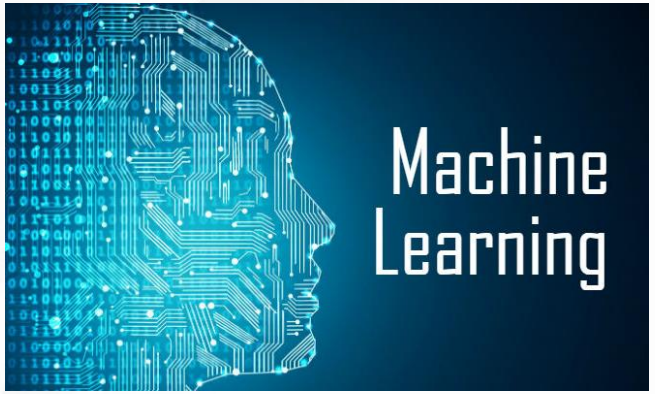
Data Science

Zakka Ugih Rizqi, S.T., M.T., M.B.A.

Researcher at Center for IoT Innovation

Ph.D. Student in Taiwan Tech



HELLO! 你好!

Zakka Ugih Rizqi

ABOUT ME

- S.T. : Industrial Engineering - UII 2020
- M.T. : Industrial Engineering - UII 2021 (Fast Track Program)
- M.B.A. : Industrial Management NTUST 2021 (Dual Degree Program)
- Ph.D. : Industrial Management – NTUST (On-Going)
- *Research Area : Data Science, Optimization, Simulation, Industry 4.0, Supply Chain Management*
- ORCID ID : https://orcid.org/0000-0003-2986-9503
- Hobbies : Music, Gym, Editing, & Writing
- Training Experience: Data Science by KOMINFO, Python by KOMINFO, Digital Entrepreneurship by Google, Artificial Intelligence by Huawei

You can find me on:

ugihzakka@gmail.com
@zakkaugih
Zakka Ugih Rizqi
Zakka Ugih Rizqi
Personal Website : https://sites.google.com/view/zakkaugihrizqi

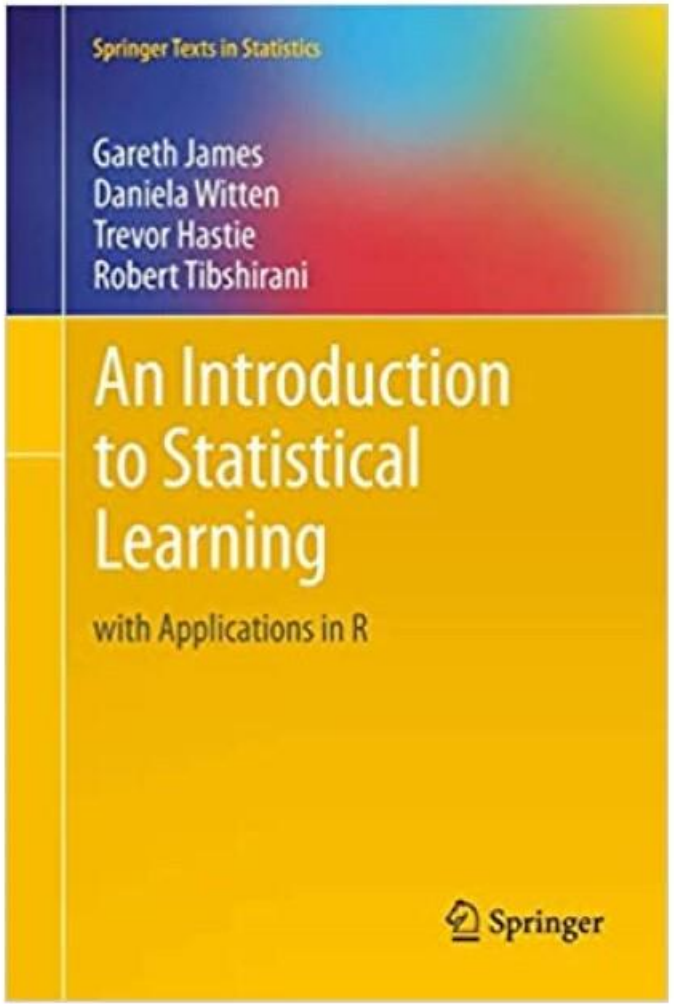
ACHIEVEMENT

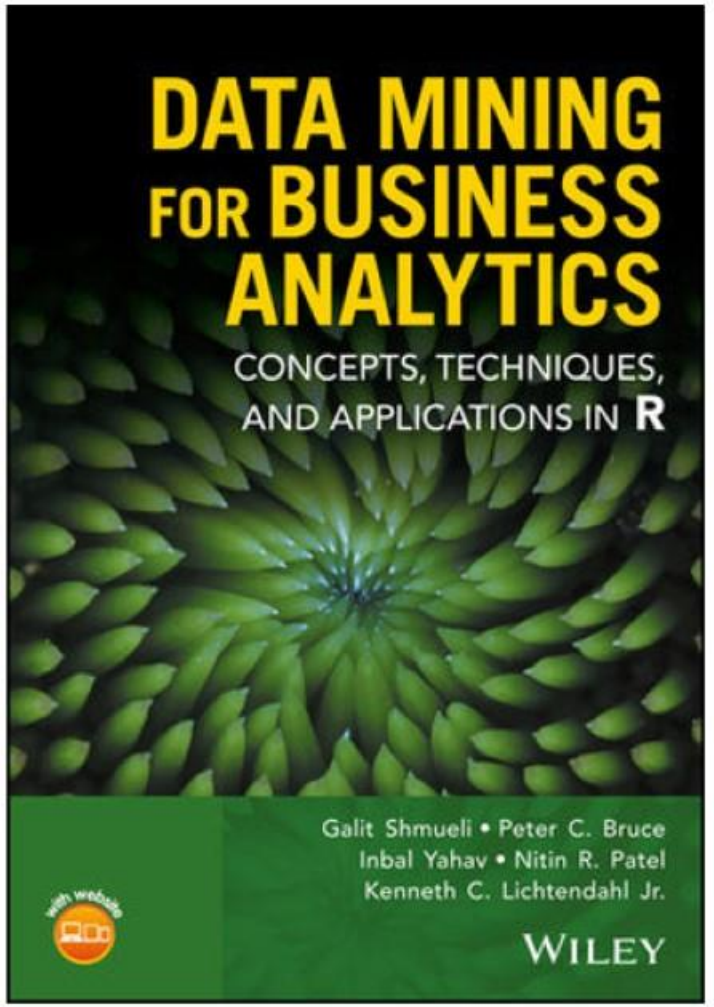
- 25++ International & National Scientific Publications
- Ph.D. & Master Scholarship Awardee in NTUST (Taiwan Tech)
- The Best Graduate in Industrial Engineering Department, UII 2020
- 3 Times Best Paper Awardee in International Conference 2019 & 2020
- 2 Times Student Creativity Program (PKM) Grant Awardee by Ministry of Education and Culture of Indonesia, 2019 & 2020
- 2nd Winner of Simulation Competition Award. IEOM Society Awards, UAE, 2020
- 3rd Winner of Industrial Engineering Paper and Action (INPACT), USU, Indonesia, 2019
- 3rd Winner of Competition of Industrial Engineering (CONSTRAIN), UNHAS, Indonesia, 2019

WORK EXPERIENCE

- Researcher in Center for IoT Innovation, Taiwan Tech. 2022 - Now
- Internship at Corporate Planning Department, PT Pos Logistik Indonesia, 2020
- Internship at Refinery Planning & Optimization Department, PT Pertamina (Persero), 2019
- Research Assistant of Industrial Modeling & Simulation Laboratory (DELSIM), 2019-2020
- Research Assistant of Basic Physics Laboratory, Faculty of Industrial Technology, 2020
- Research Assistant of Design of Integrated Industrial System, Faculty of Industrial Technology, 2020

Textbook





Outline

Week 1	*Introduction to Data Science* - Terms, Scope, Big Data, Analytics Types, Skillsets, Data Types, Supervised VS Unsupervised Learning, CRISP-DM
Week 2	*Python for Data Science* - Introduction to Python, Data Science Libraries, Exploratory Data Analysis, Tutorial for Data Cleansing & Data Visualization
Week 3	*Machine Learning for Predictive Analytics* - Classification, Regression, Time-Series, Hyperparameter Tuning
Week 4	*Machine Learning for Descriptive Analytics* - Clustering, Association Rule Mining, Moving from Machine Learning to Deep Learning
Final Project Presentation	Will be discussed later!

Part 1
Introduction to Data Science

What is Data Science?

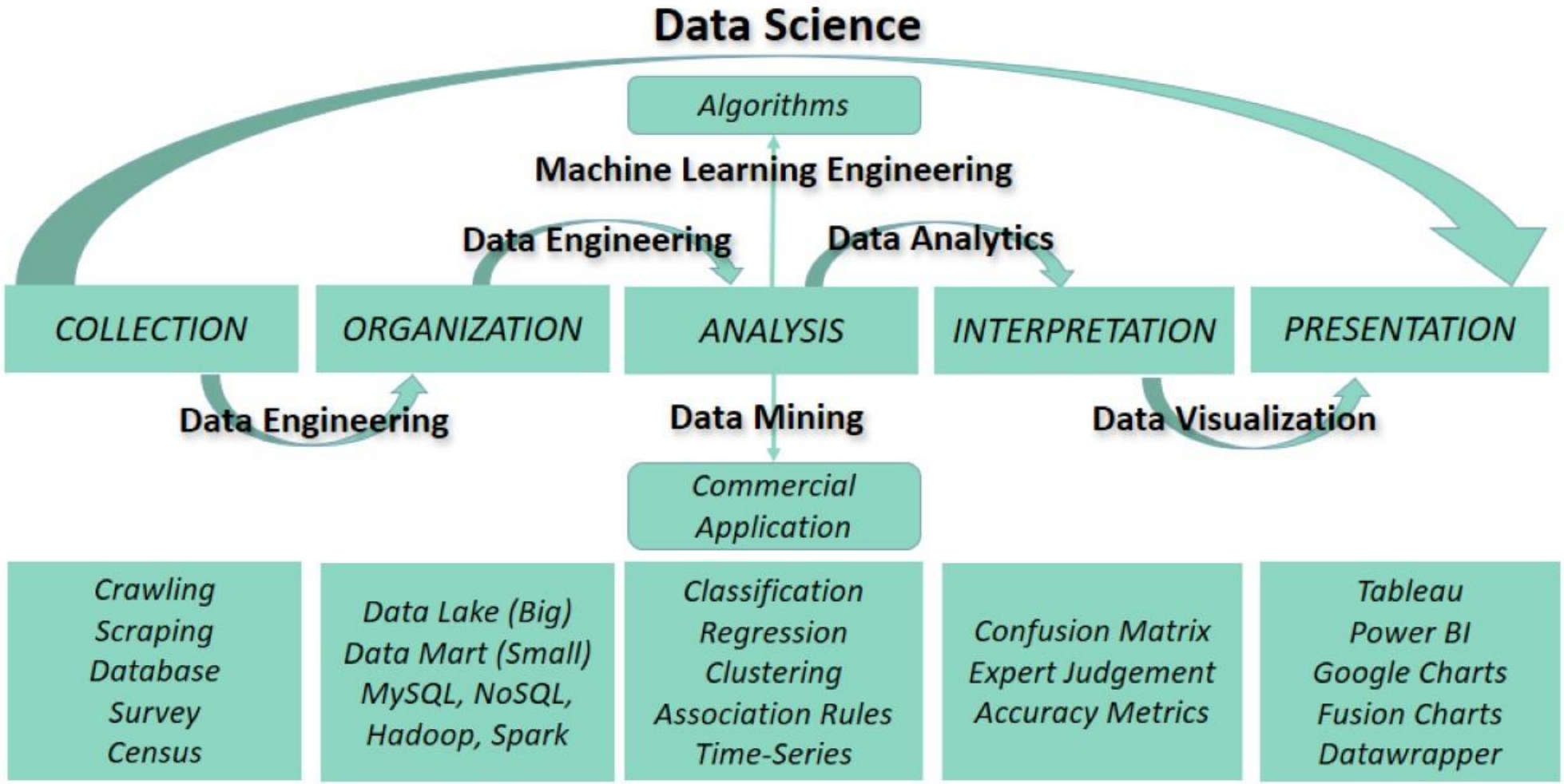
- **Data** is collection of facts, it can be number, text, etc. In singular, it is called datum
- **Science** is systematic approach that organizes specific knowledge
- In simple term, **Data Science** is a systematic approach to acquire insight based on data and use it to make powerful decisions
- Wu (1997) called statistics as data science, and statisticians as data scientists. Even tough some experts do not agree
- **Statistics** emphasizes on **inferencing sample** (explaining average effect through hypothesis) **from population,** Data Science emphasizes on **discovering pattern** and **producing the accurate ML model**

- How about the other terms? Some similar terms:

Data Mining, Data Analytics, Data Engineering, Machine Learning, etc.

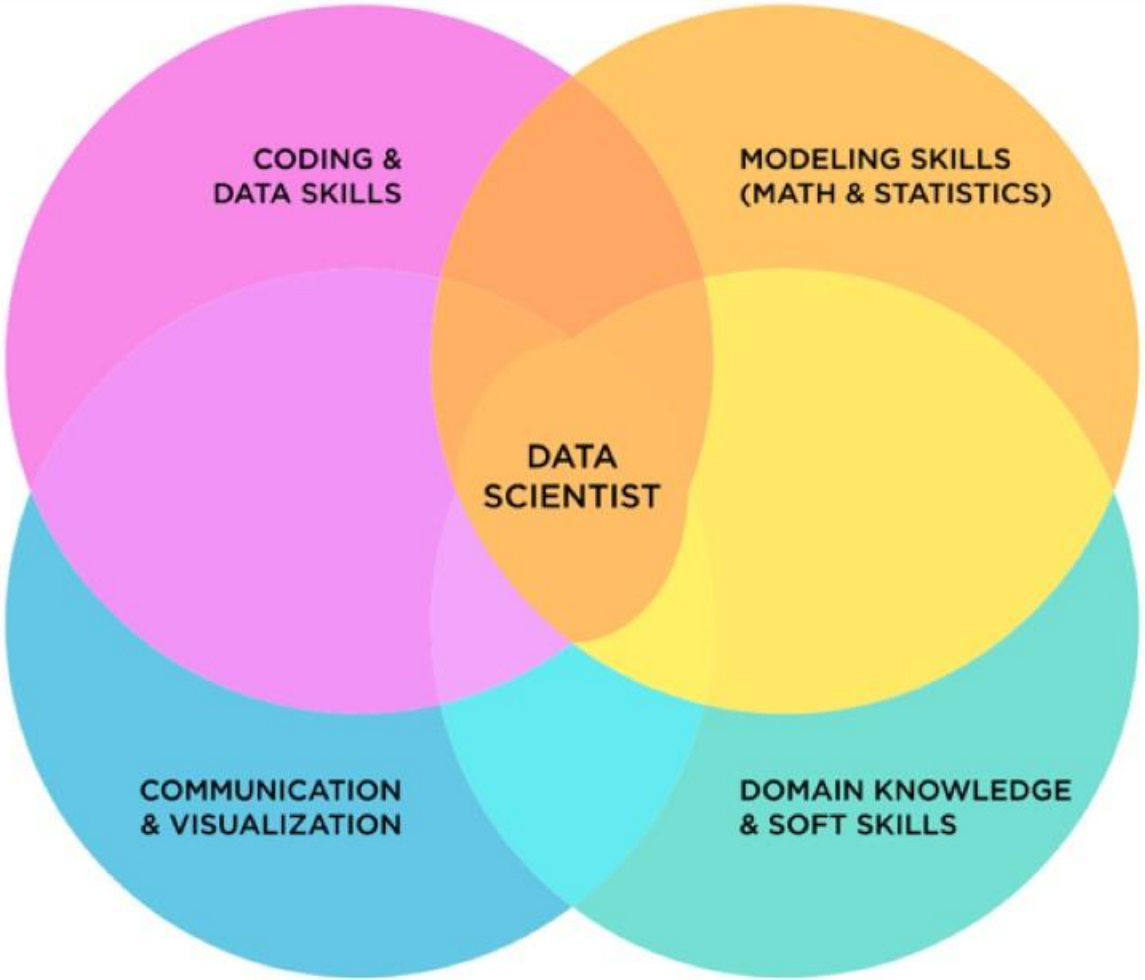
Wu, J. (1997). *Statistics = Data Science?*. Inaugural Lecture in University of Michigan

Data Science Vs Others



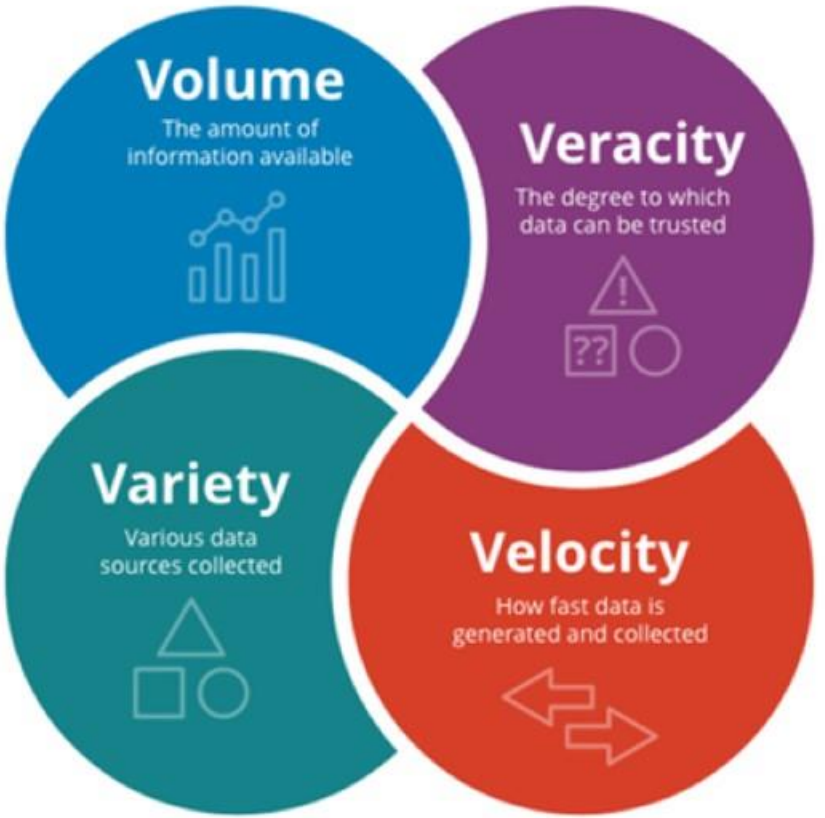
- This framework is **not absolute**, sometimes it is overlapping
- The best way is by following the **convention** where you are in

Data Science Skills



Why is Data Science Needed?

- We live in **Big Data era** → Big Data Analytics
- Characteristics of Big Data → **"4V's"**
- In 2006, Clive Humby shouted "Data is the new oil."
- How many actually the number of big data? Not certain answer.
- Some experts said > Ms. excel (1,048,576x16,384), others said if the data has > 1 gigabyte
- Where do the data come from?
  - Internal company data (excel, internal databases, cloud)
  - Web API's or Scraping (twitter, youtube)
  - Web page downloading or crawling (web page)
  - Public dataset (financial statements, tax records, job desc)
  - Open dataset (Kaggle, UCI, UNICEF, WHO, World Bank)
  - Private collection data (survey, interview)



Why is Data Science Needed?

SPOTLIGHT ON BIG DATA

Spotlight

ARTWORK Tamar Cohen, *Andrew J Buboltz*
2011, silk screen on a page from a high school yearbook, 8.5" x 12"

Data Scientist:

The Sexiest Job of the 21st Century

Meet the people who can coax treasure out of messy, unstructured data.

by Thomas H. Davenport and D.J. Patil

70 Harvard Business Review **October 2012**

- Read More: https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

How Dataset Looks Like?

- When data are in table format, it is called **Dataset**

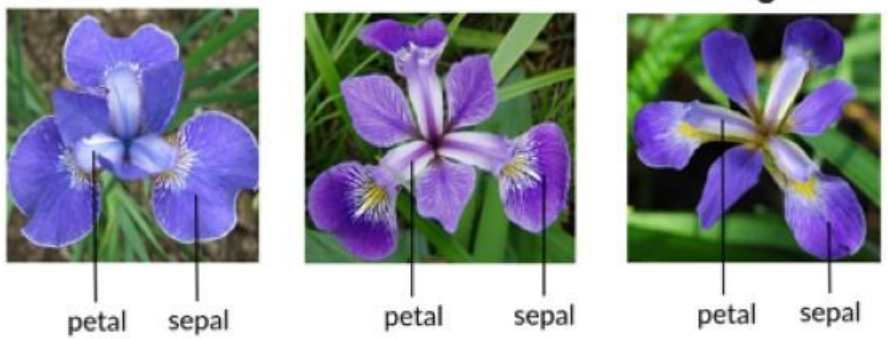
Attribute/**Feature**/Dimension/ Predictor/Explanatory/Independent Variables

Class/Label/**Target**/Response /Outcome/Dependent Variable

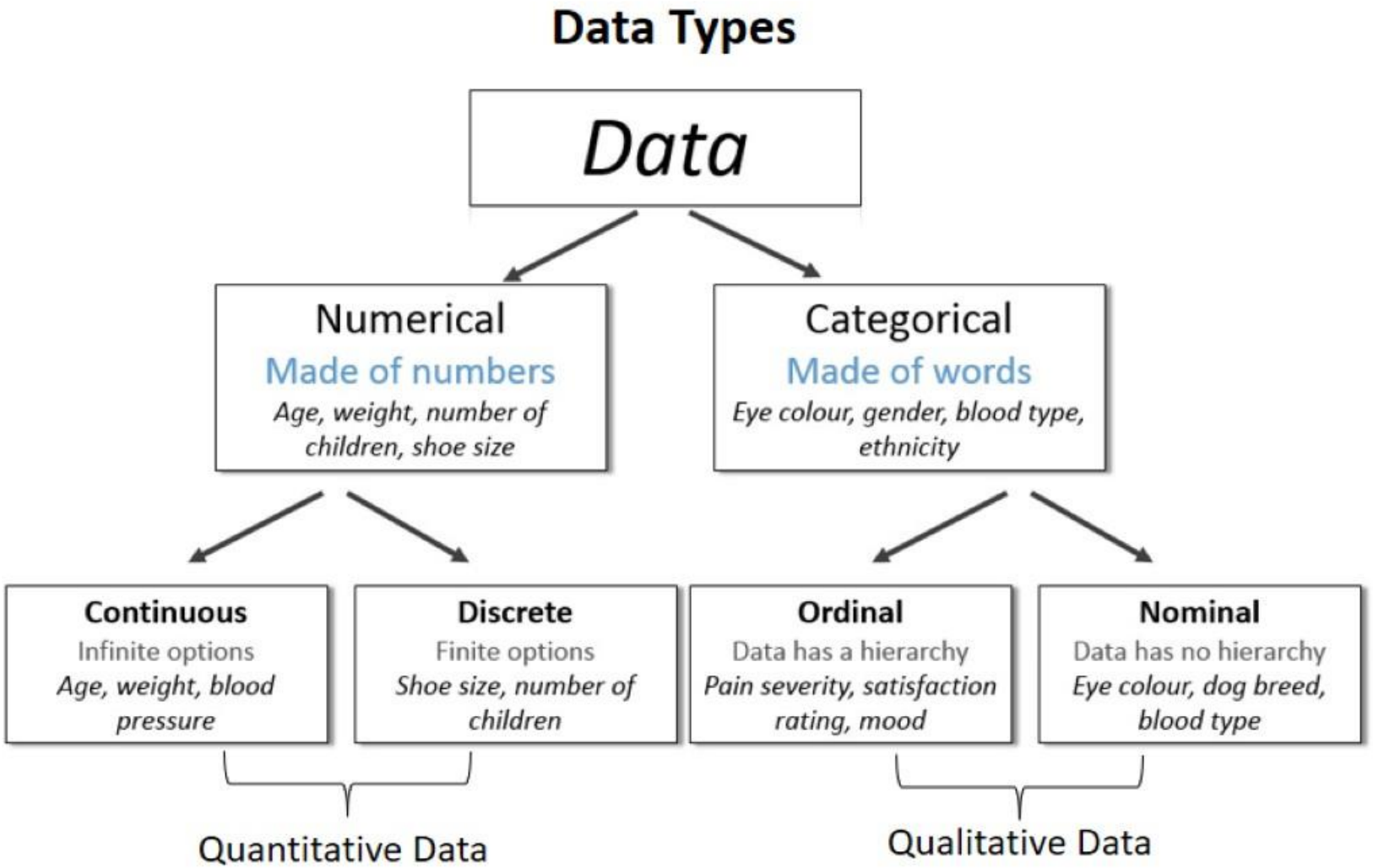
	Sepal Length (cm)	Sepal Width (cm)	Petal Length (cm)	Petal Width (cm)	Type
1	5.1	3.5	1.4	0.2	*Iris setosa*
2	4.9	3.0	1.4	0.2	*Iris setosa*
3	4.7	3.2	1.3	0.2	*Iris setosa*
4	4.6	3.1	1.5	0.2	*Iris setosa*
5	5.0	3.6	1.4	0.2	*Iris setosa*
...					
51	7.0	3.2	4.7	1.4	*Iris versicolor*
52	6.4	3.2	4.5	1.5	*Iris versicolor*
53	6.9	3.1	4.9	1.5	*Iris versicolor*
54	5.5	2.3	4.0	1.3	*Iris versicolor*
55	6.5	2.8	4.6	1.5	*Iris versicolor*
...					
101	6.3	3.3	6.0	2.5	*Iris virginica*
102	5.8	2.7	5.1	1.9	*Iris virginica*
103	7.1	3.0	5.9	2.1	*Iris virginica*
104	6.3	2.9	5.6	1.8	*Iris virginica*
105	6.5	3.0	5.8	2.2	*Iris virginica*
150					

Sample (n) /Row

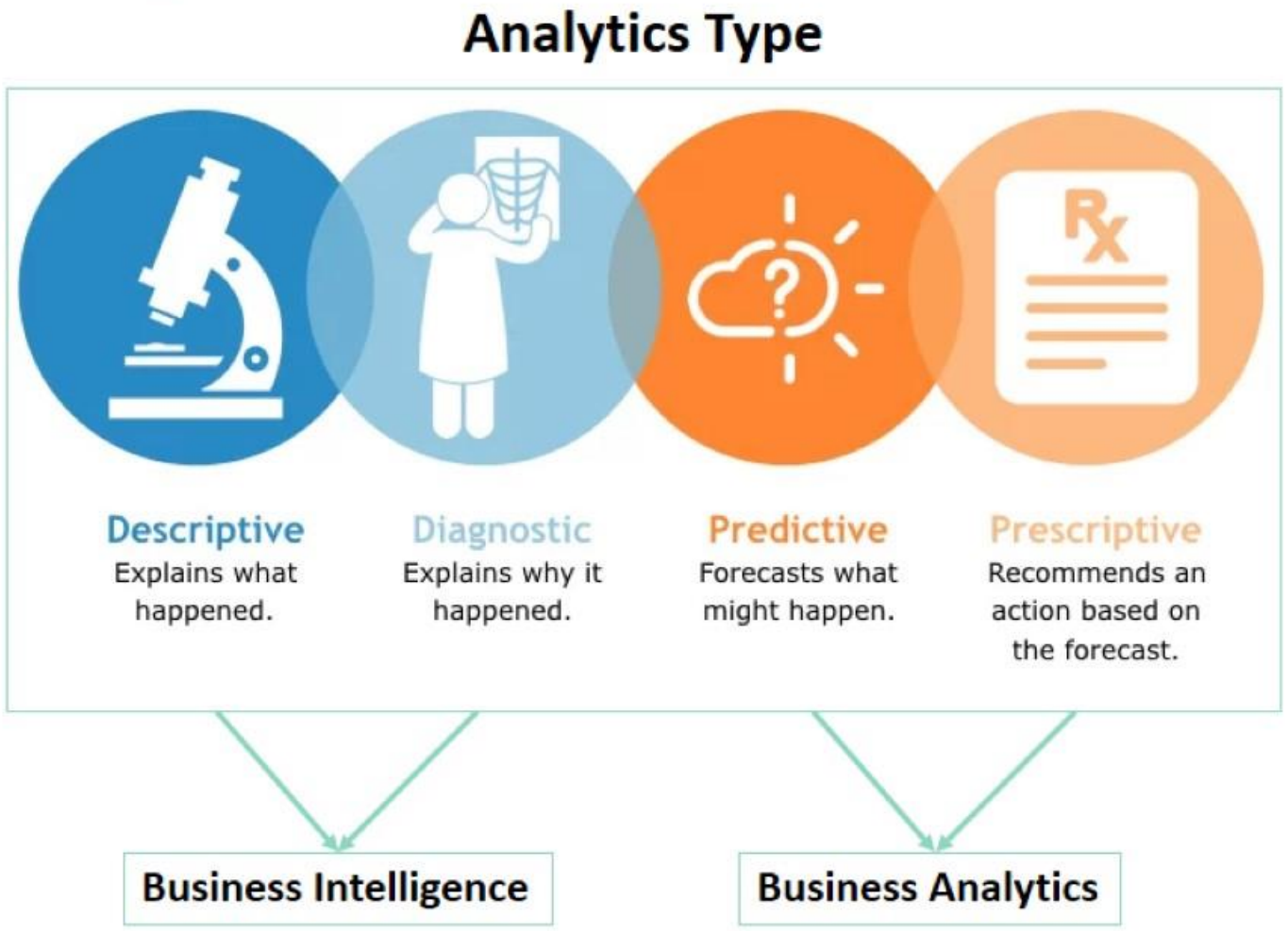
A well-known dataset – "Iris"



How Dataset Looks Like?



How Data Science Works



- Again, sometimes it is **overlapping** and **interchangeable**
- Usually, Data Science emphasizes on **Descriptive (Unsupervised)** and **Predictive (Supervised)** Analytics
- **Diagnostics** Analytics emphasizes on Statistical Tools (SPC, 7 Tools) and sometimes it is included in Descriptive
- **Prescriptive** Analytics emphasizes on Optimization, Simulation, Stochastic Process, MCDM

How Data Science Works

Supervised VS Unsupervised Learning

Supervised Learning

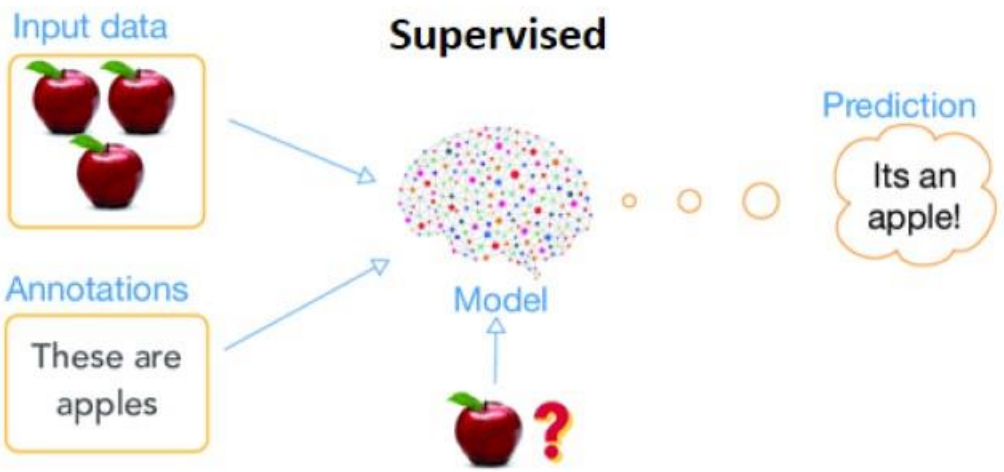
X_1	X_2	X_3	X_p	Y

Target

Un-Supervised Learning

X_1	X_2	X_3	X_p	Y

No Target



Data Science Process

CRISP-DM

- **What is that?**

- It stands for *CRoss Industry Standard Process for Data Mining*
- It was first used by 5 industries in conducting European Union project 1996
- Open standard process model that describes common approaches used by data mining expert.
- It has 6 main stages, started from Business Understanding and finished in Deployment

CRISP-DM - Detail

1. **Define the problem clearly** as well as the purpose of the data science project
2. **Obtain the dataset** related to the problem faced based on the right requirements and right sources
3. **Determine** analytics **approach**, data mining task, and algorithms
4. **Explore data** (EDA), feature analysis, and pre-process the data
5. **Determine Testing system**, partition the data
6. **Perform** ML **algorithm** & optimize it
7. **Evaluate** the **performance**, Interpret the results, and visualize it, if necessary
8. **Adjust the result** based on the business problem, do a loop, if necessary
9. **Deploy** where algorithm wants to be embedded , if necessary

- Stage 1 (BU): 1, 2
- Stage 2 (DU): 3
- Stage 3 (DP): 4,5
- Stage 4 (Mo): 6
- Stage 5 (Ev): 7, 8
- Stage 6 (Dp): 9

CRISM-DM (Example of Zillow)

1. **Define the problem & purpose** → We want to know the house price in US without observing & asking it directly to the owner. Then just build the predictive model that can predict house price based on the historical data.
2. **Define the approach** → Since we want to predict, choose predictive analytics. Since house price is numerical data type, choose regression. We choose some algorithms: Multiple Linear Regression, SVR, DCR, & Neural Network.
3. **Obtain dataset** → Use the 2014 West Roxbury Housing data from database. It contains 13 features and 5000 samples.
4. **Pre-process the data** → Some data are missing, then we need to clean it (e.g. delete it).
5. **Partition the data** → We divide the data directly. For training 80%, for testing 20%.
6. **Perform Algorithm** → All algorithms are performed using Python. Each algorithm has also been optimized.
7. **Evaluate the performance** → It shows that Neural Network is the best algorithm.
8. **Adjustment** → Is the model fulfilled the objective? Yes.
9. **Deployment** → We want the model to help not only us, but also people who need this information. Then just embed it in the website!

TABLE 2.1 DESCRIPTION OF VARIABLES IN WEST ROXBURY (BOSTON) HOME VALUE DATASET

TOTAL VALUE	Total assessed value for property, in thousands of USD
TAX	Tax bill amount based on total assessed value multiplied by the tax rate, in USD
LOT SQ FT	Total lot size of parcel in square feet
YR BUILT	Year the property was built
GROSS AREA	Gross floor area
LIVING AREA	Total living area for residential properties (ft^2)
FLOORS	Number of floors
ROOMS	Total number of rooms
BEDROOMS	Total number of bedrooms
FULL BATH	Total number of full baths
HALF BATH	Total number of half baths
KITCHEN	Total number of kitchens
FIREPLACE	Total number of fireplaces
REMODEL	When the house was remodeled (Recent/Old/None)

TABLE 2.2 FIRST 10 RECORDS IN THE WEST ROXBURY HOME VALUES DATASET

TOTAL VALUE	TAX	LOT SQ FT	YR BUILT	GROSS AREA	LIVING AREA	FLOORS	ROOMS	BED ROOMS	FULL BATH	HALF BATH	KIT CHEN	FIRE PLACE	REMODEL
344.2	4330	9965	1880	2436	1352	2	6	3	1	1	1	0	None
412.6	5190	6590	1945	3108	1976	2	10	4	2	1	1	0	Recent
330.1	4152	7500	1890	2294	1371	2	8	4	1	1	1	0	None
498.6	6272	13,773	1957	5032	2608	1	9	5	1	1	1	1	None
331.5	4170	5000	1910	2370	1438	2	7	3	2	0	1	0	None
337.4	4244	5142	1950	2124	1060	1	6	3	1	0	1	1	Old
359.4	4521	5000	1954	3220	1916	2	7	3	1	1	1	0	None
320.4	4030	10,000	1950	2208	1200	1	6	3	1	0	1	0	None
333.5	4195	6835	1958	2582	1092	1	5	3	1	0	1	1	Recent
409.4	5150	5093	1900	4818	2992	2	8	4	2	0	1	0	None

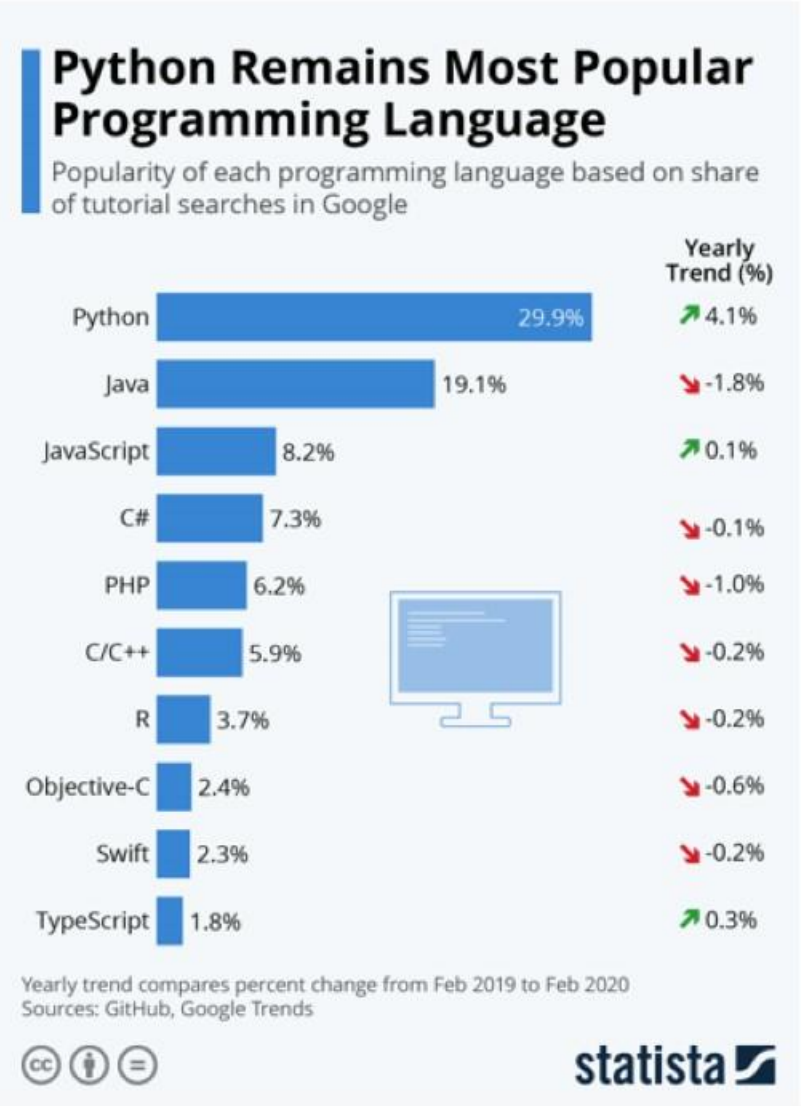
Part 2
Python & Exploratory Data Analysis (EDA)
Let's Encode!

Python History

Founder	Guido van Rossum
Degree	Master of Mathematics、 Master of Computer Science
When and Where	Created in Amsterdam during Christmas in 1989.
Meaning of Name	A big fan of Monty Python's Flying Circus

Why Python & Why Programming?

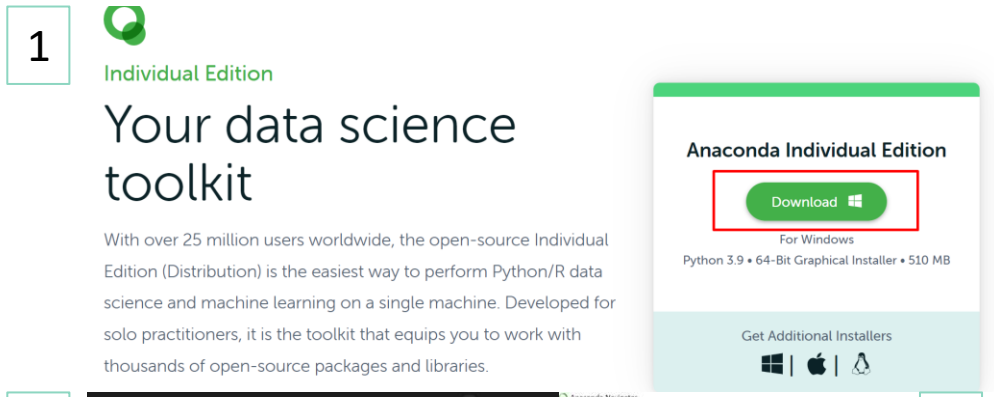
- Compared with other languages
  - Interpreted Language, not Compiled Language
  - General purpose
  - Open Source, many libraries for Data Science
  - Relatively Easy to Use
  - Predicted to be no. 1 for Data Science purpose in the near future
- Compared with packaged software (RapidMiner, Ms. Excel, Orange, etc.)
  - Flexibility → Integration or Enhancement
  - Scalability → Able to handle Small/Big Data
  - Free → Cost Effective



Python Environment

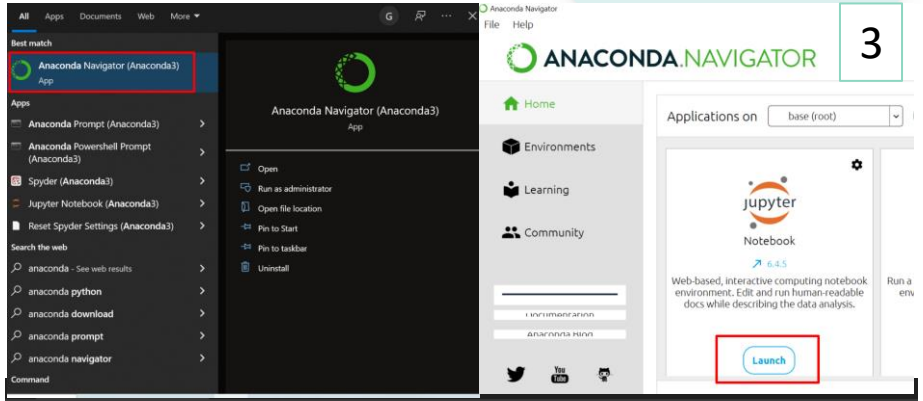
- In order to operate python easily, we need Integrated Development Environment (IDE)
- IDE used in this tutorial is Jupyter Notebook, others: PyCharm, VSCode, Google Collab
- Jupyter is interactive and enables us to run code in smaller chunks called “cells”
- Installation: 1. Click https://www.anaconda.com/products/individual, Download & Install it → 2. Open Anaconda → 3. Launch Jupyter Notebook → 4. Click New and Click Python 3
- For detail tutorial, follow this: https://docs.anaconda.com/anaconda/install/

1

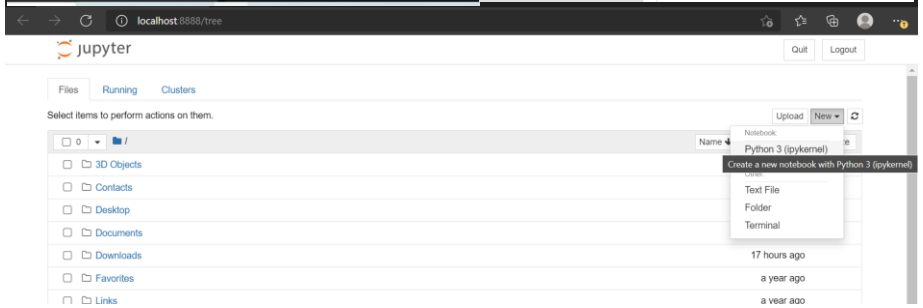


2

3

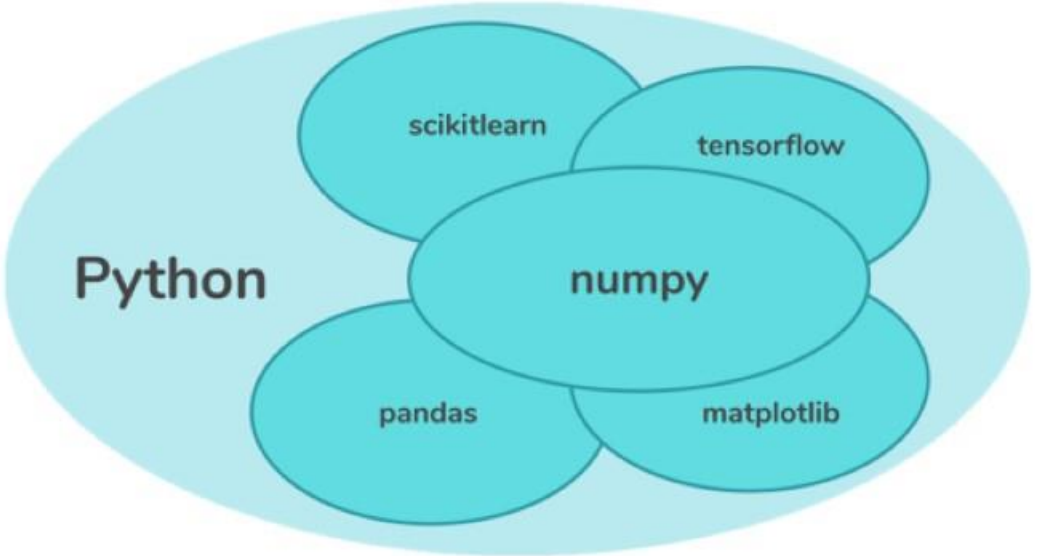


4



Important Libraries

- A Python library is a set of related modules, containing bundles of code that can be used repeatedly in different programs
- We need to install it, how?
- Open Anaconda Prompt → Type “pip install (library name)”



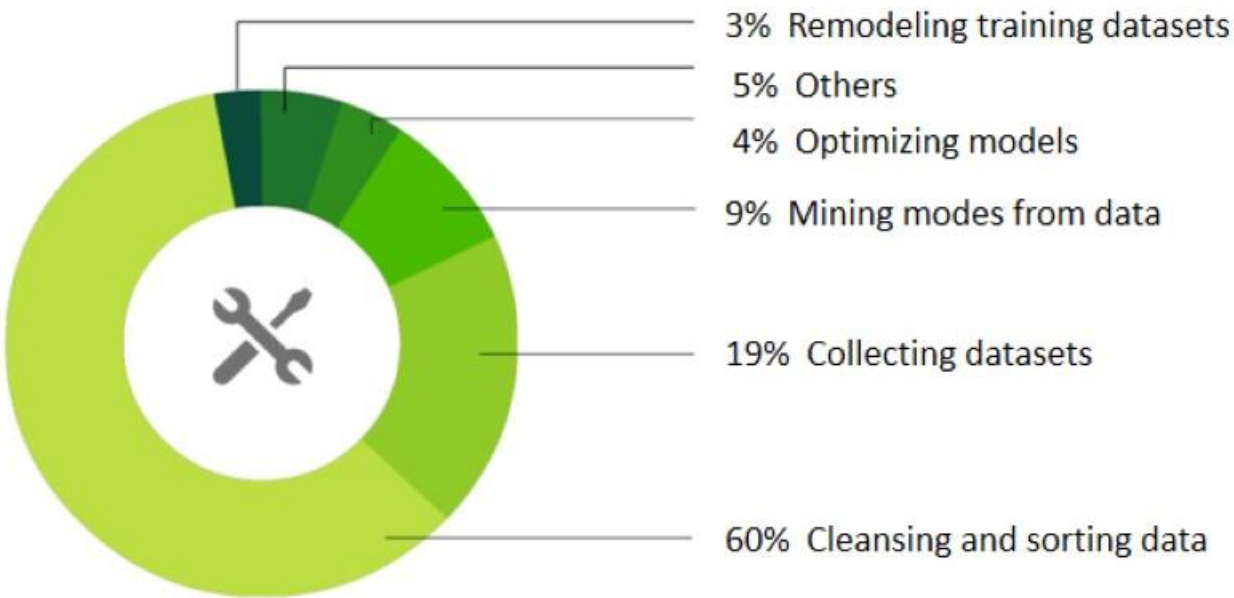
We will use 5 main libraries:

1. NumPy → Matrix for linear algebra
2. Pandas → Spreadsheet/Table for data processing
3. Scikitlearn → Machine Learning algorithms
4. Tensorflow → Deep Learning algorithms
5. Matplotlib & Seaborn → Visualization

Data Cleansing

Data Preprocessing

- Data is crucial to models. It is the ceiling of model capabilities. Without good data, there is no good model.
- This step of preprocessing the data is so-called Data Cleansing.
- Other terms are Data Cleaning, Data Scrubbing & Data Wrangling
- How hard it is?



CrowdFlower Data Science Report 2016

Data Preprocessing

- Generally, real data may have some quality problems
  - Incompleteness: contains missing values or the data that lacks attributes
  - Noise: contains incorrect records or exceptions.
  - Inconsistency: contains inconsistent records.
- Missing value & Duplication can be easily and automatically done
- Others, need attention!

Example:
International School Dataset

#	Id	Name	Birthday	Gender	IsTeacher	#Students	Country	City
1	111	John	31/12/1990	M	0	0	Ireland	Dublin
2	222	Mery	15/10/1978	F	1	15	Iceland	
3	333	Alice	19/04/2000	F	0	0	Spain	Madrid
4	444	Mark	01/11/1997	M	0	0	France	Paris
5	555	Alex	15/03/2000	A	-1	23	Germany	Berlin
6	555	Peter	1983-12-01	M	1	10	Italy	Rome
7	777	Calvin	05/05/1995	M	0	0	Italy	Italy
8	888	Roxane	03/08/1948	F	0	0	Portugal	Lisbon
9	999	Anne	05/09/1992	F	0	5	Switzerland	Geneva
10	101010	Paul	14/11/1992	M	1	26	Ytali	Rome

Missing value

Invalid value

Value that should be in another column

Misspelling

Invalid duplicate item

Incorrect format

Attribute dependency

Handling Messy Data

- **How to handle missing value?**

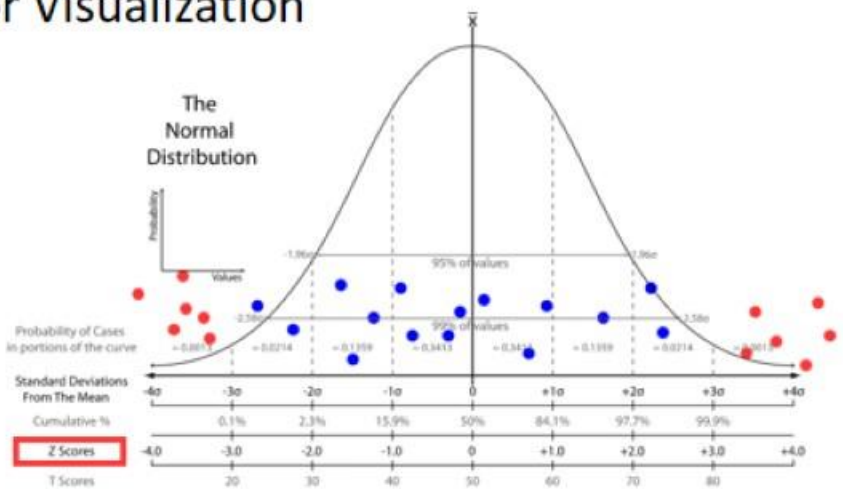
- If a row has missing value in one feature only → replace it with mean or modus in that feature (data imputation) → Advanced technique may use predictive analytics
- If a row has missing value in more than one feature → better to delete/drop it

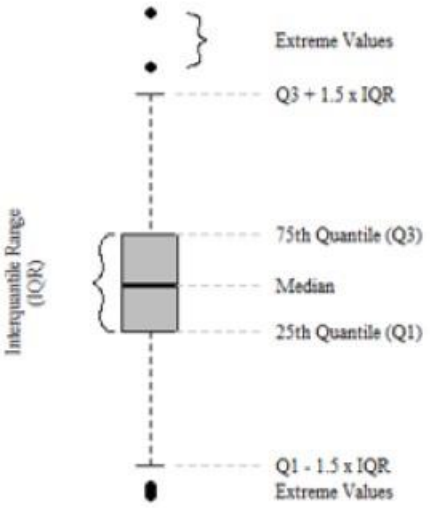
- **How to detect outlier?**

- Quartile → Data lies in the outside of lower bound = (Q1–1.5*IQR) & upper bound = (Q3+1.5*IQR) → Where IQR = Q3-Q1.
- Boxplot → Use Data Visualization
- Z-Score → Can be done by calculation or Visualization

- **How to handle outlier?**

- Trimming (Eliminating)
- Rescaling
- Doing nothing → Outlier is also data!





Handling Messy Data

- Categorical Variable needs special attention → Why? Because ML Algorithms follow math process and string cannot be calculated through math
- Therefore, we need to convert it first into numerical data by using "***Encoding***"
- **Categorical Data Handling Approach:**

1. Convert *categorical variable* to *numerical variable* with **Label Encoding** → Can we use it directly? NO → Is chicken value twice bigger than apple?
2. Convert *numerical variable* with **One-Hot Encoding** and **Dummy Variable**

Special issue in Dummy Variable

There is ***dummy variable trap*** that makes calculation into *redundancy* and *multicollinearity*. Therefore, follow this rule: **number of dummy variable columns = n-1**, where n is initial number of dummy variable columns.

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Feature Scaling

- In many cases, dataset has some features with different scale
- It will make ML algorithms hard to learn because ML algorithms basically calculate the distances between data
- The higher feature may contribute/dominate more in affecting the result → It is not fair since the lower feature seems not significantly affecting the result.
- Therefore, feature scaling is usually executed in data preprocessing stage. This approach also can be said as feature engineering.
- Two approaches of feature scaling are:
  - Standardization → new data ranges [-∞, ∞] with Mean = 0 & SD = 1 → Based on Z standard principle
  - Normalization → new data ranges [0, 1] → Based on Min Max value

Before Feature Scaling

	Country	Age	Salary	Purchased
0	France	44.0	72000.0	No
1	Spain	27.0	48000.0	Yes
2	Germany	30.0	54000.0	No
3	Spain	38.0	61000.0	No

Standardization

$$x_{stand} = \frac{x - \text{mean}(x)}{\text{standard deviation } (x)}$$

Normalization

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

After Feature Scaling

```
-0.19159184384578545 -1.0781259408412425]
-0.014117293757057777 -0.07013167641635372]
0.566708506533324 0.633562432710455]
-0.30453019390224867 -0.30786617274297867]
-1.9018011447007988 -1.420463615551582]
1.1475343068237058 1.232653363453549]
1.4379472069688968 1.5749910381638885]
-0.7401495441200351 -0.5646194287757332]]
```

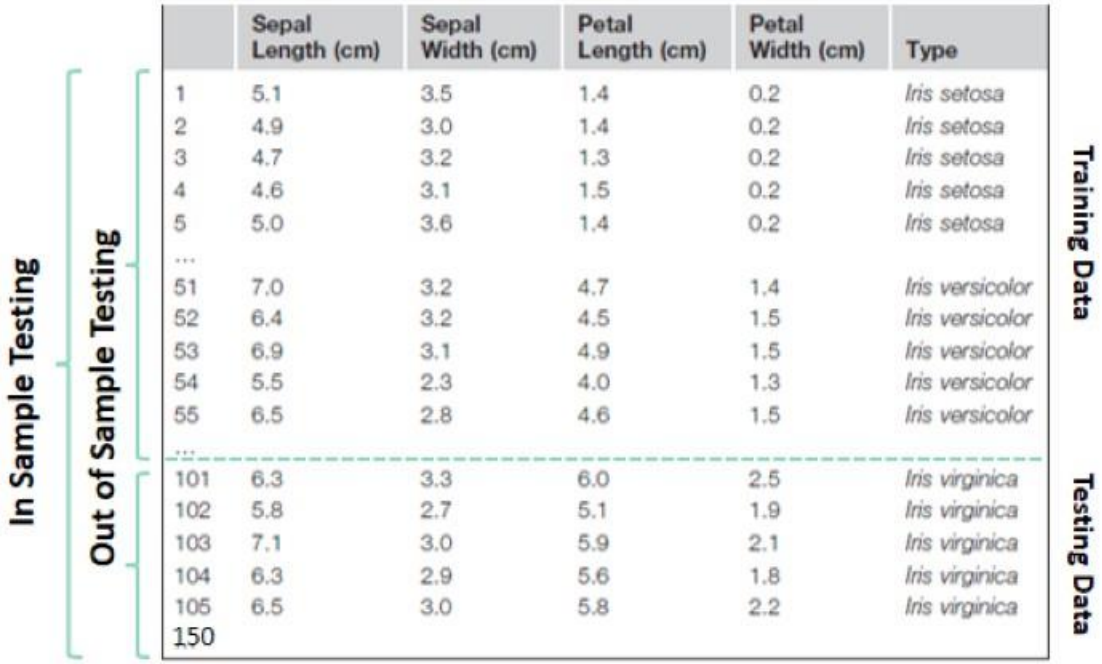
Testing System

1. In Sample Testing

- All data are used as training data
- Good choice if data are in small amount
- Training error is used as the main criteria of performance
- Not reliable enough; not fair because model has ever seen the data; can be trapped in overfitting

2. Out of Sample Testing

- Data are divided into training and testing data, even validation data (see the previous page)
- Good choice if data are in medium to big amount
- Testing/Validation error is used as the main criteria of performance
- Reliable approach; fair



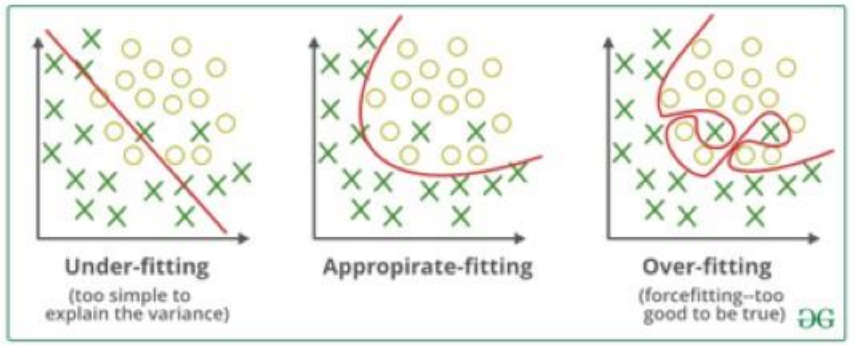
	Sepal Length (cm)	Sepal Width (cm)	Petal Length (cm)	Petal Width (cm)	Type
1	5.1	3.5	1.4	0.2	*Iris setosa*
2	4.9	3.0	1.4	0.2	*Iris setosa*
3	4.7	3.2	1.3	0.2	*Iris setosa*
4	4.6	3.1	1.5	0.2	*Iris setosa*
5	5.0	3.6	1.4	0.2	*Iris setosa*
...					
51	7.0	3.2	4.7	1.4	*Iris versicolor*
52	6.4	3.2	4.5	1.5	*Iris versicolor*
53	6.9	3.1	4.9	1.5	*Iris versicolor*
54	5.5	2.3	4.0	1.3	*Iris versicolor*
55	6.5	2.8	4.6	1.5	*Iris versicolor*
...					
101	6.3	3.3	6.0	2.5	*Iris virginica*
102	5.8	2.7	5.1	1.9	*Iris virginica*
103	7.1	3.0	5.9	2.1	*Iris virginica*
104	6.3	2.9	5.6	1.8	*Iris virginica*
105	6.5	3.0	5.8	2.2	*Iris virginica*
150					

Overfitting & Underfitting

- **Overfitting** → Model has good accuracy in training data but not in testing or validation data → Mistreated the noise in the data as a signal
- **Underfitting** → Model has bad accuracy in all data → Not all features are learned well
- In training data, data have been seen & learned by model. Others are not.

What is the worst effect?

- Overfitting may result in **selecting wrong model**, from which the best performing model is selected
- Underfitting may not only give inaccurate model, but also **useless model**
- A lot of new data scientists use the same dataset to develop the model and assess the performance → Here, we do not assess the **predictive power** but we assess the **memory power** of algorithm. It is called "**optimism bias**"



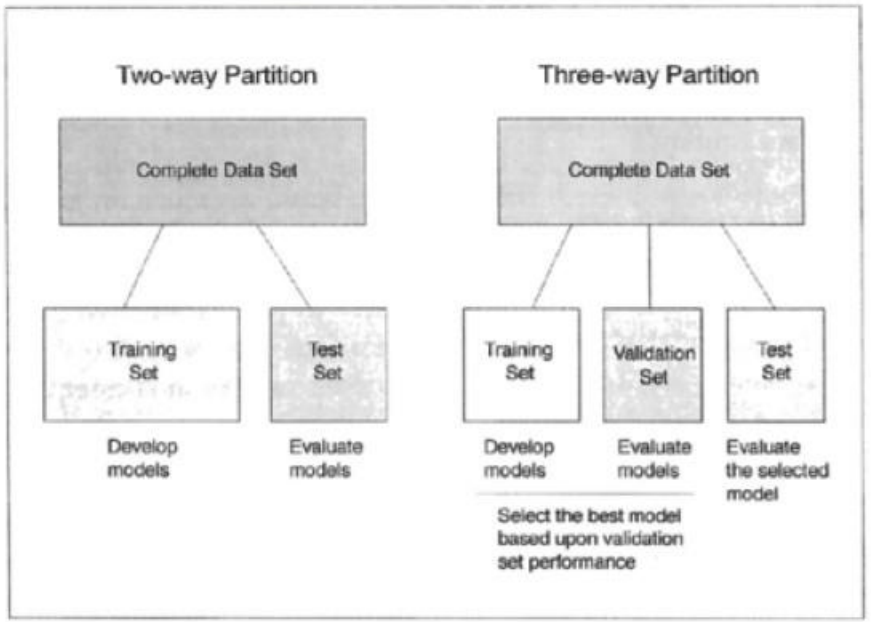
"All models are wrong, but some are useful"
George Box (1976)

George E. P. B. (1976), Science and statistics. *Journal of the American Statistical Association*, 71 (356): 791–799,

Overfitting & Underfitting

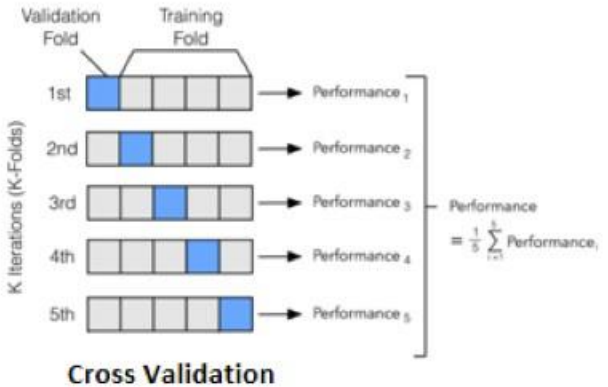
How to handle them?

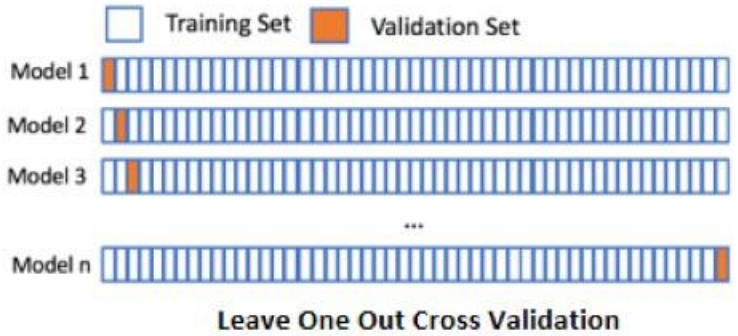
- **Overfitting** → 1. Data Partition 2. Regularization 3. Use more data & simpler algorithm



Data Partition Approaches:

1. Split directly → Good for big data. RoT is 80%/70% training & 20%/30% testing
2. Cross Validation (K-Fold) → Good for medium data and more reliable
3. Leave one out CV (LOOCV) → Good for small data and the most reliable



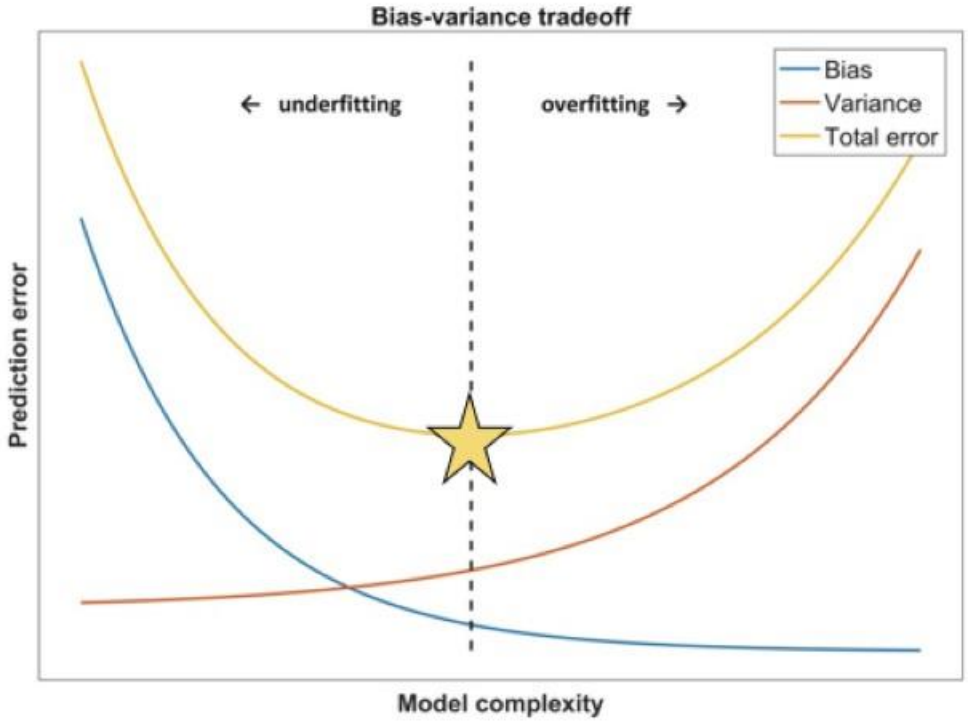


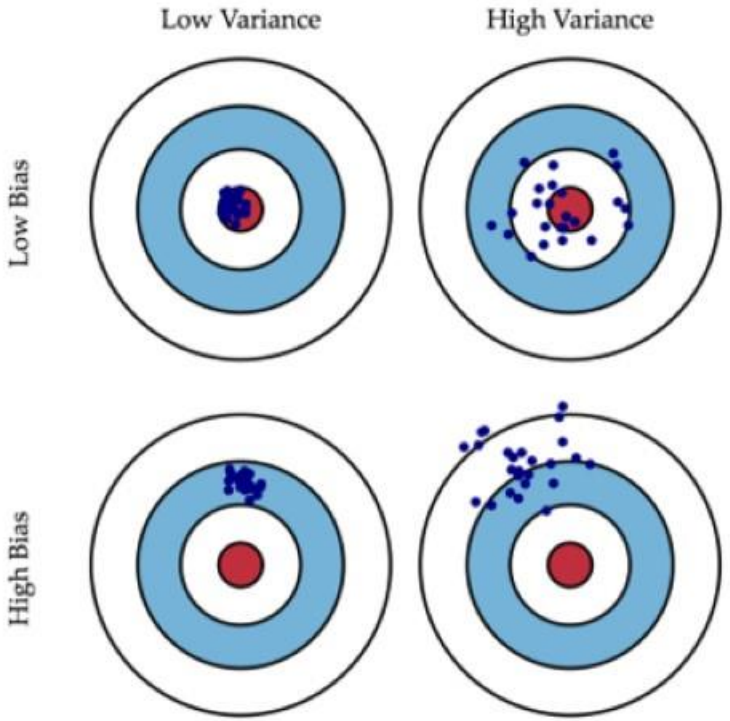
Regularization → minimizing Generalization Error = in sample error – out of sample error. Some approaches:

1. Early Stoppage
2. L1 – Lasso Regression
3. L2 – Ridge Regression
4. Dropout Regularization

- **Underfitting** → 1. Add related features 2. Add more data 3. Use more advanced algorithm

Variance VS Bias



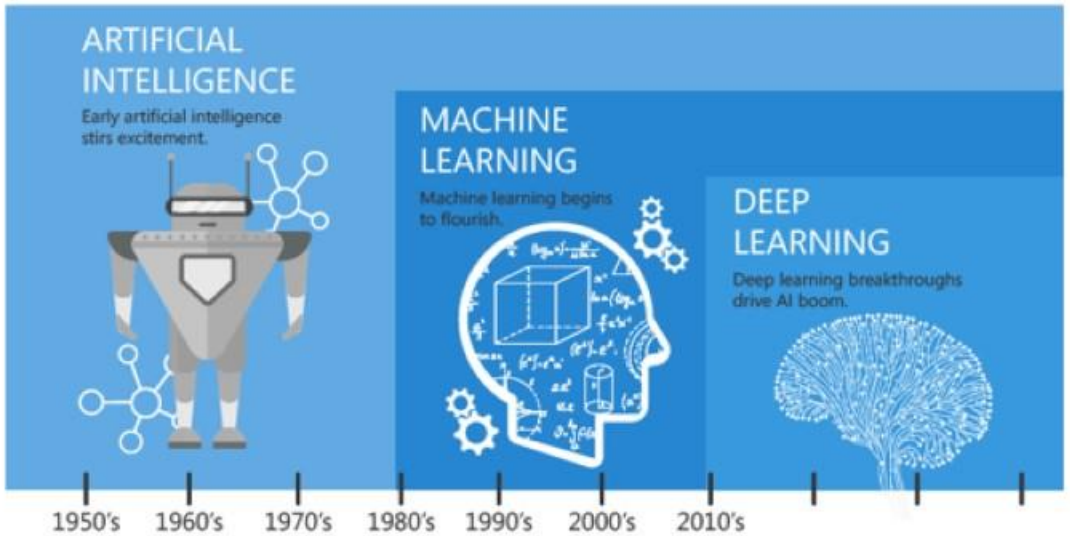


- The **best model** has **low variance** and **low bias** at the same time.
- If the model has **low variance** and **high bias**, it is called **Underfitting**
- If the model has **high variance** and **low bias**, it is called **Overfitting**
- The **worst model** has **high variance** and **high bias** at the same time.

Part 3
Machine Learning for Predictive Analytics

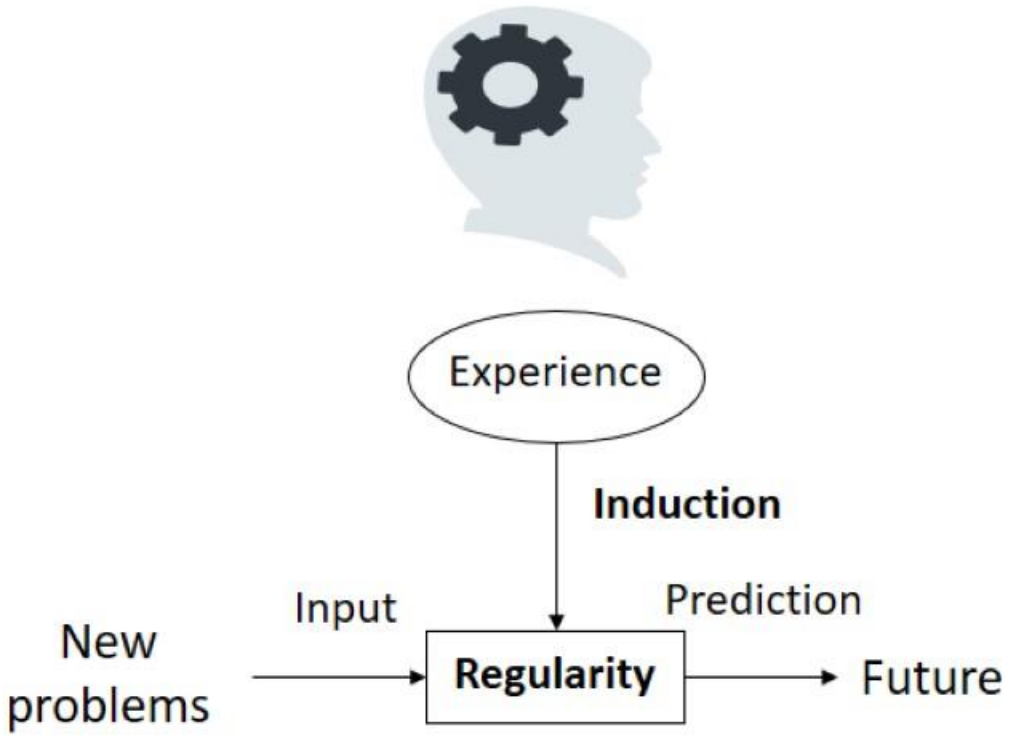
Machine Learning Overview

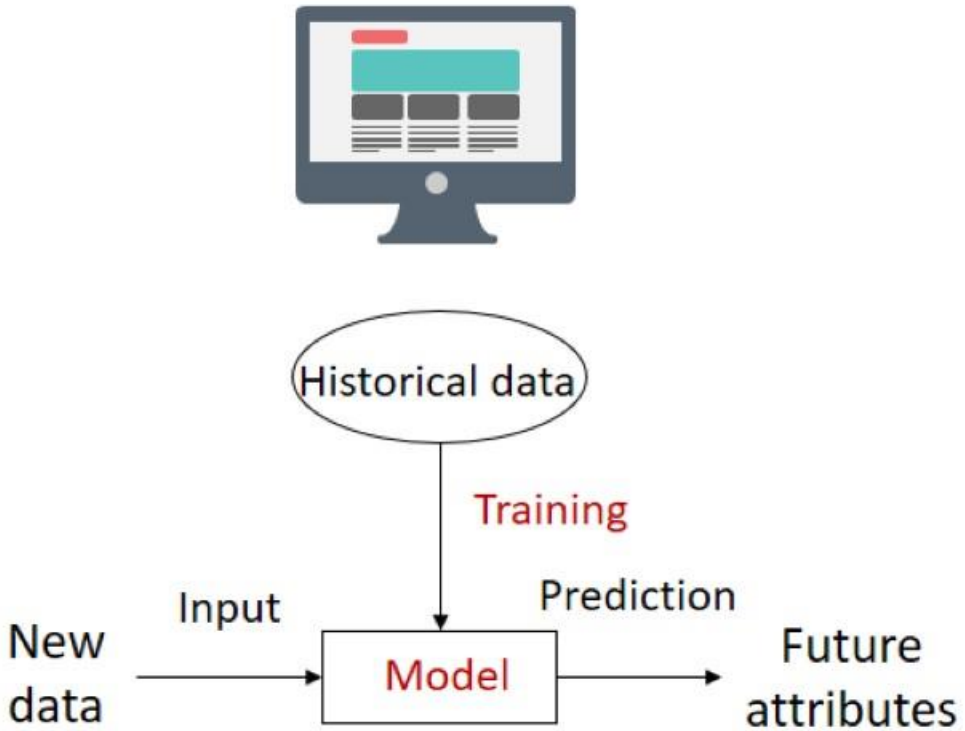
- Machine learning or ML (including deep learning (DL)) is a study of learning algorithms.
- A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.
- ML is part of AI, and DL is part of ML as well as AI



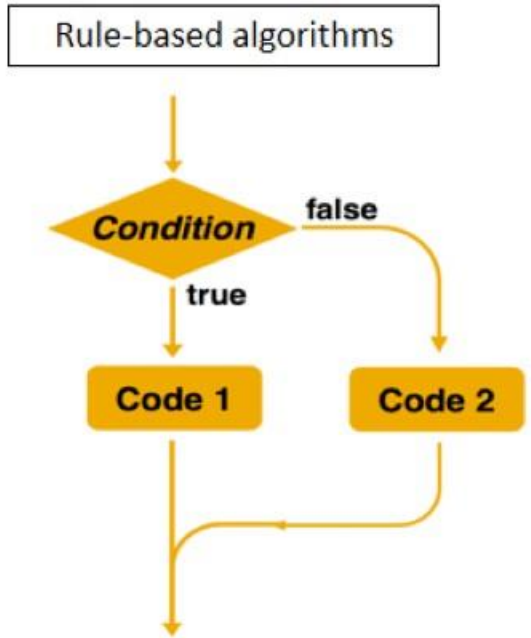
Data (Experience E) → Learning algorithms (Task T) → Basic understanding (Measure P)

Human Learning VS Machine Learning

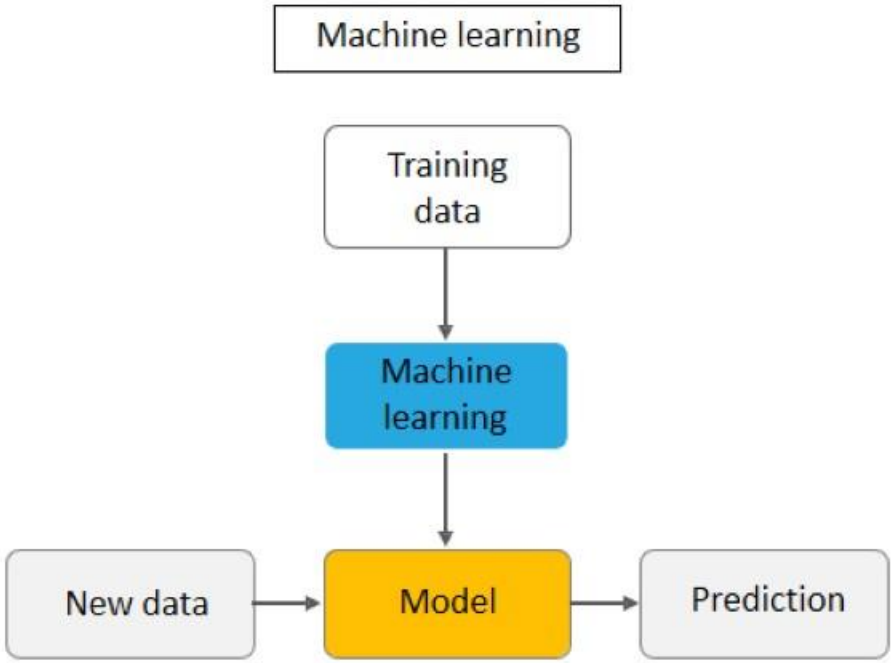




ML Algorithm VS Traditional Algorithm



- Explicit programming is used to solve problems.
- Rules are manually specified by human.
- Input needs the data and the rules
- Output is the result/value

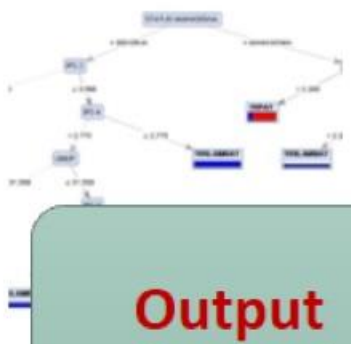


- Implicit programming is used to solve problems.
- Rules are automatically learned by machines.
- Input needs the data and the desired results
- Output is the program/model + result

Main Process of ML

Input
(Clean Dataset)

Metode
(ML/DL Algorithm)

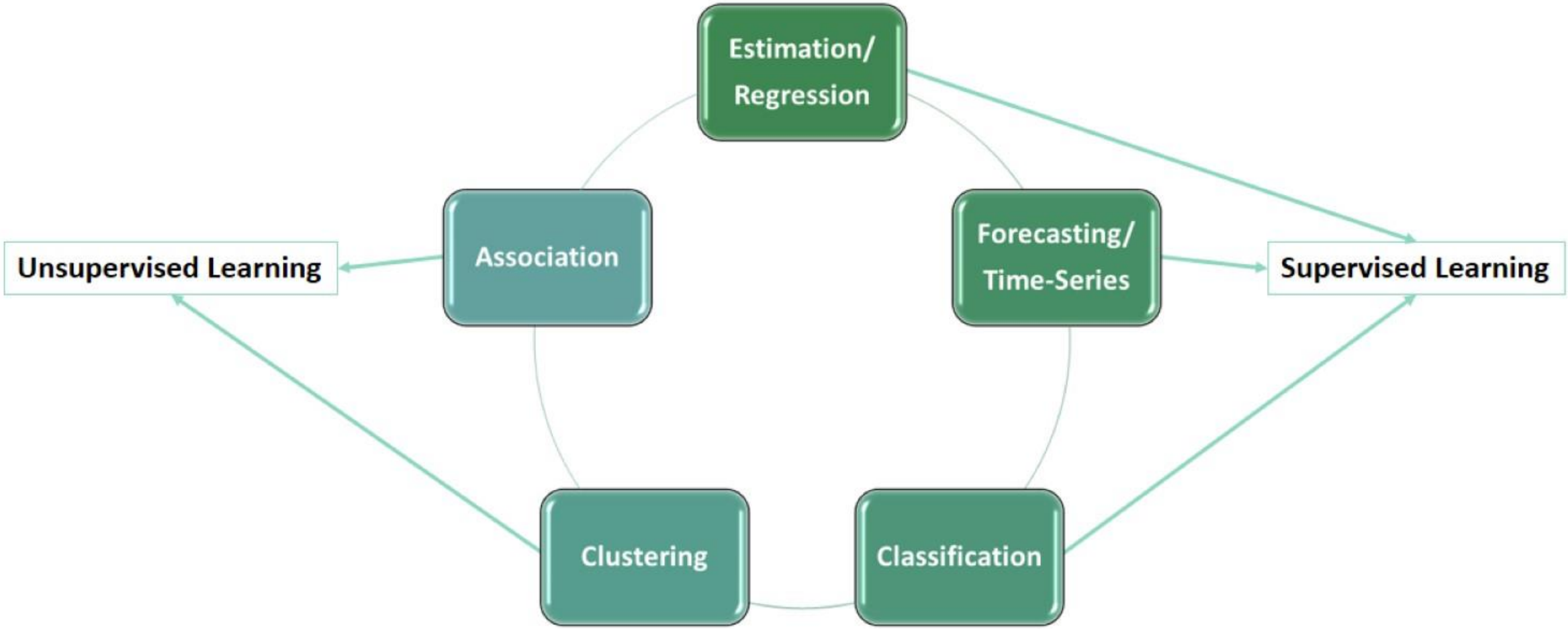


Output
(Program/ Model)

Evaluation
(Accuracy Metrics)

Output/Pattern/Knowledge of ML can be in several forms, such as formula, rules, trees, etc. depending on algorithm used

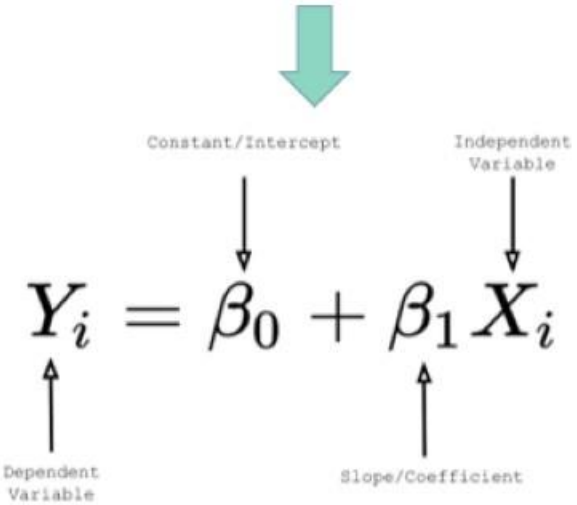
Data Science Tasks



1. Regression

- Regression → **Predict quantitative/numeric** data
- Input data may be quantitative or qualitative → If qualitative, it should be converted to **dummy variables** first
- For example, predict sales (Y) based on advertisement (X)
- If we see rigorously from the data, there is a **tendency** that the bigger the advertisement, the bigger the sales.
- That's why this method is called regression (regress → return to a former). This method believes that the factors happen in the future will be the same as the factors happen before.
- Algorithms: *Simple Linear Regression, Multiple Linear Regression, Polynomial Regression, Support Vector Regression, Decision Tree Regression, Random Forest Regression*

X	Y
Advertisement	Sales
$90	$1000
$120	$1300
$150	$1800
$100	$1200
$130	$1380
$200	??



1. Regression

- Before we go through the performance evaluation, let's try to think why do we need to do evaluation?
  1. Multiple methods are available to model the data
  2. For each method, multiple choices are available especially for model settings
  3. To choose best model, need to assess each model's performance
- Sometimes, accuracy is not the only metrics, but also inference
- Accuracy or error analysis is useful to measure the performance of predictive model, inference is useful to assess the credibility of a model
- However, not many data scientists care about inference

1. Regression

- Performance Metrics:
- **Mean Absolute Error (MAE) → Lower is better**

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

- **Mean Square Error (MSE) → Lower is better**

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

- **R-Squared (R^2) → Higher is better ($-\infty$, 1)**
  Sometimes this is called Coefficient Determination. For feature selection case, Adjusted R-Squared is preferred.

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y}_i)^2}$$

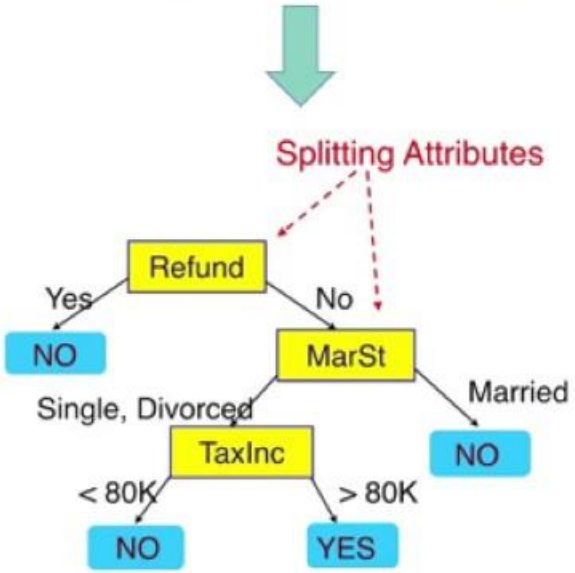
Note:

- **Predictive accuracy** is **not** the same as **goodness of fit!**
- **Naïve benchmark (average)** can be used as a baseline to check whether the model is useful or not**!**
- Metrics from **training set** explain about **model fitness** while metrics from **validation set** measure **predictive performance**. Comparing metrics of training set and validation set will **reveal** the overfitting of the model!

2. Classification

- Classification → **Predict qualitative/categorical** data
- The target can be either binary class/multi class/multi label
- Input data may be quantitative or qualitative
- For example, Loan analysis for bank customer
- How if the target is in numerical form? Categorize them first into several classes (income → from Rupiah to Low, Medium, High) → Use **Binning technique**
- Algorithms: *Decision Tree Classification, Random Forest Classification, Logistic Regression, K-nearest Neighbors (KNN), Support Vector Machines (SVM), Naive Bayes*

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



2. Classification

- Terms and definitions:

1. P: positive, indicating the number of real positive cases in the data.
2. N: negative, indicating the number of real negative cases in the data.
3. TP : true positive, indicating the number of positive cases that are correctly classified by the classifier.
4. TN: true negative, indicating the number of negative cases that are correctly classified by the classifier.
5. FP: false positive, indicating the number of positive cases that are incorrectly classified by the classifier.
6. FN: false negative, indicating the number of negative cases that are incorrectly classified by the classifier.

Confusion Matrix

Estimated amount / Actual amount	yes	no	Total
yes	TP	FN	P
no	FP	TN	N
Total	P'	N'	$P+N$

2. Classification

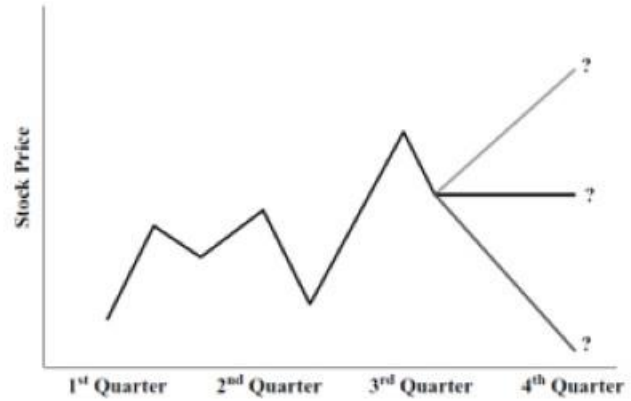
Confusion Metrics

Measurement	Ratio
Accuracy and recognition rate	$\frac{TP + TN}{P + N}$
Error rate and misclassification rate	$\frac{FP + FN}{P + N}$
Sensitivity, true positive rate, and recall	$\frac{TP}{P}$
Specificity and true negative rate	$\frac{TN}{N}$
Precision	$\frac{TP}{TP + FP}$
F_1, harmonic mean of the recall rate and precision	$\frac{2 \times precision \times recall}{precision + recall}$
F_β, where β is a non-negative real number	$\frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$

3. Time-Series

- Time-Series → **Predict time-stamped** data, the target is in the **quantitative/numerical** form
- In this method, the dataset only has 2 variables. One independent variable (X) in time-stamped unit and one dependent variable (Y) in any unit
- For example → Predict stock price (IDR)
- The comprehensive flow of time-series analysis: Plot historical data → Recognize the pattern → Plot ACF/PACF & Estimate the parameters of model → Decide and execute some algorithms → Pick the best algorithm based on Evaluation (MAD, MSE, MAPE, RMSE)

Row No.	Close	Date
1	1286.570	Apr 11, 2006
2	1288.120	Apr 12, 2006
3	1289.120	Apr 13, 2006
4	1285.330	Apr 17, 2006
5	1307.280	Apr 18, 2006
6	1309.930	Apr 19, 2006
7	1311.460	Apr 20, 2006
8	1311.280	Apr 21, 2006
9	1308.110	Apr 24, 2006



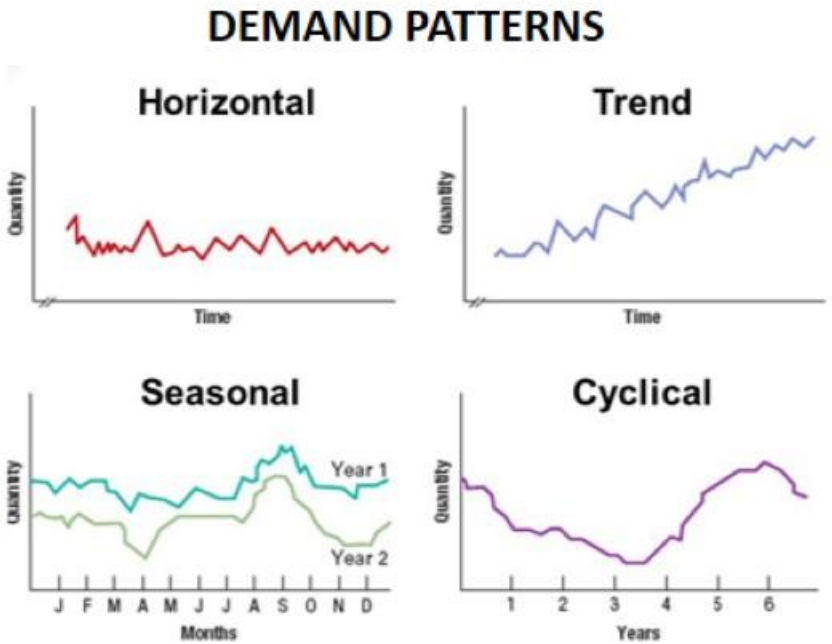
3. Time-Series

- In Time-Series, there are several **demand patterns** that can drive what kind of appropriate algorithms we should use
- Time-Series Techniques:

1. **Qualitative**: Delphi, Market Survey, Expert Judgement
2. **Quantitative**: Statistical Model & ML Model

a. Statistical Model → Naïve, Moving Average, Exponential Smoothing, ARIMA, Prophet

b. ML/Causal/Associative Model → All regression algorithms can be used → Including DL Model → LSTM and its evolution



Hyperparameter Tuning

Hyperparameter VS Parameter

- Hyperparameters are parameters that are set before learning, their values will not change until the end of learning process
- Parameters are values that cannot be set in advance, but they are set automatically when ML algorithm learns

Examples

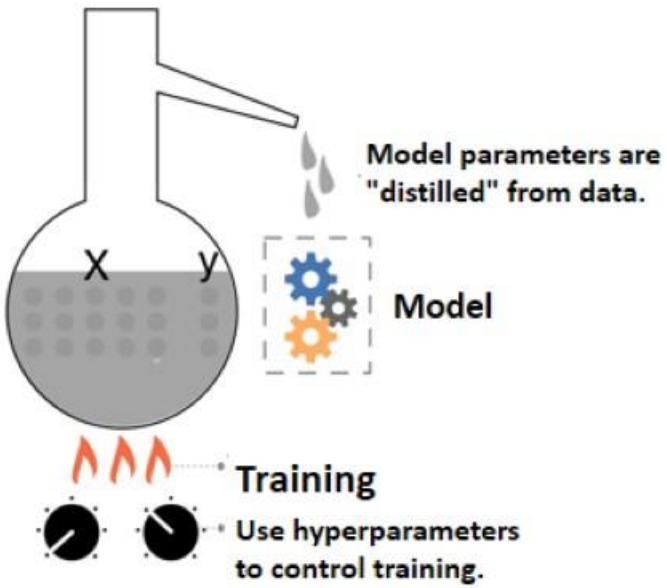
a. Neural Network (NN) → Hyperparameters are Number of Hidden Layers, the Number of Nodes, and Training Rate → Weight & Bias are parameters

b. K-NN → K or number of neighbors included in the voting process.

c. Random Forest → Number of Trees

d. Simple Linear Regression → None, but Polynomial Regression has order and Lasso Regression has lambda or regularization term

e. Support Vector Machines (SVM) → C and σ



Analogy – Distillation Process

Hyperparameter Tuning

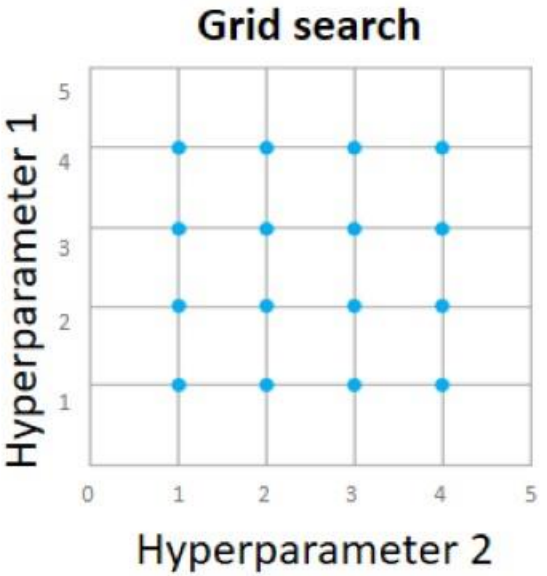
- Hyperparameter tuning can be seen as searching a process because we try to search optimal value such that giving the best performance of model
- Several ways to optimize hyperparameter:

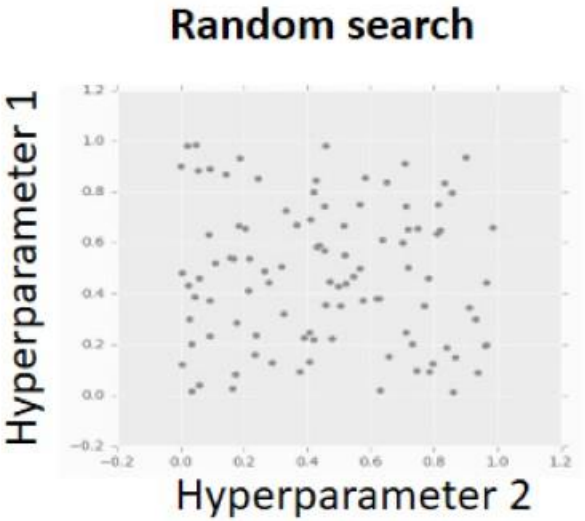
a. **Random Search** → Uses some arbitrary values randomly that is feasible, then pick the best one → Good for big space without any directions

b. **Grid Search** → Exhaustively Uses all possible hyperparameter combinations to form a hyperparameter grid → Good for small/big space with direction

c. **Bayesian Search** → Uses Bayes Theorem to direct the search.

d. **Metaheuristics Search** → Uses intelligent algorithm inspired from nature such as ACO, PSO, GA → Good for small/big space with direction





Parameter Optimization

How about optimizing parameter?

- It is also possible to optimize parameter in ML algorithms
- However, ML algorithm itself has local optimizer, such as least squares optimization in linear regression and gradient descent algorithm in Neural Network (NN)
- Whenever we use another optimizer for optimizing parameter in ML, actually ML algorithm do NOT learn by itself, we only use ML model as objective function.
- Usually, changing optimizer does not significantly affect the performance of ML and even can be worse. But for complex problem, this may help because sometimes local optimizer in ML is trapped in local optima, not global optima. So it is better to find out!
- All optimizers showed in Hyperparameter is also able to do Parameter Optimization

Examples

a. NN → Weight & Bias are optimized using PSO (PSO-based NN)

b. Linear Regression → coefficients & constant are optimized using GA (GA-based linear regression).

Which Algorithm is The Best?

- There is **no the best algorithm** for **all problems**
- The answer is "**No Free Lunch Theory**" → introduced by Wolpert and Macready (1997). They stated the overall performance of algorithms or methods on a wide range of test problems would be same.
- Thus the performance of an algorithm might be very good for small or undemanding problems but it's performance for large and difficult problems would tail off and vice versa
- What we need to do in order to get the best performance is by conducting careful **systematic experiments**

Effectivity VS Efficiency in ML

- Effectivity refers to the algorithm performance, efficiency refers to the computation time and cost
- For **practical point of view** or in real cases, **performance** is not only what we are looking for, sometimes **computational efficiency** is more important → How to **retrieve** the data, **embed** the model, and **re-train** the model is **crucial** → System development is **needed**
- For **academic point of view**, what they are focusing is to discover better and better algorithm → Usually, **no need** to **retrieve** the data as well as **embed** and **re-train** the model, but try as many as datasets is **needed** to verify the proposed algorithm → System development is **not needed** sometimes

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, *1*(1), 67-82.

Part 4
Machine Learning for Descriptive Analytics

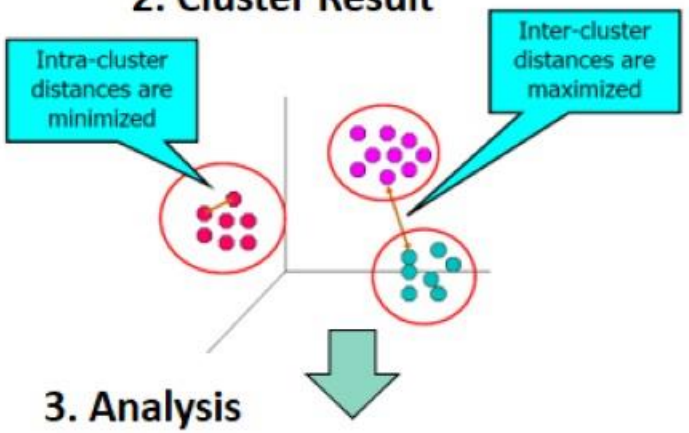
4. Clustering

- Clustering or Cluster analysis → **Group the data** based on the data proximity
- Usually used in the first phase of data science project
- Clustering algorithms try to *minimize* **intra-cluster** & *maximize* **inter-cluster** distance
- The features can be quantitative or qualitative, mostly **quantitative**
- For example → customer segmentation for marketing, case study from (Rizqi, 2020)

1. Dataset

CustomerID	TotalRevenue	FreqTransaction	DaysasCustomer
12346	77183.60	1	326
12347	4310.00	7	367
12348	1437.24	4	358
12349	1457.55	1	19
12350	294.40	1	310

2. Cluster Result



3. Analysis

	TotalRevenue	FreqTransaction	DaysasCustomer
Cluster 1	4315.750230	3.344828	221.528736
Cluster 2	2744.662642	4.431818	82.369318
Cluster 3	2012.831865	4.209902	10.938112

4. Marketing Strategy

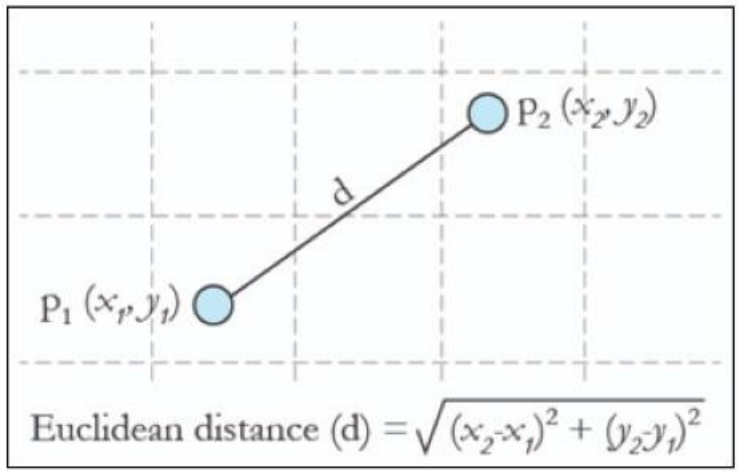
Rizqi, Z. U., Khairunisa, A., & Puspitasari, N. A. (2020). Determinasi Strategi Markdown berbasis Clustering-Data Mining (Studi Kasus pada Usaha Salon di Yogyakarta). *IENACO (Industrial Engineering National Conference) 8.*

4. Clustering

How to measure the proximity? → calculate the **similarity** and **dissimilarity** of data

- Similarity [0, 1]→ higher value is better → Cosine, Correlation, & Covariance
- Dissimilarities [0, ∞]→ lower value is better → **Euclidean Distance**, Manhattan Distance, Minkowski Distance, Jaccard Distance, & Hamming Distance

Mostly, clustering uses **Euclidean Distance** with following formula:



4. Clustering

- **Two main issues in clustering:**

1. **Determining number of clusters (K)** (depends on algorithm later) → 3 ways to solve it:

a. K can be defined by the domain expert → If there is no knowledge, use other approaches

b. Try different K → Analyze it using Elbow approach or Coefficient Silhouette

c. Automatic Clustering → Hybrid clustering algorithm with metaheuristics

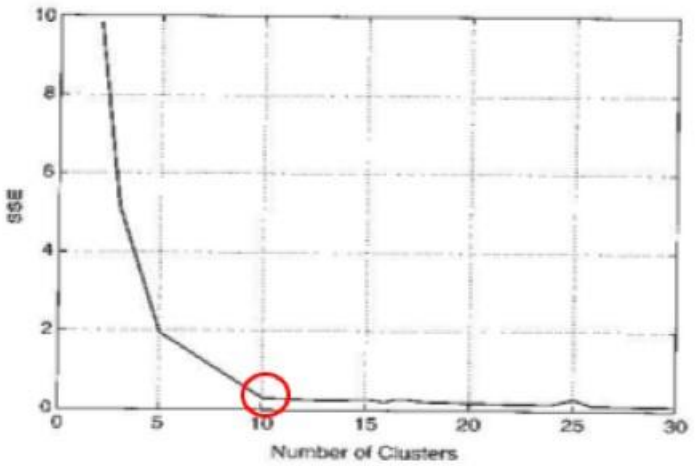
2. **Clustering data** → Use clustering algorithms

a. Partitional (K is defined first, follows iterative process) → K-means, Fuzzy C-Means, K-modes

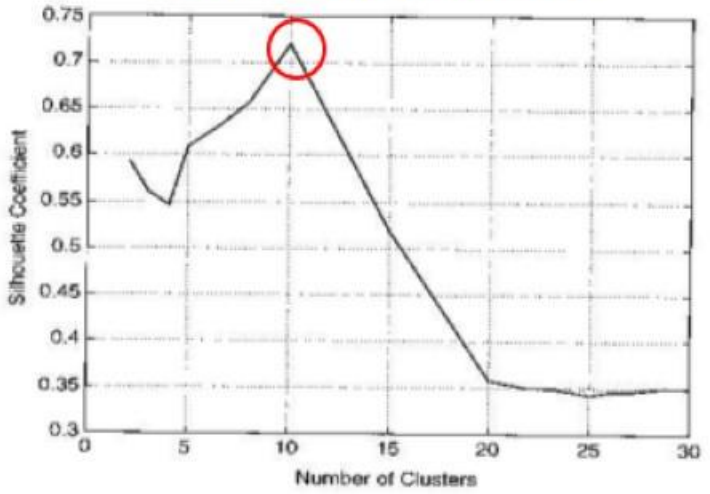
b. Hierarchical (K is defined later based on threshold in dendogram) → Agglomerative (Bottom-up), Divisive (Top-down)

c. Density-based (K is defined automatically, robust to outliers) → DBSCAN, OPTICS, DENCLUE

Elbow technique



Coefficient Silhouette



4. Clustering

- Most common metric in evaluating performance of clustering is **Sum of Squared Error (SSE)** → Lower is better

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

- *Dist* means distance or subtraction of m_i and x → x is a data point in cluster C_i and m_i is the representative or center point for cluster C_i
- SSE is also driven by the number of cluster (K) → The higher K, the lower SSE → Fair algorithm evaluation needs to be compared in the same K
- Another way to evaluate **cluster validity** is to use dataset with known target, then assess the clustering result based on known target

5. Association

- Association/Associative Learning/Association Rule Mining → **Finding the rules** from the data relationship → The rules tell us which data that **often appear together**
- In retail business, this method is usually called as ***Market Basket Analysis*** (AR-MBA) or ***Affinity Analysis***. The data mainly come from Point-of-Sales (POS) system
- In 2001, Walmart used this approach and found that Beer and Diaper has frequent rule. They then brought these 2 products closer and got 35% profit higher!
- Association output is easily represented as If (antecedent) → then (consequent)
- For example, real case from electronic retail business (Rizqi, 2019)

Transaction	Type of Goods
1	Tempered Glass, HP Case, Charger, Earphone, Powerbank
2	HP Case, USB Cable, Printer Ink, Wireless Mouse, SD Card
3	HP Case, Tempered Glass, Powerbank, Joystick, Earphone
4	HP Case, Tempered Glass, USB Cable, Charger, SD Card
5	HP Case, Tempered Glass, Earphone, Powerbank, SD Card

If {Harddisk} → Then {Charger}

Lift Ratio 1.029, Confidence 85%, & Support 39%

Rizqi, Z. U. (2019). Implementation of Association Rule-Market Basket Analysis in Determining Product Bundling Strategy: Case Study of Retail Businesses in Indonesia. *Proceedings of Industrial Engineering and Operations Management*, pp. 153-160

5. Association

- Association algorithms: Apriori, FP-Growth, & ECLAT
- Rule Evaluation Metrics:

1. Support (s) [0, 1] → eliminates ***uninteresting*** rules
2. Confidence (c) [0, 1] → measures the ***reliability*** of the rules
3. Lift Ratio (lr) [0, ∞] → measures the ***validity*** of the rules → if LR < 1, not valid; if LR > 1, valid; if LR = 1, no relationship

$$Rule: X \Rightarrow Y \rightarrow \begin{cases} Support = \frac{frq(X,Y)}{N} \\ Confidence = \frac{frq(X,Y)}{frq(X)} \\ Lift = \frac{Support(X,Y)}{Supp(X) \times Supp(Y)} \end{cases}$$

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Exercise:

- How is the support, confidence, and lift ratio of *{Milk, Diaper}* → *Beer*?
- Total transaction (N) = 5; X = Milk & Diaper; Y = Beer
- Conclusion: From all transactions, customers bought Milk, Diaper, & Beer simultaneously by 40%, & we believe that Milk & Diaper always be together with Beer by 67%, and it is valid.

$$s = \frac{2}{5} = 0.4$$

$$c = \frac{2}{3} = 0.67$$

$$lr = \frac{0.4}{3/5 \; x \; 3/5} = 1,11$$

5. Association

- Before association algorithm is performed, we need to define first ***minimum support threshold (minsup)*** [0, 1] to take frequent itemset only and ***minimum confidence threshold (minconf)*** to take the important rules [0, 1]
- The goal of association is to find all rules having **support ≥ minsup threshold** & **confidence ≥ minconf threshold**
- There is no formal convention how to define the thresholds, the best practice is to look at the data and try several values (case-by-case)
- Best practice:

1. If the data is big, then denumerator will be big. Therefore, just use small threshold first. Otherwise, use bigger value.
2. Let's say we try 50% for thresholds. Then we see, if the rules are too many (based on our judgement), cut them by increasing the thresholds into 60%, and so on until the desired rules number are obtained.

Advanced Applications

- In data science project, we can use several methods at the same time for the same dataset. For example:

1. ***Semi-supervised learning*** (from unsupervised to supervised) → For example, customer data can be used for the **clustering** first to categorize the income level (high, medium, low) → The income level can be added in the dataset as target data → Company can create **prediction** model to classify income level for prospect customer then make strategy for that customer
2. ***Two-Stage Approach*** → **Cluster** the customer data based on profitability first (potential, fickle, miserly) → Each cluster data is then analyzed using **Association Rule** to get specific marketing strategy → Each cluster will have different rules

Data Science Methods & Applications (Summary)

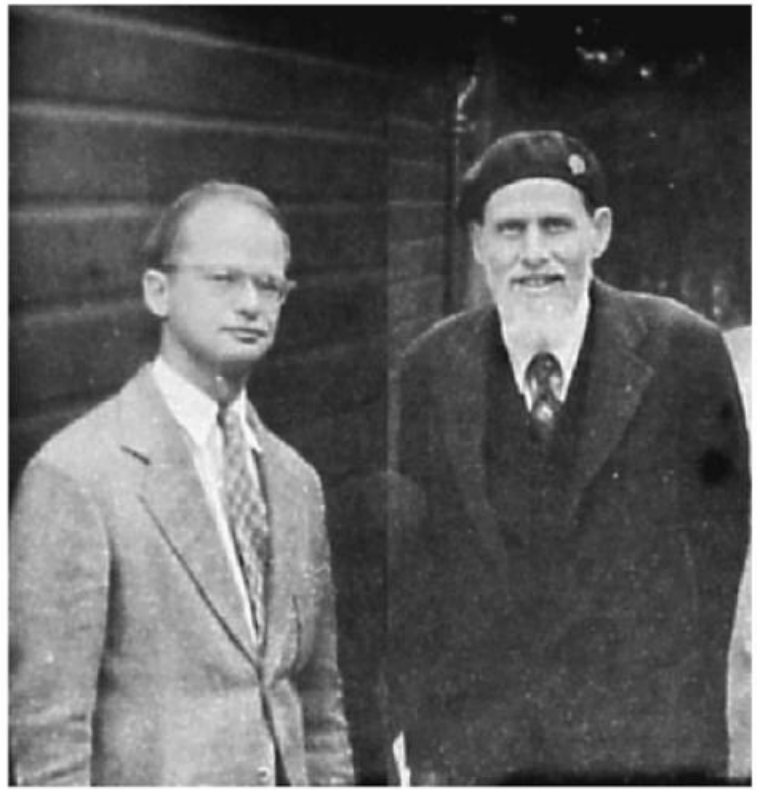
1. Regression: **Predict quantitative** data, e.g. predict sales (unit), predict house price (USD).
2. Classification: **Predict qualitative** data either binary or multiclass, e.g. fraud detection, classifying spam, disease prediction.
3. Time-Series: **Predict time-stamped** data, target is in quantitative form, e.g. predict stock price (IDR), predict demand (unit).
4. Clustering: **Group** the data, the features can be quantitative or qualitative, mostly **quantitative**, e.g. customer segmentation.
5. Association: **Finding the rule** from the data relationship, the features can be quantitative or qualitative, mostly **qualitative**, e.g. layout analysis, symptom analysis in medical, credit analysis.

Supplement
Moving from Machine Learning to Deep Learning

Deep Learning

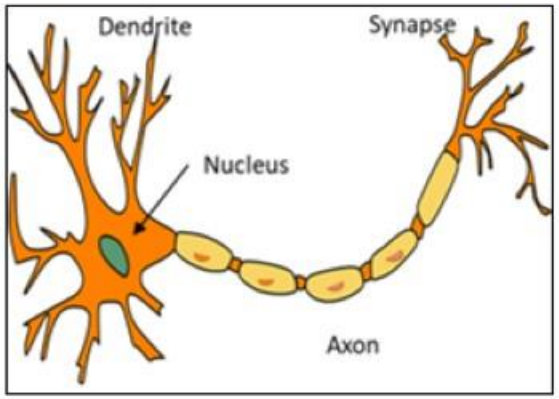
What is Deep Learning (DL)/Deep Neural Network (DNN)/Deep Structured Learning (DSL)?

- Deep learning is actually **Neural Network (NN)** with many hidden layers so that it seems to be deep (wide).
- NN was proposed by McCulloh & Pitts (1943).
- It imitates the human brain structure and functions. NN connects neurons that work in parallel. Human brain consists of about 100 billion neurons
- NN itself is actually machine learning algorithm



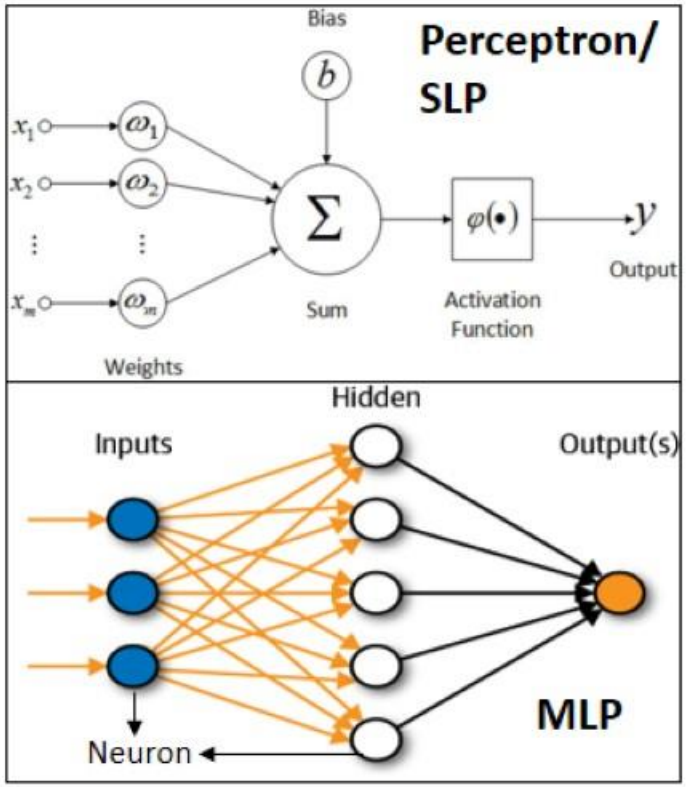
McCulloh & Pitts

Deep Learning Architecture

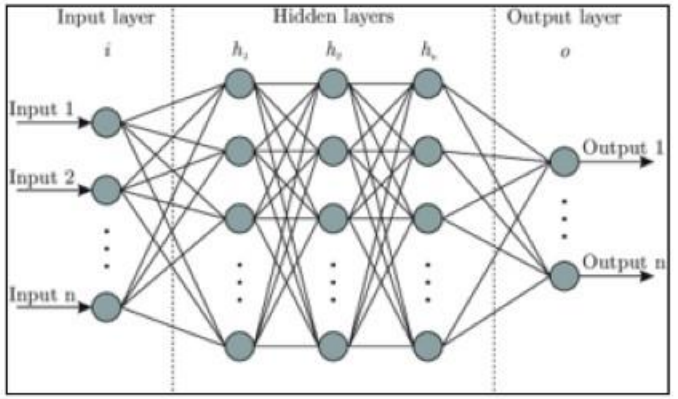


Human Neural Network

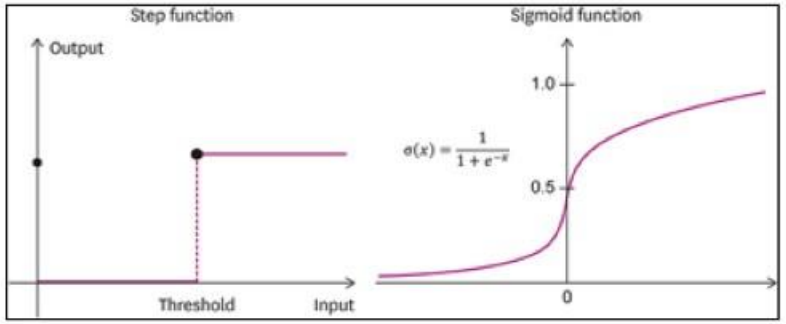
Output of Neural Network is the wight (ω) and bias (θ)!



Artificial Neural Network (ANN)

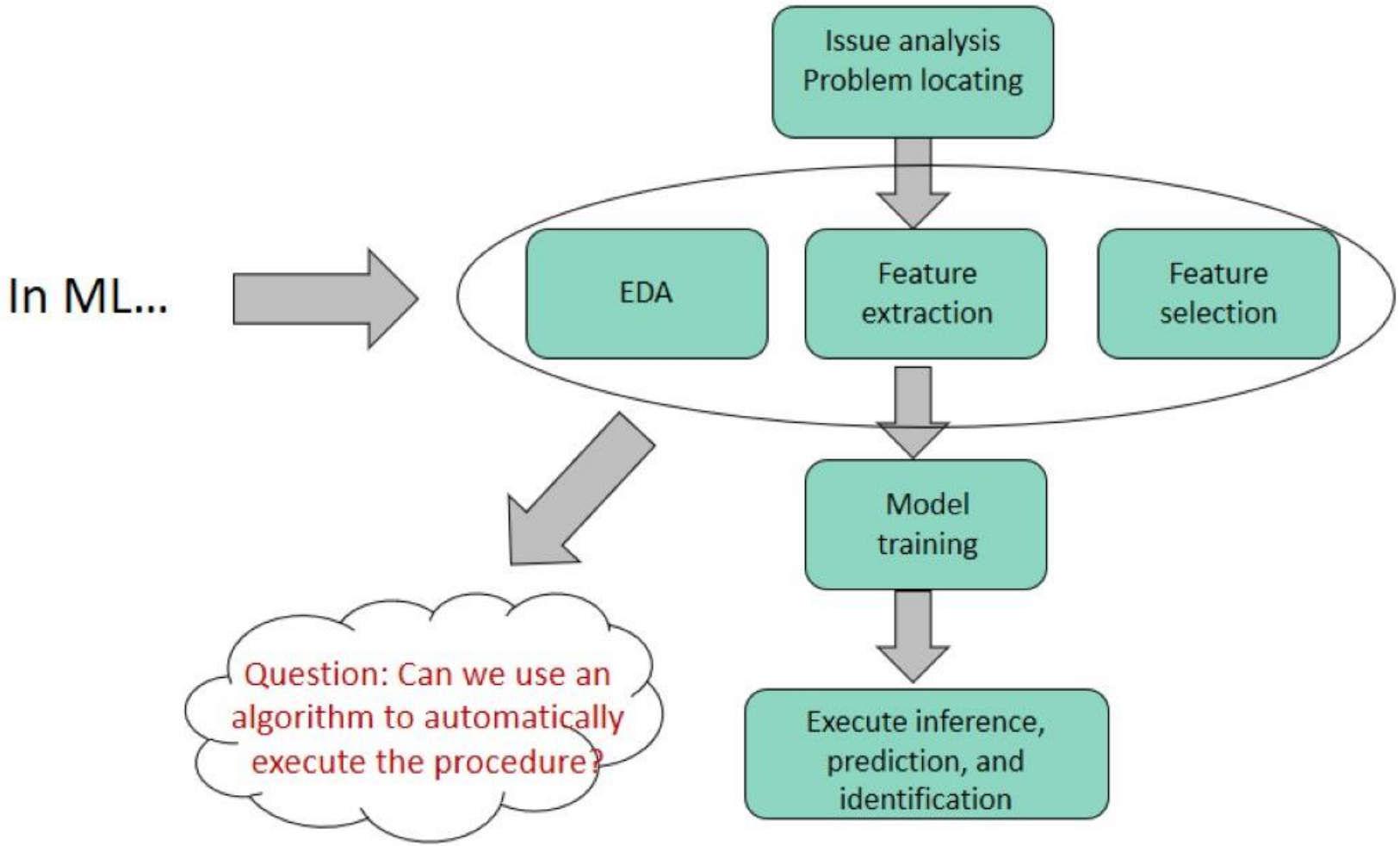


Deep Neural Network

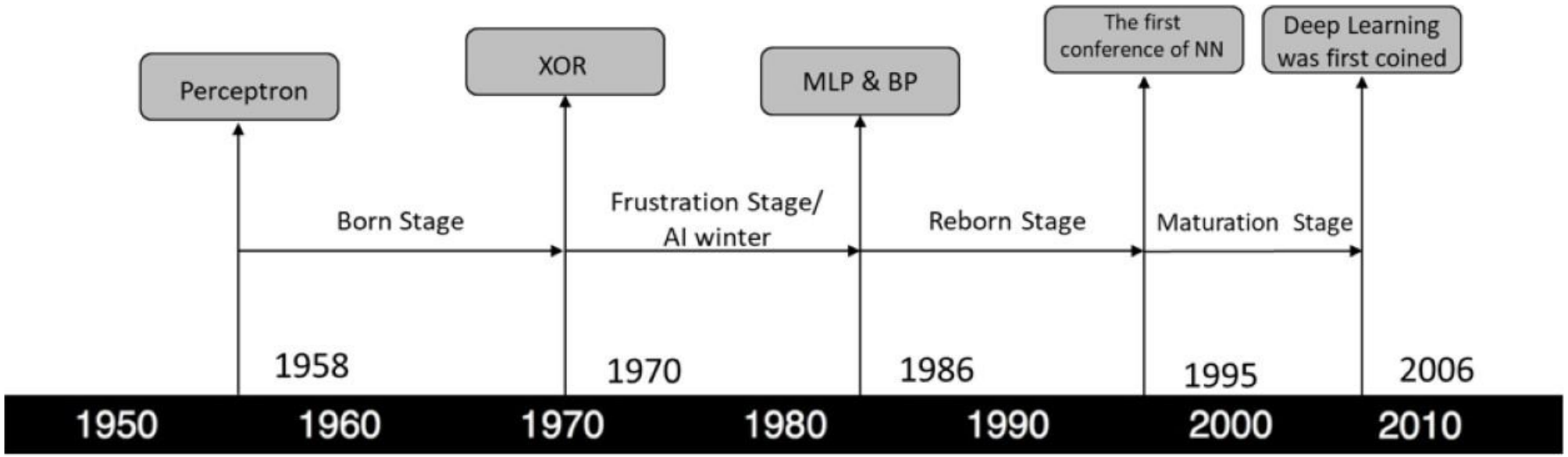


- **Perceptron** or **Single Layer Perceptron** (SLP) only has **input** and **output layer** with **step function** (**linear**)
- **Multi Layer Perceptron** (MLP) has additional layer called **hidden layer** with **sigmoid function** (**non-linear**)
- When MLP has a lot of hidden layers, it is called **Deep learning**. Usually, more than 3 layers.

DL vs ML

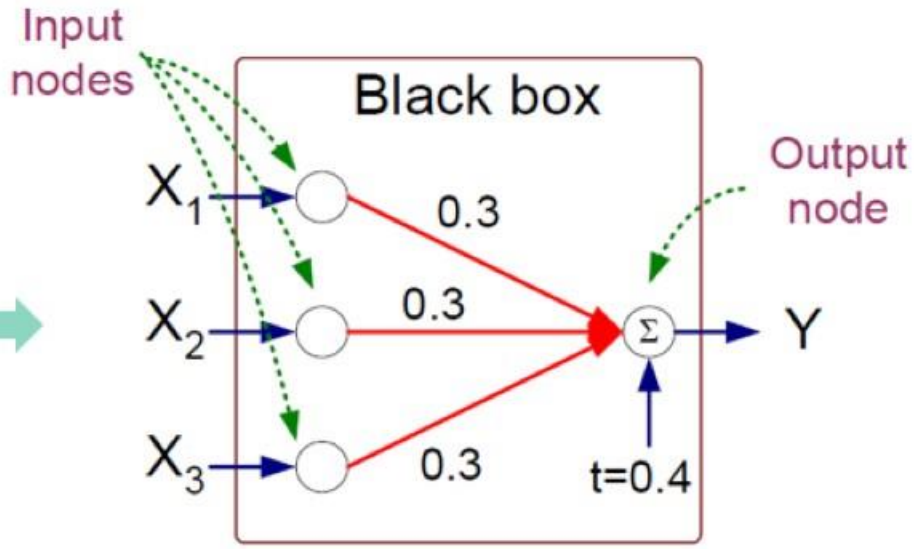


Deep Learning Milestone



Perceptron

X_1	X_2	X_3	Y
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	1	0
0	1	0	0
0	1	1	1
0	0	0	0



Perceptron Model:

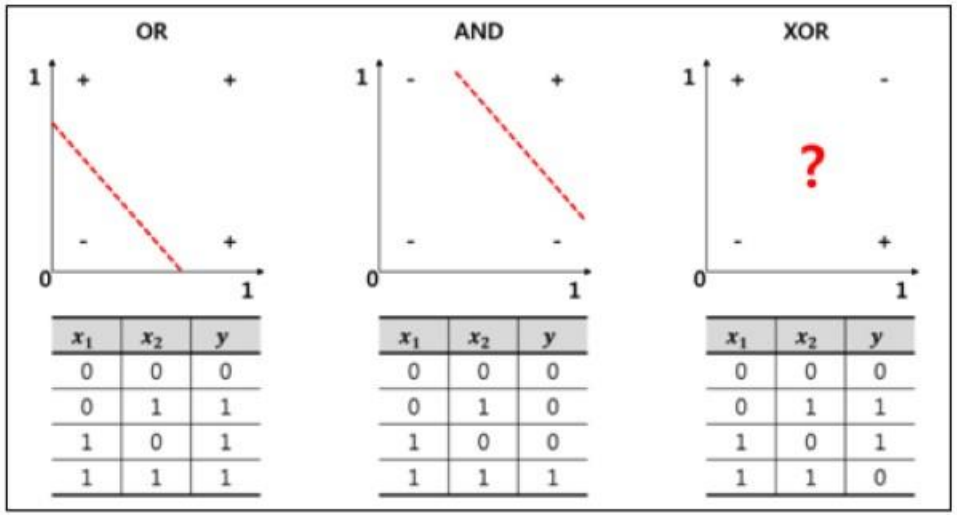
$$Y = I(\sum_i w_i X_i - t) \quad \text{or}$$

$$Y = sign(\sum_i w_i X_i - t)$$

Key Answers:
Output Y is 1, if at least two of the three inputs are equal to 1.

$$Y = I(0.3X_1 + 0.3X_2 + 0.3X_3 - 0.4 > 0)$$

$$\text{where } I(z) = \begin{cases} 1 & \text{if } z \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$



x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Perceptron can only solve linear problem. How about non-linear problems like XOR problem?

Perceptron

1. Set up network parameters, learning rate η.
2. Randomly set up weights, W, and bias θ.
3. Input a training sample with input matrix, X, and target output vector, T.
4. Calculate the output vector, Y.

$$net_j = \sum_i W_{ij} X_i - \theta_j$$

Yj = 1 if netj > 0

= 0 if netj ≦ 0

5. Calculate δ

$$\delta_j = T_j - Y_j$$

6. Calculate ΔW and Δ θ

$$\Delta W_{ij} = \eta \delta_j X_i$$
$$\theta_j = -\eta \delta_j$$

7. Update W and θ。

$$W_{ij} = W_{ij} + \Delta W_{ij}$$
$$\theta_j = \theta_j + \Delta \theta_j$$

8. Go back to Steps 3 to 7 and repeat for the next sample. If the termination criterion is met, then stop.

Multi-Layer Perceptron

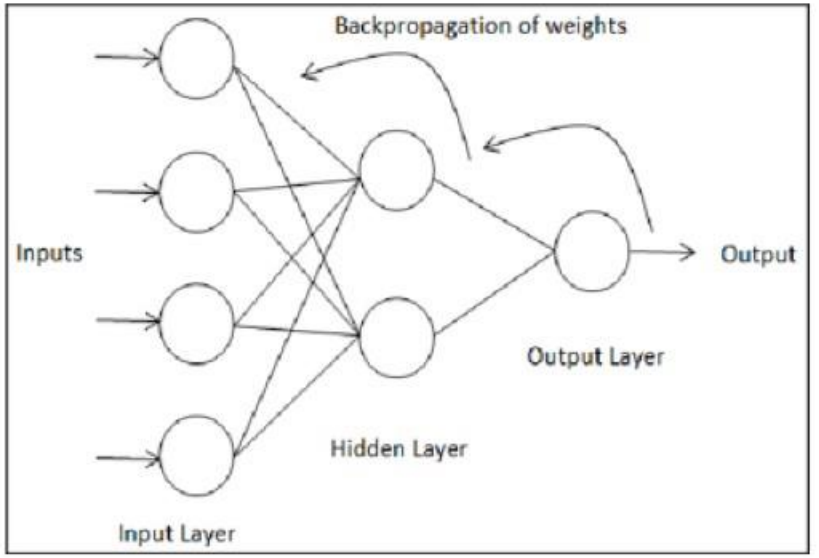
- MLP has 3 layers:

1. Input layer: Use linear activation function: f(x)=x
2. Hidden layer: Use non linear activation function (sigmoid, tanh, ReLU, etc.)
3. Output layer: Use non linear activation function.

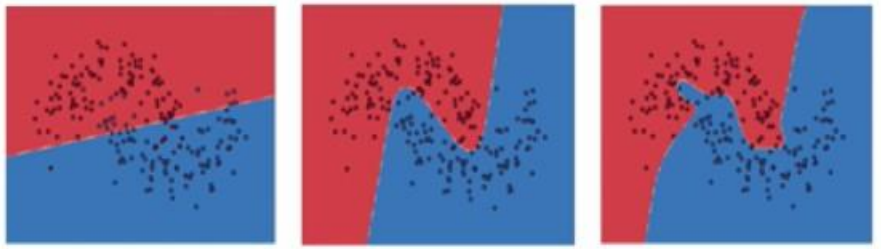
- MLP is proposed by using **Back-Propagation** (BP) Algorithm that has forward & backward propagation process with gradient steepest descent to minimize error function

- There are 3 main parameters need to be set up:

1. The number of hidden layer nodes
2. The number of hidden layers
3. Training rate [0.01, 0.1] or [0.05, 0.25]



Impacts of Hidden Layers on A Neural Network



Multi-Layer Perceptron

1. Set up network parameters.
2. Randomly set up weight matrixes, W_xh and W_hy, and bias vectors, θ_h and θ_y .
3. Input a training sample with input vector, X, and target output vector, T.
4. Calculate the output vector, Y.

(a) Calculate the output vector, H, for hidden layer.

$$net_h = \sum_i W\_xh_{ih} \cdot X_i - \theta\_h_h$$

$$H_h = f(net_h) = \frac{1}{1+\exp^{-net_h}}$$

(b) Calculate the output vector for output layer.

$$net_j = \sum_h W\_hy_{hj} \cdot H_h - \theta\_y_j$$

$$Y_j = f(net_j) = \frac{1}{1+\exp^{-net_j}}$$

5. Calculate the deltas, δ.

(a) Calculate the deltas, δ, for output layer.

$$\delta_j = Y_j(1-Y_j)(T_j - Y_j)$$

(b) Calculate the deltas for hidden layer.

$$\delta_h = H_h(1-H_h)\sum_j W\_hy_{hj}\delta_j$$

6. Calculate ΔW and Δθ.

(a) Calculate ΔW_hy and Δθ_y for output layer.

$$\Delta W\_hy_{hj} = \eta\delta_j H_h$$

$$\Delta\theta\_y_j = -\eta\delta_j$$

(b) Calculate ΔW_xh and Δθ_h for hidden layer.

$$\Delta W\_xh_{ih} = \eta\delta_h X_i$$

$$\Delta\theta\_h_h = -\eta\delta_h$$

7. Update weight matrix, W, and bias vector, θ.

(a) Update W_hy and θ_y for output layer.

$$W\_hy_{hj} = W\_hy_{hj} + \Delta W\_hy_{hj}$$

$$\theta\_y_j = \theta\_y_j + \Delta\theta\_y_j$$

(b) Update W_xh and θ_h for hidden layer.

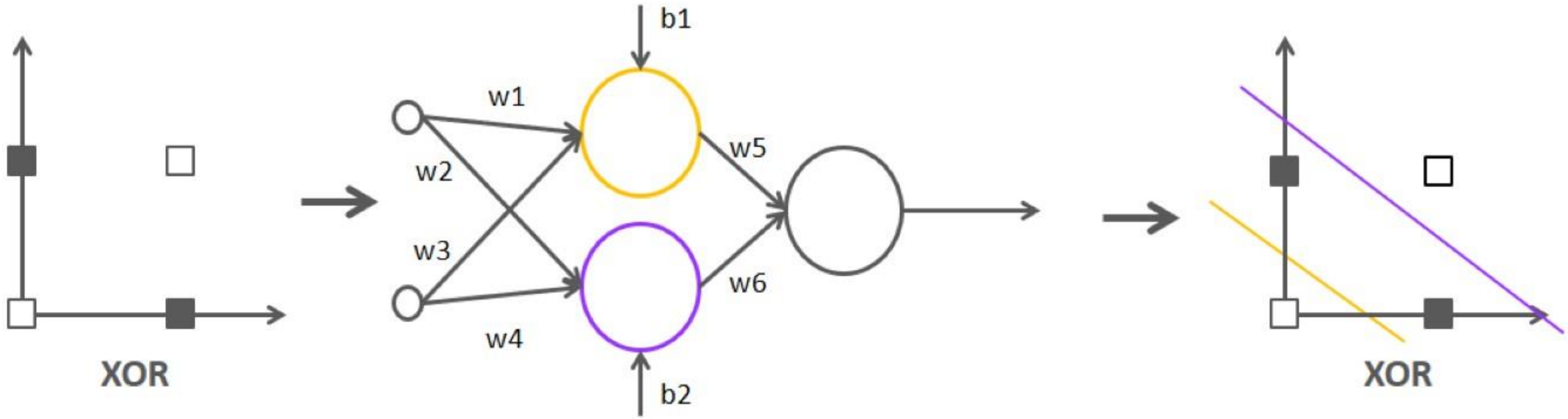
$$W\_xh_{ih} = W\_xh_{ih} + \Delta W\_xh_{ih}$$

$$\theta\_h_h = \theta\_h_h + \Delta\theta\_h_h$$

8. Repeat Steps 3 to 7. If the stop criterion is met, then stop.

Multi-Layer Perceptron

How MLP solves XOR problem

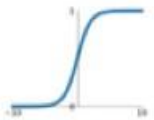


Supplements in DL

Important things in using DL:

1. Feature scaling is important part, either X or Y variable, either normalization or standardization. It affects the learning of DL.
2. Since DL use optimization algorithm embedded, consider the appropriate loss function based on your purpose → Remember the metrics is the key!
3. Input layer does not need activation function, only for hidden and output layer. Practically, we can use the same activation function for all hidden layers such as ReLU, Sigmoid, & Tanh
4. Determining the configurations of output layer is important (activation function & number of nodes): Binary Classification VS Multi-Class Classification VS Regression will be different.
5. Training rate determines learning step of DL affecting how fast the convergence and the effectivity of algorithm

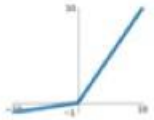
Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$



tanh
$\tanh(x)$

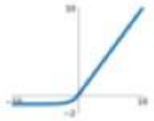
ReLU
$\max(0, x)$

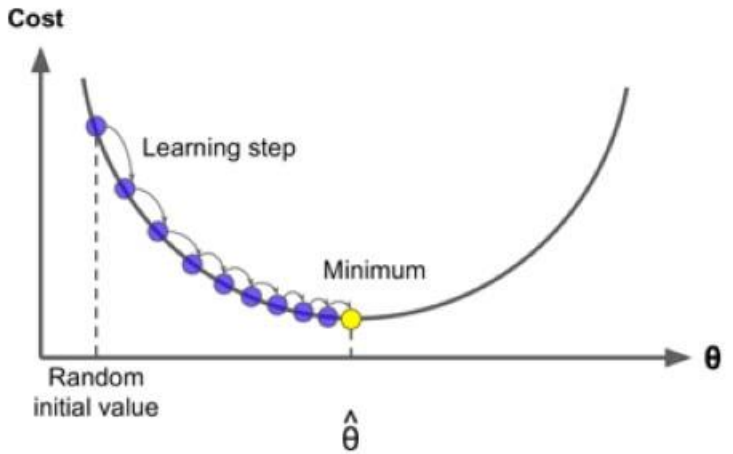
Leaky ReLU
$\max(0.1x, x)$



Maxout
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU
$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$





Architectural Decisions in ANN

- In ANN, there are 2 architectural design parameters driving ANN performance:
  1. The number of hidden layer nodes
  2. The number of hidden layers
- Some strategies to optimize it:
  1. For efficiency purpose → Literature suggestion
  2. All optimizers showed in Hyperparameter optimization

Hidden Layer Nodes Strategy

1. $(N_I \times N_O)^{\frac{1}{2}}$
2. $\frac{1}{2}(N_I \times N_O)$
3. $\frac{1}{2}(N_I \times N_O)+(N_P)^{\frac{1}{2}}$
4. $2 \times N_I + 1$

N1: The number of input layer node
No: The number of output-layer nodes
Np: The number of hidden layers

Hidden Layer Strategy

1. Practically, most of problems can be solved by using 1 or 2 hidden layers
2. If the problem is very complicated → Use > 3 hidden layers (DL)
3. Rule of thumb is by trying from 1, then increase one-by-one to see the effect

Advantage & Disadvantage of ANN

(Kumar et al., 2020)

Model Name	Advantages	Disadvantages
Single Layer Perceptron	• Less computation—time Easy to setup	• Can only be used in linearly separable data
Multi-Layer Perceptron	• Can be used for complex problems	• Need more time for training • Can get stuck in local minima

Kumar, P., Lai, S. H., Wong, J. K., Mohd, N. S., Kamal, M. R., Afan, H. A., ... & El-Shafie, A. (2020). Review of nitrogen compounds prediction in water bodies using artificial neural networks and other models. *Sustainability*, *12*(11), 4359.

Most Used DL Algorithms

1. Regression: Artificial Neural Network (ANN).
2. Classification: Artificial Neural Network (ANN).
3. Time-Series: ANN, Recurrent Neural Network (RNN), especially Long Short-term Memory (LSTM)
4. Clustering: Self-Organizing Map (SOM) or Kohonen Network (KN).
5. Association: Helps in data preprocessing (feature selection), Selected features then are inputted to Deep learning algorithm

DL is further used in other applications or areas such as:

- Computer vision: Convolutional Neural Networks (CNN), Autoencoders
- Speech recognition: RNN
- Natural language processing: CNN, RNN, Generative Adversarial Network (GAN)
- Recommendation System: Boltzmann Machines

Evaluation

1. What is standard methodology of Data Science Process?
2. What stage is the most time-consuming in Data Science Project?
3. Mention 3 reasons why we choose Python!
4. Mention 5 main tasks of data science!
5. Explain the difference between regression and classification!
6. Mention 2 algorithms that can be used for regression and classification!
7. Explain the differences between supervised and unsupervised learning!
8. Mention 2 applications of Association Rule!
9. Mention 3 algorithms that can be used for clustering!

THANK YOU
谢谢