Linear Regression

Q1. What is Linear Regression?

ANS: Linear Regression is the most commonly used supervised machine learning algorithm. Linear regression is used to discover a linear relationship between one dependent variable(Y) and one or more independent variable(X). Linear regression is also known as Ordinary Least-Squares(OLS). Linear regression is used to predict the future scores of the dependent variable(Y) based on the measured score of the independent variable(X) when the dependent variable(Y) is continuous such as salary, age, sales, product price, etc.

Q2. Why Linear Regression is Important?

ANS: The importance of Linear Regression is that it is one of the easy to understand machine learning algorithm that can help business owners to grow by understanding the data they have the factor which help their business to grow, the factor which is contributing to the growth of the business and the other factors which is not at all helping the business to grow. So after understanding the data knowing that which data is more significantly contributing to the growth and which is not contributing to the growth they can manipulate or change the data for maximum profit.

The linear regression algorithm is used for:

1.Predicting the sales of Company.

2.Predicting the house price.

3.For Insurance prediction.

Q3. What is the Equation of Linear Regression?

$Y = \beta_0 + \beta_1 X + \varepsilon$

Where,

Y : Dependent Variable

X : Independent Variable

β_0 : Y Intercept

β_1 : Slope Coefficient

ε : Error Term or Residual

Advertising Dataset:

TV	NEWSPAPER	RADIO	SALES
44	69	37	22
70	70	45	14
39	80	50	16
50	90	55	29

Description of the above Dataset: Basically it is Advertising Dataset in which we have to check what is the impact of various types of advertisement on sales.

Q3. What are Independent(X) and Dependent variables(Y)?

ANS: Explanation of these terms:

1.Dependent Variable(Y)

In a dataset we can have one or more independent variables and only one dependent variable.

A dependent variable is a variable that is dependent on the other variable it means that it can be changed by the other variables

Example of Dependent Variable:

In a dataset, we have a variable as sales which is nothing but the overall sales of a company etc. So this sales variable is dependent on many other factors or variables like advertisement by TV, RADIO, NEWSPAPER . So we can say that the sales variable is a dependent variable.

2.Independent Variable(X)

An Independent variable is a variable that is not dependent on the other variable it means that it cannot be changed by the other variables.

OR

The variable that is controlled throughout the experiment but is not affected by other variables is called as an independent variable.

Example of Independent Variable:

If the dependent variable is sales then there are many factors or variable that will definitely affect the dependent variable such as like advertisement by TV, RADIO, NEWSPAPER hence all these variables can be identified as independent variables.

Q4. What is β_0 OR Y Intercept?

ANS: Y intercept is nothing but the point where the function cuts or intersect the Y-axis, when the value of X = 0.

Q5. What is β_1 OR Slope coefficient?

ANS: Usually the slope coefficient refers to the coefficient of an independent variable(X) in a regression equation. It tells the amount of change in dependent variable(Y) that can be expected to result from a unit increase in independent variable(X).

Q6. What is ε OR Error Term OR Residual?

ANΣ: ε is nothing but the distance between the regression line and the data point.

Q7. What is correlation?

ANS:

- Correlation measures the relative strength of linear relationship between two variables Independent(X) and Dependent Variable(Y)
- Correlation ranges from -1 to 1.
- The closer to -1, the stronger is the negative linear relationship.
- The closer to 1, the stronger is the positive linear relationship.
- The closer to 0, the weaker is the positive linear relationship.

Q8. What is Positive Correlation?

ANS:

Income(X)	Savings(Y)
10	5
20	10
30	15
40	50
50	25

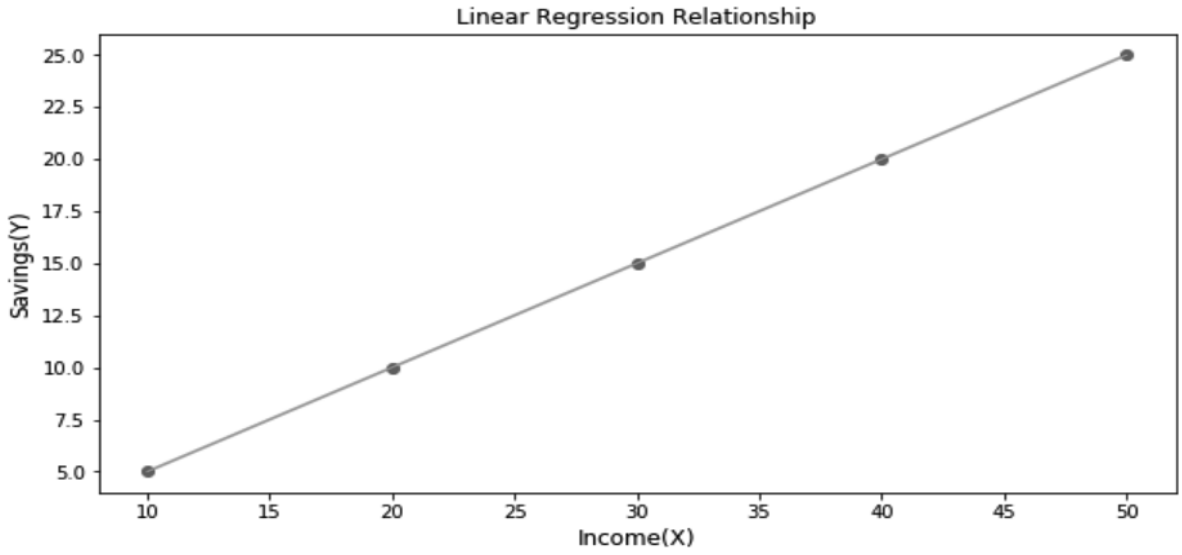
The above table shows that there are 2 columns in it Income(X) which is the Independent Variable and Savings(Y) which is the Dependent Variable. As we can see in the above table as the Income(X) increases the Savings(Y) increases this is called Positive Correlation.

Now let us try to plot the above data points on a graph using python.

Python Code:

```
import numpy as np
import matplotlib.pyplot as plt      #For ploting the graph
Income = np.array([10, 20, 30, 40,50])   #Income variable for storing income
Savings = np.array([ 5, 10, 15, 20, 25 ])  #Savings variable for storing savings
slope, intercept = np.polyfit(Income, Savings, 1)
plt.figure(figsize = (10,5))
plt.plot(Income, Savings, 'o')
plt.plot(Income, slope*Income + intercept)
plt.title("Linear Regression Relationship")
plt.ylabel("Savings(Y)",fontsize = 12)
plt.xlabel("Income(X)",fontsize = 12)
plt.show()
```

```
import numpy as np
import matplotlib.pyplot as plt
Income = np.array([10, 20, 30, 40,50])
Savings = np.array([ 5, 10, 15, 20, 25 ])
slope, intercept = np.polyfit(Income, Savings, 1)
plt.figure(figsize = (10,5))
plt.plot(Income, Savings, 'o')
plt.plot(Income, slope*Income + intercept)
plt.title("Linear Regression Relationship")
plt.ylabel("Savings(Y)",fontsize = 12)
plt.xlabel("Income(X)",fontsize = 12)
plt.show()
```



So as we can see in the graph as income increases the saving increases this is known as Positive Correlation.

Q9. What is Negative Correlation?

ANS:

Age(X)	Eggs(Y)
2	50
4	40
6	30
8	20
10	10

In the above table we have Age of Chicken and the number of eggs lays by the chicken. As we can see in the above table as the age increases the number of Eggs lays by the chicken decreases this is called Negative Correlation.

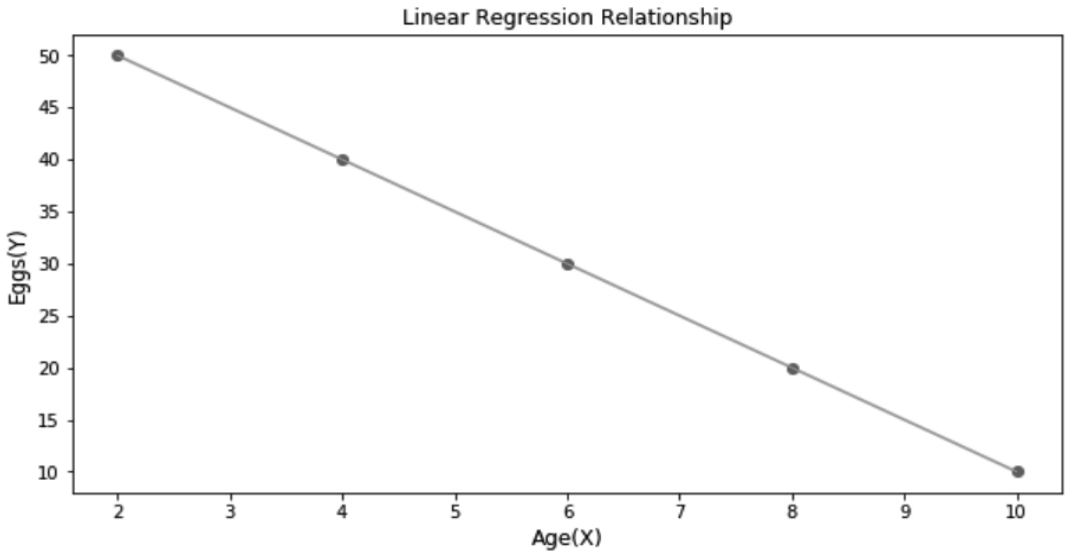
Now let us try to plot the above data points on a graph using python.

Python Code:

```
import numpy as np
import matplotlib.pyplot as plt
Age = np.array([2,4,6,8,10])
Eggs = np.array([50,40,30,20,10])
slope, intercept = np.polyfit(Age, Eggs, 1)

plt.figure(figsize = (10,5)
plt.plot(Age, Eggs, 'o')
plt.plot(Age, slope*Age + intercept)
plt.title("Linear Regression Relationship")
plt.ylabel("Eggs(Y)",fontsize = 12)
plt.xlabel("Age(X)",fontsize = 12)
plt.show()
```

```
import numpy as np
import matplotlib.pyplot as plt
Income = np.array([10, 20, 30, 40,50])
Savings = np.array([ 5, 10, 15, 20, 25 ])
slope, intercept = np.polyfit(Income, Savings, 1)
plt.figure(figsize = (10,5))
plt.plot(Income, Savings, 'o')
plt.plot(Income, slope*Income + intercept)
plt.title("Linear Regression Relationship")
plt.ylabel("Savings(Y)",fontsize = 12)
plt.xlabel("Income(X)",fontsize = 12)
plt.show()
```



So as we can clearly see in the graph as Age increases the number of Eggs lays by the chicken decreases this is known as Negative Correlation.

Q10. What are the types of Linear Regression?

ANS: There are two types of Linear Regression:

1.Simple linear regression.

2.Multiple linear regression.

Q11. What is a Simple linear regression?

ANS: Simple linear regression or Simple regression is a supervised machine learning algorithm. Simple regression as the name suggest it has only two variables in which one of them is dependent variable(Y) and other one is the independent variable(X). Simple linear regression is a technique used to discover a linear relationship between one dependent variable(Y) and one independent variable(X).

General form of Simple linear regression:

$Y = \beta_0 + \beta_1 X + \varepsilon$

Where,

Y : Dependent Variable

X : Independent Variable

β_0 : Y Intercept

β_1 : Slope Coefficient

ε : Error Term or Residual

Q12. What is a Multiple linear regression?

ANS: Multiple linear regression or Multiple regression is a supervised machine learning algorithm. Multiple regression as the name suggest contains multiple variable in which we have only one dependent variable(Y) but multiple independent variables(X). Multiple linear regression is a technique used to discover a linear relationship between the dependent variable(Y) and independent variables(X).

General form of Simple linear regression:

$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \ldots\ldots\ldots + \beta_x X_x + \varepsilon$

Where,

Y : Dependent Variable

X : Independent Variable

β_0 : Y Intercept

β_1 : Slope Coefficient

ε : Error Term or Residual

Q13. What are the advantages of Linear Regression?

ANS: The advantages of Linear Regression are as follow:

- The Linear regression is easy to understand and simple.
- Easy to interpret the output.
- Linear Regression is less complex compared to other Machine learning algorithm.
- When we have a regression problem the first choice of every individual is Linear Regression.

- Linear Regression tends towards overfitting but can be reduced by applying or implementing regularization L1 and L2.

Q14. What are the disadvantages of Linear Regression?

ANS: The disadvantages of Linear Regression are as follow:

- The major disadvantage of Linear Regression is the assumptions of Linear Regression in many real-life scenario the assumptions are not met so in this case it is very difficult to produce a useful result.
- Underfitting occurs in Linear Regression when the model fails to fit the data properly.

Q15. What is Overfitting?

ANS: Overfitting is a scenario in which the model tries to fit the training data very closely but fails to fit the testing data. Overfitting occurs when the model learns each and every detail in the training data and the noise in the training data. The problem which occurs is that we try to pass the new data to the model to predict it gives a negative result. Overfitting also occurs if the model is too complex.

Q16. How to deal with overfitting in Linear Regression?

ANS:

- Training the model with more data.
- Cross-Validation.
- Regularization
- Data Augmentation.
- Feature Selection.
- Reducing the model complexity.

Q17. What is Underfitting?

ANS: Underfitting is a scenario in which the model is not able to fit the training data and the results of the testing data is also poor. Underfitting occurs when the model is not complex enough to perfectly fit the training data.

Q18. How to deal with underfitting in Linear Regression?

ANS:

- Increasing the size and the number of features in the machine learning model.
- Increasing the complexity of the model.
- Get more training data.

Q19. What is regularisation?

ANS: Regularization is a technique which is used to solve the problem of overfitting in Linear Regression. Regularization technique is used to reduce the magnitude of the features by keeping the same number of features. Regularization works by adding a penalty term to the complex model.

$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \ldots\ldots\ldots + \beta_x X_x + \varepsilon$

In the above equation Y is the value to be predicted.

$X_1, X_2, X_3, \ldots\ldots, X_n$ are the independent Variables or the Features of Y

$\beta_0, \beta_1, \beta_3, \ldots\ldots, \beta_n$ are the weight or magnitude.

After this the cost function is optimized by adding the penalty term and loss function is added to model so that our model can accurately predict the value of the Y.

Q20. What are the types of Regularization Techniques?

ANS: There are two types of Regularization Techniques.

- Ridge Regression also known as L2 regularization.
- Lasso Regression also known as L1 regularization.

Q21. What is Ridge Regression?

ANS: Ridge Regression is a regularization technique to overcome the overfitting problem in this technique a small amount to bias is added to get a better prediction. Ridge regression is used to reduce the complexity of the

machine learning model. It is also known as L2 regularization. The cost function is changed by adding the penalty term. The penalty term helps in reducing the complexity.

Q22. What is Lasso Regression?

ANS: Lasso stands for Least Absolute Shrinkage and Selection Operator. It is a regularization technique that helps in reducing the complexity of the machine learning model and by doing so it also helps to solve the problem of overfitting. Lasso is much similar to the ridge regression. Lasso Regression uses shrinkage method. In this technique some of the independent feature are ignored during model evaluation. It can help in reducing the overfitted and it can also help in model selection.

Q23. What is Feature Selection?

ANS: Feature selection is technique which helps us choose only that feature or variable which significantly helps in contributing to the accuracy of the model. The feature which we select using feature selection technique will highly contribute to the performance of the model. In feature selection technique most of the insignificant variable or the variable which are not contributing to the accuracy of the model are eliminated which also helps in the computation cost of the model and in model performance.

Q24. Why Feature Selection is important?

ANS: Feature Selection is important because:

- Improves Accuracy
- Reduces Overfitting
- Reduces training time

Q25. What are the type of feature selection?

ANS: The type of feature selection are:

- Feature Importance
- Univariate Selection
- Correlation Matrix with Heatmap

- Manual Feature Selection.

Q26. What are the ways of improving the accuracy of a Linear Regression model?

ANS: The ways of improving Linear Regression model are:

- Outliers detection and treatment: It is very important to treat the outliers. Replacing the variables based on mean, median and mode can be very useful.
- Feature Engineering: It is a process of extracting important features from the raw data based on domain knowledge.
- Converting the categorical variable to numerical using one hot encoding method.
- Treating the missing values based on their nature like if they are categorical then imputing the missing values by mode and if they are numerical treating the missing values by mean or median.

Q27. What are the assumptions of Linear Regression?

ANS: The assumptions of linear regression are as follow:

Assumption 1: There should be no outliers.

Assumption 2: Assumption of Linearity.

Assumption 3: Assumption of Normality.

Assumption 4: Assumption of Multicollinearity.

Assumption 5: Assumption of Independence.

Q28. When to drop an outlier and when not to drop an outlier?

ANS.

- Drop an outlier: We can drop an outlier when we know that it is wrongly entered that is a data entry error. For example if we have an outlier in the age column where age is 150 which is far from the normal range and which is impossible so in that case we can remove or eliminate that is outlier. We can drop an outlier if we are having a lot of data and a very small sample of data can be dropped.

- Do not drop an outlier: When the data is critical we should not drop an outlier if we do so the results may change it will affect the accuracy of the model. We should not drop an outlier when there are lot of outlier may be something interesting will be there in the data.

Q29. What is the assumption of Linearity?

ANS. Assumption of Linearity says that there should me a linear relationship between the independent and the dependent variable.

Q30. What is the assumption of Normality?

ANS. Assumption of normality says that for any fixed value of X independent variable the Y dependent variable should be normally distributed.

If the dependent variable is not normally distributed then we have to apply Log transformation (Natural log) to get it normally distributed.

SYNTAX for Transformation:

```
Y_log = np.log(Y)

sns.distplot(Y_log,hist = True)
```

Q31. What is the assumption of multicollinearity?

ANS. Assumption of multicollinearity says that there should be no multicollinearity which means that the independent variable should not be highly correlated with each other.

Q32. What is Multicollinearity?

ANS: Multicollinearity is one of the assumptions of Linear regression. Multicollinearity is a scenario in which the predictor variables or the independent variables are somehow highly correlated with each other.

Q33. What are the functions used to check multicollinearity in python?

ANS: The function used to check multicollinearity are:

- corr() function
- sns.heatmap() function
- variance_inflation_factor()

Q34. What is corr() function?

ANS: corr() is a very useful function to check the multicollinearity in the dataset. corr() gives the correlation of the independent variable. corr() returns a correlation matrix. The values in the correlation matrix are known as correlation coffecients.

SYNTAX:

```
corr_df = X.corr(method = 'pearson')
print(corr_df)
print()
```

Q35. What is heatmap?

ANS: Heatmap helps us to plot a rectangular data visualization table in which we have the correlation values of the independent variables. With the help of the heatmap, we can discover which variables are highly correlated with each other. If the value is closer to -1 it means a strong negative correlation and if the value is closer to 1 it means a strong negative correlation in any of these conditions are met they are highly correlated.

SYNTAX:

```
a = sns.heatmap(corr_df,vmax = 1.0, vmin = -1.0, annot = True)
b, t = a.get_ylim()
a.set_ylim(b+0.5, t-0.5)
```

Q36. What is vif?

ANS: The Variance inflation factor is used to identify the correlation between the independent variables. If the vif value is 1 it means there is no correlation, if vif is between 1 to 5 there is a correlation and if the vif value is greater than 5 then it is highly correlated and we can eliminate that variable.

Example:

Calculating vif using variance_inflation_factor() function: SYNTAX:

from statsmodels.stats.outliers_influence import variance_inflation_factors as vif

```
vif_df  = pd.DataFrame()
vif_df["features"] = X.columns
vif_df["VIF Factor"] = [vif(X.values, i) for i in range(X.shape[1])]
vif_df.round(2)
```

	features	VIF Factor
0	TV	3.39
1	radio	3.63
2	newspaper	5.44

The vif score for newspaper is greater then 5 which is not acceptable so the newspaper is a problematic variable. So we can remove the newspaper variable because it creating problem.

Q37. What are the effects of Multicollinearity on Linear Regression?

ANS: Multicollinearity is a scenario in which the predictor variables or the independent variables are somehow highly correlated with each other. Multicollinearity is problematic because it is against the assumption of linear regression. If you are building the model only for the prediction purpose then the multicollinearity may not cause any problem. But in case if you want to check the regression coefficient how the independent variable affects the dependent variable in that case multicollinearity can be problematic.

Q38. How to deal with the problem of multicollinearity?

ANS: To deal with the problem of multicollinearity:

- We can remove some of the independent variables which are highly related with each other with the help of heatmap.

- We can use the variance inflation factor to identify the correlation between the independent variables. If the vif value is 1 it means there is no correlation, if vif is between 1 to 5 there is a correlation and if the vif value is greater than 5 then it is highly correlated and we can eliminate that variable.
- We can use Principle Component Analysis to eliminate the unwanted or irrelevant variables.

Q39. What is R square?

ANS. R square is also described as the coefficient of determination. R square is used to determine the strength of correlation between the independent and the dependent variable. In simple terms R square lets us know how accurate our regression model is when compared to average. R square ranges between 0 to 1 higher the number the better is the accuracy or prediction of the model. If our R square is greater than 70% which is 0.7 indicated a good fit model.

Q40. What is Adjusted R square?

ANS. The Adjusted R square is a modified version of the R square. Adding more independent variables will result in an increased value of R square irrespective of whether the new independent variable is significant or not. But in the case of Adjusted R square if the new independent variable added is insignificant the adjusted r square has the capability to decrease therefore resulting in a better, more reliable, and accurate evaluation.

Q41. Difference between R^2 and Adjusted R^2?

ANS. Adding more independent variables will result in an increased value of R. This is the disadvantage of R square adding more independent variable irrespective of whether the new independent variable is significant or not the value of R square increases. But in the case of Adjusted R square if the new independent variable added is insignificant the adjusted r square has the capability to decrease therefore resulting in a better, more reliable, and accurate evaluation.

Q42. What is RMSE?

ANS. RMSE stands for ROOT MEAN SQUARE ERROR is a standard way to measure the error rate of the model. RMSE is a standard deviation of residuals or errors. Residuals or Errors are a measure of how far the data points are from the regression line. RMSE is a value that should be closer to 0.

Q43. Difference between Correlation and Regression?

ANS. Correlation measures the relative strength of linear relationship between two variables Independent(X) and Dependent Variable(Y). Correlation ranges from -1 to 1. The closer to -1, the stronger is the negative linear relationship. The closer to 1, the stronger is the positive linear relationship. The closer to 0, the weaker is the positive linear relationship.

Regression is nothing but to describe the relationship between two variables and how change in one variable affects the other variable. Regression is described by the best fit line. It is used for model building and prediction.

Q44. What do you mean by OLS?

ANS. OLS stands for Ordinary Least Squares. OLS is most commonly known as Linear Regression. OLS is a method which is used to discover the relation between the dependent variable and one or more independent variable.

Q45. What is normal Distribution?

ANS. Normal distribution is a bell shaped curve which shows the probability distribution of the data.

Q46. What are the properties if normal distribution?

ANS. Properties of normal distribution are:

- The mean=median=mode.
- The area under the curve is 1.
- Half of the value are to the left from the centre and half of the value are to the right from the centre.

Q47. What do you mean by Box-Cox Transformation?

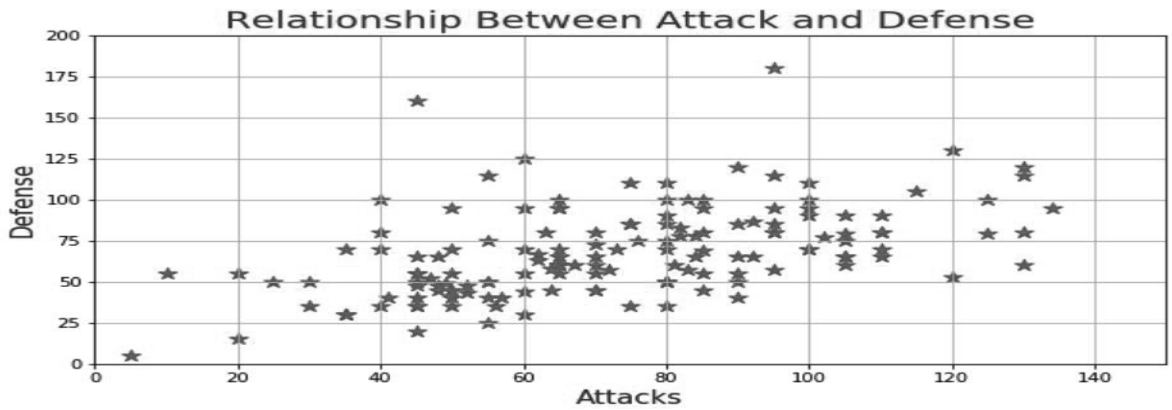
ANS. There are various assumptions that the data should be normally distributed but if the data is not normally distributed or when the data is right skewed or left skewed then the Box-Cox Transformation is used to transform the data into a normal distribution.

Q48. What is a Scatter Plot?

ANS. Scatter plot is a type of graph that is used to represent the relationship between two variables with the help of dots. The dots in the scatter plot represents the individual data points. Scatter plot is also used to find the pattern if we try to fit a regression line in the scatter plot it can show us that the relation between the variables is positive linear relation, negative linear relation, or there is no relation.

SYNTAX:

```
plt.figure(figsize=(10,5))
plt.scatter(data=df, x='Attack',y='Defense',s=100, c='green', marker='*')
#change axes ranges
plt.xlim(0,150)
plt.ylim(0,200)
#add title
plt.title('Relationship Between Attack and Defense',size=20,c='purple')
#add x and y labels
plt.xlabel('Attacks',size=15,c='blue')
plt.ylabel('Defense',size=15,c='blue')
plt.grid(True)
plt.savefig("test.jpg")
plt.show()
```



Q49. Why do we split our data into training and testing?

ANS. Splitting the data into a training data and testing data is a very important step for model evaluation. We divide our data into two sets which are training and testing most of the data is used for the training purpose and a small sample of the whole data is used for testing.

The whole process works in the following manners:

- First we train the model on the training data which we have created by splitting the original data set.
- After training the model we do our prediction on the testing data and store the predicted value in variable.
- The last step is to compare the predicted data and the testing data that we already know this is how we try to evaluate the model and calculate the accuracy of the model.

SYNTAX:

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size =0.2,
random_state = 10)
```

Basically we divide our data in X_train, Y_train which are the training data.

X_test,Y_test are the testing data.

We only have to pass the data which is X and Y and the size of the test data in our case we have pass 0.2 which is nothing but 20% of the whole data.

That means our training data is 0.8 which is 80 % of the whole data.

Q50. Which graphs are used to observed the data before building the model?

ANS. Before training the model we must understand the data like what kind of data do we have so for understanding the data very well we must look for the trends in the data, distribution, skewness and the relation of the variables. To perform all such operations we can use various graphs like histograms, scatter plots, box plots, line chart etc. We must also look for what kind of relationship do we have among the variables for checking relationship we can always use scatter plot.

Q51. What are the benefits of Linear Regression?

ANS. Benefits of Linear Regression are:

- Forecasting and prediction: Linear regression is used to forecast trends and make accurate prediction.
- Beneficial for businesses: If a business owner want to know where he should invest more so that he get maximum profit. For getting insights linear regression is very useful it can shows us the relation of the variables. By doing so one can get overall all idea that which variable is significantly contributing to the sales of the company

Q52. Steps for performing Linear Regression in Python.

ANS: Performing Linear Regression in Python:

1. Create a dataframe properly --> pd.read_csv(), pd.read_excel()
2. Assumption 1-There should be no outliers in the data --> pd.boxplot()
3. Assumption 2-Assumption of Linearity --> pairplot()
4. Create X and Y
5. Assumption 3-Assumption of Normality of Y --> distplot(), log()
6. Handle the skewness in the X --> skew(), log1p()
7. Assumption no 4-There should be no multicollinearity -->corr(), heatmap(), vif()
8. Splitting the data --> train_test_split(), manual splitting
9. Build the model:
   a.Create the model object --> obj=LinearRegression()
   b. Train the model --> obj.fit(X_train,Y_train)
   c. Predict using the model --> Y_pred=obj.predict(X_test)

10. Evaluating the model:
    a. Rsquare
    b. Adjusted Rsquare
    c. RMSE (ROOT MEAN SQUARE ERROR)
11. Tuning the model --> Manual feature selection, pvalues, Ridge Regression, Lasso Regression, Applying Feature engineering, PCA principle component analysis.

Implementation of Linear Regression with Python:

Description of the dataset: The data set which we are using for the linear regression is a Advertising dataset. In this dataset out independent variables(X) are TV, radio and newspaper. The dependent variable(Y) is sales. We have to check that what is the impact of different types of advertisement on sales. So that the one can invest only in those variables which is more significantly contributing to the data set.

- Import the required library:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

- Import the Dataset as dataframe:
  df = pd.read_csv('Advertising (1).csv', index_col=0,header=0)

- Checking the dataset using the head() function:
  df.head()

	TV	radio	newspaper	sales
1	230.1	37.8	69.2	22.1
2	44.5	39.3	45.1	10.4
3	17.2	45.9	69.3	9.3
4	151.5	41.3	58.5	18.5
5	180.8	10.8	58.4	12.9

- Checking the size of the dataset using shape function:
  df.shape

```
df.shape
```

```
(200, 4)
```

- Checking for missing values using isnull() function:
  print(df.isnull().sum())

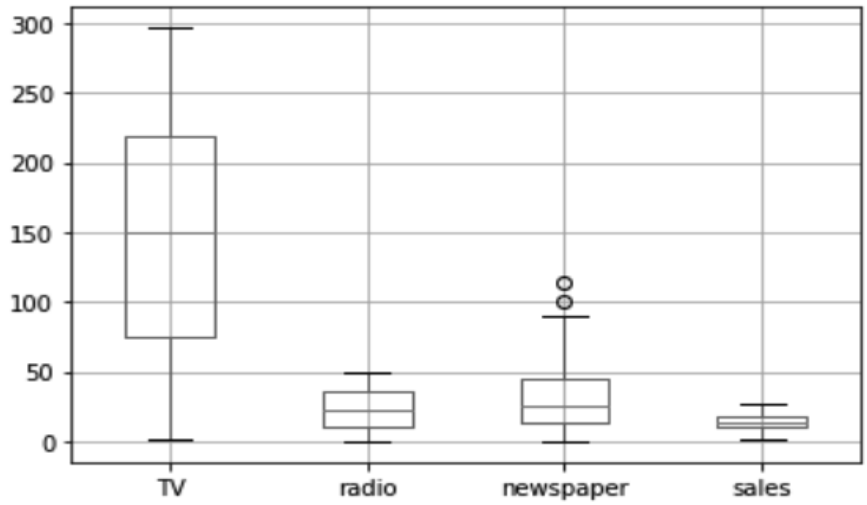
```
print(df.isnull().sum())
```

```
TV           0
radio        0
newspaper    0
sales        0
dtype: int64
```

- Checking for the outliers using boxplot():
  df.boxplot()

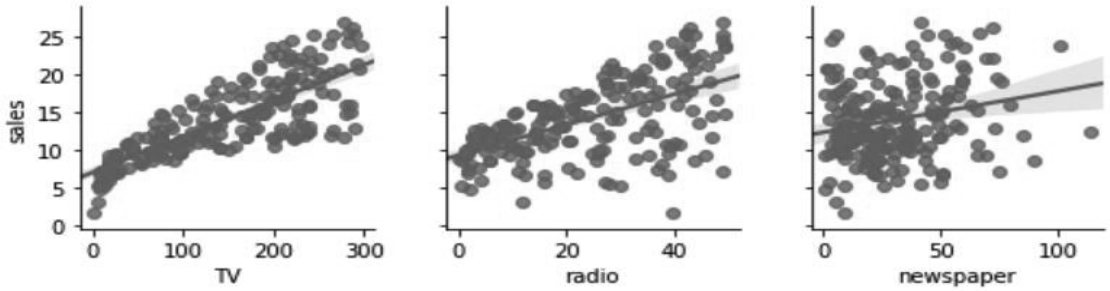
```
df.boxplot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ca98da9a88>
```



In the above graph we can clearly see that in the variable newspaper there are some outliers.

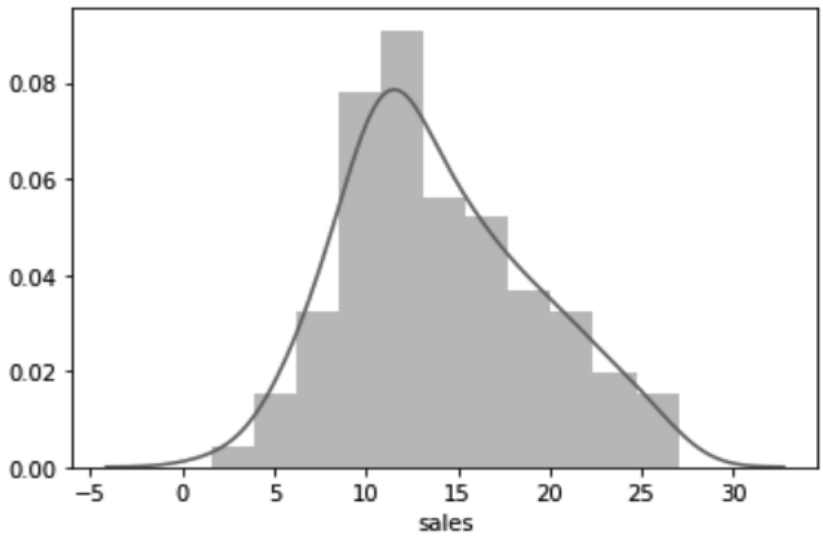
- Assumption of linearity using sns pairplot():
  sns.pairplot(df,x_vars = ['TV','radio','newspaper'], y_vars = 'sales', kind = 'reg')



In the above graph we can clearly see that the variable TV, radio have strong linear relationship but the newspaper variable have a weak linear relationship with the dependent variable sales.

- Creating variable X and Y where X is the independent variables and Y is the dependent variable:
  X = df[['TV','radio','newspaper']]
  Y = df['sales']

- Assumption of normality using sns.distplot():
  sns.distplot(Y,hist = True)



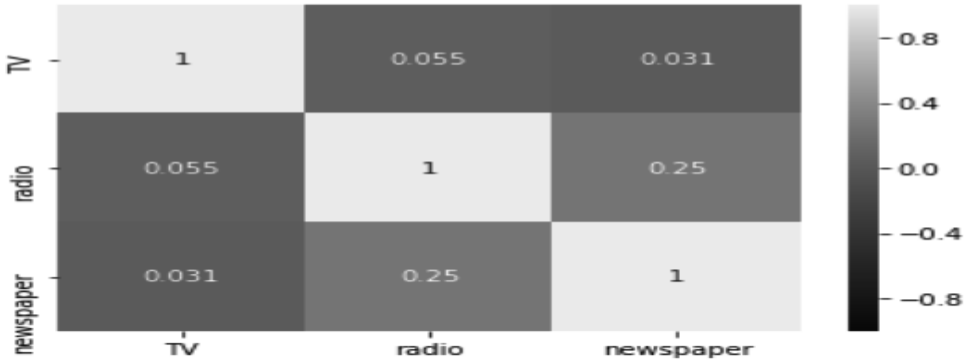
  We can clearly see that the dependent variable(Y) sales is normally distributed.
  Suppose the dependent variable(Y) sales was not normally distributed in that case we can use log transformation (Natural log) by applying log to the Y but in our case it is not needed.
  Y_log = np.log(Y)

- Assumption of multi-collinearity using sns.heatmap or corr function():
  corr_df = X.corr(method = 'pearson')
  print(corr_df)
  print()

  a = sns.heatmap(corr_df,vmax = 1.0, vmin = -1.0, annot = True)
  b, t = a.get_ylim()
  a.set_ylim(b+0.5, t-0.5)

```
                TV     radio  newspaper
TV        1.000000  0.054809   0.030951
radio     0.054809  1.000000   0.251909
newspaper 0.030951  0.251909   1.000000
```

```
(3.0, 0.0)
```



- Calculating vif using variance_inflation_factor() function:
  from statsmodels.stats.outliers_influence import variance_inflation_factor as vif
  vif_df = pd.DataFrame()
  vif_df["features"] = X.columns
  vif_df["VIF Factor"] = [vif(X.values, i) for i in range(X.shape[1])]
  vif_df.round(2)

	features	VIF Factor
0	TV	3.39
1	radio	3.63
2	newspaper	5.44

  The vif score for newspaper is greater then 5 which is not acceptable so the newspaper is a problematic variable.

- Dropping the newspaper variable because the vif score is greater then 5 ans it is a problematic variable:
  X.drop("newspaper", axis=1, inplace=True)
  X.head()

	TV	radio
1	230.1	37.8
2	44.5	39.3
3	17.2	45.9
4	151.5	41.3
5	180.8	10.8

- Splitting the data into training and testing:
  from sklearn.model_selection import train_test_split
  X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2,
  random_state = 10)

- Importing the linear regression model from sklearn.linear_model:
  from sklearn.linear_model import LinearRegression

- Creating a model object lm:
  lm = LinearRegression()

- Training the model object on the train data:
  lm.fit(X_train,Y_train)

- Printing the intercept and coefficients:
  print(lm.intercept_)
  print(lm.coef_)

```
#print the intercepts and coefficients
print(lm.intercept_)
print(lm.coef_)
```

```
3.21702610851297
[0.04372065 0.19242472]
```

- Prediction using the model we created:
  Y_pred = lm.predict(X_test)
  print(Y_pred)
  print(np.exp(Y_pred))

- Evaluating the model:

```
from sklearn.metrics import r2_score,mean_squared_error
import numpy as np

r2=r2_score(Y_test,Y_pred)
print("R square:",r2)

rmse=np.sqrt(mean_squared_error(Y_test,Y_pred))
print("RMSE:",rmse)

adjusted_r_squared = 1 - (1-r2)*(len(Y)-1)/(len(Y)-X.shape[1]-1)
print("Adjusted R Square:",adjusted_r_squared)
```

```
R square: 0.8354496662944217
RMSE: 2.5878817077378105
Adjusted R Square: 0.8337791045309133
```

For evaluation matrix we have imported r2_score for calculating R square, mean_squared_error for calculating RMSE and for Adjusted R square we have applied the formula.
Evaluating the accuracy of the model:
R square/Adjusted R square: R square is a value which is in between 0 to 1 higher the number the better is the accuracy or prediction of the model. If our R square is greater then 70% which is 0.7 indicated a good fit model. In our case it is 0.83 which in 83% which indicates a good fit model. There is not much difference between R square and Adjusted R square. So our Adjusted R square is also 0.83 which indicates a good fit model.

MSE/RMSE: RMSE stands for ROOT MEAN SQUARE ERROR is a standard way to measure the error rate of the model. RMSE is a standard deviation of residuals or errors. Residuals or Errors are a measure of how far the data points are from the regression line. RMSE is a value that should be closer to 0.

- Ridge Regression:
  Creating a Ridge regression model.

- Importing the Ridge Regression model from sklearn.linear_model:
  from sklearn.linear_model import Ridge

- Creating a model object lm:
  lm = Ridge()

- Training the model object on the train data:
  lm.fit(X_train,Y_train)

- Printing the intercept and coefficients:
  print(lm.intercept_)
  print(lm.coef_)

```
3.3522471725966003
[ 0.04374234  0.19302603 -0.04853131]
```

- Prediction using the Ridge Regression model we created:
  Y_pred = lm.predict(X_test)
  print(Y_pred)
  print(np.exp(Y_pred))

- Evaluating the model:
  from sklearn.metrics import r2_score,mean_squared_error
  import numpy as np

  r2=r2_score(Y_test,Y_pred)
  print("R square:",r2)

  rmse=np.sqrt(mean_squared_error(Y_test,Y_pred))
  print("RMSE:",rmse)

  adjusted_r_squared = 1 - (1-r2)*(len(Y)-1)/(len(Y)-X.shape[1]-1)
  print("Adjusted R Square:",adjusted_r_squared)

```
R Square: 0.8348082644975935
RMSE: 2.5929204626839977
Adjusted R Square 0.8322798195664342
```

- Lasso Regression Model:
  Creating a Lasso regression model.

- Importing the Lasso Regression model from sklearn.linear_model:
  from sklearn.linear_model import Lasso

- Creating a model object lm:
  lm = Lasso()

- Training the model object on the train data:
  lm.fit(X_train,Y_train)

- Printing the intercept and coefficients:
  print(lm.intercept_)
  print(lm.coef_)

  ```
  3.336794058220317
  [ 0.04362374  0.18766033 -0.        ]
  ```

  As we have seen while checking the outliers we have found some outliers in the newspaper variable and the vif was also greater then 5 for newspaper variable which was not acceptable. Which means that newspaper is an insignificant variable so lasso regression forcefully makes is zero(0).

- Prediction using the Ridge Regression model we created:
  Y_pred = lm.predict(X_test)
  print(Y_pred)
  print(np.exp(Y_pred))

- Evaluating the model:
  from sklearn.metrics import r2_score,mean_squared_error
  import numpy as np

  r2=r2_score(Y_test,Y_pred)

```
print("R square:",r2)

rmse=np.sqrt(mean_squared_error(Y_test,Y_pred))
print("RMSE:",rmse)

adjusted_r_squared = 1 - (1-r2)*(len(Y)-1)/(len(Y)-X.shape[1]-1)
print("Adjusted R Square:",adjusted_r_squared)
```

```
R Square: 0.8360506658527163
RMSE: 2.5831514271094234
Adjusted R Square 0.8335412372688292
```