REPUBLIC POLYTECHNIC
DISCOVER. TRANSFORM. ACHIEVE

Introduction to Pattern Matching and Anomaly Detection

Download materials at
http://bit.ly/iprad_jan20

Version 1.0

Before we start…

- No food and favored drinks in class. Plain water is fine.
- Be reminded that there is an expiry time for the food provided during tea break.
- Give me **feedback** as I need to know how you are doing so that I can adjust my pace or explain any concepts again.

source

Download from Github

http://bit.ly/iprad_jan20

Warm up!

Step 1: Go to the following url

http://bit.ly/warmup

Step 2: facilitator will walk you through the following question

1) **What do you hope to gain from this workshop.**

5 mins

Programme

Day 1	Introduction to Pattern Recognition and their techniques Activity – KNN & K-Means (iris) Common Python Lib - Numpy Activity - Numpy	Common Python Lib – Pandas, matplotlib, Scikit-learn Activity – Pandas, Matplotlib, Scikit-learn
Day 2	Data gathering to prepare for training and testing data Activity – Data Cleaning Use low-pass filter and simple moving average to detect abnormalities Activity – Low Pass Anomaly Detector	Introduction to anomaly detection and their techniques Introduction to H2O Activity - IRIS, MNIST, Fashion MNIST
Day 3	Activity – Credit card fraud with H2O/Isolation forest Activity – Intrusion detection system	Introduction to Alibi-Detect Activity – Image anomaly detection

Introduction of trainer

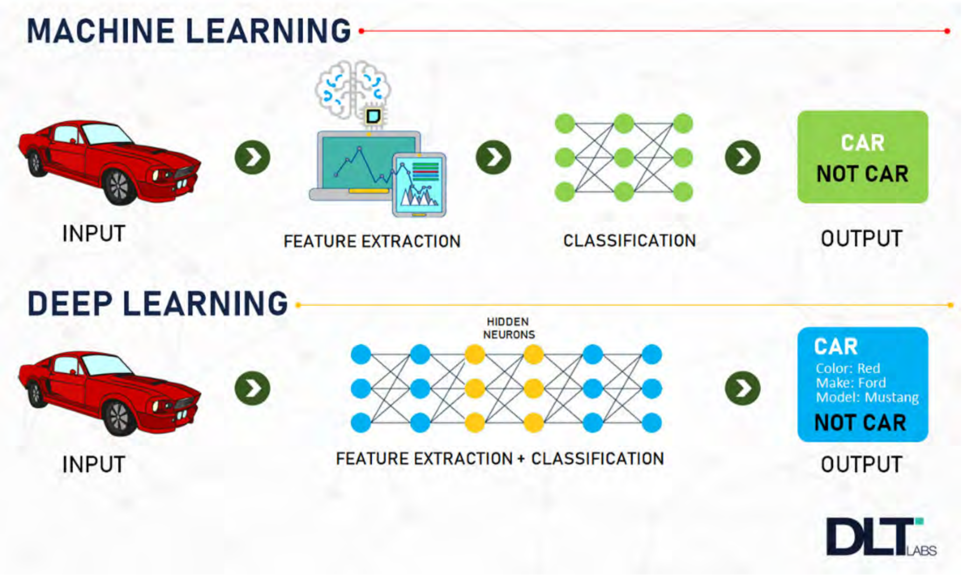
Name
Seow Khee Wei

Telegram
@kwseow

Email
seow_khee_wei@rp.edu.sg

Prerequisites

- Understanding of ML/DL
- Python



https://blog.usejournal.com/understanding-machine-learning-deep-learning-f5aa95264d61

Pattern Matching & Recognition

- Pattern Matching and Recognition is a branch of Machine Learning in identifying patterns and regularities in data
- In contrast to Pattern Recognition, in Pattern Matching, the outcome has to be an exact match
    - Pattern Recognition attempts are to find and seek for the pattern
    - Pattern Match have the knowledge of what are the pattern we are looking for

Examples

- Given the following images, are there any apples and the colour of the apples?

Yes, there are green and red apples. You are able to match the shape and colour

- Given the sequences of numbers, what is the next set of numbers (d)?

a. 1, 2, 3

b. 3, 4, 6

c. 7, 9, 24

d. ….. 15, 31, 216

You will recognize that the first number of b is the sum of (a) first and second number

The sum of (a) first and third will determine the second number of b.

Finally the third number is the product of (a) second and third

Possible pattern:

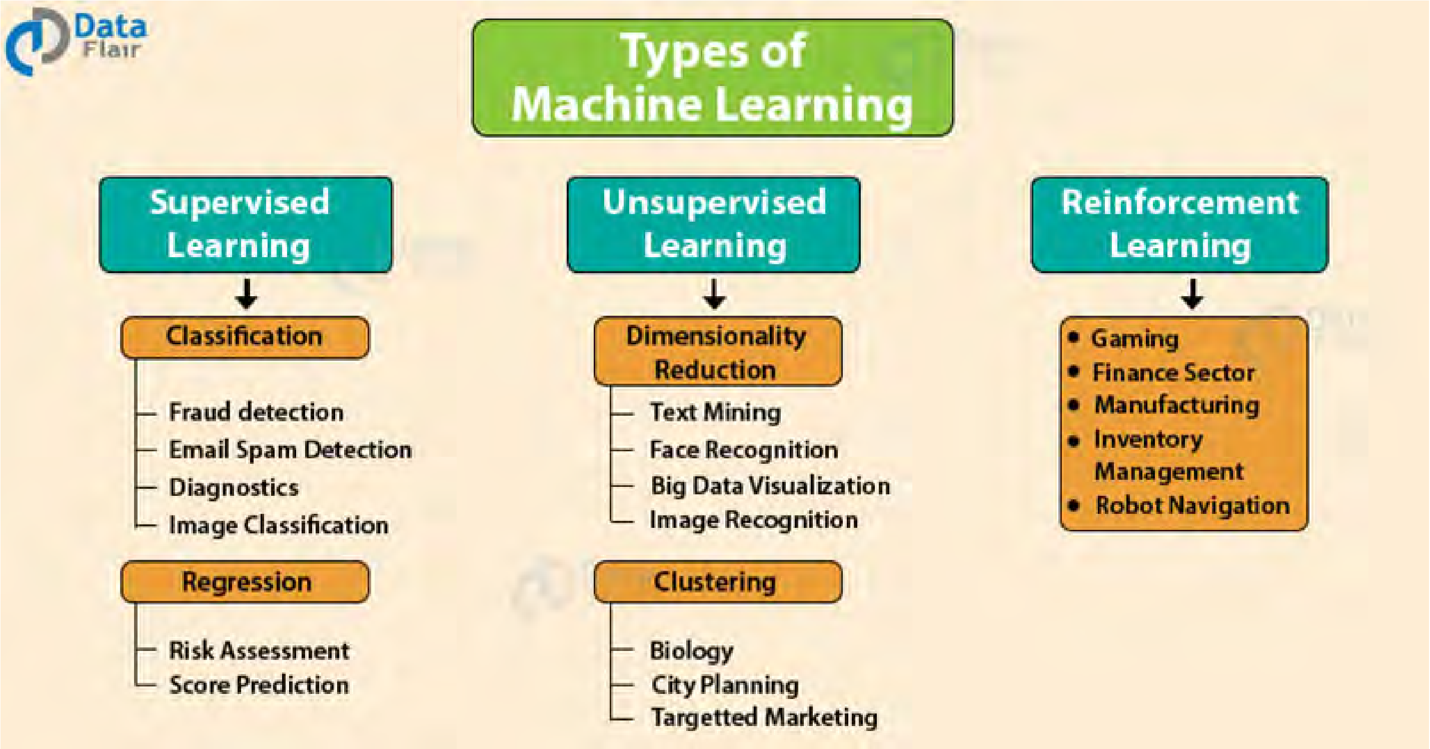
$x_{(i+1)} = x_i + y_i$

$y_{(i+1)} = x_i + z_i$

$z_{(i+1)} = y_i * z_i$

Methods

- Pattern Matching and Recognition involve a series of Machine Learning methods:
  - Supervised
  - Unsupervised
  - Reinforced

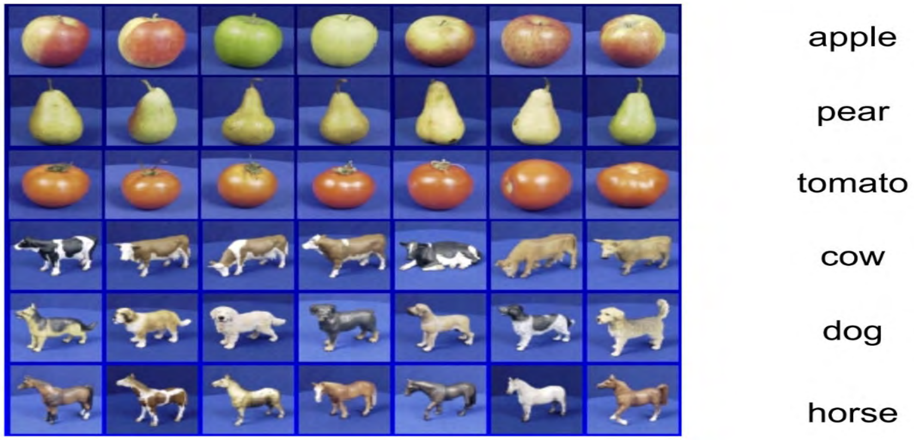


Machine Learning Methods

- Pattern matching and recognition will requires a model and predication that the machine learning system need to use to analyze the input samples and 'predict' the output
- Basically this steps is to group the different input samples into different classifications
- Typically, the input datasets are divided into the following groups:
  - Training
  - Testing
- Like a child, the machine need to be trained to understand and predict the outcome based on the given input

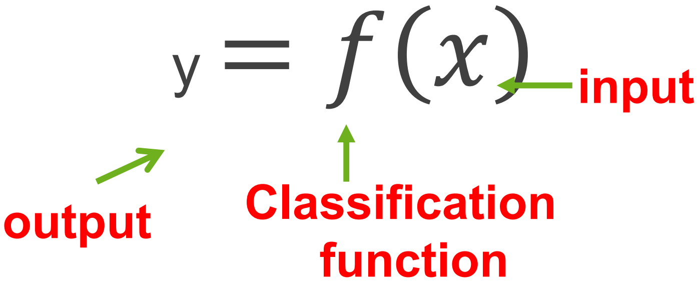
Supervised Learning

- Supervised Learning involves training the classifier using labeled examples
- Supervised Learning compares predictions by the model to known answers and makes corrections in the model
- To train the model in Supervised Learning, it will need a set of training data with label examples



https://www.youtube.com/watch?v=Ki2iHgKxRBo

Basic Framework of Supervised Learning



$$y = f(x)$$

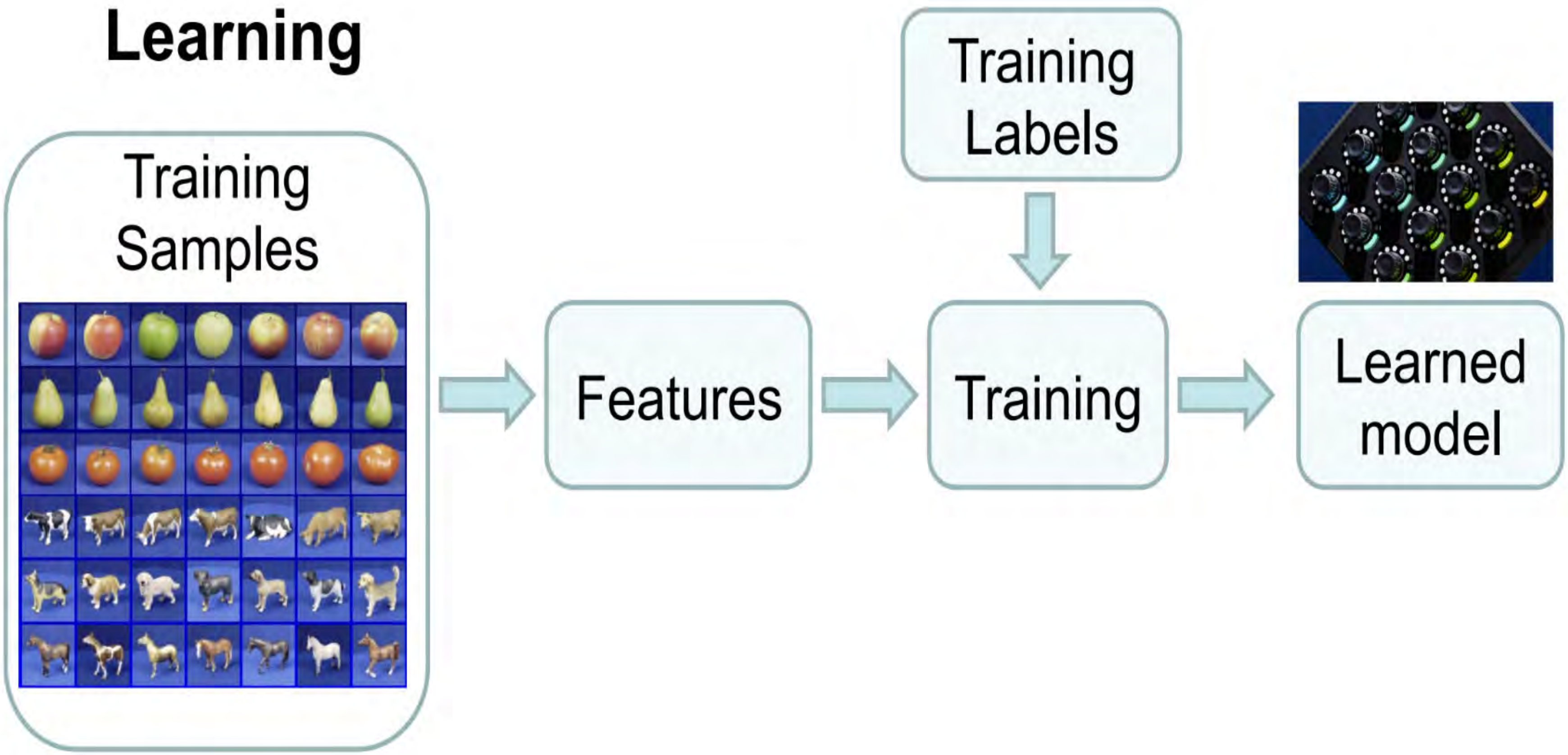
input

output

Classification function

- Learning: Given a set of training data with labelled examples $\{(x_1, y_1).....(x_n,y_n)\}$, it will attempt to build the parameters of the classification function ***f***
- Inference: After the learning, it is to apply ***f*** to a never seen test example ***x*** and output predicted the value $y = f(x)$

Basic Framework of Supervised Learning



Learning

Training Samples

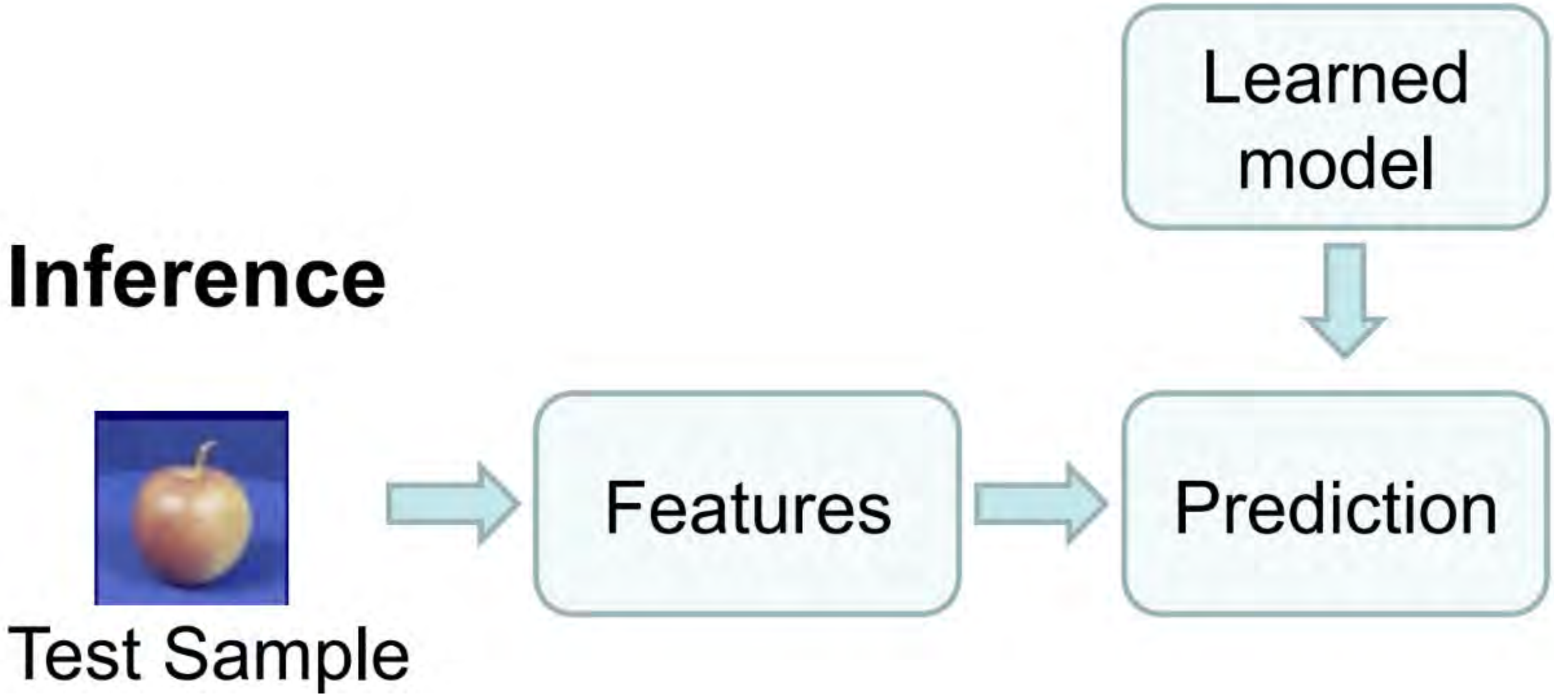
Features

Training Labels

Training

Learned model

Basic Framework of Supervised Learning



Supervised Learning Algorithms

- Some of the Supervised Learning Algorithms
  - **K-Nearest Neighbors**
  - **Linear Regression**
  - Decision Trees
  - Polynomial regression
  - Naïve Bayes
  - Logistic Regression

Choice of Supervised Learning Algorithms

- There is a wide range of Supervised Learning Algorithm, each algorithms has it strength and weakness.
- Four keys consideration for choice of Supervised Learning Algorithm:
  - Bias-variance tradeoff
  - Function complexity and amount of training data
  - Dimensionality of the input space
  - Noise in the output values

Unsupervised Learning

- Unsupervised Learning involves the machine to learn the classifier from unlabeled examples
- Fundamentally, in unsupervised learning it will group the information according to the characteristics (similarities and differences)

Georgia Tech https://youtu.be/1qtfILYSDJY

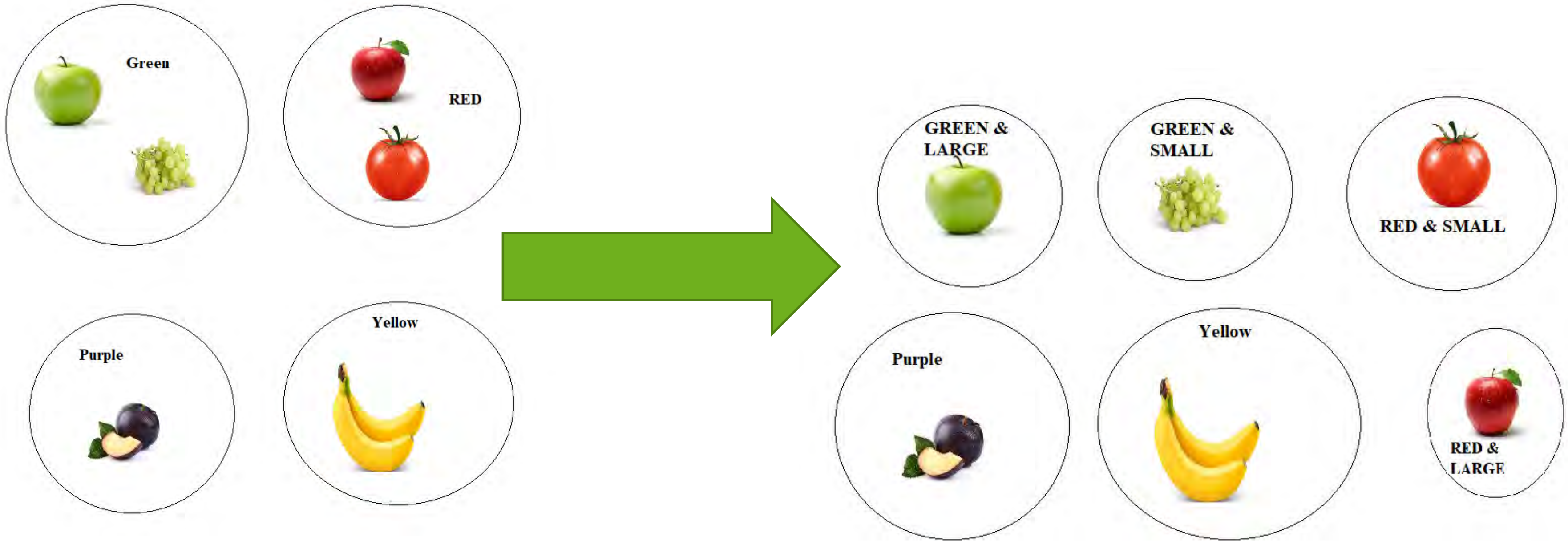
Real life example of Unsupervised Learning

- Imagine you have are given a basket of fruits
- Assuming you have not seen the fruits before
- How you will start to differentiate the different fruits?
  - You can start off with the physical attributes of the fruits
    - E.g. Colour, size

Real life example of Unsupervised Learning

- In this case, you may like to differentiate by colour
  - RED – Tomato, Apple
  - GREEN – Apple, Grape
  - ORANGE – Orange
  - PURPLE – Plum
  - YELLOW – Banana
- You may like to further classification based on size
  - RED, LARGE – Apple
  - RED, SMALL – Tomato
  - GREEN, LARGE – Apple
  - GREEN, SMALL - Grape

Real life example of Unsupervised Learning



In unsupervised learning, it is an attempt to cluster the outcome based on the characteristics.
e.g. Colour, shape, size

Unsupervised Learning Algorithms

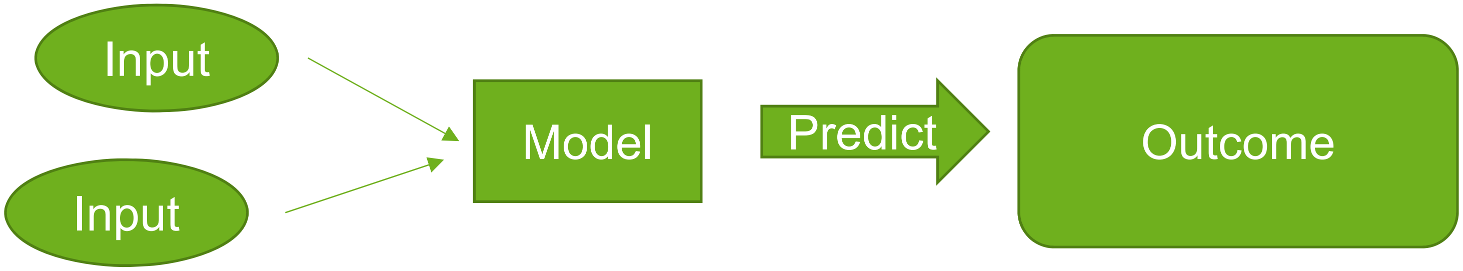
- Some of the algorithms that are used:
  - **k – means clustering**
  - Hierarchical clustering
  - Hidden Markov models

Semi Supervised Learning

- **Semi-supervised learning** is an approach to machine learning that combines a small amount of labeled data with a large amount of unlabeled data during training.
- Semi-supervised learning falls between **unsupervised learning** (with no labeled training data) and **supervised learning** (with only labeled training data).

Pattern Match and Recognition

- It is requires mathematical models to identify the patterns
- Common Mathematical approach is using statistical models to help -> Attempts to make classification to the data and values
- Based on the models created, it will be used to make predication of the possible outcome from the input(s)



Approaches

- Statistical Pattern Matching and Recognition
  - Patterns that are generated by probabilistic algorithm
  - Input data are reduced to vectors of numbers
  - Statistical techniques are used for classification of the input data vectors
- Structural Pattern Matching and Recognition
  - Process based on the structural interrelationship of features
  - Data is converted to discrete structure (e.g. graph)
  - Classification techniques include parsing and matching are used

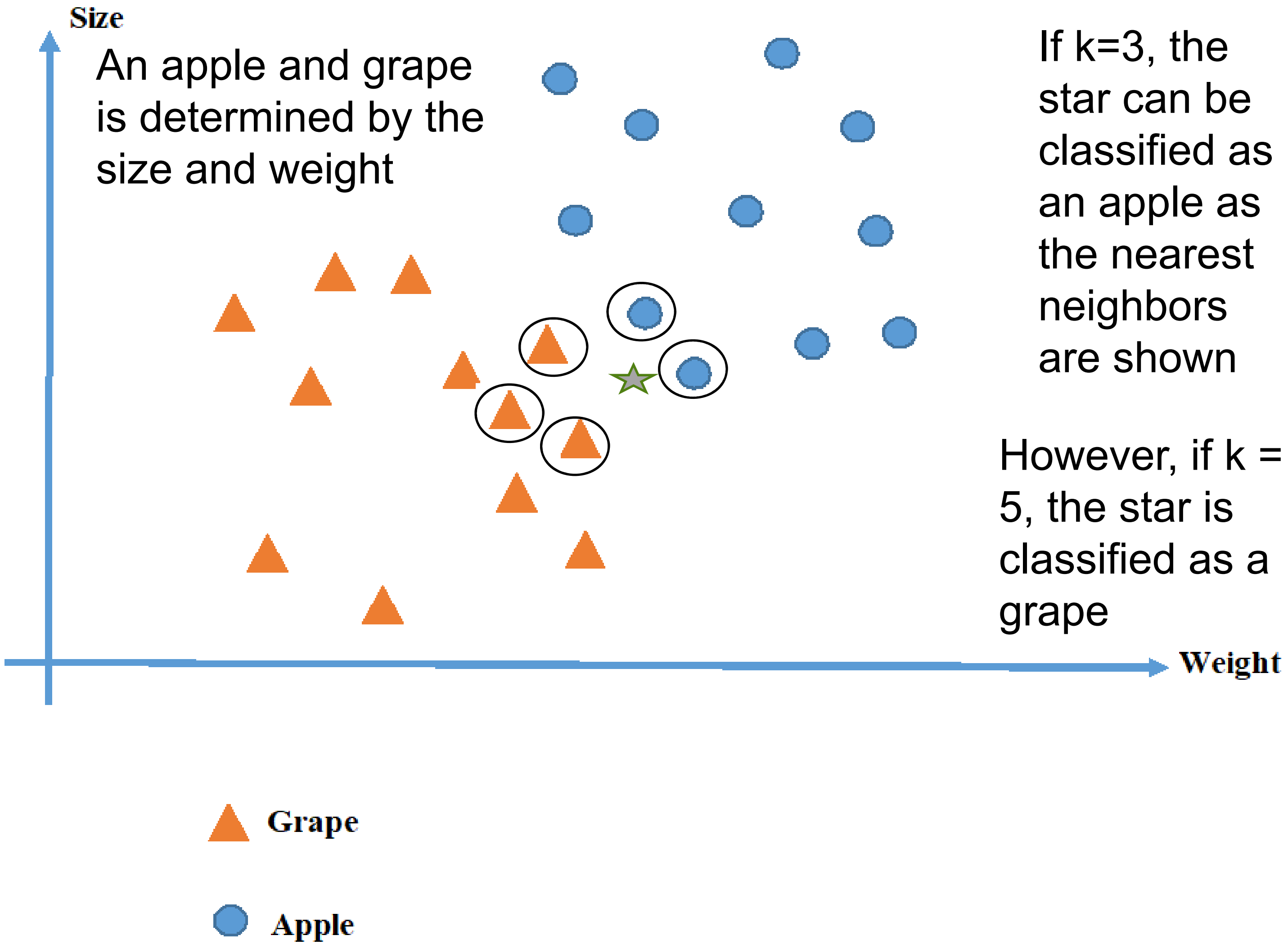
KNN & K-Mean

- K-Nearest Neighbor and K-Mean are algorithms that focus on leveraging the similarities among the examples
- KNN are used for supervised learning and K-Mean for unsupervised learning

KNN in Supervised Learning

- KNN is a type of instance-based/lazy type learning
- It is the simplest classification technique when there is little or no prior knowledge about distribution of the data
- The predicted output are determined by the ‘closest’ of the unknown input sample to the known input samples
- KNN is a robust in situation where there are noisy input samples

Example of KNN



An apple and grape is determined by the size and weight

If k=3, the star can be classified as an apple as the nearest neighbors are shown

However, if k = 5, the star is classified as a grape

Consideration of using KNN

- You need to avoid a tie in the decision, such that k value
    - must be an odd number for a TWO class problem
    - must not be a multiple of the number of classes
- Drawback of KNN is the complexity of searching the nearest neigbhours of each sample

Disadvantage of KNN

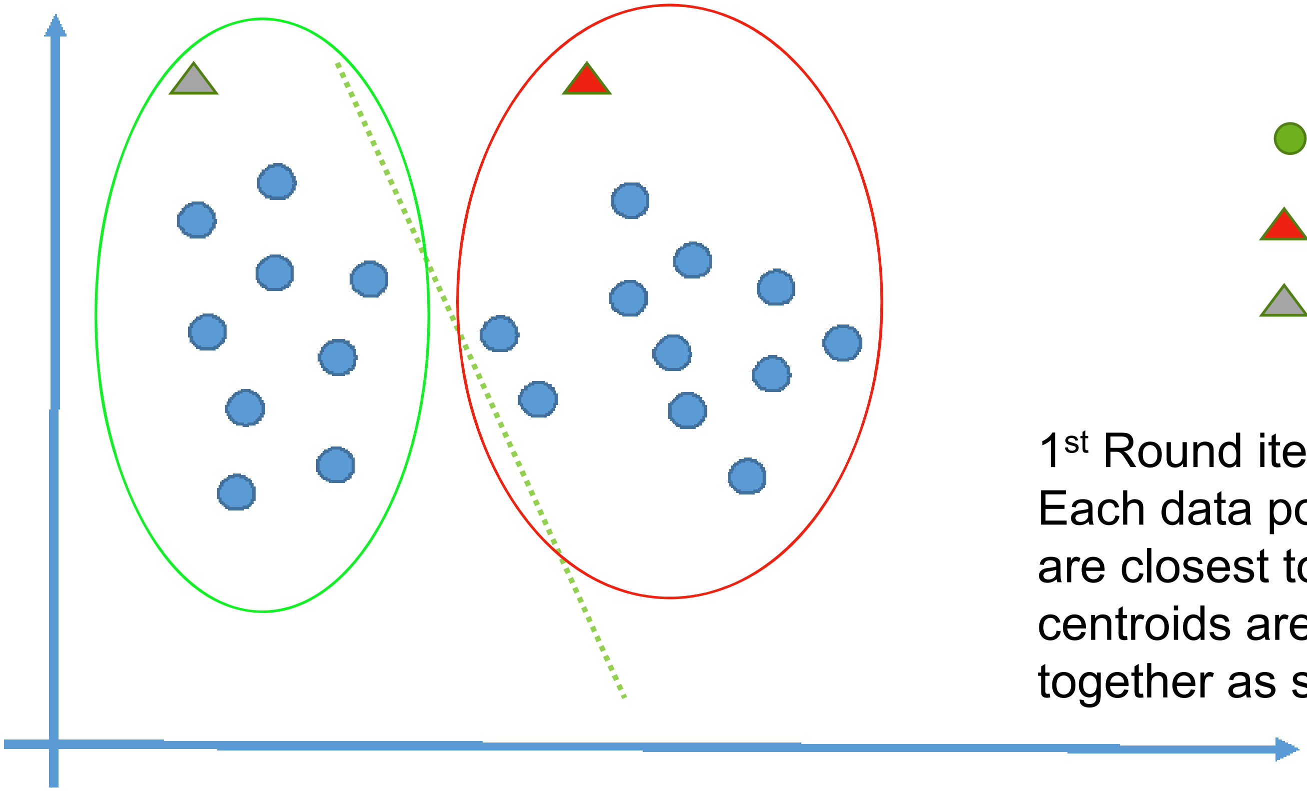
- Need to determine the value of k
- Distance of object become less distinct
- Low computation efficiency
- Data Sparsity
- Large amount of Data required
- False Intuition
- Not clear which type of distance metric to use
- Computation cost is high

K-Means clustering

- Given the input is k and a sets of points $x_1......x_n$
- Place the centroids (k) $c_1...c_k$ at initial points (will discuss later what is the ideal k locations)
- Repeat until convergence
  - Determine each point to the nearest centroid (e.g. using Euclidian distance)*
  - Recalculate the new centroids based on the mean of all points assigned to the centroids
- Stop when there is no change of cluster assignments

- Euclidian distance is straight distance between two points p to q $\sqrt{}(q - p)^2$

K-Means

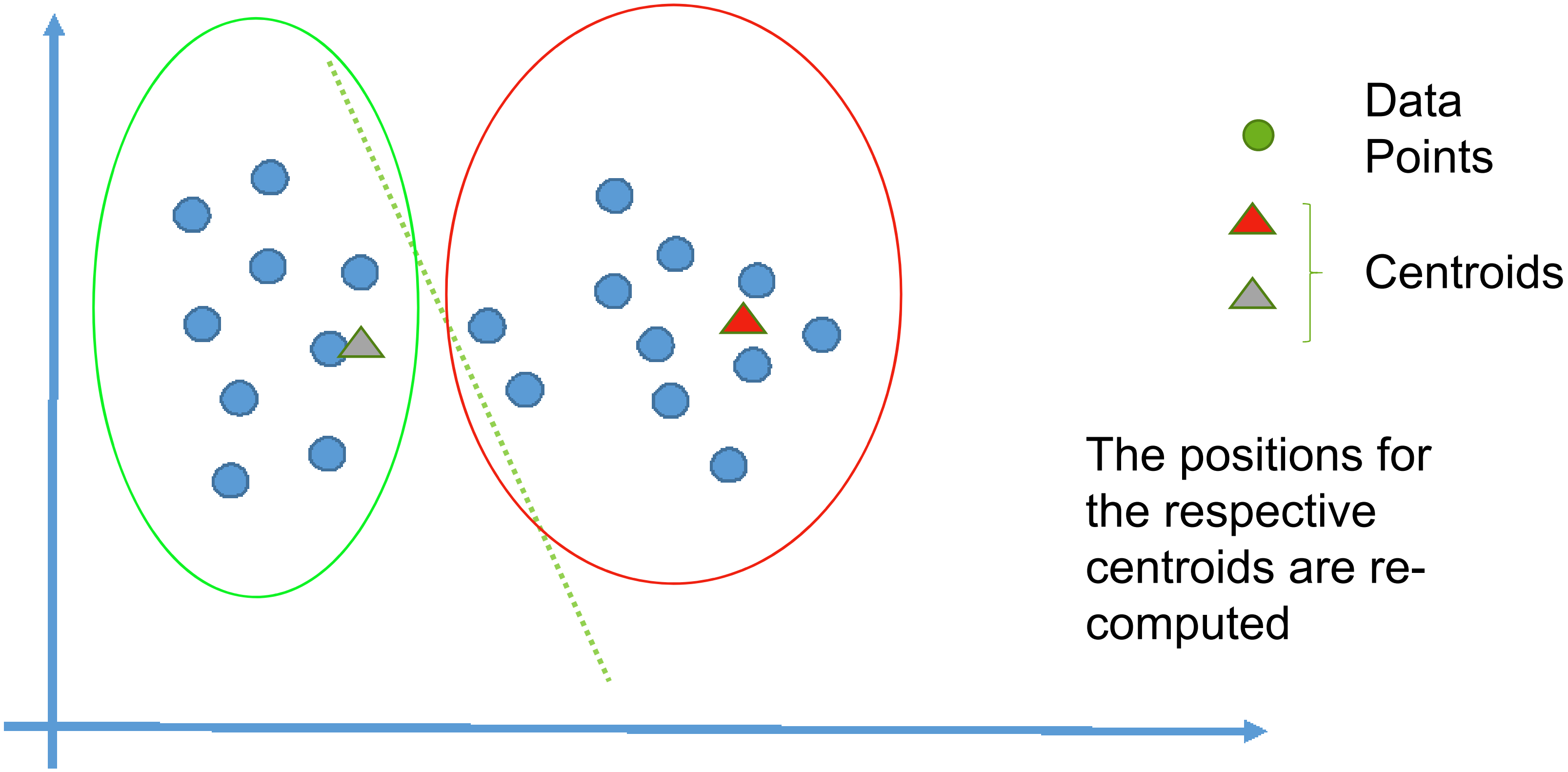


● Data Points

▲ ▲ Centroids

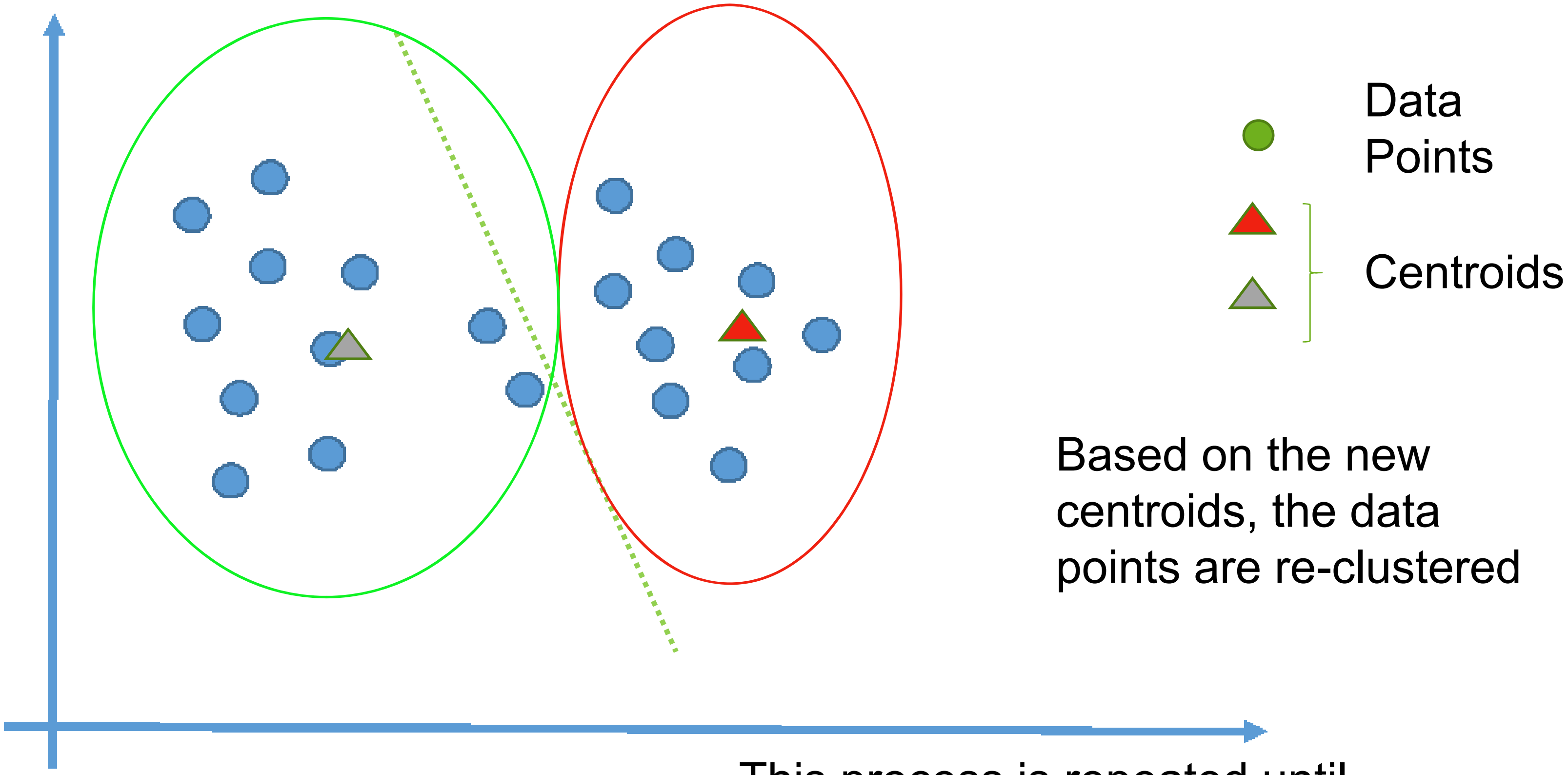
1st Round iteration
Each data points that are closest to the centroids are clustered together as shown

K-Means



The positions for the respective centroids are re-computed

K-Means



Based on the new centroids, the data points are re-clustered

This process is repeated until there are no changes in the data points to the clusters

Picking the value of k

- If too few centroids(k), the distance are too far from the centroids (Figure A)
- If too many centroids(k), there are little improvement in average distances (Figure B)

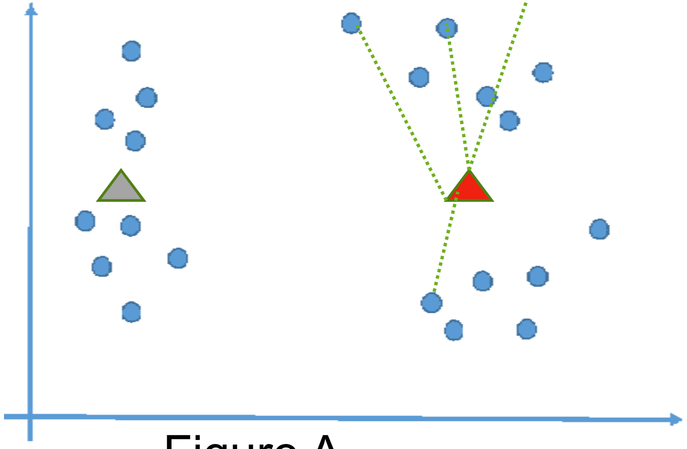


Figure A

What is the correct value of k?

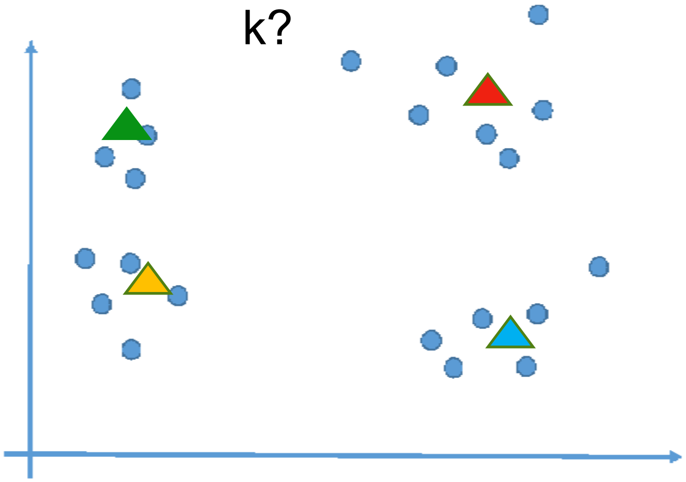
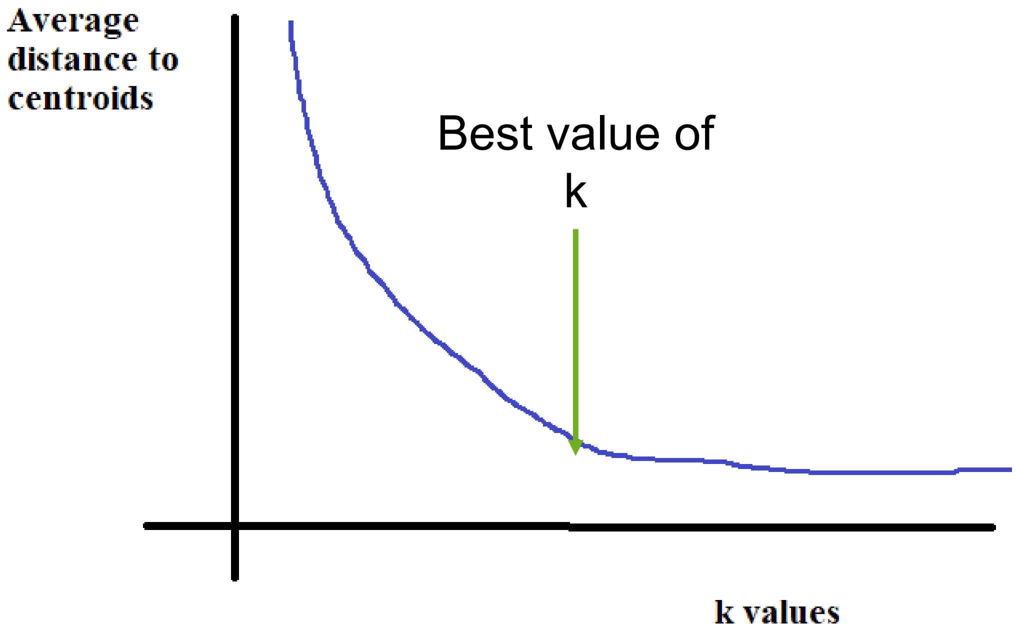


Figure B

Picking the right k value

- Average to centroids fall rapidly until the right values of k, then it falls slowly



Picking the initial k values

- May not be ideal to randomly chose the k values points
- The initial all k points may be in the same cluster
- Or the k-points are in the outlier areas
- In both situations, it may not reflect well on the final clustering
- It is important to chose the right k points

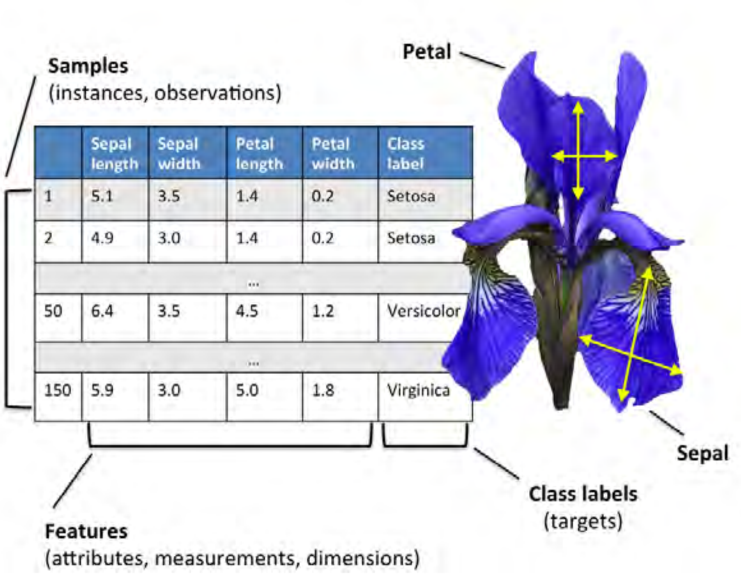
Determine the k-point locations

- Approach 1: Sampling
  - Cluster a sample data using another algorithm to determine the number of clusters (k)
  - Chose a point from each cluster (point closest to the centroid)
- Approach 2: Picked ‘dispersed’ set of points
  - Chose the first one at random
  - Pick the next k-point to be the one whose minimum distance from the selected points is as large as possible
  - Repeat until the required k points

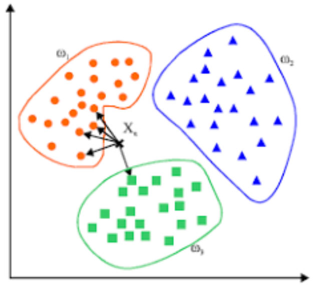
Problem with k-means

- Need to examine each input points that is closest to the centroids
- For example, there are ***N*** points for ***k*** clusters, hence each round will be O(***kN***)
- The number of rounds for each convergence can be large.

Activity 1 - KNN & K-Means in Action


Samples
(instances, observations)
Petal
Sepal
Class labels
(targets)
Features
(attributes, measurements, dimensions)

	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
			...		
50	6.4	3.5	4.5	1.2	Versicolor
			...		
150	5.9	3.0	5.0	1.8	Virginica



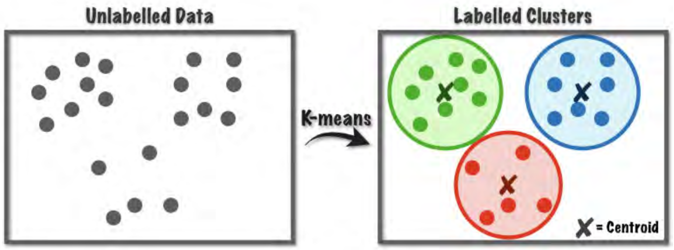


Image credits: Principal Component Analysis by Sebastian Raschka

Step 1:
Watch and listen to the instructor's demonstration

10 mins

Step 2:
Work through the activities

Individual Activity

20 mins

https://www.kaggle.com/nautna/iris-knn-python-classification

15 Mins Break

bit.ly/top10_2020

Machine Learning in Python

- Python is the lingua franca for data science application
- Able to combines the programming language with ease of use of scripting language like MATLAB or R
- Python has libraries for:
  - Data loading
  - Visualization
  - Natural Language Processing
  - Image Processing
  - Special Purpose functionality

NumPy

pandas

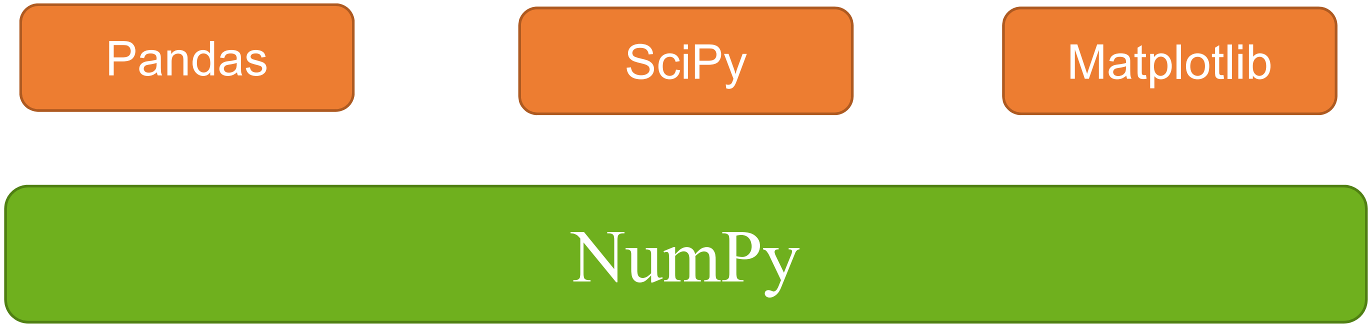
scikit learn

matplotlib

SciPy

Relationship between NumPy and Pandas

- Pandas, SciPy and Matplotlib are high level manipulation tools built on top of NumPy
- NumPy is a low level manipulation tool



Pandas

SciPy

Matplotlib

NumPy

What is NumPy?

- NumPy is the fundamental packages for scientific computing
- Numpy is a Linear Algebra Library for Python
- It is very important to Data Science as almost all of the libraries in the PyData ecosystem reply on NumPy as one of the key development block
- NumPy is very fast as it has binding to C Libraries

What is included in NumPy?

- Include functionality for
  - Multidimensional Arrays
  - High Level Mathematical functions
  - Linear Algebra operations
  - Fourier transform
  - Pseudorandom number generator

NumPy

- N-dimensional array (ndarray) is an important object define in NumPy
- It is a collections of items of the same type and the items in the collection can be accessed by zero-based index
- Each item in an ndarray has the same size memory block
- Each item in an ndarray is an object of Data-Type object (dtype)

Python List Vs NumPy Array

- The advantage of NumPy (compared to Python List) are as follow:
  - Require Less Memory
  - Processing is faster
  - Require less coding

Usage of NumPy

- NumPy arrays are the main way of storing data
- It come in Vectors and Matrices
- Vectors are 1-D arrays and matrices are 2-D matrices

Example of Vectors

$$[1, \quad 2, \quad 3]$$

Example of Matrices

$$\begin{bmatrix} 3, & 4 \\ 5, & 6 \end{bmatrix}$$

Activity 2 - Numpy

- Activity - Numpy

Step 1:
Watch and listen to the instructor’s demonstration

10 mins

Step 2:
Work through the activities

Individual Activity

30 mins

LUNCH BREAK

60 mins Lunch Break

Some interesting videos

https://www.youtube.com/watch?v=bmNaLtC6vkU

https://www.youtube.com/watch?v=Nnf8P5A_saE

Lunch break XX:XX-YY:YY

What is the role of Pandas in Machine Learning?

- A set of Python libraries for data wrangling and analysis
- Useful for early stages of data inspection, preprocessing and data cleaning
- It can work with data from a variety of sources
- It has excellent performance and built-in visualization features

Save the Panda

Pandas and NumPy

- As mentioned Pandas is built on top of NumPy, there are many features that are similar to NumPy
- Likewise, dtype defines the data type that are used in the various Pandas' various Data structure

This is the reason why NumPy is emphasized first

Pandas Data Structure and Dimension

- Pandas has the following data structure
  - Series
  - Data Frames
  - Panel
- The data structure are built on NumPy
- Pandas reduces the complexities of handling two or more dimensional arrays

Overview of Pandas Data Structure and Dimensional

Data Structure	Dimension	Description
Series	1	1D labelled homogeneous array, size immutable. Data is mutable
Data Frames	2	General 2D labelled, data and size mutable tabular structure with potentially heterogeneously typed columns.
Panels	3	General 3D labelled, data and size mutable array.

- All Data Structure except Series are mutable (can be changed)
- Data Frame is the most common Data Structure used

Key Point of Series Data Structure

- One Dimensional labelled array
- Data need to be homogenous
- The size is immutable
- Values of the data can be mutable

Activity 3 - Pandas

pandas

Step 1:
Watch and listen to the instructor's demonstration

15 mins

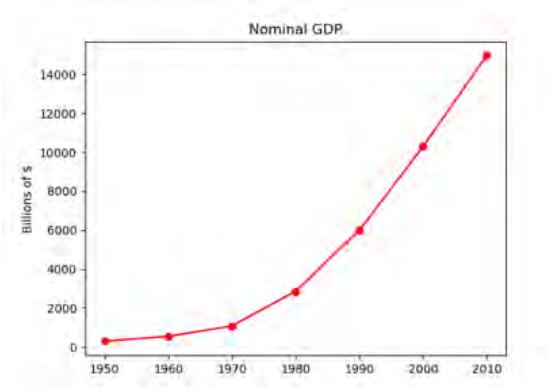
Step 2:
Work through the activities

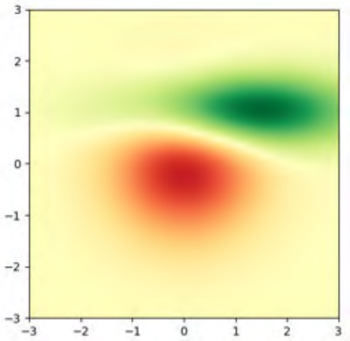
Individual Activity

30 mins

Matplotlib

- In Artificial Intelligence as well as Data Science, very often, data needs to be visualised in order to draw insights from the datasets.
- Basic examples (http://matplotlib.org/examples/)





- sudo apt-get install libpng12-dev
- sudo apt-get install python-dev
- sudo pip install matplotlib

Activity 4 - Matplotlib

Step 1:
Watch and listen to the instructor's demonstration

10 mins

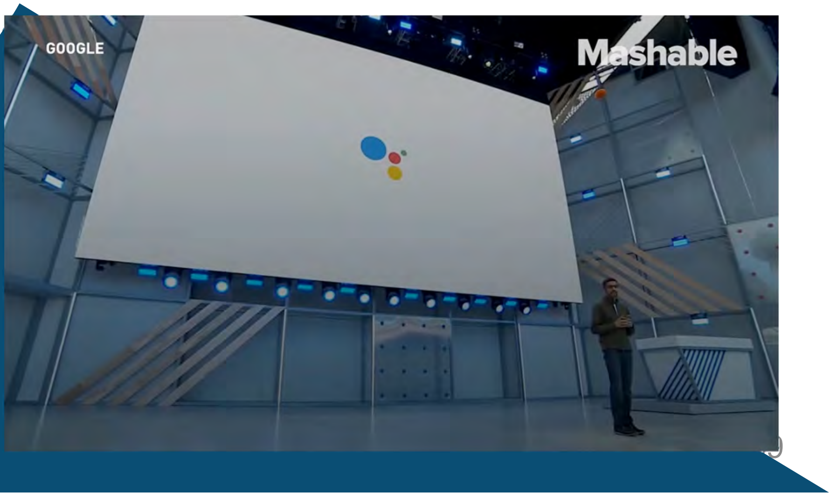
Step 2:
- Do on your own

Optional Activity

30 mins

15 Mins Break

bit.ly/google_duplex2019



Scikit-learn

- Scikit-learn is a library for doing machine learning in Python.
- Any machine learning problem is based on three concepts:
  - Define a task, T, to solve.
  - Need some experience, E, to learn to perform the task.
  - Need to measure performance, P, to know how well we are solving the task.

- pip install scikit-learn
- sudo pip install urllib3[secure]
- sudo pip install scipy

Datasets

- We will use a well-known datasets called the Iris flower dataset. (https://en.wikipedia.org/wiki/Iris_flower_data_set.)

Iris setosa

Iris versicolor

Iris virginica

Threshold Metrics

- Precision-Recall Metrics
  - Precision summarizes the fraction of examples assigned the positive class that belong to the positive class.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

  - Recall summarizes how well the positive class was predicted and is the same calculation as sensitivity

$$Recall \ = \frac{TruePositive}{TruePositive + FalseNegative}$$

Threshold Metrics

- F-Measure
  - F-measure provides a way to combine both precision and recall into a single measure that captures both properties..

$$F_{measure} = \frac{(2)\ \mathrm{x}\ Precision\ \mathrm{x}\ Recall}{Precision + \ Recall}$$

  - sometimes called the F-score or the F1-measure
  - might be the most common metric used on imbalanced classification problems.

Activity 5 – scikit-learn

Step 1:
Watch and listen to the instructor's demonstration

10 mins

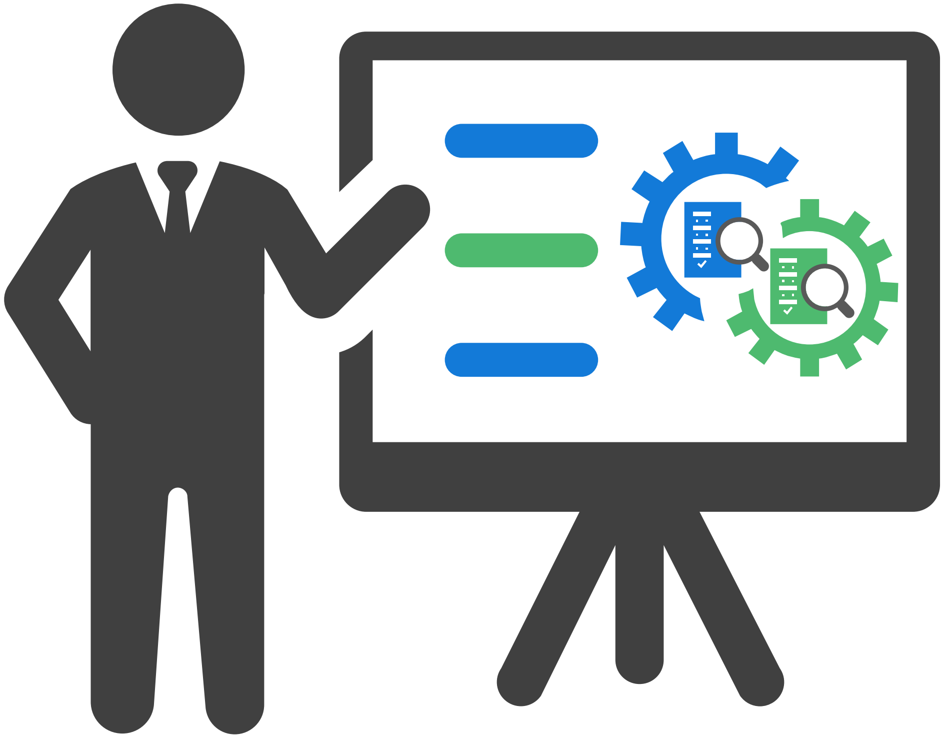
Step 2:
- Do on your own

Individual Activity

30 mins

Quiz

https://bit.ly/kw_poll



Summary

- Recap ML
- Numpy
- Pandas
- Matplotlib
- Scikit Learn

Email
seow_khee_wei@rp.edu.sg

Telegram
@kwseow

Source code:

Thank you