**INTERVIEW QUESTION & SOLUTION**

**QUESTION 1:**

About the data & exercise:

* Columns 0 – 150 represent input variable and ‘target’ represents target variable
* Goal is to use the input variables to correctly identify or predict target variable
* Usage of Python and Jupyter notebook for completing this exercise is preferred but if you are not comfortable, feel free to provide source code and summary in email
* Please summarize your thought and analytics under following sections
  + Any pre-analytics steps and your understanding of data before you perform the analytics
  + Your choice of analytical algorithms, various steps taken during analytics, any comparisons between other algorithms and your understanding of how this model is performing
  + Your final choice of model and summary.

**ANSWER 1:**

* I have been given a dataset in the form of a csv file containing 150 input Variables & 1 Target Variable.
* Goal is to use the input variables to correctly identify or predict target variable.
* Detailed Summary of my Analytics needs to be Provided.
* Please find the Detailed Report below & My Report along with complete Source Codes & other Analytical Outputs have been uploaded in my GitHub Link and Google Drive Link mentioned below.
* **I HAVE CREATED A DETAILED REPORT AS BELOW CONTAINING ALL THE STEPS FOLLOWED , DETAILED OUTPUTS OF MY ANALYSIS & SUMMARY.**

https://github.com/aishwaryar5/MICRON-DATA-SCIENCE-SOLUTION-AISHWARYA-INTERVIEW-ANSWERS.git

**MY APPROACH:** I will be following a Detailed step-by-step approach to build my Predictive Model & Implement the Solution for above Data Science Question, the complete details are provided below in the form of a Report.

|  |  |
| --- | --- |
| **S.NO** | **TITLE** |
| 1. | OBJECTIVE OF THE PROJECT,BUSINESS UNDERTANDING, PROJECT DESIGN & PLANNING | |
| 2. | COMPLETE STEPS INVOLED | |
| 3. | DATA UNDERSTANDING , CLEANING, PREPARATION & TRANSFORMATION | |
| 4. | EXPLORATORY & STATISTICAL DATA ANALYSIS | |
| 5. | PREDICTIVE MODELLING, MODEL SELECTION, JUSTIFICATION,  VALIDATION & HYPER - PARAMETER TUNING | |
| 6. | KEY FINDINGS & SUMMARY | |
| 7. | REFERENCES | |
| 8. | APPENDIX | |

**TABLE OF CONTENTS**

**1.BUSINESS OBJECTIVE** , **OBJECTIVE OF THE PROJECT,BUSINESS UNDERTANDING, PROJECT DESIGN & PLANNING:**

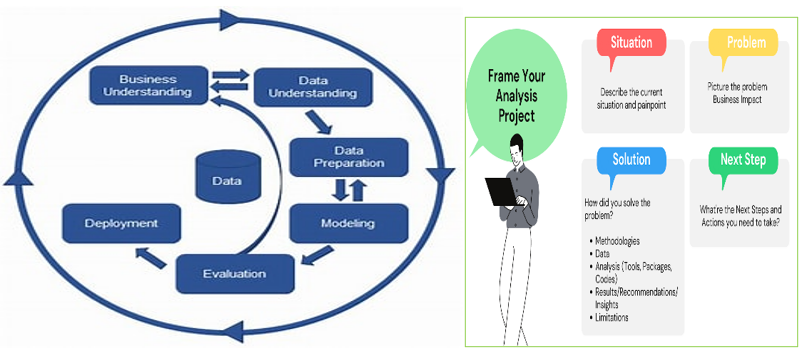
For every data science project understanding the business problem & objective is very crucial (since here I am working on an Interview Question, I am skipping this part for now as business scenario and details are not provided ). It is a good and essential practice while solving data science problems , we need to have: 1. thorough understanding and sufficient knowledge about business & technical problems, 2.do research work about business problem & objective , 3.plan and design a proper project methodology before implementation, only then we will be able to provide the right solution and build a good data science project.

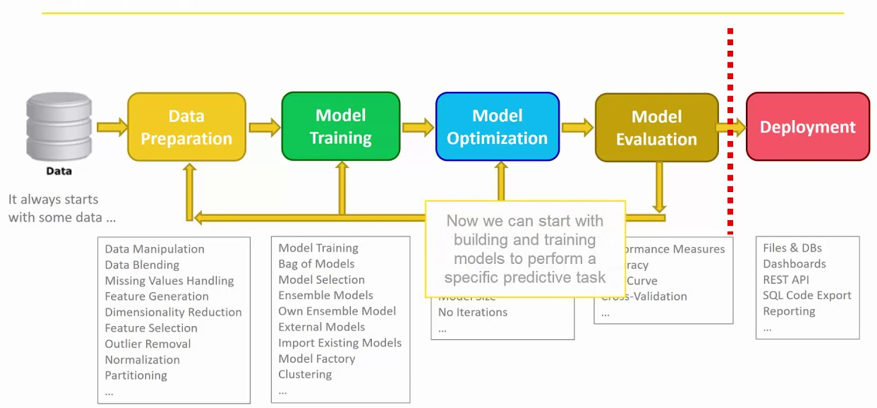
BUSINESS OBJECTIVE

BUSINESS UNDERSTANDING

PLANNING & IMPLEMENTATION

TECHNICAL OBJECTIVE & PROBLEM



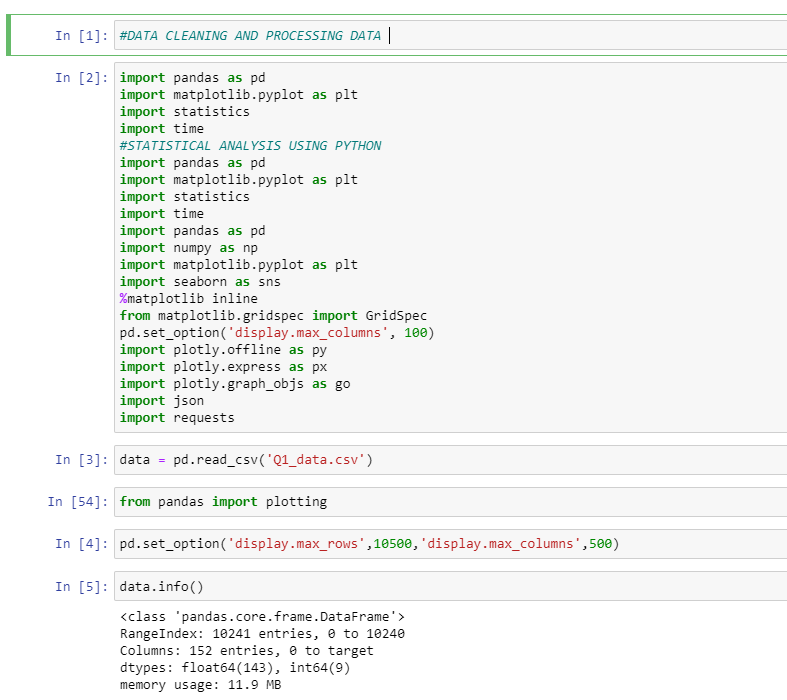
****

**2.COMPLETE STEPS INVOLVED:**

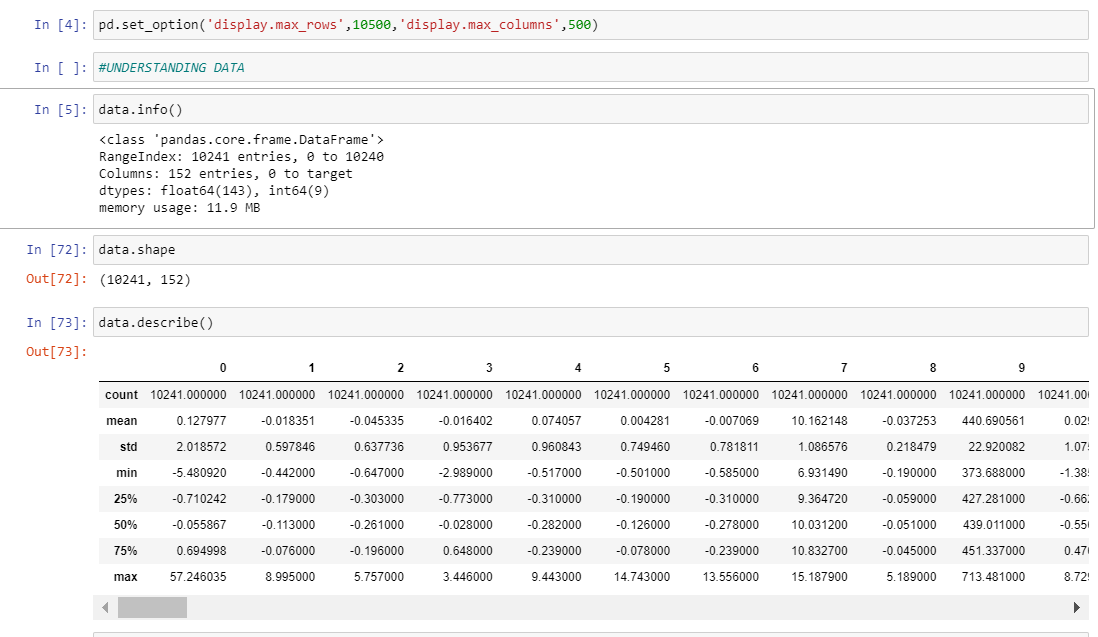
* Data Understanding, Cleaning, Preparation & Transformation
  + - * Merging Datasets
      * Handling Missing Values
      * Handling Null Values
      * De-Duplication
      * Handling Outliers (if-needed)
      * Data Type Conversion(if-needed)
      * Data Quality Analysis
      * Handling Multi-Collinearity (if-needed)
      * Feature Engineering(if-needed)
      * Feature Selection (or) Dimension Reduction(if-needed)
      * Loading Data
* Initial Statistical Analysis & Inferences
  + - * Basic Statistical Analysis to Obtain Insights about Data
* Initial Descriptive/ Exploratory Analysis & Inferences
  + - * Graphical Analysis to Obtain Insights about Data
* Predictive Modelling
  + - * Choosing Suitable Modelling Algorithms for Dataset & Justification
      * Following Pre-Data Modelling Requirements & Steps
      * Splitting Data into Train & Test
      * Steps Taken During Model Building using Each Algorithm
      * Model Validation/Testing & Scoring
      * Comparison of Algorithms & Justification
      * Model Performances Report
      * Choosing the Final best Algorithm/Model & Justification
      * Model Re-validation/Testing or Scoring
      * Performance Optimization & Prescriptive Analytics for chosen model
      * Model Re-validation/Testing or Scoring
      * Final Model Deployment
* Report Containing the Model Output & Key Insights Obtained.
* Data Governance & Protection Applied on Report & Publish Report.
* Summary & Conclusion of Analytics & Predictive Model Building performed .

**1.** **DATA UNDERSTANDING, CLEANING, PREPARATION ,TRANSFORMATION ,STATISTICAL , EXPLORATORY DATA ANALYSIS& INFERENTIAL ANALYTICS**

1. IMPORTING THE NECESSARY LIBRARIES & IMPORTING CSV FILE:



1. UNDERTSANDING DATA :



FINDINGS: DATA CONTAINS 10241 ROWS AND 152 COLUMNS

ALL THE DATA ARE NUMERICAL DATA

WE HAVE FLOAT64 🡪143 & INT64🡪 9

WE IDENTIFIED THE MEAN, MEDIAN, STD DEVIATION,MIN & MAX VALUES OF EACH COLUMN DATA

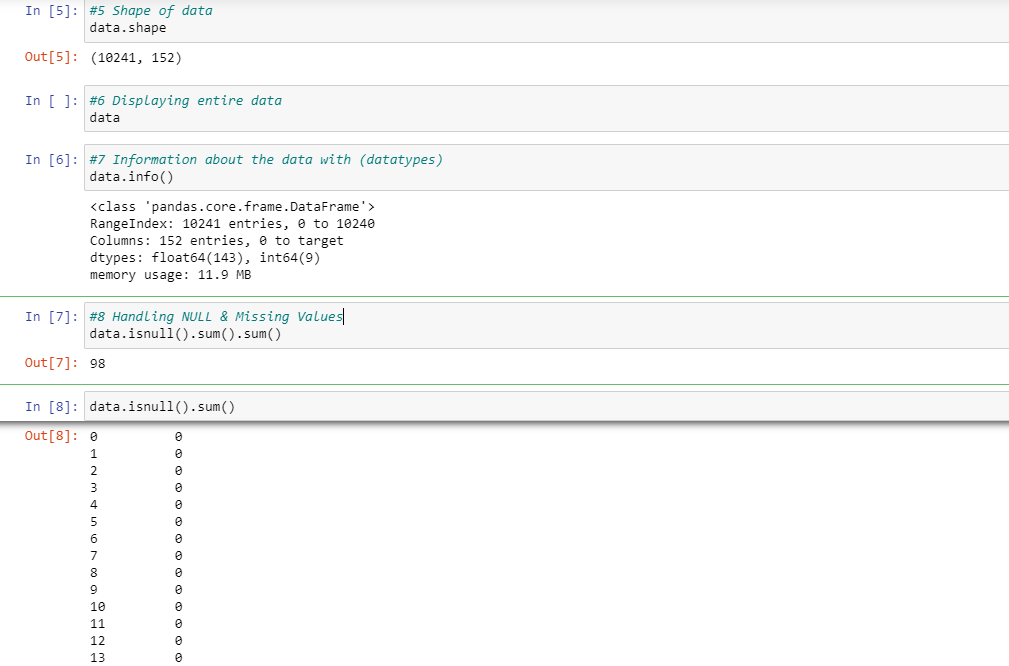
* 3. WE HANDLE WARNINGS, SET DISPLAY PROPERTIES & LOAD DATA



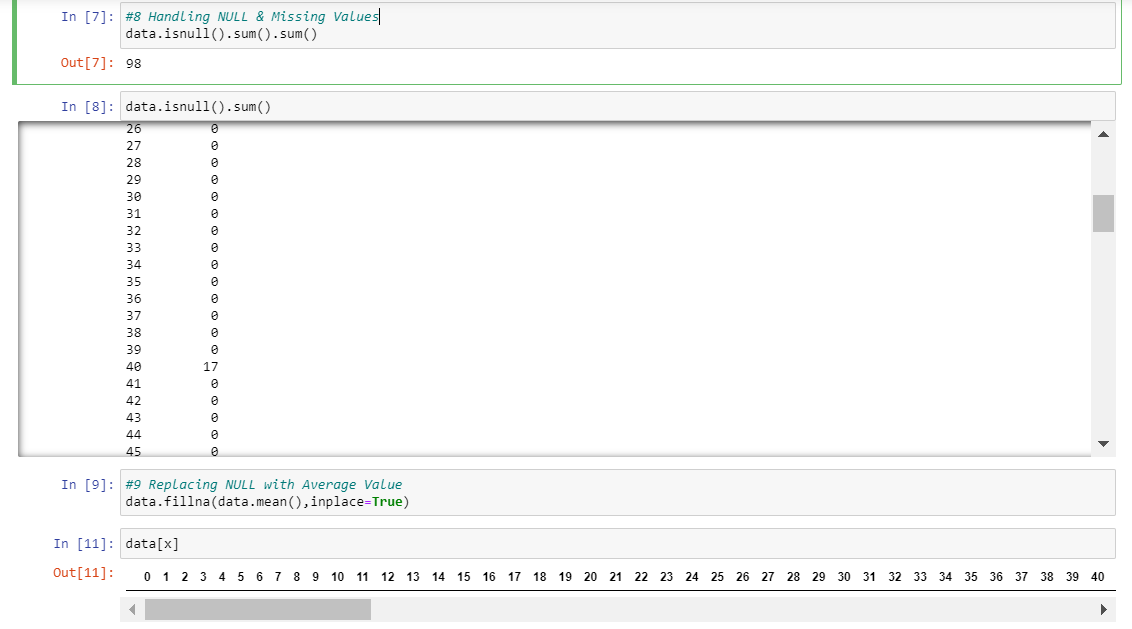
* WE GET THE DATA INFO AFTER HANDLING NULL & DUPLICATE VALUES: WE GET DATA INFO WE SEE THAT DATA HAS:

10241 ROWS & 152 COLUMNS

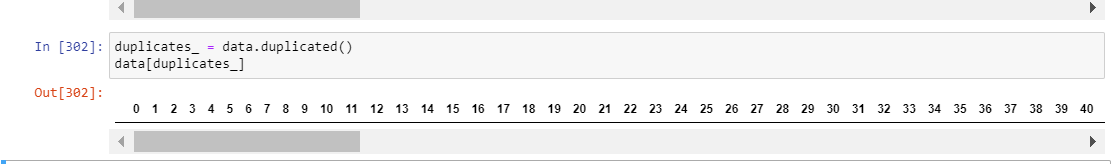
143 FLOAT & 9 INTEGER VALUES



* WE REPLACE THE MISSING VALUES WITH AVERAGE VALUE.( THERE ARE DIFFERENT METHODS TO REPLACE MISSING\NULL VALUE -> 1.REPLACING WITH ZERO 2.REMOVING THE MISSING\NULL VALUE 3.USING AVERAGE VALUE , IN OUR CASE USING AVERAGE VALUE IS MOST SUITABLE.



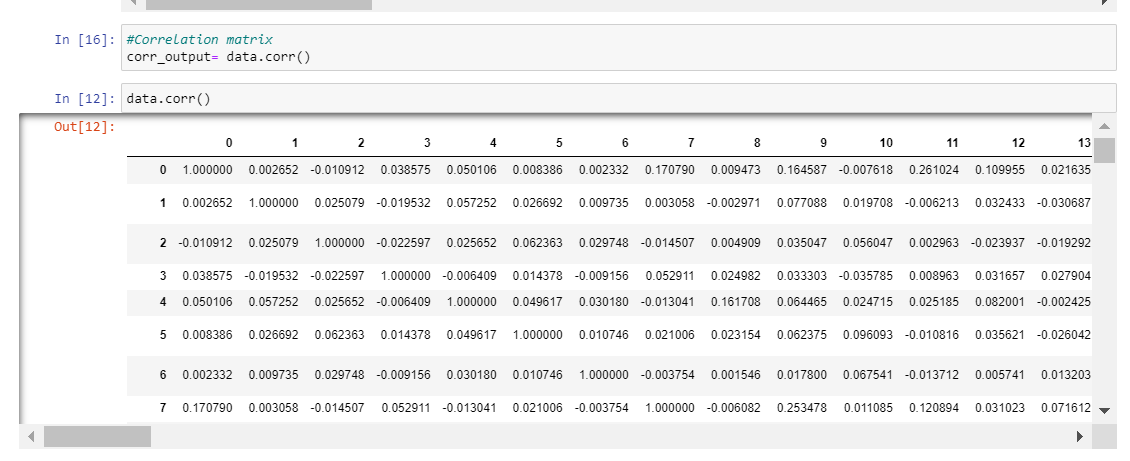
* WE PERFORM DATA DE-DUPLICATION AS SEEN BELOW WE DO NOT FIND ANY DUPLICATE VALUES IN THE CSV FILE DATA.



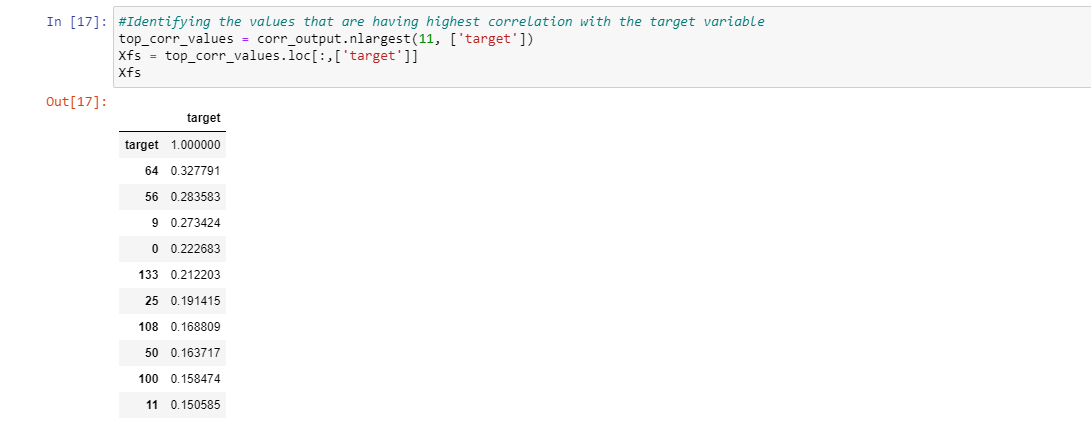
1. Handling Outliers, Data Type, Multi Collinearity :

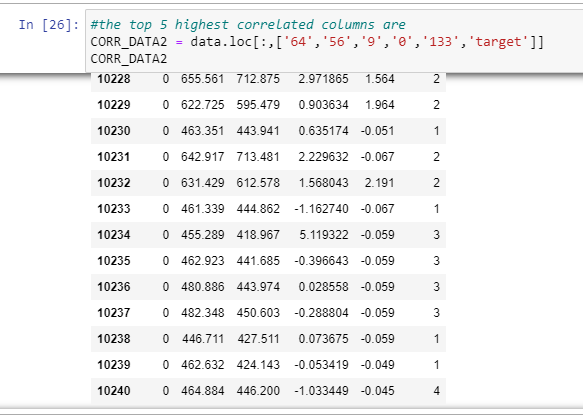
Outliers can cause a lot of distortion in the Data ; hence they need to be removed but at the same time in sensitive data sets like MES, INDUSTRIAL DATA it is not advisable to remove outliers without having complete business knowledge. Type conversion is not required as data contains float and int , as we are going to implement a multi -class classification model with PCA multi-collinearity handling is not required.

1. CORRELATION ANALYSIS: WE FIND THE CORRELATION AMOG ALL THE INPUT VARIABLES AND THE TARGET VARIABLES.

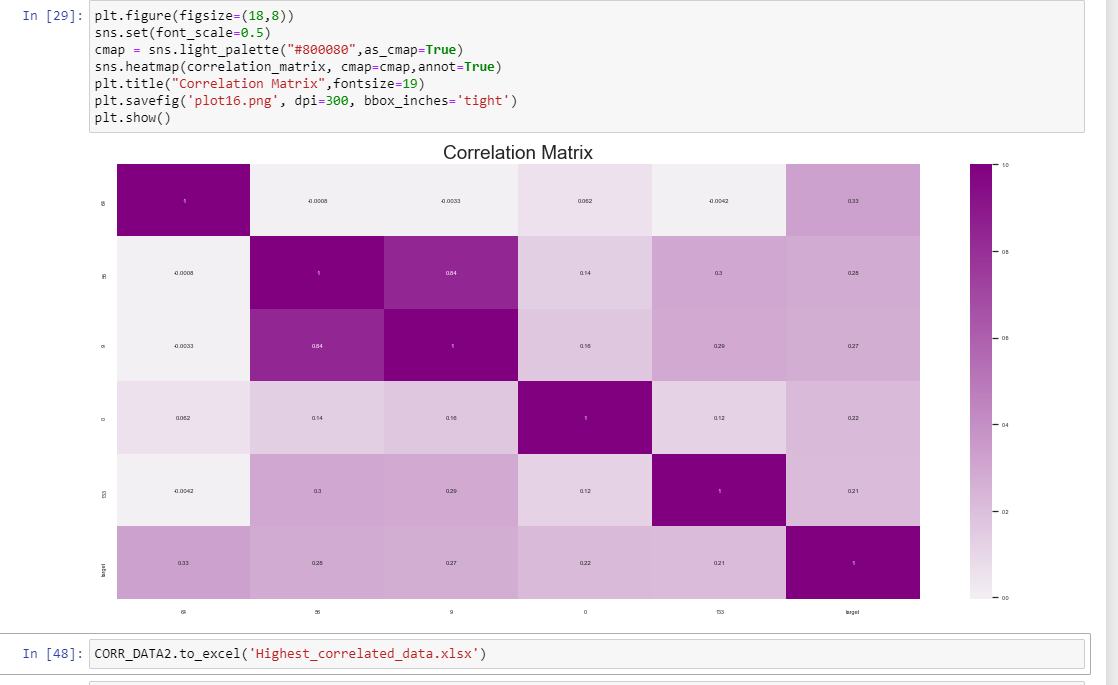


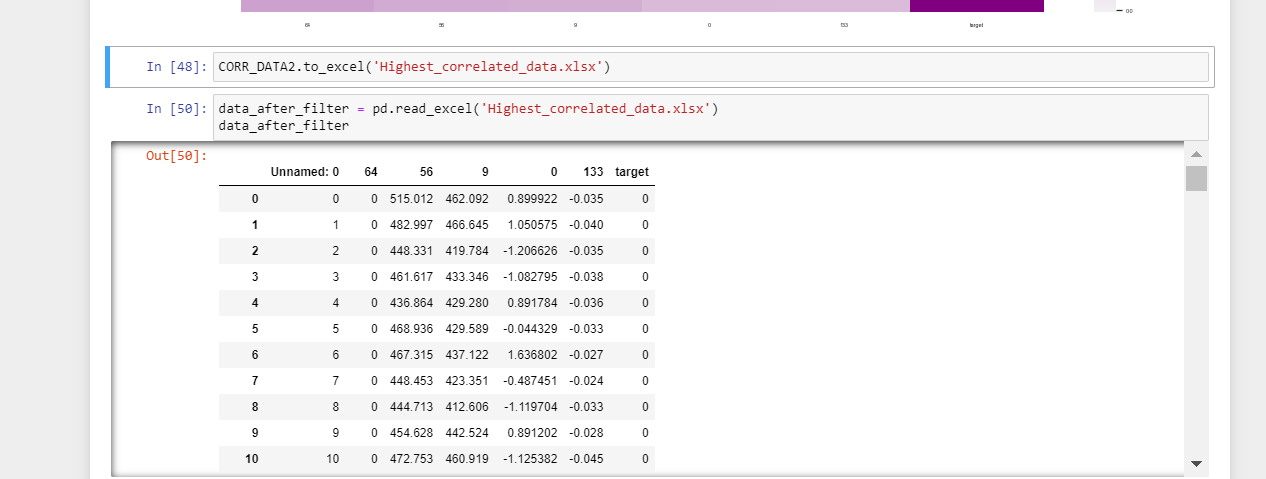
SINCE OUR FOCUS IS ON THE TARGET VARIABLE AND WE NEED TO PERFORM DESCRIPTIVE AND PREDICTIVE MODELLING FOR THE TARGET VARIABLE ATTRIBUT EWE NEE DTO CONSIDER THOSE ATTRIBUTES WHICH HAVE GOOD CORRELATION WITH OUR TARGET VARIABLE ‘target’;





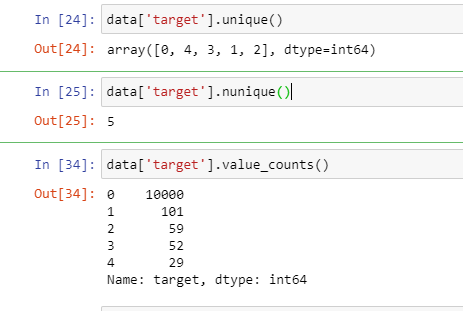
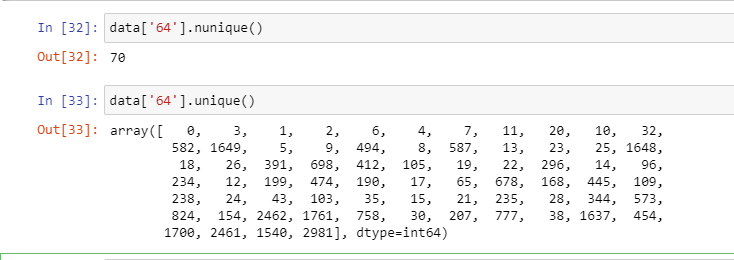
WE FIND THAT THE ATTRIBUTES 65,56,9,0,133 HAVE THE HIGHEST CORRELSTION WITH OUR TARGET VARIABLEWEWILL CONSIDER THEM FOR FURTHER ANALYSIS.

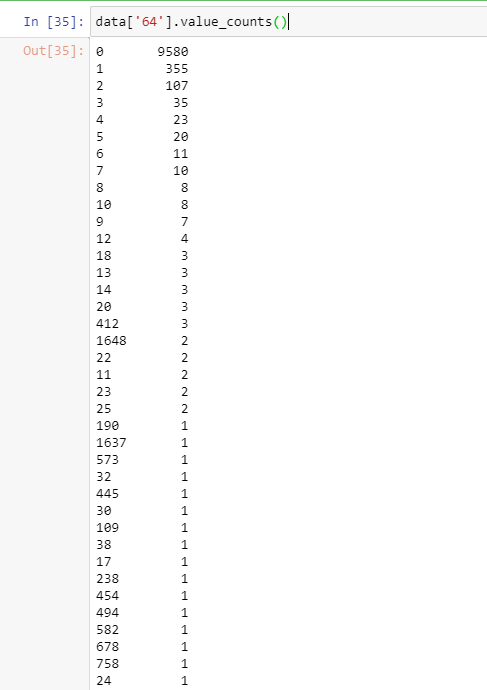




STATISTICAL AND INFERENTIAL ANALYSIS:

We perform a sample ( for few columns) data inferencing for target variable and attribute 64 which ahs maximum correlation with target:

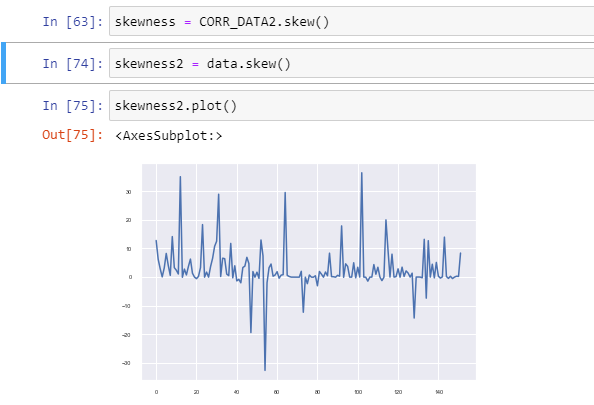


FINDINGS: WE IDENTIFIED THE TOP 5 HIGHEST CORRELATED ATTRIBUTES

WHILE GOING THROUGH THE NATURE OF DATA WE IDENTIFY THAT THERE ARE LESS NUMBEROF UNIQUE VALUES AND MANY VALUES ARE REPETITIV EIN NATURE, THIS INDICATED THAT THERE MIGHT BE A POSSIBILITY OF SEASONALITY PATTERN IN THE DATA .

**SKEWNESS AND KURTOSIS:**

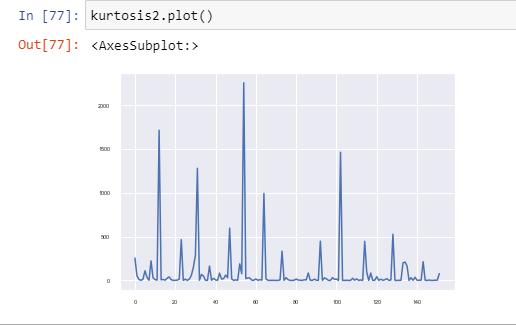
Skewness is a measure of symmetry, distribution of data set, it checks if data is symmetric i.e. if it looks the same to the left and right of the centre point.



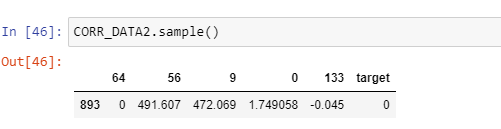
WE SEE THAT DATA HAS A PATTERN BUT ALSO HAS LOT OF NOISE ASSOCIATED WITH IT , REMOVAL OF NOISE AND DATA PROCESSING WILL GIVE A CLEARER OUTPUT. DATA HAS A PATTERN SKEWNESS.

**KURTOSIS:**

It gives the measure of tailing of data  whether the data are heavy-tailed or light-tailed relative to a normal distribution



**DATA SAMPLING & SCATTER PLOT:**



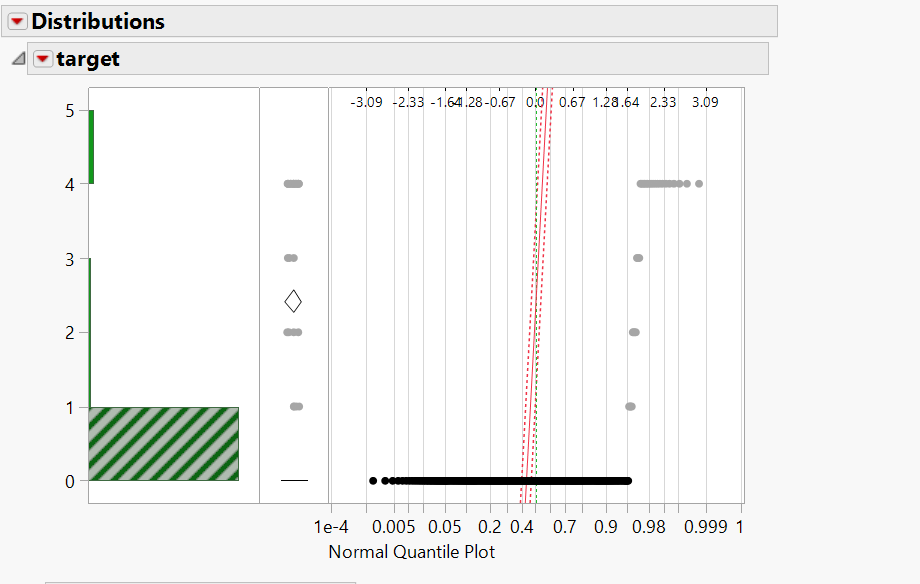
Above output returns a particular length list of values chosen from attribute list Used for random sampling without replacement. (in case we have a requirement for random sampling & decomposition)**SCATTER PLOT:**

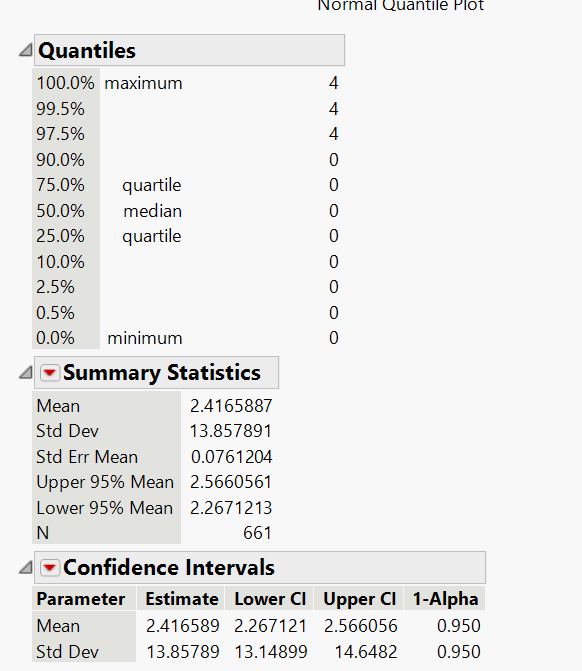
SCATTER PLOT HAS BEEN PLOTTED FOR THE 5 ATTRIBUTES HAVING HIGHEST CORRELATION WITH THE TARGET VARIABLE .

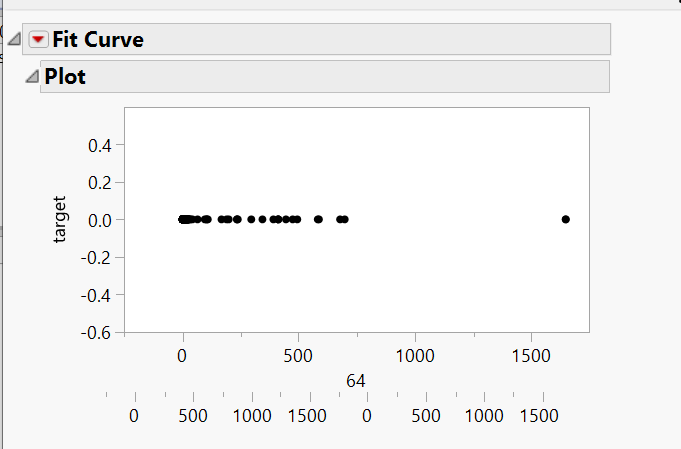
CONCLUSION: I PERFORMED BASIC STATISTICAL & EDA ANALYSIS ON DAT AAND OBTAINED SOME INFERENCES I CAN PERFORM FURTHER DETAILED ANALYSIS BASED ON THE REQUIREMENT .



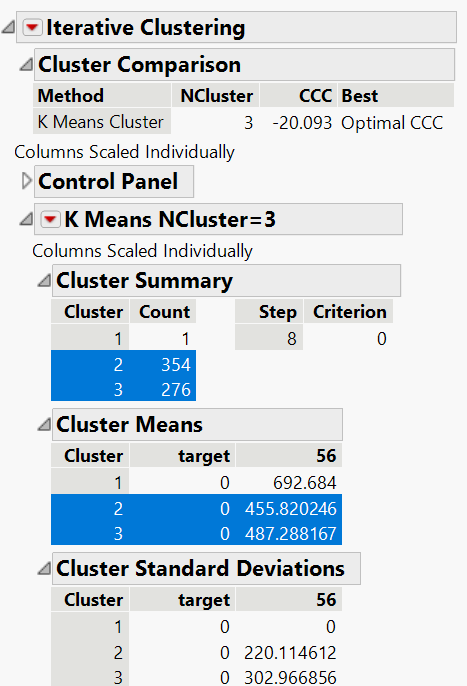
DISTRIBUTION OF TARGET VARIABLE W.R.T TOP 3 HIGHEST CORRELATED ATTRIBUTES 64,56,9:







K-MEANS CLUSTER ANALYSIS:



**2.PREDICTIVE MODELLING, MODEL SELECTION,VALIDATION & HYPER PARAMETER TUNING**

**STEP 1: ANALYZING THE DATASET :**

**1] DATA CONTAINS 151 INPUT (X) ATTRIBUTES AND 1 TARGET (Y) ATTRIBUTE .**

**2] ALL THE COLUMNS FROM 0 TO 150 ARE INTEGER OR FLOAT DATA.**

**3] IT IS A {X🡪Y, NUMERICAL 🡪 NUMERICAL} TYPE OF MODEL.**

**4] TARGET VARIABLE DATA IS NOT CONTINOUS.**

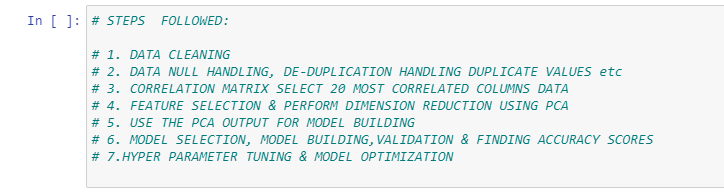
**5] TARGET VARIABLE DATA IS NOT BINARY , IT HAS MULTIPLE NUMERICAL VALUES.**

**6] HENCE WE HAVE MULTI-CLASS CLASSIFICATION SCENARIO AND WE NEED TO USE MULTI CLASS CLASSIFICATION ALGORITHMS TO BUILD MODEL AND PREDICT TARGET.**

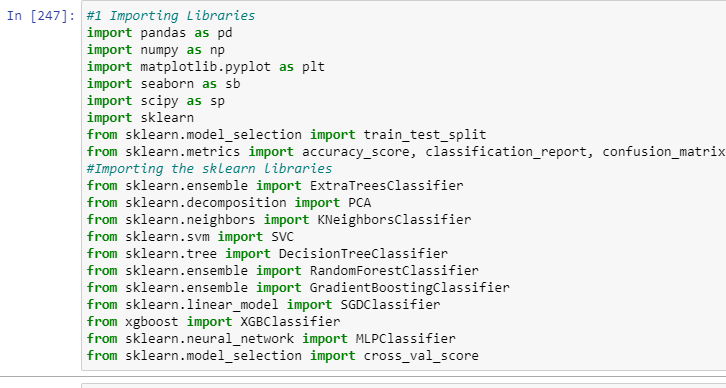
**CONCLUSION : WE NEED TO BUILD A MULTI-CLASS CLASSIFICATION MODEL USING MULTI-CLASS CLASSIFIER ALGORITHMS.**

**STEP 2: COMPLETE STEPS FOR MODEL BUILDING ,SELECTION,VALIDATION & OPTIMIZATION & CODE IMPLEMENTATION USING JUPYTER NOTEBOOK PYTHON:**

* WE FOLLOW THE BELOW STEPS



* WE IMPORT THE REQUIRED LIBRARIES



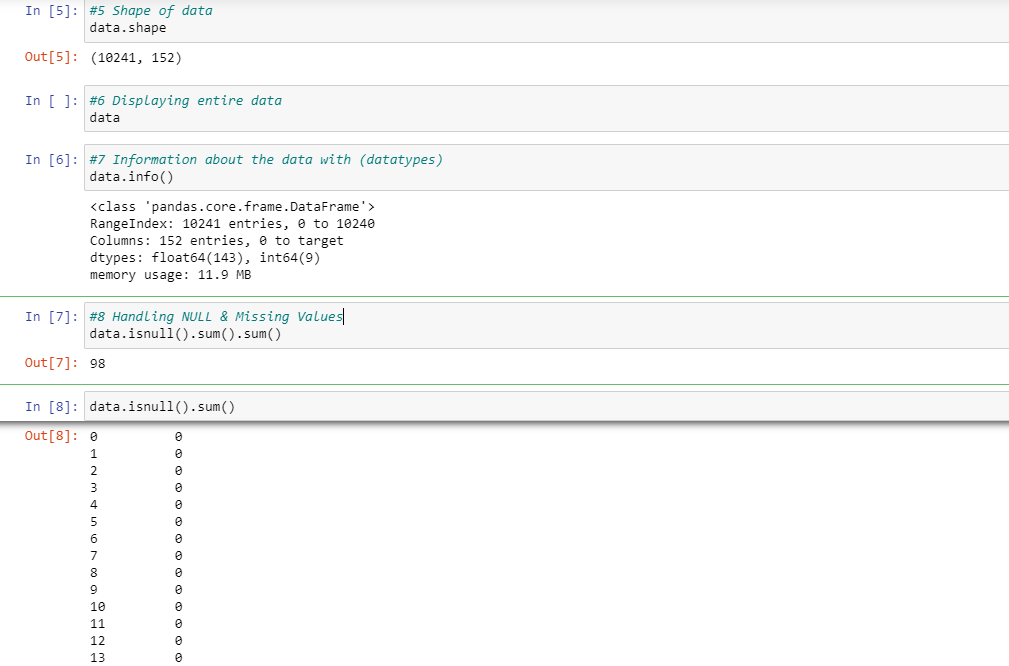
* WE HANDLE WARNINGS, SET DISPLAY PROPERTIES & LOAD DATA



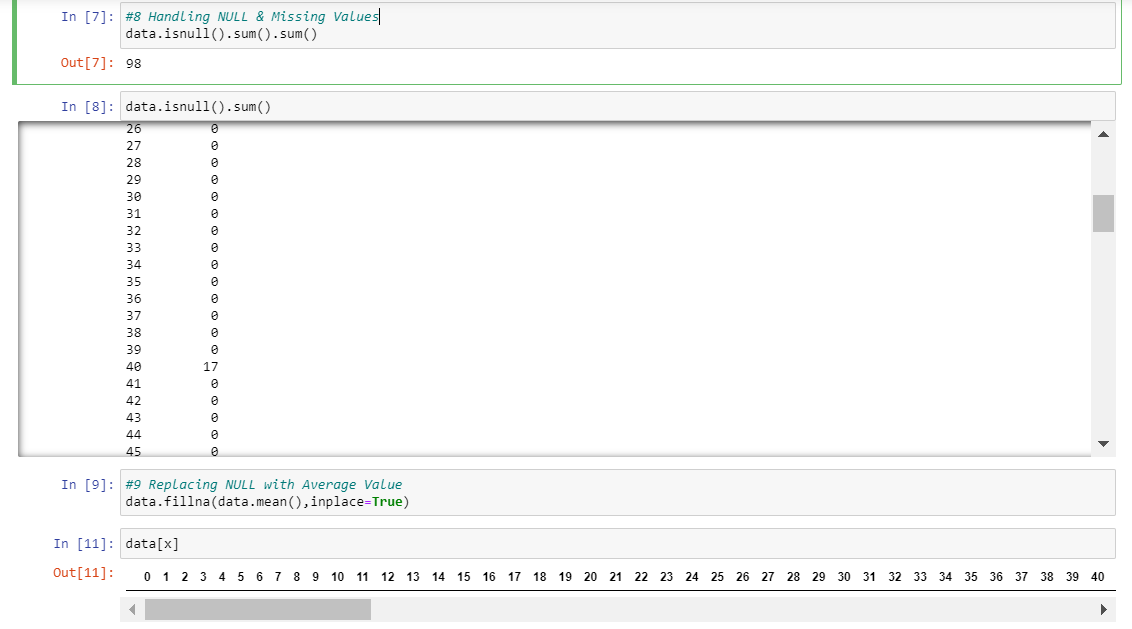
* WE GET THE DATA INFO AFTER HANDLING NULL & DUPLICATE VALUES: WE GET DATA INFO WE SEE THAT DATA HAS:

10241 ROWS & 152 COLUMNS

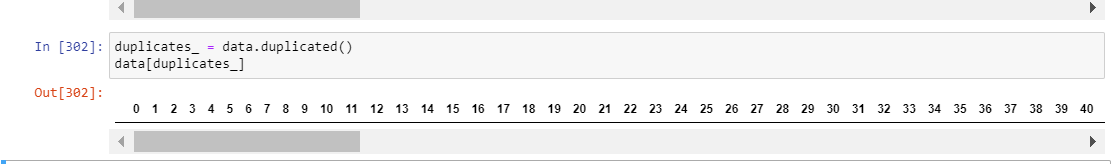
143 FLOAT & 9 INTEGER VALUES



* WE REPLACE THE MISSING VALUES WITH AVERAGE VALUE.( THERE ARE DIFFERENT METHODS TO REPLACE MISSING\NULL VALUE -> 1.REPLACING WITH ZERO 2.REMOVING THE MISSING\NULL VALUE 3.USING AVERAGE VALUE , IN OUR CASE USING AVERAGE VALUE IS MOST SUITABLE.



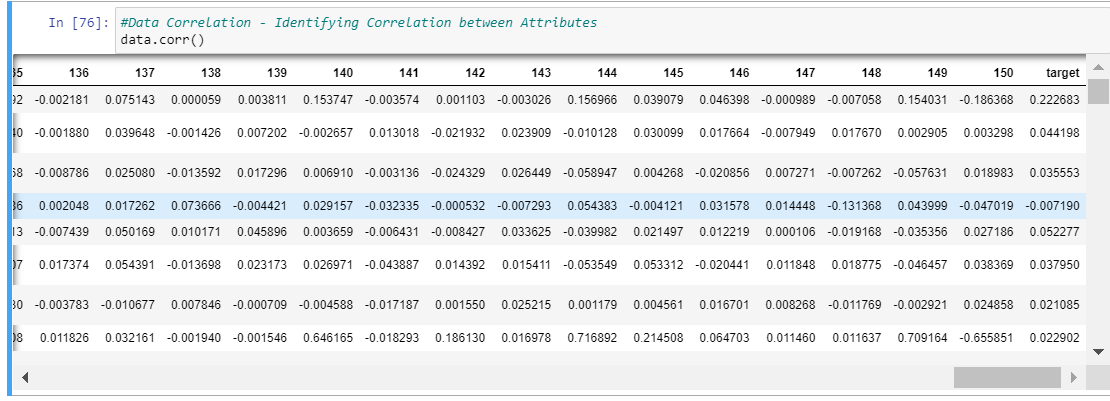
* WE PERFORM DATA DE-DUPLICATION AS SEEN BELOW WE DO NOT FIND ANY DUPLICATE VALUES IN THE CSV FILE DATA.



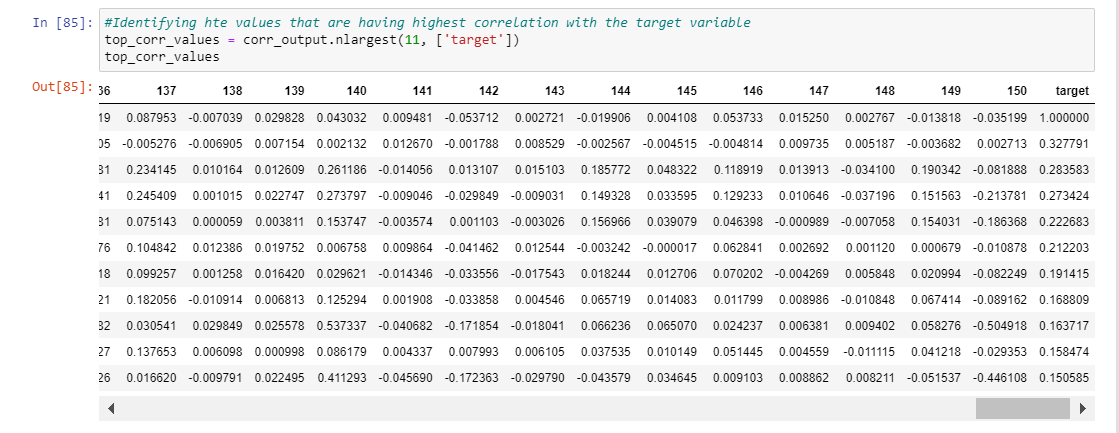
* AS WE SEE OUR DATA HAS A LARGE NUMBER OF INPUT VARIABLES , AS WE HAVE A LARGE AMOUNT OF DATA , WITH MANY VARIABLES IT IS DIFFICULT TO

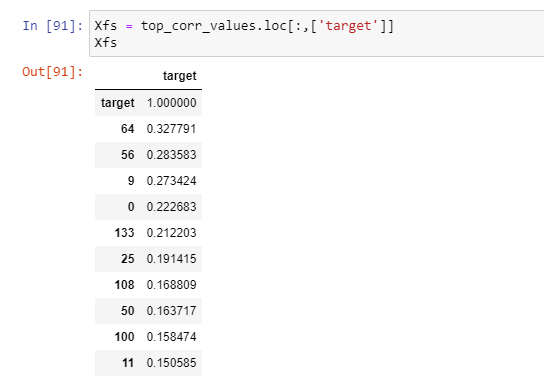
– TO MANIPULATE AND TRANSFORM THIS DATA – VISUALIZE & MODEL THIS DATA – TO PERFORM A MEANINGFUL ANALYSIS AND CREATE USEFUL MODELS -- TOO MANY ATTRIUTES MAY CAUSE OVERFITTING OR POOR MODEL PERFORMANCE ETC. WE NEED TO FOLLOW FEATURE SELECTION OR DIMENSION REDUCTION TECHNIQUE.-- ONE OF THE MAJOR APPROACHES WE USE TO PROCESS SUCH DATA IS DIMENSION REDUCTION – THE AIM IS TO EFFECTIVELY REDUCE THE NUMBER OF VARIABLES UNDER ANALYSIS BY COMBINING TOGETHER THE ORIGINAL VARIABLES, WHILST MINIMIZING THE LOSS OF INFORMATION IN THIS PROCESS

* THERE ARE TWO METHODS:
  + - FEATURE SELECTION - Feature selection is for filtering irrelevant or redundant features from your dataset. It may features data that are not very well related to target data i.e. features that are redundant. Advantages are outputs are interpretable, disadvantages are data loss may occur.
    - DIMENSION REDUCTION(PCA) – In Dimension Reduction using Principal component analysis (PCA) linear combinations of the original features are created . The new features are orthogonal, which means that they are uncorrelated. Furthermore, they are ranked in order of their variance. Advantages Include better model performance, minimises chances of over-fitting and data loss is minimised, disadvantages include outputs are not interpretable.
* I WILL BE DEMONSTRATING BOTH PCA & FEATURE SELECTION, BUT I WILL BE USING PCA TO BUILD THE MODEL.
* WE FIND THE CORRELATION OF TARGET VARIABLE WITH THE INPUT VARIABLES; WE CAN USE THE TOP MOST CORRELATED DATA ATTRIBUTES AND DISCARD THE LEAST CORRELATED ATTRIBUTES.



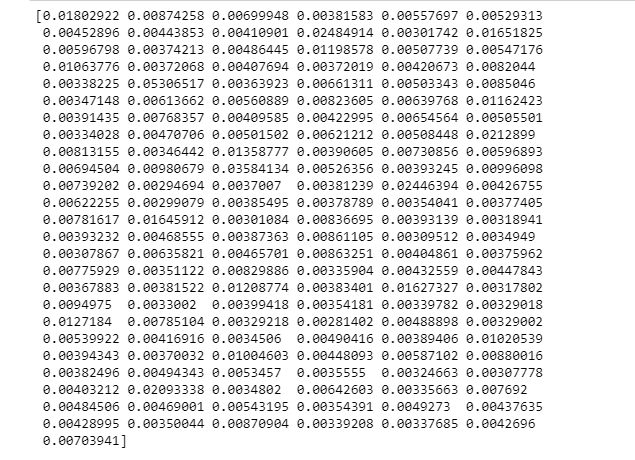


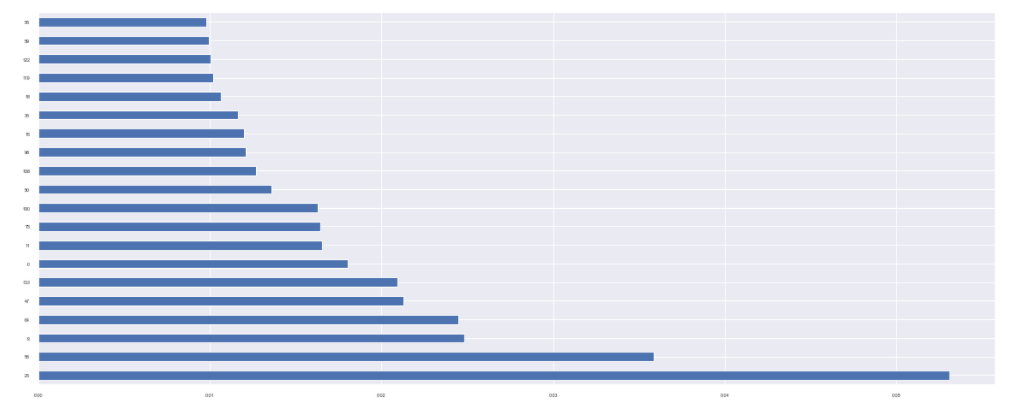


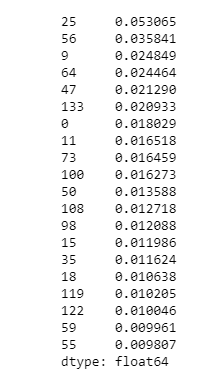
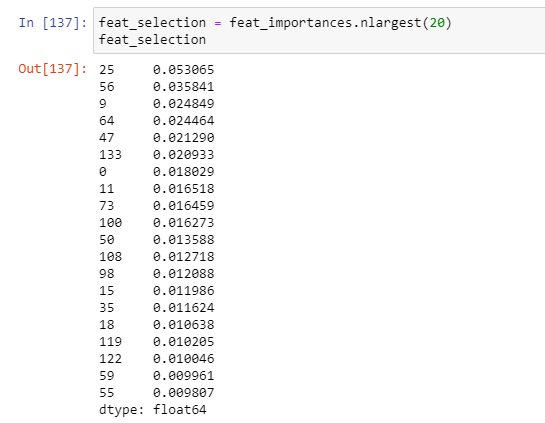


* FEATURE SELECTION USING EXTRA TREES

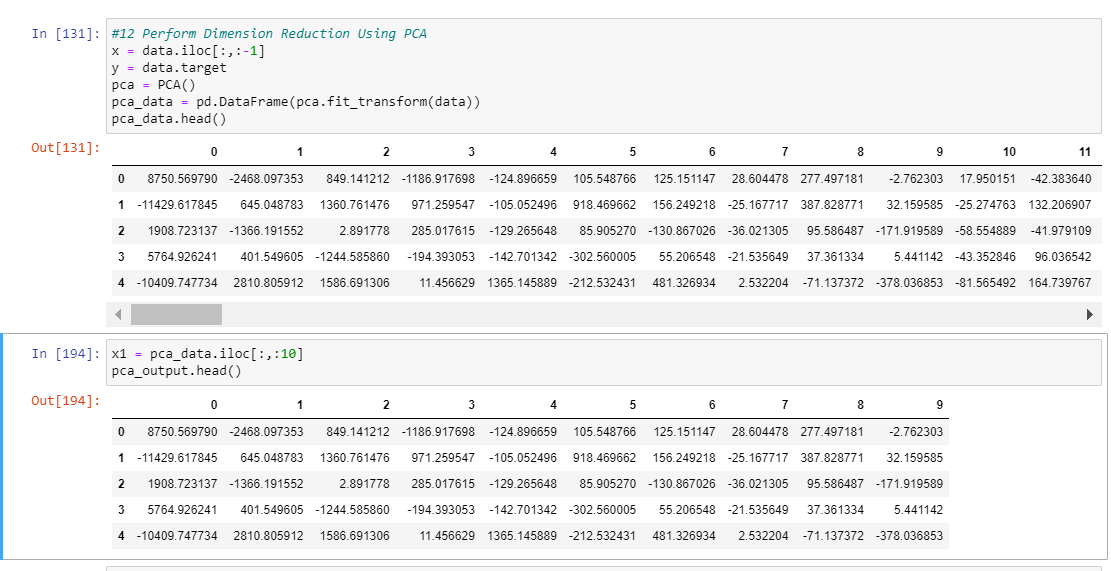


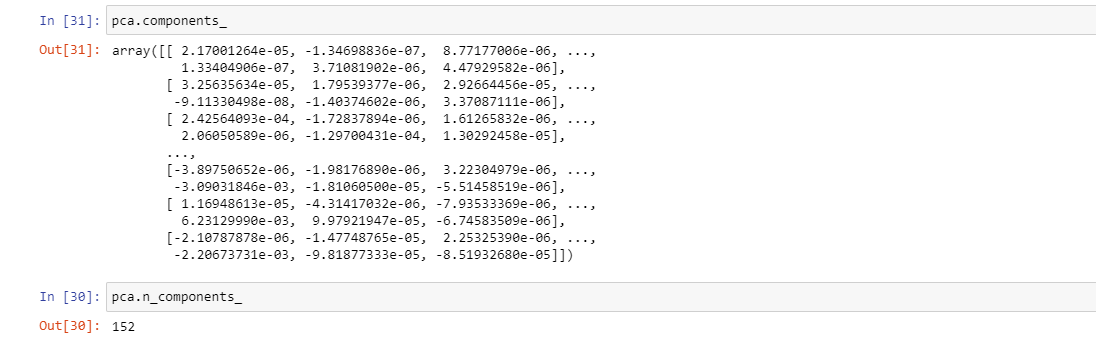




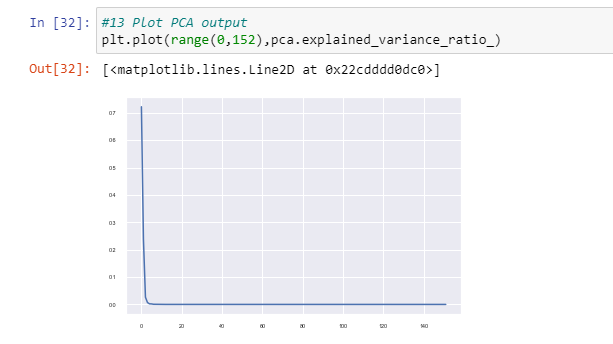
 

* PERFORMING DIMENSION REDUCTION USING PCA:





* WE STORE THE PCA() MODEL DIMENSION REDUCTION OUTPUT IN PCA\_DATA DATA FRAME
* WE SEE FROM THE PLOT THAT HOW THE VARIANCE RATIO CHANGES STEEPLY AFTER A CERTAIN POINT (AFTER 5TH VARIABLE IN THE X-AXIS)



* MODEL SELECTION, COMPARISON, JUSTIFICATION, EVALUATION, VALIDATION & HYPER PARAMETER TUNING:
* MODEL SELECTION & COMPARISON:
  + - I WILL BE CONSIDERING THE FOLLOWING MULTI CLASS CLASSIFIER ALGORITHMS TO BUILD THE MODELS:

--- K NEAREST NEIGHBORS (KNN)

--- SUPPORT VECTOR CLASSIFICATION (SVC)

--- DECISION TREE

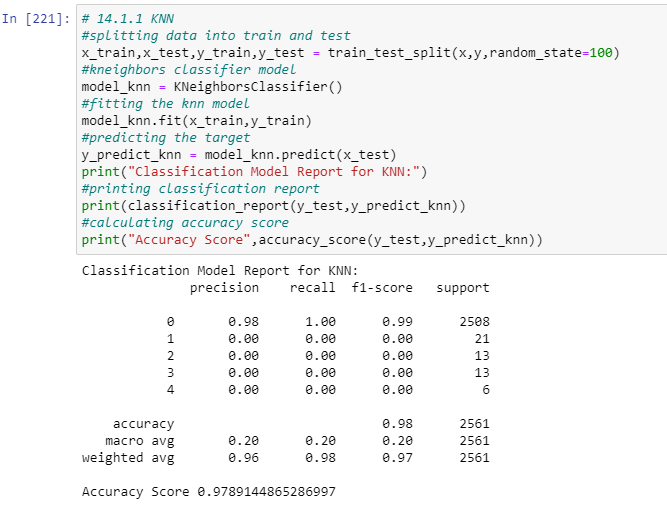
--- RANDOM FOREST

---GRADIENT BOOST

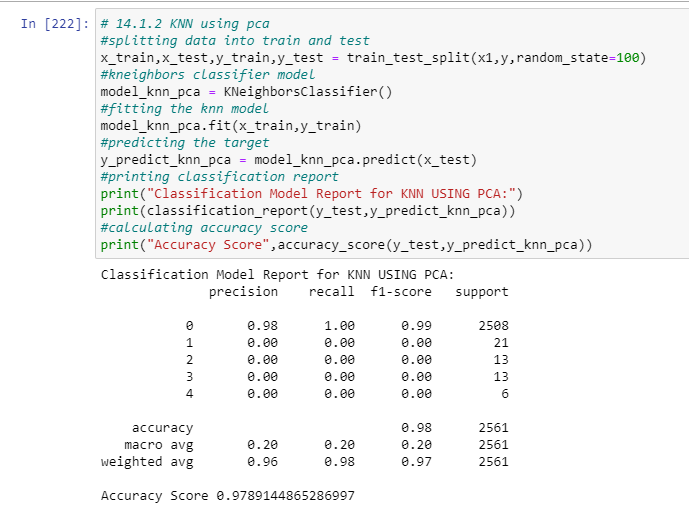
---XG BOOST

--- ARTIFICIAL NEURAL NETWORK (ANN)

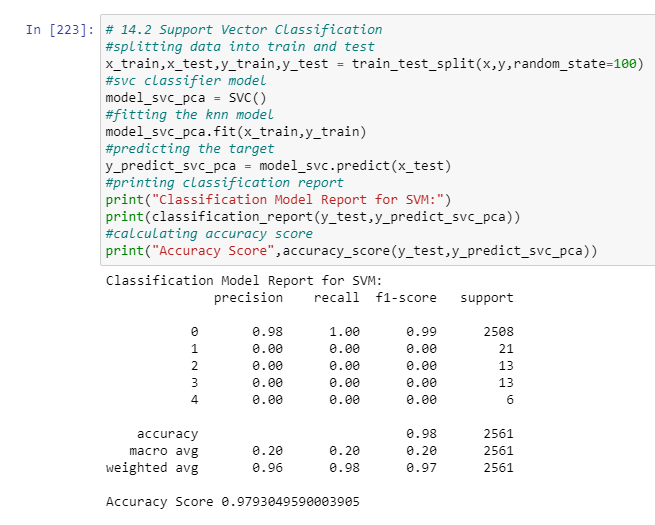
* I AM SHOWING THE MODEL BUILDING OUTPUT BY USING DATA BEFORE AND AFTER PCA TO DEMONSTRATE THE DIFFERENCE IN THE MODEL PERFORMANCE, IDEAL MEHTOD IS TO USE TH PCA DATA AS IT INCREASES THE MODEL PERFORMANCE AND PROVIDES AN OVERALL ENHANCED & OPTIMISES PERFORMANCE (SOMETIMES AT A SLIGHT COMPROMISE WITH THE ACCURACY AT CERTAIN INSTANCES BUT THEY ARE NOT VERY SIGNIFICANT & AS OVERALL PERFORMANCE OF MODEL IS BETTER, WE CAN STILL GO WITH PCA) . OVERALL MODEL PERFORMANCE WILL BE IMPROVED USING PCA WITHOUT SIGNIFICANT DATA LOSS.
* **KNN MODEL:**



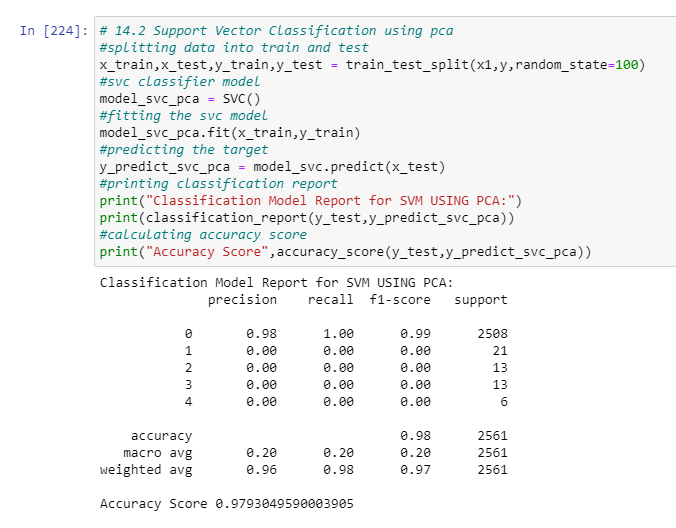
* **KNN FOR PCA DATA**



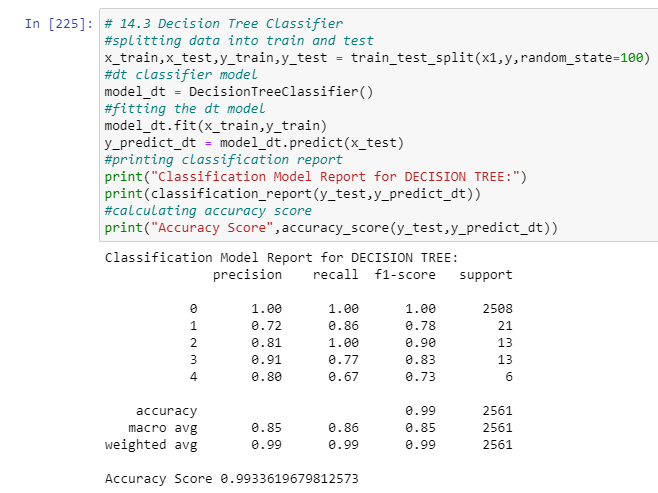
**#SVC MODEL:**



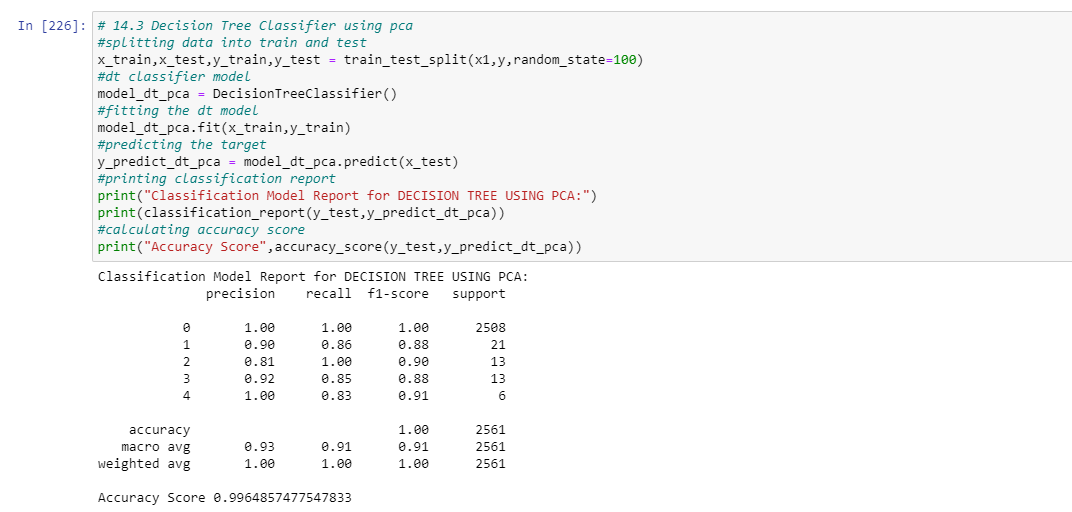
**# SVC FOR PCA DATA**



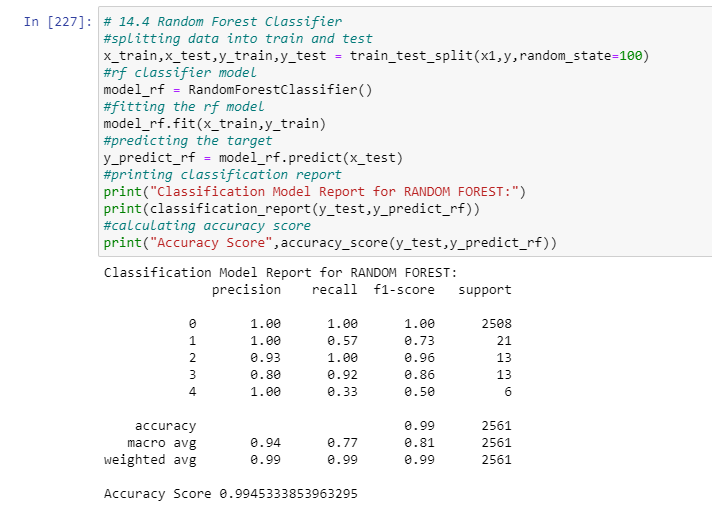
**#DECISION TREE MODEL:**



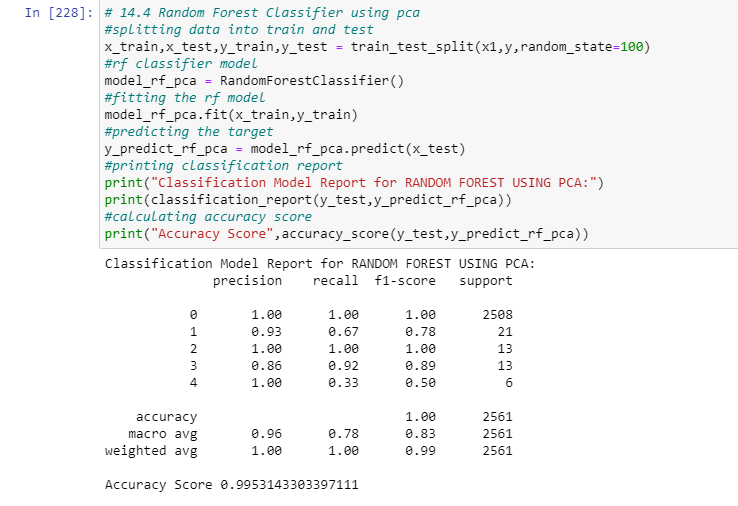
**#DECISION TREE MODEL USING PCA:**



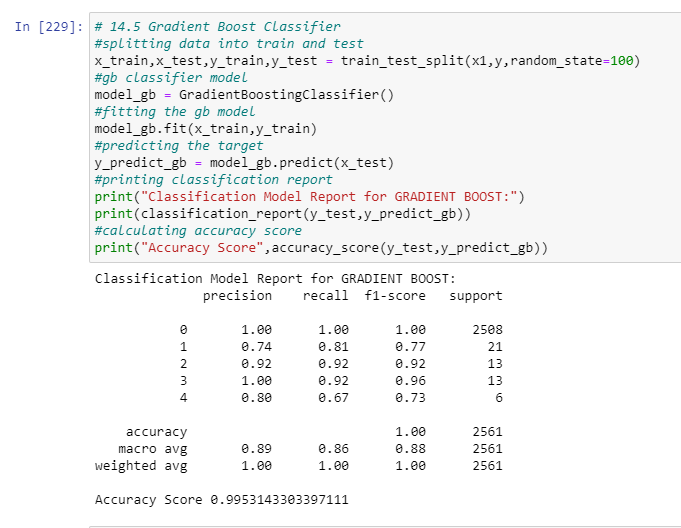
**#RANDOM FOREST MODEL:**



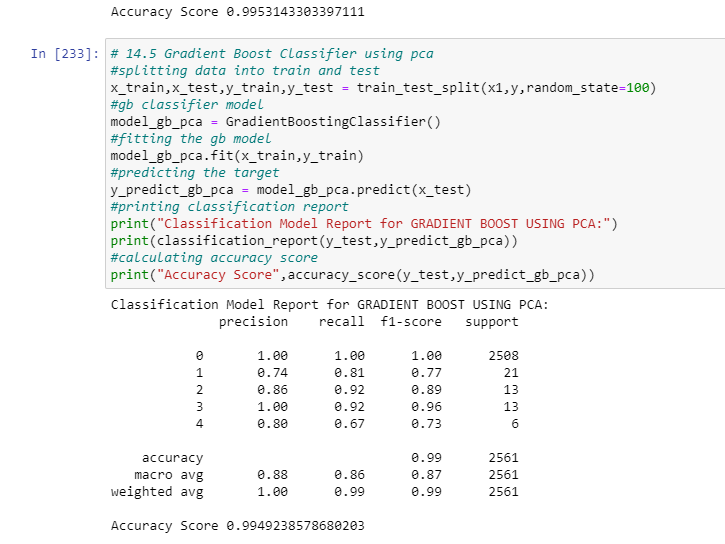
**#RANDOM FOREST USING PCA:**



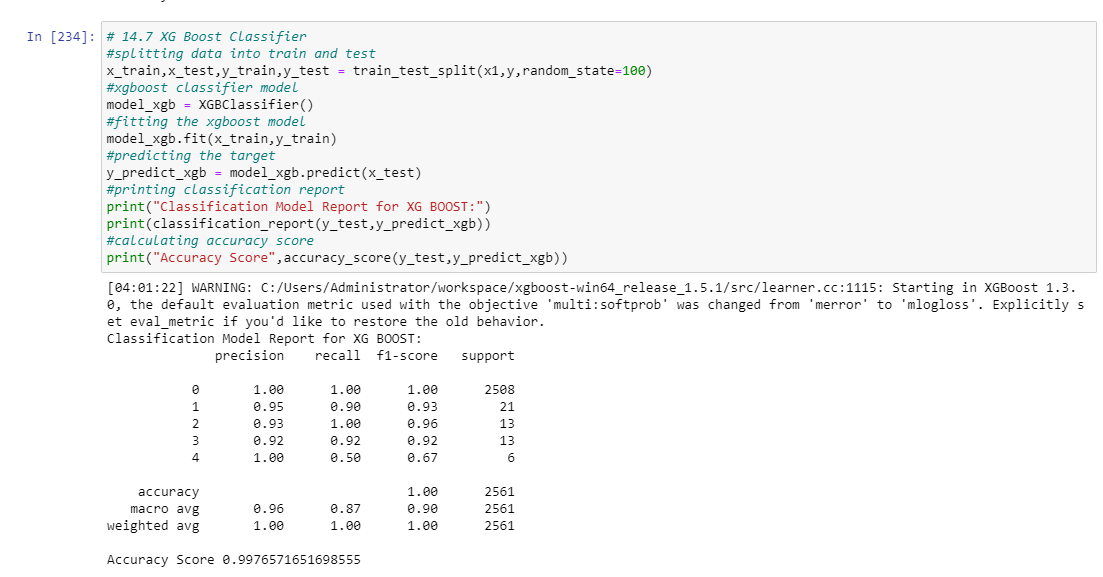
**#GRADIENT BOOST MODEL:**



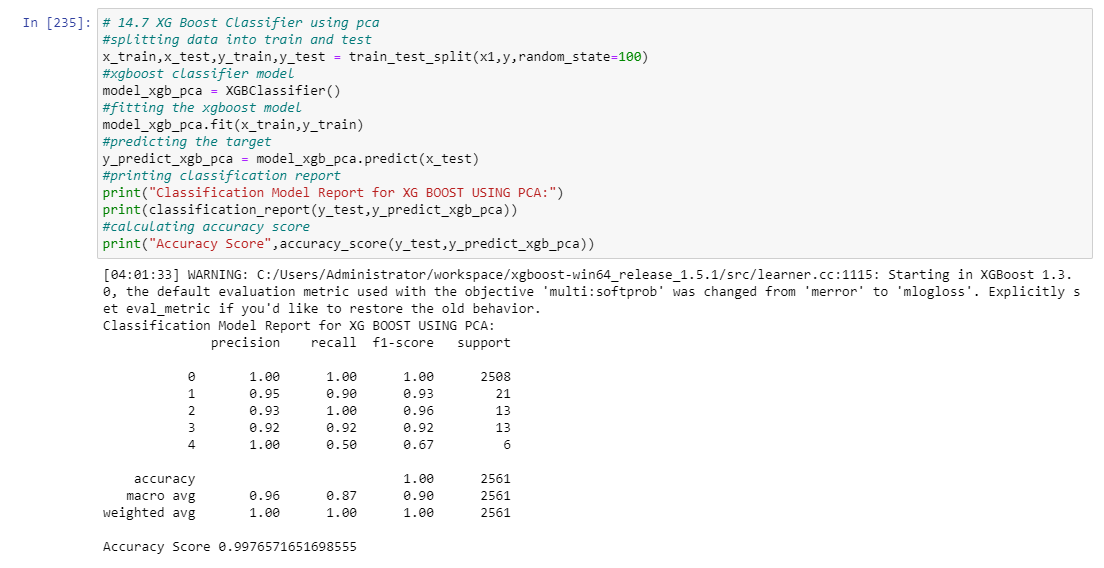
**#GRADIENT BOOST MODEL USING PCA:**



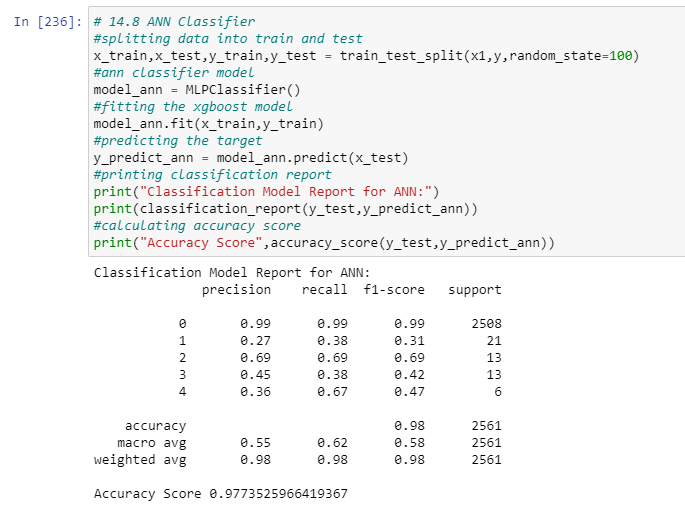
**#XG BOOST MODEL:**



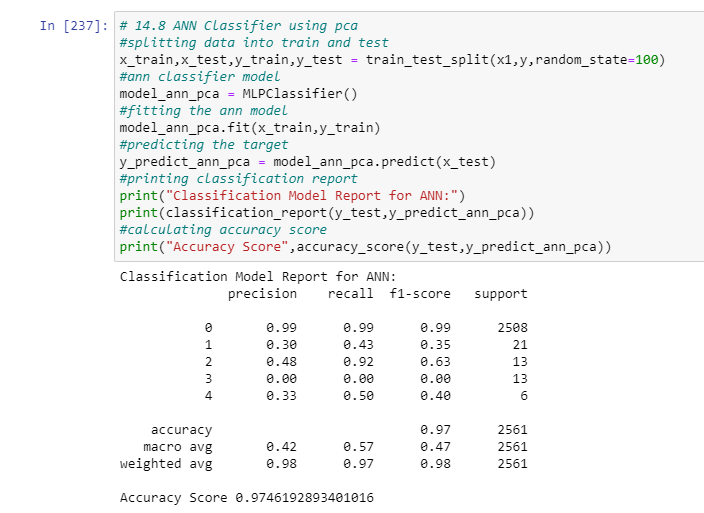
**#XG BOOST MODEL USING PCA :**



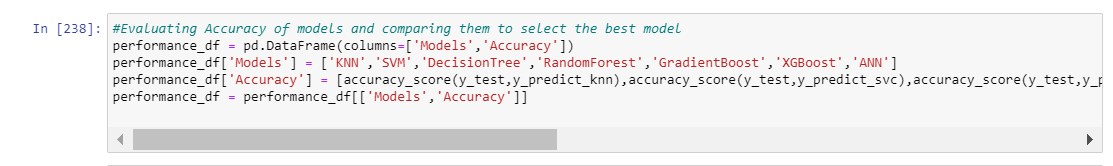
**#ANN MODEL:**

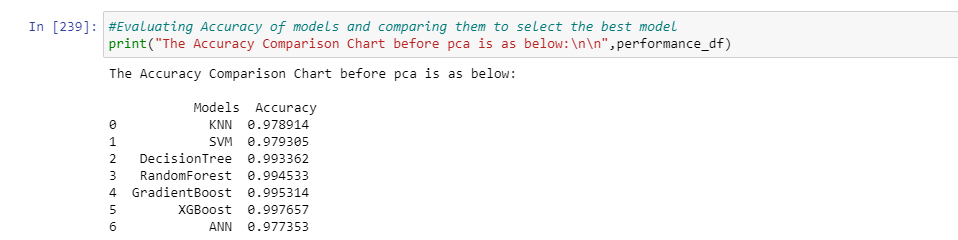


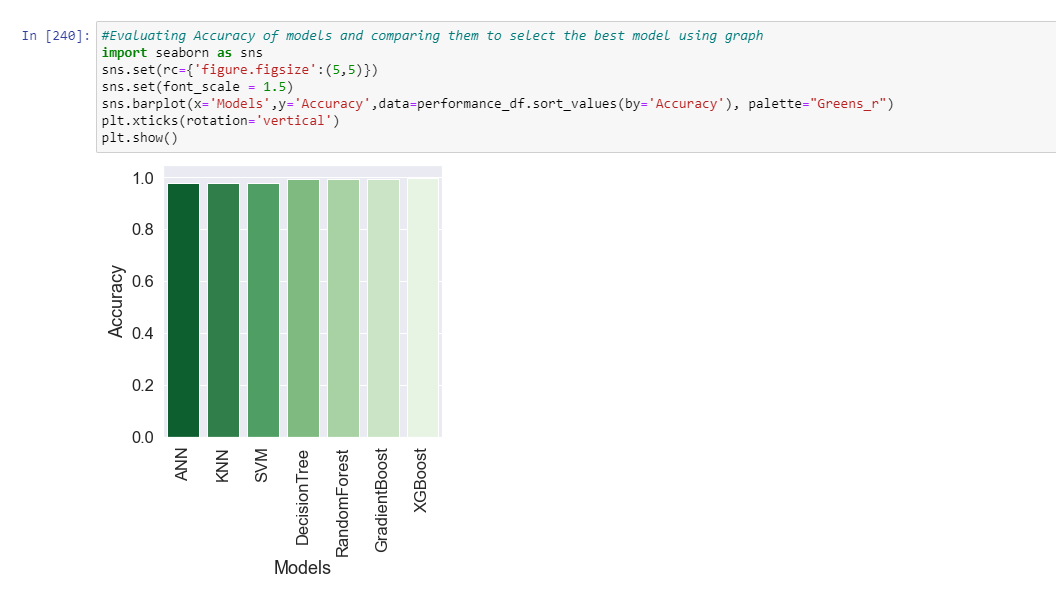
**#ANN MODEL USING PCA :**



* **EVALUATING COMPARING THE ACCURACY SCORES\PERFORMANCE OF MODELS :**
* **DIRECT DATA INPUT**





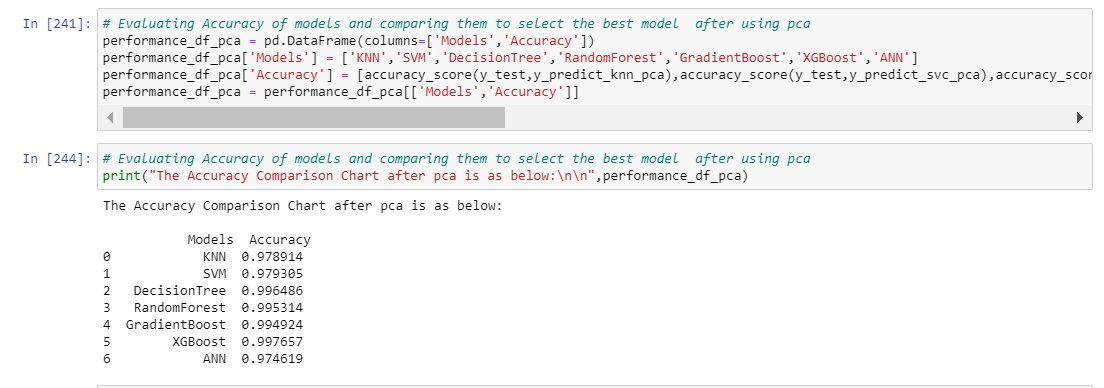


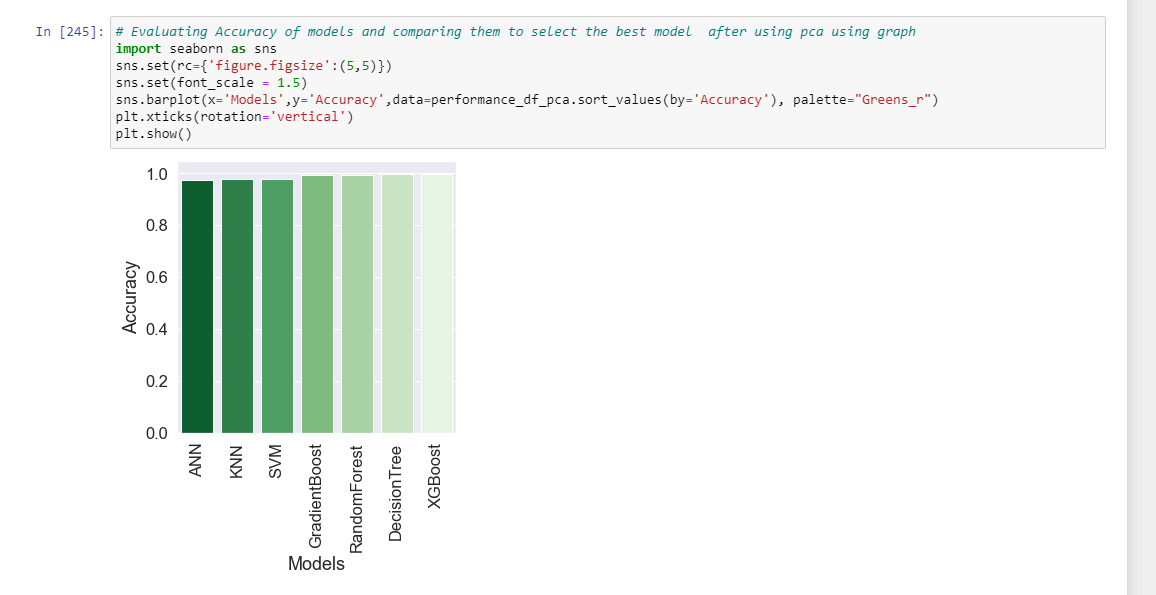
* **FINDINGS AND CONCLUSIONS:**

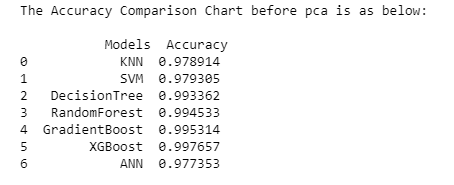
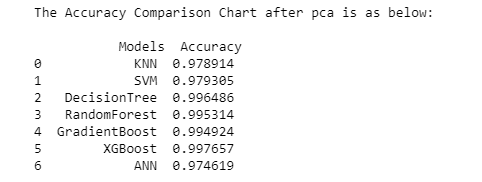
**1.XGBOOST IS THE BEST PERFORMING MODEL WITH AN ACCURACY SCORE OF 0.997 i.e. 99.7%**

**2.THE SCORES OBTAINED ARE ABOVE 96% FOR ALL MODELS. HIGHER THE ACCURACY BETTER IS THE MODEL PERFORMANCE IN GENERAL.**

* **COMPARING MODELS AFTER USING PCA DATA:**





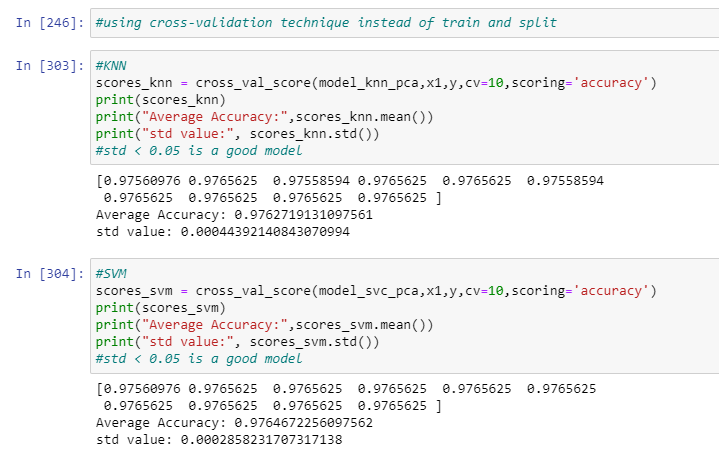
* **FINDINGS AND CONCLUSIONS:**

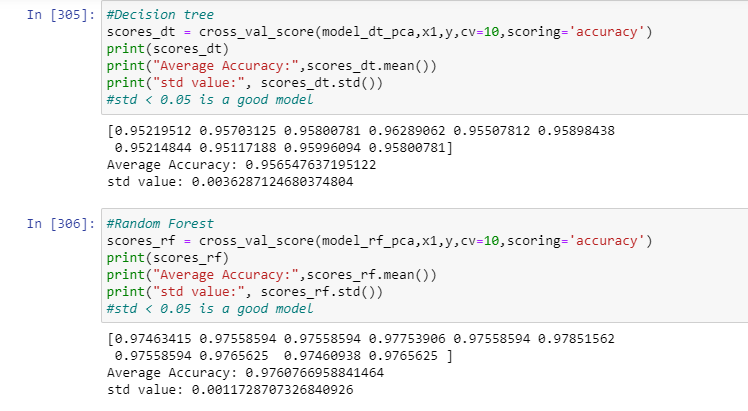
**1.XGBOOST IS THE BEST PERFORMING MODEL WITH AN ACCURACY SCORE OF 0.997 i.e. 99.7%**

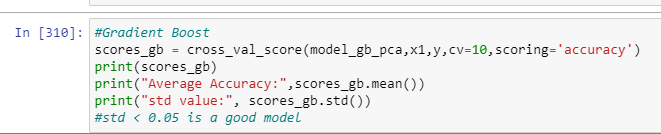
**2.THE SCORES OBTAINED ARE ABOVE 96% FOR ALL MODELS. HIGHER THE ACCURACY BETTER IS THE MODEL PERFORMANCE IN GENERAL.**

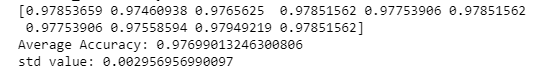
**3. WE SEE THAT THERE IS NO SIGNIFICANT DIFFERENCE IN ACCURACY AFTER PCA , FOR SOME OF THE MODELS THE ACCURACY HAS BECOME LESS BY A VERY SLIGHT MARGIN, WHICH CAN BE CONSIDERED ALRIGHT AS PCA INCREASES THE OVERALL MODEL PERFORMANCES AND MINIMISES THE DATA LOSS AND CHANCES OF OVERFITTING IS MINIMISED AT THE COST OF VERY SLIGHT REDUCTION IN ACCURACY IN CERTAIN CASES WHICH IS ACCEPTABLE.**

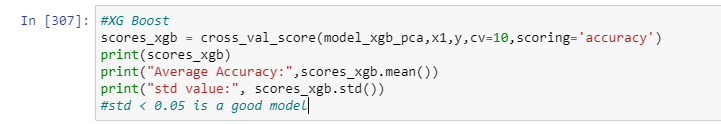
* **CROSS -VALIDATION TECHNIQUE**
* SO FAR, WE APPLIED TRAIN-TEST SPLIT VALIDATION METHODOLOGY , BUT THE TRAIN AND SPLIT METHODLOGY WE SPLIT THE DATA WHICH MAY CAUSE DATA LOSS, AND MAY POTENTIALLY LOWER MODEL ACCURACY
* SO WE GO FOR CROSS -VALIDATION TECHNIQUE :INSTEAD OF USING A SINGLE VALIDATION SET, WE CAN USE MANY VALIDATION SETS. WE CAN MAKE MANY TRAININGS : VALIDATION SPLITS AND CYCLE EACH PART OF THE DATA WE USE FOR VALIDATION EACH TIME SUCH THAT EVENTUALLY, OVER EVERY TRAINING: VALIDATION SPLIT COMBINED, ALL THE DATA HAS BEEN USED AT LEAST ONCE FOR VALIDATION, AND AT LEAST ONCE FOR TRAINING. WE GET TO USE 100% OF THE TRAINING + VALIDATION DATA FOR TRAINING, WHICH SMOOTHS OUT ISSUES WHERE THE INITIAL TRAINING SET WAS HIGHLY BIASED AND CONTAINED MANY EXAMPLES OF AN EXTREME OCCURRENCE, OR DID NOT CONTAIN ANY EXAMPLES OF AN IMPORTANT DATA TYPE/OCCURRENCE. WE GET TO VALIDATE OVER ALL OF THE DATA. WE GET TO AVERAGE OUR PERFORMANCE OVER ALL OF THE DATA, GIVING US FAR MORE CONFIDENCE IN OUR ESTIMATION OF THE MODEL’S SKILL, AS WELL AS AN ACTUAL PICTURE OF HOW VOLATILE THE MODEL IS TO PERTURBATIONS IN THE INPUT DATA.
* WE ARE AUTOMATICALLY FORCED TO BUILD A FAR LESS OVERFITTED (AND THUS MORE GENERALISABLE MODEL) BECAUSE WE ARE TRYING TO MAXIMISE THE AVERAGE PERFORMANCE OVER MANY VALIDATION SETS, NOT ONE SPECIFIC VALIDATION SET, SO WE CANNOT INADVERTENTLY TUNE TOWARDS HYPER-PARAMETER SETTINGS THAT ARE ONLY GOOD FOR A VERY SPECIFIC VALIDATION SET. AND HENCE MAKES OVERALL MODEL PERFORMANCE BETTER.
* **# K-FOLD CROSS VALIDATION OUTPUT:**

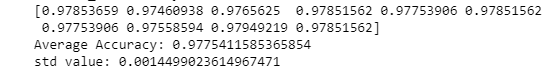


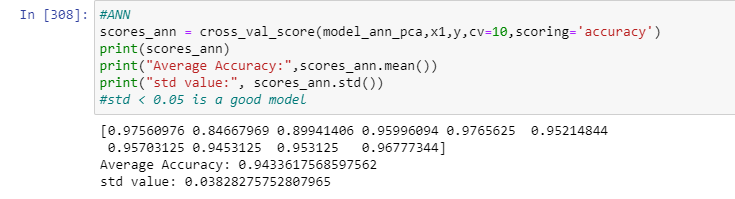




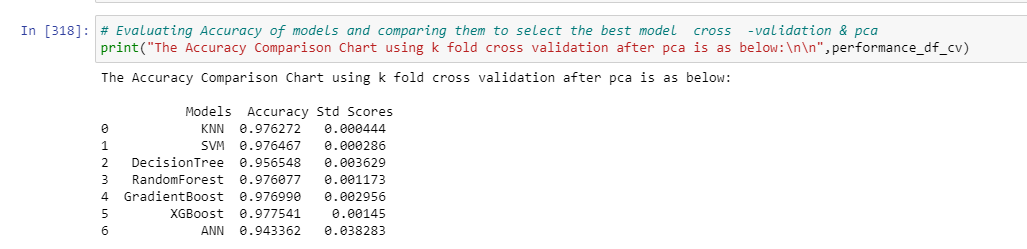


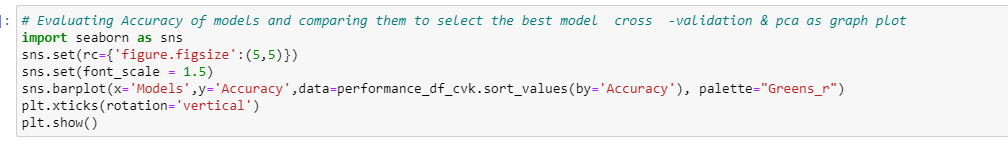


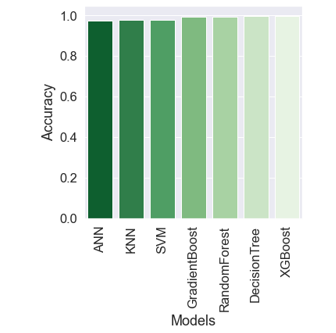




* **ACCURACY AND MODEL PERFORMANCE FOR K -FOLD CROSS VALIDATION MODELS WITH PCA INPUT:**







* **FINDINGS AND CONCLUSIONS:**

**1.XGBOOST IS THE BEST PERFORMING MODEL WITH AN ACCURACY SCORE OF 0.997 i.e. 97.7%**

**2.THE SCORES OBTAINED ARE ABOVE 96% FOR ALL MODELS. HIGHER THE ACCURACY BETTER IS THE MODEL PERFORMANCE IN GENERAL.**

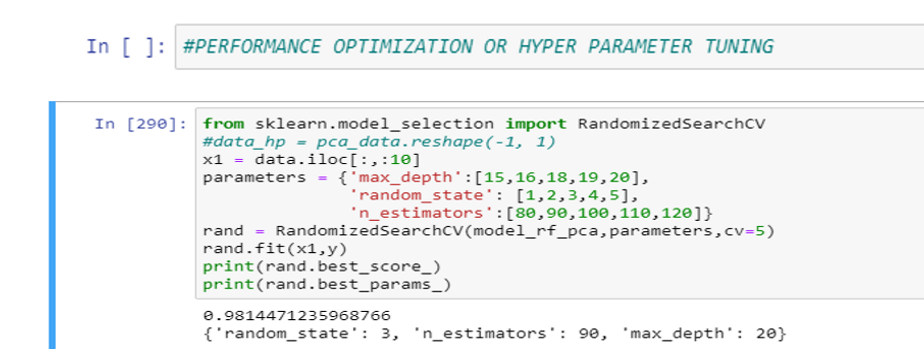
**3. WE SEE THAT THERE IS NO SIGNIFICANT DIFFERENCE IN ACCURACY AFTER K FOLD CROSS VALIDATION & PCA , FOR SOME OF THE MODELS THE ACCURACY HAS BECOME LESS BY A VERY SLIGHT MARGIN, WHICH CAN BE CONSIDERED ALRIGHT AS CROSS VALIDATION & PCA INCREASES THE OVERALL MODEL PERFORMANCES AND MINIMISES THE DATA LOSS AND CHANCES OF OVERFITTING IS MINIMISED AT THE COST OF VERY SLIGHT REDUCTION IN ACCURACY IN CERTAIN CASES WHICH IS ACCEPTABLE AND ALSO THE DATA LOSS DURING TRAINING AND TESTING IS ELIMINATED, ALONG WITH ELIMINATION OF OVER FITTING BY K-FOLD.**

* **PERFORMANCE OPTIMIZATION OR HYPER-PARAMETER TUNING:**

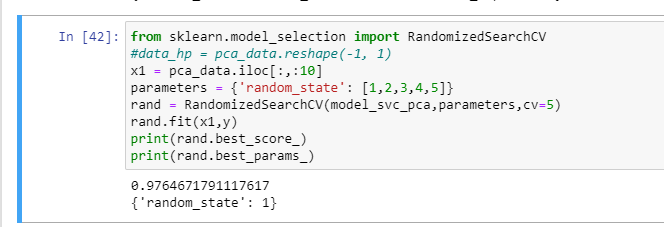
WE PERFORM HYPER – PARAMETER TUNING OR MODEL OPTIMISATION IN ORDER TO MAXIMIZE THE MODEL PERFORMANCE AND TO IMPROVE THE OVERALL ACCURACY OF THE MODEL ESPECIALLY IN THE CASES WHERE MODEL ACCURACY IS LESS . SINCE WE HAVE AN ACCURACY OF >97% FOR OUR MODELS , HYPER PARAMETER TUNING IS NOT ABSOLUTELY ESSENTIAL , BUT I WILL DEMONSTRATE HYPER-PARAMETER TUNING AND OPTIMISATION FOR MY MODELS TO UNDERSTAND HOW IT WILL ENHANCE THE MODEL FURTHER AS GIVEN BELOW . THERE ARE TWO POPULAR METHODS FOR CLASSIFIER MODELS: GRID SEARCH CV & RANDOMISED SEARCH CV.

* **I WILL BE USING RANDOMIZED SEARCH CV METHODOLOGY:**

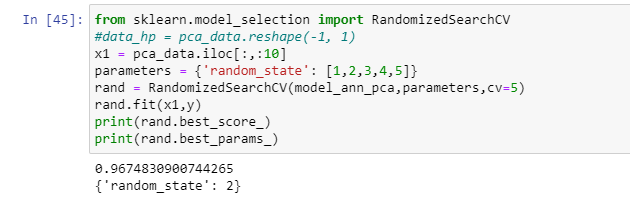
**#RANDOM FOREST TUNING:**



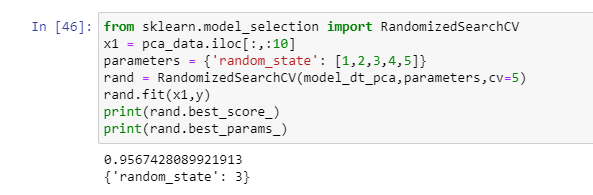
**#SVM TUNING:**



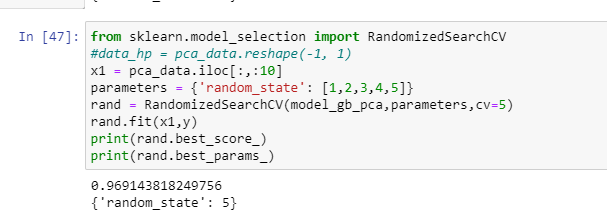
**# ANN TUNING:**



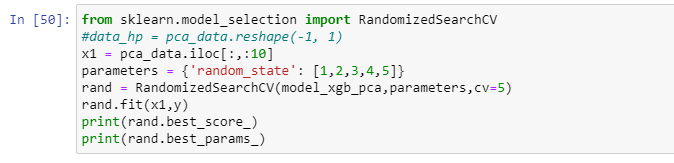
**# DECISION TREE TUNING:**



**#GRADIENT BOOST TUNING :**



**#XGBOOST TUNING:**





**FINDINGS AND CONCLUSION:**

WE PERFORMED RANDOMISED SEARCH HYPER PARAMETER TUNING OPTIMIZATION TECHNIQUE MODEL ACCURACY INCREASED SLIGHTLY FORFEW MODELS AND FOR THE REST OF THE MODELS THE CHANGE WAS NOT SIGNIFICANT.

**SUMMARY:**

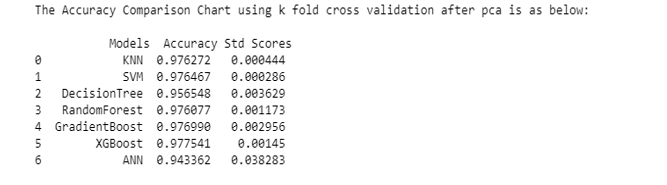
1.Any pre-analytics steps and your understanding of data before you perform the analytics

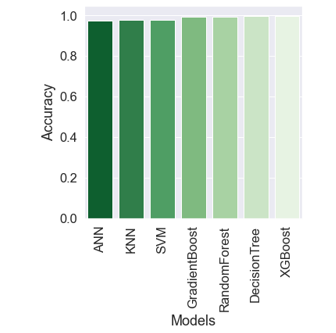
**I PERFORMED DATA UNDERSTANDING ,CLEANING & PROCESSING ALONG WITH SOME EXPLORATORY DATA ANALYSIS & STATISTICAL ANALYSIS TO OBTAIN INISGHTS ABOUT DATA**

2.Your choice of analytical algorithms, various steps taken during analytics, any comparisons between other algorithms and your understanding of how this model is performing

THE PROBLEM WAS A MULTI CLASS CLASSIFICATION PROBLEM.

I CHOSE 6 CLASSIFIER MODELS AND TRAINED & BUILT MODELS USING THEM, AND GOT THE ACCURACY SCORE AFTER PERFORMING PCA AND K CROSS FOLD VALIDATION ON DATA FOLLOWED BY HYPER PARAMETER TUNING





**FROM MY ANALYSIS XGBOOST CLASSIFIER HAD THE HIGHEST ACCURACY SO I CHOSE TO GO WITH IT.**

3.Your final choice of model and summary:

MY FINAL CHOICE OF MODEL IS XGBOOST

I FOLLOWED THE BELOW STEPS (SUMMARY OF MY WORK):

**# STEPS FOLLOWED:**

**# 1. DATA CLEANING**

**# 2. DATA NULL HANDLING, DE-DUPLICATION HANDLING DUPLICATE VALUES etc**

**# 3. EDA & STATISTICA ANALYSIS**

**#4 CORRELATION MATRIX SELECT 20 MOST CORRELATED COLUMNS DATA, INFERENTIAL ANALYSIS**

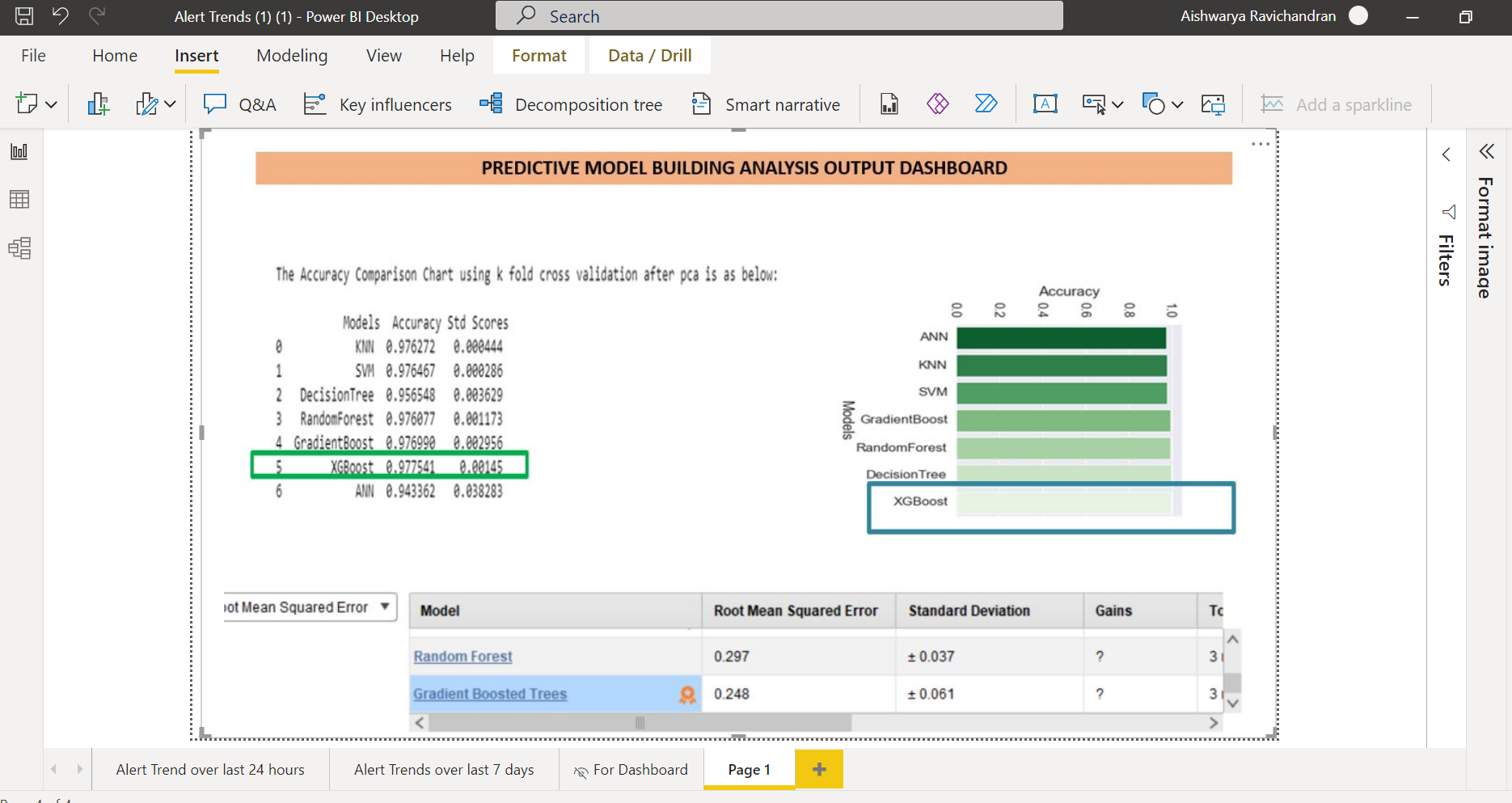
**# 4. FEATURE SELECTION & PERFORM DIMENSION REDUCTION USING PCA**

**# 5. USE THE PCA OUTPUT FOR MODEL BUILDING ( 6 CLASSIFIER MODELS)**

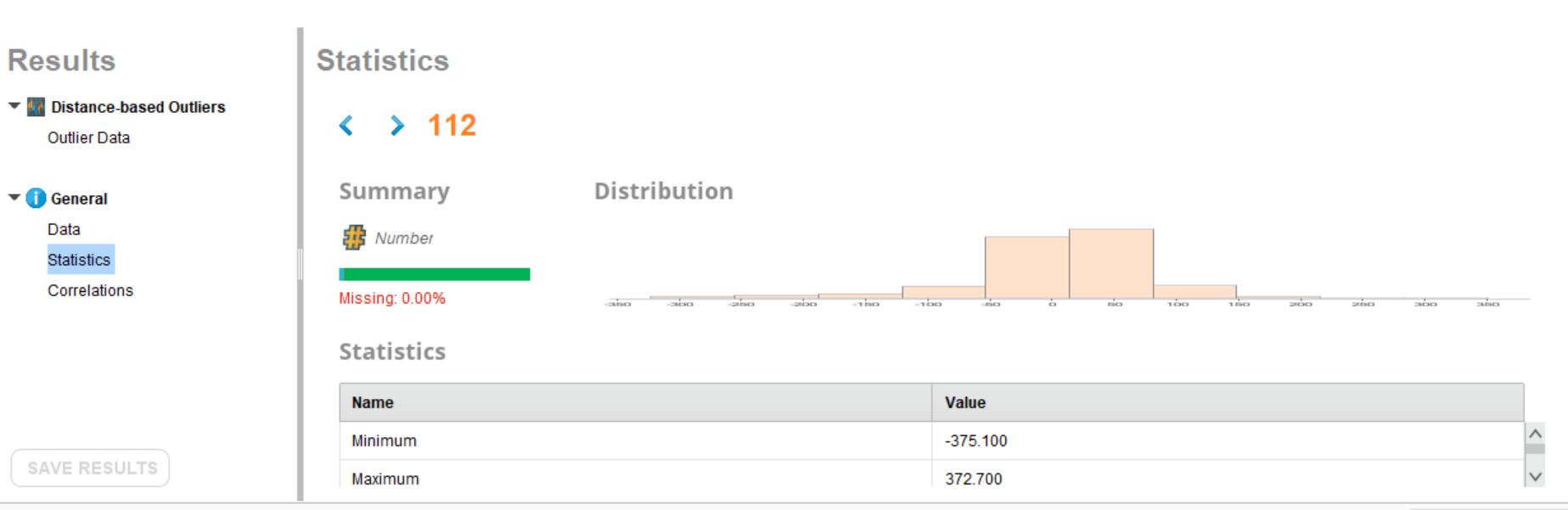
**# 6. MODEL SELECTION, MODEL BUILDING,VALIDATION & FINDING ACCURACY SCORES**

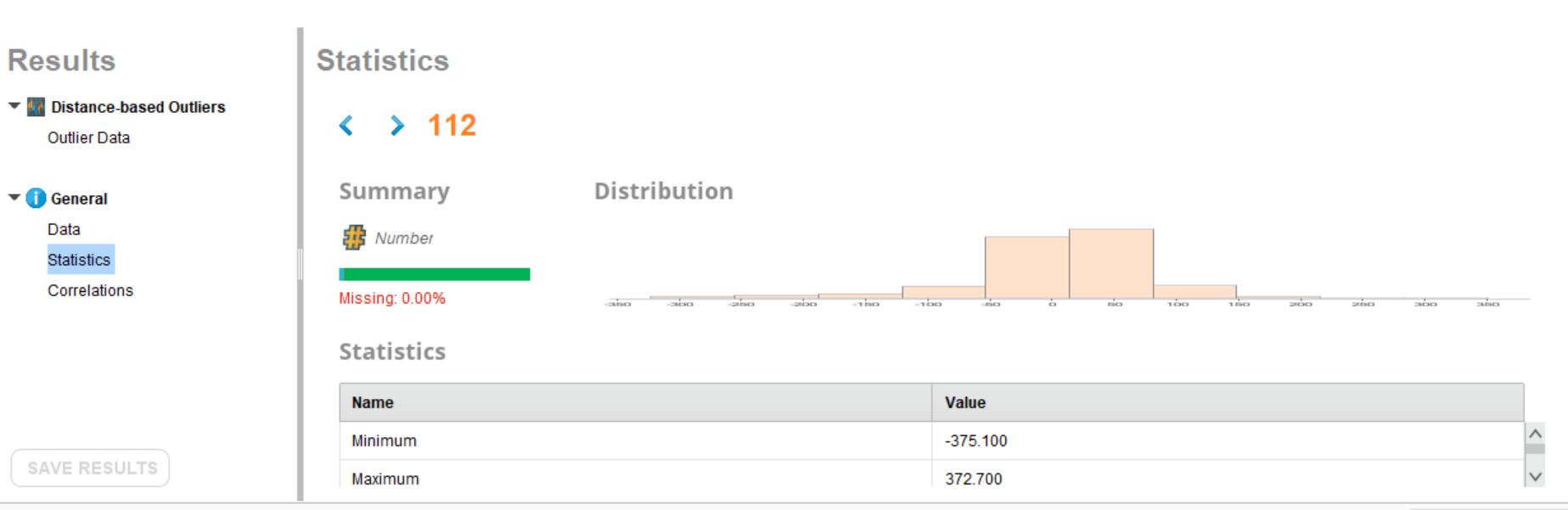
**# 7.HYPER PARAMETER TUNING & MODEL OPTIMIZATION**

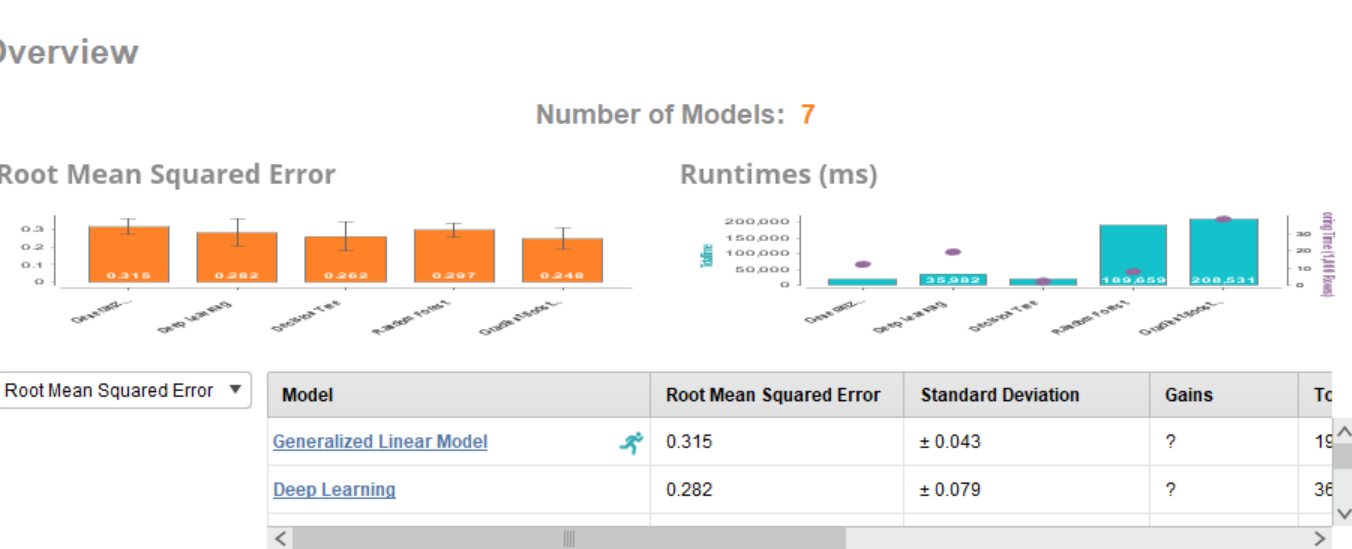
**THE FINAL MODEL WILL BE DEPLOYED AND MONITORED THE MONITORING OUTPUT RESULTS ETC WILL BE SHOWN IN REPORT(HERE POWER BI) ALONG WITH DAT AGOVERNANCE & PROTECTION APPLIED TO DATA AND INSIGHTS.**

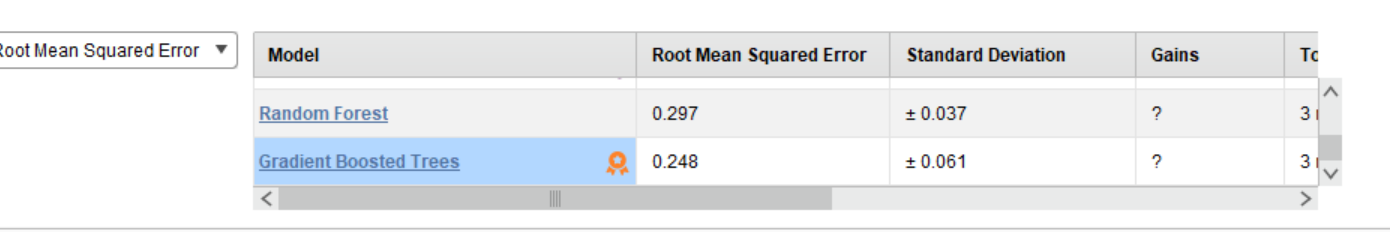


**RAPIDMINER OUTPUTS:**









**CONCLUSION: BUILT MODEL USING RAPIDMINER AND GRADIENT TRESS ,XGBOOST WAS THE BEST MODEL.**

**APPENDIX**

#Importing Libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sb

import scipy as sp

import sklearn

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

from collection import Counter

from sklearn.decomposition import pca

from sklearn.ensemble import ExtraTreesClassifier

from sklearn.decomposition import PCA

from sklearn import KNeighborsClassifier

#Importing Warnings

warnings.filterwarnings('ignore')

#Setting the display properties

pd.set\_option('display.max\_rows',10500,'display.max\_columns',500)

#Load Data

data = pd.csv('Q1\_data.csv')

data.head()

#Shape of data

data.shape

#Displaying entire data

data

#Information about the data with (datatypes)

data.info()

#Handling NULL & Missing Values

data.isnull().sum()

data.isnull().sum().sum()

#Replacing NULL with Average Value

data.fillna(data.mean(),inplace=True)

#Data De-Duplication or Handling Duplicates

x=data.duplicated()

data[x]

data[x].shape

#Data Correlation - Identifying Correlation between Attributes

data.corr()

#Correlation Matrix Plot

plt.figure(figsize=(18,8))

sb.set(font\_scale=0.5)

cmap = sb.light\_palette("#800080",as\_cmap=True)

sb.heatmap(data.corr(),cmap=cmap,annot=True)

plt.title("Correlation Matrix",fontsize=19)

plt.savefig('plot6.png', dpi=300, bbox\_inches='tight')

#Feature Selection Using Extra Trees

plt.figure(figsize=(20,8))

sb.set(font\_scale=0.5)

x = data.iloc[:,:-1]

y = data.iloc[:,151:152]

model = ExtraTreesClassifer()

model.fit(x,y)

print(model.feature\_importances\_)

feat\_importances = pd.Series(model.feature\_importances\_, index=x.columns)

feat\_importances.nlargest(20).plot(kind='barh')

plt.show()

print(feat\_importances.nlargest(20))

y1 = feat\_importances.nlargest(20)

y1.to\_csv("Features Selected.csv")

Xfs = data.loc[:,[25,56,9,64,73]]

Xfs

#Identifying Correlation of each Feature

x = data.iloc[:,:-1]

y = data.iloc[:,151:152]

corr\_mat = data.corr()

top\_corr\_features = corr\_mat.index

plt.figure(figsize=(20,20))

g=sd.heatmap(data[top\_corr\_features].corr(),annot=True,cmap="RdlGn")

#Perform Dimension Reduction Using PCA

x = data.iloc[:,:-1]

y = data.target

pca = PCA()

x1 = pd.DataFrame(pca.fit\_transform(data))

x1.head()

pca\_output = x1.iloc[:,:10]

pca\_output.head()

pca.components\_

pca.n\_components\_

plt.plot(range(0,152),pca.explained\_variance\_ratio\_)

#Model Selection & Model Building

# 1. KNN

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,random\_state=100)

model\_knn = kNeighborsClassifier()

model\_knn.fit(x\_train,y\_train)

y\_predict\_knn = model\_knn.predict(x\_test)

print("Classification Model Report for KNN:")

print(classification\_report(y\_test,y\_predict\_knn))

print("Accuracy Score",accuracy\_score(y\_test,y\_predict\_knn))

from sklearn.model\_selection import GridSearchCV

Import GridsearchCV from Scikit Learn

param\_grid = {'C': [0.1,1, 10, 100],

'gamma': [1,0.1,0.01,0.001],'kernel': ['rbf', 'poly', 'sigmoid']}

grid = GridSearchCV(SVC(),param\_grid,refit=True,verbose=2)

grid.fit(X\_train,y\_train)

print(grid.best\_estimator\_)