UNIT III SUPERVISED LEARNING

Introduction to machine learning – Linear Regression Models: Least squares, single & multiple variables, Bayesian linear regression, gradient descent, Linear Classification Models: Discriminant function – Probabilistic discriminative model - Logistic regression, Probabilistic generative model – Naive Bayes, Maximum margin classifier – Support vector machine, Decision Tree, Random forests

Introduction to machine learning

Machine learning is a field of inquiry devoted to understanding and building methods that "learn" – that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificialintelligence.

Machine learning is a growing technology which enables computers to learn automatically from past data. Machine learning uses various algorithms for building mathematical models and making predictions using historical data or information. Currently, it isbeing used for various tasks such as image recognition, speech recognition, email filtering, Facebook auto-tagging, recommender system, and many more.

This machine learning tutorial gives you an introduction to machine learning along with the wide range of machine learning techniquessuch as Supervised, Unsupervised, and Reinforcement learning. You will learn about regression and classification models, clustering methods, hidden Markov models, and various sequential models.

What is Regression?

- Define regression with example(2M,8M)
- Application of regression in real life(2M)

Regression allows researchers to predict or explain the variation in one variable based on another variable.

The variable that researchers are trying to explain or predict is called the response variable. It is also sometimes called the dependent variable because it depends on another variable.

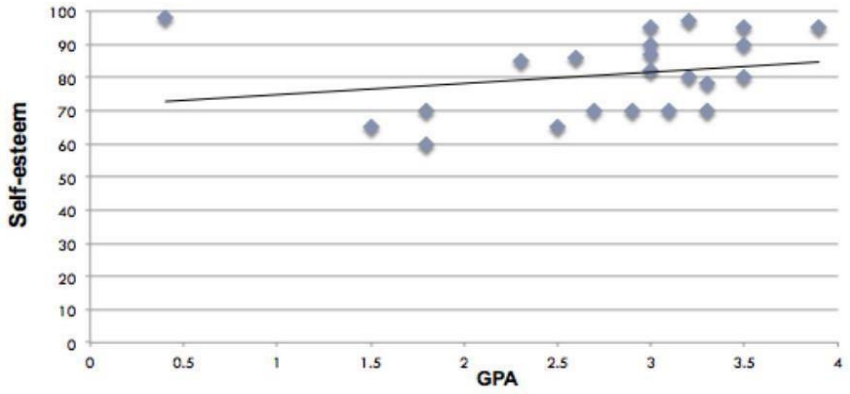
The variable that is used to explain or predict the response variable is called the explanatory variable. It isalso sometimes called the independent variable because it is independent of the other variable.

In regression, the order of the variables is very important. The explanatory variable (or the independent variable) always belongs on the x-axis. The response variable (or the dependent variable) always belongs on the y-axis.

Example:

If it is already known that there is a significant correlation between students' GPA and their self-esteem, thenext question researchers might ask is: Can students' scores on a self-esteem scale be predicted based on GPA? In other words, does GPA explain self-esteem? These are the types of questions that regression responds to.

**Note that these questions do not imply a causal relationship. In this example, GPA is the explanatory variable (or the independent variable) and self-esteem is the response variable (or the dependent variable). GPA belongs on the x-axis and self-esteem belongs on the y-axis.



Regression is essential for any machine learning problem that involves continuous numbers, which includes a vast array of real-life applications:

1. Financial forecasting, such as estimating housing or stock prices
2. Automobile testing
3. Weather analysis
4. Time series forecasting

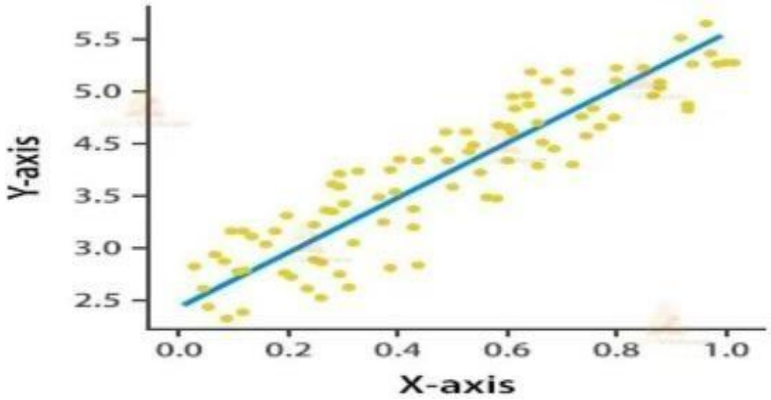
Types of Regression

- Linear Regression
- Logistic Regression
- Polynomial Regression
- Stepwise Regression
- Ridge Regression
- Lasso Regression
- Elastic Net Regression

LINEAR REGRESSION:

Simple linear regression is useful for finding relationship between two continuous variables. One is predictoror independent variable and other is response or dependent variable. It looks for statistical relationship but not deterministic relationship. Relationship between two variables is said to be deterministic if one variable can be accurately expressed by the other. For example, using temperature in degree Celsius it is possible to accurately predict Fahrenheit. Statistical relationship is not accurate in determining relationship between twovariables. For example, relationship between height and weight.

The core idea is to obtain a line that best fits the data. The best fit line is the one for which total predictionerror (all data points) are as small as possible. Error is the distance between the point to the regression line.



Calculate the regression coefficient and obtain the lines of regression for the following data

X	1	2	3	4	5	6	7
Y	9	8	10	12	11	13	14

Solution:

X	Y	X^2	Y^2	XY
1	9	1	81	9
2	8	4	64	16
3	10	9	100	30
4	12	16	144	48
5	11	25	121	55
6	13	36	169	78
7	14	49	196	98
$\sum X = 28$	$\sum Y = 77$	$\sum X^2 = 140$	$\sum Y^2 = 875$	$\sum XY = 334$

Table 9.7

$\overline{X} = \frac{\Sigma X}{N} = \frac{28}{7} = 4$,

$\overline{Y} = \frac{\Sigma Y}{N} = \frac{77}{7} = 11$

Regression coefficient of X on Y

$$b_{xy} = \frac{N\Sigma XY - (\Sigma X)(\Sigma Y)}{N\,\Sigma Y^2 - (\Sigma Y)^2}$$

$$= \frac{7(334) - (28)(77)}{7(875) - (77)^2}$$

$$= \frac{2338 - 2156}{6125 - 5929}$$

$$= \frac{182}{196}$$

$$b_{xy} = 0.929$$

Regression equation of X on Y

$$X - \overline{X} = b_{xy}(Y - \overline{Y})$$

$$X - 4 = 0.929(Y - 11)$$

$$X - 4 = 0.929Y - 10.219$$

$\therefore$ The regression equation X on Y is $X = 0.929Y - 6.219$

LEAST SQUARE METHOD:

The least squares method is a form of mathematical regression analysis used to determine the line of bestfit for a set of data, providing a visual demonstration of the relationship between the data points. Each point of data represents the relationship between a known independent variable and an unknown dependent variable. This method of regression analysis begins with a set of data points to be plotted on an x- and y-axis graph. An analyst using the least squares method will generate a line of best fit that explains the potential relationship between independent and dependent variables.

The least squares method is used in a wide variety of fields, including finance and investing. For financial analysts, the method can help to quantify the relationship between two or more variables—such as a stock's share price and its earnings per share (EPS). By performing this type of analysis investors often tryto predict the future behavior of stock prices or other factors.

FORMULA TO CALCULATE LEAST SQUARE REGRESSION:

The regression line under the Least Squares method is calculated using the following formula –

$y = a + bx$

Where,

y = dependent variable

x = independent variable

a = y-intercept

b = slope of the line

The slope of line b is calculated using the following formula :

$$b = \frac{\sum(x-\bar{x})\,(y-\bar{y})}{\sum(x-\bar{x})^2}$$

Or

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

Y-intercept, ‘a’ is calculated using the following formula –

$$a = \frac{\sum y - (b \sum x)}{n}$$

LEAST SQUARE REGRESSION LINE:

If the data shows a leaner relationship between two variables, the line that best fits this linear relationshipis known as a least-squares regression line, which minimizes the vertical distance from the data points to the regression line. The term “least squares” is used because it is the smallest sum of squares of errors, which is also called the "variance."

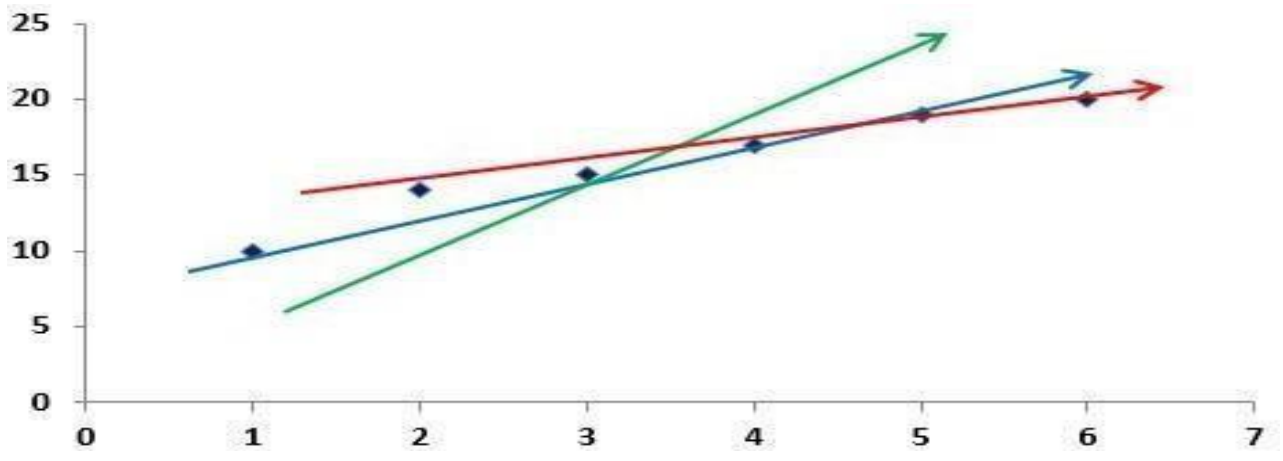
In regression analysis, dependent variables are illustrated on the vertical y-axis, while independent variables are illustrated on the horizontal x-axis. These designations will form the equation for the line ofbest fit, which is determined from the least squares method.

In contrast to a linear problem, a non-linear least-squares problem has no closed solution and is generallysolved by iteration.

EXAMPLE:

The line of best fit is a straight line drawn through a scatter of data points that best represents the relationship between them.

Let us consider the following graph wherein a set of data is plotted along the x and y-axis. These data points are represented using the blue dots. Three lines are drawn through these points – a green, a red, and a blue line. The green line passes through a single point, and the red line passes through three data points.
However, the blue line passes through four data points, and the distance between the residual points to the blue line is minimal as compared to the other two lines.



In the above graph, the blue line represents the line of best fit as it lies closest to all the values and the distance between the points outside the line to the line is minimal (i.e., the distance between the residualsto the line of best fit – also referred to as the sums of squares of residuals). In the other two lines, the orange and the green, the distance between the residuals to the lines is greater as compared to the blue line.

MULTIPLE REGRESSION:

Multiple regression is a statistical technique that can be used to analyze the relationship between a single dependent variable and several independent variables. The objective of multiple regression analysis is to use the independent variables whose values are known to predict the value of the single dependent value. Each predictor value is weighed, the weights denoting their relative contribution to the overall prediction.

$$Y=a+b1X1+b2X3+\ldots+bnXn$$

Here Y is the dependent variable, and X1,…,Xn are the *n* independent variables. In calculating the weights, a, b1,…,bn, regression analysis ensures maximal prediction of the dependent variable from the set of independent variables. This is usually done by least squares estimation.

In the case of linear regression, although it is used commonly, it is limited to just one independent and one dependent variable. Apart from that, linear regression restricts the training data set and does not predict a non-linear regression.

For the same limitations and to cover them, we use multiple regression. It focuses on overcoming one particular limitation and that is allowing to analyze more than one independent variable.

Multiple regression equation

We will start the discussion by first taking a look at the linear regression equation:

$y = bx + a$

Where,

y is a dependent variable we need to find, x is an independent variable. The constants a and b drive the equation. But according to our definition, as the multiple regression takes several independent variables(x), so for the equation we will have multiple x values too:

$$y = b1x1 + b2x2 + \dots bnxn + a$$

Here, to calculate the value of the dependent variable y, we have multiple independent variables x1, x2,and so on. The number of independent variables can grow till n and the constant b with every variable denotes its numeric value. The purpose of the constant a is to denote the dependent variable's value in case when all the independent variable values turn to zero.

Example: A researcher decides to study students' performance at a school over a period of time. He observed that as the lectures proceed to operate online, the performance of students started to declineas well. The parameters for the dependent variable "decrease in performance" are various independent variables like "lack of attention, more internet addiction, neglecting studies" and much more.

So for the above example, the multiple regression equation would be:

y = b1 * attention + b2 * internet addiction + b3 * technology support + … bnxn + a

ASSUMPTIONS OF MULTIPLE REGRESSION ANALYSIS:

- The variables considered for the model should be relevant and the model should be reliable.
- The model should be linear and not non-linear.
- Variables must have a normal distribution
- The variance should be constant for all levels of the predicted variable.

BENEFITS OF MULTIPLE REGRESSION ANALYSIS:

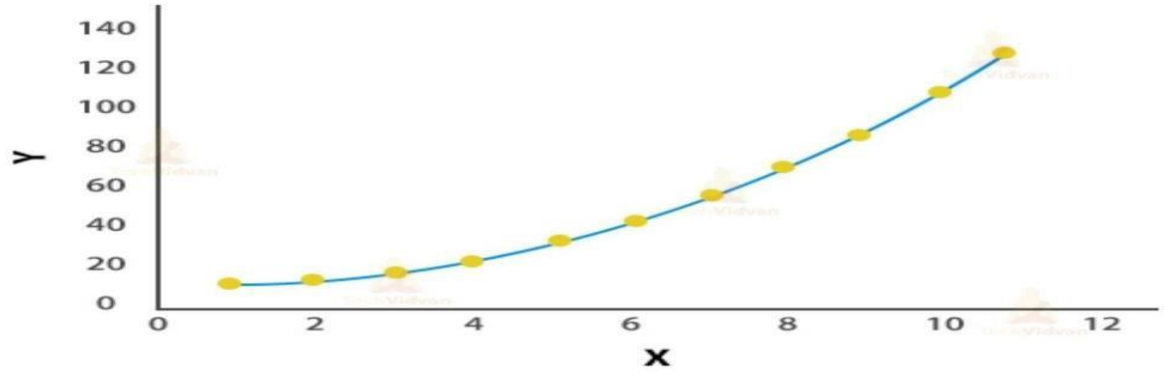
- Multiple regression analysis helps us to better study the various predictor variables at hand.
- It increases reliability by avoiding dependency on just one variable and having more than oneindependent variable to support the event.
- Multiple regression analysis permits you to study more formulated hypotheses that are possible.

Logistic regression

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based onprior observations of a data set.

A logistic regression model predicts a dependent data variable by analyzing the relationship between one ormore existing independent variables. For example, a logistic regression could be used to predict whether a political candidate will win or lose an election or whether a high school student will be admitted or not to a particular college. These binary outcomes allow straightforward decisions between two alternatives.

A logistic regression model can take into consideration multiple input criteria. In the case of college acceptance, the logistic function could consider factors such as the student's grade point average, SAT score and number of extracurricular activities. Based on historical data about earlier outcomes involving the same input criteria, it then scores new cases on their probability of falling into one of two outcome categories.



What Is Bayesian Linear Regression?

In Bayesian linear regression, the mean of one parameter is characterized by a weighted sum of other variables. This type of conditional modeling aims to determine the prior distribution of the regressors as well as other variables describing the allocation of the regress and eventually permits the out-of-sample forecasting of the regress and conditional on observations of the regression coefficients.

The normal linear equation, where the distribution of display style YY given by display style XX is Gaussian, isthe most basic and popular variant of this model. The future can be determined analytically for this model, and a specific set of prior probabilities for the parameters is known as conjugate priors. The posteriors usually have more randomly selected priors.

When the dataset has too few or poorly dispersed data, Bayesian Regression might be quite helpful. In contrast to conventional regression techniques, where the output is only derived from a single number of each attribute, aBayesian Regression model's output is derived from a probability distribution.

The result, "y," is produced by a normal distribution (where the variance and mean are normalized). The goal of the Bayesian Regression Model is to identify the 'posterior' distribution again for model parameters rather than the model parameters themselves. The model parameters will be expected to follow a distribution in addition to the output y.

The posterior expression is given below:

Posterior = (Likelihood * Prior)/Normalization

The expression parameters are explained below:

- Posterior: It is the likelihood that an event, such as H, will take place given the occurrence of another
- event, such as E, i.e., P(H | E).
- Likelihood: It is a likelihood function in which a marginalization parameter variable is used.

Priority: This refers to the likelihood that event H happened before event A, i.e., P(H) (H)

- This is the same as Bayes' Theorem, which states the following -

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

P(A) is the likelihood that event A will occur, while P(A|B) is the likelihood that event A will occur, providedthat event B has already occurred. Here, A and B seem to be events. P(B), the likelihood of event B happeningcannot be zero because it already has.

According to the aforementioned formula, we get a prior probability for the model parameters that is proportional to the probability of the data divided by the posterior distribution of the parameters, unlikeOrdinary Least Square (OLS), which is what we observed in the case of the OLS.

The value of probability will rise as more data points are collected and eventually surpass the previous value.The parameter values converge to values obtained by OLS in the case of an unlimited number of data points.Consequently, we start our regression method with an estimate (the prior value).

As we begin to include additional data points, the accuracy of our model improves. Therefore, to make a Bayesian Ridge Regression model accurate, a considerable amount of train data is required.

Let's quickly review the mathematical side of the situation now. If 'y' is the expected value in a linear model,then

$$y(w,x) = w0+w1x1+...+wpxp$$

where, The vector "w" is made up of the elements w0, w1,... The weight value is expressed as 'x'.

$$w=(w1\ldots wp)$$

As a result, the output "y" is now considered to be the Gaussian distribution around Xw for Bayesian Regression to produce a completely probabilistic model, as demonstrated below:

$$p(y|X, w. \alpha) = N(y|Xw, \alpha)$$

where the Gamma distribution prior hyper-parameter alpha is present. It is handled as a probability calculated from the data. The Bayesian Ridge Regression implementation is provided below.

The Bayesian Ridge Regression formula on which it is based is as follows:

$$p(y|\lambda)=N(w|0, \lambda^{-1}Ip)$$

where alpha is the Gamma distribution's shape parameter before the alpha parameter and lambda is thedistribution's shape parameter before the lambda parameter.

We have discussed Bayesian Linear Regression so, let us now discuss some of its real-life applications.

Real-life Application Of Bayesian Linear Regression

Some of the real-life applications of Bayesian Linear Regression are given below:

- **Using Priors:** Consider a scenario in which your supermarkets carry a new product, and we want to predict its initial Christmas sales. For the new product's Christmas effect, we may merely use the average of comparable things as a previous one.

Additionally, once we obtain data from the new item's initial Christmas sales, the previous is immediately updated. As a result, the forecast for the next Christmas is influenced by both the prior and the new item's data.

- **Regularize Priors:** With the season, day of the week, trend, holidays, and a tonne of promotion indicators, our model is severely over-parameterized. Therefore regularization is crucial to keep the forecasts in check.

Since we got an idea regarding the real-life applications of Bayesian Linear Regression, we will now learn aboutits advantages and disadvantages.

Advantages Of Bayesian Regression

Some of the main advantages of Bayesian Regression are defined below:

- Extremely efficient when the dataset is tiny.
- Particularly well-suited for online learning as opposed to batch learning, when we know the complete dataset before we begin training the model. This is so that Bayesian Regression can be used without having to save data.
- The Bayesian technique has been successfully applied and is quite strong mathematically. Therefore, using this requires no additional prior knowledge of the dataset.

Let us now look at some disadvantages of Bayesian Regression.

Disadvantages Of Bayesian Regression

Some common disadvantages of using Bayesian Regression:

- The model's inference process can take some time.
- The Bayesian strategy is not worthwhile if there is a lot of data accessible for our dataset, and the regular probability approach does the task more effectively.

After going through the definitions, applications, and advantages and disadvantages of Bayesian LinearRegression, it is time for us to explore how to implement Bayesian Regression using Python.

Implementation Of Bayesian Regression Using Python

We shall apply Bayesian Ridge Regression in this example. The Bayesian method, however, can be used in anyregression technique, including regression analysis, lasso regression, etc. To implement Probabilistic Ridge Regression, we'll use the sci-kit-learn library.

We'll make use of the Boston Housing dataset, which includes details on the average price of homes in various Boston neighborhoods.

The r2 score will be used for evaluation. The r2 score should be as high as 1.0. The value of the r2 score is zeroif the model predicts consistently independent of the attributes. Even inferior models may have a negative r2 score.

However, before we begin the coding, you must comprehend the crucial components of a Bayesian Ridge Regression model:

- n_iter: Quantity of iterations. The default value is 100.
- tol: How to know when to end the procedure after the model converges. 1e-3 is the default value.
- alpha_1: Alpha parameter over the Gamma distribution shape parameter of a regressor line. 1e-6 is the default value.
- alpha_2: Gamma distribution's inverse scale parameter relative to the alpha parameter. 1e-6 is the default value.
- lambda_1: Gamma distribution's shape parameter relative to lambda. 1e-6 is the default value.
- lambda_2: Gamma distribution's inverse scale parameter over the lambda variable. 1e-6 is the default value.

Supervised Learning : Important Concepts

Supervised learning: classification is seen as supervised learning from examples.

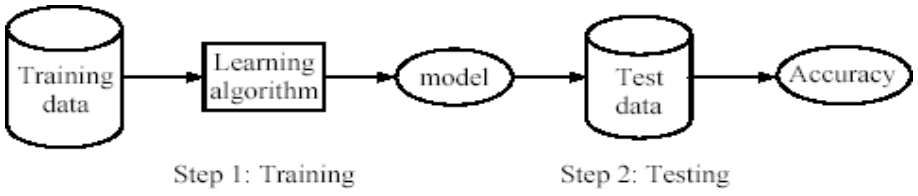
- ❖ Supervision: The data (observations, measurements, etc.) are labeled with pre-defined classes. It is like that a“teacher” gives the classes (supervision).
- ❖ Test data are classified into these classes too.

Unsupervised learning (clustering)

- ❖ Class labels of the data are unknown
- ❖ Given a set of data, the task is to establish the existence of classes or clusters in the data

Learning (training): Learn a model using the training data

Testing: Test the model using unseen test data to assess the model accuracy



Supervised Learning: Data and corresponding labels are given

Unsupervised Learning: Only data is given, no labels provided

Semi-supervised Learning: Some (if not all) labels are present

Reinforcement Learning: An agent interacting with the world makes observations, takes actions, and is rewarded or punished;it should learn to choose actions in such a way as to obtain a lot of reward.

Data: labeled instances <*xi, y*>, e.g. emails marked spam/not spamTraining Set Held-out SetTest Set

Features: attribute-value pairs which characterize each *x* Experimentation cycle Learn parameters (e.g. model probabilities) on training set (Tune hyper-parameters on held-out set)Compute accuracy of test set Very important: never "peek" at the test set!Evaluation

Accuracy: fraction of instances predicted correctlyOverfitting and generalization Want a classifier which does well on test data

Overfitting: fitting the training data very closely, but not generalizing well

An example application

- An emergency room in a hospital measures 17 variables (e.g., blood pressure, age, etc) of newly admitted patients.
- A decision is needed: whether to put a new patient in an intensive-care unit.
- Due to the high cost of ICU, those patients who may survive less than a month are given higher priority.
- Problem: to predict high-risk patients and discriminate them from low-risk patients.
- In classification, we predict labels *y* (classes) for inputs *x*

Examples:

- OCR (input: images, classes: characters)
- Medical diagnosis (input: symptoms, classes: diseases)
- Automatic essay grader (input: document, classes: grades)
- Fraud detection (input: account activity, classes: fraud / no fraud)
- Customer service email routing
- Recommended articles in a newspaper, recommended books
- DNA and protein sequence identification
- Categorization and identification of astronomical images
- Financial investments

… many more

An example: the learning task

- Learn a classification model from the data
- Use the model to classify future loan applications into
  - Yes (approved) and
  - No (not approved)
- What is the class for following case/instance?

Age	Has_Job	Own_house	Credit-Rating	Class
young	false	false	good	?

An example

Data: Loan application data

Task: Predict whether a loan should be approved or not.

Performance measure: accuracy.

No learning: classify all future applications (test data) to the majority class (i.e.,Yes):Accuracy = 9/15 = 60%.

We can do better than 60% with learning.

Fundamental assumption of learning

Assumption: The distribution of training examples is identical to the distribution of test examples (including future unseen examples).

- In practice, this assumption is often violated to certain degree.
- Strong violations will clearly result in poor classification accuracy.
- To achieve good accuracy on the test data, training examples must be sufficiently representative of the test data.

Decision tree ALGORITHM

- Decision tree learning is one of the most widely used techniques for classification.
- Its classification accuracy is competitive with other methods, and it is very efficient.
- The classification model is a tree, called decision tree.
- Basic algorithm (a greedy divide-and-conquer algorithm)
- Assume attributes are categorical now (continuous attributes can be handled too)
- Tree is constructed in a top-down recursive manner
- At start, all the training examples are at the root
- Examples are partitioned recursively based on selected attributes
- Attributes are selected on the basis of an impurity function (e.g., information gain)

- ❖ Conditions for stopping partitioning
- ❖ All examples for a given node belong to the same class
- ❖ There are no remaining attributes for further partitioning – majority class is the leaf
- ❖ There are no examples left

Principle

- Basic algorithm a greedy algorithm
- Tree is constructed in a *top-down recursive divide-and-conquer* manner

Iterations

- o At start, all the training tuples are at the root
- o Tuples are partitioned recursively based on selected attributes
- o Test attributes are selected on the basis of a heuristic or statistical measure (e.g, information gain)
- o Stopping conditions
- o All samples for a given node belong to the same class
- o There are no remaining attributes for further partitioning – – majority voting is employed for classifying the leaf
- o There are no samples left

Aim: find a small tree consistent with the training examples

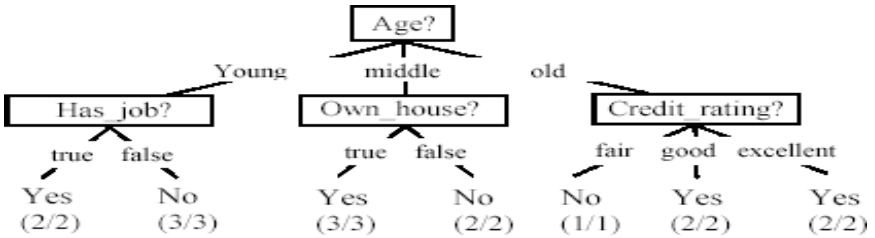
Idea: (recursively) choose "most significant" attribute as root of (sub)tree

```
function DTL(examples, attributes, default) returns a decision tree
    if examples is empty then return default
    else if all examples have the same classification then return the classification
    else if attributes is empty then return MODE(examples)
    else
        best ← CHOOSE-ATTRIBUTE(attributes, examples)
        tree ← a new decision tree with root test best
        for each value v_i of best do
            examples_i ← {elements of examples with best = v_i}
            subtree ← DTL(examples_i, attributes − best, MODE(examples))
            add a branch to tree with label v_i and subtree subtree
        return tree
```

A decision tree from the loan data

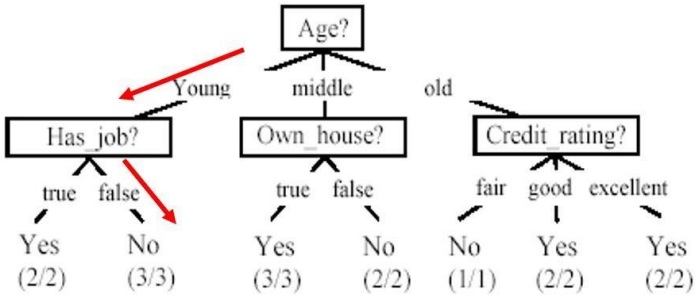
ID	Age	Has_Job	Own_House	Credit_Rating	Class
1	young	false	false	fair	**No**
2	young	false	false	good	**No**
3	young	true	false	good	**Yes**
4	young	true	true	fair	**Yes**
5	young	false	false	fair	**No**
6	middle	false	false	fair	**No**
7	middle	false	false	good	**No**
8	middle	true	true	good	**Yes**
9	middle	false	true	excellent	**Yes**
10	middle	false	true	excellent	**Yes**
11	old	false	true	excellent	**Yes**
12	old	false	true	good	**Yes**
13	old	true	false	good	**Yes**
14	old	true	false	excellent	**Yes**
15	old	false	false	fair	**No**

Decision nodes and leaf nodes (classes)



Use the decision tree

Age	Has_Job	Own_house	Credit-Rating	Class
young	false	false	good	?



From a decision tree to a set of rules

A decision tree can be converted to a set of rules Each path from the root to a leaf is a rule. Finally,Two possible roots, which is better?

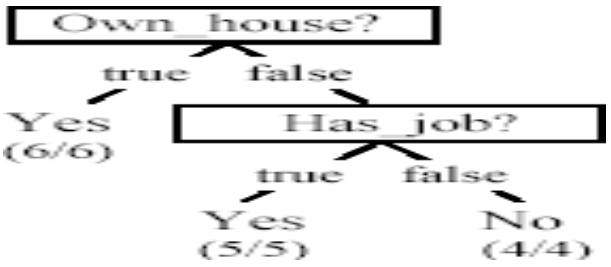
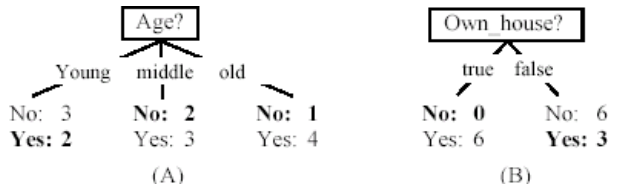


Fig. (B) seems to be better

Own_house = true → Class =Yes [sup=6/15, conf=6/6]
Own_house = false, Has_job = true → Class = Yes [sup=5/15, conf=5/5]
Own_house = false, Has_job = false → Class = No [sup=4/15, conf=4/4]



Learning decision trees:Ex2

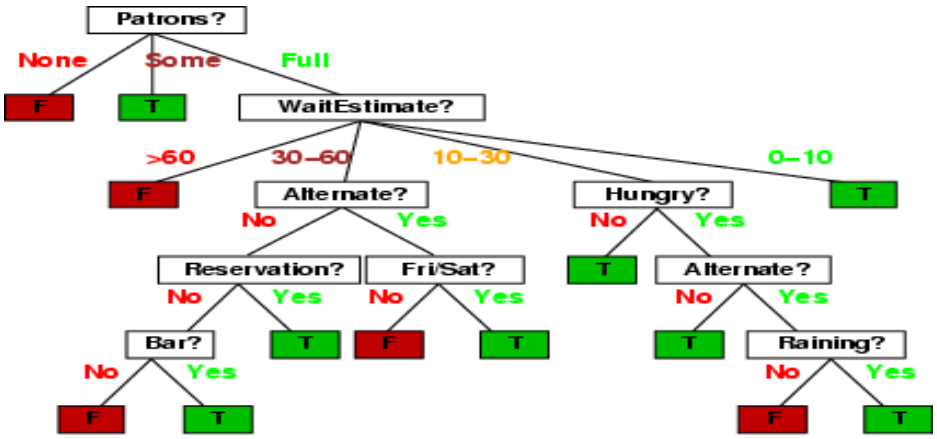
Example Problem: decide whether to wait for a table at a restaurant, based on the following attributes:

1. **Alternate**: is there an alternative restaurant nearby?
2. **Bar**: is there a comfortable bar area to wait in?
3. **Fri/Sat**: is today Friday or Saturday?
4. **Hungry**: are we hungry?
5. **Patrons**: number of people in the restaurant (None, Some, Full)
6. **Price**: price range ($, $$, $$$)
7. **Raining**: is it raining outside?
8. **Reservation**: have we made a reservation?
9. **Type**: kind of restaurant (French, Italian, Thai, Burger)
10. **WaitEstimate**: estimated waiting time (0-10, 10-30, 30-60, >60)

- Feature(Attribute)-based representations Examples described by feature(attribute) values
  - (Boolean, discrete, continuous)
- E.g., situations where I will/won't wait for a table:

Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*Wait*
X_1	T	F	F	T	Some	$$$	F	T	French	0–10	T
X_2	T	F	F	T	Full	$	F	F	Thai	30–60	F
X_3	F	T	F	F	Some	$	F	F	Burger	0–10	T
X_4	T	F	T	T	Full	$	F	F	Thai	10–30	T
X_5	T	F	T	F	Full	$$$	F	T	French	>60	F
X_6	F	T	F	T	Some	$$	T	T	Italian	0–10	T
X_7	F	T	F	F	None	$	T	F	Burger	0–10	F
X_8	F	F	F	T	Some	$$	T	T	Thai	0–10	T
X_9	F	T	T	F	Full	$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	$$$	F	T	Italian	10–30	F
X_{11}	F	F	F	F	None	$	F	F	Thai	0–10	F
X_{12}	T	T	T	T	Full	$	F	F	Burger	30–60	T

- Classification of examples is positive (T) or negative (F)
- One possible representation for hypotheses E.g., here is the “true” tree for deciding whether to wait:

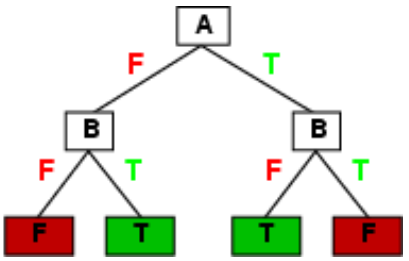


Expressiveness

Decision trees can express any function of the input attributes.

E.g., for Boolean functions, truth table row → path to leaf:

A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F



- Trivially, there is a consistent decision tree for any training set with one path to leaf for each example (unless *f* nondeterministic in *x*) but it probably won't generalize to new examples
- refer to find more compact decision trees

Evaluating classification methods

- **Predictive accuracy**

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}}$$

- Efficiency
  - time to construct the model
  - time to use the model
- Robustness: handling noise and missing values
- Scalability: efficiency in disk-resident databases
- Interpretability:
  - understandable and insight provided by the model
- Compactness of the model: size of the tree, or the number of rules.

Validation set: the available data is divided into three subsets,

- a training set,
- a validation set and
- a test set.

❖ A validation set is used frequently for estimating parameters in learning algorithms.

❖ In such cases, the values that give the best accuracy on the validation set are used as the final parameter values.

❖ Cross-validation can be used for parameter estimating as well.

- Classification measures
- Accuracy is only one measure (error = 1-accuracy).
- **Accuracy is not suitable in some applications**.
- In text mining, we may only be interested in the documents of a particular topic, which are only a small portion ofa big document collection.
- In classification involving skewed or highly imbalanced data, e.g., network intrusion and financial frauddetections, we are interested only in the minority class.
  - High accuracy does not mean any intrusion is detected.
  - E.g., 1% intrusion. Achieve 99% accuracy by doing nothing.
- The class of interest is commonly called the **positive class**, and the rest **negative classes.**
- **Precision** and **recall** measures Used in information retrieval and text classification.
- We use a confusion matrix to introduce them.

	Classified Positive	Classified Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

where

TP: the number of correct classifications of the positive examples (**true positive**),
FN: the number of incorrect classifications of positive examples (**false negative**),
FP: the number of incorrect classifications of negative examples (**false positive**), and
TN: the number of correct classifications of negative examples (**true negative**).

	Classified Positive	Classified Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

$$p = \frac{TP}{TP + FP}. \qquad r = \frac{TP}{TP + FN}.$$

- Precision *p* is the number of correctly classified positive examples divided by the total number of examples that are classified as positive.
- Recall *r* is the number of correctly classified positive examples divided by the total number of actual positive examples in the test set.

An example

	Classified Positive	Classified Negative
Actual Positive	1	99
Actual Negative	0	1000

- This confusion matrix gives
  - precision p = 100% and
  - recall r = 1%
  - because we only classified one positive example correctly and no negative examples wrongly.
- Note: precision and recall only measure classification on the positive class.

k-Nearest Neighbor Classification (kNN)

- To classify a test instance d, define k-neighborhood P as k nearest neighbors of d
- Count number n of training instances in P that belong to class cj
- Estimate Pr($cj|d$) as n/k
- No training is needed. Classification time is linear in training set size for

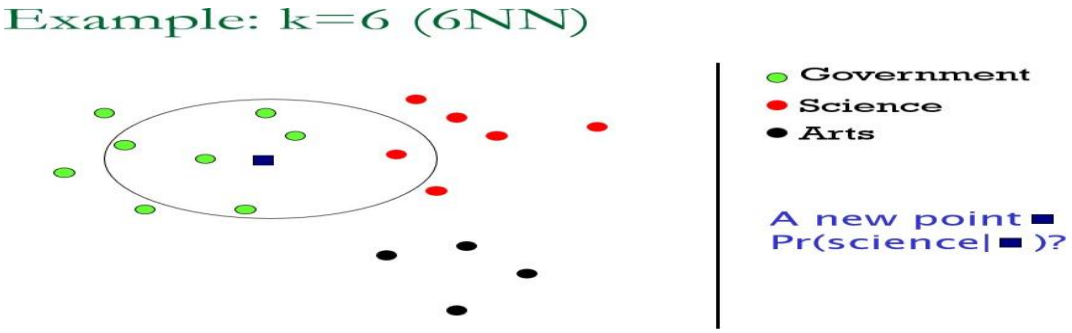
each test case.

kNNAlgorithm

Algorithm kNN(D, d, k)
1 Compute the distance between d and every example in D;
2 Choose the k examples in D that are nearest to d, denote the set by P ($\subseteq D$);
3 Assign d the class that is the most frequent class in P (or the majority class);

k is usually chosen empirically via a validation set or cross-validation by trying a range of k values.

Distance function is crucial, but depends on applications.



Discussions

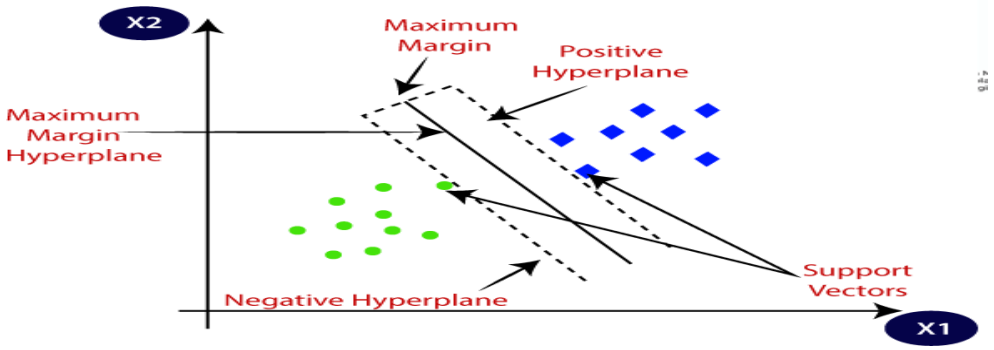
- kNN can deal with complex and arbitrary decision boundaries.
- Despite its simplicity, researchers have shown that the classification accuracy of kNN can be quite strong and in many cases as accurate as those elaborated methods.
- kNN is slow at the classification time
- kNN does not produce an understandable model

Support vector machines(SVM)

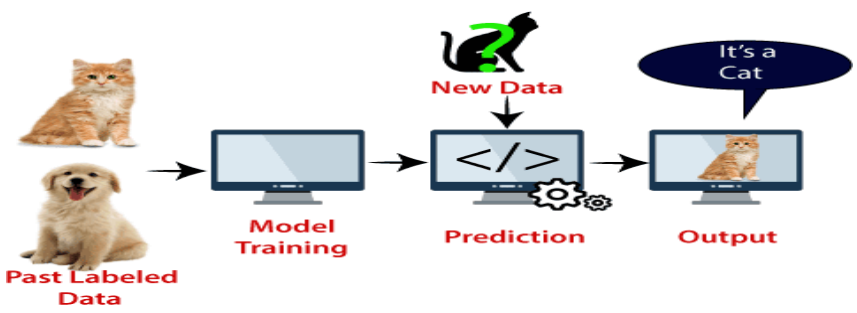
Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

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 whichthere 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 classifiedinto 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, butwe 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.

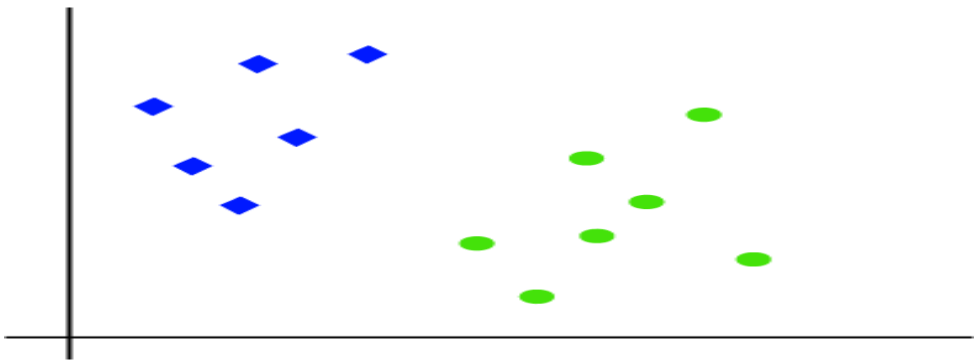
The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features(as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane. We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

Support Vectors:

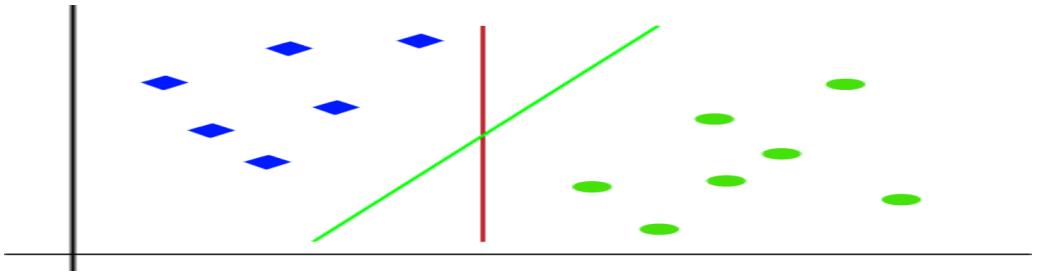
The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

How does SVM works?Linear SVM:

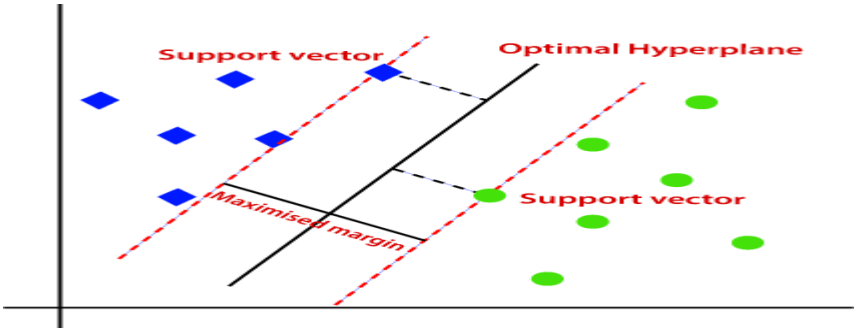
The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider the below image:



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:

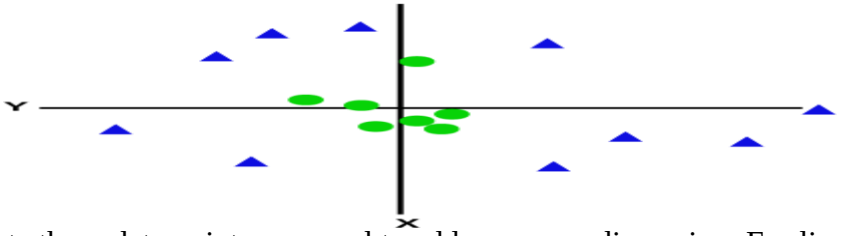


Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is calledas a hyperplane. SVM algorithm finds the closest point of the lines from both the classes. These points are calledsupport vectors. The distance between the vectors and the hyperplane is called 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:

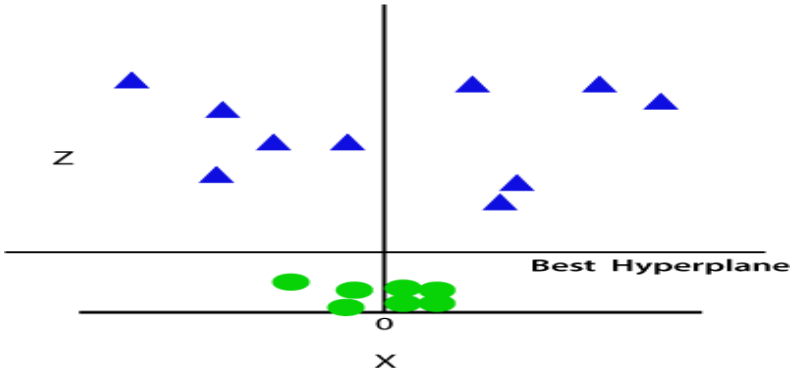
If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot drawa single straight line. Consider the below image:



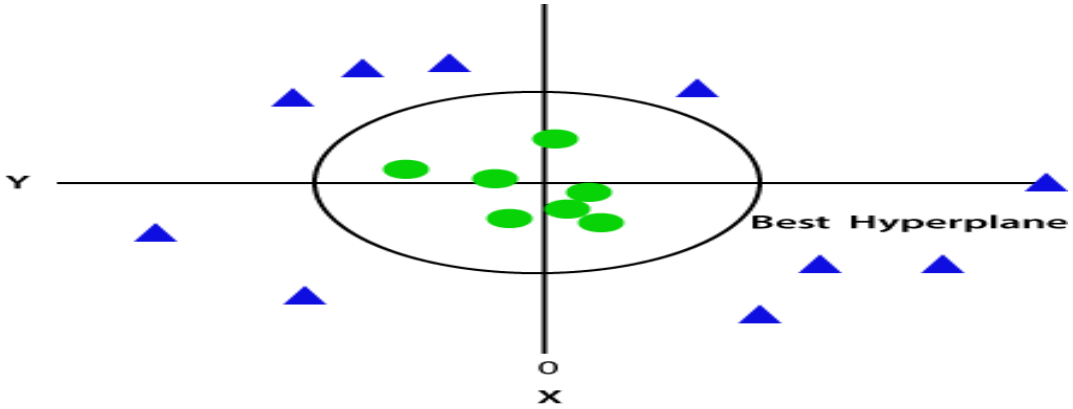
So to separate these data points, we need to add one more dimension. For linear data, we have used two dimensionsx and y, so for non-linear data, we will add a third dimension z. It can be calculated as:

$z=x^2 +y^2$

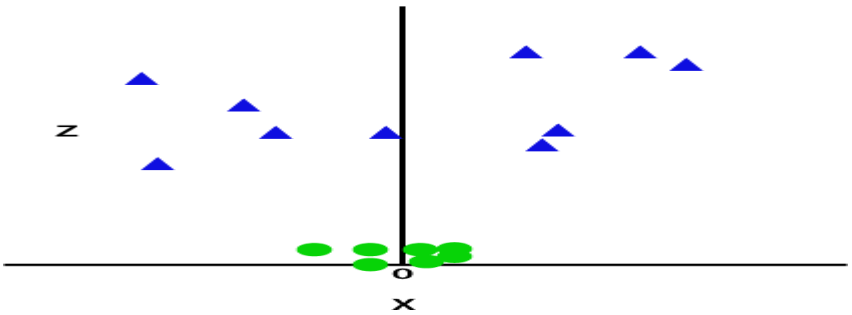
By adding the third dimension, the sample space will become as below image:



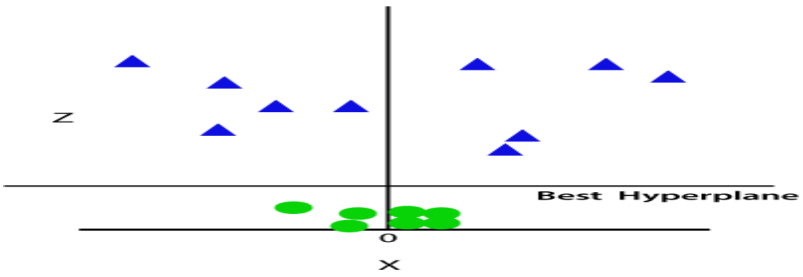
Since we are in 3-d Space, hence it is looking like a plane parallel to the x-axis. If we convert it in 2d space withz=1, then it will become as:



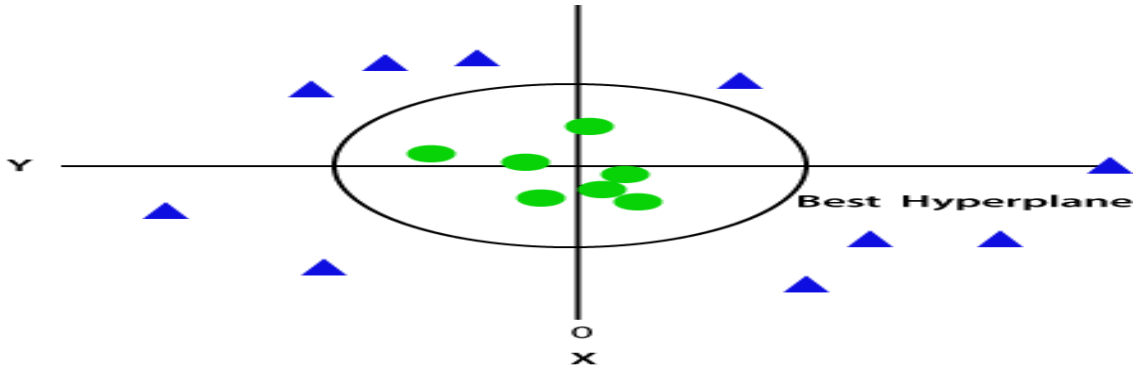
Hence we get a circumference of radius 1 in case of non-linear data.



So now, SVM will divide the datasets into classes in the following way. Consider the below image:



Since we are in 3-d Space, hence it is looking like a plane parallel to the x-axis. If weconvert it in 2d space with z=1, then it will become as:



Applications of SVM in Real World

As we have seen, SVMs depends on supervised learning algorithms. The aim of using SVM is to correctly classify unseen data. SVMs have a number of applications in several fields. Some common applications of SVM are-

- Face detection – SVMc classify parts of the image as a face and non-face and create a square boundary around the face.
- Text and hypertext categorization – SVMs allow Text and hypertext categorization for both inductive and transductive models. They use training data to classify documents into different categories. It categorizes on the basis of the score generated and then compares with the thresholdvalue.
- Classification of images – Use of SVMs provides better search accuracy for image classification.It provides better accuracy in comparison to the traditional query-based searching techniques.
- Bioinformatics – It includes protein classification and cancer classification. We use SVM foridentifying the classification of genes, patients on the basis of genes and other biological problems.
- Protein fold and remote homology detection – Apply SVM algorithms for protein remotehomology detection.
- Handwriting recognition – We use SVMs to recognize handwritten characters used widely.
- Generalized predictive control(GPC) – Use SVM based GPC to control chaotic dynamics withuseful parameters.

Random Forest Algorithm

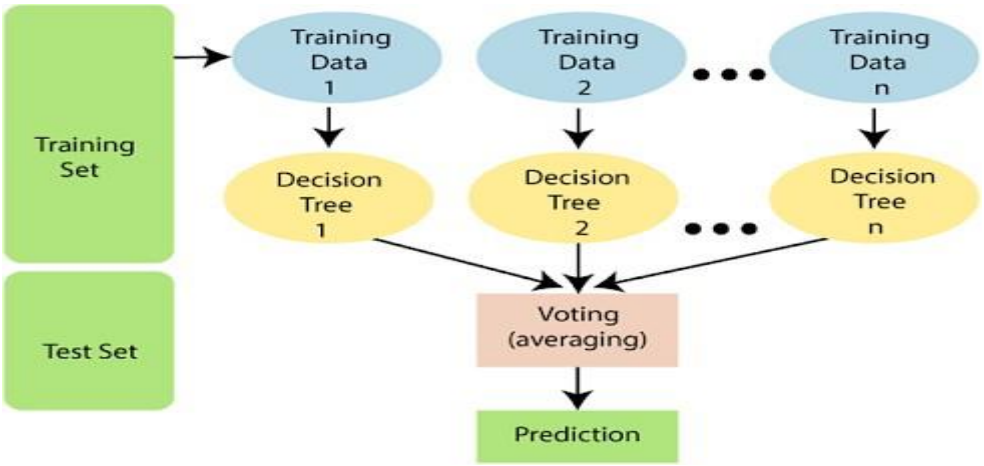
Random Forest is a supervised machine learning algorithm made up of decision trees. Random Forest is used forboth classification and regression—for example, classifying whether an email is “spam” or “not spam”.

It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

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 following steps explain the working Random Forest Algorithm:

Working of Random Forest Algorithm



How does Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and secondis to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

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.

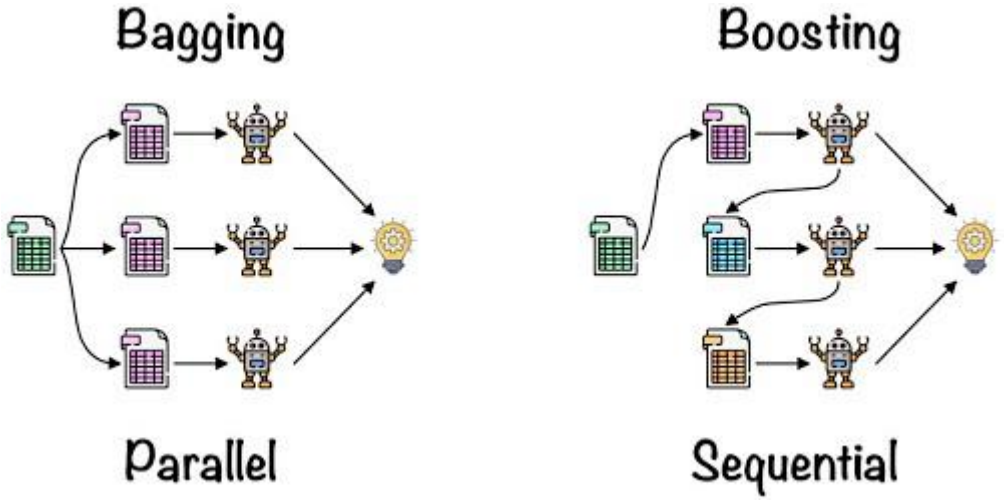
Step-4: Repeat Step 1 & 2.

Finally, select the most voted prediction result as the final prediction result.

This combination of multiple models is called Ensemble. Ensemble uses two methods:

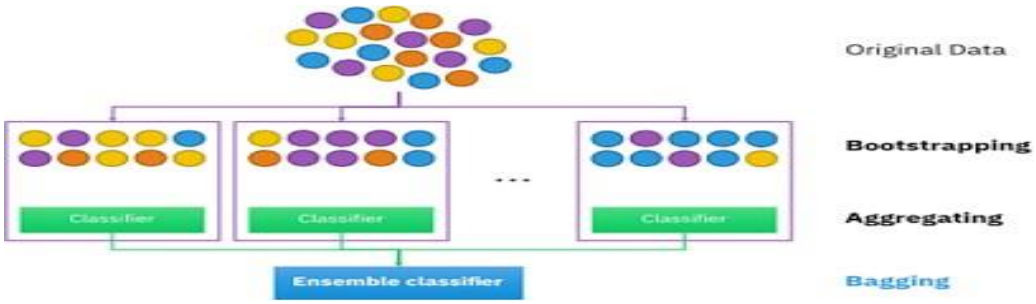
1. Bagging: Creating a different training subset from sample training data with replacement is called Bagging. The final output is based on majority voting.

2. Boosting: Combing weak learners into strong learners by creating sequential models such that the final model has the highest accuracy is called Boosting.



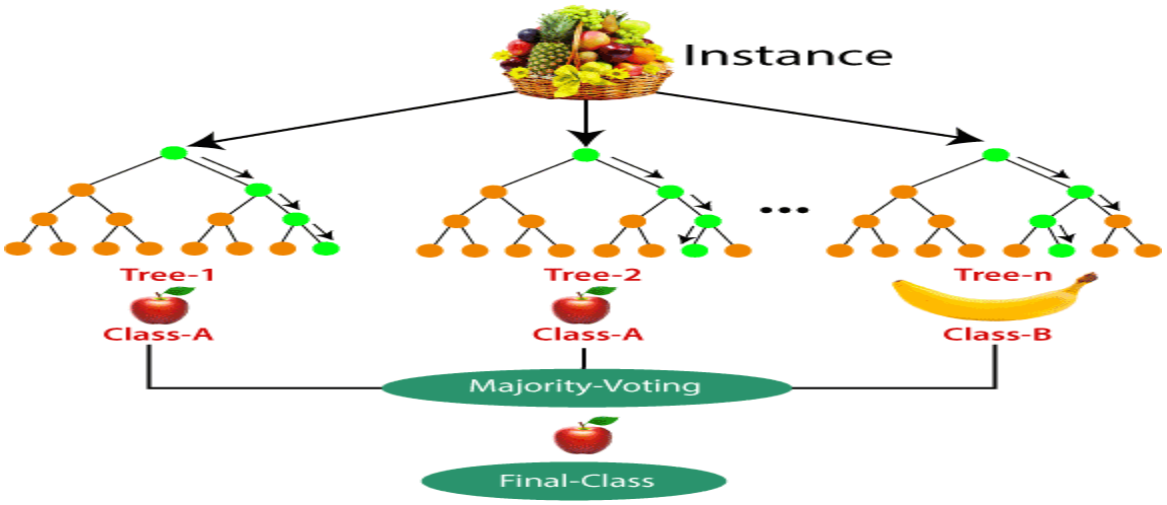
Bagging: From the principle mentioned above, we can understand Random Forest uses the Bagging code. Now, let us understand this concept in detail. Bagging is also known as Bootstrap Aggregation used by random forest.

The process begins with any original random data. After arranging, it is organised into samples known as Bootstrap Sample. This process is known as Bootstrapping. Further, the models are trained individually, yieldingdifferent results known as Aggregation. In the last step, all the results are combined, and the generated output is based on majority voting. This step is known as Bagging and is done using an Ensemble Classifier.



The working of the algorithm can be better understood by the below example:

Example: 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.

Advantages of Random Forest

- Random Forest is capable of performing both Classification and Regression tasks.
- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.

Disadvantages of Random Forest

- Although random forest can be used for both classification and regression tasks, it is not more suitablefor Regression tasks.

Implementation Steps are given below:

- Data Pre-processing step
- Fitting the Random Forest algorithm to the Training set
- Predicting the test result
- Test accuracy of the result (Creation of Confusion matrix)
- Visualizing the test set result.

- Difference Between Decision Tree and Random Forest
- Random forest is a collection of decision trees; still, there are a lot of differences in their behavior.

Decision trees	**Random Forest**
1. Decision trees normally suffer from the problemof overfitting if it's allowed to grow without any control.	1. Random forests are created from subsets of data, and the final output is based on average or majority ranking; hence the problem ofoverfitting is taken care of.
2. A single decision tree is faster in computation.	2. It is comparatively slower.
3. When a data set with features is taken as input by a decision tree, it will formulate some rules tomake predictions.	3. Random forest randomly selects observations, builds a decision tree, and takes the average result. It doesn't use any set of formulas.