Introduction to Data Science

Last Update: 1 JANUARY 2021

Chapter 4
Predictive Analysis

Papangkorn Inkeaw, PhD

Department of Computer Science, Faculty of Science

Chiang Mai University

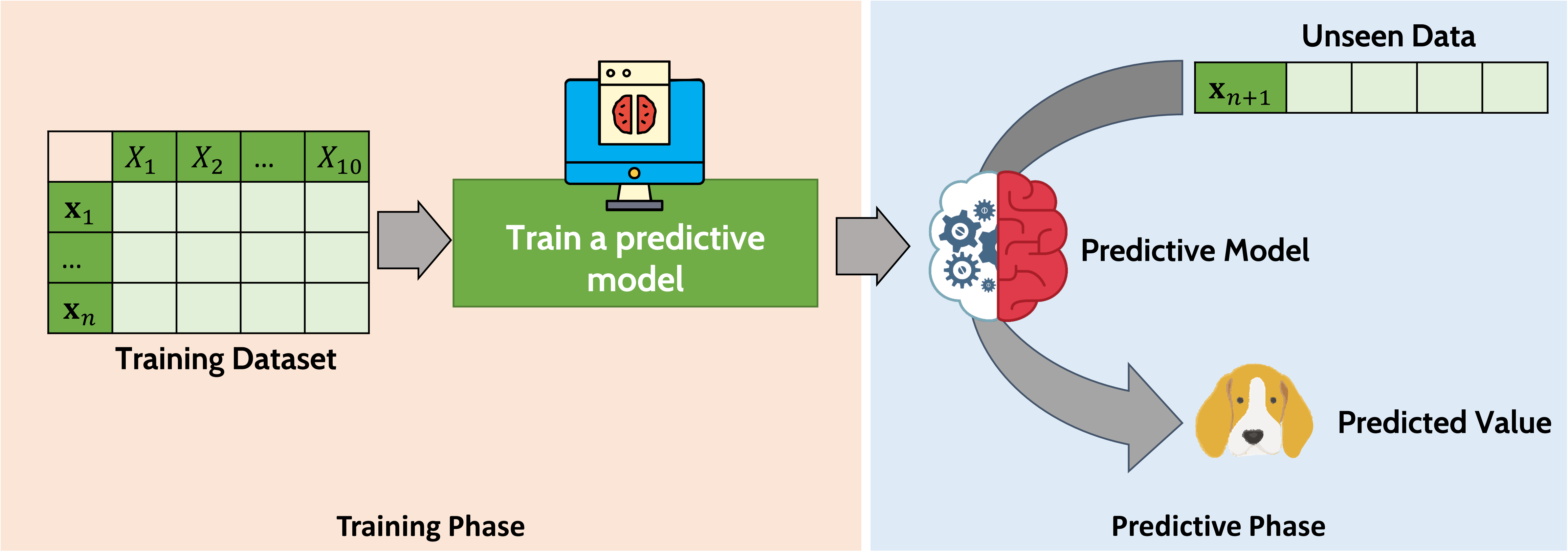
Outline

Predictive Analysis

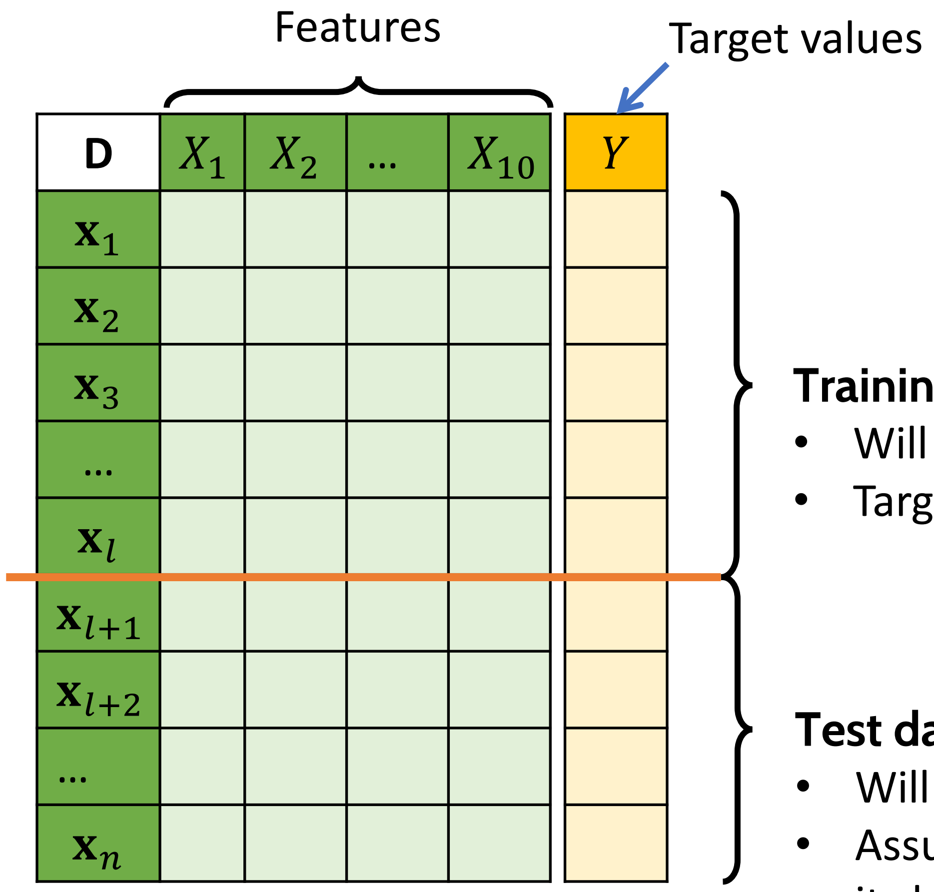
1. Predictive Analysis
   - Preparing Datasets
2. Classification Analysis
   - K-Nearest Neighbor
   - Decision Tree
   - Naïve Bayes
   - Artificial Neural Network
   - Classification Assessment
3. Regression Analysis
   - Linear Regression
   - Polynomial Regression
   - Artificial Neural Network
   - Regression Assessment
4. Time Series Analysis
   - Autoregressive Model
   - Moving Average Model
   - Autoregressive Integrated Moving Average
   - Moving Average Smoothing

Predictive Analysis

Analyze current and historical data to make predictions about future or otherwise unknown events.



Preparing Dataset
Predictive Analysis



To perform a predictive analysis:

- We should have two dataset: training and test datasets.
- The target value of each datapoint must be available.

Training dataset

- Will be used to <u>train</u> a predictive model.
- Target value of each data point must be available.

Test dataset

- Will be used to <u>evaluate</u> the predictive model
- Assume that target value of each data point is not known, but it should be available.

Classification Analysis

Features

Target class

D	X_1	X_2	…	X_{10}	Y
$\mathbf{x}_1$					
$\mathbf{x}_2$					
$\mathbf{x}_3$					
…					
$\mathbf{x}_l$					
$\mathbf{x}_{l+1}$					
$\mathbf{x}_{l+2}$					
…					
$\mathbf{x}_n$					

For classification analysis

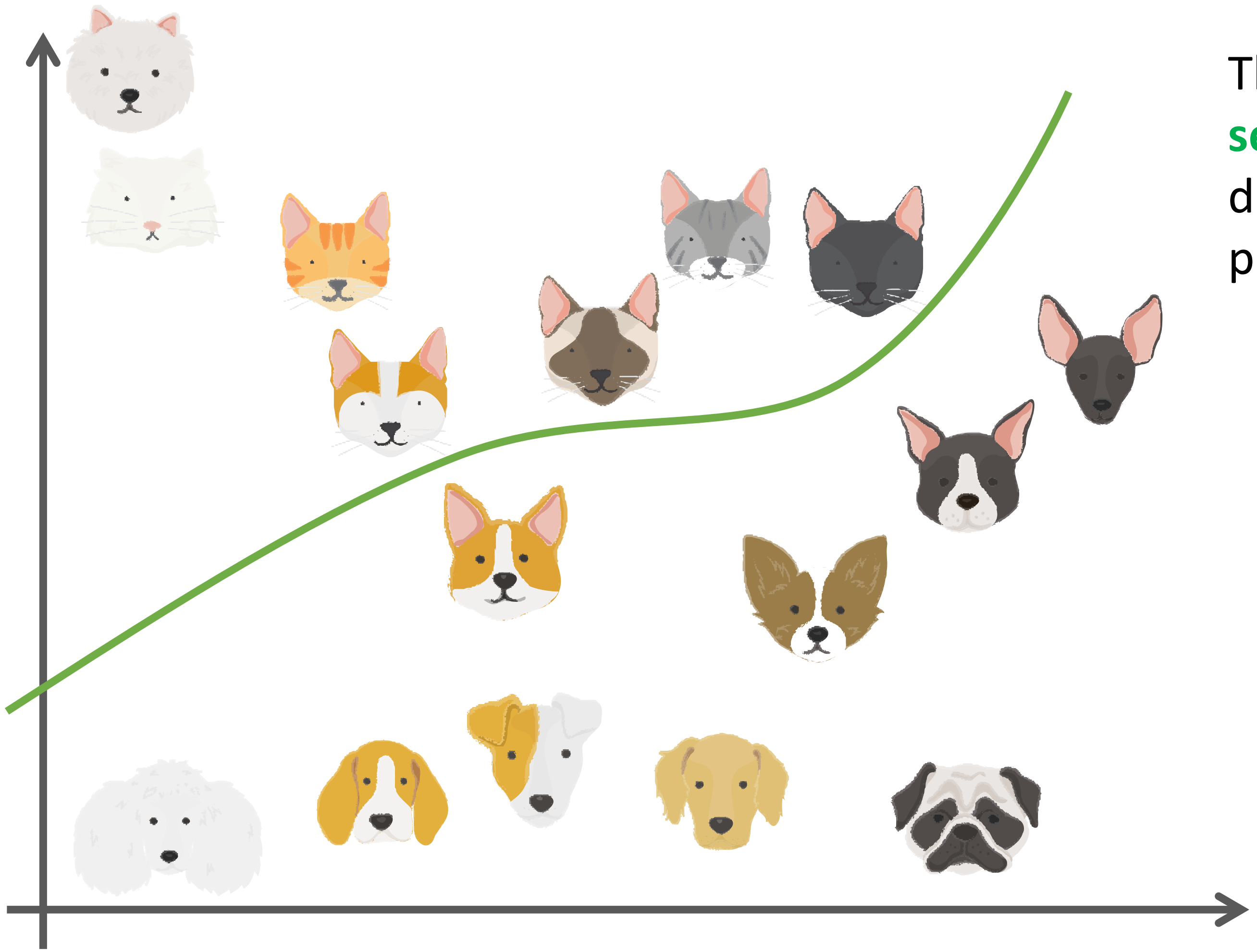
- The value we want to predict is **categorical data.**
- Known as **class**

Example

We know some characteristics of an animal, and we want to predict it is a cat or a dog.

cat or dog?

Classification Analysis



The task of classification is one of finding **separating lines** that separate classes of data from a training dataset as best as possible.

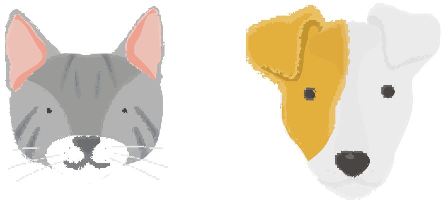
Classification Analysis

Types of Classification Problems

Number of Possible Classes

$= 2$

Binary Classification
- Want to distinguish only two categories (classes)



Cat or **Dog**?

> 2

Multi-class Classification
- Want to distinguish more than two categories (classes)

Bulldog or **Beagle** or **Shiba** ?

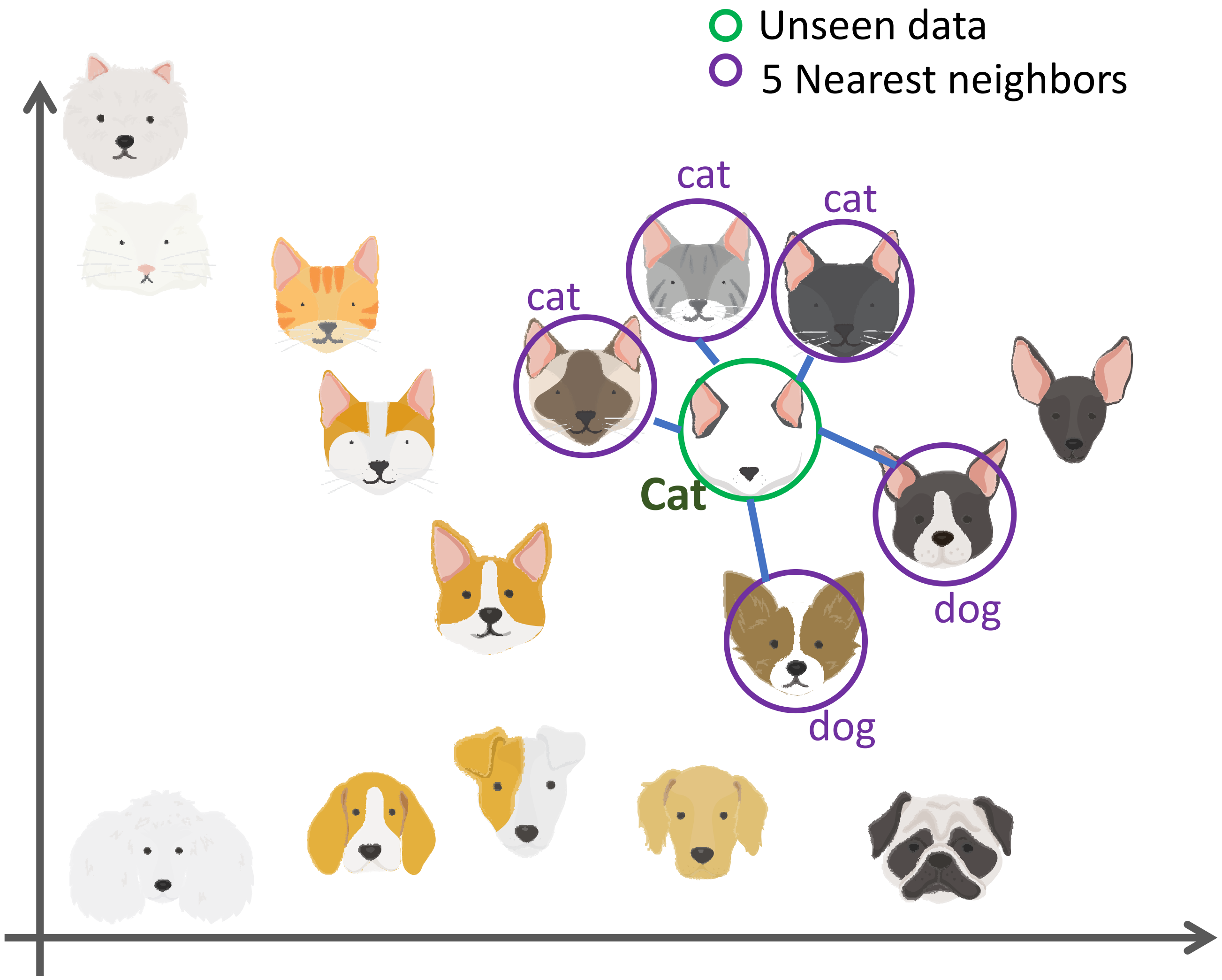
K-Nearest Neighbor

Classification Analysis

K-Nearest Neighbor classifier assigns the class label of an unseen data with the majority class labels of k neighbor data (in the training dataset)

How the k-nearest neighbor works

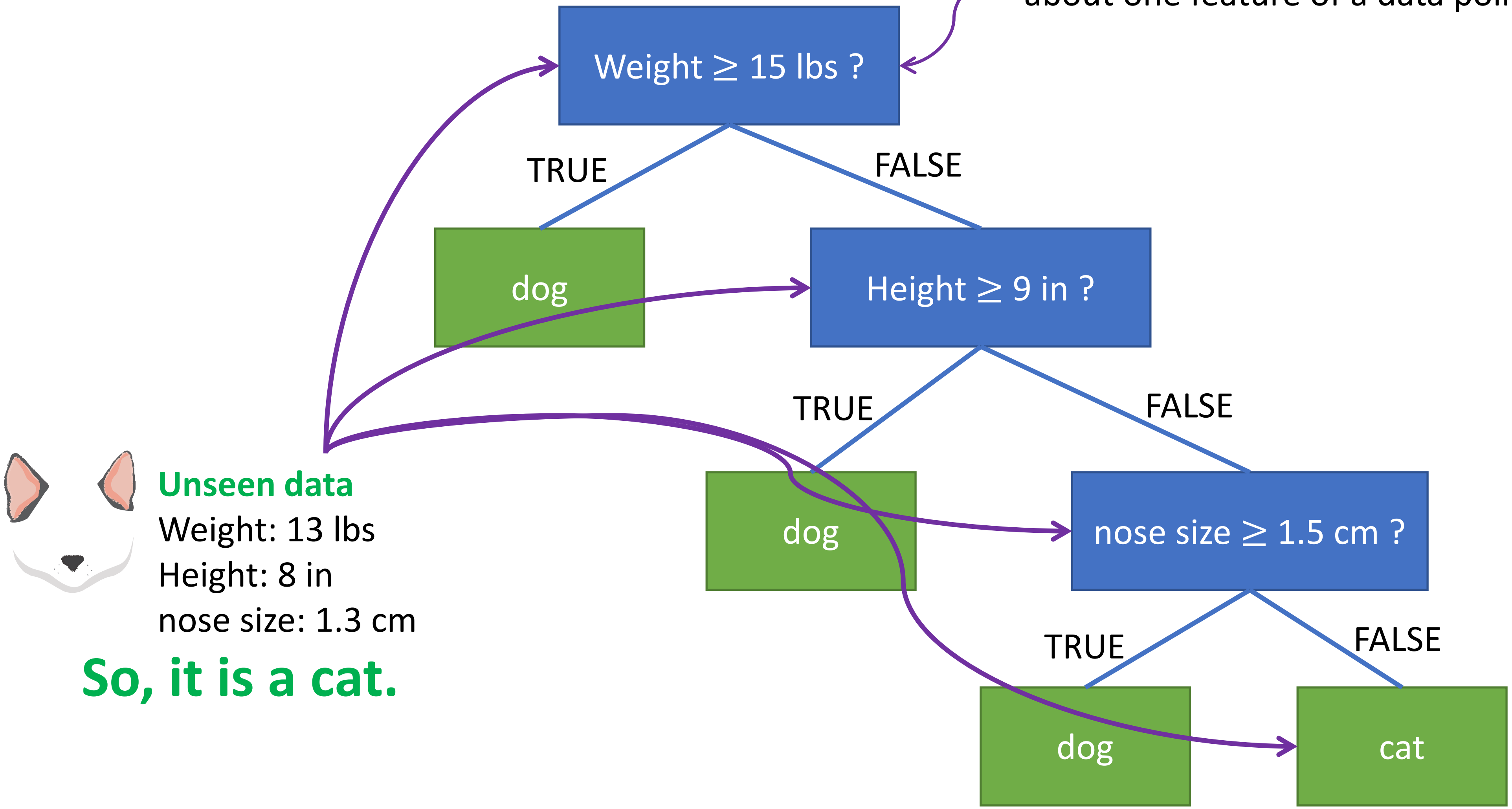
STEP 1: Calculate distances between an unseen data and training data
STEP 2: Find *k* nearest neighbor
STEP 3: Find majority class label
STEP 4: Assign the majority class label to the class label of the unseen data



Decision Tree

Classification Analysis

Every node in the tree asks a question about one feature of a data point.



So, it is a cat.

Decision Tree

Classification Analysis

Construct a decision tree

STEP 1: Given a training data D, find the single feature (and cutoff for that feature, if it's numerical) that best partitions your data into classes.

STEP 2: This single best feature/cutoff becomes the root of your decision tree.

STEP 3: Partition *D* up according to the root node.

STEP 4: Recursively train each of the child nodes on its partition of the data until all of the data points in the partition have the same label.

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog
$\mathbf{x}_2$	50	40	3	Dog
$\mathbf{x}_3$	8	9	1.3	Cat
$\mathbf{x}_4$	15	12	2.5	Dog
$\mathbf{x}_5$	9	9.8	1.4	Cat

Weight $\geq$ 15 lbs ?

TRUE

D	Weight	Height	Nose size	Label
$\mathbf{x}_2$	50	40	3	Dog
$\mathbf{x}_4$	15	12	2.5	Dog

FALSE

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog
$\mathbf{x}_3$	8	9	1.3	Cat
$\mathbf{x}_5$	9	9.8	1.4	Cat

Decision Tree

Classification Analysis

Weight $\geq$ 15 lbs ?

FALSE

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog
$\mathbf{x}_3$	8	9	1.3	Cat
$\mathbf{x}_5$	9	8.5	1.4	Cat

Height $\geq$ 9 in ?

TRUE

D	Weight	Height	Nose size	Label
$\mathbf{x}_3$	8	9	1.3	Cat

FALSE

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog
$\mathbf{x}_5$	9	8.5	1.4	Cat

Construct a decision tree

STEP 1: Given a training data D, find the single feature (and cutoff for that feature, if it's numerical) that best partitions your data into classes.

STEP 2: This single best feature/cutoff becomes the root of your decision tree.

STEP 3: Partition *D* up according to the root node.

STEP 4: Recursively train each of the child nodes on its partition of the data until all of the data points in the partition have the same label.

Decision Tree

Classification Analysis

Construct a decision tree

STEP 1: Given a training data D, find the single feature (and cutoff for that feature, if it's numerical) that best partitions your data into classes.
STEP 2: This single best feature/cutoff becomes the root of your decision tree.
STEP 3: Partition *D* up according to the root node.
STEP 4: Recursively train each of the child nodes on its partition of the data until all of the data points in the partition have the same label.

Height $\geq$ 9 in ?

FALSE

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog
$\mathbf{x}_5$	9	8.5	1.4	Cat

nose size $\geq$ 1.5 cm ?

TRUE

D	Weight	Height	Nose size	Label
$\mathbf{x}_1$	8	8	1.6	Dog

FALSE

D	Weight	Height	Nose size	Label
$\mathbf{x}_5$	9	8.5	1.4	Cat

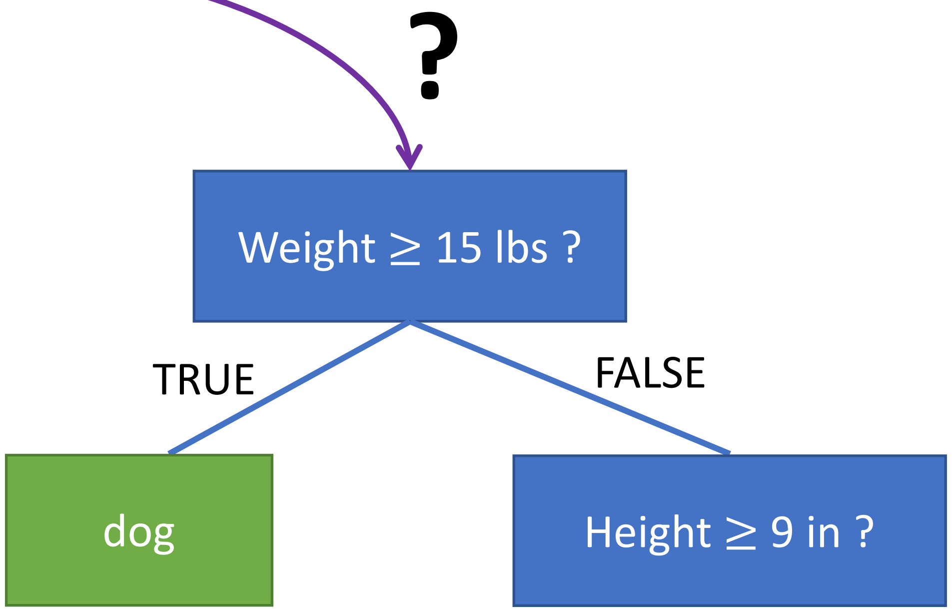
Decision Tree

Classification Analysis

How to determine the best feature and cutoff

The most common ones are:

- Information gain
- Gini impurity.



You can find more details in:

- Zaki, M., & Meira, W. (2014). Data mining and analysis : Fundamental concepts and algorithms. New York: Cambridge University Press.
- https://en.wikipedia.org/wiki/Decision_tree_learning

Naïve Bayes

Classification Analysis

Bayes Theorem:

The *prior*, the initial degree of belief in ***A***.

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)}$$

Probability of ***A*** happening, given that ***B*** has occurred

The likelihood of event ***B*** occurring given that ***A*** is true.

Thomas Bayes
1701-1761
Source:
https://en.wikipedia.org/wiki/Thomas_Bayes#/media/File:Thomas_Bayes.gif

Naïve Bayes
Classification Analysis

Classify whether the day is suitable for playing golf, given the features of the day.

Bayes theorem can be rewritten as:

$$P(y|\mathbf{x}) = \frac{P(y)P(\mathbf{x}|y)}{P(\mathbf{x})}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes
Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y) \prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

We want to classify

$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y) \prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Play golf} = \text{No}) = \frac{5}{14}$$

$$P(\text{Play golf} = \text{Yes}) = \frac{9}{14}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y) \prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Outlook} = \text{Sunny}|\text{Play golf} = \text{No}) = \frac{2}{5}$$

$$P(\text{Outlook} = \text{Sunny}|\text{Play golf} = \text{Yes}) = \frac{3}{9}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y)\prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Temperature} = \text{Hot}|\text{Play golf} = \text{No}) = \frac{2}{5}$$

$$P(\text{Temperature} = \text{Hot}|\text{Play golf} = \text{Yes}) = \frac{2}{9}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y)\prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Humidity} = \text{Normal}|\text{Play golf} = \text{No}) = \frac{1}{5}$$

$$P(\text{Humidity} = \text{Normal}|\text{Play golf} = \text{Yes}) = \frac{6}{9}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y)\prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Windy} = \text{True}|\text{Play golf} = \text{No}) = \frac{3}{5}$$

$$P(\text{Windy} = \text{True}|\text{Play golf} = \text{Yes}) = \frac{3}{9}$$

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Naïve Bayes

Classification Analysis

How the Naïve Bayes works

STEP 1: Calculate $P(y)$ for all possible value of y from the training dataset.

STEP 2: Calculate $P(\mathbf{x}|y) = \prod_{i=1}^{p} P(x_i|y)$ for all possible value of y from the training dataset.

STEP 3: Calculate $P(y|\mathbf{x}) = P(y) \prod_{i=1}^{p} P(x_i|y)$

STEP 4: Assign y that reach the highest $P(y|\mathbf{x})$ to the class label of $\mathbf{x}$

$$P(\text{Play golf} = \text{No}|\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$
$$= \frac{5}{14} \times \frac{2}{5} \times \frac{2}{5} \times \frac{1}{5} \times \frac{3}{5} = 0.0069$$

$$P(\text{Play golf} = \text{Yes}|\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$
$$= \frac{9}{14} \times \frac{3}{9} \times \frac{2}{9} \times \frac{6}{9} \times \frac{3}{9} = \mathbf{0.0106}$$

So, it is suitable to **play golf** given the conditions (Outlook = Sunny, Temperature = Hot, Humidity = Normal and Windy = True).

We want to classify

$$\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Normal}, \text{True})$$

$$P(\text{Play golf} = \text{No}) = \frac{5}{14}$$

$$P(\text{Play golf} = \text{Yes}) = \frac{9}{14}$$

$$P(\text{Outlook} = \text{Sunny}|\text{Play golf} = \text{No}) = \frac{2}{5}$$

$$P(\text{Outlook} = \text{Sunny}|\text{Play golf} = \text{Yes}) = \frac{3}{9}$$

$$P(\text{Temperature} = \text{Hot}|\text{Play golf} = \text{No}) = \frac{2}{5}$$

$$P(\text{Temperature} = \text{Hot}|\text{Play golf} = \text{Yes}) = \frac{2}{9}$$

$$P(\text{Humidity} = \text{Normal}|\text{Play golf} = \text{No}) = \frac{1}{5}$$

$$P(\text{Humidity} = \text{Normal}|\text{Play golf} = \text{Yes}) = \frac{6}{9}$$

$$P(\text{Windy} = \text{True}|\text{Play golf} = \text{No}) = \frac{3}{5}$$

$$P(\text{Windy} = \text{True}|\text{Play golf} = \text{Yes}) = \frac{3}{9}$$

Naïve Bayes

Classification Analysis

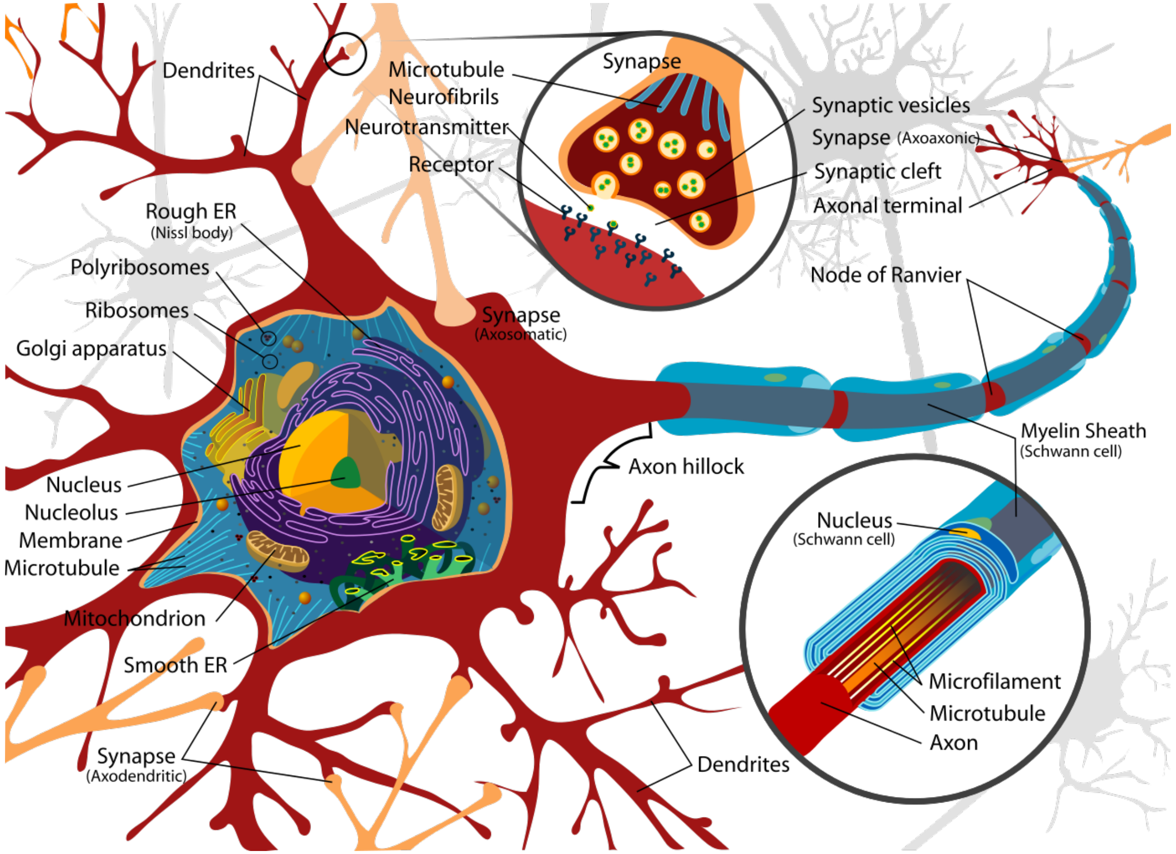
Quiz:

It is suitable to play golf or not given the conditions (Outlook = Rainy, Temperature = Mild, Humidity = Normal and Windy = False).

D	Outlook	Temperature	Humidity	Windy	Play golf
$\mathbf{x}_1$	Rainy	Hot	High	False	No
$\mathbf{x}_2$	Rainy	Hot	High	True	No
$\mathbf{x}_3$	Overcast	Hot	High	False	Yes
$\mathbf{x}_4$	Sunny	Mild	High	False	Yes
$\mathbf{x}_5$	Sunny	Cool	Normal	False	Yes
$\mathbf{x}_6$	Sunny	Cool	Normal	True	No
$\mathbf{x}_7$	Overcast	Cool	Normal	True	Yes
$\mathbf{x}_8$	Rainy	Mild	High	False	No
$\mathbf{x}_9$	Rainy	Cool	Normal	False	Yes
$\mathbf{x}_{10}$	Sunny	Mild	Normal	False	Yes
$\mathbf{x}_{11}$	Rainy	Mild	Normal	True	Yes
$\mathbf{x}_{12}$	Overcast	Mild	High	Ture	Yes
$\mathbf{x}_{13}$	Overcast	Hot	Normal	False	Yes
$\mathbf{x}_{14}$	Sunny	Mild	High	True	No

Artificial Neural Network

Classification Analysis



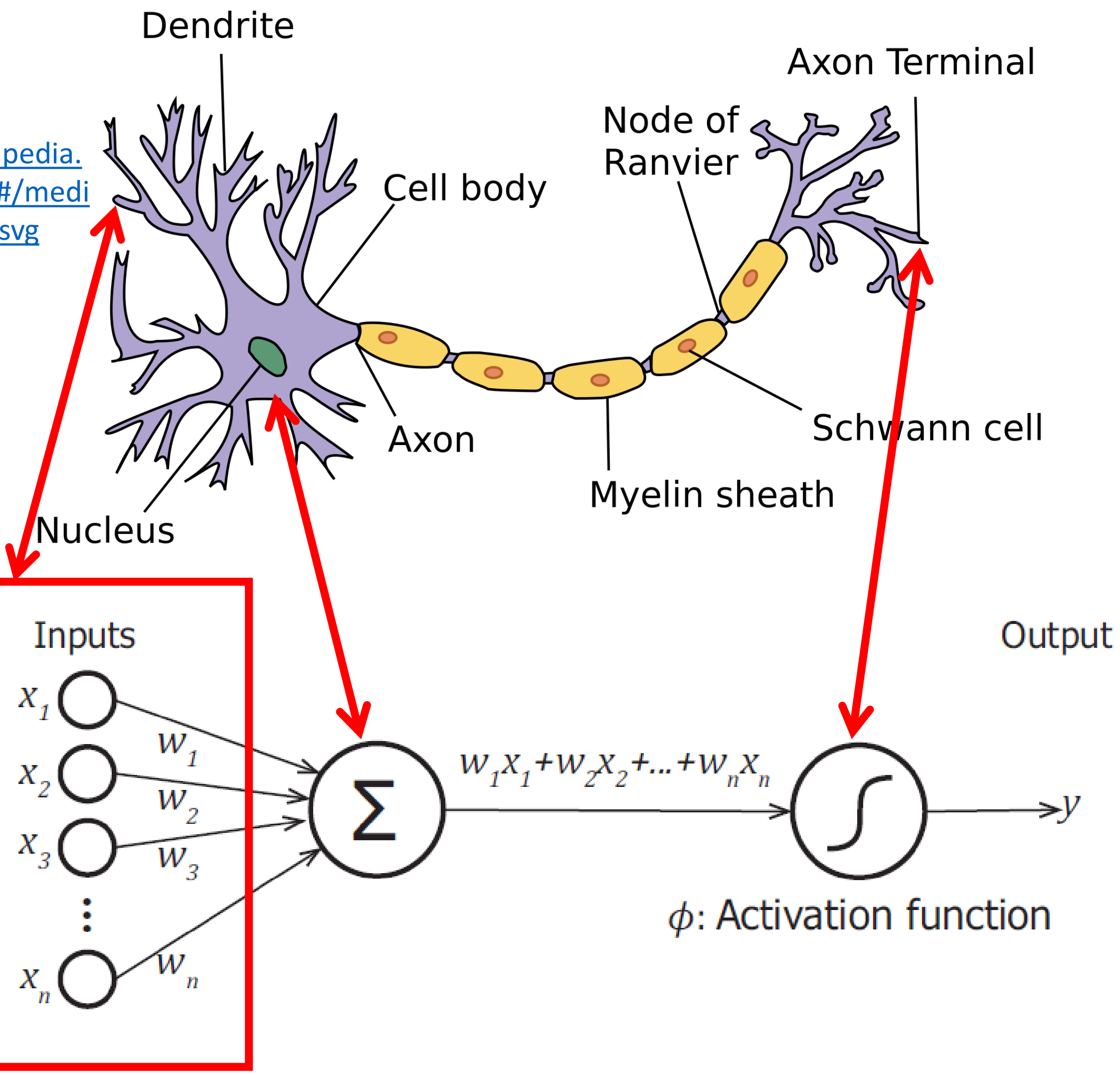
Source:
https://en.wikipedia.org/wiki/Neuron#/media/File:Complete_neuron_cell_diagram_en.svg

- **Artificial Neural Network (ANN)** is a computational model inspired structurally and functionally in biological neural networks.
- **ANN** is <u>a web of neuron nodes</u>.

Artificial Neural Network

Classification Analysis

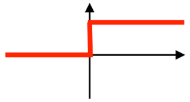
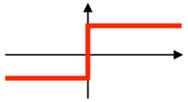
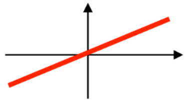
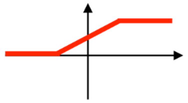
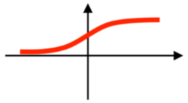
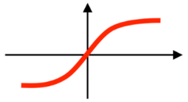
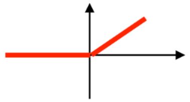
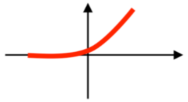
Source:
https://en.wikipedia.org/wiki/Axon#/media/File:Neuron.svg



For a neuron, the output y is produced by:

$$y = \phi\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

Artificial Neural Network
Classification Analysis

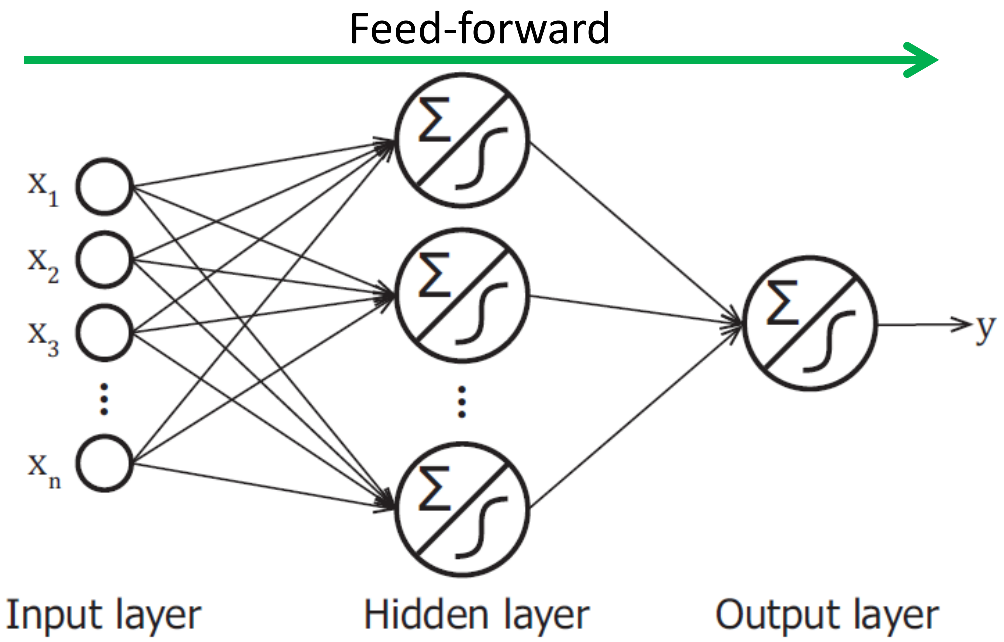
Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus Copyright © Sebastian Raschka 2016 (http://sebastianraschka.com)	$\phi(z) = \ln(1 + e^z)$	Multi-layer Neural Networks	

Source:
https://sebastianraschka.com/

Artificial Neural Network
Classification Analysis

A well-known structure of ANN is the multilayer perceptron:

- Neurons are arranged in a layer
- Output of one layer serving as the input to the next layer and possibly other layers.

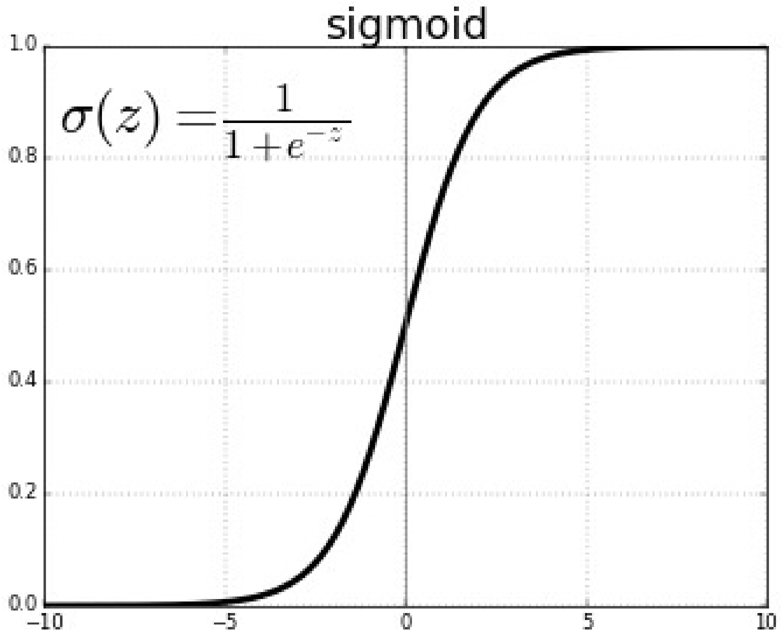


Artificial Neural Network

Classification Analysis

For Classification,

Neurons in output layer applies *sigmoid* function as the activation function.

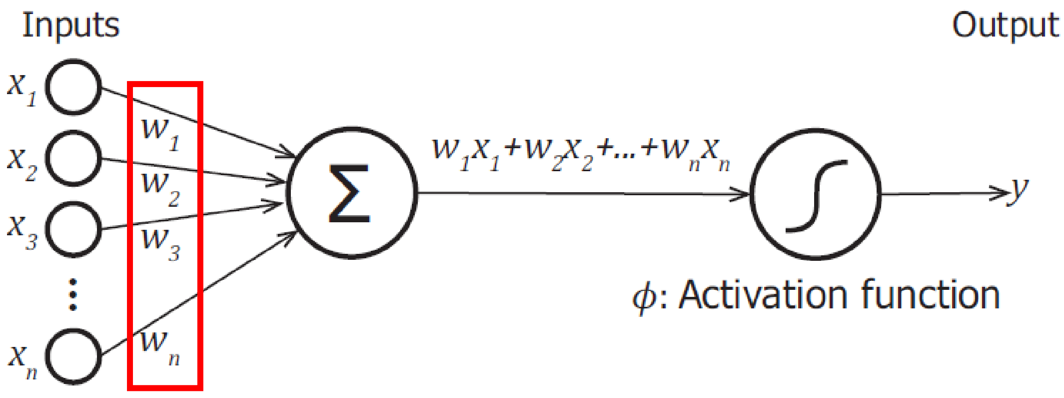


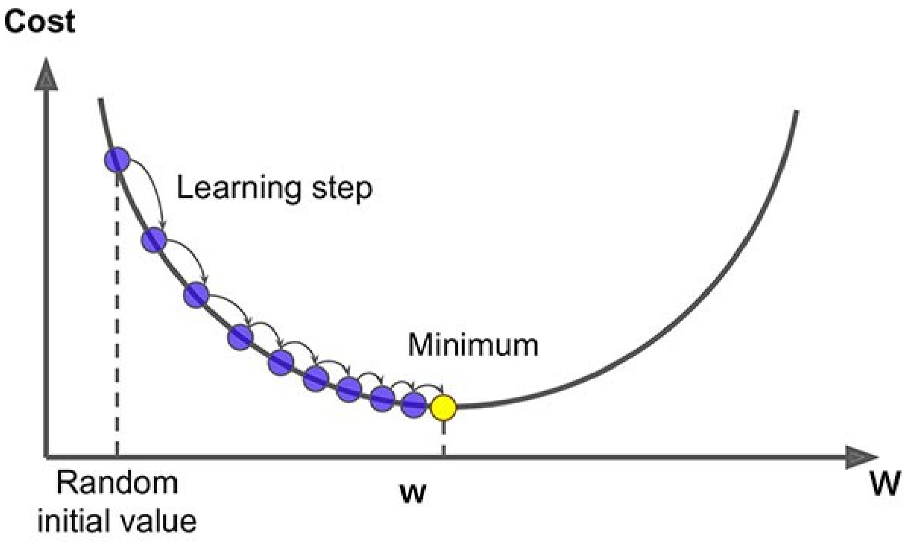
Artificial Neural Network

Classification Analysis

Using an ANN

- Design the structure of ANN
- Train the ANN using a training dataset
    - Find optimal values of weight w_i that minimize an objection function e.g. square error
    - A well-known approach to training an ANN is gradient descent

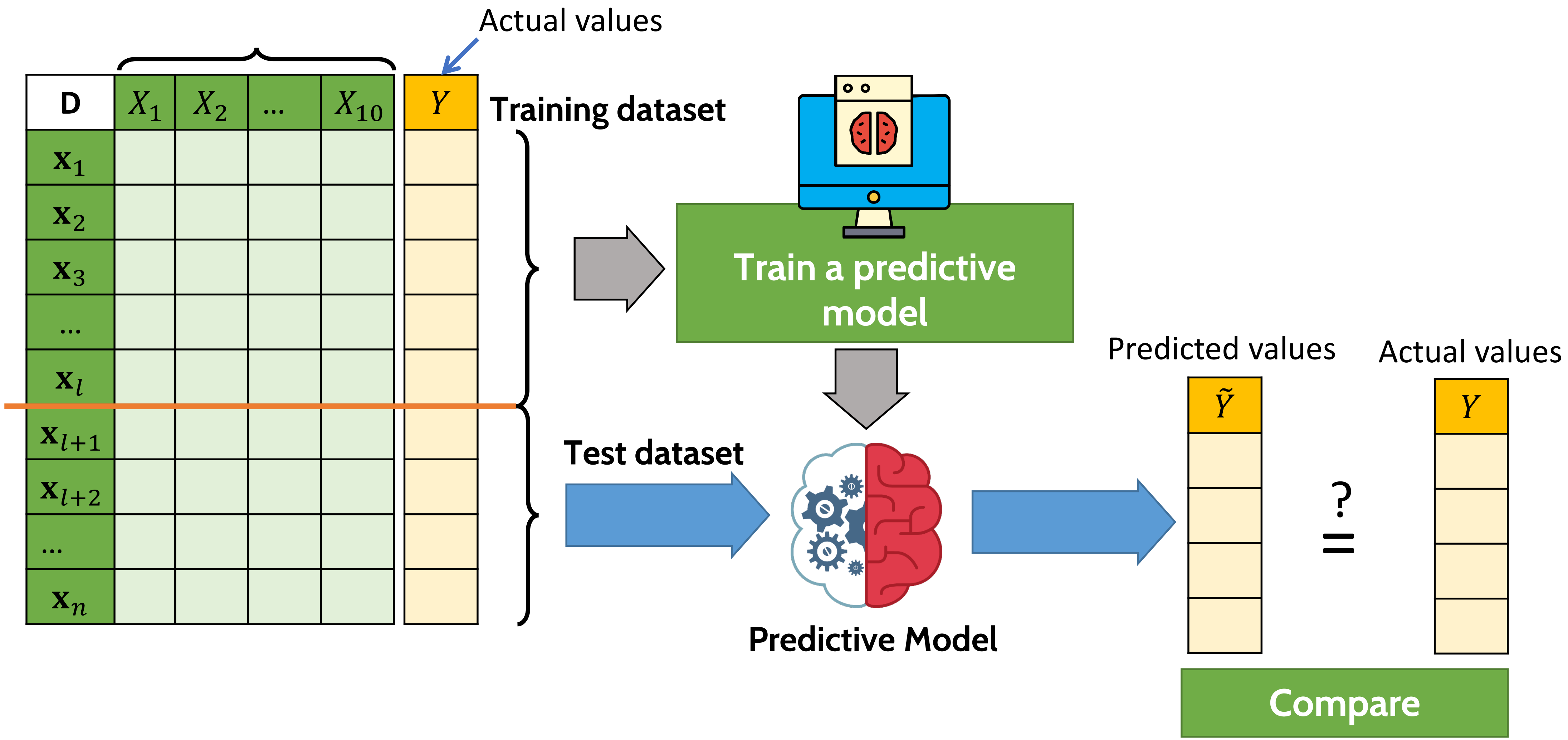




Source: https://mc.ai/an-introduction-to-gradient-descent-2/

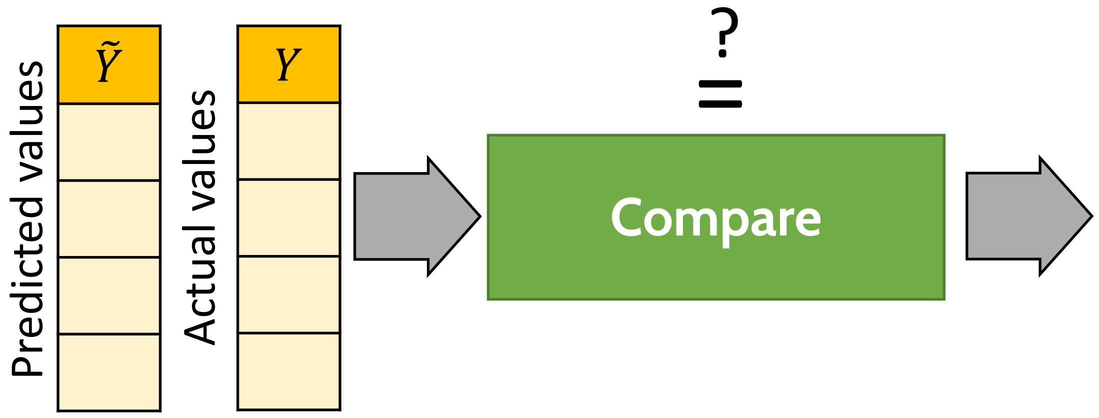
Classification Assessment

Classification Analysis



Classification Assessment

Classification Analysis



Actual values

Predicted values	Positive	Negative
Positive	TP	FP
Negative	FN	TN

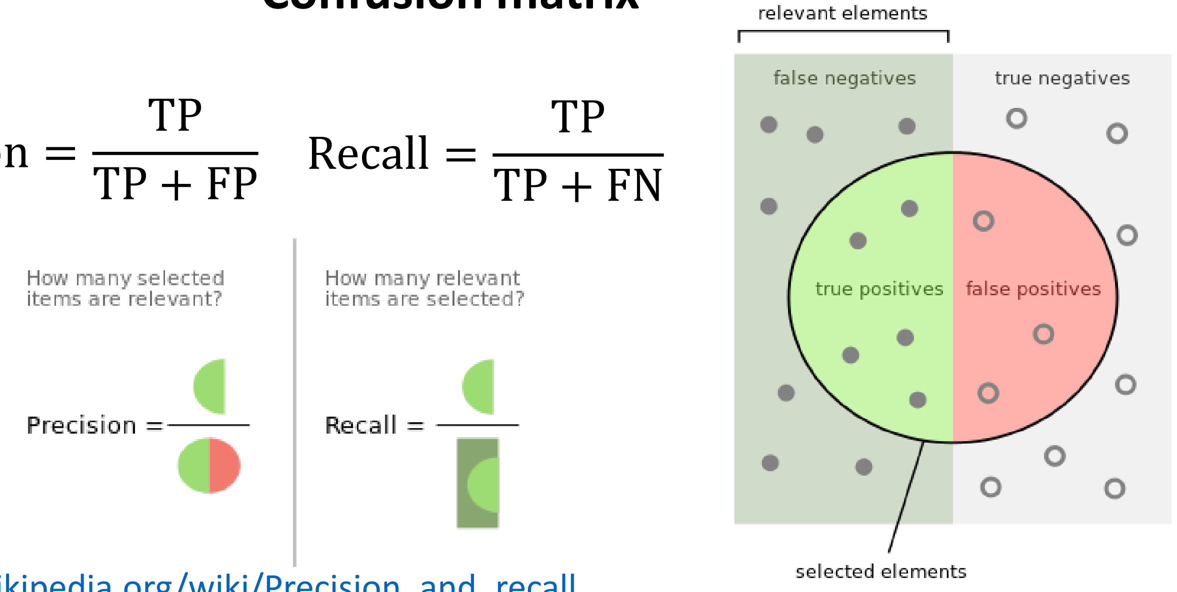
true positives (TP)
true negatives (TN)
false positives (FP)
false negatives (FN)

Confusion matrix

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{\text{Total}}$$

$$\text{Misclassification Rate} = \frac{(\text{FP} + \text{FN})}{\text{Total}} = 1 - \text{Accuracy}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$



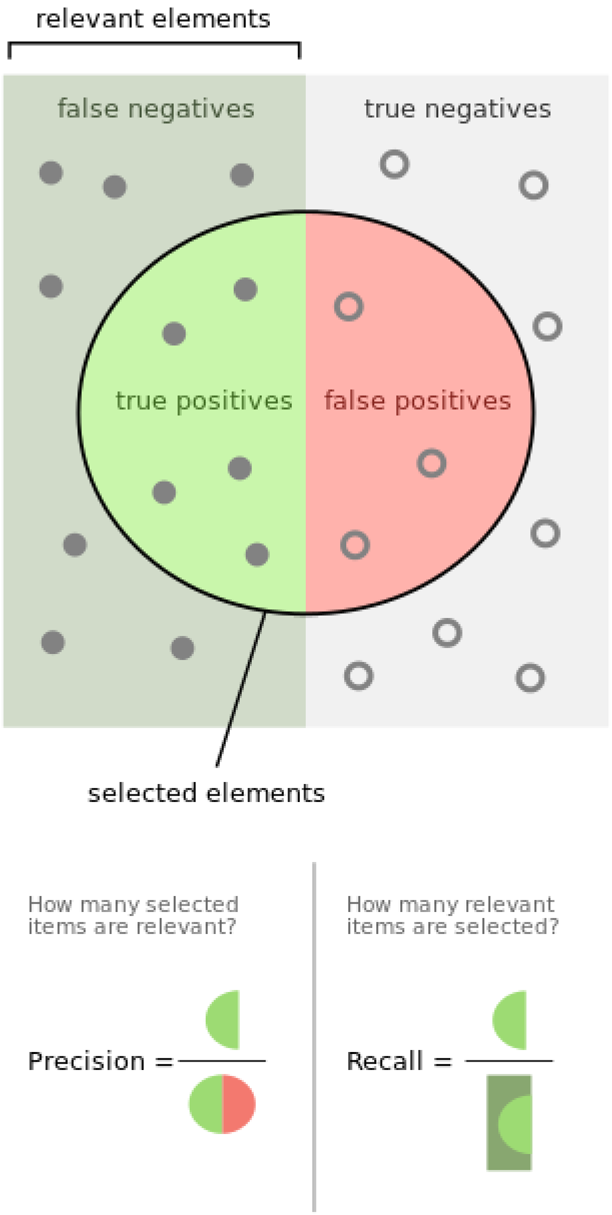
Source:
https://en.wikipedia.org/wiki/Precision_and_recall

Classification Assessment

Classification Analysis

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$



Classification Assessment

Classification Analysis

Example

Actual values (columns); Predicted values (rows)

	setosa	versicolor	virginica
setosa	**10**	2	4
versicolor	1	**16**	1
virginica	0	2	**9**

$$\text{Accuracy} = \frac{(10+16+9)}{45} = \frac{35}{45} = 0.78$$

$$\text{Misclassification Rate} = 1 - 0.78 = 0.22$$

$$\text{Recall}_{\text{setosa}} = \frac{10}{10+1+0} = \frac{10}{11} = 0.91$$

$$\text{Precision}_{\text{setosa}} = \frac{10}{10+2+4} = \frac{10}{16} = 0.625$$

$$\text{Recall}_{\text{versicolor}} = \frac{16}{2+16+2} = \frac{16}{20} = 0.8$$

$$\text{Precision}_{\text{versicolor}} = \frac{16}{1+16+1} = \frac{16}{18} = 0.89$$

$$\text{Recall}_{\text{virginica}} = ?$$

$$\text{Precision}_{\text{virginica}} = ?$$

Classification Assessment

Classification Analysis

Example

Actual values

Predicted values	Cat	Dog
Cat	**5**	2
Dog	3	**3**

$$\text{Accuracy} = \frac{(5+3)}{13} = \frac{8}{13} = 0.62$$

$$\text{Misclassification Rate} = \frac{(2+3)}{13} = \frac{5}{13} = 0.38$$

$$\text{Recall} = \frac{5}{5+3} = \frac{5}{8} = 0.625$$

$$\text{Precision} = \frac{5}{5+2} = \frac{5}{7} = 0.714$$

Classification Assessment

Classification Analysis

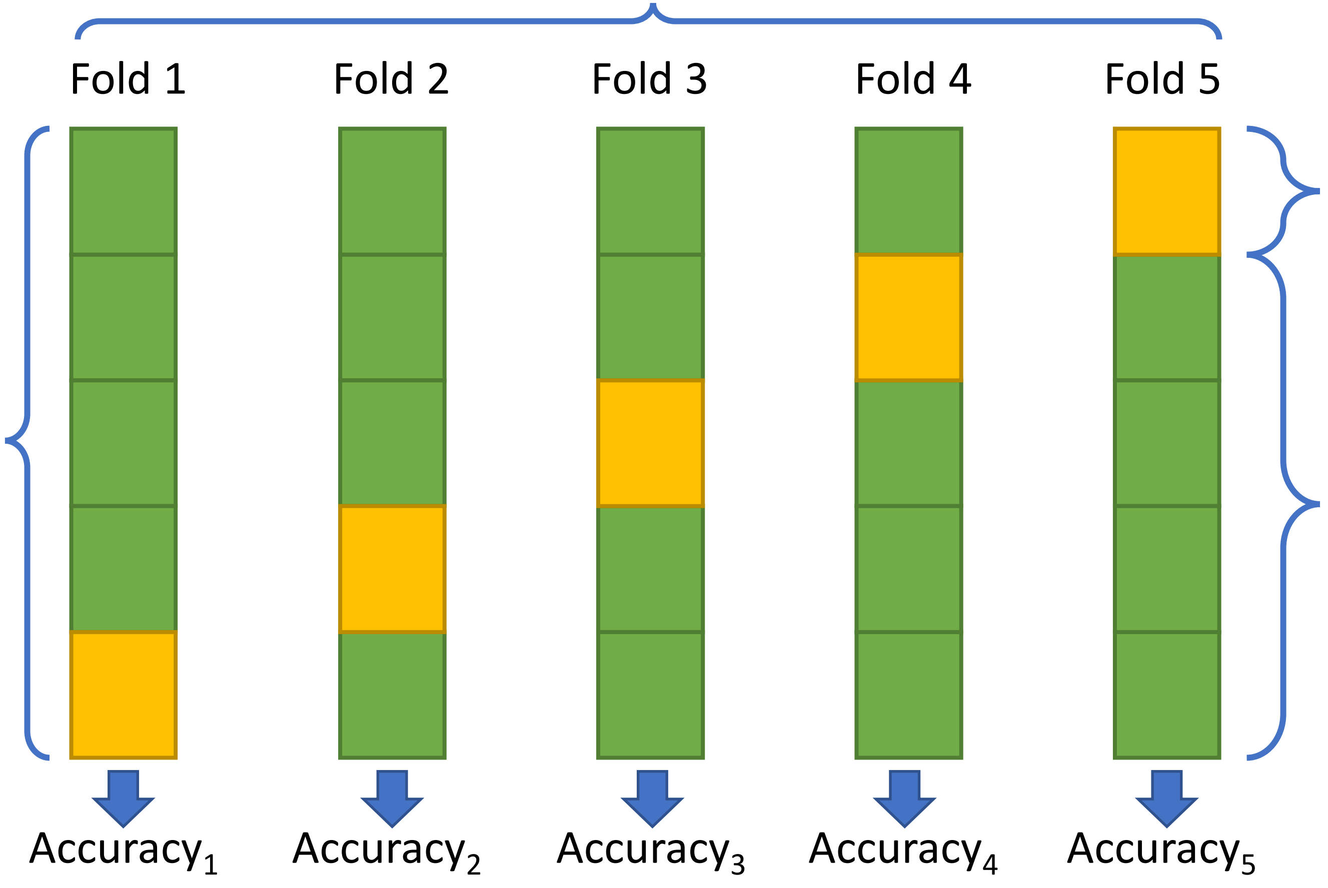
Cross-validation

Perform k times that each subset is selected to be the test dataset at one time

Training dataset

Test dataset

Divide the dataset into k (e.g., k = 5) subset



Use a subset as test dataset

Use the remaining sets as training dataset

$$Accuracy_{average} = \frac{1}{k}\sum_{i=1}^{k} Accuracy_i$$

Regression Analysis

Independent variable

Dependent variable

D	X_1	X_2	…	X_{10}	Y
$\mathbf{x}_1$					
$\mathbf{x}_2$					
$\mathbf{x}_3$					
…					
$\mathbf{x}_l$					
$\mathbf{x}_{l+1}$					
$\mathbf{x}_{l+2}$					
…					
$\mathbf{x}_n$					

For regression analysis

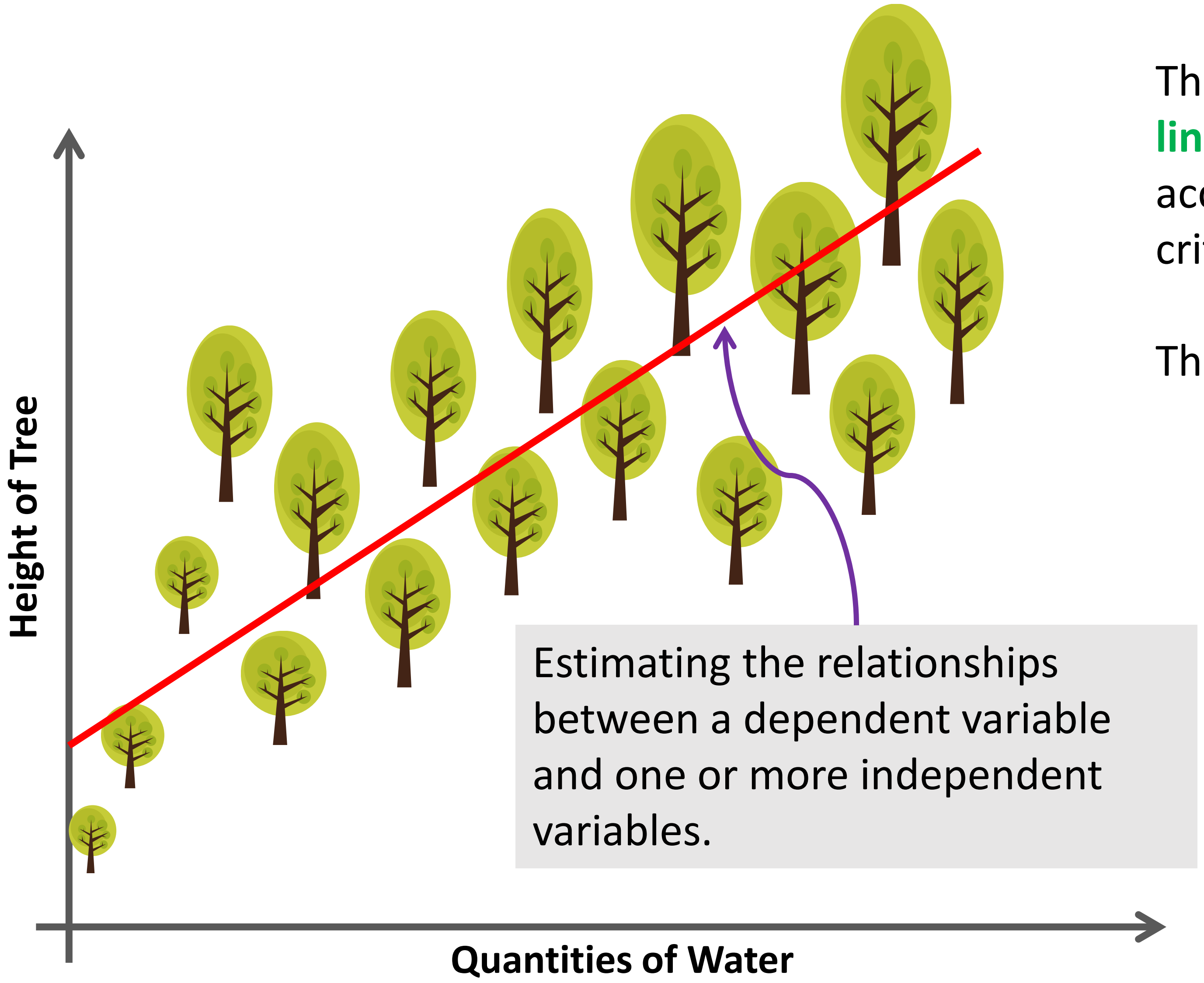
- The value we want to predict is **numeric data.**
- Known as **Dependent variable**

Example

- We know quantities of water and fertilizer providing to a tree for a month
- We want to predict the growth rate (height) of the tree.

Height?

Regression analysis

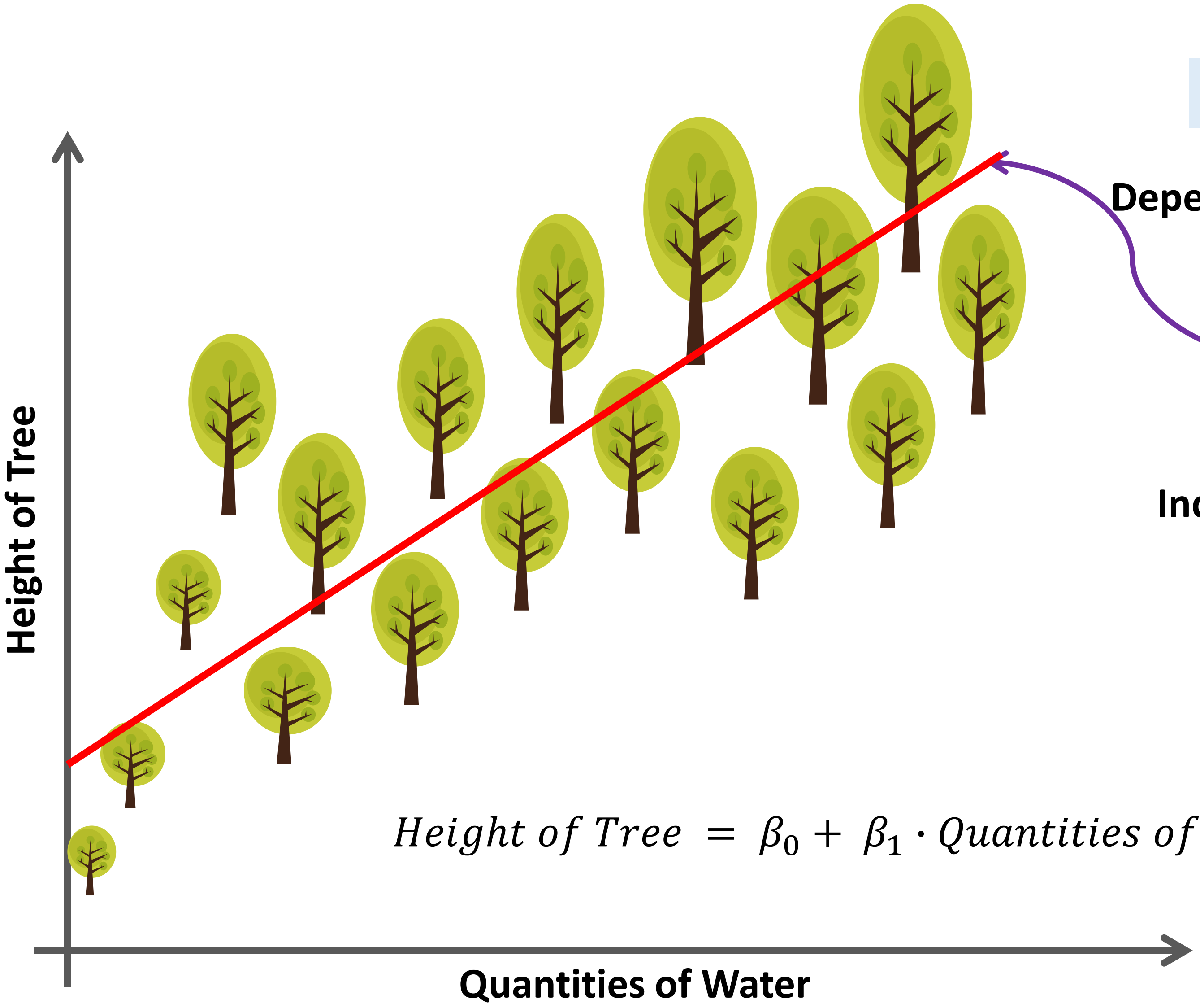


The task of regression is one of finding a **line** that most closely fits the data according to a specific mathematical criterion.

The line can be used for

- prediction and forecasting
- describing relationships between the independent and dependent variables.

Regression analysis



$$Height\ of\ Tree\ =\ \beta_0 +\ \beta_1 \cdot Quantities\ of\ Water$$

Goal is to estimate the function.

Dependent variable

Unknown parameters

$$y = \boxed{f\,(\mathrm{x}, \beta)} + e$$

Independent variables

Error terms

Regression Analysis

Types of Regression Problems

Number of Independent Variable

$= 1$

> 1

Simple Regression
Concerns two-dimensional sample points:

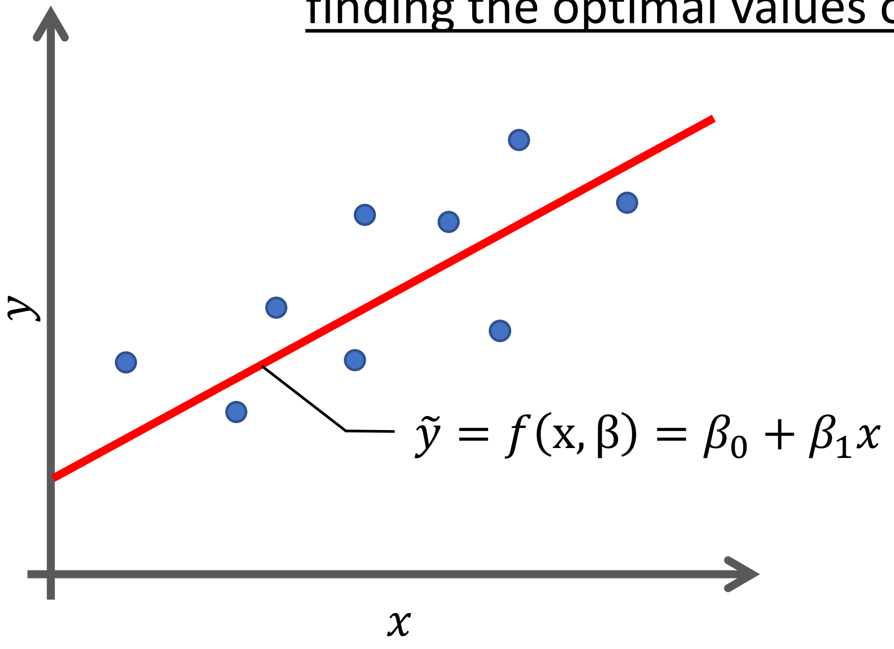
- one independent variable
- one dependent variable

Multiple Regression
Uses several independent variables to predict the outcome of a dependent variable.

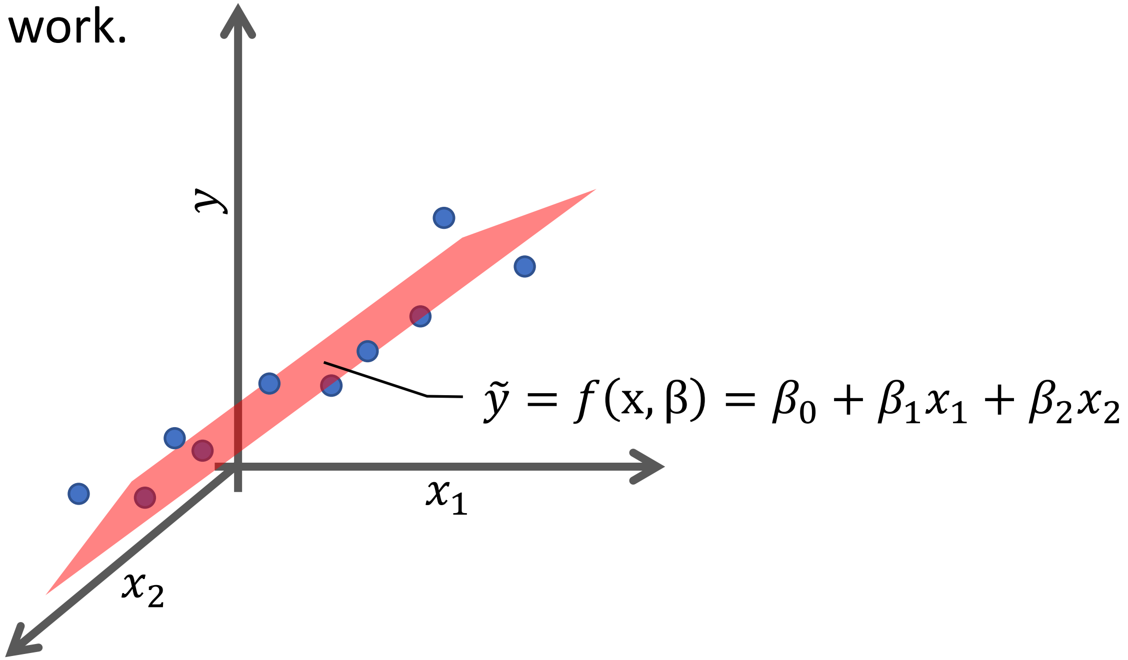
Linear Regression

Regression Analysis

We aim to fit **a line** or **hyperplane** to a scattering of data.
As the line or hyperplane is described by the parameters β,
<u>finding the optimal values of β</u> is our work.



Simple Linear Regression



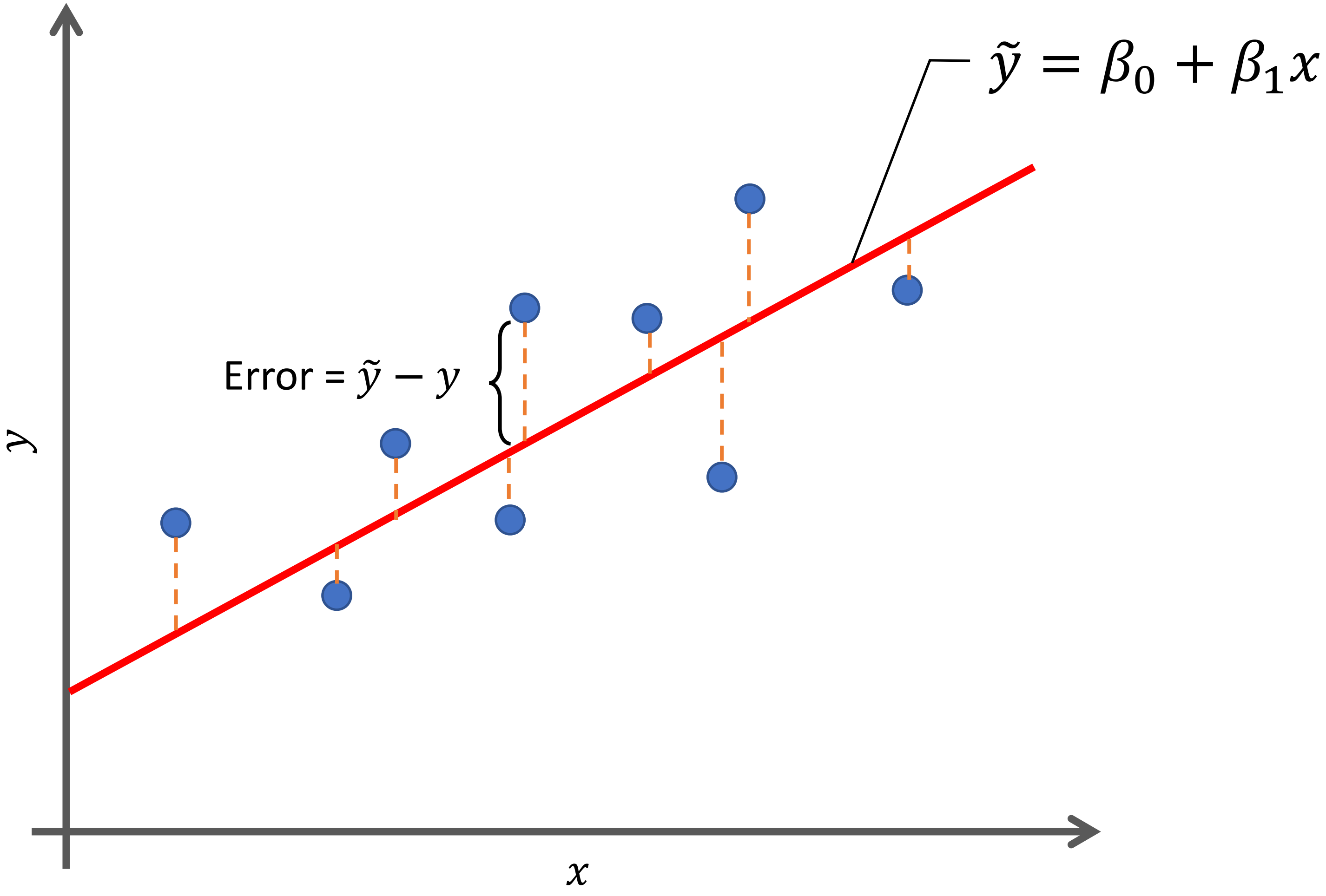
Multiple Linear Regression

Linear Regression

Regression Analysis

The value of parameters will be determined by fitting the line to training data.

Done by: minimize an *error function.*



Linear Regression

Regression Analysis

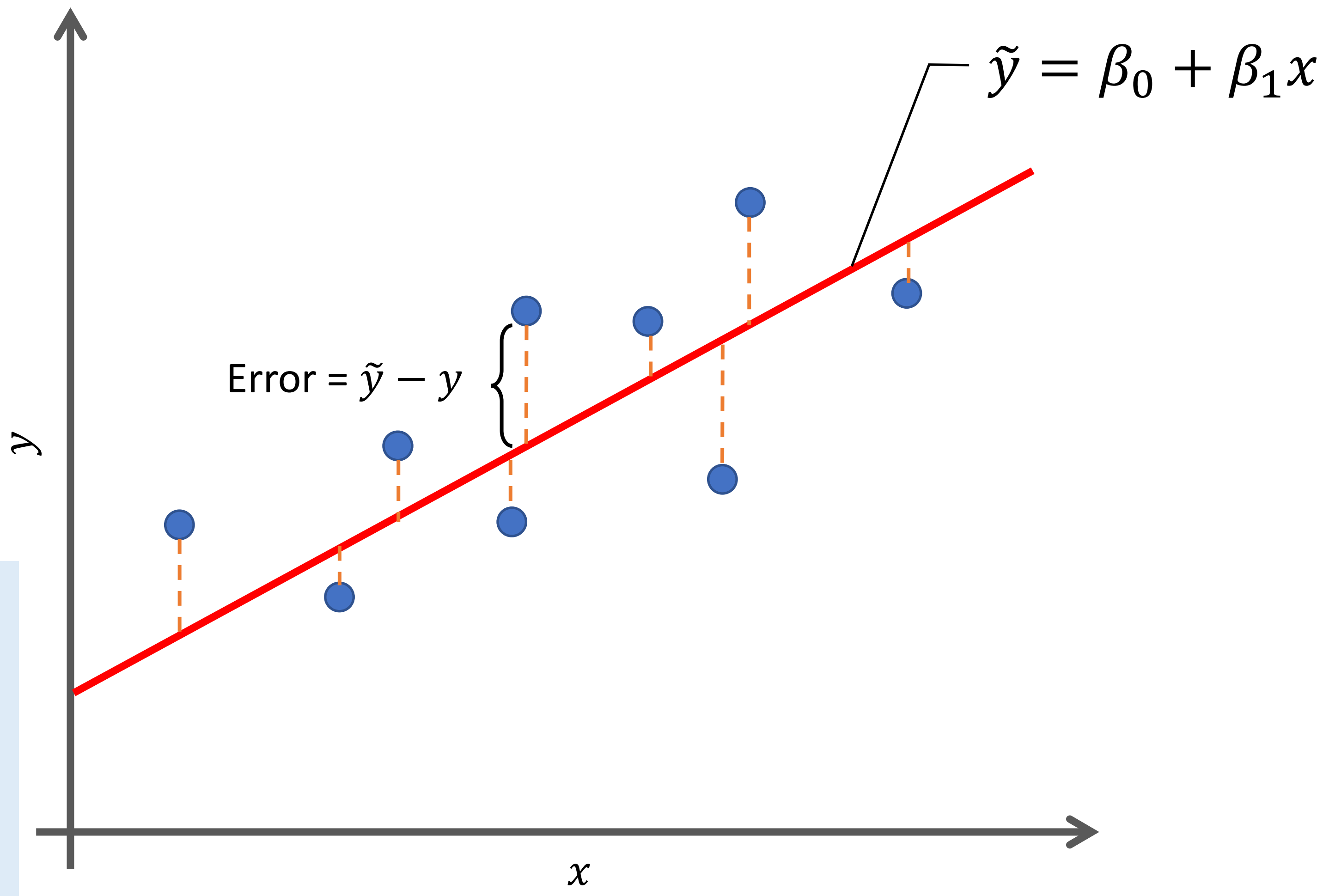
Sum of squared errors

$$E(\beta) = \sum_{i=1}^{n} (\tilde{y}_i - y_i)^2$$

$$= \sum_{i=1}^{n} (\beta_0 + \beta_1 x_i - y_i)^2$$

So, we find the parameter $\beta = [\beta_0, \beta_1]$ that provide a small value for $E(\beta)$.

This problem can be solved by optimization tools.



Linear Regression

Regression Analysis

Extend to multiple linear regression

$$\tilde{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

$$\tilde{y} = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$$

- The sum of squared error function can be defined by

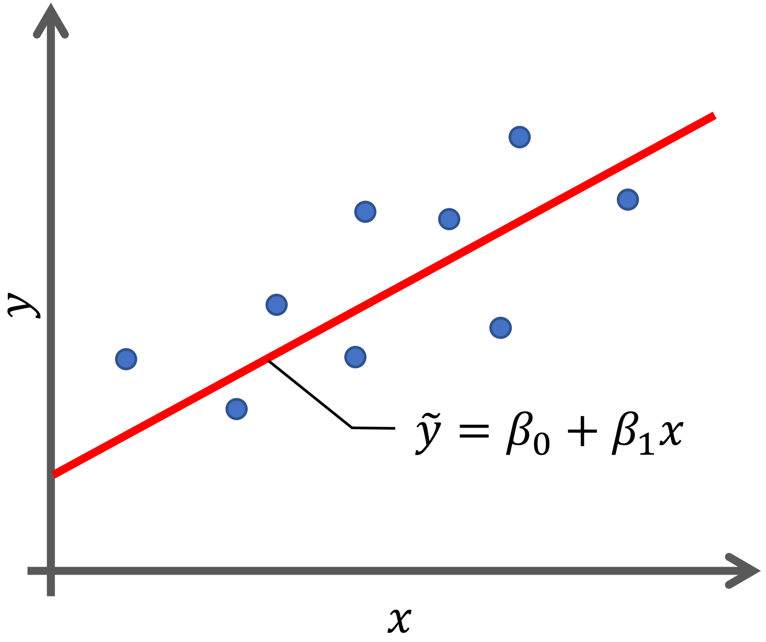
$$E(\beta) = \sum_{i=1}^{n} (\tilde{y}_i - y_i)^2$$

$$E(\beta) = \sum_{i=1}^{n} \left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j - y_i \right)^2$$

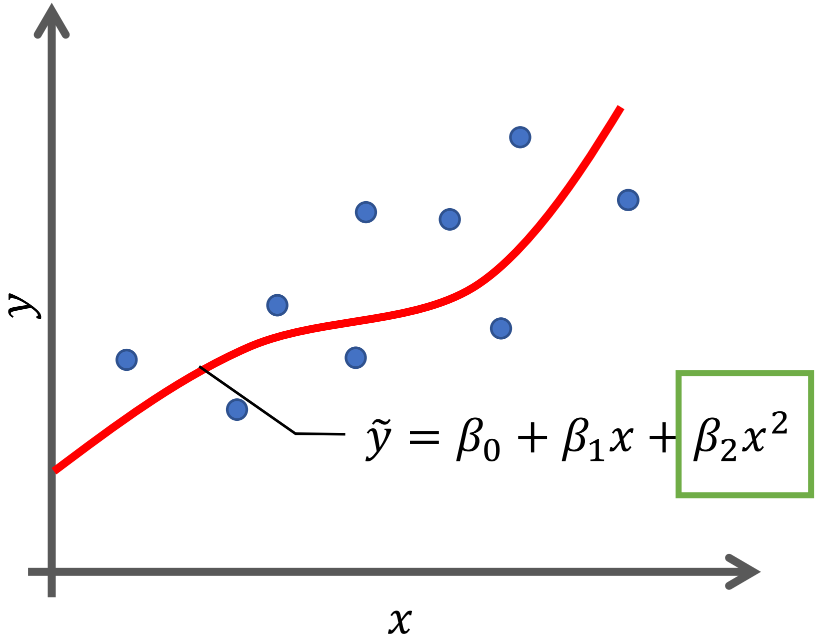
Polynomial Regression

Regression Analysis

Linear Regression	Polynomial Regression



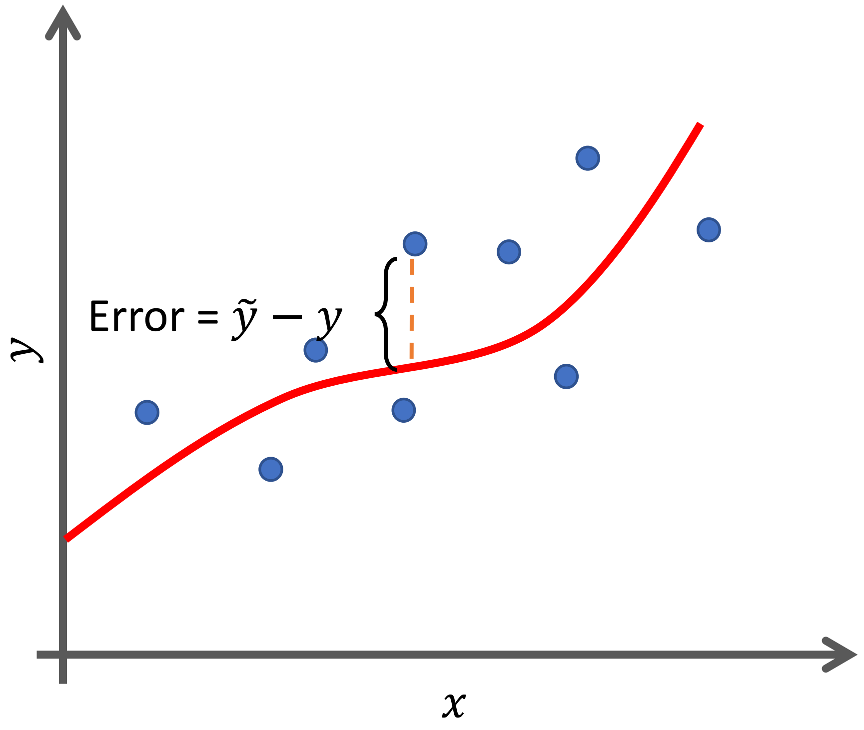
Relationship between the independent variable x and the dependent variable y is a linear model.



Relationship between the independent variable x and the dependent variable y is modelled as an n^{th} degree polynomial in x. (i.e. n=2)

Polynomial Regression

Regression Analysis



The general form of polynomial regression model:

$$\tilde{y} = \beta_0 + \sum_{d=1}^{M} \beta_d x^d$$

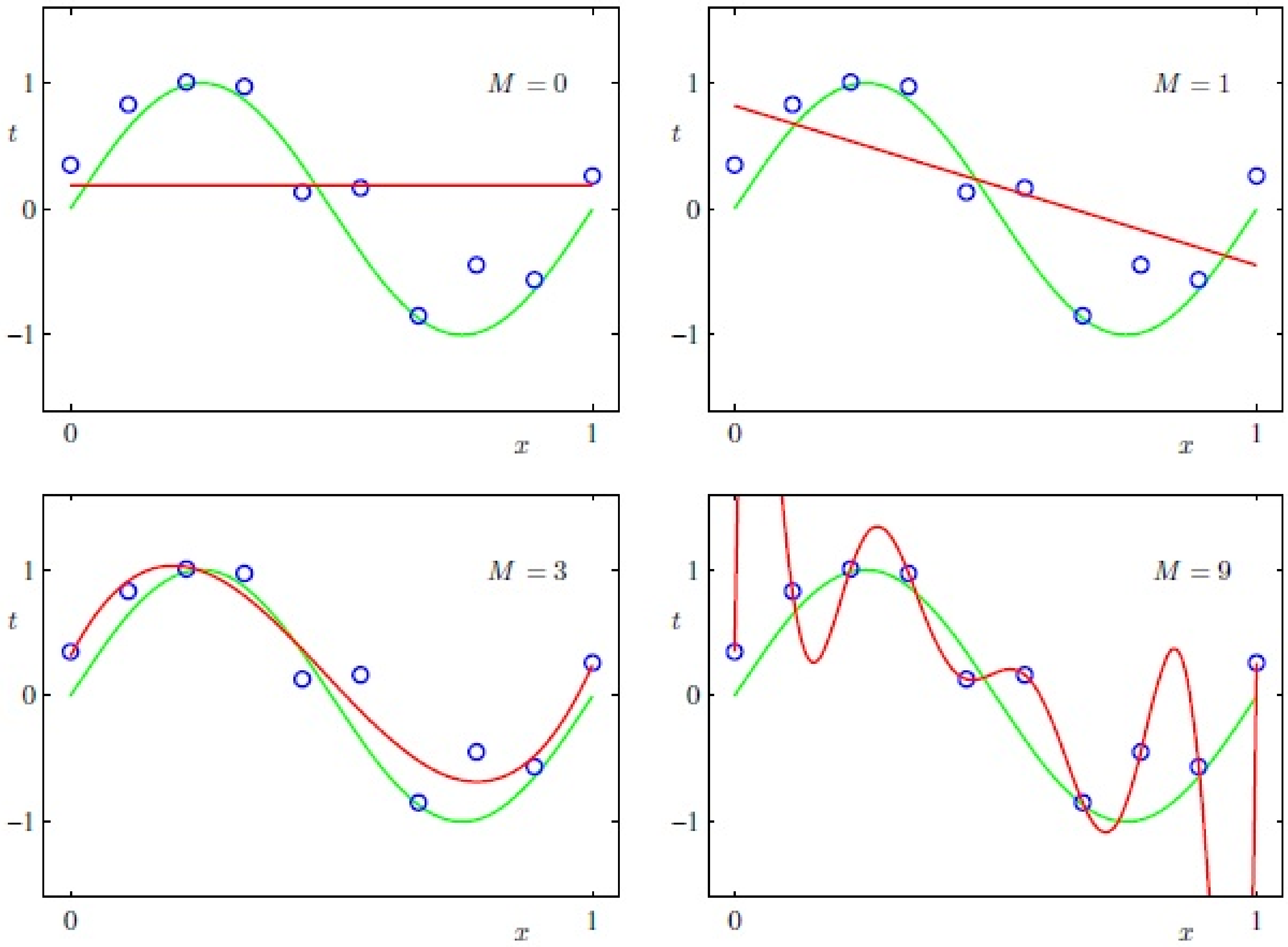
The best values of parameter $\beta = [\beta_0, \beta_1, \dots, \beta_M]$ can be determined by minimizing the sum of squared errors:

$$E(\beta) = \sum_{i=1}^{n} (\tilde{y}_i - y_i)^2$$

$$E(\beta) = \sum_{i=1}^{n} \left(\beta_0 + \sum_{d=1}^{M} \beta_d x^d - y_i \right)^2$$

Polynomial Regression

Regression Analysis



Plot of polynomials having various orders M, shown as red curves.

Source: Christopher M. Bishop (2006). Pattern Recognition and Machine Learning. New York: Springer-Verlag.

What happens when we go to a much higher order polynomial?

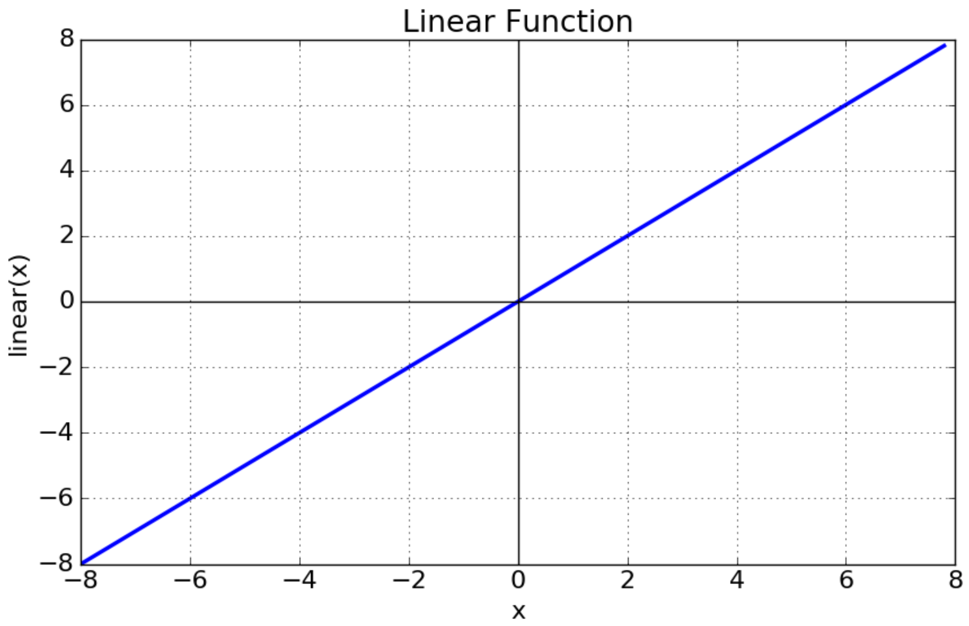
Over-fitting!

Artificial Neural Network
Regression Analysis

Artificial Neural Network can be also applied to regression analysis

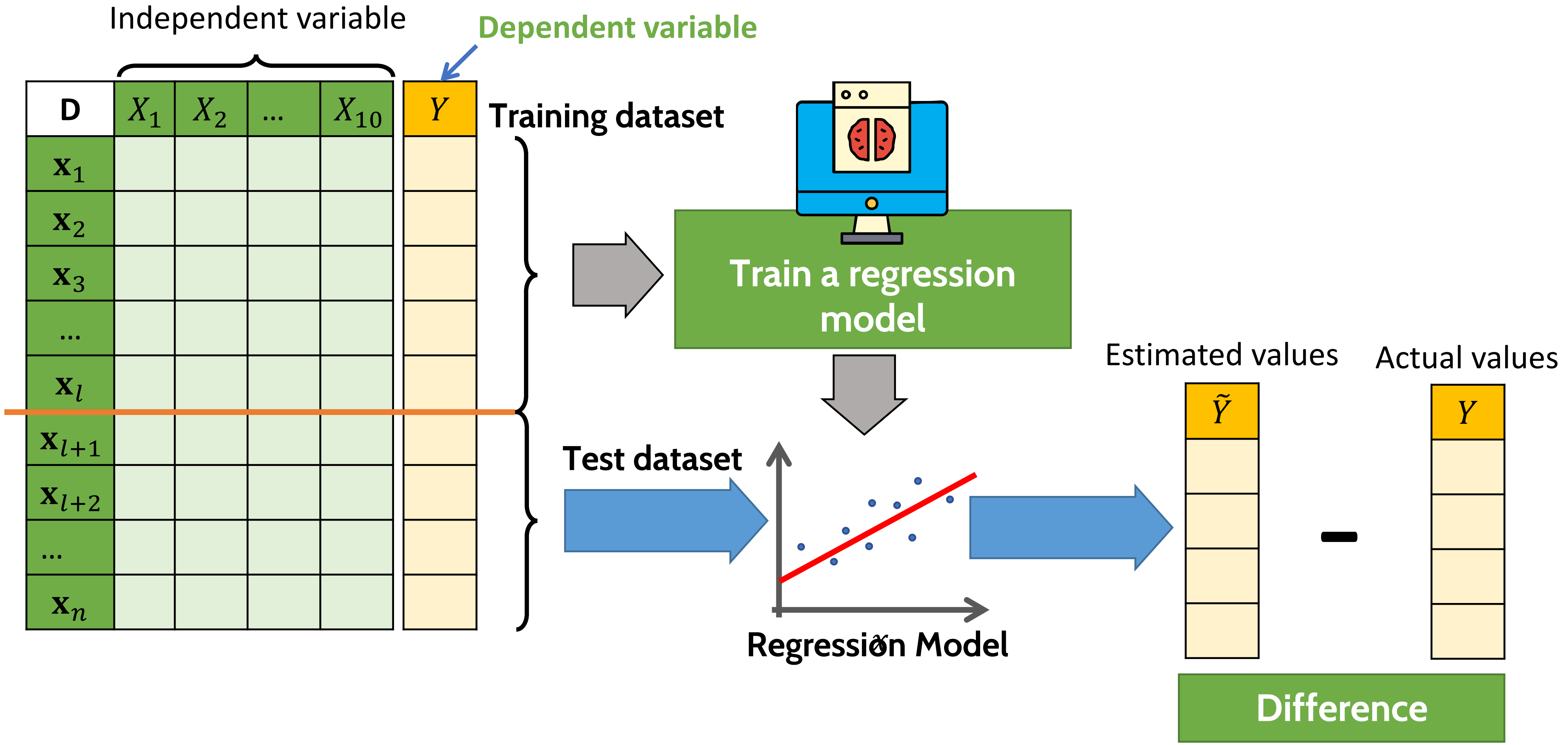
For regression,

Neurons in output layer applies *linear* function as the activation function.



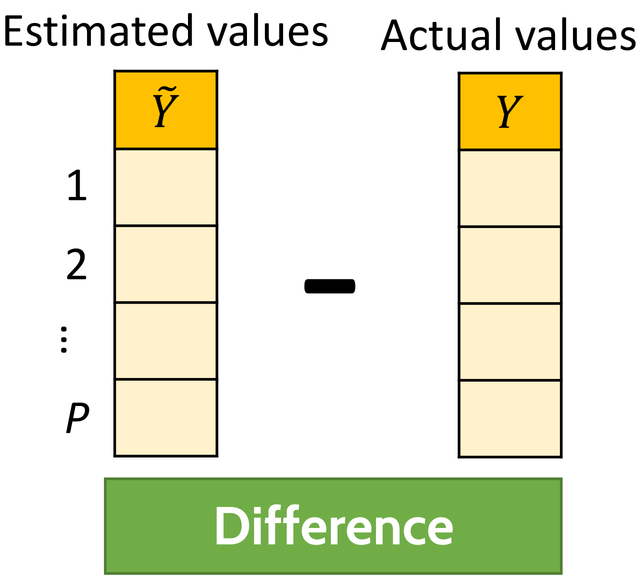
Regression Assessment

Regression Analysis



Regression Assessment

Regression Analysis



Mean Squared Error (MSE)

$$MSE = \frac{1}{P}\sum_{i=1}^{P}(\tilde{y}_i - y_i)^2$$

Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{P}\sum_{i=1}^{P}(\tilde{y}_i - y_i)^2}$$

Mean Absolute Error (MAE)

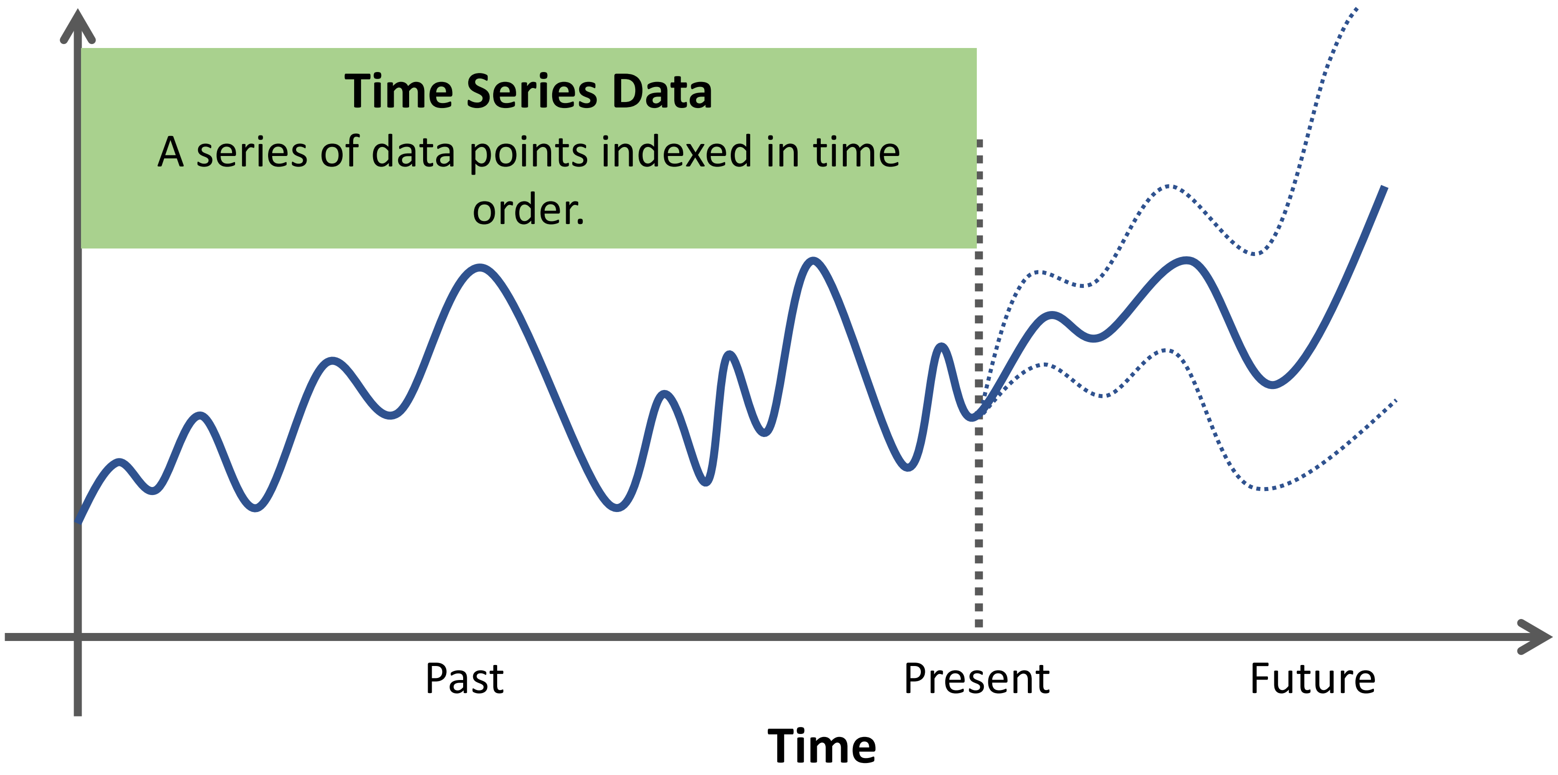
$$MAE = \frac{1}{P}\sum_{i=1}^{P}|\tilde{y}_i - y_i|$$

MSE, RMSE and MAE ≥ 0

A lower value and is better than a higher one.

Time Series Analysis

Time Series Data



Time series data can be found in **signal processing, econometrics, mathematical finance, weather forecasting, control engineering, astronomy, communications engineering, etc.**

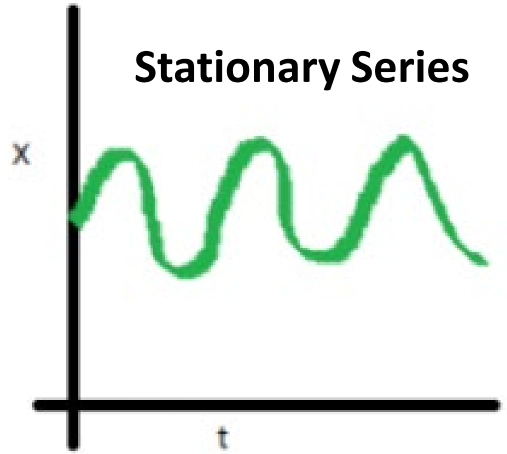
Time Series Analysis

Characteristics of Time Series Data

Stationary

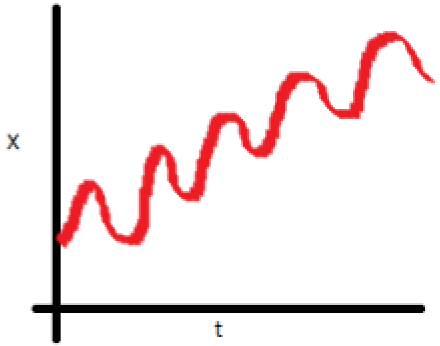
Statistical properties do not change over time.

- Mean
- Variance
- Covariance

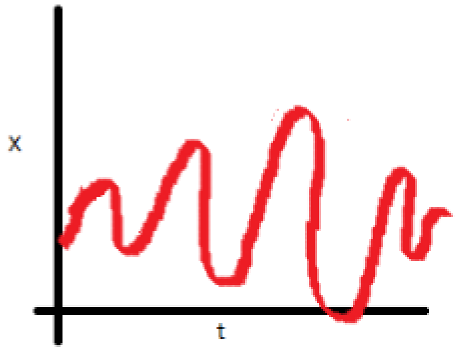


Source: https://medium.com/greyatom/time-series-b6ef79c27d31

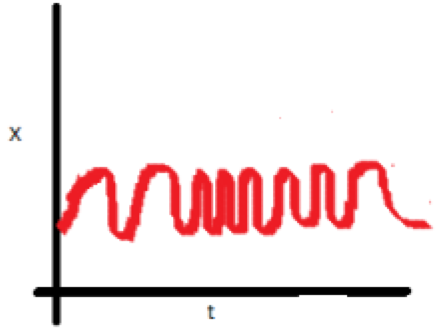
Non-stationary Series



Mean increases with time.



Variance of the series is a function of time.



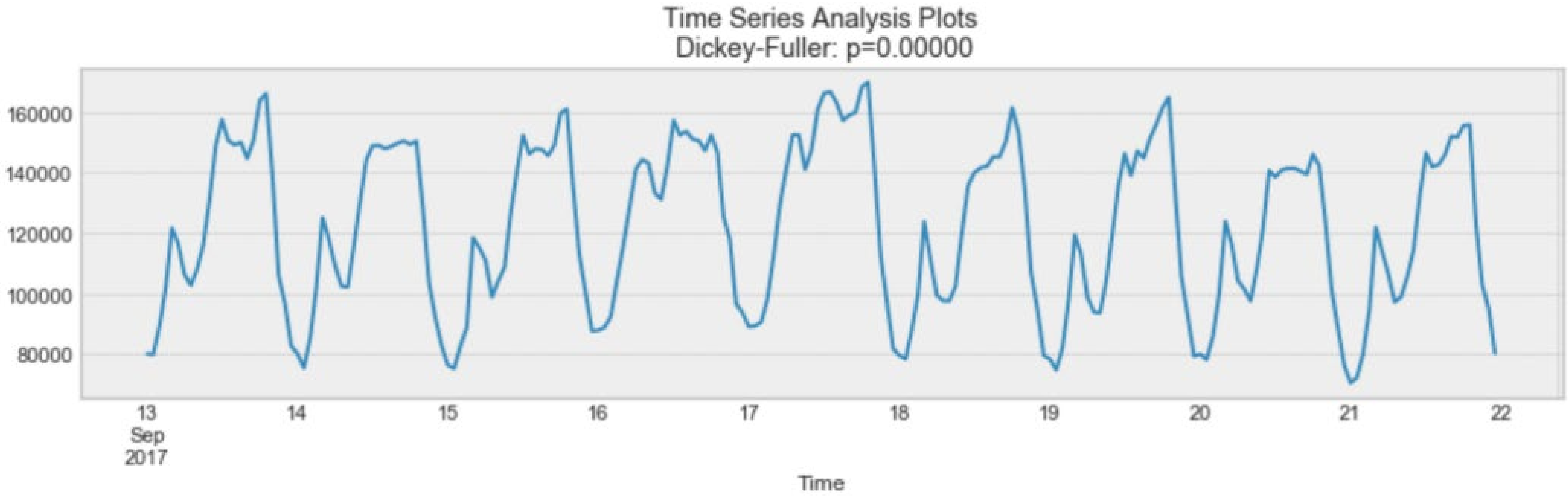
The spread becomes closer as the time increases.

Time Series Analysis

Characteristics of Time Series Data

Seasonality

Periodic fluctuations - pattern that recurs or repeats over regular intervals.



Example of seasonality

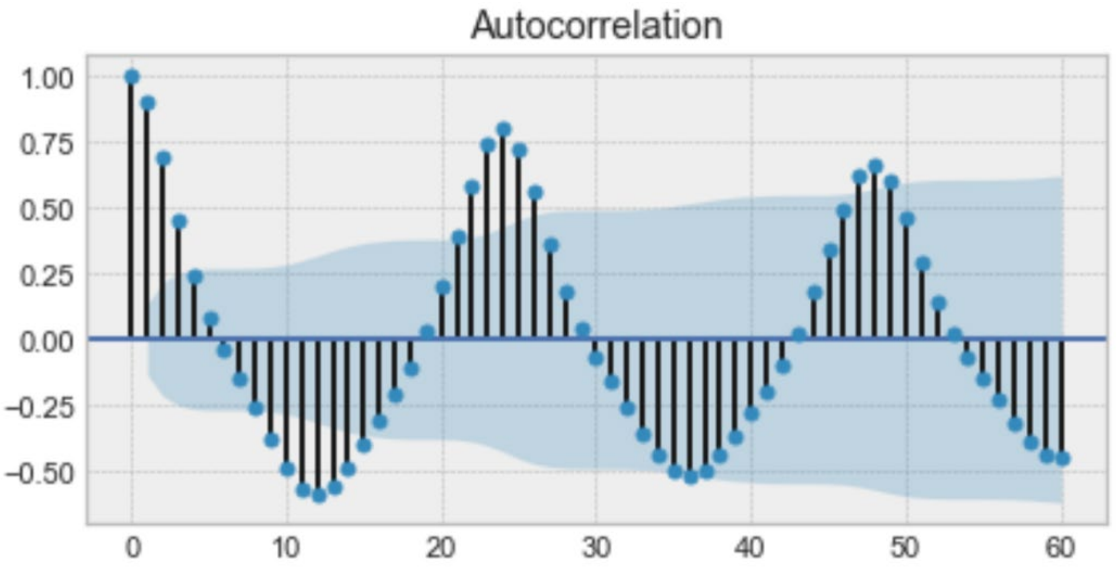
Source: https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775

Time Series Analysis

Characteristics of Time Series Data

Autocorrelation

- Internal correlation in a time series.
- The similarity between observations as a function of the time lag between them.



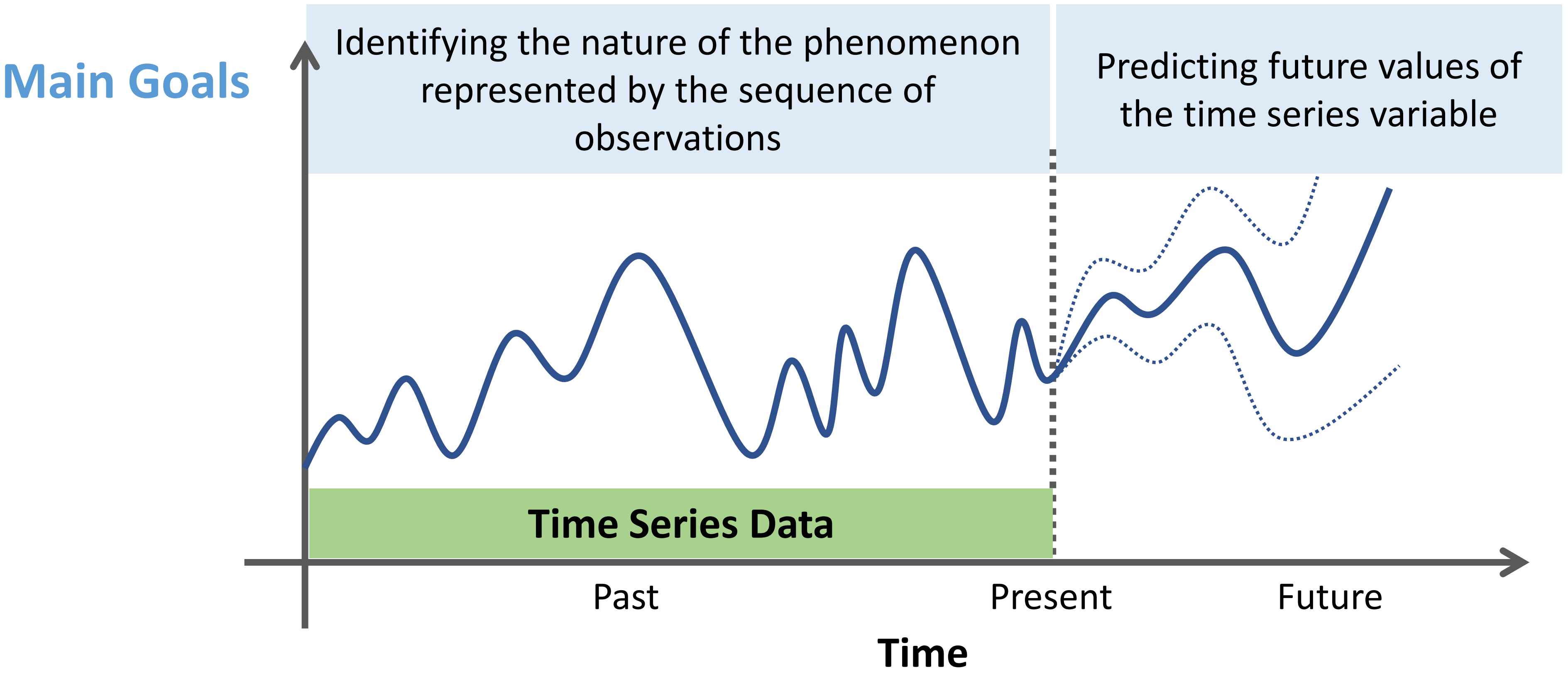
Example of an autocorrelation plot - we will find a very similar value at every 24 unit of time.

Source: https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775

Time Series Analysis

Time Series Analysis

Analysis techniques that deal with time series data.

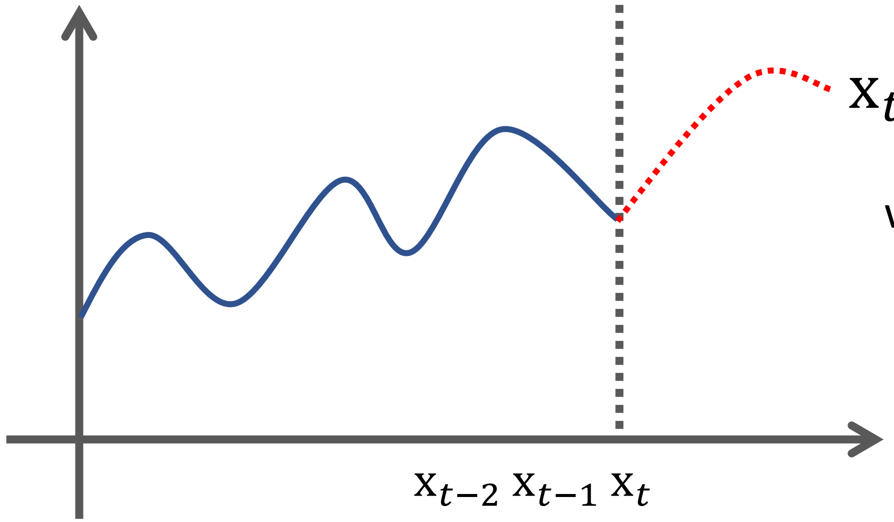


Autoregressive Model

Time Series Analysis

The output variable depends linearly on:

- Its own previous values
- A stochastic term (an imperfectly predictable term)



$$x_t = c + \underbrace{\varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p}}_{\text{Linear combination of } p \text{ previous observations}} + \underbrace{\varepsilon_t}_{\text{stochastic term}}$$

where c is a constant

$\varphi_1, \varphi_2, \ldots, \varphi_p$ are the autoregressive model parameters

ε_t is write noise

Finding the optimal values of $\boldsymbol{\varphi_1, \varphi_2, \ldots, \varphi_p}$ is the work for fitting the model.

There are many ways to estimate the parameters, such as

- The ordinary least squares procedure
- Method of moments (through Yule–Walker equations).

Autoregressive Model

Time Series Analysis

AR(p) model : $x_t = c + \sum_{i=1}^{p} \varphi_i x_{t-i} + \varepsilon_t$

How can we determine the maximum lag *p*?

Decide based on:

- **Autocorrelation *function***
- ***Partial autocorrelation function***

Autoregressive Model

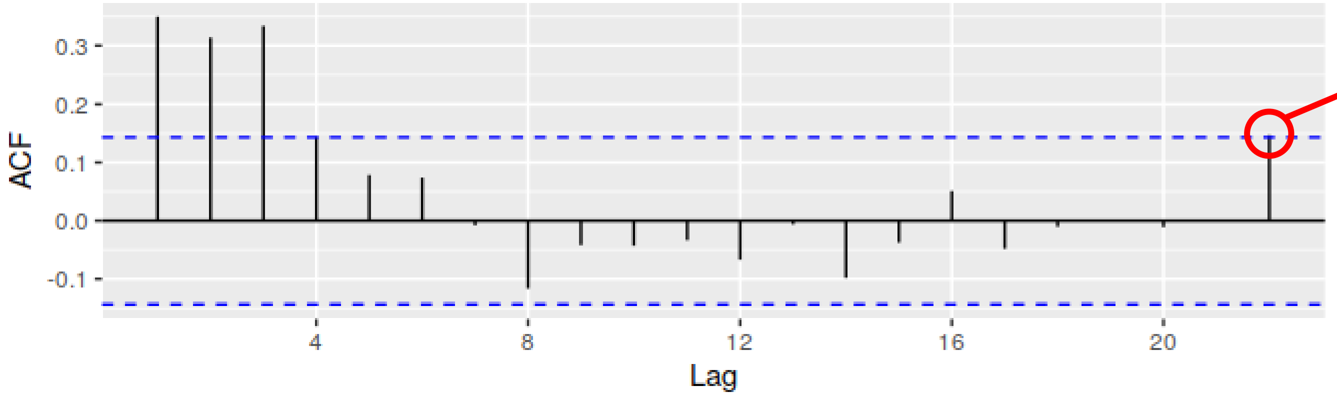
Time Series Analysis

Autocorrelation Function

- Autocorrelation refers to how correlated a time series is with its past values.
- It measures the linear relationship between *lagged values* of a time series.

$$ACF(k) = \frac{\sum_{t=k+1}^{T} (\mathrm{x}_t - \bar{\mathrm{x}})(\mathrm{x}_{t-k} - \bar{\mathrm{x}})}{\sum_{t=1}^{k} (\mathrm{x}_t - \bar{\mathrm{x}})^2}$$

where T is the length of the time series.



Always measured between +1 and -1.

- +1 : a strong positive association
- -1 : a strong negative association
- 0 : no association.

ACF of quarterly percentage change in US consumption.

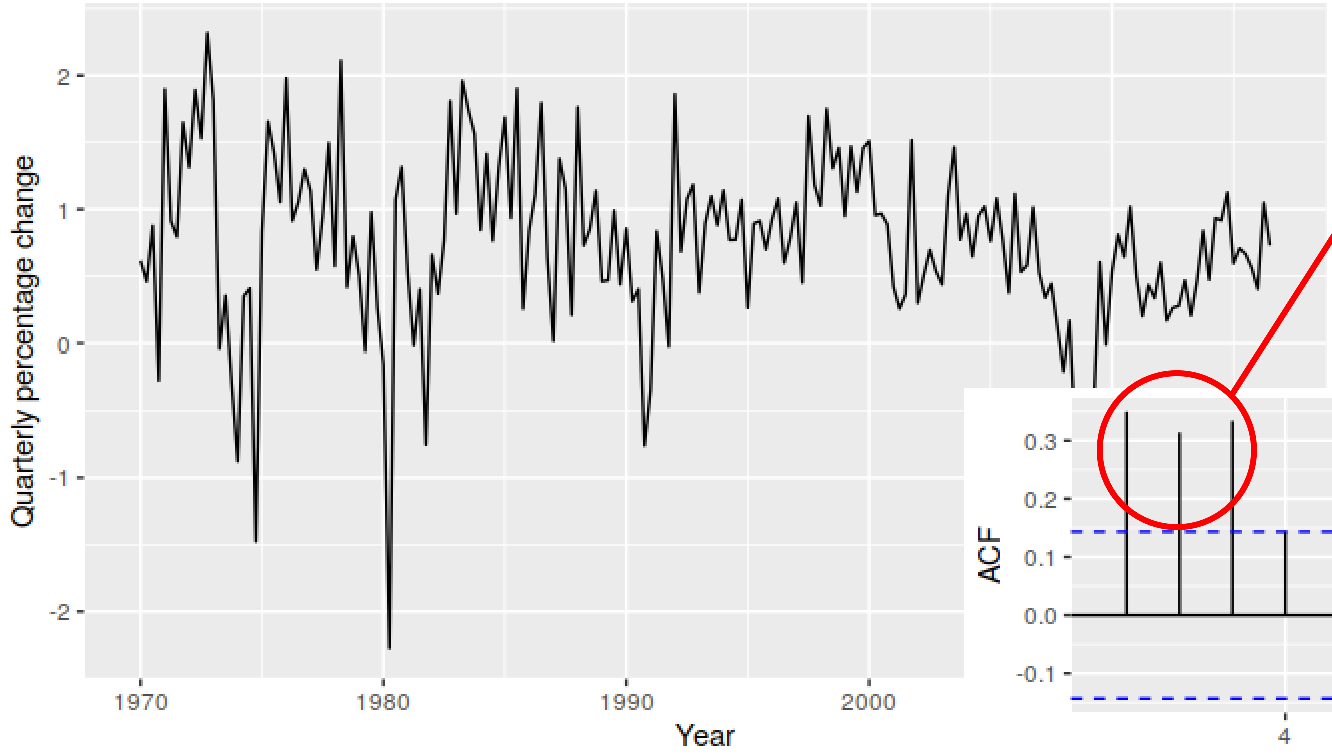
Source: https://otexts.com/fpp2/non-seasonal-arima.html

Autoregressive Model

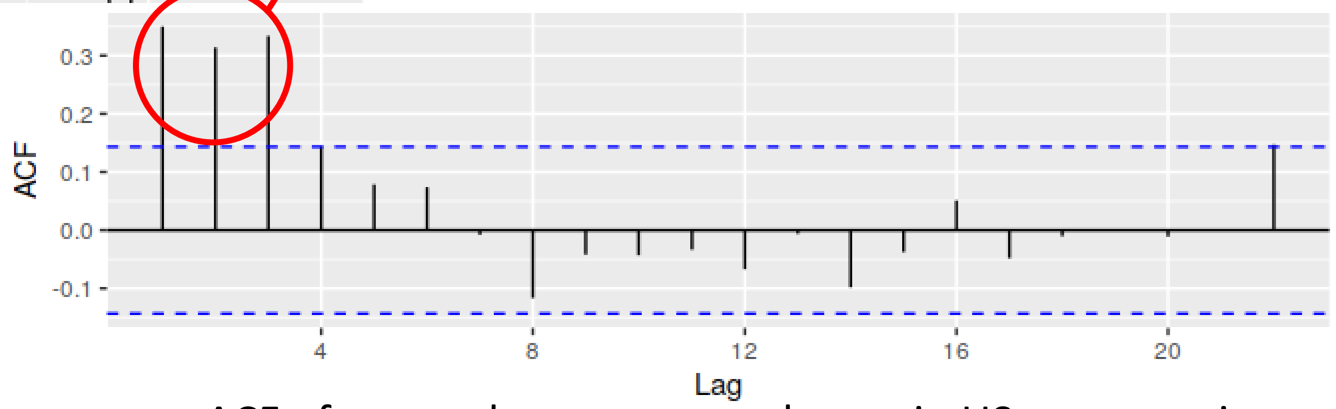
Time Series Analysis

Quarterly percentage change in US consumption expenditure.

Source: https://otexts.com/fpp2/non-seasonal-arima.html



We notice that there are significant spike at a lag of 1, 2 and 3 and much lower spikes for the subsequent lags.



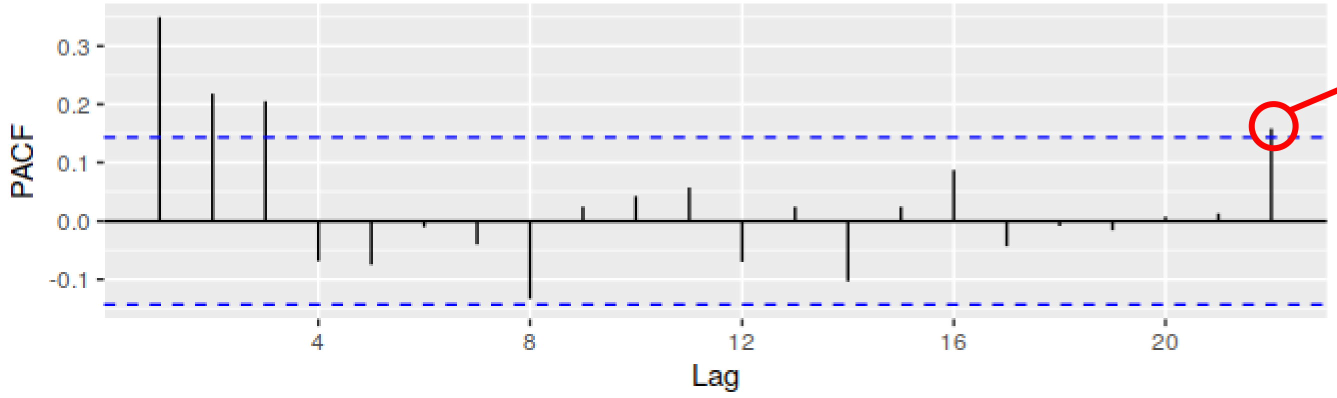
ACF of quarterly percentage change in US consumption

So, our AR model becomes $x_t = c + \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \varphi_3 x_{t-3} + \varepsilon_t$ **AR(3)**

Autoregressive Model
Time Series Analysis

Partial Autocorrelation Function

- It measures the relationship between x_t and x_{t-k} after removing the effects of lags $1,2,3, \ldots, k-1$.



Always measured between +1 and -1.

- +1 : a strong positive association
- -1 : a strong negative association
- 0 : no association.

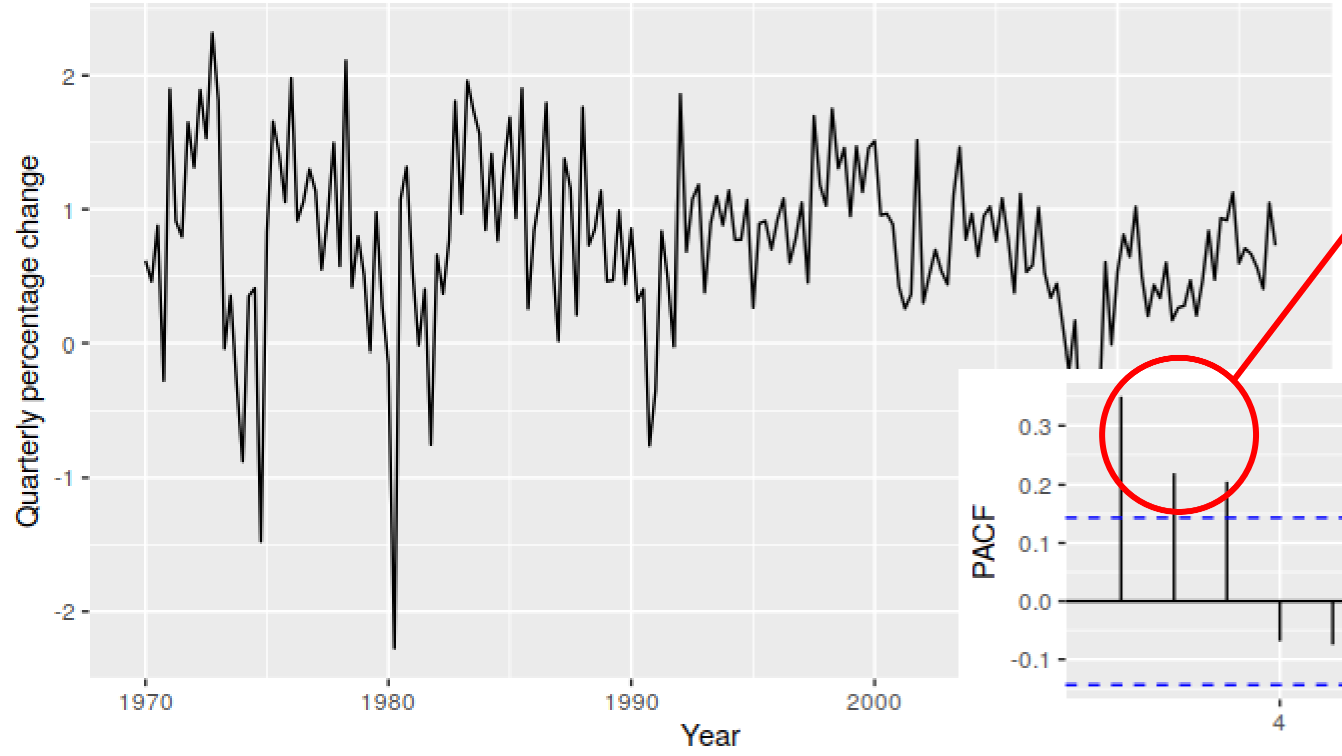
PACF of quarterly percentage change in US consumption.
Source: https://otexts.com/fpp2/non-seasonal-arima.html

Autoregressive Model

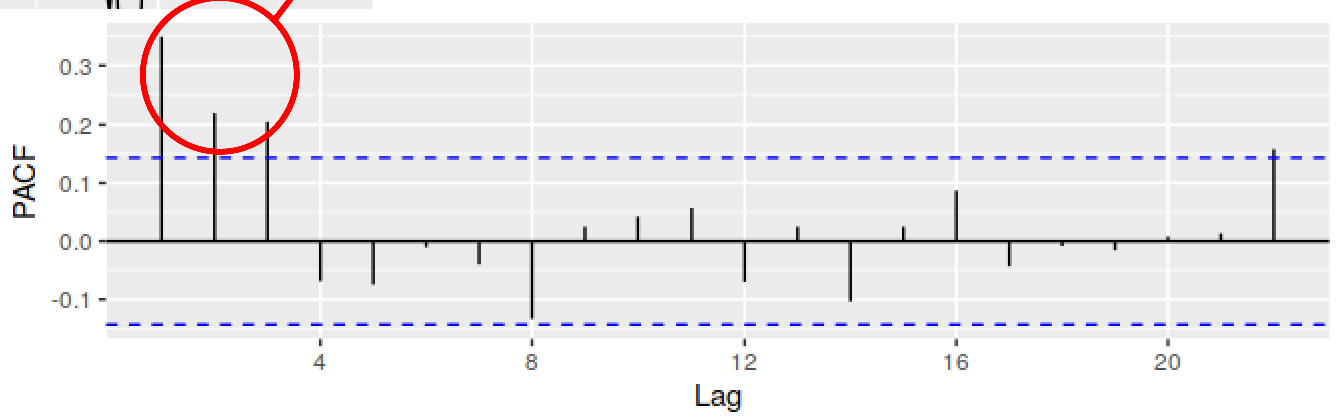
Time Series Analysis

Quarterly percentage change in US consumption expenditure.

Source: https://otexts.com/fpp2/non-seasonal-arima.html



We notice that there are significant spike at a lag of 1, 2 and 3 and much lower spikes for the subsequent lags.



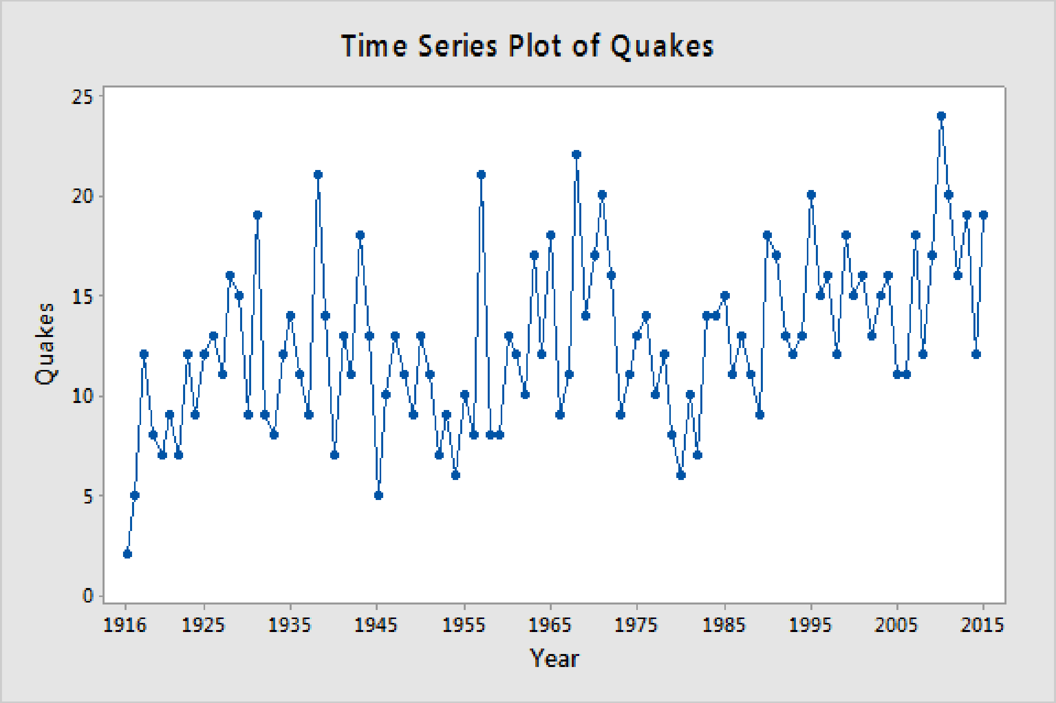
PACF of quarterly percentage change in US consumption

So, our AR model becomes $x_t = c + \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \varphi_3 x_{t-3} + \varepsilon_t$ **AR(3)**

Autoregressive Model

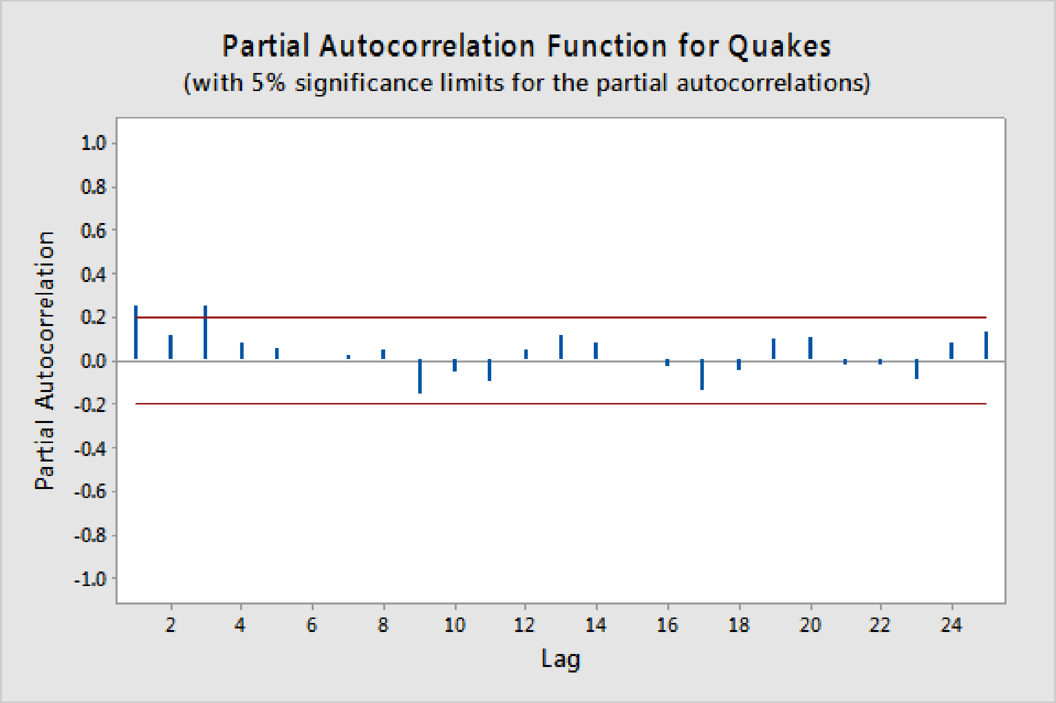
Time Series Analysis

The annual number of worldwide earthquakes with magnitude greater than 7 on the Richter scale for *n* = 100 years



Quiz:

What is an appropriate AR model of quake?



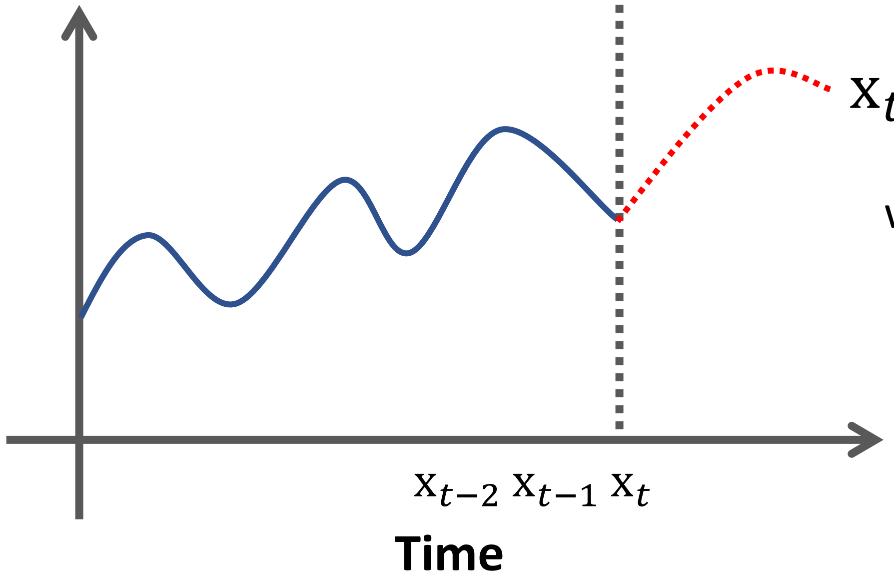
Source: https://online.stat.psu.edu/stat501/lesson/14/14.1

Moving Average Model

Time Series Analysis

The output variable depends linearly on:

- Past forecast errors
- A stochastic term (an imperfectly predictable term)



Linear combination of q previous forecast errors

$$x_t = \mu + \varepsilon_t + \underbrace{\theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \cdots + \theta_p\,\varepsilon_{t-q}}$$

where μ is the mean of the series
$\theta_1, \theta_2, \ldots, \theta_q$ are the moving average model parameters
ε_t is white noise

Finding the optimal values of $\boldsymbol{\theta_1, \theta_2, \ldots, \theta_q}$ is the work for fitting the model.

- Fitting the MA estimates is more complicated than it is in autoregressive models, because the <u>lagged error terms are not observable</u>.
- Iterative non-linear fitting procedures need to be used.

Moving Average Model

Time Series Analysis

MA(q) model : $\mathrm{x}_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$

How can we determine the maximum lag *q*?

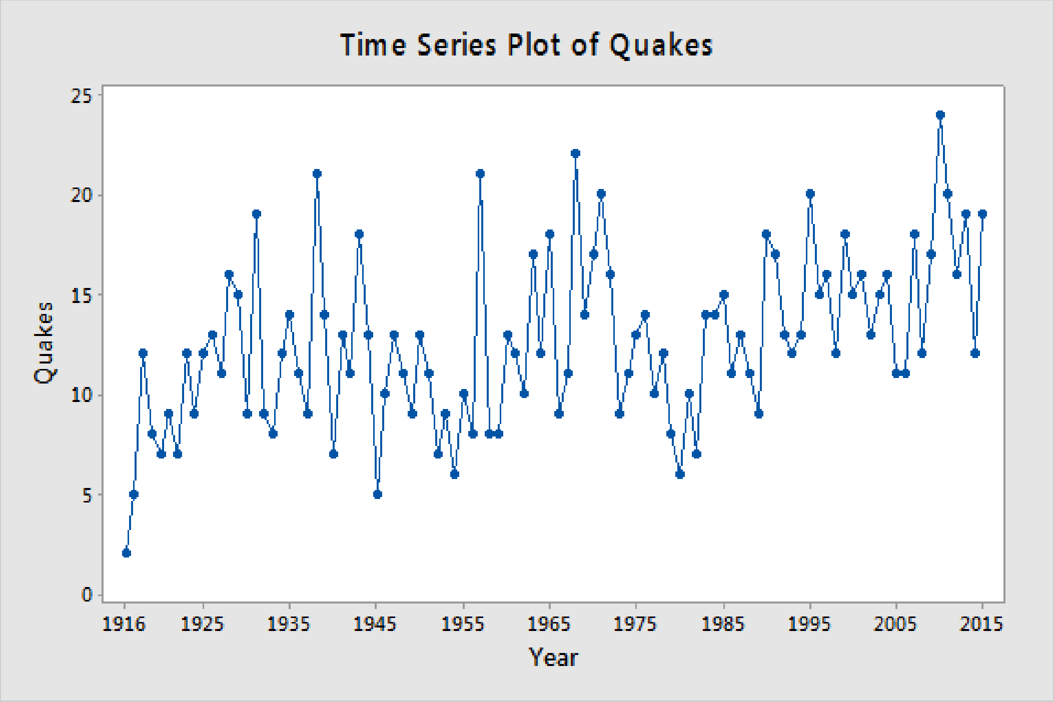
Decide based on:

- **Autocorrelation *function***
- ***Partial autocorrelation function***

Moving Average Model

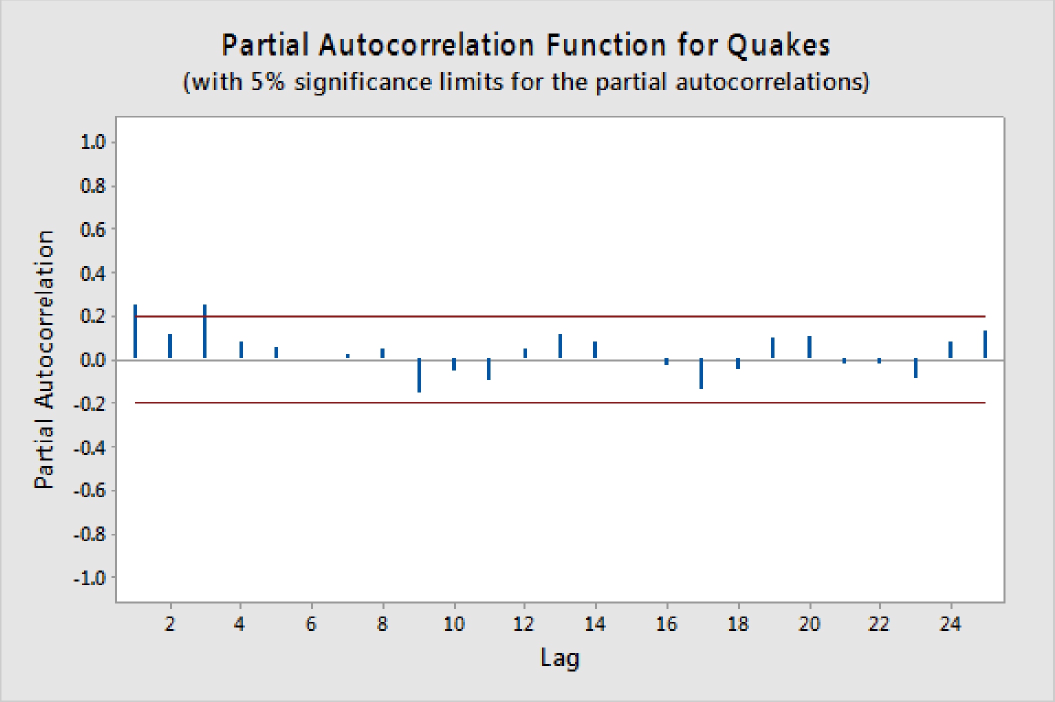
Time Series Analysis

The annual number of worldwide earthquakes with magnitude greater than 7 on the Richter scale for n = 100 years



Quiz:

What is an appropriate MA model of quake?



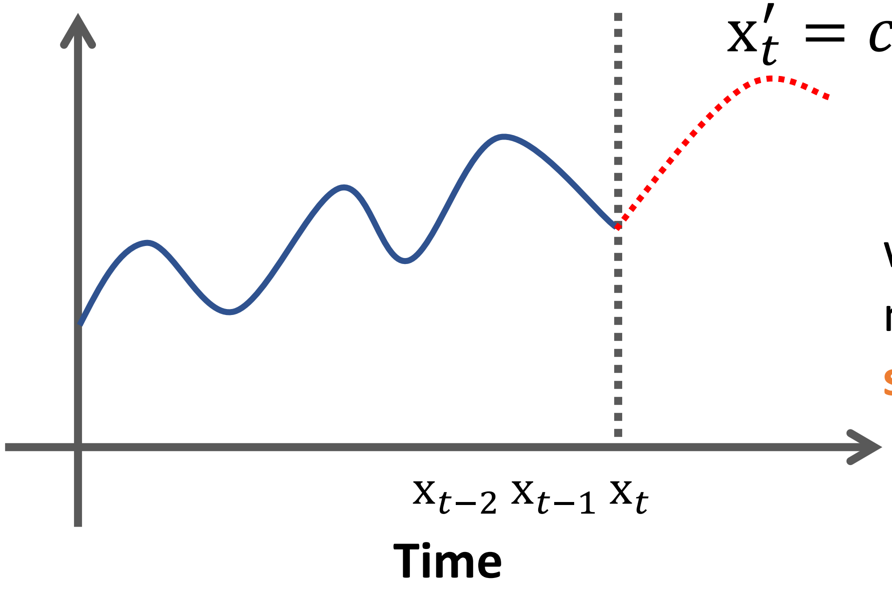
Source: https://online.stat.psu.edu/stat501/lesson/14/14.1

Autoregressive Integrated Moving Average (ARIMA)

Time Series Analysis

Combination of autoregressive and moving average models.

- **Autoregression** - AR(p): $x_t = c + \sum_{i=1}^{p} \varphi_i x_{t-i} + \varepsilon_t$
- **Moving Average** - MA(q): $x_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$
- **Integration** - the reverse of differencing (transform non-stationarity to stationarity)



$$x'_t = c + \underbrace{\varphi_1 x'_{t-1} + \cdots + \varphi_p x'_{t-p}}_{\text{AR(p)}} + \underbrace{\theta_1 \varepsilon_{t-1} + \cdots + \theta_p \varepsilon_{t-q}}_{\text{MA(q)}} + \varepsilon_t$$

where x'_t is the differenced series. It may have been differenced more than once (**take d times of the first difference until the time series is not stationary**)

Finding the optimal values of $\boldsymbol{\varphi_1, \varphi_2, \ldots, \varphi_p}$ and $\boldsymbol{\theta_1, \theta_2, \ldots, \theta_q}$ is the work for fitting the model.

Autoregressive Integrated Moving Average (ARIMA)

Time Series Analysis

$$x'_t = c + \varphi_1 x'_{t-1} + \cdots + \varphi_p\, x'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_p\, \varepsilon_{t-q} + \varepsilon_t$$

AR(p) **MA(q)**

where x'_t is the differenced series. It may have been differenced more than once (**take *d* times of the first difference until the time series is not stationary**)

ARIMA(p,d,q)

p, d and q are hyper-parameters that we need to determine.

Autoregressive Integrated Moving Average (ARIMA)

Time Series Analysis

Perform ARIMA

Step 1 **Check stationarity**	If a time series has a trend or seasonality component, it must be made stationary before we can use ARIMA to forecast.
Step 2 **Difference**	If the time series is not stationary, it needs to be stationarized through differencing.
Step 3 **Filter out a validation sample**	This will be used to validate how accurate our model is. Use train test validation split to achieve this
Step 4 **Select AR and MA terms**	Use the ACF and PACF to decide whether to include an AR term(s), MA term(s), or both.
Step 5 **Build the model**	Build the model and set the number of periods to forecast to N (depends on your needs).
Step 6 **Validate model**	Compare the predicted values to the actuals in the validation sample.

Parameter ***d*** is determined here.

Autoregressive Integrated Moving Average (ARIMA)

Time Series Analysis

Determine suitable values of p and q using either AIC, AICc or BIC value.

Akaike information criterion (AIC)

$$\mathrm{AIC} = -2\log(L) + 2(p+q+k+1)$$

where L is the likelihood of the data,
$k = 1$ if $c \neq 0$ and $k = 0$ if $c = 0$.

Corrected AIC (AICc)

$$\mathrm{AICc} = \mathrm{AIC} + \frac{2(p+q+k+1)(p+q+k+2)}{T-p-q-k-2}$$

Bayesian Information Criterion (BIC)

$$\mathrm{BIC} = \mathrm{AIC} + [\log(T) - 2](p+q+k+1)$$

Good models are obtained by <u>minimizing</u> the AIC, AICc or BIC.

Autoregressive Integrated Moving Average (ARIMA)

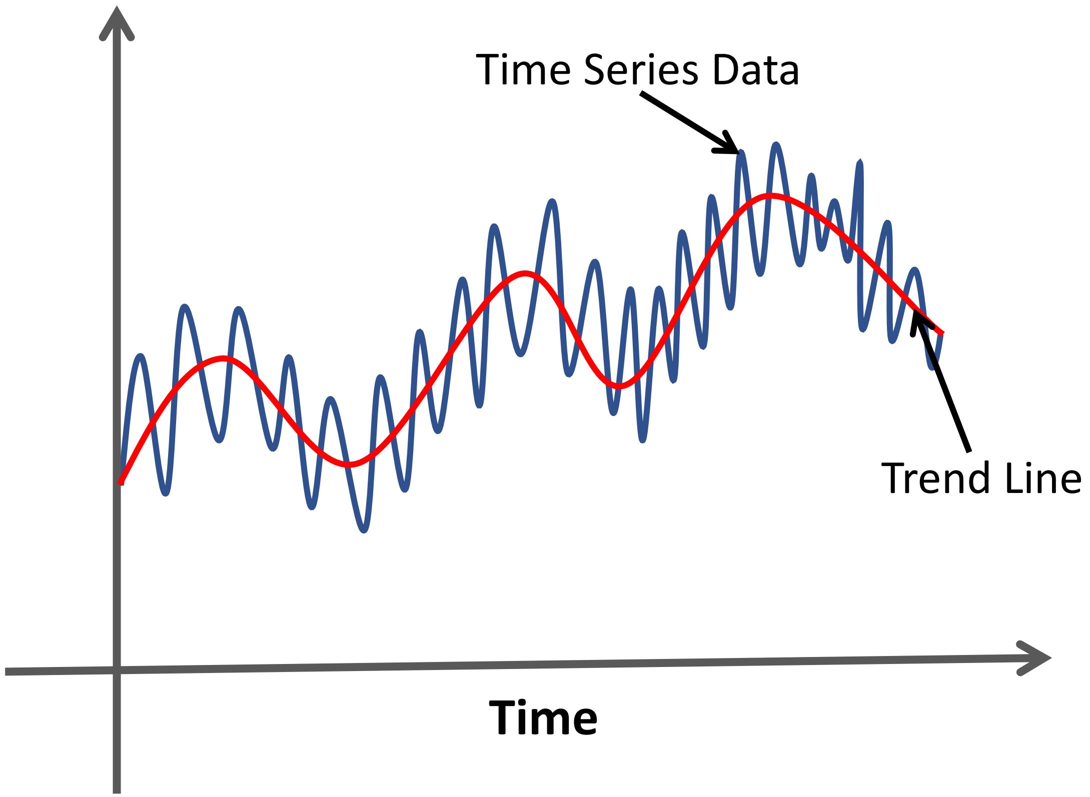
Time Series Analysis

Determine suitable values of p and q using either AIC, AICc or BIC value.

		p in AR(p)					
		0	1	2	3	4	5
q in MA(q)	0	4588.666	4588.472	4589.884	4591.619	4592.181	4593.312
	1	4588.618	4584.675	4586.262	4588.261	4590.172	4592.002
	2	4590.031	4586.263	4588.317	4590.25	4590.726	4594.104
	3	4591.883	4589.089	4583.762	4593.013	4589.644	4590.99
	4	4592.883	4590.161	4592.254	4594.099	4583.88	4586.875
	5	4594.055	4590.793	4594.07	4596.018	4586.779	4587.788

Moving Average Smoothing

Time Series Analysis



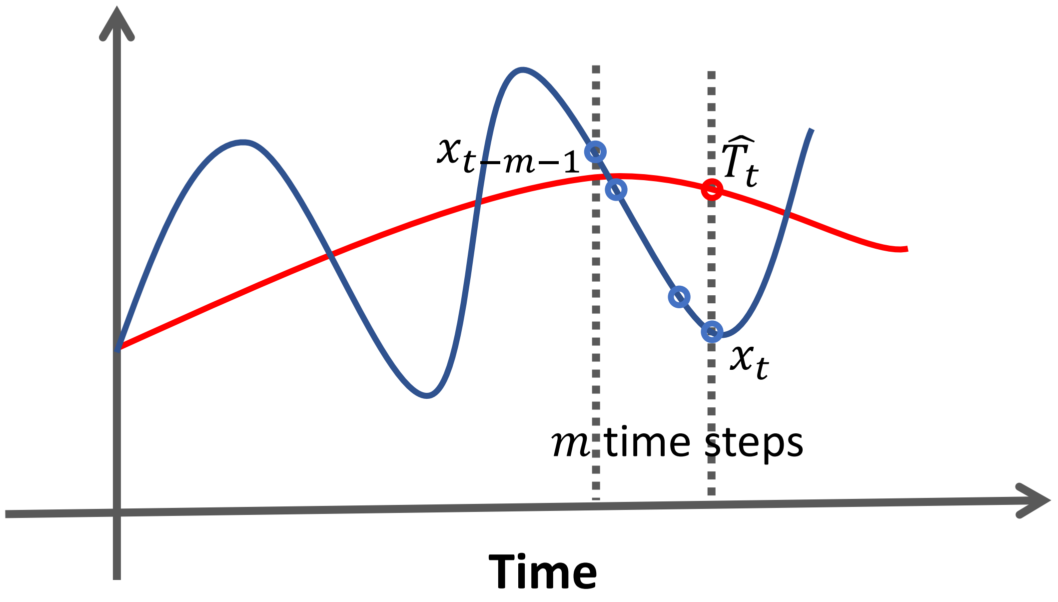
- Smooth out short-term fluctuations
- Highlight longer-term trends or cycles.

Purpose: to help improve understanding of the time series

Moving Average Smoothing

Time Series Analysis

Simple Moving Average



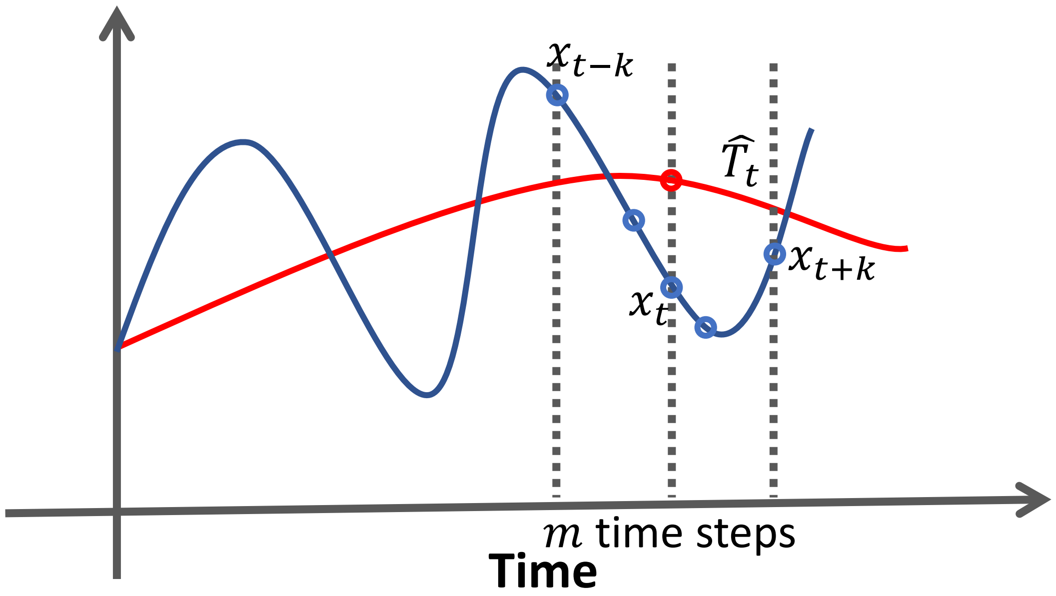
Financial Applications: the unweighted mean of the previous n data.

$$\widehat{T}_t = \frac{1}{m}\sum_{i=0}^{m-1} x_{t-i}$$

Moving Average Smoothing

Time Series Analysis

Simple Moving Average



Science and Engineering: the mean is taken from an equal number of data on either side of a central value.

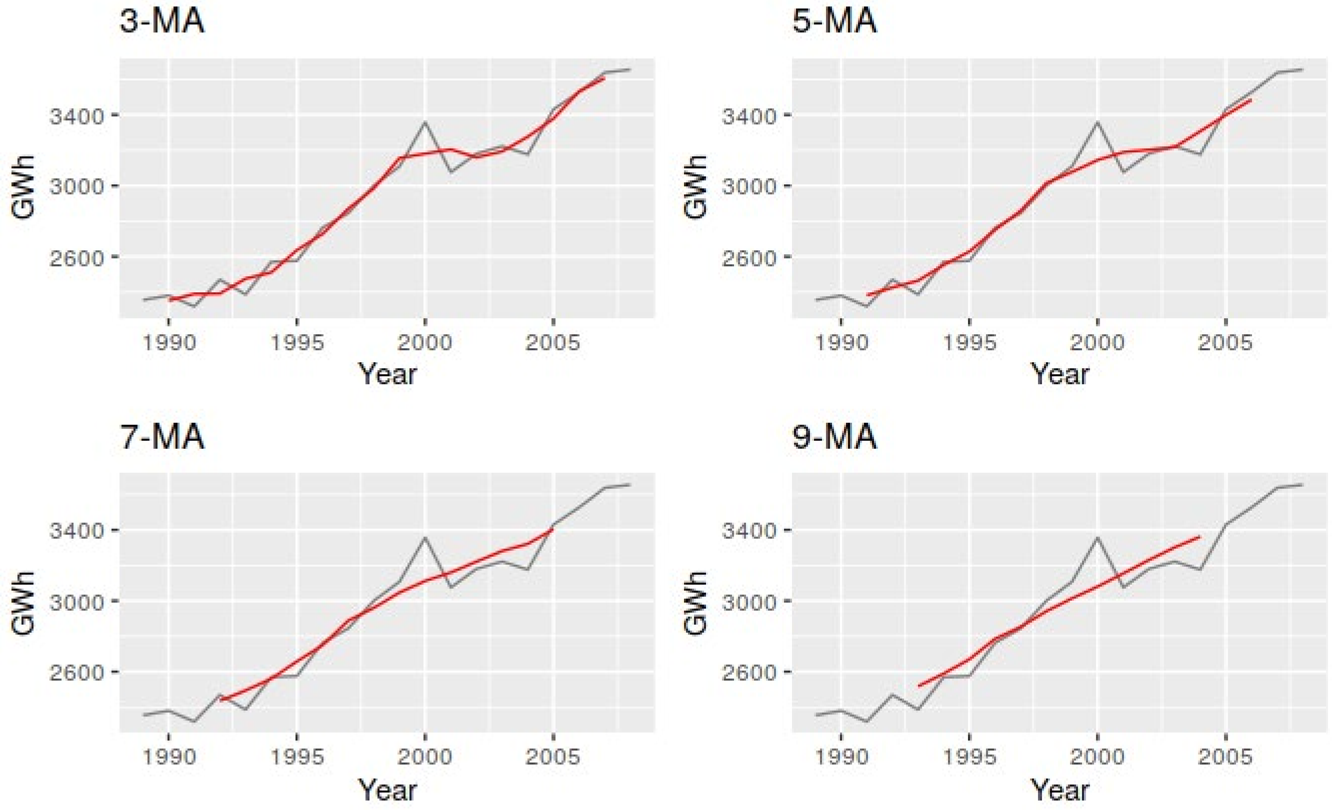
$$\hat{T}_t = \frac{1}{m}\sum_{i=-k}^{k} x_{t+k}$$

where $m = 2k + 1$

Moving Average Smoothing

Time Series Analysis

Simple Moving Average



Example: Different moving averages applied to the residential electricity sales data.

Source: https://otexts.com/fpp2/moving-averages.html

Further Study

- **Book**:
  - Zaki, M., & Meira, W. (2014). Data mining and analysis : Fundamental concepts and algorithms. New York: Cambridge University Press.
  - Christopher M. Bishop (2006). Pattern Recognition and Machine Learning. New York: Springer-Verlag.
  - Jeremy Watt, Reza Borhani & Aggelos K. Katsaggelos (2016). Machine Learning Refined: Foundations, Algorithm, and Application. New York: Cambridge University Press.
- Website
  - https://medium.com/swlh/an-introduction-to-time-series-analysis-ef1a9200717a
  - https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775
  - https://en.wikipedia.org/wiki/Autoregressive_model
  - https://otexts.com/fpp2/
  - https://online.stat.psu.edu/stat510/lesson/5/5.2