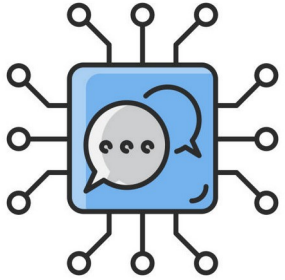
Statistical Approaches

Tushar B. Kute,
http://tusharkute.com



Probability

- Probability means possibility. It is a branch of mathematics that deals with the occurrence of a random event.
- The value is expressed from zero to one. Probability has been introduced in Maths to predict how likely events are to happen.
- The meaning of probability is basically the extent to which something is likely to happen. This is the basic probability theory, which is also used in the probability distribution, where you will learn the possibility of outcomes for a random experiment.
- To find the probability of a single event to occur, first, we should know the total number of possible outcomes.

Probability: Definition in Maths

- Probability is a measure of the likelihood of an event to occur. Many events cannot be predicted with total certainty.
- We can predict only the chance of an event to occur i.e., how likely they are going to happen, using it. Probability can range from 0 to 1, where 0 means the event to be an impossible one and 1 indicates a certain event.
- Probability for Class 10 is an important topic for the students which explains all the basic concepts of this topic.
- The probability of all the events in a sample space adds up to 1.

Probability: Definition in Maths

- For example, when we toss a coin, either we get Head OR Tail, only two possible outcomes are possible (H, T).
- But when two coins are tossed then there will be four possible outcomes, i.e {(H, H), (H, T), (T, H), (T, T)}.

Probability: Definition in Maths

- The probability formula is defined as the possibility of an event to happen is equal to the ratio of the number of favourable outcomes and the total number of outcomes.

  Probability of event to happen P(E) =

  Number of favourable outcomes/Total Number of outcomes

- Sometimes students get mistaken for “favourable outcome” with “desirable outcome”.
- This is the basic formula. But there are some more formulas for different situations or events.

Probability: Example

- 1) There are 6 pillows in a bed, 3 are red, 2 are yellow and 1 is blue. What is the probability of picking a yellow pillow?
  - Ans: The probability is equal to the number of yellow pillows in the bed divided by the total number of pillows, i.e. 2/6 = 1/3.

Probability: Example

- 2) There is a container full of coloured bottles, red, blue, green and orange. Some of the bottles are picked out and displaced. Sumit did this 1000 times and got the following results:
  - No. of blue bottles picked out: 300
  - No. of red bottles: 200
  - No. of green bottles: 450
  - No. of orange bottles: 50

Probability: Example Continued.

- a) What is the probability that Sumit will pick a green bottle?

  Ans: For every 1000 bottles picked out, 450 are green.

  Therefore, P(green) = 450/1000 = 0.45

- b) If there are 100 bottles in the container, how many of them are likely to be green?

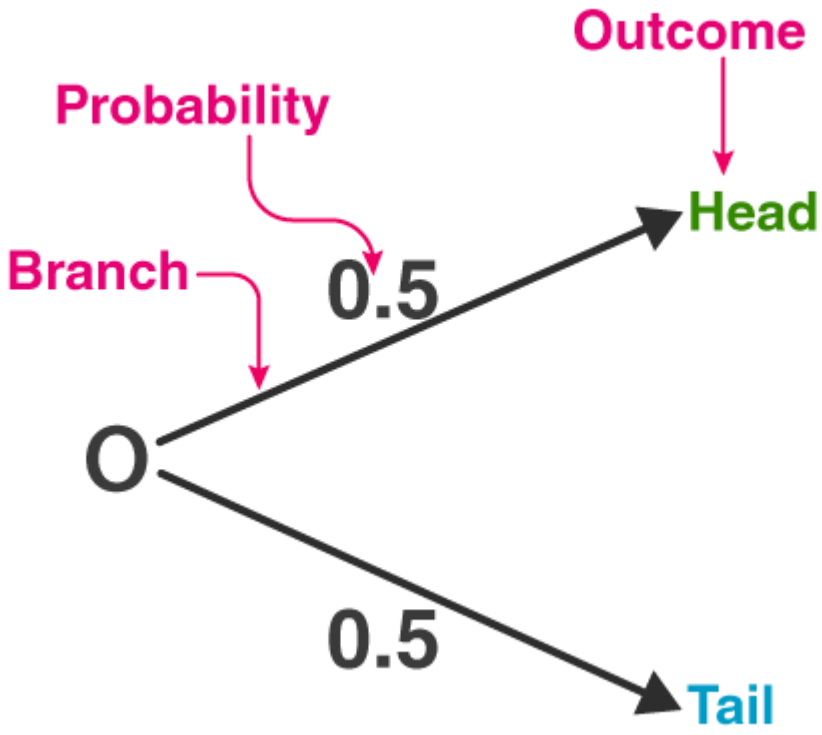
  Ans: The experiment implies that 450 out of 1000 bottles are green.

  Therefore, out of 100 bottles, 45 are green.

Probability Tree

- The tree diagram helps to organize and visualize the different possible outcomes. Branches and ends of the tree are two main positions.
- Probability of each branch is written on the branch, whereas the ends are containing the final outcome.
- Tree diagrams are used to figure out when to multiply and when to add.

Probability Tree



Probability Types

- There are three major types of probabilities:
  - Theoretical Probability
  - Experimental Probability
  - Axiomatic Probability

Theoretical Probability

- It is based on the possible chances of something to happen.
- The theoretical probability is mainly based on the reasoning behind probability.
- For example, if a coin is tossed, the theoretical probability of getting a head will be ½.

Experimental Probability

- It is based on the basis of the observations of an experiment.
- The experimental probability can be calculated based on the number of possible outcomes by the total number of trials.
- For example, if a coin is tossed 10 times and head is recorded 6 times then, the experimental probability for heads is 6/10 or, 3/5.

Axiomatic Probability

- In axiomatic probability, a set of rules or axioms are set which applies to all types.
- These axioms are set by Kolmogorov and are known as Kolmogorov's three axioms. With the axiomatic approach to probability, the chances of occurrence or non-occurrence of the events can be quantified.
- The axiomatic probability lesson covers this concept in detail with Kolmogorov's three rules (axioms) along with various examples.

Conditional Probability

- Conditional Probability is the likelihood of an event or outcome occurring based on the occurrence of a previous event or outcome.

Probability of an event

- Assume an event E can occur in r ways out of a sum of n probable or possible equally likely ways. Then the probability of happening of the event or its success is expressed as;

$$P(E) = r/n$$

- The probability that the event will not occur or known as its failure is expressed as:

$$P(E') = (n-r)/n = 1-(r/n)$$

- E' represents that the event will not occur.
- Therefore, now we can say;

$$P(E) + P(E') = 1$$

- This means that the total of all the probabilities in any random test or experiment is equal to 1.

What are Equally Likely Events?

- When the events have the same theoretical probability of happening, then they are called equally likely events. The results of a sample space are called equally likely if all of them have the same probability of occurring.
- For example, if you throw a die, then the probability of getting 1 is 1/6. Similarly, the probability of getting all the numbers from 2,3,4,5 and 6, one at a time is 1/6. Hence, the following are some examples of equally likely events when throwing a die:
  - Getting 3 and 5 on throwing a die
  - Getting an even number and an odd number on a die
  - Getting 1, 2 or 3 on rolling a die
- are equally likely events, since the probabilities of each event are equal.

Complementary Events

- The possibility that there will be only two outcomes which states that an event will occur or not.
- Like a person will come or not come to your house, getting a job or not getting a job, etc. are examples of complementary events.
- Basically, the complement of an event occurring in the exact opposite that the probability of it is not occurring. Some more examples are:
  - It will rain or not rain today
  - The student will pass the exam or not pass.
  - You win the lottery or you don’t.

Probability Theory

- Probability theory had its root in the 16th century when J. Cardan, an Italian mathematician and physician, addressed the first work on the topic,
- The Book on Games of Chance. After its inception, the knowledge of probability has brought to the attention of great mathematicians.
- Thus, Probability theory is the branch of mathematics that deals with the possibility of the happening of events.

Probability Theory

- Although there are many distinct probability interpretations, probability theory interprets the concept precisely by expressing it through a set of axioms or hypotheses.
- These hypotheses help form the probability in terms of a possibility space, which allows a measure holding values between 0 and 1.
- This is known as the probability measure, to a set of possible outcomes of the sample space.

Probability Density Function

- The Probability Density Function (PDF) is the probability function which is represented for the density of a continuous random variable lying between a certain range of values.
- Probability Density Function explains the normal distribution and how mean and deviation exists.
- The standard normal distribution is used to create a database or statistics, which are often used in science to represent the real-valued variables, whose distribution is not known.

Probability Terms and Definitions

Term	Definition	Example
Sample Space	The set of all the possible outcomes to occur in any trial	1. Tossing a coin, Sample Space (S) = {H,T} 2. Rolling a die, Sample Space (S) = {1,2,3,4,5,6}
Sample Point	It is one of the possible results	In a deck of Cards: • 4 of hearts is a sample point. • The queen of clubs is a sample point.

Probability Terms and Definitions

Experiment or Trial	A series of actions where the outcomes are always uncertain.	The tossing of a coin, Selecting a card from a deck of cards, throwing a dice.
Event	It is a single outcome of an experiment.	Getting a Heads while tossing a coin is an event.
Outcome	Possible result of a trial/experiment	T (tail) is a possible outcome when a coin is tossed.
Complimentary event	The non-happening events. The complement of an event A is the event, not A (or A')	In a standard 52-card deck, A = Draw a heart, then A' = Don't draw a heart
Impossible Event	The event cannot happen	In tossing a coin, impossible to get both head and tail at the same time

Probability: Applications

- Probability has a wide variety of applications in real life. Some of the common applications which we see in our everyday life while checking the results of the following events:
  - Choosing a card from the deck of cards
  - Flipping a coin
  - Throwing a dice in the air
  - Pulling a red ball out of a bucket of red and white balls
  - Winning a lucky draw

Probability: Applications

- Other Major Applications of Probability
  - It is used for risk assessment and modelling in various industries
  - Weather forecasting or prediction of weather changes
  - Probability of a team winning in a sport based on players and strength of team
  - In the share market, chances of getting the hike of share prices

Probability: Sample Problem

- Question 1: Find the probability of ‘getting 3 on rolling a die’.
- Solution:

  Sample Space = S = {1, 2, 3, 4, 5, 6}

  Total number of outcomes = n(S) = 6

  Let A be the event of getting 3.

  Number of favourable outcomes = n(A) = 1

  i.e. A = {3}

  Probability, P(A) = n(A)/n(S) = 1/6

  Hence, P(getting 3 on rolling a die) = 1/6

Probability: Sample Problem

- Question 2: Draw a random card from a pack of cards. What is the probability that the card drawn is a face card?
- Solution:

  A standard deck has 52 cards.

  Total number of outcomes = n(S) = 52

  Let E be the event of drawing a face card.

  Number of favourable events = n(E) = 4 x 3 = 12 (considered Jack, Queen and King only)

  Probability, P = Number of Favourable Outcomes/Total Number of Outcomes

  P(E) = n(E)/n(S)

  = 12/52

  = 3/13

  P(the card drawn is a face card) = 3/13

Probability Model

- Describing randomness
- Building a probability model involves a few simple steps.
- First, you identify the random variables of interest in your system. A random variable is just a numerical summary of an uncertain outcome.
  - In our airline example, we could have any possible combination of passengers fail to show up (seat 2C, 14G, etc). But at the end of the day, if we want to know whether any passengers are likely to get bumped to the next flight, all we care about is how many ticketed passengers are no-shows, not their specific identities or seat numbers. So that's our numerical summary, i.e. our random variable: X = the number of no-shows.

Probability Model

- Second, you identify the set of possible outcomes for your random variable, which we refer to as the sample space.
- In our airline example, the sample space is the set of whole numbers between 0 and 140 (the maximum number of no-shows possible, because that's how many tickets were sold).

Probability Model

- Finally, you provide a probability distribution, which is a rule for calculating probabilities associated with each possible outcome in the sample space.
- In the airline example, this distribution might be described using a simple lookup table based on historical data, e.g. 1% of all flights have 1 no-show, 1.2% have 2 no-shows, 1.7% have 3 no-shows, and so forth.
- In building a probability model, this final step is usually where the action is, and it's what we'll discuss extensively in this lesson.

Probability Model

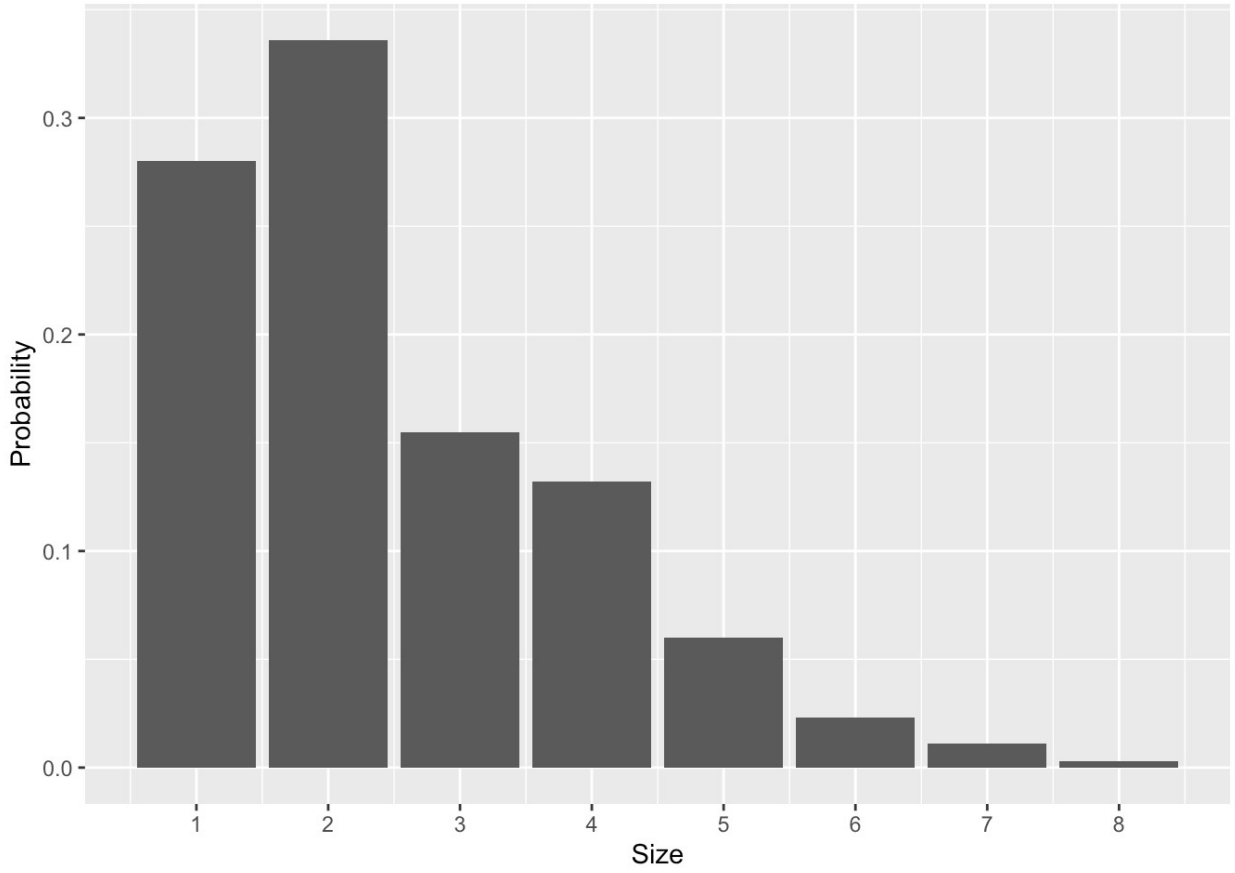
- There are two common types of random variables, corresponding to two common types of outcomes.
  - Discrete:
    - the sample space consists of whole numbers (0, 1, 2, 3, etc.).71 Both the number of airline no-shows and the score of a soccer game are discrete random variables: you can’t have 2.4 no-shows or 3.7 goals.
  - Continuous:
    - the random variable could be anything within a continuous range of numbers, like the price of Apple stock tomorrow, or the volume of a subsurface oil reservoir.

Probability Model : Example

- The table below shows a probability distribution for X, taken from U.S. census data in 2015, in the form of a simple look-up table:

Size	1	2	3	4	5	6	7	8
Probability	0.28	0.336	0.155	0.132	0.06	0.023	0.011	0.003

Probability Model : Example



Probability Model

- This probability distribution provides a complete representation of your uncertainty in this situation. It has all the key features of any probability distribution:
  - There is a random variable, or a numerical summary of an uncertain situation—here, the size of the household next door (X).
- There is a set of possible outcomes for the random variable —here, the numbers 1 through 8. (There's a tiny probability of 9 or more members of the household, which we've truncated to 0.)
- Finally, there are probabilities for each possible outcome—here provided via a simple look-up table or a bar graph.

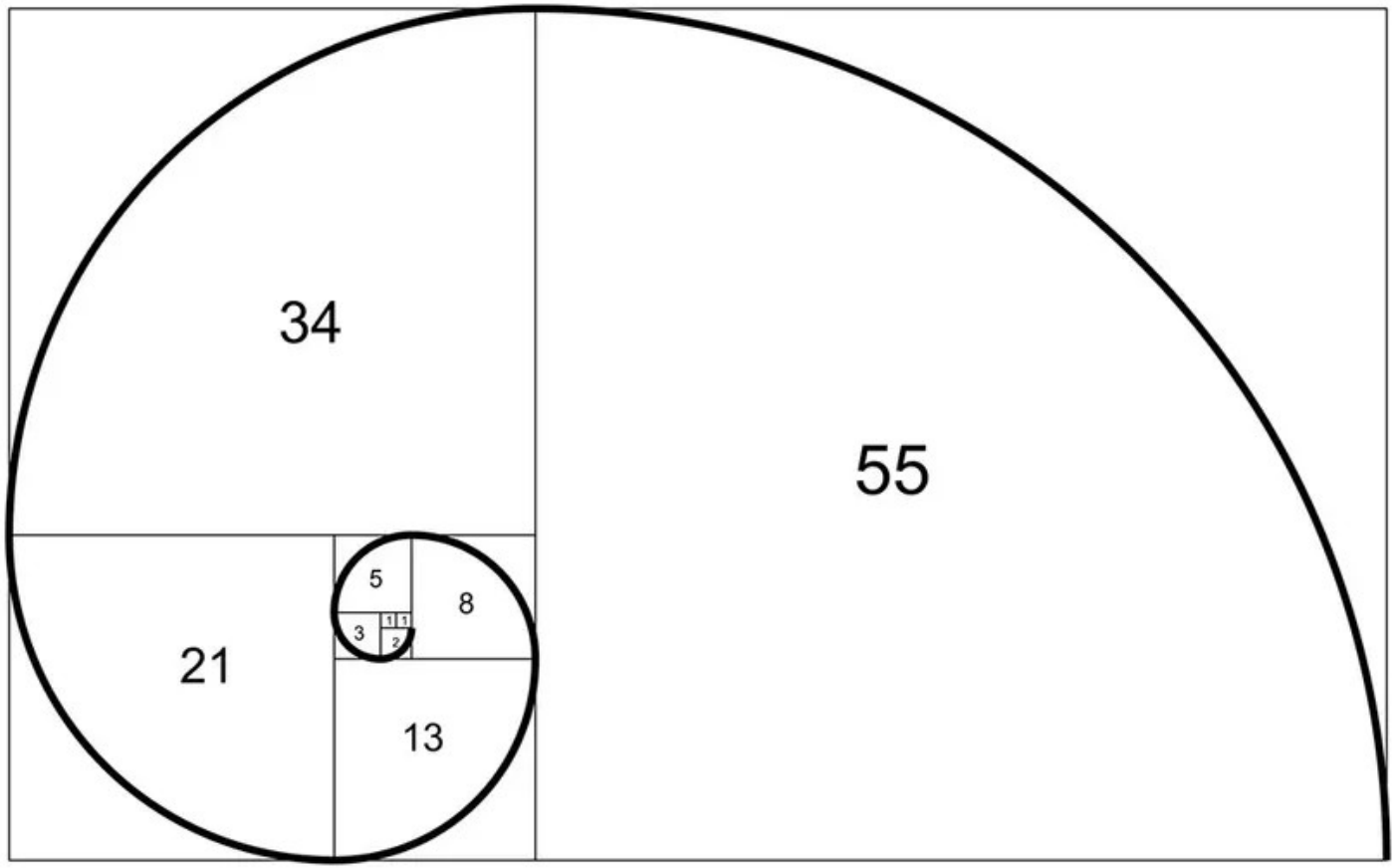
Discrete-time models

- Conceptually, discrete-time models in one variable are the simplest type of dynamical model, also because understanding them does not require knowledge of any advanced mathematical techniques.

Sequences and recursions

- A good starting point for thinking about discrete-time models are sequences of numbers that follow certain rules.
- Consider the following examples:
  - 1 -2 4 -8 16 -32
  - 4 8 32 512 131072
  - 4 5 7 10 14 19
  - 2 4 8 7 5 1

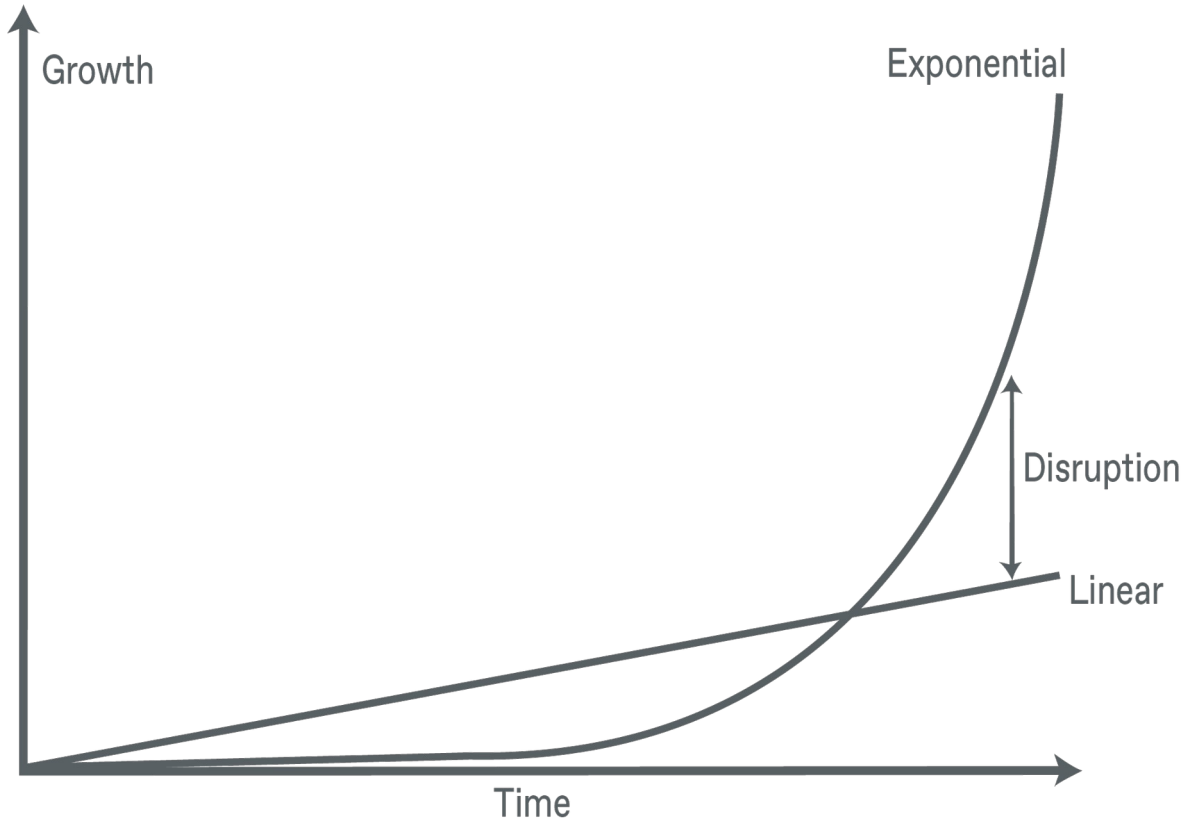
Fibonacci Growth



Exponential Growth

- Consider a population of asexually reproducing individuals. In each generation, each individual produces on average a offspring individuals, following which the entire parental generation dies.
- This assumption is often referred to as “non-overlapping generations” and is met in many plant and animal species. The population is completely homogeneous (no sexes, no age classes, no geographic structure).
- If we denote by Nt the number of individuals in generation t, we can write the recursion equation for this model as $N_{t+1}=aN_t$

Exponential Growth



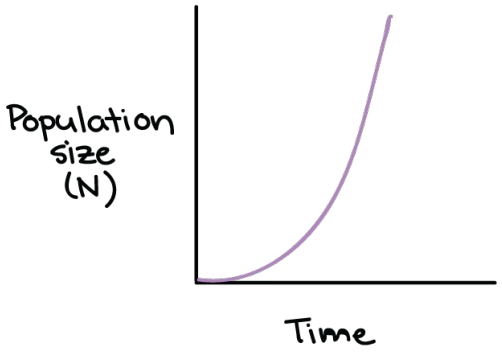
Logistic Growth

$$\frac{dN}{dt} = rN$$

Exponential growth

Per capita growth rate (r) doesn't change, even if pop. gets very large.

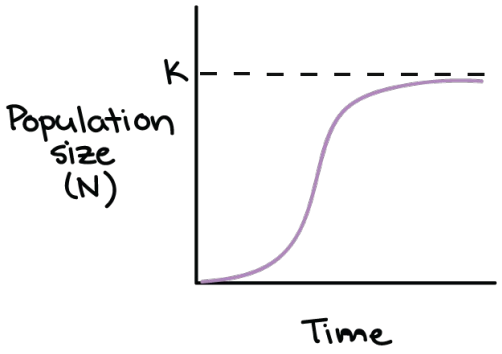
$$\frac{dN}{dt} = r_{max}N$$



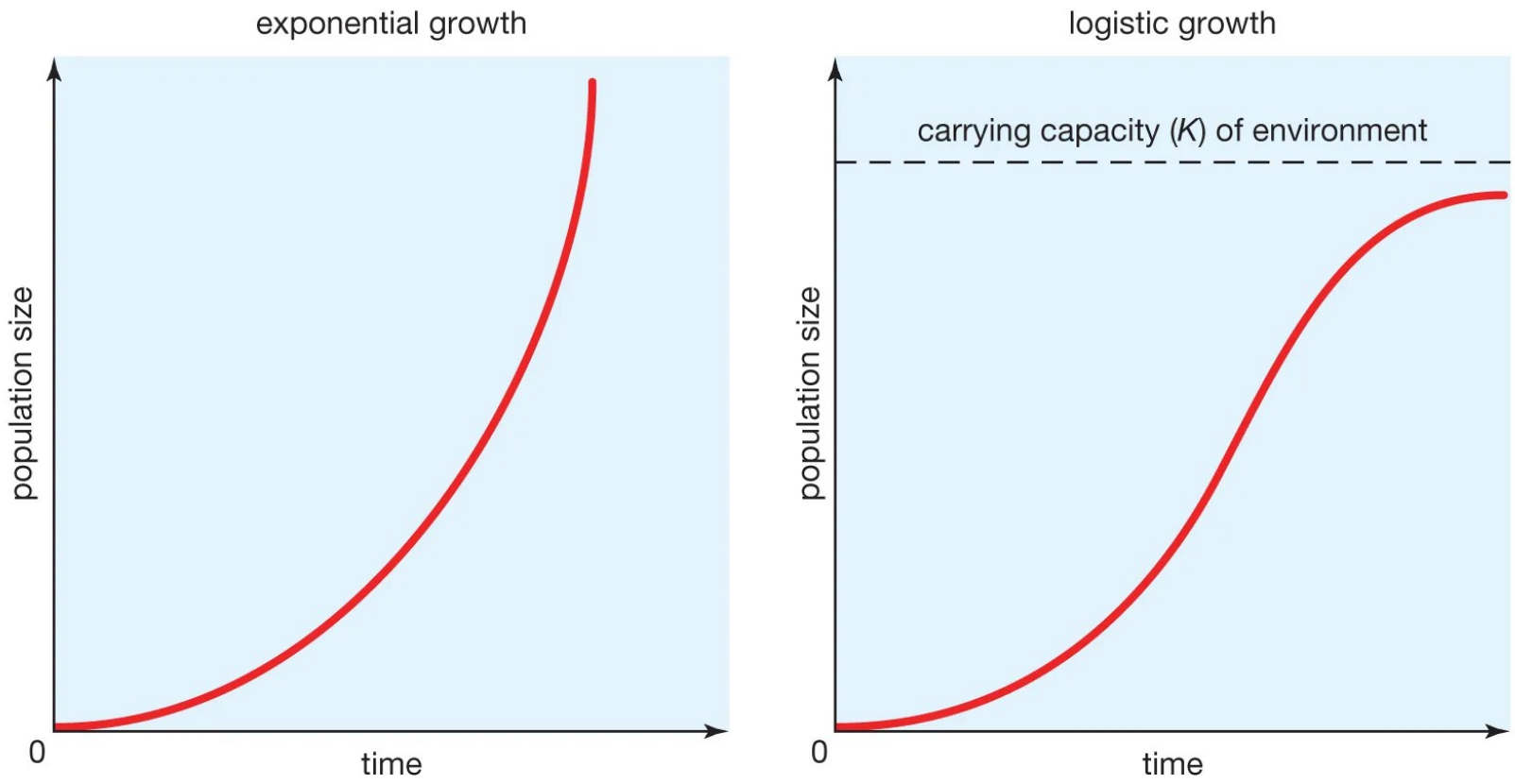
Logistic growth

Per capita growth rate (r) gets smaller as pop. approaches its max. size.

$$\frac{dN}{dt} = r_{max}\left(\frac{K-N}{K}\right)N$$



Logistic Growth



Markov Model

- A Markov model, also known as a Markov chain or Markov process, is a stochastic model used to represent and analyze systems that change over time in a probabilistic manner.
- Here are some key characteristics of Markov models:
  - Memorylessness: The future state of the system depends only on its present state, not on any of the previous states.
  - Discrete: The state space and the time are discrete.
  - Homogeneous: The transition probabilities between states are constant over time.

Markov Model

- Markov models are used in various applications, including:
  - Modeling random phenomena like stock prices and weather patterns.
  - Predicting future behavior in systems like queuing networks and traffic flow.
  - Generating text, music, and other creative content.
  - Implementing natural language processing tasks like speech recognition and machine translation.

Markov Model

- Here are some different types of Markov models:
  - Hidden Markov Models (HMMs): These models include hidden states that cannot be directly observed but influence the observed states.
  - Markov Decision Processes (MDPs): These models allow for decision-making within the system, where the transition probabilities depend on the chosen actions.

Markov Model

- Benefits of using Markov models:
  - Simple and easy to understand.
  - Efficient to compute.
  - Widely used and highly adaptable to various applications.
- Limitations of Markov models:
  - Assumes memorylessness, which may not be true for all real-world systems.
  - Limited to discrete state spaces and time steps.

Markov Model Process

- 1. Define the States:
  - Identify the possible states the system can be in.
  - These states can be discrete categories, values, or even hidden states not directly observable.
- 2. Calculate Transition Probabilities:
  - Determine the probability of transitioning from one state to another.
  - These probabilities are represented by a transition matrix, where each row represents the probability of transitioning from a specific state to all other states.

Markov Model Process

- 3. Define the Initial State:
  - Specify the starting state of the system at time zero.
  - This initial state distribution can be based on known information or assumed probabilities.
- 4. Analyze Future States:
  - Using the transition matrix and initial state, calculate the probability of the system being in different states at future time steps.
  - This can be done through matrix multiplication or recursive calculations.

Markov Model Process

- 5. Apply the Model:
  - Utilize the calculated probabilities to predict future behavior, make decisions, or analyze the overall system dynamics.
  - This can involve simulating the system over time, optimizing actions, or drawing inferences from observed data.

Markov Model Example

- Imagine a simple weather model with two states: sunny and rainy.
  - States: Sunny, Rainy
  - Transition Matrix:

Current State	Sunny	Rainy
Sunny	0.7 (stay sunny)	0.3 (transition to rainy)
Rainy	0.2 (transition to sunny)	0.8 (stay rainy)

Initial State: Assume the initial state is Sunny with a probability of 1.

Markov Model Example

- Step 1: We identified the states (Sunny, Rainy).
- Step 2: We calculated the transition probabilities based on the weather patterns (e.g., the probability of staying sunny after a sunny day is 0.7).
- Step 3: We defined the initial state as Sunny.
- Step 4: We can now use the matrix and initial state to calculate the probability of the weather being sunny or rainy at the next time step.
- Step 5: We can further use this model to simulate the weather patterns for several days, analyze the long-term weather trends, or compare different scenarios with different initial states or transition probabilities.

Markov Chain

- These are the simplest type of Markov model and are used to represent systems where all states are observable.
- Markov chains show all possible states, and between states, they show the transition rate, which is the probability of moving from one state to another per unit of time.
- Applications of this type of model include prediction of market crashes, speech recognition and search engine algorithms.

Application in NLP

- Markov analysis is also used in natural language processing (NLP) and in machine learning.
- For NLP, a Markov chain can be used to generate a sequence of words that form a complete sentence, or a hidden Markov model can be used for named-entity recognition and tagging parts of speech.
- For machine learning, Markov decision processes are used to represent reward in reinforcement learning.

How are Markov models represented?

- The simplest Markov model is a Markov chain, which can be expressed in equations, as a transition matrix or as a graph.
- A transition matrix is used to indicate the probability of moving from each state to each other state.
- Generally, the current states are listed in rows, and the next states are represented as columns. Each cell then contains the probability of moving from the current state to the next state.
- For any given row, all the cell values must then add up to one.

How are Markov models represented?

- A graph consists of circles, each of which represents a state, and directional arrows to indicate possible transitions between states.
- The directional arrows are labeled with the transition probability. The transition probabilities on the directional arrows coming out of any given circle must add up to one.
- Other Markov models are based on the chain representations but with added information, such as observations and observation likelihoods.

How are Markov models represented?

- The transition matrix below represents shifting gears in a car with a manual transmission.
- Six states are possible, and a transition from any given state to any other state depends only on the current state -- that is, where the car goes from second gear isn't influenced by where it was before second gear.
- Such a transition matrix might be built from empirical observations that show, for example, that the most probable transitions from first gear are to second or neutral.

How are Markov models represented?

Shifting gears

	First gear	Second gear	Third gear	Fourth gear	Neutral	Reverse
First gear	.005	.5	.05	.04	.4	.005
Second gear	.4	.005	.5	.05	.04	.005
Third gear	.05	.4	.005	.5	.04	.005
Fourth gear	.04	.05	.8	.005	.1	.005
Neutral	.6	.005	.005	.005	.005	.38
Reverse	.1	.005	.005	.005	.88	.005

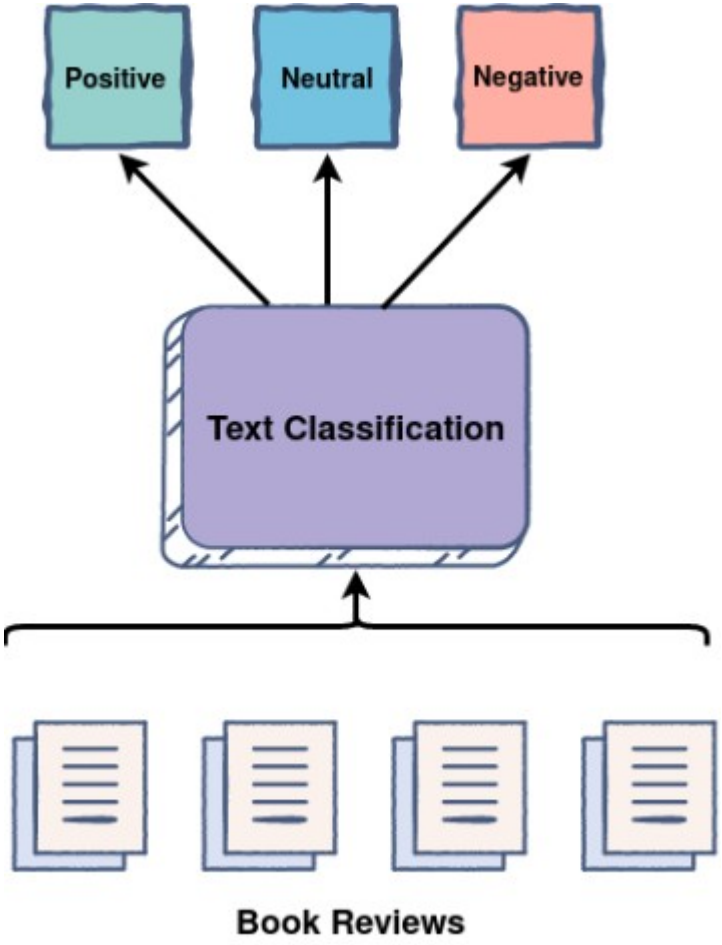
Markov Model

- Practical

Text Classification

- Text Classification is the processing of labeling or organizing text data into groups.
- It forms a fundamental part of Natural Language Processing. In the digital age that we live in we are surrounded by text on our social media accounts, in commercials, on websites, Ebooks, etc.
- The majority of this text data is unstructured, so classifying this data can be extremely useful.

Text Classification



Text Classification: Applications

- Spam detection in emails
- Sentiment analysis of online reviews
- Topic labeling documents like research papers
- Language detection like in Google Translate
- Age/gender identification of anonymous users
- Tagging online content
- Speech recognition used in virtual assistants like Siri and Alexa

Rule Based Approach

- These approaches make use of handcrafted linguistic rules to classify text.
- One way to group text is to create a list of words related to a certain column and then judge the text based on the occurrences of these words.
- For example, words like “fur”, “feathers”, “claws”, and “scales” could help a zoologist identify texts talking about animals online.
- These approaches require a lot of domain knowledge to be extensive, take a lot of time to compile, and are difficult to scale.

Bag of Words

The Bag of Words Representation

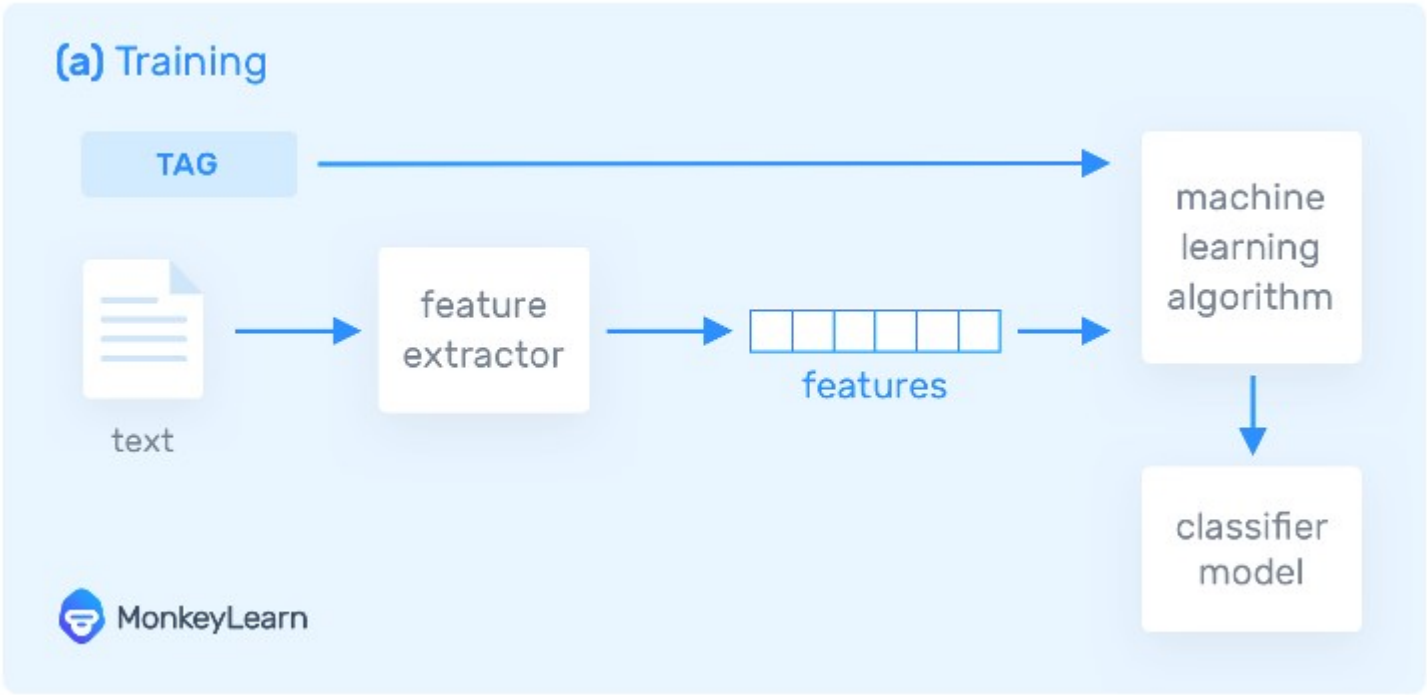
I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!

it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1

Machine Learning Approach

- We can use machine learning to train models on large sets of text data to predict categories of new text.
- To train models, we need to transform text data into numerical data – this is known as feature extraction.
- Important feature extraction techniques include bag of words and n-grams.

Machine Learning Approach



Machine Learning Approach

- There are several useful machine learning algorithms we can use for text classification.
- The most popular ones are:
  - Decision Tree
  - K-Nearest Neighbors
  - Naive Bayes classifiers
  - Support vector machines
  - Logistic Regression
  - Deep learning algorithms

Hybrid Approach

- These approaches are a combination of the two algorithms above.
- They make use of both rule-based and machine learning techniques to model a classifier that can be fine-tuned in certain scenarios.

Bag of words

- Bag of words is a Natural Language Processing technique of text modelling.
- In technical terms, we can say that it is a method of feature extraction with text data.
- This approach is a simple and flexible way of extracting features from documents.

Bag of words

- A bag of words is a representation of text that describes the occurrence of words within a document.
- We just keep track of word counts and disregard the grammatical details and the word order.
- It is called a “bag” of words because any information about the order or structure of words in the document is discarded.
- The model is only concerned with whether known words occur in the document, not where in the document.

Bag of words: Why?

- One of the biggest problems with text is that it is messy and unstructured, and machine learning algorithms prefer structured, well defined fixed-length inputs and by using the Bag-of-Words technique we can convert variable-length texts into a fixed-length vector.
- Also, at a much granular level, the machine learning models work with numerical data rather than textual data.
- So to be more specific, by using the bag-of-words (BoW) technique, we convert a text into its equivalent vector of numbers.

Bag of words: Example

- Sentences:
  - The quick brown fox jumps over the lazy dog.
  - The cat chases the mouse and it squeaks loudly.

Bag of words: Example

Word	Sentence 1	Sentence 2
the	2	2
quick	1	0
brown	1	0
fox	1	0
jumps	1	0
over	1	0
lazy	1	0
dog	1	0
cat	0	1
chases	0	1
mouse	0	1
and	0	1
it	0	1
squeaks	0	1
loudly	0	1

Example:

- Practical

N-grams

- Again same questions, what are n-grams and why do we use them? Let us understand this with an example below-

- Sentence 1: “This is a good job. I will not miss it for anything”

- Sentence 2: ”This is not good at all”

N-grams

- For this example, let us take the vocabulary of 5 words only. The five words being-
  - good
  - job
  - miss
  - not
  - all
- So, the respective vectors for these sentences are:

  “This is a good job. I will not miss it for anything”=[1,1,1,1,0]

  ”This is not good at all”=[1,0,0,1,1]

N-grams

- Can you guess what is the problem here? Sentence 2 is a negative sentence and sentence 1 is a positive sentence. Does this reflect in any way in the vectors above? Not at all.
- So how can we solve this problem? Here come the N-grams to our rescue.
- An N-gram is an N-token sequence of words: a 2-gram (more commonly called a bigram) is a two-word sequence of words like “really good”, “not good”, or “your homework”, and a 3-gram (more commonly called a trigram) is a three-word sequence of words like “not at all”, or “turn off light”.

N-grams

- For example, the bigrams in the first line of text in the previous section: “This is not good at all” are as follows:
  - “This is”
  - “is not”
  - “not good”
  - “good at”
  - “at all”
- Now if instead of using just words in the above example, we use bigrams (Bag-of-bigrams) as shown above. The model can differentiate between sentence 1 and sentence 2.
- So, using bi-grams makes tokens more understandable (for example, “HSR Layout”, in Bengaluru, is more informative than “HSR” and “layout”)

N-grams

- Practical

The TF-IDF Vectorizer

- The TF*IDF algorithm is used to weigh a keyword in any document and assign the importance to that keyword based on the number of times it appears in the document.
- Put simply, the higher the TF*IDF score (weight), the rarer and more important the term, and vice versa.
- Each word or term has its respective TF and IDF score. The product of the TF and IDF scores of a term is called the TF*IDF weight of that term.

The TF-IDF Vectorizer

- The TF (term frequency) of a word is the number of times it appears in a document. When you know it, you're able to see if you're using a term too often or too infrequently.
  - TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).
- The IDF (inverse document frequency) of a word is the measure of how significant that term is in the whole corpus.
  - IDF(t) = log_e(Total number of documents / Number of documents with term t in it).

The TF-IDF Vectorizer

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

N = total number of documents

Example:

- 1. It was a beautiful rainy day that made by whole day awesome.
- 2. We made it awesome by adding more flavors on that day.

Example:

Word	Sentence 1	Sentence 2	TF-IDF Score (Sentence 1)	TF-IDF Score (Sentence 2)
beautiful	1	0	0.5772	0
rainy	1	0	0.5772	0
day	1	1	0.2886	0.2886
made	1	1	0.2886	0.2886
by	1	0	0.5772	0
whole	1	0	0.5772	0
awesome	1	1	0.3772	0.2886
it	0	1	0	1.3863
was	1	0	0.5772	0
we	1	0	0.5772	0
more	0	1	0	0.772
flavors	0	1	0	1.0792
on	0	1	0	0.772
that	0	1	0	0.772

Example:

- Practical

TF*IDF Transformer

- With Tfidftransformer you will systematically compute word counts using CountVectorizer and then compute the Inverse Document Frequency (IDF) values and only then compute the Tf-idf scores.

Example:

- Practical

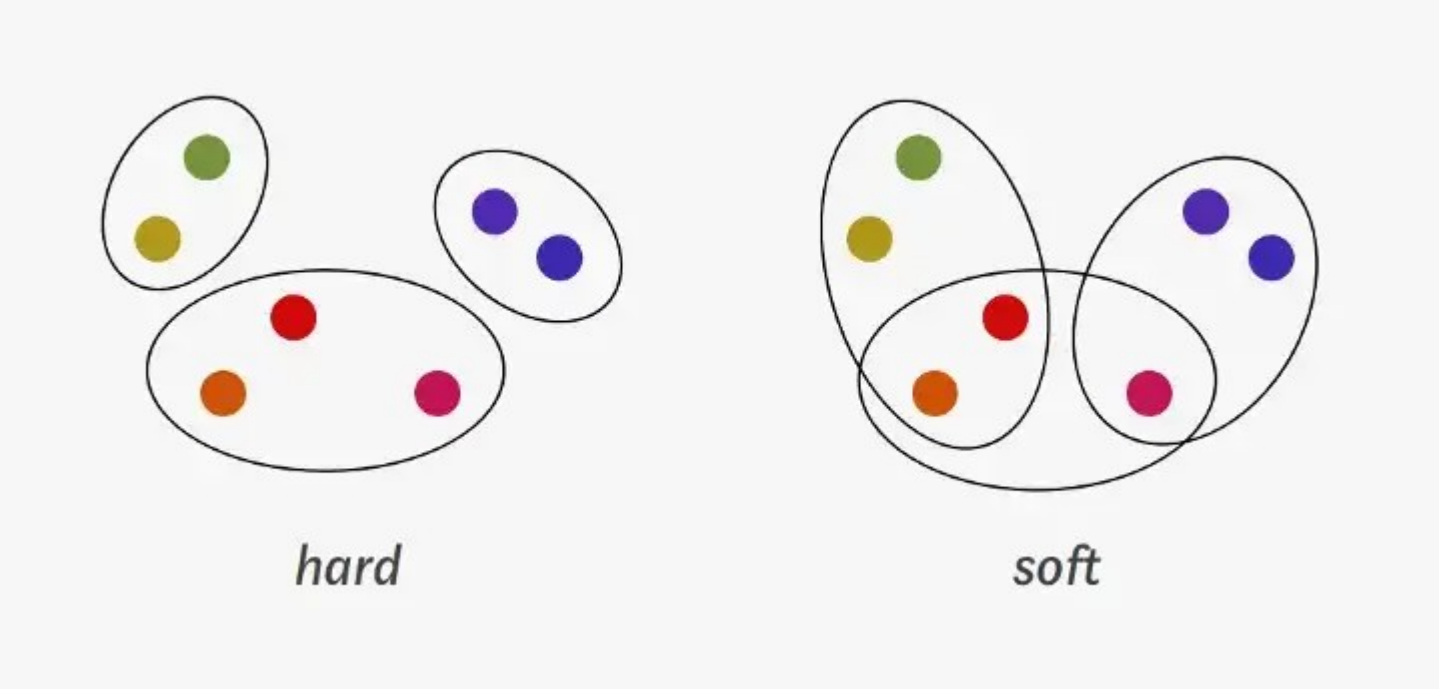
When to use what?

- If you need the term frequency (term count) vectors for different tasks, use Tfidftransformer.
- If you need to compute tf-idf scores on documents within your “training” dataset, use Tfidfvectorizer
- If you need to compute tf-idf scores on documents outside your “training” dataset, use either one, both will work.

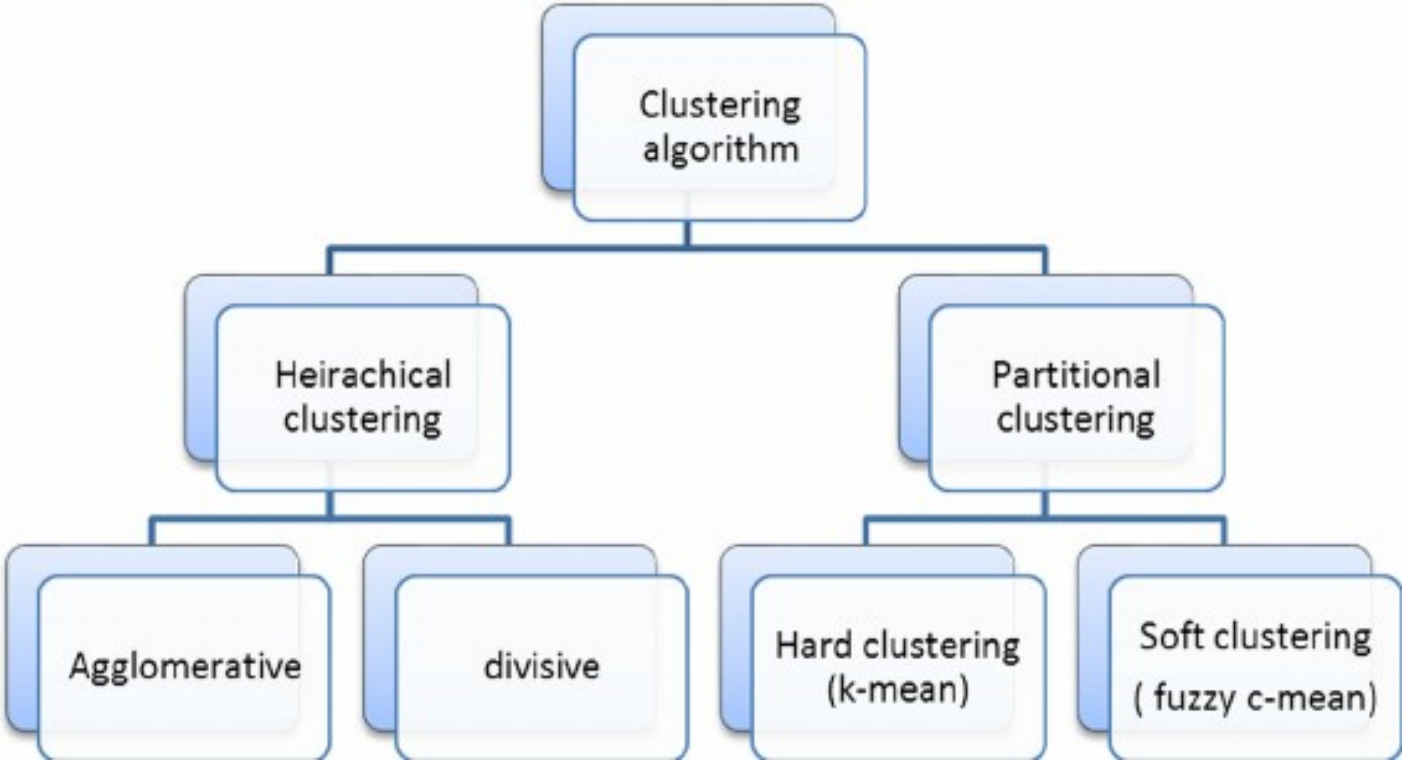
Text Clustering

- Clustering is a data mining technique which groups unlabeled data based on their similarities or differences.
- Clustering algorithms are used to process raw, unclassified data objects into groups represented by structures or patterns in the information.
- Clustering algorithms can be categorized into a few types, specifically exclusive, overlapping, hierarchical, and probabilistic.

Common Clustering Types



Common Clustering Algorithms



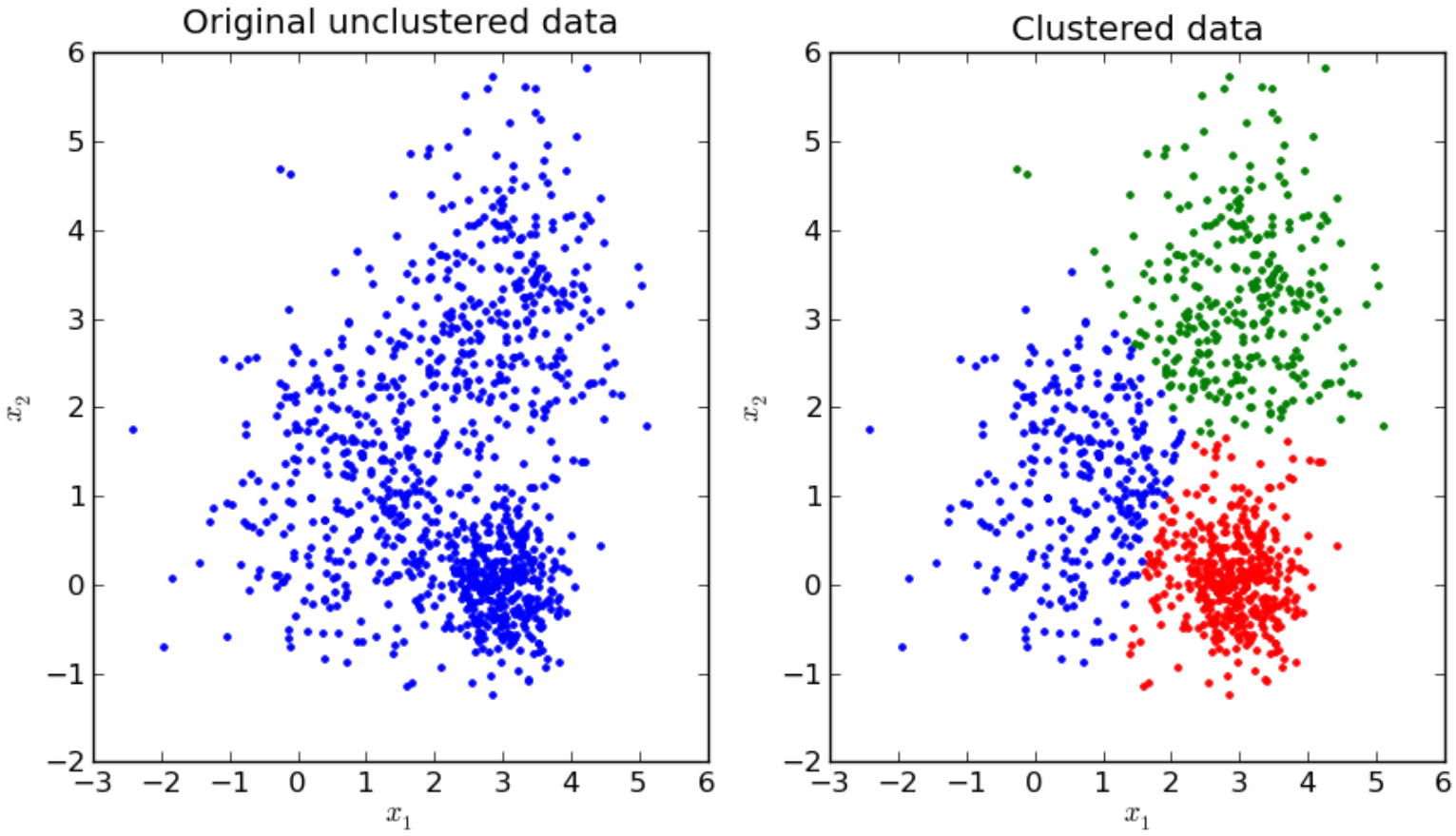
K-means clustering

- The key objective of a k-means algorithm is to organize data into clusters such that there is high intra-cluster similarity and low inter-cluster similarity.
- An item will only belong to one cluster, not several, that is, it generates a specific number of disjoint, non-hierarchical clusters.

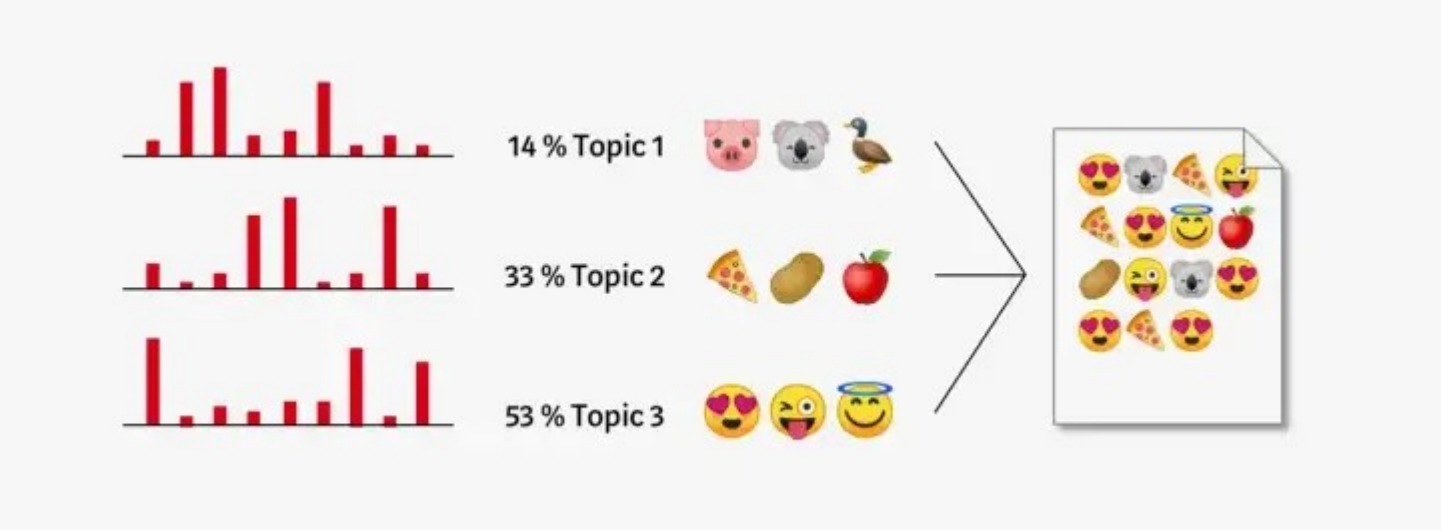
K-means clustering

- K-means uses the strategy of divide and concur, and it is a classic example for an expectation maximization (EM) algorithm.
- EM algorithms are made up of two steps:
  - The first step is known as expectation(E) and is used to find the expected point associated with a cluster; and
  - The second step is known as maximization(M) and is used to improve the estimation of the cluster using knowledge from the first step.
- The two steps are processed repeatedly until convergence is reached.

Clustering



Clustering



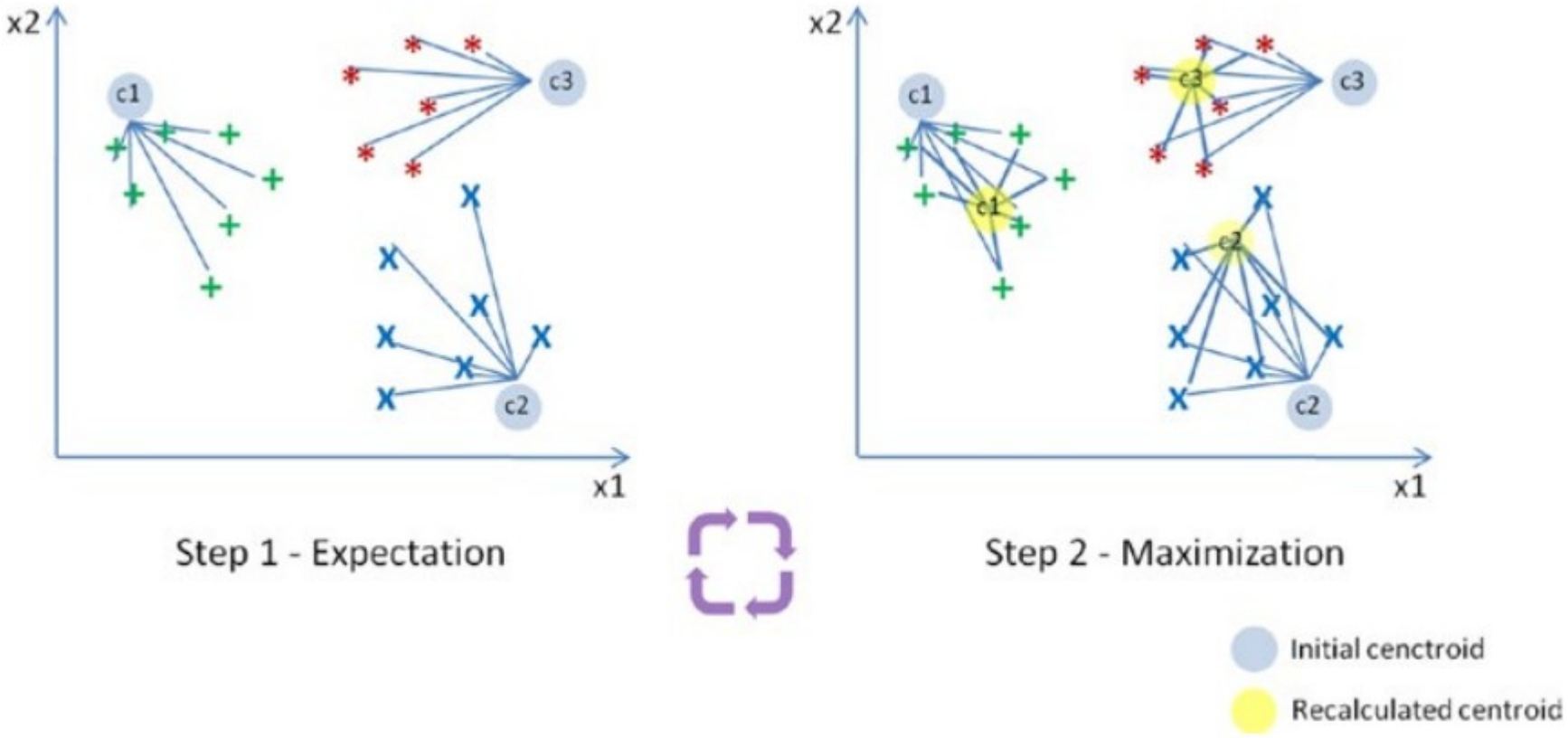
Generalized algorithm

- Our algorithm works as follows, assuming we have inputs x1, x2, x3,…,xn and value of K
  - Step 1 - Pick K random points as cluster centers called centroids.
  - Step 2 - Assign each xi to nearest cluster by calculating its distance to each centroid.
  - Step 3 - Find new cluster center by taking the average of the assigned points.
  - Step 4 - Repeat Step 2 and 3 until none of the cluster assignments change.

Algorithm work flow

- Step 1: In the first step k centroids (in above case k=3) is randomly picked (only in the first iteration) and all the points that are nearest to each centroid point are assigned to that specific cluster. Centroid is the arithmetic mean or average position of all the points.
- Step 2: Here the centroid point is recalculated using the average of the coordinates of all the points in that cluster. Then step one is repeated (assign nearest point) until the clusters converge.

Algorithm work flow



Limitations

- K-means clustering needs the number of clusters to be specified.
- K-means has problems when clusters are of differing sized, densities, and non-globular shapes.
- Presence of outlier can skew the results.

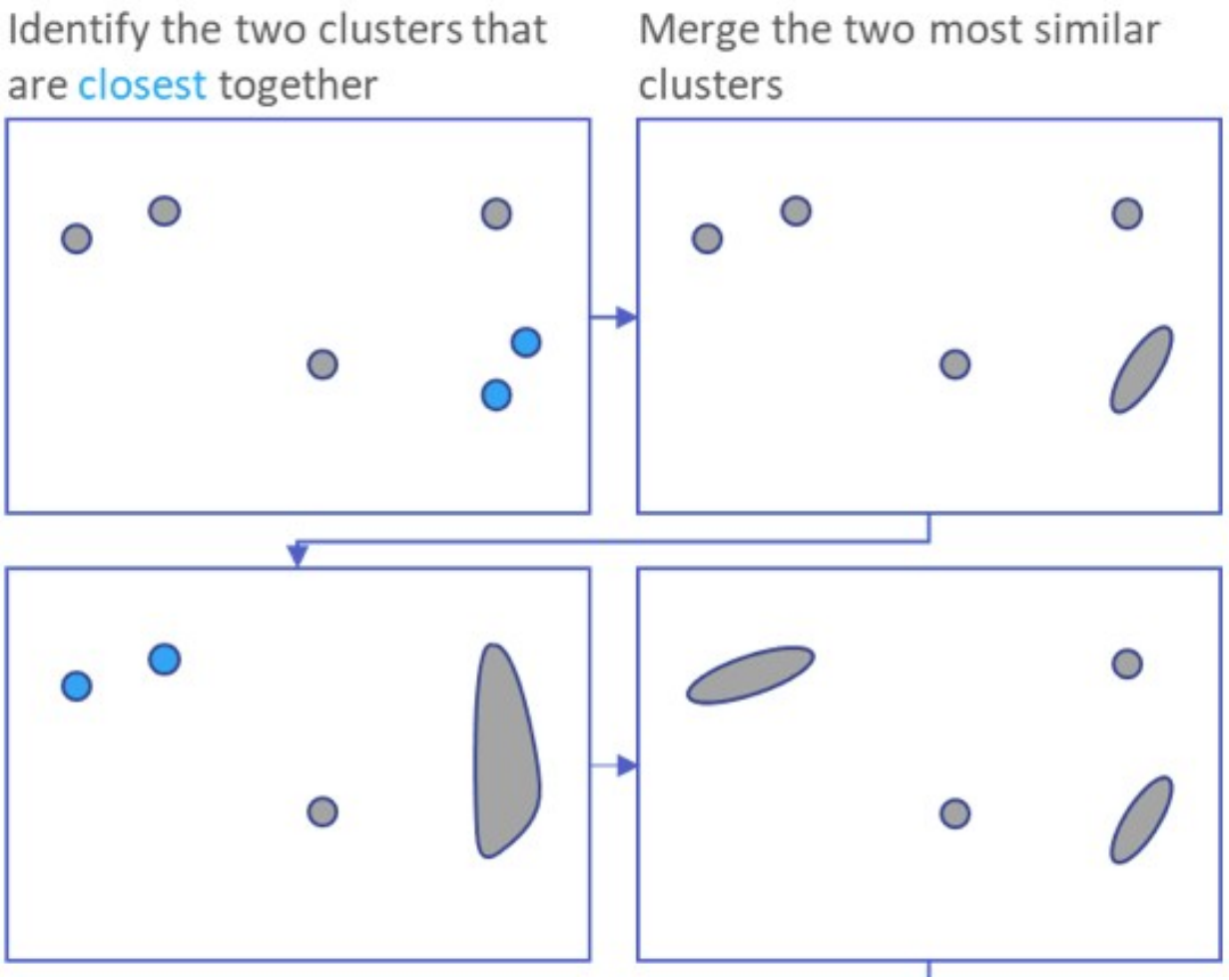
Hierarchical Clustering

- Hierarchical clustering, also known as hierarchical cluster analysis, is an algorithm that groups similar objects into groups called clusters.
- The endpoint is a set of clusters, where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other.
- Hierarchical clustering starts by treating each observation as a separate cluster. Then, it repeatedly executes the following two steps: (1) identify the two clusters that are closest together, and (2) merge the two most similar clusters. This continues until all the clusters are merged together.

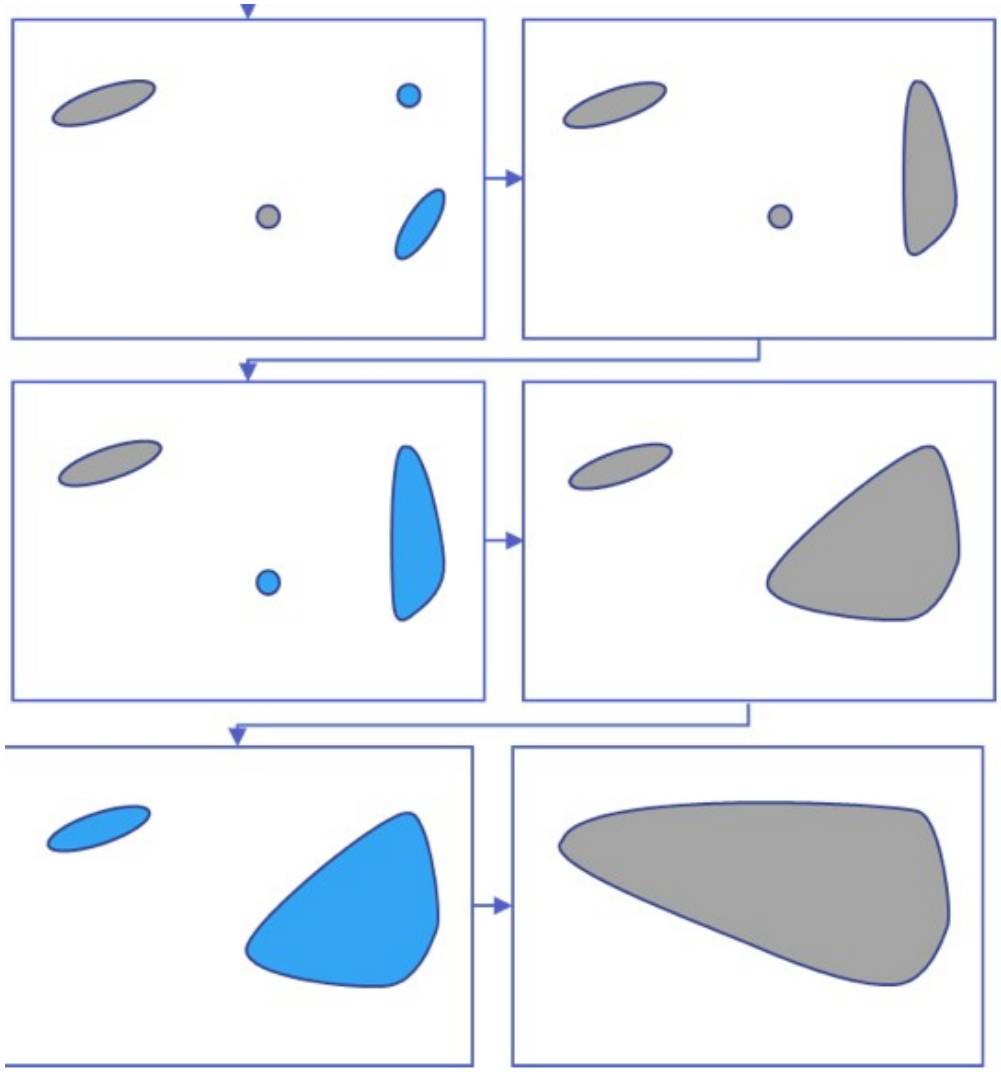
Hierarchical Clustering – Steps

- At the start, treat each data point as one cluster. Therefore, the number of clusters at the start will be K, while K is an integer representing the number of data points.
- Form a cluster by joining the two closest data points resulting in K-1 clusters.
- Form more clusters by joining the two closest clusters resulting in K-2 clusters.
- Repeat the above three steps until one big cluster is formed.
- Once single cluster is formed, dendrograms are used to divide into multiple clusters depending upon the problem.

How it works?

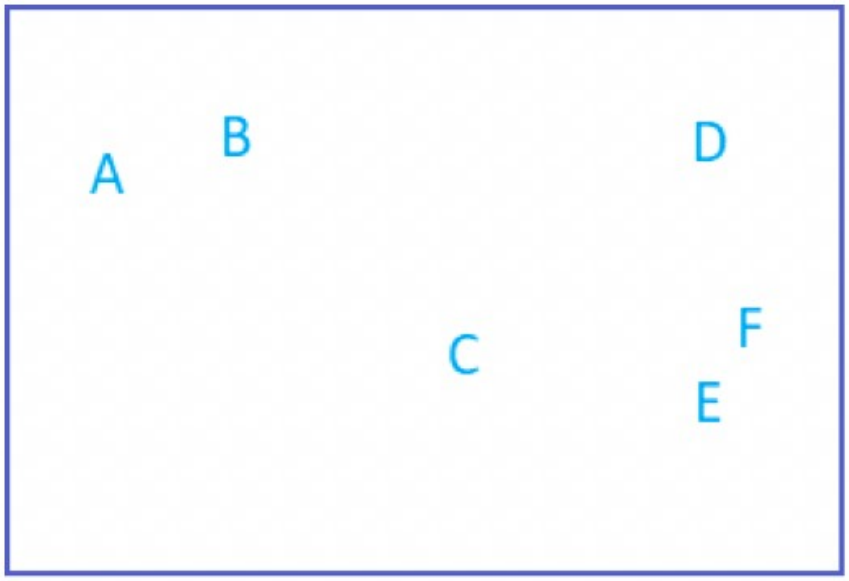

Identify the two clusters that are closest together
Merge the two most similar clusters

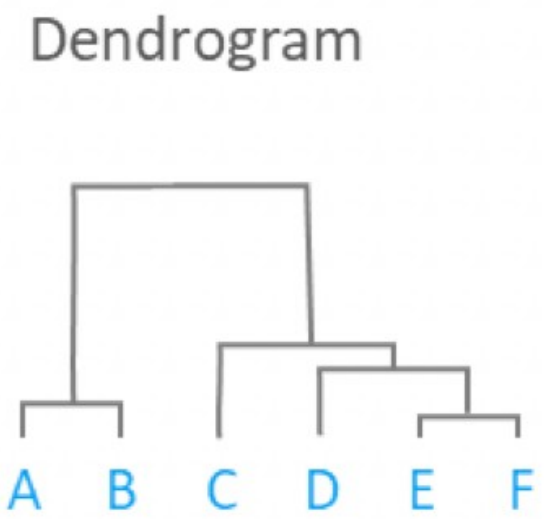
How it works?



Dendrogram

- The main output of Hierarchical Clustering is a dendrogram, which shows the hierarchical relationship between the clusters:





Measures of distance

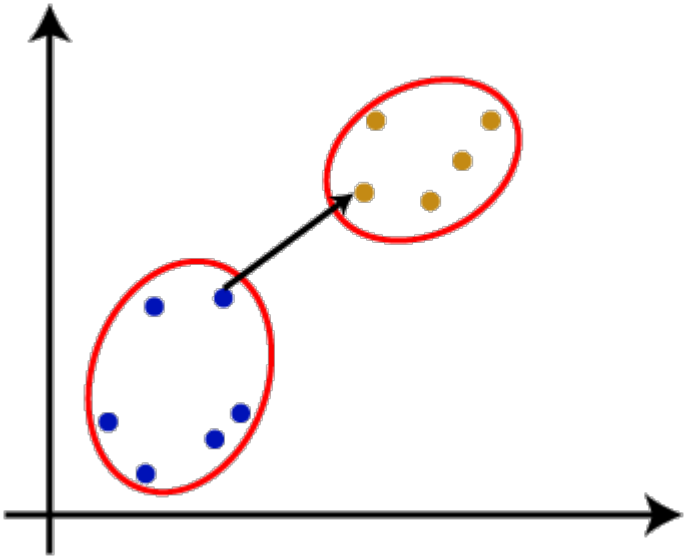
- Similarity:
  - The distance between two clusters has been computed based on length of the straight line drawn from one cluster to another.
  - This is commonly referred to as the Euclidean distance. Many other distance metrics have been developed.

Linkage Criteria

- After selecting a distance metric, it is necessary to determine from where distance is computed.
- For example, it can be computed between the two most similar parts of a cluster (single-linkage), the two least similar bits of a cluster (complete-linkage), the center of the clusters (mean or average-linkage), or some other criterion.
- Many linkage criteria have been developed.
  - Distance metrics
  - Ward's Method

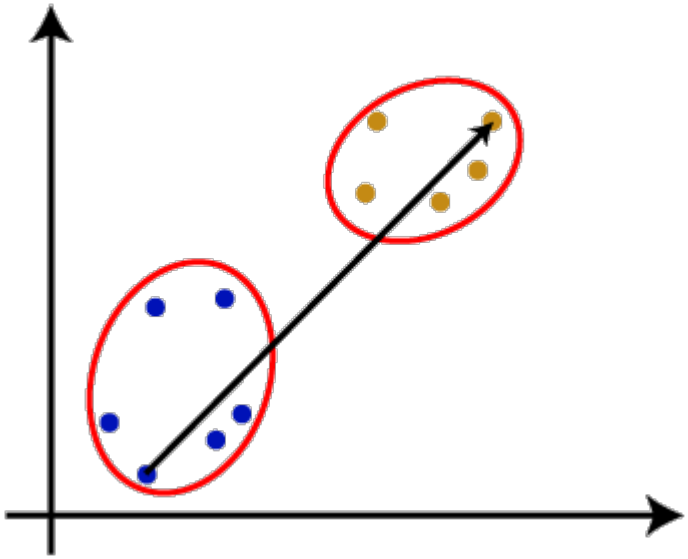
Single Linkage

- Single Linkage: It is the Shortest Distance between the closest points of the clusters.
- Consider the below image:



Complete Linkage

- It is the farthest distance between the two points of two different clusters. It is one of the popular linkage methods as it forms tighter clusters than single-linkage.

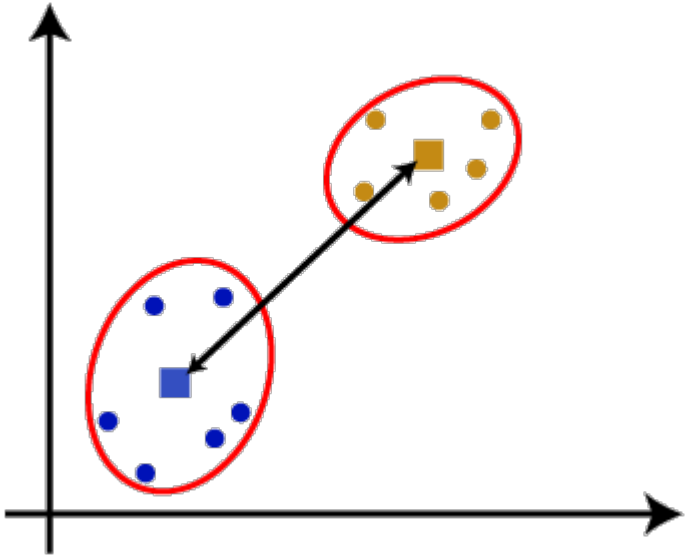


Average Linkage

- Average Linkage: It is the linkage method in which the distance between each pair of datasets is added up and then divided by the total number of datasets to calculate the average distance between two clusters.
- It is also one of the most popular linkage methods.

Centroid Linkage

- Centroid Linkage: It is the linkage method in which the distance between the centroid of the clusters is calculated.
- Consider the below image:



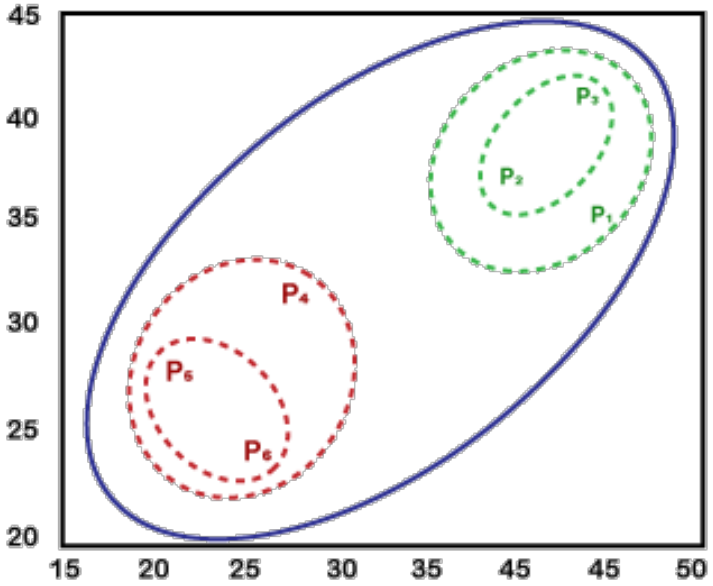
Ward's Method

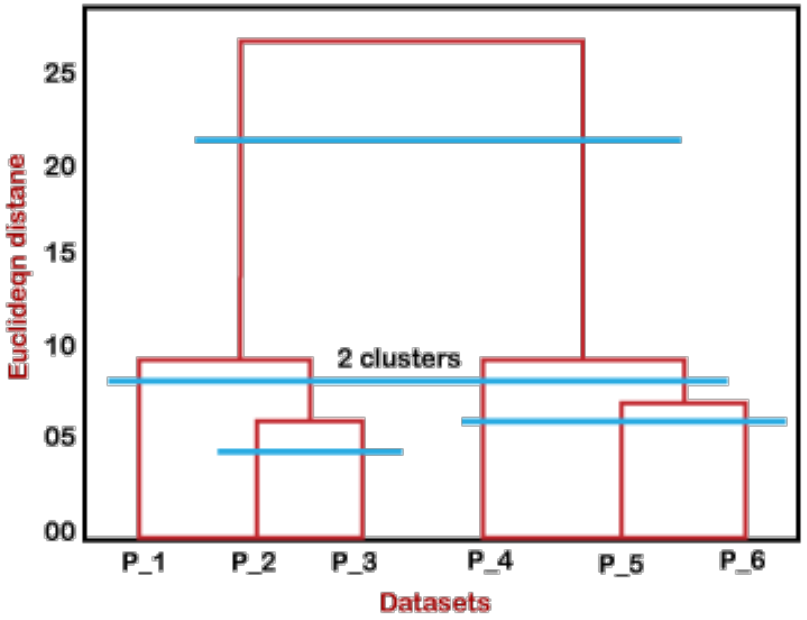
- Ward's method (a.k.a. Minimum variance method or Ward's Minimum Variance Clustering Method) is an alternative to single-link clustering. Popular in fields like linguistics, it's liked because it usually creates compact, even-sized clusters.
- Like most other clustering methods, Ward's method is computationally intensive. However, Ward's has significantly fewer computations than other methods.
- The drawback is this usually results in less than optimal clusters. That said, the resulting clusters are usually good enough for most purposes.

Ward's Method

- Like other clustering methods, Ward's method starts with n clusters, each containing a single object. These n clusters are combined to make one cluster containing all objects.
- At each step, the process makes a new cluster that minimizes variance, measured by an index called E (also called the sum of squares index).
- At each step, the following calculations are made to find E:
  - Find the mean of each cluster.
  - Calculate the distance between each object in a particular cluster, and that cluster's mean.
  - Square the differences from Step 2.
  - Sum (add up) the squared values from Step 3.
  - Add up all the sums of squares from Step 4.

Working Summary

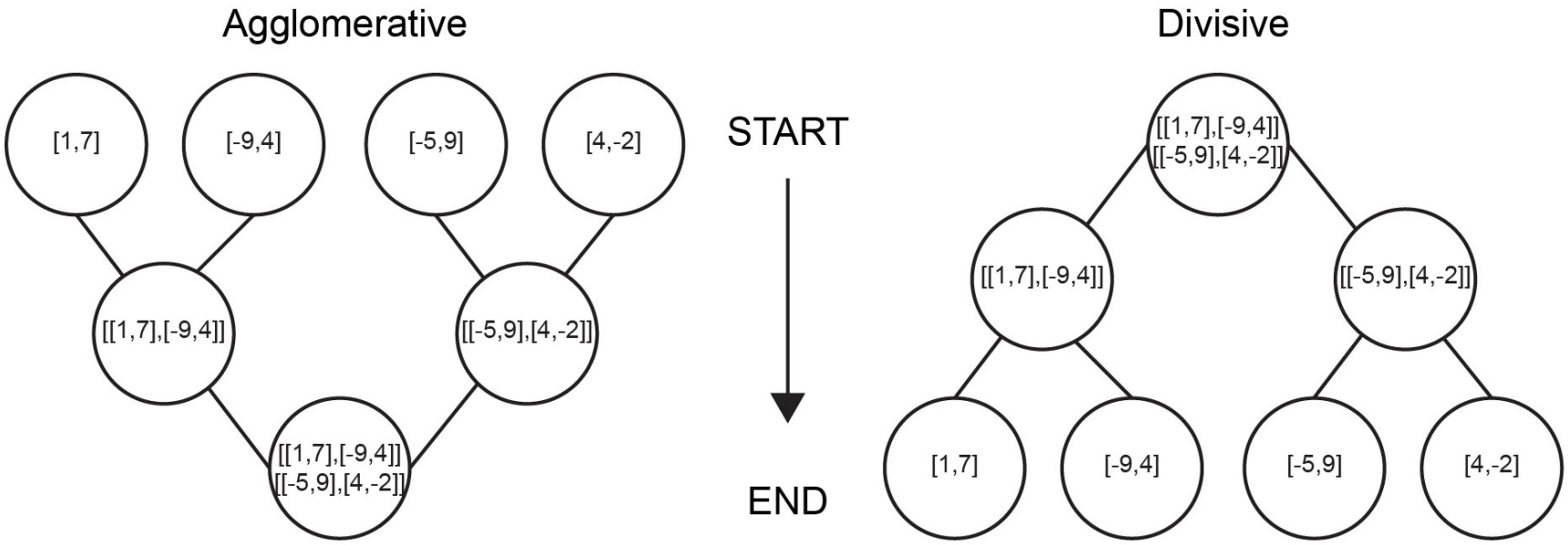




Agglomerative vs. Divisive

- Hierarchical clustering typically works by sequentially merging similar clusters. This is known as agglomerative hierarchical clustering.
- In theory, it can also be done by initially grouping all the observations into one cluster, and then successively splitting these clusters.
- This is known as divisive hierarchical clustering. Divisive clustering is rarely done in practice.

Agglomerative vs. Divisive



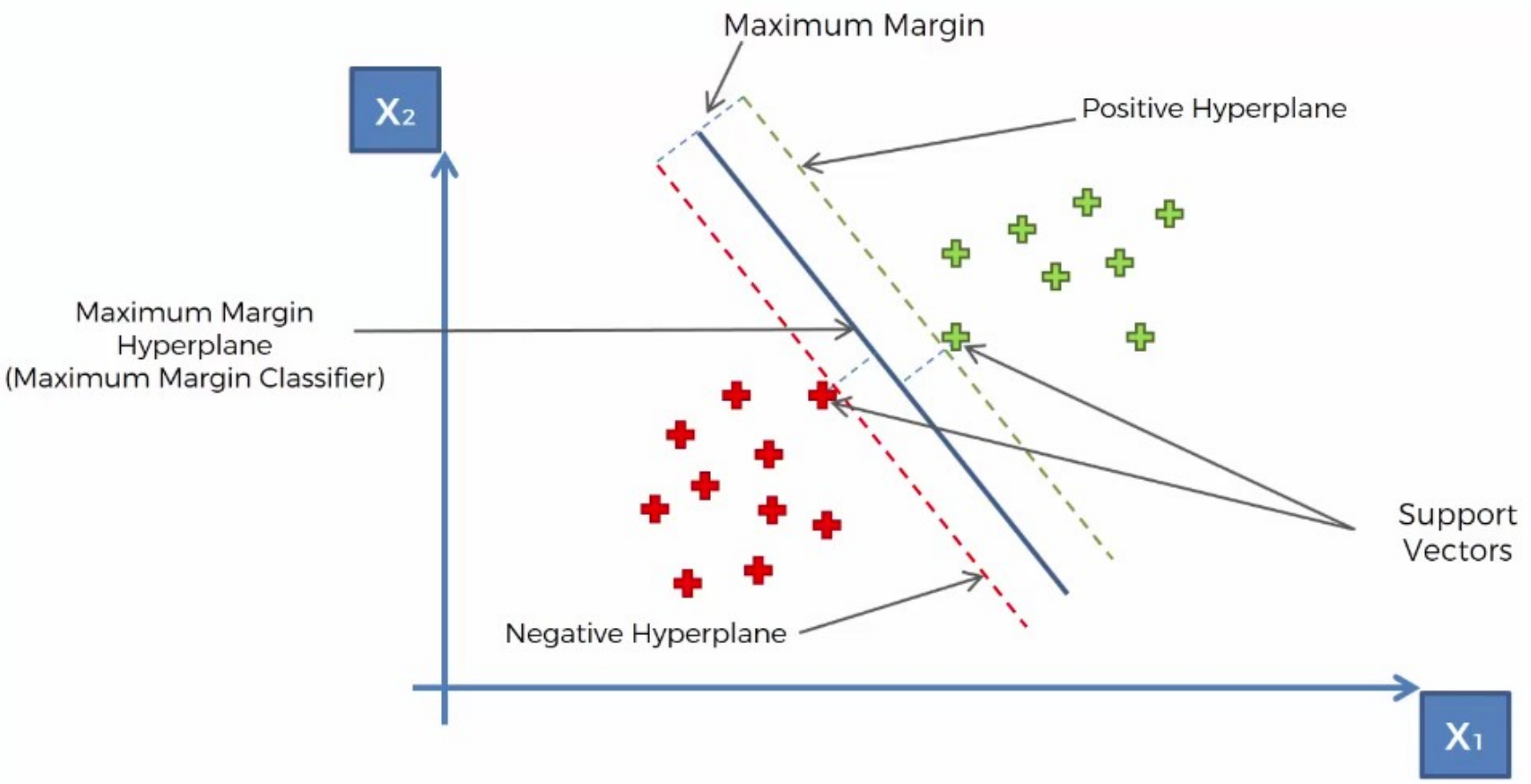
Example:

- Practical

What is support vector?

- “Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems.
- In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate.
- Then, we perform classification by finding the hyperplane that differentiate the two classes very well.

Decision Vectors



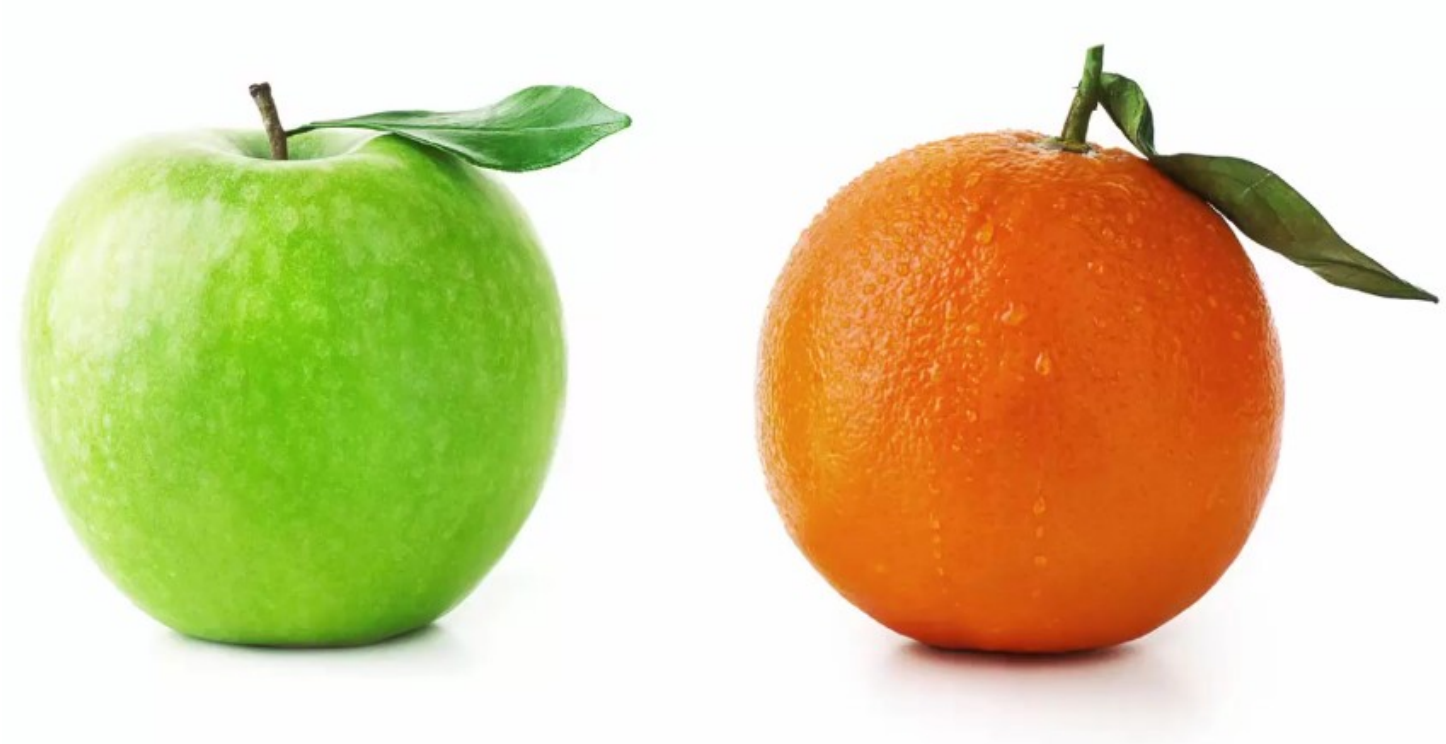
Definitions

- Support Vectors
  - Support vectors are the data points, which are closest to the hyperplane. These points will define the separating line better by calculating margins. These points are more relevant to the construction of the classifier.
- Hyperplane
  - A hyperplane is a decision plane which separates between a set of objects having different class memberships.

Definitions

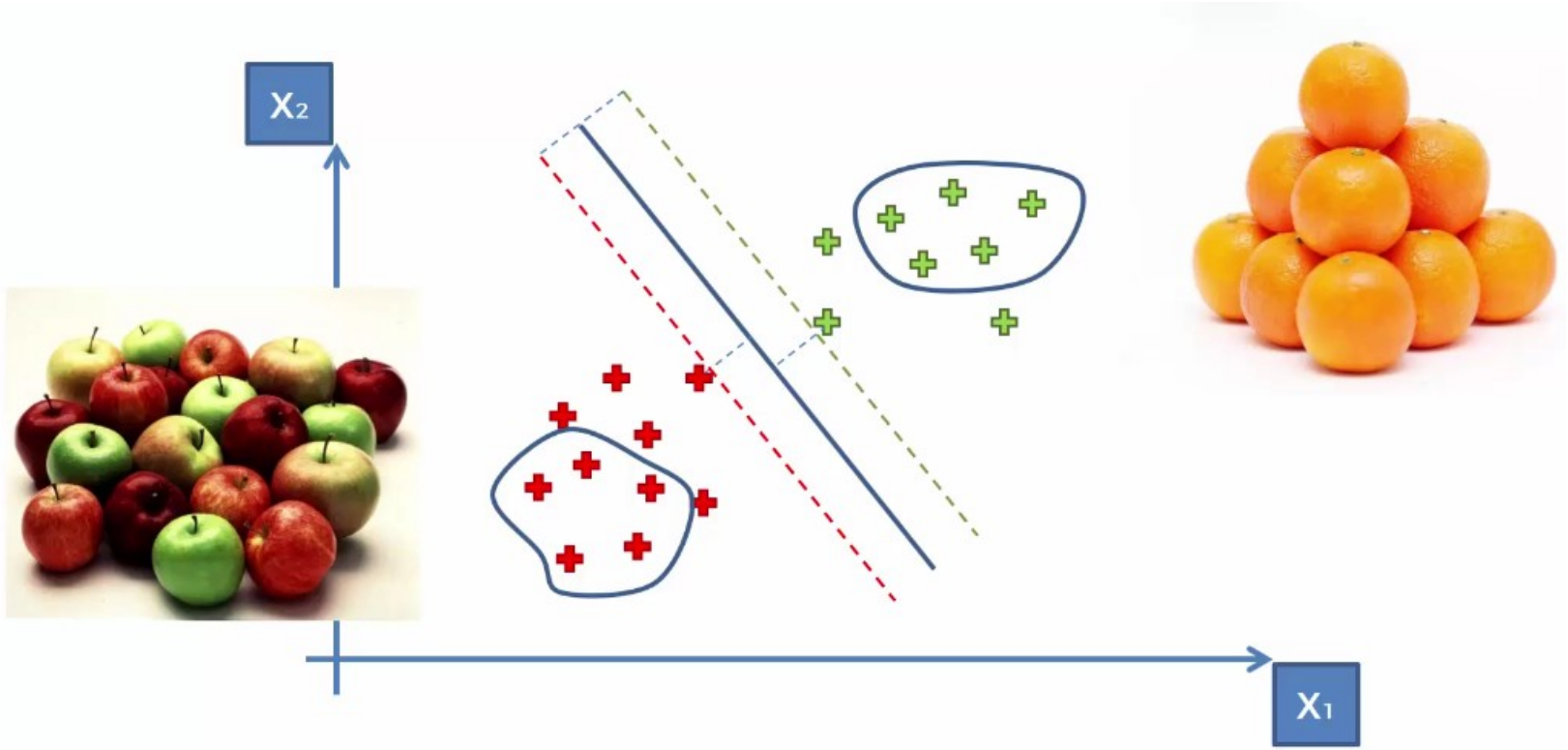
- Margin
  - A margin is a gap between the two lines on the closest class points.
  - This is calculated as the perpendicular distance from the line to support vectors or closest points.
  - If the margin is larger in between the classes, then it is considered a good margin, a smaller margin is a bad margin.

Why SVM is so special ?

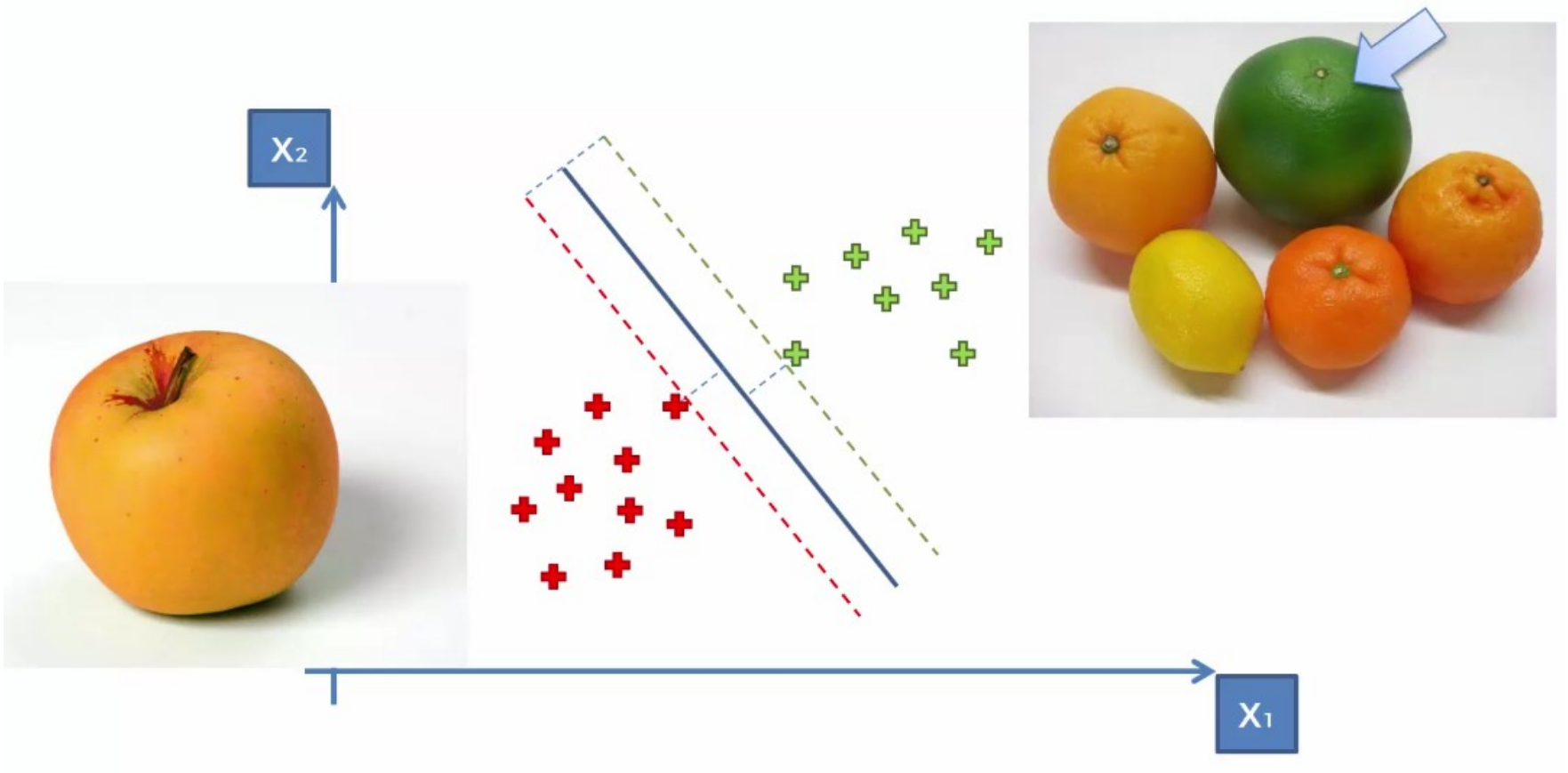


Example Reference: Super Data Science

How SVM works for this?



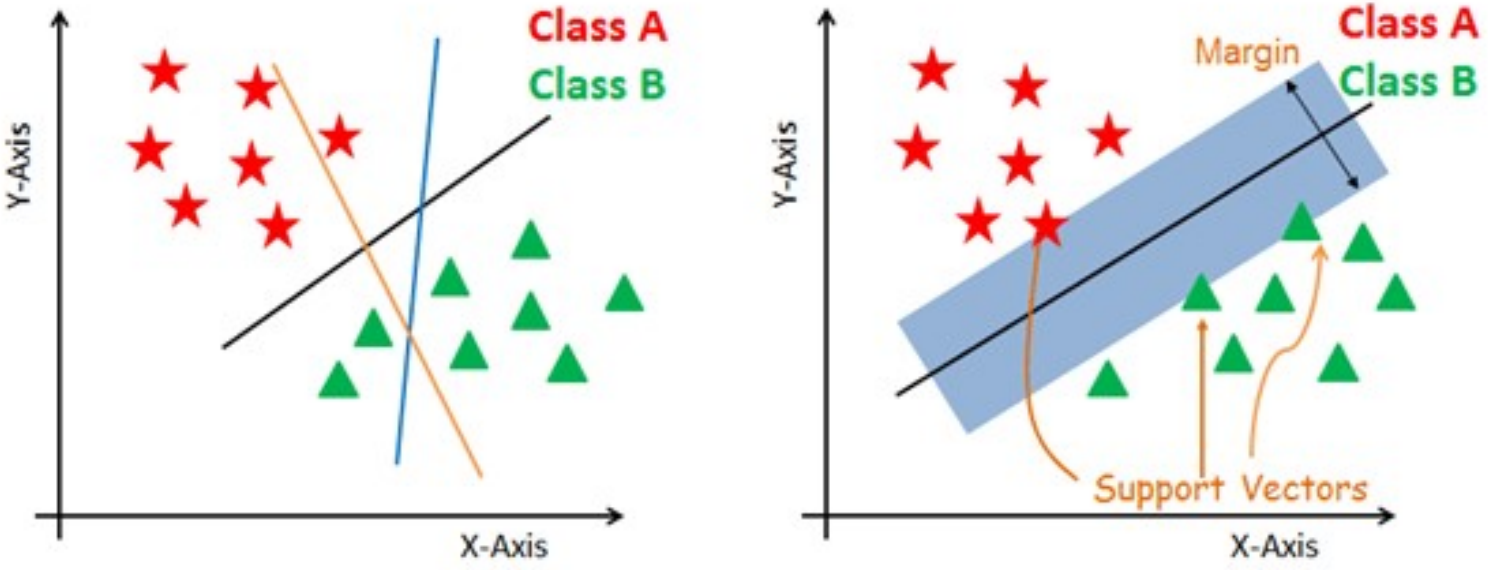
How classification will work ?



How SVM works ?

- The main objective is to segregate the given dataset in the best possible way.
- The distance between the either nearest points is known as the margin.
- The objective is to select a hyperplane with the maximum possible margin between support vectors in the given dataset. SVM searches for the maximum marginal hyperplane in the following steps:
  - Generate hyperplanes which segregates the classes in the best way.
  - Select the right hyperplane with the maximum segregation from the either nearest data points.

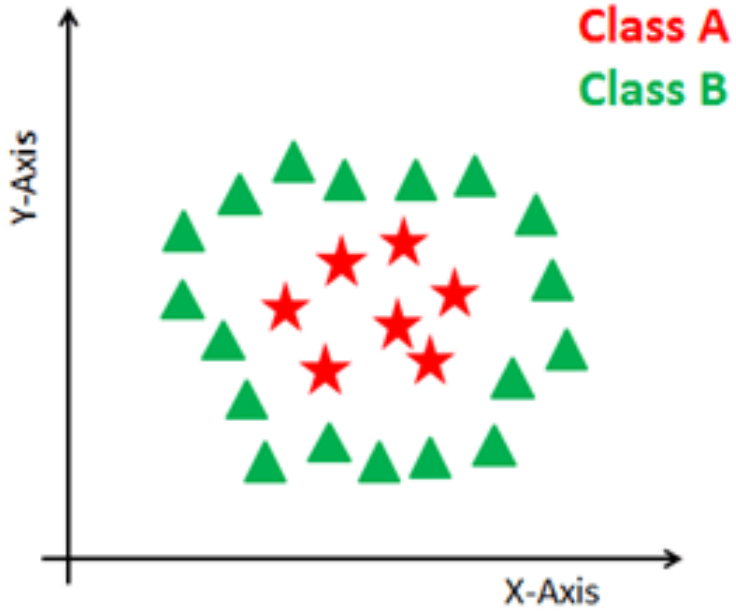
How SVM works ?

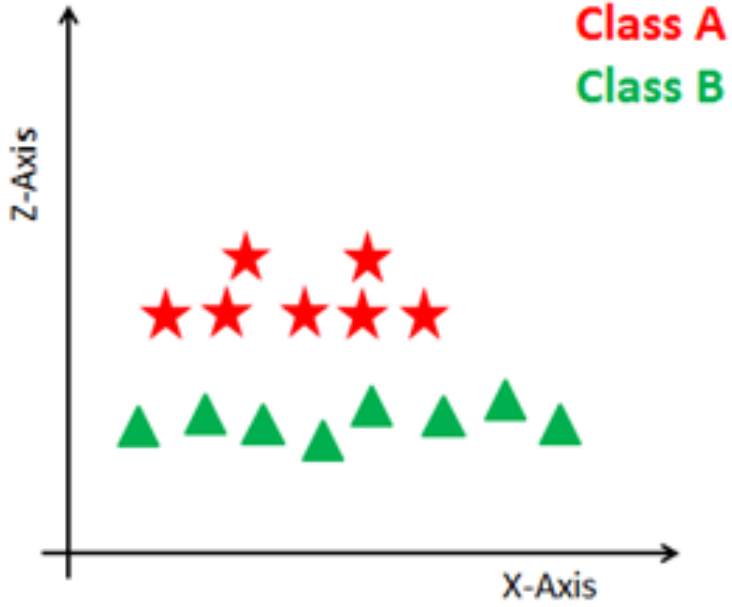


Non-linear and inseparable planes

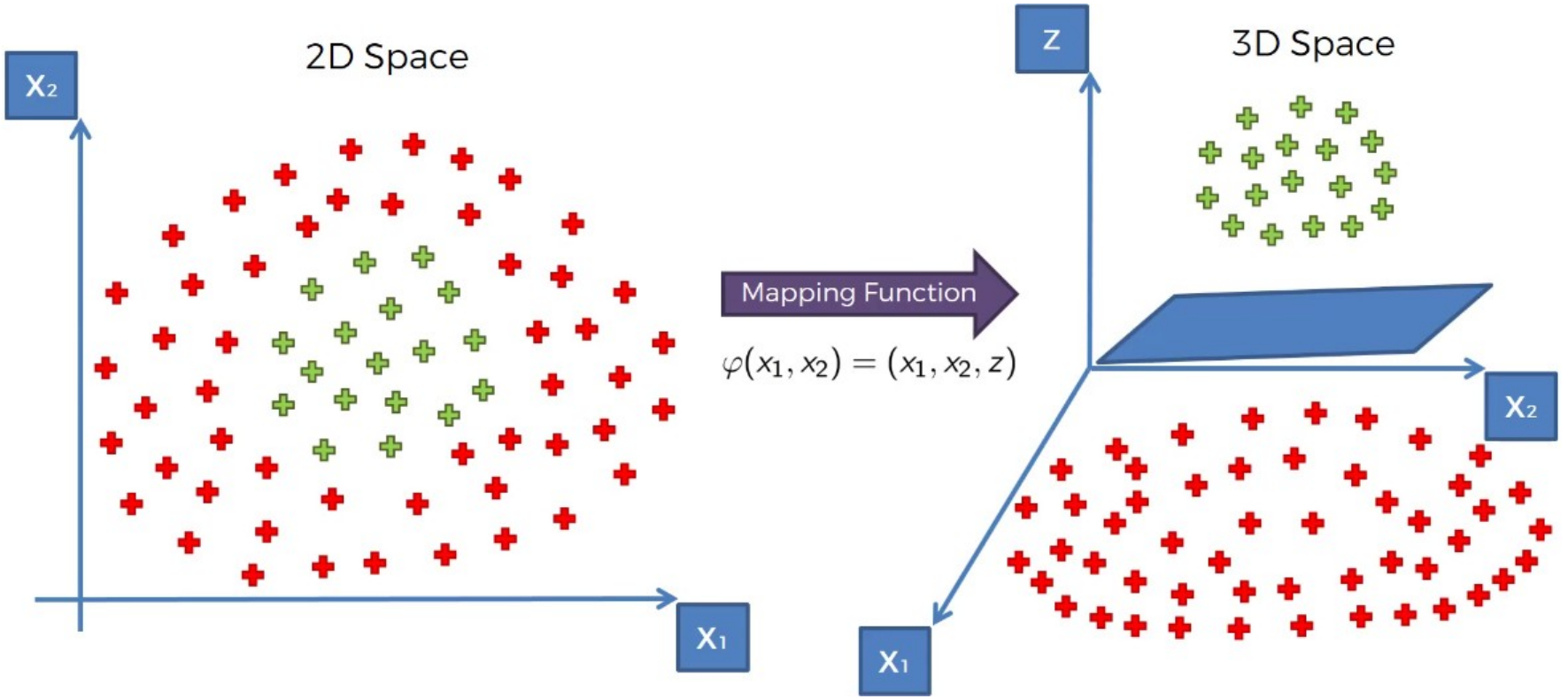
- Some problems can’t be solved using linear hyperplane.
- In such situation, SVM uses a kernel trick to transform the input space to a higher dimensional space as shown on the right.
- The data points are plotted on the x-axis and z-axis (Z is the squared sum of both x and y: z=x^2=y^2).
- Now you can easily segregate these points using linear separation.

Non-linear and inseparable planes

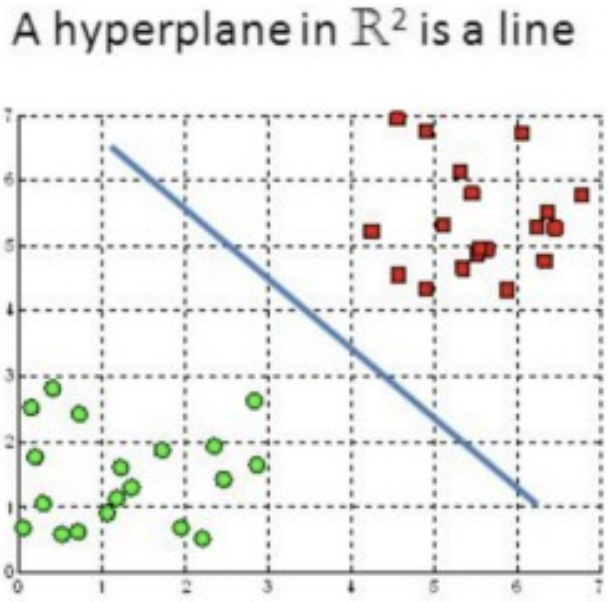


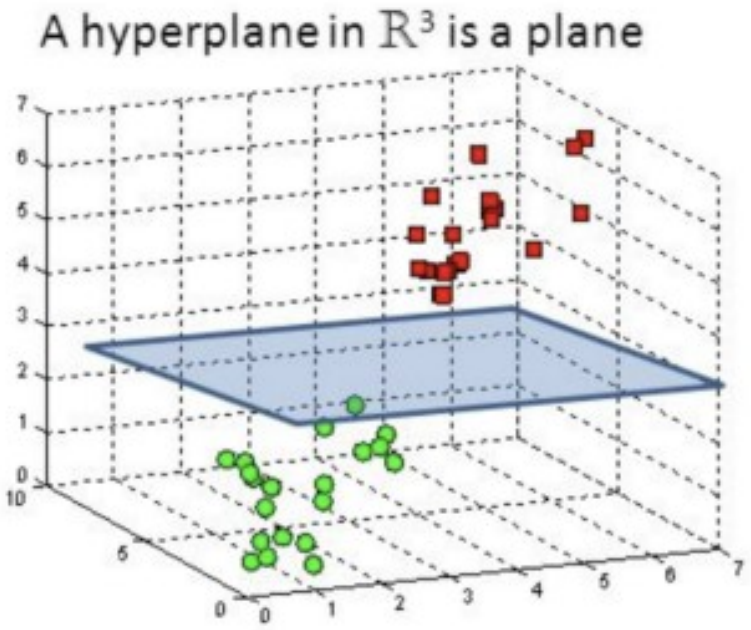


High Dimensional Space Mapping



High Dimensional Space Mapping





SVM Kernels

- The SVM algorithm is implemented in practice using a kernel. A kernel transforms an input data space into the required form.
- SVM uses a technique called the kernel trick. Here, the kernel takes a low-dimensional input space and transforms it into a higher dimensional space.
- In other words, you can say that it converts non-separable problem to separable problems by adding more dimension to it.
- It is most useful in non-linear separation problem. Kernel trick helps you to build a more accurate classifier.

Kernel Types

- Linear Kernel
- Polynomial Kernel
- Radial Basis Function Kernel
- Sigmoid Kernel

Radial Basis Function Kernel

- The Radial basis function kernel is a popular kernel function commonly used in support vector machine classification. RBF can map an input space in infinite dimensional space.

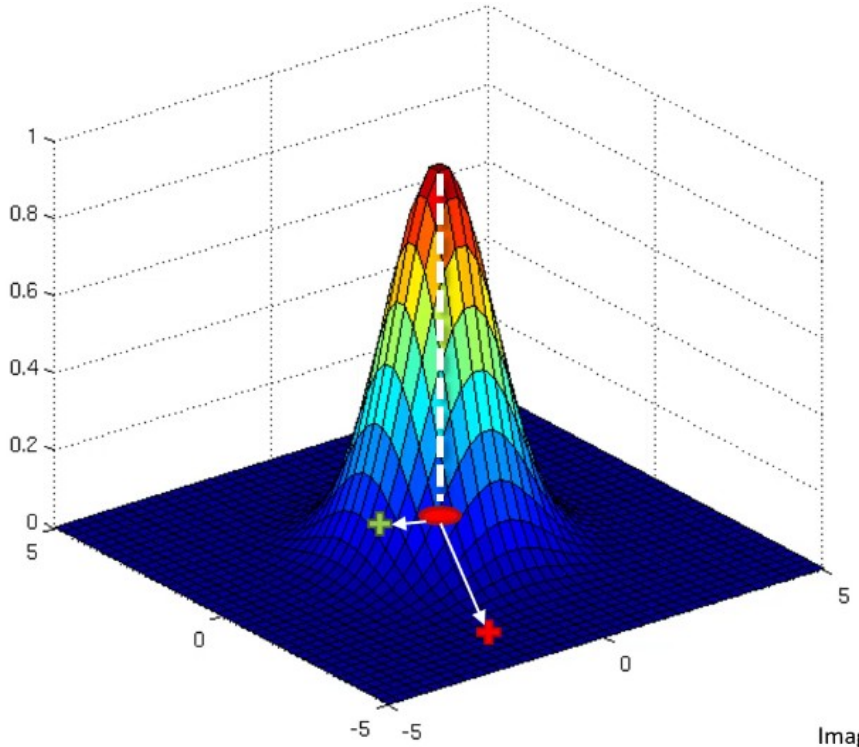
$$K(x,xi) = \exp(-gamma * sum((x - xi^2))$$

- Here gamma is a parameter, which ranges from 0 to 1. A higher value of gamma will perfectly fit the training dataset, which causes over-fitting. Gamma=0.1 is considered to be a good default value.
- The value of gamma needs to be manually specified in the learning algorithm.

Radial Basis Function Kernel

$$K(\vec{x}, \vec{l}^{i}) = e^{-\frac{\|\vec{x}-\vec{l}^{i}\|^2}{2\sigma^2}}$$

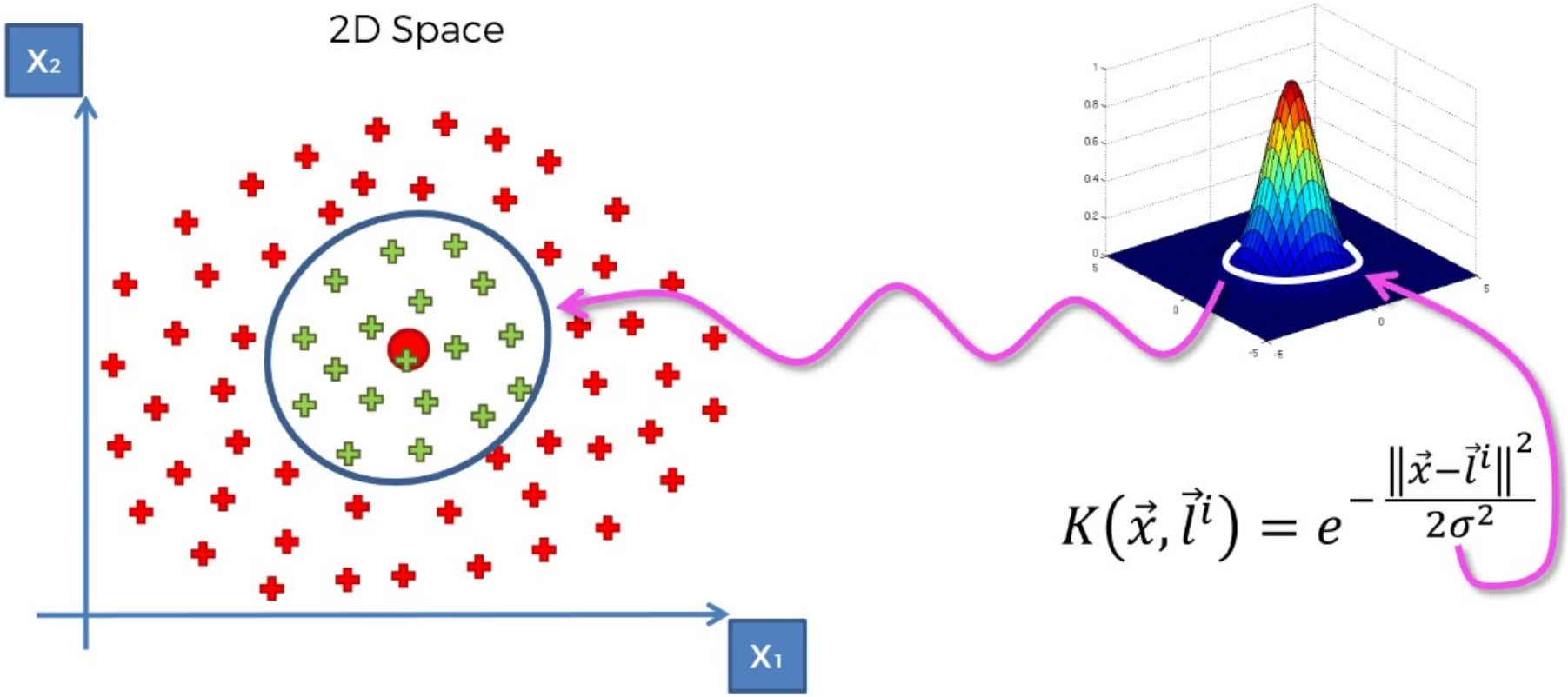
Radial Basis Function Kernel



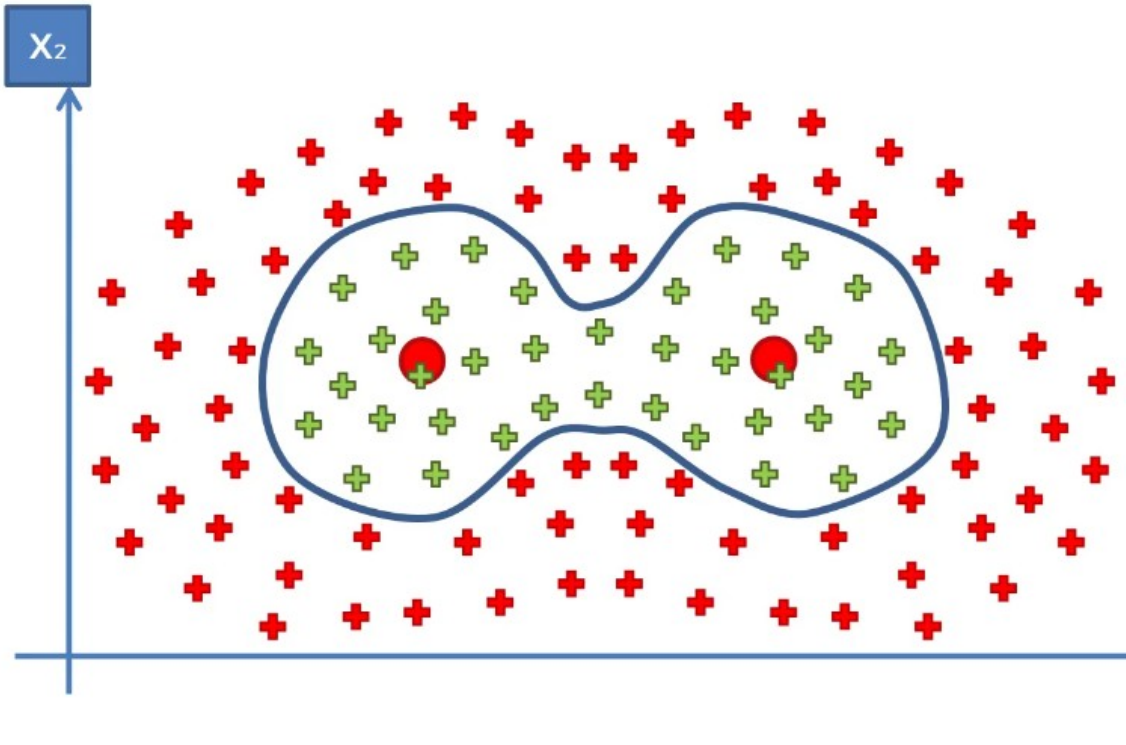
$$K(\vec{x}, \vec{l}^i) = e^{-\frac{\|\vec{x}-\vec{l}^i\|^2}{2\sigma^2}}$$

Image source: http://www.cs.toronto.edu/~duvenaud/cookbook/index.html

Radial Basis Function Kernel



Radial Basis Function Kernel



$$K(\vec{x}, \vec{l}^1) + K(\vec{x}, \vec{l}^2)$$

(Simplified Formula)

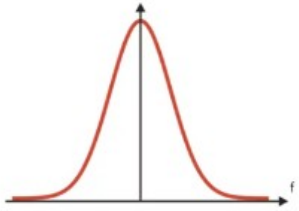
Green when:

$$K(\vec{x}, \vec{l}^1) + K(\vec{x}, \vec{l}^2) > 0$$

Red when:

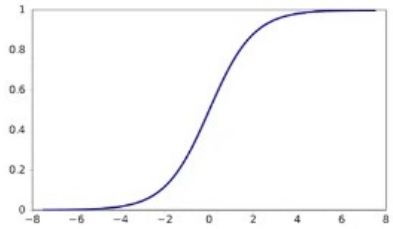
$$K(\vec{x}, \vec{l}^1) + K(\vec{x}, \vec{l}^2) = 0$$

Types of Kernels



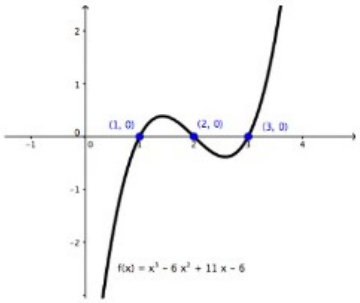
Gaussian RBF Kernel

$$K\left(\vec{x}, \vec{l}^{i}\right) = e^{-\frac{\left\|\vec{x}-\vec{l}^{i}\right\|^{2}}{2\sigma^{2}}}$$



Sigmoid Kernel

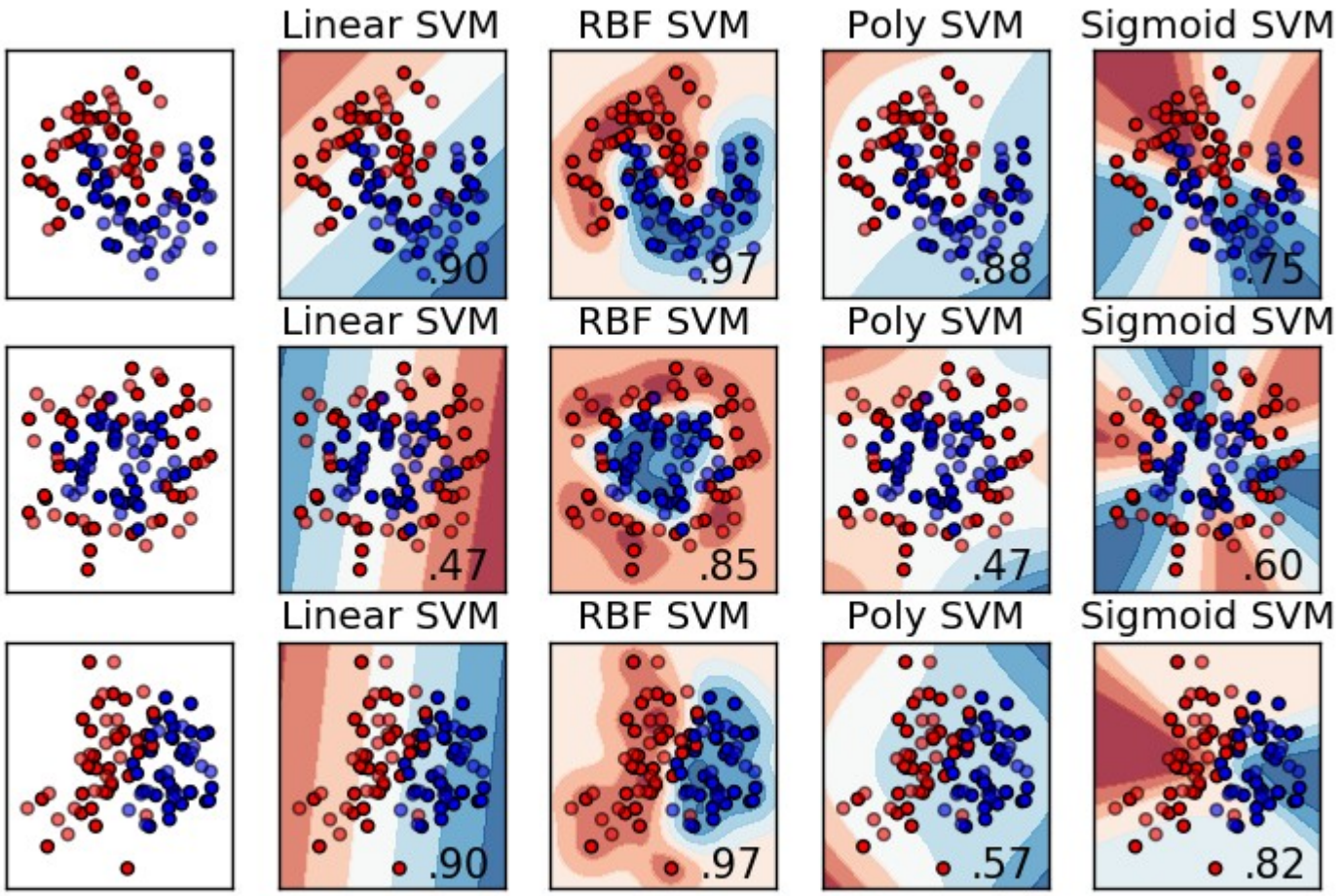
$$K(X, Y) = \tanh(\gamma \cdot X^{T}Y + r)$$



Polynomial Kernel

$$K(X, Y) = (\gamma \cdot X^{T}Y + r)^{d}, \gamma > 0$$

Types of Kernels



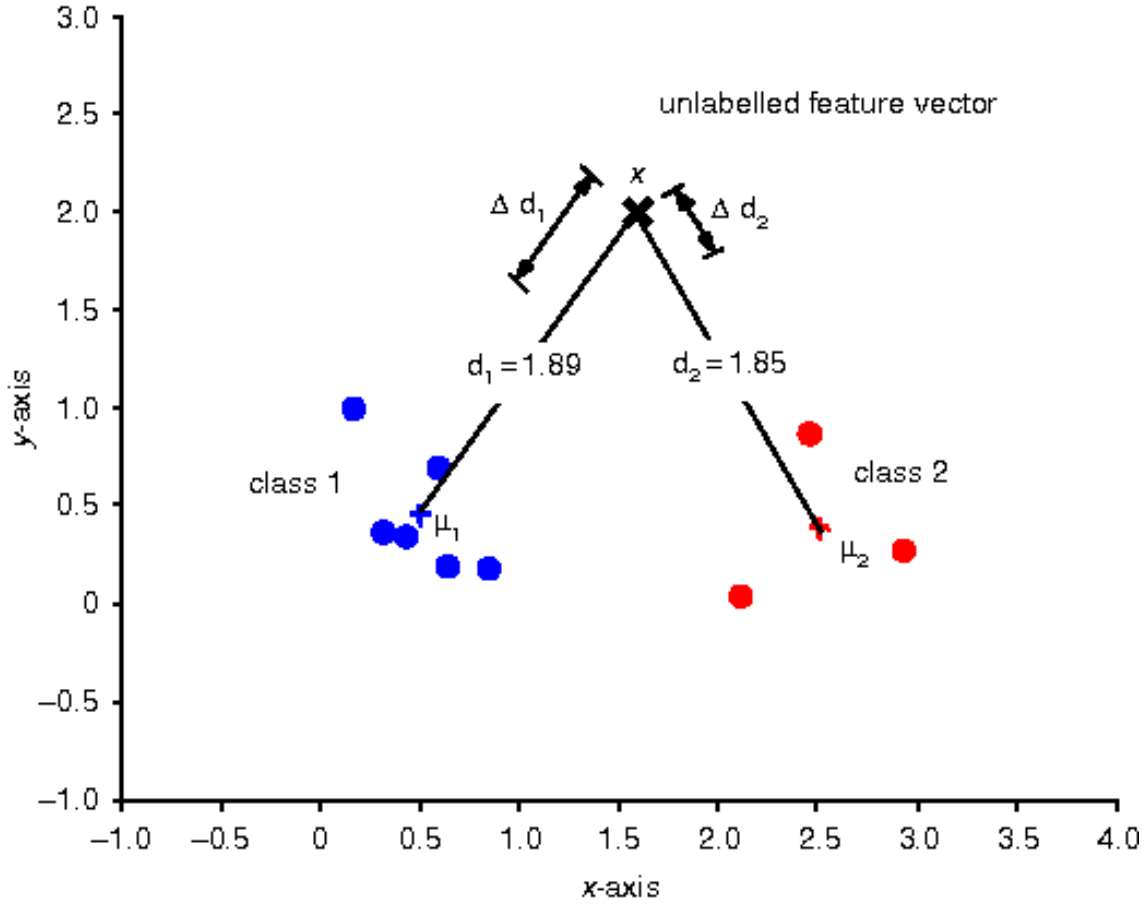
Example:

- Practical

Centroid Based Classification

- In machine learning, a nearest centroid classifier or nearest prototype classifier is a classification model that assigns to observations the label of the class of training samples whose mean (centroid) is closest to the observation.
- When applied to text classification using word vectors containing tf*idf weights to represent documents, the nearest centroid classifier is known as the Rocchio classifier because of its similarity to the Rocchio algorithm for relevance feedback.

Centroid Based Classification



Example:

- Practical

Thank you

@mitu_skillologies

@mITuSkillologies

@mitu_group

@mitu-skillologies

@MITUSkillologies

kaggle

@mituskillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

@mituskillologies

contact@mitu.co.in

tushar@tusharkute.com