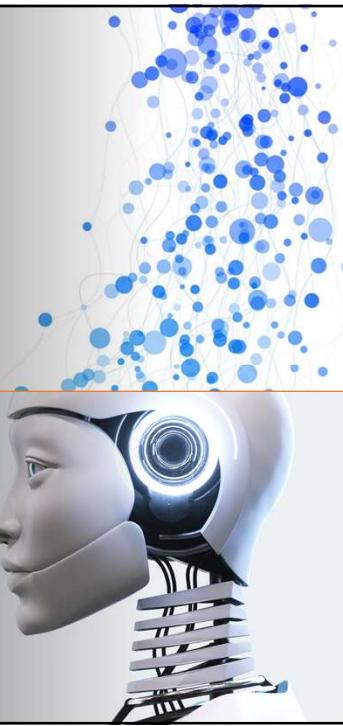


# Machine Learning Foundations

## An Introduction to your AI Journey

Sumudu Tennakoon, PhD



© 2021 Sumudu Tennakoon

1

## Course Objective

Provide comprehensive understanding of the core principles of Machine Learning with hands-on training on applying machine learning to solve real-world problems.

A learner who completes this course should be able to define a machine learning problem, understand the solution path, and display the ability to carry out the end-to-end process of building a machine learning application.

© 2021 Sumudu Tennakoon

2

## Machine Learning Career Prospectus

- Data Scientist
- AI Scientist
- ML/AI Engineer
- Data Engineer
- Data Analyst
- AI/ML Developer
- IoT Developer
- Solutions Architect
- Freelancer
- ...

© 2021 Sumudu Tennakoon

3

## Schedule and Format

**Duration:** 60 hours

**Schedule:** 3-month program/12 weeks, two sessions per week.

**Format:** Live/Recorded Lectures, Demonstrations, Hands-on Exercises/Labs.

**Evaluation:** Quizzes (2), Project (1)

**Additional Practice:** Students must spend extra time on exercises and the capstone project.

© 2021 Sumudu Tennakoon

4

# Prerequisites

- Basics of computer programming, mathematics, and statistics.
- Basic knowledge in computer applications:
  - Spreadsheet
  - word processor
  - presentation authoring

© 2021 Sumudu Tennakoon

5

# Platform and Data for Hands-on Exercises and Project

**Programing Language:** Python 3 will be used as the primary programming language in teaching, practice examples and assignments.

**Python Libraries:** Scikit-learn, TensorFlow, Pandas, NumPy, Matplotlib, Seaborn, Flask.

**Applications/Tools:** Jupyter Notebook/Lab, IDE (Spyder/VS Code/Atom/PyCharm), Spreadsheet (MS Excel/LibreOffice Calc).

**Data:** Data for exercises, case studies, and projects will be obtained from open data repositories.

**Computing Environment:** Cloud platform (will be decided on class consensus and service availability) or locally installed Python distribution in student's PC.

© 2021 Sumudu Tennakoon

6

# Session Topics

#	Topic Name	Training Week #
1	Introduction to Machine Learning (ML), History, and Applications	1
2	Setting up a Computing Environment, Python and Required Libraries.	2
3	Knowledge Foundations to ML (Computing, Statistics, and Mathematics) *	2-3
4	Exploratory Data Analysis (EDA) and Feature Engineering *	4-5
5	Supervised and Unsupervised Learning (concepts)	6
6	Machine Learning Algorithms *	6-7
7	Explaining ML Models and Predictions (introduction) *	7
8	Deep Learning and Neural Networks (introduction) *	8
9	Design and Develop and Deploy ML Solutions *	9-10
10	Capstone Project *	11-12

© 2021 Sumudu Tennakoon

7

# Evaluations and Grading

## Completion Requirement:

- 80 % Attendance (at least 19 out of 24 sessions)
- Final Grade > 70 %

## Completion with Distinction:

- Final Grade > 90 %

Topic #	%
Quiz1 (Basic Concepts)	1-6 20
Quiz 2 (Advanced Concepts, Deep Learning and Application Building)	7-9 20
Deliverable and Project Report	10 50
Presentation (video narration)	10 10
	100

© 2021 Sumudu Tennakoon

8

# Introduction to Machine Learning

History and core concepts of ML to navigate the future lessons.

Applications of ML.

© 2021 Sumudu Tennakoon



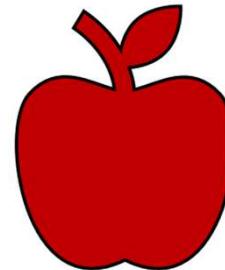
9

## Example: Identify Objects

What facts you consider to identify these objects?



Pineapple



Apple

© 2021 Sumudu Tennakoon

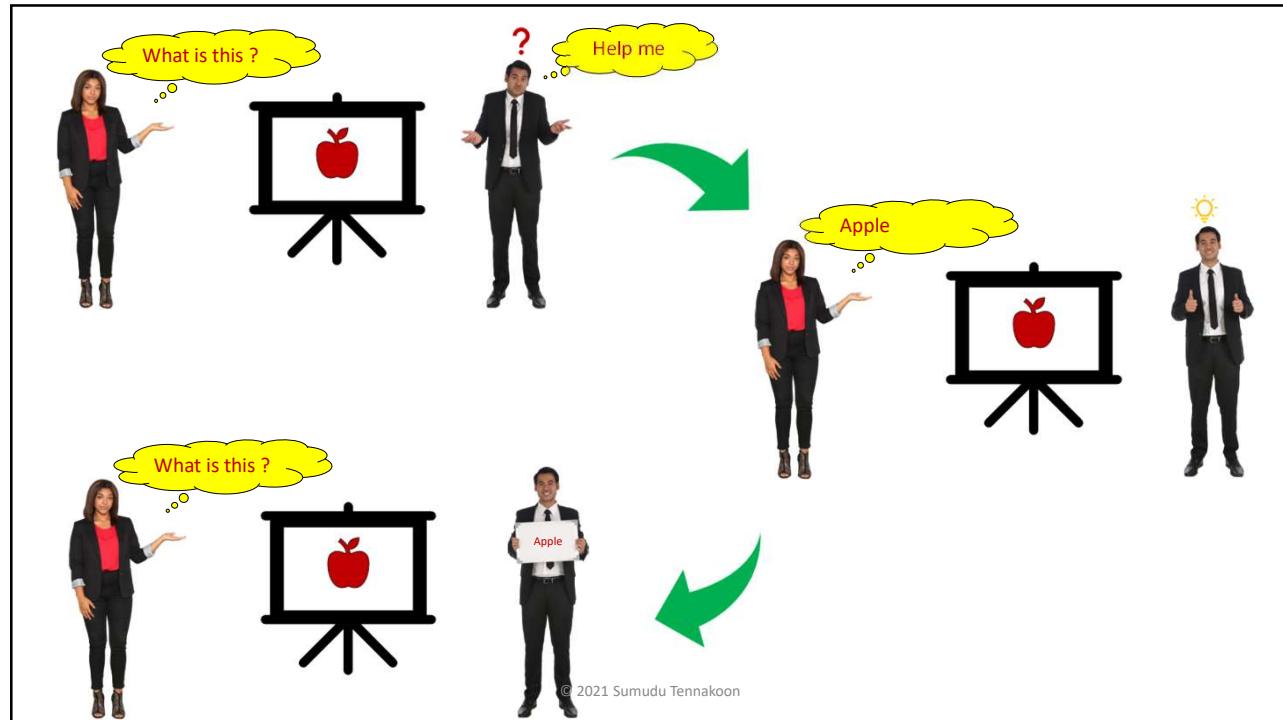
10

# How we Learn ?

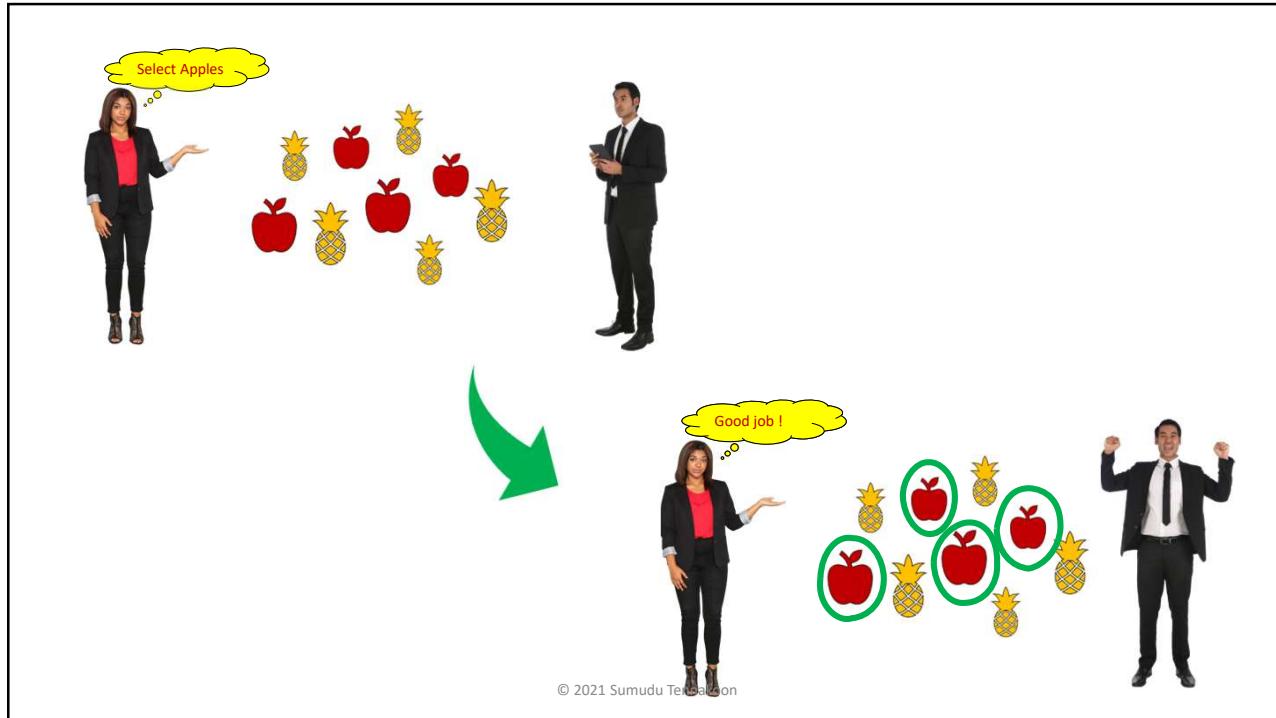
- Memorize Facts
  - Declarative Knowledge
  - Limited by memory and time to observe
- Infer (deduce new information from previously known facts)
  - Imperative Knowledge
  - Limited by accuracy of predictions and drifts (present is not behaving the same way as past)

© 2021 Sumudu Tennakoon

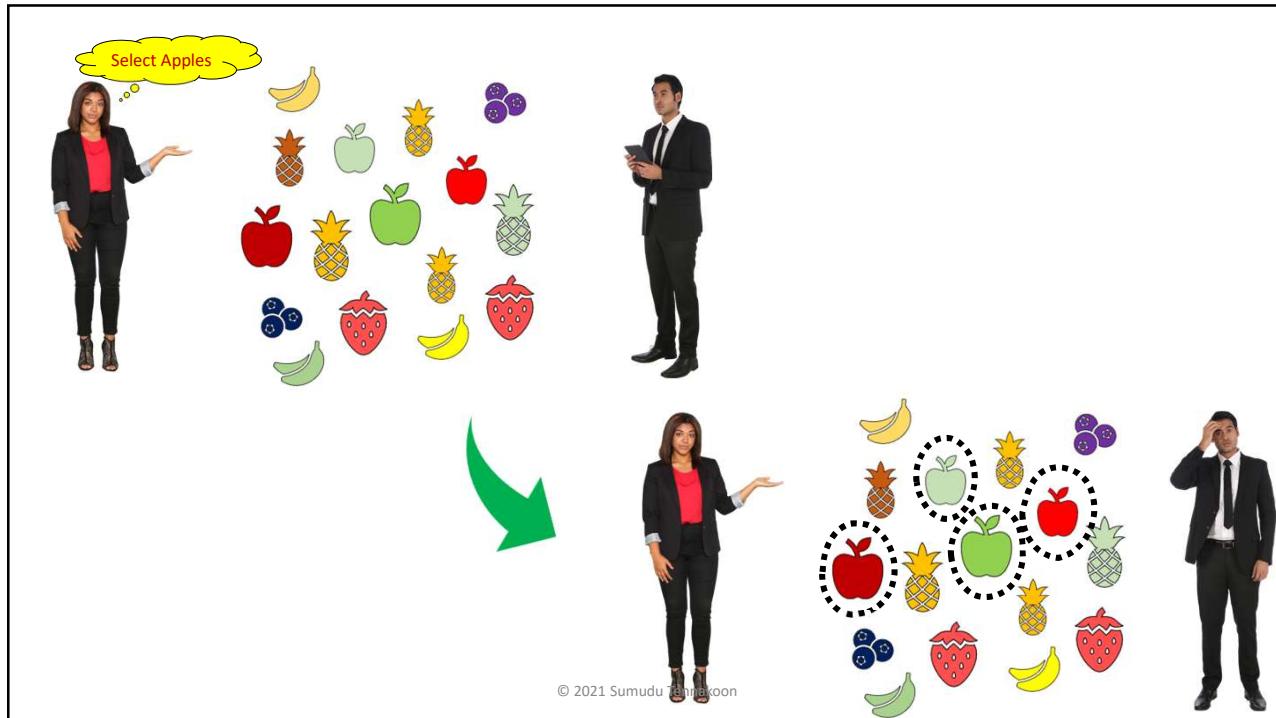
11



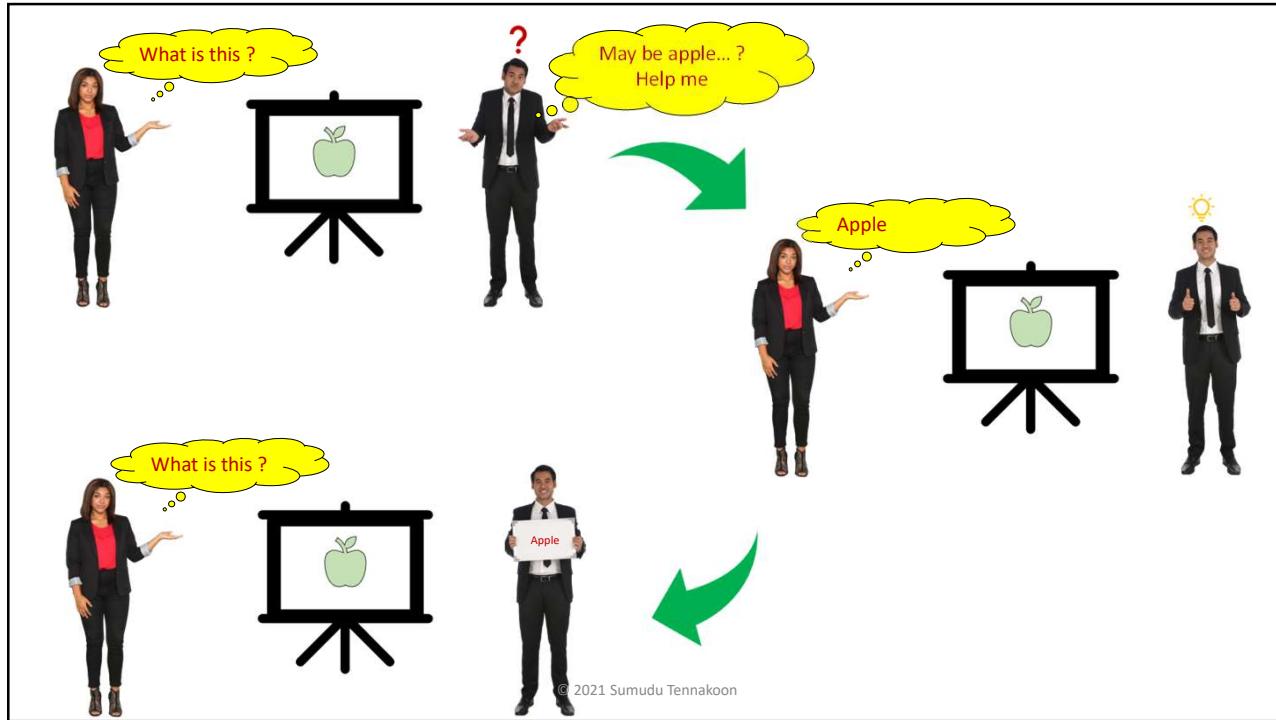
12



13



14



15

## Exercise

What are the Observations/Measurements can be used to make a determination.

Design a simple classifier logic.

Is it easy or difficult to convert this logic to a computer program (code)?

What are the considerations when converting this logic to a computer program (code)?

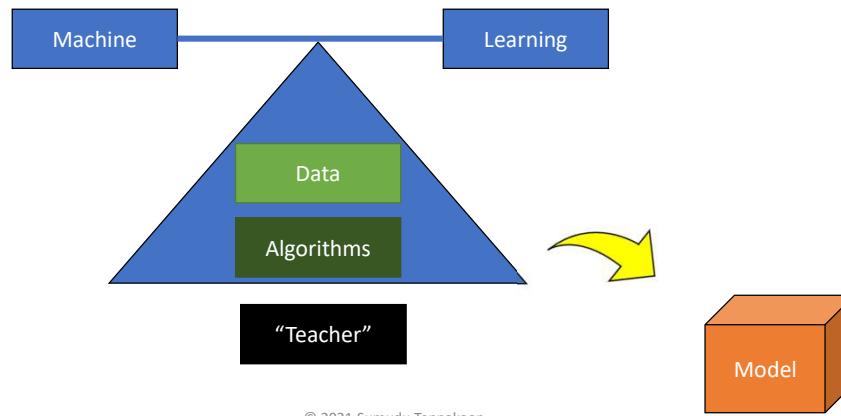
What are the points of failures in this approach?

© 2021 Sumudu Tennakoon

16

# What is Machine Learning ?

- Learn from Data



17

## Machine learning model

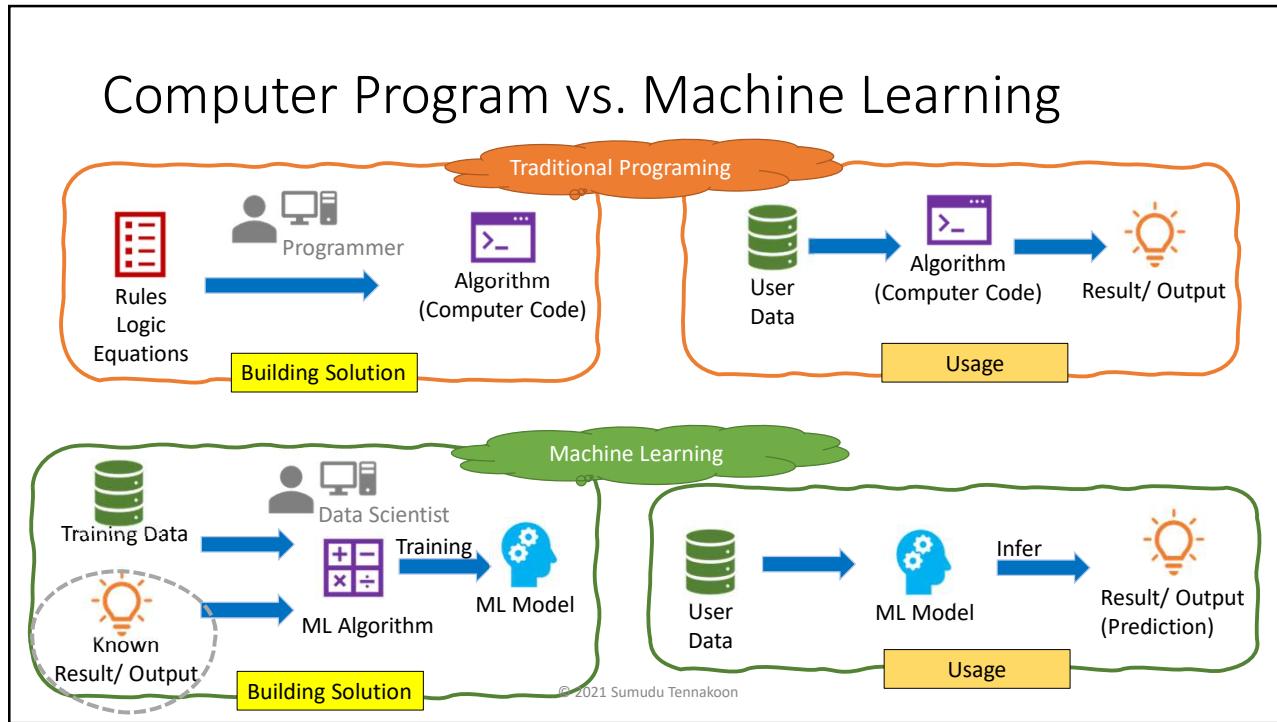


“Machine learning models are built on mathematical algorithms and are trained using data and human expertise to help us accurately predict outcomes based on input data such as images, text, or language.”

<https://developer.nvidia.com/ai-models>

© 2021 Sumudu Tennakoon

18



19

What is Machine Learning?

“The field of study that gives computers the ability to learn without being explicitly programmed.”  
~ Arthur Samuel (1959)

*Author of first self-learning program to learn how to play checkers by learning from experience (past movements and results)*

© 2021 Sumudu Tennakoon

20

## What is Machine Learning?

“ 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**.“

~ Tom Mitchell (1997)

Example: playing checkers.

- E = the experience of playing many games of checkers
- T = the task of playing checkers.
- P = the probability that the program will win the next game.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill.  
© 2021 Sumudu Tennakoon  
p. 2. [ISBN 978-0-07-042807-2](#).

21

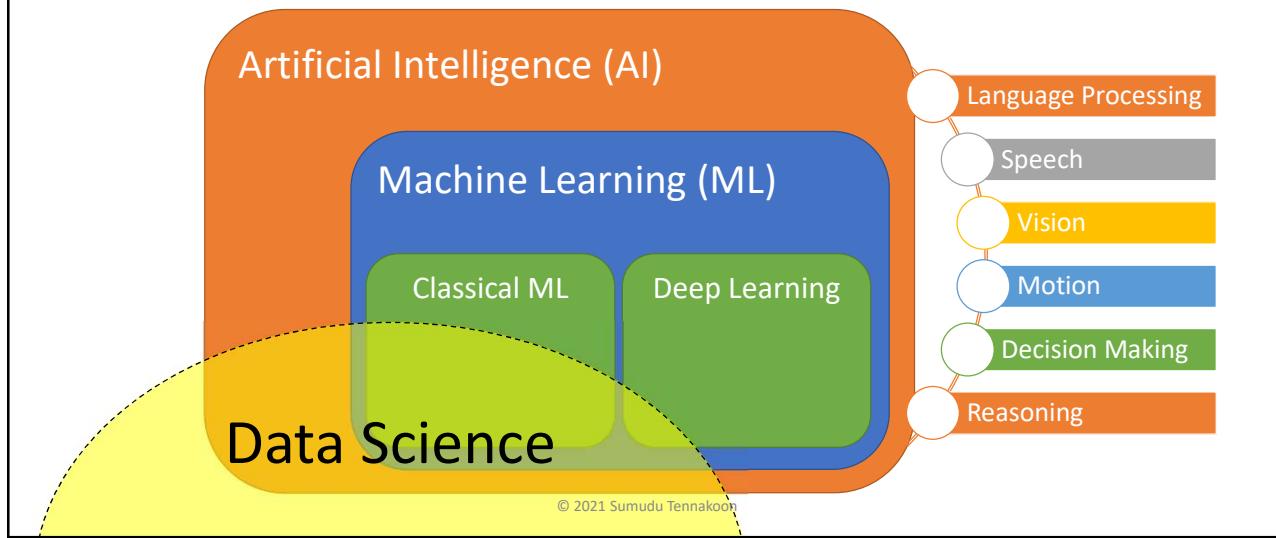
## AI and Machine Learning

- AI (Mimic Cognitive Functions of Human)
  - Computer Vision
  - Speech Recognition and Synthesis
  - Language Processing and Understanding
  - Motion
  - Decision Making
  - Prescribe or Predict
  - Reasoning
- Machine Learning (ML)
  - Machines learn on Data/Prior Knowledge
  - Statistical Modeling/Algorithms
  - Backbone of AI is Machine Learning
  - Algorithms to Find meanings of data
  - Find Relationships
  - Making Predictions
  - Problem-Solution Types
    - Classification
    - Regression
    - Clustering

© 2021 Sumudu Tennakoon

22

## Machine Learning/Deep Learning/AI



23

## Levels of AI

### Artificial General Intelligence (AGI) known as “Strong AI”

- AGI is the ability to solve *any* problem rather than finding a solution to a particular problem.
- Machine can understand or learn any intellectual task that a human being can.
- The machine can think and perform tasks on its own, just like a human being.
- In the Movies! We are not there yet.

### Weak Artificial Intelligence (Weak AI),

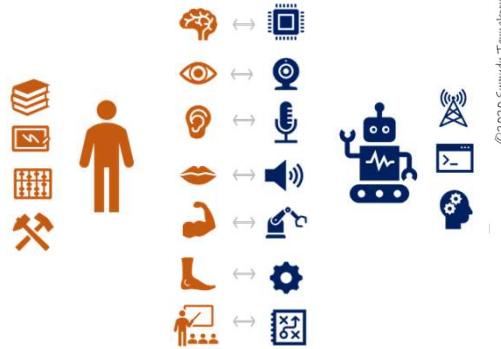
- Implements a limited part of human cognitive abilities.
- **Narrow AI** is a special case of Weak AI focused on a specific problem or task.
- Currently, existing AI systems are likely operating as a narrow AI.
- devices cannot follow these tasks independently but are made to look intelligent (simulate human behavior).

© 2021 Sumudu Tennakoon

24

# Building Blocks of an AI System

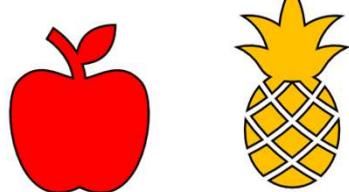
- Image recognition (computer vision)
- Signal processing (sound, sensor data feed, etc.)
- Speech Recognition (Speech to text/STT)
- Natural language processing (NLP)
- Visual Synthesis (Computer Graphics)
- Sound Synthesis (Text to Speech/TTS)
- Software/Algorithms
- Applications (Anomaly Detection, Classification, Prediction, Pattern Recognition)
- Memory (Storage, RAM, Cache)
- Processor (GPU, CPU, TPU)
- Connectivity (Wi-Fi, Satellite, 5G, ethernet, etc.)
- Hardware (Computer, Mechanical Components, etc.)



© 2021 Sumudu Tennakoon

25

## Exercise



What are Observations/Measurements  
can be used to make a determination.

Design a simple classifier logic.

Is it easy or difficult to converting this  
logic to a computer program (code)?

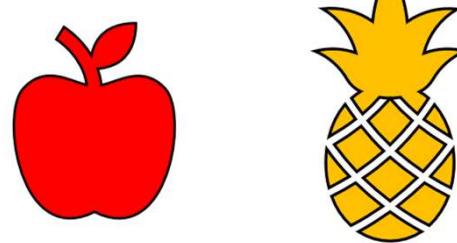
What are the considerations when  
converting this logic to a code?

What are the points of failure in this  
approach?

© 2021 Sumudu Tennakoon

26

# Input and Output ?



Input →  
(Features/Attributes)

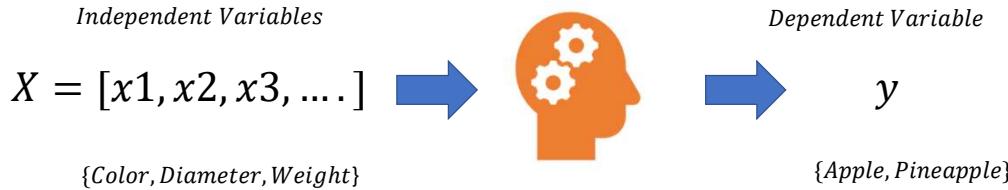
Color	red	yellow
Weight	50 g	200 g
Diameter	10 cm	20 cm

© 2021 Sumudu Tennakoon

Output →  
Apple (A)  
Pineapple (P)

27

$$y = f(X)$$



© 2021 Sumudu Tennakoon

28



29

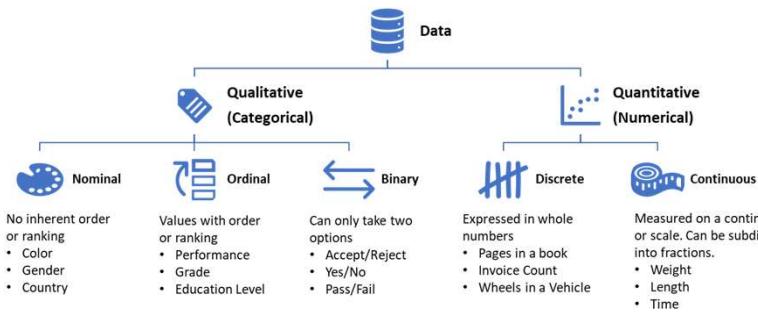
# Tabular Data

	Column 1	Column 2	Column 3	
	ID	Name	DOB	← Column Names
Row 1	10001	John Doe	1988-01-01	
Row 2	10002	Jane Doe	1990-12-31	Row (Record)
Row 3	...	...	...	

<https://pub.towardsai.net/data-science-for-everyone-getting-to-know-your-data-part-1-bb8b6d7782b1>

30

# Understanding Data



© 2021 Sumudu Tennakoon

© 2021 Sumudu Tennakoon

31

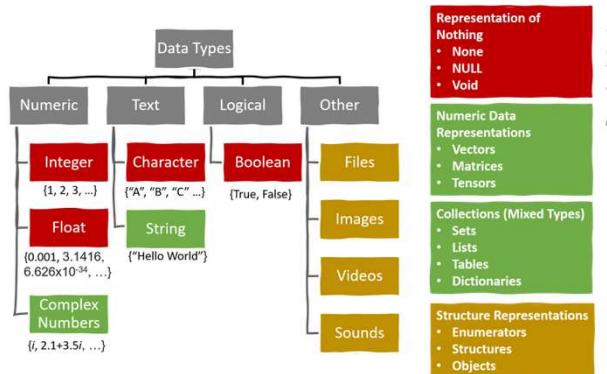


© 2021 Sumudu Tennakoon

© 2020 Sumudu Tennakoon

32

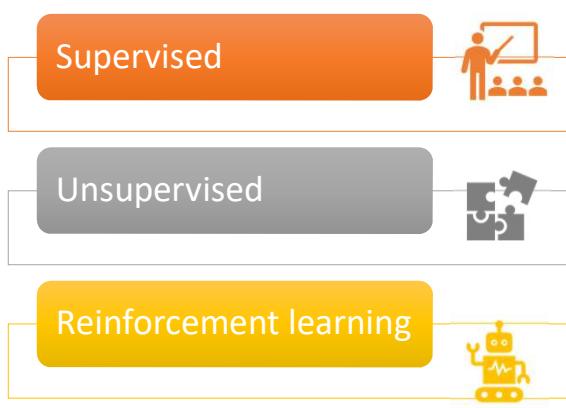
# Data Types and Representations



<https://pub.towardsai.net/data-science-for-everyone-getting-to-know-your-data-part-1-bb8b6d7782b1>

33

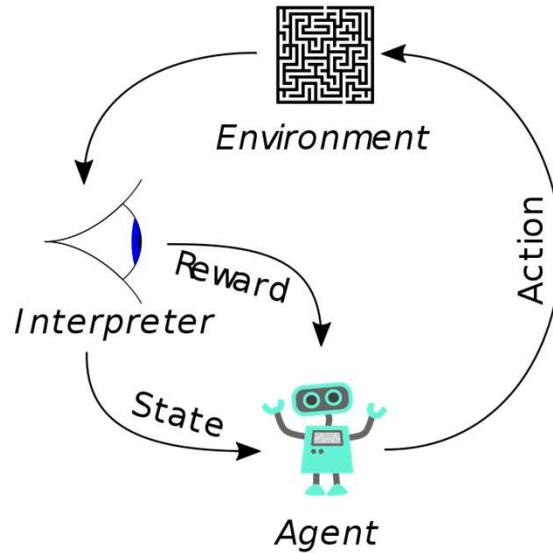
# Machine Learning Approaches



© 2021 Sumudu Tennakoon



34



© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Reinforcement\\_learning](https://en.wikipedia.org/wiki/Reinforcement_learning)

35

## Types of ML Algorithms



### Regression

Linear

Polynomial



### Classification

Tree Classifiers

Logistic Regression

Support Vector Machines (SVM)



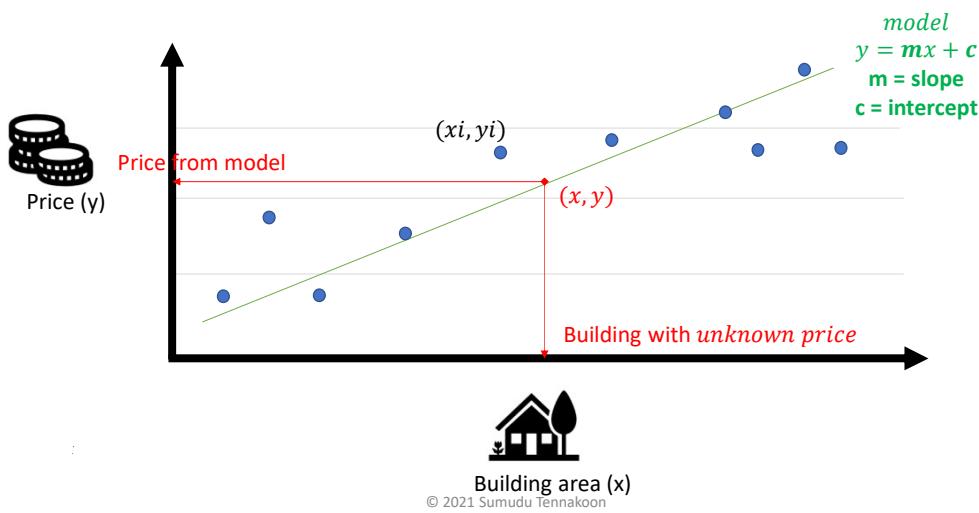
### Clustering

K-Means

© 2021 Sumudu Tennakoon

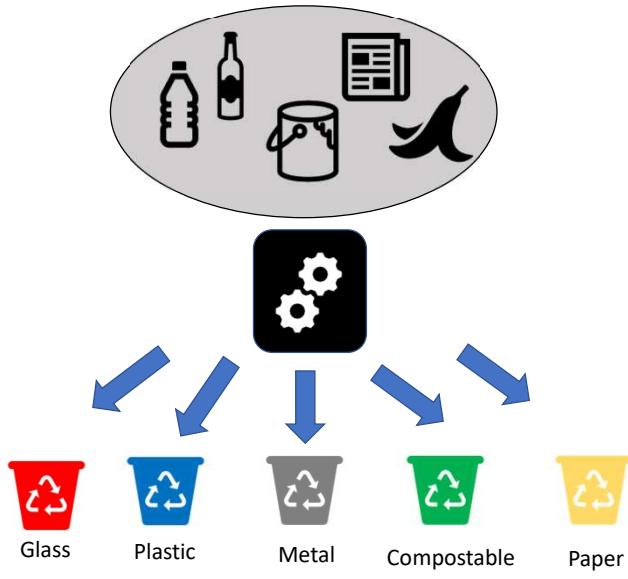
36

## Regression



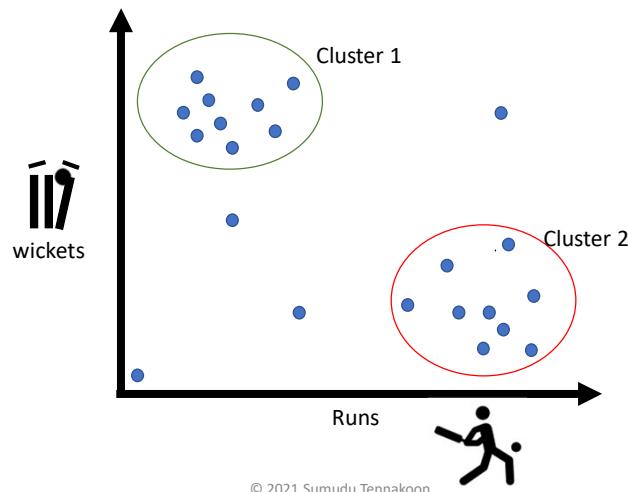
37

## Classification



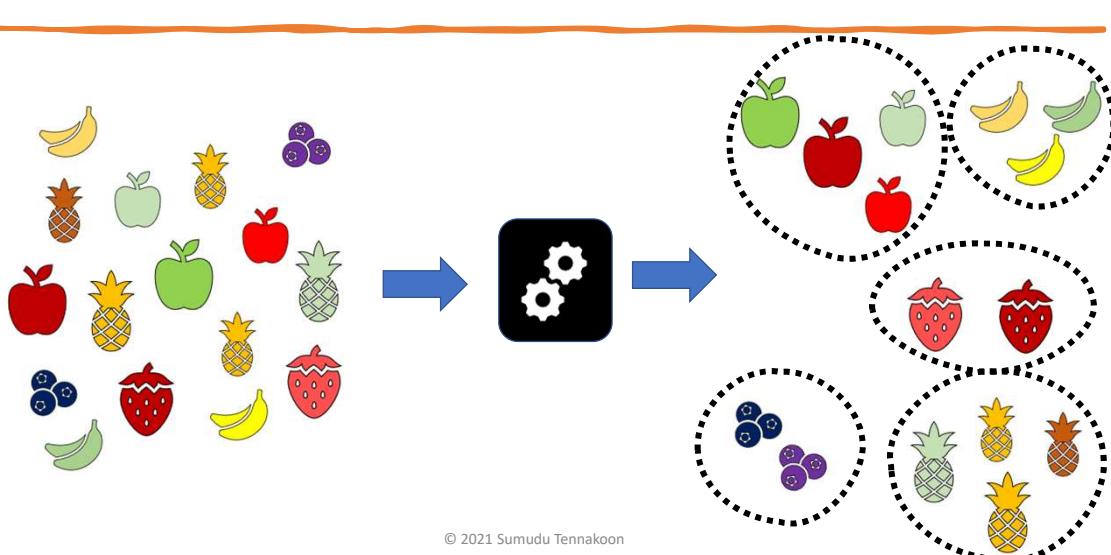
38

## Clustering



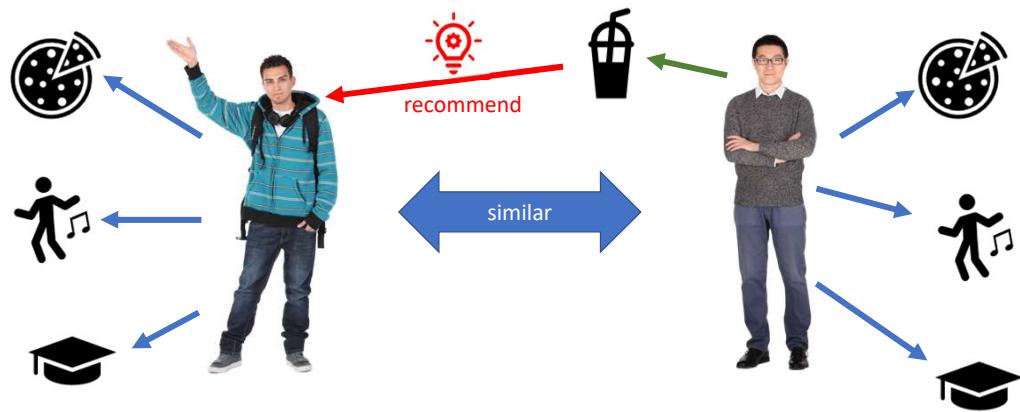
39

## Clustering



40

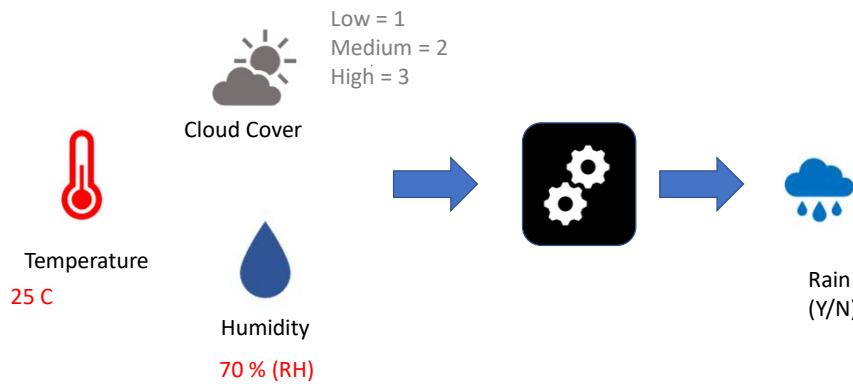
## Recommender Systems



© 2021 Sumudu Tennakoon

41

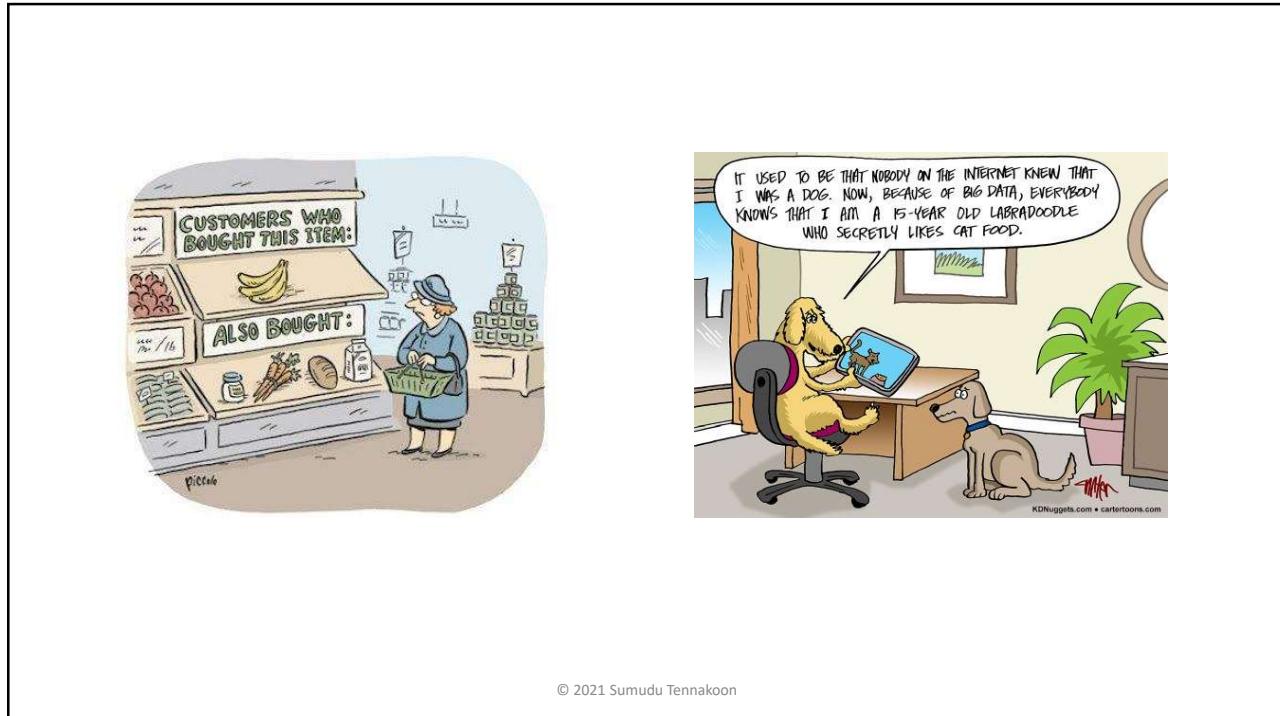
## Prediction



$$y = f(X) ?$$

© 2021 Sumudu Tennakoon

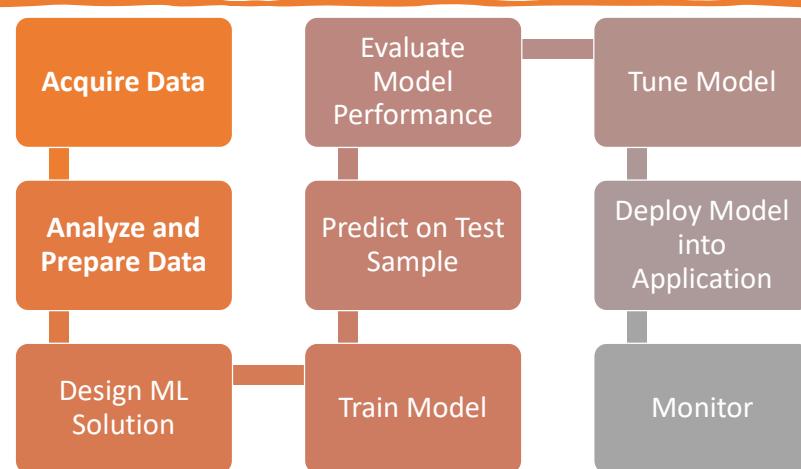
42



© 2021 Sumudu Tennakoon

43

## Steps of Building ML Model



© 2021 Sumudu Tennakoon

44

# Why we need Machine Learning?

Simulate human intelligence

Automation

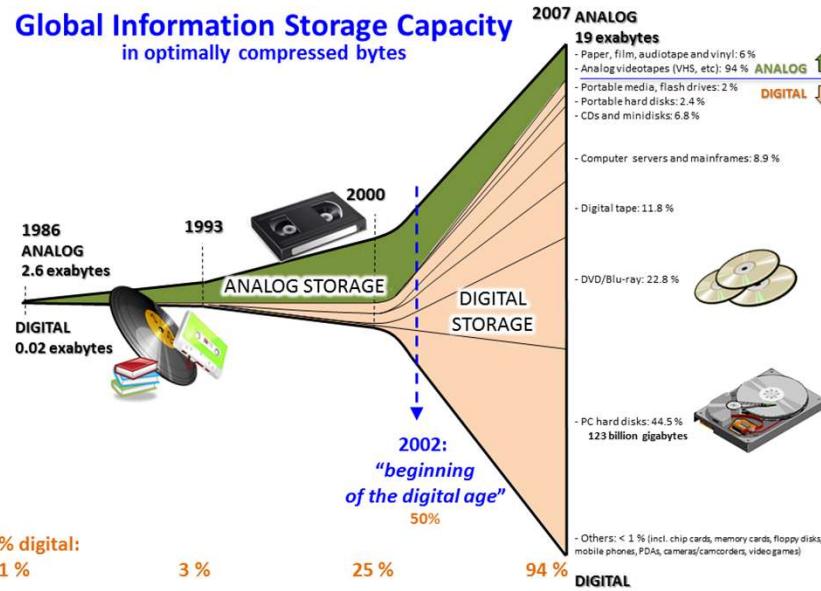
Help humans with informed decision making

Solve multidimensional problems

Predict future outcome based on historical observations

© 2021 Sumudu Tennakoon

45



Source: Hilbert, M., & López, P. (2011). The World's Technological Capacity to Store, Communicate, and Compute Information. *Science*, 332(6025), 60–65. <http://www.martinhilbert.net/WorldInfoCapacity.html>

Scholz, R.. "Sustainable Digital Environments: What Major Challenges Is Humankind Facing?" *Sustainability* 8 (2016): 1-31.

46

# Machine Learning Applications

- Spam Email Filtering
- Approve or Reject Loan Application
- Predicting Stock Price
- Credit Card Fraud Detection
- Recommending Items to Purchase (Advertising)

© 2021 Sumudu Tennakoon

47

# Application Areas

- Finance
- Marketing
- Information Technology
- Cyber Security
- Agriculture
- Government
- Automobile
- Manufacturing
- Retail
- Entertainment
- ...
- Everywhere!

© 2021 Sumudu Tennakoon

48

## Why should everyone get familiar with ML?



Applications of machine learning are all around us.



ML is used in many industries and domains.



Job opportunities.



It is fun to learn and helps train your brain.



© 2021 Sumudu Tennakoon

49

## Machine Learning Career Prospectus

- Data Scientist
- AI Scientist
- ML/AI Engineer
- Data Engineer
- Data Analyst
- AI/ML Developer
- IoT Developer
- Solutions Architect
- ...

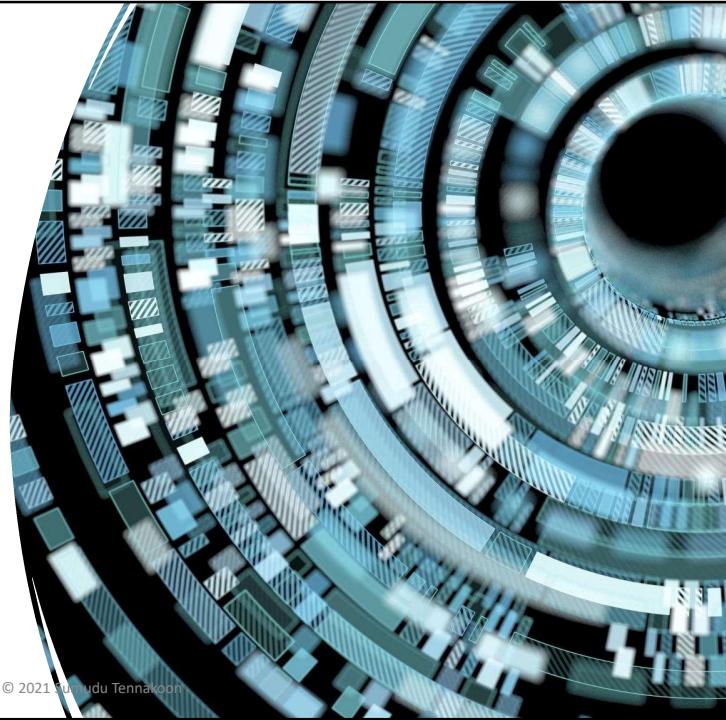
© 2021 Sumudu Tennakoon

50

# Setting up Computing Environment

Cloud Computing Platform  
(Google Colab)

Python (Install, Libraries)



© 2021 Sumudu Tennakoon

51

## Install Python in Local Computer

- Python:  
<https://www.python.org/downloads/>
- Anaconda Python:  
<https://www.anaconda.com/products/individual>
- Python: Libraries:  
<https://www.anaconda.com/open-source>

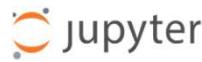


© 2021 Sumudu Tennakoon

52

## Python Libraries

- Data Handling
  - Pandas
  - Dask (distributed)
- Machine Learning
  - Scikit-learn
  - TensorFlow
  - PyTorch
- Visualizing
  - Matplotlib
  - Seaborn
- Numerical and Scientific Computing
  - SciPy
  - NumPy
- Machine Learning Model Interpretation
  - LIME
  - SHAP
- Web Services/API
  - Flask
  - Django



© 2021 Sumudu Tennakoon

53

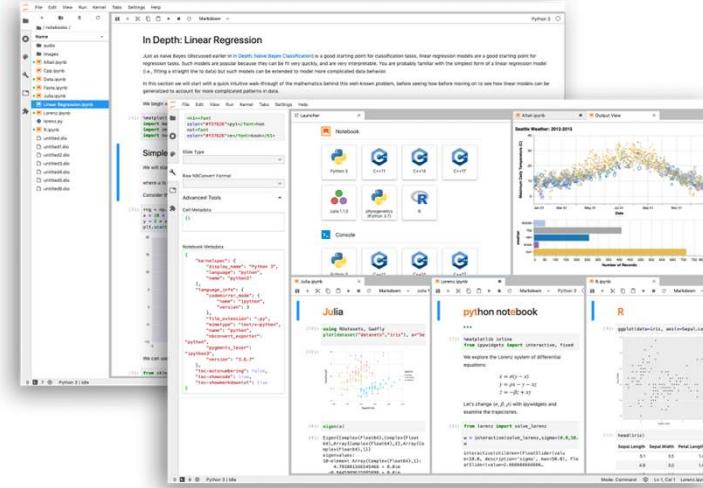
## Install Python Libraries

- Using pip package manager for Python
  - <https://packaging.python.org/tutorials/installing-packages>
  - pip install some-package-name
  - NumPy: pip install numpy
  - Pandas: pip install pandas
  - Scikit-Learn: pip install scikit-learn
  - Matplotlib: pip install matplotlib
- Using Conda package manager
  - <https://docs.conda.io/projects/conda/en/latest/commands/install.html>
  - conda install -c conda-forge some-package-name
  - NumPy: conda install -c conda-forge numpy
  - Pandas: conda install -c conda-forge pandas
  - Scikit-Learn: conda install -c conda-forge scikit-learn
  - Matplotlib: conda install -c conda-forge matplotlib

© 2021 Sumudu Tennakoon

54

# Jupyter Notebook/Lab



© 2021 Sumudu Tennakoon

55

# Google Colab

<https://colab.research.google.com>

© 2021 Sumudu Tennakoon

56

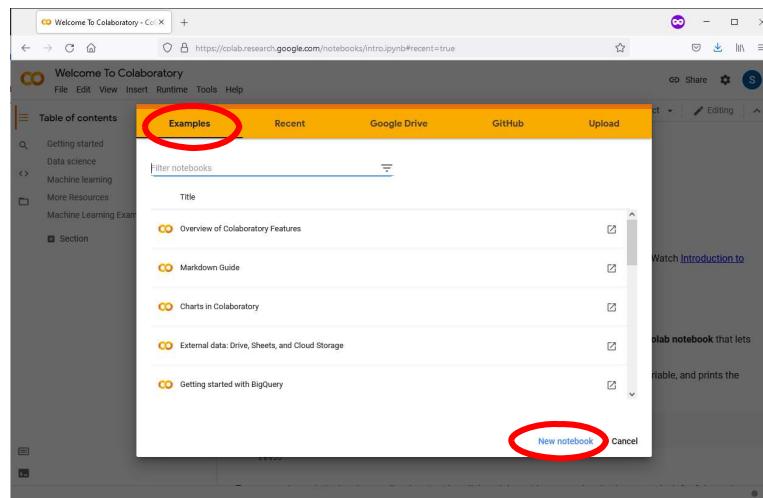
## Getting Started

- Creating New Notebook
- Opening Notebook from GitHub
- Opening Notebook from file

© 2021 Sumudu Tennakoon

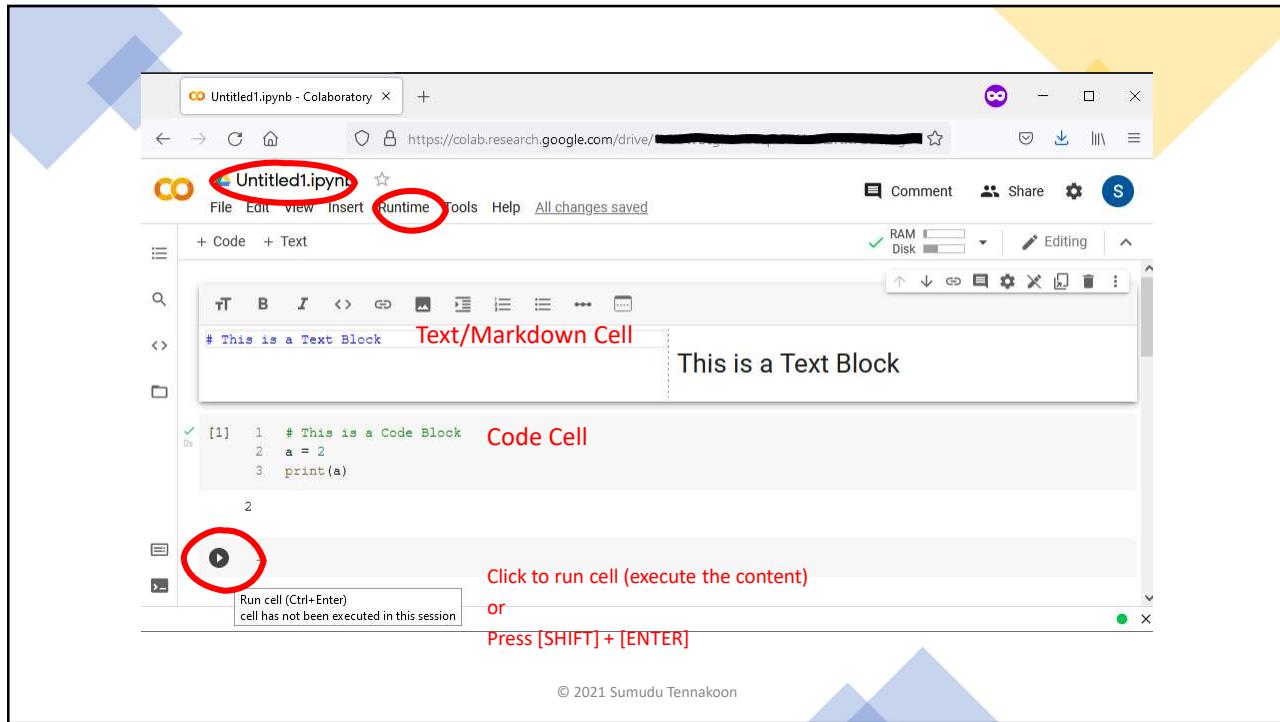
57

## Creating New Notebook



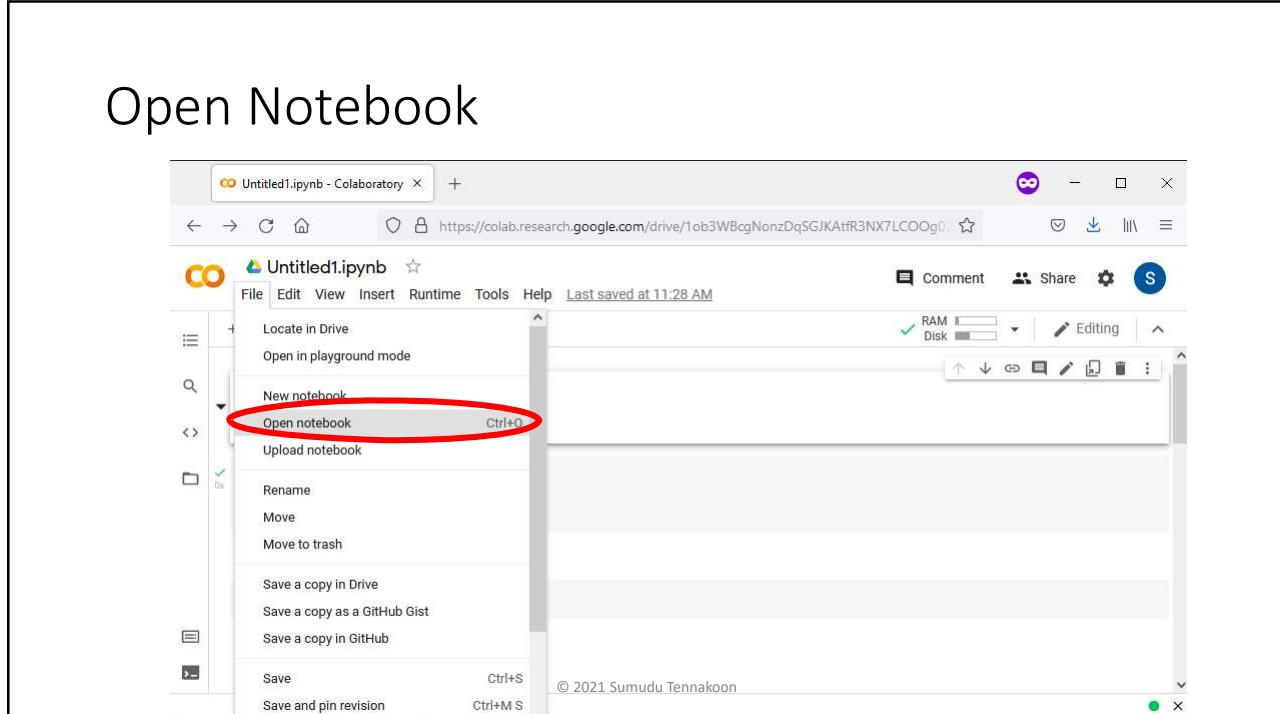
© 2021 Sumudu Tennakoon

58



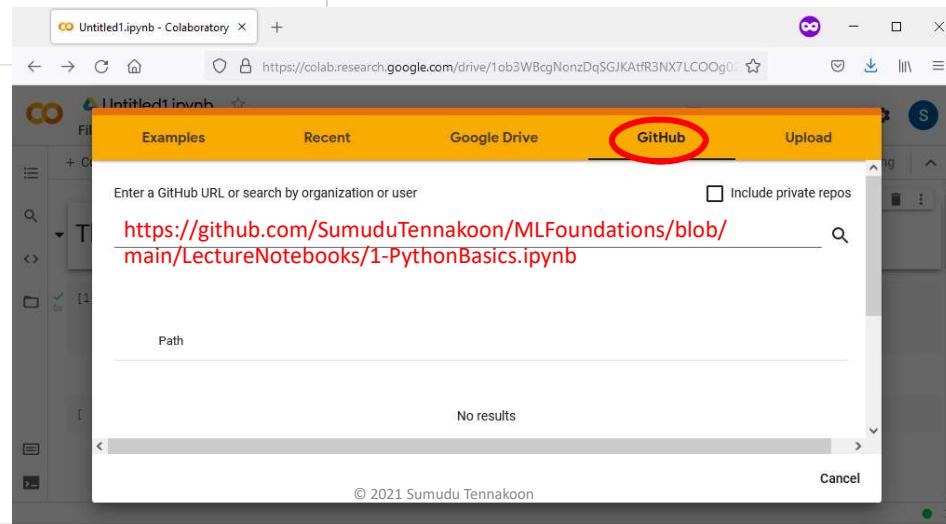
59

## Open Notebook



60

## Open Notebook from GitHub



61

https://github.com/SumuduTennakoon/MLFoundations/blob/main/LectureNotebooks/1-PythonBasics.ipynb

Open notebook in new tab

62

The screenshot shows the Google Colab interface. The top navigation bar includes back, forward, home, and search icons, along with a URL bar showing <https://colab.research.google.com/github/SumuduTennakoon/MLFoundations/blob/main/Untitled5.ipynb>. The main menu has options: File, Edit, View, Insert, Runtime, Tools, Help, Share, and Settings. A sidebar on the left is titled 'Table of contents' and lists sections: Machine Learning Foundations (which is expanded), Python Basics, Variables, Data Types, Lists, Tuples, Sets, Dictionary, Conditions, Functions, and Exercise 1. The main content area displays the title 'Machine Learning Foundations' and author 'Sumudu Tennakoon, PhD'. Below this, a section titled 'Python Basics' is shown with the text: 'In this notebook we will explore some basic features on Python programming language for those who have a prior programming experience.' It also includes a note: 'To learn more about Python, refer to the following websites' with a bullet point: 'Python : [www.python.org](http://www.python.org)'. At the bottom of the content area, there's a footer with the text '© 2021 Sumudu Tennakoon'.

63

The screenshot shows the Google Colab interface with a modal dialog box for uploading a notebook. The dialog has tabs: Examples, Recent, Google Drive, GitHub, and Upload, with 'Upload' being the active tab and circled in red. Below the tabs is a large input field with a dashed border, containing the text 'Browse... No file selected.' at the bottom. The background of the dialog is dark grey, while the tabs and input field are highlighted in yellow. The bottom right corner of the dialog has a 'Cancel' button. The URL in the browser bar is <https://colab.research.google.com/drive/1ob3WBcgNonzDqSGJKAtfR3NX7LCOOg0>.

64

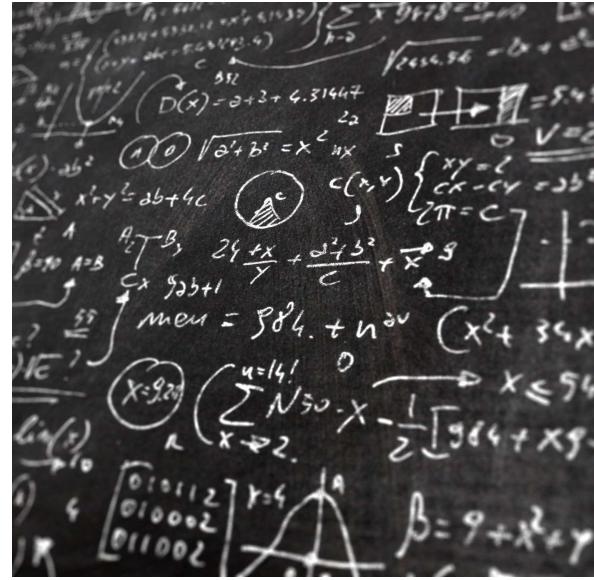
The screenshot shows a Google Colab notebook titled "1-PythonBasics.ipynb". The left sidebar displays a "Table of contents" with sections like Machine Learning Foundations, Python Basics, Variables, Data Types, Lists, Tuples, Sets, Dictionary, Conditions, Functions, and Exercise 1. The main content area is titled "Machine Learning Foundations" by Sumudu Tennakoon, PhD. It contains a section titled "Python Basics" with a description of basic features for programmers with prior experience. It also lists websites for learning more about Python, including [www.python.org](http://www.python.org). The footer of the notebook page includes the copyright notice "© 2021 Sumudu Tennakoon".

65

The slide has a title "GitHub Link to Lecture Notebooks". Below the title are two circular icons: one orange folder icon and one grey book icon with stars. The left side of the slide contains a "Folder:" label and a link to a GitHub repository: <https://github.com/sumudutennakoon/mlfoundations/tree/main/lecturenotebooks>. The right side contains a "Python basics notebook:" label and a link to a specific notebook: <https://github.com/sumudutennakoon/mlfoundations/blob/main/lecturenotebooks/1-pythonbasics.ipynb>. The footer of the slide includes the copyright notice "© 2021 Sumudu Tennakoon".

66

# Mathematics for Machine Learning



© 2021 Sumudu Tennakoon

67

# Data Representations

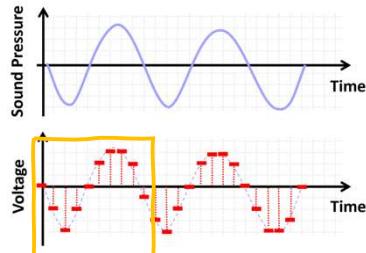
- In computing everything must convert into numbers !
- Numeric Data Structures:
  - Scalars: [3.14](#)
  - Vectors (1D): [\[1,2,3\]](#)
  - Matrices (2D): [\[ \[1,2\], \[3,4\] \]](#)
  - Tensors (3D+): [\[ \[ \[1,2\], \[3,4\] \], \[ \[5,6\], \[7,8\] \] \]](#)

© 2021 Sumudu Tennakoon

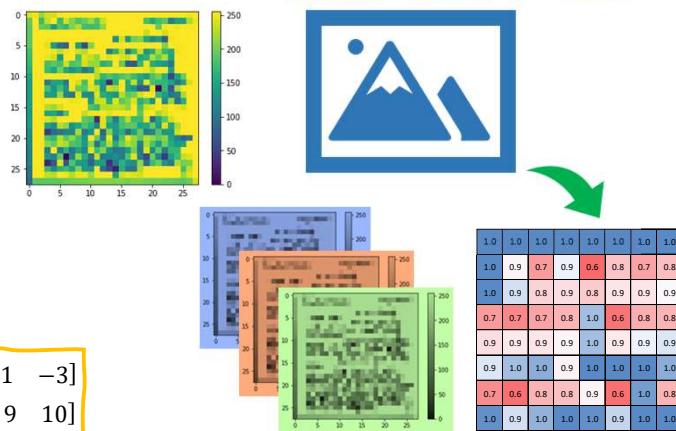


68

## Why we need Vectors, Matrices and Tensors in Machine Learning?



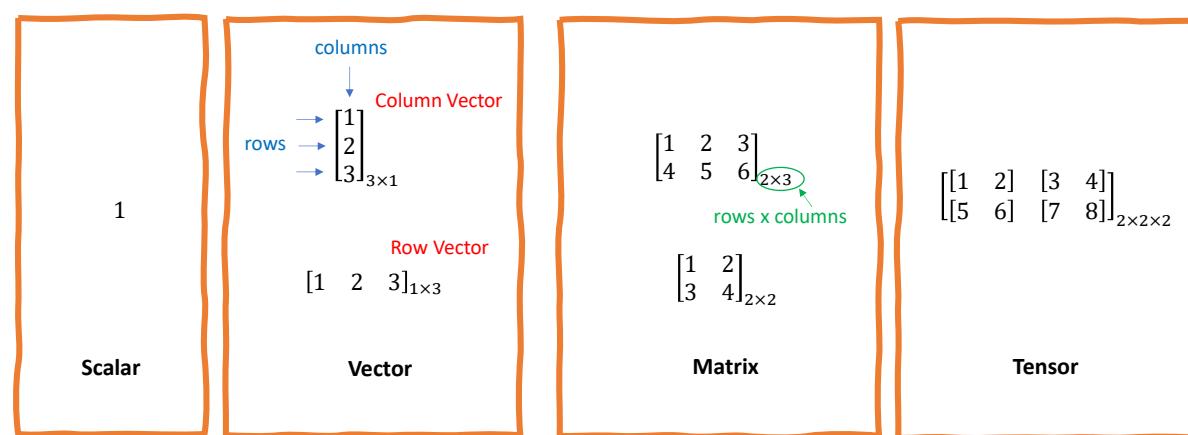
`signal = [0 -2 -4 -2 0 2 3 3 2 -1 -3]  
time = [0 1 2 3 4 5 6 7 8 9 10]`



<https://pub.towardsai.net/data-science-for-everyone-getting-to-know-your-data-part-1-bb8b6d7782b1>

69

## Vectors and Matrices



- Temperature
- Mass
- Speed

- Distance  $[x, y, z]$
- Velocity  $[V_x, V_y, V_z]$

© 2021 Sumudu Tennakoon

[What's a Tensor? - YouTube](https://www.youtube.com/watch?v=JyfjwvXWzqU)

70



# Matrix Algebra

- Transpose
- Sum
- Diagonal
- Determinant
- Adjugate
- Inverse

© 2021 Sumudu Tennakoon

71

## Matrix Notation

- Matrix Notation

$$\bullet A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \ddots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ddots & a_{mn} \end{bmatrix}_{m \times n}$$

- Representation of elements in python index numbers

$$\bullet A = \begin{bmatrix} A[0][0] & A[0][1] & \dots & A[0][n-1] \\ A[1][0] & A[1][1] & \ddots & A[1][n-1] \\ \vdots & \vdots & \ddots & \vdots \\ A[m-1][0] & A[m-1][1] & \dots & A[m-1][n-1] \end{bmatrix}$$

© 2021 Sumudu Tennakoon

72

# Transpose, Sum, Diagonal

•  $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = [a_{ij}]$

•  $A^T = \begin{bmatrix} a_{11} & a_{21} & a_{31} & a_{41} \\ a_{12} & a_{22} & a_{32} & a_{42} \\ a_{13} & a_{23} & a_{33} & a_{43} \\ a_{14} & a_{24} & a_{34} & a_{44} \end{bmatrix} = [a_{ji}]$

•  $\text{Sum}(A) = \sum_{ij} a_{ij}$

•  $\text{Diagonal}(A) = [a_{11} \ a_{22} \ a_{33} \ a_{44}]$

•  $\text{Trace}(A) = a_{11} + a_{22} + a_{33} + a_{44}$

E.g.,

- $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
- $A^T = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$
- $\text{Sum}(A) = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$
- $\text{Diagonal}(A) = [1 \ 5 \ 9]$
- $\text{Trace}(A) = 1 + 5 + 9 = 15$

© 2021 Sumudu Tennakoon

73

# Determinant

**2 × 2 Matrix**

$$\det \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

$$\det \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = 1 \times 4 - 2 \times 3 = -2$$

**3 × 3 Matrix**

$$\det \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} = a \begin{vmatrix} e & f \\ h & i \end{vmatrix} - b \begin{vmatrix} d & f \\ g & i \end{vmatrix} + c \begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

$$\det \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = 1 \begin{vmatrix} 5 & 6 \\ 8 & 9 \end{vmatrix} - 2 \begin{vmatrix} 4 & 6 \\ 7 & 9 \end{vmatrix} + 3 \begin{vmatrix} 4 & 5 \\ 7 & 8 \end{vmatrix} = 0$$

© 2021 Sumudu Tennakoon

74

## Inverse Matrix

- $A^{-1} = \frac{1}{|A|} adj(A)$
- $A^{-1}A = AA^{-1} = I$

- $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$
- $|A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$
- $adj(A) = \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$
- $A^{-1} = \frac{1}{ad-bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

[https://en.wikipedia.org/wiki/Invertible\\_matrix](https://en.wikipedia.org/wiki/Invertible_matrix)

© 2021 Sumudu Tennakoon

[https://en.wikipedia.org/wiki/Adjugate\\_matrix](https://en.wikipedia.org/wiki/Adjugate_matrix)

75

## Properties of Matrix

$$\begin{aligned} A + B &= B + A \\ A + 0 &= A \\ AB &\neq BA \end{aligned}$$

$$\begin{aligned} A(BC) &= (AB)C \\ A(B+C) &= AB+AC \\ (A+B)C &= AC+BC \end{aligned}$$

$$\begin{aligned} \alpha(A+B) &= \alpha A + \alpha B \\ (\alpha + \beta)A &= \alpha A + \beta A \end{aligned}$$

$$\begin{aligned} 0.A &= 0 \\ A.0 &= 0 \end{aligned}$$

$$\begin{aligned} (A^T)^T &= A \\ (A+B)^T &= A^T + B^T \\ (AB)^T &= B^T A^T \end{aligned}$$

$$\begin{aligned} (A^{-1})^{-1} &= A \\ A^{-1}A &= AA^{-1} = I \end{aligned}$$

$$\begin{aligned} (A^T)^{-1} &= (A^{-1})^T \\ (AB)^{-1} &= B^{-1}A^{-1} \end{aligned}$$

$$\begin{aligned} |A^{-1}| &= \frac{1}{|A|} \\ |AB| &= |A||B| \end{aligned}$$

© 2021 Sumudu Tennakoon

76

## Special Vectors/Matrices

- Square Matrix:  $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}_{2 \times 2}$

- Symmetric Matrix:  $S = \begin{bmatrix} 1 & 4 & 5 \\ 4 & 2 & 6 \\ 5 & 6 & 3 \end{bmatrix}$

- Zero Matrix:  $0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

- Unit Vector:  $\hat{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

- Diagonal Matrix:  $D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$

- Identity Matrix:  $I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

- Scalar Diagonal Matrix:  $D = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \lambda I$

- Upper Triangular matrix:  $U = \begin{bmatrix} 1 & 4 & 5 \\ 0 & 2 & 6 \\ 0 & 0 & 3 \end{bmatrix}$

- Lower Triangular Matrix:  $L = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 2 & 0 \\ 5 & 6 & 3 \end{bmatrix}$

© 2021 Sumudu Tennakoon

77

## Matrix Operations and Applications

- 
- Addition/Subtraction
  - Scalar Multiplication
  - Matrix Multiplication (Dot Product)
  - Matrix-Vector Multiplication
  - Row operations
  - Solving Linear Equations
  - Linear transformations
  - Decomposition

© 2021 Sumudu Tennakoon

78

# Matrix Multiplication

$$\bullet \ A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\bullet A \cdot B = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} p & q \\ r & s \end{bmatrix}$$

$$\bullet \ B = \begin{bmatrix} p & q \\ r & s \end{bmatrix}$$

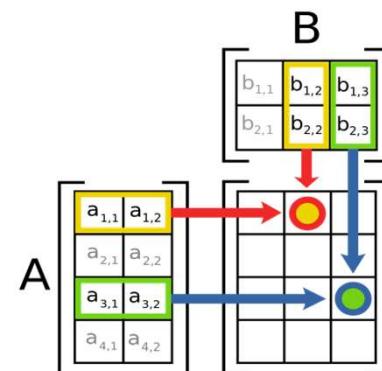
$$\bullet A \cdot B = \begin{bmatrix} ap + br & aq + bs \\ cp + dr & cq + ds \end{bmatrix}$$

© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Matrix\\_multiplication](https://en.wikipedia.org/wiki/Matrix_multiplication)

79

# Matrix Multiplication (Dot Product)

$$\begin{array}{ccc}
 \text{A} & \cdot & \text{B} \\
 5 \times 3 & & 3 \times 4
 \end{array}
 =
 \begin{array}{c}
 \text{C} \\
 5 \times 4
 \end{array}$$



© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Matrix\\_multiplication](https://en.wikipedia.org/wiki/Matrix_multiplication)

80

Handwritten notes on a blackboard:

$$\frac{2x^2 + 3x^2}{2} = x^2$$

$$c(x, y) \left\{ \begin{array}{l} xy = c \\ cx - cy = z \end{array} \right.$$

$$2x^2 + 3x^2 + x^2 = 2x^2 + 3x^2 + x^2$$

$$= 384. + n \text{ av } (x^2 + 3)$$

$$x = 2.$$

$$r = 4 \quad b = 9 + x^2$$

## Solving Linear Equations

**Problem:**

$$\begin{aligned} 3x + 2y &= 12 \rightarrow (1) \\ 4x + 5y &= 23 \rightarrow (2) \end{aligned}$$

$$\boxed{x = ?}$$

$$\boxed{y = ?}$$

**Solution Approach:**

$$\begin{aligned} x &= \frac{12 - 2y}{3} = 4 - \frac{2}{3}y \\ 4\left(4 - \frac{2}{3}y\right) + 5y &= 23 \\ 16 - \frac{8}{3}y + 5y &= 16 + \frac{15 - 8}{3}y = 16 + \frac{7}{3}y = 23 \\ y &= \frac{3(23 - 16)}{7} = \frac{3(7)}{7} = 3 \\ x &= 4 - \frac{2}{3}(3) = 4 - 2 = 2 \end{aligned}$$

$$\boxed{x = 2}$$

$$\boxed{y = 3}$$

© 2021 Sumudu Tennakoon

81

## Formulating the Problem in Matrix Form

$$\begin{cases} x_1 = x \\ x_2 = y \end{cases}$$

$$3x_1 + 2x_2 = 12 \rightarrow (1)$$

$$4x_1 + 5x_2 = 23 \rightarrow (2)$$

Solve using Matrices:

$$\begin{bmatrix} 3 \\ 4 \end{bmatrix} x_1 + \begin{bmatrix} 2 \\ 5 \end{bmatrix} x_2 = \begin{bmatrix} 12 \\ 23 \end{bmatrix}$$

$$\begin{bmatrix} 3 & 2 \\ 4 & 5 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 12 \\ 23 \end{bmatrix}$$

$$Ax = b$$

$$\begin{bmatrix} a_{11} & a_{11} \\ a_{11} & a_{11} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 2 \\ 4 & 5 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{11} \\ a_{11} & a_{11} \end{bmatrix}$$

© 2021 Sumudu Tennakoon

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 12 \\ 23 \end{bmatrix}$$

82

## Obtaining the Solution in Matrix Form

$$\begin{aligned} Ax &= b \\ A^{-1}Ax &= A^{-1}b \\ Ix &= A^{-1}b \\ x &= A^{-1}b \end{aligned}$$

$$\begin{aligned} x &= A^{-1}b \\ x &= \left( \frac{1}{|A|} \times adj(A) \right) b \end{aligned}$$

Example  $A = \begin{bmatrix} 3 & 2 \\ 4 & 5 \end{bmatrix}$

$$b = \begin{bmatrix} 12 \\ 23 \end{bmatrix}$$

$$A^{-1} = \frac{1}{|A|} adj(A)$$

$$|A| = 3 \times 5 - 4 \times 2 = 7$$

$$A^{-1} = \frac{1}{7} \begin{bmatrix} 5 & -2 \\ -4 & 3 \end{bmatrix}$$

$$x = \frac{1}{7} \begin{bmatrix} 5 & -2 \\ -4 & 3 \end{bmatrix} \cdot \begin{bmatrix} 12 \\ 23 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

© 2021 Sumudu Tennakoon

83

## Solving Linear Equations (General Form)

$$Ax = b$$

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}_{m \times n} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{n \times 1} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}_{m \times 1}$$

$m = \text{number of linearly independent equations}$

$n = \text{number of unknowns } (x_i)$

$m = n \rightarrow \text{unique solution}$

$m < n \rightarrow \text{infinitely many solutions}$

$$x = A^{-1}b$$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{n \times 1} = \underbrace{\begin{bmatrix} a'_{11} & a'_{12} & \dots & a'_{1m} \\ a'_{21} & a'_{22} & \dots & a'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a'_{n1} & a'_{n2} & \dots & a'_{nm} \end{bmatrix}_{n \times m}}_{A^{-1}} \cdot \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}_{m \times 1}$$

© 2021 Sumudu Tennakoon

84

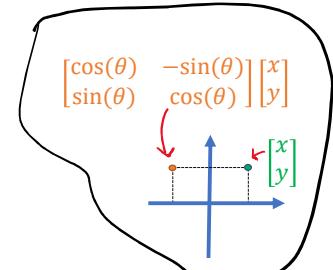


# Linear Transformations

- Stretching =  $\begin{bmatrix} k & 0 \\ 0 & 1 \end{bmatrix}$
- Squeezing =  $\begin{bmatrix} k & 0 \\ 0 & 1/k \end{bmatrix}$
- Rotation =  $\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$   
counterclockwise

© 2021 Sumudu Tennakoon

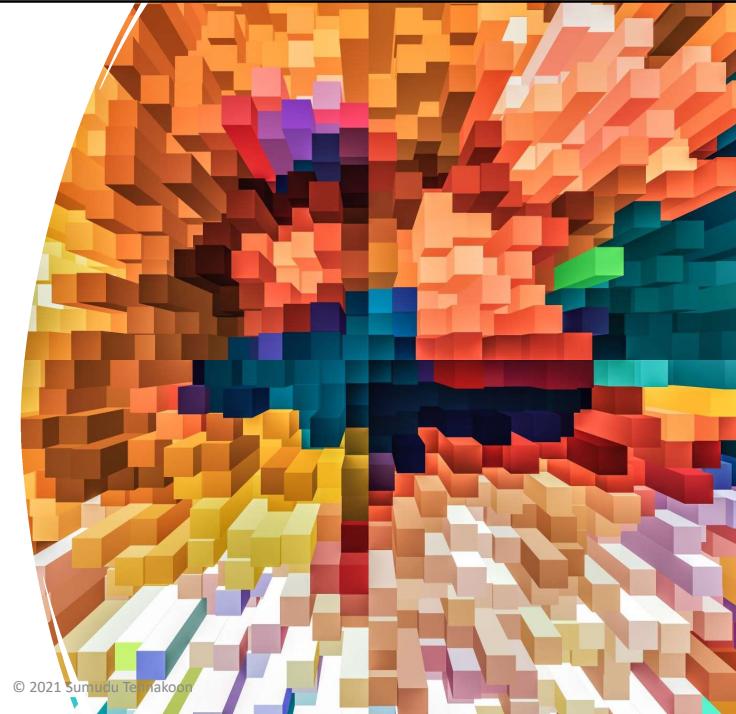
$$p = \begin{bmatrix} x \\ y \end{bmatrix}$$
$$p' = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$



85

# Sets

- 
- Basics of Set Theory
  - Set Operators
    - Union
    - Intersection
    - Complement
    - Difference



86

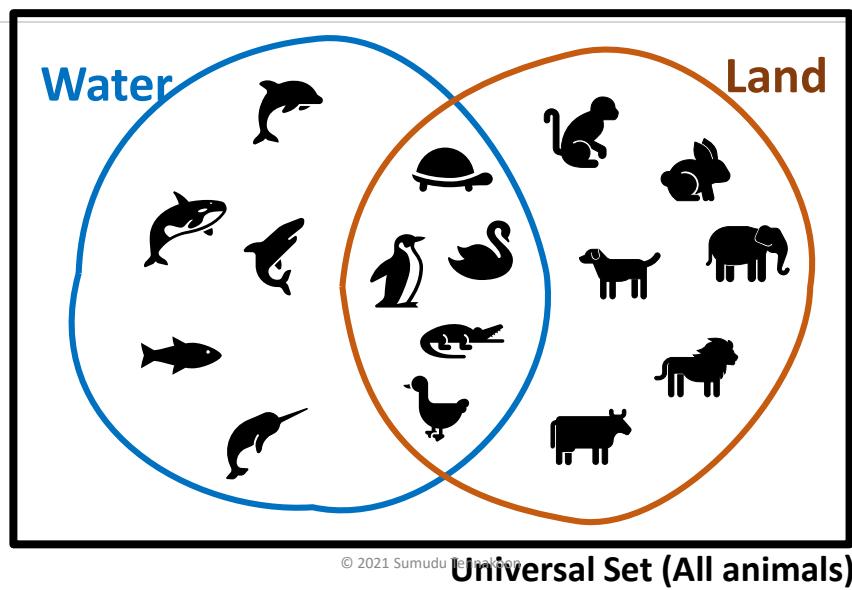
## Basics of Set Theory

- A set is a collection of things (elements).
  - $A = \{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$
  - $B = \{\times, \div, +, -\}$
  - $C = \{apple, orange, mango, banana\}$
  - $D = \{x | x \text{ satisfies some property}\}$
  - $\mathbb{N} = \{1, 2, 3, 4, \dots\}$  is set of natural numbers
  - $\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, 4, \dots\}$  is set of integers
  - $E = \{x | x \in \mathbb{Z}, -2 \leq x < 10\}$
  - $\emptyset = \{\}$  is Null set
- Items belongs to a set (element of)
  - $\heartsuit \in A$
  - $\div \in B$
  - $apple \in C$
- Items not belongs to a set (not an element of)
  - $x \notin A$
  - $*$   $\notin B$
  - $strawberry \notin C$
- Subset
  - $\{\diamondsuit, \heartsuit\} \subset A$
  - $\mathbb{N} \subset \mathbb{Z}$

© 2021 Sumudu Tennakoon

87

## Venn Diagrams



88

## Cardinality (size of the set)

$$A = \{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$$

$$|A| = 4$$

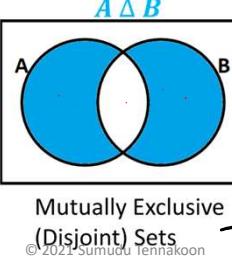
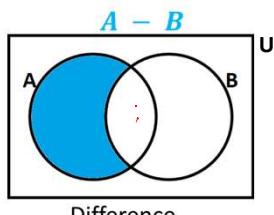
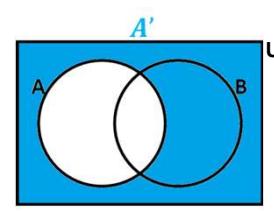
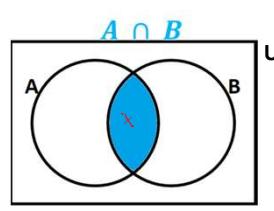
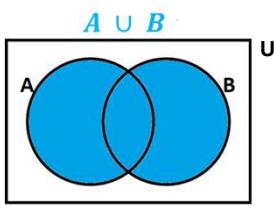
$$B = \{1, 2, 3, 4, 5, 6\}$$

$$|B| = 6$$

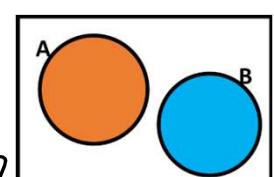
© 2021 Sumudu Tennakoon <https://en.wikipedia.org/wiki/Cardinality>

89

## Set Operations



© 2021 Sumudu Tennakoon



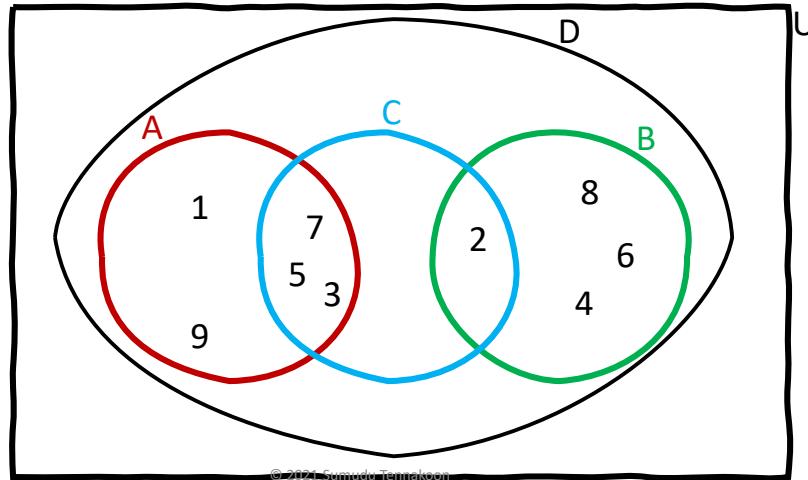
90

$$A = \{x \mid x \in \mathbb{Z}, 1 \leq x < 10, x \text{ is an odd number}\}$$

$$B = \{x \mid x \in \mathbb{Z}, 1 \leq x < 10, x \text{ is an even number}\}$$

$$C = \{x \mid x \in \mathbb{Z}, 1 \leq x < 10, x \text{ is a prime number}\}$$

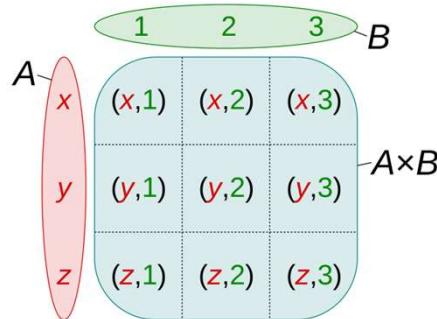
$$D = \{x \mid x \in \mathbb{Z}, 1 \leq x < 10\}$$



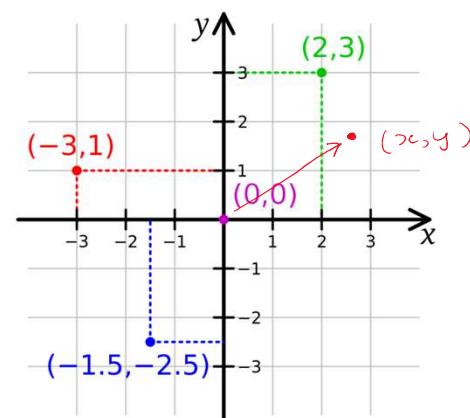
91

## Cartesian product

- $A \times B = \{(a, b) \mid a \in A \text{ and } b \in B\}$
- $A \times B \neq B \times A$



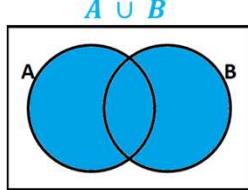
© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Cartesian\\_product](https://en.wikipedia.org/wiki/Cartesian_product)



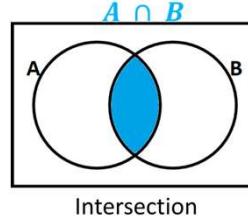
92

## Laws of Set Theory

- Identity
  - $A \cap U = A$
  - $A \cup \emptyset = A$



- Dominance
  - $A \cap \emptyset = \emptyset$
  - $A \cup U = U$



- Idempotence
  - $A \cap A = A$
  - $A \cup A = A$

- Complement
  - $A \cap A' = \emptyset$
  - $A \cup A' = U$

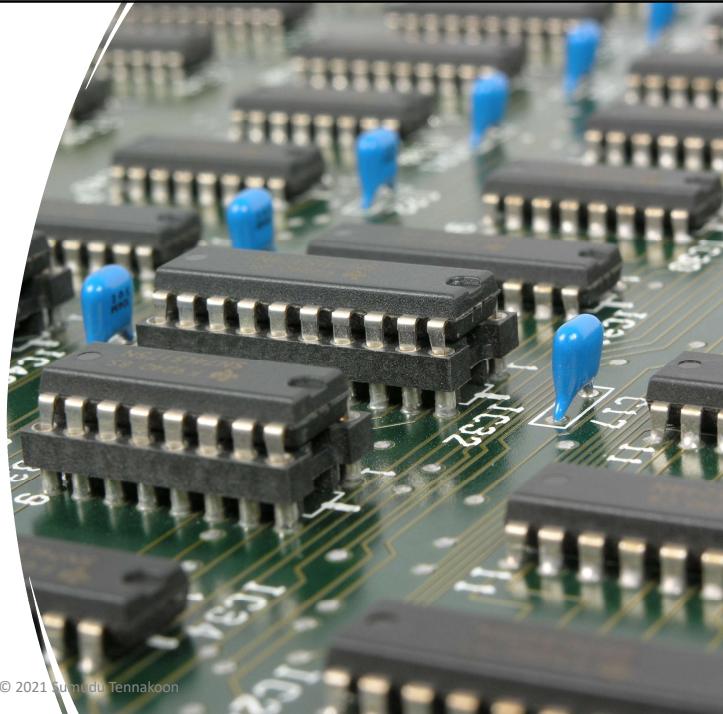
- Double Compliment
  - $(A')' = A$

© 2021 Sumudu Tennakoon

93

## Logic Gates

- Logical Statements
- Logical Operators
  - AND
  - OR
  - NOT



© 2021 Sumudu Tennakoon

94

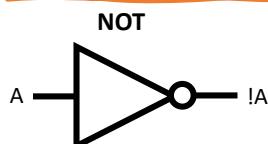
# Logical Statements and Binary Logic

- TRUE = 1
- FALSE = 0
- “5 is an odd number” is **TRUE**
- “4 is a prime number” is **FALSE**
- “Kandy is the Capital City of Sri Lanka” is **FALSE**
- “A triangle has three sides” is **TRUE**

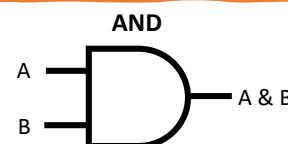
© 2021 Sumudu Tennakoon

95

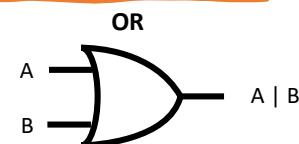
# Logic Gates



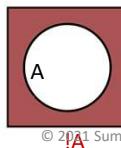
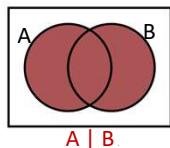
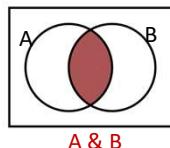
A	!A
1	0
0	1



A	B	A & B
1	1	1
1	0	0
0	1	0
0	0	0



A	B	A   B
1	1	1
1	0	1
0	1	1
0	0	0



© 2021 Sumudu Tennakoon

[https://en.wikipedia.org/wiki/Boolean\\_algebra](https://en.wikipedia.org/wiki/Boolean_algebra)

96

# Laws of Boolean Algebra

- Identity
  - A & 1 = A
  - A | 0 = A
- Annihilator
  - A & 0 = 0
  - A | 1 = 1
- Idempotence
  - A & A = A
  - A | A = A
- A is a Boolean variable
- A can take either 1 or 0
- Associativity
  - $A \& (B \& C) = (A \& B) \& C$
  - $A | (B | C) = (A | B) | C$
- Commutativity
  - $A \& B = B \& A$
  - $A | B = B | A$
- Distributivity
  - $A \& (B | C) = (A \& B) | (A \& C)$
  - $A | (B \& C) = (A | B) \& (A | C)$

© 2021 Sumudu Tennakoon

97

# Laws of Boolean Algebra

- Complement
  - $A \& !A = 0$
  - $A | !A = 1$
- Double Negation
  - $!(!A) = A$
- De Morgan's laws:
  - $!A \& !B = !(A | B)$
  - $!A | !B = !(A \& B)$

© 2021 Sumudu Tennakoon

98

# Exponents and Logarithms

	Exponent Representation	Logarithmic representation
Base n	$n^x = y$	$\log_n y = x$
Base 2	$2^3 = 8$	$\log_2 8 = 3$
Base 10	$10^3 = 1000$	$\log_{10} 1000 = 3$
Base e	$e^{6.907755} \approx 1000$	$\log_e 1000 \approx 6.907755$

$$\log_e(1000) = \ln(1000)$$

© 2021 Sumudu Tennakoon

99

# Functions

- $y = f(x)$

- $f(x) = \frac{1}{x}$

- $f(x) = a_0 + a_1x + a_2x^2$

- *Step Functions*

- $$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

- *Logistic Function*

- $$f(x) = \frac{L}{1+e^{-k(x-x_0)}}$$

© 2021 Sumudu Tennakoon

100

## Series

$$\bullet S_n = \sum_{i=1}^n a_i = a_1 + a_2 + \dots + a_n$$

- Example

- Let  $a_i = 2(i - 1) + 1$
- $a_1 = 1$
- $a_2 = 2 + 1 = 3$
- $a_3 = 2 \times 2 + 1 = 5$
- $a_4 = 2 \times 3 + 1 = 7$
- $S_4 = 1 + 3 + 5 + 7 = 16$

© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Series\\_\(mathematics\)](https://en.wikipedia.org/wiki/Series_(mathematics))

101

## Sequence

$$\bullet (a_i)_{i=1}^n = [a_0, a_1, a_2, \dots, a_n]$$

- Example

- Let  $a_i = i^2$
- $a_1 = 1$
- $a_2 = 2^2 = 4$
- $a_3 = 3^2 = 9$
- $a_4 = 4^2 = 16$
- $(i^2)_{i=1}^4 = 1, 4, 9, 16$

© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Series\\_\(mathematics\)](https://en.wikipedia.org/wiki/Series_(mathematics))

102

# Trigonometric Functions

- $\sin(x)$
- $\cos(x)$
- $\tan(x)$
- Sound Signal:
  - amplitude =  $\sin(\omega t) = \sin(2\pi ft)$

© 2021 Sumudu Tennakoon

103

## Take-home Exercise

- Mention one machine learning problem (specify whether it is a classification, regression or clustering problem) and the training approach you would take (supervised or unsupervised).
- Your answer must be 240 characters or fewer.

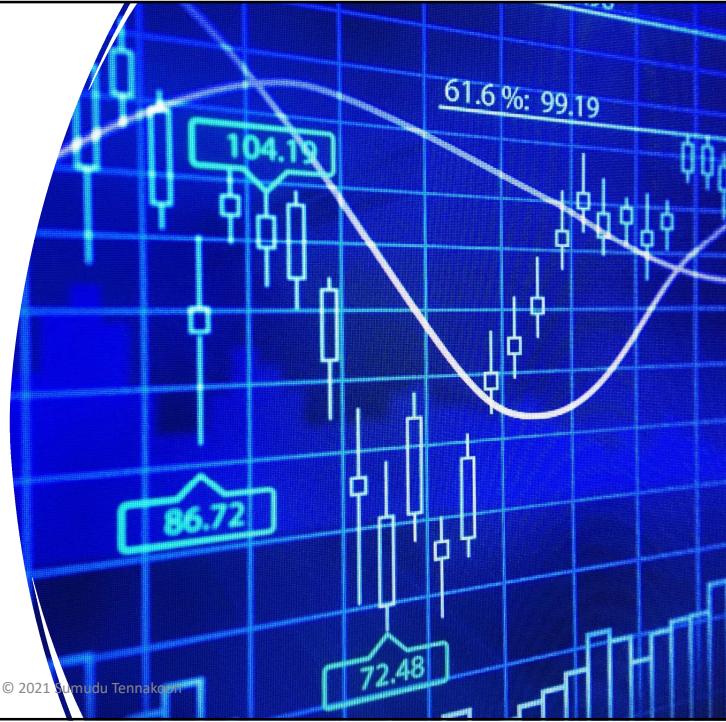


© 2021 Sumudu Tennakoon

104

# Probability and Statistics Concepts for Machine Learning

- Probability
- Statistical Distributions
- Descriptive Statistics



105

## Probability



© 2021 Sumudu Tennakoon

106

## Probability Example

- **Coin (single toss)**
  - Possible outcomes: {Head, Tail}
- **Dice (single roll)**
  - Possible outcomes: {1, 2, 3, 4, 5, 6}

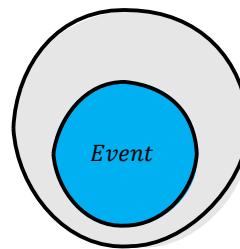


© 2021 Sumudu Tennakoon

107

## Term Definitions

- **Experiment:** A procedure that can be repeated and has a well-defined set of possible outcomes.
  - **Random experiment:** outcome is unknown before the experiment. Results different outcomes when repeating the experiment in the same manner.
  - **Deterministic experiment:** outcome may be predicted with certainty beforehand. Has a definite outcome. Outcome is predictable.
- **Sample Space ( $\Omega$ ):** Set of all possible outcomes or results of an experiment.
- **Outcome:** Possible result of an experiment. An element of  $\Omega$ .
- **Event:** A set of selected outcomes of an experiment. Subset of the sample space.
  - Independent Event: probabilities of occurrence do not depend on one another
- **Random Variable:** A variable that can take different values of the sample space (describes the outcomes of a random experiment).



© 2021 Sumudu Tennakoon

108

# Probability

- Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_n\}$  be a sample space of an experiment.
- Let the event  $A \subset \Omega$
- Probability (How likely the event A can occur) :  $P(A) = \frac{|A|}{|\Omega|}$
- $0 \leq P(A) \leq 1$
- Random Variable ( $X$ ):  $\{\omega \in \Omega | u < X(\omega) \leq v\}$
- Cumulative Probability:  $\sum_x P(X = x) = 1$

© 2021 Sumudu Tennakoon

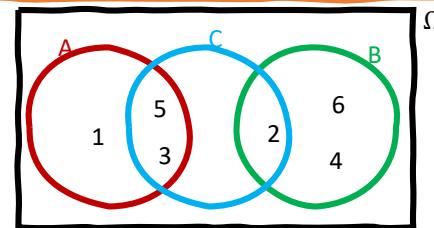
109

# Probability Example

- Coin (single toss)
  - Two possible outcomes
  - $\Omega = \{H, T\}$
  - $|\Omega| = 2$
  - $P(A = \{H\}) = \frac{1}{2}$
  - $P(A = \{T\}) = \frac{1}{2}$



- Dice (single roll)
  - Six possible outcomes
  - $\Omega = \{1, 2, 3, 4, 5, 6\}$
  - $|\Omega| = 6$
  - $P(A = \{1\}) = \frac{1}{6}$
  - $P(A = \{6\}) = \frac{1}{6}$
  - $P(A = \{1, 3, 5\}) = \frac{3}{6} = \frac{1}{2}$  (get odd number as outcome)



$$\text{Odd number: } P(A) = \frac{|A|}{|\Omega|} = \frac{1}{6} = \frac{1}{2}$$

$$\text{Even number: } P(B) = \frac{|B|}{|\Omega|} = \frac{1}{2}$$

$$\text{Prime number: } P(C) = \frac{|C|}{|\Omega|} = \frac{1}{2}$$

$$P(A \cup C) = \frac{|A \cup C|}{|\Omega|} = \frac{4}{6} = \frac{2}{3}$$

$$P(A \cap C) = \frac{|A \cap C|}{|\Omega|} = \frac{2}{6} = \frac{1}{3}$$

© 2021 Sumudu Tennakoon

110

# Conditional Probability

- Probability of A given B :  $P(A|B) = \frac{P(A \cap B)}{P(B)}$ , if  $P(B) \neq 0$

- Example

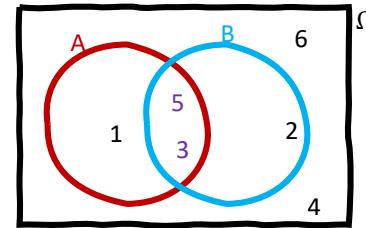
- $A = \{2,3,5\}$ , prime numbers
- $B = \{1,3,5\}$ , odd numbers



- If the outcome is an odd number, probability of that Number being a prime.

$$\begin{aligned} P(A|B) &= \frac{P(A \cap B)}{P(B)} = \frac{\frac{1}{3}}{\frac{1}{2}} = \frac{2}{3} \\ P(A \cap B) &= \frac{|A \cap B|}{|\Omega|} = \frac{|\{3,5\}|}{|\{1,2,3,4,5,6\}|} = \frac{2}{6} = \frac{1}{3} \\ P(B) &= \frac{|B|}{|\Omega|} = \frac{|\{1,3,5\}|}{|\{1,2,3,4,5,6\}|} = \frac{3}{6} = \frac{1}{2} \end{aligned}$$

© 2021 Sumudu Tennakoon



111

# Conditional Probability: Example

- $A$ : Selected person is vaccinated
- $B$ : Age of the selected person  $\geq 30$

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

- If a person with age  $\geq 30$  is selected, probability that the person is vaccinated?

$$\begin{aligned} \textcolor{red}{\bullet} \quad P(A) &= \frac{|A|}{|\Omega|} = \frac{55}{100} \\ \textcolor{magenta}{\bullet} \quad P(A \cap B) &= \frac{|A \cap B|}{|\Omega|} = \frac{45}{100} \\ \textcolor{green}{\bullet} \quad P(B) &= \frac{|B|}{|\Omega|} = \frac{50}{100} \\ \bullet \quad P(A|B) &= \frac{P(A \cap B)}{P(B)} = \frac{\frac{45}{100}}{\frac{50}{100}} = \frac{45}{50} = 0.9 \end{aligned}$$

	Vaccinated (YES)	Vaccinated (NO)	Total
Age < 30	10	40	50
Age $\geq 30$	45	5	50
Total	55	45	100

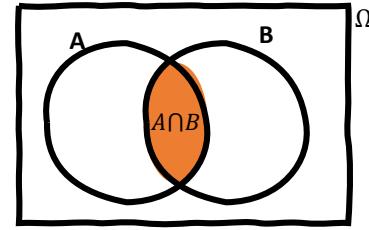
90 %

© 2021 Sumudu Tennakoon

112

# Bayes Rule

- Conditional Probability of A given B
  - $P(A|B) = \frac{P(A \cap B)}{P(B)}$ , if  $P(B) \neq 0 \rightarrow (1)$
- Commutativity in Set theory
  - $P(A \cap B) = P(B \cap A) \rightarrow (2)$
- (2) in (1)
  - $P(B \cap A) = P(A|B) \cdot P(B) \rightarrow (3)$
- Conditional Probability of B given A
  - $P(B|A) = \frac{P(B \cap A)}{P(A)} \rightarrow (4)$
- (3) in (4)
  - $P(B|A) = \frac{P(A|B) \cdot P(B)}{P(A)}$ , if  $P(A) \neq 0$



© 2021 Sumudu Tennakoon

[https://en.wikipedia.org/wiki/Bayes%27\\_theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem)

113

# Conditional Probability: Example

## Events

- $I$ : Infected
- $N$ : Not infected
- $T_P$ : Positive Test Result
- $T_N$ : Negative Test Result

## Measurements/Probabilities

- Infected population ratio:  $P(I) = 0.04 = 4\%$
- Positive Test Result if infected:  $P(T_P|I) = 0.98 = 98\%$
- Positive Test Result if not infected:  $P(T_P|N) = 0.01 = 1\%$
- A person get positive result regardless of the infection status:  $P(T_P)$ 
  - $P(N) = 1 - P(I)$
  - $P(T_P) = \sum_{x=\{I,N\}} P(x)P(T_P|x)$
  - $P(T_P) = P(I)P(T_P|I) + P(N)P(T_P|N)$
  - $P(T_P) = 0.04 \times 0.98 + 0.96 \times 0.01 = 0.0488$

John got a positive test result. What is probability that John is infected.

$$P(I|T_P) = \frac{P(T_P|I) \cdot P(I)}{P(T_P)} = \frac{0.98 \times 0.04}{0.0488} = 0.803$$

80 %

known

114

# Probability Distributions

## Discrete

- Probability distribution of a random variable that can take only a finite set of values.
- Probability mass function (PMF)
  - $\sum_{x \in X} P(x) = 1$

## Continuous

- Probability distribution of a random variable that can take an infinite number of values.
- Probability density function (PDF)
  - $\int P(x) dx = 1$

© 2021 Sumudu Tennakoon

[https://en.wikipedia.org/wiki/Probability\\_distribution](https://en.wikipedia.org/wiki/Probability_distribution)

115

# Combinatorics

## • Permutations

- Order matters
- Without repetition:  $P_x^n = \frac{n!}{(n-x)!}$
- With repetitions:  $n^x$

## • Combinations

- Order does not matter
- Can take with or without repetitions
- Without repetitions:  $C_x^n = \frac{n!}{x!(n-x)!}$
- With repetitions:  $C_x^{n+x-1} = \frac{(n+x-1)!}{x!(n-x)!}$

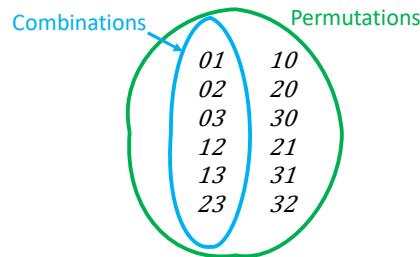
## Example

Set of Digits

- $S = \{0, 1, 2, 3\}$
- $n = |S| = 4$

Pick 2 elements (digits at a time) without repetition.

- $x = 2$



© 2021 Sumudu Tennakoon

116

# Permutations and Combinations

Set of Digits

- $S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- $|S| = 10$

$$P_x^n = \frac{n!}{(n-x)!}$$

$$C_x^n = \frac{n!}{x! (n-x)!}$$

Pick 4 elements (digits at a time)

E.g.,

- $A = 1234$
- $B = 2341$

How many permutations can be made?

How many combinations can be made?

© 2021 Sumudu Tennakoon

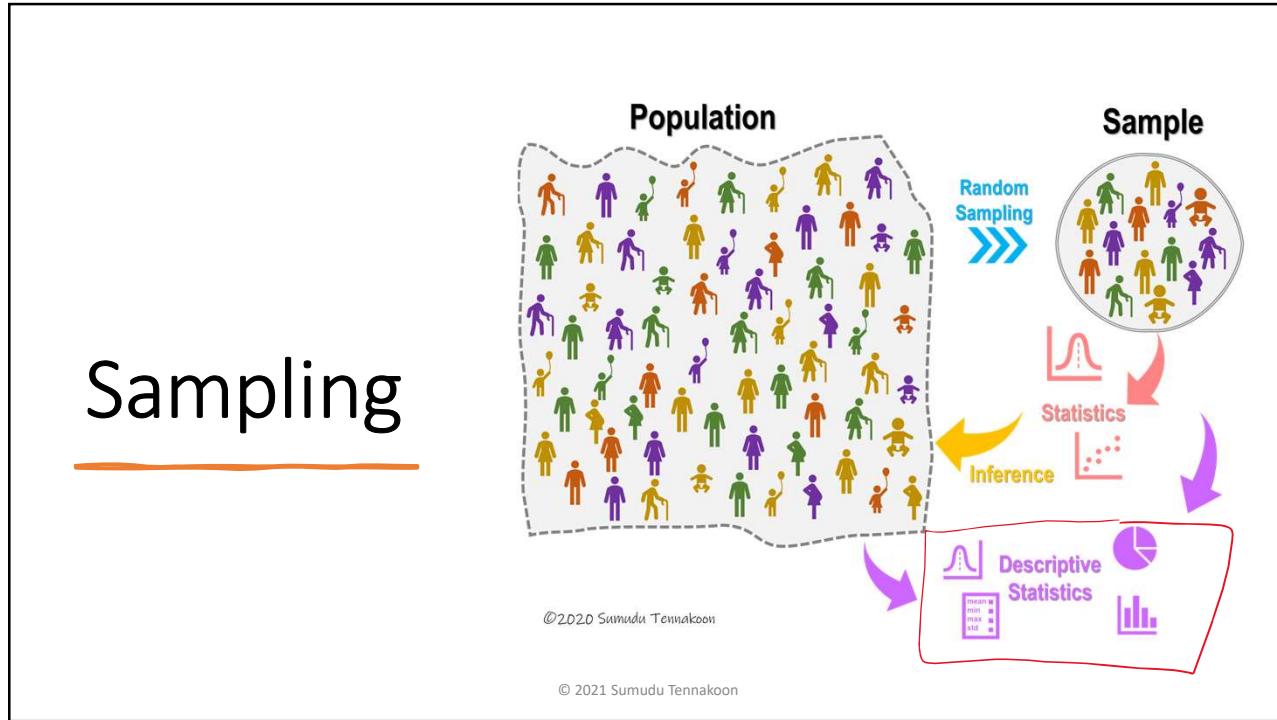
117

# Statistics

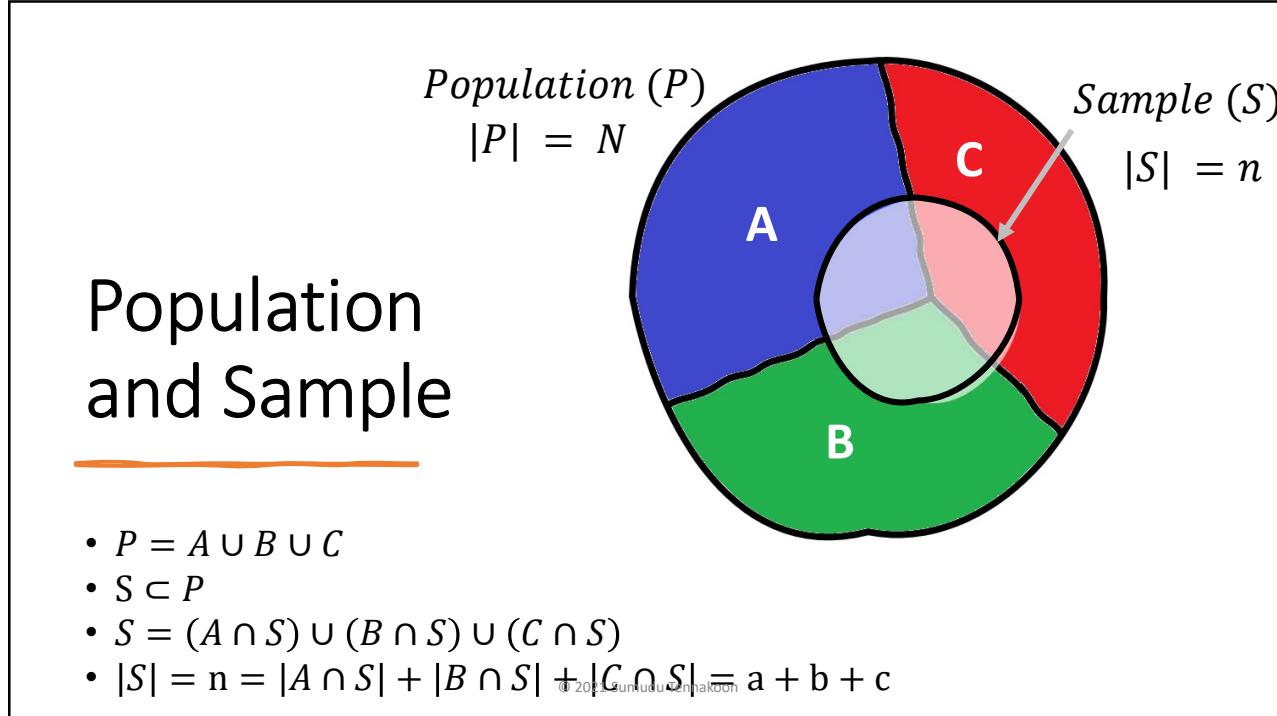


© 2021 Sumudu Tennakoon

118



119



120

# Sampling Method depends on

---

- Nature of population
- Regulatory Restrictions
- Resources available
  - Time
  - Money
- Data Collectors/Probes

© 2021 Sumudu Tennakoon

121

# Sampling Considerations

---

---

**Unbiased:** each element is equally likely to be chosen

---

**Representative of the Population**

---

**Minimize the Sampling Error**

---

© 2021 Sumudu Tennakoon

122

Sampling Technique	Description
Random Sampling	Sample in a random manner
Stratified Sampling	Represent subgroups
Systematic sampling	Include observations in population study
Cluster Sampling	Clusters observed represent population identified population
...	
	<a href="https://en.wikipedia.org/wiki/Sampling_(statistics)">https://en.wikipedia.org/wiki/Sampling_(statistics)</a>
	<a href="https://en.wikipedia.org/wiki/Stratified_sampling">https://en.wikipedia.org/wiki/Stratified_sampling</a>

123

Non-Probabilistic Sampling Techniques	<a href="https://en.wikipedia.org/wiki/Sampling_(statistics)">https://en.wikipedia.org/wiki/Sampling_(statistics)</a> <a href="https://en.wikipedia.org/wiki/Stratified_sampling">https://en.wikipedia.org/wiki/Stratified_sampling</a>
---------------------------------------	--

124

## Types of Statistics

**Descriptive Statistics:** Organizing and summarizing data. Use data from the population or sample.

Mean, Median, Mode, Range, Variance

**Inferential Statistics:** Use the sample data to make an inference or draw the population's conclusion. It uses probability to find out the confidence of the predictions we make.

Distributions, Regression, ...

© 2021 Sumudu Tennakoon

125

## Types of Analytics

**Descriptive:** What do we have now/ had

**Predictive:** What can happen if the conditions are set to

**Prescriptive:** Define rules/ constrains

© 2021 Sumudu Tennakoon

126

# Descriptive Statistics

- Range
- Mean
- Median
- Mode
- Variance
- Standard Deviation
- Coefficient of Variation
- Co-variance
- Co-relation Coefficient

© 2021 Sumudu Tennakoon

127

## Range, Mean, Median and Mode

- Range
  - Max Value - Min value
- Mean
  - Population Mean:  $\mu = \frac{\sum_{i=1}^n x_i}{N}$
  - Sample Mean:  $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$
  - Has influence of outliers
- Weighted Mean ( $w_i$ s are weights)
  - $\bar{x}_w = \frac{\sum_i^n w_i x_i}{\sum_{i=1}^n w_i}$
- Geometric Mean:
  - $\bar{x}_g = \sqrt[n]{\prod_{i=1}^n x_i}$
- Median:
  - Mid value when ordered
  - Removes influence of outliers
- Mode
  - Value that occur most often

© 2021 Sumudu Tennakoon

128

# Variance , Standard Deviation and Coefficient of variance

- Variance
  - How data spread around mean
  - Population :  $\sigma^2 = \frac{\sum_{i=1}^n (x_i - \mu)^2}{N}$
  - Sample :  $S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$
- Standard Deviation (STD)
  - Population :  $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{N}}$
  - Sample :  $S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$
- Coefficient of variance (CV)
  - Also known as relative standard deviation (RSD)
  - Population :  $CV = \frac{\sigma}{\mu}$
  - Sample :  $CV = \frac{S}{\bar{x}}$

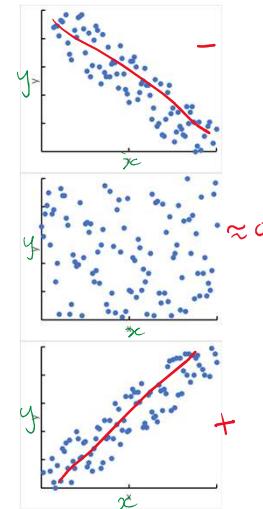
© 2021 Sumudu Tennakoon

129

## Co-variance and Co-relation Coefficient

- Co-variance:
  - Check whether 2 variables moving together
  - $\text{COV}(x,y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}$
  - $\text{COV} \begin{cases} < 0, & \text{if variables moving opposite direction} \\ = 0, & \text{if independent variables} \\ > 0, & \text{if variables moving together} \end{cases}$
- Co-relation Coefficient
  - $r = \frac{\text{COV}(x,y)}{S(x) \cdot S(y)}$
  - $-1 \leq r \leq 1$
  - $r = 1$  and  $r = -1$ : perfect corelation
  - $r \begin{cases} < 0, & \text{if negative corelation} \\ = 0, & \text{if independent variables} \\ > 0, & \text{if positive corelation} \end{cases}$

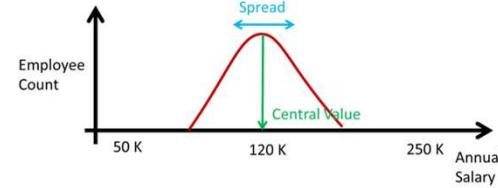
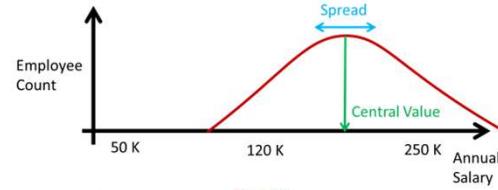
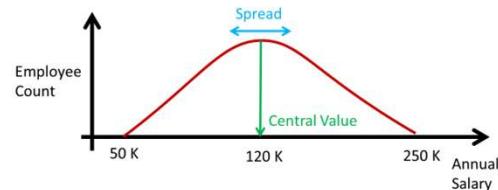
© 2021 Sumudu Tennakoon



130

## Distributions

### Central value and Spread

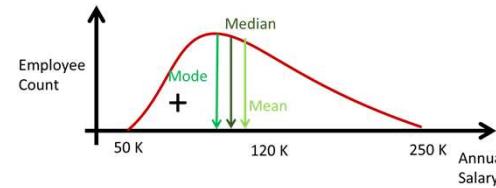
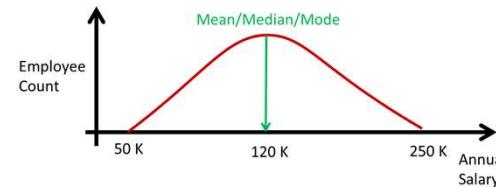
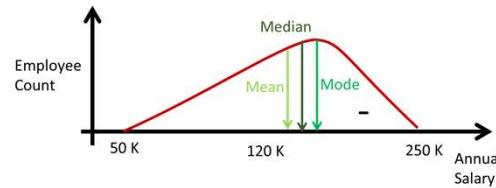


© 2021 Sumudu Tennakoon

131

## Distributions

### Skewness



<https://en.wikipedia.org/wiki/Skewness>

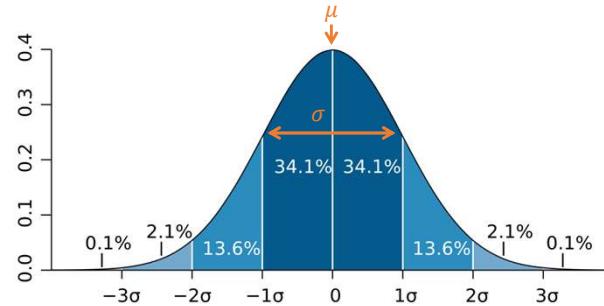
<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.skew.html>

<https://mathworld.wolfram.com/Pearson'sSkewnessCoefficients.html>

132

# Standard Normal Distribution

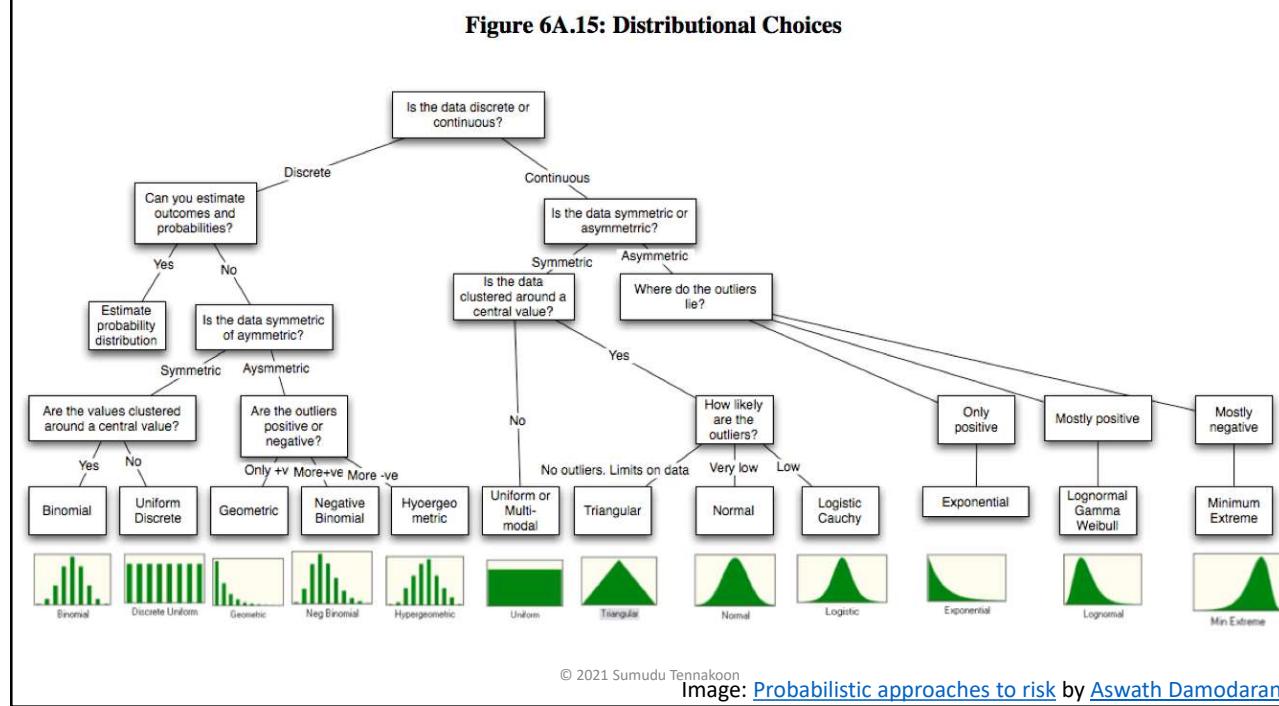
- Standard Deviation ( $\sigma$ ) = 1
- Mean ( $\mu$ ) = 0 = Median = Mode



© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Probability\\_distribution](https://en.wikipedia.org/wiki/Probability_distribution)

133

**Figure 6A.15: Distributional Choices**



© 2021 Sumudu Tennakoon  
Image: [Probabilistic approaches to risk](#) by Aswath Damodaran

134

# Exploratory Data Analysis and Feature Engineering

- Descriptive Statistics
- Univariate Analysis
- Multivariate Analysis



© 2021 Sumudu Tennakoon

135

## Range, Mean, Median and Mode

- Range
  - Max Value - Min value
- Mean
  - Population Mean:  $\mu = \frac{\sum_{i=1}^n x_i}{N}$
  - Sample Mean:  $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$
  - Has influence of outliers
- Weighted Mean ( $w_i$ s are weights)
  - $\bar{x}_w = \frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i}$
- Geometric Mean:
  - $\bar{x}_g = \sqrt[n]{\prod_{i=1}^n x_i}$
- Median:
  - Mid value when ordered
  - Removes influence of outliers
- Mod
  - Value that occur most often

© 2021 Sumudu Tennakoon

136

# Variance , Standard Deviation and Coefficient of variance

- Variance
  - How data spread around mean
  - Population :  $\sigma^2 = \frac{\sum_{i=1}^n (x_i - \mu)^2}{N}$
  - Sample :  $S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$
- Standard Deviation (STD)
  - Population :  $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{N}}$
  - Sample :  $S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$
- Coefficient of variance (CV)
  - Also known as relative standard deviation (RSD)
  - Population :  $CV = \frac{\sigma}{\mu}$
  - Sample :  $CV = \frac{S}{\bar{x}}$

© 2021 Sumudu Tennakoon

137

## Data Pre-Processing

### Treat Missing Values

### Treat Outliers

### Scaling Variables

© 2021 Sumudu Tennakoon

138

# Treat Missing Values

- Drop
  - Drop rows/columns with missing values ✗
- Impute
  - Impute (replacing with value)
    - Interpolate
    - Descriptive statistics: Median, Mean, Mode
    - Use machine learning model
    - Derive using other column (e.g., gender by title, name)
    - Replace with a dummy value (e.g., NA, N/A, null, -9999)
- Revisit
  - Revisit data collection
- Request
  - Request missing information from data provider

© 2021 Sumudu Tennakoon

139

# Treating Outliers

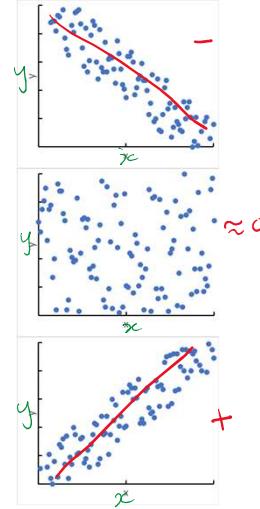
- Detecting Outliers
  - Box Plot
  - Percentiles
- Remove Outliers
  - Drop rows/columns
  - Replace outlier values
    - Set cutoff value/clip (piecewise function)
    - Convert to ranges

© 2021 Sumudu Tennakoon

140

## Co-variance and Co-relation Coefficient

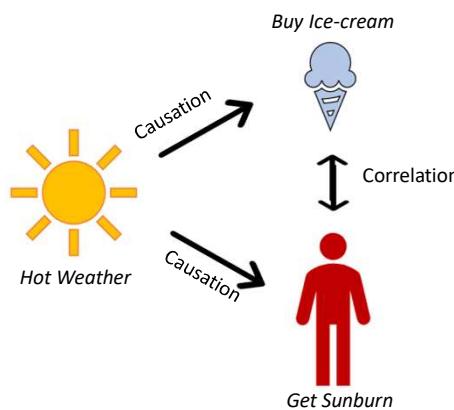
- Co-variance:
  - Check whether 2 variables moving together
  - $\text{COV}(x,y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}$
  - $\text{COV} \begin{cases} < 0, & \text{if variables moving opposite direction} \\ = 0, & \text{if independent variables} \\ > 0, & \text{if variables moving together} \end{cases}$
- Co-relation Coefficient
  - $r = \frac{\text{COV}(x,y)}{S(x) \cdot S(y)}$
  - $-1 \leq r \leq 1$
  - $r = 1$  and  $r = -1$ : perfect corelation
  - $r \begin{cases} < 0, & \text{if negative corelation} \\ = 0, & \text{if independent variables} \\ > 0, & \text{if positive corelation} \end{cases}$



© 2021 Sumudu Tennakoon

141

## Correlation & Causation



© 2021 Sumudu Tennakoon

142

# Univariate Analysis

- Single Variable
- Techniques
  - Descriptive Statistics
  - Frequency Table
  - Count Plot/Bar plot (categorical)
  - Histogram (numerical)
  - Box Plot

© 2021 Sumudu Tennakoon

143

# Bivariate Analysis

- Two variables
- Techniques
  - Cross tables (two-way tables)
  - Scatter Plot
  - Correlation coefficient
  - Stacked plot
  - Heatmaps
  - Pair Plot
  - Marginal Probability/Conditional Probability/Joint Probability
  - Regression

© 2021 Sumudu Tennakoon

144

# Preparing Dataset for Machine Learning

- Dimensionality Reduction
- Feature Selection
  - Select best features based on highest predictive power
- Feature Engineering
  - Derive new features from existing features
  - Transform features
  - Combine multiple features (create interactions)

© 2021 Sumudu Tennakoon

145

# Dimensionality Reduction

- Transform data from higher dimension space to lower dimension space (e.g., 4D to 3D, 3D to 2D)
- Retains characteristics of original data.
- Useful for
  - visualization purpose
  - Observe patterns more clearly
- Methods
  - Principle Component Analysis (PCA)
  - Linear discriminant analysis (LDA)
  - Kernel PCA
  - Autoencoder

© 2021 Sumudu Tennakoon

146

## Feature Scaling

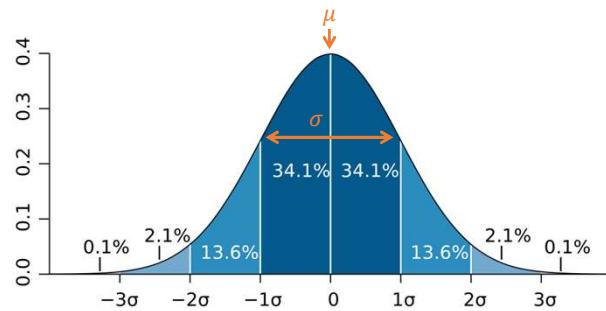
- Normalization (value between 0 and 1)
  - Min-max
  - Mean
- Standardization (scale based on standard normal distribution)
  - Z-score
- Other Scaling Techniques
  - Log transform
  - Power transform
- Not necessary in Tree based models

© 2021 Sumudu Tennakoon

147

## Standard Normal Distribution

- *Standard Deviation ( $\sigma$ ) = 1*
- *Mean ( $\mu$ ) = 0 = Median = Mode*



© 2021 Sumudu Tennakoon [https://en.wikipedia.org/wiki/Probability\\_distribution](https://en.wikipedia.org/wiki/Probability_distribution)

148

# Encoding

- Numerical Variables -> Categorical Variables
- Categorical Variables
  - One hot encoding (1/0, create dummy variables)
  - Mean Encoding (mean of each category)
  - Label Encoding (ordinal)
  - Target guided ordinal encoding (rank of mean)

ID#	Grade	A	B	C	Other	N/A
1	A	1	0	0	0	0
2	B	0	1	0	0	0
3	C	0	0	1	0	0
4	D	0	0	0	0	0
5		0	0	0	0	1

ID#	Gender	Male	Female	Is_male?	Gender_ME	Gender_TGOE
1	M	1	0	1	0.48	2
2	F	0	1	0	0.52	1
3	F	0	1	0	0.52	1
4	M	1	0	1	0.48	2
5	M	1	0	1	0.48	1

© 2021 Sumudu Tennakoon

149

# Feature Selection

- Based on Missing Values
  - Remove features with high % or missing values (define a threshold)
  - Create new feature to indicate missingness (binary column)
- Based on Variance
  - Remove features with zero variation
  - Remove features with low variation (define a threshold)
- Based on correlation
  - Analyze correlation between features (x variables) and those with the target (y) variable
  - Keep from a set of highly correlated variables
    - Keep one having highest correlation coefficient with the target (y) variable.
    - Drop features with low correlation with the target (y)
    - This could miss a useful feature, therefore use with caution

© 2021 Sumudu Tennakoon

150

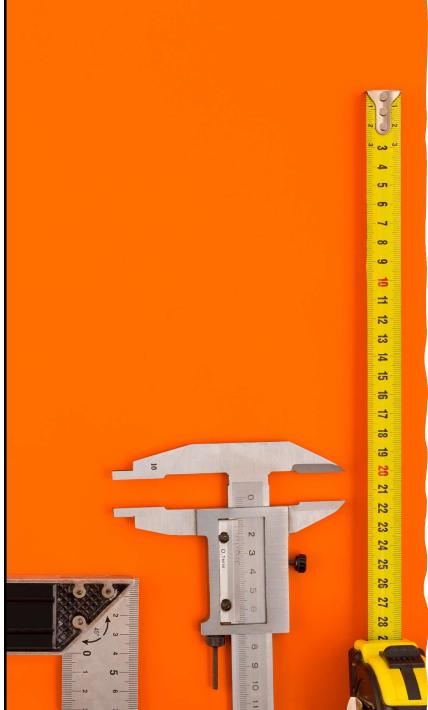
# Feature Selection

- Forward Selection
  - Start with best feature (feature with most predictive power) to build model
  - Evaluate the model performance
  - Add next feature, build model and evaluate
  - If the mode performance increase, keep feature if not, drop
- Backward Elimination
  - Start with all features, build model, evaluate
  - Drop least useful feature in each iteration

© 2021 Sumudu Tennakoon

151

# Evaluating Machine Learning Models



© 2021 Sumudu Tennakoon

152

	Positive	Negative	Total
Infected	48	2	50
Not Infected	1	49	50
Total	49	51	100

© 2021 Sumudu Tennakoon

153

## Confusion Matrix: Terms

- 
- condition positive (P)
    - the number of real positive cases in the data
  - condition negative (N)
    - the number of real negative cases in the data
  - true positive (TP)
  - true negative (TN)
  - false positive (FP)
  - false negative (FN)

© 2021 Sumudu Tennakoon

154

# Confusion Matrix

		Predicted condition		Sources: [13][14][15][16][17][18][19][20] view talk edit		
		Total population = P + N	Positive (PP)	Negative (PN)	Informedness, bookmaker informedness (BM) = TPR + TNR - 1	Prevalence threshold (PT) = $\frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	Positive (P)	True positive (TP), hit	False negative (FN), type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$	
	Negative (N)	False positive (FP), type I error, false alarm, overestimation	True negative (TN), correct rejection	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$	
	Prevalence $= \frac{P}{P+N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{N} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR-) $= \frac{FNR}{TNR}$	
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{TP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{N} = 1 - FOR$	Markedness (MK), deltaP ( $\Delta p$ ) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) = $\frac{LR+}{LR-}$	
	Balanced accuracy (BA) = $\frac{TPR + TNR}{2}$	F <sub>1</sub> score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times DFR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$	

© 2021 Sumudu Tennakoon

[https://en.wikipedia.org/wiki/Confusion\\_matrix](https://en.wikipedia.org/wiki/Confusion_matrix)

155

## Evaluating Classification Models

### Accuracy (ACC):

$$\frac{TP + TN}{P + N}$$

### Precision: Positive Predictive Value (PPV)

$$\frac{TP}{TP + FP}$$

### Recall: Sensitivity/Hit rate/True Positive Rate (TPR)

$$\frac{TP}{P}$$

### F1 score:

$$2 \times \frac{PPV \times TPR}{PPV + TPR} = \frac{2 \times TP}{2 \times TP + TN + FP + FN}$$

© 2021 Sumudu Tennakoon

156

## Visual Methods

Receiver operating characteristic(ROC) curve

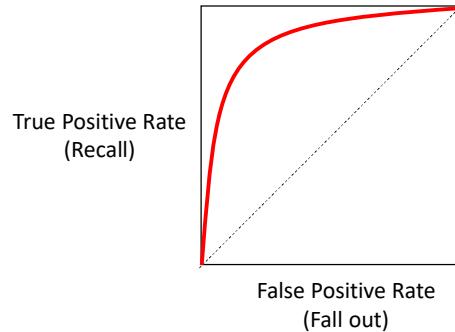
Precision Recall Curve

Sample Mean vs.  
Predicted Mean

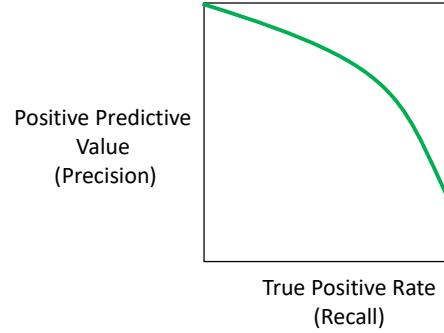
© 2021 Sumudu Tennakoon

157

**Receiver operating characteristic  
(ROC) curve**



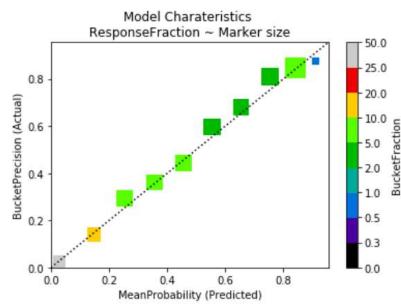
**Precision Recall Curve**



© 2021 Sumudu Tennakoon

158

## Sample Mean vs. Predicted Mean



Quantile	Mean Probability	Bucket Count	Response Count	Bucket Fraction	Response Fraction	Bucket Precision
1	0.0027	659	4	0.101	0.0026	0.0061
2	0.008	646	5	0.0992	0.0032	0.0077
3	0.0181	649	13	0.0996	0.0083	0.02
4	0.0376	651	23	0.1	0.0147	0.0353
5	0.0801	652	57	0.1	0.0365	0.0874
6	0.152	651	94	0.1	0.0602	0.144
7	0.252	651	191	0.1	0.122	0.293
8	0.385	651	253	0.1	0.162	0.389
9	0.561	651	381	0.1	0.244	0.585
10	0.8	652	540	0.1	0.346	0.828
<b>DataSet</b>	<b>0.23</b>	<b>6513</b>	<b>1561</b>	<b>1</b>	<b>1</b>	<b>0.24</b>

© 2021 Sumudu Tennakoon

159

## Evaluating Regression Models

### Mean Absolute Error (MAE):

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

### Mean Squared Error (MSE):

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

### Root Mean Squared Error (RMSE):

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

### Coefficient of Determination (R-squared):

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

© 2021 Sumudu Tennakoon

160

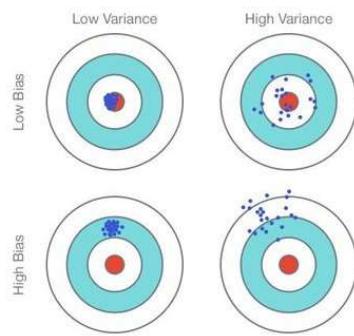


Fig. 1: Graphical Illustration of bias-variance trade-off , Source: Scott Fortmann-Roe., Understanding Bias-Variance Trade-off

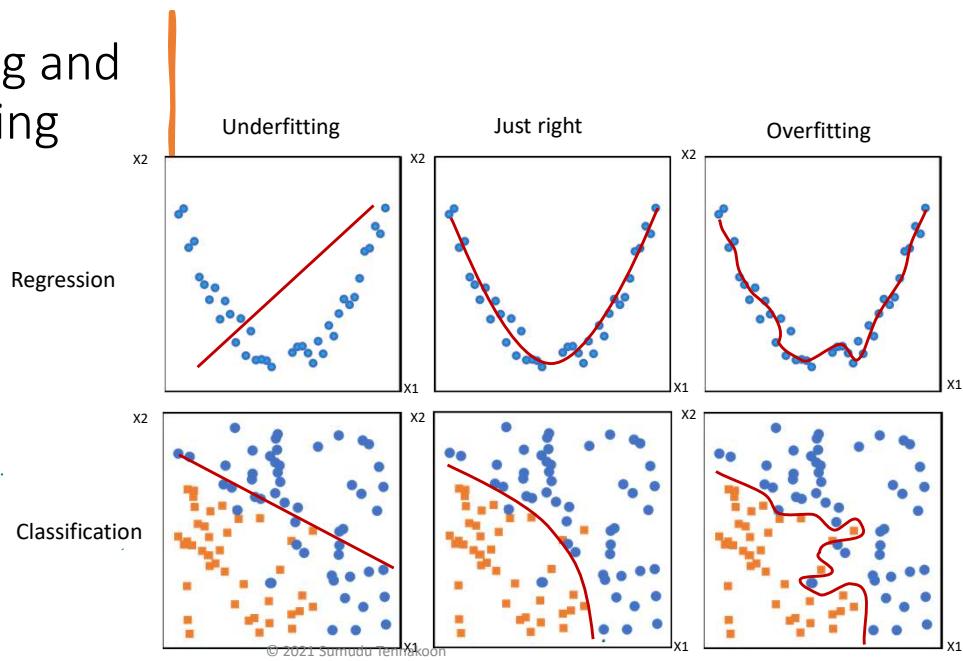
## Bias and Variance

- **Bias:** deviation from the actual value due to the model's simplistic assumptions in fitting the data.
  - High bias: model is unable to capture the patterns in the data and this results in **under-fitting**.
- **Variance:** deviation from the actual value due to the complex model trying to fit the data.
  - High variance: model passes through most of the data points, and it results in **overfitting** the data.

© 2021 Sumudu Tennakoon

161

## Overfitting and Underfitting

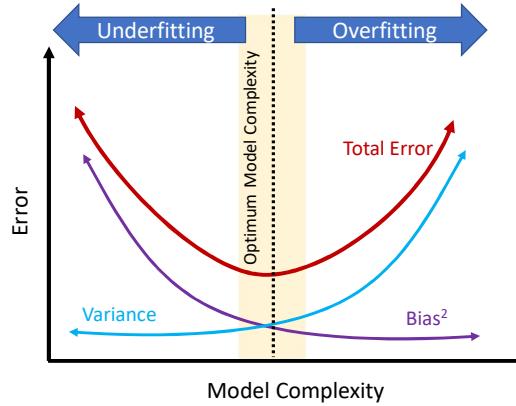


© 2021 Sumudu Tennakoon

162

# Bias-Variance Tradeoff

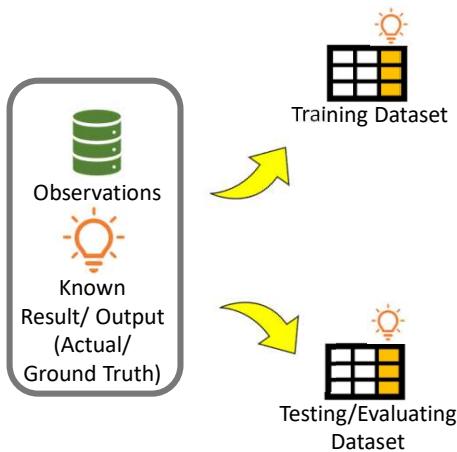
- Ideal Case for a machine learning model :
  - low variance and low bias.
- Reality
  - Bias-Variance trade-off
  - Find the model that performs well on both the train and unseen(test) data.



© 2021 Sumudu Tennakoon

163

# Train-Test Split

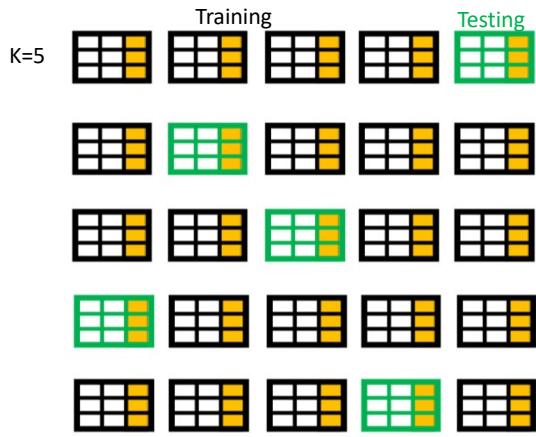


- No Training data should be used for Testing/Evaluating and vise versa.
- Keep Training and Testing Datasets Separate or properly labeled in the same dataset.
- Do train test split once and use the same two datasets over model tuning process.
- Use random Sampling
- Commonly used Train to Test ratio is 70:30

© 2021 Sumudu Tennakoon

164

# Cross Validation

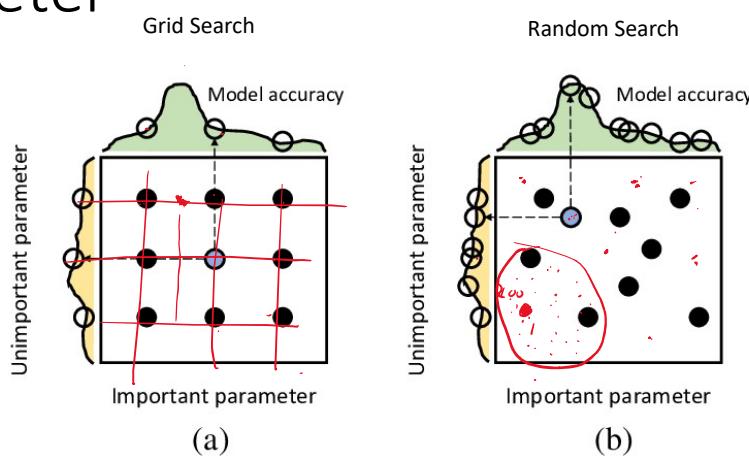


- If required to use of all data in the Training Process
  - E.g., small number of observations available.
- K-Fold Cross validation
- Split Dataset into K number of Samples
- Create K number of configurations holding one out of K sample as the Testing Sample
- Train Models using K folds and average results

© 2021 Sumudu Tennakoon

165

# Hyperparameter Tuning

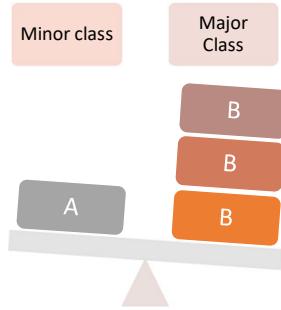


© 2021 Sumudu Tennakoon, Pilario, Karl Ezra & Cao, Yi & Shafiee, Mahmood. (2020)

166

## Class Imbalance

- Majority Class Minority Class
- Examples
  - Fraud Detection.
  - Churn Prediction.
  - Spam Detection.
- Ways to treat Imbalance
  - Collect More Data
  - (Re)Sampling
    - Stratified sample
    - Over Sampling (multiple copies of some of the minority classes)
      - SMOTE: Synthetic Minority Over-sampling Technique.
    - Under Sampling (Randomly remove samples from the majority class)
  - Data Augmentation (Generate Synthetic Data)
  - Using proper evaluation metric
    - Precision, Recall, F1 Score, AUC (ROC)
    - Do not use accuracy !
  - Split Larger Class into sub classes.
  - Use Ensemble Models



<https://arxiv.org/abs/1106.1813>

© 2021 Sumudu Tennakoon

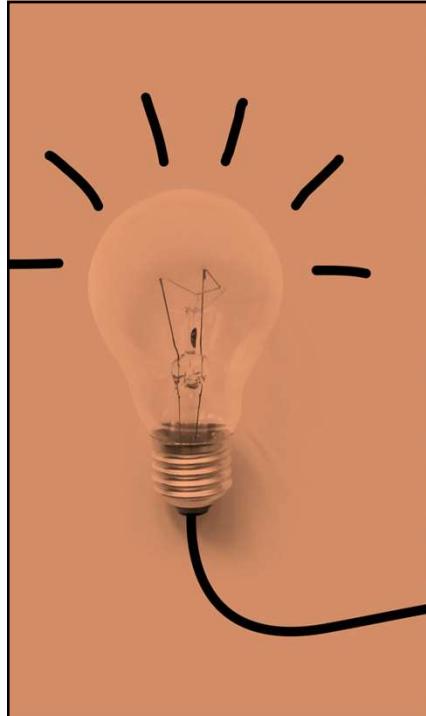
<https://link.springer.com/article/10.1007/s13748-016-0094-0>

167

## Machine Learning Algorithms

© 2021 Sumudu Tennakoon

168



## Supervised and Unsupervised Machine Learning

### Supervised learning

- $y = f(x_1, x_2, \dots, x_n)$
- $x_1, x_2, \dots, x_n$  are independent variables used to analyze the dependent variable ( $y$ ) and the relation between them.

### Unsupervised Learning

- No dependent variable.
- Starts with a collection of variables ( $x_1, x_2, \dots, x_n$ ) to find out similarity between them and classify them into clusters.

© 2021 Sumudu Tennakoon

169

## Supervised Machine Learning

### Regression

- Linear Regression
- Polynomial Regression
- Random Forest Regressor

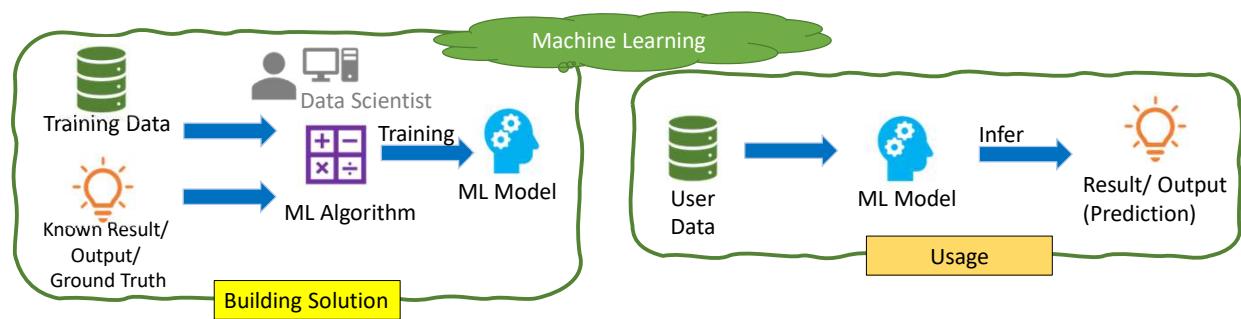
### Classification

- Logistic Regression
- Decision Trees
- Random Forest Classifier
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)

© 2021 Sumudu Tennakoon

170

## Unsupervised Machine Learning

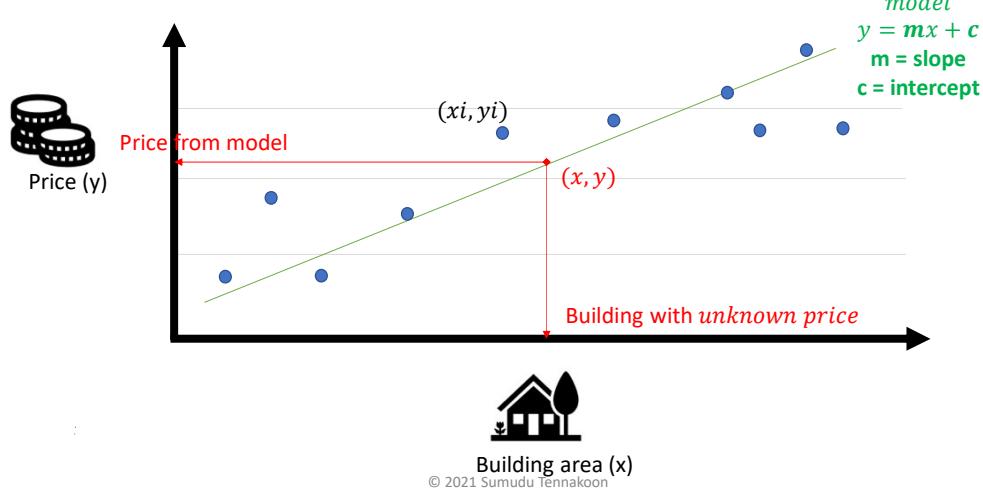


© 2021 Sumudu Tennakoon



171

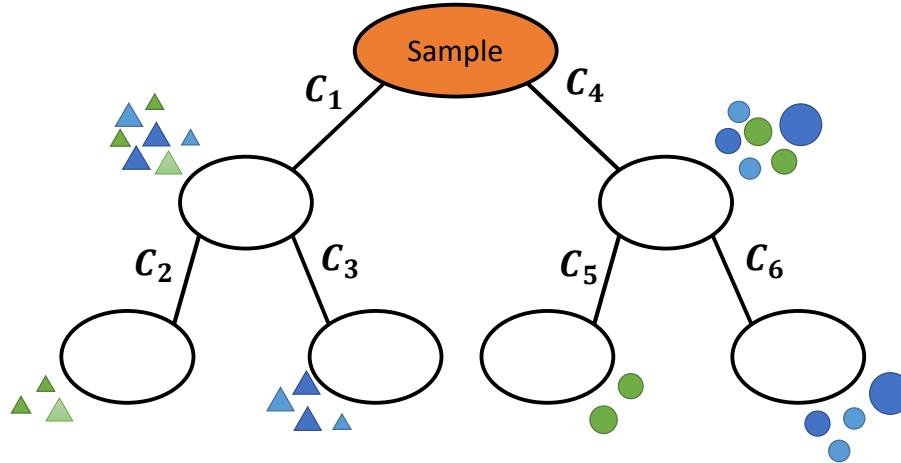
## Linear Regression



© 2021 Sumudu Tennakoon

172

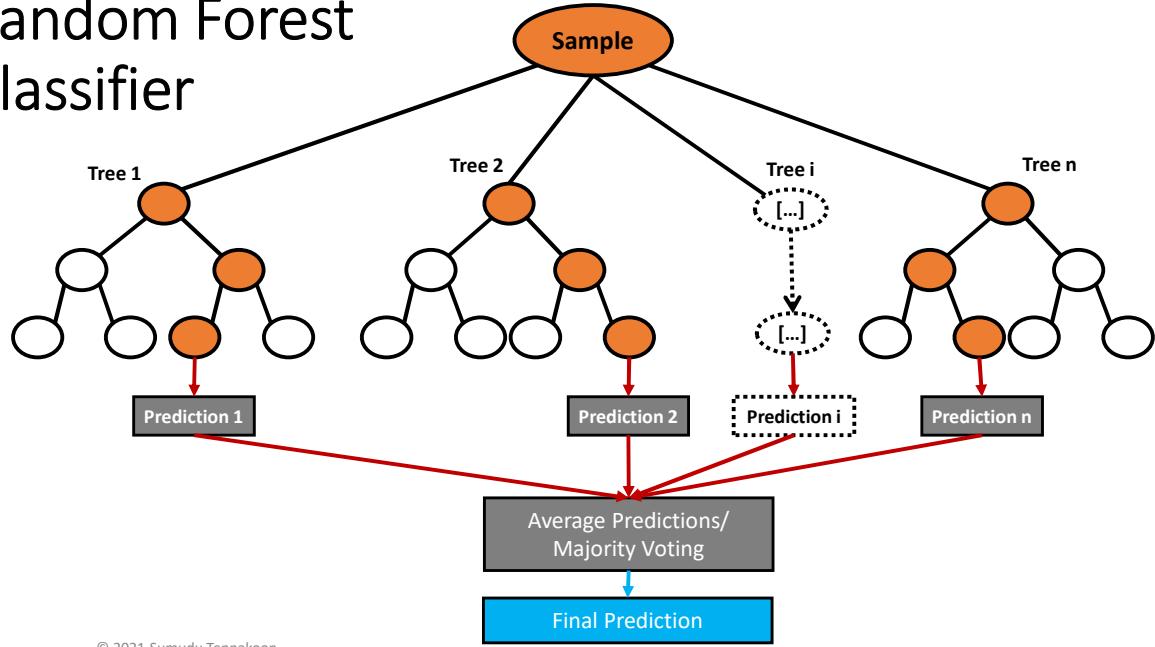
## Decision Tree Classifier



© 2021 Sumudu Tennakoon

173

## Random Forest Classifier



© 2021 Sumudu Tennakoon

174

# Unsupervised Machine Learning

## Clustering

- K-Means
- DBSCAN

## Dimensionality Reduction

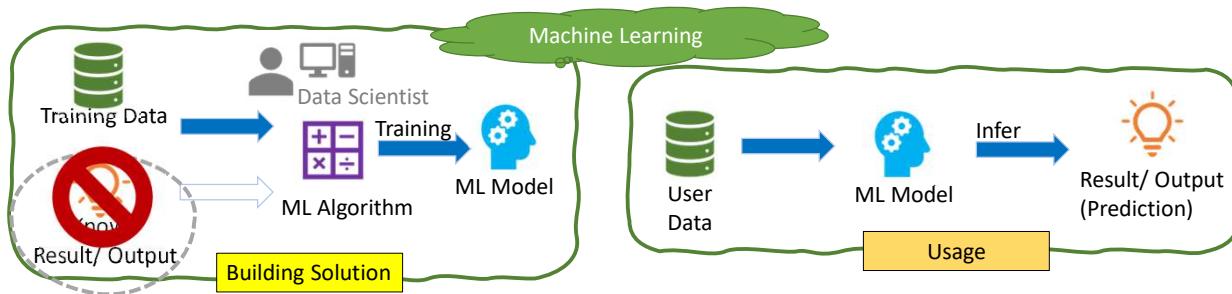
- PCA
- Autoencoder

Generative adversarial networks (GANs)

© 2021 Sumudu Tennakoon

175

# Unsupervised Machine Learning



© 2021 Sumudu Tennakoon

176

# Clustering

© 2021 Sumudu Tennakoon

Unsupervised learning

Group data points based on their similarities

Considerations

- Representation of a cluster with more than one data points.
- How we define the qualification for a datapoint to be a member of a cluster (nearness)
- When to stop the data point assignments to cluster

177

## Applications of Clustering

Market and Customer Segmentation

Recommendation engines

Anomaly Detection / Outlier Detection

Medical Data Analysis (Imaging, Disease Control, etc.)

Data Labeling (Cluster and Label)

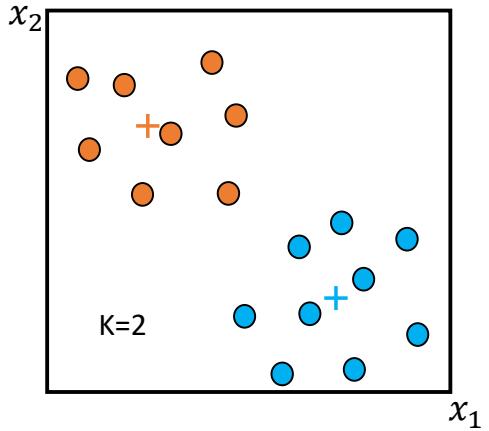
Understand the subgroups exists in data

- Number of groups
- Determine Group size
- Study group characteristics (aggregate values).
- Cohesion (togetherness) and separation
- Can those groups combine or further split?

© 2021 Sumudu Tennakoon

178

## Centroid Based Clustering

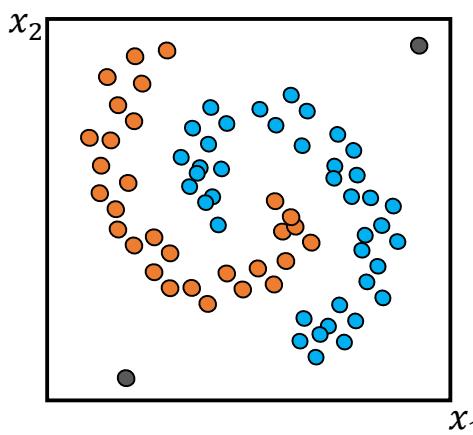


- Step 1: Place K centroids in random locations
- Step 2: Calculate the **distance** between each point to the centroid centers.
- Step 3: Find the nearest centroid to each data point using the get **min(distance)** and assign point to that centroid.
- Step 4: Calculate new center (mean of all assigned points) for the centroid based on the points.
- Loop: Repeat the Steps 2 -4 until converge (no points change cluster assignments)

© 2021 Sumudu Tennakoon

179

## Density Based Clustering

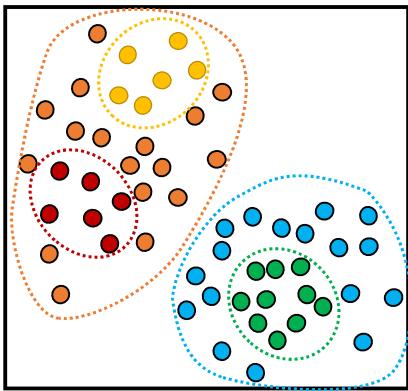


- Step 1: Select a random point previously unlabeled.
- Step 2: Count the number of other points ( $n$ ) enclosed (density) in the defined region.
- Step 3:
  - If  $n = 0$  where the current point is the only point in the region, mark the point as noise or outlier.
  - If  $n < N_{\min}$ 
    - If no core point is included in the region, mark the point as noise or outlier.
    - If at least one other core point is included in the region, mark it as a border point.
  - If  $n \geq N_{\min}$  mark the point as core point and initiate a cluster or assign the point to the current cluster.
- Step 4 (Loop):
  - If the point is a core point or border point, go to Step 2 covering all points in the cluster.
  - If the point is an outlier, go to Step 1
- Step 4 (Loop Break):
  - If all possible points covered in the current cluster (no more core points to be created), go to Step 1

© 2021 Sumudu Tennakoon

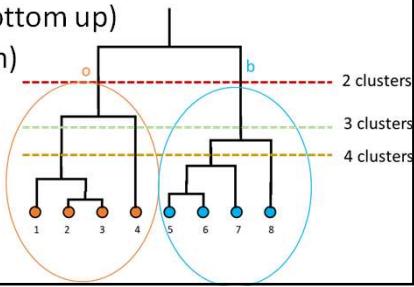
180

# Hierarchical Clustering



- Work well with data can be represented as heatmaps
- Dendrograms
- Methods
  - Agglomerative (bottom up)
  - Divisive (top-down)

© 2021 Sumudu Tennakoon



181

# Explaining ML Models and Predictions

---

Explainable AI (XAI)

© 2021 Sumudu Tennakoon

182

# Why do we need to Explain Models and their Predictions?

© 2021 Sumudu Tennakoon



183

## Explaining ML Models and Predictions

- Global
  - Overall understanding of how the model makes a prediction.
  - Feature Importance
- Local
  - Understand how model makes a prediction for a given observation.
  - Positive and Negative contribution of x variables (features) to the prediction.
  - Scale of contribution could be very different from model feature Importance
  - What are the variables contributed the most.

© 2021 Sumudu Tennakoon



184

## Explainable AI local methods

Local interpretable model-agnostic explanation (LIME)

Kernel Shapley additive explanations (KernalSHAP)

Integrated gradients (IG)

Explainable explanations through AI (XRAI)

© 2021 Sumudu Tennakoon

185

## Explainability

Models can be easily explained

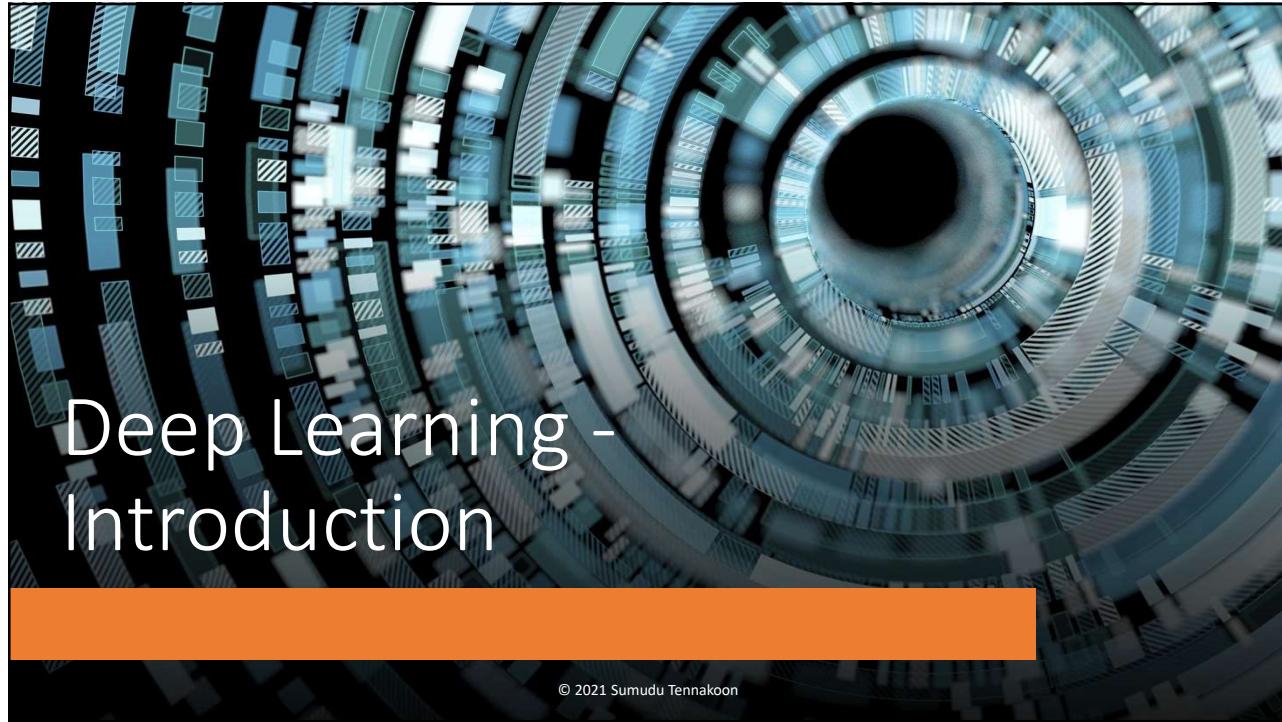
- Regression
- Decision Trees

Models cannot be easily explained (Blackbox Models)

- Ensemble Models (e.g., Random Forest)
- Neural Network (e.g., CNN, RNN)

© 2021 Sumudu Tennakoon

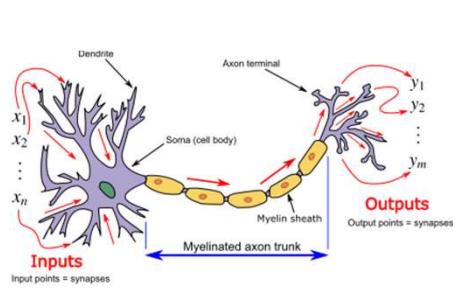
186



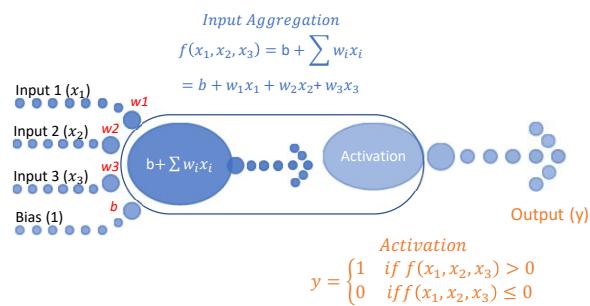
187

## Neuron and Perceptron

- We have approximately 86 billion neurons in our brain.



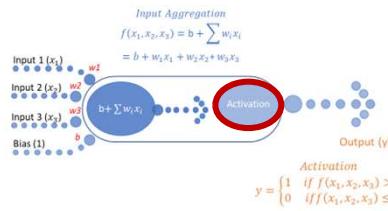
<https://en.wikipedia.org/wiki/Neuron>



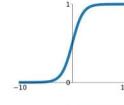
© 2021 Sumudu Tennakoon

188

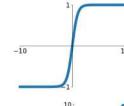
# Activation functions

**Sigmoid**

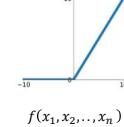
$$\sigma(x) = \frac{1}{1+e^{-x}}$$

**tanh**

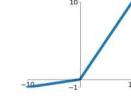
$$\tanh(x)$$

**ReLU**

$$\max(0, x)$$

**Leaky ReLU**

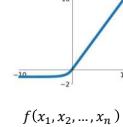
$$\max(0.1x, x)$$

**Maxout**

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

**ELU**

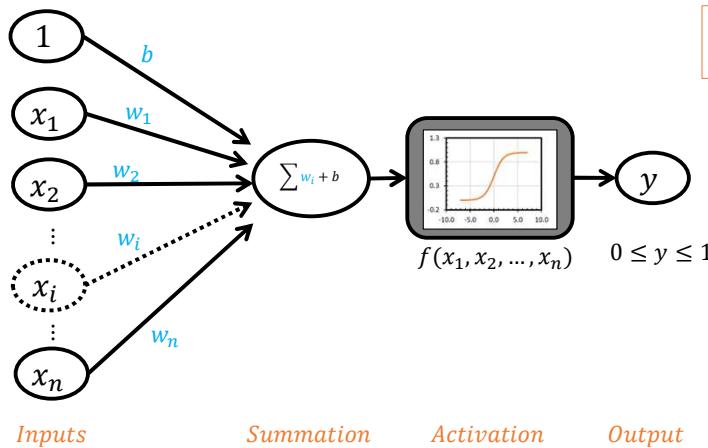
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



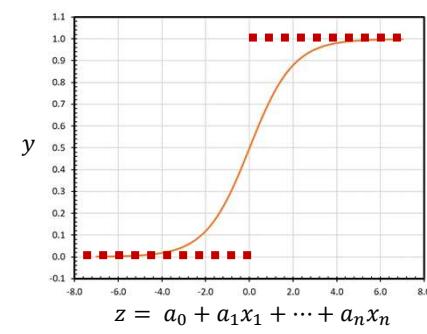
© 2021 Sumudu Tennakoon

189

## Logistic Regression as Perceptron



$$f(x_1, x_2, \dots, x_n) = \frac{1}{1 + e^{-(b + w_1 x_1 + \dots + w_n x_n)}}$$

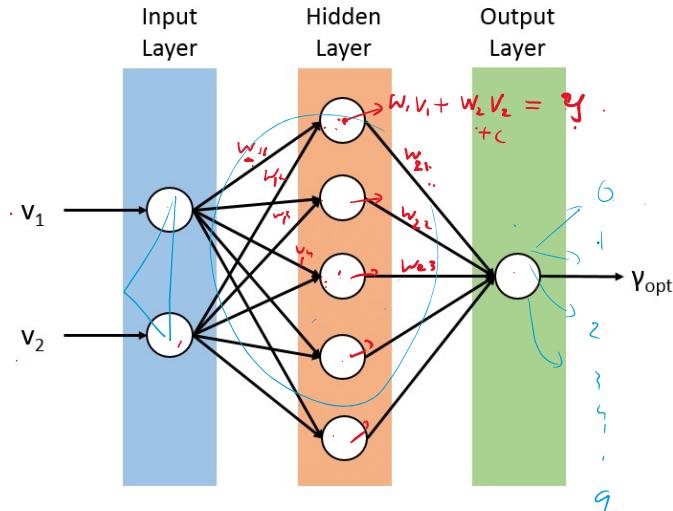


© 2021 Sumudu Tennakoon

190

# Neural Network

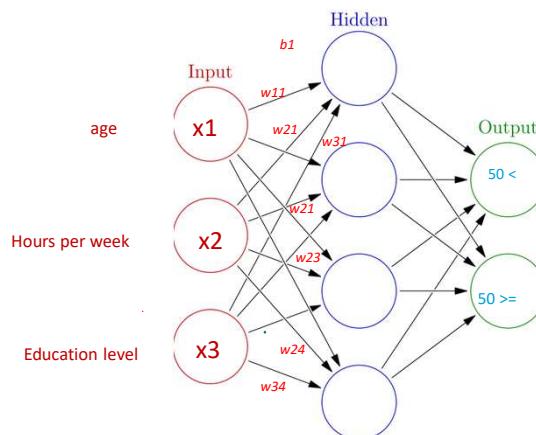
---



[https://en.wikipedia.org/wiki/Artificial\\_neural\\_network](https://en.wikipedia.org/wiki/Artificial_neural_network)

© 2021 Sumudu Tennakoon

191

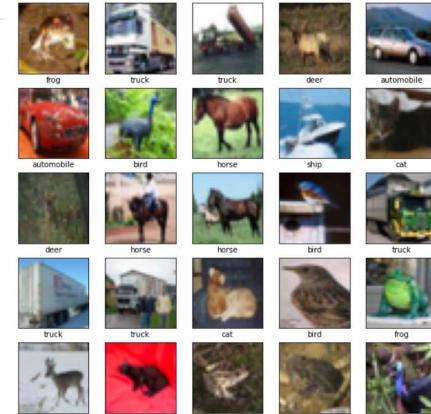
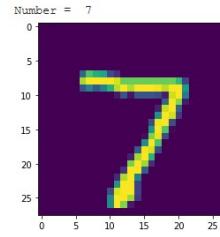
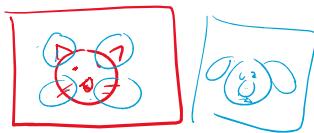


© 2021 Sumudu Tennakoon

192

## Traditional ML vs. Deep Learning

- Feature Engineering Workload
- Model Training Recourses
- Amount of Training Data Required
- Solving Complex Problems

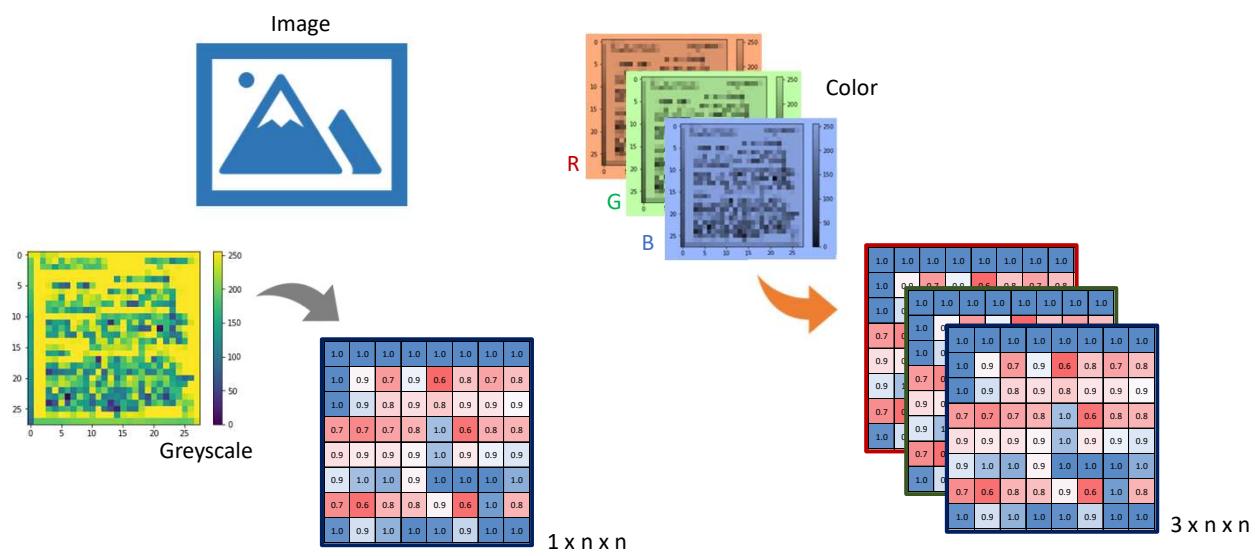


CIFAR10

© 2021 Sumudu Tennakoon

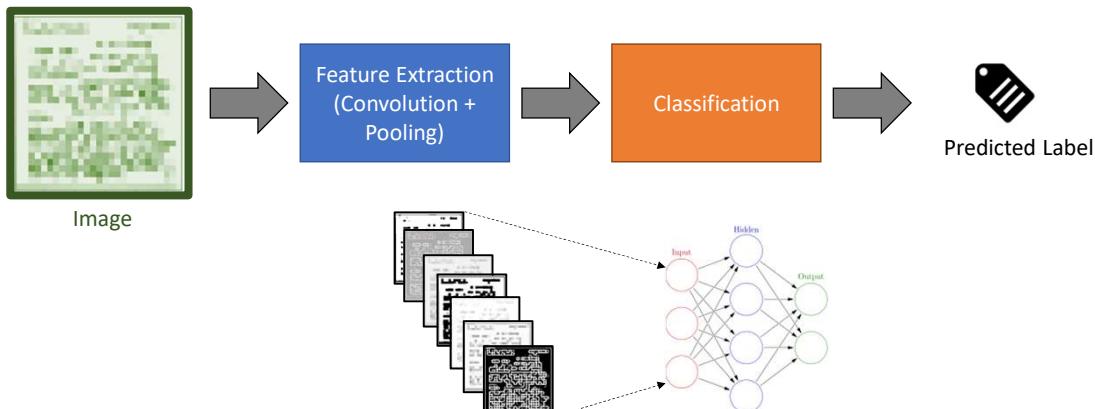
193

## Convolutional Neural Network



194

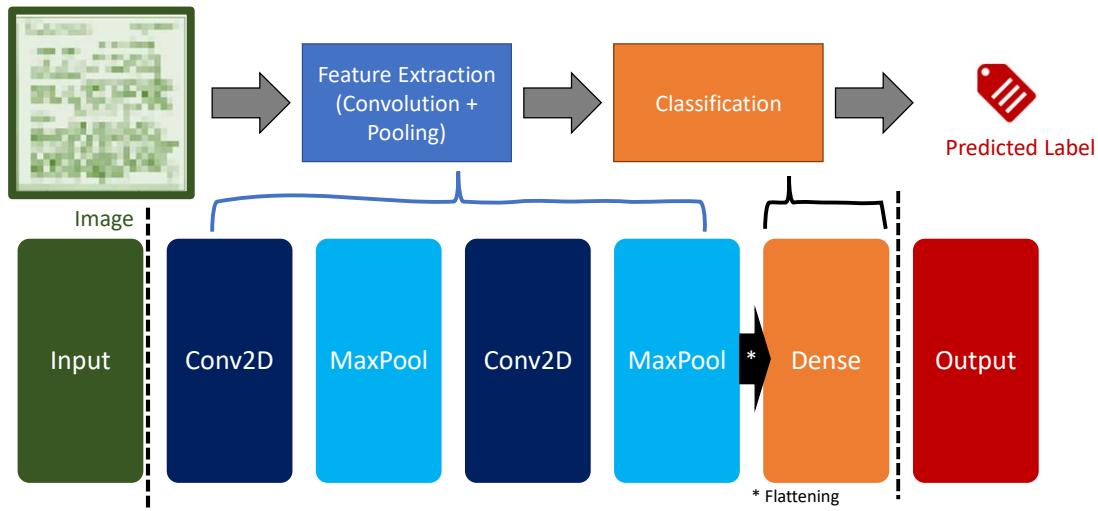
## Convolutional Neural Network



© 2021 Sumudu Tennakoon

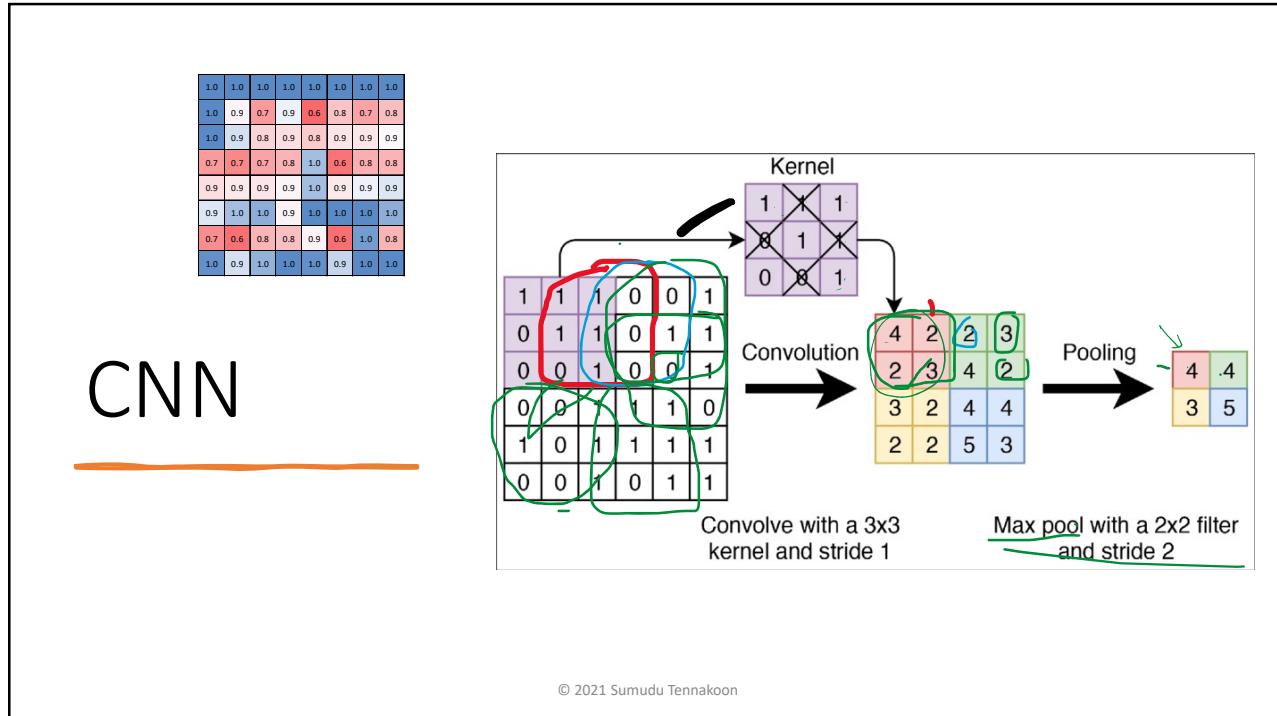
195

## Convolutional Neural Network

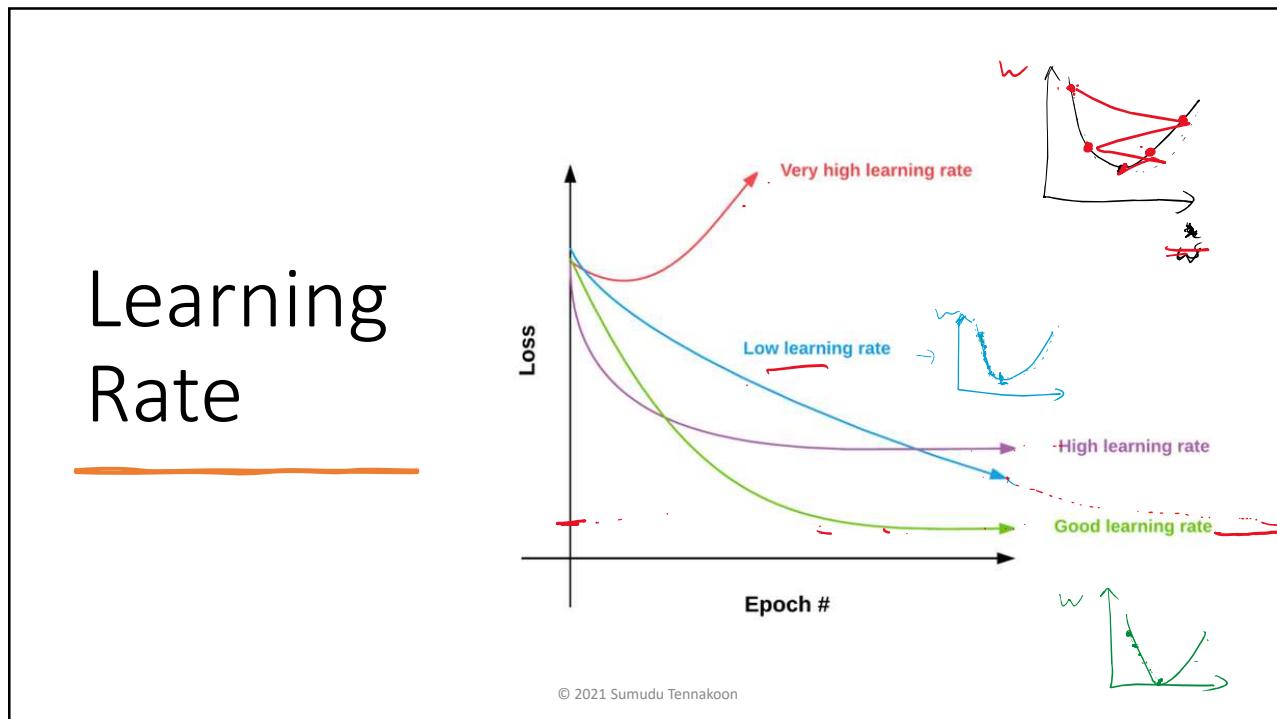


© 2021 Sumudu Tennakoon

196



197



198

# Regularization

---

$$\hat{y} = w_1x_1 + w_2x_2 + \dots + w_Nx_N + b$$

$$Loss = Error(y, \hat{y})$$

Loss function with no regularisation

$$Loss = Error(y, \hat{y}) + \lambda \sum_{i=1}^N |w_i|$$

Loss function with L1 regularisation

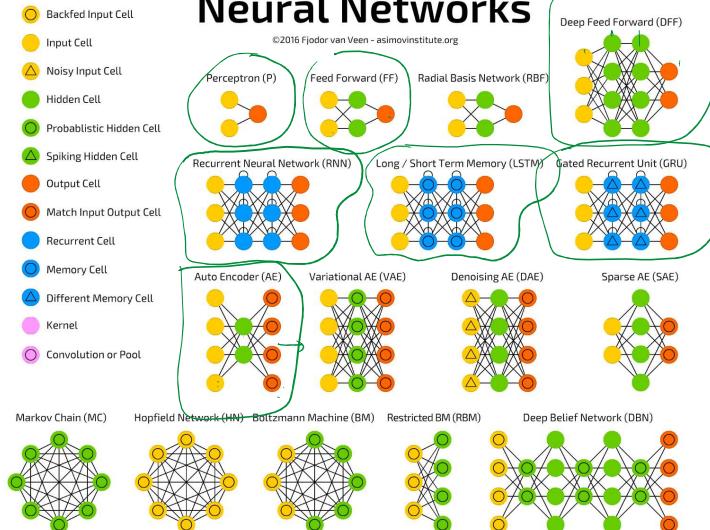
$$Loss = Error(y, \hat{y}) + \lambda \sum_{i=1}^N w_i^2$$

Loss function with L2 regularisation

© 2021 Sumudu Tennakoon

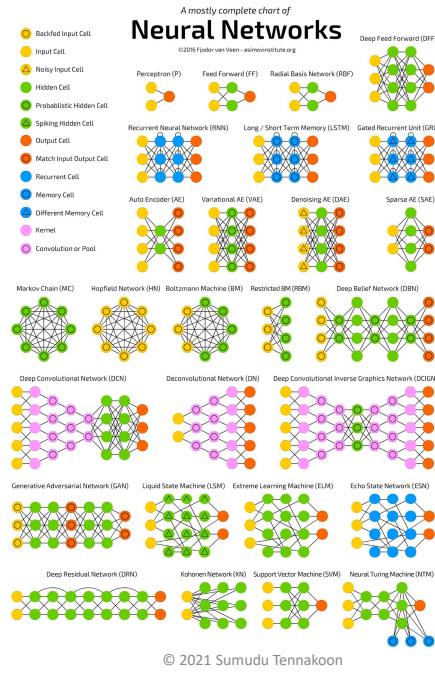
199

## A mostly complete chart of Neural Networks



© 2021 Sumudu Tennakoon

200

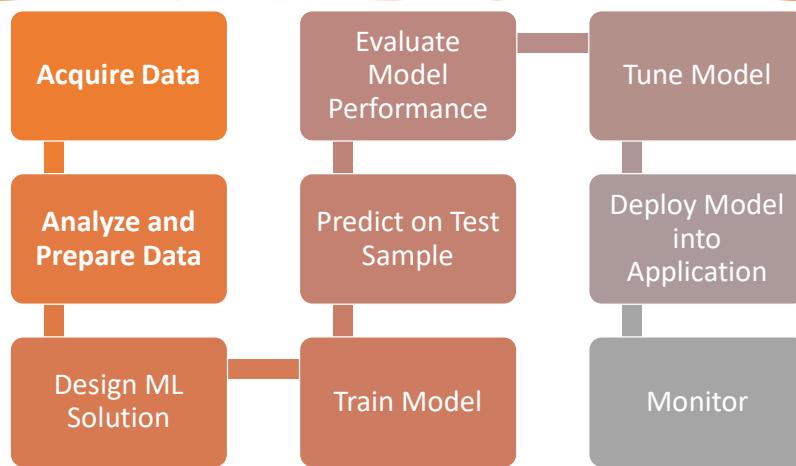


201



202

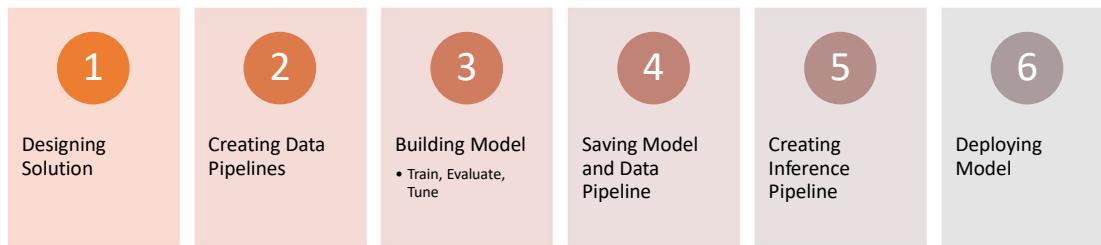
## Steps of Building ML Model



© 2021 Sumudu Tennakoon

203

## Design, Develop and Deploy ML Solutions



© 2021 Sumudu Tennakoon

204

# ML Solution Parts to Build

Data Pre-Processing Pipeline

**Input Module**

- Web form Interface
- API
- UI Application

**Pre-processing Pipeline**

- Feature Transformations
- Column Selection

Machine Learning Model Building Pipeline

**Post-processing Pipeline**

- Transform Prediction

Machine Learning Model

**Scoring Module**

- Model Prediction

Prediction (Scoring) Module

Post Processing Pipeline

**Output Module**

- Return result to user interface

Interface to accept User Input

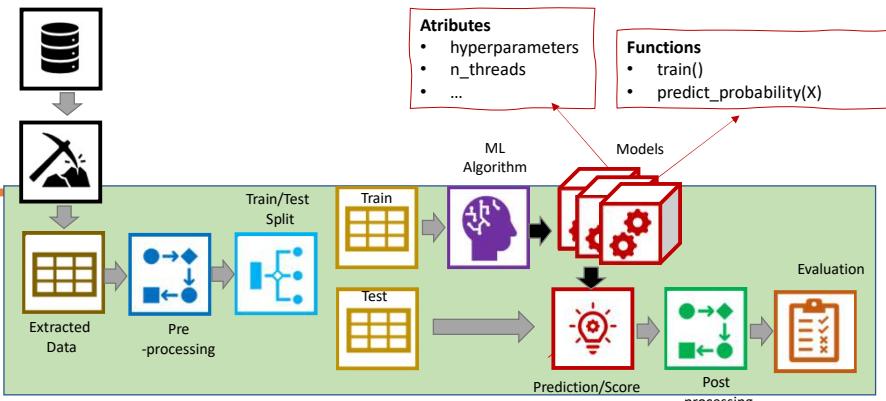
Interface to Return Prediction to User

© 2021 Sumudu Tennakoon

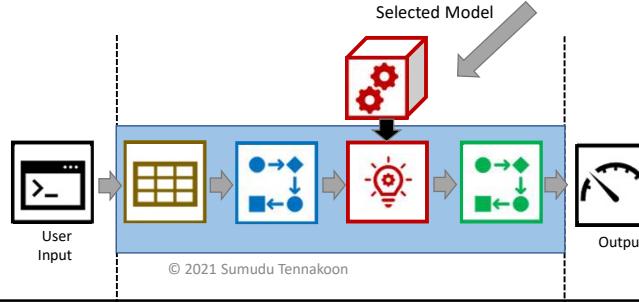
205

# Pipelines

**Model Training Pipeline**



**Inference Pipeline**



206

## Deployment

- Standalone Application
- Virtual Machines (VM)
- Containerized application
  - Docker  
(<https://www.ibm.com/cloud/learn/docker>)
  - Kubernetes  
(<https://www.ibm.com/cloud/learn/Kubernetes>)
- Locally hosted Web API
- Cloud API

© 2021 Sumudu Tennakoon

207

## Introduction to Azure ML Platform for Machine Learning Workflow

© 2021 Sumudu Tennakoon

208