



# COS10022 – DATA SCIENCE PRINCIPLES

Dr Pei-Wei Tsai (Lecturer, Unit Convenor)  
ptsai@swin.edu.au, EN508d

SWIN  
BUR  
NE

SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY





# Week 12



**COS10022**  
**Revisit**





# ABOUT DATA SCIENCE

## Data scientists are in high demand

Data scientist job postings, per 1 million postings on Indeed



glassdoor Jobs Company Reviews Salaries Interviews Salary Calculator Write Review For Employers

Q Job Title, Keywords, or Company Jobs Location Search

### 50 Best Jobs in America

Best Jobs 2018 United States

Job Title	Median Base Salary	Job Satisfaction	Job Openings
#1 Data Scientist	\$110,000	4.2/s	4,524
#2 DevOps Engineer	\$105,000	4.0/s	3,369
#3 Marketing Manager	\$85,000	4.0/s	6,439

glassdoor Jobs Company Reviews Salaries Interviews Salary Calculator Write Review For Employers

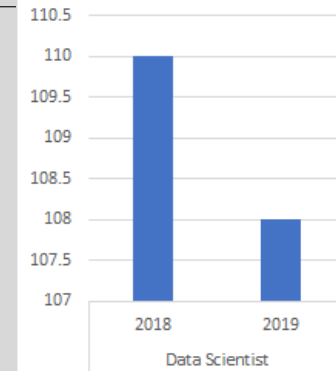
Q Job Title, Keywords, or Company Jobs Location Search

### 50 Best Jobs in America for 2019

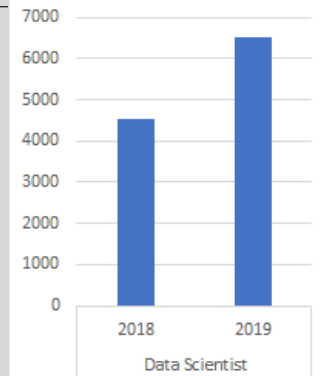
Best Jobs 2019 United States

Job Title	Median Base Salary	Job Satisfaction	Job Openings
#1 Data Scientist	\$108,000	4.3/s	6,510
#2 Nursing Manager	\$83,000	4/s	13,931
#3 Marketing Manager	\$82,000	4.2/s	7,395

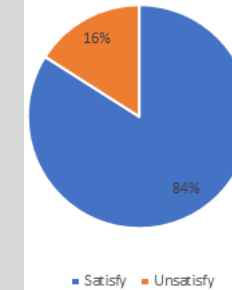
## Median Base Salary



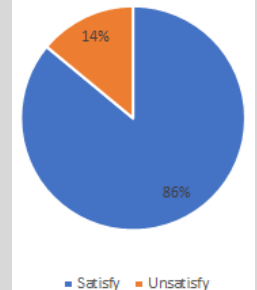
## Job Openings



## Job Satisfaction 2018



## Job Satisfaction 2019



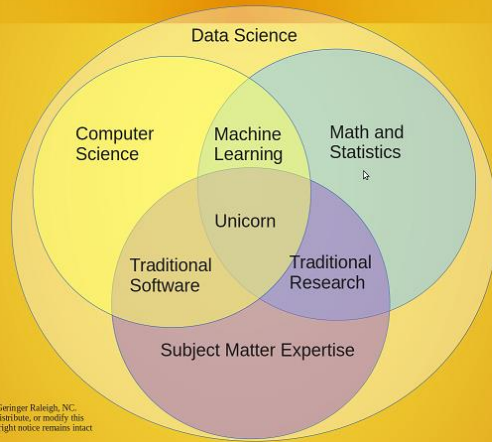
<https://www.hiringlab.org/2019/01/17/data-scientist-job-outlook/>, [https://www.glassdoor.com/List/Best-Jobs-in-America-2018-LST\\_KQ0,25.htm](https://www.glassdoor.com/List/Best-Jobs-in-America-2018-LST_KQ0,25.htm)

# About Data Scientist

- The demand for data scientist is still growing – Indeed 2018.
- Data scientist is the top ranked job in US in 2018 and 2019 – glassdoor data.



## Data Science Venn Diagram v2.0



Copyright © 2014 by Steven Goring Raleigh, NC.  
Permission is granted to use, distribute, or modify this  
image, provided that this copyright notice remains intact

	Data Analyst	Machine Learning Engineer	Data Engineer	Data Scientist
Programming Tools	Very important	Very important	Very important	Very important
Data Visualization and Communication	Very important	Somewhat important	Somewhat important	Very important
Data Intuition	Somewhat important	Very important	Somewhat important	Very important
Statistics	Somewhat important	Very important	Somewhat important	Very important
Data Wrangling	Not that important	Not that important	Very important	Very important
Machine Learning	Not that important	Very important	Not that important	Very important
Software Engineering	Not that important	Somewhat important	Very important	Somewhat important
Multivariable Calculus and Linear Algebra	Not that important	Very important	Not that important	Somewhat important

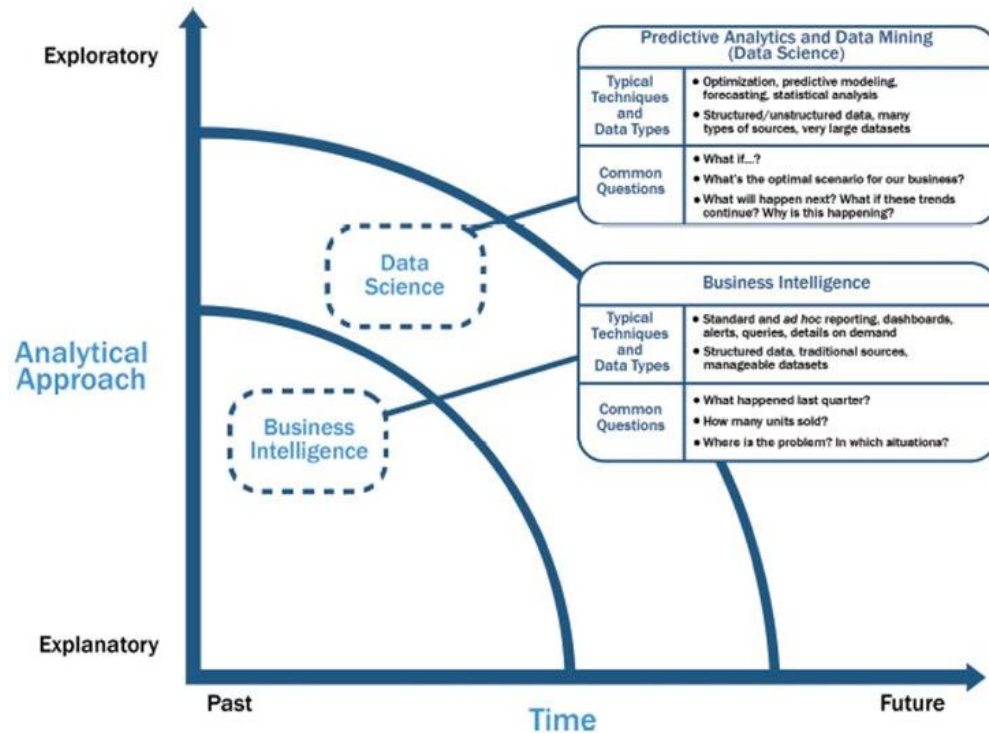
Not that important
Somewhat important
Very important



# Data Scientist

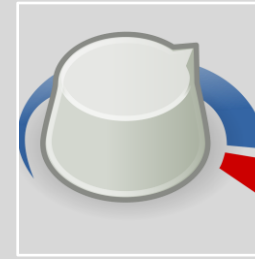
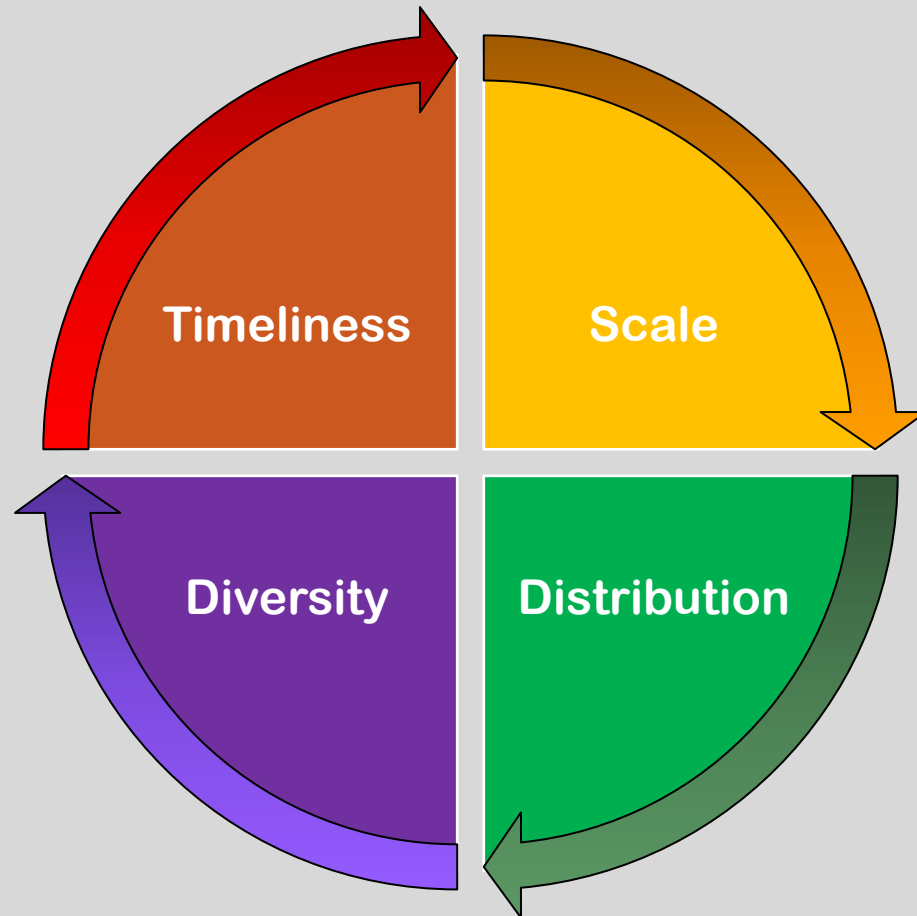
- Capable of analyzing and interpreting complex digital data to assist a business in its decision-making.

# Data Science and Business Intelligence



# DATA SCIENCE VS BUSINESS INTELLIGENCE

# Big Data 4 Vs Revisit



## Volume

- Scale

## Variety

- Diversity
- Distribution



## Velocity

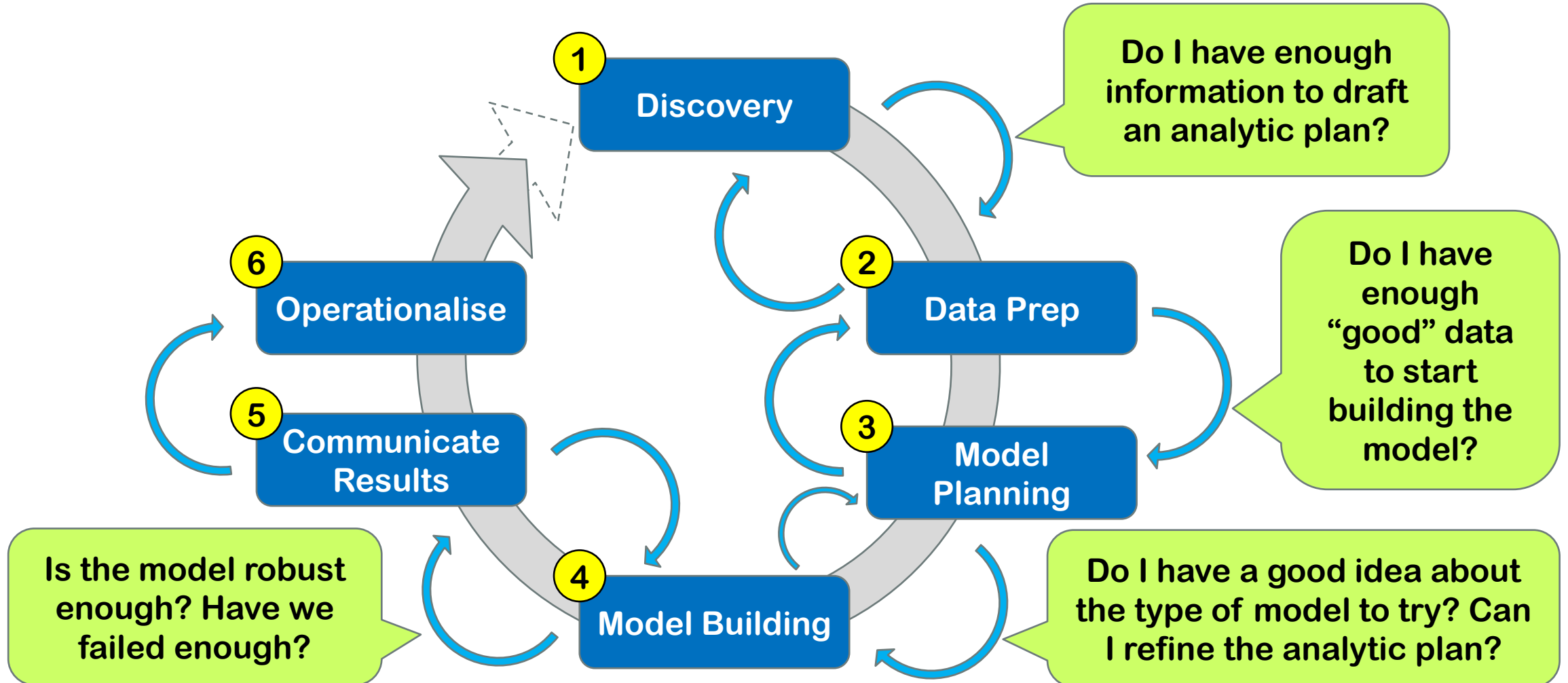
- Timeliness

## Veracity

- i.e. pertaining to the accuracy of data.



# Data Analytics Lifecycle





# Structured or Unstructured Data?

## Structured Data

- Structured data is arranged in a specific manner in tables.
- It is more suitable for structured database such as MySQL, PostgreSQL, etc.

## Semi-structured Data

- Semi-structured data is a form of structured data that does not obey the formal structure of data models associated with relational databases or other forms of data tables, but nonetheless contains tags or other markers to separate semantic elements and enforce hierarchies of records and fields within the data.

## Unstructured Data

- Unstructured data is information that either does not have a predefined data model or is not organised in a pre-defined manner.
- Unstructured data is typically text-heavy.
- Unstructured database such as MongoDB is generally used to store the unstructured data.



# DATA PREPARATION

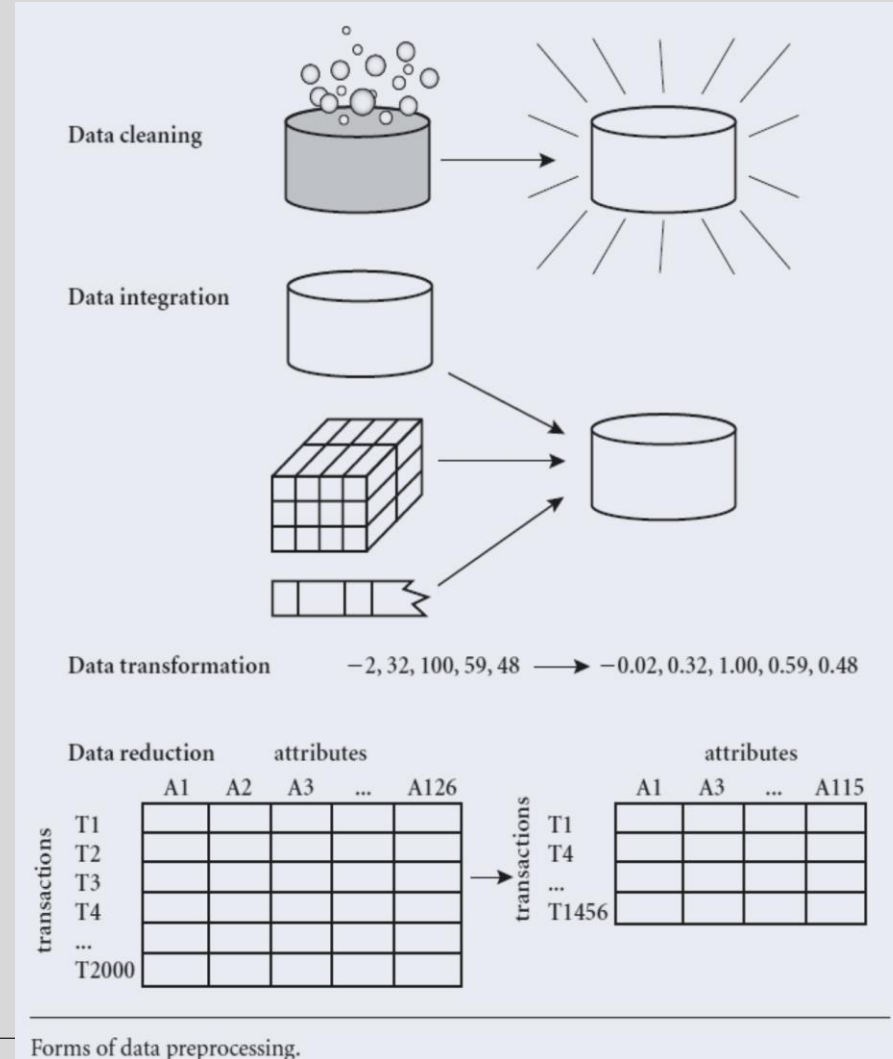
# Major Tasks in Data Preparation

## Data Cleaning

To fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.

## Data Transformation

To modify the source data into different formats in terms of data types and values so that it is useful for mining and to make the output easier to understand.



## Data Integration

To merge data from multiple data stores to help reduce redundancies and inconsistencies in the resulting dataset.

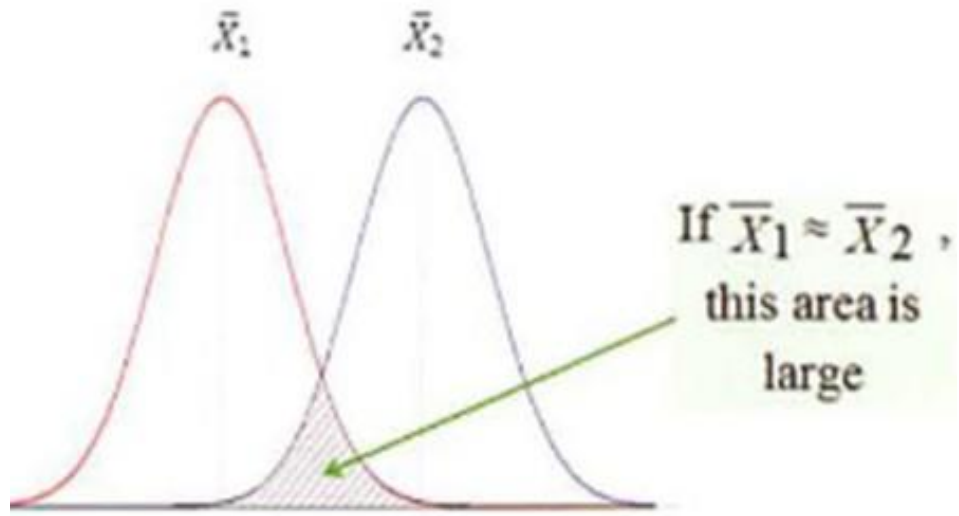
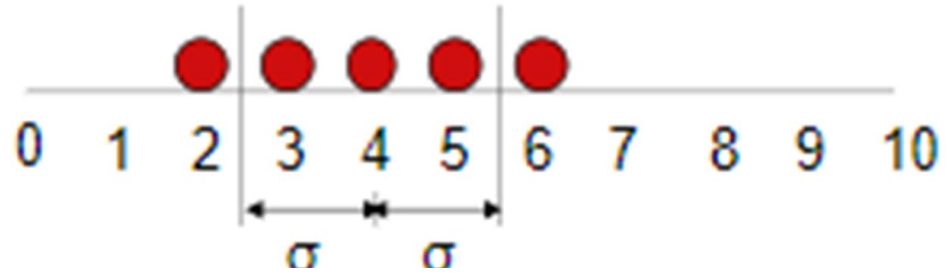
## Data Reduction

To obtain a reduced representation of the dataset that is much smaller in volume, yet produces the same (or almost the same) analytical results.



# Normalization

Min-Max	Z-score	Decimal scaling
Transforms the data into a desired range, usually [0, 1].	Useful when the actual min and max of attribute are unknown.	Transform data into a range between [-1, 1].
$v' = \frac{v - \min_A}{\max_A - \min_A} (\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_A$ <p>Where, <math>[\min_A, \max_A]</math> is the initial range and <math>[\text{new\_min}_A, \text{new\_max}_A]</math> is the new range.</p>	$v' = \frac{v - \mu_A}{\sigma_A}$ <p>Where <math>\mu_A</math> and <math>\sigma_A</math> are the mean and standard deviation of the initial data values.</p>	$v' = \frac{v}{10^j}$ <p>Where <math>j</math> is the smallest integer such that <math>\text{Max}( v' ) &lt; 1</math>.</p>
<p>Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to:</p> $\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0) + 0 = 0.716$	<p>Let <math>\mu = \\$54,000</math>, <math>\sigma = \\$16,000</math>. Then \$73,600 is transformed to:</p> $\frac{73,600 - 54,000}{16,000} = 1.225$	<p>Suppose that the values of A range from -986 to 917. Divide each value by 1000 (i.e. <math>j = 3</math>): -986 normalizes to -0.986 and 917 normalizes to 0.917.</p>



# Describing Your Data

- **Descriptive Statistics**

- Mean
- SD
- Median
- ...

- **Inferential Statistics**

- T-Test
- ANOVA
- ...

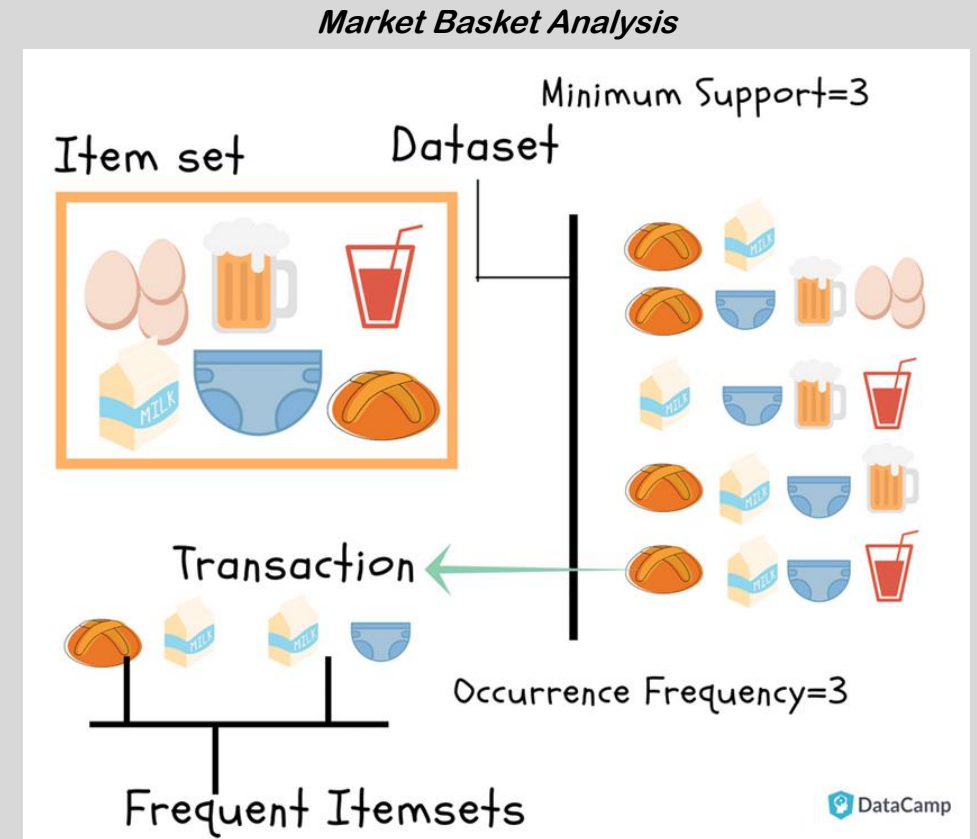


# ASSOCIATION RULES



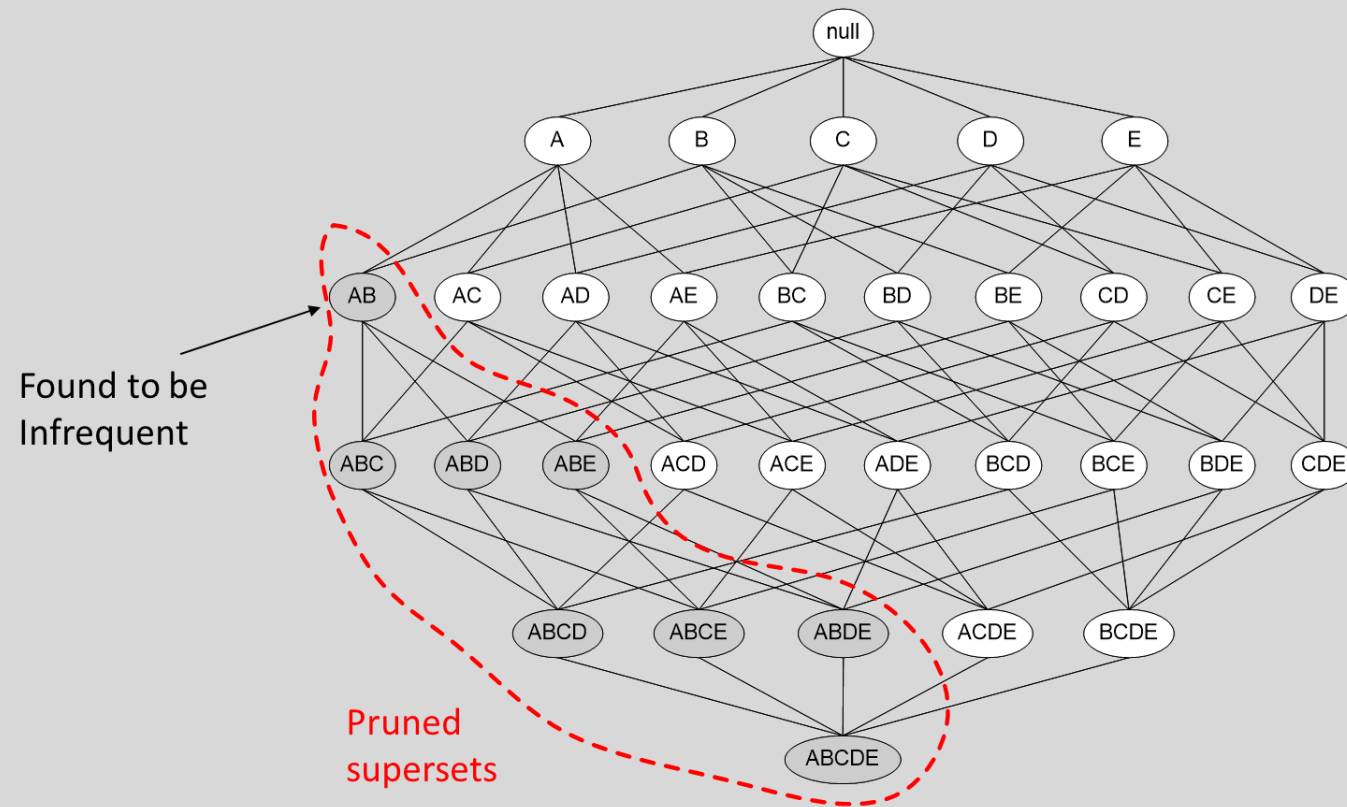
# Overview

- Association rules
  - An **unsupervised** learning method.
  - A **descriptive**, **not predictive**, method.
  - Used to discover **interesting hidden relationships** in a large dataset.
  - The disclosed relationships are represented as **rules** or **frequent itemsets**.
  - Commonly used for mining transactions in database.



# Apriori Algorithm

## Illustrating Apriori Principle



- Any subset of a frequent itemset must also be frequent.
- Itemsets that do not meet the minimum support threshold are **pruned** away.

# Evaluation of Candidate Rules

The process of creating association rules is two-staged.

- First, a set of candidate rule based on frequent itemsets is generated.
  - If {Bread, Egg, Milk, Butter} is the frequent itemset, candidate rules will look like:
    - {Egg, Milk, Butter} → {Bread}
    - {Bread, Milk, Butter} → {Egg}
    - {Bread, Egg} → {Milk, Butter}
    - Etc.
- Second, the appropriateness of these candidate rules are evaluated using:
  - Confidence
  - Lift
  - Leverage



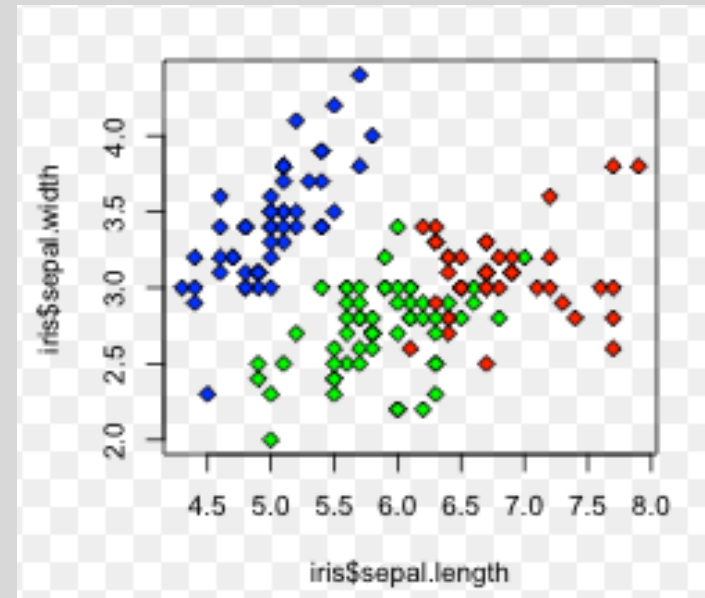


# K-MEANS CLUSTERING

# Exploratory model: K-Means Clustering

Exploratory = Unsupervised

- Clustering methods aim at discovering natural grouping of objects of interests (i.e. customers, images, documents, etc.).
- Generally, this objective is achieved through:
  1. Finding the similarities between the objects based on their attributes/properties/variables.
  2. Group similar objects into clusters.



# K-Means Clustering: choosing the value of $k$

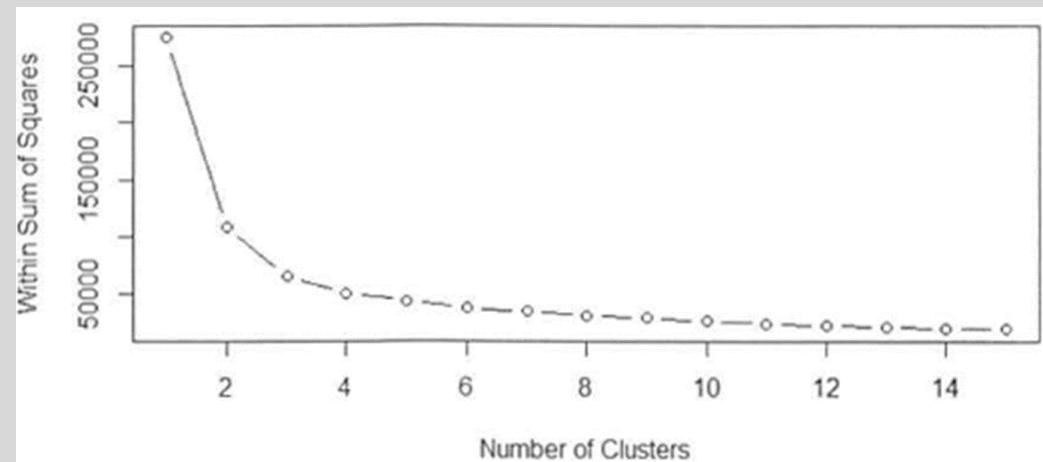
As mentioned before, the K-Means clustering model assumes that you already know the 'right' number of  $k$  clusters to be found before executing the clustering model.

In practice, the optimal value of  $k$  can be determined by either:

- a 'reasonable' guess;
- predefined requirements, e.g. a company wishes to segment its customers to exactly 5 clusters given its traditional way of grouping customers;
- using the **Within Sum of Squares (WSS)** metric as a heuristic. The basic idea is that if having  $k+1$  clusters does not greatly reduce the value of WSS compared to having  $k$  clusters, then there is little benefit to adding another cluster.

<http://www.learnbymarketing.com/methods/k-means-clustering/>

WSS: Sum of the squared difference from the cluster center





# K-Means Clustering: choosing the value of $k$

- Silhouette analysis can be used to determine the **degree of separation** between resulting clusters.
- The silhouette plot displays a measure of how close each point in one cluster is to points in the neighbouring clusters. For each sample:
  - Compute the average distance from all data points in the **same cluster** ( $a^i$ ).
  - Compute the average distance from all data points in the **closest cluster** ( $b^i$ ).
  - Compute the coefficient:

$$\frac{b^i - a^i}{\max(a^i, b^i)}$$

- The coefficient can take values in the interval  $[-1, 1]$ .
  - If it is 0 → the sample is very close to the neighbouring clusters.
  - If it is 1 → the sample is far away from the neighbouring clusters.
  - If it is -1 → the sample is assigned to the **wrong clusters**. (not yet converged.)
- A good cluster has a coefficient close to 1.



# LINEAR REGRESSION

# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

Given the following  
data:

Table 1. Example data.

X	Y
1.00	1.00
2.00	2.00
3.00	1.30
4.00	3.75
5.00	2.25

**Task.**

Build a simple Linear Regression model that predicts the value of Y when the value of X is known.

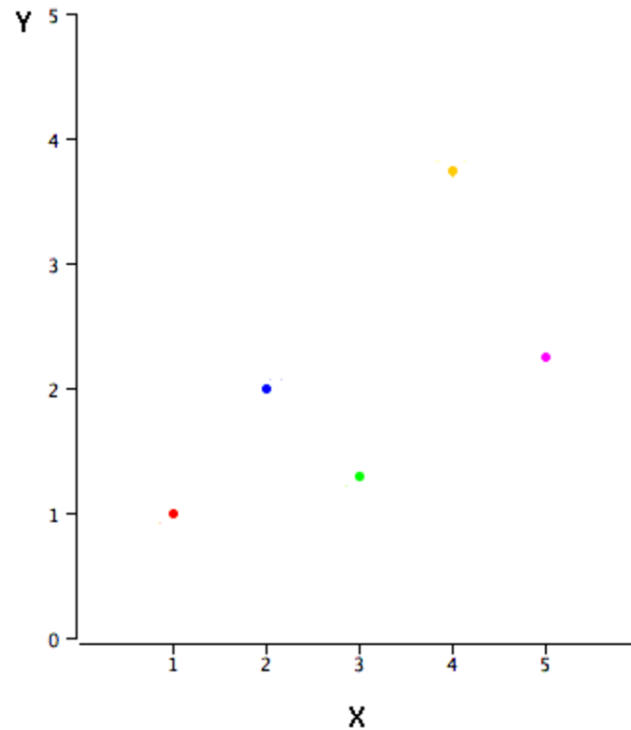


Figure 1. A scatter plot of the example data.

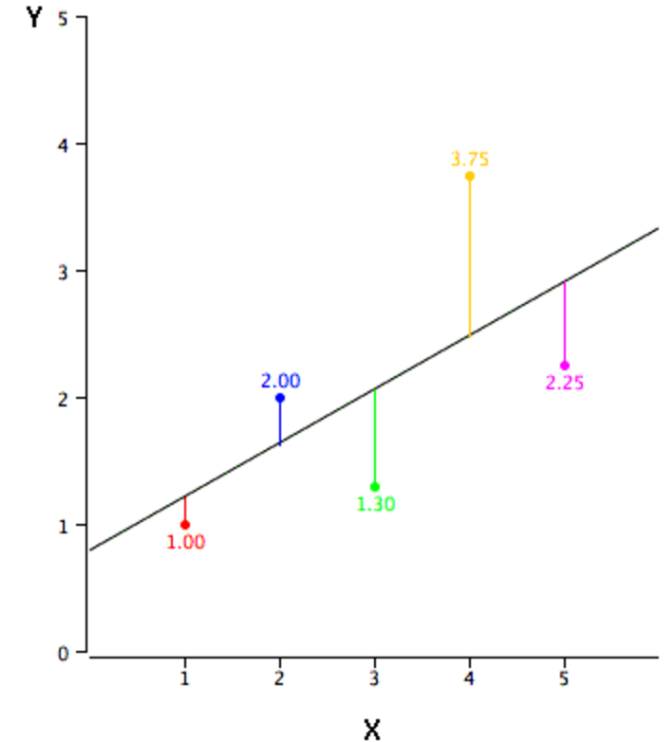


Figure 2. A scatter plot of the example data. The black line consists of the predictions, the points are the actual data, and the vertical lines between the points and the black line represent errors of prediction.



## Outliers:

A data point that differs significantly from other data points.

# Anscombe's Quartet

Four (4) datasets with nearly identical descriptive statistics (*mean, variance*) but strikingly different shapes when graphed.

Property	Value	Accuracy
Mean of $x$	9	exact
Sample variance of $x$	11	exact
Mean of $y$	7.50	to 2 decimal places
Sample variance of $y$	4.125	plus/minus 0.003
Correlation between $x$ and $y$	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively



Source: [https://en.wikipedia.org/wiki/Anscombe's\\_quartet](https://en.wikipedia.org/wiki/Anscombe's_quartet)



# NAÏVE BAYES CLASSIFIER

# Bayes' Theorem

The **Bayes' Theorem** is algebraically derived from the previous conditional probability formula to obtain:

$$P(C | A) = \frac{P(A | C) \cdot P(C)}{P(A)}$$

$$P(C|A) = \frac{P(A \cap C)}{P(A)} \dots\dots\dots (1)$$

$$P(A|C) = \frac{P(A \cap C)}{P(C)}$$
$$P(A \cap C) = P(A|C) \cdot P(C) \dots (2)$$

where:

$$C \in \{c_1, c_2, \dots, c_n\}$$

$A$  is the class label  $A \in \{a_1, a_2, \dots, a_m\}$


is the observed attributes,



# Naïve Bayes Classifier

Following the two simplifications above, calculating the probability of a class label  $c_i$  given a set of attributes  $a_1, a_2, \dots, a_m$  is proportional to the product of  $P(a_j | c_i)$  multiplied by  $P(c_i)$ , as shown below:

$$P(c_i | A) \propto P(c_i) \cdot \prod_{j=1}^m P(a_j | c_i), \quad i = 1, 2, \dots, n$$

 LHS: the probability of a class label  $c_i$  given a set of attributes  $a_1, a_2, \dots, a_m$

 RHS: the product of  $P(a_j | c_i)$  multiplied by  $P(c_i)$

 This symbol means 'directly proportional to'

# Supervised Models: Metrics and Methods

Popular metrics for evaluating the performance of supervised models:

1. **Accuracy**
2. **TPR**
3. **FPR (Type 1 Error Rate)**
4. **FNR (Type 2 Error Rate, Miss Rate)**
5. **Precision**
6. **Area Under the Curve (AUC)**

These metrics can be calculated by utilizing a **confusion matrix**.

# Supervised Models: Metrics and Methods

**Example 1.** A confusion matrix of Naïve Bayes classifier for 100 customers in predicting whether they would subscribe to the term deposit (refer to the Naïve Bayes example in the previous lecture).

		Predicted Class		Total
		Subscribed	Not Subscribed	
Actual Class	Subscribed	3 ( <b>correct prediction</b> )	8 ( <b>error</b> )	11
	Not Subscribed	2 ( <b>error</b> )	87 ( <b>correct prediction</b> )	89
Total		5	95	100





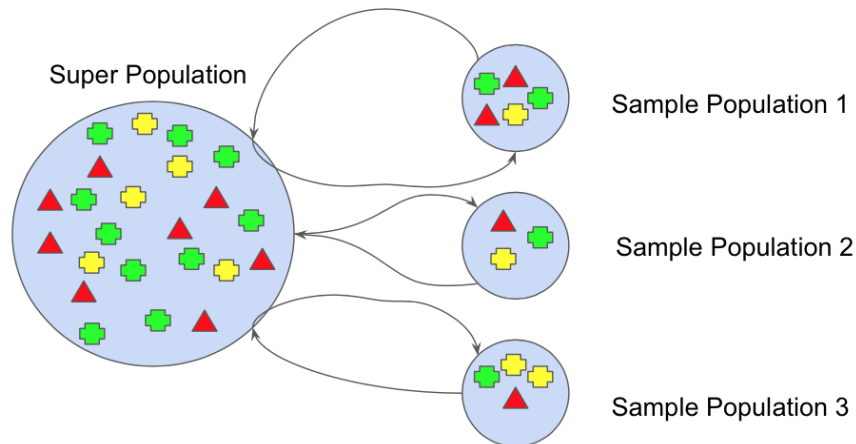
# DECISION TREE





# ENSEMBLE LEARNING

# Bootstrapping



## Bootstrap refers to

- Random sampling with replacement.

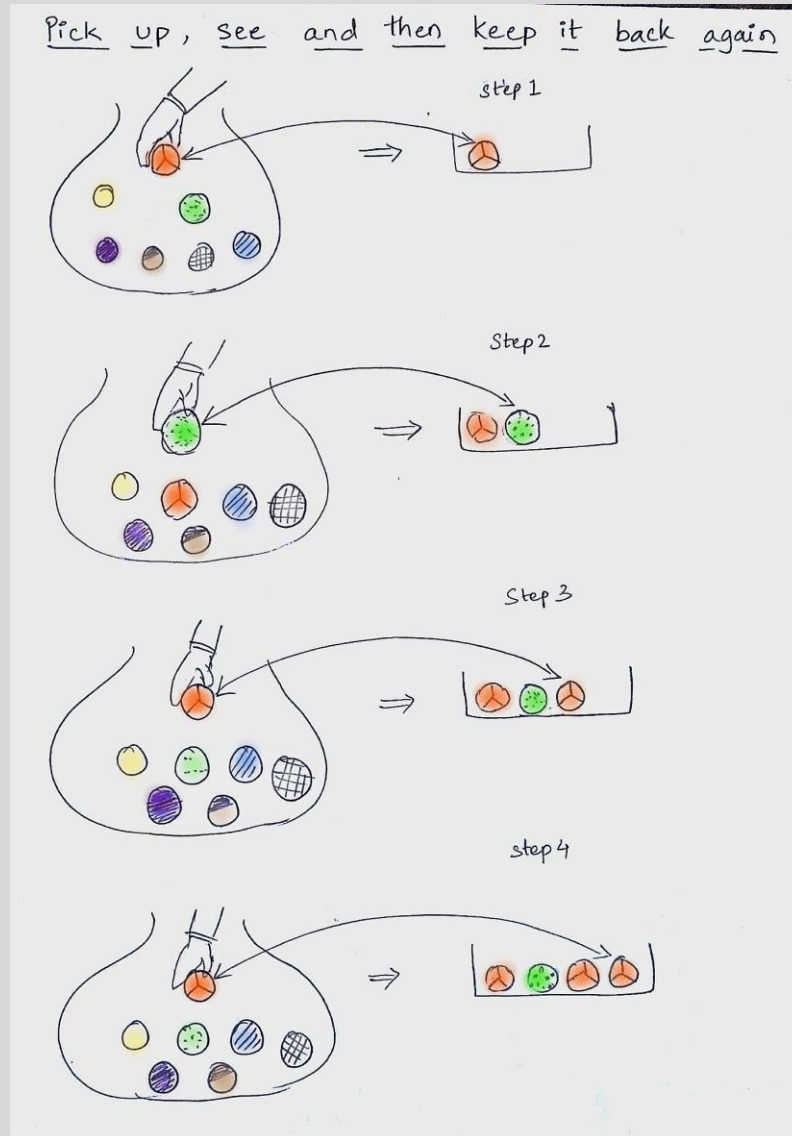
## Strong points

- To better understand the bias, mean, the variance, and the standard deviation from the dataset.

## Key operations

- Random sampling of small subset of data from the dataset.
- The subset can be replaced.
- The selection of all examples in the dataset has equal probability.

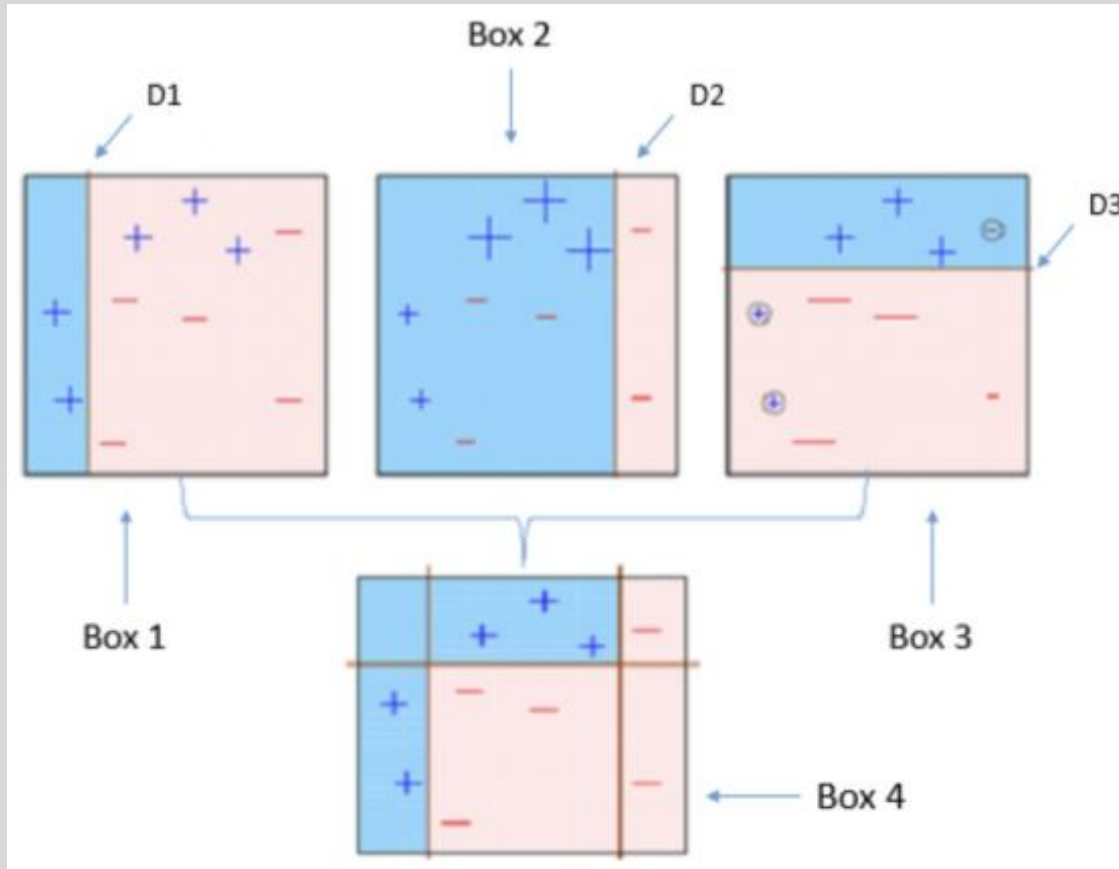




# Bagging

- **Bagging** is also known as **Bootstrap Aggregation**.
- A simple but a powerful ensemble method.
- It is typically an application of the Bootstrap procedure to **decision tree**, which is a high-variance machine learning algorithm.
- In bagging, training instances can be **sampled several times for the same predictor**.

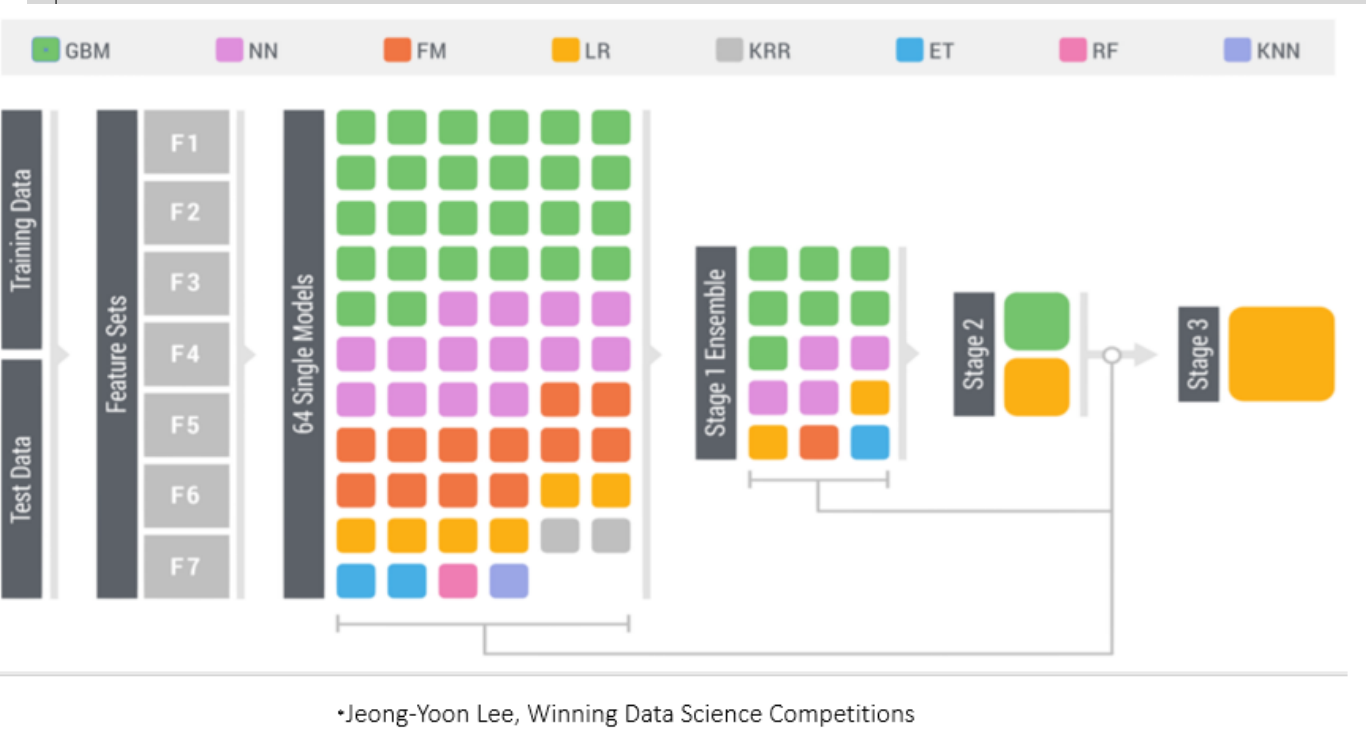
# Boosting



- In boosting algorithms, they utilise the weighted averages to mark weak learners into strong learners.
- Different from bagging, which runs each model independently before aggregating the outputs at the end without preference to any model, boosting is all about “teamwork.”
- Each model that runs in boosting, dictates what features the next model will focus on.

# Stacking

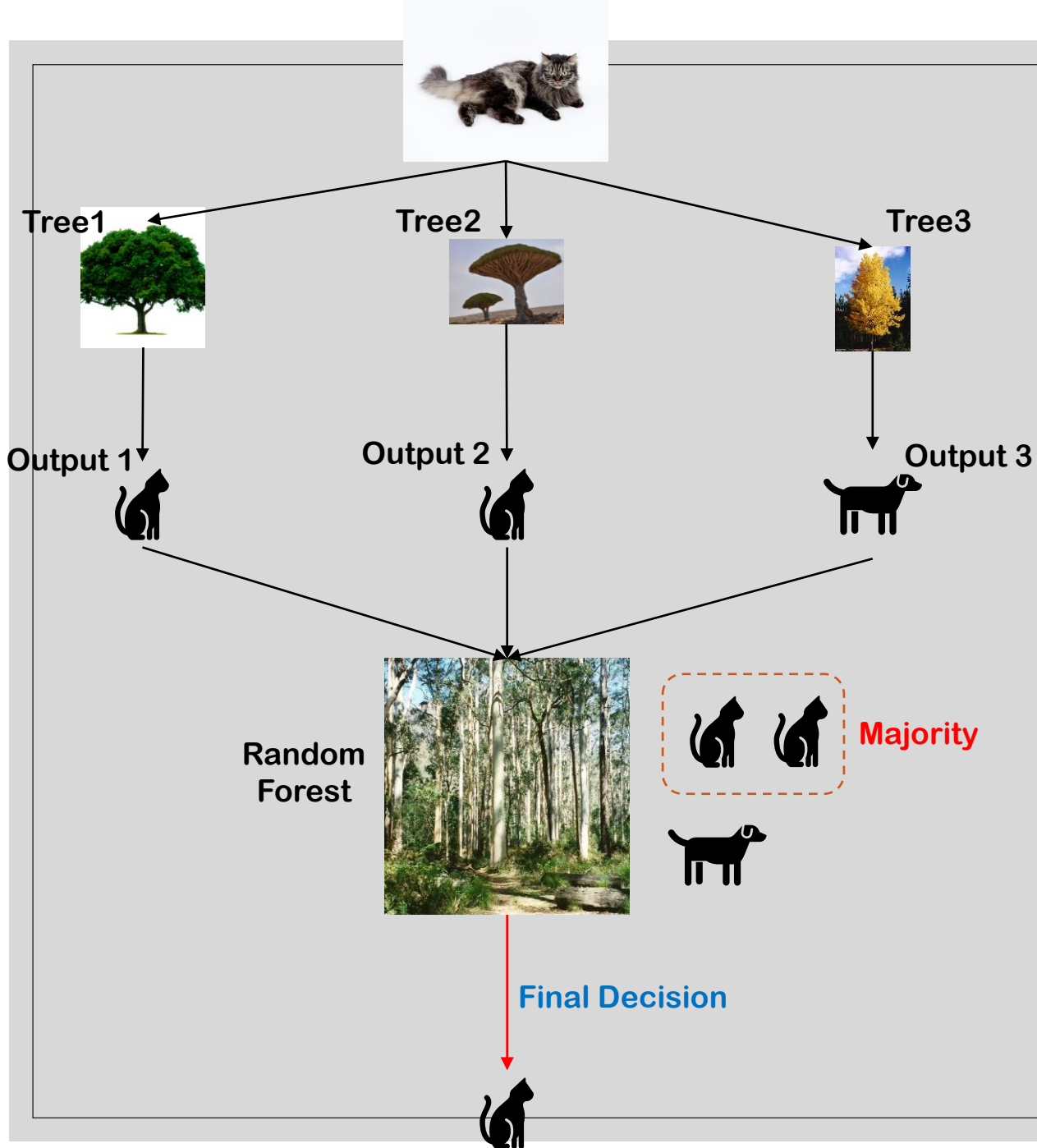
- On the popular data science competition site Kaggle (<https://www.kaggle.com/>), you can explore numerous winning solutions through its discussion forums to get a flavour of the state of the art.
- Another popular data science competition is the KDD Cup (<http://www.kdd.org/kdd-cup>). The figure shown on the left is the winning solution for the 2015 competition, which used a three-stage stacked modelling approach.







# RANDOM FOREST



# What is Random Forest?

## Definition

- Random Forest or Random Decision Forest is a method that operates by constructing multiple Decision Trees during training phases.

## Operation

- It's an application of stacking.

# Why using Random Forest?

## No Overfitting

- Use of multiple trees to reduce the risk of overfitting.
- Less training time.

## High Accuracy

- Runs efficiently on large dataset.
- Produces highly accurate predictions in large dataset.

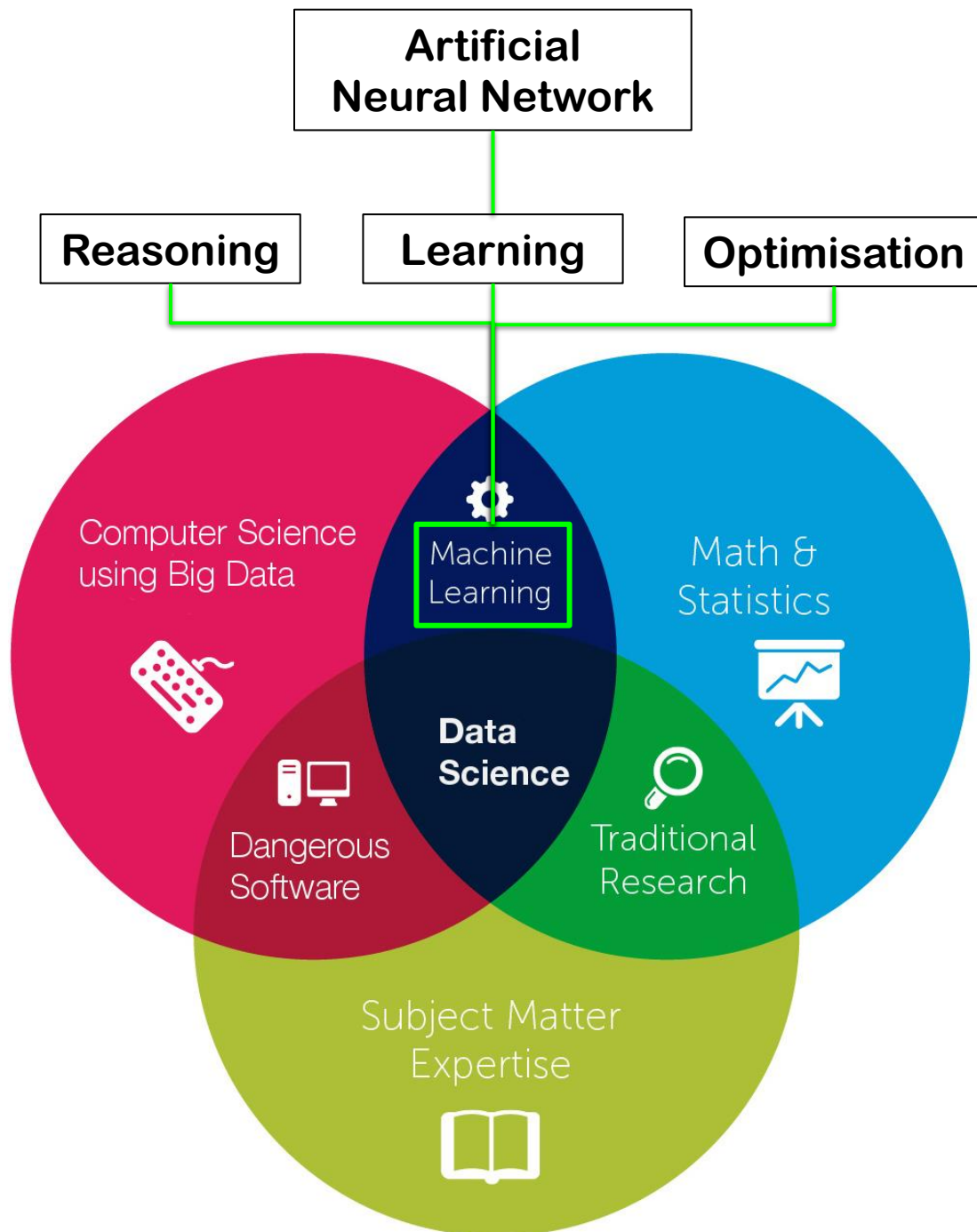
## Estimates missing data

- Random Forest can maintain accuracy when a large proportion of data is missing.





# ARTIFICIAL NEURAL NETWORK

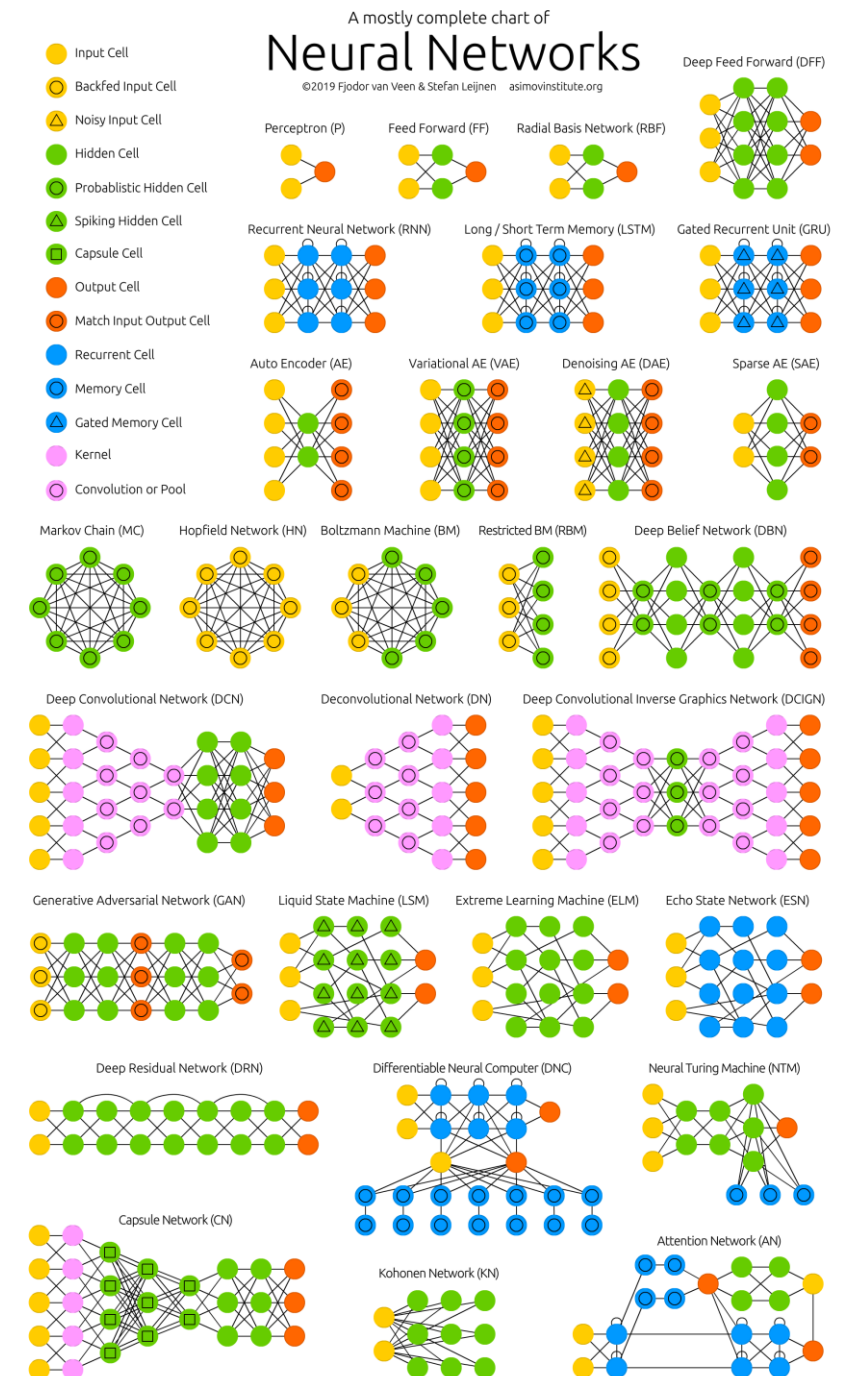


**WHERE  
DOES  
ANN  
STAND?**



# TYPES OF ARTIFICIAL NEURAL NETWORKS

- Artificial Neural Network (ANN) can be considered as a set of nodes (neurons) combined with the designed patterns.
- The weights on the connections between nodes are adjusted in the training process.
- Alike other classifiers introduced before,
- In IDS, we will focus on the shallow networks instead of the deep structures.





# Parameters (Coefficients) of a Perceptron

$$\omega = \begin{bmatrix} \omega_0 \\ \omega_1 \\ \vdots \\ \omega_n \end{bmatrix}$$

$\omega$ : A vector, which length is the no. of dimension you have plus one.

$\nu$ : The step size controlling how fast the learning process will terminate.

A constant  $X_j = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix},$   
 $j = 1, \dots, m$  and  
 $X = \{X_1, \dots, X_m\}$

Input

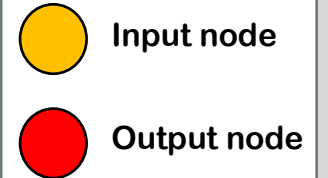
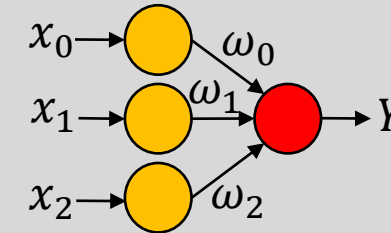
Weights

Learning Rate

Input Data

Output

Final result



$$y = \omega^T X = \sum_{i=0}^n \omega_i \cdot x_i$$

$$Y = \begin{cases} \text{green circle}, & \text{if } y > 0 \\ \text{red X}, & \text{if } y \leq 0 \end{cases}$$

$$X_j = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}, \text{ where } x_0 = 1$$

# Let's Get Back to the Given Example

## Update

- Update  $\omega$  to adjust the line into the right location by  $\omega'_i = \omega_i + \sum_{j=1}^m v \cdot d_j \cdot x_i$ , where  $\omega'_i$  and  $\omega_i$  represents the new and the old  $\omega$ , respectively.

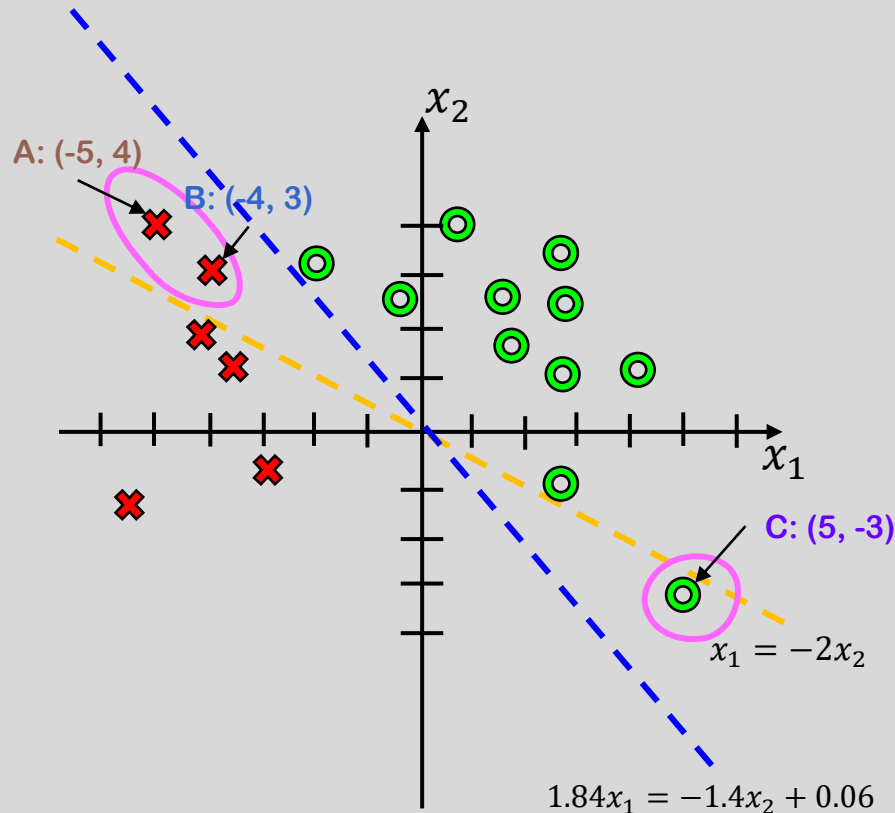
$$d_j = \begin{cases} 1, & \text{if } X_j \text{ should be in the upper plan} \\ -1, & \text{if } X_j \text{ should be in the lower plan} \\ 0, & \text{if } X_j \text{ is already in the correct plan} \end{cases}$$

$$\omega'_0 = \omega_0 + [(0.06)(-1)(1)] + [(0.06)(-1)(1)] + [(0.06)(1)(1)] = -0.06$$

$$\omega'_1 = \omega_1 + [(0.06)(-1)(-5)] + [(0.06)(-1)(-4)] + [(0.06)(1)(5)] = 1.84$$

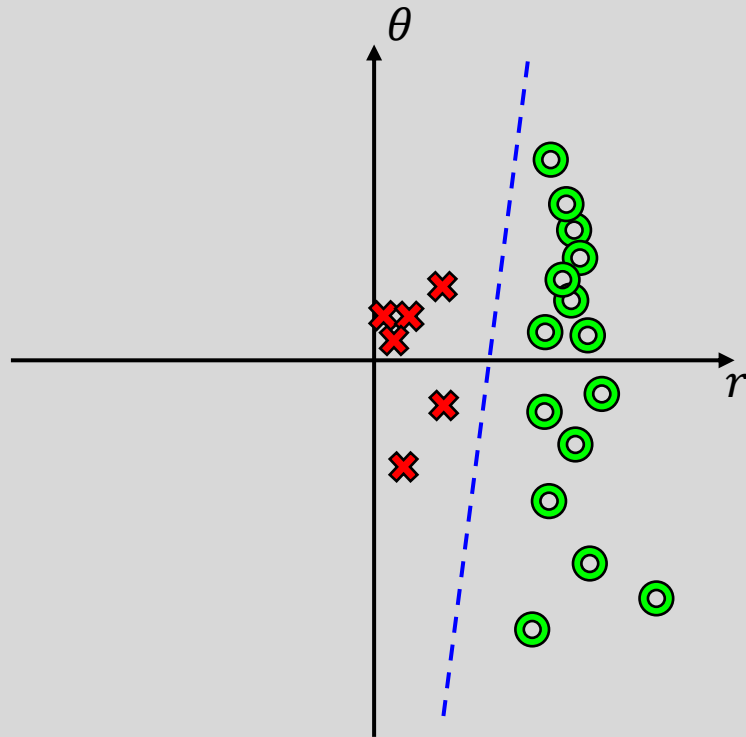
$$\omega'_2 = \omega_2 + [(0.06)(-1)(4)] + [(0.06)(-1)(3)] + [(0.06)(1)(-3)] = 1.4$$

$$\rightarrow y = \omega^T X = \begin{bmatrix} 0 & 1.84 & 1.4 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = 1.84x_1 + 1.4x_2 - 0.06$$



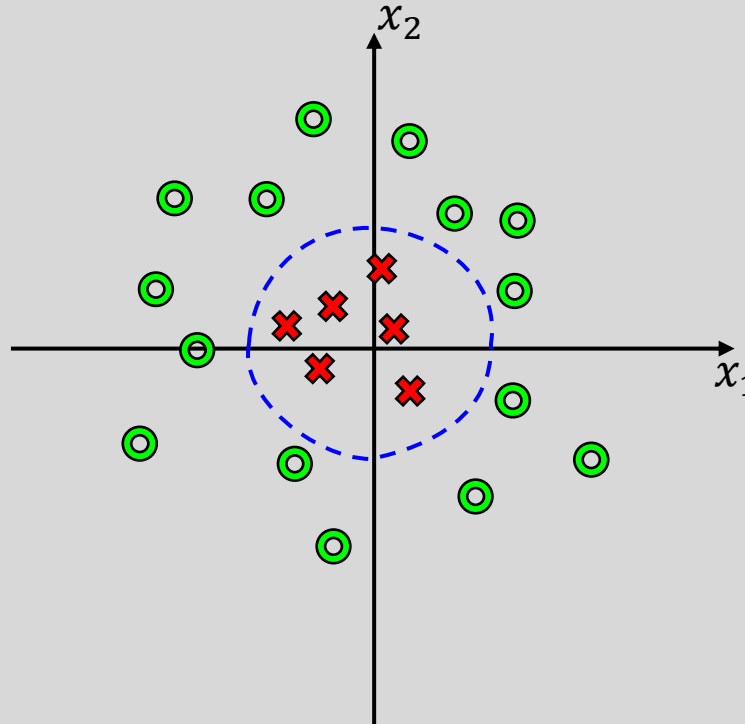
# What About the Not Linearly Separable Case?

Convert to Different  
Coordination System

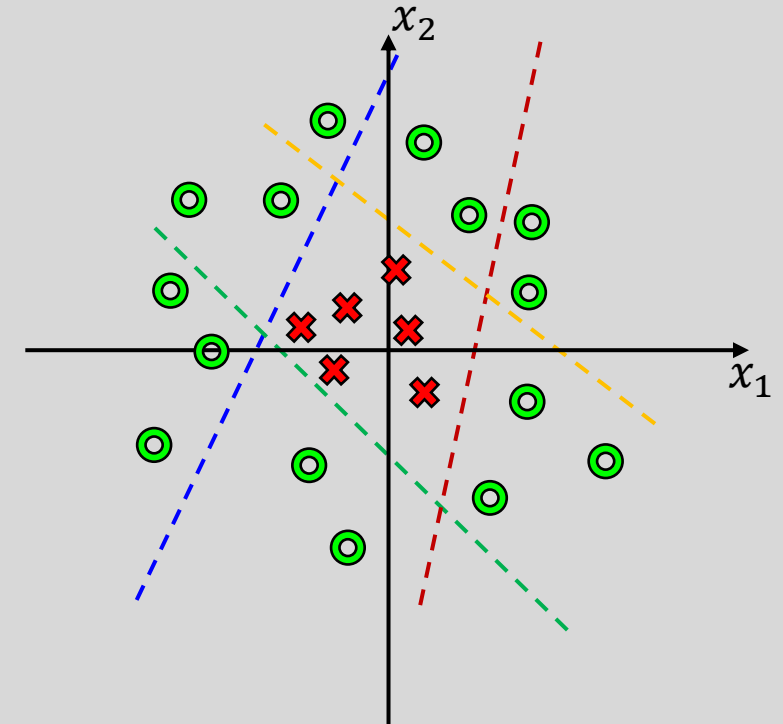


$(x_1, x_2) \rightarrow (r, \theta)$  in Polar Coordination

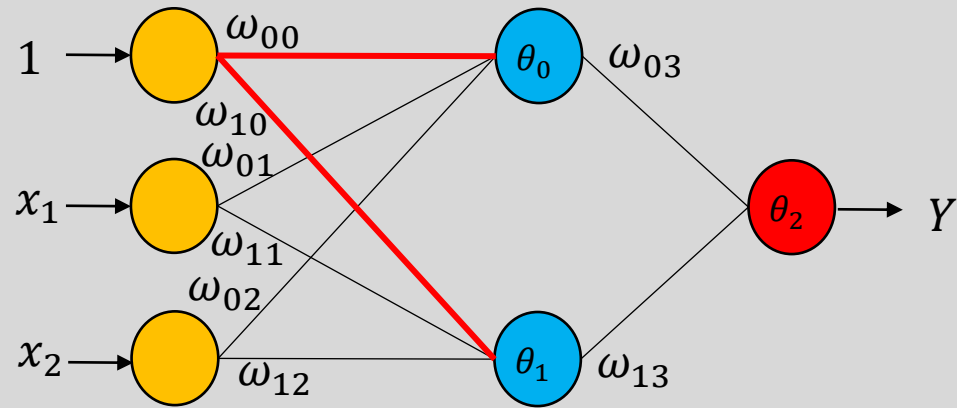
Not Linearly Separable



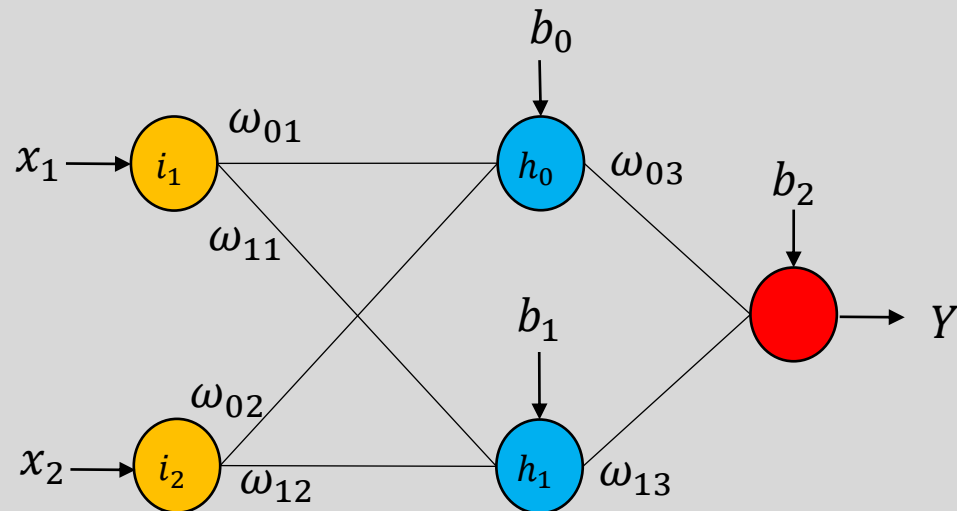
Stacking of Perceptrons



# Example 2-3: XOR Gate



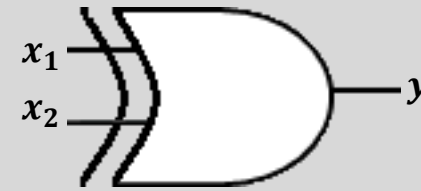
||



$$h_m = \sum_{k=1}^n i_k \times \omega_{mk} + b_m$$

If  $\omega = \begin{bmatrix} b_0 \\ \omega_{01} \\ \omega_{02} \\ \omega_{03} \\ b_1 \\ \omega_{11} \\ \omega_{12} \\ \omega_{13} \\ b_2 \end{bmatrix} = \begin{bmatrix} -10 \\ 20 \\ 20 \\ 20 \\ 30 \\ -20 \\ -20 \\ 20 \\ -30 \end{bmatrix}$ , the result values will be

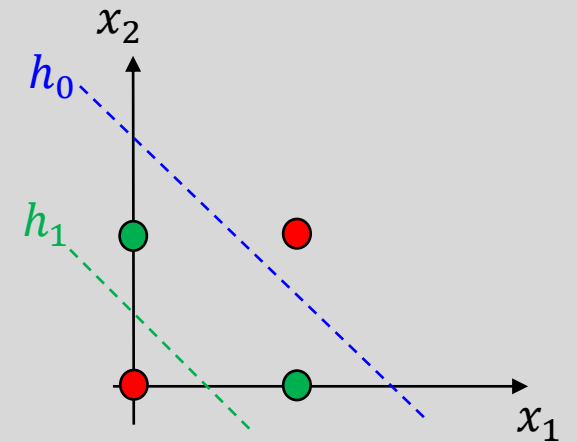
$4.5 \times 10^{-5}$  and  $9.9 \times 10^{-1}$ .



$$y = x_1 \oplus x_2$$

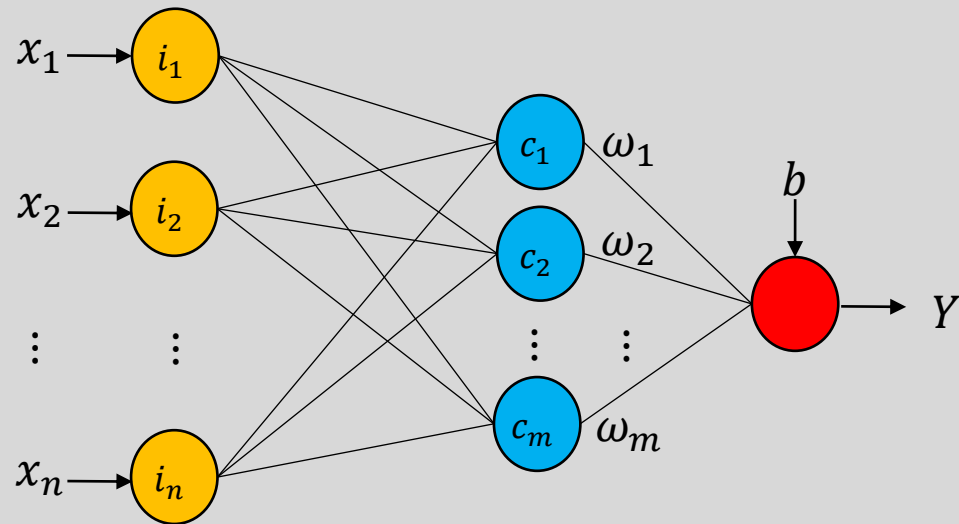
Input		Output
$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

$h_0 = \text{sigmoid}(\omega_{01}x_1 + \omega_{02}x_2 + b_0) = 1.5$   
 $h_1 = \text{sigmoid}(\omega_{11}x_1 + \omega_{12}x_2 + b_1) = 0.5$   
 $\rightarrow$  For  $h_0: x_1 + x_2 \approx 1.5$   
 For  $h_1: x_1 + x_2 \approx 0.5$





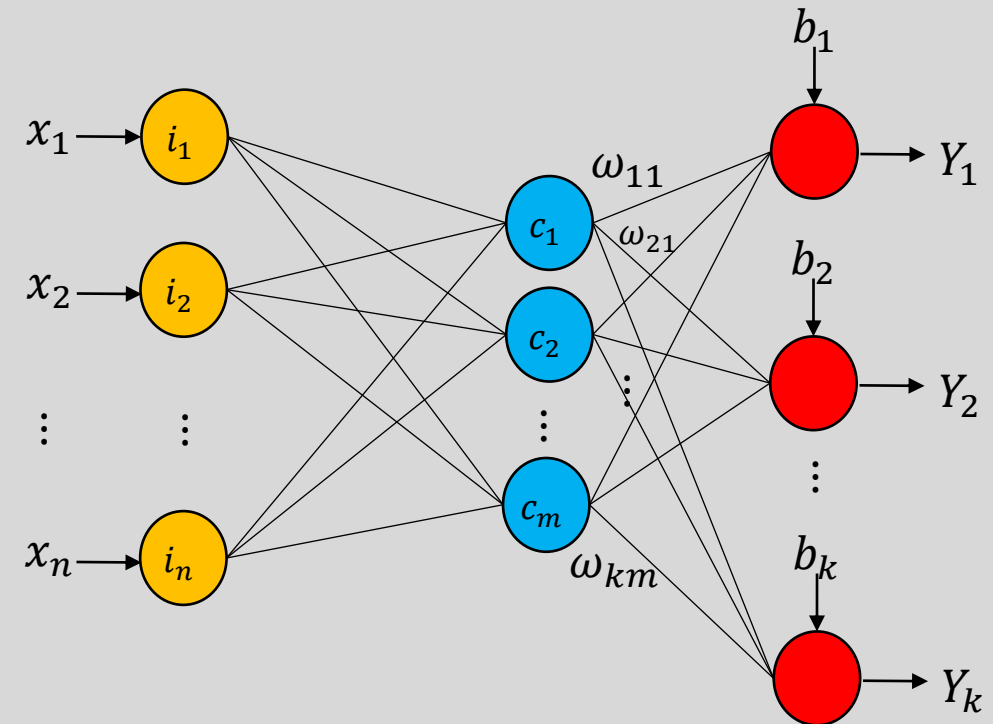
# RBF Model

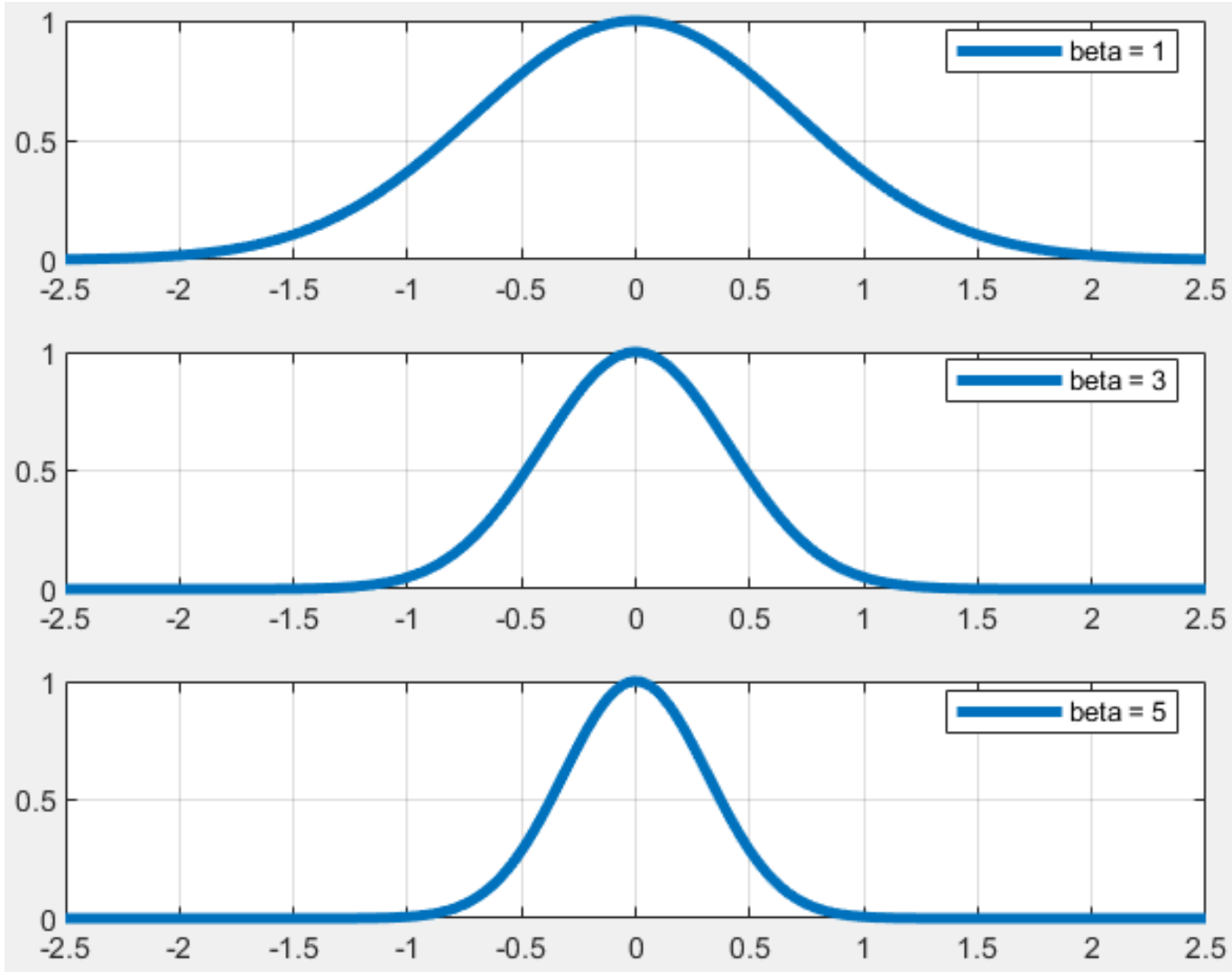


$$c_j = e^{-\beta \cdot D(x, c_j)^2}$$

$$Y = \sum_{j=1}^m \omega_j \cdot c_j + b$$

- If you have more outputs, you'll have multiple output nodes.

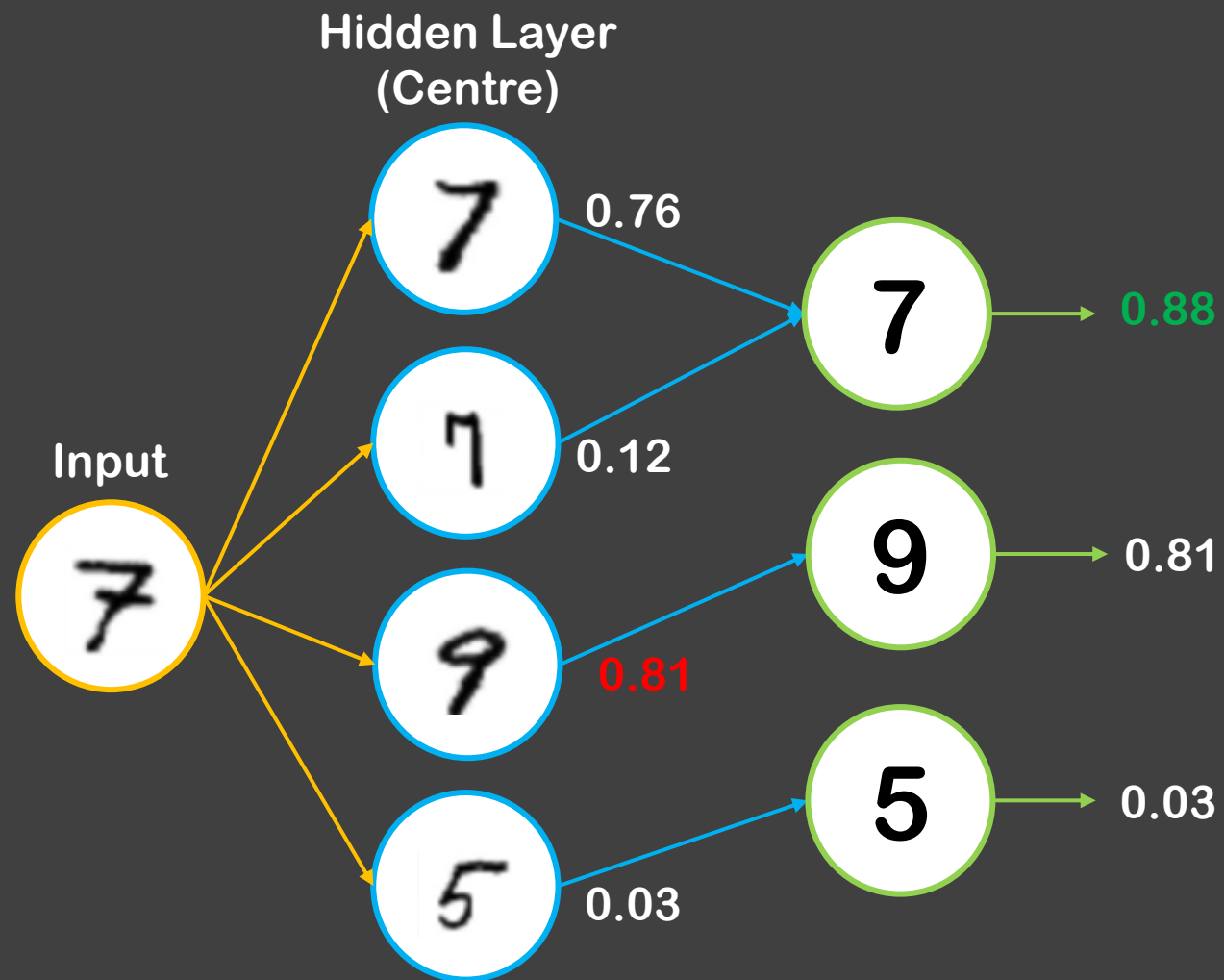




## RBF: How to decide the radius of a circle?

- $\phi(D) = e^{-\beta D^2}$   
where  $D$  is the distance between the data point to the centre of a circle and  $\beta$  is the parameter controls the radius of the circle.

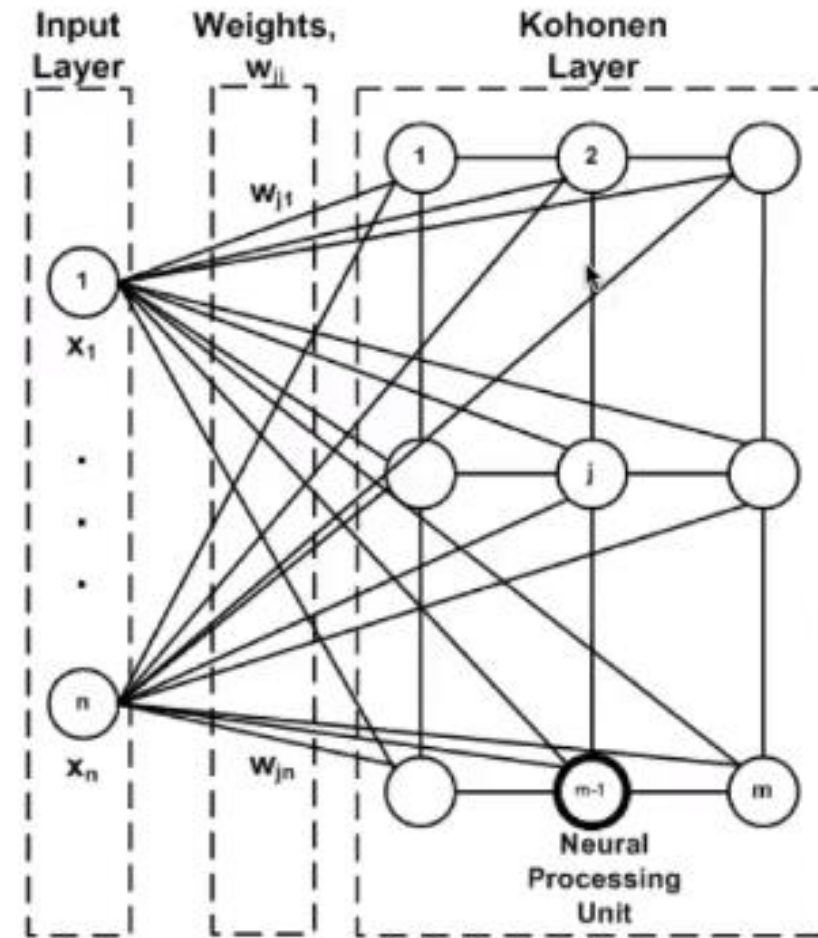
**RBF NN  
EXAMPLE:  
HAND-WRITTEN  
DIGIT  
RECOGNITION**





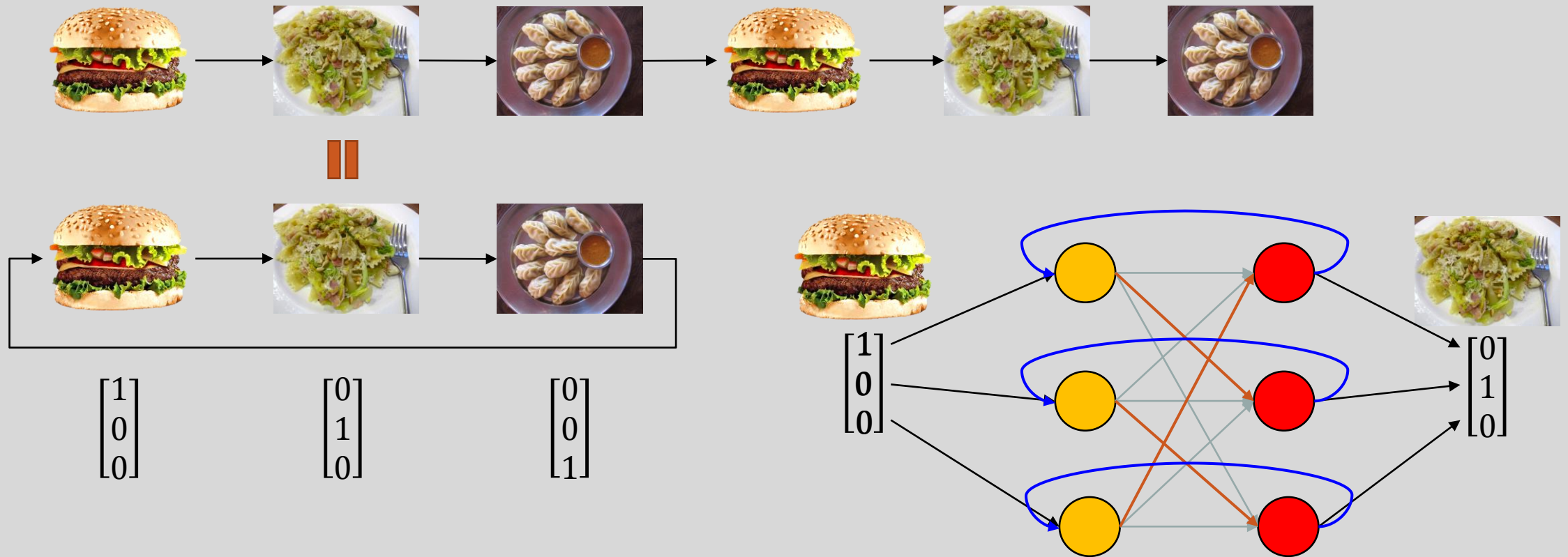
# SELF-ORGANISING MAPS

- The Kohonen layer is a 2-D plane for displaying the results.
- All nodes in the input layer are fully connected to nodes in every Kohonen layer.
- By adjusting the weights in the learning process, data with similar features will be sent to the similar location.

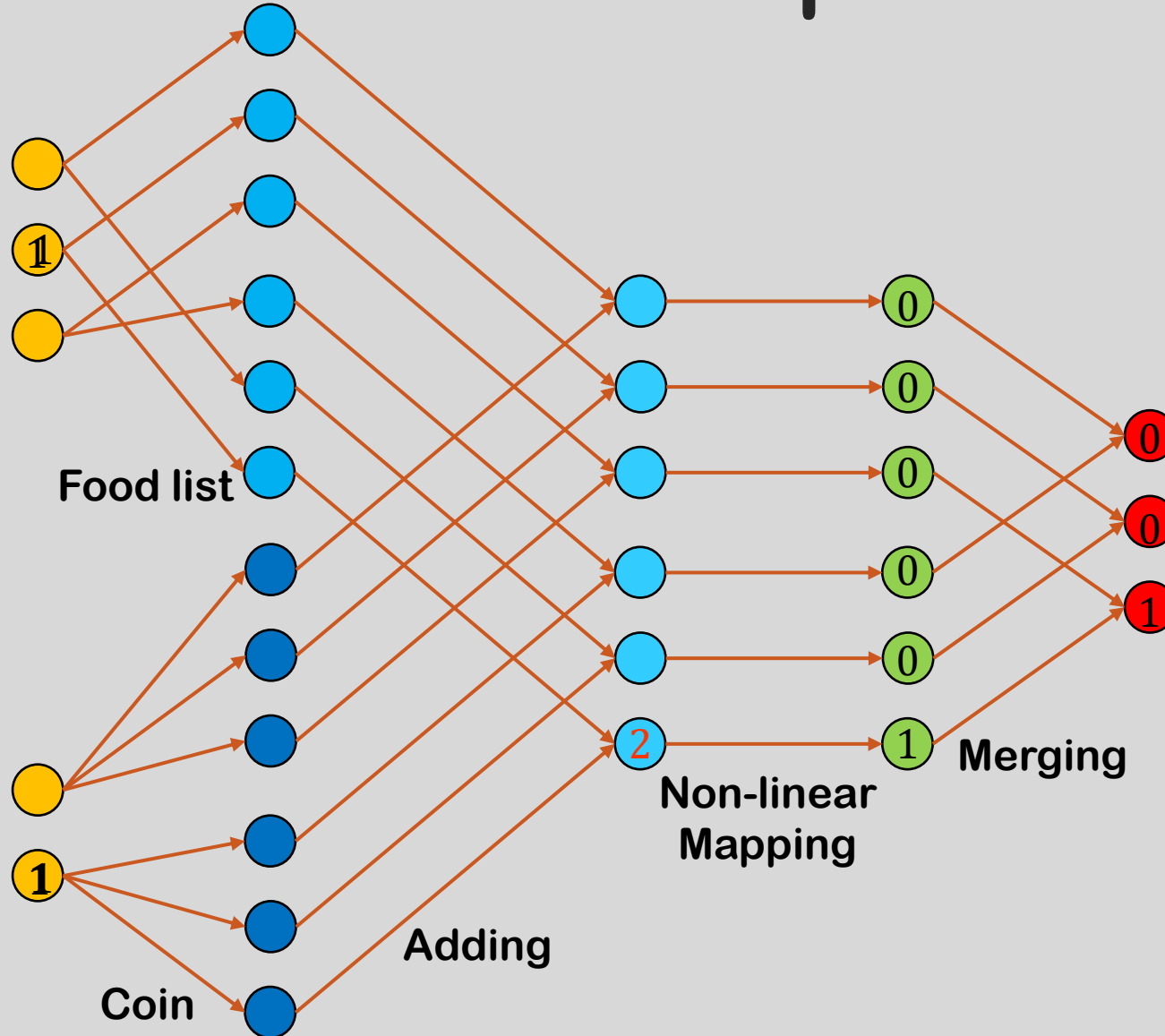


Araujo, Ernesto & R. Silva, Cassiano & J. B. S. Sampaio, Daniel. (2008). Video Target Tracking by using Competitive Neural Networks. WSEAS Transactions on Signal Processing. 4.

# Let's Learn RNN from Examples



# Let's Learn RNN from Examples

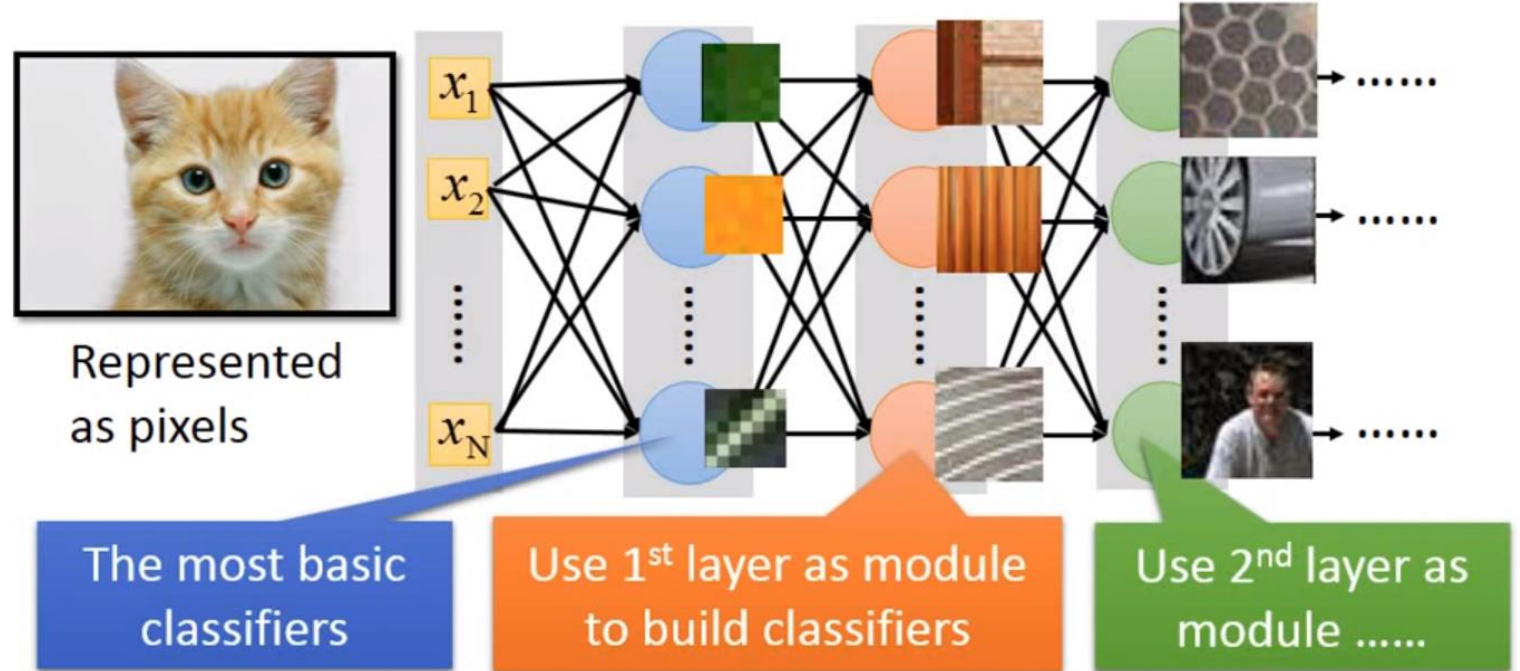

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$



# CNN Structure Overview

- Each neuron in the network is expected to be a simple classifier.
- Along with the layers, a single neuron is more specialised in recognising particular information.



# CNN Structure



Convolution

## Property 1

- Some patterns are much smaller than the whole image. (Localisation)

## Property 2

- The same patterns appear in different regions.

Max Pooling

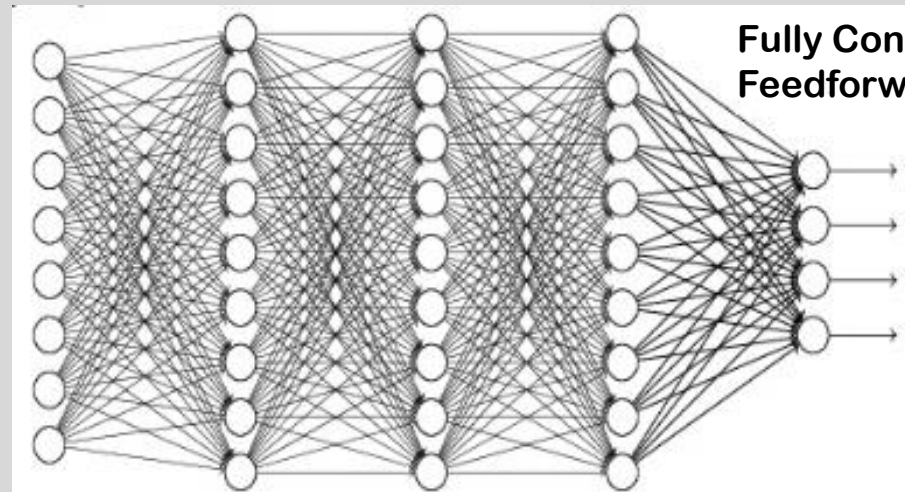
## Property 3

- Subsampling the pixels will not change the object.

Convolution

Max Pooling

Flatten



Fully Connected  
Feedforward Network

Output: Cat/Dog

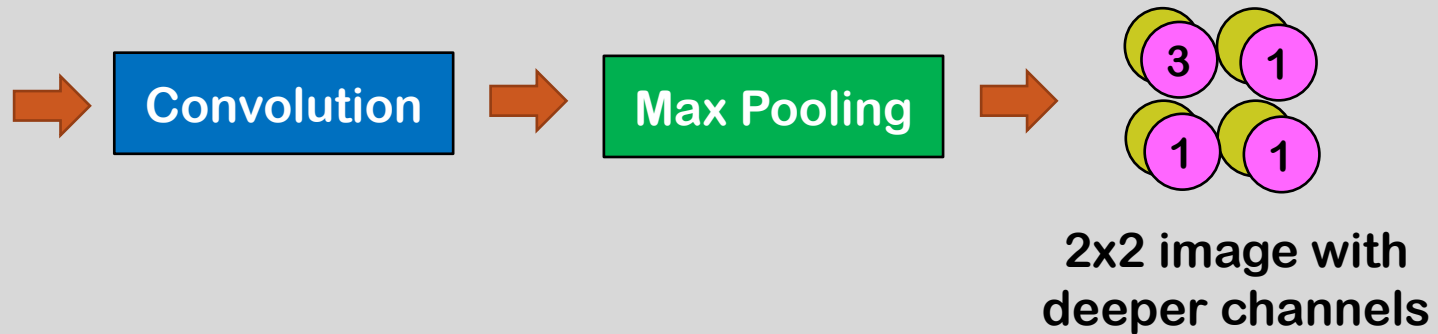
## Max Pooling

# CNN – Max Pooling

- Every time when going through the Convolution plus Max Pooling, the dimension of the image is reduced and multiple feature maps are generated.

0	1	0	0	1	0
0	1	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
0	1	0	0	1	0
1	0	0	0	0	1

6x6 image



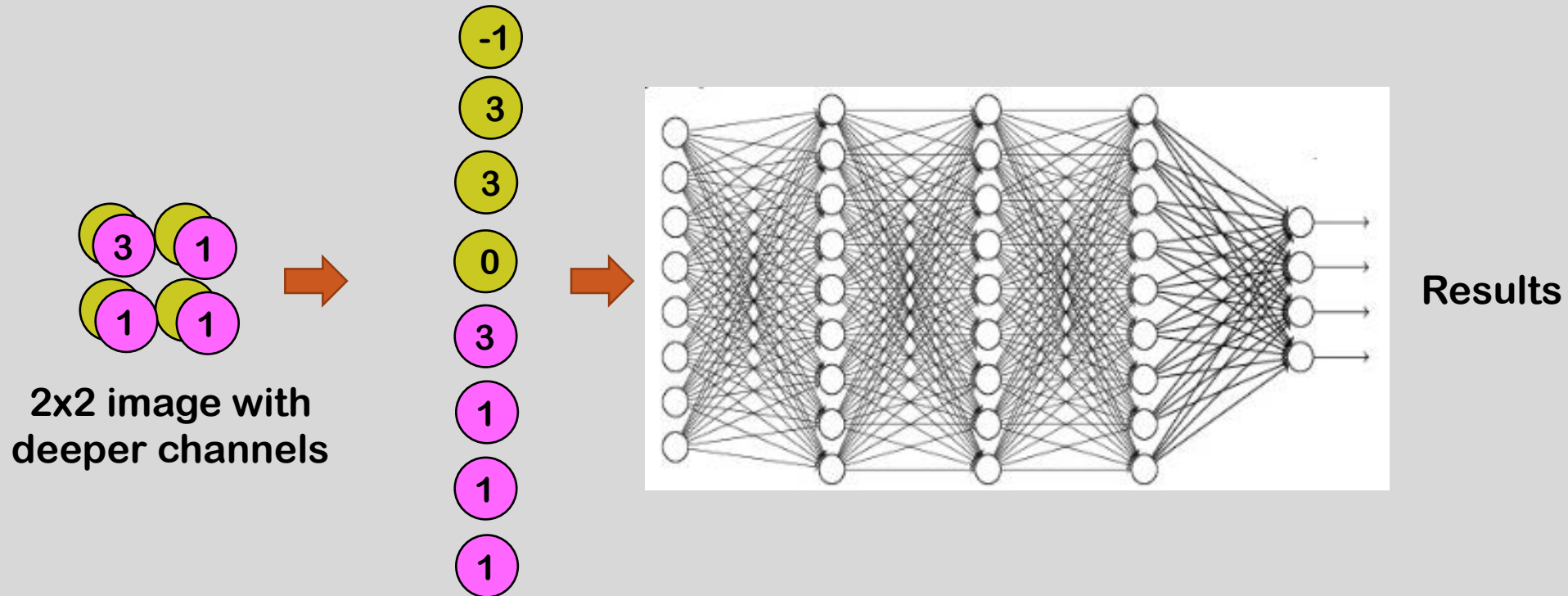
- The feature maps is considered as composed of a voxel (3-D cubes).



## Flatten

# CNN – Flatten and Fully Connected Network

- After straightening the feature map into a 1-D vector, it is pushed into the fully connected network for output.



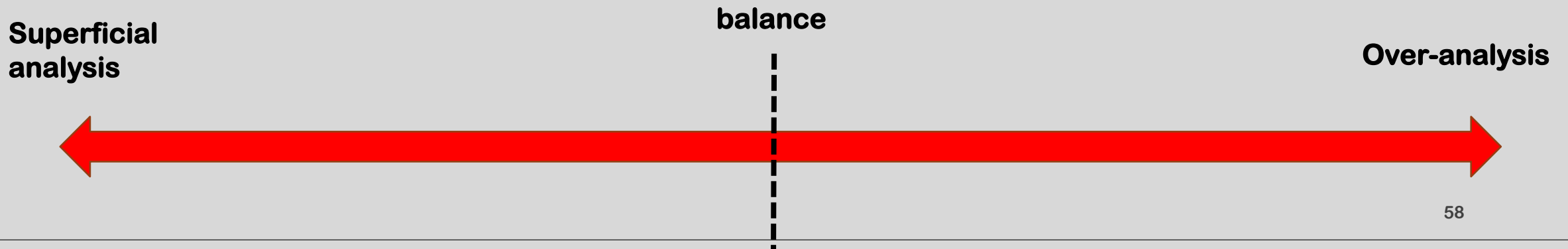


# COMMUNICATION RESULT

# Success or failure criteria

The success or failure criteria are determined by whether the data and the chosen analytics models are able to accept or reject the initial hypotheses formulated in Phase 1.

Rejecting a hypothesis does **not** always equate a failure. Instead, a failure usually refers to the inability to strike the balance between two possible analytics extremes.





# Data visualisation basics

It is important to know when to use a **particular type of chart or graph** to express a given kind of data. The objective is to find the best chart for expressing the data **clearly** so the visual does not impede the message, but supports the audience in taking away the intended message.

Data to be visualised	Type of chart
Components (parts of whole)	Pie chart
Item	Bar chart
Time series	Line chart
Frequency	Line chart of histogram
Correlation	Scatterplot, side-by-side bar charts

# Data visualisation best practices

**Be aware of “Data-Ink Ratio”.**

**Data-Ink** refers to the actual portion of a graphic that portrays the data, while **non-Data Ink** refers to labels, edges, colours, and other decoration.

Hence, **Data-Ink Ratio** could be thought of as:

$$\frac{\text{Data - ink}}{\text{Total ink used to print the graphic}}$$

Also equals to:  
1.0 – proportion of a graphic that can be erased without loss of data-information

**The greater the ratio, the more data rich it is and the fewer distractions it has.** According to Edward Tufte who pioneered the concept, the goal of data visualisation is to design display with the highest possible data-ink ratio (that is, as close to the total of 1.0), without eliminating something that is necessary for effective communication.



YOUR UNIT. YOUR SAY.