Machine learning and Inferential stastitical analysis

The major difference between machine learning and statistics is their purpose. Machine learning models are designed to make the most accurate predictions possible. Statistical models are designed for inference about the relationships between variables.

Machine Learning with inferential statistics, i.e a discipline which aims to understand the underlying probability distribution of a phenomenon within a specific population. This is also what we want to do in Machine Learning in order to generate a prediction for a new element of the population.

Inferential stats relies on assumptions: the first step of the statistical method is to choose a model with unknown parameters for the underlying law governing the observed property. correlations and other statistical tools help us determine the values for the parameters of this model. Thus if your assumptions about the data are wrong, computation of the parameters will make no sense and our model will never fit in data with enough accuracy.

We can choose better hypothesis and make sure to pick the right model but there are an infinite number of possible families of distributions.

Descriptive analysis to identify the shape of the distribution of our data. But again some questions pop up in mind :-

- what if the data has many features i.e more than two ?
- How do we visualize this data to make a model proposition?
- What if we cannot identify the exact shape of the model?
- What if the subtle difference between two families of models can not be distinguished by the human eye?

The stage of creating model is the most difficult part of the inferential statistics methodology. However, that is right! This is also what we do in Machine Learning when we make out that the relationship in our data is linear and then run a linear regression.

Machine Learning methods enable us to identify complex correlations in data sets for which Inferential stats doesn't provide determination of the shape of the underlying model. We do not want to give an explicit formula for the distribution of data, rather, we want the machine to figure out the pattern on its own directly from the data using algorithms. Learning methods help us to throw off assumptions attached to the statistical methodology.

Some real-world applications for them regressions and correlations are sufficient. Sometimes we want to know the common trend of a variable against another. In this case, a simple correlation will determine the coefficient related to this trend. But how would one determine a model to classify non-linear separable data?

We can also mention the point that Machine Learning is inherent to Computer Science that allow faster learning on a huge dataset, which is not a concern of statistics. We understand that the goal of Machine Learning is not to come up with knowledge about the data ('this is the real phenomenon, this is how it works') but with a working and reproducible model for which the error tolerance is determined by the project.

Descriptive Statistics in learning techniques

What is Statistics?

Statistics is the science of collecting data and analyzing them to infer proportions (sample) that are representative of the population. In other words, statistics is interpreting data in order to make predictions for the population.

Branches of Statistics:

There are two branches of Statistics.

- DESCRIPTIVE STATISTICS : Descriptive Statistics is a statistics or a measure that describes the data.
- INFERENTIAL STATISTICS : Using a random sample of data taken from a population to describe and make inferences about the population is called Inferential Statistics.

Descriptive Statistics

Descriptive Statistics is summarizing the data at hand through certain numbers like mean, median etc. so as to make the understanding of the data easier. It does not involve any generalization or inference beyond what is available. This means that the descriptive statistics are just the representation of the data (sample) available and not based on any theory of probability.

Commonly Used Measures

1. Measures of Central Tendency

2. Measures of Dispersion (or Variability)

Measures of Central Tendency

A Measure of Central Tendency is a one number summary of the data that typically describes the center of the data. These one number summary is of three types.

1. **Mean :** Mean is defined as the ratio of the sum of all the observations in the data to the total number of observations. This is also known as Average. Thus mean is a number around which the entire data set is spread.

2. **Median :** Median is the point which divides the entire data into two equal halves. One-half of the data is less than the median, and the other half is greater than the same. Median is calculated by first arranging the data in either ascending or descending order.

- If the number of observations are odd, median is given by the middle observation in the sorted form.

- If the number of observations are even, median is given by the mean of the two middle observation in the sorted form.

An important point to note that the order of the data (ascending or descending) does not effect the median.

3. Mode : Mode is the number which has the maximum frequency in the entire data set, or in other words,mode is the number that appears the maximum number of times. A data can have one or more than one mode.

- If there is only one number that appears maximum number of times, the data has one mode, and is called **Uni-modal**.

- If there are two numbers that appear maximum number of times, the data has two modes, and is called **Bi-modal**.

- If there are more than two numbers that appear maximum number of times, the data has more than two modes, and is called **Multi-modal**.

Example to compute the Measures of Central Tendency

Consider the following data points.

17, 16, 21, 18, 15, 17, 21, 19, 11, 23

- Mean — Mean is calculated as

$$Mean = \frac{17+16+21+18+15+17+21+19+11+23}{10} = \frac{178}{10} = 17.8$$

- Median — To calculate Median, lets arrange the data in ascending order.

11, 15, 16, 17, 17, 18, 19, 21, 21, 23

Since the number of observations is even (10), median is given by the average of the two middle observations (5th and 6th here).

$$Median = \frac{5^{th}\ Obs + 6^{th}\ Obs}{2} = \frac{17+18}{2} = 17.5$$

- Mode — Mode is given by the number that occurs maximum number of times. Here, 17 and 21 both occur twice. Hence, this is a Bimodal data and the modes are 17 and 21.

Note-

1. Since Median and Mode does not take all the data points for calculations, these are robust to outliers, i.e. these are not effected by outliers.

2. At the same time, Mean shifts towards the outlier as it considers all the data points. This means if the outlier is big, mean overestimates the data and if it is small, the data is underestimated.

3. If the distribution is symmetrical, Mean = Median = Mode. Normal distribution is an example.

Measures of Dispersion (or Variability)

Measures of Dispersion describes the spread of the data around the central value (or the Measures of Central Tendency)

1. **Absolute Deviation from Mean** — The Absolute Deviation from Mean, also called Mean Absolute Deviation (MAD), describe the variation in the data set, in sense that it tells the average absolute distance of each data point in the set. It is calculated as

$$Mean\ Absolute\ Deviation = \frac{1}{N}\sum_{i=1}^{N}|X_i - \overline{X}|$$

2. Variance — Variance measures how far are data points spread out from the mean. A high variance indicates that data points are spread widely and a small variance indicates that the data points are closer to the mean of the data set. It is calculated as

$$Variance = \frac{1}{N}\sum_{i=1}^{N}(X_i - \overline{X})^2$$

3. Standard Deviation — The square root of Variance is called the Standard Deviation. It is calculated as

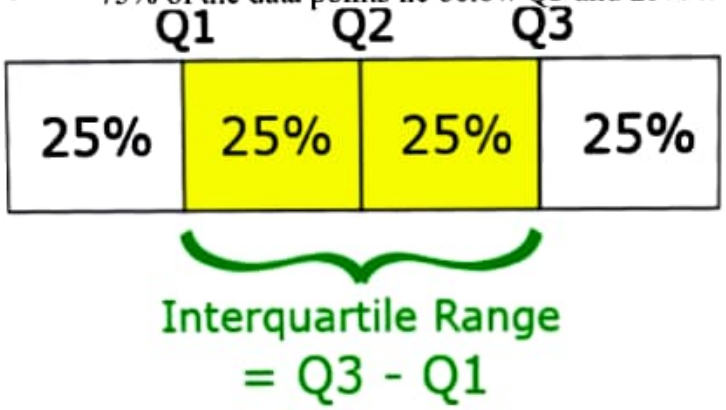
$$Std\ Deviation = \sqrt{Variance} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - \overline{X})^2}$$

4. Range — Range is the difference between the Maximum value and the Minimum value in the data set. It is given as

$$Range = Maximum - Minimum$$

5. Quartiles — Quartiles are the points in the data set that divides the data set into four equal parts. Q1, Q2 and Q3 are the first, second and third quartile of the data set.

- 25% of the data points lie below Q1 and 75% lie above it.
- 50% of the data points lie below Q2 and 50% lie above it. Q2 is nothing but Median.
- 75% of the data points lie below Q3 and 25% lie above it.



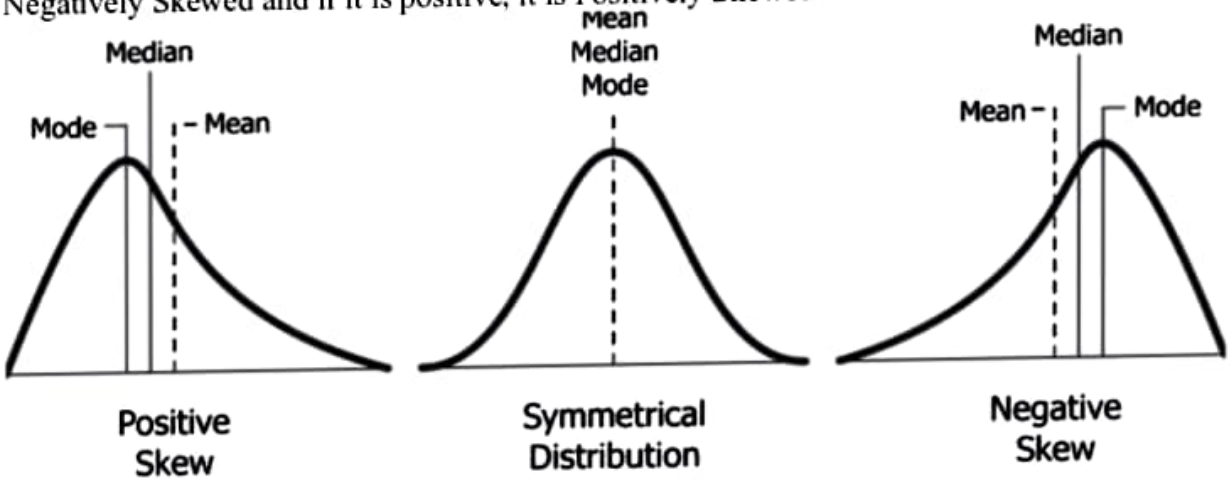
6. Skewness — The measure of asymmetry in a probability distribution is defined by Skewness. It can either be positive, negative or undefined.

- Positive Skew — This is the case when the tail on the right side of the curve is bigger than that on the left side. For these distributions, mean is greater than the mode.
- Negative Skew — This is the case when the tail on the left side of the curve is bigger than that on the right side. For these distributions, mean is smaller than the mode.

The most commonly used method of calculating Skewness is

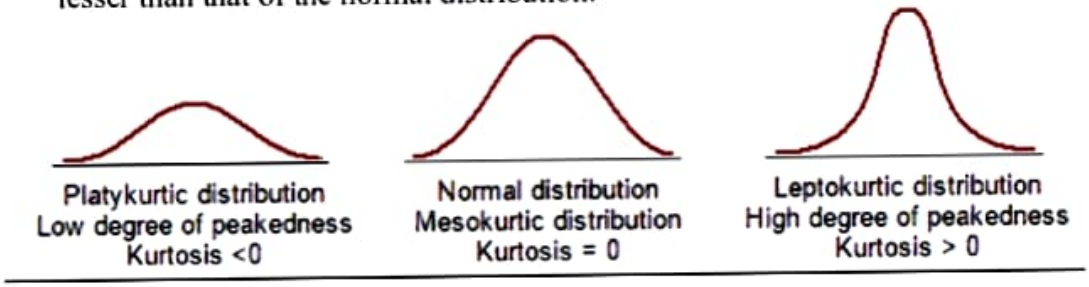
$$Skewness = \frac{3\,(Mean - Median)}{Std\ \ Deviation}$$

If the skewness is zero, the distribution is symmetrical. If it is negative, the distribution is Negatively Skewed and if it is positive, it is Positively Skewed.



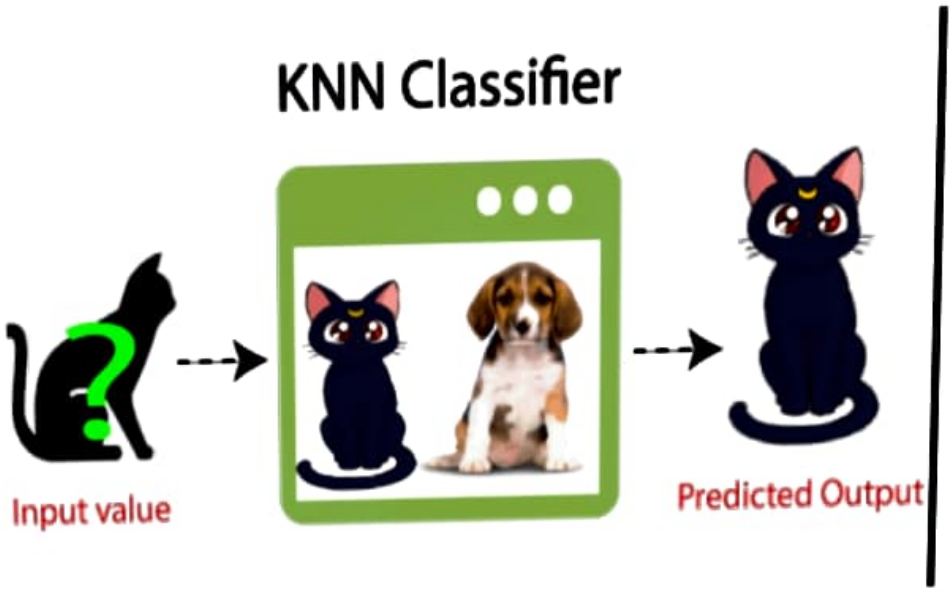
7. Kurtosis — Kurtosis describes the whether the data is light tailed (lack of outliers) or heavy tailed (outliers present) when compared to a Normal distribution. There are three kinds of Kurtosis:

- Mesokurtic — This is the case when the kurtosis is zero, similar to the normal distributions.
- Leptokurtic — This is when the tail of the distribution is heavy (outlier present) and kurtosis is higher than that of the normal distribution.
- Platykurtic — This is when the tail of the distribution is light(no outlier) and kurtosis is lesser than that of the normal distribution.



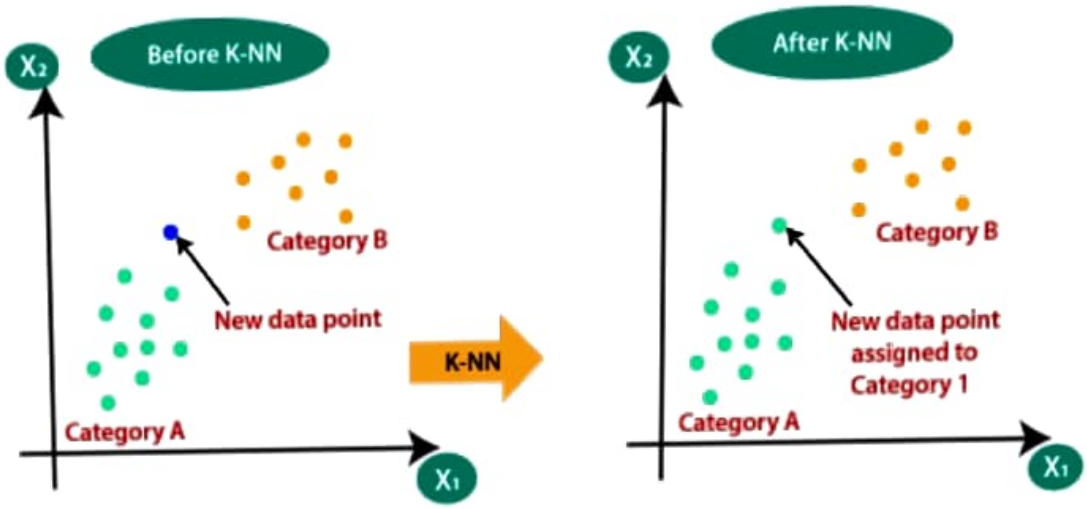
K-Nearest Neighbor(KNN) Algorithm

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



Explanation of K-NN Algorithm

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset. Consider the below diagram:



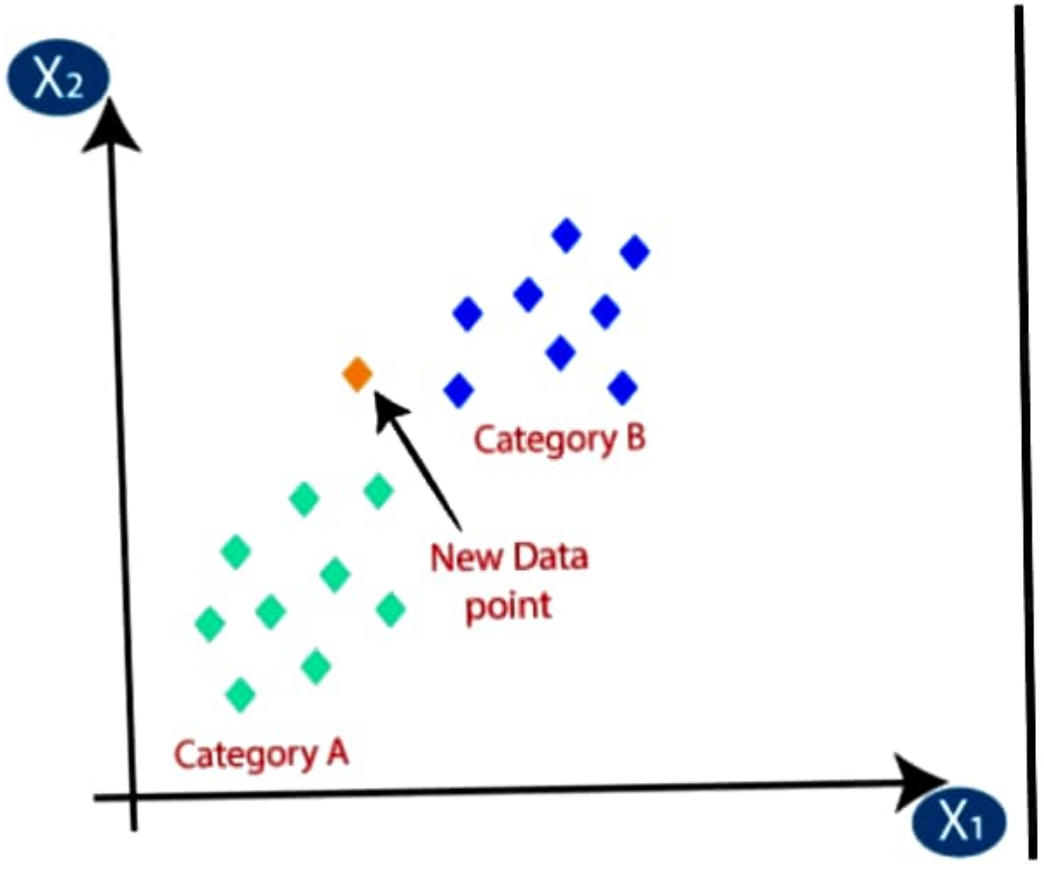
K-NN working

The K-NN working can be explained on the basis of the below algorithm:

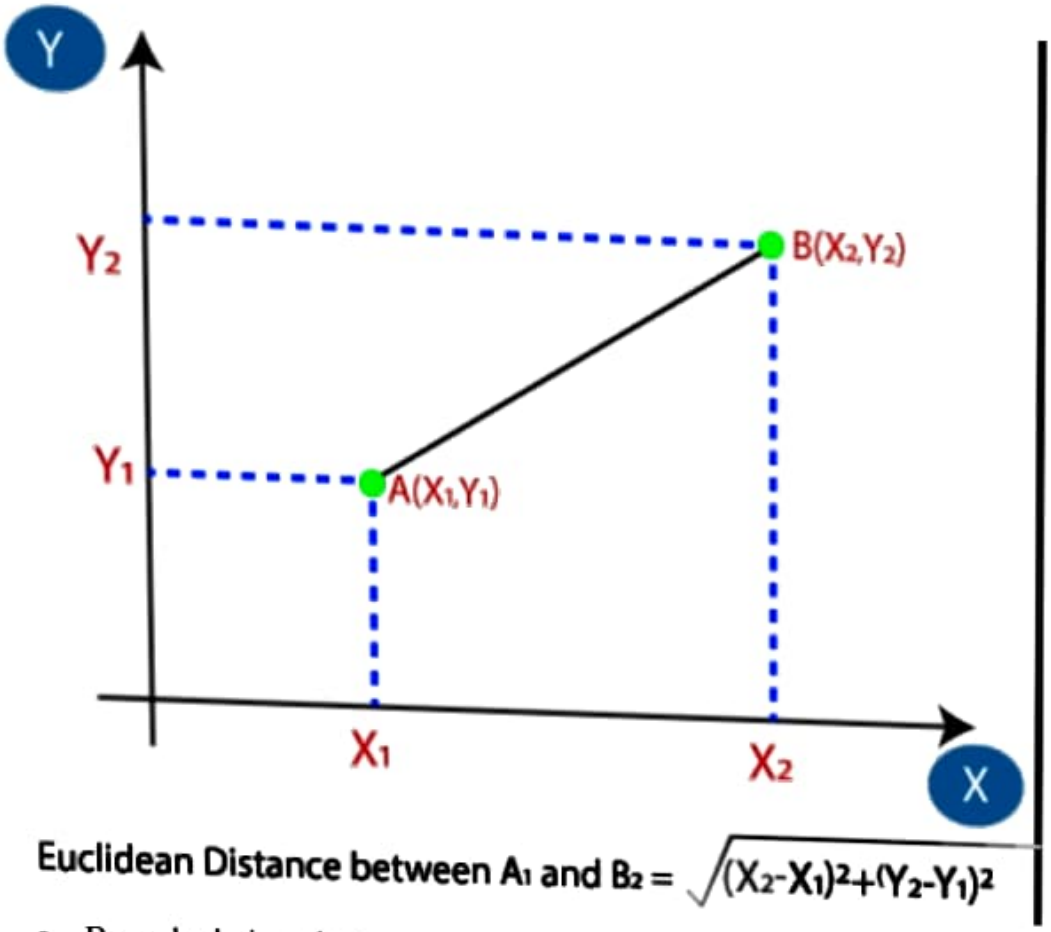
- **Step-1:** Select the number K of the neighbors

- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our model is ready.

Suppose we have a new data point and we need to put it in the required category. Consider the below image:

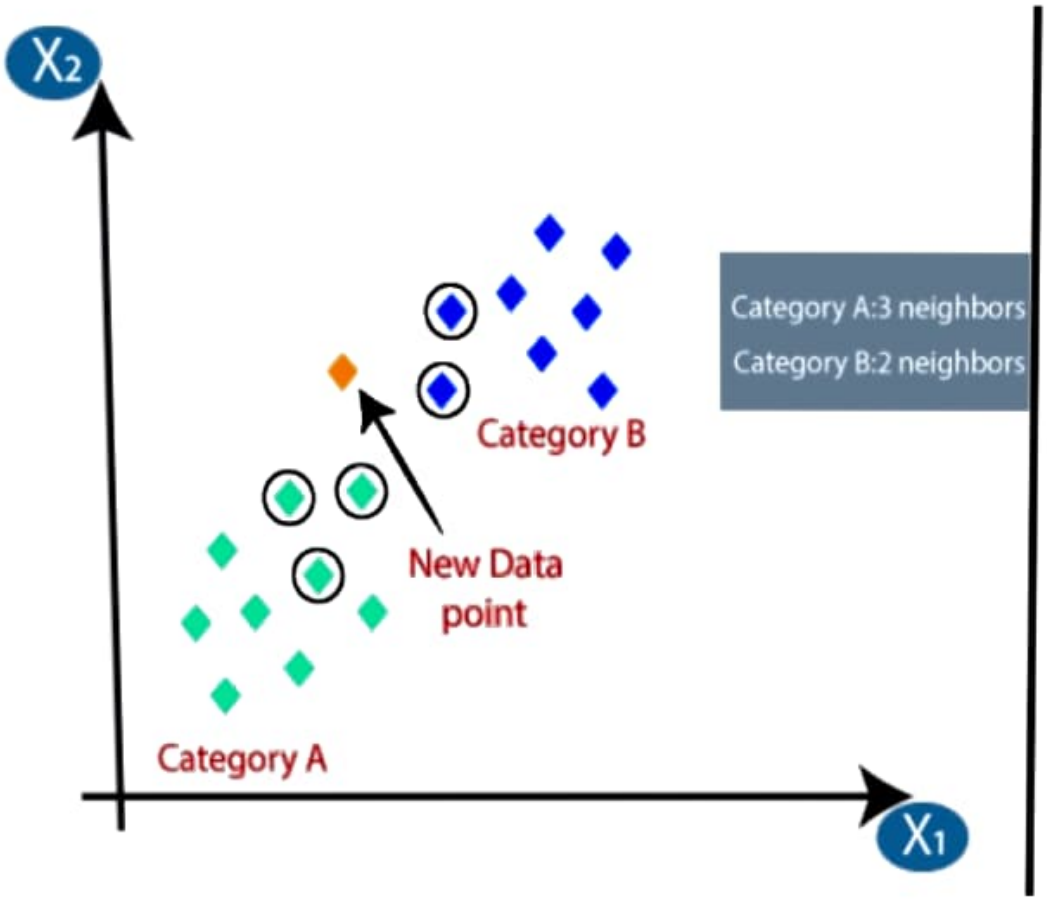


- Firstly, we will choose the number of neighbors, so we will choose the k=5.
- Next, we will calculate the **Euclidean distance** between the data points. The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:



Euclidean Distance between A_1 and B_2 = $\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$

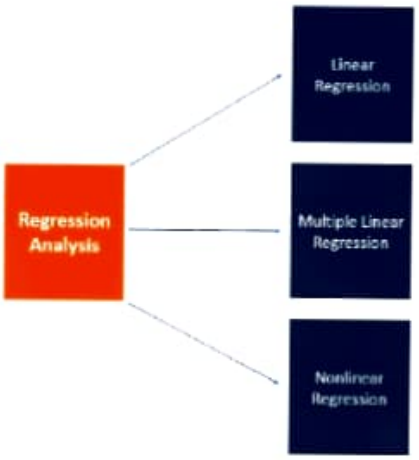
- By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B. Consider the below image:



- As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

Regression Functions

Regression analysis is a set of statistical methods used for the estimation of relationships between a dependent variable and one or more independent variables. It can be utilized to assess the strength of the relationship between variables and for modeling the future relationship between them.



Regression analysis includes several variations, such as linear, multiple linear, and nonlinear. The most common models are simple linear and multiple linear. Nonlinear regression analysis is commonly used for more complicated data sets in which the dependent and independent variables show a nonlinear relationship.

Regression Analysis – Linear Model Assumptions

Linear regression analysis is based on six fundamental assumptions:

1. The dependent and independent variables show a linear relationship between the slope and the intercept.
2. The independent variable is not random.
3. The value of the residual (error) is zero.
4. The value of the residual (error) is constant across all observations.
5. The value of the residual (error) is not correlated across all observations.
6. The residual (error) values follow the normal distribution.

Regression Analysis – Simple Linear Regression

Simple linear regression is a model that assesses the relationship between a dependent variable and an independent variable. The simple linear model is expressed using the following equation:

$$Y = a + bX + \epsilon$$

Where:

- **Y** – Dependent variable
- **X** – Independent (explanatory) variable
- **a** – Intercept
- **b** – Slope
- **ϵ** – Residual (error)

Regression Analysis – Multiple Linear Regression

Multiple linear regression analysis is essentially similar to the simple linear model, with the exception that multiple independent variables are used in the model. The mathematical representation of multiple linear regression is:

$$Y = a + bX_1 + cX_2 + dX_3 + \epsilon$$

Where:

- **Y** – Dependent variable
- X_1, X_2, X_3 – Independent (explanatory) variables
- **a** – Intercept
- **b, c, d** – Slopes
- ϵ – Residual (error)

Multiple linear regression follows the same conditions as the simple linear model. However, since there are several independent variables in multiple linear analysis, there is another mandatory condition for the model:

- **Non-collinearity:** Independent variables should show a minimum correlation with each other. If the independent variables are highly correlated with each other, it will be difficult to assess the true relationships between the dependent and independent variables.

Regression Analysis in Finance

Regression analysis comes with several applications in finance. For example, the statistical method is fundamental to the Capital Asset Pricing Model (CAPM). Essentially, the CAPM equation is a model that determines the relationship between the expected return of an asset and the market risk premium.

The analysis is also used to forecast the returns of securities, based on different factors, or to forecast the performance of a business. Learn more forecasting methods in CFI's Budgeting and Forecasting Course!

1. Beta and CAPM

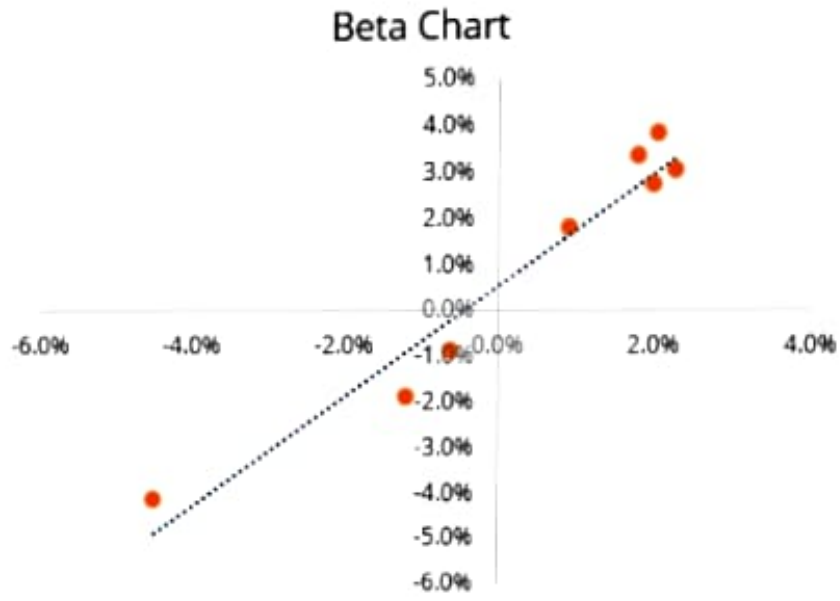
In finance, regression analysis is used to calculate the Beta (volatility of returns relative to the overall market) for a stock. It can be done in Excel using the Slope function.

Beta (β) Calculator

Individual Stock		
Date	**Price**	**Return**
1/2/2018	15.78	
1/9/2018	16.38	3.8%
1/16/2018	16.67	1.8%
1/23/2018	17.17	3.0%
1/30/2018	17.02	-0.9%
2/6/2018	16.31	-4.2%
2/13/2018	16.00	-1.9%
2/20/2018	16.43	2.7%
2/27/2018	16.97	3.3%

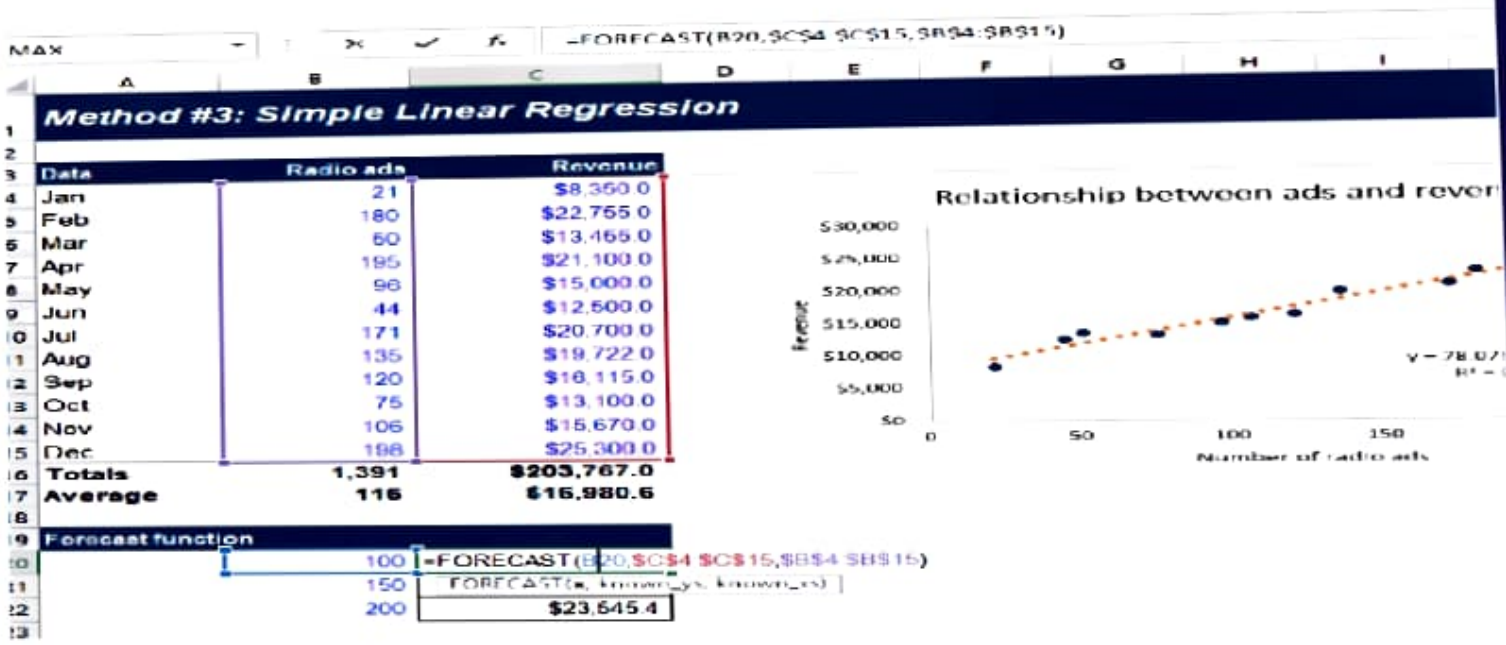
S&P 500 Index		
Date	**Price**	**Return**
1/2/2018	2,696	
1/9/2018	2,751	2.0%
1/16/2018	2,776	0.9%
1/23/2018	2,839	2.3%
1/30/2018	2,822	-0.6%
2/6/2018	2,695	-4.5%
2/13/2018	2,663	-1.2%
2/20/2018	2,716	2.0%
2/27/2018	2,765	1.8%

Beta (β)	1.21



2. Forecasting Revenues and Expenses

When forecasting financial statements for a company, it may be useful to do a multiple regression analysis to determine how changes in certain assumptions or drivers of the business will impact revenue or expenses in the future. For example, there may be a very high correlation between the number of salespeople employed by a company, the number of stores they operate, and the revenue the business generates.



The above example shows how to use the Forecast function in Excel to calculate a company's revenue, based on the number of ads it runs.

Regression Tools

Excel remains a popular tool to conduct basic regression analysis in finance, however, there are many more advanced statistical tools that can be used.

Python and R are both powerful coding languages that have become popular for all types of financial modeling, including regression. These techniques form a core part of data science and machine learning where models are trained to detect these relationships in data.

Lenier Regression with least square error

The least-squares regression method is a technique commonly used in Regression Analysis. It is a mathematical method used to find the best fit line that represents the relationship between an independent and dependent variable.

To understand the least-squares regression method lets get familiar with the concepts involved in formulating the line of best fit.

Line Of Best Fit?

Line of best fit is drawn to represent the relationship between 2 or more variables. To be more specific, the best fit line is drawn across a scatter plot of data points in order to represent a relationship between those data points.

Regression analysis makes use of mathematical methods such as least squares to obtain a definite relationship between the predictor variable (s) and the target variable. The least-squares method is one of the most effective ways used to draw the line of best fit. It is based on the idea that the square of the errors obtained must be minimized to the most possible extent and hence the name least squares method.

If we were to plot the best fit line that shows the depicts the sales of a company over a period of time, it would look something like this:

Notice that the line is as close as possible to all the scattered data points. This is what an ideal best fit line looks like.

To better understand the whole process let's see how to calculate the line using the Least Squares Regression.

Steps to calculate the Line of Best Fit

To start constructing the line that best depicts the relationship between variables in the data, we first need to get our basics right. Take a look at the equation below:

$$y = mx + c$$

It is a simple equation that represents a straight line along 2 Dimensional data, i.e. x-axis and y-axis. To better understand this, let's break down the equation:

- y: dependent variable
- m: the slope of the line
- x: independent variable
- c: y-intercept

So the aim is to calculate the values of slope, y-intercept and substitute the corresponding 'x' values in the equation in order to derive the value of the dependent variable.

Let's see how this can be done.

As an assumption, let's consider that there are 'n' data points.

Step 1: *Calculate the slope 'm' by using the following formula:*

$$m = \frac{n\sum xy - (\Sigma x)(\Sigma y)}{n\Sigma x^2 - (\Sigma x)^2}$$

Step 2: *Compute the y-intercept (the value of y at the point where the line crosses the y-axis):*

$$c = y - mx$$

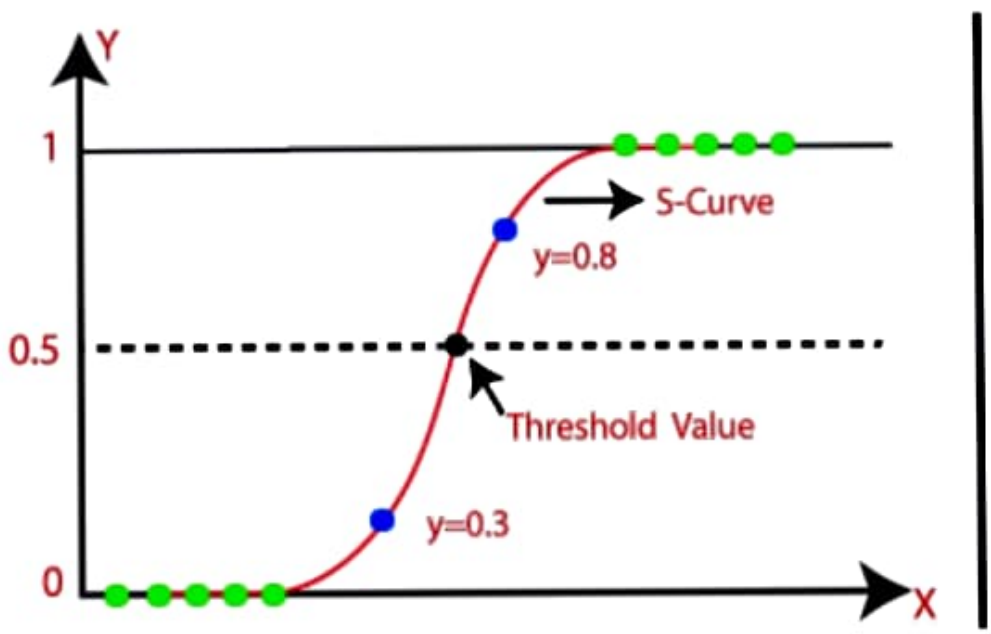
Step 3: *Substitute the values in the final equation:*

$$y = mx + c$$

Logistic Regression for classification tasks

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

- Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, **it gives the probabilistic values which lie between 0 and 1**.
- Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas **Logistic regression is used for solving the classification problems**.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.
- Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:



Supervised Machine Learning algorithm can be broadly classified into Regression and Classification Algorithms. In Regression algorithms, we have predicted the output for continuous values, but to predict the categorical values, we need Classification algorithms.

Classification Algorithm

The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations on the basis of training data. In Classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, **Yes or No, 0 or 1, Spam or Not Spam, cat or dog,** etc. Classes can be called as targets/labels or categories.

Unlike **regression,** the output variable of Classification is a category, not a value, such as "Green or Blue", "fruit or animal", etc. Since the Classification algorithm is a Supervised learning technique, hence it takes labeled input data, which means it contains input with the corresponding output.

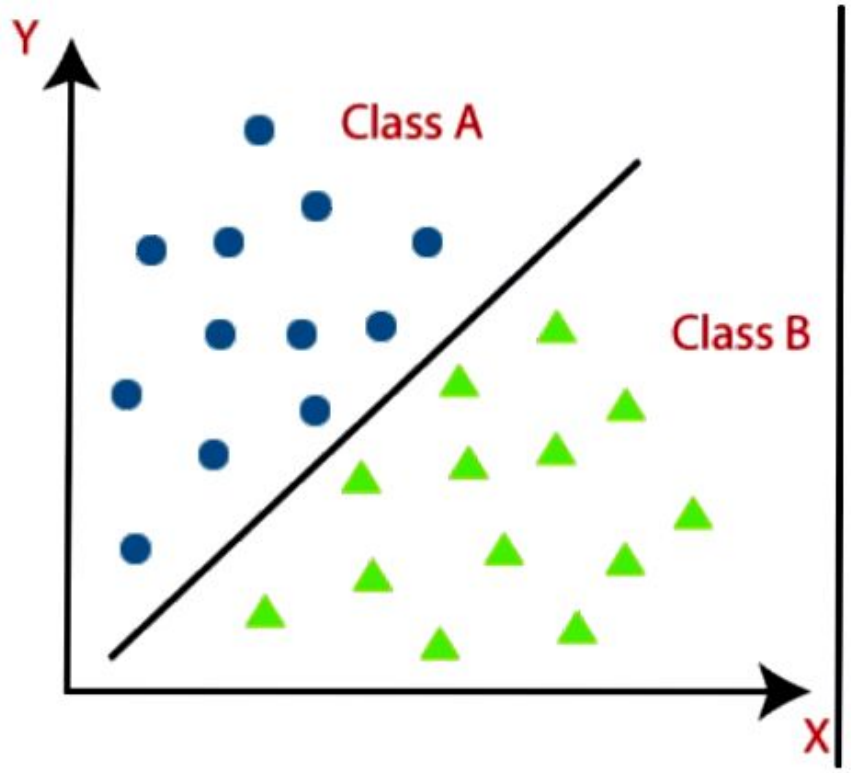
In classification algorithm, a discrete output function(y) is mapped to input variable(x).

1. $y=f(x)$, where y = categorical output

The best example of an ML classification algorithm is **Email Spam Detector**.

The main goal of the Classification algorithm is to identify the category of a given dataset, and these algorithms are mainly used to predict the output for the categorical data.

Classification algorithms can be better understood using the below diagram. In the below diagram, there are two classes, class A and Class B. These classes have features that are similar to each other and dissimilar to other classes.



The algorithm which implements the classification on a dataset is known as a classifier. There are two types of Classifications:

- **Binary Classifier:** If the classification problem has only two possible outcomes, then it is called as Binary Classifier.
  Examples: YES or NO, MALE or FEMALE, SPAM or NOT SPAM, CAT or DOG, etc.
- **Multi-class Classifier:** If a classification problem has more than two outcomes, then it is called as Multi-class Classifier.
  Example: Classifications of types of crops, Classification of types of music.

Classification Problems:

In the classification problems, there are two types of learners:

1. **Lazy Learners:** Lazy Learner firstly stores the training dataset and wait until it receives the test dataset. In Lazy learner case, classification is done on the basis of the most related data stored in the training dataset. It takes less time in training but more time for predictions.
   Example: K-NN algorithm, Case-based reasoning

2. **Eager Learners:**Eager Learners develop a classification model based on a training dataset before receiving a test dataset. Opposite to Lazy learners, Eager Learner takes more time in learning, and less time in prediction. **Example:** Decision Trees, Naïve Bayes, ANN.

Types of ML Classification Algorithms:

Classification Algorithms can be further divided into the Mainly two category:

- **Linear Models**
  - Logistic Regression
  - Support Vector Machines
- **Non-linear Models**
  - K-Nearest Neighbours
  - Kernel SVM
  - Naïve Bayes
  - Decision Tree Classification
  - Random Forest Classification

Evaluating a Classification model:

Once our model is completed, it is necessary to evaluate its performance; either it is a Classification or Regression model. So for evaluating a Classification model, we have the following ways:

1. Log Loss or Cross-Entropy

- It is used for evaluating the performance of a classifier, whose output is a probability value between the 0 and 1.
- For a good binary Classification model, the value of log loss should be near to 0.
- The value of log loss increases if the predicted value deviates from the actual value.
- The lower log loss represents the higher accuracy of the model.
- For Binary classification, cross-entropy can be calculated as:

1. ?(ylog(p)+(1?y)log(1?p))

Where y= Actual output, p= predicted output.

2. Confusion Matrix:

- The confusion matrix provides us a matrix/table as output and describes the performance of the model.
- It is also known as the error matrix.
- The matrix consists of predictions result in a summarized form, which has a total number of correct predictions and incorrect predictions. The matrix looks like as below table:

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{Total Population}}$$

3. AUC-ROC curve:

- ROC curve stands for **Receiver Operating Characteristics Curve** and AUC stands for **Area Under the Curve**.
- It is a graph that shows the performance of the classification model at different thresholds.
- To visualize the performance of the multi-class classification model, we use the AUC-ROC Curve.
- The ROC curve is plotted with TPR and FPR, where TPR (True Positive Rate) on Y-axis and FPR(False Positive Rate) on X-axis.

Uses of Classification Algorithms

Classification algorithms can be used in different places. Below are some popular use cases of Classification Algorithms:

- Email Spam Detection

- Speech Recognition
- Identifications of Cancer tumor cells.
- Drugs Classification
- Biometric Identification, etc.

Fisher's Linear Discriminant and thersholding for classification

We can view linear classification models in terms of dimensionality reduction.

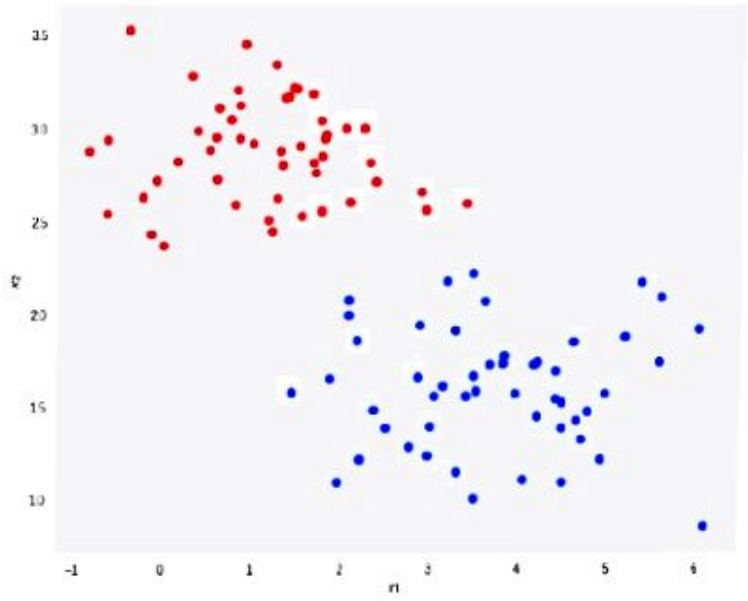
To begin, consider the case of a two-class classification problem **(K=2)**. Blue and red points in R^2. In general, we can take any D-dimensional input vector and project it down to D'-dimensions. Here, **D** represents the original input dimensions while **D'** is the projected space dimensions. Throughout this article, consider **D'** less than **D**.

In the case of projecting to one dimension (the number line), i.e. **D'=1**, we can pick a threshold **t** to separate the classes in the new space. Given an input vector **x**:

- if the predicted value $y >= t$ then, **x** belongs to class C1 (class 1) - where $y = W^T x$.
- otherwise, it is classified as C2 (class 2).

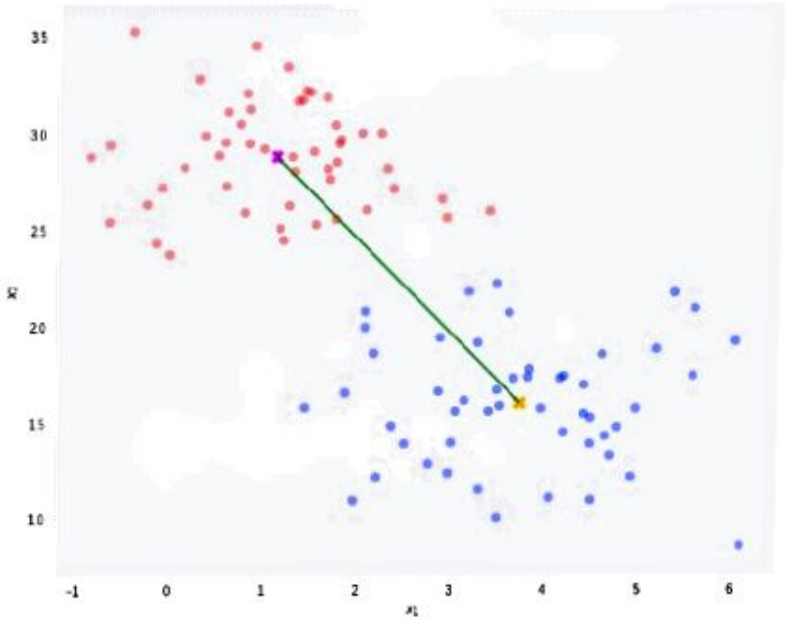
Take the dataset below as a toy example. We want to reduce the original data dimensions from D=2 to D'=1. In other words, we want a transformation T that maps vectors in 2D to 1D-T(v) = $\mathbb{R}^2 \rightarrow \mathbb{R}^1$.

First, let's compute the mean vectors **m1** and **m2** for the two classes.



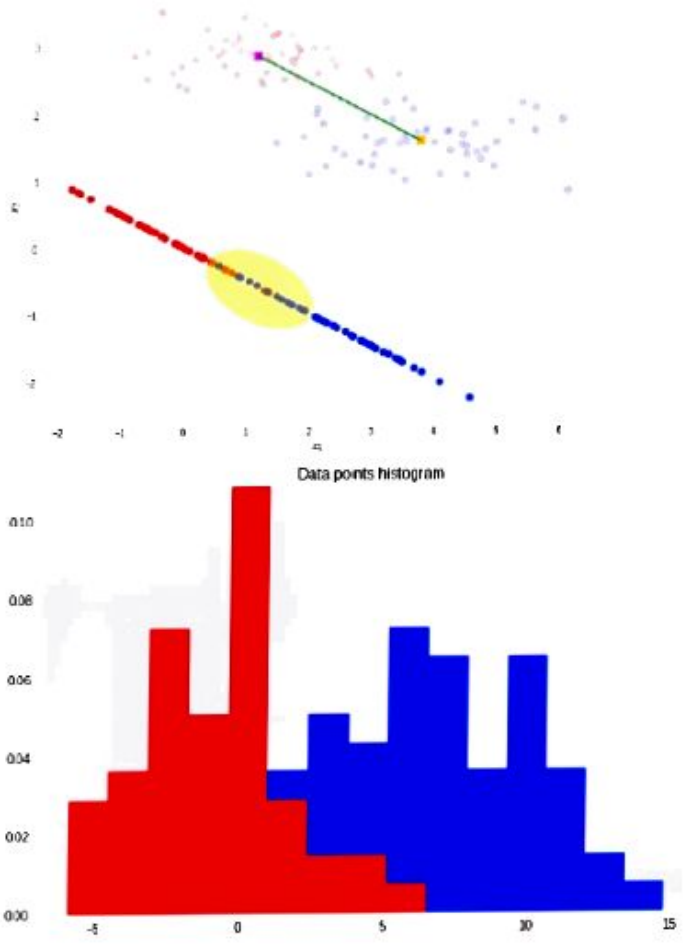
$$m_1 = \frac{1}{N_1} \sum_{n \in C1} x_n \qquad m_2 = \frac{1}{N_2} \sum_{n \in C2} x_n \qquad (1)$$

Note that N1 and N2 denote the number of points in classes C1 and C2 respectively. Now, consider using the class means as a measure of separation. In other words, we want to project the data onto the vector **W** joining the 2 class means.



It is important to note that any kind of projection to a smaller dimension might involve some loss of information. In this scenario, note that the two classes are clearly separable (by a line) in their original space.

However, after re-projection, the data exhibit some sort of class overlapping-shown by the yellow ellipse on the plot and the histogram below.



That is where the Fisher's Linear Discriminant comes into play.

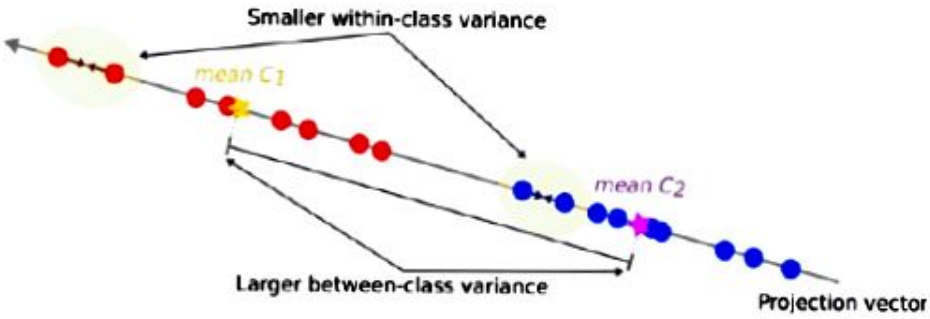
The idea proposed by Fisher is to maximize a function that will give a large separation between the projected class means while also giving a small variance within each class, thereby minimizing the class overlap.

In other words, FLD selects a projection that maximizes the class separation. To do that, it maximizes the ratio between the between-class variance to the within-class variance.

In short, to project the data to a smaller dimension and to avoid class overlapping, FLD maintains 2 properties.

- A large variance among the dataset classes.
- A small variance within each of the dataset classes.

Note that a large between-class variance means that the projected class averages should be as far apart as possible. On the contrary, a small within-class variance has the effect of keeping the projected data points closer to one another.



To find the projection with the following properties, FLD learns a weight vector W with the following criterion.

$$J(\boldsymbol{W}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} \qquad (1)$$

Between-class variance

Within-class variance

If we substitute the mean vectors **m1** and **m2** as well as the variance **s** as given by equations (1) and (2) we arrive at equation (3). If we take the derivative of (3) w.r.t **W** (after some simplifications) we get the learning equation for **W** (equation 4). That is, W (our desired transformation) is directly proportional to the inverse of the within-class covariance matrix times the difference of the class means.

BAYESIAN LEARNING

Bayesian reasoning provides a probabilistic approach to inference. It is based on the assumption that the quantities of interest are governed by probability distributions and that optimal decisions can be made by reasoning about these probabilities together with observed data

INTRODUCTION

Bayesian learning methods are relevant to study of machine learning for two different reasons.

1. First, Bayesian learning algorithms that calculate explicit probabilities for hypotheses, such as the naive Bayes classifier, are among the most practical approaches to certain types of learning problems
2. The second reason is that they provide a useful perspective for understanding many learning algorithms that do not explicitly manipulate probabilities.

Features of Bayesian Learning Methods

- Each observed training example can incrementally decrease or increase the estimated probability that a hypothesis is correct. This provides a more flexible approach to learning than algorithms that completely eliminate a hypothesis if it is found to be inconsistent with any single example
- Prior knowledge can be combined with observed data to determine the final probability of a hypothesis. In Bayesian learning, prior knowledge is provided by asserting (1) a prior probability for each candidate hypothesis, and (2) a probability distribution over observed data for each possible hypothesis.
- Bayesian methods can accommodate hypotheses that make probabilistic predictions
- New instances can be classified by combining the predictions of multiple hypotheses, weighted by their probabilities.
- Even in cases where Bayesian methods prove computationally intractable, they can provide a standard of optimal decision making against which other practical methods can be measured.

Practical difficulty in applying Bayesian methods

1. One practical difficulty in applying Bayesian methods is that they typically require initial knowledge of many probabilities. When these probabilities are not known in advance they are often estimated based on background knowledge, previously available data, and assumptions about the form of the underlying distributions.
2. A second practical difficulty is the significant computational cost required to determine the Bayes optimal hypothesis in the general case. In certain specialized situations, this computational cost can be significantly reduced.

BAYES THEOREM

Bayes theorem provides a way to calculate the probability of a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself.

Notations

- P(h) prior probability of h, reflects any background knowledge about the chance that h is correct
- P(D) prior probability of D, probability that D will be observed
- P(D|h) probability of observing D given a world in which h holds
- P(h|D) posterior probability of h, reflects confidence that h holds after D has been observed

Bayes theorem is the cornerstone of Bayesian learning methods because it provides a way to calculate the posterior probability P(h|D), from the prior probability P(h), together with P(D) and P(D|h).

Bayes Theorem:

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- P(h|D) increases with P(h) and with P(D|h) according to Bayes theorem.
- P(h|D) decreases as P(D) increases, because the more probable it is that D will be observed independent of h, the less evidence D provides in support of h.

Maximum a Posteriori (MAP) Hypothesis

- In many learning scenarios, the learner considers some set of candidate hypotheses H and is interested in finding the most probable hypothesis h ∈ H given the observed data D. Any such maximally probable hypothesis is called a maximum a posteriori (MAP) hypothesis.
- Bayes theorem to calculate the posterior probability of each candidate hypothesis is h_{MAP} is a MAP hypothesis provided

$$\begin{aligned} h_{MAP} &= \underset{h \in H}{argmax}\ P(h|D) \\ &= \underset{h \in H}{argmax}\ \frac{P(D|h)P(h)}{P(D)} \\ &= \underset{h \in H}{argmax}\ P(D|h)P(h) \end{aligned}$$

- P(D) can be dropped, because it is a constant independent of h

Maximum Likelihood (ML) Hypothesis

- In some cases, it is assumed that every hypothesis in H is equally probable a priori ($P(h_i) = P(h_j)$ for all h_i and h_j in H).
- In this case the below equation can be simplified and need only consider the term P(D|h) to find the most probable hypothesis.

$$h_{MAP} = \underset{h \in H}{argmax}\ P(D|h)P(h)$$

the equation can be simplified

$$h_{ML} = \underset{h \in H}{argmax}\ P(D|h)$$

P(D|h) is often called the likelihood of the data D given h, and any hypothesis that maximizes P(D|h) is called a maximum likelihood (ML) hypothesis

Example

- Consider a medical diagnosis problem in which there are two alternative hypotheses: (1) that the patient has particular form of cancer, and (2) that the patient does not. The available data is from a particular laboratory test with two possible outcomes: + (positive) and - (negative).

- We have prior knowledge that over the entire population of people only .008 have this disease. Furthermore, the lab test is only an imperfect indicator of the disease.
- The test returns a correct positive result in only 98% of the cases in which the disease is actually present and a correct negative result in only 97% of the cases in which the disease is not present. In other cases, the test returns the opposite result.
- The above situation can be summarized by the following probabilities:

$$\begin{aligned} P(cancer) &= .008 & P(\neg cancer) &= 0.992 \\ P(\oplus|cancer) &= .98 & P(\ominus|cancer) &= .02 \\ P(\oplus|\neg cancer) &= .03 & P(\ominus|\neg cancer) &= .97 \end{aligned}$$

Suppose a new patient is observed for whom the lab test returns a positive (+) result. Should we diagnose the patient as having cancer or not?

$$P(\ominus|cancer)P(cancer) = (.98).008 = .0078$$
$$P(\oplus|\neg cancer)P(\neg cancer) = (.03).992 = .0298$$
$$\Rightarrow h_{MAP} = \neg cancer$$

The exact posterior probabilities can also be determined by normalizing the above quantities so that they sum to 1

$$P(cancer|\oplus) = \frac{0.0078}{0.0078 + 0.0298} = 0.21$$

$$P(\neg cancer|\oplus) = \frac{0.0298}{0.0078 + 0.0298} = 0.79$$

Basic formulas for calculating probabilities are summarized in Table

---

- ***Product rule***: **probability $P(A \wedge B)$ of a conjunction of two events A and B**

$$P(A \wedge B) = P(A|B)P(B) = P(B|A)P(A)$$

- ***Sum rule***: **probability of a disjunction of two events A and B**

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

- ***Bayes theorem***: **the posterior probability $P(h|D)$ of h given D**

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- ***Theorem of total probability***: **if events $A_1, \ldots, A_n$ are mutually exclusive with $\sum_{i=1}^{n} P(A_i) = 1$, then**

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$
