MACHINE LEARNING LAB RECORD

**Task 1:**

**How to create a dataframe using the pandas library.**

**AIM:**

Creating a dataframe in Pandas from csv files.

**Dataset Description and Methods:**

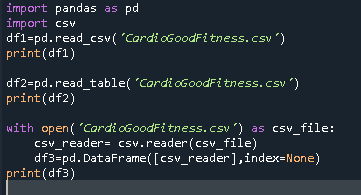
1. CardioGoodFitness.csv is a dataset of the size 180 rows x 9 columns
2. Bank.csv is a dataset of size 11162 rows x 17 columns
3. Housing.csv is a dataset of size 20640 rows x 10 columns
4. Taxi-trip-duration.csv is a dataset of size 1458644 rows x 11 columns
5. Auto-mpg is a dataset of size 398 rows x 9 columns

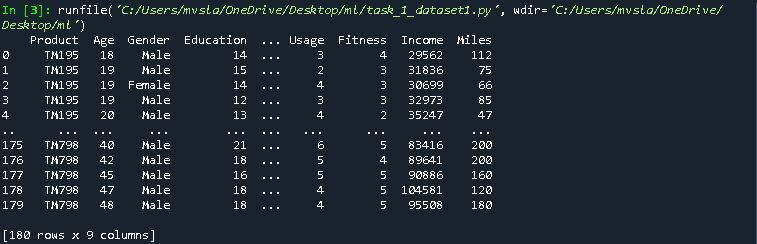
Methods to create dataframes from csv files are:

1. pd.read\_csv()
2. pd.read\_table()
3. Using csv module

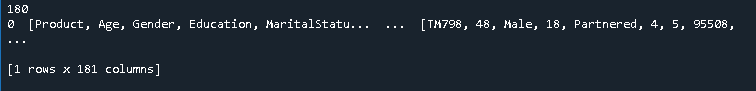
**Source Code:**

**For CardioGoodFitness.csv dataset**

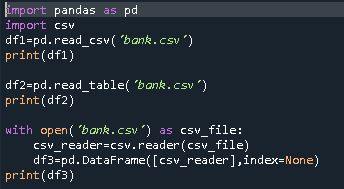
****

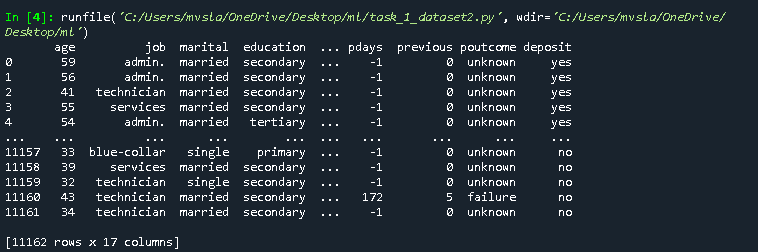
****

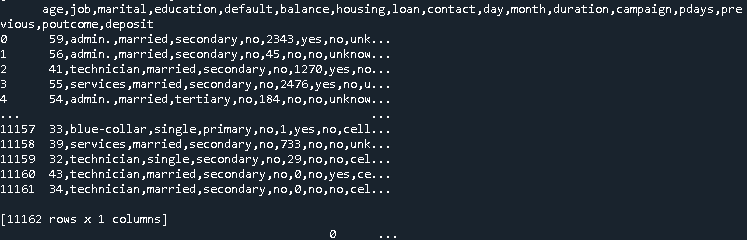
****

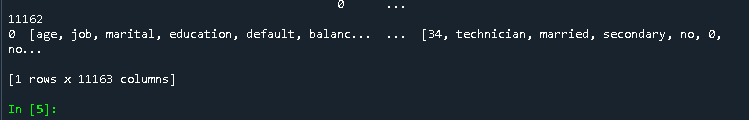
****

**For bank.csv dataset:**

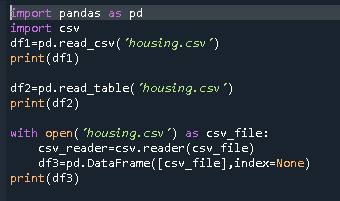
****

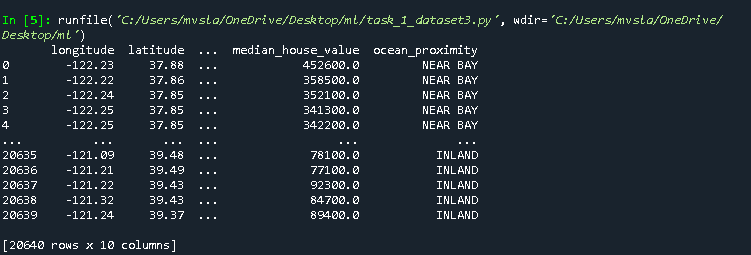
****

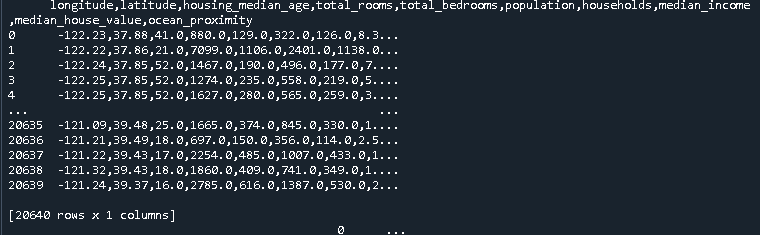
****

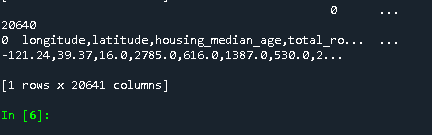
****

**For housing.csv dataset:**

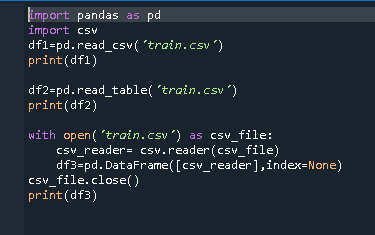
****

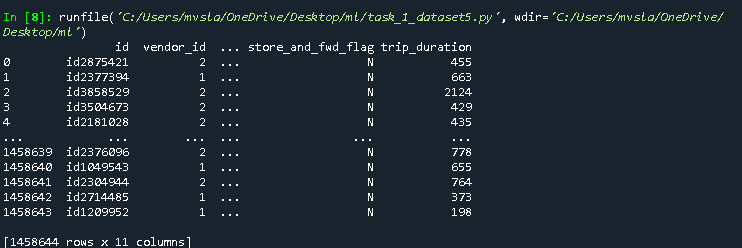
****

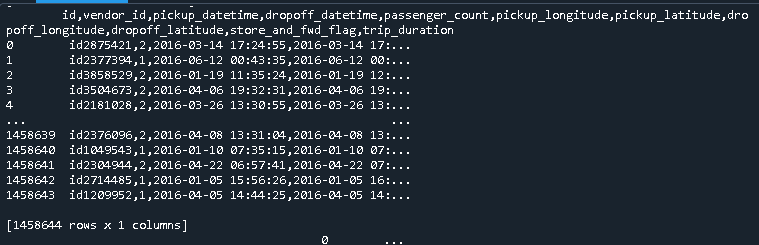
****

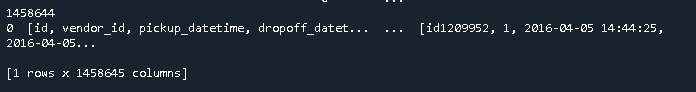
****

**For taxi-trip-duration dataset (training split):**

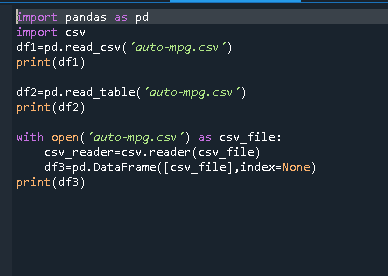
****

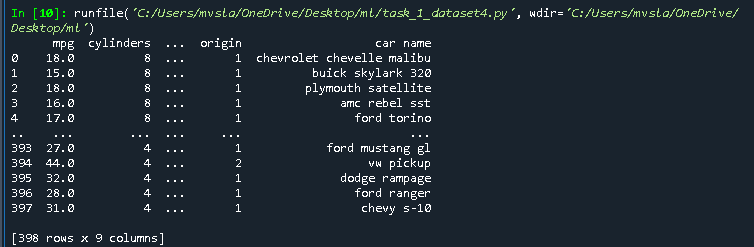
****

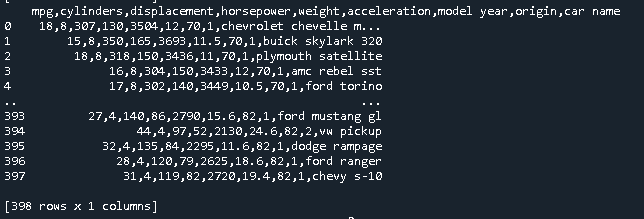
****

****

**For AutoMPG dataset:**

****

****

****

****

**Observation:**

1. Pd.read\_csv() method is giving the dataframe with exact dimensions with proper indexing and the values are separated by spaces, giving the dataframe a matrix format for ease of analysis.
2. Pd.read\_table() method is giving the dataframe with all the values in one column and the values are separated by commas.
3. Csv module and pd.DataFrame() is giving the transpose of the pd.read\_table() and proper indexing is not present.

**Conclusion:**

pd.read\_csv() method is more suitable amongst the other methods because it is having proper indexing and the dataframe produced by this method is easy to understand and interpret.

**Task 2:**

**How to add a new column to a dataframe.**

**AIM:** To add a new column to an existing dataframe

**Methods and dataset description:**

* AutoMPG dataset is a dataset about cars and the dataset is of the size

398 rows x 9 columns.

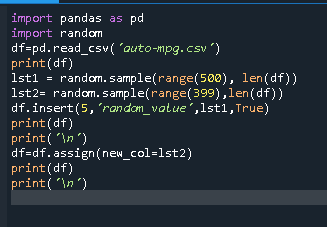
* There are 8 numerical variables and 1 categorical variable in the dataset.

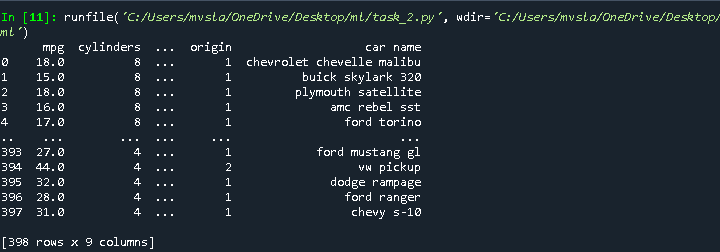
Adding a new column to the dataframe can be done in the following methods.

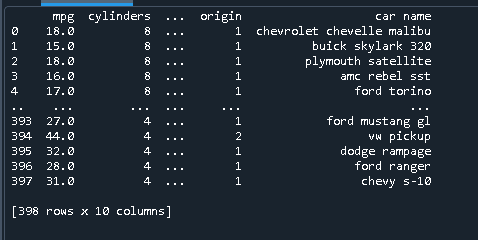
* + - Df.insert()
    - Df.assign()

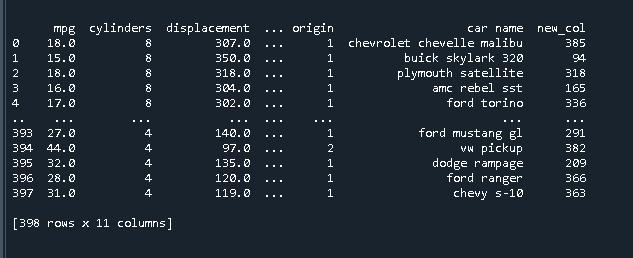
**Source Code:**

**For AutoMPG dataset:**









**Observations:**

* Pd.insert() method inserts a new column in the original dataframe.
* Pd.assign() method creates a new dataframe in the runtime with the new column and that dataframe must be assigned to a new variable.

**Conclusion:**

Both the methods can be used as per the requirement and the are quite easy to use.

**Task 3:**

**How to handle missing values in a dataset**

**AIM:** Handling missing values in a pandas dataframe

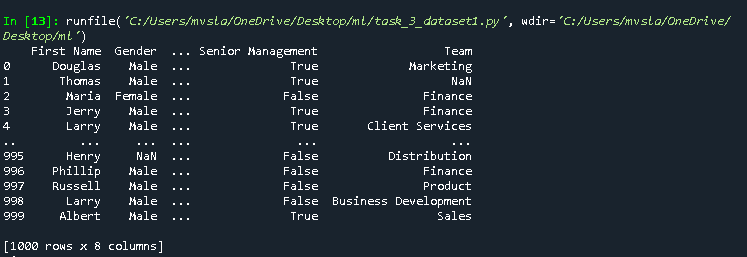
**Methods and Dataset Description:**

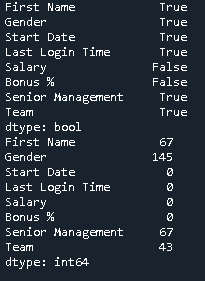
* Missing values can be a problem in the data analysis and machine learing model decision.
* So they must be handled in the proper way possible and it can be done in the following ways.
  + - Df.dropna()
    - Df.fillna()
    - Df.replace()
    - Df.interpolate()
* This task has been done on 2 datasets taken from kaggle:
  + - Employees.csv is a dataset of the size 1000 rows x 8 columns.
    - It has 6 categorical varaibles and 2 numerical variables.
    - AutoMPG.csv is a dataset of the size 398 rows x 9 columns.
    - There are 8 numerical variables and 1 categorical variable in the dataset.

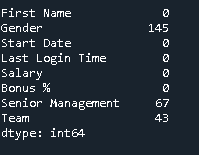
**Source code:**

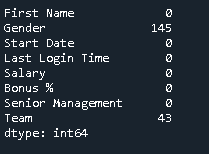
**For employees.csv dataset.**

****

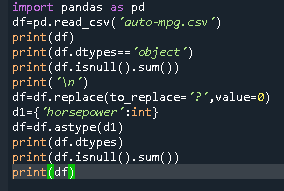
****

** **

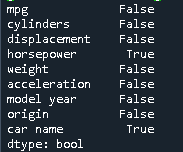
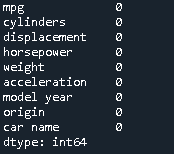
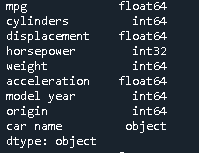
****

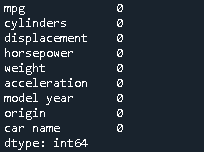
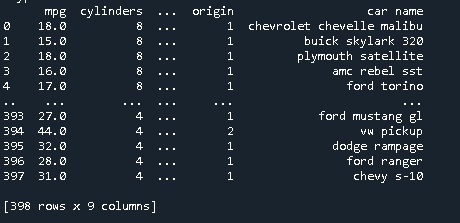
****

**For AutoMPG dataset:**

****

****

**  **

** **

**Observations:**

* df.dropna() is a method which removes the records with null avlues and reduces the size of the dataset.
* df.fillna() is method which can fill the null values as per the method for one column or for many columns.
* Df.replace() method creates a new dataframe and needs a dictionary as the input and can even work with replacing not null but wrong entries in the dataframe.
* Df.interpolate() is not used in this particular task as it needs much data to understand the distribution and fill the null values accordingly.

**Conclusion:**

It is very important to understand the distribution and the type of the dataset to use the appropriate missing data handling mechanism.

**Task 4:**

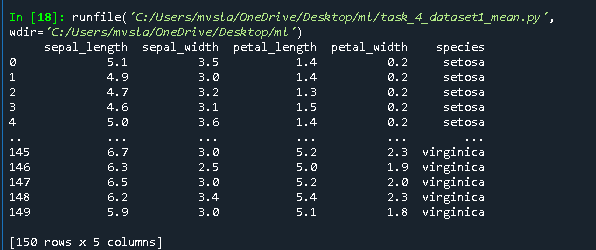
**How to calculate arithmetic mean, median, trimmed mean and weighted mean in a dataframe.**

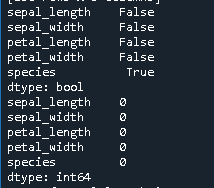
**AIM :** To calculate the arithmetic mean, median, trimmed mean and weighted mean to a dataset.

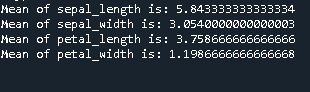
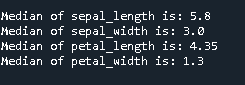
**Methods and dataset description:**

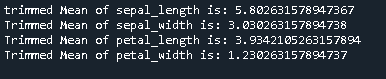
* We used two datasets taken from kaggle to caluculate the given measures of central tendency, which are AutoMPG and iris datsets.
* AutoMPG is a datset about cars which has a size of 398 columns x 9 rows with 8 numerical variables and 1 categorical variable, however we can consider model year and origin as the nominal variables.
* Iris dataset is a datset about three types of flowers with a size of 150 rows x 5 columns, with 4 numerical variables and 1 categorical variable.
* We used numpy, pandas and scipy libraries to complete the given task.

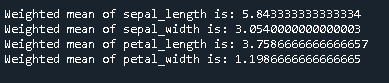
**Source Code: For Iris Dataset: **

****

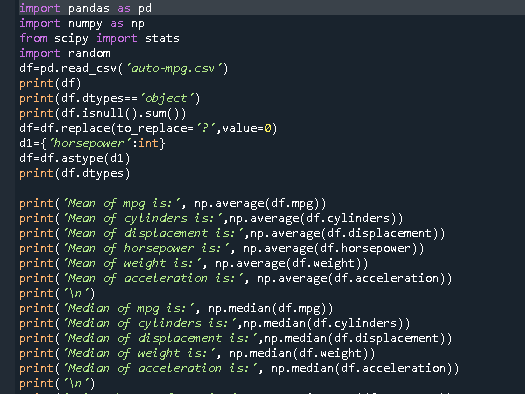
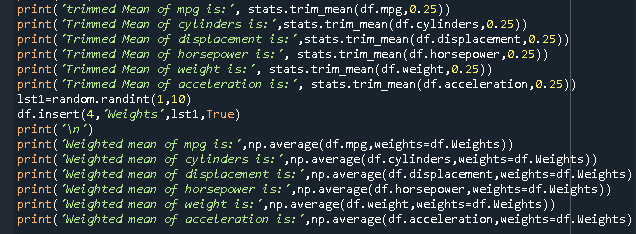
****

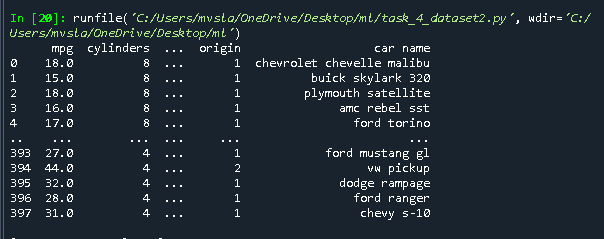
 

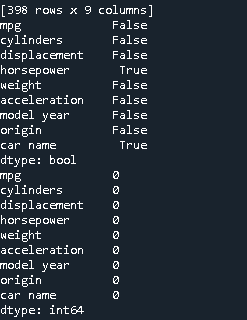
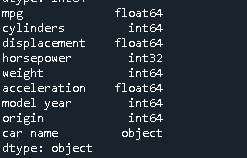


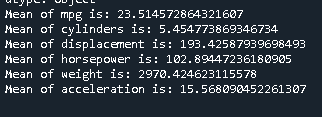
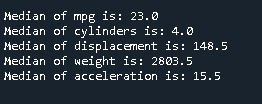


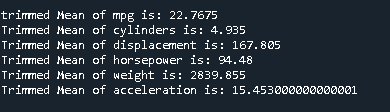
**For AutoMPG dataset:**

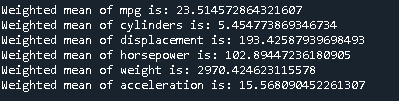
**** 

****

** **

** **

****

****

**Observations:**

* Each measure of central tendency needed a separate library to execute and they are giving different results as per their formulae.
* All these measures can be applied only on numerical data.

**Conclusion:**

According to the requirement and as per the datatype, appropriate measure of central tendency must be applied for a given dataset in a dataframe.

**Task 5:**

**How do you index your dataframe and select the desired data from the dataframe?**

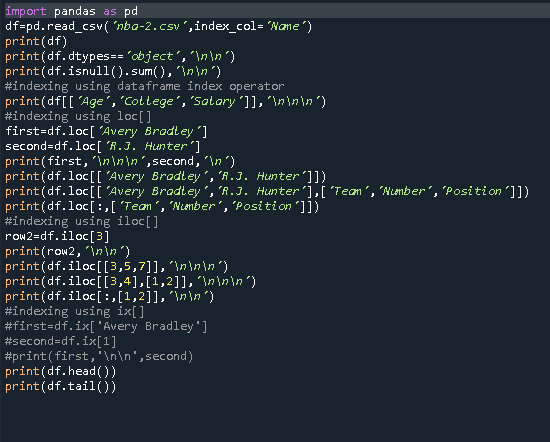
**AIM:** To index and select data from the dataframe.

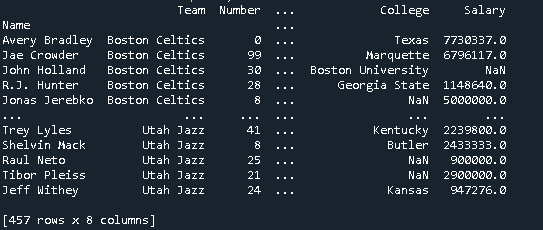
**Methods and dataset description:**

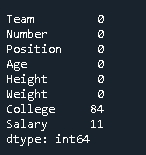
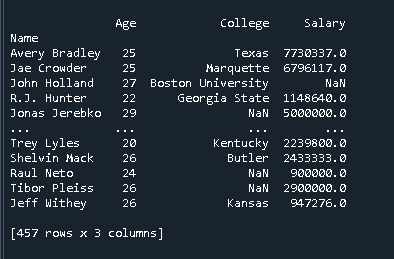
* We are taking two datasets into consideration for applying the indexing and selecting techniques.
* The datasets are: nba-2.csv dataset and iris dataset.
* The nba dataset has 4 numerical variables and 4 categorical variables and it has 84 missing values in college and 11 missing values in salary column and is of the size 457 rows x 8 columns.
* The iris dataset has 150 rows x 5 columns and has no missing values. It has 4 numerical variables and 1 class variable.
* Indexing in a dataframe is nothing but the process of selecting the set of rows required for the analysis and training dataset preparation, especially in the case of supervised learning.
* The indexing can be done in 4 ways:
  + - Index operator []
    - df.loc[]
    - df.iloc[]
    - df.ix[]

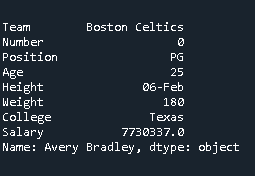
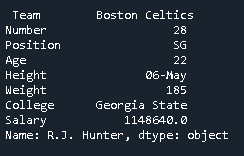
**Source code and output:**

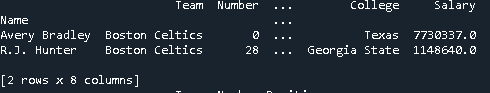
**For nba-dataset:**

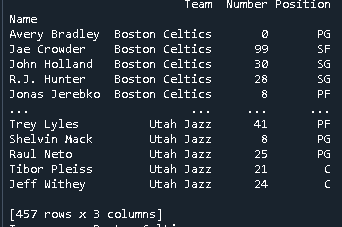
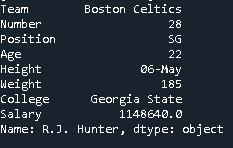
****

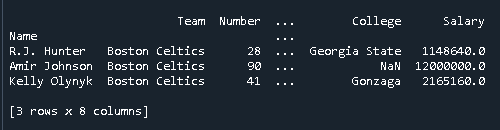
****

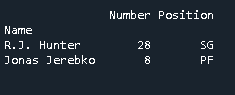
**  **

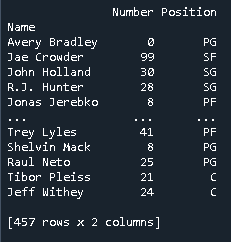
** **

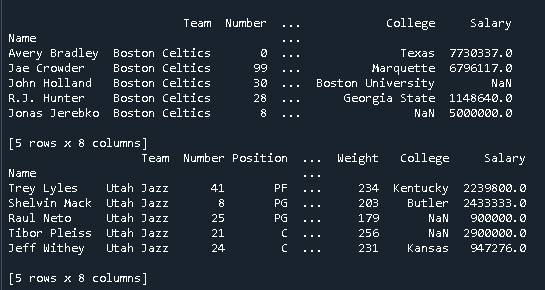
****

** **

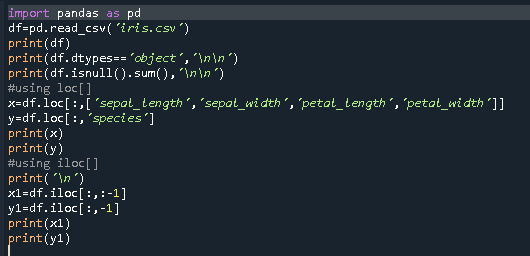
****

****

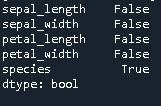
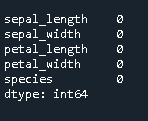
****

****

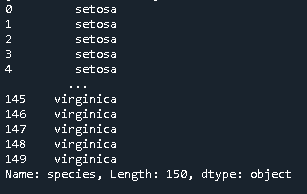
**For Iris dataset:**

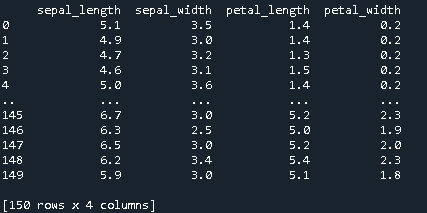
****

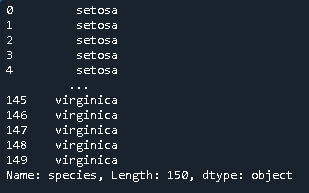
****

** **

****

****

****

****

**Observations:**

* Index operator does the selection of columns based on the column labels, however we can select all the rows not a particular set of row values.
* df.loc[]method works based on column values and we can also provide the range of row values to be selected.
* df.iloc[] method works based on the column indexes and row indexes we can select the dataframes.
* df.ix[] method method can work based on both index values and labels, however this method has been depricated due to inconsistency.

**Conclusion:**

loc[] and iloc[] methods can be used as per the requirement and iloc[] is more preferable as the generalization to training set preparation for model fitting.

**TASK 6: Perform Principal Component Analysis for dimensionality reduction**

**AIM :** To perform principal component analysis and also identify the best number of principal components for a given dataset using the explained variance ratio and scree plot

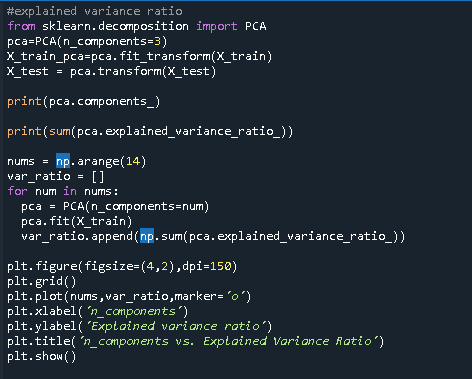
**Methods and dataset description:**

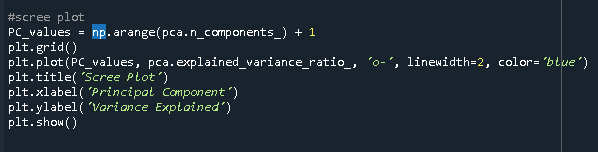
* Principal component analysis the one of the most important dimensionality reduction technique for feature transformation.
* PCA reduces the size of the dataset from the n dimensional space to m dimansional space where m<n, without the loss of data.
* PCA converts the dependent features into independent features by applying the concept of Eigen vectors and Eigen values to the features in a dataset.
* The optimal number of features for the data analysis can be decided using evaluation metrics like ‘Explained Variance Ratio’ (EVR) and visualization metric like scree plot.
* PCA is performed on two datasets : wine dataset and iris dataset.

**Source Code and output:**

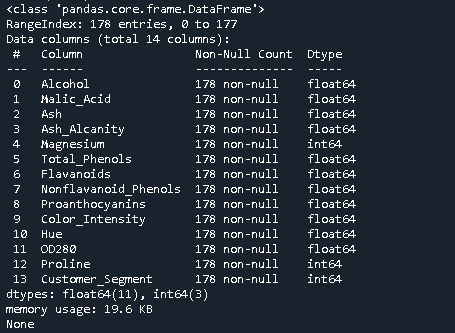
**Source Code for wine dataset:**

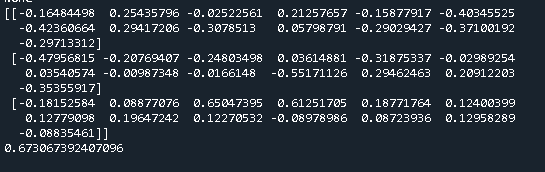
****

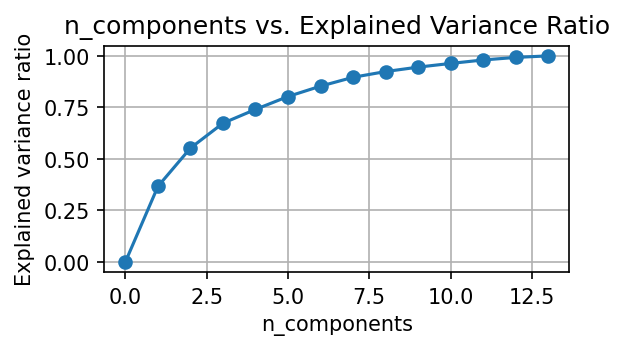
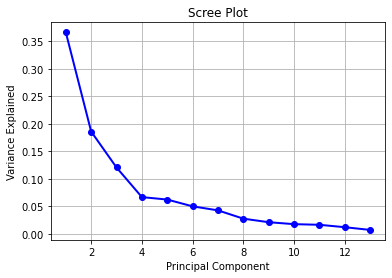
****

****

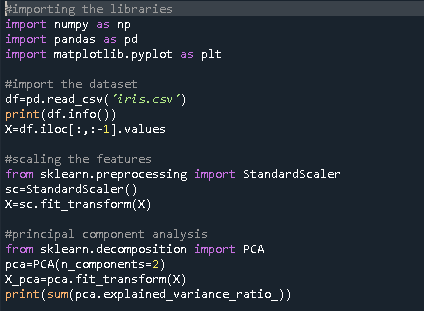
**Output:**

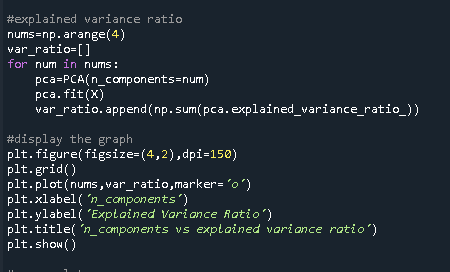
****

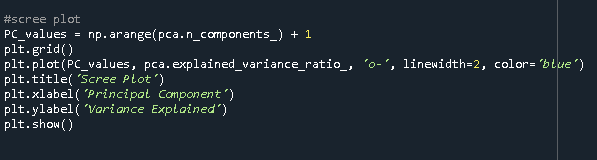
****

**Source Code for iris dataset:**

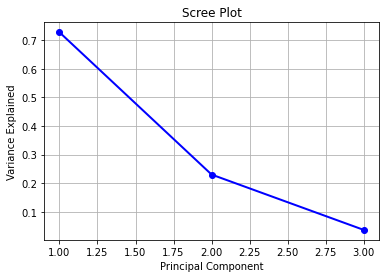
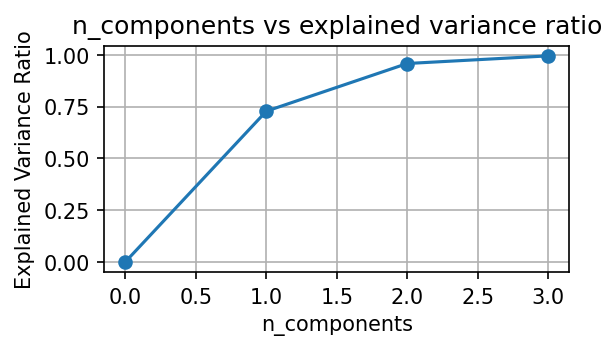
****

****

****

**Output:**

****



**Note:** Source for scree plot is in the link given below

<https://www.statology.org/scree-plot-python/>

**Observations:**

* The optimal principal components for the wine dataset is 4 or 5 can be considered, as per the EVR and scree plot visualizations.
* The optimal principal components for the iris dataset is 2.

**Conclusion:**

We can say that the number of features in the wine dataset are reduced from 13 to 5 and for iris dataset, the number of features can be reduced from 4 to 2 features using Principal Component Analysis (PCA) and the machine learning algorithms can be implemented accordingly.

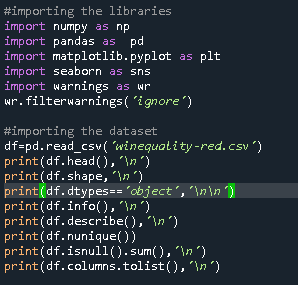
**TASK 7: Performing Exploratory data analysis Using data visualizations and Statistics**

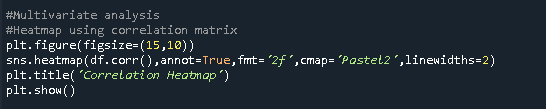
**AIM:** To perform EDA on the given dataset and explore various types of data visualizations for univariate, bivariate and multivariate analysis.

**Methods and dataset description:**

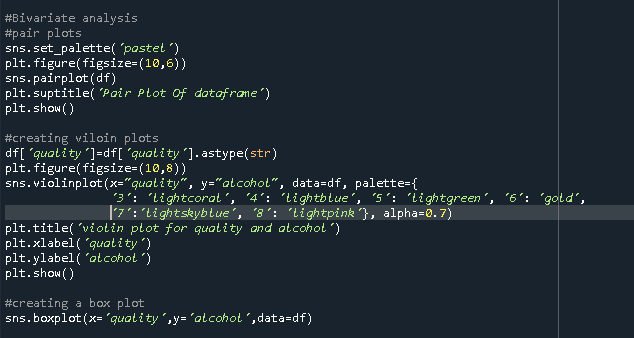
* The red wine quality dataset is used to perform Exploratory Data Analysis.
* It has the dimensions as 1599 x 12 and has no missing values.
* It has 12 attributes as numeric and no categorical attributes.
* Exploratory Data analysis involves the process of exploring the data using statistical and data visulaization techniques to understand and improve the quality of the data.
* EDA also helps in understanding the type of machine learning algorithm is most suitable to the given dataset.

**Source Code:**

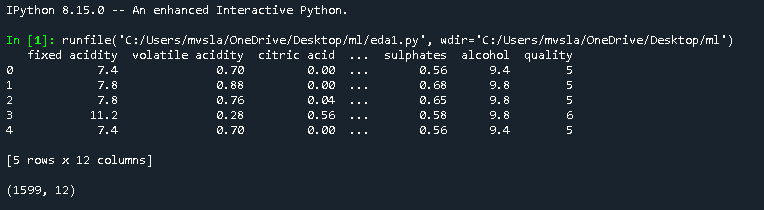
****

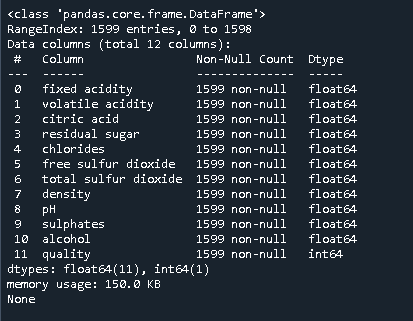
****

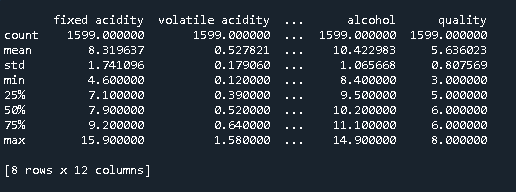
****

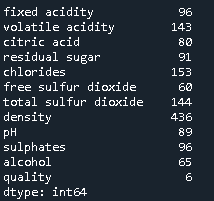
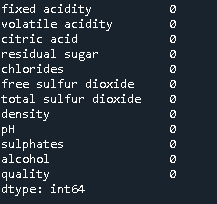
****

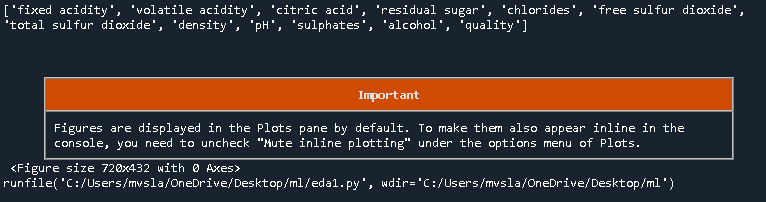
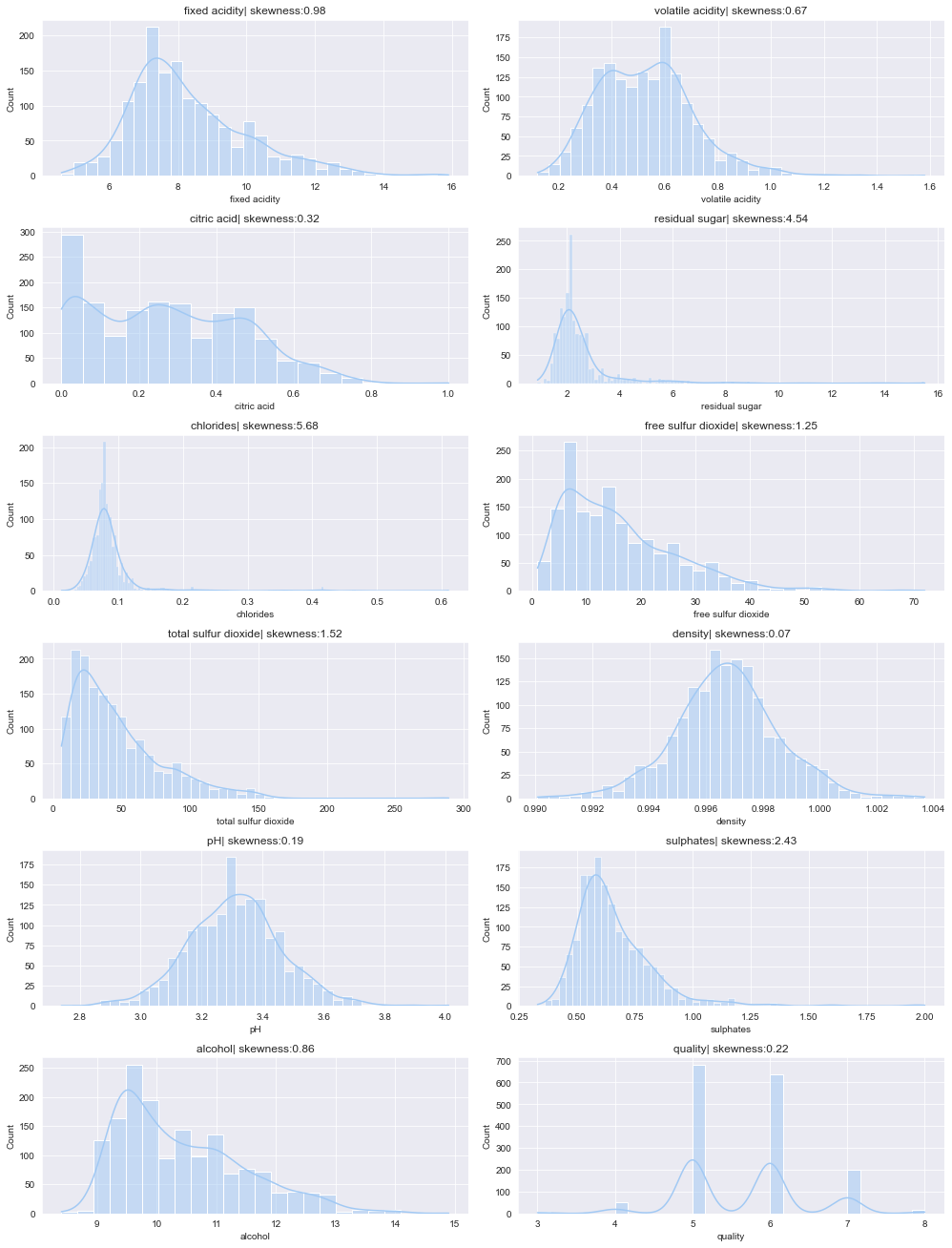
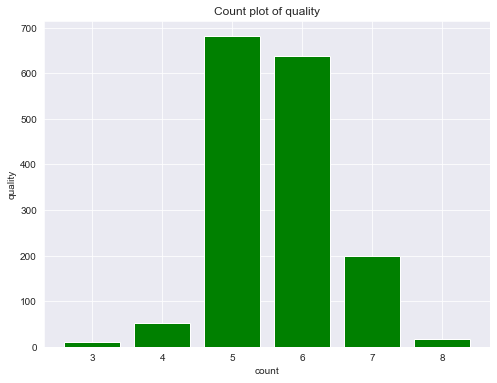
**Output:**

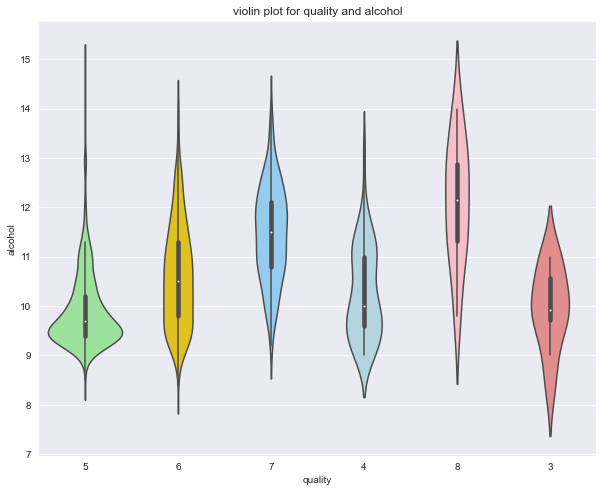
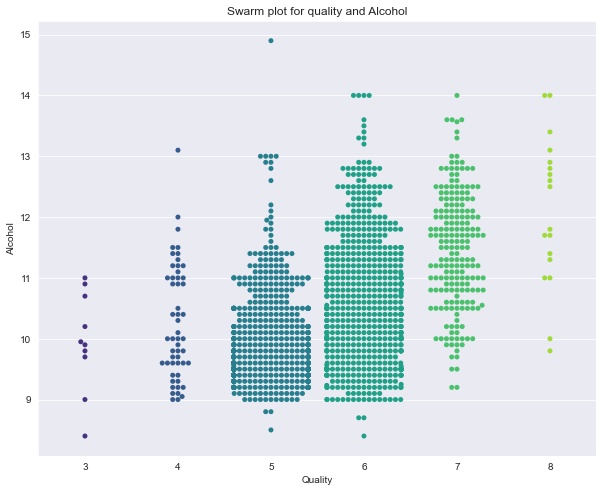
****

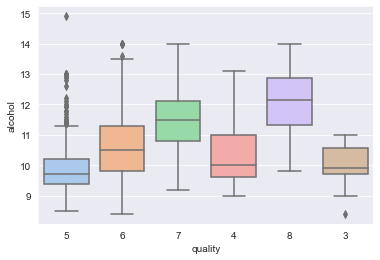
****

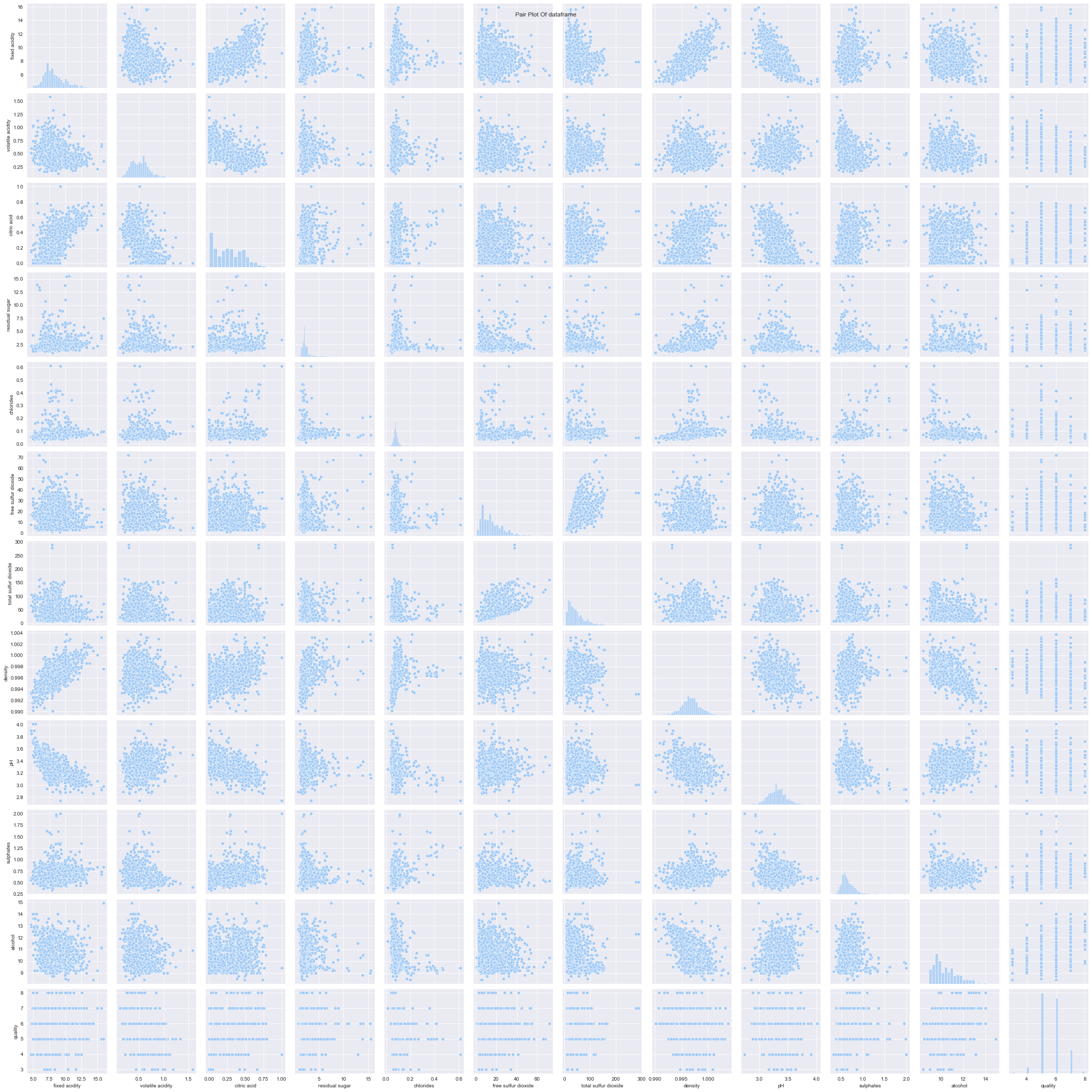
****

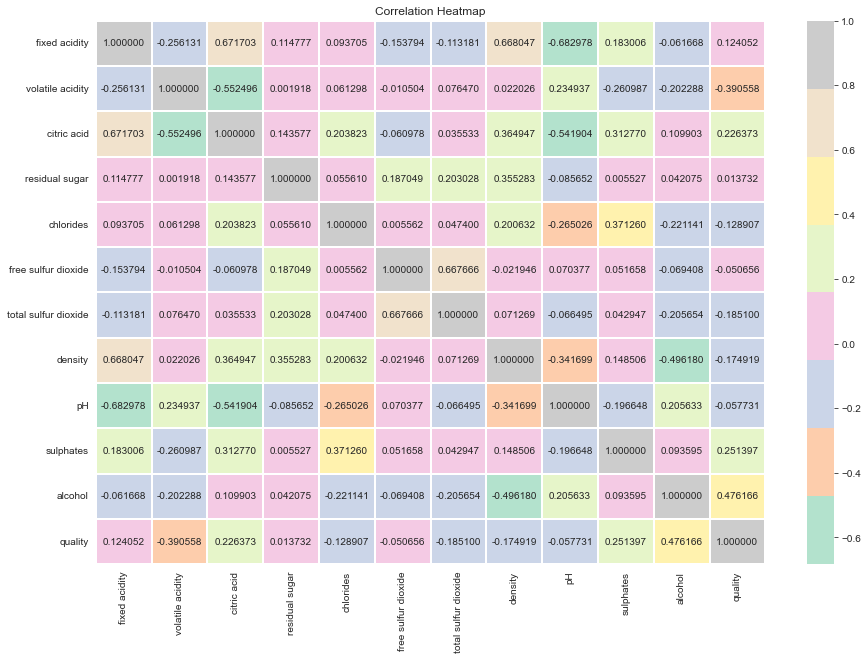
** **

****









**Observations:**

* The count plot is showing that most of the wine is of quality mark 5 or 6.
  + From the KDE plot we can infer that:
    - Fixed acidity attribute has the skewness as 0.98
    - volatile acidity has the skewness as 0.67
    - Citric acid has skewness as 0.32
    - Residual sugar has skewness as 4.54
    - Chlorides has the skewness as 5.68
    - Free Sulphur dioxide has the skewness as 1.25
    - Total sulphur dioxide has the skewness as 1.52
    - Density has the skewness as 0.07
    - pH level has the skewness as 0.19
    - Sulphates has the skewness as 2.43
    - Alcohol skewness is 0.86
    - Quality attribute skewness is 0.22
* Swarm plot is showing that the density of the data points is highest in the quality bars of 5 and then 6, later comes the bar of 7
* Violin plot is giving the observation that in the target feature quality has most of the data points in the quality value 5 and they are highly dense in between 9 and 10 percentage of alcohol
* The box plot is showing that there are multiple outliers in the 5 quality category with respective to alcohol percentage
* Pair plot is giving the correlation between each pair of features so it gives us a result of 144 plot diagrams
* The correlation heatmap is giving the correlation between each pair of the observations in the numeric format i.e. it is giving the results of the pairplots in the numeric format. The correlation heatmap is a visual representation of the covariance matrix of the entire dataset.

**Conclusion:**

Each of the data visualization technique has its own unique use and they can be used to understand the data better and assist in employing the appropriate machine learning algorithm to solve the problem.

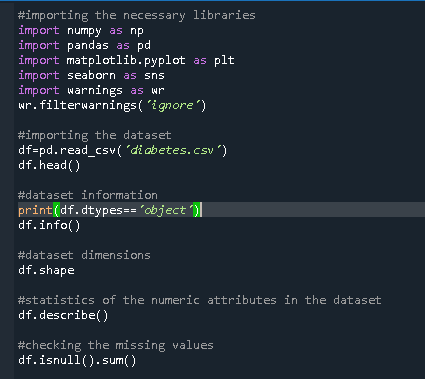
**TASK 8: K Nearest Neighbours Classification with elbow method**

**AIM:** Creating a Classifier to for the given dataset using the K nearest neighbours algorithm and deciding the optimal k value using the elbow method and cross validation.

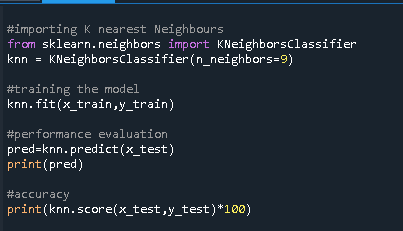
**Methods and Description:**

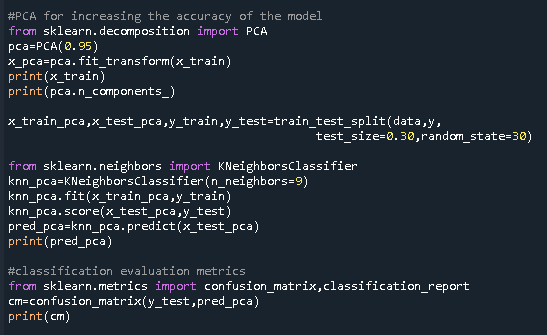
* K Nearest neighbours, or simply called as KNN is a simple, yet powerful classifier algorithm.
* It belongs to the class of Lazy learners, so the training phase time is almost negligible.
* When the test set arrives, then KNN takes the training data and then try to find the nearest ‘k’ neighbours to each test data element and assign the appropriate class to the test data item.
* The KNN algorithm is implemented on the diabetes datase, which is of dimensions 768 x 9 and has no missing values and categorical variables.
* The target feature is labelled as ‘Outcome’.
* The training and testing data split is in the ratio of 70:30.
* Principal component Analysis is used to increase the accuracy of the model.
* The most important hyperparameter for this model is ‘k’ value and the most optimal ‘k’ value can be predicted using elbow method and cross validation.

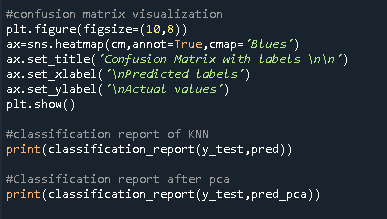
**Source Code:**

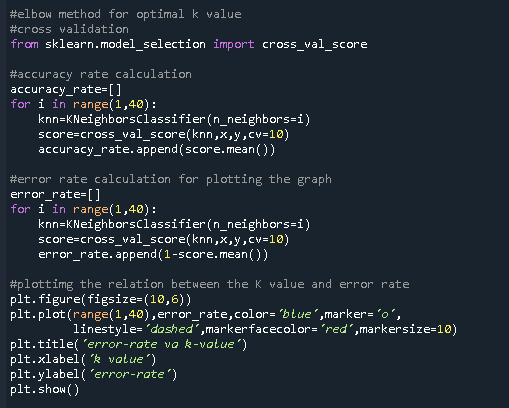
****

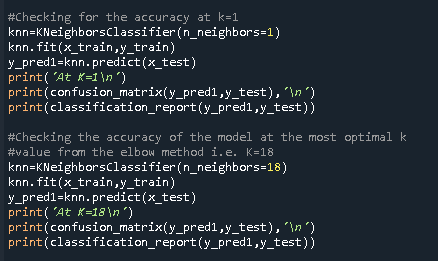
****

****

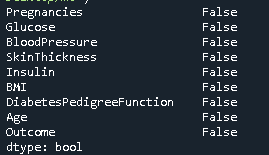
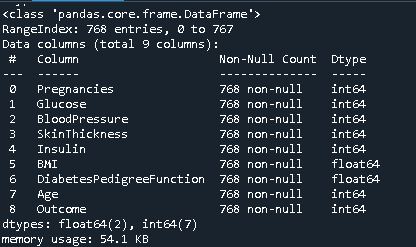
****

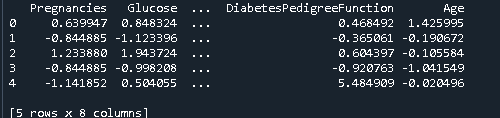
****

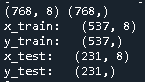
****

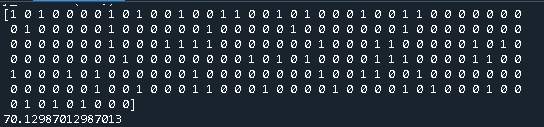
****

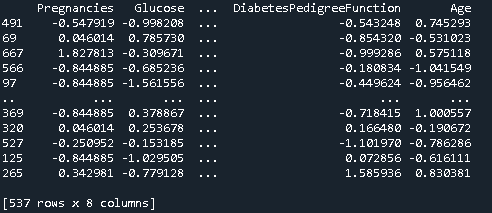
**Output:**

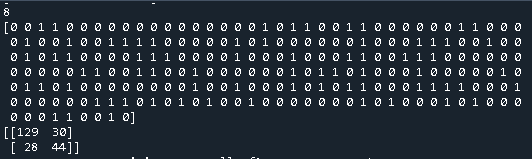
** **

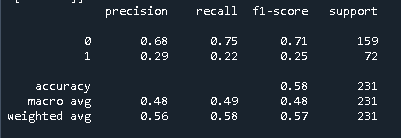
****

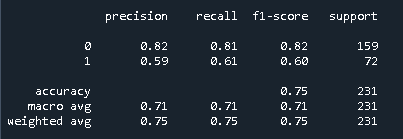
****

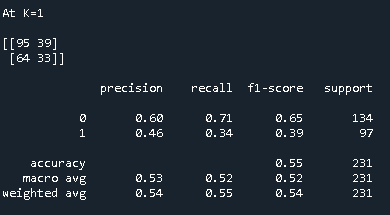
****

****

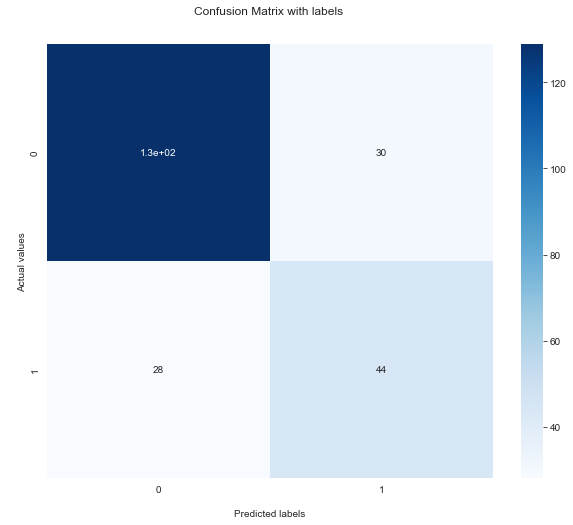
****

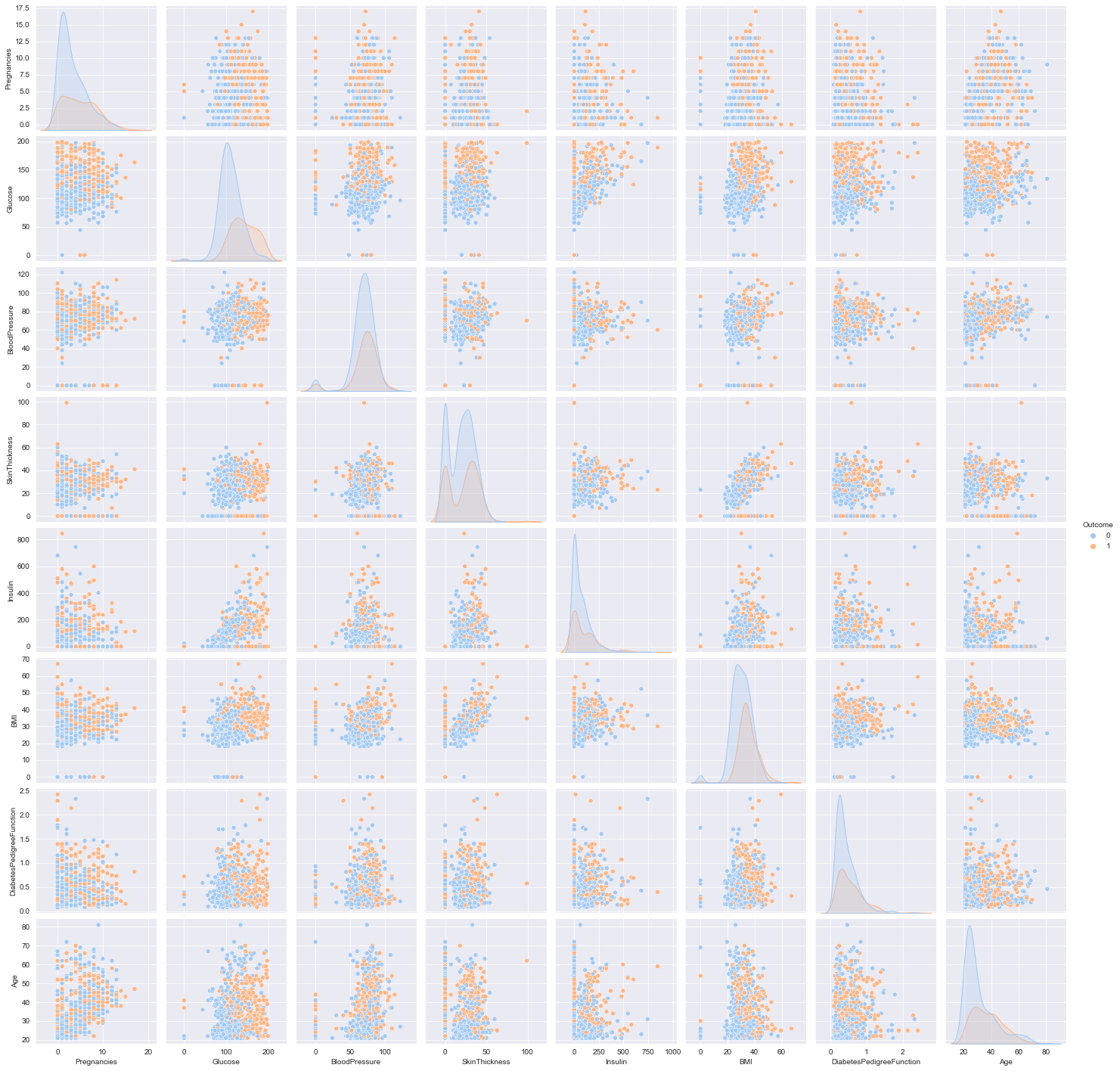
****

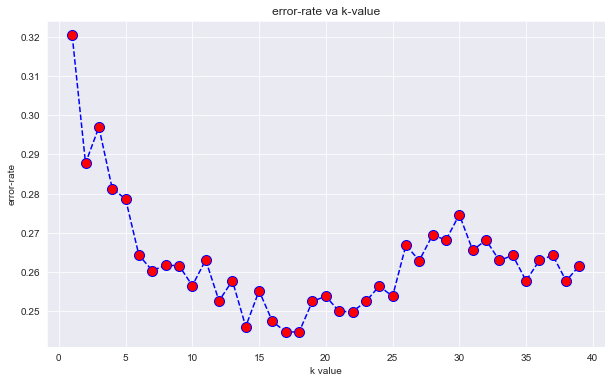
****

****

****







**Observations:**

* The KNN is showing an accuracy score as 70% at k=9
* After Principal Component Analysis, the score level has improved and reached to 74% at k=9
* F1 score at k=9 before PCA is 0.58, after PCA it is found to be 0.75.
* After using the elbow method to find the optimal K value, we found that at k=18 without PCA the F1 score value is found to be is 0.68.
* After PCA the F1 score has improved to 0.77 at k=18.

**Conclusion:**

The principal component analysis is very useful in improving the accuracy of the model by transforming the features into useful features, by reducing the noise and irrelevancy amongst them.

The elbow method is very useful in finding the most optimal ‘k’ value which has improved the performance of the model.

When both of these techniques are used together, they are resulting in the highest level of the accuracy and F1 score for the prediction.

**TASK 9 : Fit Linear SVM to the non linear datasets**

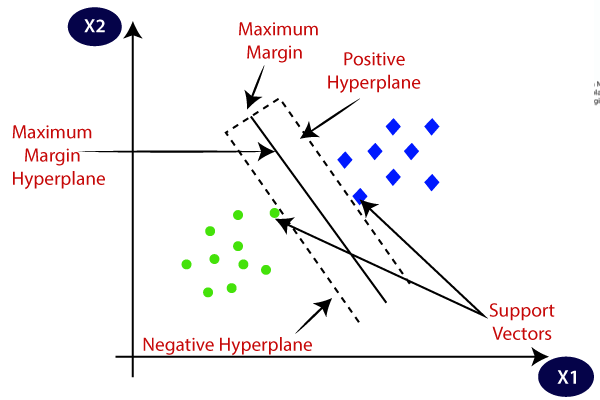
**AIM :** To fit linear Support Vector Machine on thre datasets : Breast Cancer,titanic dataset and Iris dataset.

**Algorithm and Dataset Description :**

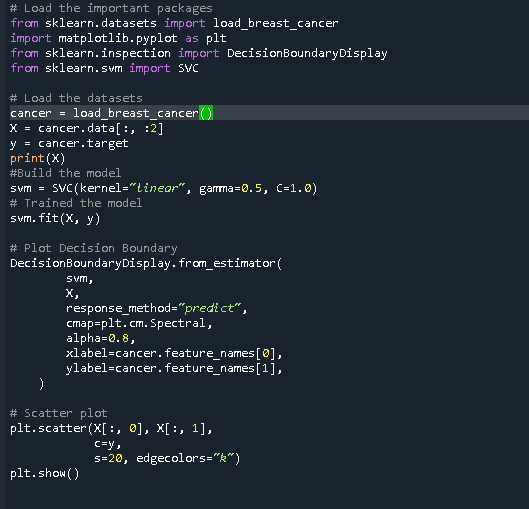
* Breast Cancer Dataset is a famous dataset primarily used for prediction tasks.
* The target feature is diagnosis where we predict if the cancer is malignant or benign.
* It has 32 features but we selected only 2 features i.e. radius1 and texture 1
* It has no missing values and the features are numeric.

**Support Vector Machine :** Support Vector Machine is a powerful machine learning algorithm.

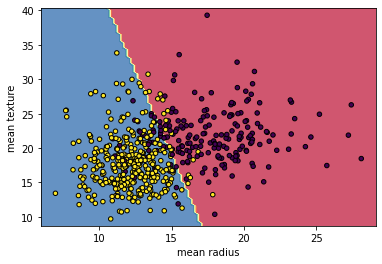
* It classifies the data based on finding the decision boundary with maximum margin.
* If the data is not linearly separable, SVM tries to map the data into higher dimensional feature space where the data can become linearly separable, called as kernel trick. Hence, that type of SVM is called kernel SVM.
* The distance parameter can be Euclidean distance.
* SVM has multiple hyperparameters involved in it like C, gamma.
* The distance metric considered and the kernel function used are also important hyperparameters.



**Source Code:**

****

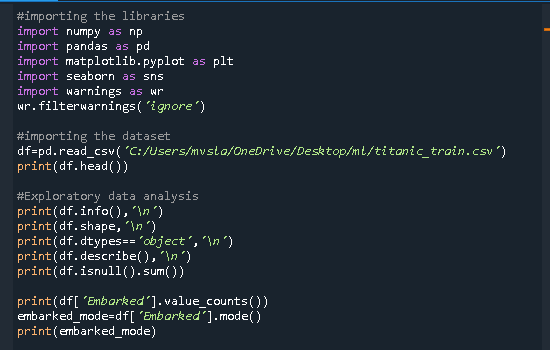
**Output:**

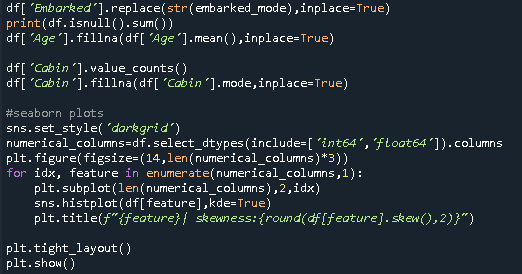
**** 

**Titanic Dataset:**

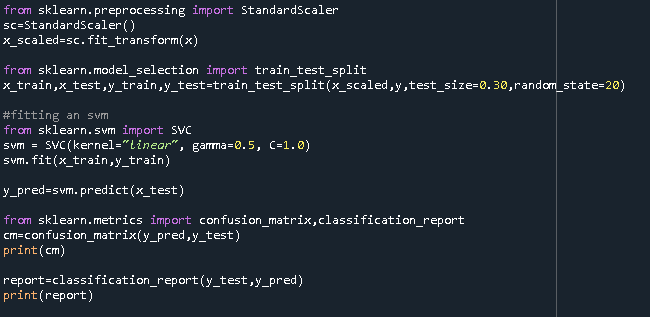
* Titanic Dataset is mainly used for classification tasks.
* It has 11 features and the target class is Survived or not.Hence it is binary classification.
* There are missing values and they are handled using fillna() method.
* There are some unnecessary features in the dataset like the address, ticket type and many more. They were removed and only Pclass, Sex, Age, Fare, Embarked are selected as feature set.

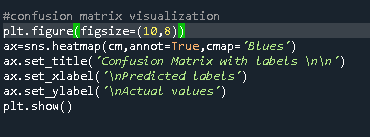
**Source Code:**

****

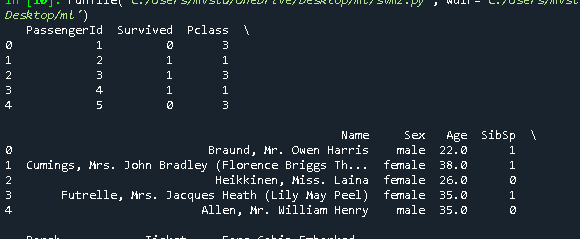
****

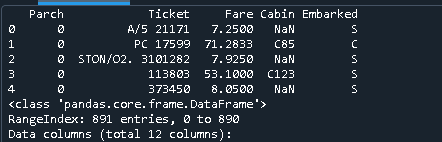
****

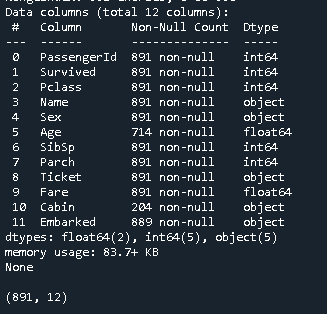
****

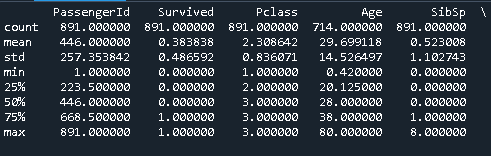
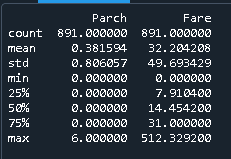
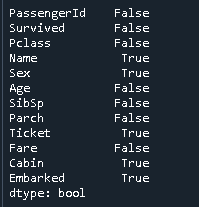
****

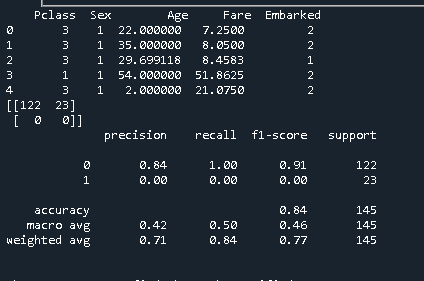
**Outputs:**

****

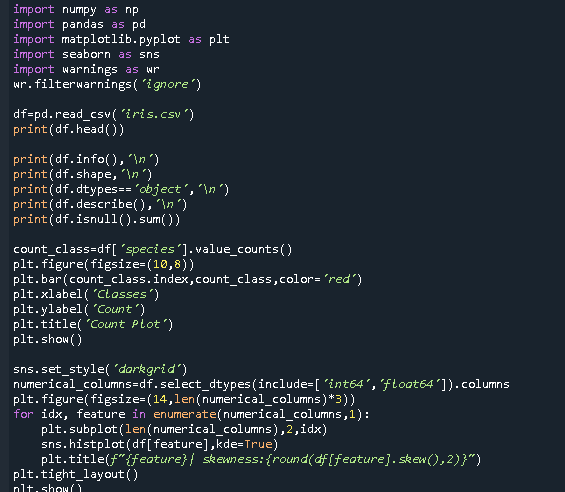
****

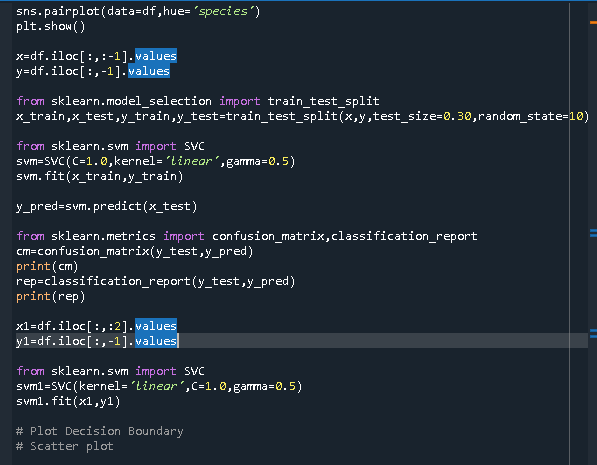
**]**

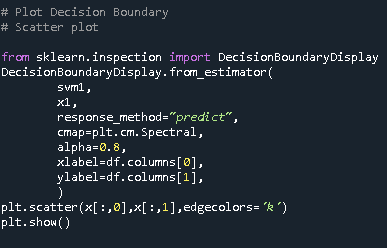
****

****

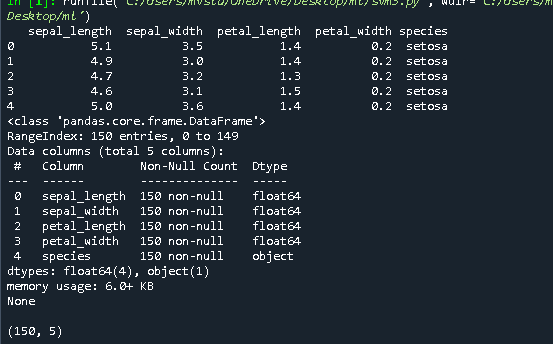
**Source Code for Iris Dataset:**

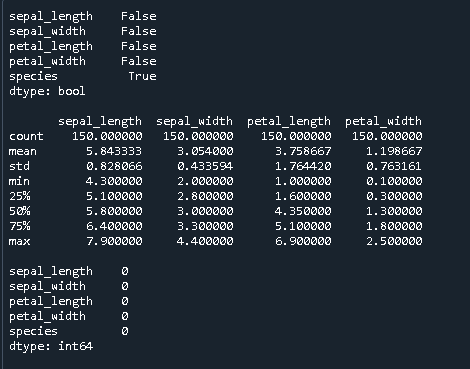
****

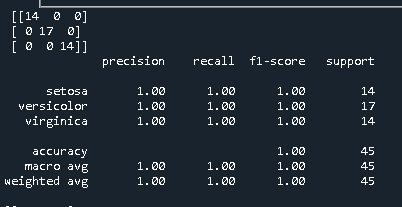
****

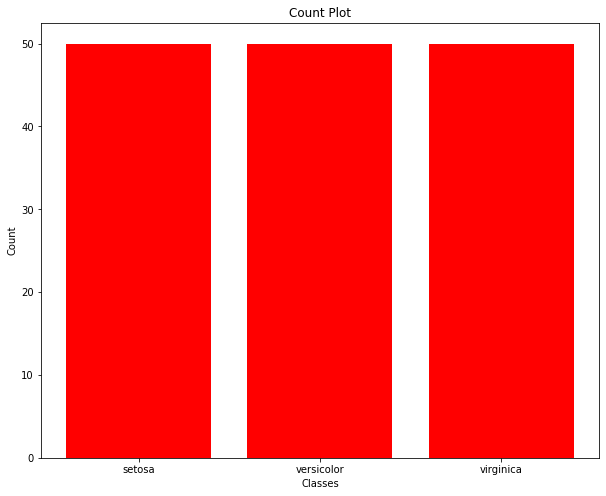
****

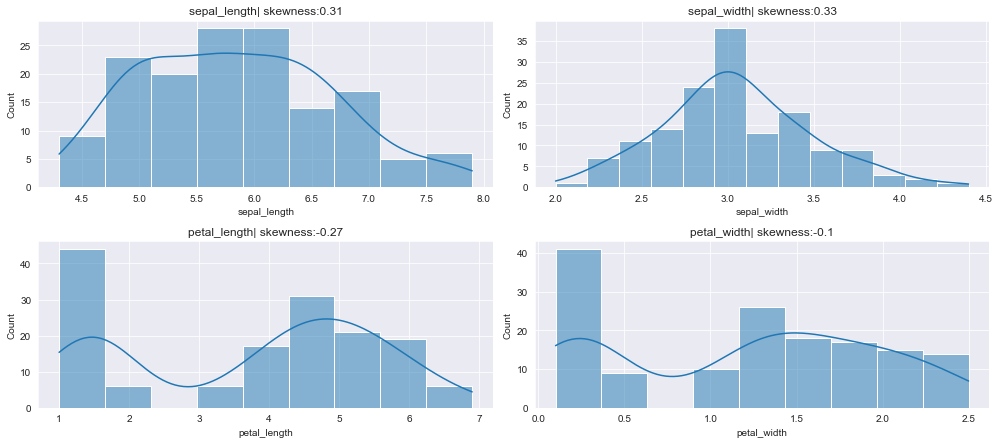
**Outputs:**

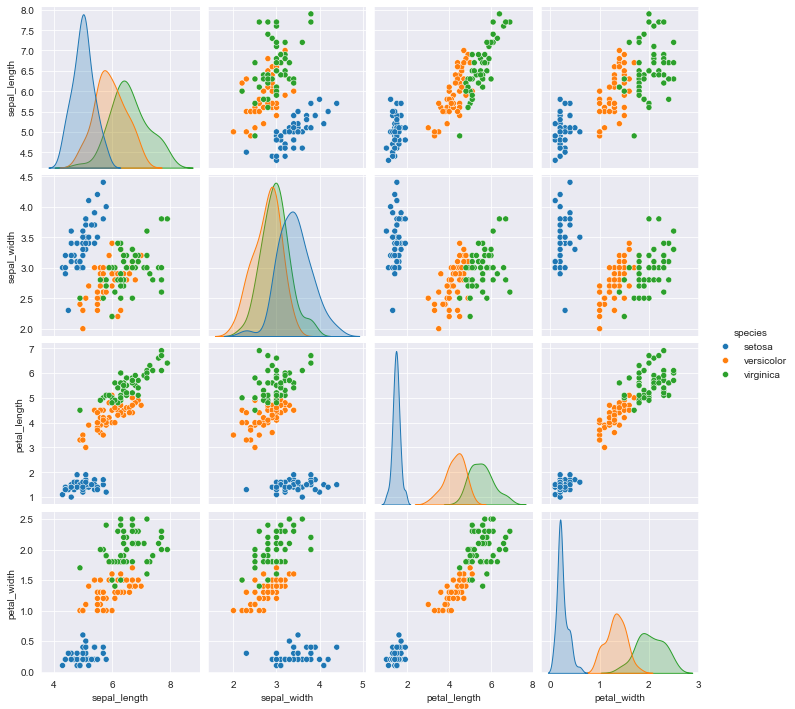
****

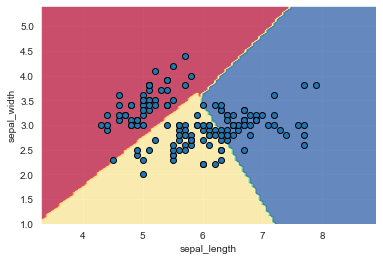
****

****









**Observations:**

* SVM tries to separate the data in the linear feature space and the accuracy scores are calculated.
* The accuracy score for titanic dataset is calculated to be 84%.

**Conclusion:**

SVM results are a decent level of accuracy. The model performance can be improved using cross validation and going for different support vector kernels.

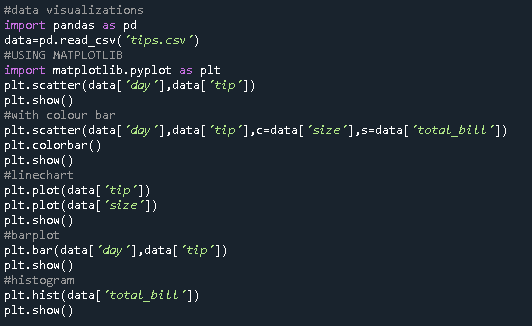
**TASK 10: PERFORM DATA VISUALIZATION USING DIFFERENT LIBRARIES**

**AIM:**Performing Data Visualization using different libraries

**Dataset Description:**

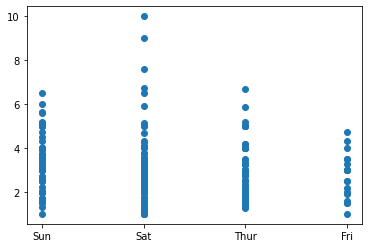
* The tips dataset is used for performing data visualizations/
* It is a dataset of dimensions 244 rows x 7 columns and has no missing values.

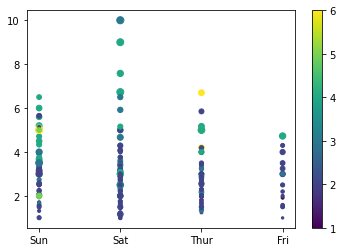
**Source Code:**

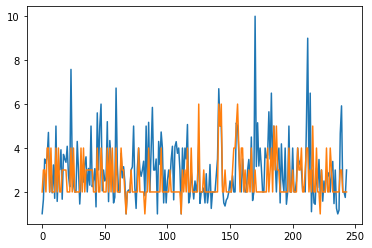
****

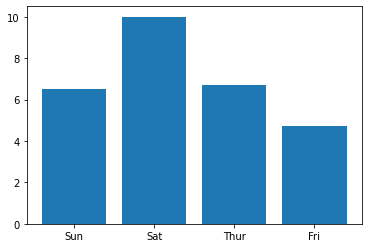
****

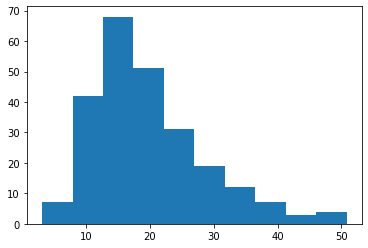
**Outputs:**

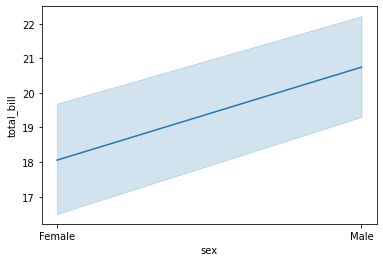


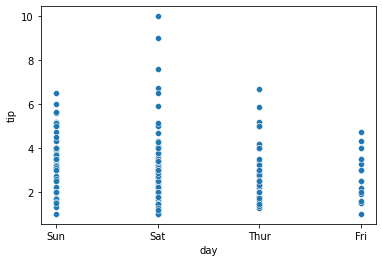


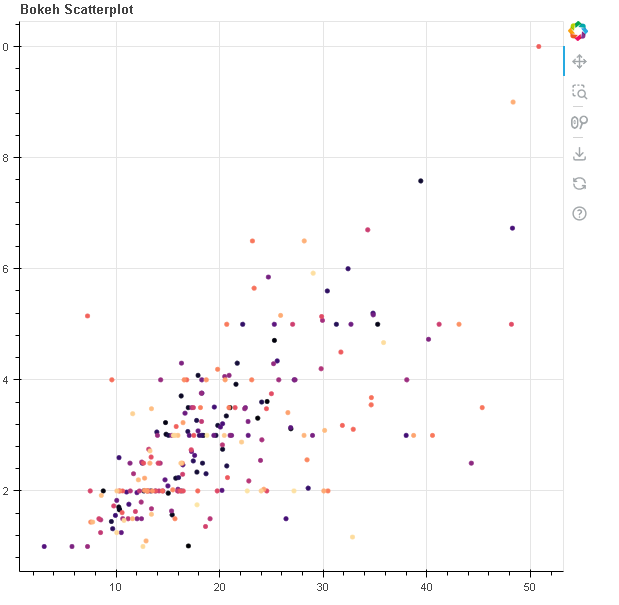


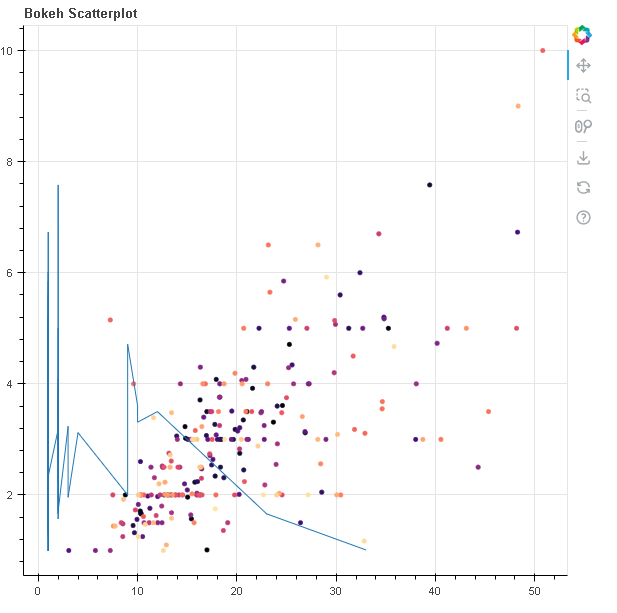


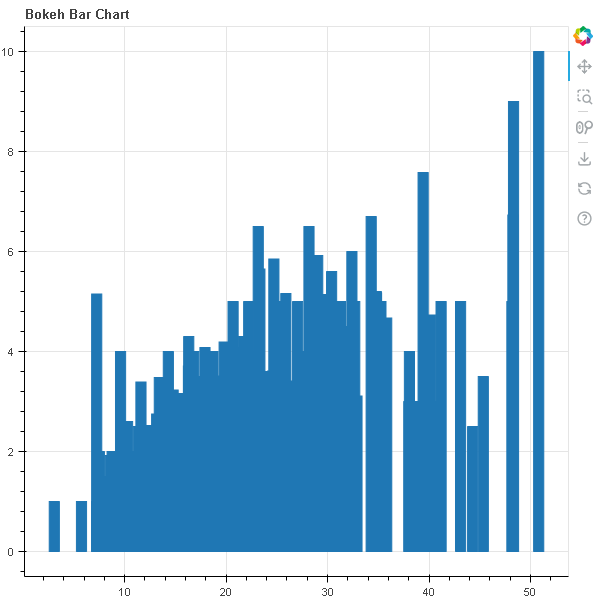






****

****

****

**Observations:**

* Matplotlib is a powerful library where we can construct many plots.However, matplotlib is having many parameters to pass.
* Seaborn is another powerful library for visualization which can construct many plots and is easier to use
* Bokeh is a great data visualization library which can create some advanced visualizations and can generate the visualizations in web format.

**Conclusion:**

The type of library is to be chosen as per the requirement and as per the plot.

**TASK 11: Fit a Classification model i.e. Decision Tree.**

**AIM :** To fit a decision tree model the two datasets on balance scale and distance database and daily weather prediction.

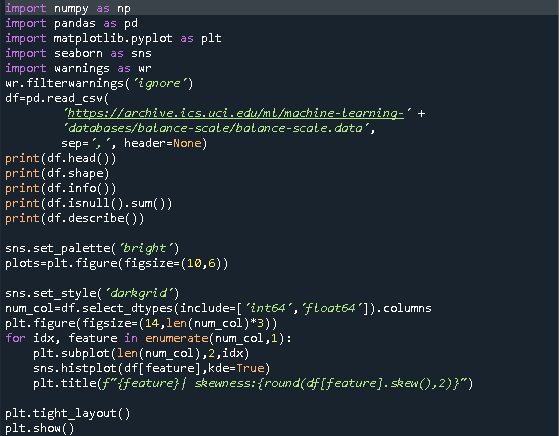
**Algorithm and Dataset Description:**

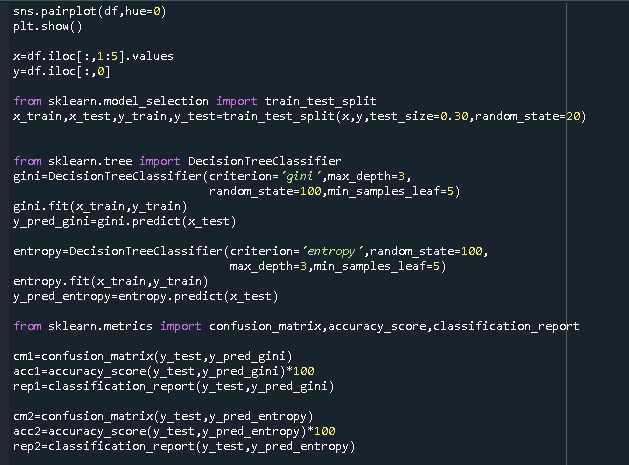
* **Decision Tree:** Decision Tree is a supervised machine learning algorithm mainly used for classification tasks.
* The goal of Decision Tree is to predict the best possible decision from the past data.
* It is in tree like form where the decisions are represented by nodes and the edges contain the outcomes of the decision and the leaf nodes contain the final decision.

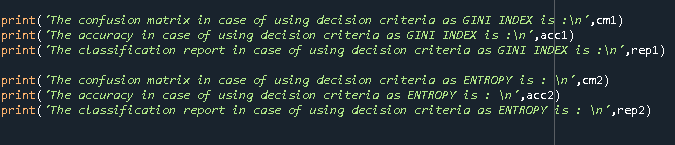
**Dataset Description:**

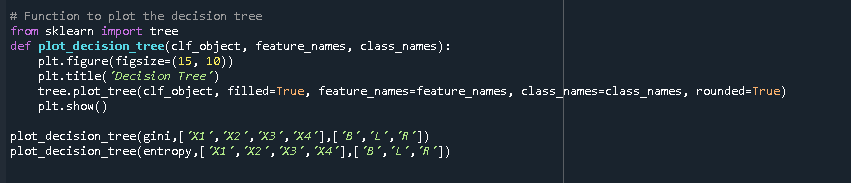
* The balance scale and distance database is present in UCI machine learning repository which is about psychological data.
* It is primarily used for classification tasks and the data set does not contain any missing values.
* The daily weather dataset is about the weather recorded on San Diego, California over a period of three years.
* The dataset has the dimensions as 1095 rows x 11 columns.
* The target feature is humidity level.

**Source Code:**

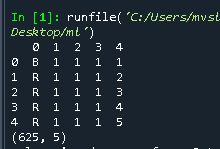
****

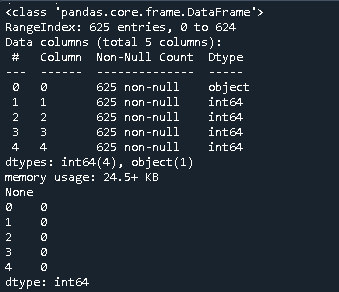
****

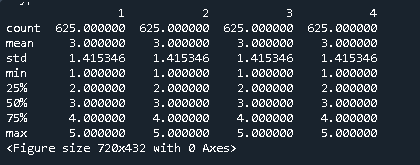
****

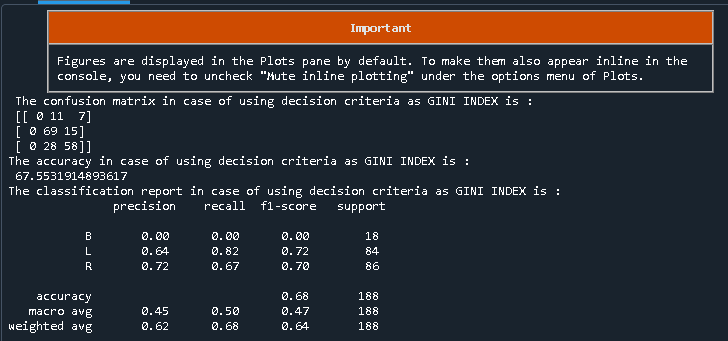
****

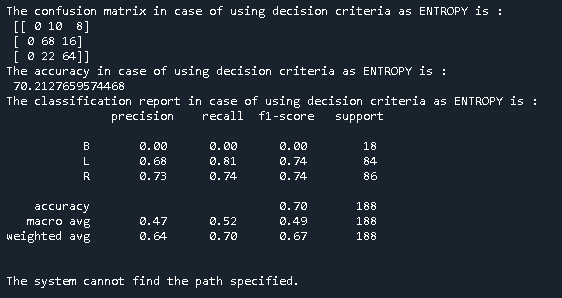
**Outputs:**

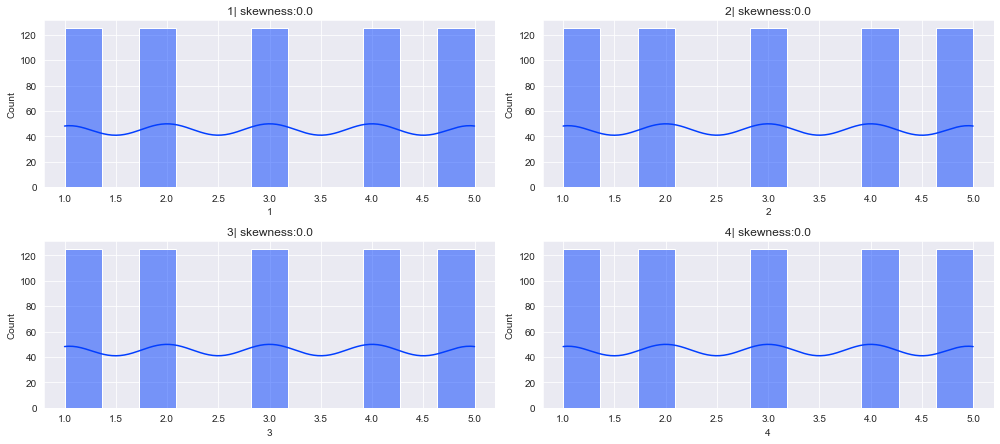
****

****

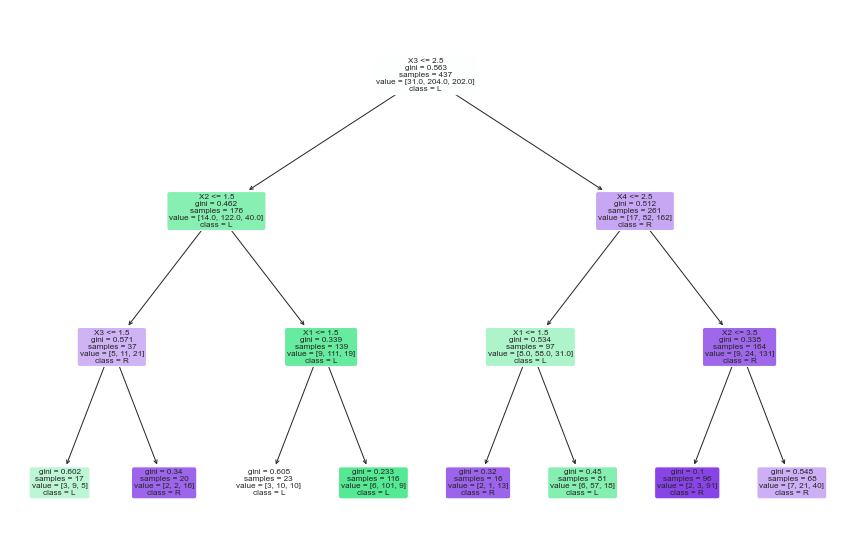
****

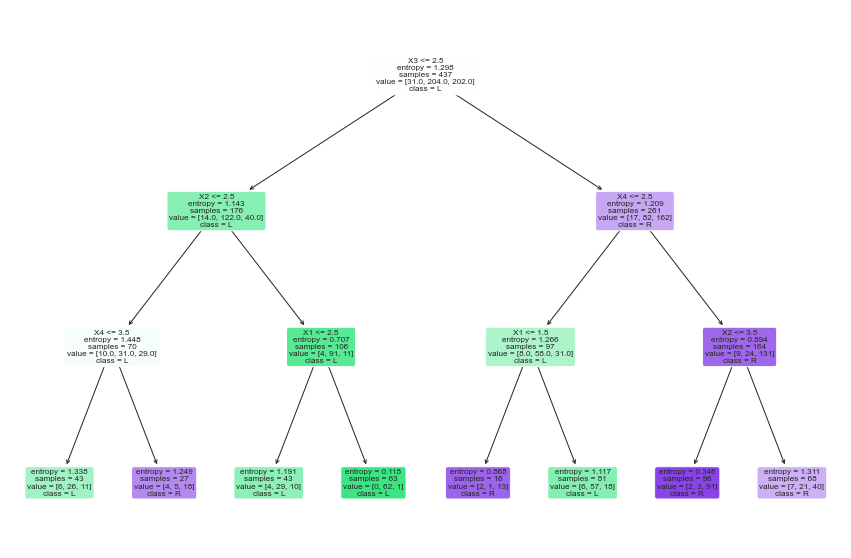
****

****

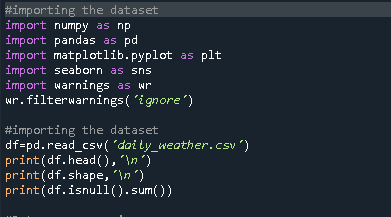


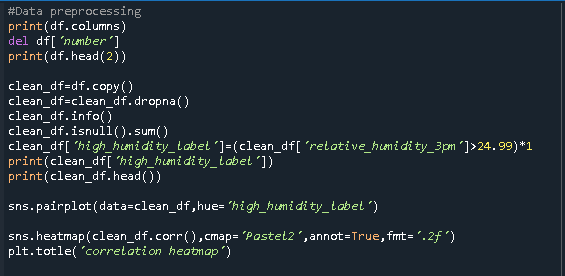


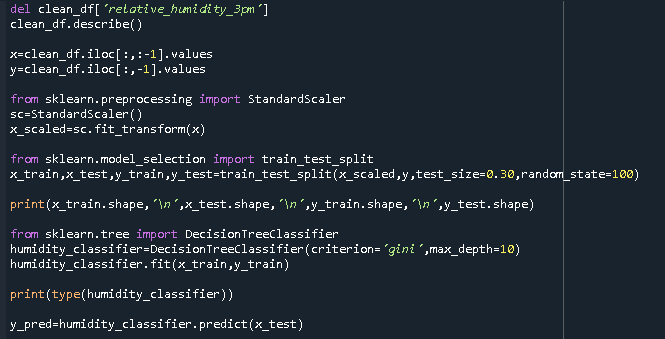


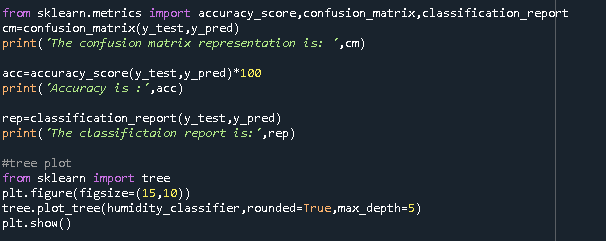


**Source Code for Daily Weather Data:**

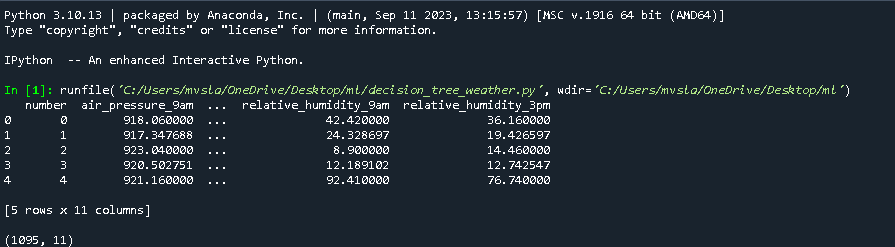
****

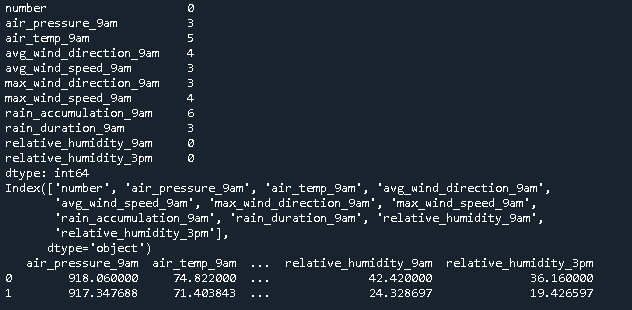
****

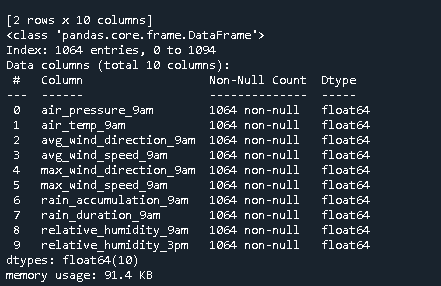
****

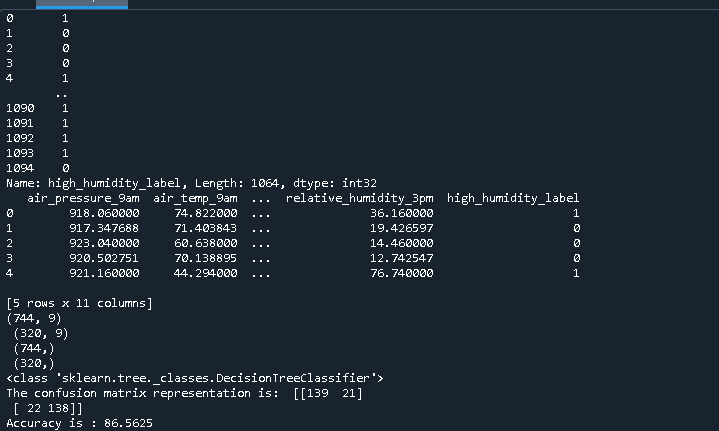
****

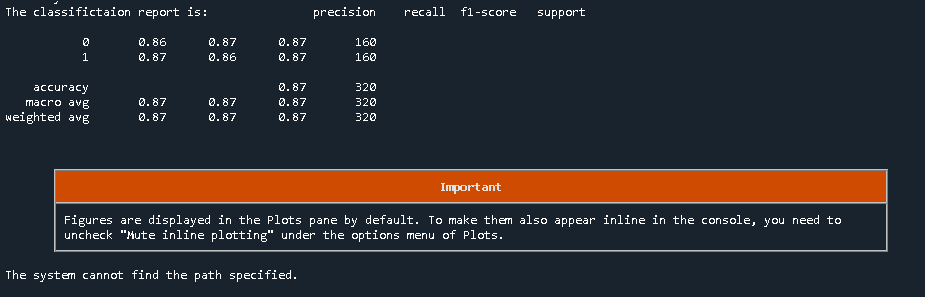
**Output:**

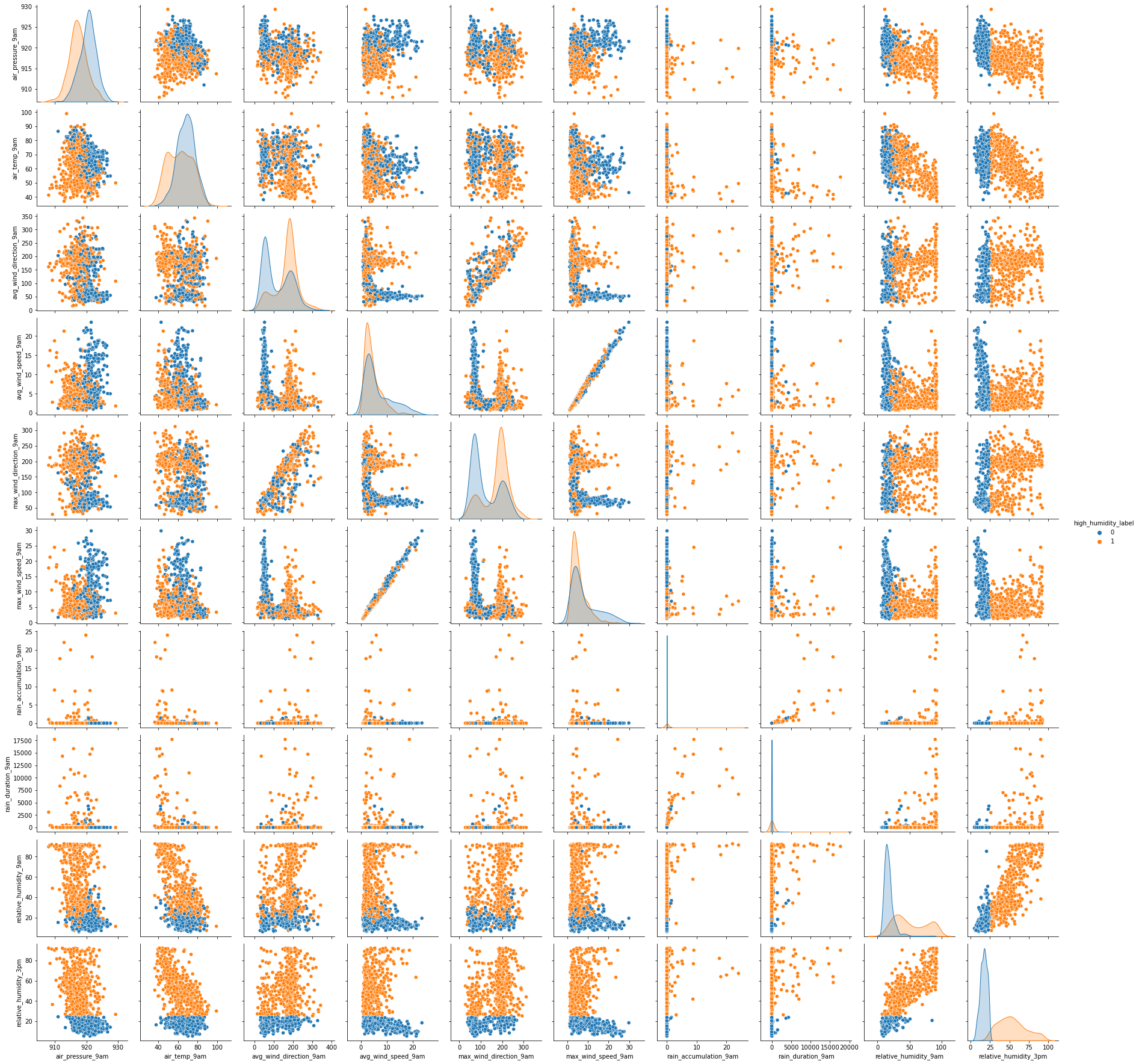
****

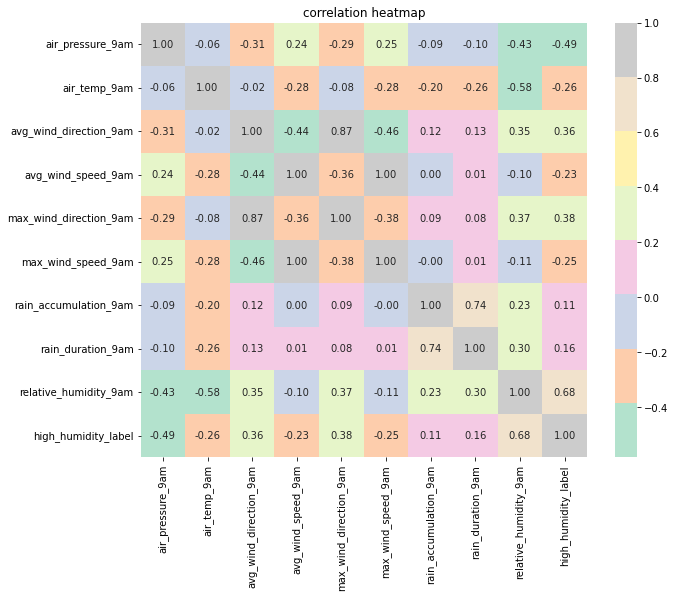
****

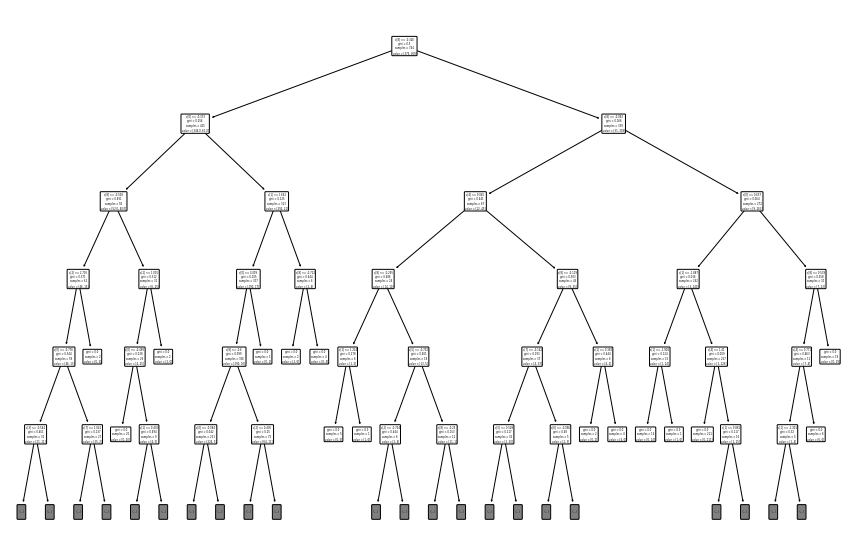
****

****

****







**Observations:**

* The accuracy of the decision tree in case of using gini index for Balance weights and distance database is 68%
* The accuracy of the decision tree model in case of entropy for the Balance dataset is 70%
* The accuarcy of the decision tree model in case of using gini index on the daily weather dataset is 87%
* Decision tree is highly prone to overfitting as the model tries to learn from the training data in the most perfect form.
* So to avoid the problem of overfitting, techniques like prepruning and postpruning are used.
* Random Forest model is an ensembling model machine learning where multiple decision trees are used and the overfitting problem is resolved.

**Conclusion:**

The Decision tree model has been fitted to the given datasets.

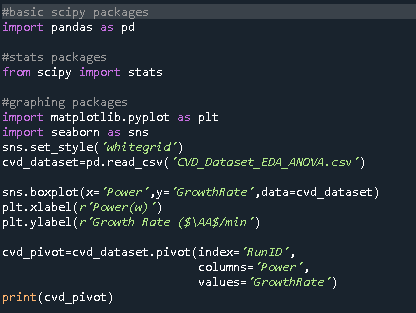
**TASK 12: Perform One Way ANOVA**

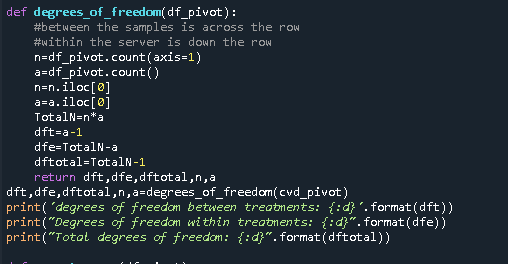
**AIM:** To perform one way ANOVA

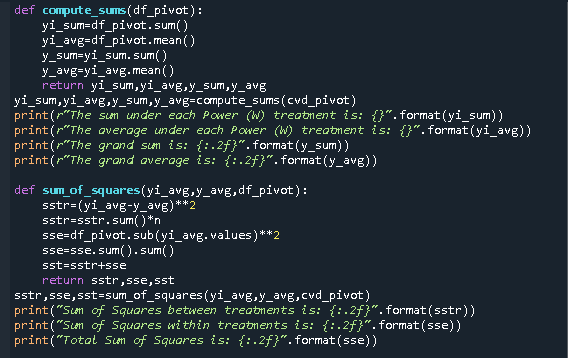
**Concept and Dataset Description:**

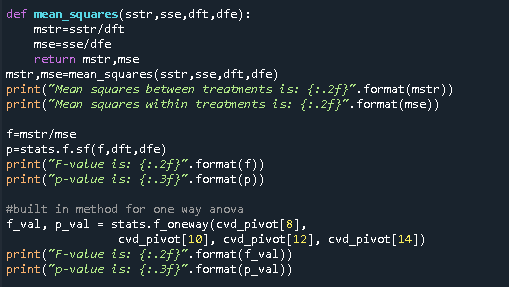
* Concept of ANOVA: ANOVA or Analysis Of Variance is a statistical technique where we divide the variations into variations due to known factors and variations due to unknown factors.
* ANOVA is based on agricultural experiments and is a powerful technique used in statistics to test various factors as an alternative to independence of two means t-test in statistics.
* Dataset used is CVD Dataset.

**Source Code :**

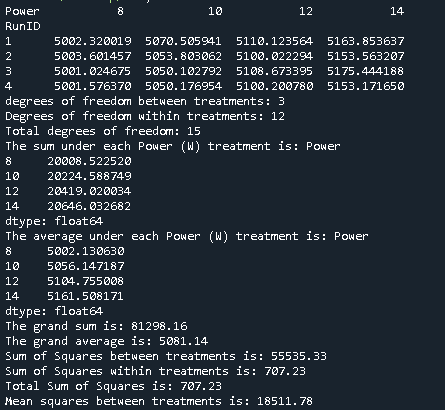
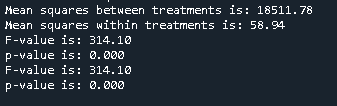
****

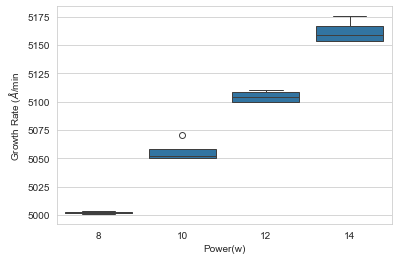
****

****

****

**Outputs:**

** **



**Observations:**

* The method of least squares is used to estimate different types of least square values in ANOVA.
* The calculated F- value is found to be 314.10
* The p-value is found to be 0.00
* Since the p-value is found to be very small, we may reject the null hypothesis that the power does not have an influence on the growth rate.

**Conclusion**:

From the Analysis Of Variance, we can conclude that the power has influence on the growth rate from the given CVD dataset.

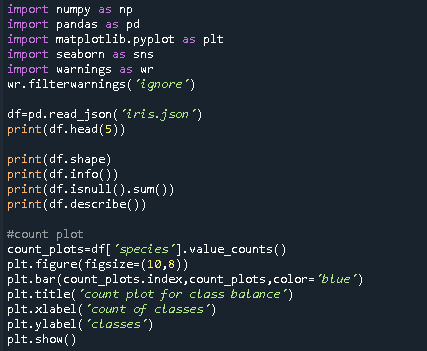
**TASK 13: Work with JSON data**

**AIM :** To work with JSON data i.e. JavaScript Object Notation Data format.

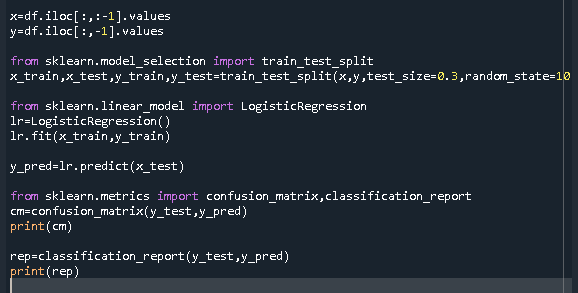
**Concept:**

* Most of the data in the real world is in the form of JSON format.
* JSON is a semi structured format of the data where the values are found in the key-value pairs.
* Most of the online data is found in HTML/DHTML/JSON format.
* So, it is imperative to work with JSON data when working in real-time projects.
* Here, iris data is loaded in the JSON format.

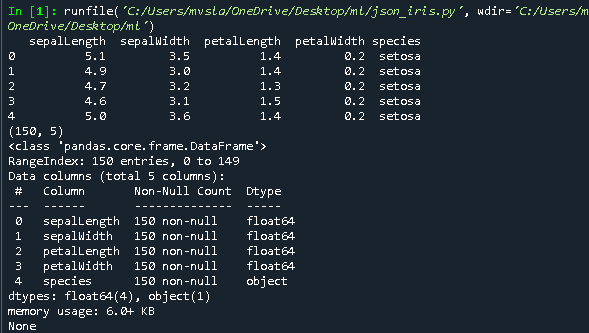
**Source Code :**

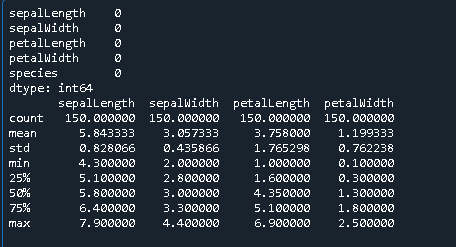
****

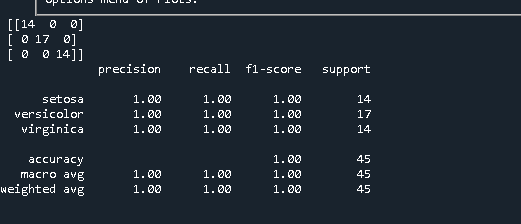
****

****

**Outputs:**

****

****

****

**Observations:**

* The JSON data is easier to work with when pandas method pd.read\_json() is used.

**Conclusion:**

JSON data is easy and very important to understand and work with if we are in the data engineering field.

**TASK 14: Apply Simple and Multiple Linear Regression**

**AIM :** Application of simple linear regression on Salary Dataset and Multiple Linear Regression on USA Housing Dataset.

**Algorithm and Dataset Description:**

**Algorithm: Linear Regression**

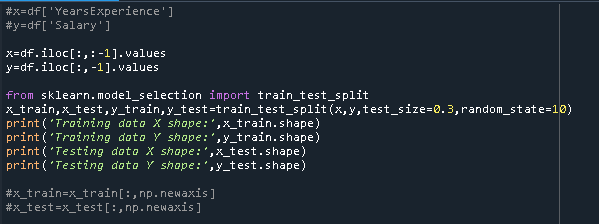
* Regression is another predictive analytics task where we try to predict a real number instead of a category.
* Linear regression uses a straight line to model the data and then uses that model to predict the value.
* Simple linear regression equation is **y=a+bx** implying that the target feature has only one predictor variable.
* Multiple linear regression equation is **y= b0+b1x1+b2x2+…..bnxn** implying that the target feature is predicted using multiple predictor variables.

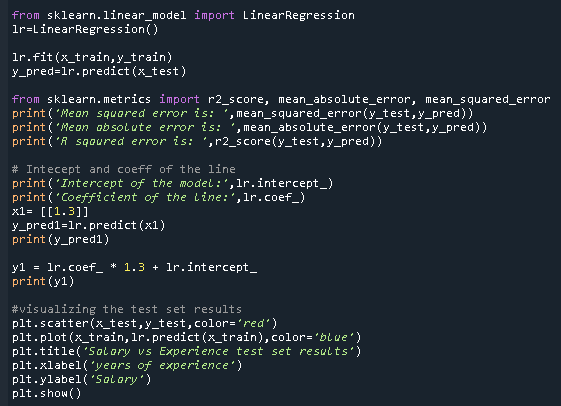
**Datasets Description:**

* Salary dataset is a simple dataset of size 30 rows x 2 columns.
* USA housing dataset is a dataset of size 5000 rows x 7 columns where price is the target feature.

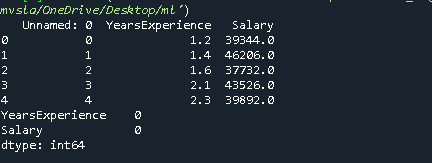
**Source Code For Salary Dataset:**

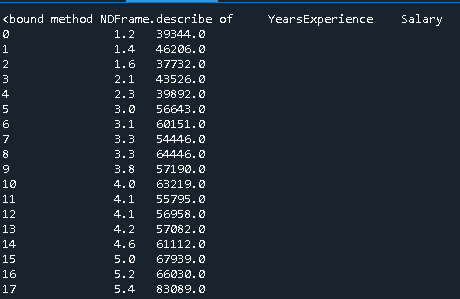
****

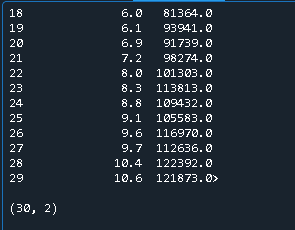
****

****

**Output:**

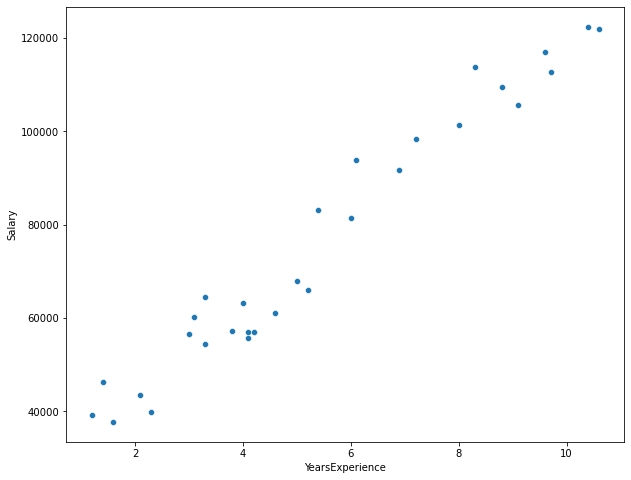
****

****

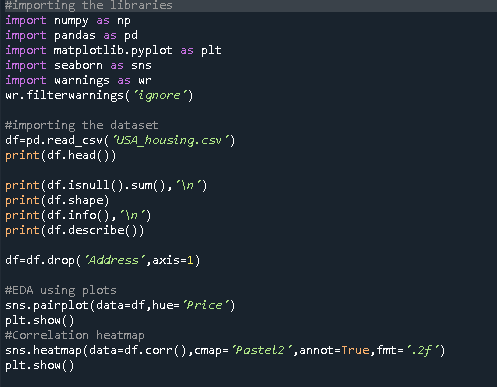
****

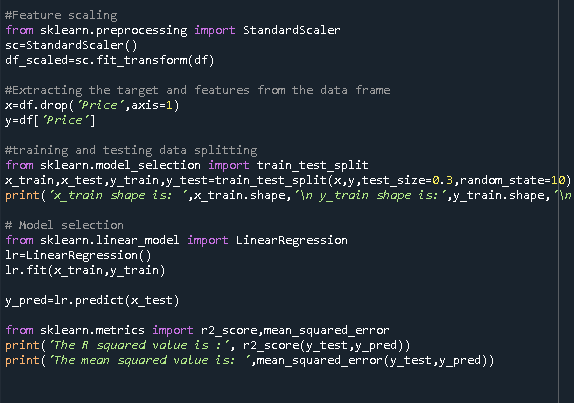
****

****

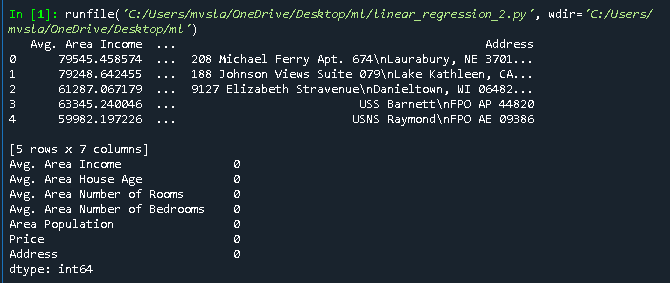


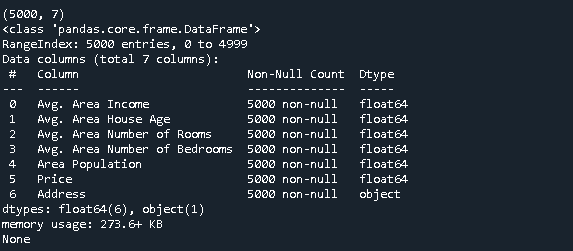
**Source Code for USA Housing:**

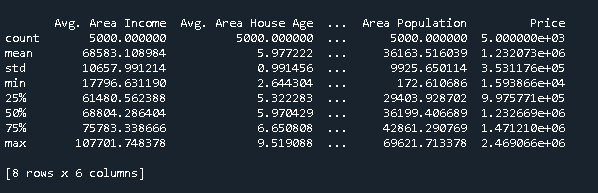
****

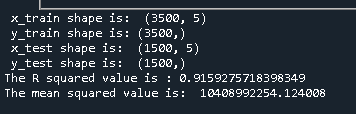
****

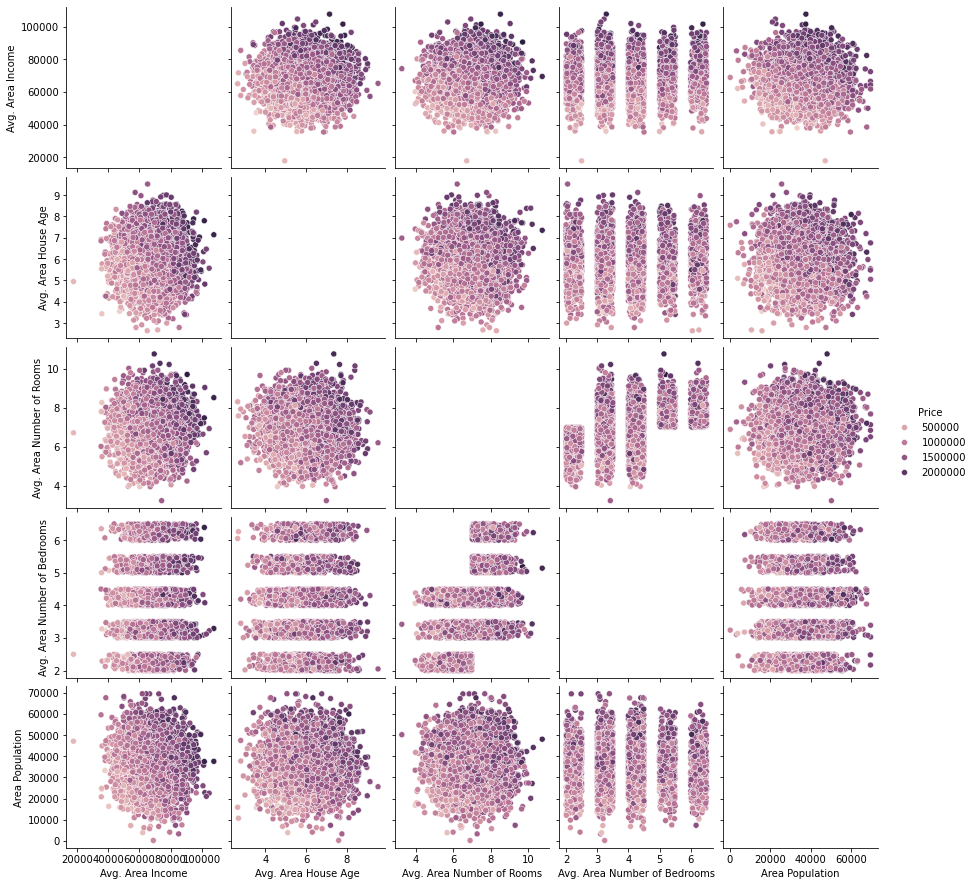
**Outputs:**

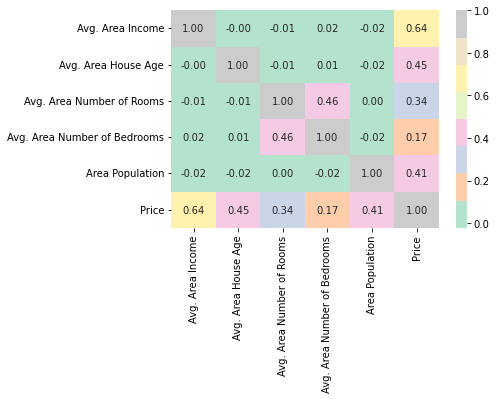
****

****

****

****





**Observations:**

* The linear regression is used to predict the future value using the given data using the regression equations.
* The R squared value is found to be 0.96 for salary dataset and 0.91 for USA Housing dataset.
* The mean squared error in salary dataset is : 21713548.637118697
* The mean squared error in USA Housing dataset is :10408992254.124008

**Conclusion:**

The mean squared error value must be minimised and the R squared value must be maximized in regression problems. Simple linear regression is picked when the dataset has only one feature and multiple linear regression is picked when the dataset has more than one feature.

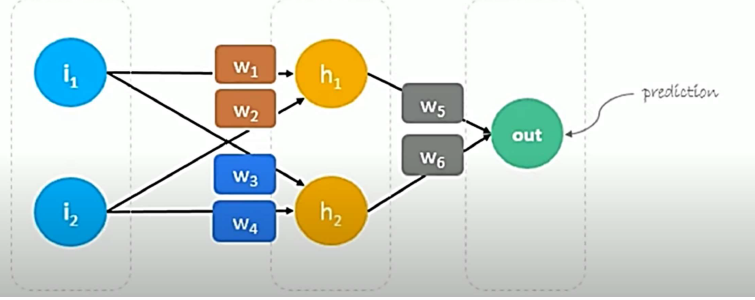
**TASK 15: Apply an artificial neural network**

**AIM :** Applying of an artificial neural network using keras on churn modelling dataset.

**Concept and dataset description:**

**Concept of ANN:**

* Artificial Neural Networks are inspired from the working of human brain.
* Neural Networks consist of small units called neurons or perceptrons, which can have multiple inputs **x1,x2…xn** and outputs y1.
* The inputs of the neurons with corresponding number of weights and one bias term is added.
* Multiple neurons are comined together to form layers.
* These layers are connected to form the network, by feeding the outputs of one layer as the inputs of the next layers, hence forming the multi layer architecture.
* A simple diagram of the ANN is as follows:

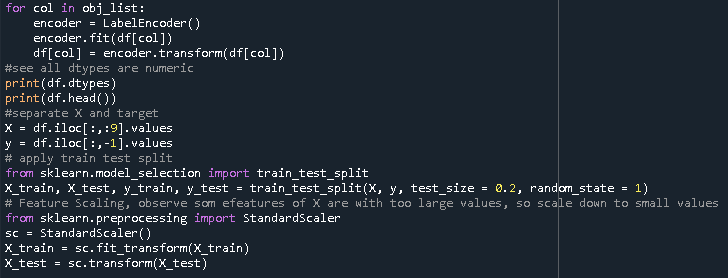


**Dataset Description:**

* The dataset is called as ‘Churn Modelling’.
* It is a very important dataset where the data is about the customers’ data of them staying with the bank or leaving the bank.
* It has 10,000 rows and 14 columns, with no missing values.
* Since it is a bigger dataset, it is important to move to deep learning models i.e. ANN to solve the predictive problem
* The ANN was implemented using keras library, with tensorflow framework at the backend.

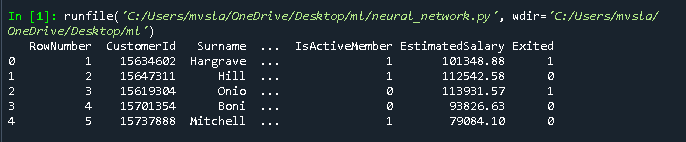
**Source Code:**

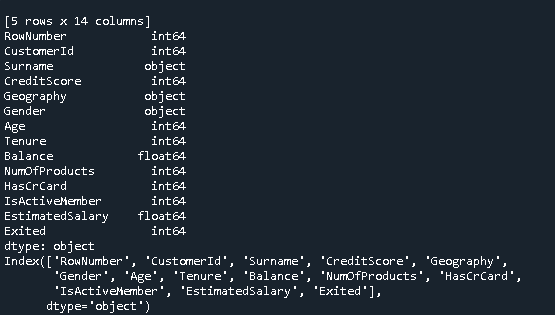


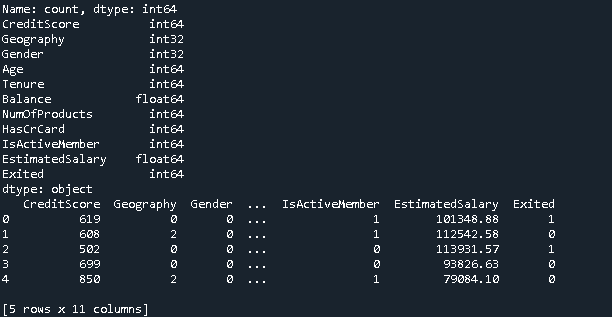


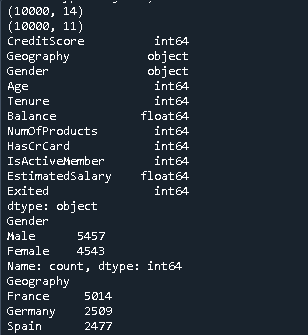


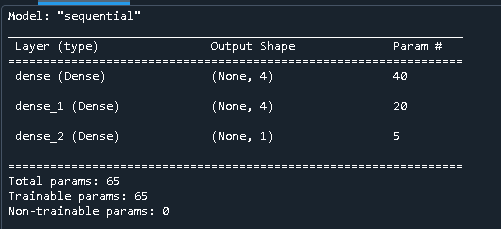
**Outputs:**

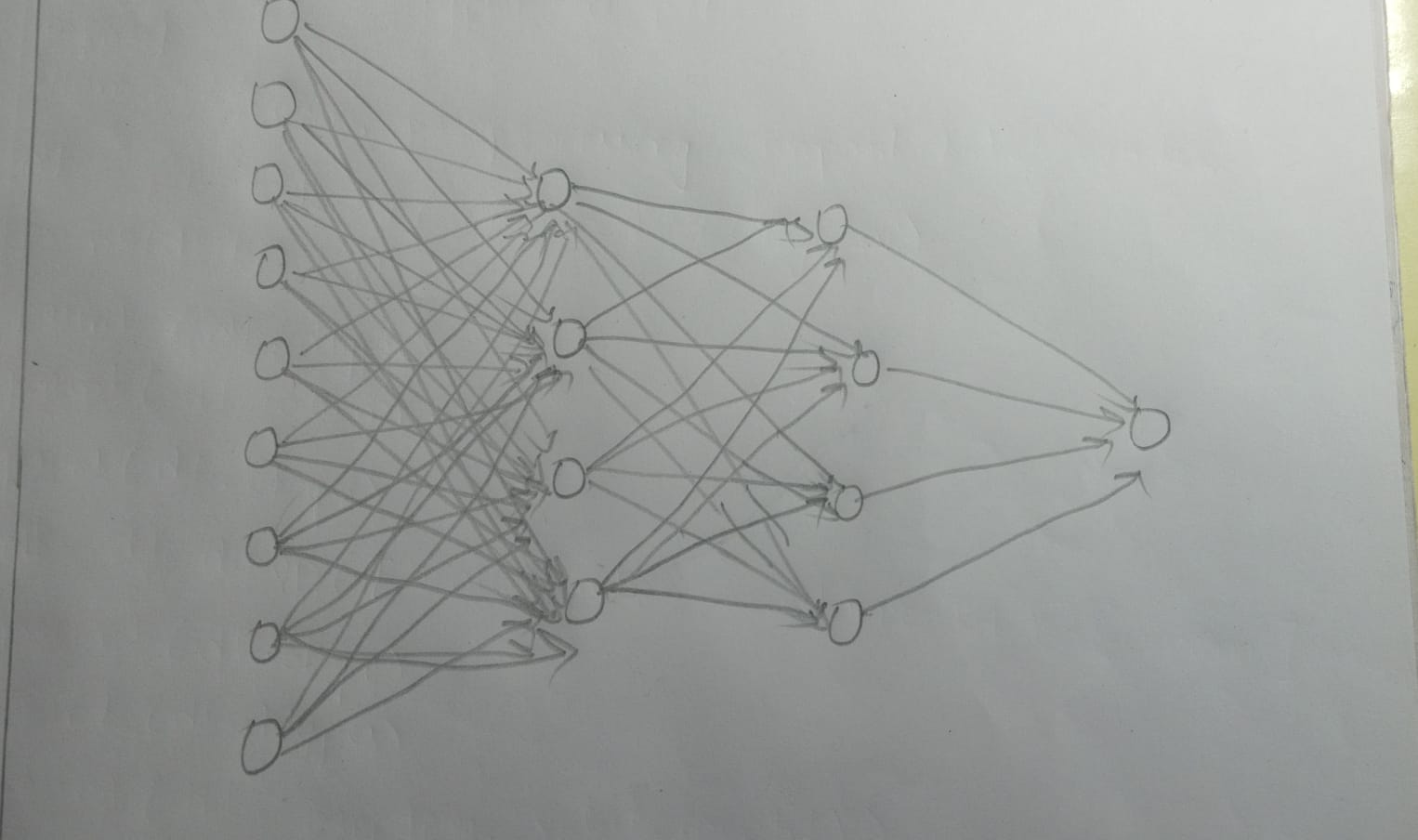
****

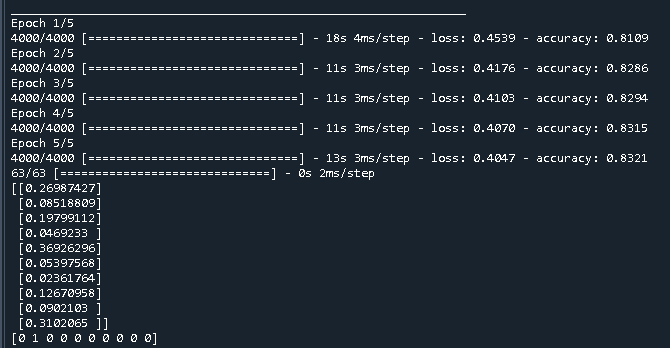
****

****

****

****



****

**Observations:**

* Each epoch consists of one forward pa(ss and one backward pass over the entire data.
* The gradient descent type is set to stochastic gradient descent.
* The weights and bias i.e. the total number of trainable parameters are : 65
* The weights and bias are updated using the concept of gradient descent with backpropagation to optimize the cost function.
* The accuracy of the network has increased from 81% to 83% in 5 epochs or 5 iterations.
* We can see that the predicted values are closer to the actual values and are lying between 0 and 1 but not exactly 0 and 1 as the activation function used at the output layer is sigmoid activation function which constitutes the range (0,1).
* The activation functions are used to introduce the non-linearity into the linear computation in the neural network.
* The hyperparameters such as learning rate, no.of neurons, no.of hidden layers, gradient descent type, epochs and many more control the model’s learning of the weights and bias term.
* The implementation of NN using keras is quite easy compared to that in tensorflow.

**Conclusion:**

The neural network is applied to the dataset and 83% accuracy is achieved.

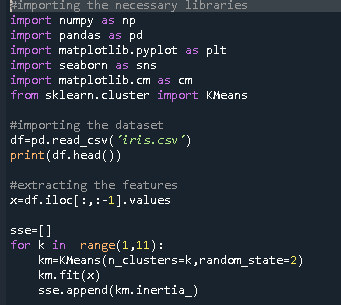
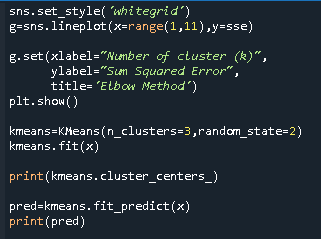
**TASK 16: PERFORM K-MEANS CLUSTERING ON THE DATA**

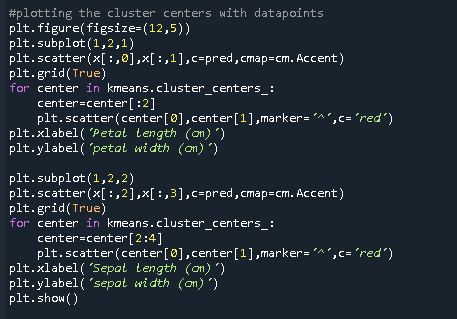
**AIM :**To perform k-means clustering on the given dataset.

**Concept and Dataset Description :**

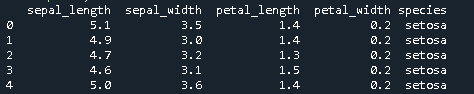
* K-means clustering is an unsupervised machine learning where we try to group the data elements based on similarity.
* K-means works on the concept that the similar items are closer to each other and dissimilar items are farther from each other.
* So, it is a distance based approach in the concept of clustering.
* The centroids of the k clusters are randomly initialized from the data elements and the distances from each data points are calculated.
* The datapoint with lower distance from the centroid are assigned to that cluster, and centroids are recalculated by comupting the average of the values in the cluster.
* The distances are computed again and the reassignment takes place. This process is repeated until no re-assignment takes place.
* The entire clustering process is applied on iris dataset for this particular task.

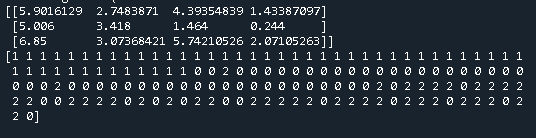
**Source Code:**

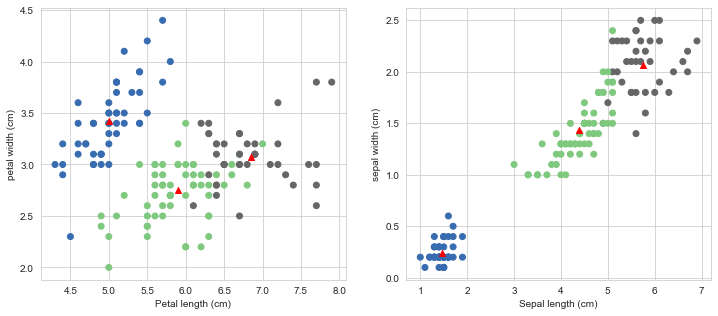
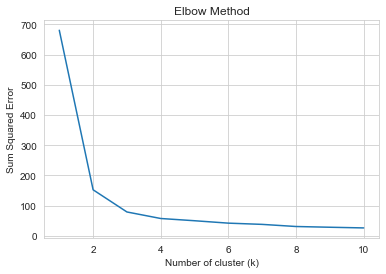
** **

****

**Outputs:**

****

****



**Observations:**

* The elbow method was chosen to choose the optimal k value which gives us the optimal number of clusters and it is found to be 3 clusters.
* The clusters are visualized using the scatter plots and we can identify three different clusters with their centroids marked with red.

**Conclusion:**

Elbow method is chosen to pick up the optimal number of clusters because the choice of K value is very important as K means clustering is computationally intensive.