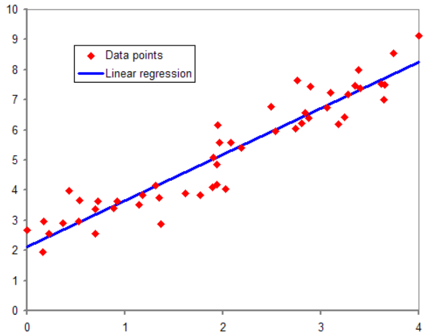
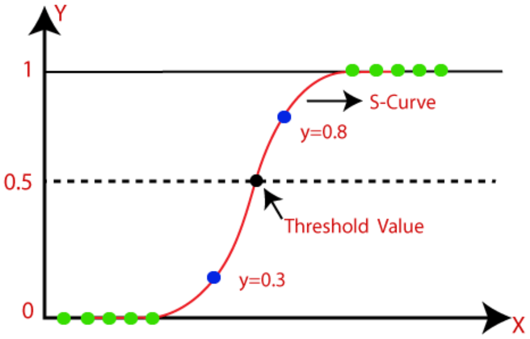
DATA ANALYTICS

(Professional Elective - I)
Subject Code: CS513PE

NOTES MATERIAL

UNIT 3





For B. TECH (CSE)

3rd YEAR – 1st SEM (R18)

Faculty:

B. RAVIKRISHNA

DEPARTMENT OF CSE

VIGNAN INSTITUTE OF TECHNOLOGY & SCIENCE

DESHMUKHI

UNIT - III
Linear & Logistic Regression

Syllabus

Regression – Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

Topics:

1. Regression – Concepts
2. Blue property assumptions
3. Least Square Estimation
4. Variable Rationalization
5. Model Building etc.
6. Logistic Regression - Model Theory
7. Model fit Statistics
8. Model Construction
9. Analytics applications to various Business Domains

Unit-2 Objectives:

1. To explore the Concept of Regression
2. To learn the Linear Regression
3. To explore Blue Property Assumptions
4. To Learn the Logistic Regression
5. To understand the Model Theory and Applications

Unit-2 Outcomes:

After completion of this course students will be able to

1. To Describe the Concept of Regression
2. To demonstrate Linear Regression
3. To analyze the Blue Property Assumptions
4. To explore the Logistic Regression
5. To describe the Model Theory and Applications

Regression – Concepts:

Introduction:

- The term regression is used to indicate the estimation or prediction of the average value of one variable for a specified value of another variable.
- Regression analysis is a very widely used statistical tool to establish a relationship model between two variables.

"Regression Analysis is a statistical process for estimating the relationships between the

Dependent Variables /Criterion Variables / Response Variables

&

One or More Independent variables / Predictor variables.

- Regression describes how an independent variable is numerically related to the dependent variable.
- Regression can be used for prediction, estimation and hypothesis testing, and modeling causal relationships.

When Regression is chosen?

- A regression problem is when the output variable is a real or continuous value, such as "salary" or "weight".
- Many different models can be used, the simplest is linear regression. It tries to fit data with the best hyperplane which goes through the points.
- Mathematically a linear relationship represents a straight line when plotted as a graph.
- A non-linear relationship where the exponent of any variable is not equal to 1 creates a curve.

Types of Regression Analysis Techniques:

1. Linear Regression
2. Logistic Regression
3. Ridge Regression
4. Lasso Regression
5. Polynomial Regression
6. Bayesian Linear Regression

Advantages & Limitations:

- Fast and easy to model and is particularly useful when the relationship to be modeled is not extremely complex and if you don't have a lot of data.
- Very intuitive to understand and interpret.
- Linear Regression is very sensitive to outliers.

Linear regression:

- Linear Regression is a very simple method but has proven to be very useful for a large number of situations.
- When we have a single input attribute (x) and we want to use linear regression, this is called simple linear regression.

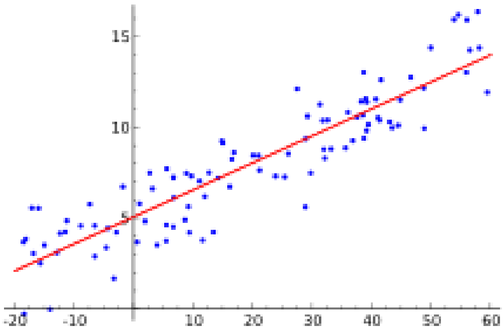


Figure Linear Regression plot

- simple linear regression we want to model our data as follows:

  y = B0 + B1 * x

- we know and B0 and B1 are coefficients that we need to estimate that move the line around.
- Simple regression is great, because rather than having to search for values by trial and error or calculate them analytically using more advanced linear algebra, we can estimate them directly from our data.

OLS Regression:

Linear Regression using Ordinary Least Squares Approximation

Based on Gauss Markov Theorem:

We can start off by estimating the value for B1 as:

$$B1 = \frac{\sum_{i=1}^{n} \left(x_i - mean(x)\right) * \left(y_i - mean(y)\right)}{\sum_{i=1}^{n} \left(x_i - mean(x)\right)^2}$$

$$B0 = mean(y) - B1 * mean(x)$$

- If we had multiple input attributes (e.g. x1, x2, x3, etc.) This would be called multiple linear regression. The procedure for linear regression is different and simpler than that for multiple linear regression.

Let us consider the following Example:

for an equation y=2*x+3.

x	y	xi-mean(x)	yi-mean(y)	xi-mean(x) * yi-mean(y)	(xi-mean(xi)^2
-3	-3	-4.4	-8.8	38.72	19.36
-1	1	-2.4	-4.8	11.52	5.76
2	7	0.6	1.2	0.72	0.36
4	11	2.6	5.2	13.52	6.76
5	13	3.6	7.2	25.92	12.96
1.4	5.8			Sum = 90.4	Sum = 45.2

Mean(x) = 1.4 and Mean(y) = 5.8

$$B1 = \frac{\sum_{i=1}^{n}(x_i - mean(x)) * (y_i - mean(y))}{\sum_{i=1}^{n}(x_i - mean(x))^2}$$

$$B0 = mean(y) - B1 * mean(x)$$

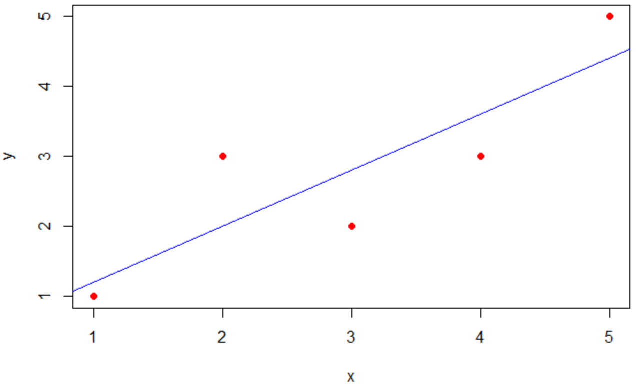
We can find from the above formulas,

B1=2 and B0=3

Example for Linear Regression using R:

Consider the following data set:
x = {1,2,4,3,5} and y = {1,3,3,2,5}
We use R to apply Linear Regression for the above data.

```
> rm(list=ls()) #removes the list of variables in the current session of R
> x<-c(1,2,4,3,5)      #assigns values to x
> y<-c(1,3,3,2,5)      #assigns values to y
> x;y
[1] 1 2 4 3 5
 [1] 1 3 3 2 5
> graphics.off() #to clear the existing plot/s
> plot(x,y,pch=16, col="red")
> relxy<-lm(y~x)
> relxy
        Call:
        lm(formula = y ~ x)
        Coefficients:
        (Intercept)
        x
        0.4         0.8
> abline(relxy,col="Blue")
```



```
> a <- data.frame(x = 7)
> a
 7
> result <-  predict(relxy,a)
> print(result)
6
> #Note: you can observe that
> 0.8*7+0.4
[1] 6    #The same calculated using the line equation y= 0.8*x +0.4.
```

Simple linear regression is the simplest form of regression and the most studied.

Calculating B1 & B0 using Correlations and Standard Deviations:

B1 = corr(x, y) * stdev(y) / stdev(x)

$$B1 = \frac{Correlation(x, y) * St.Deviation(y)}{St.Deviation(x)}$$

Where cor (x,y) is the correlation between x & y and stdev() is the calculation of the standard deviation for a variable. The same is calculated in R as follows:

```
> x<-c(1,2,4,3,5)
> y<-c(1,3,3,2,5)
> x;y
[1] 1 2 4 3 5
[1] 1 3 3 2 5
> B1=cor(x,y)*sd(y)/sd(x)
> B1
[1] 0.8
> B0=mean(y)-B1*mean(x)
> B0
[1] 0.4
```

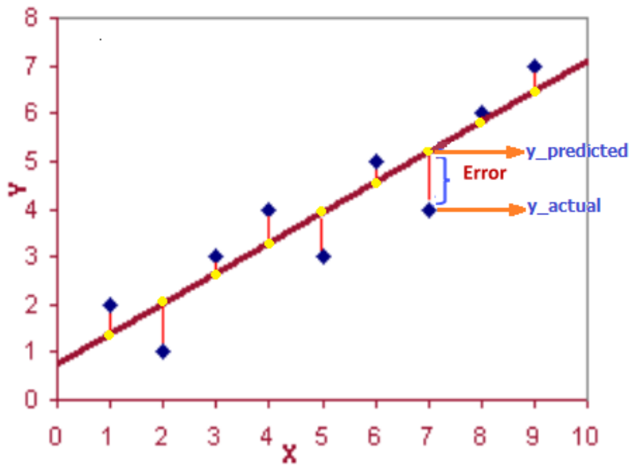
Estimating Error: (RMSE: Root Mean Squared Error)

We can calculate the error for our predictions called the Root Mean Squared Error or RMSE.

Root Mean Square Error can be calculated by

$$Err = \sqrt{\frac{\sum_{i=1}^{n}(p_i - y_i)^2}{n}}$$

p is the predicted value and y is the actual value, i is the index for a specific instance, n is the number of predictions, because we must calculate the error across all predicted values.



Estimating Error for y=0.8*x+0.4

x	y = y-actual	p = y-predicted	p-y	(p-y)^2
1	1	1.2	0.2	0.04
2	3	2	-1	1
4	3	3.6	0.6	0.36
3	2	2.8	0.8	0.64
5	5	4.4	-0.6	0.36

⇨ mean(x)= 3

s = sum of $(p\text{-}y)^2$ = 2.4

⇨ s/n = 2.4 / 5 = 0.48

⇨ sqrt(s/n) = sqrt(0.48) = 0.692

⇨ **RMSE = 0.692**

Properties and Assumptions of OLS approximation:

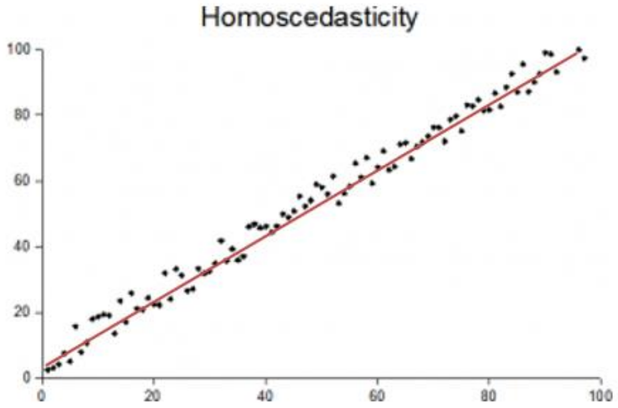
1. **Unbiasedness:**
   - i. Biased estimator is defined as the difference between its expected value and the true value. i.e., e(y)=y_actual – y_predited
   - ii. If the biased error (bias) is zero then estimator become unbiased.
   - iii. Unbiasedness is important only when it is combined with small variance
2. **Least Variance:**
   - i. An estimator is best when it has the smallest or least variance
   - ii. Least variance property is more important when it combined with small biased.
3. **Efficient estimator:**
   - i. An estimator said to be efficient when it fulfilled both conditions.
   - **ii.** Estimator should unbiased and have least variance
4. **Best Linear Unbiased Estimator (BLUE Properties):**
   - i. An estimator is said to be BLUE when it fulfill the above properties
   - ii. An estimator is BLUE if it is Unbiased, Least Variance and Linear Estimator
5. **Minimum Mean Square Error (MSE):**
   - i. An estimator is said to be MSE estimator if it has smallest mean square error.
   - ii. Less difference between estimated value and True Value
6. **Sufficient Estimator:**
   - i. An estimator is sufficient if it utilizes all the information of a sample about the True parameter.
   - ii. It must use all the observations of the sample.

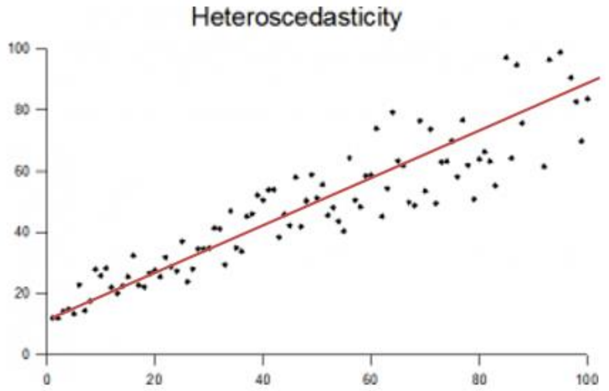
Assumptions of OLS Regression:

1. There are random sampling of observations.
2. The conditional mean should be zero
3. There is homoscedasticity and no Auto-correlation.
4. Error terms should be normally distributed(optional)
5. The Properties of OLS estimates of simple linear regression equation is

   y = B0+B1*x + **µ** (µ -> Error)
6. The above equation is based on the following assumptions
   - a. Randomness of **µ**
   - b. Mean of **µ** is Zero
   - c. Variance of **µ** is constant
   - d. The variance of **µ** has normal distribution
   - e. **Error µ** of different observations are independent.

Homoscedasticity vs Heteroscedasticity:





- The Assumption of homoscedasticity (meaning "same variance") is central to linear regression models. Homoscedasticity describes a situation in which the error term (that is, the "noise" or random disturbance in the relationship between the independent variables and the dependent variable) is the same across all values of the independent variables.
- Heteroscedasticity (the violation of homoscedasticity) is present when the size of the error term differs across values of an independent variable.
- The impact of violating the assumption of homoscedasticity is a matter of degree, increasing as heteroscedasticity increases.
- Homoscedasticity means "having the same scatter." For it to exist in a set of data, the points must be about the same distance from the line, as shown in the picture above.
- The opposite is heteroscedasticity ("different scatter"), where points are at widely varying distances from the regression line.

Variable Rationalization:

- The data set may have a large number of attributes. But some of those attributes can be irrelevant or redundant. The goal of Variable Rationalization is to improve the Data Processing in an optimal way through attribute subset selection.
- This process is to find a minimum set of attributes such that dropping of those irrelevant attributes does not much affect the utility of data and the cost of data analysis could be reduced.
- Mining on a reduced data set also makes the discovered pattern easier to understand. As part of Data processing, we use the below methods of Attribute subset selection
  1. Stepwise Forward Selection
  2. Stepwise Backward Elimination
  3. Combination of Forward Selection and Backward Elimination
  4. Decision Tree Induction.

All the above methods are greedy approaches for attribute subset selection.

1. **Stepwise Forward Selection**: This procedure starts with an empty set of attributes as the minimal set. The most relevant attributes are chosen (having minimum p-value) and are added to the minimal set. In each iteration, one attribute is added to a reduced set.

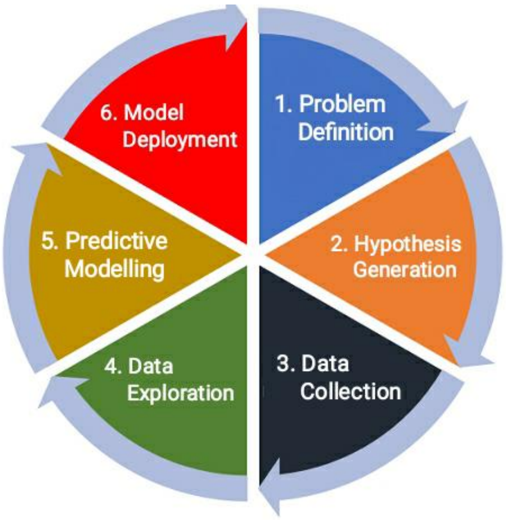
- **Stepwise Backward Elimination**: Here all the attributes are considered in the initial set of attributes. In each iteration, one attribute is eliminated from the set of attributes whose p-value is higher than significance level.
- **Combination of Forward Selection and Backward Elimination:** The stepwise forward selection and backward elimination are combined so as to select the relevant attributes most efficiently. This is the most common technique which is generally used for attribute selection.
- **Decision Tree Induction**: This approach uses decision tree for attribute selection. It constructs a flow chart like structure having nodes denoting a test on an attribute. Each branch corresponds to the outcome of test and leaf nodes is a class prediction. The attribute that is not the part of tree is considered irrelevant and hence discarded.

Model Building Life Cycle in Data Analytics:

When we come across a business analytical problem, without acknowledging the stumbling blocks, we proceed towards the execution. Before realizing the misfortunes, we try to implement and predict the outcomes. The problem-solving steps involved in the data science model-building life cycle. Let's understand every model building step in-depth,

The data science model-building life cycle includes some important steps to follow. The following are the steps to follow to build a Data Model

1. **Problem Definition**
2. **Hypothesis Generation**
3. **Data Collection**
4. **Data Exploration/Transformation**
5. **Predictive Modelling**
6. **Model Deployment**



1. Problem Definition

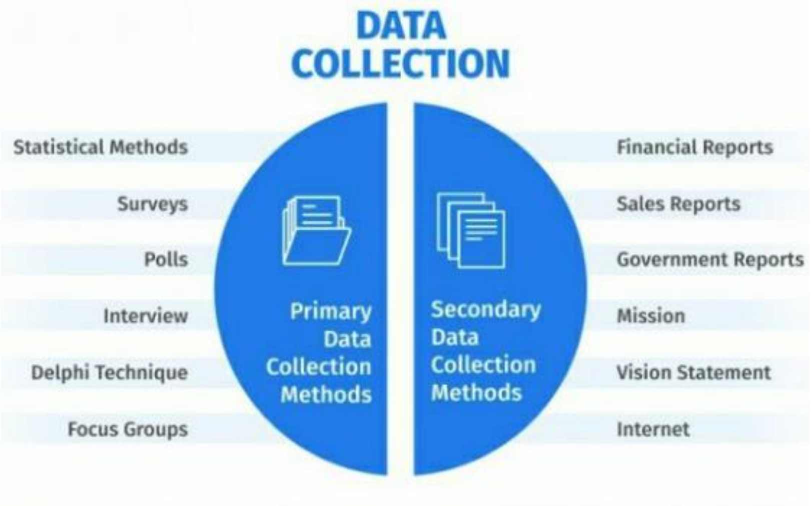
- The first step in constructing a model is to understand the industrial problem in a more comprehensive way. To identify the purpose of the problem and the prediction target, we must define the project objectives appropriately.
- Therefore, to proceed with an analytical approach, we have to recognize the obstacles first. Remember, excellent results always depend on a better understanding of the problem.

2. Hypothesis Generation

- Hypothesis generation is the guessing approach through which we derive some essential data parameters that have a significant correlation with the prediction target.
- Your hypothesis research must be in-depth, looking for every perceptive of all stakeholders into account. We search for every suitable factor that can influence the outcome.
- Hypothesis generation focuses on what you can create rather than what is available in the dataset.

3. Data Collection

- Data collection is gathering data from relevant sources regarding the analytical problem, then we extract meaningful insights from the data for prediction.



The data gathered must have:

- Proficiency in answer hypothesis questions.
- Capacity to elaborate on every data parameter.
- Effectiveness to justify your research.
- Competency to predict outcomes accurately.

4. Data Exploration/Transformation

- The data you collected may be in unfamiliar shapes and sizes. It may contain unnecessary features, null values, unanticipated small values, or immense values. So, before applying any algorithmic model to data, we have to explore it first.
- By inspecting the data, we get to understand the explicit and hidden trends in data. We find the relation between data features and the target variable.
- Usually, a data scientist invests his 60–70% of project time dealing with data exploration only.
- There are several sub steps involved in data exploration:
  - **Feature Identification:**
    - You need to analyze which data features are available and which ones are not.
    - Identify independent and target variables.
    - Identify data types and categories of these variables.

- **Univariate Analysis:**
    - We inspect each variable one by one. This kind of analysis depends on the variable type whether it is categorical and continuous.
        - Continuous variable: We mainly look for statistical trends like mean, median, standard deviation, skewness, and many more in the dataset.
        - Categorical variable: We use a frequency table to understand the spread of data for each category. We can measure the counts and frequency of occurrence of values.
- **Multi-variate Analysis:**
    - The bi-variate analysis helps to discover the relation between two or more variables.
    - We can find the correlation in case of continuous variables and the case of categorical, we look for association and dissociation between them.
- **Filling Null Values:**
    - Usually, the dataset contains null values which lead to lower the potential of the model. With a continuous variable, we fill these null values using the mean or mode of that specific column. For the null values present in the categorical column, we replace them with the most frequently occurred categorical value. Remember, don't delete that rows because you may lose the information.

5. **Predictive Modeling**

- Predictive modeling is a mathematical approach to create a statistical model to forecast future behavior based on input test data.

Steps involved in predictive modeling:

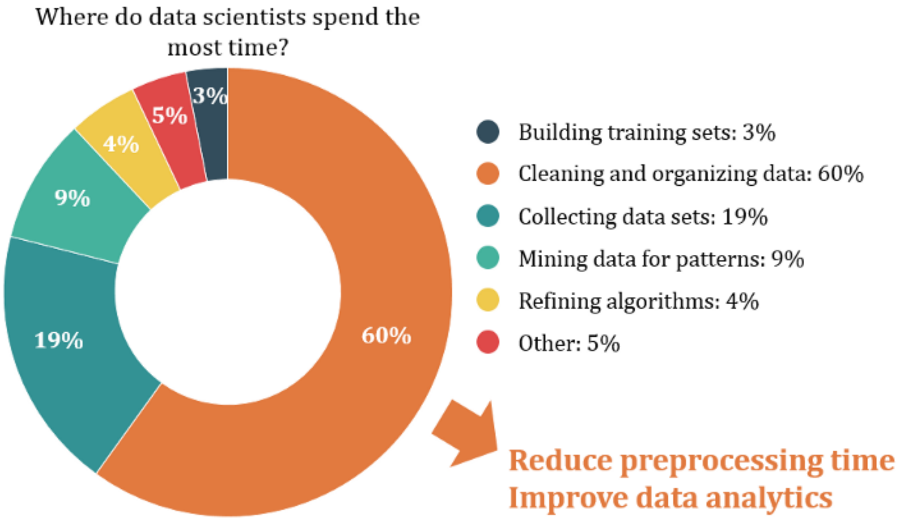
- **Algorithm Selection:**
    - When we have the structured dataset, and we want to estimate the continuous or categorical outcome then we use supervised machine learning methodologies like regression and classification techniques. When we have unstructured data and want to predict the clusters of items to which a particular input test sample belongs, we use unsupervised algorithms. An actual data scientist applies multiple algorithms to get a more accurate model.
- **Train Model:**
    - After assigning the algorithm and getting the data handy, we train our model using the input data applying the preferred algorithm. It is an action to determine the correspondence between independent variables, and the prediction targets.
- **Model Prediction:**

- We make predictions by giving the input test data to the trained model. We measure the accuracy by using a cross-validation strategy or ROC curve which performs well to derive model output for test data.

6. Model Deployment

- There is nothing better than deploying the model in a real-time environment. It helps us to gain analytical insights into the decision-making procedure. You constantly need to update the model with additional features for customer satisfaction.
- To predict business decisions, plan market strategies, and create personalized customer interests, we integrate the machine learning model into the existing production domain.
- When you go through the Amazon website and notice the product recommendations completely based on your curiosities. You can experience the increase in the involvement of the customers utilizing these services. That's how a deployed model changes the mindset of the customer and convince him to purchase the product.

Key Takeaways



SUMMARY OF DA MODEL LIFE CYCLE:

- Understand the purpose of the business analytical problem.
- Generate hypotheses before looking at data.
- Collect reliable data from well-known resources.
- Invest most of the time in data exploration to extract meaningful insights from the data.
- Choose the signature algorithm to train the model and use test data to evaluate.
- Deploy the model into the production environment so it will be available to users and strategize to make business decisions effectively.

Logistic Regression:

Model Theory, Model fit Statistics, Model Construction

Introduction:

- Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
- The outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
- The curve from the logistic function indicates the likelihood of something such as whether or not the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- Logistic regression uses the concept of predictive modeling as regression; therefore, it is called logistic regression, but is used to classify samples; therefore, it falls under the classification algorithm.
- In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

Types of Logistic Regressions:

On the basis of the categories, Logistic Regression can be classified into three types:

- **Binomial:** In binomial Logistic regression, there can be only two possible types of the dependent variables, such as 0 or 1, Pass or Fail, etc.
- **Multinomial:** In multinomial Logistic regression, there can be 3 or more possible unordered types of the dependent variable, such as "cat", "dogs", or "sheep"
- **Ordinal:** In ordinal Logistic regression, there can be 3 or more possible ordered types of dependent variables, such as "low", "Medium", or "High".

Definition: Multi-collinearity:

- Multicollinearity is a statistical phenomenon in which multiple independent variables show high correlation between each other and they are too inter-related.
- Multicollinearity also called as Collinearity and it is an undesired situation for any statistical regression model since it diminishes the reliability of the model itself.

- If two or more independent variables are too correlated, the data obtained from the regression will be disturbed because the independent variables are actually dependent between each other.

Assumptions for Logistic Regression:

- The dependent variable must be categorical in nature.
- The independent variable should not have multi-collinearity.

Logistic Regression Equation:

- The Logistic regression equation can be obtained from the Linear Regression equation. The mathematical steps to get Logistic Regression equations are given below:
- Logistic Regression uses a more complex cost function, this cost function can be defined as the 'Sigmoid function' or also known as the 'logistic function' instead of a linear function.
- The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

$$0 \le h_\theta(x) \le 1$$ --- *Logistic Regression Hypothesis Expectation*

Logistic Function (Sigmoid Function):

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- The sigmoid function maps any real value into another value within a range of 0 and 1, and so forma S-Form curve.
- The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form.
- The below image is showing the logistic function:

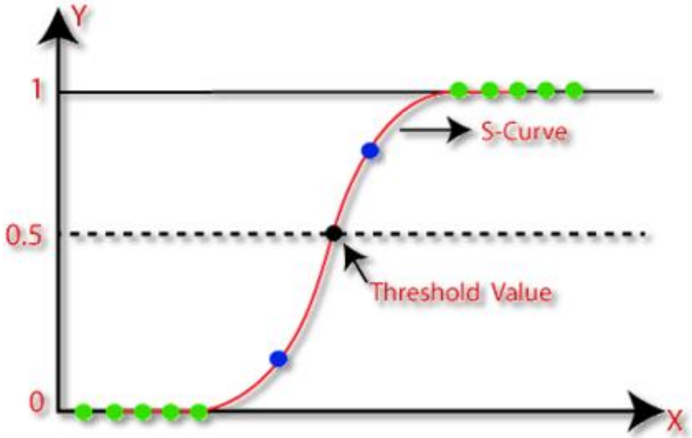


Fig: Sigmoid Function Graph

The Sigmoid function can be interpreted as a probability indicating to a Class-1 or Class-0. So the Regression model makes the following predictions as

$$z = sigmoid(y) = \sigma(y) = \frac{1}{1+e^{-y}}$$

Hypothesis Representation

- When using *linear regression,* we used a formula for the line equation as:

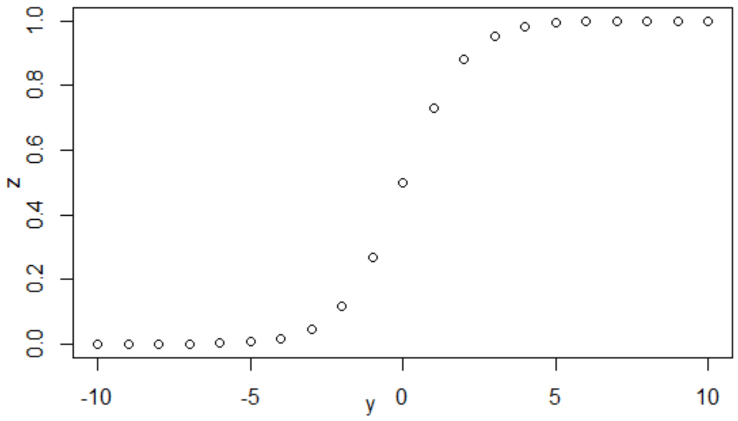
$$y = b_0 + b_1x_1 + b_2x_2 + ... + b_nx_n$$

- In the above equation y is a response variable, $x_1, x_2, ...x_n$ are the predictor variables, and $b_0, b_1, b_2, ..., b_n$ are the coefficients, which are numeric constants.
- For logistic regression, we need the maximum likelihood hypothesis $h_\theta(y)$
- Apply Sigmoid function on y as

$$z = \sigma(y) = \sigma(b_0 + b_1x_1 + b_2x_2 + ... + b_nx_n)$$

$$z = \sigma(y) = \frac{1}{1+e^{-(b_0+b_1x_1+b_2x_2+...+b_nx_n)}}$$

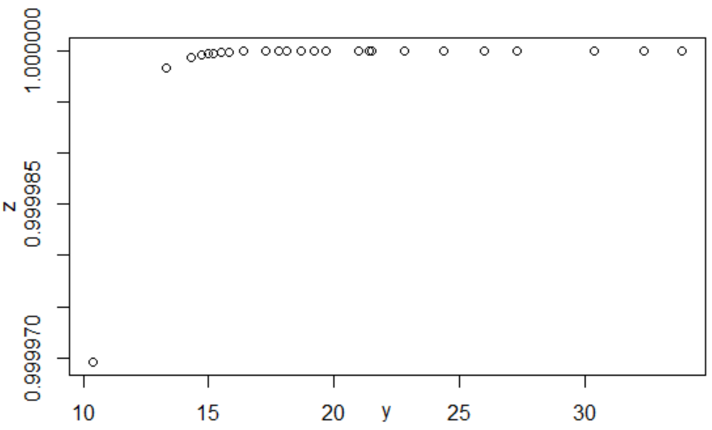
```
Example for Sigmoid Function in R:
> #Example for Sigmoid Function
> y<-c(-10:10);y
 [1] -10  -9  -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5   6   7   8   9  10
> z<-1/(1+exp(-y));z
 [1]  4.539787e-05  1.233946e-04  3.353501e-04  9.110512e-04  2.472623e-03  6.692851e-03
1.798621e-02 4.742587e-02
 [9]    1.192029e-01    2.689414e-01
5.000000e-01            7.310586e-01
8.807971e-01            9.525741e-01
9.820138e-01 9.933071e-01
[17]    9.975274e-01    9.990889e-01
9.996646e-01            9.998766e-01
9.999546e-01
> plot(y,z)
```



```
> rm(list=ls())
> attach(mtcars)    #attaching a
data set into the R environment
> input <- mtcars[,c("mpg","disp","hp","wt")]
> head(input)
                   mpg  disp  hp    wt
Mazda RX4         21.0   160 110 2.620
Mazda RX4 Wag     21.0   160 110 2.875
Datsun 710        22.8   108  93 2.320
Hornet 4 Drive    21.4   258 110 3.215
Hornet Sportabout 18.7  360 175 3.440
Valiant           18.1  225 105 3.460
> #model<-lm(mpg~disp+hp+wt);model1# Show the model
> model<-glm(mpg~disp+hp+wt);model

Call:  glm(formula = mpg ~ disp + hp + wt)

Coefficients:
(Intercept)         disp           hp           wt
  37.105505    -0.000937    -0.031157    -3.800891

Degrees of Freedom: 31 Total (i.e. Null);  28 Residual
Null Deviance:        1126
Residual Deviance: 195  AIC: 158.6
> newx<-data.frame(disp=150,hp=150,wt=4)  #new input for prediction
> predict(model,newx)
       1
17.08791
> 37.15+(-0.000937)*150+(-0.0311)*150+(-3.8008)*4 #checking with the data newx
[1] 17.14125
```

```
y<-input[,c("mpg")]; y
z=1/(1+exp(-y));z
plot(y,z)
```

```
> y<-input[,c("mpg")]
> y
 [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2 10.4
10.4 14.7 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4 15.8 19.7 15.0
21.4
> z=1/(1+exp(-y));z
1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.9999994 1.0000000
1.0000000   1.0000000   1.0000000
0.9999999   1.0000000   0.9999997
0.9999696   0.9999696   0.9999996
1.0000000   1.0000000   1.0000000
1.0000000   0.9999998   0.9999997
0.9999983   1.0000000   1.0000000
1.0000000   1.0000000   0.9999999
1.0000000 0.9999997 1.0000000
> plot(y,z)
```

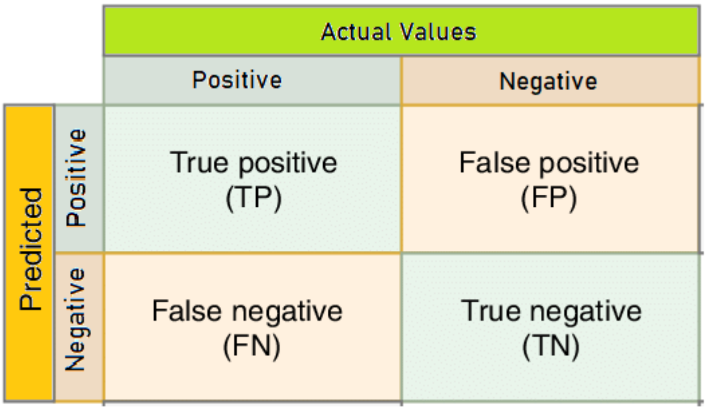


Confusion Matrix (or) Error Matrix (or) Contingency Table:

What is a Confusion Matrix?

"A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making. It is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one (in unsupervised learning it is usually called a matching matrix)."

For a binary classification problem, we would have a 2 x 2 matrix as shown below with 4 values:



		Actual Values	
		Positive	Negative
Predicted	Positive	True positive (TP)	False positive (FP)
	Negative	False negative (FN)	True negative (TN)

Fig: Confusion Matrix

Let's decipher the matrix:

- The target variable has two values: Positive or Negative
- The columns represent the actual values of the target variable
- The rows represent the predicted values of the target variable

  - True Positive
  - True Negative
  - False Positive – Type 1 Error
  - False Negative – Type 2 Error

Why we need a Confusion matrix?

  - Precision vs Recall
  - F1-score

Understanding True Positive, True Negative, False Positive and False Negative in a Confusion Matrix

True Positive (TP)

- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

- The predicted value matches the actual value
- The actual value was negative and the model predicted a negative value

False Positive (FP) – Type 1 error

- The predicted value was falsely predicted
- The actual value was negative but the model predicted a positive value
- Also known as the Type 1 error

False Negative (FN) – Type 2 error

- The predicted value was falsely predicted
- The actual value was positive but the model predicted a negative value
- Also known as the Type 2 error

To evaluate the performance of a model, we have the performance metrics called,

Accuracy, Precision, Recall & F1-Score metrics

Accuracy:

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

- Accuracy is a great measure to understand that the model is Best.
- Accuracy is dependable only when you have symmetric datasets where values of false positive and false negatives are almost same.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision:

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

It tells us how many of the correctly predicted cases actually turned out to be positive.

$$Precision = \frac{TP}{TP + FP}$$

- Precision is a useful metric in cases where False Positive is a higher concern than False Negatives.

- Precision is important in music or video recommendation systems, e-commerce websites, etc. Wrong results could lead to customer churn and be harmful to the business.

Recall: (Sensitivity)

Recall is the ratio of correctly predicted positive observations to the all observations in actual class.

$$Recall = \frac{TP}{TP + FN}$$

- Recall is a useful metric in cases where False Negative trumps False Positive.
- Recall is important in medical cases where it doesn't matter whether we raise a false alarm but the actual positive cases should not go undetected!

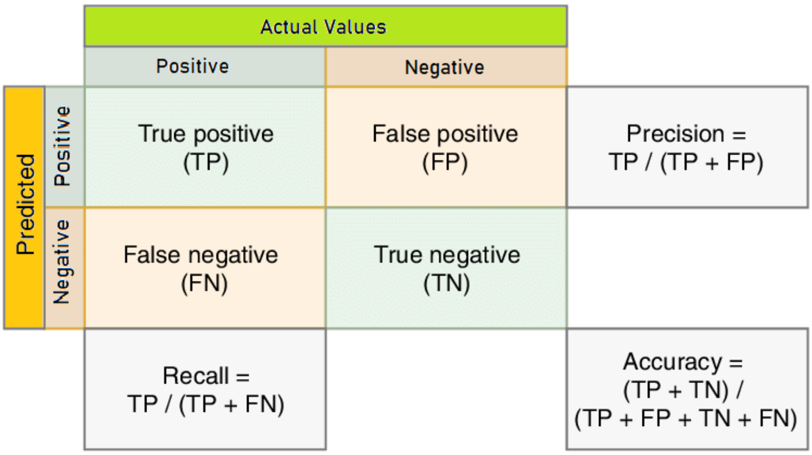
F1-Score:

F1-score is a harmonic mean of Precision and Recall. It gives a combined idea about these two metrics. It is maximum when Precision is equal to Recall.

Therefore, this score takes both false positives and false negatives into account.

$$F_1 - Score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precession}} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- F1 is usually more useful than accuracy, especially if you have an uneven class distribution.
- Accuracy works best if false positives and false negatives have similar cost.
- If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall.
- But there is a catch here. If the interpretability of the F1-score is poor, means that we don't know what our classifier is maximizing – precision or recall? So, we use it in combination with other evaluation metrics which gives us a complete picture of the result.

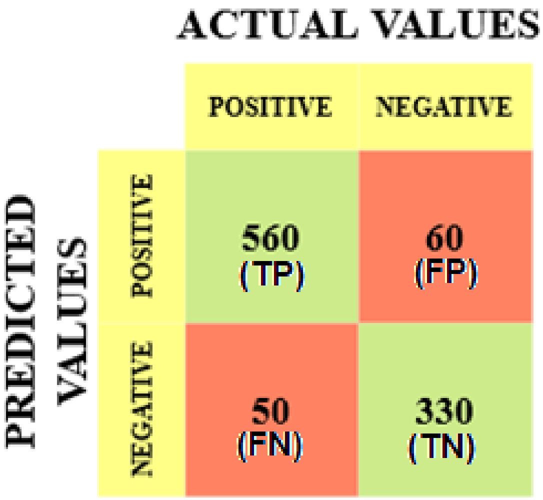


		Actual Values		
		Positive	Negative	
Predicted	Positive	True positive (TP)	False positive (FP)	Precision = TP / (TP + FP)
	Negative	False negative (FN)	True negative (TN)	
		Recall = TP / (TP + FN)		Accuracy = (TP + TN) / (TP + FP + TN + FN)

Fig: Confusion Matrix

Example:

Suppose we had a classification dataset with 1000 data points. We fit a classifier on it and get the below confusion matrix:



		ACTUAL VALUES	
		POSITIVE	NEGATIVE
PREDICTED VALUES	POSITIVE	560 (TP)	60 (FP)
	NEGATIVE	50 (FN)	330 (TN)

The different values of the Confusion matrix would be as follows:

- True Positive (TP) = 560

  -Means 560 positive class data points were correctly classified by the model.

- True Negative (TN) = 330

  -Means 330 negative class data points were correctly classified by the model.

- False Positive (FP) = 60

  -Means 60 negative class data points were incorrectly classified as belonging to the positive class by the model.

- False Negative (FN) = 50

  -Means 50 positive class data points were incorrectly classified as belonging to the negative class by the model.

This turned out to be a pretty decent classifier for our dataset considering the relatively larger number of true positive and true negative values.

Precisely we have the outcomes represented in Confusion Matrix as:

TP = 560, TN = 330, FP = 60, FN = 50

Accuracy:

The accuracy for our model turns out to be:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$=> Accuracy = \frac{560 + 330}{560 + 60 + 330 + 50} = \frac{890}{1000} = 0.89$$

Hence Accuracy is 89%...Not bad!

Precision:

It tells us how many of the correctly predicted cases actually turned out to be positive.

$$Precision = \frac{TP}{TP + FP}$$

This would determine whether our model is reliable or not.

Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

$$Precision = \frac{TP}{TP + FP} = \frac{560}{560 + 60} = 0.903$$

We can easily calculate Precision and Recall for our model by plugging in the values into the above questions:

$$Recall = \frac{TP}{TP + FN} = \frac{560}{560 + 50} = 0.918$$

F1-Score

$$F_1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

$$\Rightarrow F_1 - Score = 2 * \frac{0.903 * 0.918}{0.903 + 0.918} = \frac{0.8289}{1.821} = 0.4552$$

AUC (Area Under Curve) ROC (Receiver Operating Characteristics) Curves:

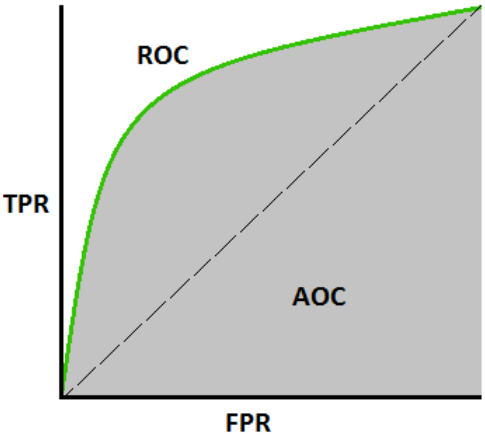
Performance measurement is an essential task in Data Modelling Evaluation. It is one of the most important evaluation metrics for checking any classification model’s performance. It is also written as AUROC (Area Under the Receiver Operating Characteristics) So when it comes to a classification problem, we can count on an AUC - ROC Curve.

When we need to check or visualize the performance of the multi-class classification problem, we use the AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve.

What is the AUC - ROC Curve?

AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. By analogy, the Higher the AUC, the better the model is at distinguishing between patients with the disease and no disease.

The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.



TPR (True Positive Rate) / Recall /Sensitivity

$$\text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

Specificity

$$\text{Specificity} = \frac{TN}{TN + FP}$$

FPR (False Positive Rate)

$$\text{FPR} = 1 - \text{Specificity}$$

$$= \frac{FP}{TN + FP}$$

ROC curve

An **ROC curve** (**receiver operating characteristic curve**) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

- True Positive Rate
- False Positive Rate
- **True Positive Rate** (**TPR**) is a synonym for recall and is therefore defined as follows:
    - TPR=TPTP+FN

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate (**FPR**) is defined as follows:

$$FPR = \frac{FP}{FP + TN}$$

An ROC curve plots TPR vs. FPR at different classification thresholds. Lowering the classification threshold classifies more items as positive, thus increasing both False Positives and True Positives. The following figure shows a typical ROC curve.

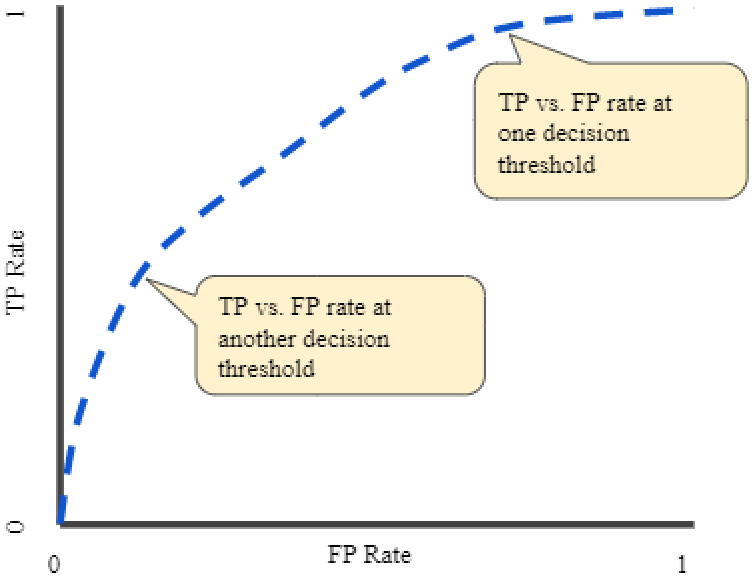


Figure 4. TP vs. FP rate at different classification thresholds.

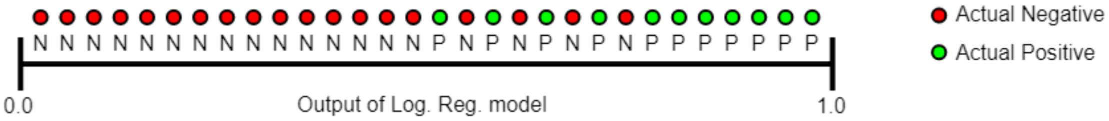

N N N N N N N N N N N N N N N P N P N P N P N P P P P P P P
0.0
Output of Log. Reg. model
1.0
● Actual Negative
● Actual Positive

Figure 6. Predictions ranked in ascending order of logistic regression score.

Analytics applications to various Business Domains:

Application of Modelling in Business:

- Applications of Data Modelling can be termed as Business analytics.
- Business analytics involves the collating, sorting, processing, and studying of business-related data using statistical models and iterative methodologies. The goal of BA is to narrow down which datasets are useful and which can increase revenue, productivity, and efficiency.
- Business analytics (BA) is the combination of skills, technologies, and practices used to examine an organization's data and performance as a way to gain insights and make data-driven decisions in the future using statistical analysis.

Although business analytics is being leveraged in most commercial sectors and industries, the following applications are the most common.

1. Credit Card Companies

   Credit and debit cards are an everyday part of consumer spending, and they are an ideal way of gathering information about a purchaser's spending habits, financial situation, behaviour trends, demographics, and lifestyle preferences.

2. Customer Relationship Management (CRM)

   Excellent customer relations is critical for any company that wants to retain customer loyalty to stay in business for the long haul. CRM systems analyze important performance indicators such as demographics, buying patterns, socio-economic information, and lifestyle.

3. Finance

   The financial world is a volatile place, and business analytics helps to extract insights that help organizations maneuver their way through tricky terrain. Corporations turn to business analysts to optimize budgeting, banking, financial planning, forecasting, and portfolio management.

4. Human Resources

   Business analysts help the process by pouring through data that characterizes high performing candidates, such as educational background, attrition rate, the average length of employment, etc. By working with this information, business analysts help HR by forecasting the best fits between the company and candidates.

5. Manufacturing

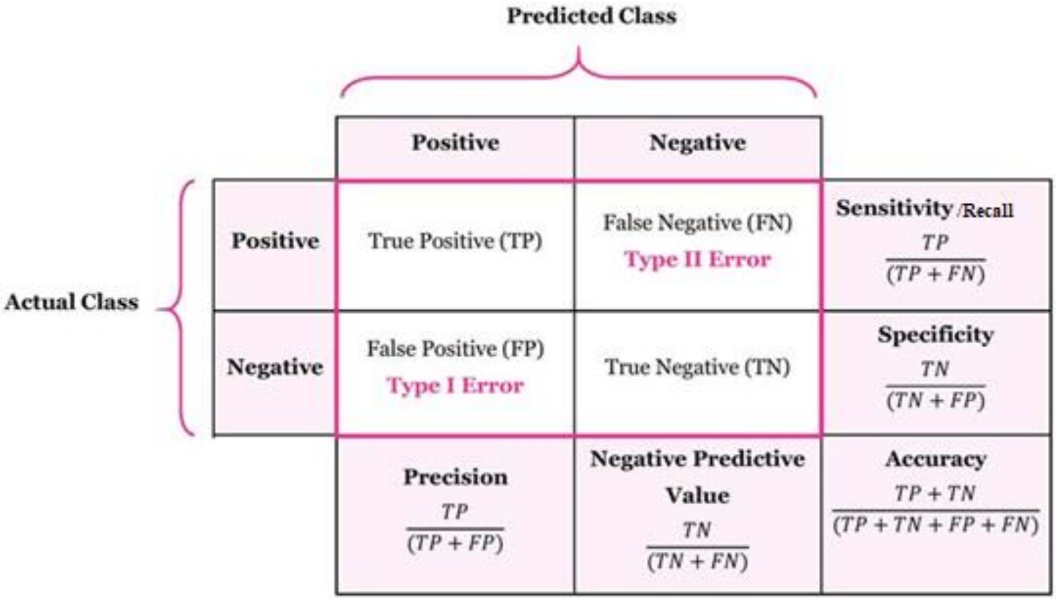
Business analysts work with data to help stakeholders understand the things that affect operations and the bottom line. Identifying things like equipment downtime, inventory levels, and maintenance costs help companies streamline inventory management, risks, and supply-chain management to create maximum efficiency.

6. Marketing

   Business analysts help answer these questions and so many more, by measuring marketing and advertising metrics, identifying consumer behaviour and the target audience, and analyzing market trends.

***** End of Unit-3 *****

Add-ons for Unit-3



		Predicted Class		
		Positive	Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity /Recall $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	Negative Predictive Value $\frac{TN}{(TN+FN)}$	Accuracy $\frac{TP+TN}{(TP+TN+FP+FN)}$

TOBE DISCUSSED:

Receiver Operating Characteristics:

ROC & AUC

Derivation for Logistic Regression:

The logistic regression model assumes that the log-odds of an observation *y* can be expressed as a linear function of the K input variables x:

$$\log \frac{P(\mathbf{x})}{1-P(\mathbf{x})} = \sum_{j=0}^{K} b_j x_j$$

Here, we add the constant term b_0, by setting $x_0 = 1$. This gives us K+1 parameters. The left hand side of the above equation is called the *logit* of P (hence, the name logistic regression).

Let's take the exponent of both sides of the logit equation.

$$\frac{P(\mathbf{x})}{1-P(\mathbf{x})} = \exp(\sum_{j=0}^{K} b_j x_j)$$

$$= \prod_{j=0}^{K} \exp(b_j x_j)$$

(Since ln(ab)=ln(a)+ln(b) and exp(a+b)=exp(a)exp(b).)

We can also invert the logit equation to get a new expression for P(x):

$$P(\mathbf{x}) = \frac{\exp z}{1 + \exp z},$$
$$z = \sum_{j=0}^{K} b_j x_j$$

The right hand side of the top equation is the sigmoid of *z*, which maps the real line to the interval (0, 1), and is approximately linear near the origin. A useful fact about P(*z*) is that the derivative P'(*z*) = P(*z*) (1 – P(*z*)). Here's the derivation:

$$\begin{aligned}
P(z) &= \frac{\exp z}{1 + \exp z} = (\exp z)(1 + \exp z)^{-1} \\
P'(z) &= (\exp z)(1 + \exp z)^{-1} + (\exp z)(-1)(1 + \exp z)^{-2}(\exp z) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(by chain rule)} \\
&= \frac{(\exp z)(1 + \exp z)}{(1 + \exp z)^2} - \frac{(\exp z)^2}{(1 + \exp z)^2} \\
&= \frac{\exp z}{(1 + \exp z)^2} \\
&= \frac{\exp z}{1 + \exp z} \cdot \frac{1}{1 + \exp z} \\
&= P(z)(1 - P(z))
\end{aligned}$$

Later, we will want to take the gradient of P with respect to the set of coefficients b, rather than *z*. In that case, P'(*z*) = P(*z*) (1 – P(*z*))*z*', where ‘ is the gradient taken with respect to b.