Unit I

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

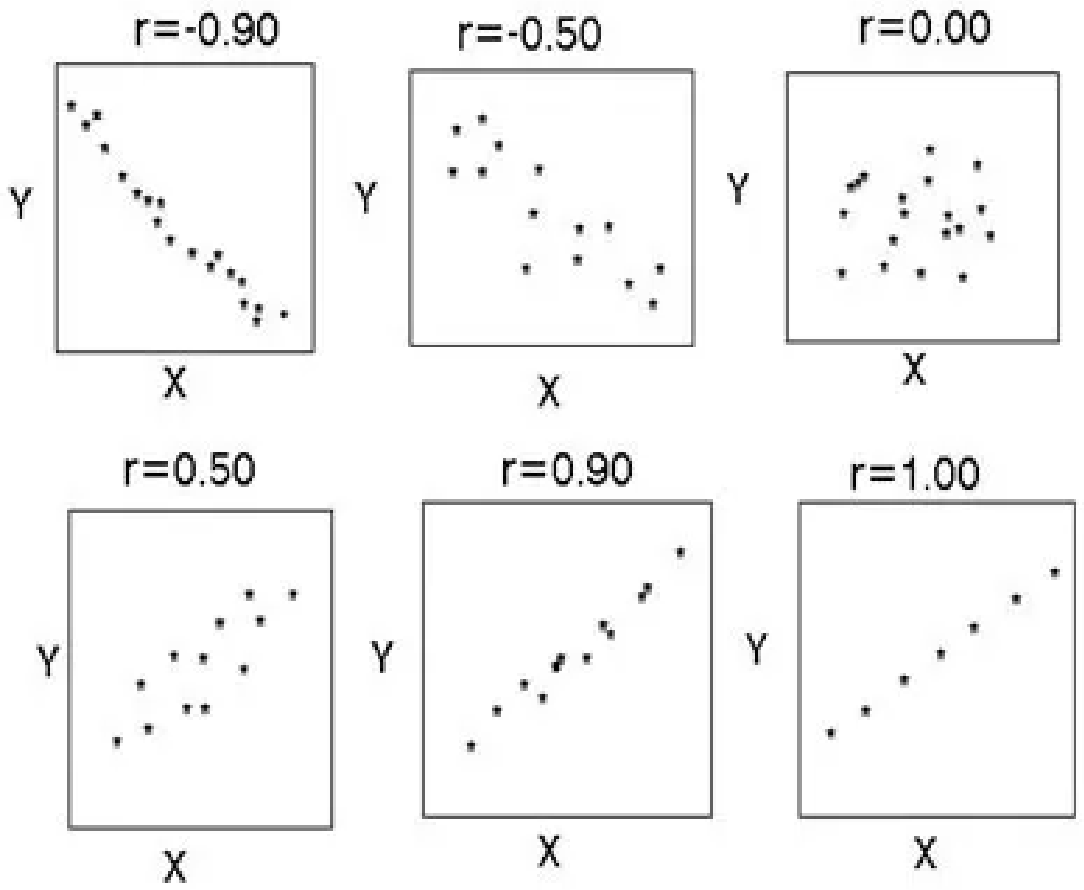
Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

Correlation:

- Correlation means association - more precisely it is a measure of the extent to which two variables are related.
- There are three possible results of a correlational study:
  - ✓ positive correlation
  - ✓ negative correlation and
  - ✓ no correlation.
- A **positive correlation** is a relationship between two variables in which both variables move in the same direction. Therefore, when one variable increases as the other variable increases, or one variable decreases while the other decreases.
- A **negative correlation** is a relationship between two variables in which an increase in one variable is associated with a decrease in the other.
- A **zero correlation** exists when there is no relationship between two variables.

$$Corr(x,y) = \frac{\sum_{i=1}^{n}(x_i - x')\,(y_i - y')}{\sqrt{\sum_{i=1}^{n}(x_i - x')^2 \sum_{i=1}^{n}(y_i - y')^2}}$$

- ✓ The correlation coefficient (*r*) indicates the extent to which the pairs of numbers for these two variables lie on a straight line.
- ✓ Values over zero indicate a **positive correlation,** while values under zero indicate a **negative correlation.**
- ✓ A correlation of –1 indicates a perfect negative correlation, meaning that as one variable goes up, the other goes down.
- ✓ A correlation of +1 indicates a perfect positive correlation, meaning that as one variable goes up, the other goes up.



- ✓ *+1 : Perfectly positive*
- ✓ *-1 : Perfectly negative*
- ✓ *0 – 0.2 : No or very weak association*
- ✓ *0.2 – 0.4 : Weak association*
- ✓ *0.4 – 0.6 : Moderate association*
- ✓ *0.6 – 0.8 : Strong association*
- ✓ *0.8 – 1 : Very strong to perfect association*

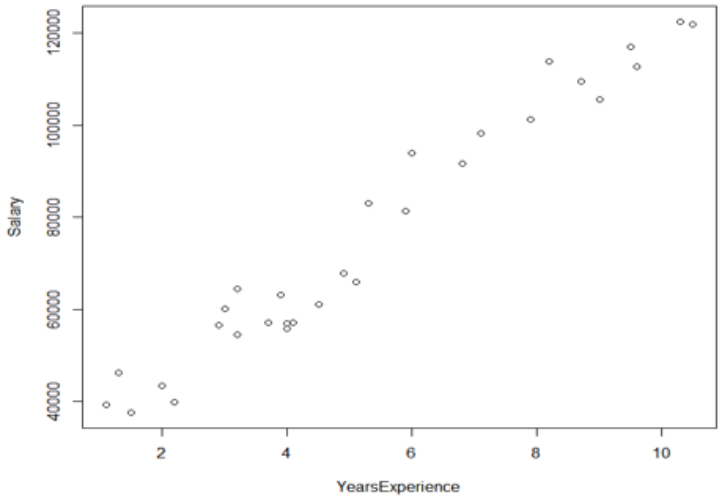
- Consider the following dataset (download from https://www.kaggle.com/rohankayan/years-of-experience-and-salary-dataset#Salary_Data.csv) which contains two variables yearsExperience and Salary. Let find correlation among two variables in R Programming

YearsExperience	Salary
1.1	39343
1.3	46205
1.5	37731
2	43525
2.2	39891
2.9	56642
3	60150
3.2	54445
3.2	64445

R-Code For Correlation

```
d<-read.csv("E:/kranthi reddy/AY 2021-2022/Unit-3/data set/Salary_Data.csv")
plot(d)
cor(d$YearsExperience,d$Salary)
cor(d)
```

Correlation Output




```
> cor(d$YearsExperience,d$Salary)
[1] 0.9782416
> cor(d)
                 YearsExperience    Salary
YearsExperience        1.0000000 0.9782416
Salary                 0.9782416 1.0000000
```

Regression:

- ***Correlation measures*** the strength of the relationship between two variables.
- **In Regression Analysis,** we can estimate the value of one variable with value of the other variable which is known.
- The ***Regression*** is statistical method used to estimate the unknown value of one variable from the known value of the related variable is called regression.
- ***Regression analysis*** is a set of statistical methods used for the estimation of relationships between a dependent variable and one or more independent variables. It can be utilized to assess the strength of the relationship between variables and for modeling the future relationship between them.

Advertisement	Sales
$90	$1000
$120	$1300
$150	$1800
$100	$1200
$130	$1380
$200	??

- Now, the company wants to do the advertisement of $200 in the year 2019 **and wants to know the prediction about the sales for this year**.
- To solve such type of prediction problems in machine learning, we need regression analysis.

Some examples of regression can be as:

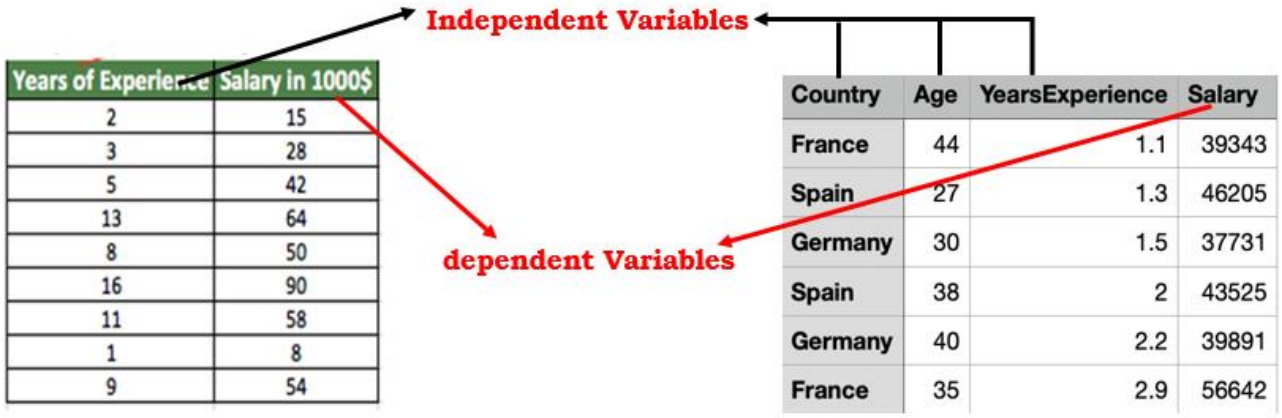
- ✓ Prediction of rain using temperature and other factors
- ✓ Determining Market trends
- ✓ Prediction of road accidents due to rash driving.

Independent Variable or Explanatory Variable

- *The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variable, also called as a predictor.*

Dependent Variable or response variable

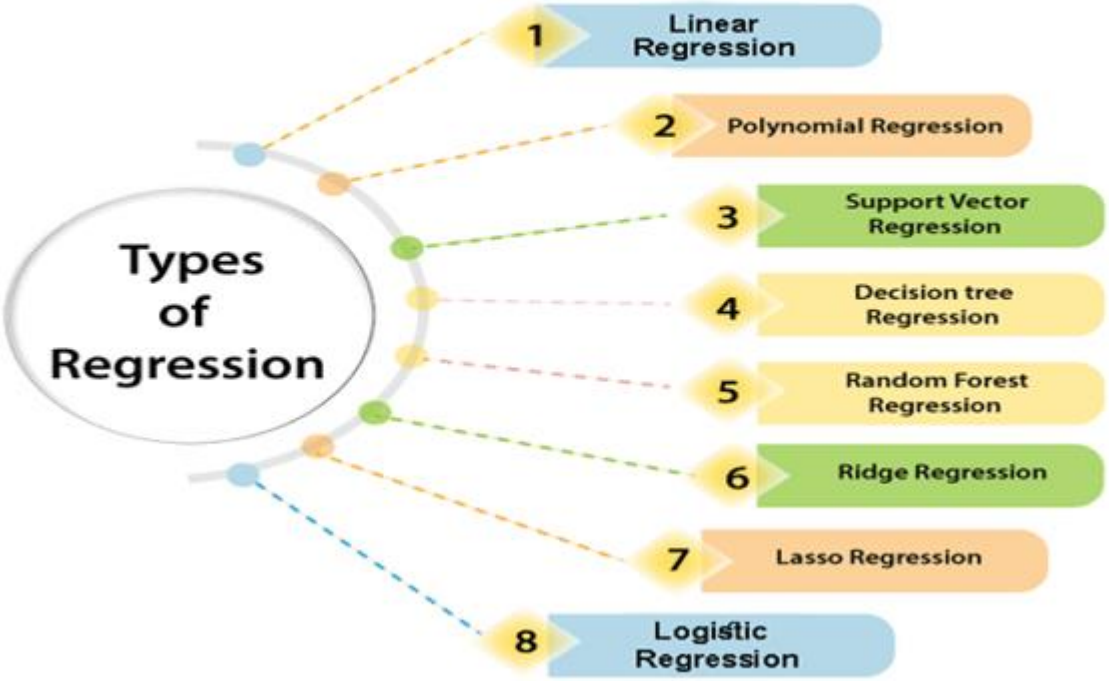
- *The main factor in Regression analysis which we want to predict or understand is called the dependent variable. It is also called target variable.*
- ***Independent variable causes an effect on the dependent variable.***
- ***Example: How long you sleep (independent variable) affects your test score (dependent variable).***



Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

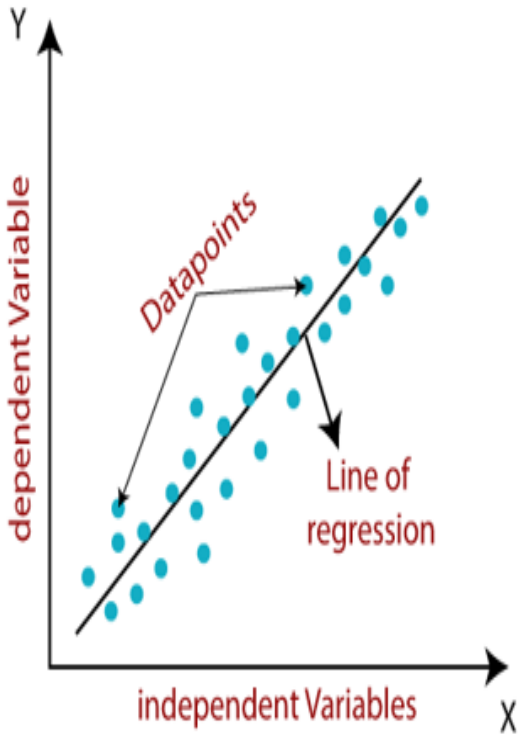
Country	Age	YearsExperience	Salary
France	44	1.1	39343
Spain	27	1.3	46205
Germany	30	1.5	37731
Spain	38	2	43525
Germany	40	2.2	39891
France	35	2.9	56642

Types of Regression



Linear Regression:

- Linear regression is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as ***sales, salary, age, product price, etc.***
- ***Linear regression algorithm*** shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression.
- ***Linear regression*** shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.
- The linear regression model provides a sloped straight line representing the relationship between the variables.



$$Y = a_0 + a_1 x$$

Y = Dependent Variable (Target Variable)

X = Independent Variable (Predictor Variable)

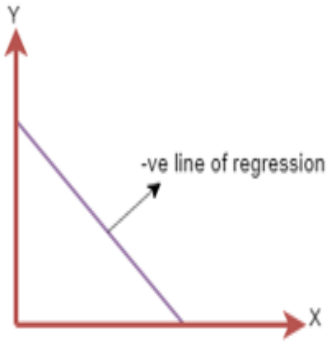
a0 = Intercept of the line

a1 = Linear regression coefficient

Regression Line:

A linear line showing the relationship between the dependent and independent variables is called a **regression line**. A regression line can show two types of relationship:

The line equation will be: $Y = a_0 + a_1 x$



The line of equation will be: $Y = -a_0 + a_1 x$

Consider the following data set which contains years experience and salary as independent and dependent variable. Let us intercept and slope for this data set values by using mathematical formulas.

Independent Variables **dependent Variables**

Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

Linear Regression function:

$$y=mx+c$$

Slope(m) of regression line

r -- Correlation

Sy -- standard deviation of Y

Sx -- standard deviation of X

$$m = r * \frac{Sy}{Sx}$$

Y – Intercept of Regression line

$$c = \bar{y} - m * \bar{x}$$

Independent Variables **dependent Variables**

Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

Slope(m) of regression line

r -- Correlation

Sy -- standard deviation of Y

Sx -- standard deviation of X

$$m = r * \frac{Sy}{Sx}$$

$$Corr(x,y) = \frac{\sum_{i=1}^{n}(x_i - x')\,(y_i - y')}{\sqrt{\sum_{i=1}^{n}(x_i - x')^2 \sum_{i=1}^{n}(y_i - y')^2}}$$

$$S_y = \sqrt{\frac{\sum (y - \bar{y})^2}{n-1}} \qquad S_x = \sqrt{\frac{\sum (x - \bar{x})^2}{n-1}}$$

Slope m = 3.86 **Intercept y = 20.55**

Linear Regression in R Programming:

Independent Variables **dependent Variables**

Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

R CODE for Linear Regression

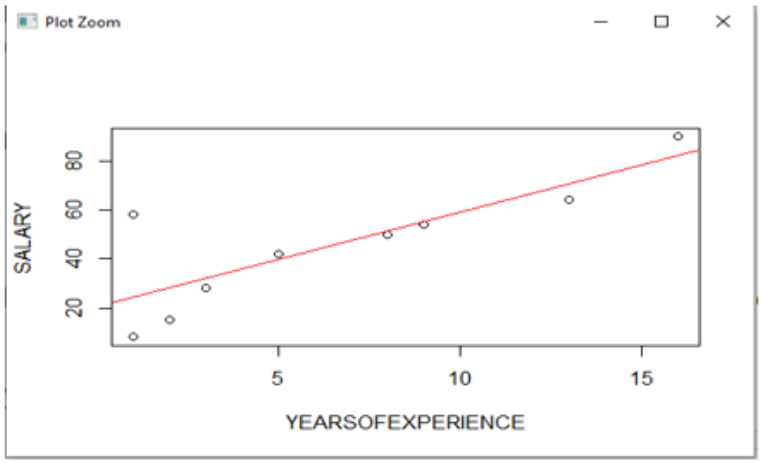
```
d1<-read.csv("E:/kranthi reddy/AY 2021-2022/Unit-3/data set/s3.csv")
attach(d1)
nrow(d1)
plot(d1)
l3<-lm(SALARY~YEARSOFEXPERIENCE,d1)
abline(l3,col="red")
summary(l3)
l3
```

The following graph generated in R Programming

Independent Variables **dependent Variables**

Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

R CODE for Linear Regression



As per R- Code '13' is linear regression model contains slope and intercept values as follows:

Independent Variables **dependent Variables**

Years of Experience	Salary in 1000$
2	15
3	28
5	42
13	64
8	50
16	90
11	58
1	8
9	54

R CODE for Linear Regression

```
> l3

Call:
lm(formula = SALARY ~ YEARSOFEXPERIENCE, data = d1)

Coefficients:
      (Intercept)  YEARSOFEXPERIENCE
           20.558              3.862
```

Predict Salary for an new input year of experience by using predict() in R programming as follows:

R CODE for Linear Regression

```
> predict(l3,list(YEARSOFEXPERIENCE=9))
       1
55.31326
> predict(l3,list(YEARSOFEXPERIENCE=5))
       1
39.86642
> predict(l3,list(YEARSOFEXPERIENCE=9))
       1
55.31326
> predict(l3,list(YEARSOFEXPERIENCE=1))
       1
24.41957
```

R Code for Linear Regression:

```
> #Linear Regression
```

Loading dataset

```
> d1<-read.csv("E:/kranthi reddy/AY 2021-2022/Unit-3/data set/s3.csv")
> View(d1)
```

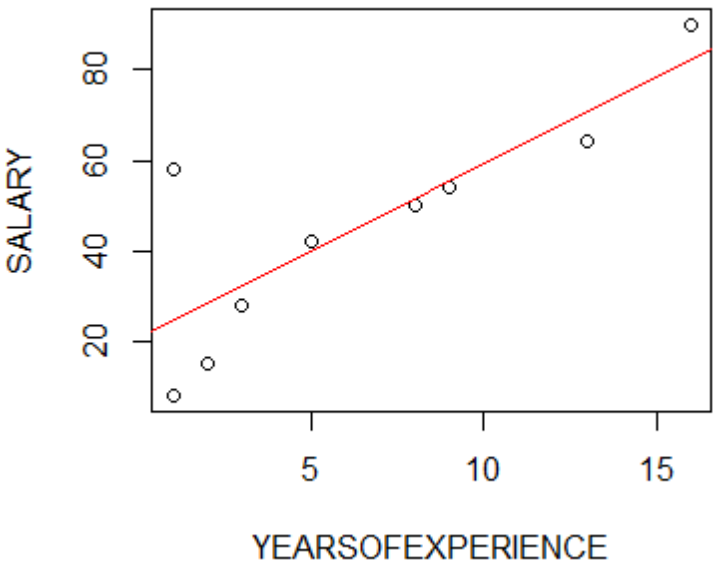
YEARSOFEXPERIENCE	SALARY
2	15
3	28
5	42
13	64
8	50
16	90
1	58
1	8
9	54

Finding number of rows in dataset

```
> nrow(d1)
[1] 9
```

Generating graph

```
> plot(d1)
```



Generating linear regression model

```
> l3<-lm(SALARY~YEARSOFEXPERIENCE,d1)
> l3

Call:
lm(formula = SALARY ~ YEARSOFEXPERIENCE, data = d1)
```

```
Coefficients:
     (Intercept)  YEARSOFEXPERIENCE
          20.558              3.862

> abline(l3,col="red")
> #summary(l3)
> l3

Call:
lm(formula = SALARY ~ YEARSOFEXPERIENCE, data = d1)

Coefficients:
     (Intercept)  YEARSOFEXPERIENCE
          20.558              3.862

> predict(l3,list(YEARSOFEXPERIENCE=5))
       1
39.86642
> predict(l3,list(YEARSOFEXPERIENCE=7))
       1
47.58984
```

Regression Residuals:

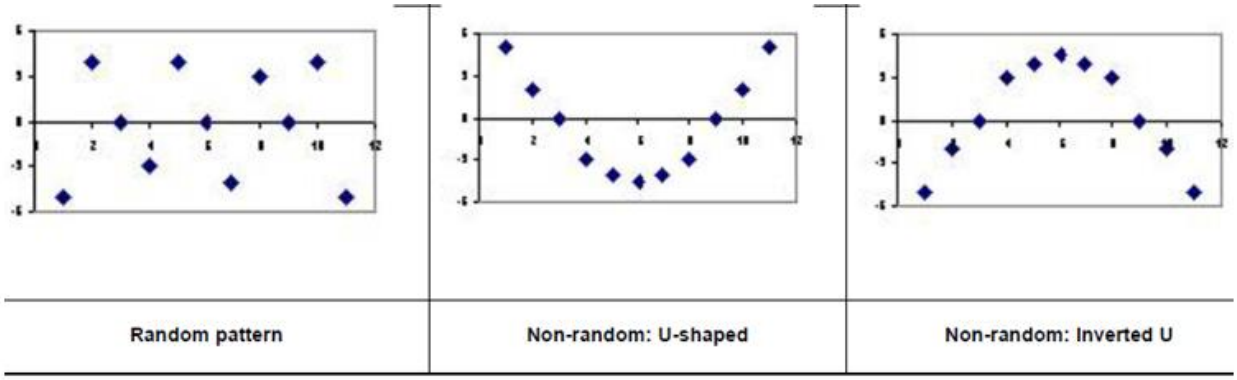
- The residual of an observed value is the difference between the observed value
  and the estimated value of the quantity of interest.
- Because a linear regression model is not always appropriate for the data,
- we should assess the appropriateness of the model by defining residuals and examining residual plots.

Residuals

- The difference between the observed value of the dependent variable (y) and the predicted value (ŷ) is called the **residual (e).** Each data point has one residual.

  Residual = Observed value - Predicted value

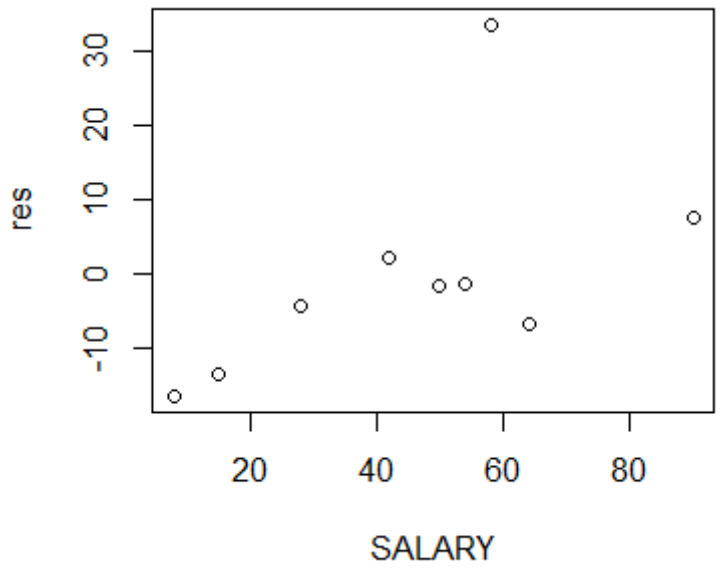
  $$e = y - \hat{y}$$
- A residual plot is a graph that shows the residuals on the vertical axis and the independent variable on the horizontal axis.
- If the points in a residual plot are randomly dispersed around the horizontal axis, then linear regression model is appropriate for the data; otherwise, a non-linear model is more appropriate for the data.
- The following figure shows the input and output from a simple linear regression analysis.



✓ **The first plot shows a random pattern, indicating a good fit for a linear model.**

✓ **The other plot patterns are non-random (U-shaped and inverted U), suggesting a better fit for a non-linear model.**

```
>  res=resid(l3)
> res
         1          2          3          4          5          6          7
8          9
-13.281279  -4.142992   2.133584  -6.760113  -1.451552   7.654751  33.580433 -16
.419567  -1.313264
> pv=predict(l3)
> SALARY-pv
         1          2          3          4          5          6          7
8          9
-13.281279  -4.142992   2.133584  -6.760113  -1.451552   7.654751  33.580433 -16
.419567  -1.313264
> plot(SALARY,res)
```



The above graph is random pattern then the model is consider as efficient model

Multiple Linear Regression:

- Multiple regression is an extension of linear regression into relationship between more than two variables. In simple linear relation we have one Dependent variable and one Independent variable, but in multiple regression we have more than one independent variable and one dependent variable.
- The general mathematical equation for multiple regression is –
  y = a + b1x1 + b2x2 +...bnxn
  - ✓ **y** is the response variable.
  - ✓ **a, b1, b2...bn** are the coefficients.
  - ✓ **x1, x2, ...xn** are the predictor variables.

- We create the regression model using the lm() function in R.
- The model determines the value of the coefficients using the input data. Next we can predict the value of the response variable for a given set of independent variables using these coefficients.
- **lm():** This function creates the relationship model between the independent and the dependent variable.
  - ✓ **Syntax:**
  - ✓ lm(y ~ x1+x2+x3...,data)
  - ✓ Formula is a symbol presenting the relation between the response variable and predictor variables.
  - ✓ data is the vector on which the formula will be applied.

➢Apply Multiple Regression on employee_data data set:

YearsExperience	age	Salary
1.1	36	39343
1.3	55	46205
1.5	61	37731
2	29	43525
2.2	34	39891
2.9	42	56642
3	53	60150
3.2	41	54445
3.2	47	64445

R-Code for Multiple Regression:

```
d2 <- read.csv("E:/datasets/new/employee_data.csv")
d2
attach(d2)
model <- lm(Salary~age+YearsExperience, input)
model
summary(model)
```

➢ How to Predict dependent variable when we know the value of independent variable.

```
predict(l,data.frame(age=29,YearsExperience=3.6))
 1
59812.06
```

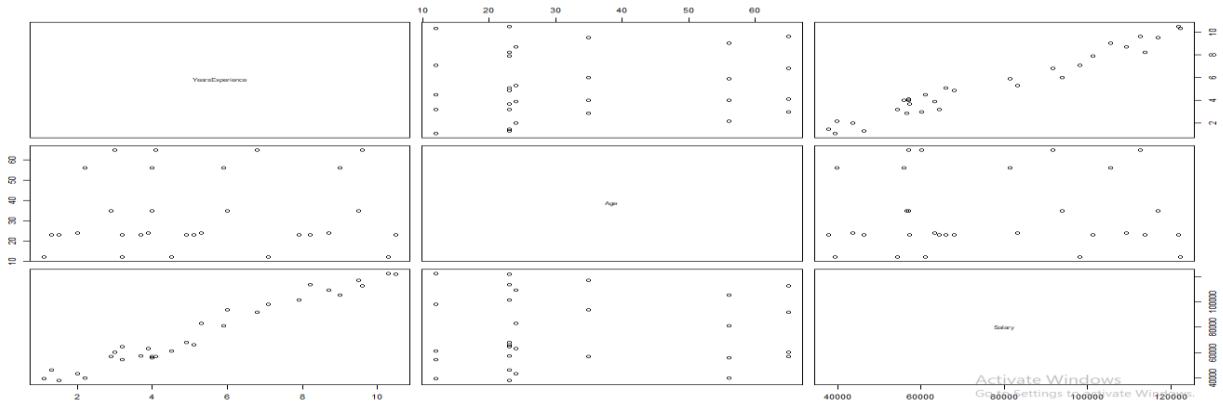
R Code for Multiple Linear Regression:

Loading dataset

```
> #Multiple Linear Regression
> d2<-read.csv("E:/kranthi reddy/AY 2021-2022/Unit-3/data set/Salary_Data1.csv")
> attach(d2)
> nrow(d2)
[1] 30
> View(d2)
```

YearsExperience	Age	Salary
1.1	12	39343
1.3	23	46205
1.5	23	37731
2.0	24	43525
2.2	56	39891
2.9	35	56642
3.0	65	60150
3.2	12	54445
3.2	23	64445
3.7	23	57189
3.9	24	63218
4.0	56	55794
4.0	35	56957
4.1	65	57081
4.5	12	61111

```
> plot(d2)
```




```
> l3<-lm(Salary~YearsExperience+Age,d2)
> l3

Call:
lm(formula = Salary ~ YearsExperience + Age, data = d2)

Coefficients:
    (Intercept)  YearsExperience              Age
       27936.93          9481.80          -70.33

> predict(l3,list(Age=24,YearsExperience=3.9))
    1
63228
> predict(l3,list(Age=24,YearsExperience=2))
       1
45212.57
```

Least Square Estimation

Least Squares Estimation is a method used in statistical modelling to find the parameters of a model that minimize the sum of the squared differences between the observed (actual) and predicted values. This approach is commonly employed in linear regression but can be extended to other types of models as well.

For linear regression, the goal is to find the slope (m) and the intercept (b) of a line that best fits the given data points. The model equation is typically represented as

y=mx+b,

where y is the dependent variable,

x is the independent variable,

m is the slope, and

b is the y-intercept.

The sum of squared differences (residuals) between the observed y values (yi) and the predicted values (yi) for each data point is defined as:

$$\text{Minimize:} \quad \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

This minimization problem is solved using the method of least squares, which involves finding the values of m and b that minimize the sum of squared differences. For a simple linear regression model, the formulas for the slope (m) and intercept (b) that minimize the sum of squared differences are:

Blue Property Assumptions and Least Square Estimations:

- Linear regression models have several applications in real life. In econometrics, Ordinary Least Squares (OLS) method is widely used to estimate the parameters of a linear regression model.
- For the validity of OLS estimates, there are assumptions made while running linear regression models.
  - ✓ A1:The linear regression model is “linear in parameters.”
  - ✓ A2:There is no multi-collinearity (or perfect collinearity).
  - ✓ A3:There is homoscedasticity and no auto-correlation
  - ✓ A4:Error terms should be normally distributed.

Multi-collinearity

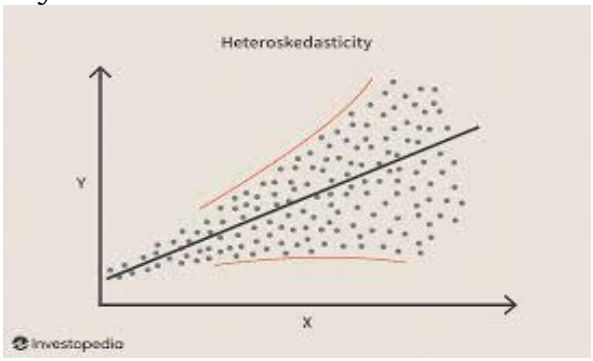
- Multicollinearity is a situation in linear regression where two or more independent variables in a model are highly correlated with each other.In a model with correlated variables, it becomes a tough task to figure out the true relationship of a predictors with response variable. In other words, it becomes difficult to find out which variable is actually contributing to predict the response variable.
- The presence of multicollinearity leads to Uncertain Variable Importance & Unstable Coefficients.
- To overcome multicollinearity choose appropriate variables while constructing model & Collect more diverse data

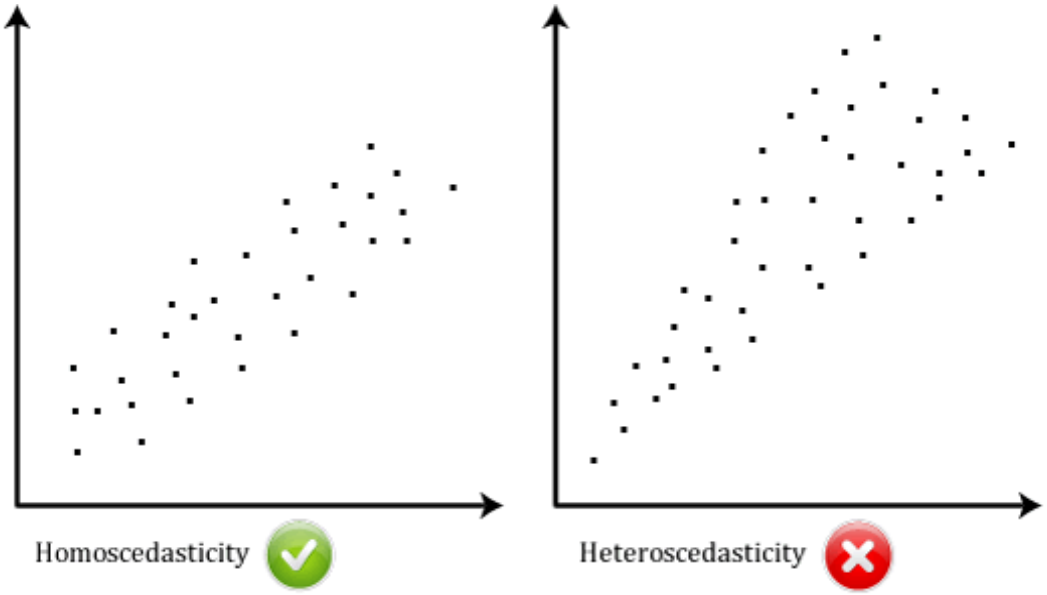
Autocorrelation

- Autocorrelation, also known as serial correlation or cross-autocorrelation. autocorrelation (also known as serial correlation) occurs when the error terms (residuals) exhibit correlation across different time periods. The presence of correlation in error terms drastically reduces model's accuracy. This usually occurs in time series models where the next instant is dependent on previous instant.
- Consider that you are attempting to forecast monthly sales using advertising costs as a basis. After fitting a regression model, you see that the errors have a pattern. According to autocorrelation, if you make a mistake (overestimated or underestimated sales) in a given month, there's a good chance that the following month will see a similar error trend. can u concise it
- Positive Autocorrelation: If you underestimated sales in June, there's a higher chance you'll also underestimate sales in July.
- Negative Autocorrelation: If you overestimated sales in June, there's a higher chance you'll underestimate sales in July.

Homoscedasticity

- In regression analysis , homoscedasticity means a situation in which the variance of the dependent variable is the same for all the data. Homoscedasticity is facilitates analysis because most methods are based on the assumption of equal variance.





- Homoscedasticity in regression, in simple words, means that the spread or variability of the residuals (the differences between the observed and predicted values) is roughly the same across all levels of the independent variable(s).
- **Homoscedasticity (Good):** The differences between predicted and actual values are spread out evenly as you move along the range of predictor values.
- **Heteroscedasticity (Not Ideal):** The differences between predicted and actual values show a pattern where the spread increases or decreases systematically with changes in the predictor variable.

Homoscedasticity:

- Scatter Plot: Picture points scattered evenly around the regression line, creating a consistent and roughly rectangular band.

- Residual Plot: Envision the residuals (vertical distances between points and the line) scattered randomly without any clear pattern, maintaining a consistent spread across all values of the independent variable.

Heteroscedasticity:

- Scatter Plot: Visualize points forming a cone-like shape, either widening or narrowing as the independent variable increases.
- Residual Plot: Notice a clear pattern in the residuals, like a cone shape or inverted funnel shape

- These assumptions are extremely important because violation of any of these assumptions would make OLS estimates unreliable and incorrect.
- **Properties of OLS Regression Estimator in details:**
  - ✓ **Property 1 : Linear**
  - ✓ **Property 2 : Unbiasedness**
  - ✓ **Property 3 : Best Minimum Variance**
  - ✓ **Property 4 : Consistency**

Property 1 : Linear

In assumption A1, the focus was that the linear regression should be "linear in parameters."

OLS estimators are linear only with respect to the dependent variable and not necessarily with respect to the independent variables.

The linear property of OLS estimators doesn't depend only on assumption A1 but on all assumptions A1 to A5.

Property 2: Unbiasedness

In regression analysis, bias and unbiasedness refer to the properties of estimators used to estimate the coefficients of the regression model.

Biased Estimator: An estimator for a regression coefficient is considered biased if, on average, it systematically overestimates or underestimates the true population value of that coefficient. For example, if the average value of the estimated coefficient from different samples consistently deviates from the true population value, the estimator is biased.

Unbiased Estimator: An estimator for a regression coefficient is considered unbiased if, on average, it provides accurate estimates of the true population value of that coefficient.

For example, if the average value of the estimated coefficient from different samples converge to the true population value, the estimator is unbiased.

Let us consider dataset that contains x independent variable and y dependent variable. A regression line/model is fitted (y=mx+c). Assume that the value of m is 3 and value of c is 2.

- ✓ Assume you collect multiple random samples from the population, fit a linear regression model for each sample, and calculate the average of the estimated intercepts(c) and slopes
- ✓ If, on average, the estimated intercepts and slopes converge to the true population intercept (2) and slope(3), then the estimator for the intercept and slope are unbiased.

Property 3: Minimum Variance

In a regression model, you estimate coefficients (like slope and intercept) to describe the relationship between the independent variable(s) and the dependent variable.

The variance of these coefficient estimates reflects how much the estimated values for the coefficients might vary from sample to sample.

Ex: Let us consider dataset that contains x independent variable and y dependent variable. A regression line/model is fitted (y=mx+c).

Case 1:

- ✓ The estimator m1 for the slope coefficient that has a high variance. This means , that across different samples the estimator m1 tends to vary widely.
- ✓ In Sample1, m1=3.2, in Sample2 m1=2.8 and in sample3 m1=2.9 The estimated coefficients significantly indicating high variance.

Case 2:

- ✓ The estimator m1 for the slope coefficient that has a low variance. This means that across different samples the estimator m1 tends to less variation.
- ✓ In Sample1, m1=3.0, in Sample2 m1=3.1 and in sample3 m1=2.9 The estimated coefficients are more consistent and don't deviates, indicating high variance.

Property 5: Consistency

An estimator is said to be consistent if its value approaches the actual, true parameter (population) value as the sample size increases. An estimator is consistent if it satisfies two conditions:

a. It is asymptotically unbiased

b. Its variance converges to 0 as the sample size increases.

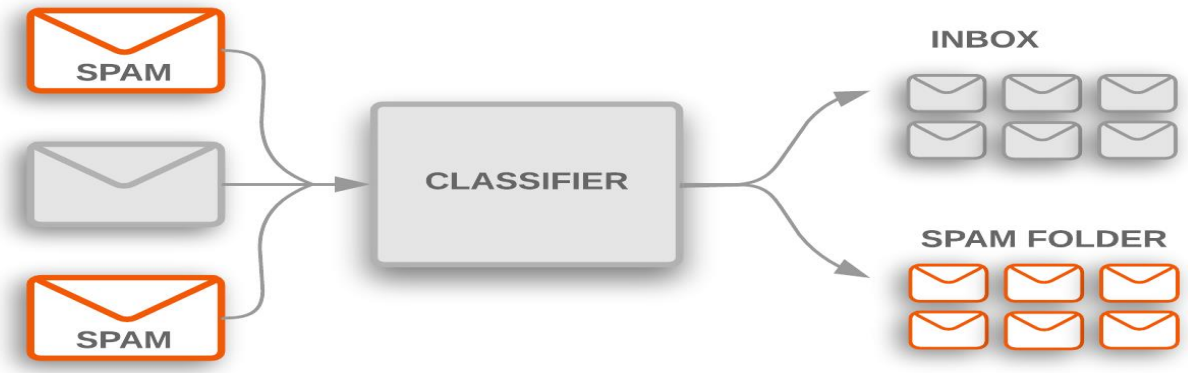
Both these hold true for OLS estimators and, hence, they are consistent estimators.

Best Linear Unbiased Estimators (BLUE):

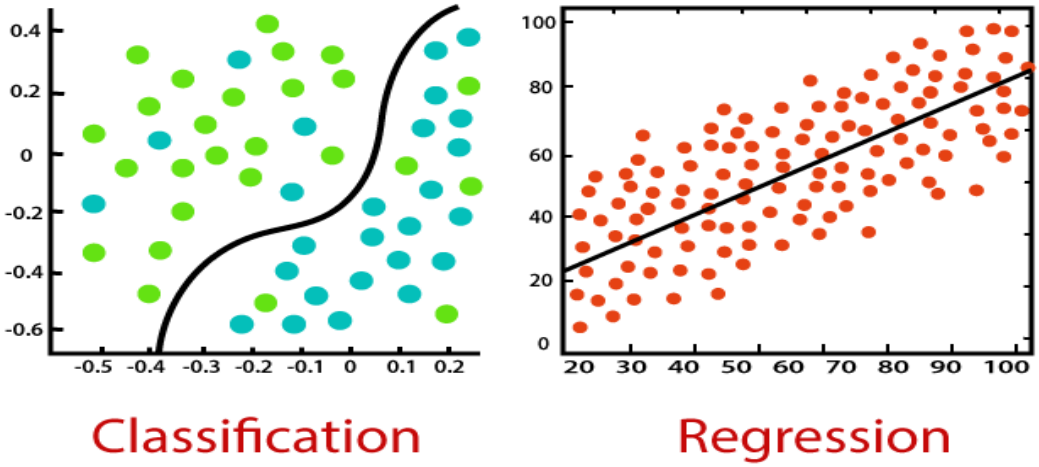
- In ordinary least squares (OLS) regression, the estimators for the regression coefficients are considered BLUE when they are unbiased and have minimum variance.
- The term "Best" in BLUE implies the efficiency aspect, with variance being a key factor.

Classification Methods

- Classification is a technique where we categorize data into a given number of classes. The main goal of a classification problem is to identify the category/class to which a new data will fall under.



- Regression and Classification algorithms are Supervised Learning algorithms. Both the algorithms are used for prediction in Machine learning and work with the labeled datasets. But the difference between both is how they are used for different machine learning problems.
- The main difference between Regression and Classification algorithms that ***Regression algorithms are used to predict the continuous values such as price, salary, age, etc.*** and ***Classification algorithms are used to predict/Classify the discrete values such as Male or Female, True or False, Spam or Not Spam, etc.***



Classification Algorithms can be further divided into the following types:

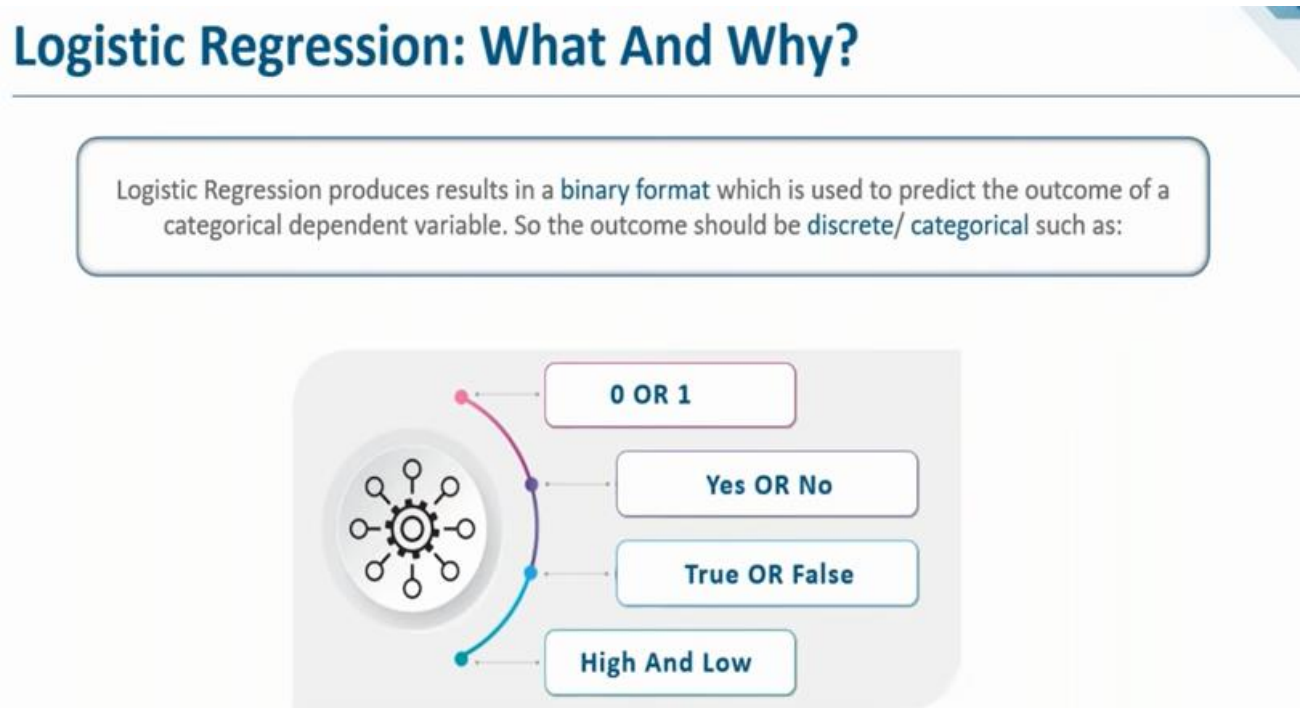
- ✓ Logistic Regression
- ✓ K-Nearest Neighbours

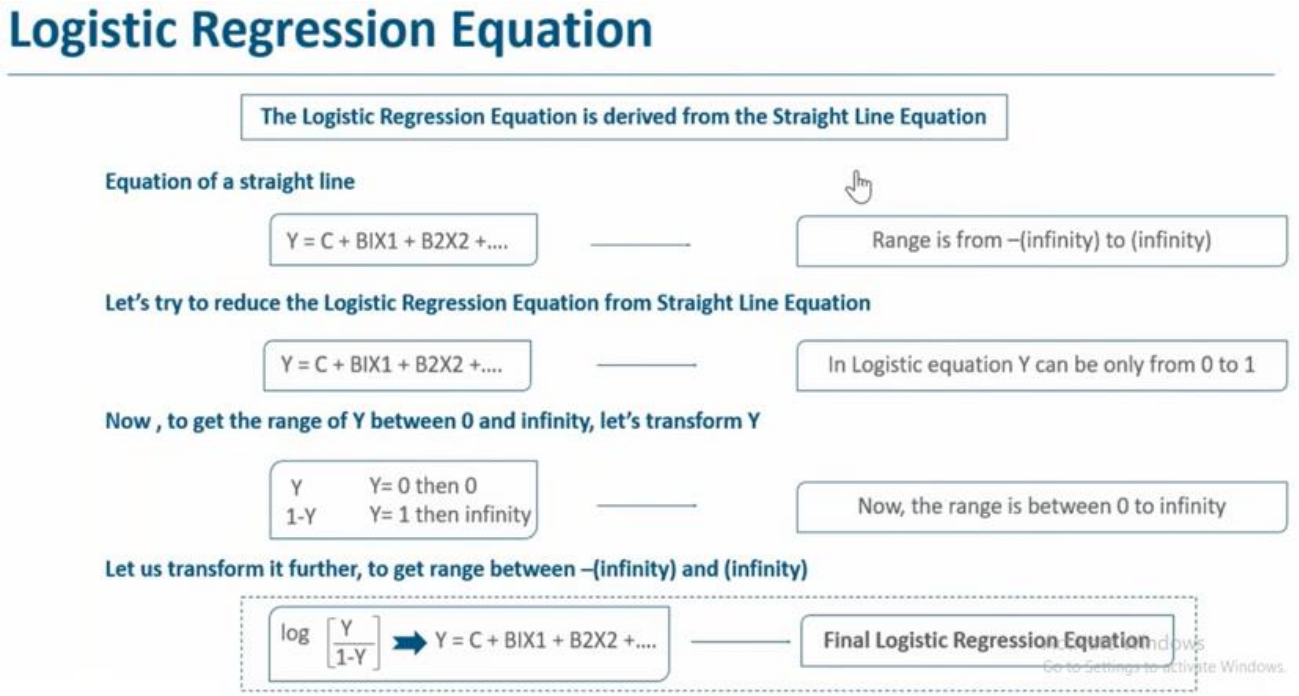
- ✓ Support Vector Machines
- ✓ Kernel SVM
- ✓ Naïve Bayes
- ✓ Decision Tree Classification
- ✓ Random Forest Classification

Classification Methods:

- **Classifier:** An algorithm that maps the input data to a specific category.
- **Classification model:** A classification model tries to draw some conclusion from the input values given for training. It will predict the class labels/categories for the new data.
- **Binary Classification:** Classification task with two possible outcomes. Eg: Gender classification (Male / Female)
- **Multi-class classification:** Classification with more than two classes. In multi class classification each sample is assigned to one and only one target label. Eg: An animal can be cat or dog but not both at the same time
- **Multi-label classification:** Classification task where each sample is mapped to a set of target labels (more than one class). Eg: A news article can be about sports, a person, and location at the same time.

Logistic Regression:





- A Logistic Regression model is similar to a Linear Regression model, except that the Logistic Regression utilizes a more sophisticated cost function, which is known as the ***"Sigmoid function" or "logistic function"*** instead of a linear function.
- Logistic Regression is a method used to predict a dependent variable (Y), given an independent variable (X), such that the dependent variable is categorical.
- **Dependent Variable(Y):** The response binary variable holding values like 0 or 1, Yes or No, A or B
- **Independent Variable (X):** The predictor or explanatory variable used to represent a linear regression model.

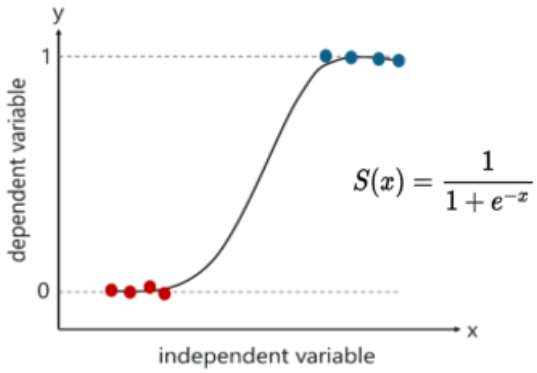
➢**Logistic Regression is a method used to predict a dependent variable (Y), given an independent variable (X), such that the dependent variable is categorical.**

- **Dependent Variable(Y):**

The response binary variable holding values like 0 or 1, Yes or No, A or B

- **Independent Variable (X):**

The predictor or explanatory variable used to represent a linear regression model.



A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

Confusion matrix is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with four different combinations of predicted and actual values.

It is extremely useful for measuring Recall, Precision, Specificity, Accuracy, and most importantly AUC-ROC curves.

		Actual Values: Positive (1)	Actual Values: Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

- **True Positive (TP):** Instances where the model correctly predicts the positive class.
- **True Negative (TN):** Instances where the model correctly predicts the negative class.

- **False Positive (FP):** Instances where the model predicts the positive class, but the actual class is negative.
- **False Negative (FN):** Instances where the model predicts the negative class, but the actual class is positive.

	Actual Not Spam	Actual Spam
Predicted Not Spam	850	20
Predicted Spam	30	100

- True Positive (TP): 100 emails were correctly predicted as spam.
- True Negative (TN): 850 emails were correctly predicted as not spam.
- False Positive (FP): 20 emails were incorrectly predicted as spam.
- False Negative (FN): 30 emails were incorrectly predicted as not spam.

Logistic Model Construction:

Let us apply logistic regression on following data set:

admit	gre	gpa	ranl
0	380	3.61	3
1	660	3.67	3
1	800	4.00	1
1	640	3.19	4
0	520	2.93	4
1	760	3.00	2
1	560	2.98	1
0	400	3.08	2
1	540	3.39	3
0	700	3.92	2
0	800	4.00	4
0	440	3.22	1
1	760	4.00	1
0	700	3.08	2
1	700	4.00	1

The above data set contains three attributes admit, gre, gpa and rank.

The outcome variable is admit which is a binary outcome variable.

1. Load the data set by using read.csv()

```
> d=read.csv("E:/kranthi reddy/AY 2021-2022/Unit-3/data set/stud_admit.csv") #lo
ads data set.
```

2. Finding number of rows

```
> nrow(d) #number of rows
[1] 400
```

3. Checking whether the missing values exist in the data set

```
> sum(is.na(d)) #checking for missing values
[1] 0
```

4. viewing the descriptive statistics of data set.

```
> summary(d)
     admit             gre             gpa             rank
 Min.   :0.0000   Min.   :220.0   Min.   :2.260   Min.   :1.000
 1st Qu.:0.0000   1st Qu.:520.0   1st Qu.:3.130   1st Qu.:2.000
 Median :0.0000   Median :580.0   Median :3.395   Median :2.000
 Mean   :0.3175   Mean   :587.7   Mean   :3.390   Mean   :2.485
 3rd Qu.:1.0000   3rd Qu.:660.0   3rd Qu.:3.670   3rd Qu.:3.000
 Max.   :1.0000   Max.   :800.0   Max.   :4.000   Max.   :4.000
```

5. Checking type of variables

```
> str(d)  #checking type of variables
'data.frame':  400 obs. of  4 variables:
 $ admit: int  0 1 1 1 0 1 1 0 1 0 ...
 $ gre  : int  380 660 800 640 520 760 560 400 540 700 ...
 $ gpa  : num  3.61 3.67 4 3.19 2.93 3 2.98 3.08 3.39 3.92 ...
 $ rank : int  3 3 1 4 4 2 1 2 3 2 ...
```

6. Converting admit and rank attributes into categorical variables by using factors Function.

```
> d$admit=as.factor(d$admit)
> d$rank=as.factor(d$rank)
> str(d)
'data.frame':  400 obs. of  4 variables:
 $ admit: Factor w/ 2 levels "0","1": 1 2 2 2 1 2 2 1 2 1 ...
 $ gre  : int  380 660 800 640 520 760 560 400 540 700 ...
```

```
 $ gpa  : num  3.61 3.67 4 3.19 2.93 3 2.98 3.08 3.39 3.92 ...
 $ rank : Factor w/ 4 levels "1","2","3","4": 3 3 1 4 4 2 1 2 3 2 ...
> set.seed(1234)
```

7. Dividing the data set into two sets training set as 80% and testing set as 20%. The total data set consist of 400 entries out of which train set contains 325 where as test set contains 75

```
> train_test<-sample(2,nrow(d),replace=TRUE, prob=c(0.8,0.2))
> train=d[train_test==1,]
> nrow(train)
[1] 325
> test=d[train_test==2,]
> nrow(test)
[1] 75
```

8. applying logistic regression by using glm()

```
> m=glm(admit~gre+gpa+rank,d=train,family='binomial')
> summary(m)
Call:
glm(formula = admit ~ gre + gpa + rank, family = "binomial",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5873  -0.8679  -0.6181   1.1301   2.1178

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.009514   1.316514  -3.805 0.000142 ***
gre          0.001631   0.001217   1.340 0.180180
gpa          1.166408   0.388899   2.999 0.002706 **
rank2       -0.570976   0.358273  -1.594 0.111005
rank3       -1.125341   0.383372  -2.935 0.003331 **
```

```
rank4       -1.532942   0.477377  -3.211 0.001322 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 404.39  on 324  degrees of freedom
Residual deviance: 369.99  on 319  degrees of freedom
AIC: 381.99

Number of Fisher Scoring iterations: 4
```

9. Predicting the outcome of new input

```
> p1<-predict(m,data.frame(gre=700,gpa=4,rank=as.factor(1)),type='response')
> p1
        1
0.6895506
> res=ifelse(p1>0.5,1,0)
> res
1
1
```

9. Predicting outcome of test data set:

```
> p2<-predict(m, test, type='response')
> pred2<-ifelse(p2>0.5,1,0)
```

10. Generating confusion matrix for test data set.

```
> tab2<-table(Predicted=pred2, Actual=test$admit)
> tab2
         Actual
Predicted  0  1
        0 48 21
        1  2  4
> 1-sum(diag(tab2))/sum(tab2)
[1] 0.3066667
```

The model 'm' generates accuracy as 70% for student admission data set.

Applications of Business Analytics with Examples:

- **Finance**: BA is of utmost importance to the finance sector. Data Scientists are in high demand in investment banking, portfolio management, financial planning, budgeting, forecasting, etc.
- **Marketing:** Studying buying patterns of consumer behavior, analyzing trends, help in identifying the target audience, employing advertising techniques that can appeal to the consumers, forecast supply requirements, etc
- **HR Professionals**: HR professionals can make use of data to find information about educational background of high performing candidates, employee attrition rate, number of years of service of employees, age, gender, etc. This information can play a pivotal role in the selection procedure of a candidate.
- **Manufacturing:** Business Analytics can help you in supply chain management, inventory management, measure performance of targets, risk mitigation plans, improve efficiency in the basis of product data, etc.
- **Credit Card Companies:** Credit card transactions of a customer can determine many factors: financial health, life style, preferences of purchases, behavioral trends, etc.