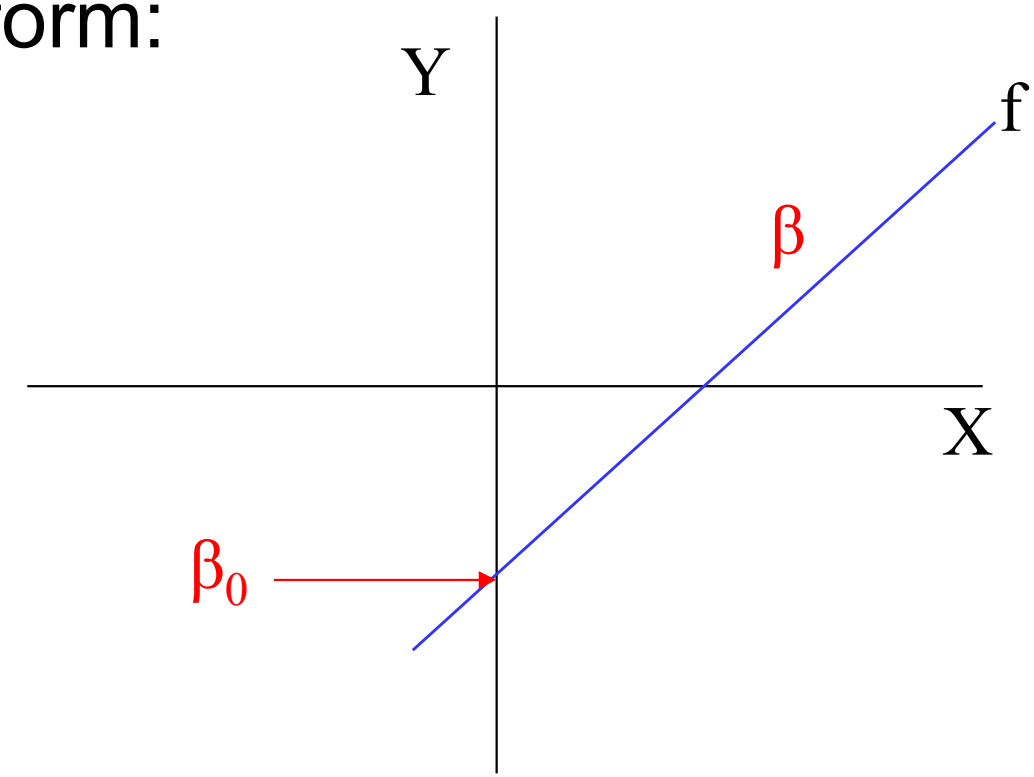
Supervised Learning



Moore & McCabe (1998, p.136)

Supervised Learning

- A straight-line relating Y to X has the following form:
  - $f(X) = \beta_0 + \beta X$
    - β_0 = intercept
    - β = slope



$$Y = f(X) + e$$
$$= \beta_0 + \beta X + e$$

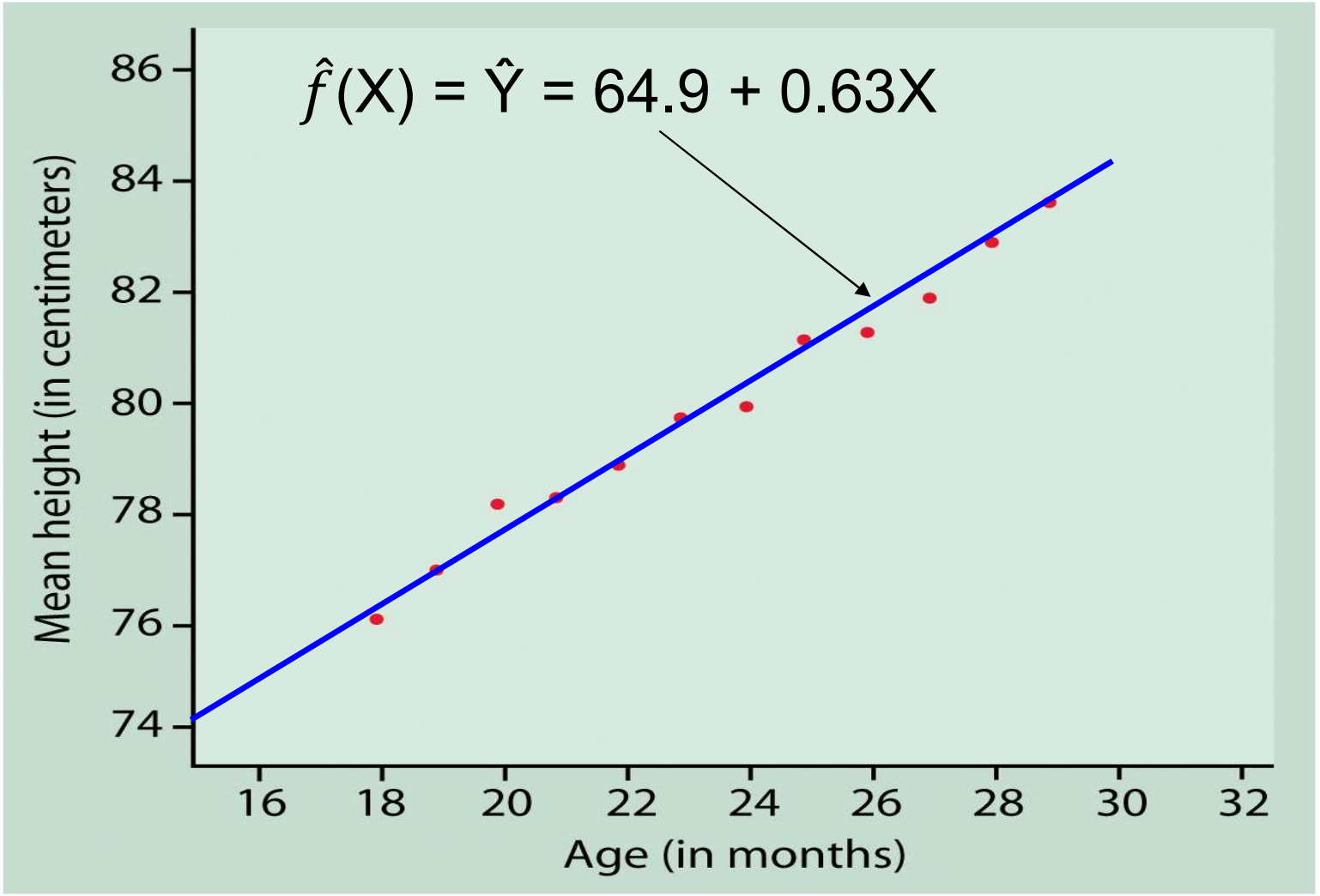
Training Data

Age X (month)	Height Y (cm)
18	76.1
19	77.0
20	78.1
21	78.2
22	78.8
23	79.7
24	79.9
25	81.1
26	81.2
27	81.8
28	82.8
29	83.5

Moore & McCabe (1998, p.136)

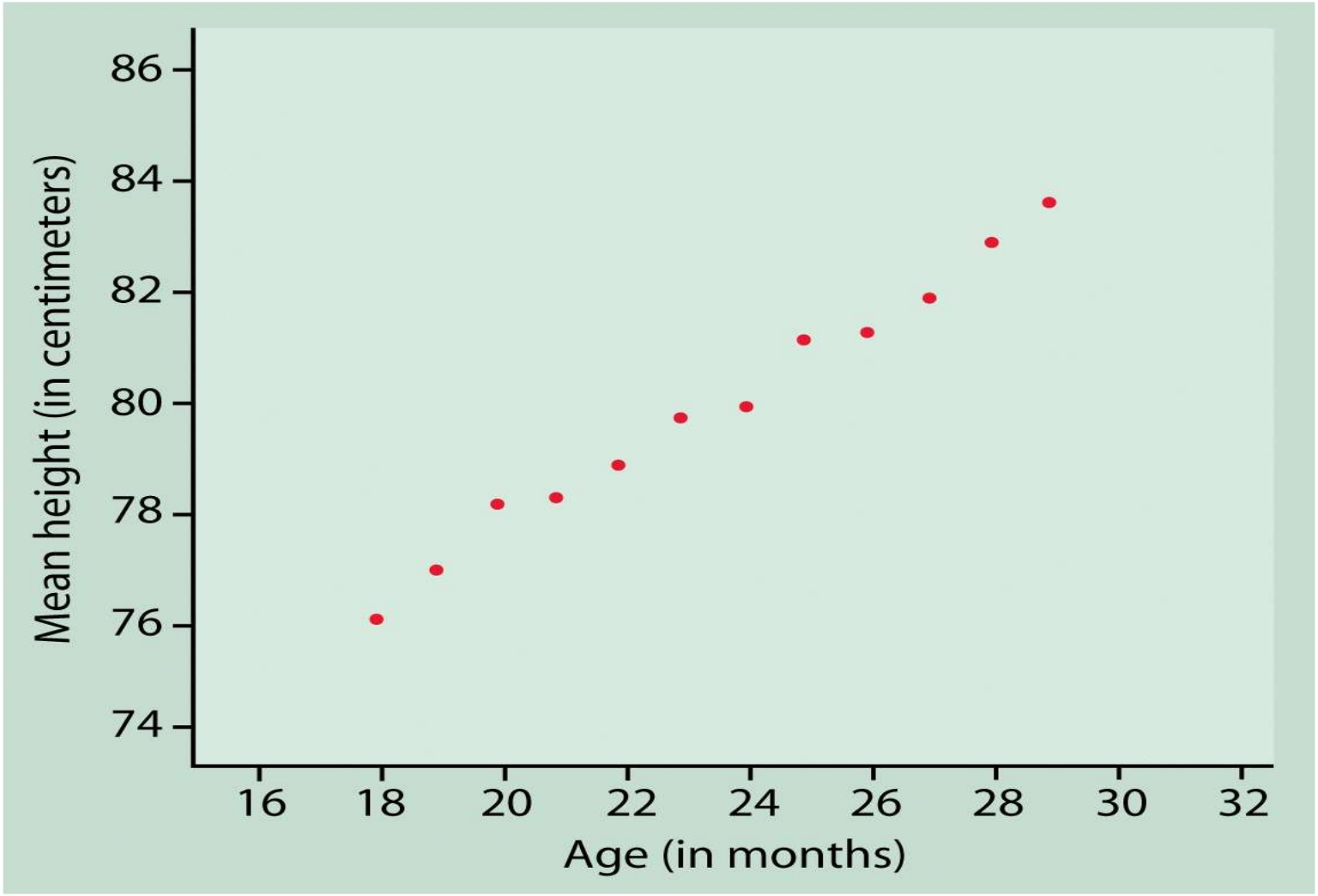
- These observations are called the training data because they are used to train or teach our method how to estimate the unknown function *f*.

Supervised Learning



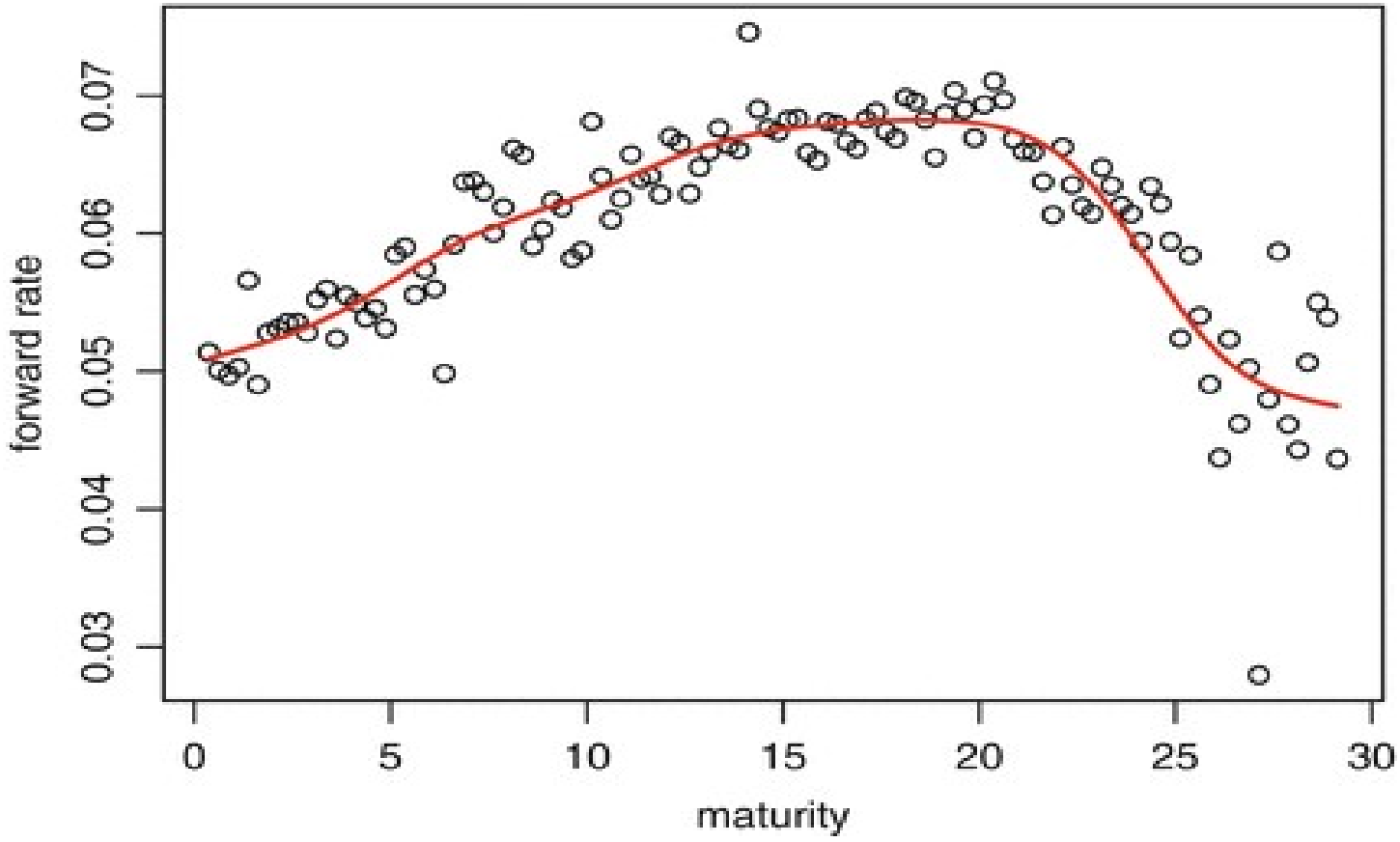
Moore & McCabe (1998, p.136)

Supervised Learning



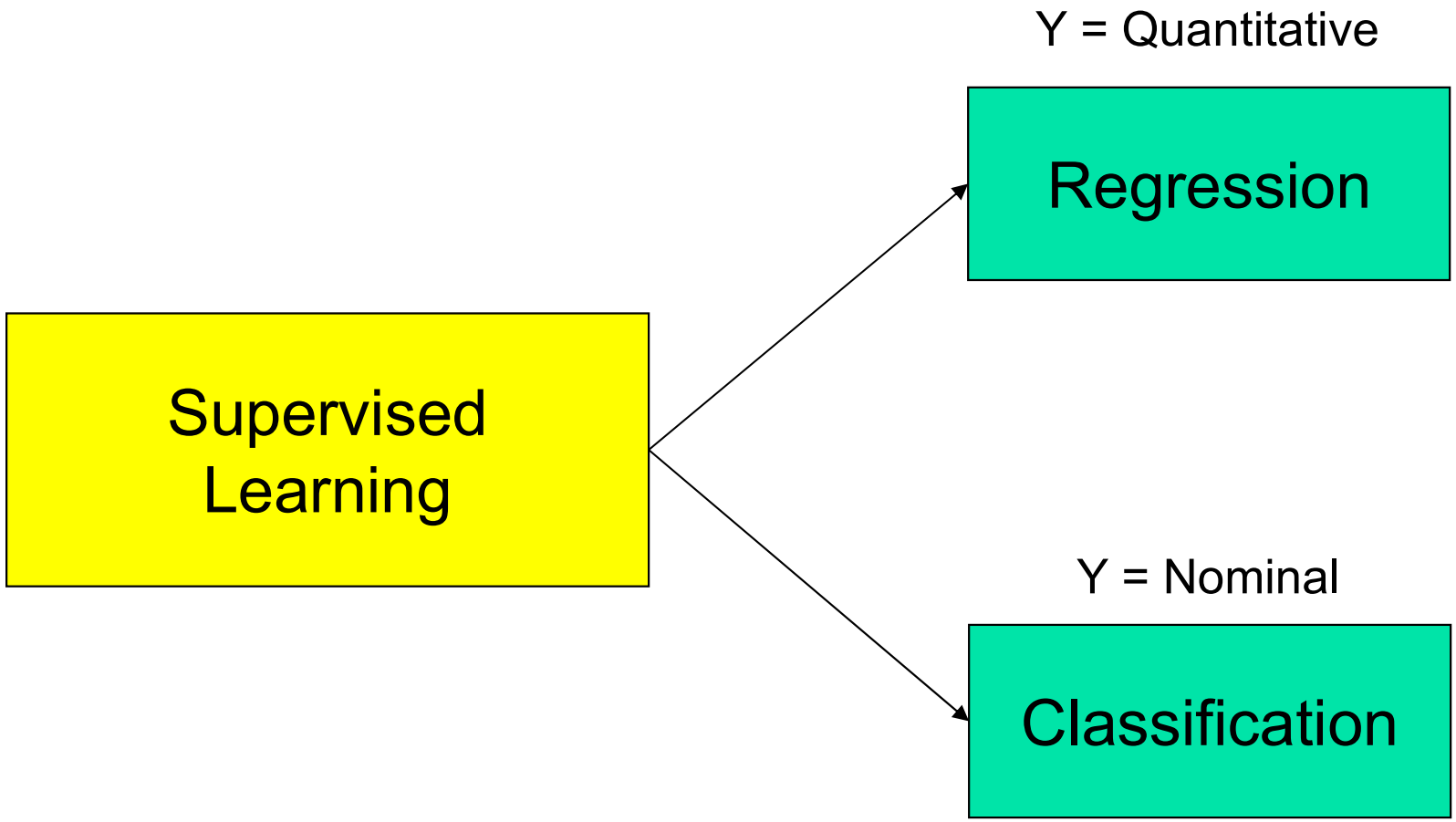
Moore & McCabe (1998, p.136)

Supervised Learning



Ruppert D. & Matteson D.S. (2015) Nonparametric Regression and Splines. In: Statistics and Data Analysis for Financial Engineering. Springer Texts in Statistics. Springer, New York, NY. https://doi.org/10.1007/978-1-4939-2614-5_21

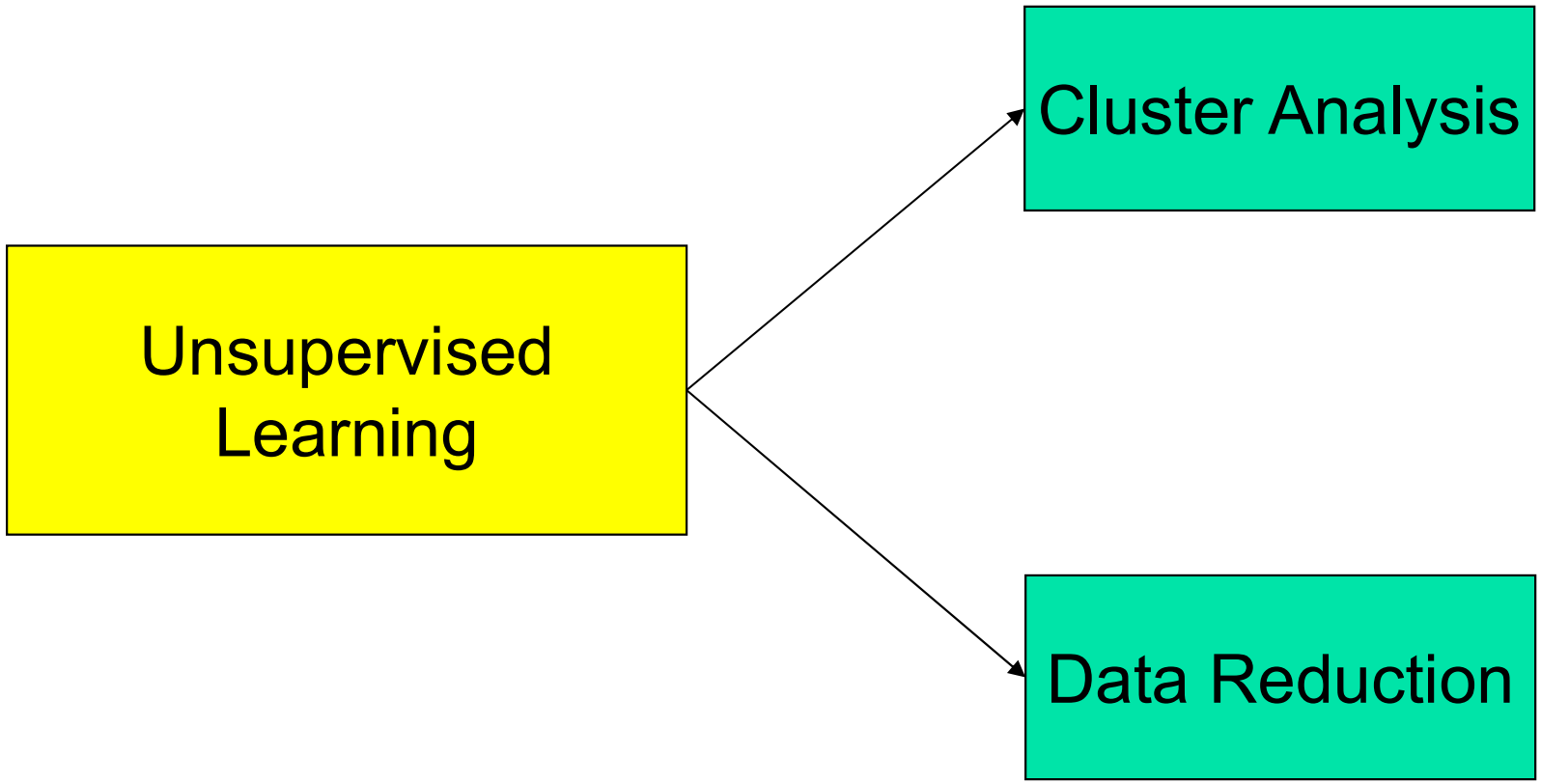
Supervised Learning

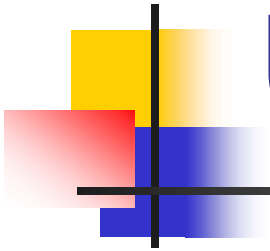


Supervised Learning Methods

- Linear Regression
- K-Nearest Neighbors Regression
- Logistic Regression
- Discriminant Analysis
- Naïve Bayes
- K-Nearest Nearest Neighbors
- Regularized Regression
- Tree-Based Methods
- Support Vector Machines

Unsupervised Learning





Unsupervised Learning

- No outcome variable, just a set of variables (features) measured on a set of samples.
- The objective is to find groups of samples that behave similarly, find features that behave similarly, or find linear combinations of features with the most variation.
- It can be useful as a pre-processing step for supervised learning.

Unsupervised Learning Methods

- Clustering Methods
  - K-means Clustering
  - Hierarchical Clustering
  - Finite Mixture Models
- Data Reduction Methods
  - Principal Components Analysis
  - Factor Analysis
  - Canonical Correlation Analysis

Unsupervised Learning Problems - Clustering

sample

Cluster/group

https://www.guru99.com/unsupervised-machine-learning.html#2

Unsupervised Learning Problems - Data Reduction: PCA

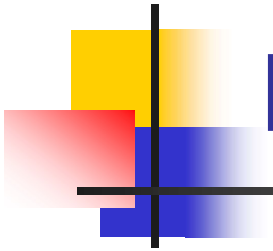
- Buchala, Davey, Gale, & Frank (2005)
  - A subset of FERET (Facial Recognition Technology) database
  - 2670 grey scale frontal face images

Property	No. Categories	Categories	No. Faces
Gender	2	Male	1603
		Female	1067
Ethnicity	3	Caucasian	1758
		African	320
		East Asian	363
Age	5	20 – 29	665
		30 – 39	1264
		40 – 49	429
		50 – 59	206
		60+	106
Identity	358	Individuals with 3 or more examples	1161

Unsupervised Learning Problems - Data Reduction: PCA

- Each image is pre-processed to a 65 X 75 resolution
- Aligned based on eye locations
- Cropped such that little or no hair information is available

Unsupervised Learning Problems - Data Reduction: PCA

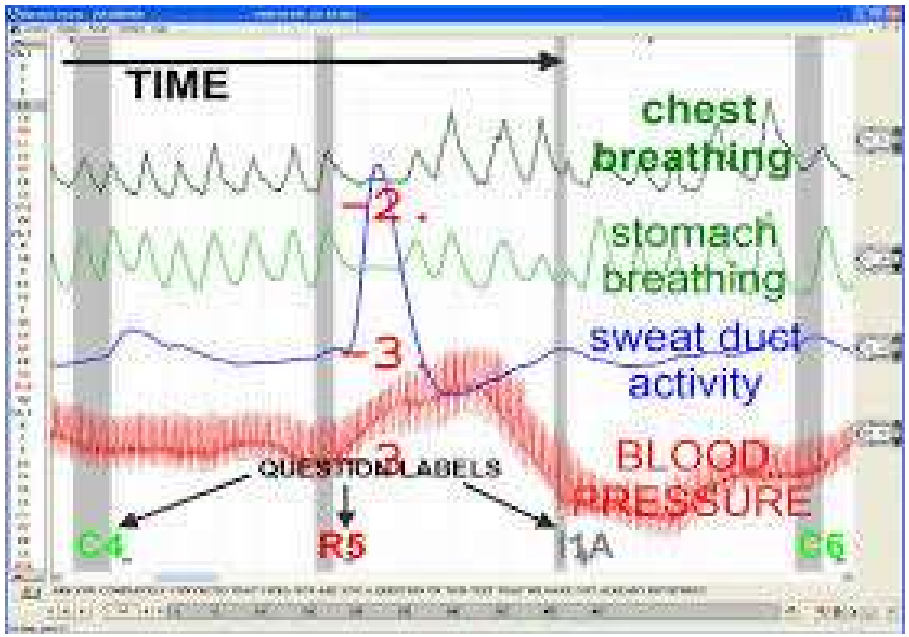


- Images of 65 × 75 resolution leads to a dimensionality of 4875 (= num of variables)
- The first 350 components accounted for 90% variance of the data
- Each face is thus represented using 350 components instead of 4875 dimensions
- In this example, D = 350 << P = 4875

Unsupervised Learning Problems - Data Reduction: PCA

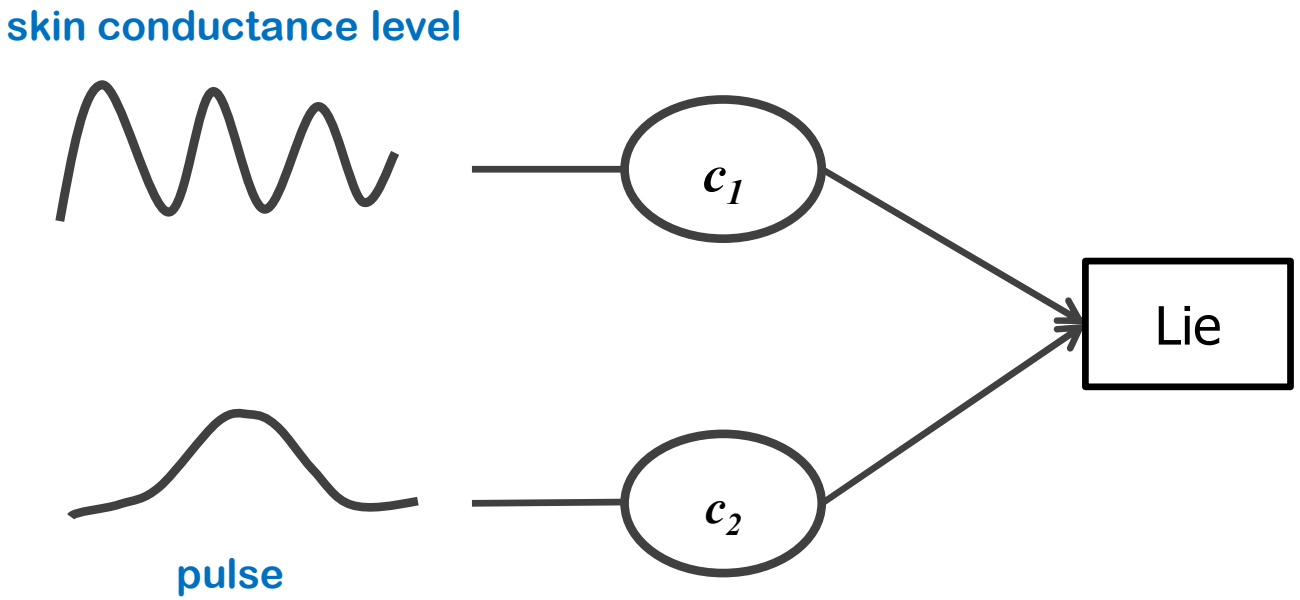
- By a single component (3rd) – related to gender

The Lie Detection Example



The Lie Detection Example

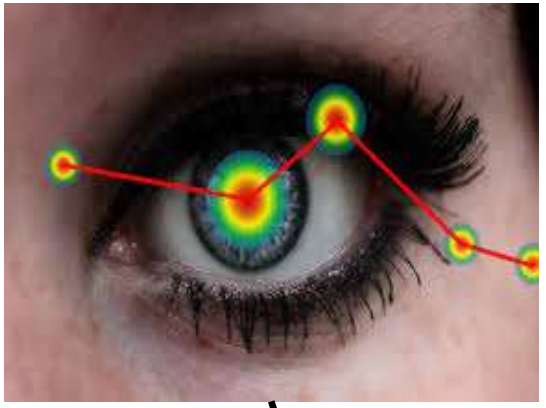
- We examine how the components of two intensive longitudinal predictors “skin conductance level” and “pulse” affect a binary outcome (1 = lies and 0 = truth).



(Hwang et al., 2015, *Psychometrika*)

Device

Data



c_1 c_2 ... c_k c_K

Lie (1/0)

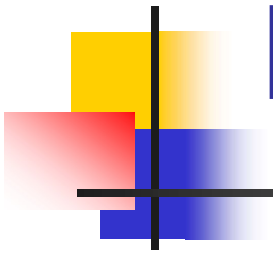
https://www.r-project.org/

https://www.rstudio.com/

Session 2
Linear Regression

An Introduction to Machine Learning for the Behavioural and Social Sciences

Heungsun Hwang & Gyeongcheol Cho

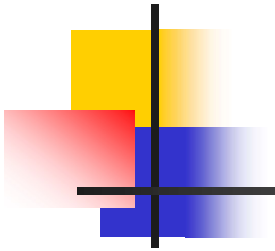


Linear Regression

- Linear regression is a very simple approach for supervised learning.
- It is a good starting point for more complex supervised learning tools.

Linear Regression

- It is useful for explaining or predicting a quantitative response (DV).
  - Two main goals:
    - To investigate the relationship between a DV and predictors
    - To find the best prediction equation for a DV regardless of the meaning of predictors in the equation

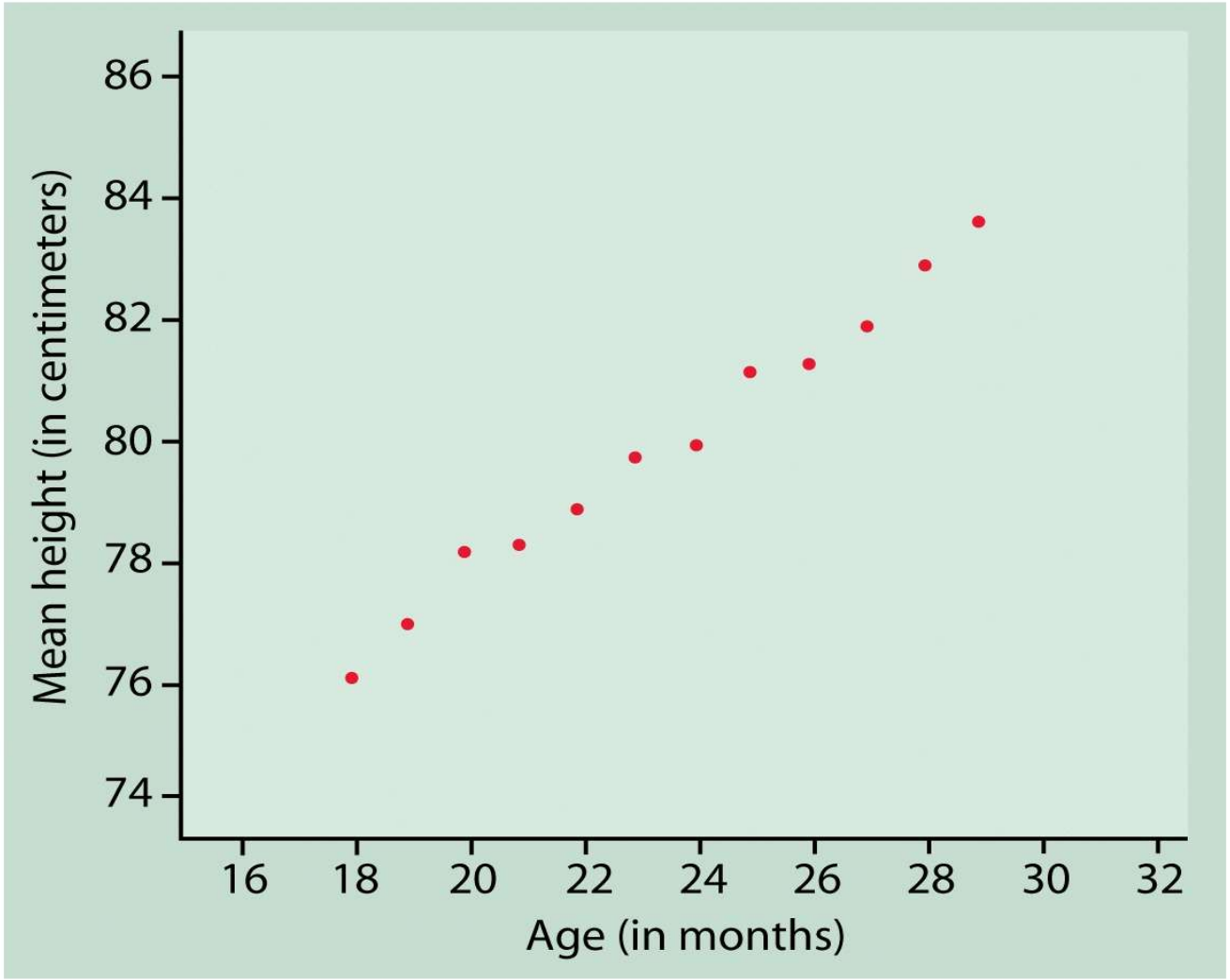


Linear Regression

- Linear regression assumes that there is approximately a “linear” relationship between predictors and a response.

$$Y = f(X) + e$$

Linear Regression - Linearity



Moore & McCabe (1998, p.136)

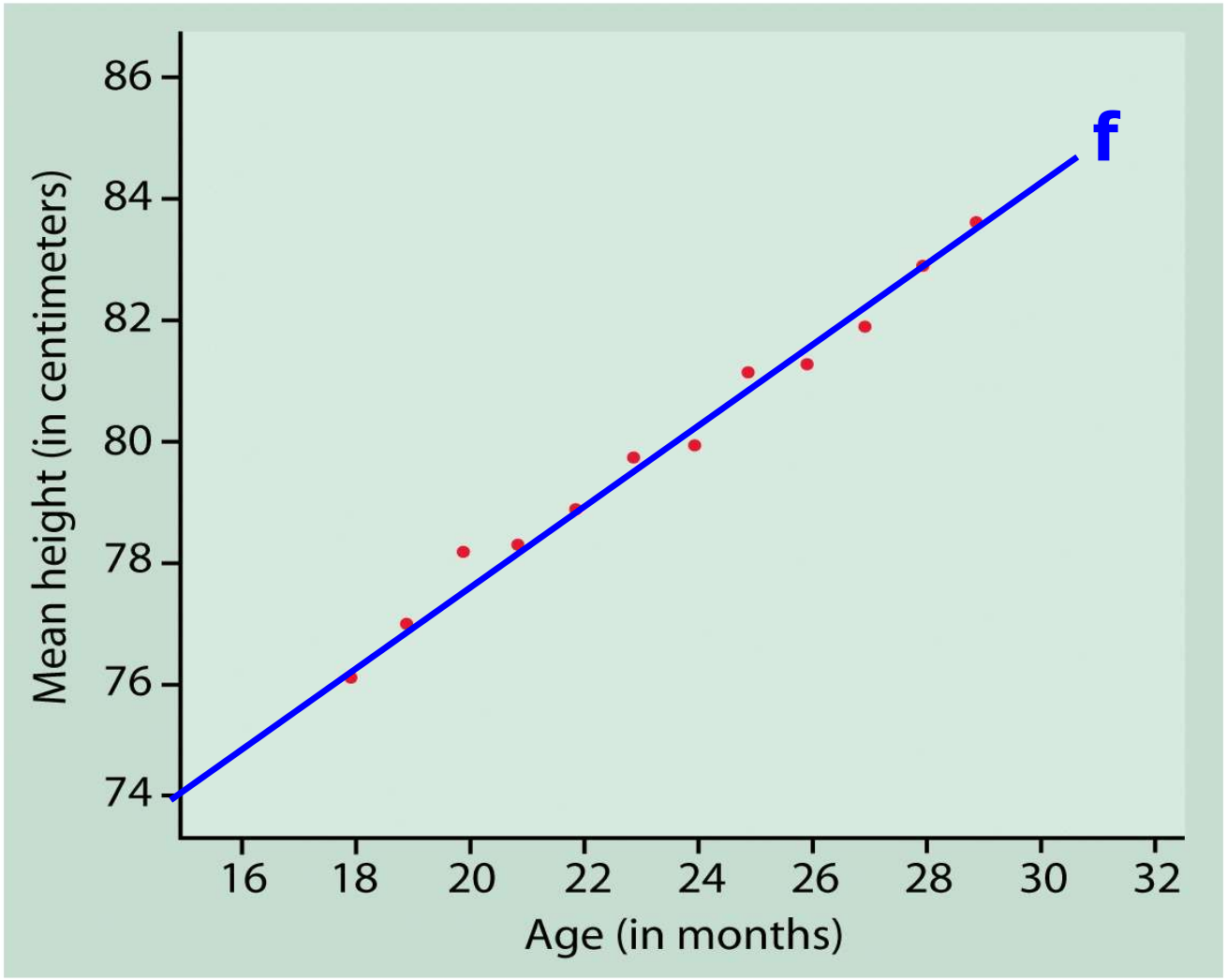
Simple Linear Regression

- When there is a linear relationship between the DV (Y) and a single predictor (X), we can represent this relationship by a straight line, called a regression line.

- A regression line gives a compact description of the dependency of Y on X.

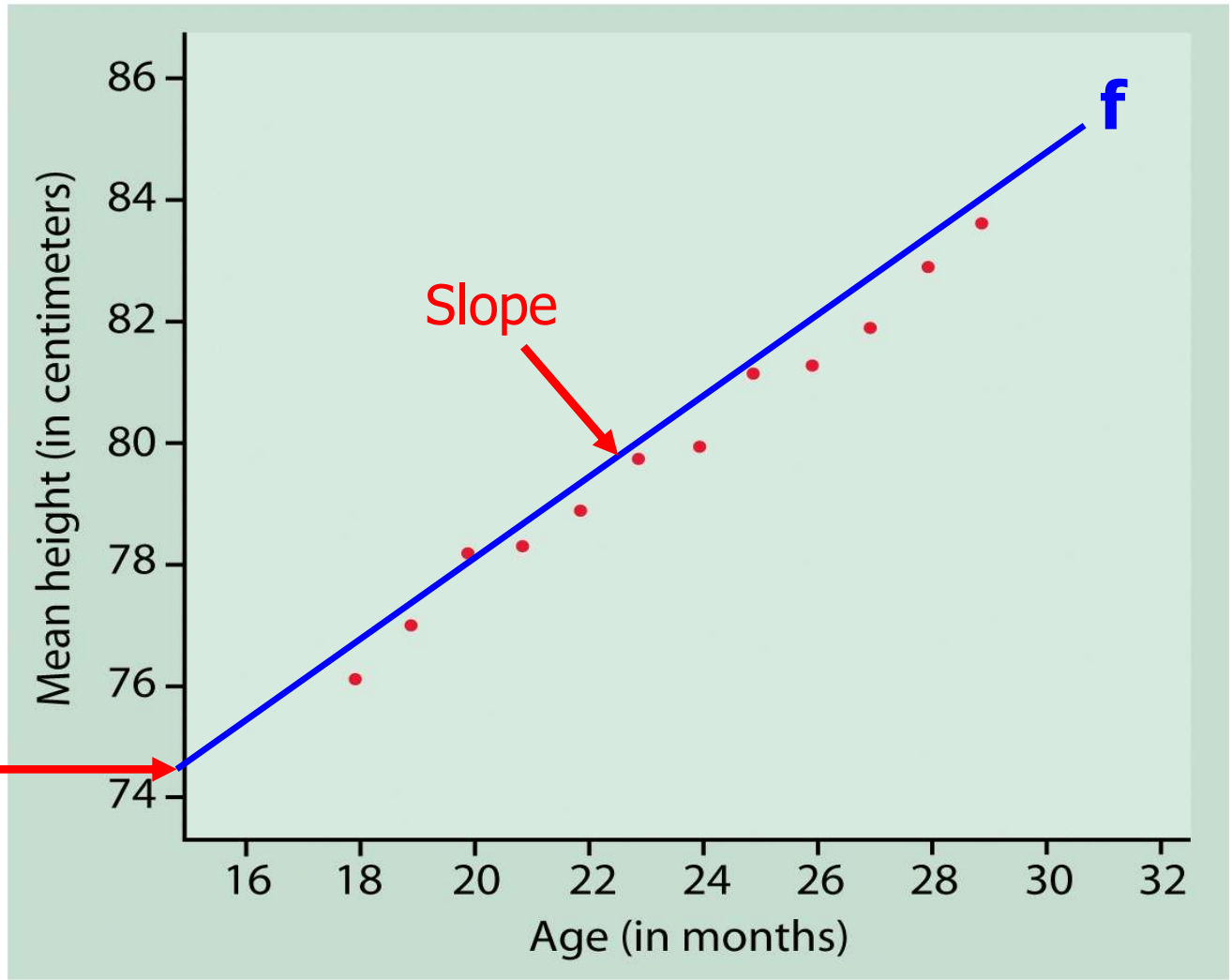
- The equation of a line is a mathematical model for the linear relationship – simple linear regression model.

Simple Linear Regression



Moore & McCabe (1998, p.136)

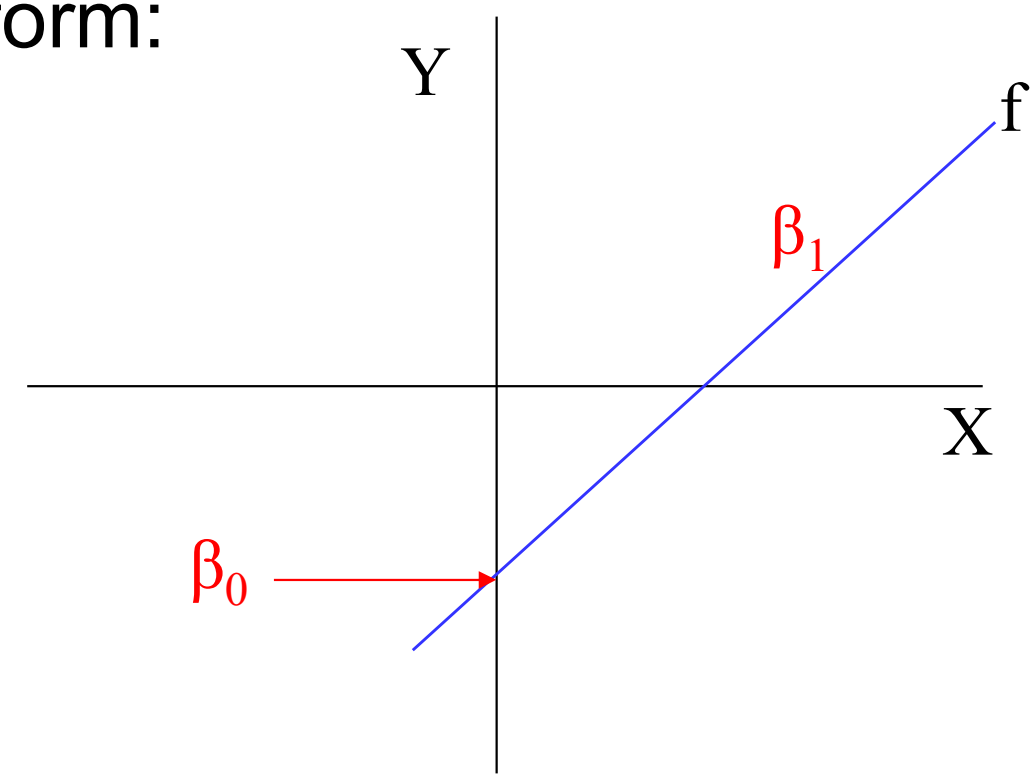
What is the "equation" of a line?



Moore & McCabe (1998, p.136)

Simple Linear Regression Model

- A straight-line relating Y to X has the following form:
  - $f(X) = \beta_0 + \beta_1 X$
    - β_0 = intercept
    - β_1 = slope



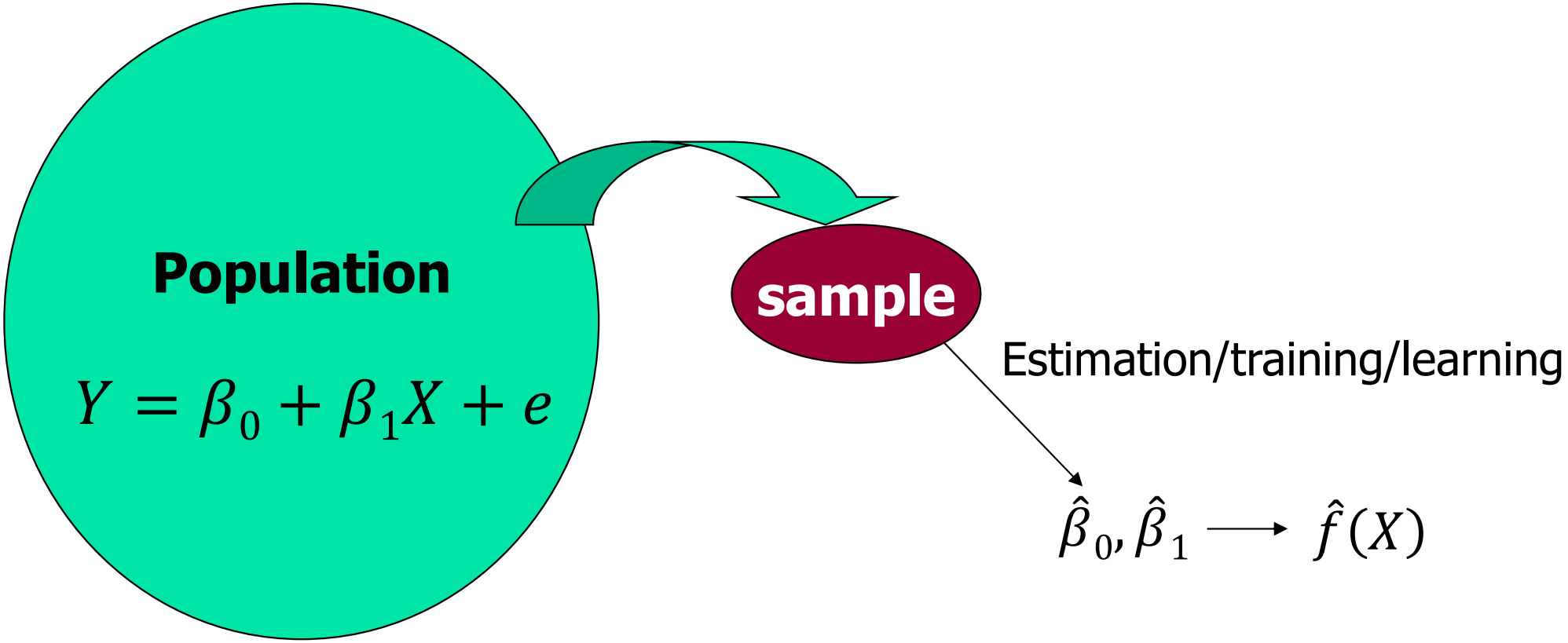
$$Y = f(X) + e$$
$$= \beta_0 + \beta_1 X + e$$

Simple Linear Regression Model

$$Y = f(X) + e$$
$$= \beta_0 + \beta_1 X + e$$

- β_0 = Average value of Y when X = 0.
- β_1 = Amount by which Y changes on average when X changes by one unit.

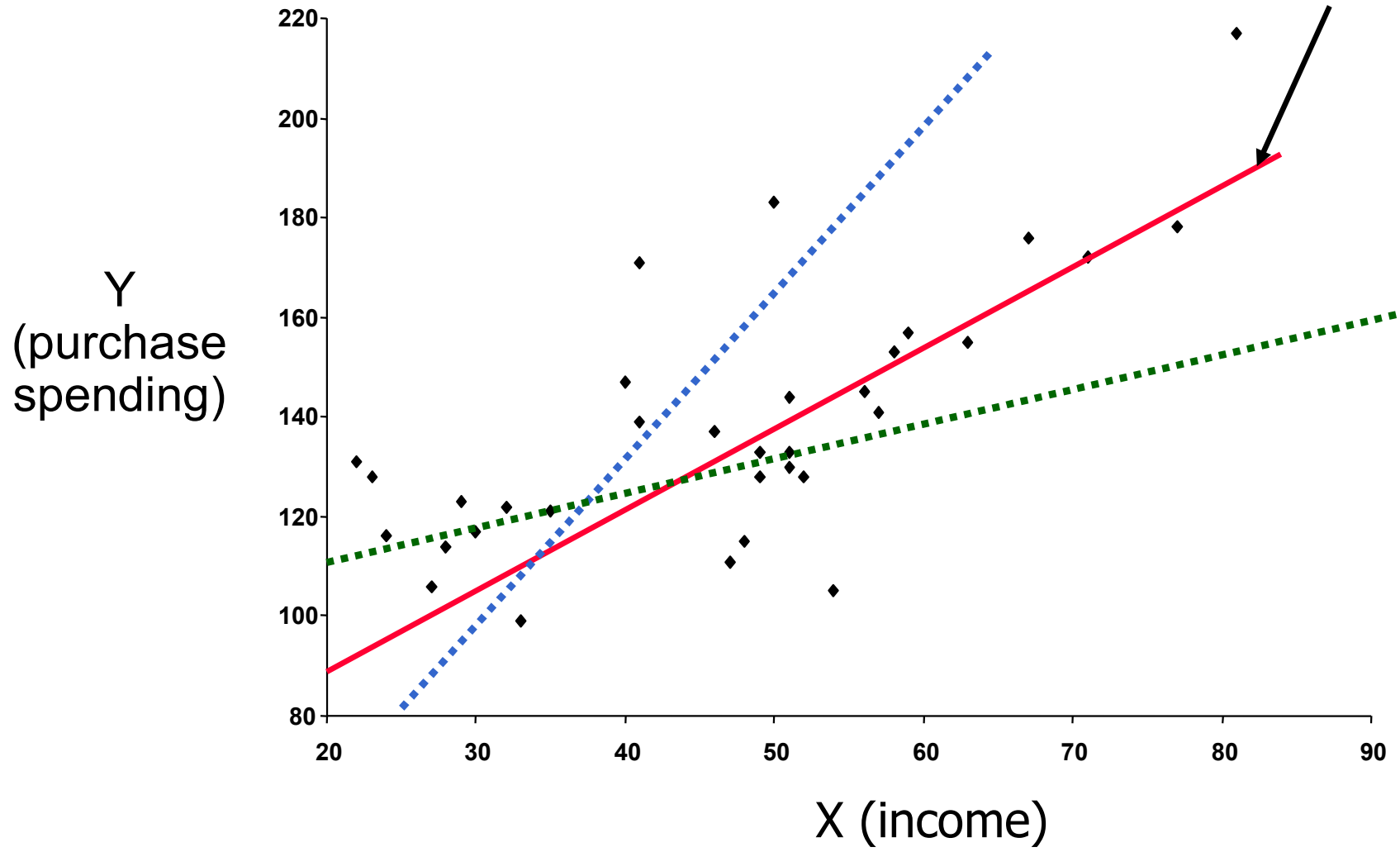
Parameter Estimation - How to determine the best regression line?



Training data/sample

Simple Linear Regression Model

$$\hat{f}(X) = \hat{Y} = 81.54 + 1.22X$$

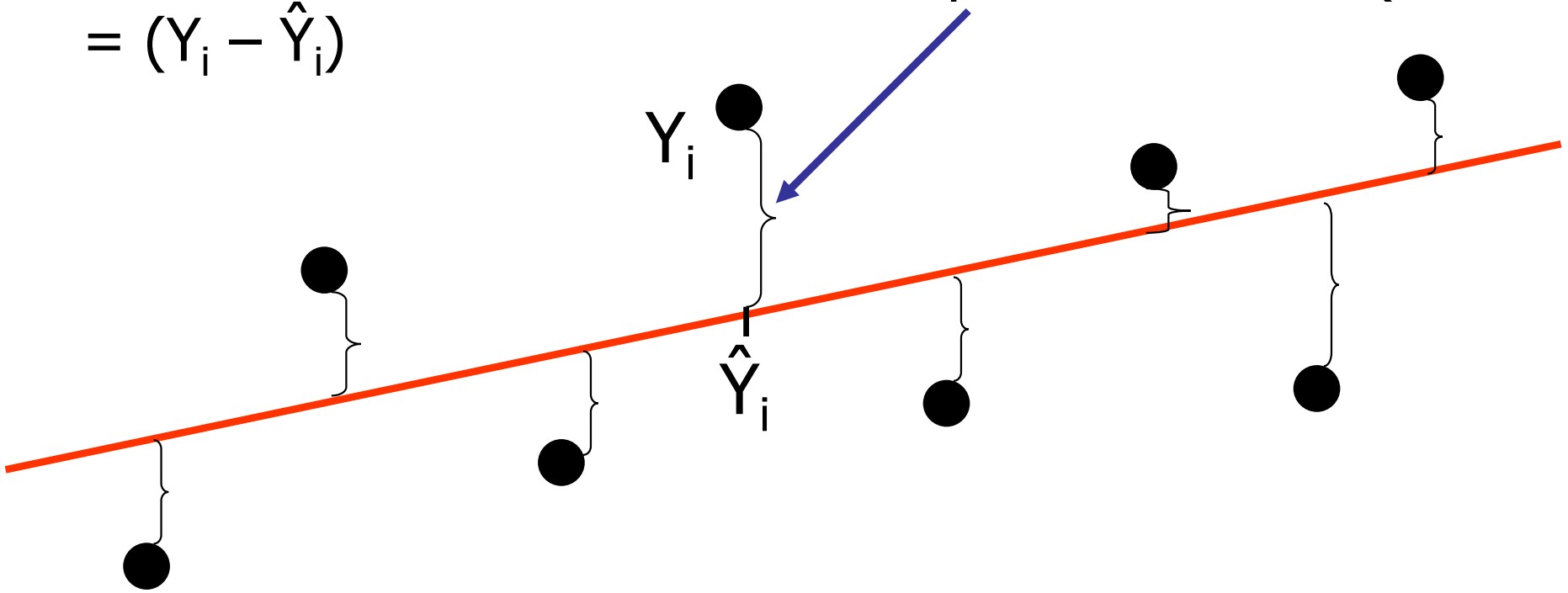


Parameter Estimation - How to determine the best regression line?

- The “best” regression line is the one that comes the closest to the data points in the vertical direction. There are many ways to make this distance “as small as possible.”
  - **The method of least squares** – the most common method. The least-squares regression line of Y on X is the line that makes the sum of the squares of the vertical distances of the data points from the line as small as possible.

Concept of Least Squares

vertical distance between a data point and a line (residual)
$= (Y_i - \hat{Y}_i)$



Sum of the **S**quares of all residuals = SS(Residual) = $\sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2$

Concept of Least Squares

- The least-squares regression line of Y on X is determined in such a way that it makes SS(Residual) as small as possible.

$$\text{SS(Residual)} = \sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} \left(Y_i - \beta_0 - \beta_1 X_i \right)^2$$

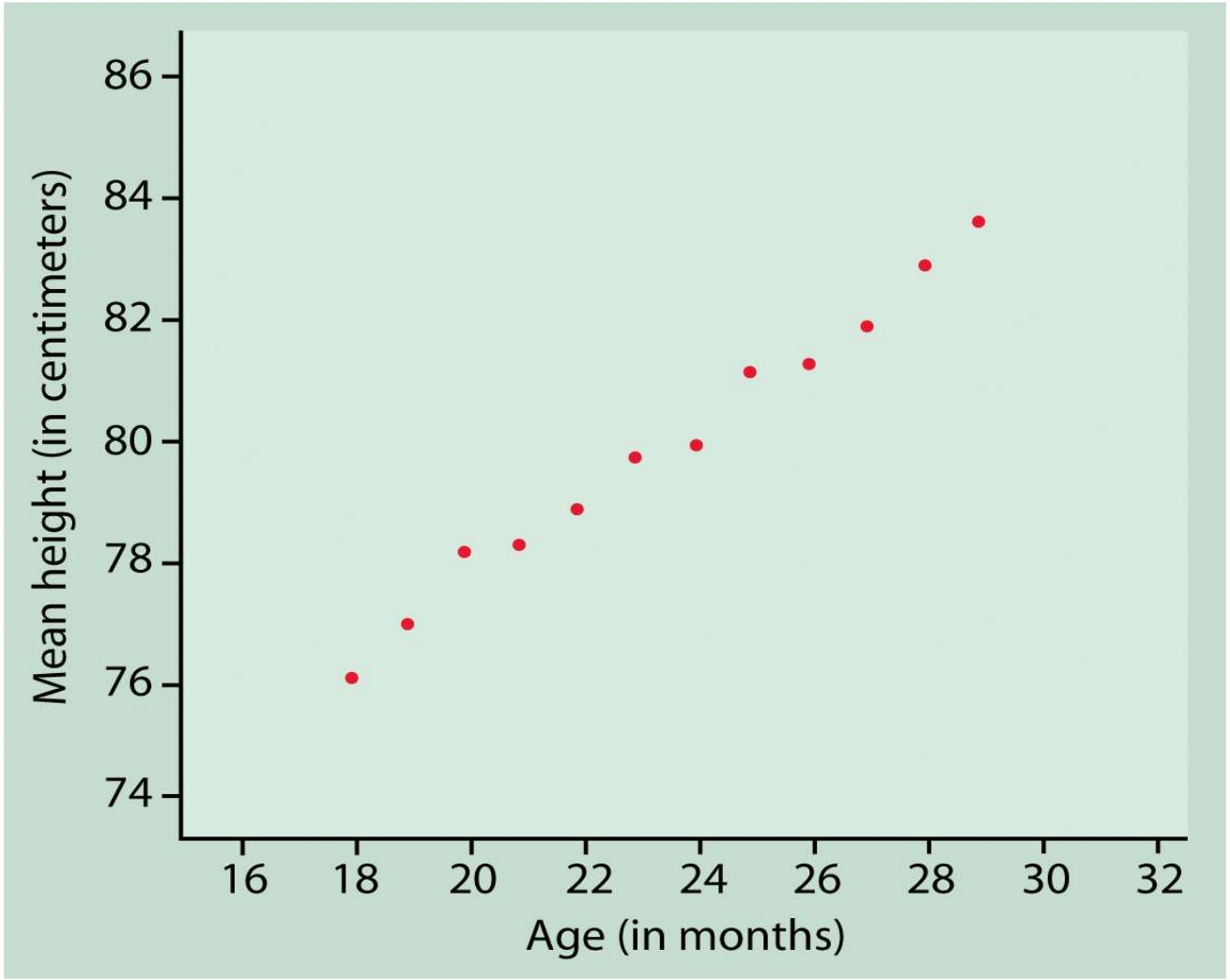
- β_0 and β_1 are estimated to minimize SS(Residual).
- In other words, this method aims to minimize the unexplained portion of Y by the regression line.

Least Squares Coefficient Estimates

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{N}\left(\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)\right)}{\sum_{i=1}^{N}\left(X_i - \overline{X}\right)^2}$$

$$\hat{\beta}_0 = \overline{Y} - \hat{\beta}_1 \overline{X}$$

Example: LS Coefficient Estimates



Moore & McCabe (1998, p.136)

Example: LS Coefficient Estimates

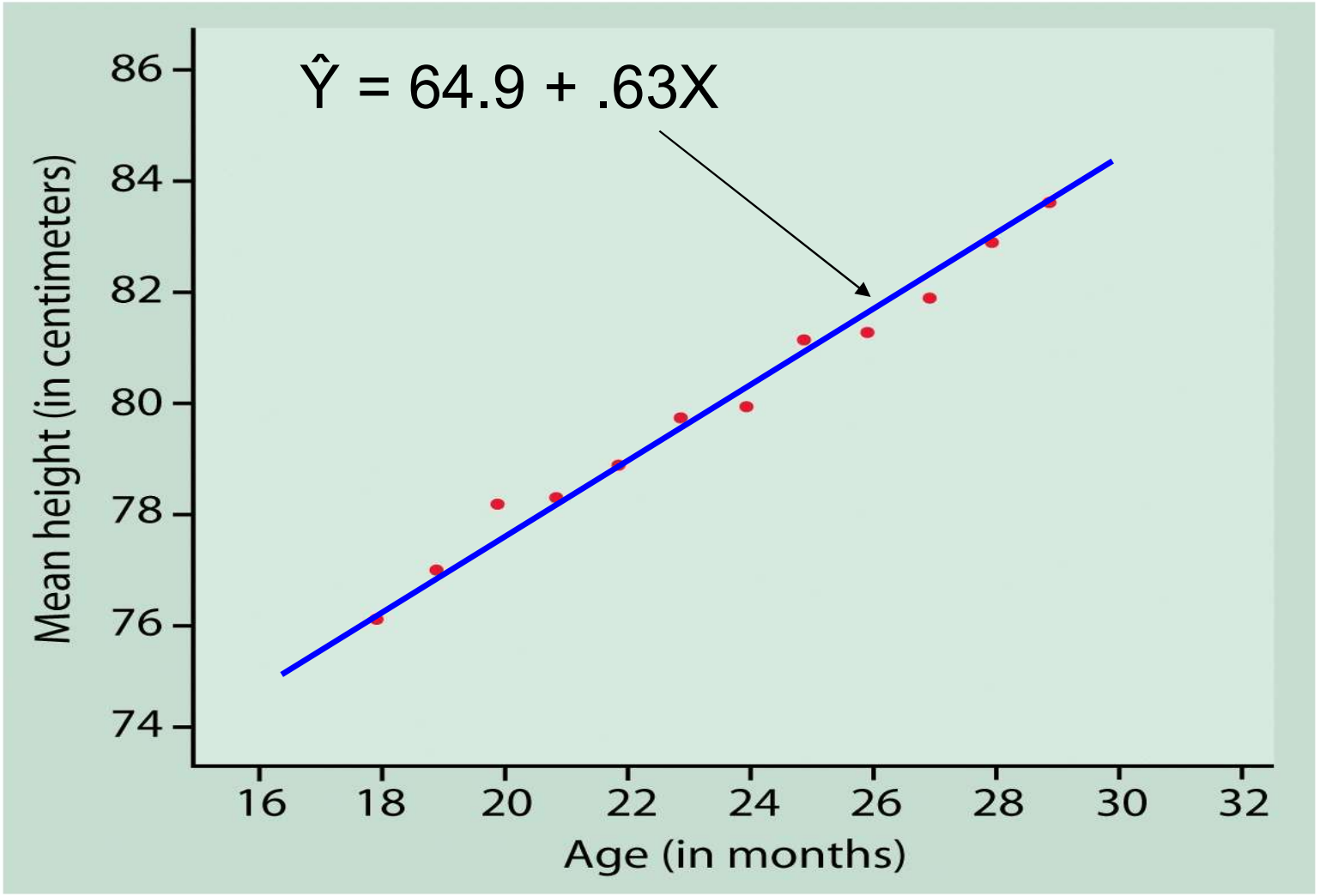
Age X (month)	Height Y (cm)
18	76.1
19	77.0
20	78.1
21	78.2
22	78.8
23	79.7
24	79.9
25	81.1
26	81.2
27	81.8
28	82.8
29	83.5

Moore & McCabe (1998, p.136)

- N = 12
- $\bar{X}$ = 23.5
- $\bar{Y}$ = 79.85

- $\hat{\beta}_1$=.6348
- $\hat{\beta}_0$= 64.93

Example: LS Coefficient Estimates



Moore & McCabe (1998, p.136)

Statistical test for the significance of slope

- We can perform a hypothesis test on the relationship between X and Y in simple linear regression (the effect of X on Y)
  - $H_0 : \beta_1 = 0$
    - There is no linear relationship between X and Y (no effect of X on Y)
  - $H_1 : \beta_1 \neq 0$
    - There is a linear relationship (an effect of X on Y)

Statistical test for the significance of slope

- We compute a t-statistic, given by

$$t = \frac{\hat{\beta}_1 - \beta_1}{SE(\hat{\beta}_1)}$$

where $SE(\hat{\beta}_1)$ is the standard error of the estimate.

Statistical test for the significance of slope

- $SE(\hat{\beta}_1)$ is computed as follows:

$$SE(\hat{\beta}_1) = \sqrt{\frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}{N-2}} \Bigg/ \sqrt{\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

Statistical test for the significance of slope

- If the p-value of the t statistic is small enough, e.g., $p < \alpha = .05$ (or .01), we may reject the null hypothesis.

- This indicates that the slope is different from zero, suggesting a statistically significant effect of X on Y.

Statistical test for the significance of slope

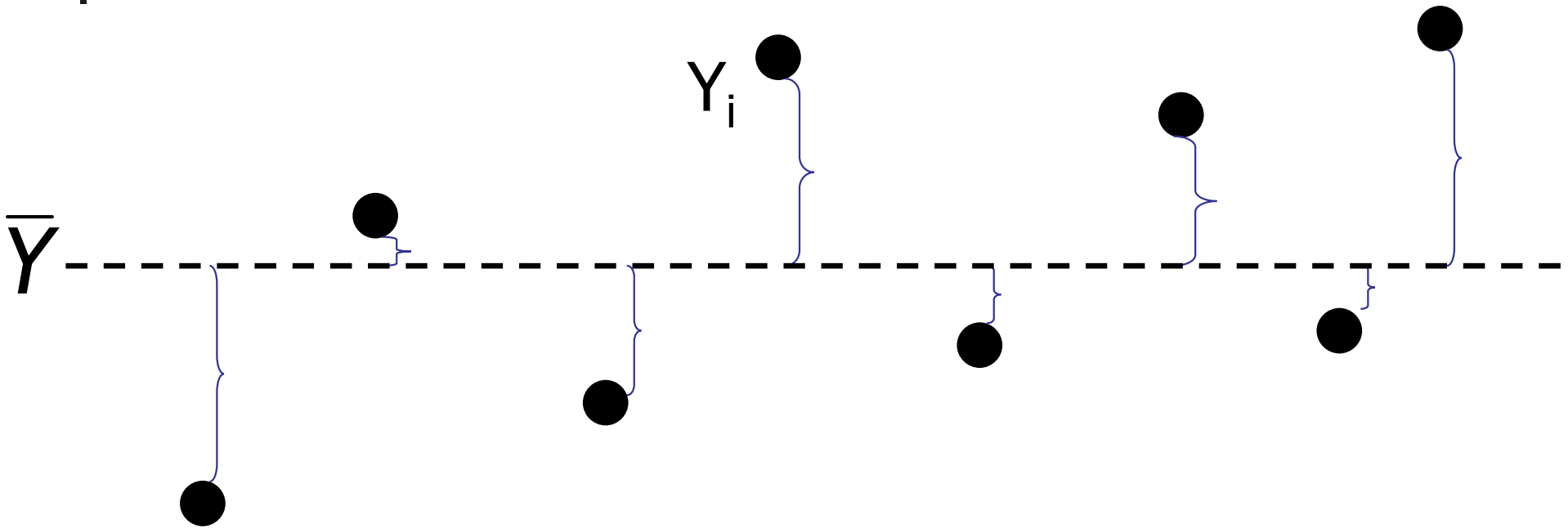
- To apply the *t* test for the slope, the following assumptions are required:
  - Normal distribution
  - Independent observations

Partitioning Variation: Simple Linear Regression

- As in ANOVA, we can also divide the variance/variation in the DV (Y) into different parts resulting from different sources.

- In regression analysis, the total variation in Y is partitioned into:
  - **SS(Regression):** The variation in Y explained by the regression line.
  - **SS(Residual):** The variation in Y unexplained by the regression line (residuals).

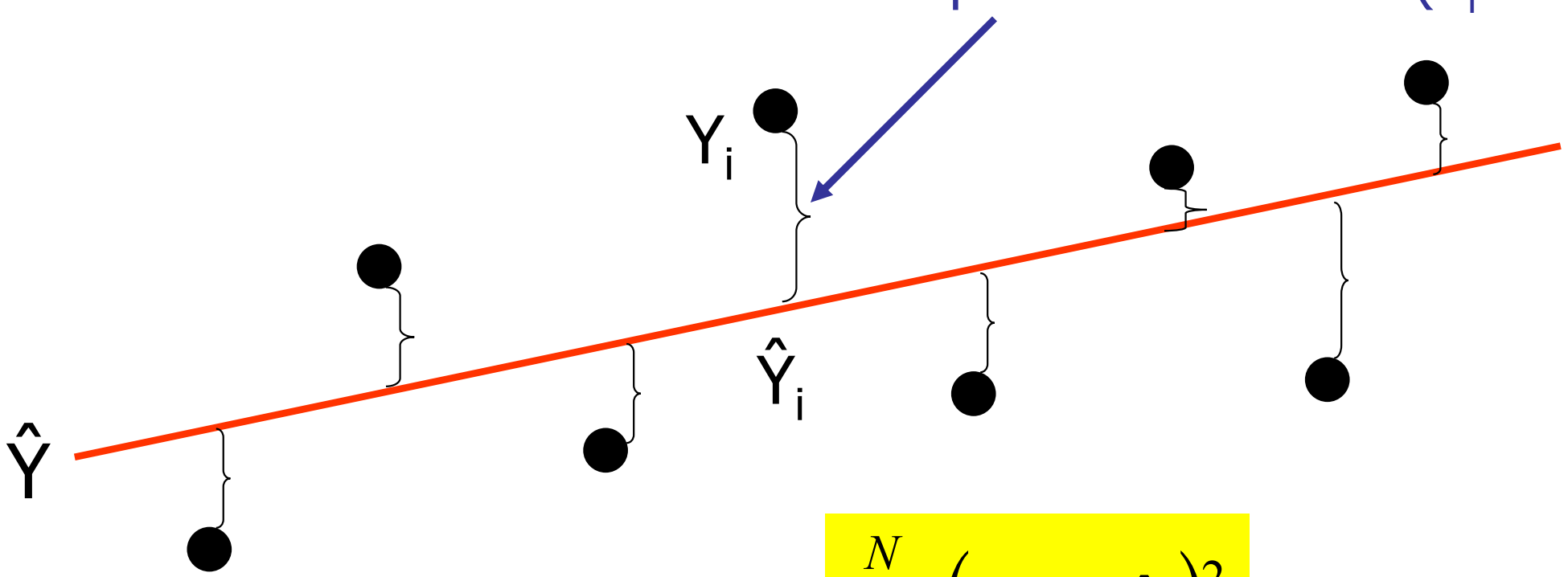
Partitioning Variation: SS(T)



$$\mathbf{SS(T)} = \sum_{i=1}^{N} \left(Y_i - \bar{Y}\right)^2$$

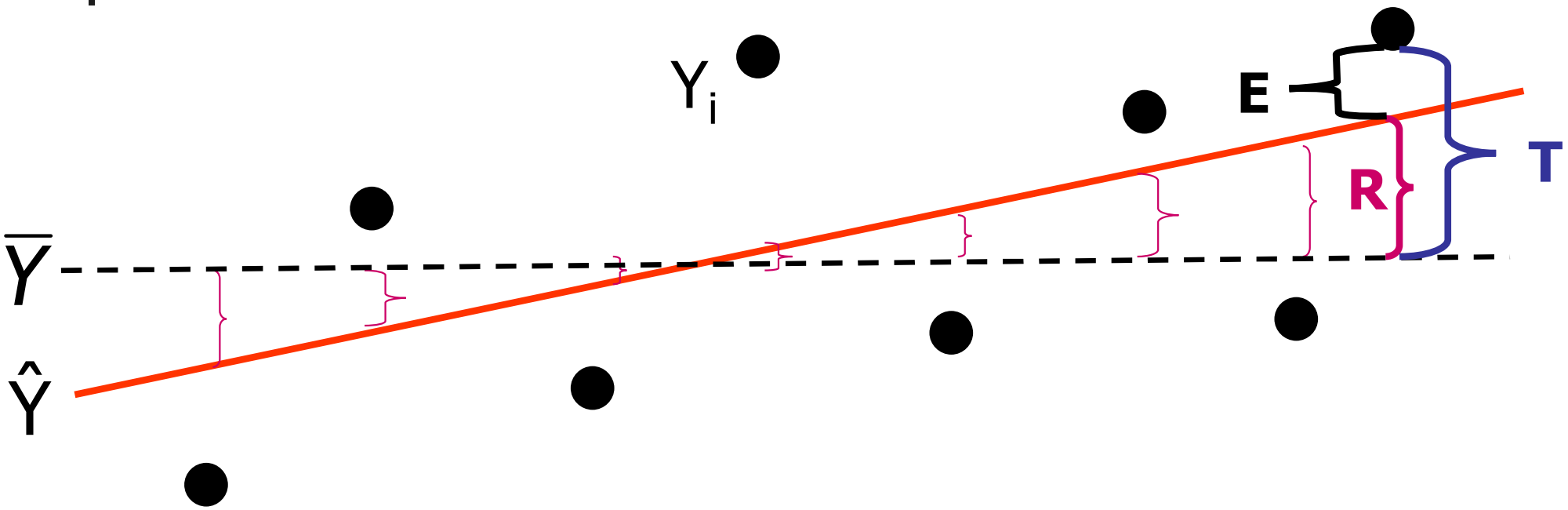
Partitioning Variation: SS(Residual)

Vertical distance between a data point and a line = $(Y_i - \hat{Y}_i)$



SS(Residual) = $\sum_{i=1}^{N} \left(Y_i - \hat{Y}_i \right)^2$

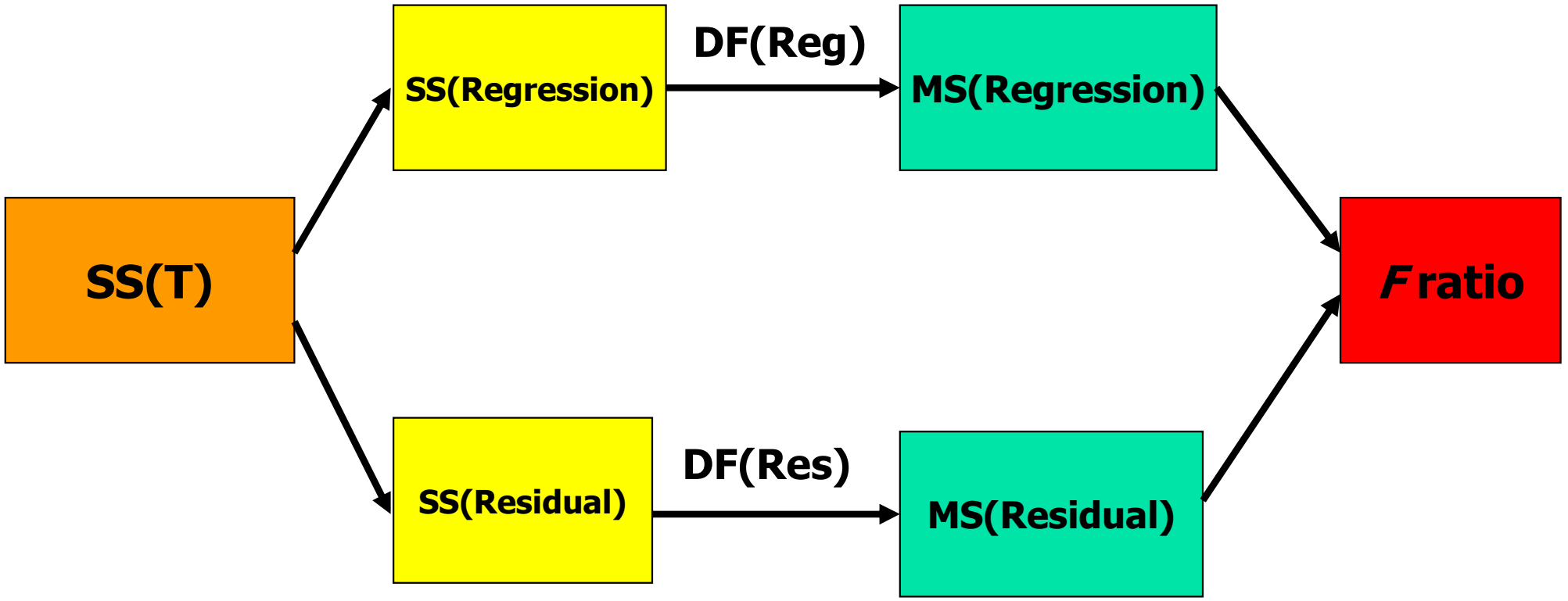
Partitioning Variation: SS(Regression)



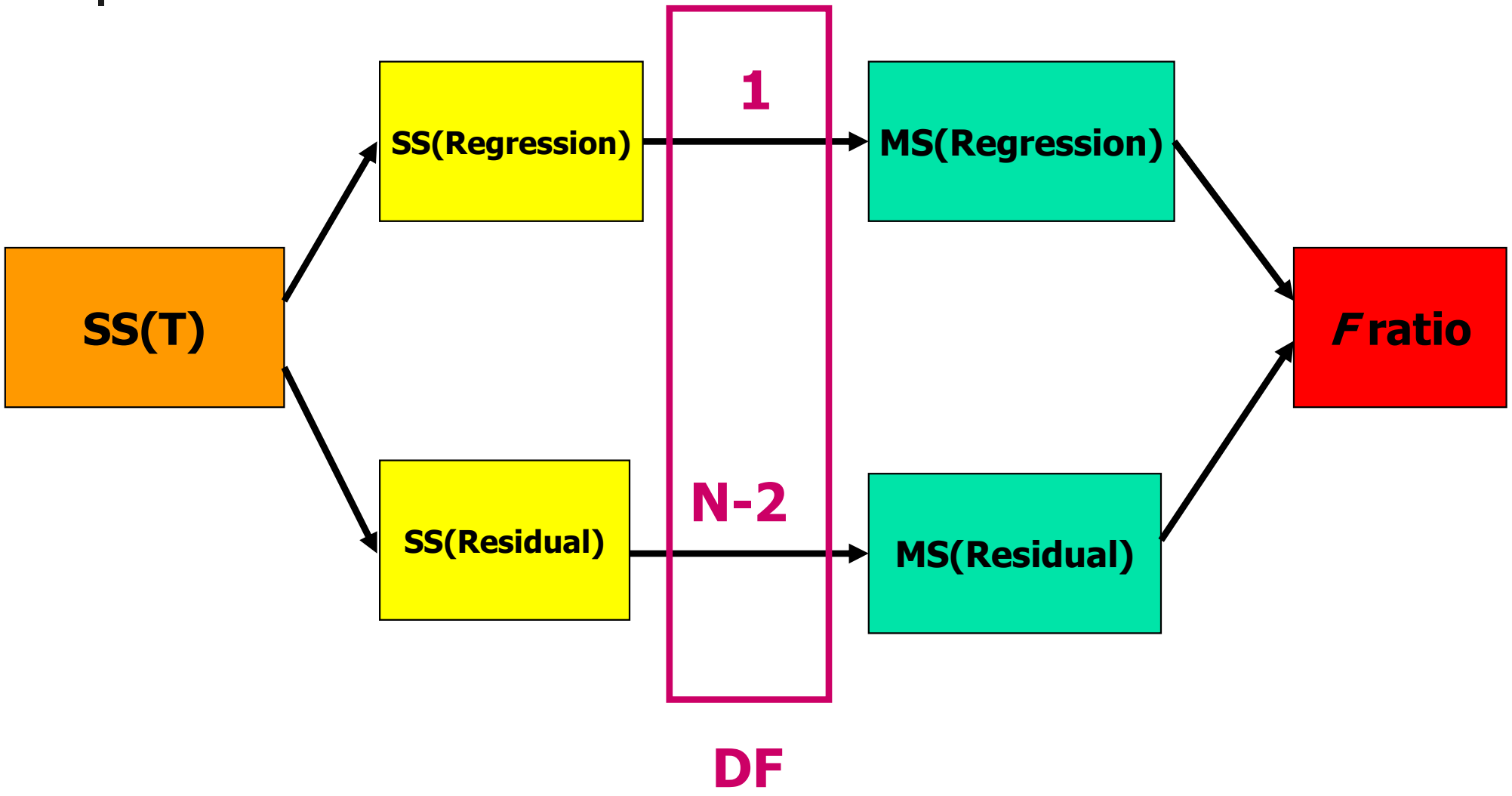
SS(Regression) = SS(T) – SS(Residual)

$$= \sum_{i=1}^{N} \left(\hat{Y}_i - \overline{Y} \right)^2$$

Simple Linear Regression – ANOVA Table



Simple Linear Regression – ANOVA Table



Simple Linear Regression

- The *F* statistic is used for testing
  - $H_0 : \beta_1 = 0$
  - $H_1 : \beta_1 \neq 0$
- If the observed *F* value is greater than a critical value of *F* with DF(Reg) and DF(Res) at $\alpha = .05$, we may reject H_0.
- This is the same as using the *t* test for β_1.
  - In fact, $t^2 = F$

Assessing the goodness-of-fit of the regression model: Coefficient of Determination (R^2)

- Proportion of the total variation in Y accounted for by the regression model.

$$R^2 = 1 - \frac{\text{SS(Residual)}}{\text{SS(Total)}}$$

- Ranges from 0 to 1.
  - The larger R^2, the more variance of Y explained
  - 0 = No explanation at all
  - 1 = Perfect explanation

- In simple linear regression, $r = \sqrt{R^2}$

Assessing the goodness-of-fit of the model: Residual Standard Error (RSE)

- The residual standard error (RSE) is an estimate of the standard deviation of e.

$$\mathrm{RSE} = \sqrt{\frac{1}{N-P-1}\mathrm{SS(Residual)}}$$

- Roughly speaking, it is the average amount that the response deviates from the true regression line.
  - RSE is considered a measure of the lack of fit of the model.

Assessing the goodness-of-fit of the model: Mean Squared Error (MSE)

- The mean squared error (MSE) is the mean of the sum of squared residuals, i.e., it measures the average of the squares of the errors.

$$\mathrm{MSE} = \frac{1}{N}\mathrm{SS}(\mathrm{Residual})$$

- Root mean squared error (RMSE) is the square root of MSE:

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

Example: Simple Linear Regression

- Wine consumption and heart attacks (Data from M. H. Criqui, reported in *New York Times*, Dec. 12., 1994) – wineheartattach.csv

TABLE 2.2 Wine consumption and heart attacks

Country	Alcohol from wine	Heart disease deaths	Country	Alcohol from wine	Heart disease deaths
Australia	2.5	211	Netherlands	1.8	167
Austria	3.9	167	New Zealand	1.9	266
Belgium	2.9	131	Norway	0.8	227
Canada	2.4	191	Spain	6.5	86
Denmark	2.9	220	Sweden	1.6	207
Finland	0.8	297	Switzerland	5.8	115
France	9.1	71	United Kingdom	1.3	285
Iceland	0.8	211	United States	1.2	199
Ireland	0.7	300	West Germany	2.7	172
Italy	7.9	107			

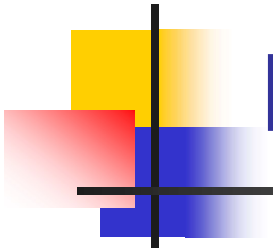
Moore & McCabe (1998, p.119)

Example: Simple Linear Regression

```
Call:
lm(formula = heartattack ~ wine, data = mydata)

Residuals:
   Min     1Q Median     3Q    Max
-62.95 -25.91 -12.35  26.97  55.52

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  260.563     13.835  18.833 7.97e-13 ***
wine         -22.969      3.557  -6.457 5.91e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 37.88 on 17 degrees of freedom
Multiple R-squared:  0.7103,  Adjusted R-squared:  0.6933
F-statistic: 41.69 on 1 and 17 DF,  p-value: 5.913e-06
```

Example: Simple Linear Regression

- This model summary table shows the goodness of fit of the fitted regression model/line. It indicates that about 71% of the total variance of the DV (heart disease deaths) is explained by the predictor (wine) (R^2 = .71).
- We may reject the null hypothesis that the population slope is zero because the p-value of the *t* statistic is less than .05 (t = -6.457, p < .00).
  - Note that in simple regression analysis, both F and t tests are used for testing the significance of the single slope, resulting in the same conclusion.

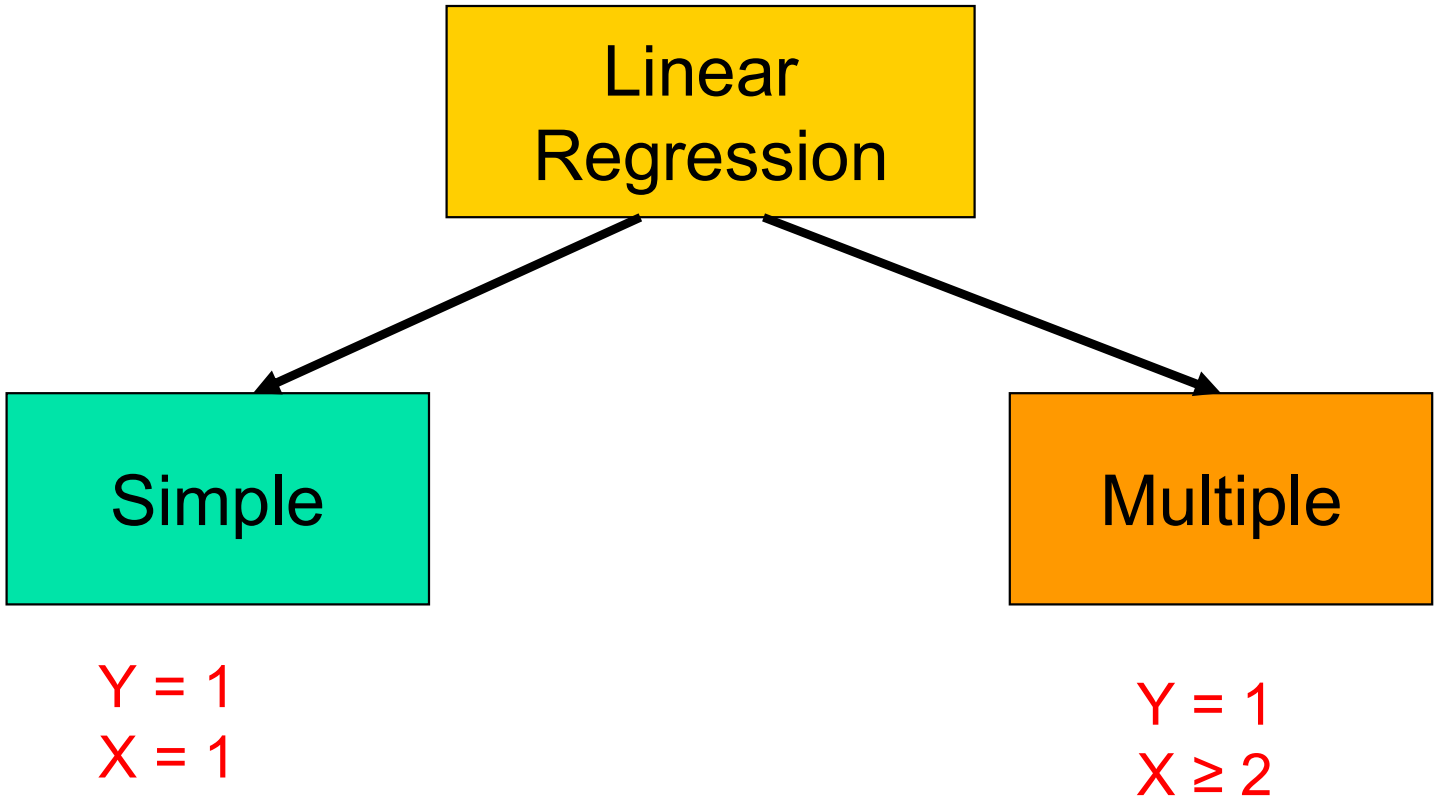


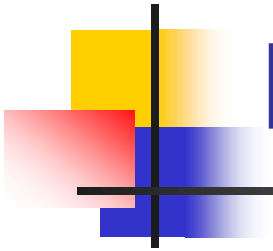
Example: Simple Linear Regression

- This indicates that the slope is statistically significantly different from zero, suggesting a statistically significant and negative effect of wine on heart disease deaths.

- More specifically, the estimated slope (wine = -22.97) indicates that as wine consumption increases by one unit, heart disease deaths decrease by 22.97 units on average.

Multiple Linear Regression

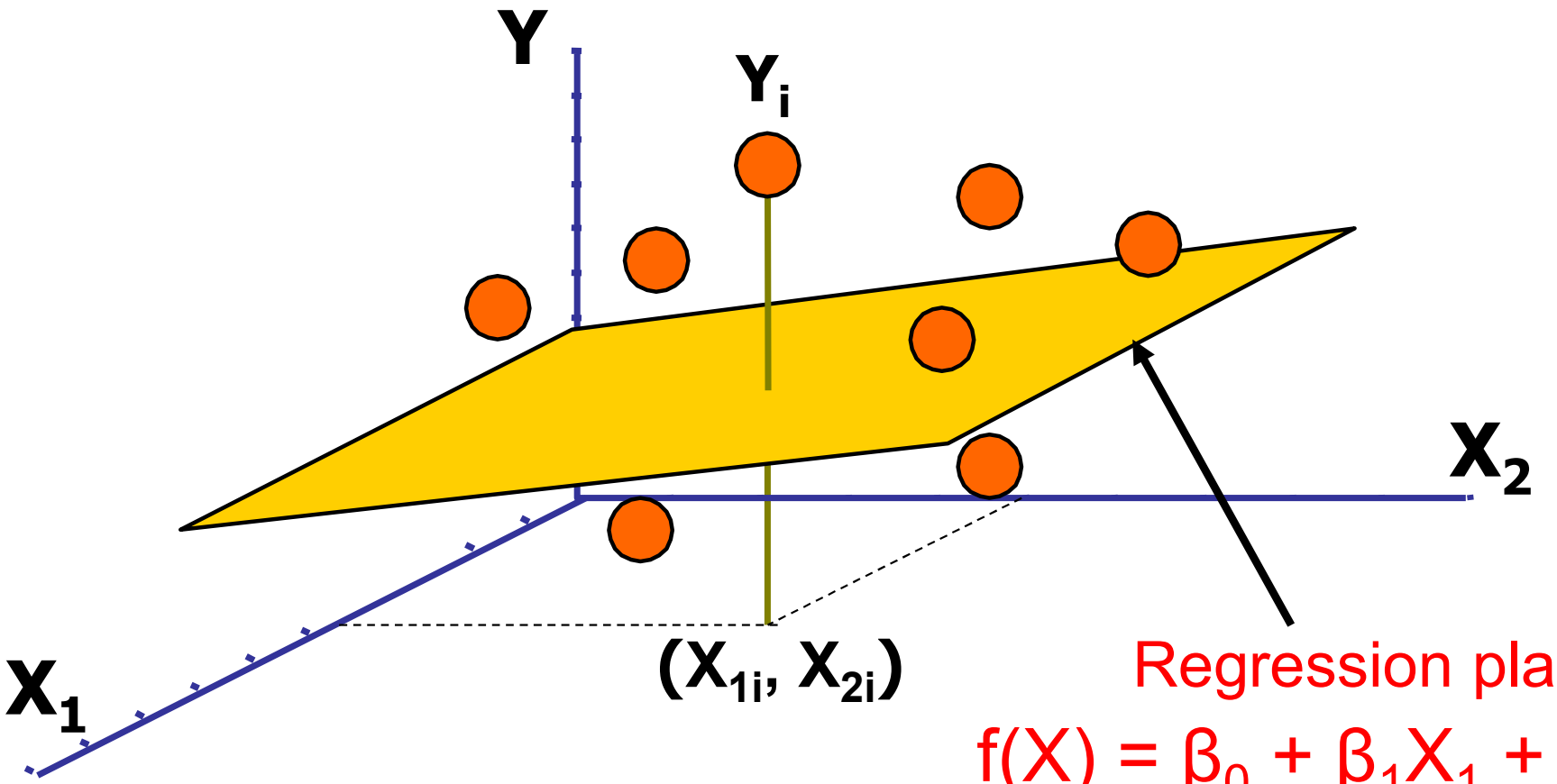




Multiple Linear Regression

- Describes how the DV (Y) changes as multiple predictors (X_P) ($P \geq 2$) change.

Multiple Linear Regression: P = 2



Multiple Linear Regression Model

- Linear relation between a continuous DV and P predictors ($P \geq 2$):

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_P X_P$$

- Predictors can be continuous or discrete.

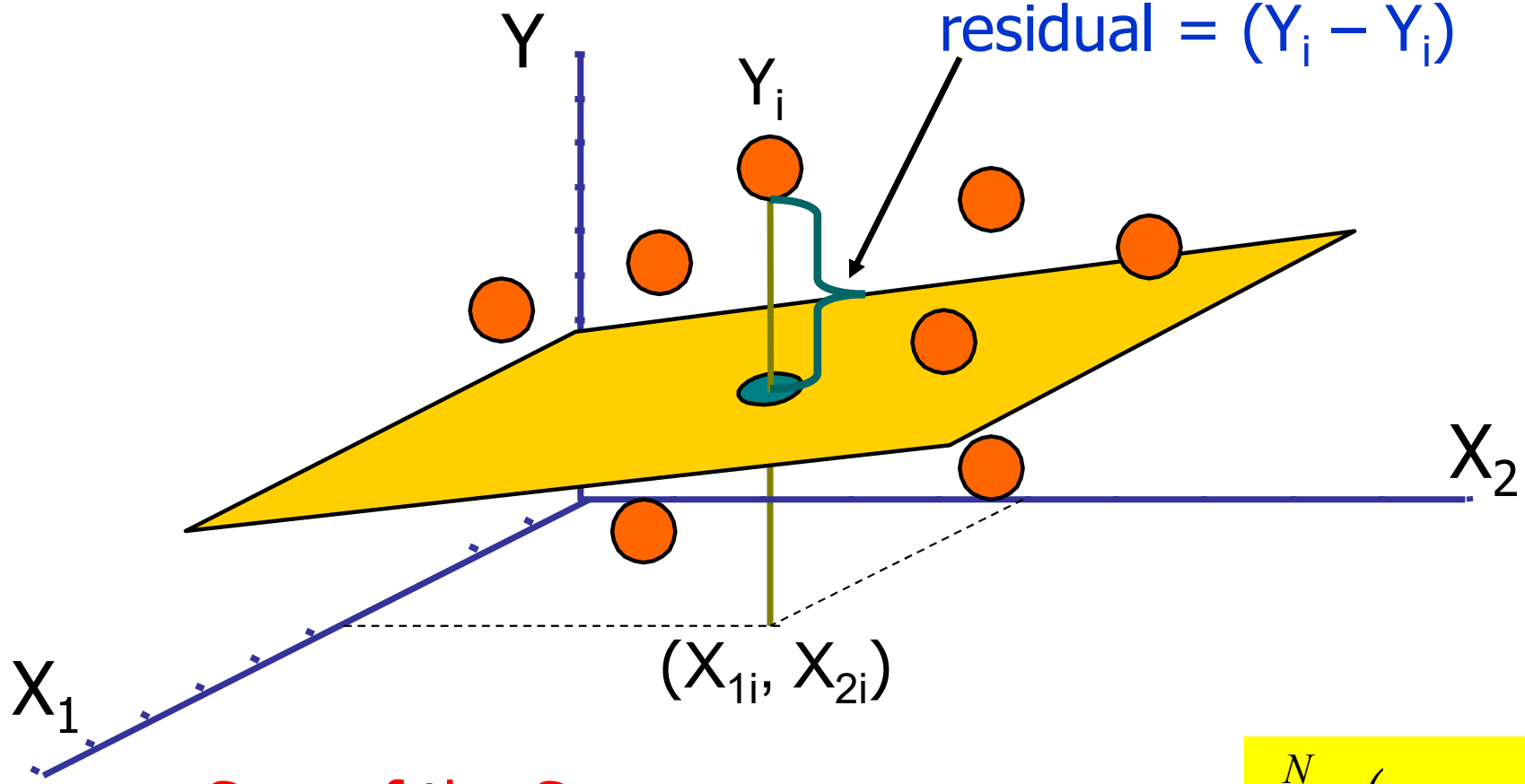
Multiple Linear Regression Model

- Intercept (β_0):
  - Average value of *Y* when all predictors are zero.
- Slope (β_p):
  - Amount by which Y changes on average when X_p changes by one unit, holding the other predictors remain constant (partialled out/controlled).
    - Example: If $\beta_1 = 2$, on average, Y is expected to increase by 2 for each 1 unit increase in X_1 with the other predictors constant.

How to estimate coefficients?

- As in simple linear regression, the method of least squares can be used for estimating the coefficients in the regression model.
  - This method chooses the values of the intercept and slopes that make the sum of the squared residuals as small as possible.

Multiple Linear Regression: Least Squares



Sum of the Squares of all residuals = SS(Error) = $\sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2$

Multiple Linear Regression: Least Squares

- In other words, the coefficients are estimated to minimize SS(Residual).

SS(Residual)

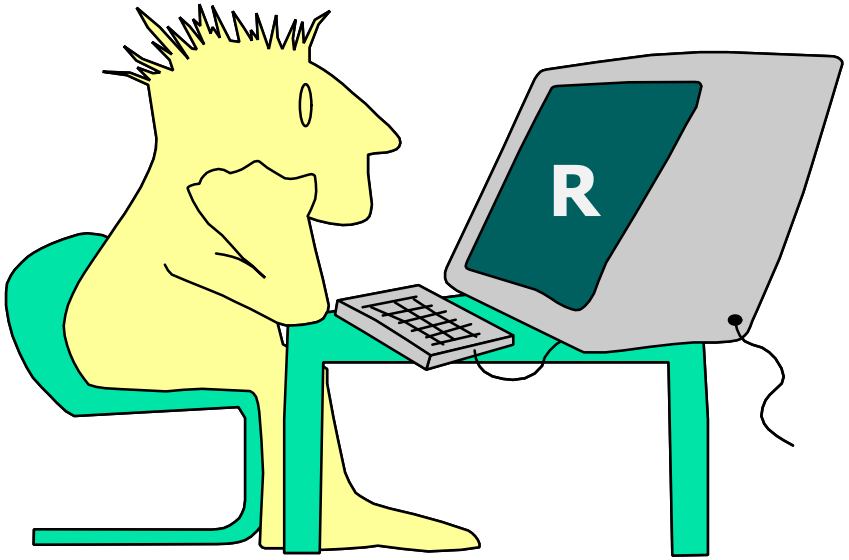
$$= \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{N} (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_1 - \cdots - \hat{\beta}_P X_P)^2$$

- The computations for obtaining the least squares estimates are complicated…

Multiple Linear Regression: Least Squares

Too complicated by hand!

Just let software do the computations!



Partitioning Variation in Multiple Linear Regression

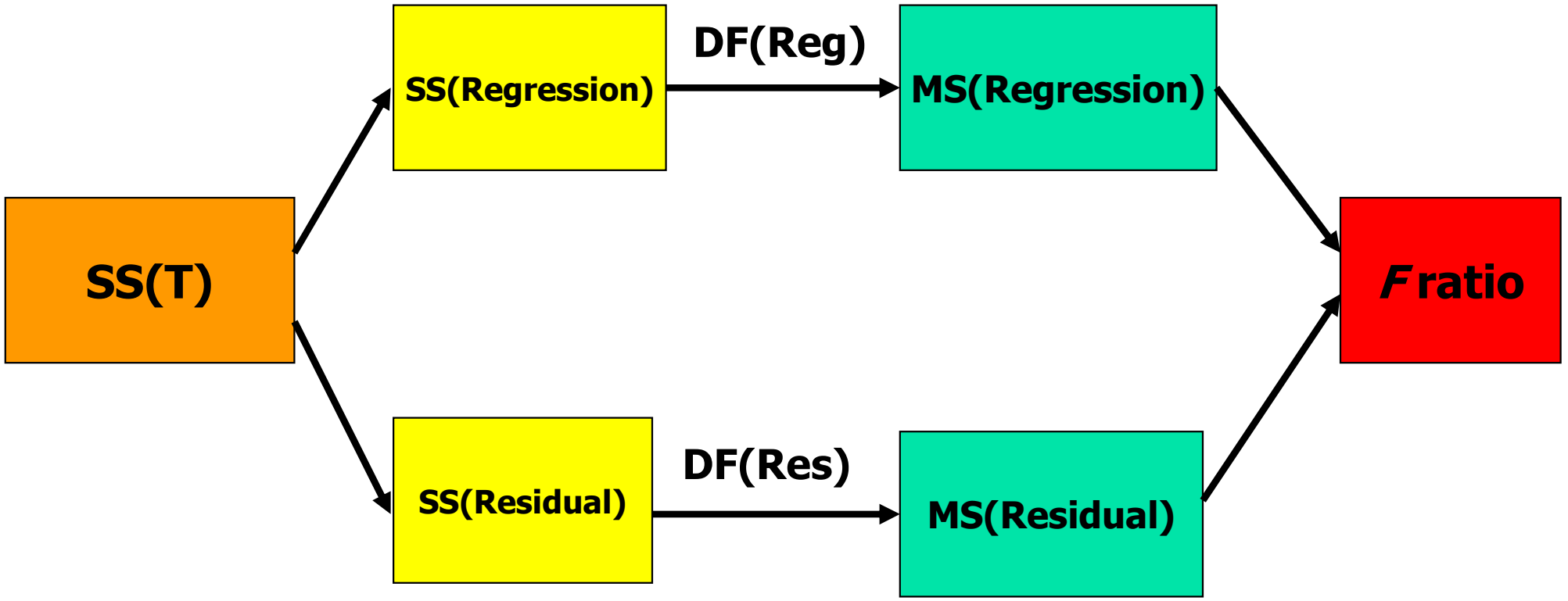
- As in simple regression analysis, the total variation in Y (SS(T)) is partitioned into:
  - SS(Regression): The variation in Y explained by the regression model.
  - SS(Residual): The variation in Y unexplained by the regression model.

$$SS(T) = \sum_{i=1}^{N} \left(Y_i - \bar{Y}\right)^2$$

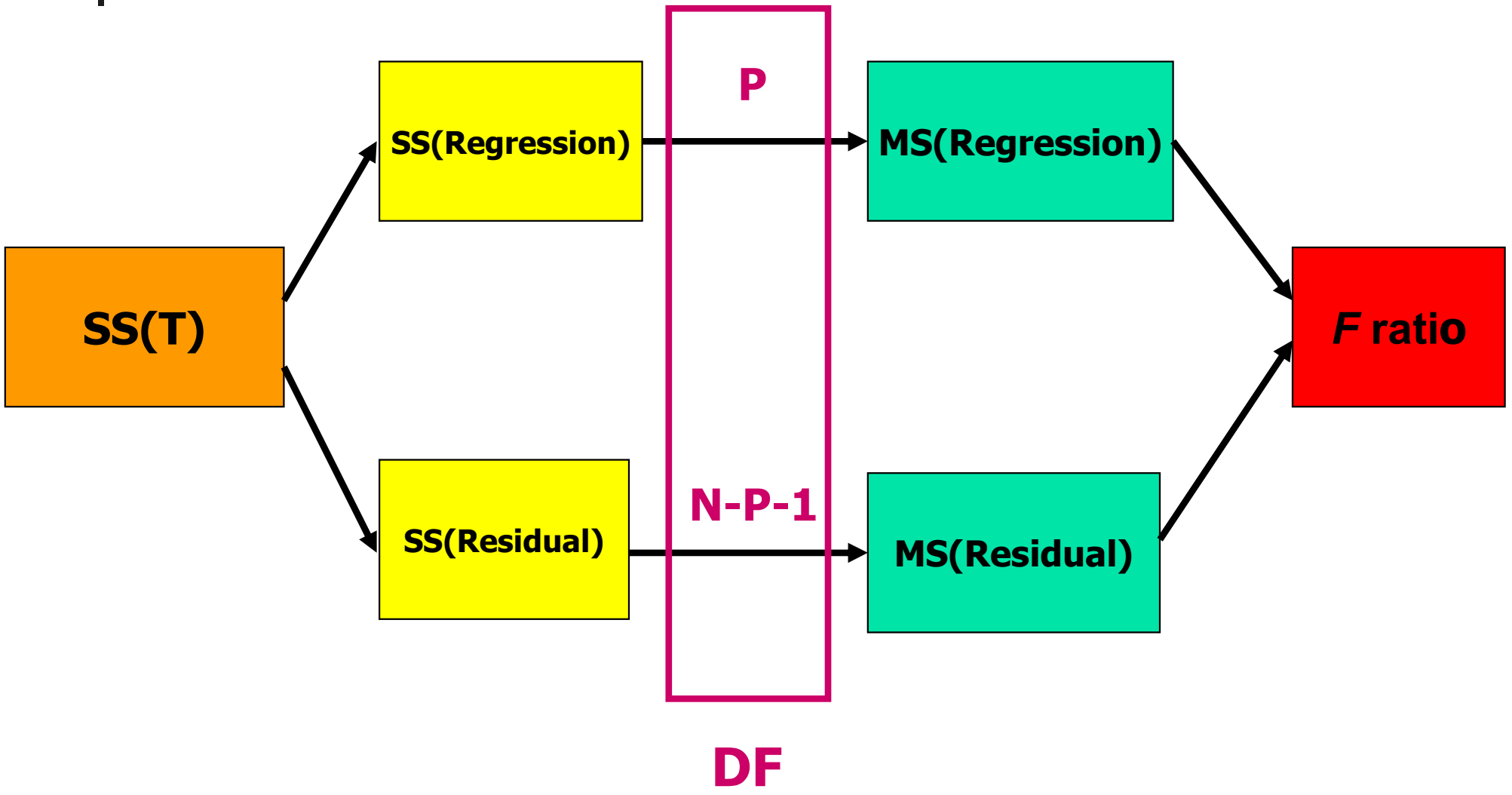
$$SS(Res) = \sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2$$

$$SS(Reg) = \sum_{i=1}^{N} \left(\hat{Y}_i - \bar{Y}\right)^2$$

Multiple Linear Regression – ANOVA Table



Multiple Linear Regression – ANOVA Table



Testing Overall Significance

- The *F* statistic is used to examine if there is a linear relationship between all *X* variables together and Y.
- Hypotheses
  - H_0: $\beta_1 = \beta_2 = \ldots = \beta_P = 0$
    - None of the Xs are linearly related to Y.
  - H_1: At least one coefficient is not 0
    - At least one X is linearly related to Y

Testing Overall Significance

- If the observed value of *F* is greater than the critical value of *F*(DF(Reg), DF(Res)) at α = .05, we may reject the null hypothesis.

- This indicates that at least one regression coefficient is statistically significantly different from zero.

Testing Individual coefficients

- $H_0 : \beta_p = 0$
  - X_p is not linearly related to Y (no linear relationship)
- $H_1 : \beta_p \neq 0$

- We can apply a t test for testing the significance of an individual coefficient.

Assessing the goodness-of-fit of the regression model: Coefficient of Determination (R^2)

- Proportion of the total variation in Y accounted for by the regression model.

$$R^2 = 1 - \frac{\text{SS(Residual)}}{\text{SS(Total)}}$$

- Also called the Squared Multiple Correlation (SMC).
  - Ranges from 0 to 1.
  - The larger R^2, the more variance of Y explained
  - 0 = No explanation at all
  - 1 = Perfect explanation

Assessing the goodness-of-fit of the model: Residual Standard Error (RSE)

- The residual standard error (RSE) is an estimate of the standard deviation of e.

$$\mathrm{RSE} = \sqrt{\frac{1}{N-P-1}\mathrm{SS(Residual)}}$$

- Roughly speaking, it is the average amount that the response deviates from the true regression line.
  - RSE is considered a measure of the lack of fit of the model.

Assessing the goodness-of-fit of the model: Mean Squared Error (MSE)

- The mean squared error (MSE) is the mean of the sum of squared residuals, i.e., it measures the average of the squares of the errors.

$$\mathrm{MSE} = \frac{1}{N}\mathrm{SS}(\mathrm{Residual})$$

- Root mean squared error (RMSE) is the square root of MSE:

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

Example: Multiple Linear Regression

- The children's antisocial behaviour data: Part of the National Longitudinal Survey of Youth (NLSY) reported in Curran (1998) (curran_training.csv).
  - In the NLSY, a large sample of children and their mothers were administered a set of assessment instruments every other year starting from 1986 to 1992.
  - From the original NLSY sample, Curran (1998) selected 221 pairs of children and mothers based on three selection criteria. First, children must have aged between 6 and 8 years at the first time point of assessment. Second, they had to complete interviews at all four time points. Finally, only one biological child was considered from each mother.
  - The average child's age was 6.9 years (SD = .62) and the average mother's age was 25.5 years (SD = 1.87) at the first time point.
  - We used 186 pairs as a training sample and 35 pairs as a test sample.

Example: Multiple Linear Regression

- DV = The antisocial behaviour of children measured at the first time point (0-12)

- Predictors (measured at the first time point):
  - Gender (female = 0 and male = 1)
  - Cognitive stimulation for children at home (0-14)
  - Emotional support for children at home (0-13)

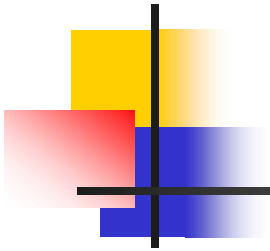
- N = 186

Example: Multiple Linear Regression

```
Call:
lm(formula = anti1 ~ gender + cogstm + emotsup, data = mydata)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5089 -1.1187 -0.3290  0.9261  5.4064

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.619633   0.560243   4.676 5.69e-06 ***
gender       0.646018   0.229772   2.812  0.00547 **
cogstm       0.008342   0.049073   0.170  0.86521
emotsup     -0.159691   0.053881  -2.964  0.00345 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.557 on 182 degrees of freedom
Multiple R-squared:  0.08062, Adjusted R-squared:  0.06546
F-statistic:  5.32 on 3 and 182 DF,  p-value: 0.001549
```

Example: Multiple Linear Regression

```
Call:
lm(formula = anti1 ~ gender + emotsup, data = mydata)

Residuals:
    Min      1Q  Median      3Q     Max
-2.4247 -1.0625 -0.3547  0.9149  5.4992

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.52342    0.42880   5.885 1.49e-08 ***
gender       0.63172    0.20035   3.153  0.00184 **
emotsup     -0.14610    0.04424  -3.302  0.00112 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.482 on 218 degrees of freedom
Multiple R-squared:  0.08113, Adjusted R-squared:  0.0727
F-statistic: 9.624 on 2 and 218 DF,  p-value: 9.874e-05
```



Linear Regression

- Two main goals:
  - To investigate the relationship between a DV and multiple predictors.
  - To find the best prediction equation for a DV regardless of the meaning of predictors in the equation.

Linear Regression: Prediction

- Once we have fit the regression model, it is straightforward to use $\hat{Y}$ to predict the response Y based on a set of values for the predictors.

$$Y = \hat{Y} + e$$

Example - Linear Regression: Prediction

- The original antisocial behaviour data were divided into training (curran_training.csv, N = 186) and test (curran_test.csv, N = 35) samples.

- We applied the same linear regression model to the training data to estimate the regression coefficients.

- Then, we used the coefficient estimates to predict new observations of the DV in the test sample and calculated RSE, MSE, and RMSE for the test sample.

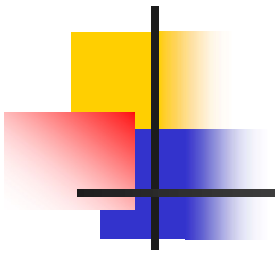
Example - Linear Regression: Prediction

```
Call:
lm(formula = anti1 ~ gender + cogstm + emotsup, data =
mydata_training)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5089 -1.1187 -0.3290  0.9261  5.4064

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.619633   0.560243   4.676 5.69e-06 ***
gender       0.646018   0.229772   2.812  0.00547 **
cogstm       0.008342   0.049073   0.170  0.86521
emotsup     -0.159691   0.053881  -2.964  0.00345 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.557 on 182 degrees of freedom
Multiple R-squared:  0.08062, Adjusted R-squared:  0.06546
F-statistic:  5.32 on 3 and 182 DF,  p-value: 0.0015494
```

Example - Linear Regression: Prediction

```
mydata_test$anti1
[1] 3 0 0 0 1 0 1 2 3 0 1 1 0 0 0 2 2 1 2 2 1 2 0 3 0 3 1 0 1 1 1 2 2 2 0

pred_y_test
        1         2         3         4         5         6         7         8
9        10        11
1.9035081 1.4338636 0.6437511 0.9631327 2.0559438 1.2491482 2.2228897 1.6091511
2.0965651 0.9047419 2.4922221
       12        13        14        15        16        17        18        19
20        21        22
1.7688419 1.2741728 1.2408066 0.7784172 1.7438172 2.0559438 0.7784172 1.1061404
1.5674433 1.4244356 1.7271342
       23        24        25        26        27        28        29        30
31        32        33
1.7521588 0.8034419 1.4244356 1.6008095 0.9381081 1.6008095 0.9130834 1.5757849
1.4077525 1.7605003 2.0298327
       34        35
0.9214250 1.92019

RSE
[1] 1.112484
MSE
[1] 1.096179
RMSE
[1] 1.046986
```

Linear Regression: Non-linear Relationship



- The linear assumption states that the changes in the response Y due to a one-unit change in a predictor is constant, regardless of the value of the predictor.

- We consider a very simple way of accommodating non-linear relationships, using polynomial regression.

Linear Regression: Non-linear Relationship

- To capture a quadratic shape, we may add X^2:

$$f(X) = \beta_0 + \beta_1 X + \beta_2 X^2$$

- Note that this is still a linear regression model with two predictors, so we can use standard linear regression software to estimate the coefficients $\beta_0, \beta_1, \text{and } \beta_2$.

- We can add higher-degree polynomials (e.g., X^3, X^4, X^5, etc.).

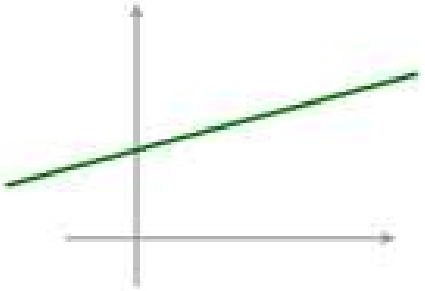
Polynomial Regression

1st degree polynomial

$y = a + bx^1$

straight line with no peaks and no valleys

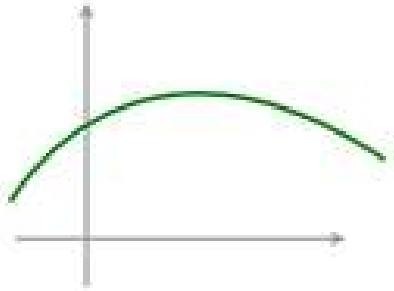
often written as $y = a + bx$



2nd degree polynomial

$y = a + bx^1 + cx^2$

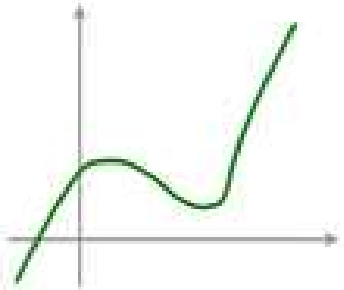
curved line with only one peak or one valley.



3rd degree polynomial

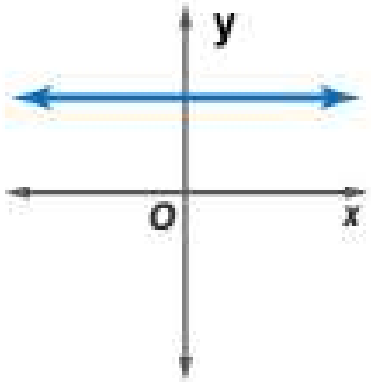
$y = a + bx^1 + cx^2 + dx^3$

curved line with multiple peaks & valley.

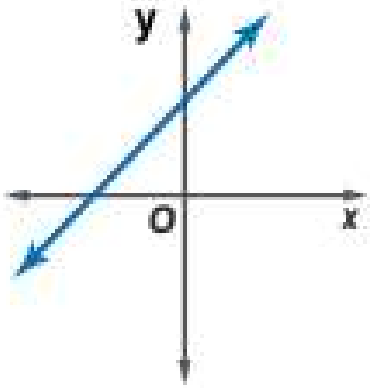


https://www.davidlongstreet.com/polynomials.html

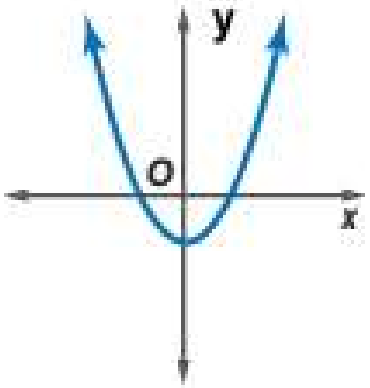
Constant function
Degree 0



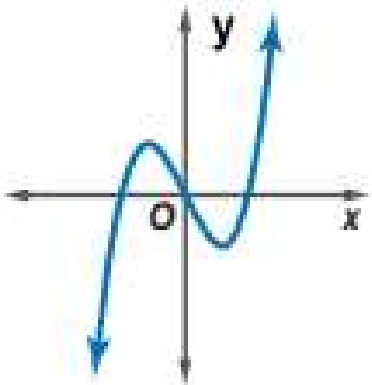
Linear function
Degree 1



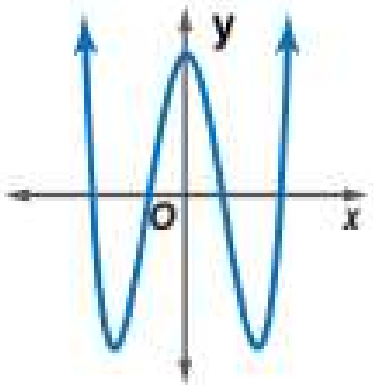
Quadratic function
Degree 2



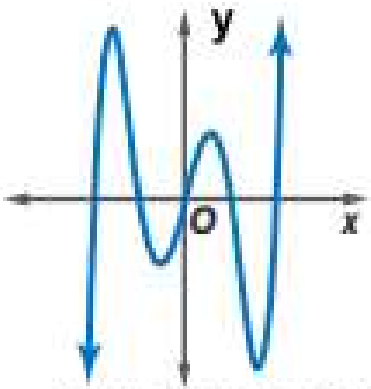
Cubic function
Degree 3



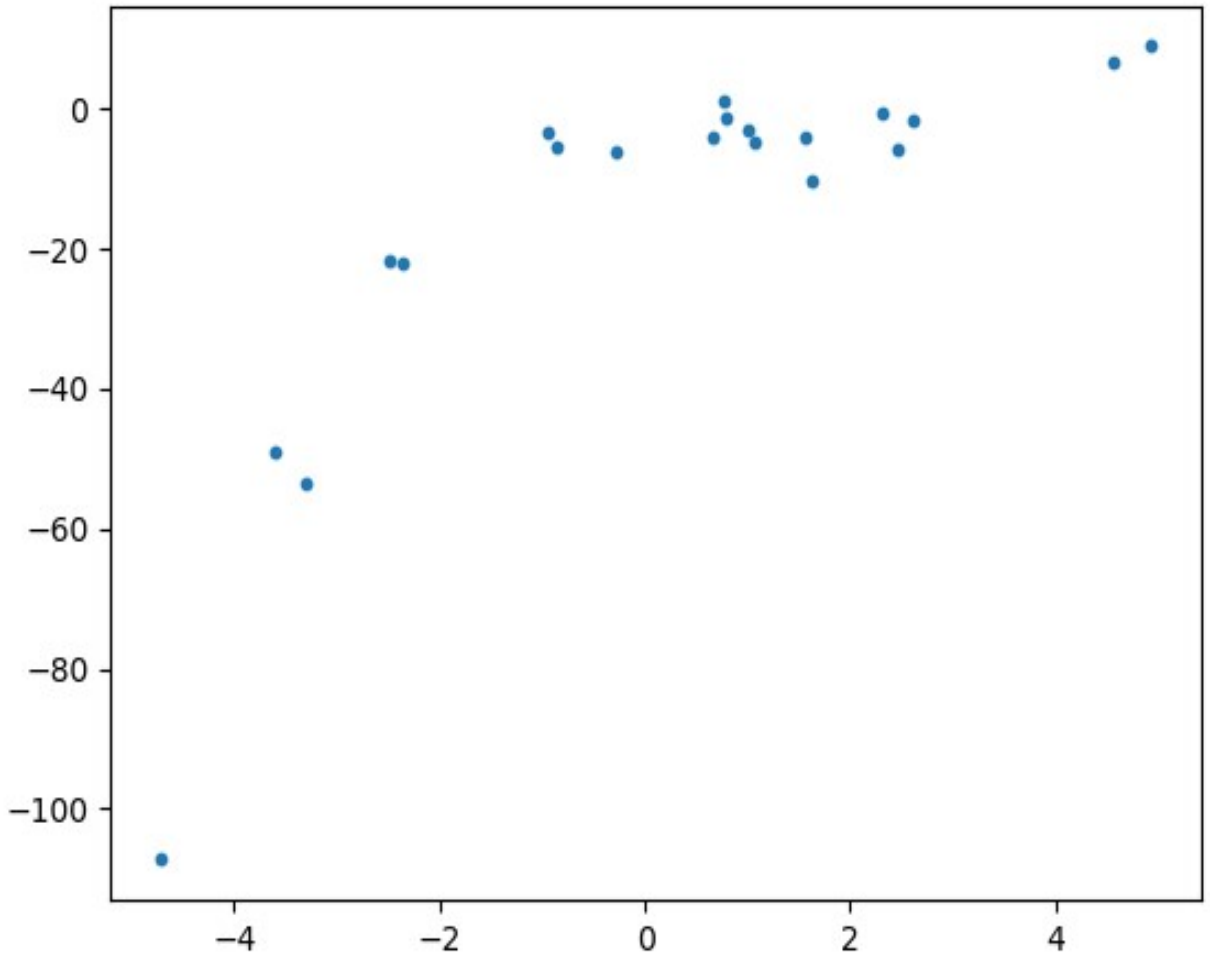
Quartic function
Degree 4



Quintic function
Degree 5

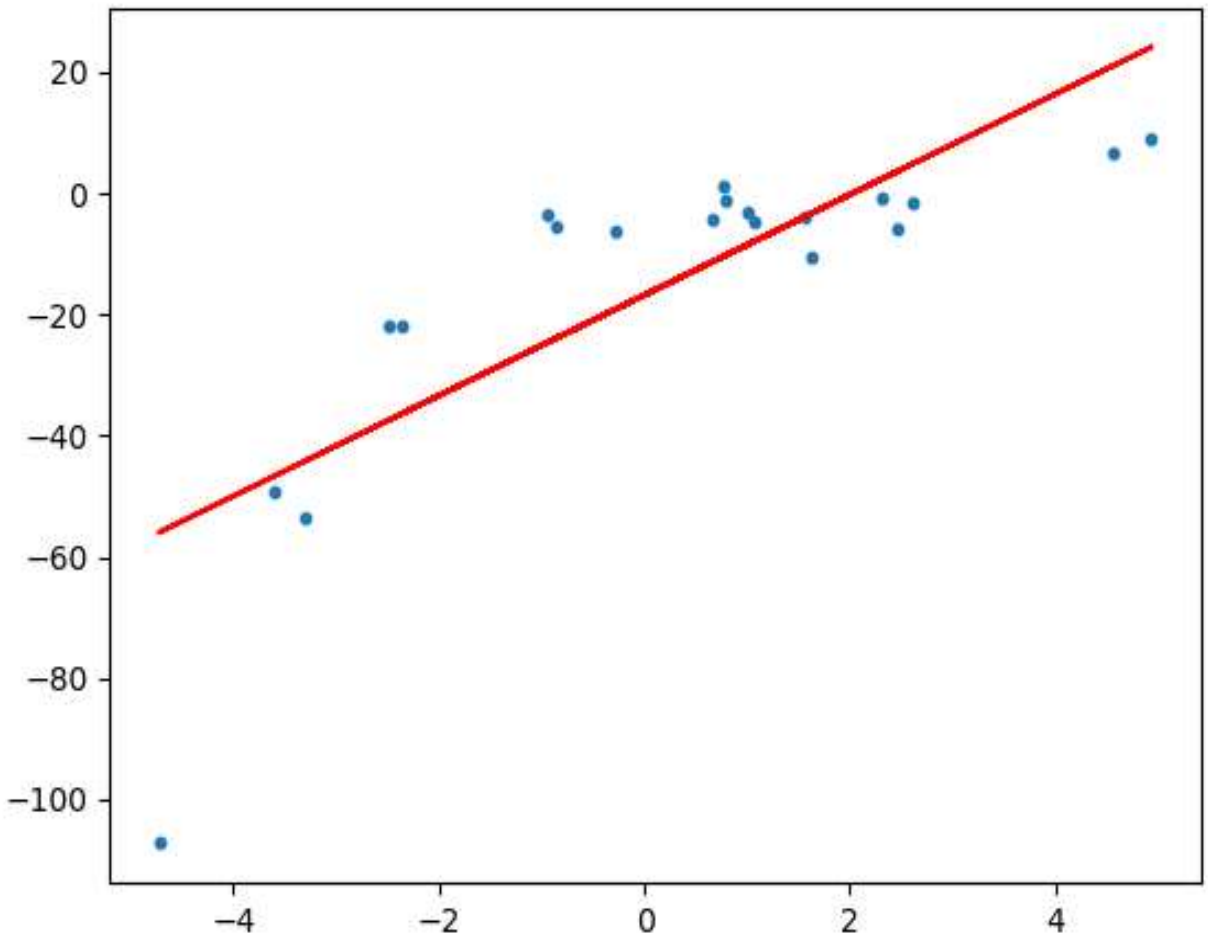


http://www.math.glencoe.com/



https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

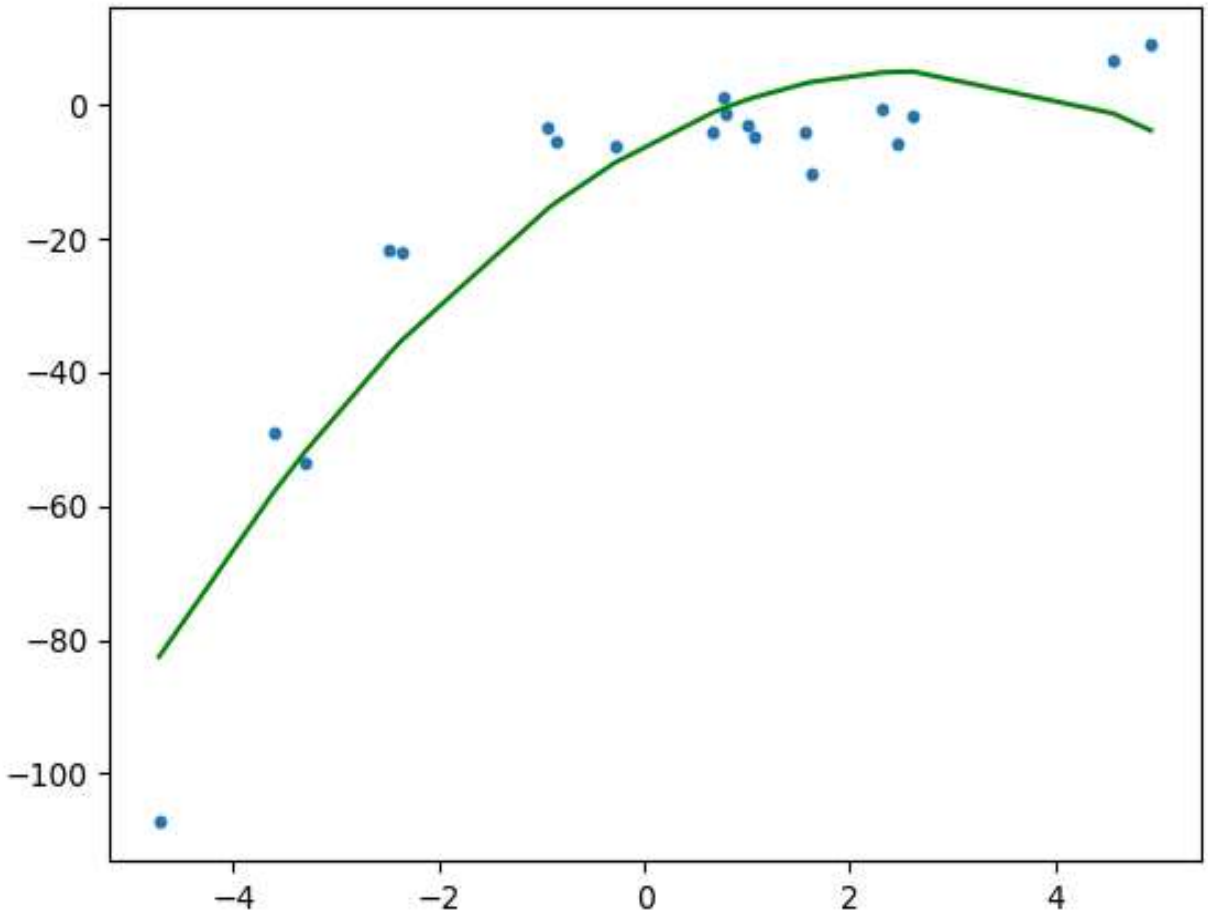
Linear regression



$R^2 = 0.639$

https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

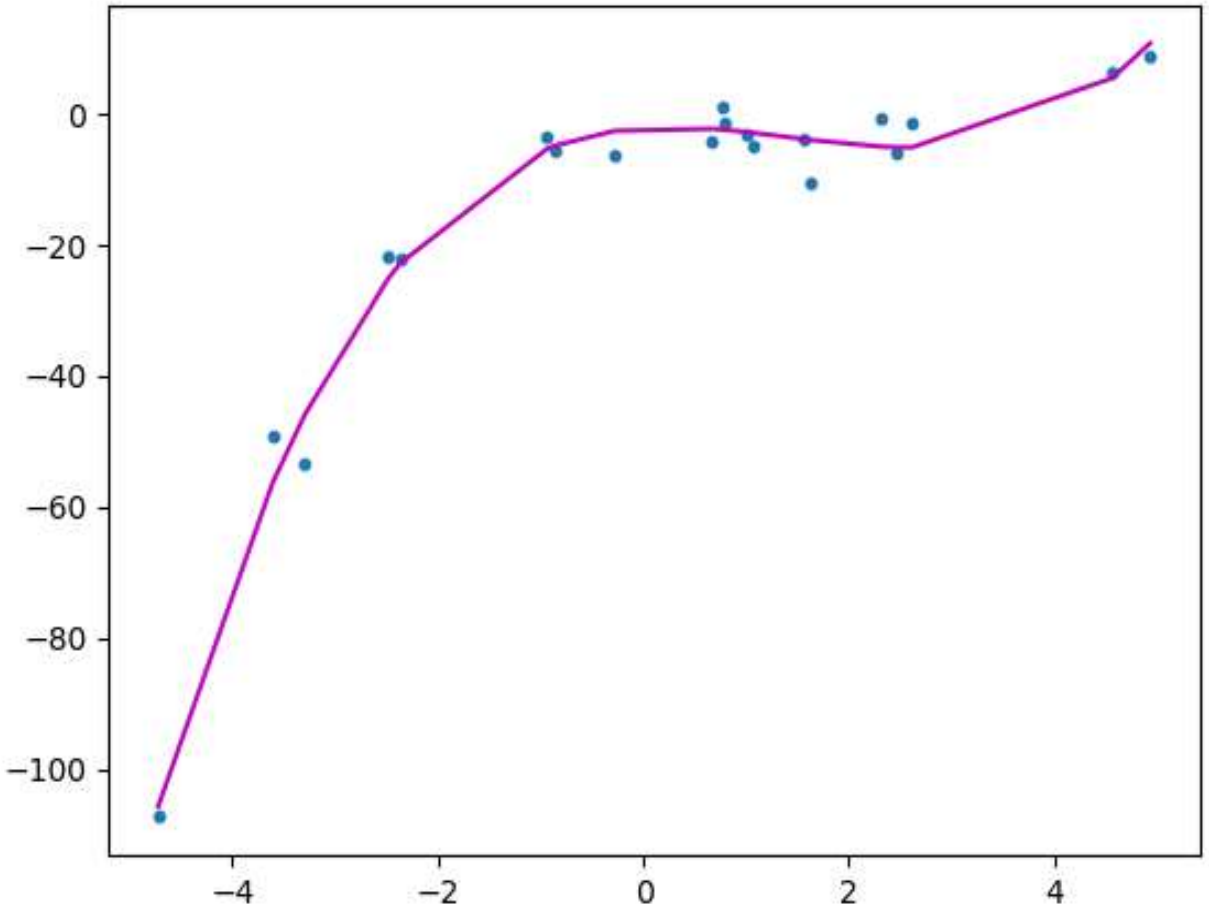
Polynomial regression (degree = 2)



$R^2 = 0.854$

https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

Polynomial regression (degree = 3)



$R^2 = 0.983$

https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

The Bias–Variance Tradeoff

- The bias of a statistical method/model refers to the error that is introduced by approximating a real-life problem, which may be extremely complicated, by much simpler models/methods.
  - For example, linear regression assumes that there is a linear relationship between Y and Xs, which is often unlikely in a real-life situation.

The Bias–Variance Tradeoff

- The variance of a statistical method/model is the amount by which $\hat{f}(X)$ would change if it is estimated using a different training set.

- Different training datasets will result in a different $\hat{f}(X)$. But ideally the estimate for *f(X)* should not vary too much between training sets. However, if a method has high variance, then small changes in the training data can result in large changes in $\hat{f}(X)$.

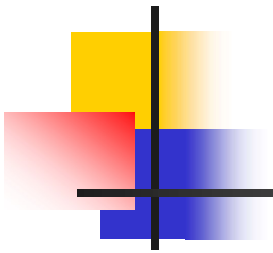
- In general, more flexible statistical methods/models have higher variance.

The Bias–Variance Tradeoff

- A statistical method's expected generalization error beyond the training sample is a sum of three terms, the bias, variance, and a quantity called the irreducible error, resulting from noise in the problem itself.

- That is, the expected test mean square error (MSE) is:

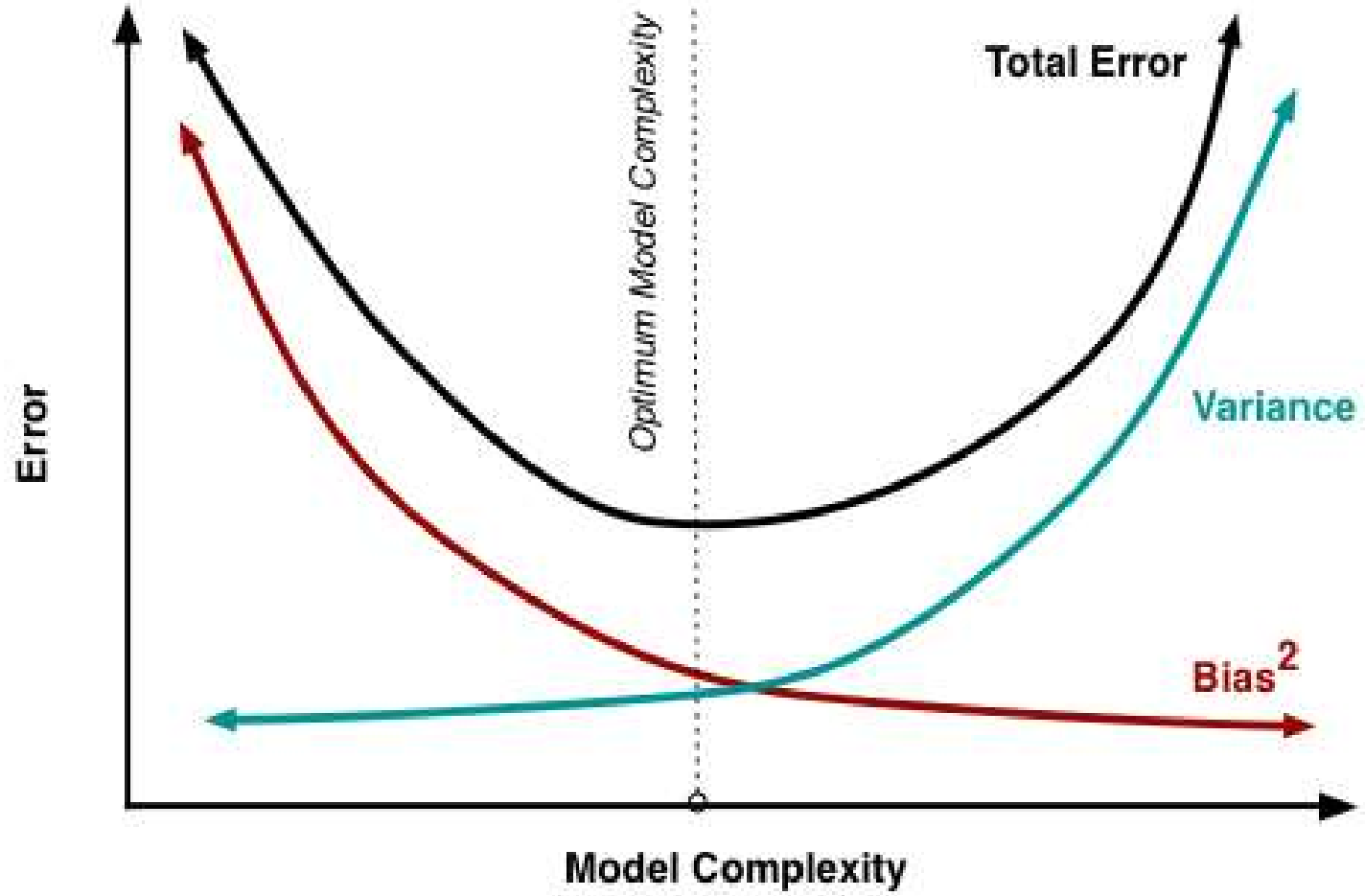
$$E(y_0 - \hat{f}(x_0))^2 = \mathrm{Var}\left(\hat{f}(x_0)\right) + [\mathrm{Bias}\left(\hat{f}(x_0)\right)]^2 + \mathrm{Var}(e)$$



The Bias–Variance Tradeoff

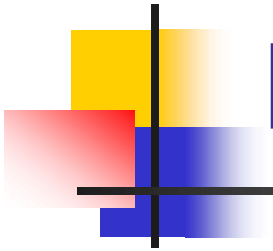
- This indicates that good test set performance of a statistical learning method requires low variance as well as low (squared) bias.
  - It is not easy to simultaneously minimize these two sources of error that prevent learning methods from generalizing beyond their training set.
- This is referred to as a trade-off because it is easy to obtain a method with extremely low bias but high variance or a method with very low variance but high bias.
- The challenge lies in finding a method for which both the variance and squared bias are low.

The Bias–Variance Tradeoff



http://scott.fortmann-roe.com/docs/BiasVariance.html

Generalizations of Linear Regression



- We will discuss methods that expand the scope of linear models:
  - Classification problems: logistic regression, support vector machines
  - Non-linearity: K-nearest neighbors regression
  - Interactions: Tree-based methods, bagging, random forests, boosting
  - Regularized methods: Ridge/lasso regression

K-Nearest Neighbors Regression

- K-nearest neighbors regression (KNN regression) is one of the simplest and best-known non-parametric (nonlinear) regression method.

- Given K and a prediction point x_0, KNN regression identifies the K training observations that are closest to x_0, represented by Q. It then estimates $f(x_0)$ using the average of all the training responses, as follows.

$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in Q} y_i$$

K-Nearest Neighbors Regression

Age X (month)	Height Y (cm)
18	76.1
19	77.0
20	78.1
21	78.2
22	78.8
23	79.7
24	79.9
25	81.1
26	81.2
27	81.8
28	82.8
29	83.5

Moore & McCabe (1998, p.136)

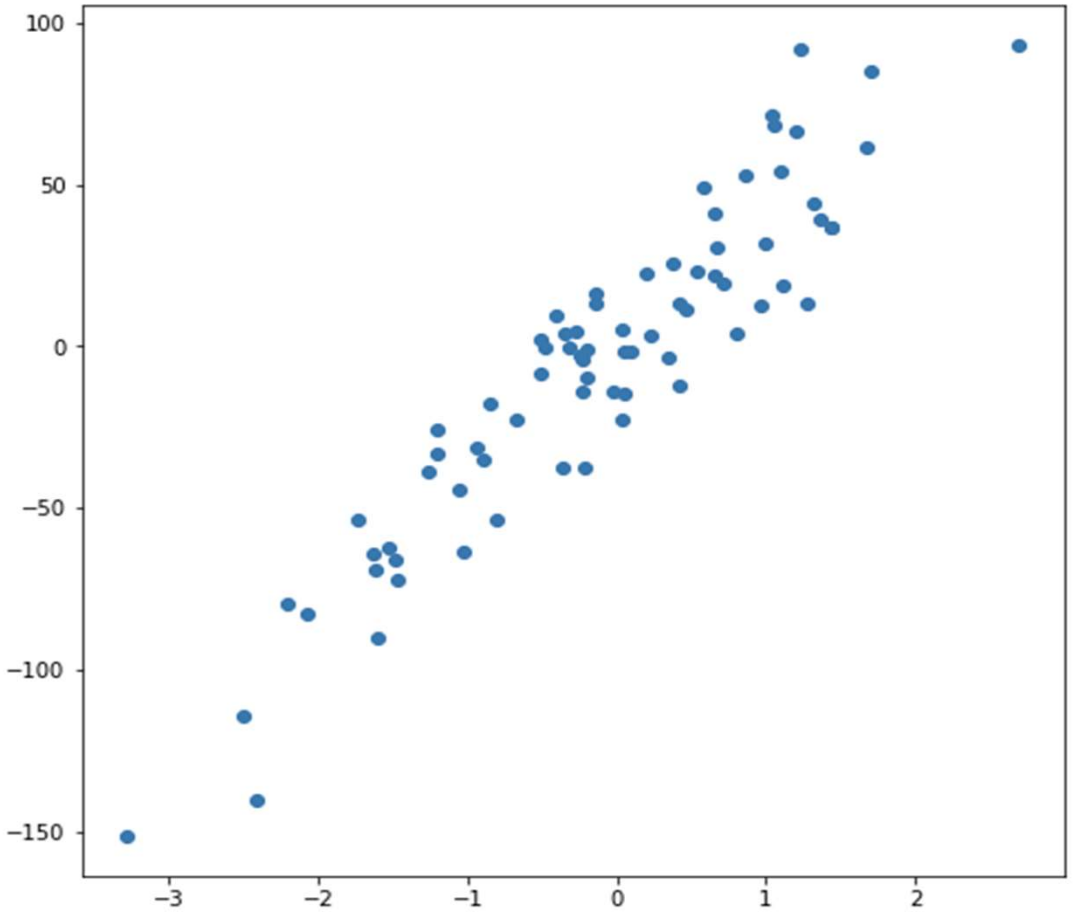
If age = 30 (X_0), then height?

K = 1, $\hat{Y}$ = 83.5

K = 2, $\hat{Y}$ = (83.5+82.8)/2

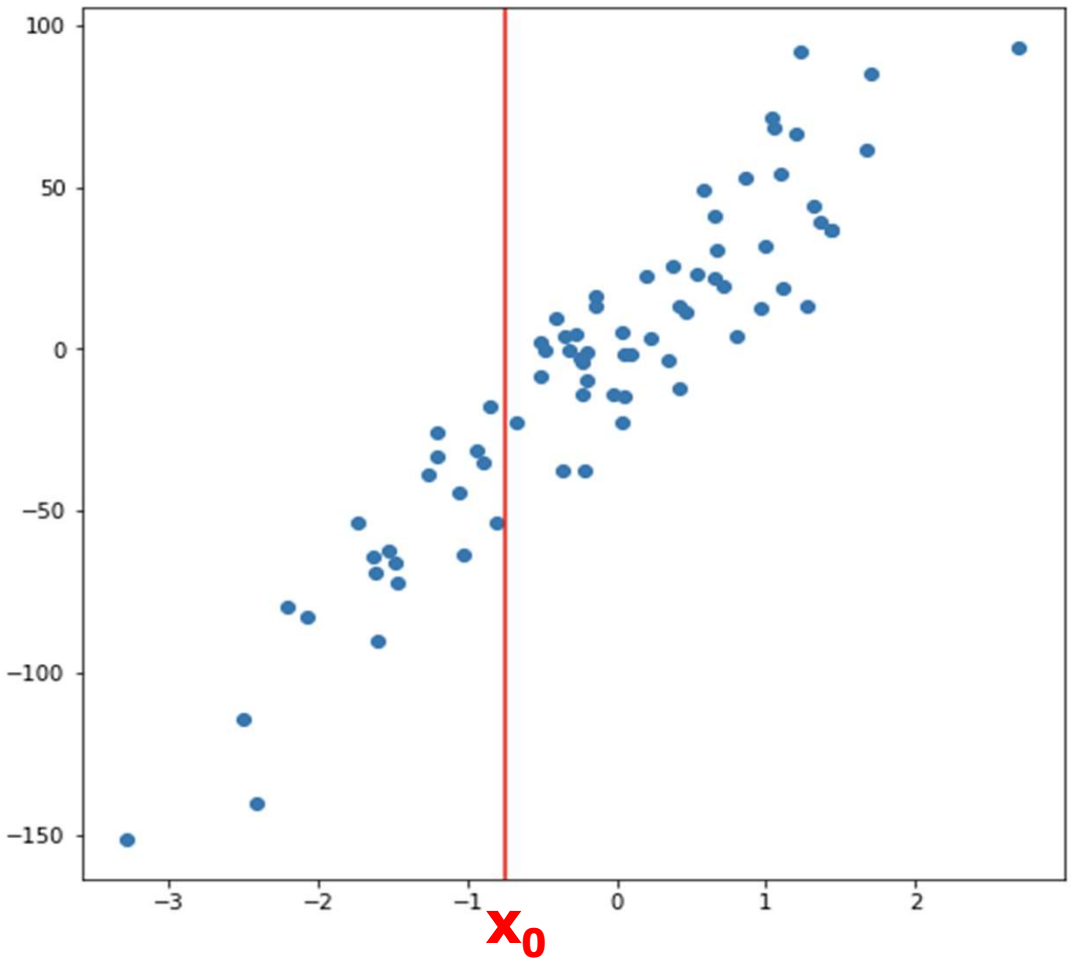
K = 3, $\hat{Y}$ = (83.5+82.8+81.8)/3

K-Nearest Neighbors Regression



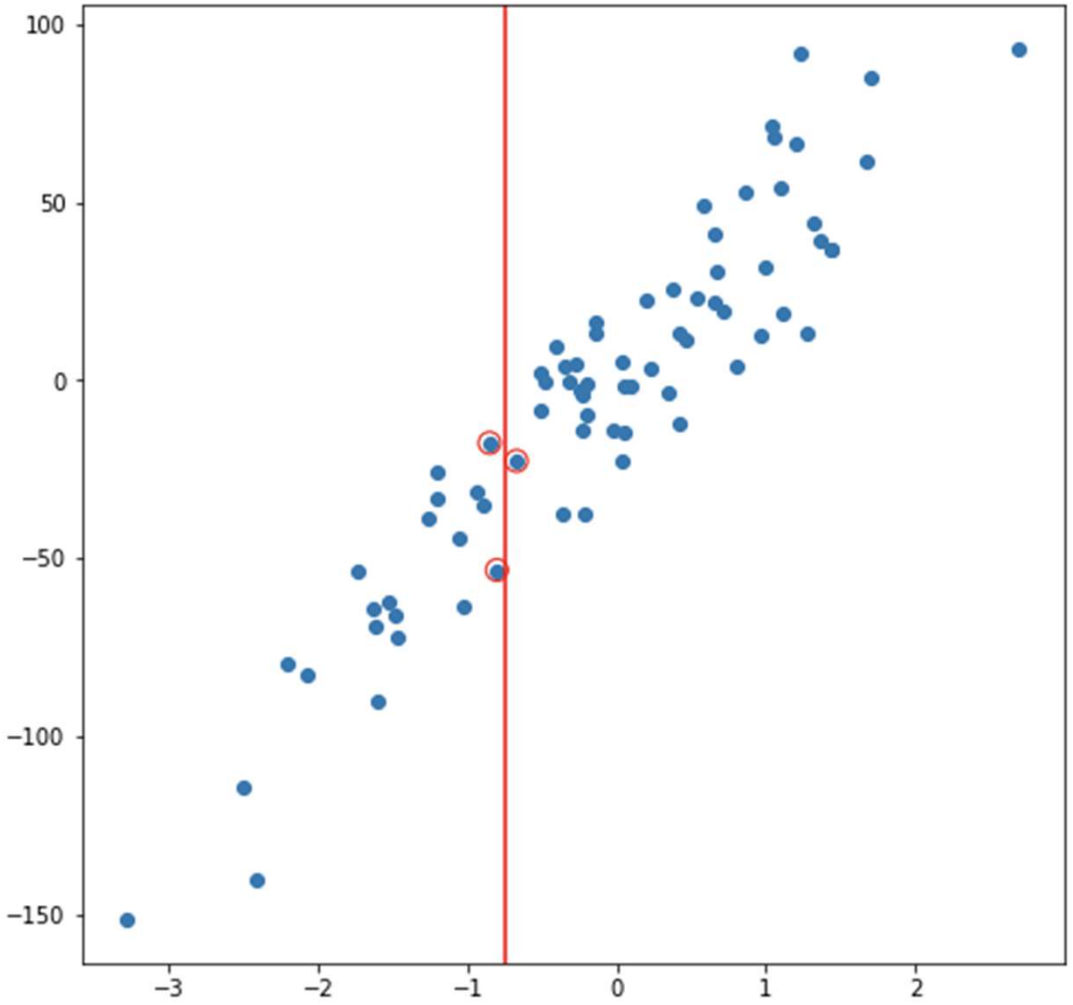
https://towardsdatascience.com/the-basics-knn-for-classification-and-regression-c1e8a6c955

K-Nearest Neighbors Regression

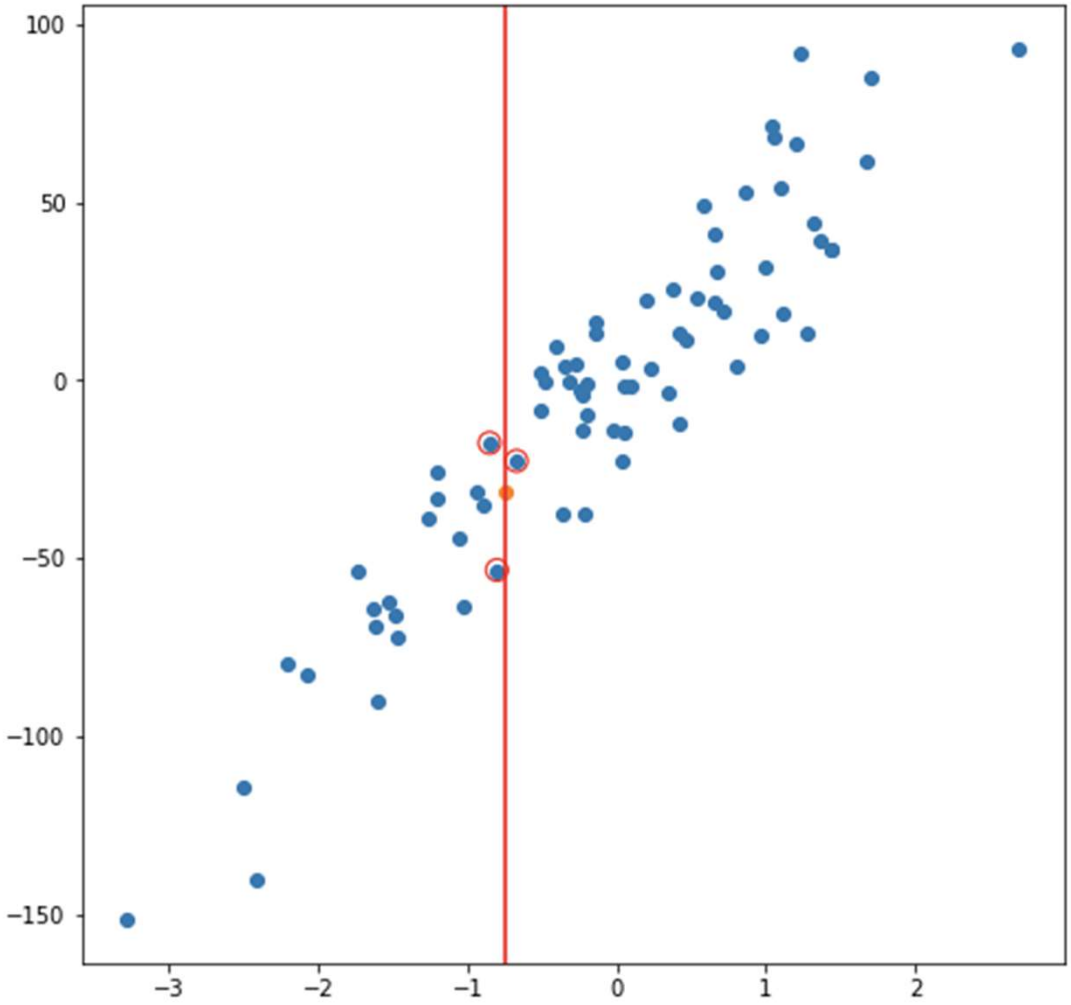


K-Nearest Neighbors Regression

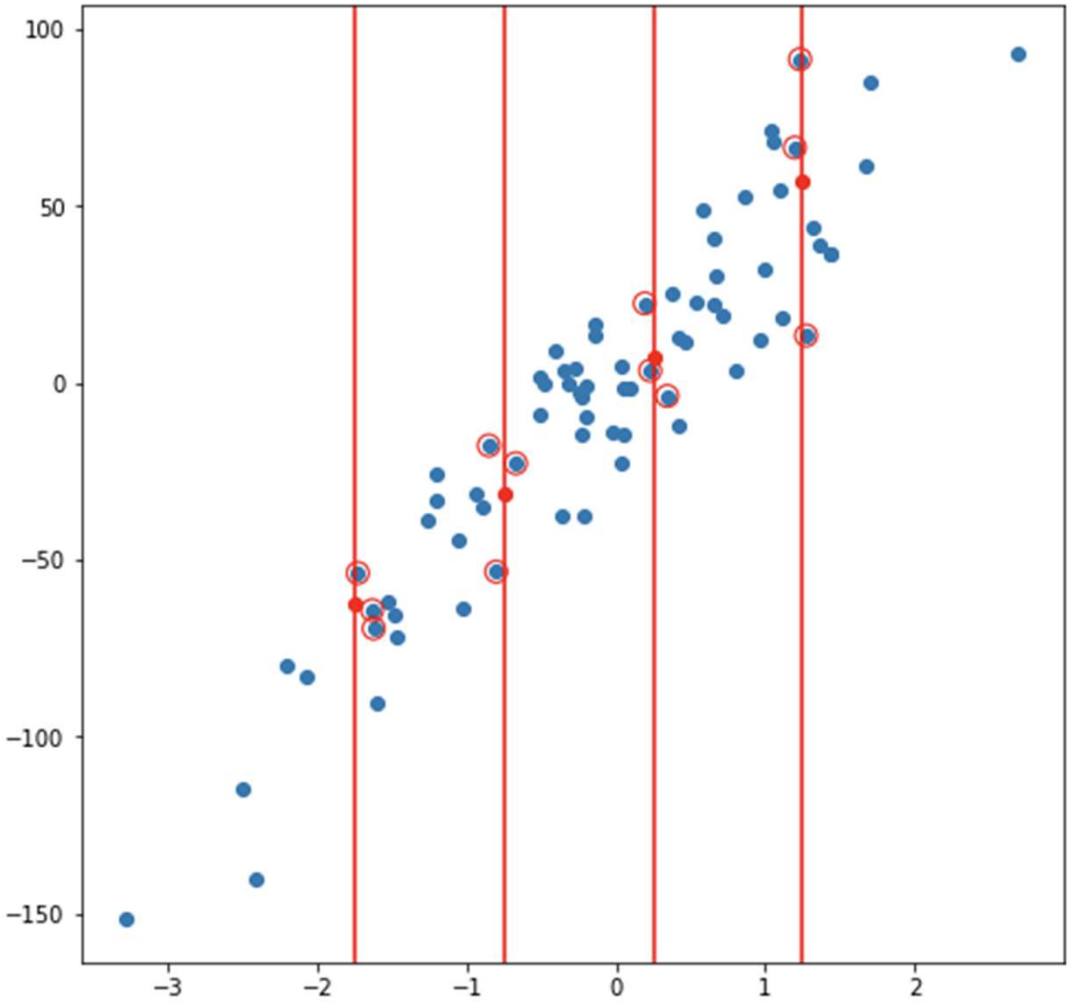
K = 3



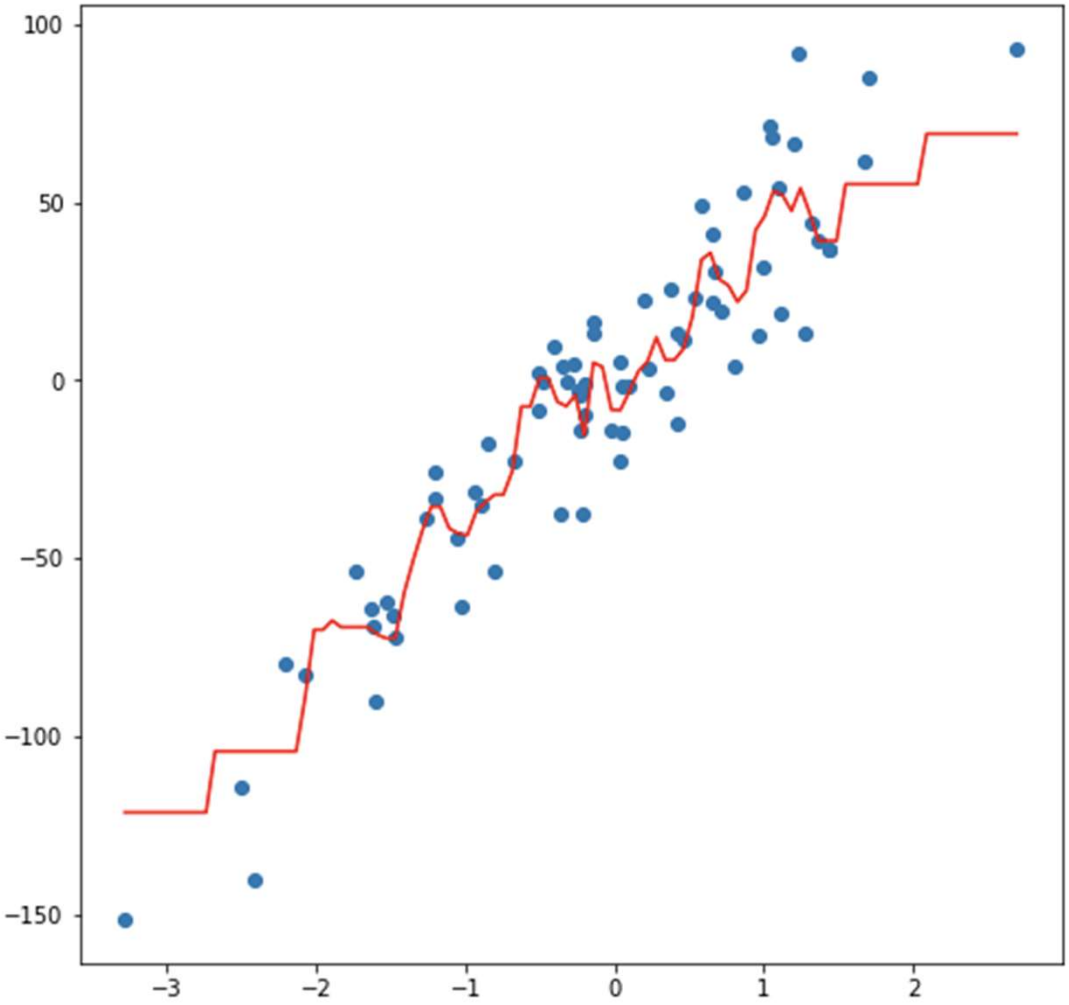
K-Nearest Neighbors Regression



K-Nearest Neighbors Regression



K-Nearest Neighbors Regression

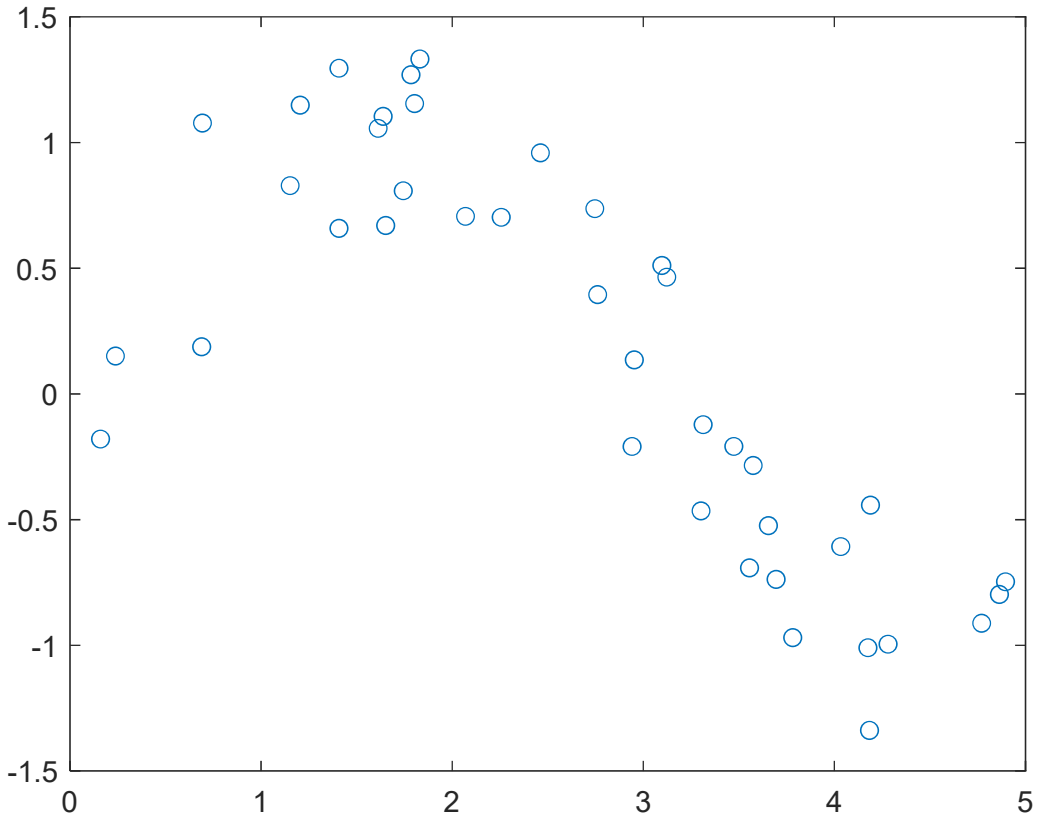


K-Nearest Neighbors Regression

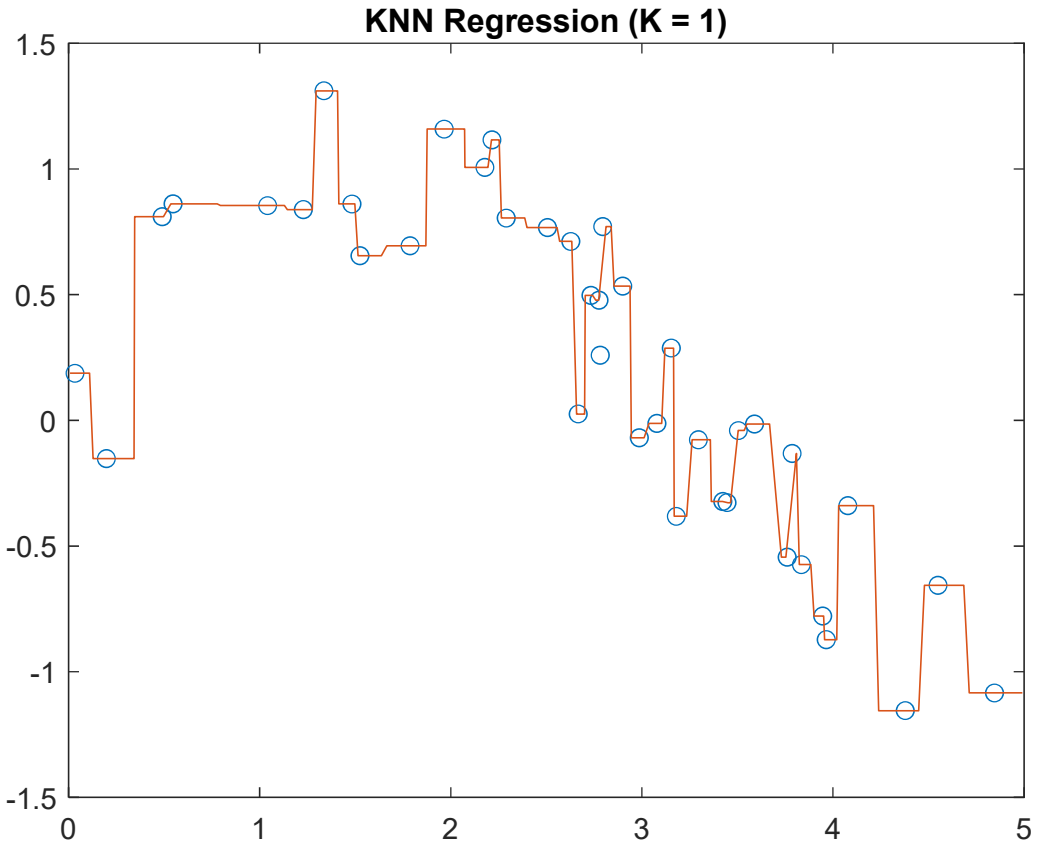
- K (the number of neighbors): When K = 1, KNN regression interpolates the training observations and takes the form of a step function. Increasing K will tend to give a smooth or less wiggly function. Yet, there is no single value of K that will work for every case. The optimal value for K will depend on the bias-variance tradeoff. Cross validation can be used for deciding K.

- Distance metric: There are different ways to measure how close two points are to each other, and the differences between these methods can become significant in higher dimensions. Most commonly used is Euclidean distance.
  - $d(x, x_0) = \sqrt{\sum_{p=1}^{P}(x_p - x_{0p})^2}$

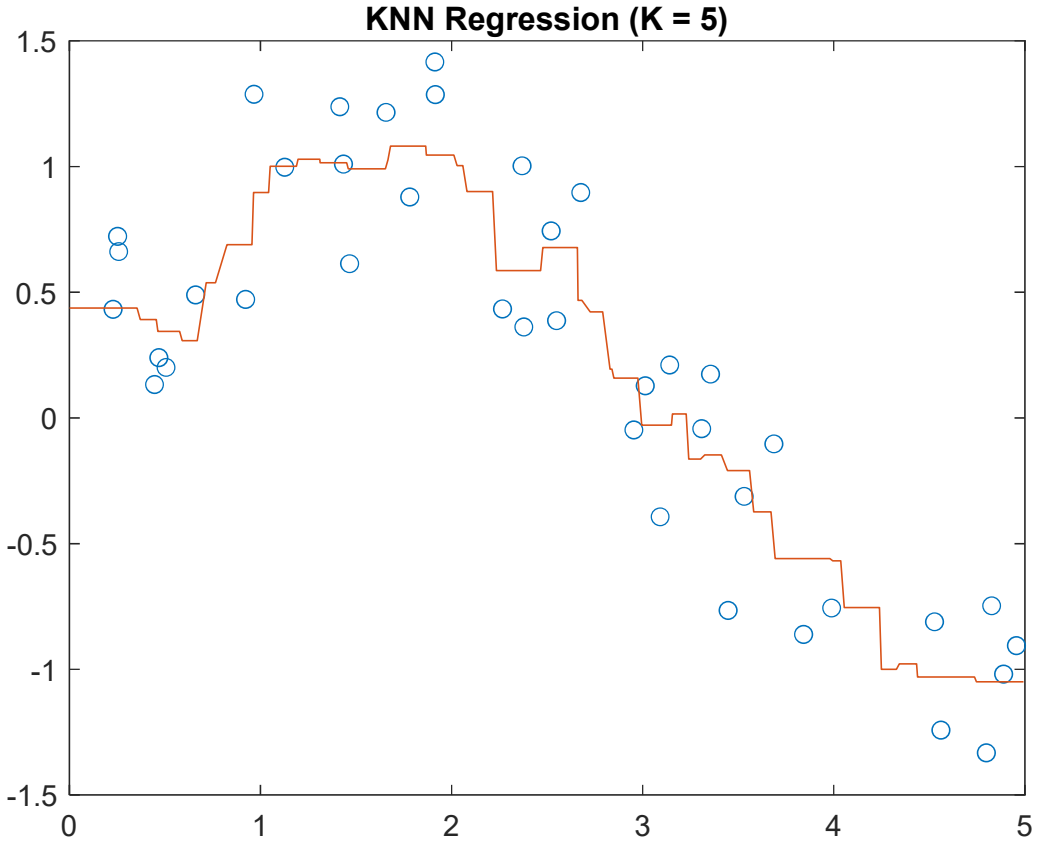
K-Nearest Neighbors Regression



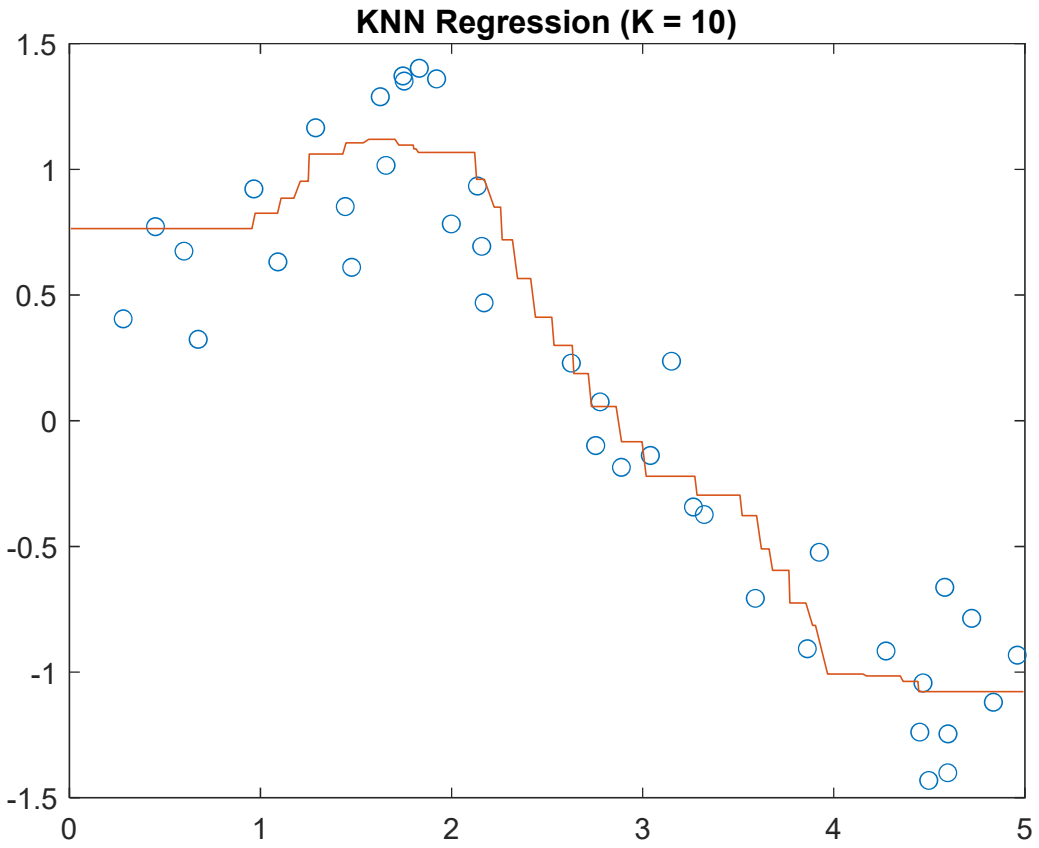
K-Nearest Neighbors Regression



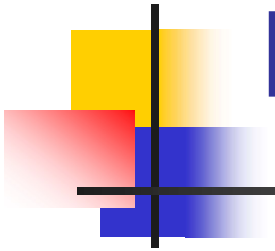
K-Nearest Neighbors Regression



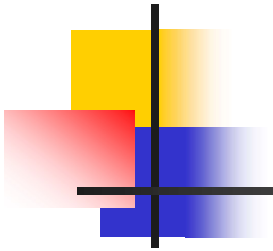
K-Nearest Neighbors Regression



K-Nearest Neighbors Regression



- KNN regression is easy to implement and will outperform linear regression when the true relationship between X and Y is substantially nonlinear.
- KNN regression can be chosen over linear regression when the true relationship between X and Y is unknown.
- But, in high dimensions (the number of predictors is large), KNN regression often performs worse than linear regression – the curse of dimensionality.
- Another issue with KNN regression is that it lacks interpretability.



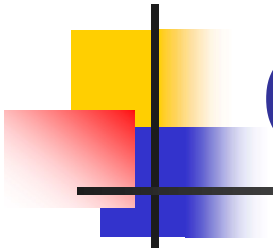
Lab: Linear Regression

1. Multiple Linear Regression – curran_training.csv
2. Polynomial Regression – curran_training.csv
3. KNN Regression - curran_training.csv
4. Performance of linear regression models and KNN regression for test data (in MSE) – curran_training.csv & curran_test.csv

Session 3
Classification Methods

An Introduction to Machine Learning for the Behavioural and Social Sciences

Heungsun Hwang & Gyeongcheol Cho



Classification

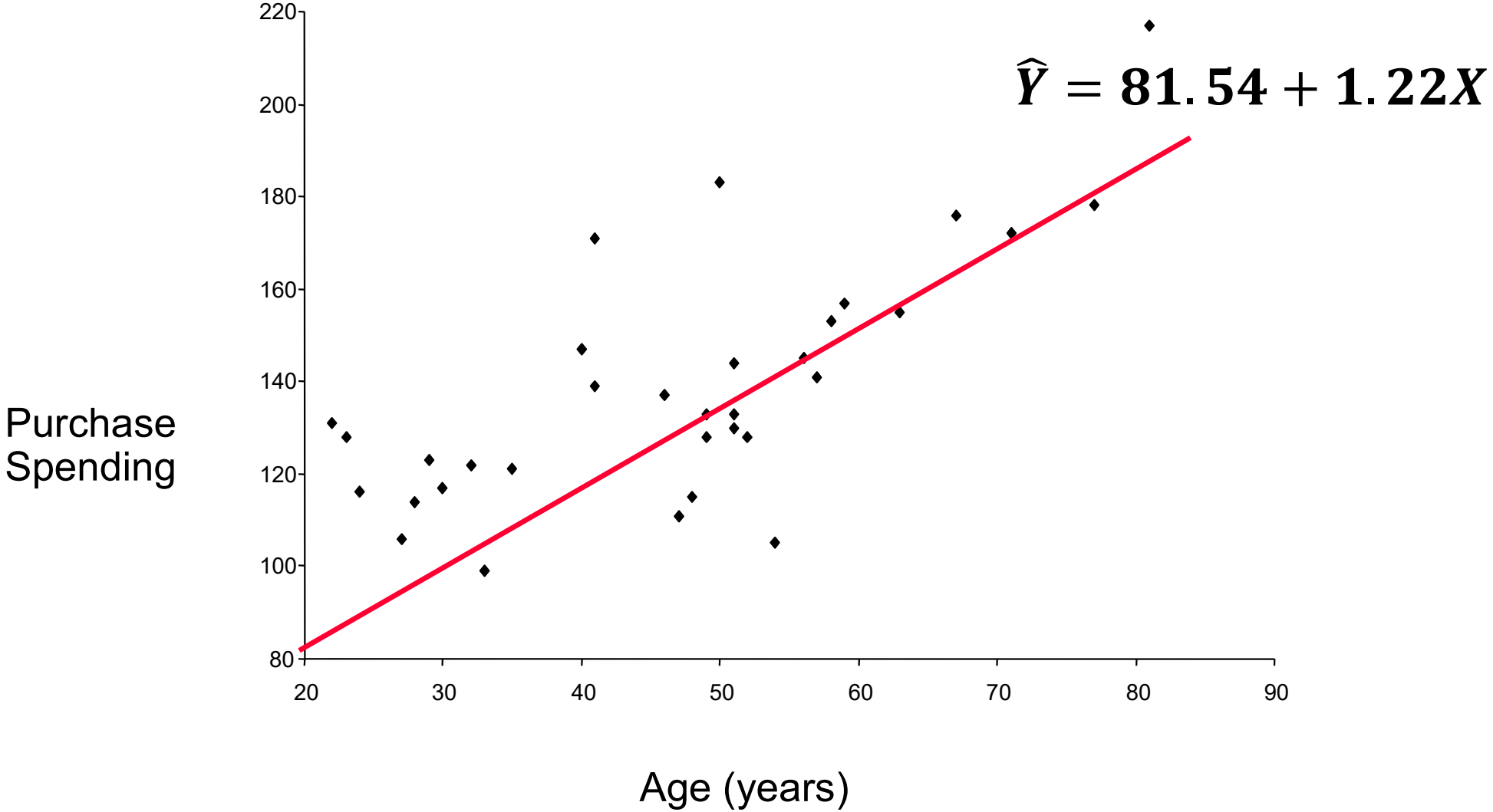
- Classification is a process of predicting a nominal variable with multiple response categories, classes or labels
  - Assigns an observation to a category

- Popular classification methods or *classifiers*:
  - Logistic regression
  - Discriminant analysis
  - Naïve Bayes
  - K-nearest neighbors

Linear Regression

Table 1 Age and purchase spending ($) among 33 adult women

Age	$	Age	$	Age	$
22	131	41	139	52	128
23	128	41	171	54	105
24	116	46	137	56	145
27	106	47	111	57	141
28	114	48	115	58	153
29	123	49	133	59	157
30	117	49	128	63	155
32	122	50	183	67	176
33	99	51	130	71	172
35	121	51	133	77	178
40	147	51	144	81	217


$\widehat{Y} = 81.54 + 1.22X$
Purchase Spending
220
200
180
160
140
120
100
80
20
30
40
50
60
70
80
90
Age (years)

Linear Regression Model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_P X_P + e$$

- Intercept (β_0):
  - Average value of Y when $X_p = 0$
- Slope (β_p):
  - Amount by which y changes on average when X_j changes by one unit, holding all the other X_ps constant.
- Regression coefficients are typically estimated via least squares.

What is Logistic Regression?

- Used when you have a binary response:
  - Yes – no
  - Positive – negative
  - Good credit – bad credit
  - Buyer – not buyer
  - Left – stayed
  - Dead – alive

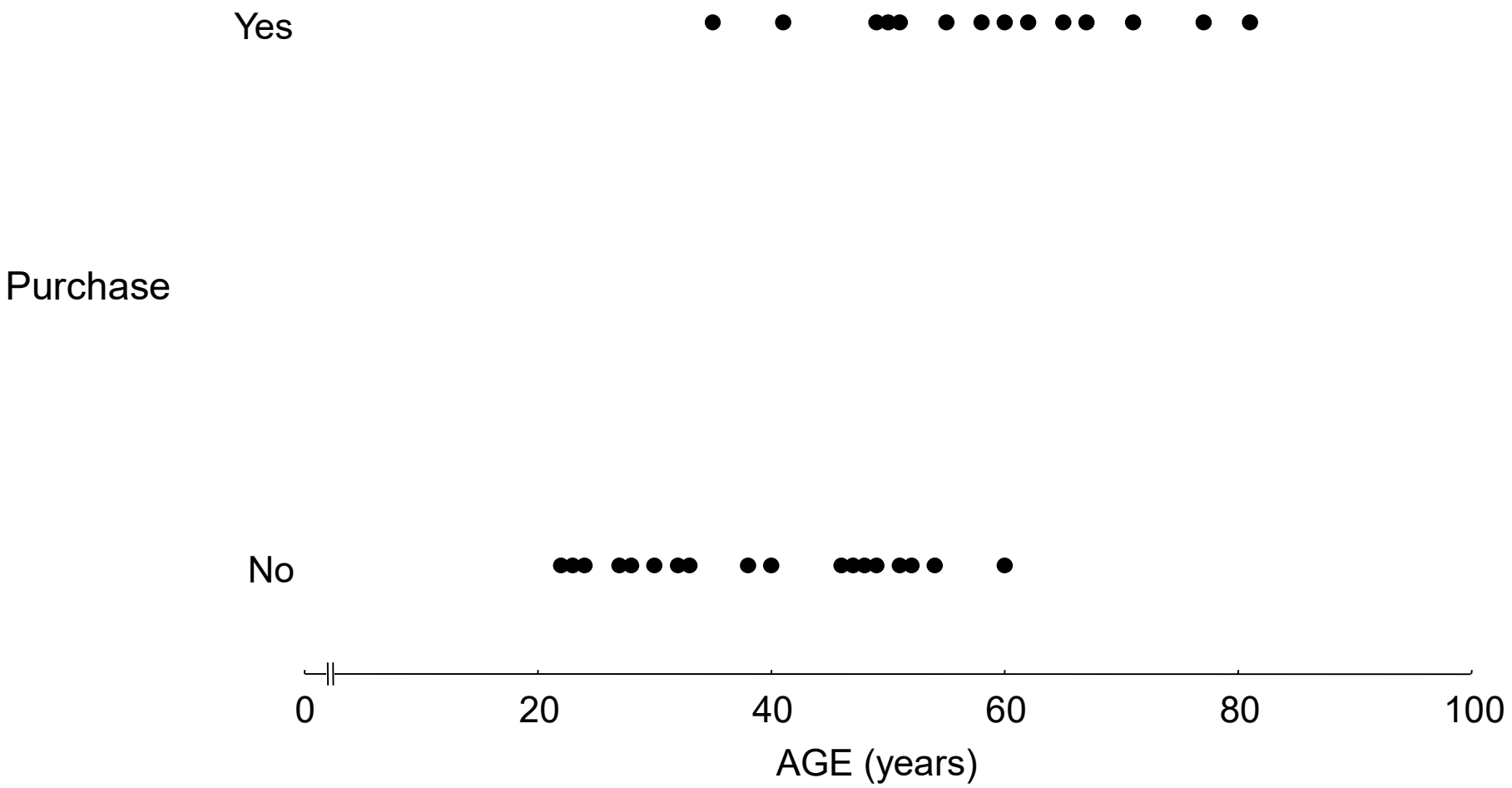
Logistic Regression

Table 2. Age (x) and CD Purchase (yes = 1, no = 0)

Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1

Logistic Regression

Data from Table 2



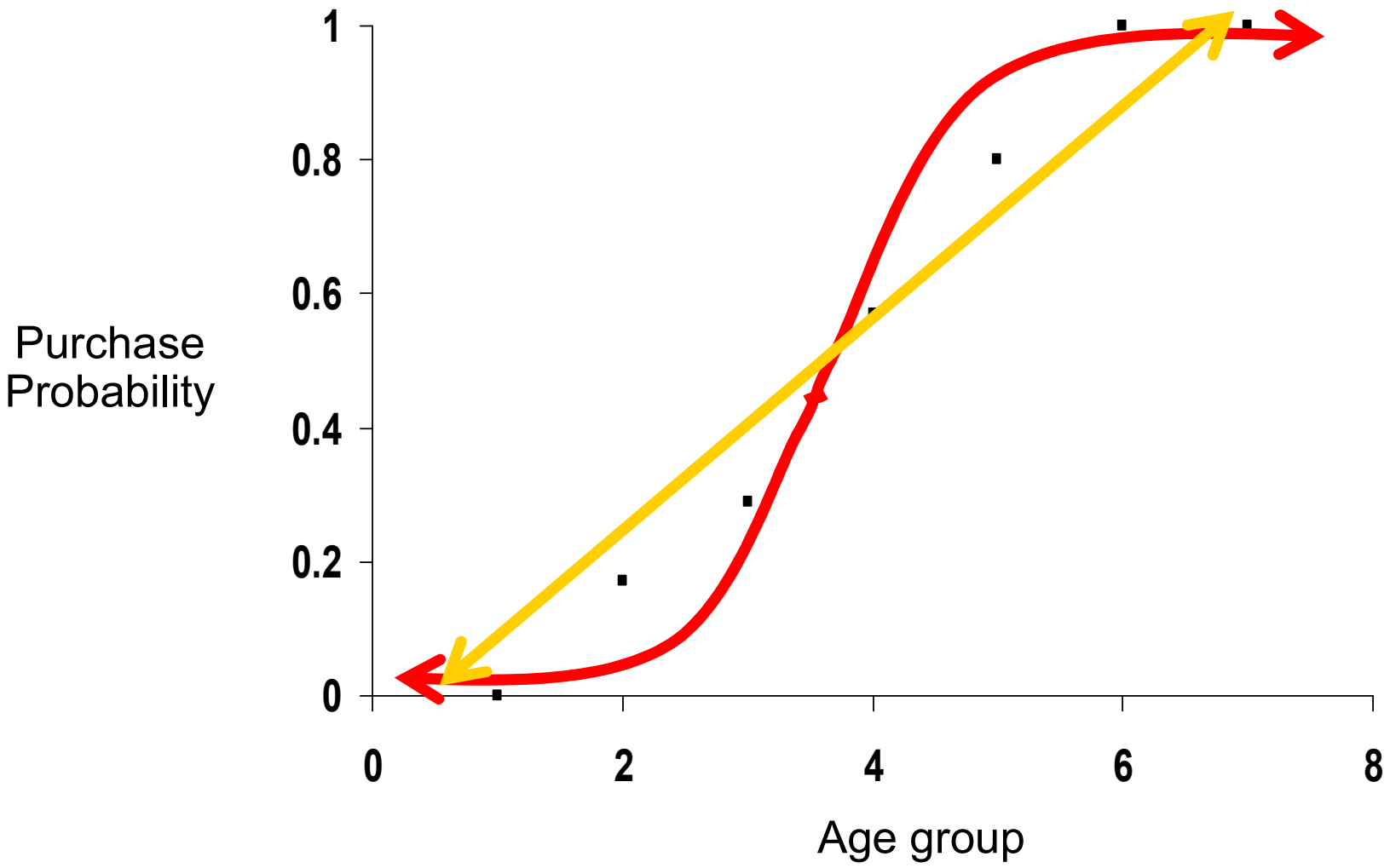
Logistic Regression

Table 3. Probability of CD purchase per age group

		CD Purchase	
Age group	# in group	#	prob
20 - 29	5	0	0
30 - 39	6	1	.17
40 - 49	7	2	.29
50 - 59	7	4	.57
60 - 69	5	4	.80
70 - 79	2	2	1.0
80 - 89	1	1	1.0

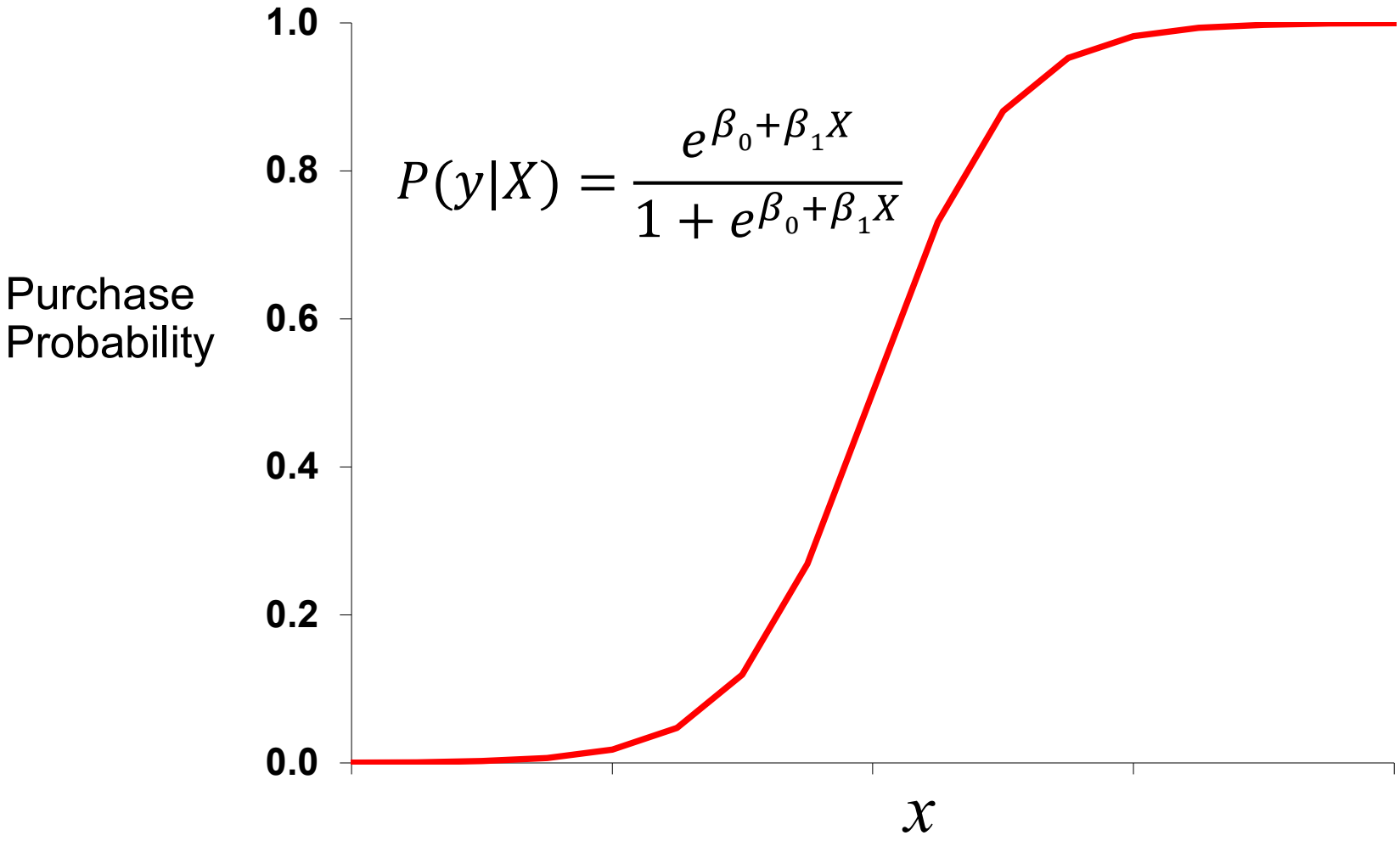
$p(y = 1 | x = \text{age group})$

Logistic Regression



Logistic Function

$$P(y|X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$



Logistic Transformation

$$P(y|X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

$$\underbrace{ln\left[\frac{P(y|X)}{1-P(y|X)}\right]}_{\text{logit of } P(y|x)} = \beta_0 + \beta_1 X$$

$$f(X) = \beta_0 + \beta_1 X \quad \Leftarrow \text{ Linearity}$$

Logistic Regression Model

- The statistical model for logistic regression is

$$ln\left[\frac{P(y|X)}{1-P(y|X)}\right] = \beta_0 + \beta_1 X,$$

where

$$P(y|X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

Multiple Logistic Regression

- More than one predictor
  - Can be discrete or continuous

$$ln\left[\frac{P(y|X)}{1-P(y|X)}\right] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_P X_P$$

- We typically use a method called maximum likelihood to estimate the coefficients from the training data.

Interpreting Coefficients

- Odds: The ratio of the proportions for two possible outcomes. If p is the proportion for one outcome, then 1-p is the proportion for the other outcome.

$$\text{Odds} = \frac{p}{1-p}$$

  - For example, p = the proportion of binge drinkers in college = .2, and 1-p = the proportion of students who are not binge drinkers = .8. Then, the odds of a college student being a binge drinker are .25 (or ¼).
  - The odds can take on any value between 0 and ∞. The larger, the higher probability of y = 1.

Interpreting Coefficients

- For the binge-drinking example, let's consider a predictor (X): Sex (1 = men & 0 = women).
- The log-odds for men:

$$\ln\left(\frac{P}{1-P}\right)_M = \beta_0 + \beta_1$$

- The log-odds for women:

$$\ln\left(\frac{P}{1-P}\right)_W = \beta_0$$

Interpreting Coefficients

- The slope β_1 indicates the difference between the log-odds for men and women.

$$\ln\left(\frac{P}{1-P}\right)_M - \ln\left(\frac{P}{1-P}\right)_W = \beta_1$$

- This can be re-expressed as Odds Ratio (OR)

$$\frac{\mathbf{Odds}_m}{\mathbf{Odds}_w} = e^{\beta_1}$$

Interpreting Coefficients

- For example,

$$\frac{\mathbf{Odds_m}}{\mathbf{Odds_w}} = 1.4 \text{ or } \mathbf{Odds_m} = 1.4 \times \mathbf{Odds_w}$$

Then, the odds for men are 1.4 times for the odds for women.

- $\text{Exp}(\beta_p)$ = Odds Ratio (OR): Change in the odds by one unit increase in X_p with all the other X's constant.

Interpreting Coefficients

- Exp(β_p):
  - indicates how the odds that y = 1 will change if X_p increases by one unit with all the other Xs constant.
  - positive $\beta_p \rightarrow \exp(\beta_p) > 1$ (= odds increased)
  - negative $\beta_p \rightarrow \exp(\beta_p) < 1$ (= odds decreased)

- The statistical significance of an individual coefficient.
  - Wald test: $W_p = \dfrac{\hat{\beta}_p^2}{SE(\hat{\beta}_p)^2}$

Estimating Probabilities

- Once the coefficients are estimated, it is simple to compute the probability of y = 1 for any given X values in a training or test sample. For example,

$$\hat{P}(y|X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}}$$

Performance Checking: Classification Table/Confusion Matrix

- One way of assessing the training/test sample performance of a model is to look at the Classification Table or Confusion Matrix.
  - This is a 2 x 2 table which shows how correctly a model predicts the outcome category of cases.
  - The columns of the table are the two predicted values of the DV, while the rows are the two observed (actual) values of the DV.
  - In a perfect model, all cases will be on the diagonal and the overall percent correct will be 100%.
  - Note that this table should not be used as a goodness-of-fit measure because it ignores actual predicted probabilities and instead use dichotomized predictions based on a cutoff (e.g., .5).

*BBB_training.sav [DataSet1] - IBM SPSS Statistics Data Editor

File Edit View Data Transform Analyze Graphs Utilities Extensions Window Help

Visible: 14 of 14 Varia

	acctnum	gender	last	book$	child	youth	cook	do_it	reference	art	geog	buyer	probability	binary_pred	var	var	var	var	var
1	10003	1	15	25.00	0	0	2	0	0	0	0	0	.01515	0					
2	10006	1	7	15.00	0	1	0	0	0	0	0	1	.04725	0					
3	10013	1	13	15.00	0	0	0	1	0	0	0	0	.01886	0					
4	10015	1	25	15.00	0	0	1	0	0	0	0	0	.00747	0					
5	10016	1	1	23.00	2	0	0	0	0	0	0	0	.06565	0					
6	10017	1	7	39.00	0	0	2	0	0	0	1	0	.05761	0					
7	10019	1	11	26.00	0	0	1	1	0	0	0	0	.01740	0					
8	10022	1	15	15.00	0	0	0	0	1	0	0	0	.03198	0					
9	10025	1	13	25.00	0	1	1	0	0	0	0	0	.02107	0					
10	10026	1	15	15.00	1	0	0	0	0	0	0	0	.02060	0					
11	10030	1	13	15.00	1	0	0	0	0	0	0	0	.02489	0					
12	10035	1	13	29.00	0	0	0	0	0	1	1	0	.16501	0					
13	10036	1	9	15.00	0	0	0	1	0	0	0	0	.02754	0					
14	10037	1	7	15.00	0	0	0	0	0	0	1	0	.09985	0					
15	10040	1	9	15.00	1	0	0	0	0	0	0	0	.03624	0					
16	10042	1	11	25.00	1	0	1	0	0	0	0	0	.02348	0					
17	10044	1	13	101.00	1	1	3	2	0	1	1	0	.02445	0					
18	10045	1	33	15.00	0	0	1	0	0	0	0	0	.00345	0					
19	10046	1	21	78.00	2	1	2	0	1	0	1	0	.01250	0					
20	10048	0	29	126.00	3	0	2	2	0	3	1	0	.10771	0					
21	10049	1	11	26.00	1	0	0	1	0	0	0	0	.01845	0					
22	10051	1	9	15.00	0	0	1	0	0	0	0	0	.03422	0					
23	10052	1	21	23.00	2	0	0	0	0	0	0	0	.01003	0					
24	10054	1	17	27.00	0	0	1	0	1	0	0	0	.01981	0					
25	10055	1	5	137.00	2	1	3	0	1	1	4	1	.51409	1					
26	10058	0	3	59.00	1	1	1	1	0	0	1	0	.10203	0					
27	10062	1	13	15.00	0	0	0	0	0	1	0	0	.09738	0					
28	10063	0	13	67.00	1	0	3	1	0	1	0	0	.05013	0					
29	10064	1	1	29.00	0	1	0	0	0	0	1	0	.13973	0					

Data View Variable View

IBM SPSS Statistics Processor is ready Unicode:ON

Performance Checking: Classification Table/Confusion Matrix

Observed \ Predicted	No (0)	Yes (1)	Total
No (0)	True Negative (TN)	False Positive (FP)	N
Yes (1)	False Negative (FN)	True Positive (TP)	P
Total	N*	P*	

Performance Checking: Classification Table/Confusion Matrix

- Two types of correct classification
  - Sensitivity (민감도): the percentage of Y = 1 (yes) cases that are correctly identified
    - 100*(TP/P)
  - Specificity (특이성): the percentage of Y = 0 (no) cases that are correctly identified
    - 100*(TN/N)

True state (population information)	Decision (sample information): Retain H_0	Decision (sample information): Reject H_0
H_0 true	Correct Decision	**Type I Error**
H_0 false	**Type II Error**	Correct Decision

- α = probability of committing Type I error
- β = probability of committing Type II error
- 1- β (power) = probability of correctly rejecting a false H_0

Performance Checking: Classification Table/Confusion Matrix

Observed	Predicted: No (0)	Predicted: Yes (1)	Total
No (0)	True Negative (TN)	False Positive (FP)	N
Yes (1)	False Negative (FN)	True Positive (TP)	P
Total	N*	P*	

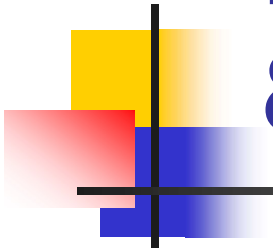
Performance Checking: Classification Table/Confusion Matrix

Name	Definition	Synonyms
False Positive Rate	FP/N	Type I error, 1 – Specificity
True Positive Rate	TP/P	Power, Sensitivity, Recall
Positive Predictive Value	TP/P*	Precision, 1 – False Discovery Proportion
Negative Predictive Value	TN/N*	

Performance Checking: Classification Table/Confusion Matrix

Name	Definition	Synonyms
Accuracy	(TP + TN) / (TP + TN + FP + FN)	
Misclassification	(FP + FN) / (TP + TN + FP + FN)	

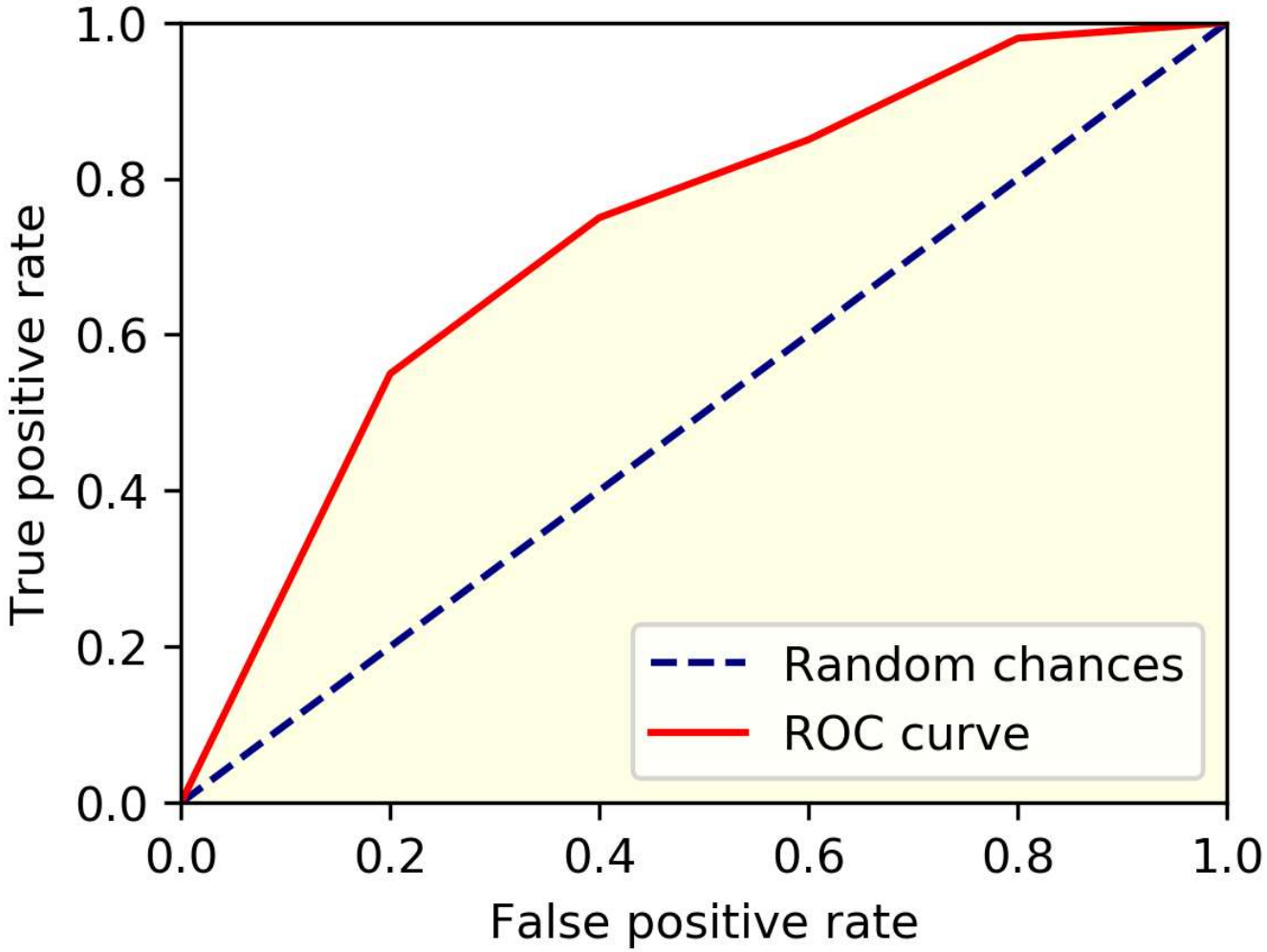
Performance Checking: ROC Curve & AUC



- By default, an observation is assigned to class 1 if $p(Y = 1|X = x) > .5$. That is, a threshold of 50% for the probability is used.

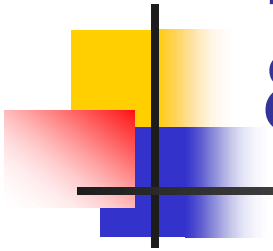
- The ROC (Receiver Operating Characteristics) curve is a popular graphic for simultaneously displaying two types of classification for all possible thresholds.
  - True Positive Rate vs. False Positive Rate

Performance Checking: ROC Curve & AUC



https://towardsdatascience.com/a-simple-explanation-of-the-roc-curve-and-auc-64db32d75541

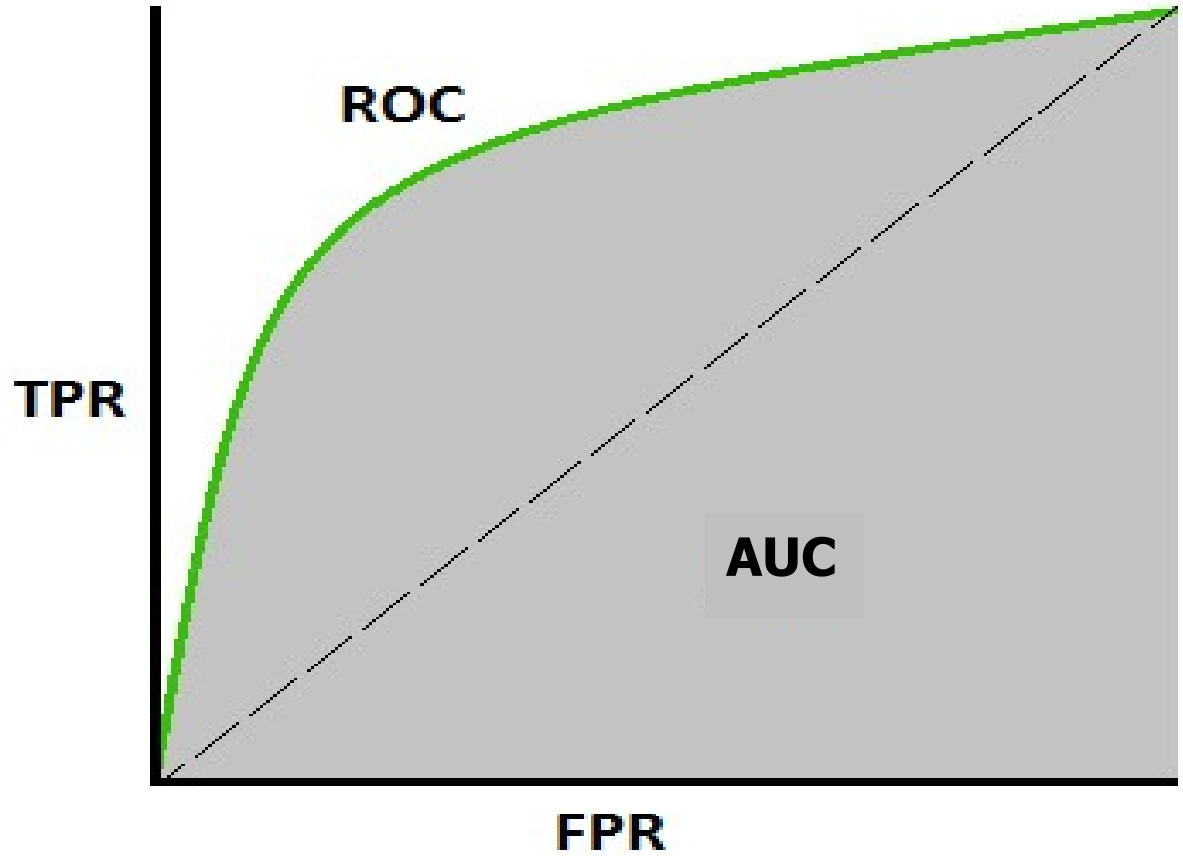
Performance Checking: ROC Curve & AUC



- An ideal ROC curve will hug the top left corner, indicating a high true positive rate and a low false positive rate.

- The overall performance of a classifier, summarized over all possible thresholds, is given by the Area Under the Curve (AUC).
  - The larger the AUC, the better the classifier.
  - If a classifier's AUC = .5, it performs no better than chance.

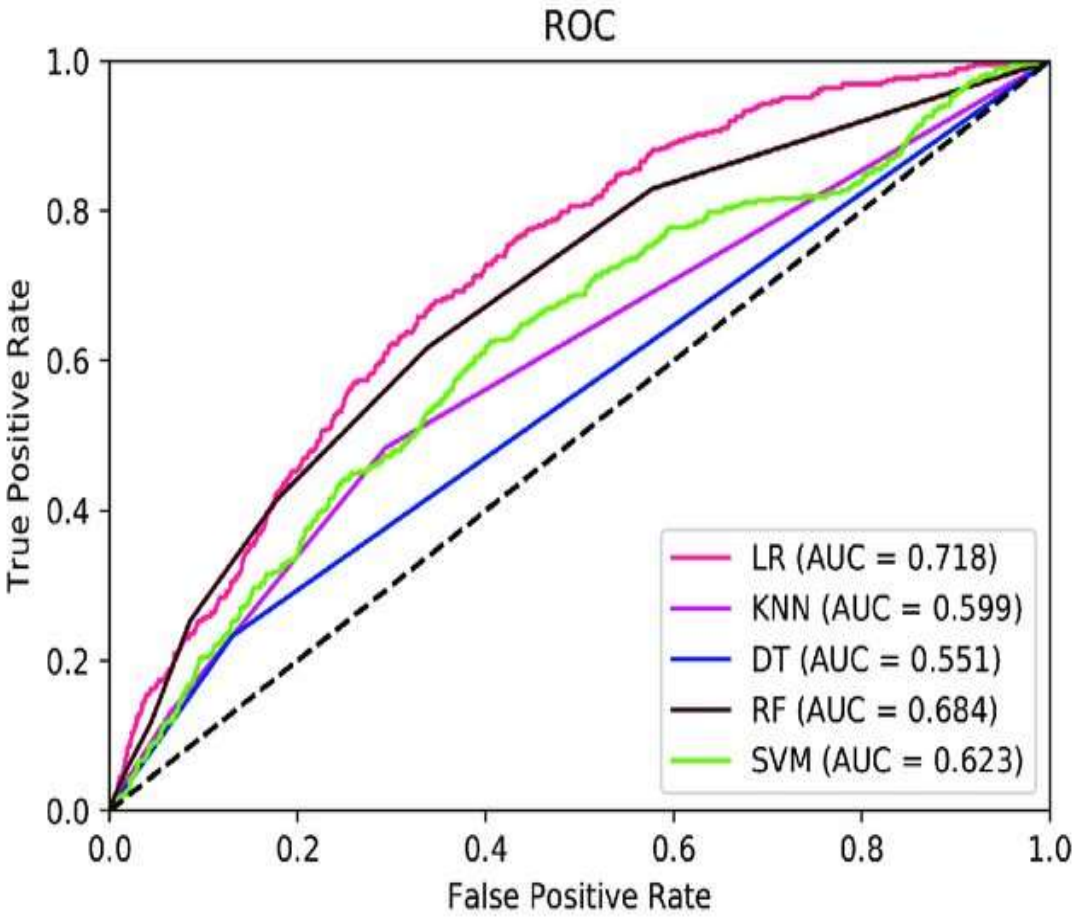
Performance Checking: ROC Curve & AUC



https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Performance Checking: ROC Curve & AUC

- ROC curves (and AUC values) are useful for comparing different classifiers.



Park H, Kim K. Comparisons among Machine Learning Models for the Prediction of Hypercholestrolemia Associated with Exposure to Lead, Mercury, and Cadmium. *International Journal of Environmental Research and Public Health*. 2019; 16(15):2666. https://doi.org/10.3390/ijerph16152666

Example: Logistic Regression

- The BookBinder Book Club data (BBB_training.csv & BBB_test.csv)
  - DV:
    - Buyer: Bought "Art History of Florence?"
  - Predictors:
    - Gender: 0 = male, 1 = female
    - Last : Months since last purchase
    - Book: Total $ spent on books
    - Art: # purchases of Art books
    - Child: # purchases of Children's books
    - Youth: # purchases of Youth books
    - Cook: # purchases of Cookbooks
    - Do_it: # purchases of Do-it-yourself books
    - Reference: # purchases of Reference books
    - Geog: # purchases of Geography books

Example: Logistic Regression

```
Call:
glm(formula = buyer ~ gender + last + book + art + child + youth +
    cook + do_it + reference + geog, family = binomial, data = mydata)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5014  -0.4092  -0.2734  -0.1791   3.3182

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.342705   0.075834 -17.706  < 2e-16 ***
gender      -0.761378   0.050699 -15.018  < 2e-16 ***
last        -0.096848   0.003962 -24.443  < 2e-16 ***
book        -0.022195   0.012458  -1.782  0.07481 .
art          1.469313   0.157176   9.348  < 2e-16 ***
child        0.027970   0.124555   0.225  0.82232
youth        0.111025   0.124815   0.890  0.37372
cook        -0.031559   0.135510  -0.233  0.81584
do_it       -0.255634   0.143092  -1.787  0.07402 .
reference    0.479709   0.150708   3.183  0.00146 **
geog         0.916020   0.164505   5.568 2.57e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Example: Logistic Regression

Exp($\widehat{\beta}$) = OR

		2.5 %	97.5 %
(Intercept)	0.2611383	0.2250304	0.3029367
gender	0.4670225	0.4228475	0.5158256
last	0.9076938	0.9006102	0.9147080
book	0.9780497	0.9544354	1.0022062
art	4.3462469	3.1958410	5.9182464
child	1.0283649	0.8055285	1.3126619
youth	1.1174233	0.8749569	1.4272501
cook	0.9689335	0.7428623	1.2636643
do_it	0.7744253	0.5850183	1.0251891
reference	1.6156037	1.2026832	2.1714549
geog	2.4993232	1.8109789	3.4514501

Example: Logistic Regression

Training sample

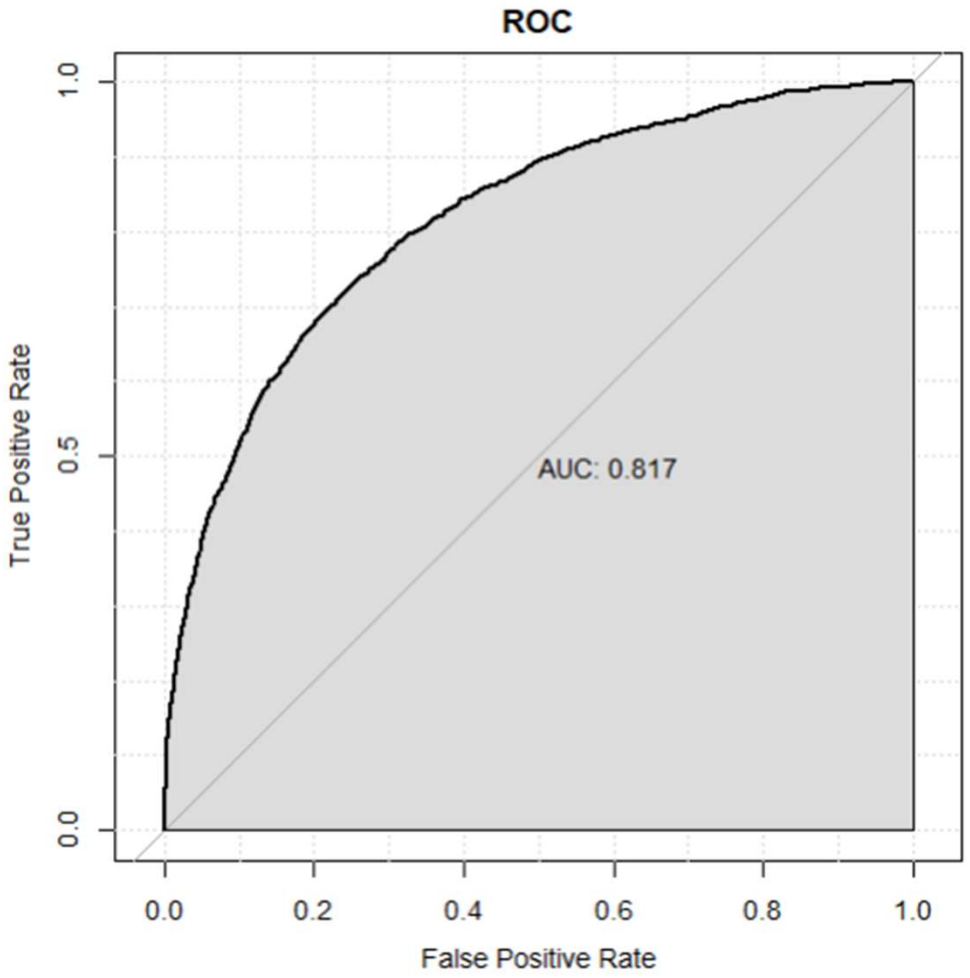
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 22736  1901
         1   190   368

               Accuracy : 0.917
                 95% CI : (0.9135, 0.9204)
    No Information Rate : 0.9099
    P-Value [Acc > NIR] : 3.894e-05

                  Kappa : 0.2331

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.16219
            Specificity : 0.99171
         Pos Pred Value : 0.65950
         Neg Pred Value : 0.92284
             Prevalence : 0.09006
         Detection Rate : 0.01461
   Detection Prevalence : 0.02215
      Balanced Accuracy : 0.57695

       'Positive' Class : 1
```

Test sample

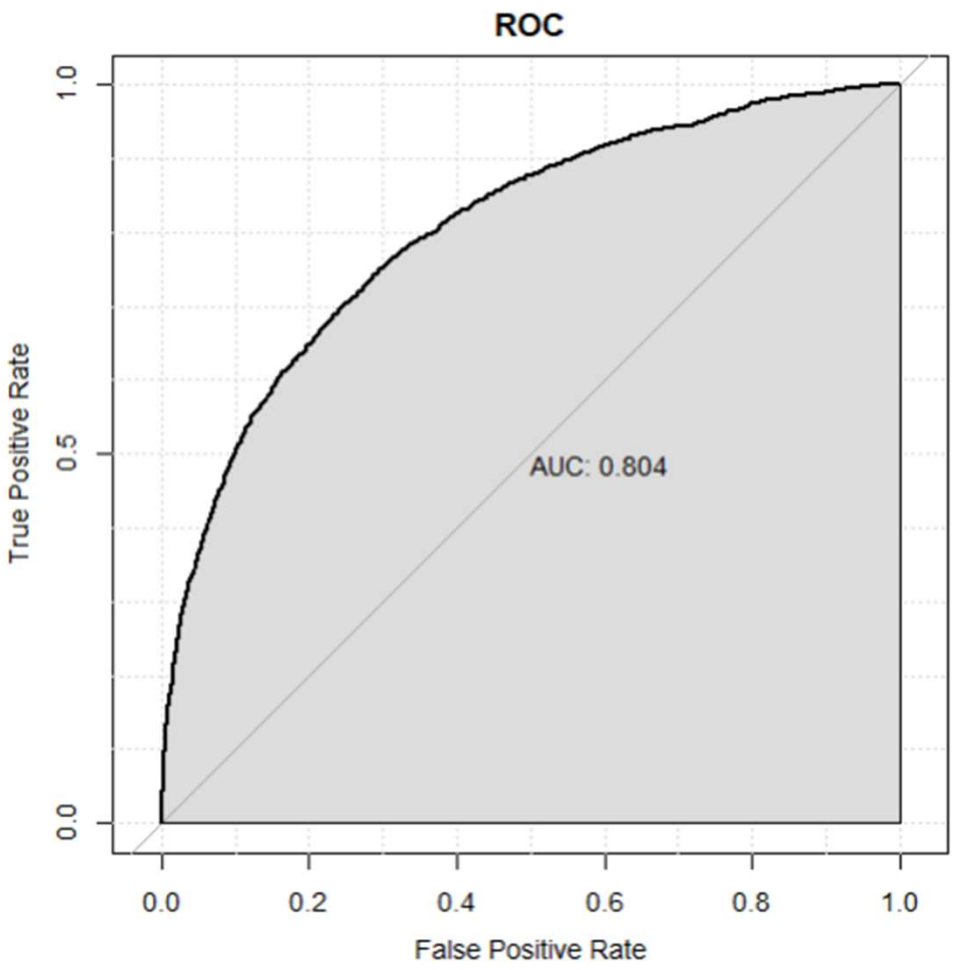
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 22365  1921
         1   187   332

               Accuracy : 0.915
                 95% CI : (0.9115, 0.9185)
    No Information Rate : 0.9092
    P-Value [Acc > NIR] : 0.0006381

                  Kappa : 0.2128

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.14736
            Specificity : 0.99171
         Pos Pred Value : 0.63969
         Neg Pred Value : 0.92090
             Prevalence : 0.09083
         Detection Rate : 0.01338
   Detection Prevalence : 0.02092
      Balanced Accuracy : 0.56953

       'Positive' Class : 1
```

Example: Logistic Regression

Training sample



Test sample

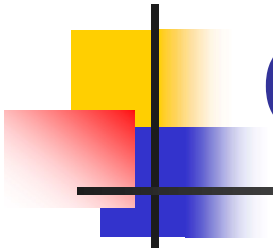


Logistic Regression for > 2 Classes

- We may need to classify a response variable that has more than two classes. For example,

$$Y \begin{cases} 1 = \text{stroke} \\ 2 = \text{drug overdose} \\ 3 = \text{epileptic seizure} \end{cases}$$

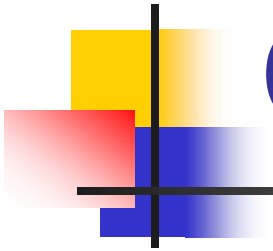
- It is straightforward to generalize (binary) logistic regression to more than two classes. This extension is known as multinomial logistic regression (or multiclass logistic regression).



Other Classification Methods

- Logistic regression involves directly modeling P(Y|X) using the logistic function.
  - We model the conditional probability of Y given X.

- We now consider alternative and less direct methods for estimating these probabilities.
  - We model the distributions of predictors (X) separately in each of the response classes. We then use Bayes' theorem to flip these around into estimates for P(Y|X).



Other Classification Methods

- Why consider another method over logistic regression?
  - When the classes are well-separated, the parameter estimates for the logistic regression model are unstable. The alternative methods that we will consider do not suffer from this problem.
  - If sample size is small and the distribution of X is approximately normal per class, the alternative methods are more stable that logistic regression.
  - The alternative methods are popular when we have more than 2 response classes.

Other Classification Methods

- Suppose that we wish to classify an observation into one of *K* classes ($K \geq 2$).
- π_k = the overall or prior probability that an observation comes from the *k*th class
  - ≈ class size
- $f_k(X) = P(X|Y = k)$: The density function of X for an observation that comes from the *k*th class

Other Classification Methods

- Then, Bayes' theorem states that

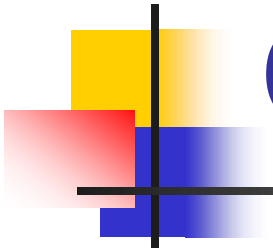
$$P(Y = k|X = x) = \frac{\pi_k f_k(X)}{\sum_{l=1}^{K} \pi_l f_l(X)}. \qquad \text{Eq.(1)}$$

- $P(Y = k|X = x)$ is called the posterior probability that an observation belongs to the *k*th class given the predictor value for the observation.

Other Classification Methods

- Instead of directly computing the posterior probability as in logistic regression, we can plug in estimates of π_k and $f_k(X)$ into Eq. (1).

- In general, it is easy to estimate π_k (the proportion of the training observations that belong to the *k*th class). Yet, estimating $f_k(X)$ is more difficult.



Other Classification Methods

- We discuss three classifiers that use different estimates of $f_k(X)$.
  - Linear discriminant analysis (LDA)
  - Quadratic discriminant analysis (QDA)
  - Naïve Bayes

Linear Discriminant Analysis with One Predictor

- Assume that we have only one predictor. Our task is to estimate $f_k(X)$ (and estimate Eq. (1)).

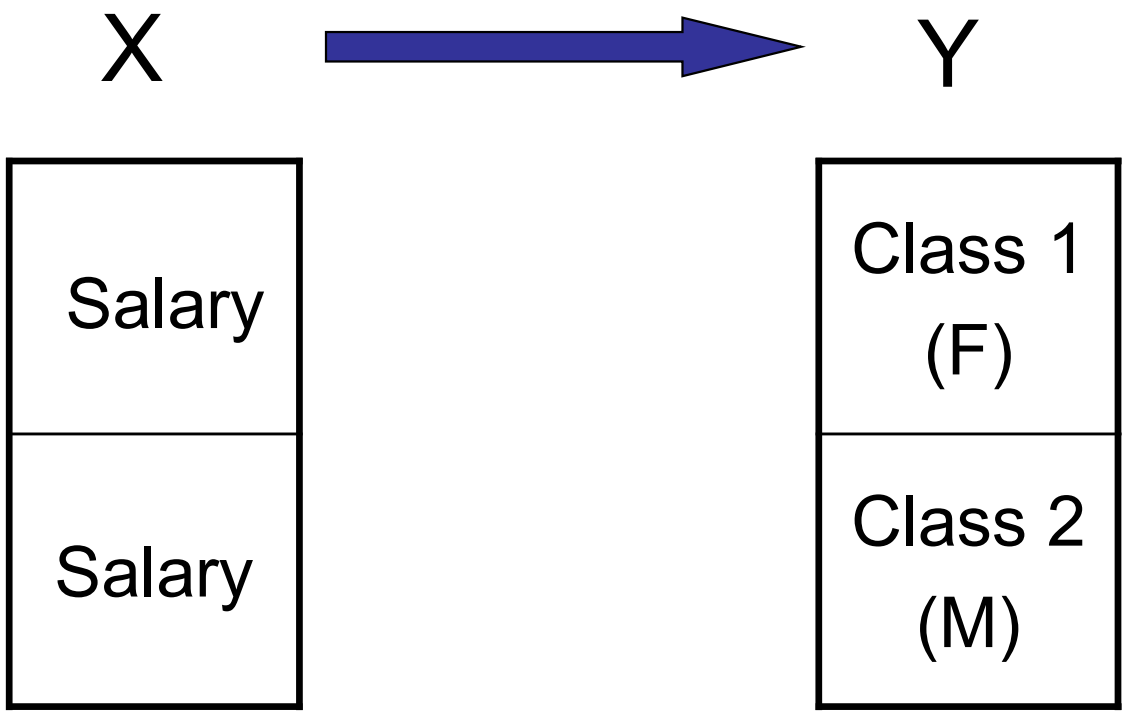
- In LDA, we assume that $f_k(X)$ is normal or Gaussian. When *P* = 1, the normal density takes the form

$$f_k(X) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(X-\mu_k)^2\right). \quad \text{Eq. (2)}$$

- We further assume that all variances are the same across *K* classes ($\sigma_1^2 = \ldots = \sigma_K^2$).

- $X|Y = k \sim N(\mu_k, \sigma^2)$

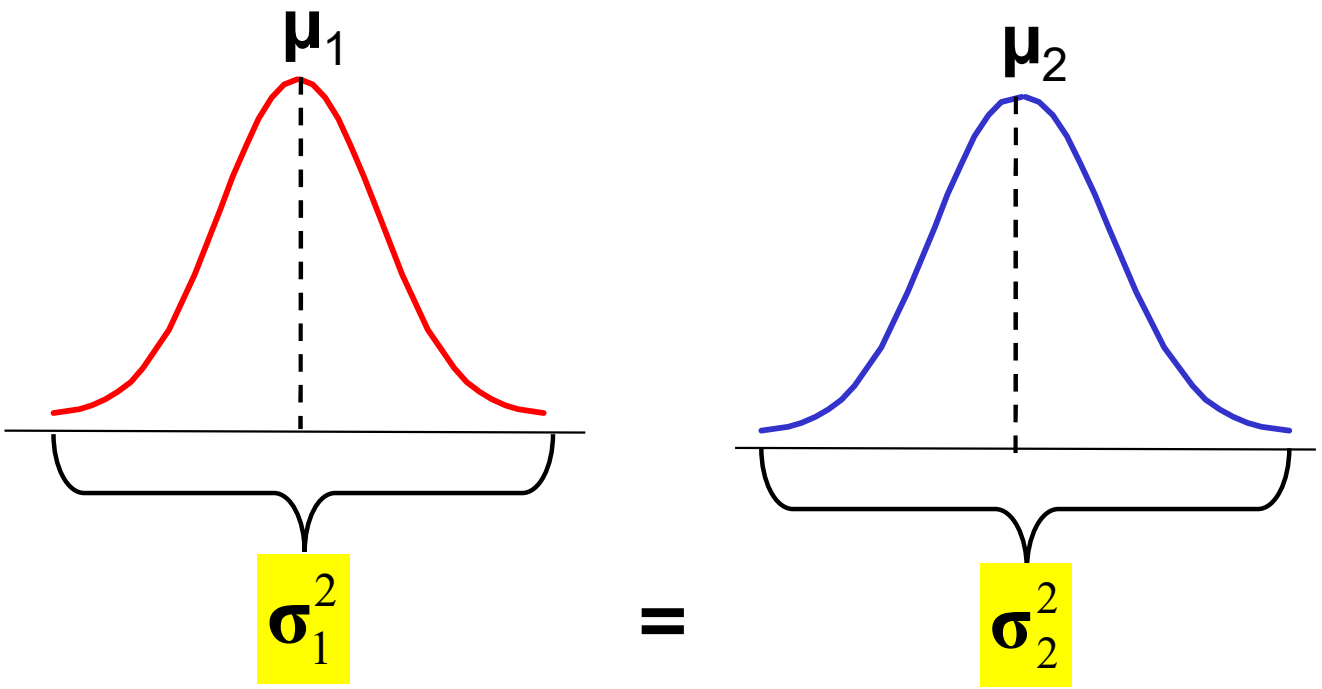
Linear Discriminant Analysis with One Predictor



Linear Discriminant Analysis with One Predictor

$X|Y = 1, 2$

Salary | Y = F	Salary | Y = M



$X|Y = 1 \sim N(\mu_1, \sigma^2)$

$X|Y = 2 \sim N(\mu_2, \sigma^2)$

Linear Discriminant Analysis with One Predictor

- Plugging Eq. (2) into Eq. (1), we find that

$$P(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left(-\frac{1}{2\sigma_l^2}(x-\mu_l)^2\right)}. \quad \text{Eq. (3)}$$

- We can classify an observation $X = x$ to the class for which Eq. (3) is largest.

Linear Discriminant Analysis with One Predictor

- Taking the log of Eq. (3) and rearranging the terms, this is equivalent to assigning the observation to the class for which

$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

  is largest.

- In LDA, we estimate μ_k as the average of all the training observations from the *k*th class; σ^2 as a weighed average of the sample variances for *K* classes; and π_k as the proportion of the training observations that belong to the *k*th class.

Linear Discriminant Analysis with One Predictor

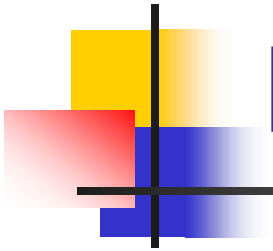
- The LDA classifier assigns an observation $X = x$ to the class for which

$$\hat{\delta}_k(x) = x\frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k)$$

  is largest.

- The word "linear" in LDA stems from the fact that the discriminant functions $\hat{\delta}_k(x)$ are linear functions of x.

Linear Discriminant Analysis with Multiple Predictors

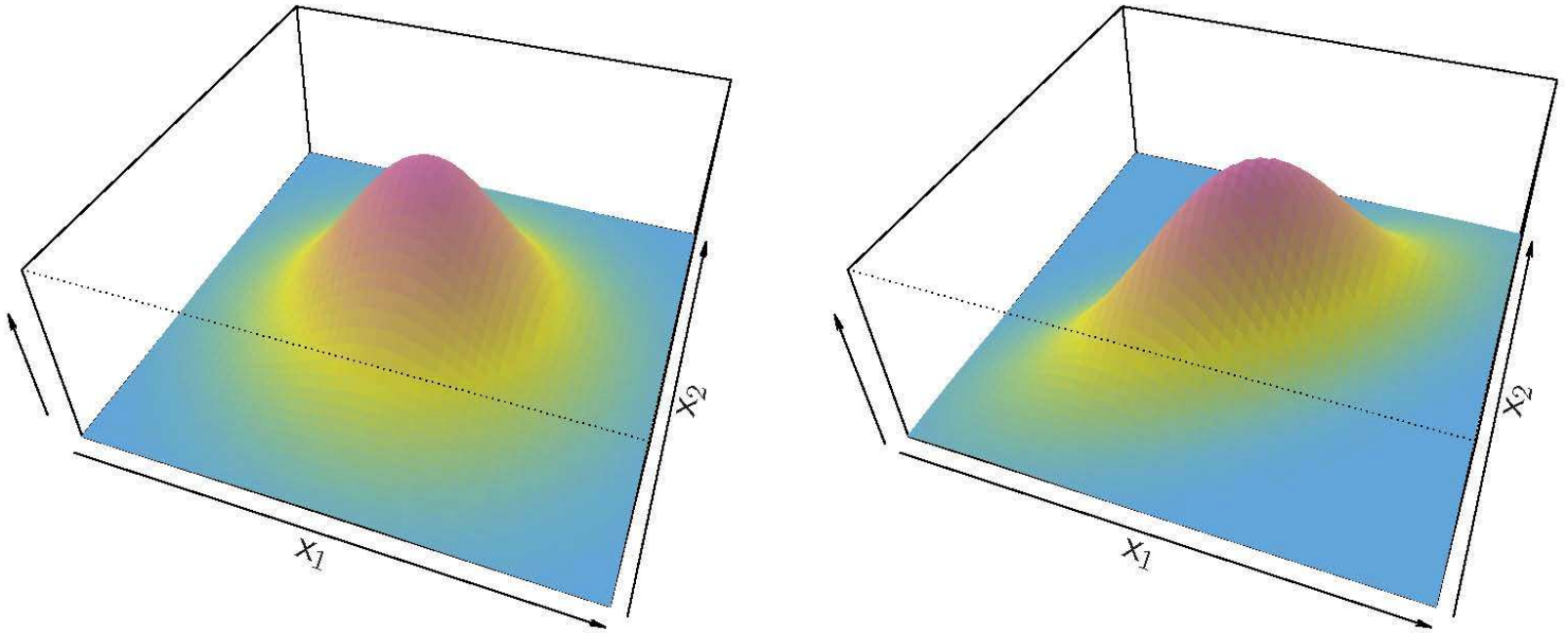


- In the case of P > 1 predictors, the LDA classifier assumes that the observations in the *k*th class are drawn from a multivariate normal distribution with P means and the covariance matrix of the predictors.

- The covariance matrix is assumed to be common to all *K* classes.
  - $\boldsymbol{X} | \, Y = k \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$

Linear Discriminant Analysis with Multiple Predictors

Multivariate normal distributions



James et al., (2021). Figure 4.5: Left: X1 and X2 are uncorrelated. Right: X1 and X2 are correlated (r = .7)

Linear Discriminant Analysis with Multiple Predictors

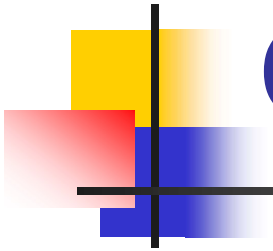
- The LDA classifier assigns an observation $X = \boldsymbol{x}$ to the class for which

$$\hat{\delta}_k(\boldsymbol{x}) = \boldsymbol{x}^T \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\mu}}_k - \frac{1}{2} \hat{\boldsymbol{\mu}}_k^T \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\mu}}_k + \log(\hat{\pi}_k)$$

  is largest.

- Again, the discriminant function $\hat{\delta}_k(\boldsymbol{x})$ is a linear function of $\boldsymbol{x}$.

Quadratic Discriminant Analysis



- Like LDA, quadratic discriminant analysis (QDA) assumes that the observations in the *k*th class are drawn from a multivariate normal distribution.

- Unlike LDA, QDA assumes that each class has its own covariance matrix.
  - $\boldsymbol{x}|\, Y = k \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

Quadratic Discriminant Analysis

- This leads the discriminant function to be a quadratic function of $\boldsymbol{x}$.

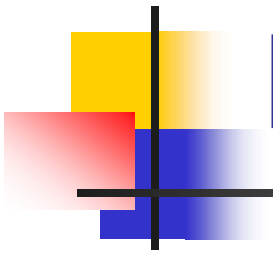
$$\hat{\delta}_k(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^T\widehat{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{x} + \boldsymbol{x}^T\widehat{\boldsymbol{\Sigma}}_k^{-1}\widehat{\boldsymbol{\mu}}_k - \frac{1}{2}\widehat{\boldsymbol{\mu}}_k^T\widehat{\boldsymbol{\Sigma}}_k^{-1}\widehat{\boldsymbol{\mu}}_k - \frac{1}{2}\log|\widehat{\boldsymbol{\Sigma}}_k| + \log(\hat{\pi}_k)$$

From $\hat{\delta}_k(\boldsymbol{x})$ to Probabilities

- Once we have estimates $\hat{\delta}_k(\boldsymbol{X})$, we can turn these into estimates for posterior probabilities:

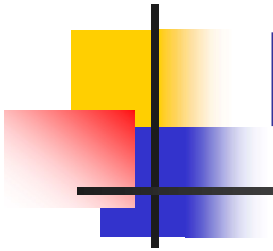
$$\hat{P}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{K} e^{\hat{\delta}_l(x)}}.$$

- Thus, classifying to the largest $\hat{\delta}_k(\boldsymbol{X})$ is equivalent to classifying to the class for which $\hat{P}(Y = k|X = x)$ is largest.



LDA VS. QDA

- Why would one prefer LDA to QDA or vice-versa?
- The answer lies in the bias-variance trade-off.
  - QDA involves much more parameters (i.e., each class's covariances) than LDA.
  - LDA is a much less flexible classifier than QDA, so has lower variance. This may improve prediction.
  - However, if LDA's assumption of a common covariance matrix for K classes is badly off, it can suffer from higher bias.



LDA VS. QDA

- Roughly speaking, LDA may be chosen over QDA if the number of training observations is relatively small.

- QDA may be recommended if the number of training observations is very large or if the assumption of a common covariance matrix for K classes is clearly untenable.

Example: Linear Discriminant Analysis

Training sample

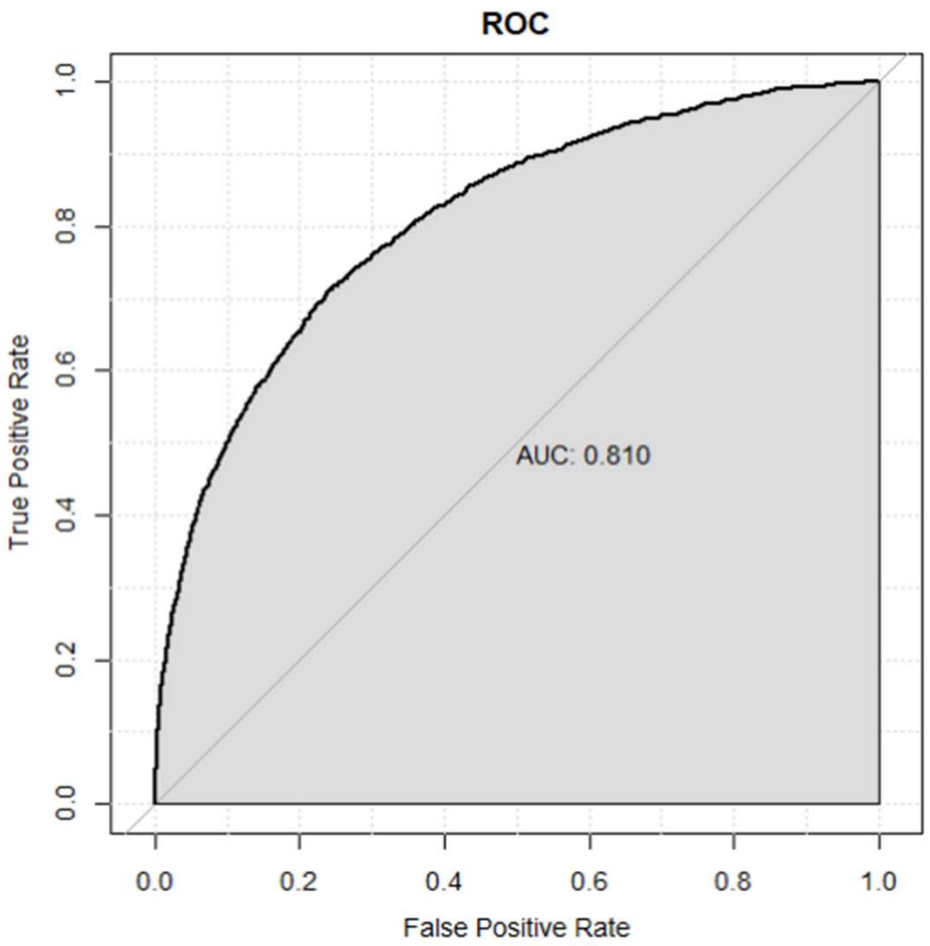
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 22455  1718
         1   471   551

               Accuracy : 0.9131
                 95% CI : (0.9096, 0.9166)
    No Information Rate : 0.9099
    P-Value [Acc > NIR] : 0.03955

                  Kappa : 0.2954

 Mcnemar's Test P-Value : < 2e-16

            Sensitivity : 0.24284
            Specificity : 0.97946
         Pos Pred Value : 0.53914
         Neg Pred Value : 0.92893
             Prevalence : 0.09006
         Detection Rate : 0.02187
   Detection Prevalence : 0.04056
      Balanced Accuracy : 0.61115

       'Positive' Class : 1
```

Test sample

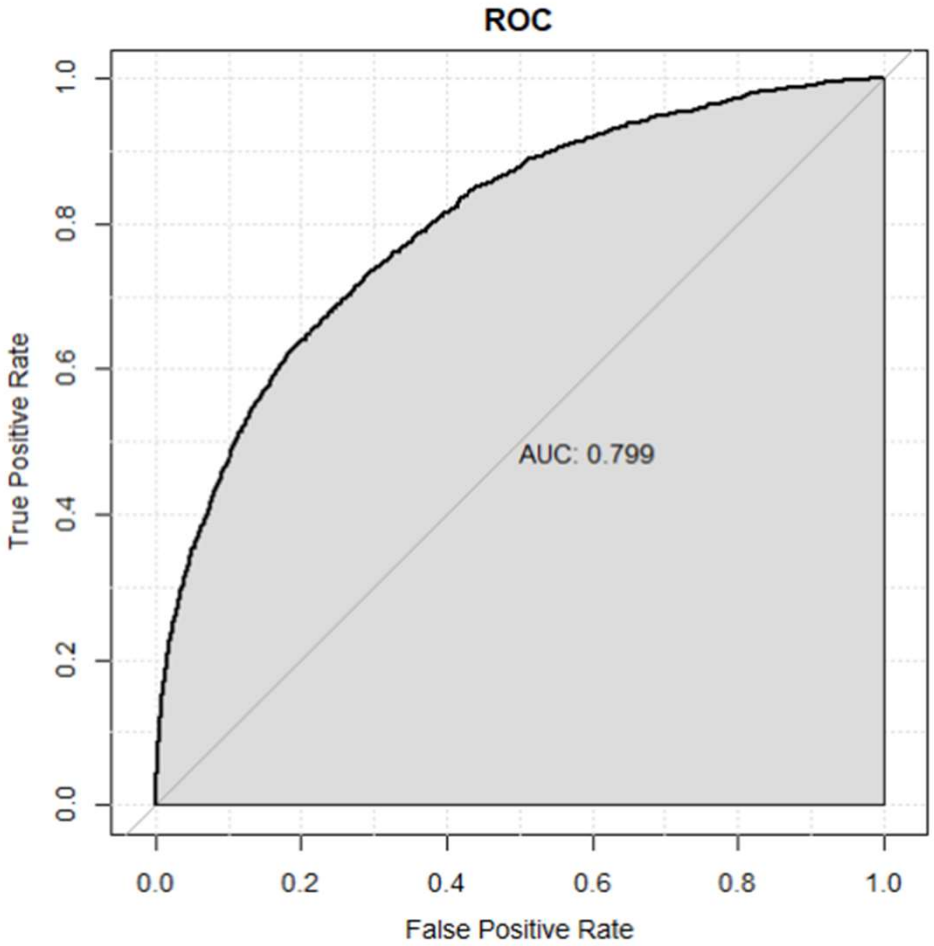
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 22084  1731
         1   468   522

               Accuracy : 0.9113
                 95% CI : (0.9077, 0.9149)
    No Information Rate : 0.9092
    P-Value [Acc > NIR] : 0.1183

                  Kappa : 0.2821

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.23169
            Specificity : 0.97925
         Pos Pred Value : 0.52727
         Neg Pred Value : 0.92731
             Prevalence : 0.09083
         Detection Rate : 0.02104
   Detection Prevalence : 0.03991
      Balanced Accuracy : 0.60547

       'Positive' Class : 1
```

Example: Linear Discriminant Analysis

Training sample



Test sample



Example: Quadratic Discriminant Analysis

Training sample

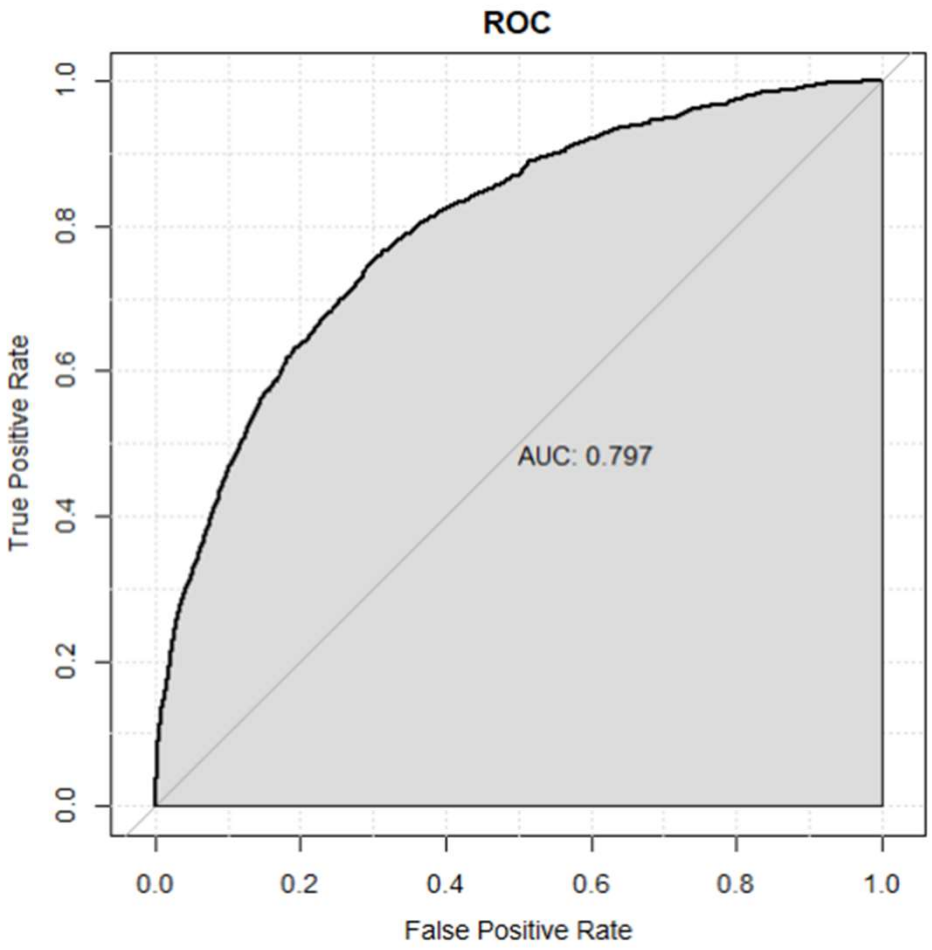
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 21738  1533
         1  1188   736

               Accuracy : 0.892
                 95% CI : (0.8881, 0.8958)
    No Information Rate : 0.9099
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2926

 Mcnemar's Test P-Value : 4.262e-11

            Sensitivity : 0.32437
            Specificity : 0.94818
         Pos Pred Value : 0.38254
         Neg Pred Value : 0.93412
             Prevalence : 0.09006
         Detection Rate : 0.02921
   Detection Prevalence : 0.07636
      Balanced Accuracy : 0.63628

       'Positive' Class : 1
```

Test sample

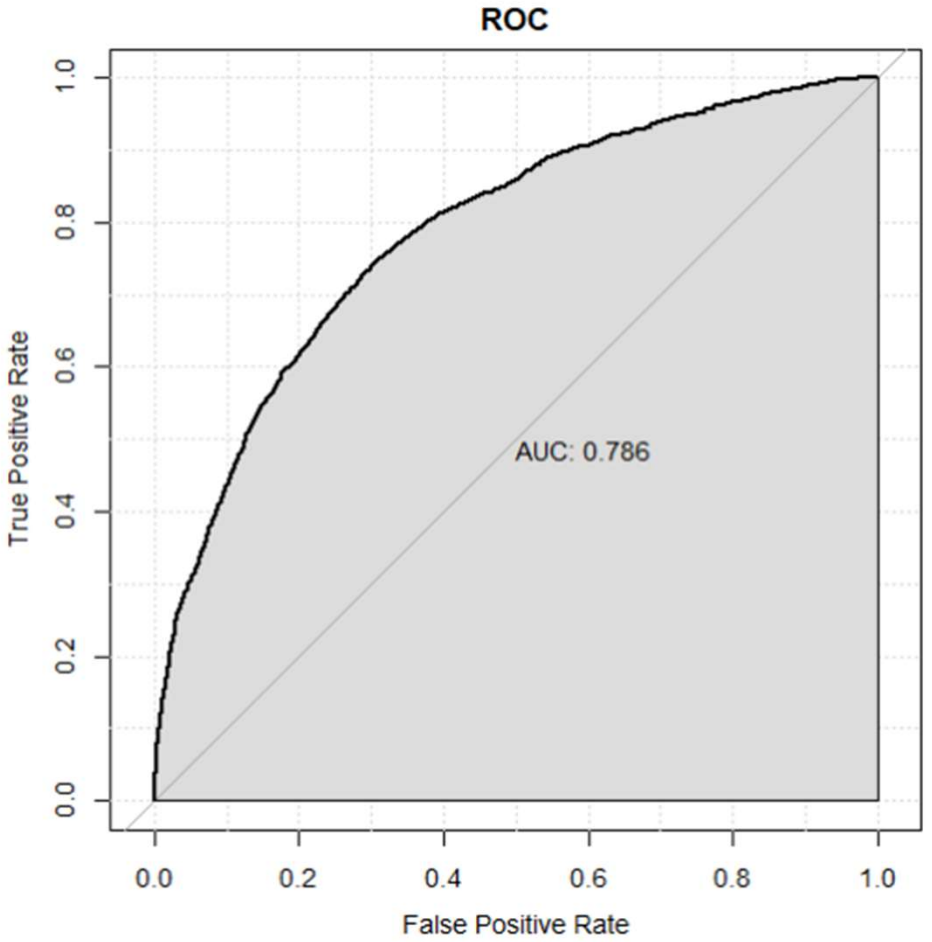
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 21377  1559
         1  1175   694

               Accuracy : 0.8898
                 95% CI : (0.8858, 0.8937)
    No Information Rate : 0.9092
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2772

 Mcnemar's Test P-Value : 2.391e-13

            Sensitivity : 0.30803
            Specificity : 0.94790
         Pos Pred Value : 0.37132
         Neg Pred Value : 0.93203
             Prevalence : 0.09083
         Detection Rate : 0.02798
   Detection Prevalence : 0.07535
      Balanced Accuracy : 0.62797

       'Positive' Class : 1
```

Example: Quadratic Discriminant Analysis

Training sample



Test sample



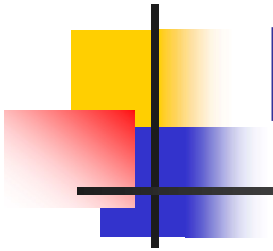
Naïve Bayes

- The naïve Bayes (also known as "Idiot's Bayes") classifier estimates $f_k(X)$ in a different way. Instead of assuming a particular family of distributions (e.g., multivariate normal) for the density function, it makes a single assumption: "Within the *k*th class, the P predictors are independent."

- Mathematically, this assumption means that for *k* = 1,…, *K*,

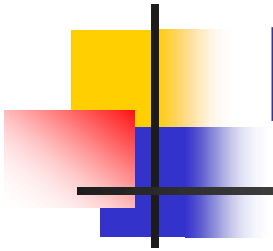
$$f_k(X) = f_{k1}(X_1) \times f_{k2}(X_2) \times \cdots \times f_{kp}(X_p) = \prod_{p=1}^{P} f_{kp}(X_p),$$

where f_{kp} is the density function of the *p*th predictor among observations in the *k*th class.



Naïve Bayes

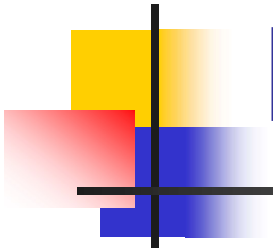
- Why is this assumption so powerful? Essentially, estimating a P-dimensional density function is challenging because we must consider not only the marginal distribution of each predictor — that is, the distribution of each predictor on its own — but also the joint distribution of the predictors — that is, the association between the different predictors.
  - In the case of a multivariate normal distribution, the association between the different predictors is summarized by the off-diagonal elements of the covariance matrix.
- However, in general, this association can be very challenging to estimate.



Naïve Bayes

- By assuming that the P predictors are independent within each class, we eliminate the need to worry about the association between the predictors, because we have assumed that there is no association between the predictors. Thus, it can simplify the estimation of the density function dramatically.

- In most settings, the naïve Bayes assumption is rather too optimistic and is generally not true.



Naïve Bayes

- Nonetheless, although this assumption is made for convenience, it often leads to quite decent results, especially when sample size is not large enough relative to the number of predictors to effectively estimate the joint distribution of the predictors within each class.

- In fact, since estimating a joint distribution requires such a huge amount of data, naïve Bayes is a good choice in a wide range of settings.

- The naïve Bayes assumption introduces some bias, but reduces variance, leading to a classifier that works quite well in practice as a result of the bias-variance trade-off.

Example: Naïve Bayes

Training sample

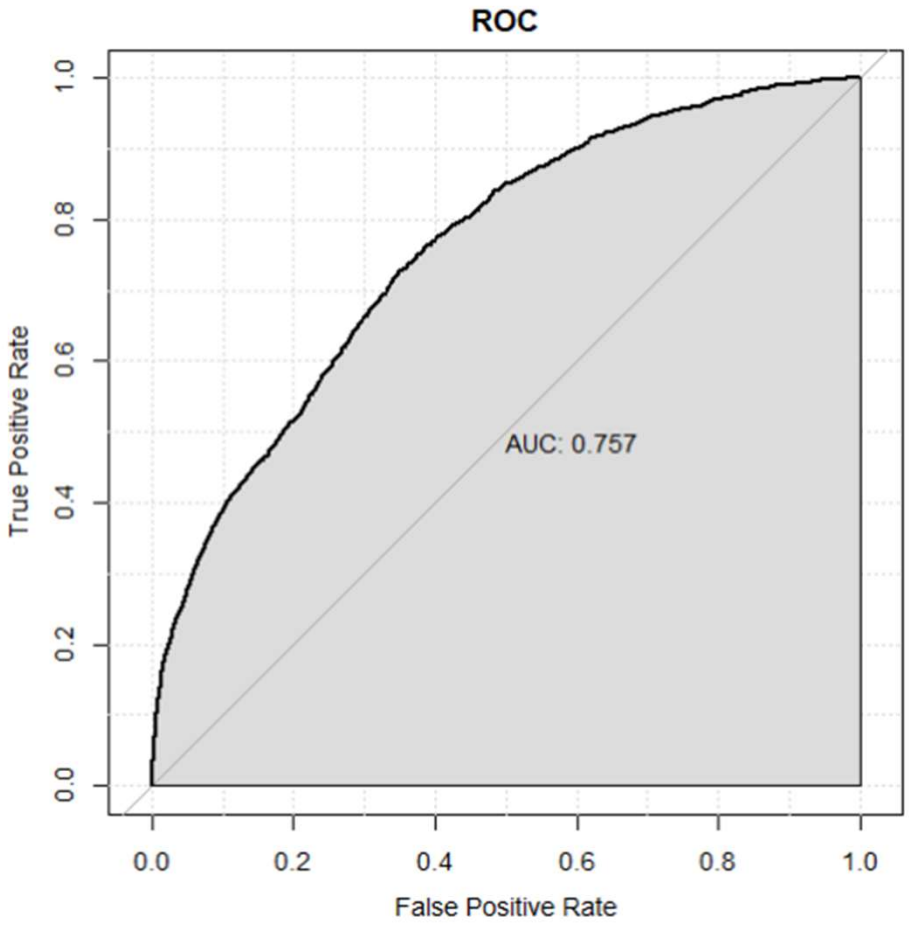
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 20968  1446
         1  1958   823

               Accuracy : 0.8649
                 95% CI : (0.8606, 0.8691)
    No Information Rate : 0.9099
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2517

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.36271
            Specificity : 0.91459
         Pos Pred Value : 0.29594
         Neg Pred Value : 0.93549
             Prevalence : 0.09006
         Detection Rate : 0.03267
   Detection Prevalence : 0.11038
      Balanced Accuracy : 0.63865

       'Positive' Class : 1
```

Test sample

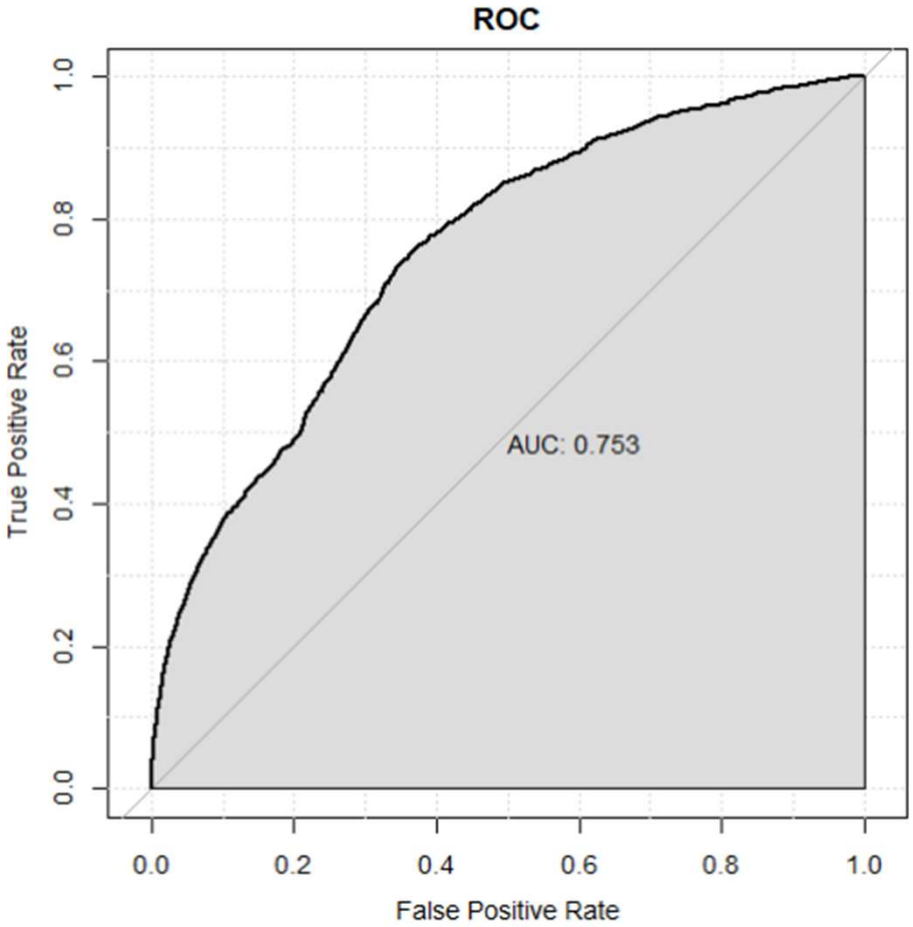
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 20706  1481
         1  1846   772

               Accuracy : 0.8659
                 95% CI : (0.8616, 0.8701)
    No Information Rate : 0.9092
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2431

 Mcnemar's Test P-Value : 2.778e-10

            Sensitivity : 0.34265
            Specificity : 0.91814
         Pos Pred Value : 0.29488
         Neg Pred Value : 0.93325
             Prevalence : 0.09083
         Detection Rate : 0.03112
   Detection Prevalence : 0.10554
      Balanced Accuracy : 0.63040

       'Positive' Class : 1
```

Example: Naïve Bayes

Training sample



Test sample



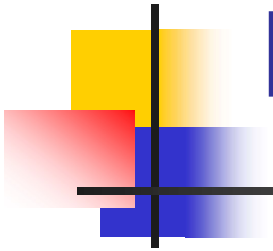
K-Nearest Neighbors (KNN)

- KNN is a completely non-parametric approach.
  - No assumptions are made about the shape of the decision boundary.
- Given a value for K and a prediction point x_0, KNN identifies the K points in the training data that are closest to x_0, represented by Q. It then estimates the conditional probability for class *k* as the fraction of points in Q whose response values equal to *k*:

$$P(Y = k|X = x_0) = \frac{1}{K}\sum_{x_i \in Q} I(y_i = k)$$

- KNN then classifies x_0 the class with the largest probability.

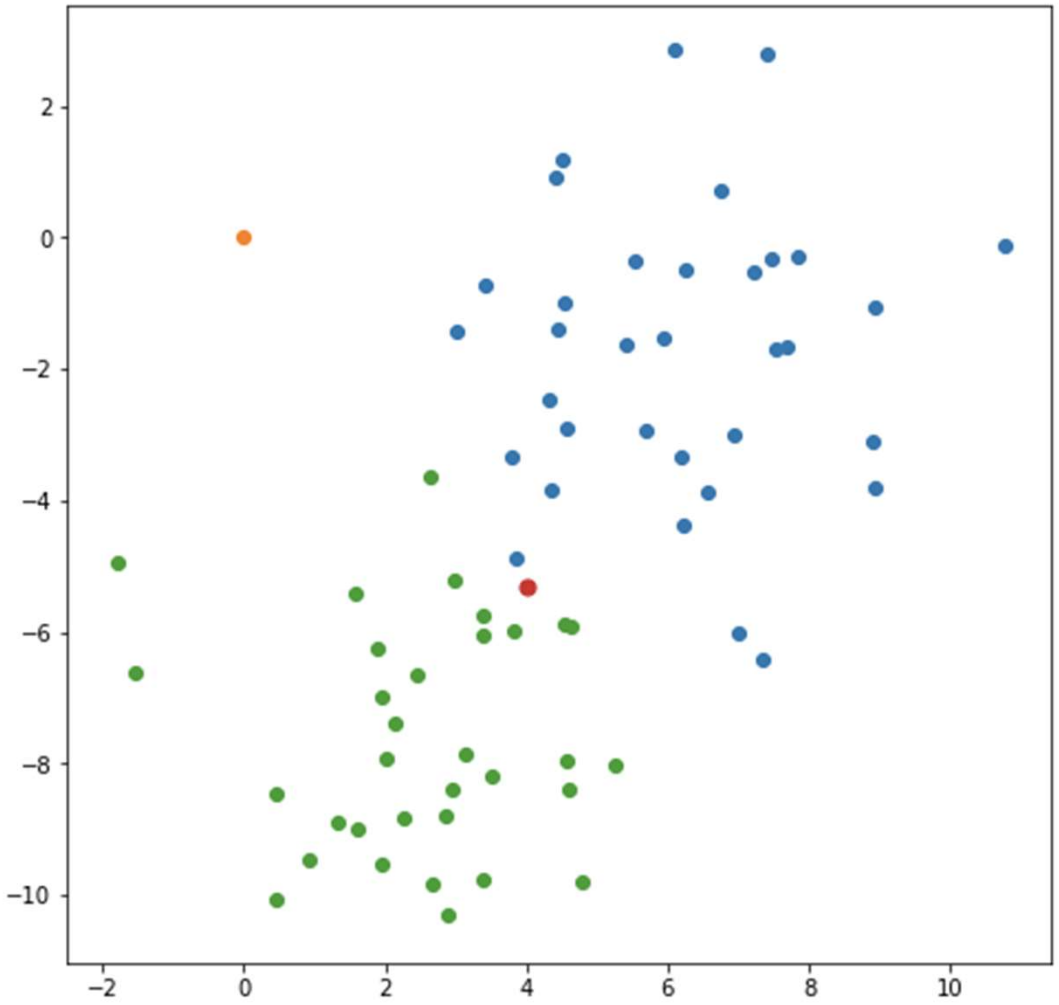
K-Nearest Neighbors (KNN)



- As in KNN regression, the choice of K has a drastic effect on the KNN classifier obtained.

- When K = 1, the decision boundary is overly flexible, resulting in low bias yet very high variance. As K increases, the method becomes less flexible and produces a decision boundary that is close to linear (so, high bias yet low variance).

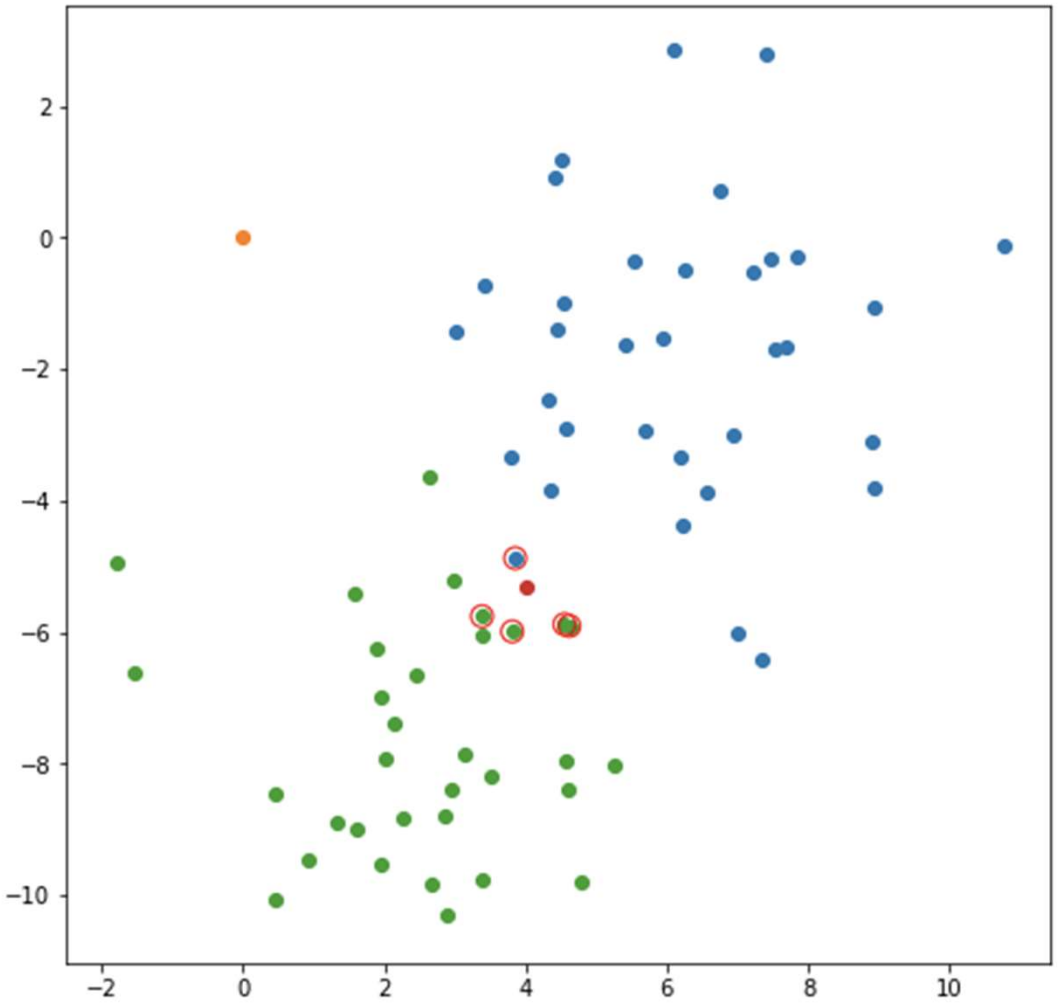
K-Nearest Neighbors (KNN)



https://towardsdatascience.com/the-basics-knn-for-classification-and-regression-c1e8a6c955

K-Nearest Neighbors (KNN)

K = 5



https://towardsdatascience.com/the-basics-knn-for-classification-and-regression-c1e8a6c955

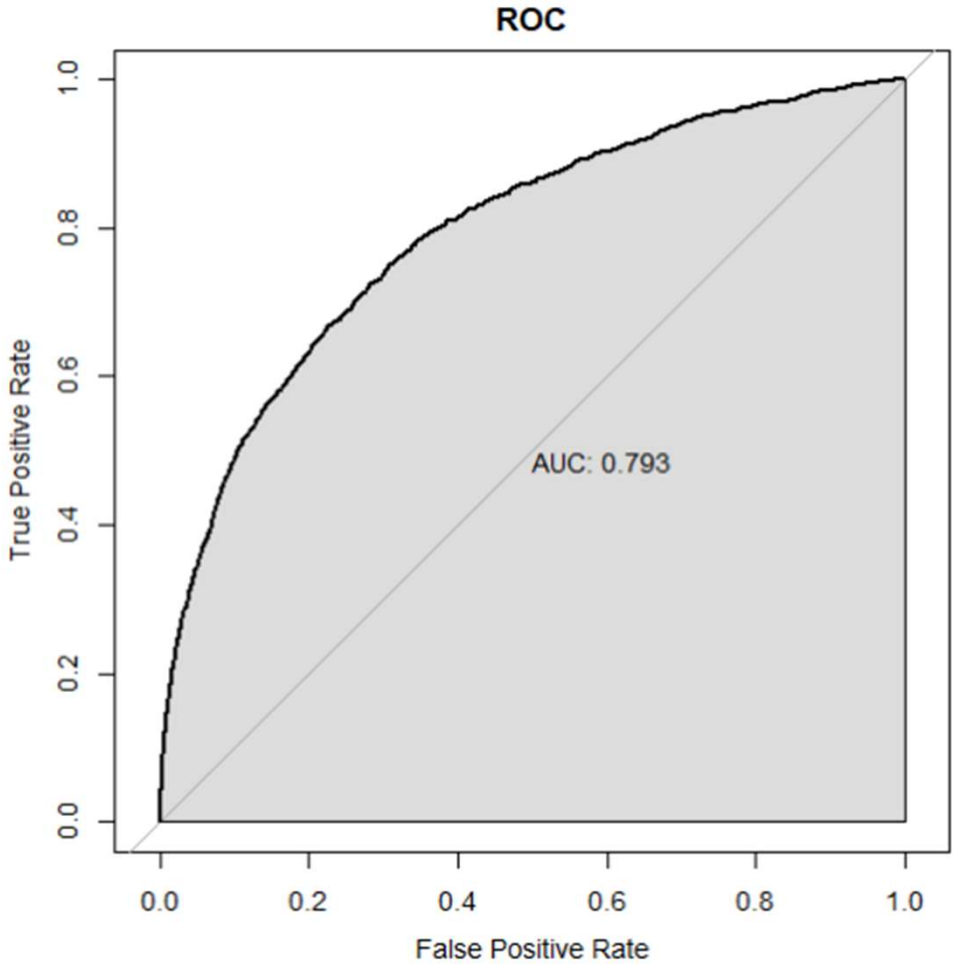
Example: K-Nearest Neighbors

Test sample

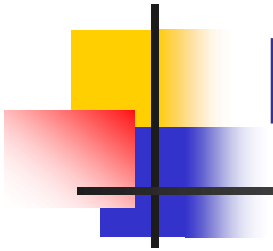
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 22541  2202
         1    11    51

               Accuracy : 0.9108
                 95% CI : (0.9072, 0.9143)
    No Information Rate : 0.9092
    P-Value [Acc > NIR] : 0.1916

                  Kappa : 0.0394

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.022636
            Specificity : 0.999512
         Pos Pred Value : 0.822581
         Neg Pred Value : 0.911005
             Prevalence : 0.090828
         Detection Rate : 0.002056
   Detection Prevalence : 0.002499
      Balanced Accuracy : 0.511074

       'Positive' Class : 1
```

Example: K-Nearest Neighbors

Test sample

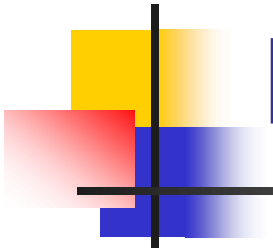


A Comparison of Classification Methods



- Logistic regression can outperform LDA if LDA's assumptions (a normal distribution with the same covariance matrix) are violated.
- LDA can provide some improvements (lower variance) over logistic regression if the assumptions are met.
- However, in practice, the LDA assumptions are never correct. So, logistic regression seems to be a safer, more robust bet over LDA, relying on fewer assumptions (Hastie et al., 2009, p. 128).

A Comparison of Classification Methods



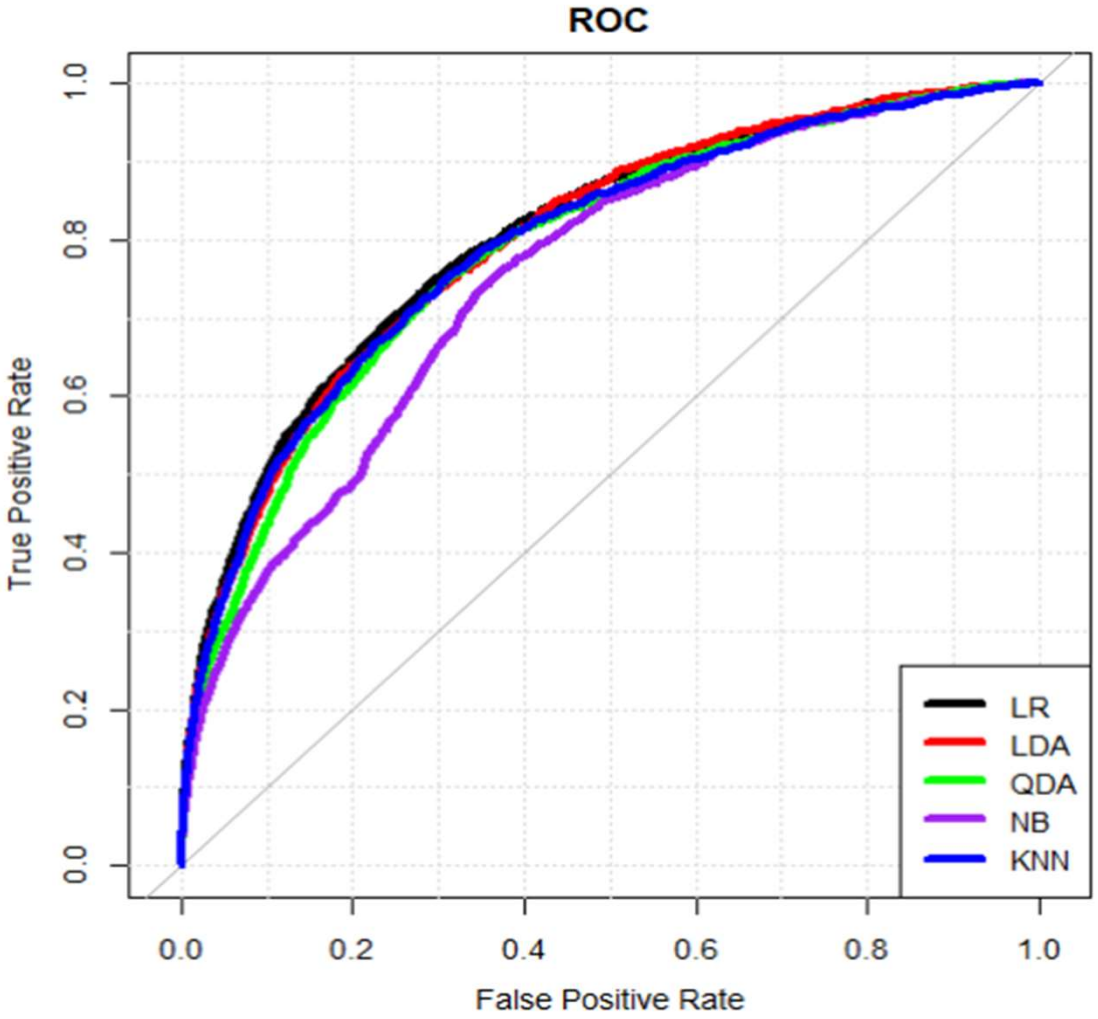
- For accurate classification, KNN requires a lot of observations relative to the number of predictors because it is non-parametric and tends to reduce the bias while incurring a lot of variance.

- KNN is expected to outperform LDA and logistic regression when the decision boundary is highly non-linear, provided that sample size is very large and the number of predictors is small.

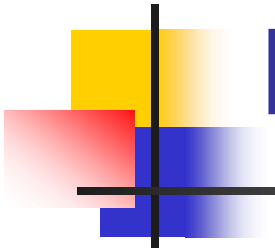
- When the decision boundary is non-linear but sample size is only modest or the number of predictors is not very small, QDA may be preferred to KNN.

Example: A Comparison of Classification Methods

Test sample




```
AUC_LR:    0.804052
AUC_LDA:   0.7987326
AUC_QDA:   0.7862619
AUC_NB:    0.7526996
AUC_KNN:   0.7934951
```



Lab: Classification Methods

- BBB_training.csv & BBB_test.csv
  - Logistic regression
  - LDA
  - QDA
  - Naïve Bayes
  - KNN