UNIT-5
PART-1
TRANSFORM SUPERSTEP

UNIVARIATE ANALYSIS

This type of data consists of **only one variable**. The analysis of univariate data is thus the simplest form of analysis since the information deals with only one quantity that changes. It does not deal with causes or relationships and the main purpose of the analysis is to describe the data and find patterns that exist within it. The example of a univariate data can be height.

Heights (in cm)	164	167.3	170	174.2	178	180	186

BIVARIATE ANALYSIS

This type of data involves **two different variables**. The analysis of this type of data deals with causes and relationships and the analysis is done to find out the relationship among the two variables. Example of bivariate data can be temperature and ice cream sales in summer season.

TEMPERATURE(IN CELSIUS)	ICE CREAM SALES
20	2000
25	2500
35	5000
43	7800

Here, the relationship is visible from the table that temperature and sales are directly proportional to each other and thus related because as the temperature increases, the sales also increase.

MULTIVARIATE ANALYSIS

When the data involves **three or more variables**, it is categorized under multivariate. Example of this type of data is suppose an advertiser wants to compare the popularity of four advertisements on a website, then their click rates could be measured for both men and women and relationships between variables can then be examined. Some of the techniques are regression analysis, path analysis, factor analysis and multivariate analysis of variance (MANOVA).

Univariate	Bivariate	Multivariate
It only summarize single variable at a time.	It only summarize two variables	It only summarize more than 2 variables.
It does not deal with causes and relationships.	It does deal with causes and relationships and analysis is done.	It does not deal with causes and relationships and analysis is done.
It does not contain any dependent variable.	It does contain only one dependent variable.	It is similar to bivariate but it contains more than 2 variables.
The main purpose is to describe.	The main purpose is to explain.	The main purpose is to study the relationship among them.
The example of a univariate can be height.	The example of bivariate can be temperature and ice sales in summer vacation.	Example, Suppose an advertiser wants to compare the popularity of four advertisements on a website. Then their click rates could be measured for both men and women and relationships between variable can be examined

LINEAR REGRESSION

- Linear regression is a type of statistical analysis used to predict the relationship between two variables. it assumes a linear relationship between the independent variable and the dependent variable, and aims to find the best-fitting line that describes the relationship.
- The line is determined by minimizing the sum of the squared differences between the predicted values and the actual values.

REAL-LIFE APPLICATIONS OF LINEAR REGRESSION

- BUSINESS
- DEMAND FORECASTING
- MEDICAL

CONCEPT OF DEPENDENT AND INDEPENDENT VARIABLES

Independent variables and dependent variables are the two fundamental types of variables in statistical modeling and experimental designs. Analysts use these methods to understand the relationships between the variables and estimate effect sizes. What effect does one variable have on another?

What is an Independent Variable?

Independent variables (IVs) are the ones that you include in the model to explain or predict changes in the dependent variable. The name helps you understand their role in statistical analysis. These variables are *independent*. In this context, independent indicates that they stand alone and other variables in the model do not influence them. The researchers are not seeking to understand what causes the independent variables to change.

Independent variables are also known as predictors, factors, treatment variables, explanatory variables, input variables, x-variables, and right-hand variables—because they appear on the right side of the equals sign in a regression equation. In notation, statisticians commonly denote them using Xs. On graphs, analysts place independent variables on the horizontal, or X, axis.

What is a Dependent Variable?

The dependent variable (DV) is what you want to use the model to explain or predict. The values of this variable *depend* on other variables. It is the outcome that you're studying. It's also known as the response variable, outcome variable, and left-hand variable. Statisticians commonly denote them using a Y. Traditionally, graphs place dependent variables on the vertical, or Y, axis.

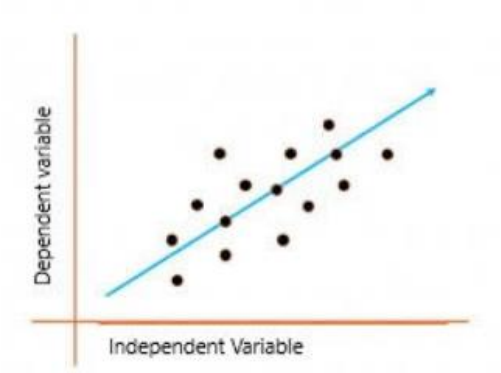
What is a Dependent Variable?

The dependent variable (DV) is what you want to use the model to explain or predict. The values of this variable *depend* on other variables. It is the outcome that you're studying. It's also known as the response variable, outcome variable, and left-hand variable. Statisticians commonly denote them using a Y. Traditionally, graphs place dependent variables on the vertical, or Y, axis.

For example, in the plant growth study example, a measure of plant growth is the dependent variable. That is the outcome of the experiment, and we want to determine what affects it.

SIMPLE LINEAR REGRESSION

- In a simple linear regression, there is one independent variable and one dependent variable. The model estimates the slope and intercept of the line of best fit, which represents the relationship between the variables.
- Linear regression shows the linear relationship between the independent(predictor) variable i.e. x-axis and the dependent(output) variable i.e. y-axis, called linear regression.

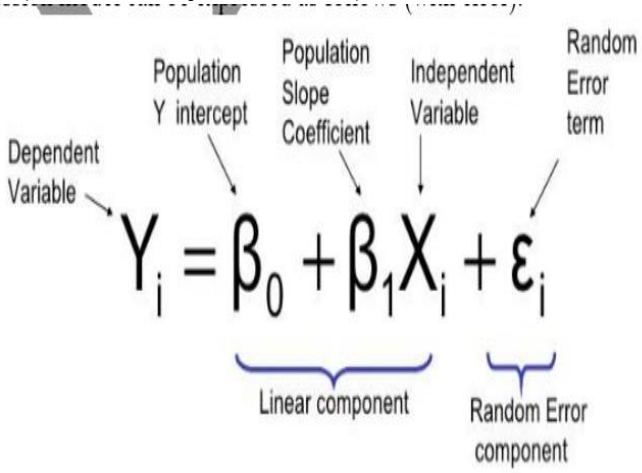


The graph above presents the linear relationship between the output(y) and predictor(X) variables.

The blue line is referred to as the *best-fit* straight line. Based on the given data points, we attempt to plot a line that fits the points the best.

EQUATION FOR THE LINEAR REGRESSION LINE

- Y = a + bX, Where, X = explanatory variable
- Y = dependent variable
- b = slope of the line
- a = intercept (the value of y when x = 0)



RANSAC Linear Regression

- RANSAC is an acronym for Random Sample Consensus. What this algorithm does is fit a regression model on a subset of data that the algorithm judges as inliers while removing outliers. This naturally improves the fit of the model due to the removal of some data points.
- An advantage of RANSAC is its ability to do robust estimation of the model parameters, i.e., it can estimate the parameters with a high degree of accuracy, even when a significant number of outliers is present in the data set.

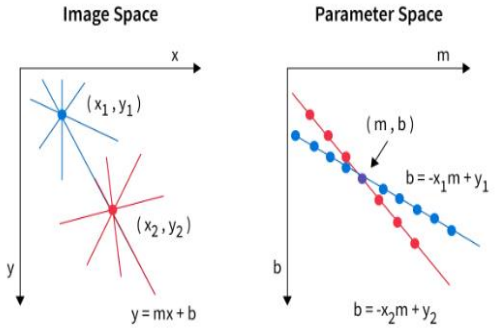
Hough Transform

- The Hough transform is a mathematical technique used in computer vision and image analysis to detect simple geometric shapes like lines, circles, and ellipses.

- The basic idea behind the Hough transform is to represent lines or curves in an image as points in a parameter space. The Hough transform is a powerful tool for line and curve detection in images.
- Its ability to handle broken and incomplete lines makes it a valuable addition to any image processing toolkit.

HISTORY

- The Hough transform was first proposed by Paul Hough in 1962 as a method for detecting lines in images. It was later extended to detect other shapes like circles and ellipses.

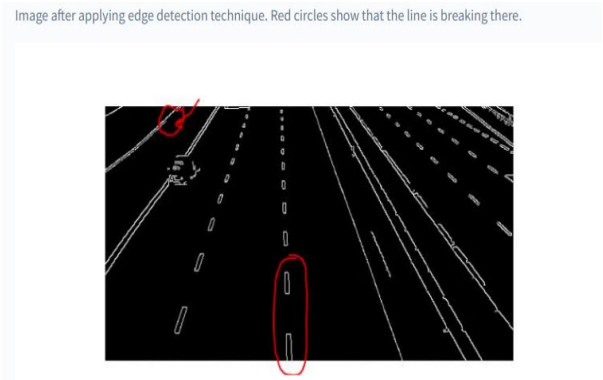


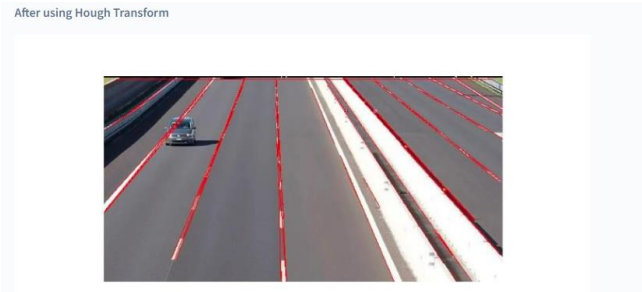
Why is Hough Transform Needed?

- The Hough transform can detect these shapes by transforming the image space into a parameter space where the shapes can be more easily identified.

How does it Work?

- The Hough transform in image processing works by transforming the image space into a parameter space. For example, in the case of detecting lines in images,
- The image space is transformed into a parameter space consisting of two parameters: the slope and the y-intercept of the line.
- Each pixel in the image space is then mapped to a curve in the parameter space that represents all the possible lines that could pass through that pixel. The curves in the parameter space are then analyzed to detect the presence of lines in the image.





- **Applications**

The following are the applications of hough transform

- Biometric and Man-machine Interaction
- Medical Application
- Object Recognition

LOGISTIC REGRESSION

Logistic regression is a data analysis technique that uses mathematics to find the relationships between two data factors. It then uses this relationship to predict the value of one of those factors based on the other. The prediction usually has a finite number of outcomes, like yes or no.

For example, let's say you want to guess if your website visitor will click the checkout button in their shopping cart or not. Logistic regression analysis looks at past visitor behavior, such as time spent on the website and the number of items in the cart. It determines that, in the past, if

visitors spent more than five minutes on the site and added more than three items to the cart, they clicked the checkout button. Using this information, the logistic regression function can then predict the behavior of a new website visitor.

Why is logistic regression important?

Logistic regression is an important technique in the field of artificial intelligence and machine learning (AI/ML). ML models are software programs that you can train to perform complex data processing tasks without human intervention.

For example, businesses can uncover patterns that improve employee retention or lead to more profitable product design.

Below, we list some benefits of using logistic regression over other ML techniques.

Simplicity

Logistic regression models are mathematically less complex than other ML methods. Therefore, you can implement them even if no one on your team has in-depth ML expertise.

Speed

Logistic regression models can process large volumes of data at high speed because they require less computational capacity, such as memory and processing power. This makes them ideal for organizations that are starting with ML projects to gain some quick wins.

Flexibility

You can use logistic regression to find answers to questions that have two or more finite outcomes. You can also use it to preprocess data. For example, you can sort data with a large range of values, such as bank transactions, into a smaller, finite range of values by using logistic regression. You can then process this smaller data set by using other ML techniques for more accurate analysis.

Visibility

Logistic regression analysis gives developers greater visibility into internal software processes than do other data analysis techniques. Troubleshooting and error correction are also easier because the calculations are less complex.

What are the applications of logistic regression?

Logistic regression has several real-world applications in many different industries.

Manufacturing

Healthcare

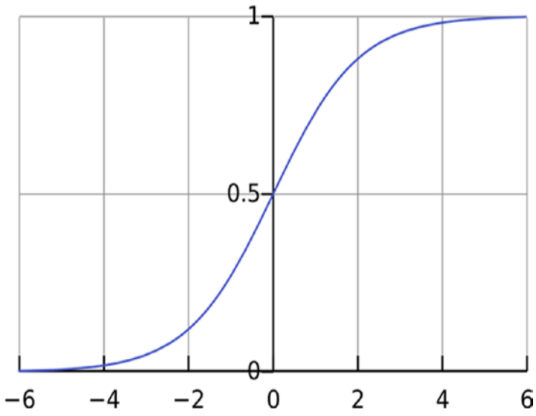
Finance

Marketing

Logistic regression function

Logistic regression is a statistical model that uses the logistic function, or logit function, in mathematics as the equation between x and y. The logit function maps y as a sigmoid function of x.

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



As you can see, the logistic function returns only values between 0 and 1 for the dependent variable, irrespective of the values of the independent variable. This is how logistic regression estimates the value of the dependent variable. Logistic regression methods also model equations between multiple independent variables and one dependent variable.

What are the types of logistic regression analysis?

SIMPLE LOGISTIC REGRESSION

- Simple logistic regression is analogous to linear regression, except that the dependent variable is nominal, not a measurement.
- When the outcome variable is categorical in nature, logistic regression can be used to predict the likelihood of an outcome based on the input variables.
-

EXAMPLE OF SIMPLE LOGISTIC REGRESSION

- Using a binary variable The output is a categorical: yes or no. Hence, is there a traffic jam? Yes or no?

- **The probability of occurrence of traffic jams can be dependent on attributes such as weather condition, day of the week and month, time of day, number of vehicles, etc.**
-

MULTINOMIAL LOGISTIC REGRESSION

- Multinomial logistic regression (often just called 'multinomial regression') is used to predict a nominal dependent variable given one or more independent variables.
- It is sometimes considered an extension of binomial logistic regression to allow for a dependent variable with more than two categories.

EXAMPLE OF MULTINOMIAL LOGISTIC REGRESSION

Which type of drink consumers prefer based on location in the UK and age (i.e., the dependent variable would be "type of drink", with four categories – Coffee, Soft Drink, Tea and Water – and your independent variables would be the nominal variable, "location in UK", assessed using three categories – London, South UK and North UK – and the continuous variable, "age", measured in years).

ORDINAL LOGISTIC REGRESSION

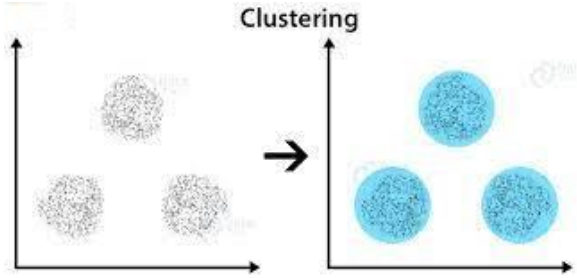
- Ordinal logistic regression (often just called 'ordinal regression') is used to predict an ordinal dependent variable given one or more independent variables.
- As with other types of regression, ordinal regression can also use interactions between independent variables to predict the dependent variable.

EXAMPLE OF ORDINAL LOGISTIC REGRESSION

- For example, you could use ordinal regression to predict the belief that "tax is too high" (your ordinal dependent variable, measured on a 4-point Likert item from "Strongly Disagree" to "Strongly Agree"), based on two independent variables: "age" and "income".

CLUSTERING TECHNIQUES

- Clustering is the use of unsupervised techniques for grouping similar objects. In machine learning, unsupervised refers to the problem of finding hidden structure within unlabelled data.
- In clustering, there are no predictions made. Rather, clustering methods find the similarities between objects according to the object attributes and group the similar objects into clusters. Clustering techniques are utilized in marketing, economics, and various branches of science.



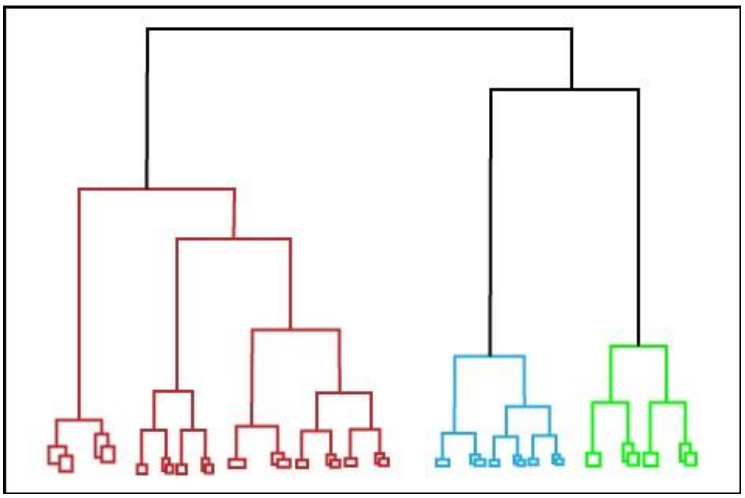
Why Clustering?

Clustering is very much important as it determines the intrinsic grouping among the unlabelled data present. There are no criteria for good clustering. It depends on the user, and what criteria they may use which satisfy their need. For instance, we could be interested in finding representatives for homogeneous groups (data reduction), finding "natural clusters" and describing their unknown properties ("natural" data types), in finding useful and suitable groupings ("useful" data classes) or in finding unusual data objects (outlier detection). This algorithm must make some assumptions that constitute the similarity of points and each assumption make different and equally valid clusters.

Clustering Methods:

HIERARCHICAL CLUSTERING

- Hierarchical clustering is a method of cluster analysis whereby you build a hierarchy of clusters. This works well for data sets that are complex .
- Also called Hierarchical cluster analysis or HCA is an unsupervised clustering algorithm which involves creating clusters that have predominant ordering from top to bottom.



The clusters formed in this method form a tree-type structure based on the hierarchy. New clusters are formed using the previously formed one. It is divided into two category

- **Agglomerative** (bottom-up *approach*)
- **Divisive** (top-down *approach*)

AGGLOMERATIVE CLUSTERING

- It's also known as AGNES (Agglomerative Nesting). It's a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

How does it work?

1. Make each data point a single-point cluster → forms N clusters

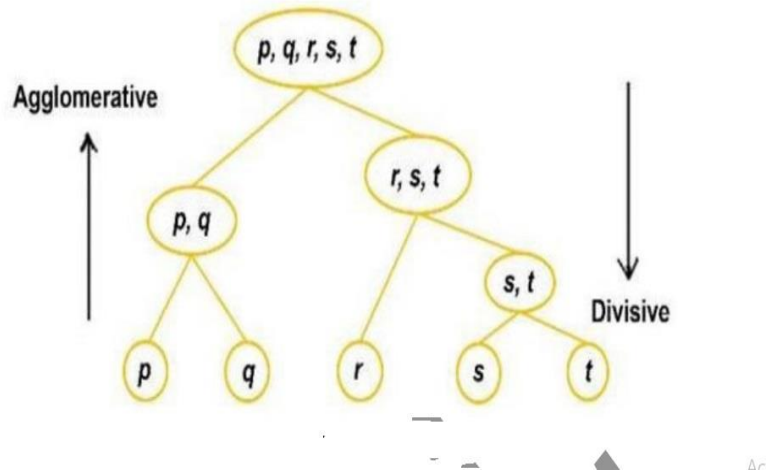
2. Take the two closest data points and make them one cluster → forms N-1 clusters

3. Take the two closest clusters and make them one cluster → Forms N-2 clusters.

4. Repeat step-3 until you are left with only one cluster.

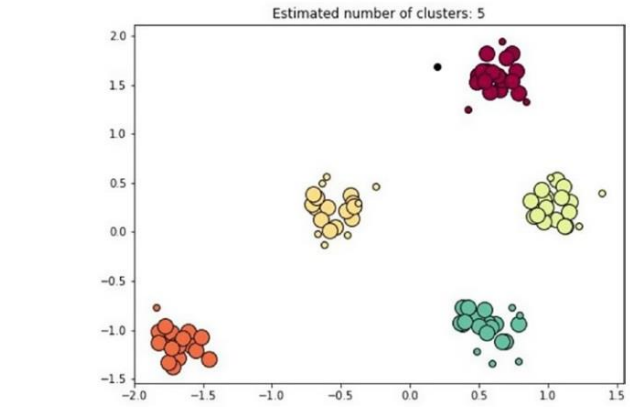
DIVISIVE CLUSTERING

- In Divisive or DIANA (DIvisive ANAlysis Clustering) is a top-down clustering method where we assign all of the observations to a single cluster and then partition the cluster to two least similar clusters.
- Finally, we proceed recursively on each cluster until there is one cluster for each observation.



PARTITIONAL CLUSTERING

- . Partitional clustering decomposes a data set into a set of disjoint clusters.
- it classifies the data into K groups by satisfying the following requirements:
- (1) each group contains at least one point,
- (2) each point belongs to exactly one group



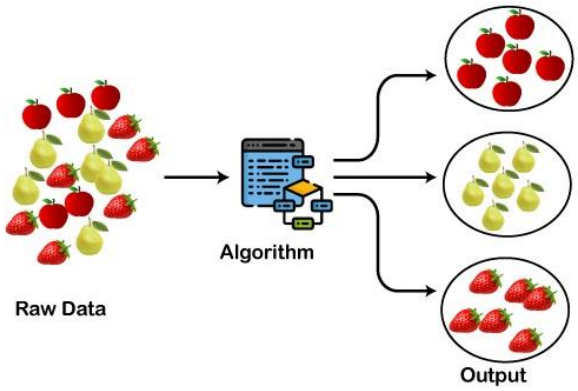
Partitional clustering

Density-Based Clustering

- In density-based clustering, an area of higher density is separated from the remainder of the data set. Data entries in sparse areas are placed in separate clusters. These clusters are considered to be noise, outliers, and border data entries

Applications of Clustering in different fields:

1. **Marketing:** It can be used to characterize & discover customer segments for marketing purposes.
2. **Biology:** It can be used for classification among different species of plants and animals.
3. **Libraries:** It is used in clustering different books on the basis of topics and information.
4. **Insurance:** It is used to acknowledge the customers, their policies and identifying the frauds.
5. **City Planning:** It is used to make groups of houses and to study their values based on their geographical locations and other factors present.
6. **Earthquake studies:** By learning the earthquake-affected areas we can determine the dangerous zones.
7. **Image Processing**: Clustering can be used to group similar images together, classify images based on content, and identify patterns in image data.
8. **Genetics:** Clustering is used to group genes that have similar expression patterns and identify gene networks that work together in biological processes.



ANOVA

- The one-way analysis of variance (ANOVA) test is used to determine whether the mean of more than two groups of data sets is significantly different from each data set.
- The core of this technique lies in assessing whether all the groups are in fact part of one larger population or a completely different population with different characteristics.

EXAMPLE OF ANOVA

A BOGOF (buy-one-get-one-free) campaign is executed on 5 groups of 100 customers each. Each group is different in terms of its demographic attributes. We would like to determine whether these five respond differently to the campaign. This would help us optimize the right campaign for the right demographic group, increase the response rate, and reduce the cost of the campaign

TYPES OF ANOVA

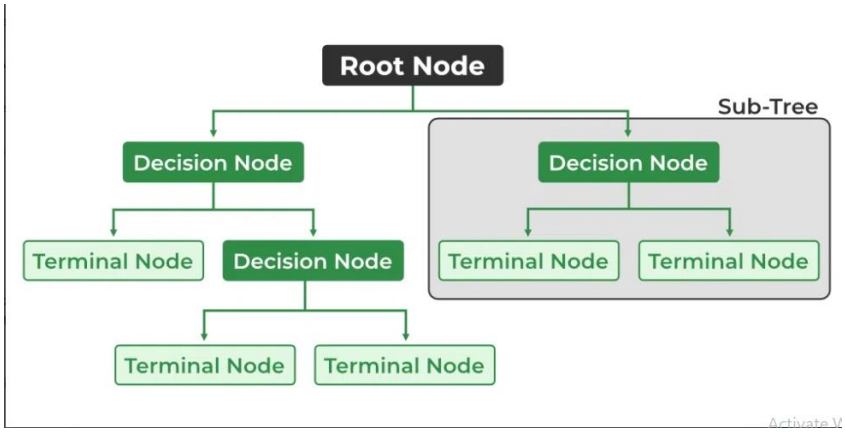
- ONE WAY ANOVA
- With a one-way, you have one independent variable affecting a dependent variable.
- TWO WAY ANOVA
- With a two-way ANOVA, there are two independents. For example, a two-way ANOVA allows a company to compare worker productivity based on two independent variables, such as salary and skill set.

PRINCIPAL COMPONENT ANALYSIS

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5.

DECISION TREES

- A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks.
- It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

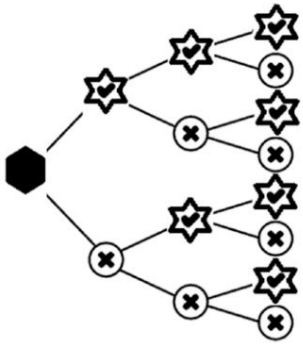


VARIANCE

- Variance: Variance measures how much the predicted and the target variables vary in different samples of a dataset. It is used for regression problems in decision trees. Mean squared error, Mean Absolute Error are used to measure the variance for the regression tasks in the decision tree.

Pruning

The process of removing branches from the tree that do not provide any additional information or lead to overfitting.



Simple decision tree

Entropy AND Gini Impurity or index:

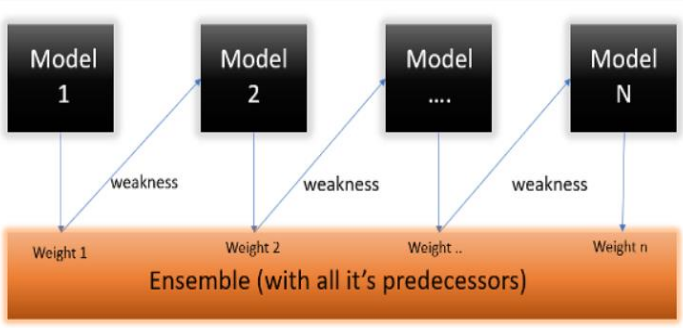
- Entropy is the measure of the degree of randomness or uncertainty in the dataset.
- Gini Impurity is a score that evaluates how accurate a split is among the classified groups. The Gini Impurity evaluates a score in the range between 0 and 1, where 0 is when all observations belong to one class, and 1 is a random distribution of the elements within classes.

How does the Decision Tree algorithm Work?

- The decision tree operates by analyzing the data set to predict its classification.
- Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
- Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
- Step-3: Divide the S into subsets that contains possible values for the best attributes.
- Step-4: Generate the decision tree node, which contains the best attribute.
- Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node Classification and Regression Tree algorithm.

ADABOOST ALGORITHM FOR DECISION TREES

- The principle behind boosting algorithms is that we first build a model on the training dataset and then build a second model to rectify the errors present in the first model.
- This procedure is continued until and unless the errors are minimized and the dataset is predicted correctly.
- What this algorithm does is that it builds a model and gives equal weights to all the data points. It then assigns higher weights to points that are wrongly classified. Now all the points with higher weights are given more importance in the next model. It will keep training models until and unless a lower error is received.



Understanding the Working of the AdaBoost Algorithm

- **Step 1: Assigning Weights**

The formula to calculate the sample weights is:

$$w(x_i, y_i) = \frac{1}{N}, \; i = 1, 2, \ldots . n$$

Where N is the total number of data points

Here since we have 5 data points, the sample weights assigned will be 1/5.

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

Step 2: Classify the Samples

- We start by seeing how well "*Gender*" classifies the samples and will see how the variables (Age, Income) classify the samples.

Step 3: Calculate the Influence

- We'll now calculate the "Amount of Say" or "Importance" or "Influence" for this classifier in classifying the data points using this formula:

$$\frac{1}{2}\log\frac{1 - Total\ Error}{Total\ Error}$$

$$\alpha = \frac{1}{2}\log_e\left(\frac{1-\frac{1}{5}}{\frac{1}{5}}\right)$$

$$\alpha = \frac{1}{2}\log_e\left(\frac{0.8}{0.2}\right)$$

$$\alpha = \frac{1}{2}\log_e(4) = \frac{1}{2} * (1.38)$$

$$\alpha = 0.69$$

Note: Total error will always be between 0 and 1.

Step 4: Calculate TE and Performance

- The wrong predictions will be given more weight, whereas the correct predictions weights will be decreased. Now when we build our next model after updating the weights, more preference will be given to the points with higher weights.

$$New\ sample\ weight = old\ weight * e^{\pm Amount\ of\ say\ (\alpha)}$$

The amount of, say (alpha) will be ***negative*** when the sample is **correctly classified**.

The amount of, say (alpha) will be ***positive*** when the sample is **miss-classified.**

New weights for *correctly classified* samples are:

$$New\ sample\ weight = \frac{1}{5} * \exp(-0.69)$$

$$New\ sample\ weight = 0.2 * 0.502 = 0.1004$$

For *wrongly classified* samples, the updated weights will be:

$$New\ sample\ weight = \frac{1}{5} * \exp(0.69)$$

$$New\ sample\ weight = 0.2 * 1.994 = 0.3988$$

Row No.	Gender	Age	Income	Illness	Sample Weights	New Sample Weights
1	Male	41	40000	Yes	1/5	0.1004
2	Male	54	30000	No	1/5	0.1004
3	Female	42	25000	No	1/5	0.1004
4	Female	40	60000	Yes	1/5	0.3988
5	Male	46	50000	Yes	1/5	0.1004

Row No.	Gender	Age	Income	Illness	Sample Weights	New Sample Weights
1	Male	41	40000	Yes	1/5	0.1004/0.8004 =0.1254
2	Male	54	30000	No	1/5	0.1004/0.8004 =0.1254
3	Female	42	25000	No	1/5	0.1004/0.8004 =0.1254
4	Female	40	60000	Yes	1/5	0.3988/0.8004 =0.4982
5	Male	46	50000	Yes	1/5	0.1004/0.8004 =0.1254

Step 5: Decrease Errors

- Now, we need to make a new dataset to see if the errors decreased or not. For this, we will remove the "sample weights" and "new sample weights" columns and then, based on the "new sample weights," divide our data points into buckets.

Row No.	Gender	Age	Income	Illness	New Sample Weights	Buckets
1	Male	41	40000	Yes	0.1004/0.8004= 0.1254	0 to 0.1254
2	Male	54	30000	No	0.1004/0.8004= 0.1254	0.1254 to 0.2508
3	Female	42	25000	No	0.1004/0.8004= 0.1254	0.2508 to 0.3762
4	Female	40	60000	Yes	0.3988/0.8004= 0.4982	0.3762 to 0.8744
5	Male	46	50000	Yes	0.1004/0.8004= 0.1254	0.8744 to 0.9998

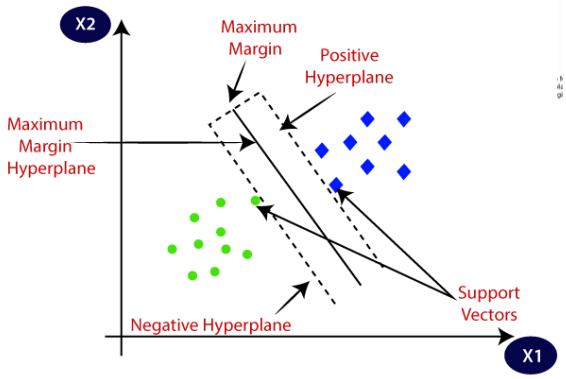
STEP-6:CREATE THE NEW DATASET AND REPEAT THE PREVIOUS STEPS

SUPPORT VECTOR MACHINES

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.

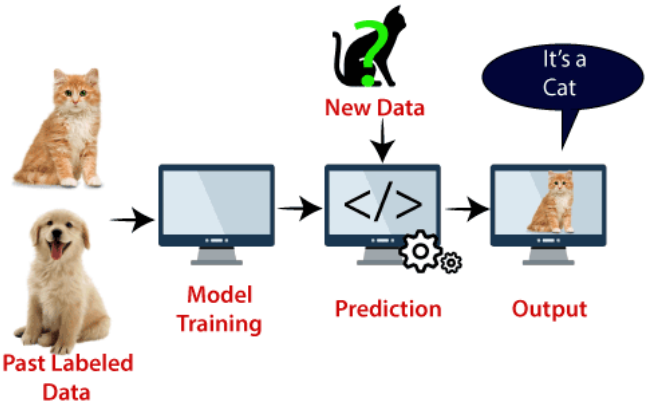
GOAL OF SVM

- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.
- This best decision boundary is called a hyperplane.
- SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



Example: SVM can be understood with the example that we have used in the KNN classifier. Suppose we see a strange cat that also has some features of dogs, so if we want a model that

can accurately identify whether it is a cat or dog, so such a model can be created by using the SVM algorithm. We will first train our model with lots of images of cats and dogs so that it can learn about different features of cats and dogs, and then we test it with this strange creature. So as support vector creates a decision boundary between these two data (cat and dog) and choose extreme cases (support vectors), it will see the extreme case of cat and dog. On the basis of the support vectors, it will classify it as a cat. Consider the below diagram:



SVM algorithm can be used for **Face detection, image classification, text categorization,** etc.

Types of SVM

SVM can be of two types:

- **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

Hyperplane and Support Vectors in the SVM algorithm:

Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

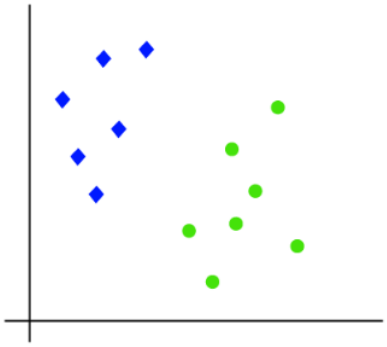
Support Vectors:

The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

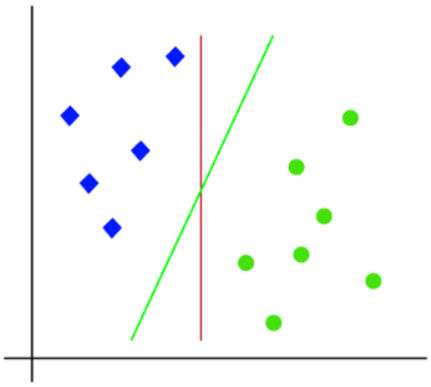
How does SVM works?

Linear SVM:

The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider the below image:



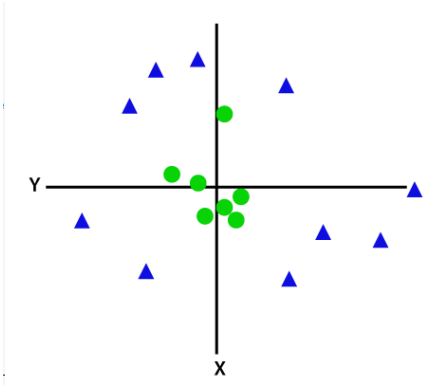
So as it is 2-d space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the below image:



Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called as a **hyperplane**. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane is called as **margin**. And the goal of SVM is to maximize this margin. The **hyperplane** with maximum margin is called the **optimal hyperplane**.

Non-Linear SVM:

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image:



NETWORKS,CLUSTERS AND GRIDS

- The support vector network is the ensemble of a network of support vector machines that together classify the same data set, by using different parameters .
- Support vector clustering is used were the data points are classified into clusters, with support vector machines performing the classification at the cluster level.
- The support vector grid (SVG) is an SVC of an SVN or an SVN of an SVC.
- It uses SVMs to handle smaller clusters of the data, to apply specific transform steps.

Linear SVM	Non-Linear SVM
It can be easily separated with a linear line.	It cannot be easily separated with a linear line.
Data is classified with the help of hyperplane.	We use Kernels to make non-separable data into separable data.
Data can be easily classified by drawing a straight line.	We map data into high dimensional space to classify.

DATA MINING

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5.

PATTERN RECOGNITION

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5.

MACHINE LEARNING

- Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.
- Machine learning contains a set of algorithms that work on a huge amount of data. Data is fed to these algorithms to train them, and on the basis of training, they build the model & perform a specific task.

These ML algorithms help to solve different business problems like Regression, Classification, Forecasting, Clustering, and Associations, etc.

Based on the methods and way of learning, machine learning is divided into mainly four types, which are:

- Supervised Machine Learning
- Unsupervised Machine Learning
- Reinforcement Learning

Supervised Machine Learning

As its name suggests, Supervised machine learning is based on supervision. It means in the supervised learning technique, we train the machines using the "labelled" dataset, and based on the training, the machine predicts the output. Here, the labelled data specifies that some of the inputs are already mapped to the output. More preciously, we can say; first, we train the machine with the input and corresponding output, and then we ask the machine to predict the output using the test dataset.

Let's understand supervised learning with an example. Suppose we have an input dataset of cats and dog images. So, first, we will provide the training to the machine to understand the images, such as the **shape & size of the tail of cat and dog, Shape of eyes, colour, height (dogs are taller, cats are smaller), etc.** After completion of training, we input the picture of a cat and ask the machine to identify the object and predict the output. Now, the machine is well trained, so it will check all the features of the object, such as height, shape, colour, eyes, ears, tail, etc., and find that it's a cat. So, it will put it in the Cat category. This is the process of how the machine identifies the objects in Supervised Learning.

The main goal of the supervised learning technique is to map the input variable(x) with the output variable(y). Some real-world applications of supervised learning are **Risk Assessment, Fraud Detection, Spam filtering,** etc.

Categories of Supervised Machine Learning

Supervised machine learning can be classified into two types of problems, which are given below:

- **Classification**
- **Regression**

a) Classification

Classification algorithms are used to solve the classification problems in which the output variable is categorical, such as "**Yes" or No, Male or Female, Red or Blue, etc**. The classification algorithms predict the categories present in the dataset. Some real-world examples of classification algorithms are **Spam Detection, Email filtering, etc.**

Some popular classification algorithms are given below:

- **Random Forest Algorithm**
- **Decision Tree Algorithm**
- **Logistic Regression Algorithm**
- **Support Vector Machine Algorithm**

b) Regression

Regression algorithms are used to solve regression problems in which there is a linear relationship between input and output variables. These are used to predict continuous output variables, such as market trends, weather prediction, etc.

Some popular Regression algorithms are given below:

- **Simple Linear Regression Algorithm**
- **Multivariate Regression Algorithm**
- **Decision Tree Algorithm**
- **Lasso Regression**

Advantages and Disadvantages of Supervised Learning

Advantages:

- Since supervised learning work with the labelled dataset so we can have an exact idea about the classes of objects.
- These algorithms are helpful in predicting the output on the basis of prior experience.

Disadvantages:

- These algorithms are not able to solve complex tasks.
- It may predict the wrong output if the test data is different from the training data.
- It requires lots of computational time to train the algorithm.

Applications of Supervised Learning

Some common applications of Supervised Learning are given below:

- **Image Segmentation:**
  Supervised Learning algorithms are used in image segmentation. In this process, image classification is performed on different image data with pre-defined labels.
- **Medical Diagnosis:**
  Supervised algorithms are also used in the medical field for diagnosis purposes. It is done by using medical images and past labelled data with labels for disease conditions. With such a process, the machine can identify a disease for the new patients.

- **Fraud Detection -** Supervised Learning classification algorithms are used for identifying fraud transactions, fraud customers, etc. It is done by using historic data to identify the patterns that can lead to possible fraud.
- **Spam detection -** In spam detection & filtering, classification algorithms are used. These algorithms classify an email as spam or not spam. The spam emails are sent to the spam folder.
- **Speech Recognition -** Supervised learning algorithms are also used in speech recognition. The algorithm is trained wit

2. Unsupervised Machine Learning

Unsupervised learning is different from the Supervised learning technique; as its name suggests, there is no need for supervision. It means, in unsupervised machine learning, the machine is trained using the unlabeled dataset, and the machine predicts the output without any supervision.

In unsupervised learning, the models are trained with the data that is neither classified nor labelled, and the model acts on that data without any supervision.

The main aim of the unsupervised learning algorithm is to group or categories the unsorted dataset according to the similarities, patterns, and differences. Machines are instructed to find the hidden patterns from the input dataset.

Let's take an example to understand it more preciously; suppose there is a basket of fruit images, and we input it into the machine learning model. The images are totally unknown to the model, and the task of the machine is to find the patterns and categories of the objects.
So, now the machine will discover its patterns and differences, such as colour difference, shape difference, and predict the output when it is tested with the test dataset.

Categories of Unsupervised Machine Learning

Unsupervised Learning can be further classified into two types, which are given below:

- **Clustering**
- **Association**

1) Clustering

The clustering technique is used when we want to find the inherent groups from the data. It is a way to group the objects into a cluster such that the objects with the most similarities remain in one group and have fewer or no similarities with the objects of other groups. An example of the clustering algorithm is grouping the customers by their purchasing behaviour.

Some of the popular clustering algorithms are given below:

- **K-Means Clustering algorithm**
- **Mean-shift algorithm**

- **DBSCAN Algorithm**
- **Principal Component Analysis**
- **Independent Component Analysis**

2) Association

Association rule learning is an unsupervised learning technique, which finds interesting relations among variables within a large dataset. The main aim of this learning algorithm is to find the dependency of one data item on another data item and map those variables accordingly so that it can generate maximum profit. This algorithm is mainly applied in **Market Basket analysis, Web usage mining, continuous production**, etc.

Some popular algorithms of Association rule learning are **Apriori Algorithm, Eclat, FP-growth algorithm.**

Advantages and Disadvantages of Unsupervised Learning Algorithm

Advantages:

- These algorithms can be used for complicated tasks compared to the supervised ones because these algorithms work on the unlabeled dataset.
- Unsupervised algorithms are preferable for various tasks as getting the unlabeled dataset is easier as compared to the labelled dataset.

Disadvantages:

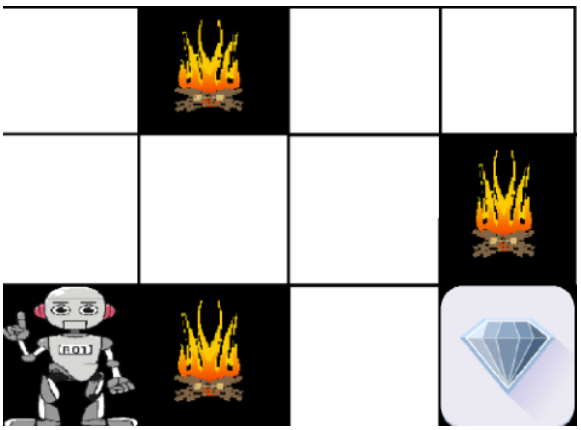
- The output of an unsupervised algorithm can be less accurate as the dataset is not labelled, and algorithms are not trained with the exact output in prior.
- Working with Unsupervised learning is more difficult as it works with the unlabelled dataset that does not map with the output.

Applications of Unsupervised Learning

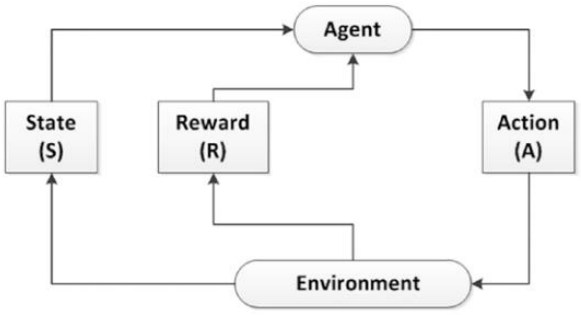
- **Network Analysis:** Unsupervised learning is used for identifying plagiarism and copyright in document network analysis of text data for scholarly articles.
- **Recommendation Systems:** Recommendation systems widely use unsupervised learning techniques for building recommendation applications for different web applications and e-commerce websites.
- **Anomaly Detection:** Anomaly detection is a popular application of unsupervised learning, which can identify unusual data points within the dataset. It is used to discover fraudulent transactions.
- **Singular Value Decomposition:** Singular Value Decomposition or SVD is used to extract particular information from the database. For example, extracting information of each user located at a particular location.

Reinforcement Learning

- Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation.
- This is used in several different areas, such as game theory, operations research, simulation-based optimization, multi-agent systems, statistics, and genetic algorithms.
- Reinforcement learning is an autonomous, self-teaching system that essentially learns by trial and error.



The robot learns by trying all the possible paths and then choosing the path which gives him the reward with the least hurdles. Each right step will give the robot a reward and each wrong step will subtract the reward of the robot.



. *Reinforced learning diagram*

Categories of Reinforcement Learning

Reinforcement learning is categorized mainly into two types of methods/algorithms:

- **Positive Reinforcement Learning:** Positive reinforcement learning specifies increasing the tendency that the required behaviour would occur again by adding something. It enhances the strength of the behaviour of the agent and positively impacts it.
- **Negative Reinforcement Learning:** Negative reinforcement learning works exactly opposite to the positive RL. It increases the tendency that the specific behaviour would occur again by avoiding the negative condition.

Real-world Use cases of Reinforcement Learning

- **Video Games:**
  RL algorithms are much popular in gaming applications. It is used to gain super-human

performance. Some popular games that use RL algorithms are **AlphaGO** and **AlphaGO Zero**.

- **Resource Management:**
  The "Resource Management with Deep Reinforcement Learning" paper showed that how to use RL in computer to automatically learn and schedule resources to wait for different jobs in order to minimize average job slowdown.
- **Robotics:**
  RL is widely being used in Robotics applications. Robots are used in the industrial and manufacturing area, and these robots are made more powerful with reinforcement learning. There are different industries that have their vision of building intelligent robots using AI and Machine learning technology.
- **Text Mining**
  Text-mining, one of the great applications of NLP, is now being implemented with the help of Reinforcement Learning by Salesforce company.

BAGGING DATA

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5.

RANDOM FORESTS

- It is a supervised learning technique that constructs an ensemble of decision-tree classifiers and uses random selection to create multiple forests from which the final prediction is made.
- The random forest has been utilised in various applications, ranging from healthcare to finance.

Why is random forests algorithm so called?

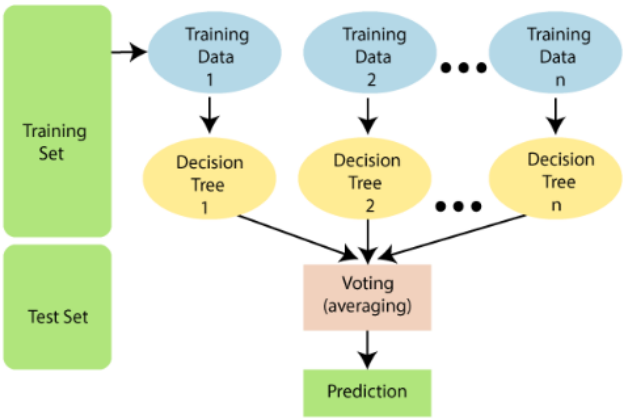
As the name suggests, ***"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."*** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Assumptions for Random Forest

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier:

- There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.
- The predictions from each tree must have very low correlations.



Why use Random Forest?

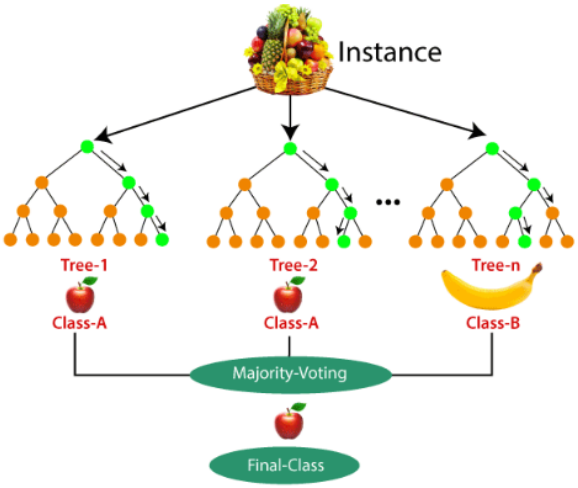
- It takes less training time as compared to other algorithms.
- It predicts output with high accuracy, even for the large dataset it runs efficiently.
- It can also maintain accuracy when a large proportion of data is missing.

STEPS OF RANDOM FOREST ALGORITHM

- Step-1: Select random K data points from the training set.
- Step-2: Build the decision trees associated with the selected data points (Subsets).
- Step-3: Choose the number N for decision trees that you want to build.
- Step-4: Repeat Step 1 & 2.
- Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

EXAMPLE OF RANDOM FOREST

Suppose there is a dataset that contains multiple fruit images. So, this dataset is given to the Random forest classifier. The dataset is divided into subsets and given to each decision tree. During the training phase, each decision tree produces a prediction result, and when a new data point occurs, then based on the majority of results, the Random Forest classifier predicts the final decision. Consider the below image:



Applications of Random Forest

There are mainly four sectors where Random forest mostly used:

1. **Banking:** Banking sector mostly uses this algorithm for the identification of loan risk.
2. **Medicine:** With the help of this algorithm, disease trends and risks of the disease can be identified.
3. **Land Use:** We can identify the areas of similar land use by this algorithm.
4. **Marketing:** Marketing trends can be identified using this algorithm.

LIMITATIONS OF RANDOM FOREST

- **It may not work well on too small datasets.**
- **Random forests can also be slow to train and predict as the number of trees in the forest increases.**

COMPUTER VISION

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5

NATURAL LANGUAGE PROCESSING

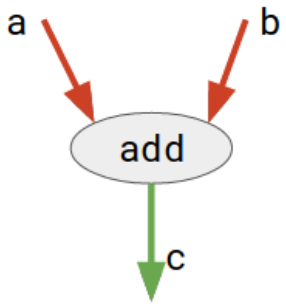
STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5

NEURAL NETWORKS

STUDENTS CAN STUDY THIS TOPIC FROM THE POWERPOINT PRESENTATION OF THE UNIT-5

TENSOR FLOW

- TensorFlow is basically a software library for numerical computation using data flow graphs where:
- nodes in the graph represent mathematical operations.
- edges in the graph represent the multidimensional data arrays (called tensors) communicated between them.



Here, add is a node which represents addition operation. a and b are input tensors and c is the resultant tensor.

INSTALLING TENSORFLOW

import tensorflow as tf

The Computational Graph

- A computational graph is nothing but a series of TensorFlow operations arranged into a graph of nodes.

```
# importing tensorflow
import tensorflow as tf

# creating nodes in computation graph
node1 = tf.constant(3, dtype=tf.int32)
node2 = tf.constant(5, dtype=tf.int32)
node3 = tf.add(node1, node2)

# create tensorflow session object
sess = tf.compat.v1.Session()

# evaluating node3 and printing the result
print("sum of node1 and node2 is :",sess.run(node3))
# closing the session
sess.close()
```

In order to run the computational graph, we need to create a session.

VARIABLES IN TENSORFLOW

- TensorFlow has Variable nodes too which can hold variable data. They are mainly used to hold and update parameters of a training model.

```
b = tf.Variable(2.5, name='b')
c = tf.Variable(10.0, name='c')
```

PLACEHOLDERS IN TENSORFLOW

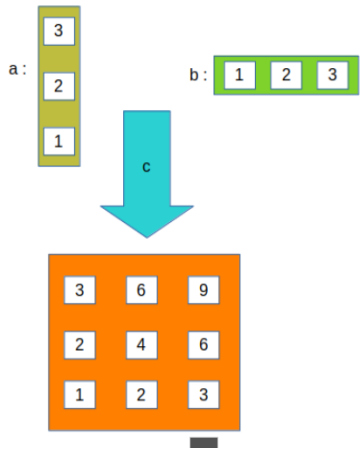
- A graph can be parameterized to accept external inputs, known as placeholders.

```
# importing tensorflow
import tensorflow as tf

# creating nodes in computation graph
a = tf.placeholder(tf.int32, shape=(3,1))
b = tf.placeholder(tf.int32, shape=(1,3))
c = tf.matmul(a,b)

# running computation graph
with tf.Session() as sess:
    print(sess.run(c, feed_dict={a:[[3],[2],[1]], b:[[1,2,3]]}))
```

OUTPUT

[[3 6 9]
[2 4 6]
[1 2 3]]

- After sess.run:



Real-World Uses of TensorFlow

- Basket analysis: What do people buy? What do they buy together?
- Forex trading: Providing recommendations on when to purchase forex for company requirements.
- Commodities trading: Buying and selling futures.

UNIT-5
PART-2
ORGANIZE AND REPORT SUPERSTEPS

ORGANIZE SUPERSTEP

The Organize superstep takes the complete data warehouse you built at the end of the Transform superstep and subsections it into business-specific data marts. A data mart is the access layer of the data warehouse environment built to expose data to the users. The data mart is a subset of the data warehouse and is generally oriented to a specific business group.

DATA MART VS. DATA WAREHOUSE VS DATA LAKE

- A data warehouse is a system that aggregates data from multiple sources into a single, central, consistent data store to support data mining, artificial intelligence (AI), and machine learning.
- A data mart (as noted above) is a focused version of a data warehouse that contains a smaller subset of data important to and needed by a single team or a select group of users within an organization.
- A data lake, too, is a repository for data. A data lake provides massive storage of unstructured or raw data fed via multiple sources, but the information has not yet been processed or prepared for analysis.

DIFFERENT STYLES OF DATASLICING

Horizontal Style

- **Horizontal style slicing selects the subset of rows from the population while preserving the columns.**

```
# -*- coding: utf-8 -*-
#####################################################################
import sys
import os
import pandas as pd
import sqlite3 as sq
#####################################################################

   If sys.platform == 'linux' or sys.platform == ' darwin':

    Base=os.path.expanduser('~') + '/VKHCG'
else:
    Base='C:/VKHCG'
print('#################################')
print('Working Base :',Base, ' using ', sys.platform)
print('#################################')
```

```
Company='01-Vermeulen'
################################################################
sDataWarehouseDir=Base + '/99-DW'
if not os.path.exists(sDataWarehouseDir):
    os.makedirs(sDataWarehouseDir)
```

-
```
################################################################
sDatabaseName=sDataWarehouseDir + '/datawarehouse.db'
conn1 = sq.connect(sDatabaseName)
################################################################
sDatabaseName=sDataWarehouseDir + '/datamart.db'
conn2 = sq.connect(sDatabaseName)
################################################################
```

Load the complete BMI data set from the data warehouse.

- **The next query loads all the data into memory, and that means you will have the complete data set ready in memory**

```
print('#################')
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT * FROM [Dim-BMI];"
PersonFrame0=pd.read_sql_query(sSQL, conn1)
```

-
- **LOADING THE HORIZONTAL DATA SLICE FOR BMI**

Load the horizontal data slice for body mass index (BMI) from the d

```
print('################')
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT PersonID,\
      Height,\
      Weight,\
      bmi,\
      Indicator\
```

```
  FROM [Dim-BMI]\
  WHERE \
  Height > 1.5 \
  and Indicator = 1\
  ORDER BY  \
       Height,\
       Weight;"
PersonFrame1=pd.read_sql_query(sSQL, conn1)
#####################################################################
DimPerson=PersonFrame1
DimPersonIndex=DimPerson.set_index(['PersonID'],inplace=False)
#####################################################################
```

- **Store the horizontal data slice for BMI into the data warehouse.**

```
sTable = 'Dim-BMI'
print('\n#################################')
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n#################################')
DimPersonIndex.to_sql(sTable, conn2, if_exists="replace")
################################################################
print('################')
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT * FROM [Dim-BMI];"
PersonFrame2=pd.read_sql_query(sSQL, conn2)
```

You can show your results by printing the following code. You can see the improvement you achieved.

```
print('Full Data Set (Rows):', PersonFrame0.shape[0])
print('Full Data Set (Columns):', PersonFrame0.shape[1])
print('Horizontal Data Set (Rows):', PersonFrame2.shape[0])
print('Horizontal Data Set (Columns):', PersonFrame2.shape[1])
```

-

```
####################################
Full Data Set (Rows): 1080
Full Data Set (Columns): 5
####################################
Horizontal Data Set (Rows): 194
Horizontal Data Set (Columns): 5
####################################
```

-

VERTICAL STYLE

- The vertical-style slicing selects the subset of columns from the population, while preserving the rows.
- The use of vertical-style data slicing is common in systems in which specific data columns may not be shown to everybody, owing to security or privacy regulations

Island Style

- Performing island-style slicing or subsetting of the data warehouse is achieved by applying a combination of horizontal- and vertical-style slicing. This generates a subset of specific rows and specific columns reduced at the same time.

- These types of island slices are typical for snapshotting a reduced data set of data each month. Items may include unpaid accounts, overdue deliveries, and damaged billboards. You do not have to store the complete warehouse. You only want the specific subset of data.

Secure Vault Style

- The secure vault is a version of one of the horizontal, vertical, or island slicing techniques, but the outcome is also attached to the person who performs the query. This is common in multi-security environments, where different users are allowed to see different data sets.

REPORT SUPERSTEP

The Report superstep is the step in the ecosystem that enhances the data science findings with the art of storytelling and data visualization. You can perform the best data science, but if you cannot execute a respectable and trustworthy Report step by turning your data science into actionable business insights, you have achieved no advantage for your business.

Summary of the Results

The most important step in any analysis is the summary of the results. Your data science techniques and algorithms can produce the most methodically, most advanced mathematical or most specific statistical results to the requirements, but if you cannot summarize those into a good story, you have not achieved your requirements.

Understand the Context

What differentiates good data scientists from the best data scientists are not the algorithms or data engineering; it is the ability of the data scientist to apply the context of his findings to the customer

Appropriate Visualization

It is true that a picture tells a thousand words. But in data science, you only want your visualizations to tell one story: the findings of the data science you prepared. It is absolutely necessity to ensure that your audience will get your most important message clearly and without any other meanings

Eliminate Clutter

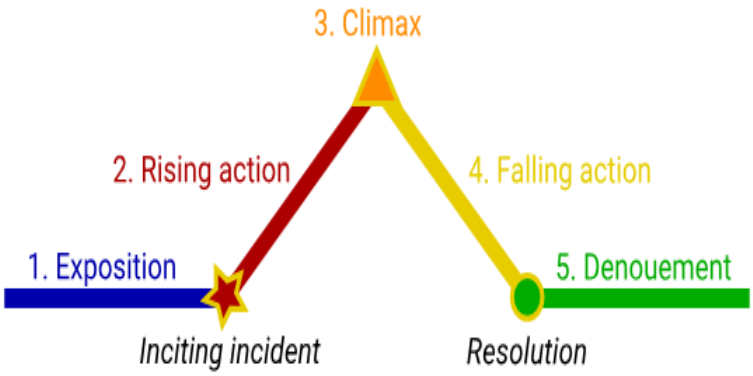
The biggest task of a data scientist is to eliminate clutter in the data sets. There are various algorithms, such as principal component analysis (PCA), multicollinearity using the variance inflation factor to eliminate dimensions and impute or eliminate missing values, decision trees to subdivide, and backward feature elimination, but the biggest contributor to eliminating clutter is good and solid feature engineering

Draw Attention Where You Want It

Your purpose as a data scientist is to deliver insights to your customer, so that they can implement solutions to resolve a problem they may not even know about. You must place the attention on the insight and not the process. However, you must ensure that your process is verified and can support an accredited algorithm or technique.

FREYTAG'S PYRAMID

Under Freytag's pyramid, the plot of a story consists of five parts: exposition, rising action, climax, falling action, and resolution. This is used by writers of books and screenplays as the basic framework of any story. In the same way, you must take your business through the data science process.



- **Freytag used these five parts to analyze the structure:**
- **exposition,**
- **rising action,**
- **climax,**
- **falling action,**
- **resolution**

EXPOSITION AND RISING ACTION

- Exposition is the portion of a story that introduces important background information to the audience. In data science, you tell the background of the investigation you performed.
- Rising action refers to a series of events that build toward the point of greatest interest. In data science, you point out the important findings or results. Keep it simple and to the point.

CLIMAX AND FALLING ACTION

- The climax is the turning point that determines a good or bad outcome for the story's characters. In data science, you show how your solution or findings will change the outcome of the work you performed.
- During the falling action, the conflict between what occurred before and after the climax takes place. In data science, you prove that after your suggestion has been implemented in a pilot, the same techniques can be used to find the issues now proving that the issues can inevitably be resolved

RESOLUTION

- Resolution is the outcome of the story. In data science, you produce the solution and make the improvements permanent.

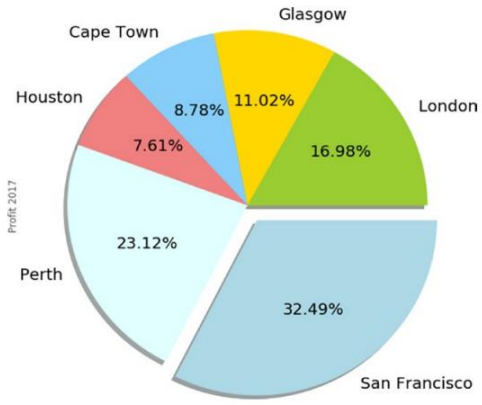
GRAPHICS

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.
In simple words, it is very difficult to gain knowledge from a large amount of data and this is where data visualization comes into the picture. Be it numerical or categorical or mixed type of data, visualization techniques help see the trends, outliers, or any kind of patterns in the data. All this information helps data scientists or anyone in any field of work to make better decisions to achieve their objective.

Plot Options

- **Pie Graph**

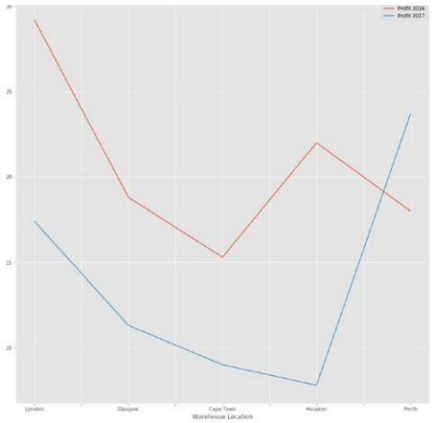
A pie chart (or a circle chart) is a circular statistical graphic which is divided into slices to illustrate numerical proportion. In a pie chart, the arc length of each slice (and consequently its central angle and area), is proportional to the quantity it represents.



Pie graph

- **Line Graph**

Matplotlib is a data visualization library in Python. The pyplot, a sublibrary of matplotlib, is a collection of functions that helps in creating a variety of charts. *Line charts* are used to represent the relation between two data X and Y on a different axis.

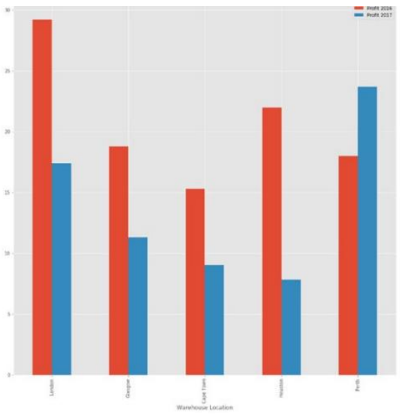


Line graph

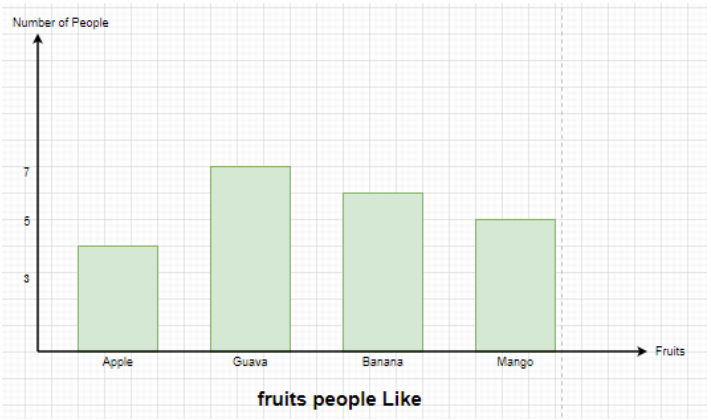
-

- **Bar Graph**

A bar chart or bar graph is a chart that represents categorical data with rectangular bars with heights proportional to the values that they represent. Here one axis of the chart plots categories and the other axis represents the value scale. The bars are of equal width which allows for instant comparison of data.



Bar graph



In order to read a Bar graph, we need to ask questions to ourselves looking at the displayed graph

What does the X-axis and Y-axis on the graph are representing?
The X-axis represents the different types of fruits like apple, guava. while Y-axis represents the Number of people.
What is the Common base for the Bars?
The bars are showing a common base of category of fruits.
What is the scale used on the Y-axis?
The scale used is normal, i.e; 1 Unit = 1 person
Overall, what kind of information the bar graph displaying?
The bar graph is displaying the number of People liking different types of fruits.
Looking at the bar Graph, can one answer, how many people like Mango?
Yes, By observing the length of the bar, one can tell that there are 5 people who like Mango.

Properties of Bar Graph

- All Bars have a common base.
- The length of each bar corresponds to its respective data mentioned on the axis (Y-axis for Vertical Graph, X-axis for Horizontal Graph).
- Each bar displayed has the same width.

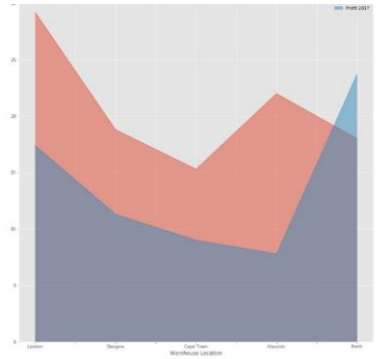
- The distance between consecutive bars is the same

Significance of a Bar Graph

It is always easier and more comfortable to visually understand something than to look at the large table of Numerical data. Bar graphs are extensively used in presentations and reports. It is very prominently used as it summarizes data and displays it in a frequency distribution.

- **Area Graph**

  An area chart or area graph displays graphically quantitative data. It is based on the line chart. The area between axis and line are commonly emphasized with colors, textures and hatchings.



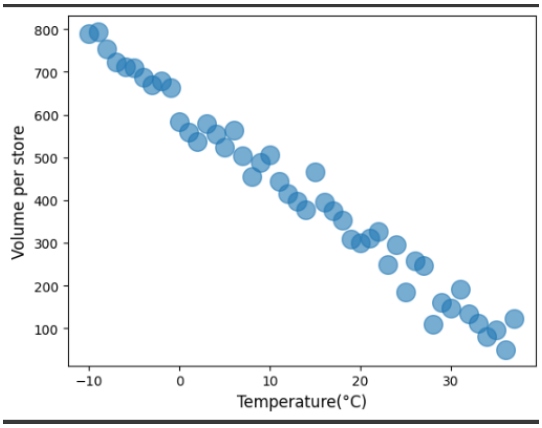
Area graph

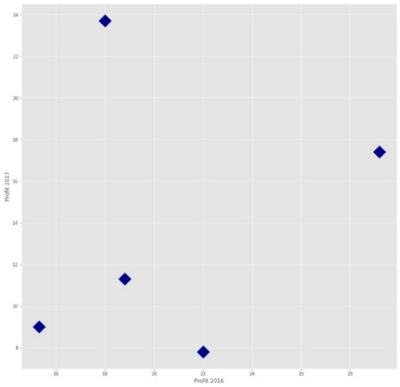
- **Scatter Graph**

Scatter plots are used when you want to show the relationship between two variables. Scatter plots are sometimes called correlation plots because they show how two variables are correlated.

Scatter plots are used when you want to show the relationship between two variables. Scatter plots are sometimes called correlation plots because they show how two variables are correlated.

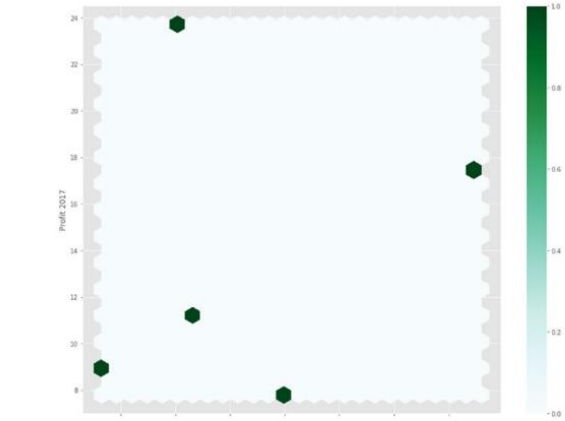
```
import numpy as np
import matplotlib.pyplot as plt
plt.scatter(x=range(-10, 38, 1), y=range(770, 60, -15)-
np.random.randn(48)*40,s=200,alpha=0.6)
plt.xlabel('Temperature(°C)', size=12)
plt.ylabel('Volume per store', size=12)
```





- *Scatter graph*

- **Hex Bin Graph**



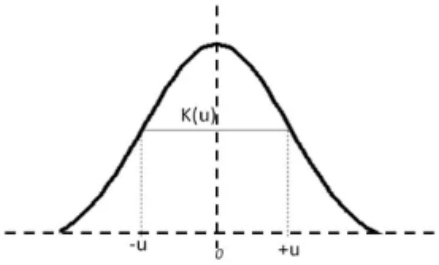
Hex bin graph for farm

- **Kernel Density Estimation (KDE) Graph**

What Is a Kernel?

Kernel is simply a function which satisfies following three properties as mentioned below. Kernel functions are used to estimate density of random variables and as weighing function in non-parametric regression. This function is also used in machine learning as kernel method to perform classification and clustering.

1. The first property of a kernel function is that **it must be symmetrical**. This means the values of kernel function is same for both +u and –u as shown in the plot below. This can be mathematically expressed as K (-u) = K (+u).



2. The symmetric property of kernel function enables the maximum value of the function (*max(K(u)*) to lie in the middle of the curve.

$$\int_{-\infty}^{+\infty} K(u)du = 1$$

3. The **area under the curve of the function must be equal to one**. Mathematically, this property is expressed as

Gaussian density function is used as a kernel function because the area under Gaussian density curve is one and it is symmetrical too.

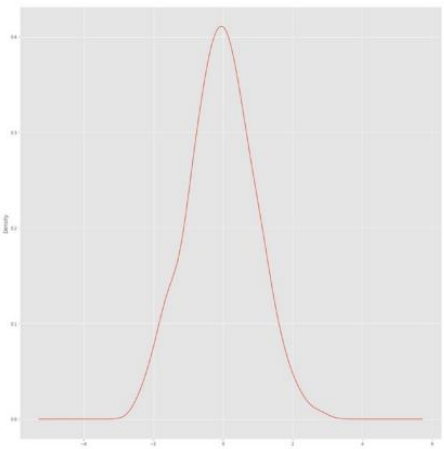
4. The value of kernel function, which is the density, **can not be negative**, $K(u) \geq 0$ for all $-\infty < u < \infty$.

It is used for visualizing the Probability Density of a continuous variable. It depicts the probability density at different values in a continuous variable.

KDE is a composite function made up of one kind of building block referred to as a kernel function.

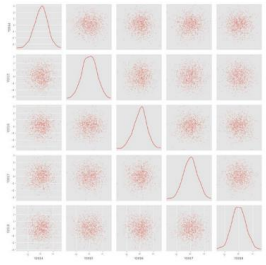
The kernel function is evaluated for each datapoint separately, and these partial results are summed to form the KDE.

```
import sys
import os
import pandas as pd
import matplotlib as ml
import numpy as np
from matplotlib import pyplot as plt
ml.style.use('ggplot')
fig1=plt.figure(figsize=(10, 10))
ser = pd.Series(np.random.randn(1000))
ser.plot(figsize=(10, 10),kind='kde')

sPicNameOut1='/content/kde.png'
plt.savefig(sPicNameOut1,dpi=600)
plt.tight_layout()
plt.show()
```



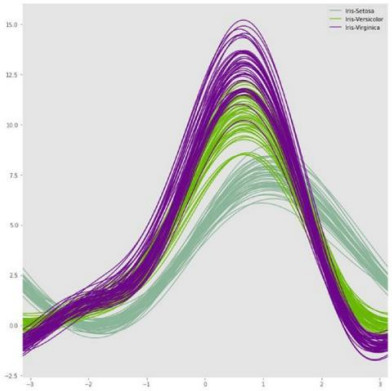
Kernel density estimation graph

- **Scatter Matrix Graph**
- A scatter plot matrix visualizes bivariate relationships between combinations of numeric variables.



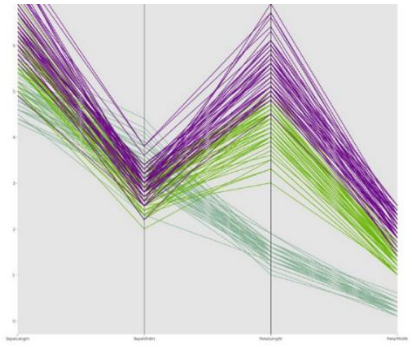
Scatter matrix graph

- **Andrews' Curves**



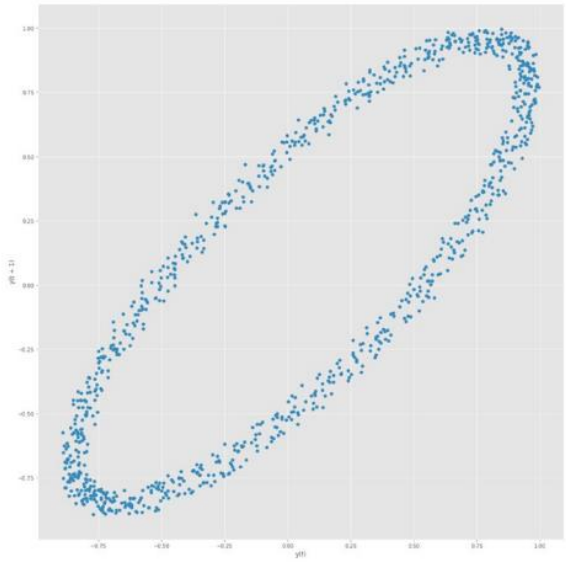
Graph produced using Andrews' curves

- **Parallel Coordinates**



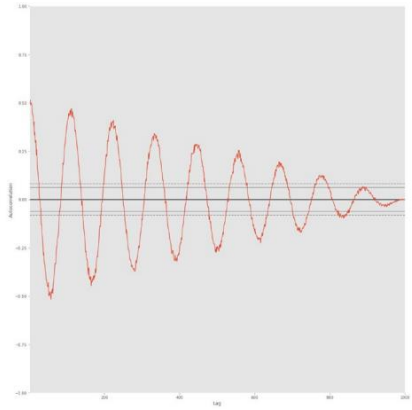
- *Graph produced using parallel coordinates*

- **Lag Plot**



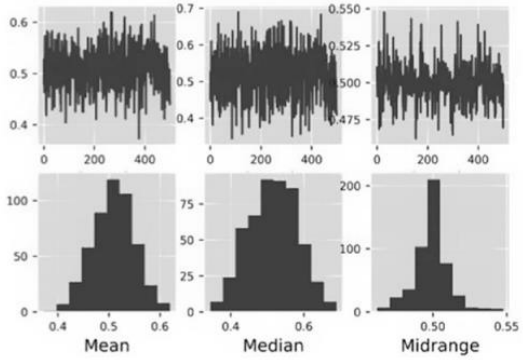
Lag plot graph

- **Autocorrelation Plot**
- An autocorrelation plot is a design of the sample autocorrelations vs. the time lags.



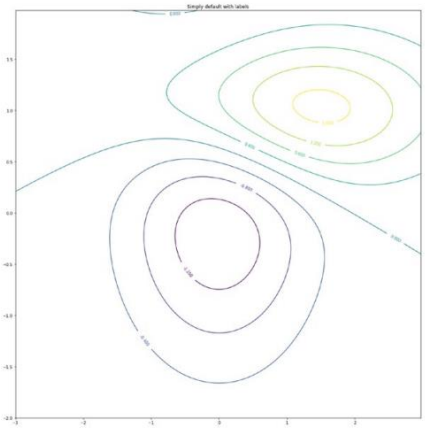
Autocorrelation plot graph

- **Bootstrap Plot**



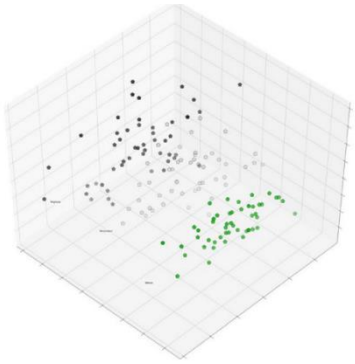
Bootstrap plot graph

- **Contour Graphs**



Contour plot graph

- **3D Graphs**



Three-dimensional graph

PICTURES

- Pictures are an interesting data science specialty. The processing of movies and pictures is a science on its own.

Channels of Images

- The interesting fact about any picture is that it is a complex data set in every image. Pictures are built using many layers or channels that assists the visualization tools to render the required image.

```
import sys
import os
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
```

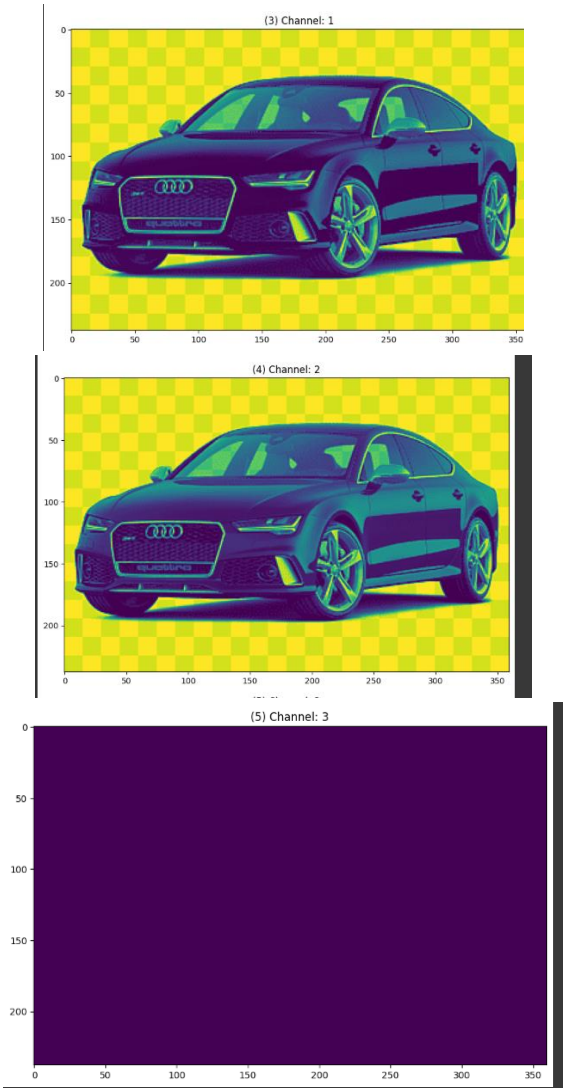
The image module in matplotlib library is used for working with images in Python.

The image module also includes two useful methods which are imread which is used to read images and imshow which is used to display the image.
The shape of an image is accessed by img.shape. It returns a tuple of the number of rows, columns, and channels (if the image is color):

If an image is grayscale, the tuple returned contains only the number of rows and columns, so it is a good method to check whether the loaded image is grayscale or color.

```
import sys
import os
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
sPicName='/content/1.png'
t=0
img=mpimg.imread(sPicName)
print('Size:', img.shape)
plt.figure(figsize=(10, 10))
t+=1
sTitle= '(' + str(t) + ') Original'
plt.title(sTitle)
plt.imshow(img)
plt.show()
for c in range(img.shape[2]):
 t+=1
 plt.figure(figsize=(10, 10))
 sTitle= '(' + str(t) + ') Channel: ' + str(c)
 plt.title(sTitle)
 lum_img = img[:,:,c]
 plt.imshow(lum_img)
 plt.show()
```

OUTPUT



OBSERVATIONS

- You can clearly see that the image is of the following size: Size: (238, 360, 4)
- Number of channels = 4
- This means you have 238 × 360 pixels per channel.
- That is a total of 85,680 pixels per layer.

CUTTING THE EDGES

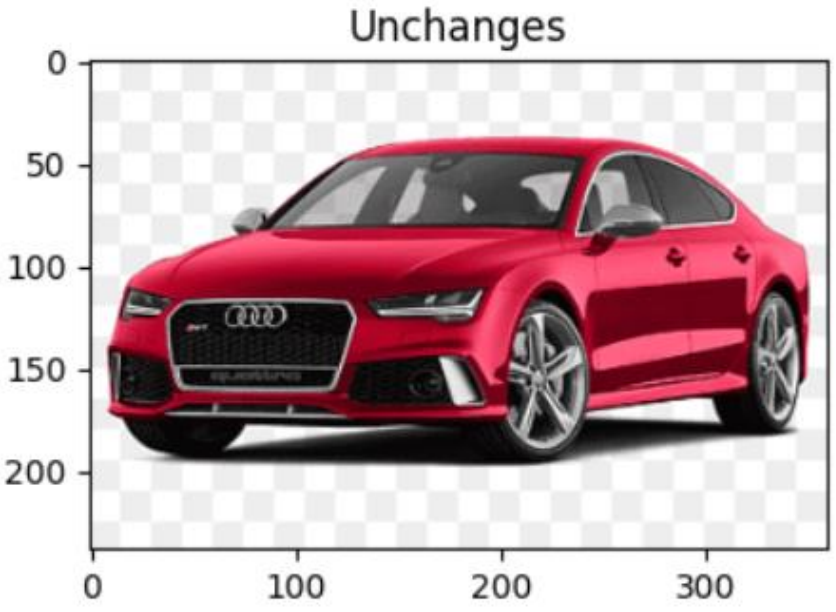
- One of the most common techniques that most data science projects require is the determination of the edge of an item's image. This is useful in areas such as robotics object selection and face recognition.

```
import sys
import os
import matplotlib.pyplot as plt
from PIL import Image

sPicNameIn='/content/1.png'
sPicNameOut='/content/audi.png'
imageIn = Image.open(sPicNameIn)
fig1=plt.figure(figsize=(10, 10))
fig1.suptitle('Audi R8', fontsize=20)
imgplot = plt.imshow(imageIn)
mask=imageIn.convert("L")
th=49


imageOut = mask.point(lambda i: i < th and 255)
imageOut.save(sPicNameOut)

imageTest = Image.open(sPicNameOut)
fig2=plt.figure(figsize=(10, 10))
fig2.suptitle('Audi R8 Edge', fontsize=20)
imgplot = plt.imshow(imageTest)
```

- The images we get to process are mostly of different sizes and quality. You will have to size images to specific sizes for most of your data science.

WHAT HAPPENS TO AN IMAGE IF YOU REDUCE A PIXEL QUALITY

```
import sys
import os
import matplotlib.pyplot as plt
from PIL import Image
sPicName='/content/1.png'
nSize=4
img = Image.open(sPicName)
plt.figure(figsize=(nSize, nSize))
sTitle='Unchanges'
plt.title(sTitle)
imgplot = plt.imshow(img)
```



You now apply a thumbnail function that creates a 64 × 64 pixel thumbnail image.

APPLY A THUMBNAIL FUNCTION

```
#thumbnail

img.thumbnail((64, 64), Image.ANTIALIAS)
 # resizes image in-place
plt.figure(figsize=(nSize, nSize))
sTitle='Resized'
plt.title(sTitle)
imgplot = plt.imshow(img)
plt.figure(figsize=(nSize, nSize))
sTitle='Resized with Bi-Cubic'
plt.title(sTitle)
imgplot = plt.imshow(img, interpolation="bicubic")
print('### Done!! ############################################')
```

Interpolation in Python is a technique used to estimate unknown data points between two known data points.
Bicubic interpolation determines the pixel value from the weighted average of the 16 closest neighboring pixels.
In Python, Interpolation is a technique mostly used to impute missing values in the data frame or series while preprocessing data.