Capstone Project

Mobile Price Range Prediction

By

Shivani Thakur

CONTENT

- Introduction
- Data description and summary
- Problem Statement
- Knowing the data
- Data pre-processing
- Outliers Detection
- Exploratory data analysis

- Machine learning algorithms

1. Naïve Bayes
2. Decision tree
3. Random forest classifier
4. Logistic Regression
5. KNN
6. XG Boost

- Model Performance
- Challenges
- Conclusion

INTRODUCTION

Mobile phones come in all sorts of prices, features, specifications and all. Price estimation and prediction is an essential part of consumer strategy. Deciding on the correct price of a product is very important for the market success of a product. A new product that has to be launched, must have the correct price so that consumers find it appropriate to buy the product.

The objective is to find out some relation between features of a mobile phone (e.g:-RAM, Internal Memory, etc) and its selling price. In this problem, we do not have to predict the actual price but a price range indicating how high the price is.

DATA DESCRIPTION

The data contains information regarding mobile phone features, specifications etc and their price range. The various features and information can be used to predict the price range of a mobile phone.

The data features are as follows:

- ❑ Battery_power - Total energy a battery can store in one time measured in mAh
- ❑ Blue - Has bluetooth or not
- ❑ Clock_speed - speed at which microprocessor executes instructions
- ❑ Dual_sim - Has dual sim support or not
- ❑ Fc - Front Camera mega pixels
- ❑ Four_g - Has 4G or not
- ❑ Int_memory - Internal Memory in Gigabytes
- ❑ M_dep - Mobile Depth in cm
- ❑ Mobile_wt - Weight of mobile phone
- ❑ N_cores - Number of cores of processor

DATA DESCRIPTION (Cont..)

- ❑ Pc - Primary Camera mega pixels
- ❑ Px_height - Pixel Resolution Height
- ❑ Px_width - Pixel Resolution Width
- ❑ Ram - Random Access Memory in Mega Bytes
- ❑ Sc_h - Screen Height of mobile in cm
- ❑ Sc_w - Screen Width of mobile in cm
- ❑ Talk_time - longest time that a single battery charge will last
- ❑ Three_g - Has 3G or not
- ❑ Touch_screen - Has touch screen or not
- ❑ Wifi - Has wifi or not
- ❑ Price_range - This is the target variable with value of 0(low cost), 1(medium cost), 2(high cost) and 3(very high cost).

PROBLEM STATEMENT

- The problem statement is to predict the price range of mobile phones based on the features available (price range indicating how high the price is).
- Here is the description of target classes:

  0 - Low cost phones

  1 - Medium cost phones

  2 - High cost phones

  3 - Very high cost phones
- This will basically help companies to estimate price of mobiles to give tough competition to other mobile manufacturer. Also, it will be useful for consumers to verify that they are paying best price for a mobile.

KNOWING THE DATA

- ❑ We have a record of 2000 mobile phones with 20 features.
- ❑ We have perfectly balanced dataset with 500 observations for each class.
- ❑ Each column represents the feature of the mobile.
- ❑ There are two types of data

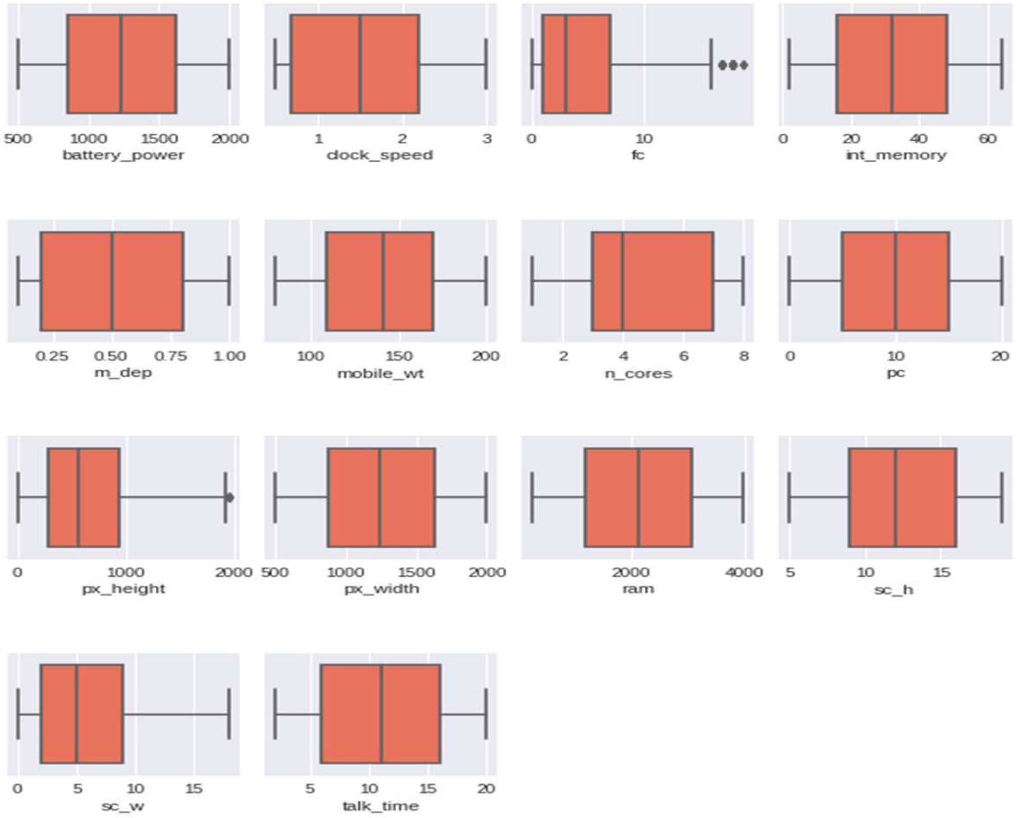
  1. Numerical Data
  2. Categorical Data.

- ❑ There are No Null Values present
- ❑ There are No Duplicate values present.
- ❑ There are no outliers
- ❑ We implemented different model to find out best model to predict the mobile price range with respect to the mobile features.
- ❑ We have applied – Decision Tree, Random Forest, Naïve Bayes, KNN, Logistic Regression and XG Boost.
- ❑ At last we conclude that Logistic Regression is performing better than any other model

DATA PREPROCESSING

- Step 1: Problem Description
- Step 2: Understanding & Pre-processing of data:
  Null values , Duplicate values and Outlier Detection .
- Step 3: Creating numerical and categorical Columns .
- Step 4: Univariate analysis and check multicollinearity .
- Step 5: Splitting the dataset into the training and test sets.
- Step 6: Model Implementation .
- Step 7: Model performance .
- Step 8: Challenges .
- Step 9: Conclusion

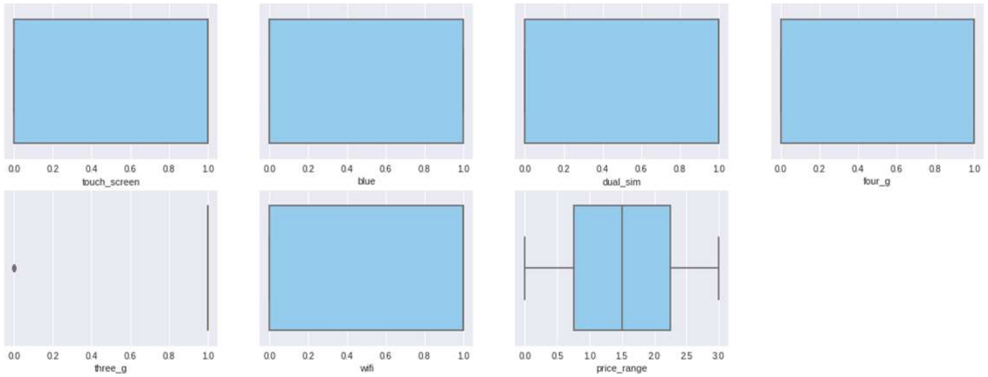
OUTLIERS DETECTION



For Numerical Columns

- Outliers can cause serious issues in statistical analysis.
- Hence we have checked this before starting analysing our data. So that we can visualise.
- In our data, there seems to be no outlier.

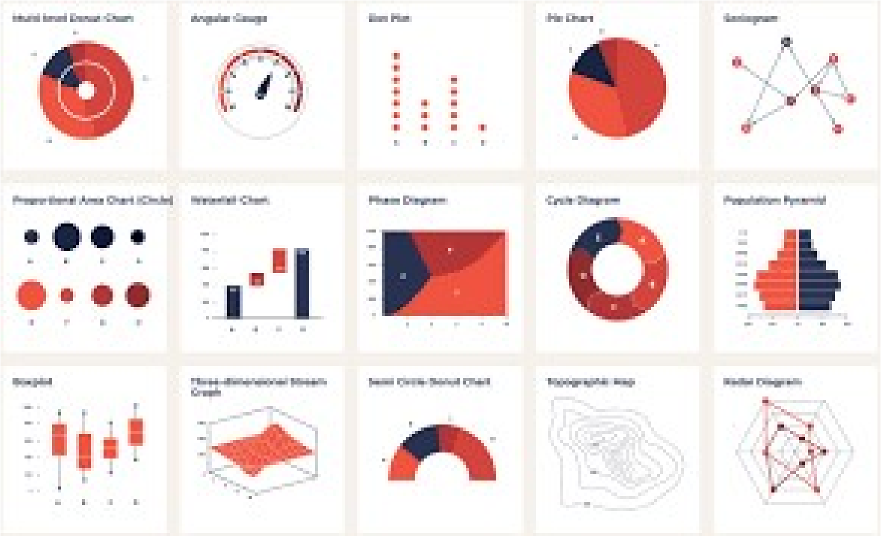
For Categorical Columns



Exploratory Data Analysis

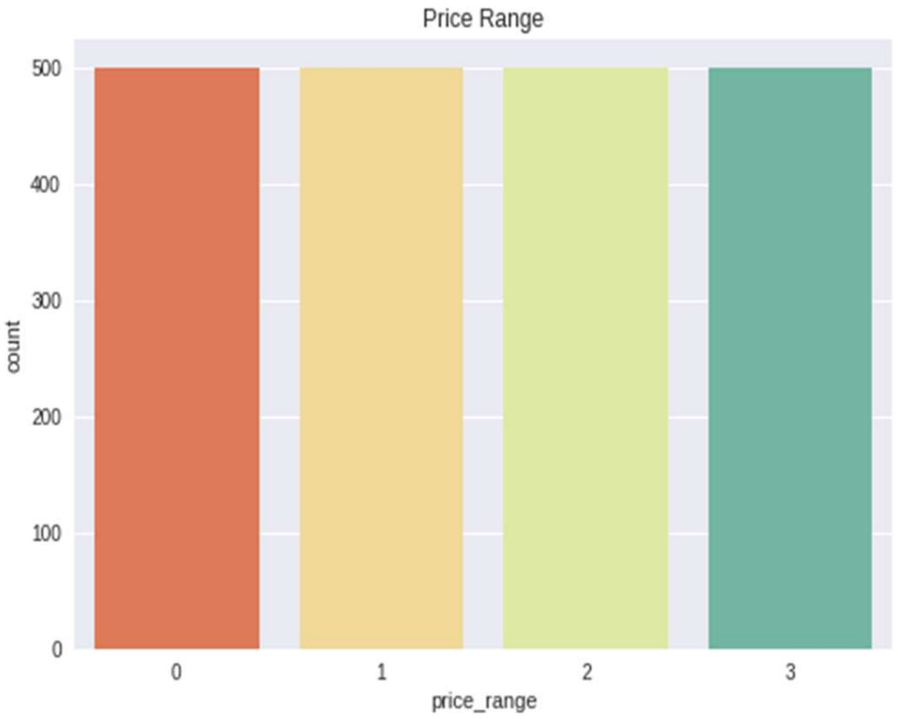
We are done with pre-processing our data.

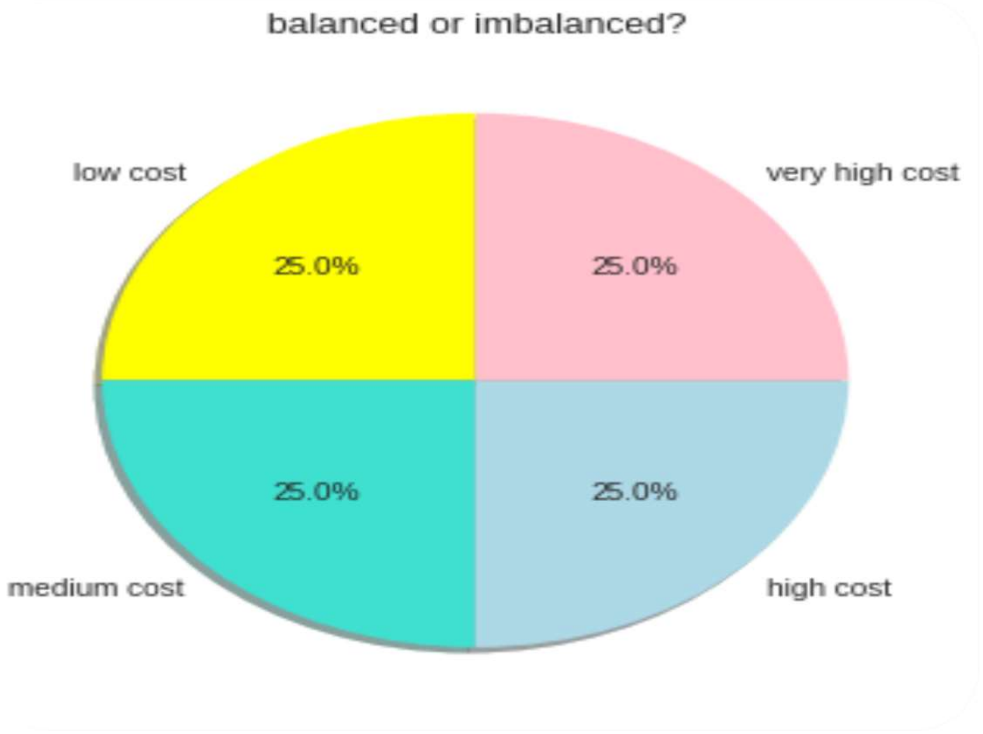
Our next step is to perform EDA.



VISUALIZING PRICE RANGE

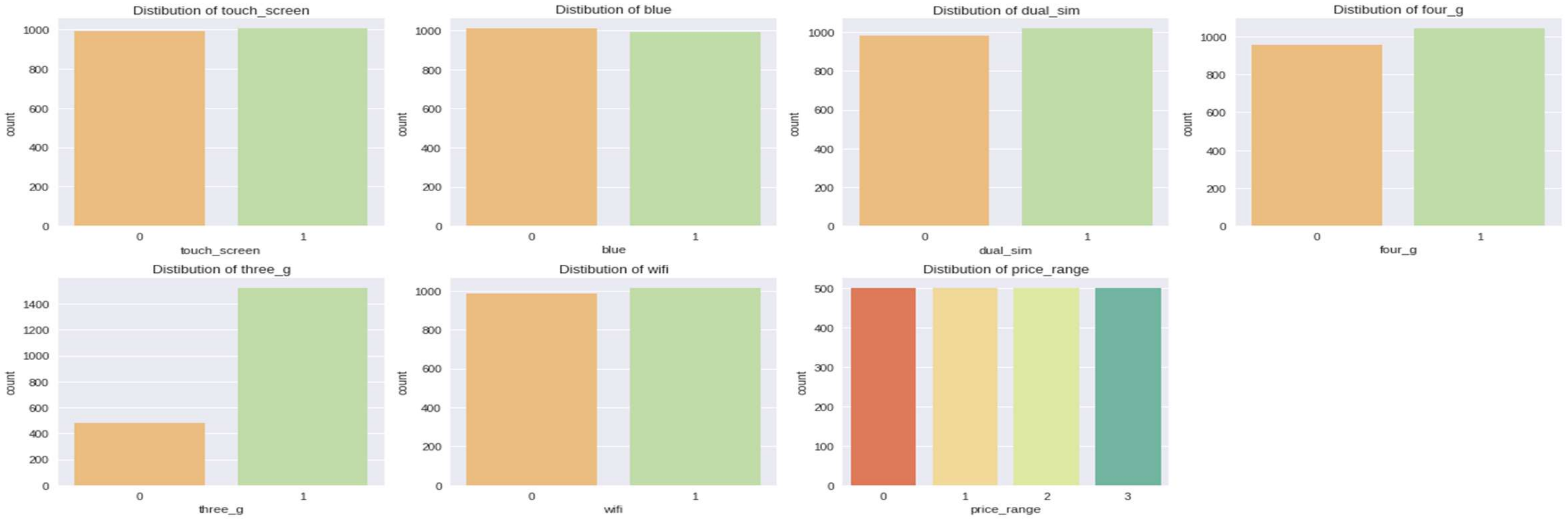
- Both Count Plot and Bar plot of our Price Range columns label(dependent variable is very balanced for all type of mobile price range categories.





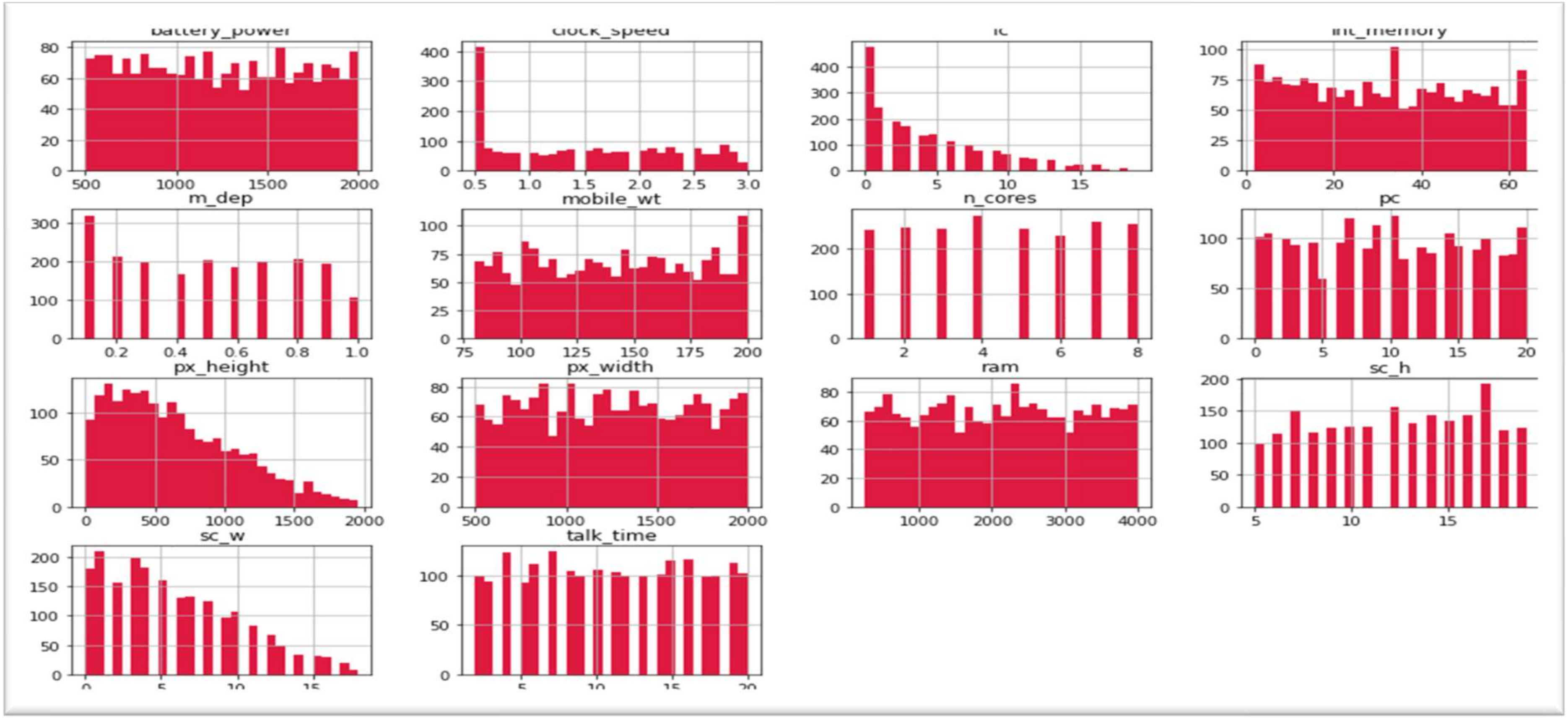
Our data is equally distributed

VISUALIZING CATEGORICAL COLUMNS



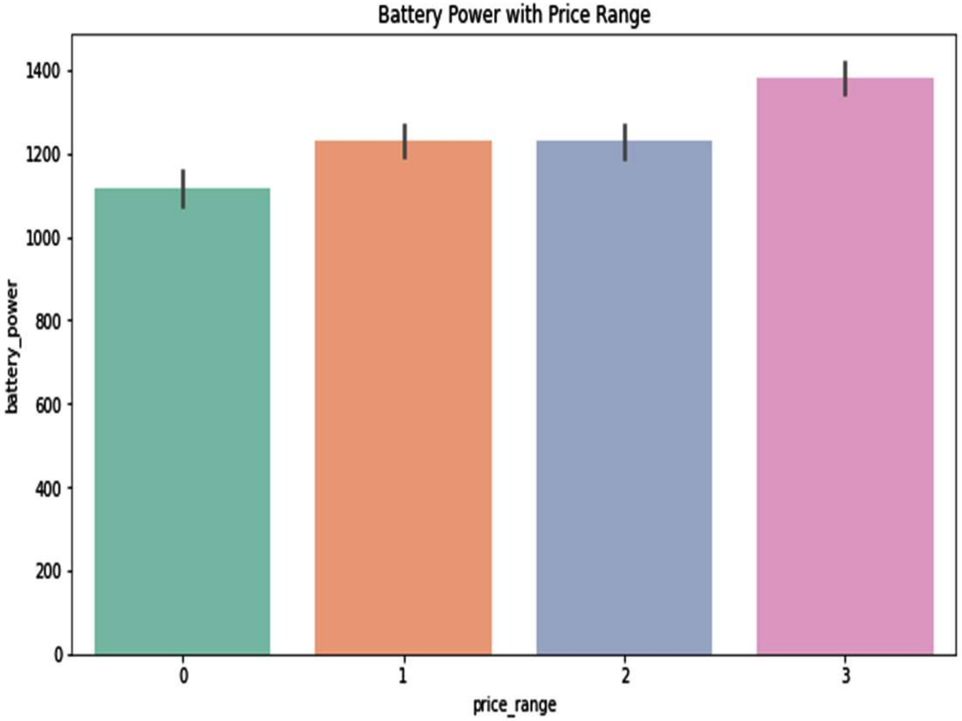
- There are almost every phone has some common features like touch screen ,Bluetooth, dual sim and wifi.
- There are mobile phones in 4 different price ranges and they are equally distributed.
- But the count of 3-G phones are least in low cost phones . However, count of 4-G phones is higher .
- In mid cost both 3-G and 4-G phones are available at high count.

VISUALIZING NUMERICAL COLUMNS

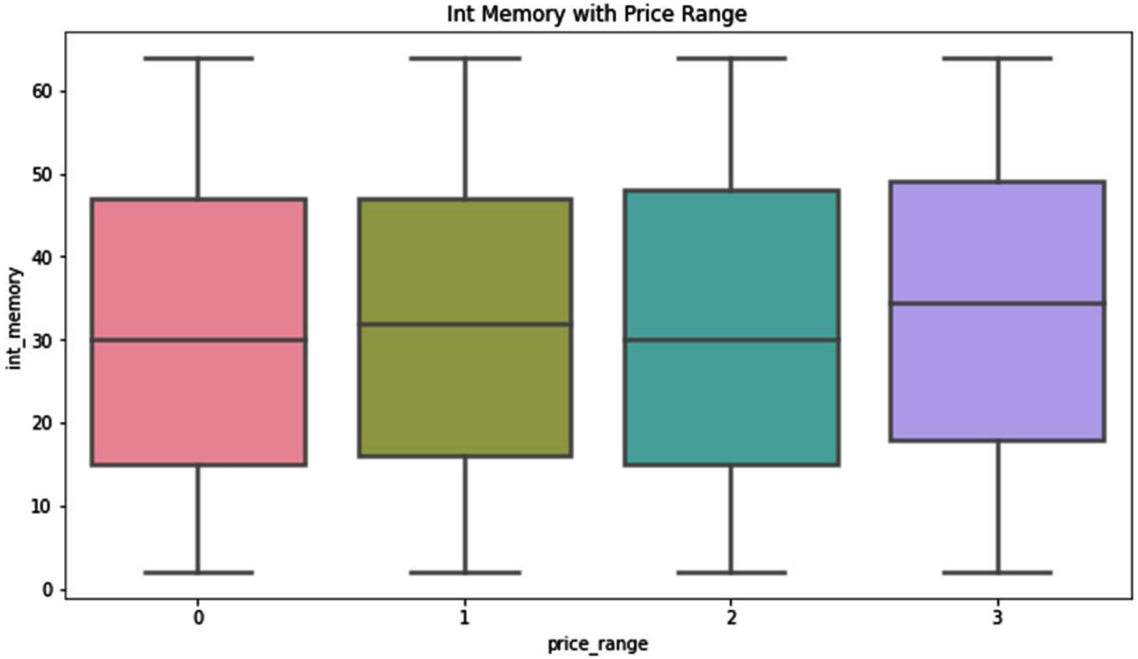


- Let’s discuss in brief each columns

RELATION PRICE WITH BATTERY POWER & INT MEMORY

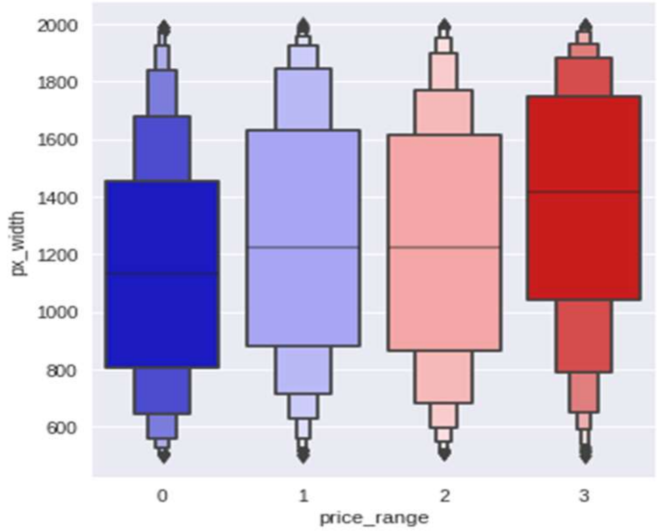


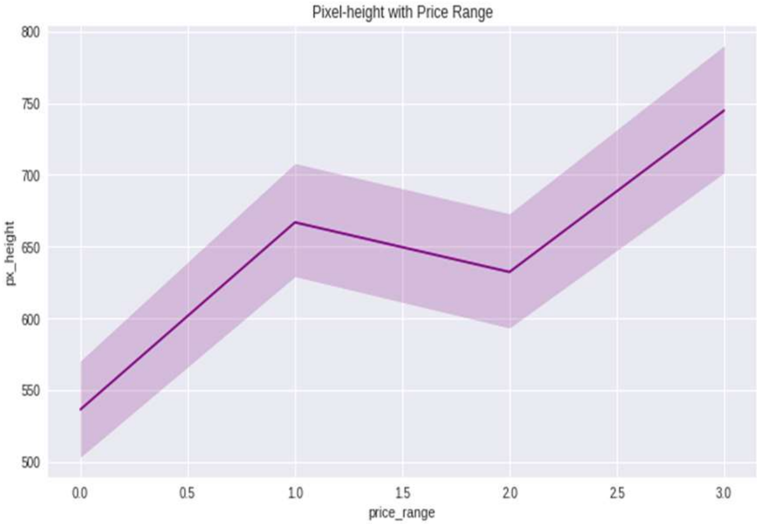
we can see from barplot of battery against price range that more expensive phone has higher battery power .



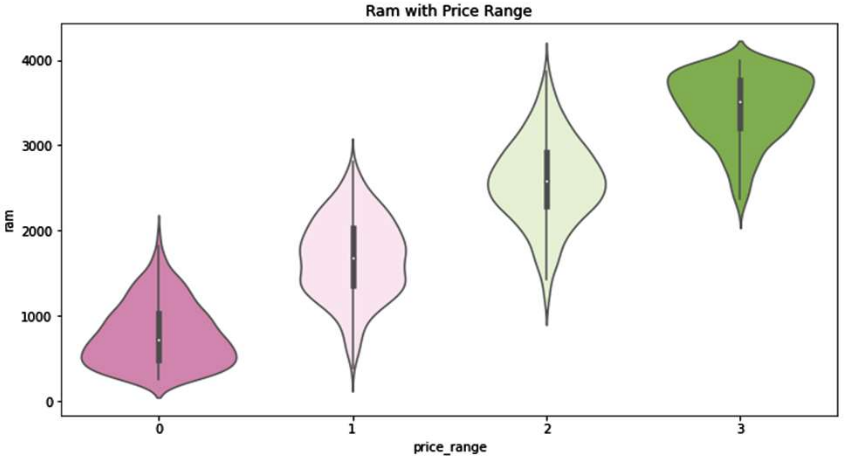
Internal memory is same for every range of phones.

PRICE RANGE WITH PIXEL HEIGHT-WIDTH & RAM





Price is increasing with Ram.



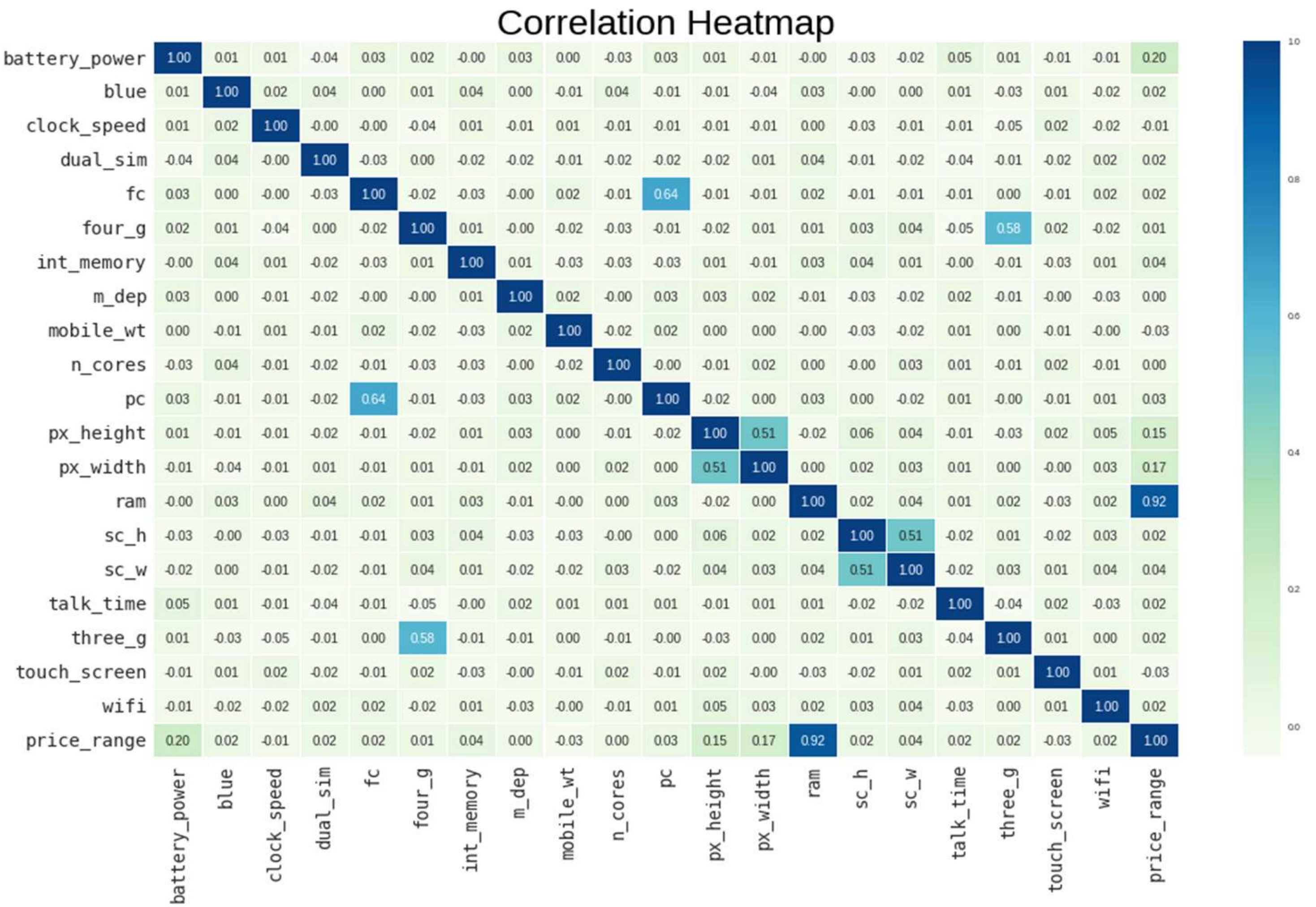
Price of mobile phones are directly related with median value of pixel height i.e both are increasing or vice versa.

Expensive mobile phone's pixel width is more.

CORRELATION

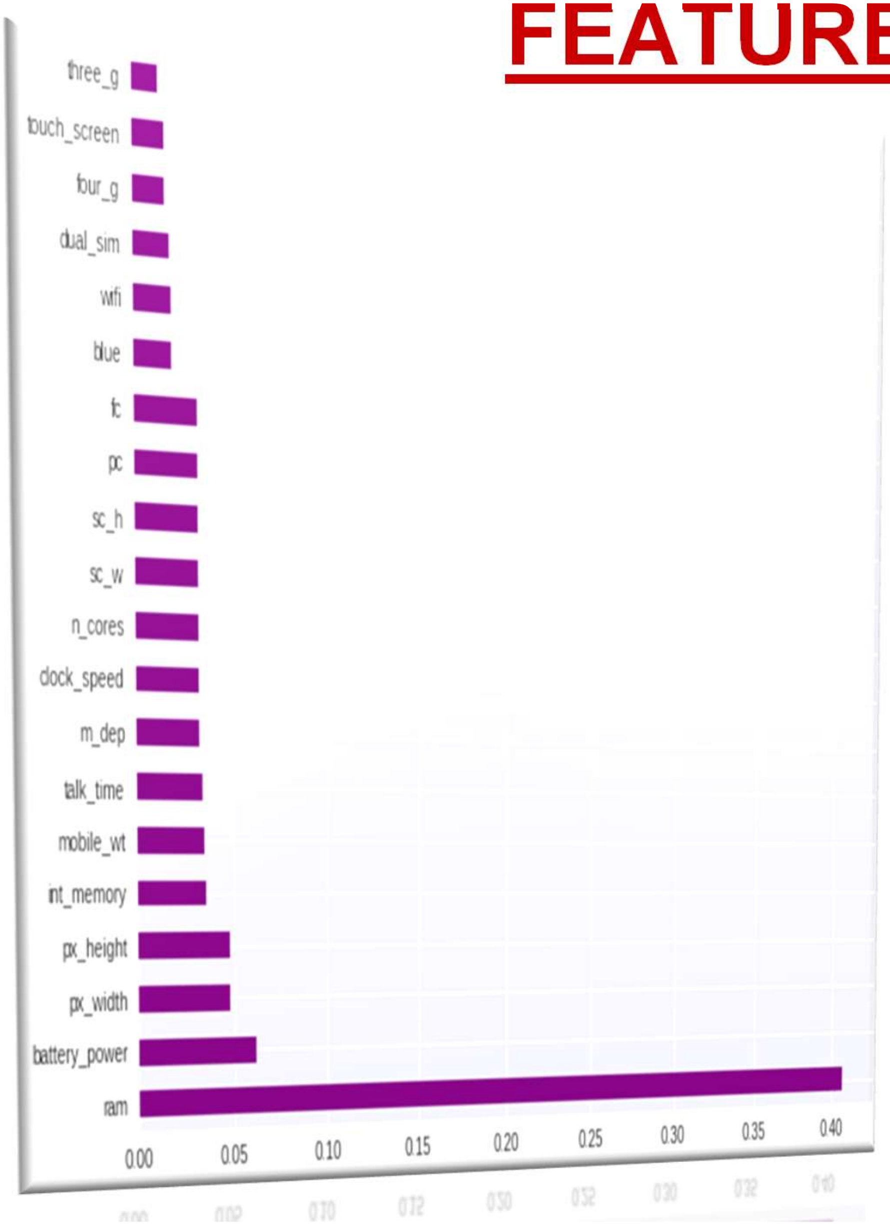
We see from the heatmap:

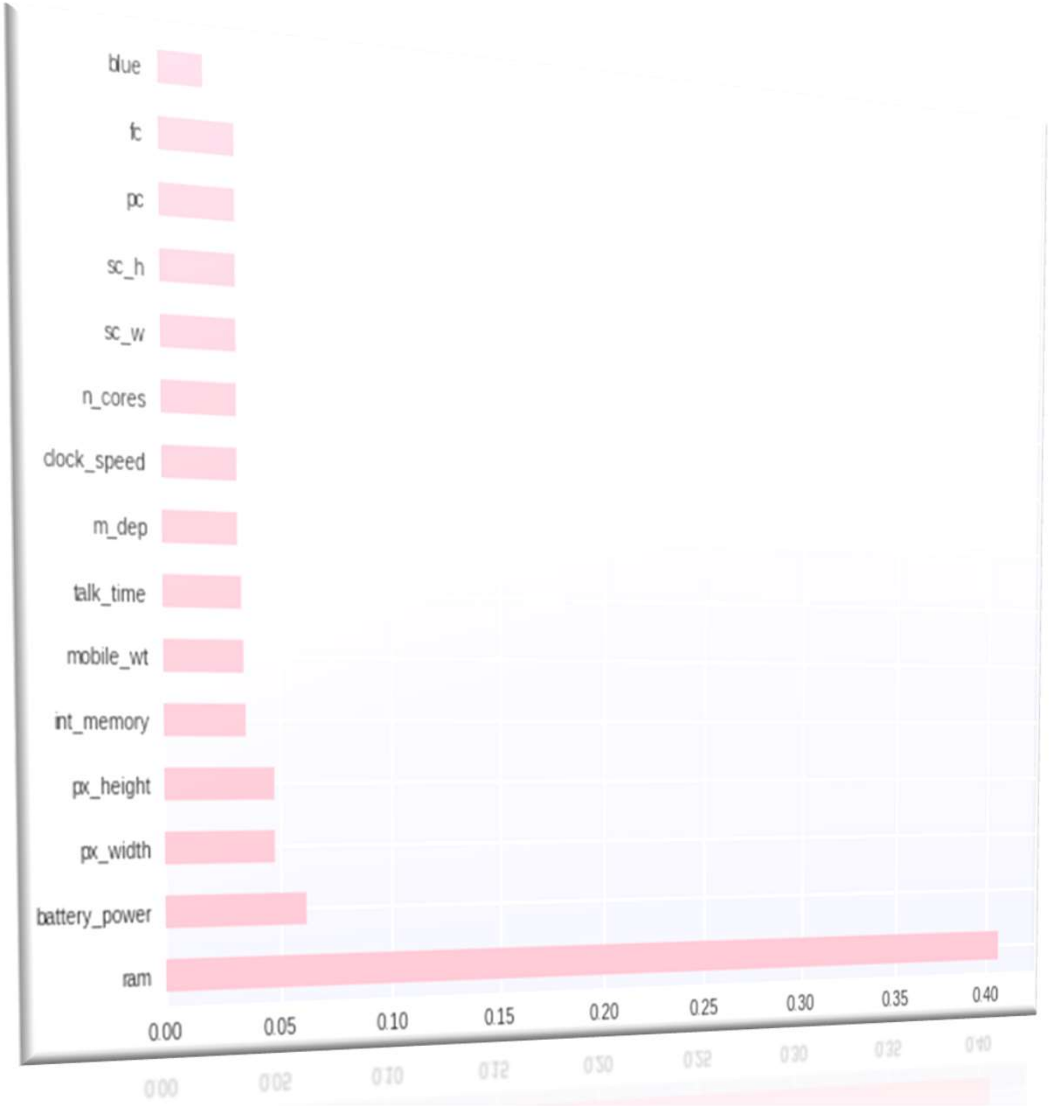
- The most influential variable is ram.
- Most of the variables have very little correlation to price range
- Primary camera mega pixels and front Camera mega pixels have correlation (it make sense because both of them reflect technology level of resolution of the related phone model) but they do not effect price range.
- Having 3G and 4G is somewhat correlated
- There is no highly correlated inputs in our dataset, so there is no multicollinearity problem.



	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	pc	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
battery_power	1.00	0.01	0.01	-0.04	0.03	0.02	-0.00	0.03	0.00	-0.03	0.03	0.01	-0.01	-0.00	-0.03	-0.02	0.05	0.01	-0.01	-0.01	0.20
blue	0.01	1.00	0.02	0.04	0.00	0.01	0.04	0.00	-0.01	0.04	-0.01	-0.01	-0.04	0.03	-0.00	0.00	0.01	-0.03	0.01	-0.02	0.02
clock_speed	0.01	0.02	1.00	-0.00	-0.00	-0.04	0.01	-0.01	0.01	-0.01	-0.01	-0.01	-0.01	0.00	-0.03	-0.01	-0.01	-0.05	0.02	-0.02	-0.01
dual_sim	-0.04	0.04	-0.00	1.00	-0.03	0.00	-0.02	-0.02	-0.01	-0.02	-0.02	-0.02	0.01	0.04	-0.01	-0.02	-0.04	-0.01	-0.02	0.02	0.02
fc	0.03	0.00	-0.00	-0.03	1.00	-0.02	-0.03	-0.00	0.02	-0.01	0.64	-0.01	-0.01	0.02	-0.01	-0.01	-0.01	0.00	-0.01	0.02	0.02
four_g	0.02	0.01	-0.04	0.00	-0.02	1.00	0.01	-0.00	-0.02	-0.03	-0.01	-0.02	0.01	0.01	0.03	0.04	-0.05	0.58	0.02	-0.02	0.01
int_memory	-0.00	0.04	0.01	-0.02	-0.03	0.01	1.00	0.01	-0.03	-0.03	-0.03	0.01	-0.01	0.03	0.04	0.01	-0.00	-0.01	-0.03	0.01	0.04
m_dep	0.03	0.00	-0.01	-0.02	-0.00	-0.00	0.01	1.00	0.02	-0.00	0.03	0.03	0.02	-0.01	-0.03	-0.02	0.02	-0.01	-0.00	-0.03	0.00
mobile_wt	0.00	-0.01	0.01	-0.01	0.02	-0.02	-0.03	0.02	1.00	-0.02	0.02	0.00	0.00	-0.00	-0.03	-0.02	0.01	0.00	-0.01	-0.00	-0.03
n_cores	-0.03	0.04	-0.01	-0.02	-0.01	-0.03	-0.03	-0.00	-0.02	1.00	-0.00	-0.01	0.02	0.00	-0.00	0.03	0.01	-0.01	0.02	-0.01	0.00
pc	0.03	-0.01	-0.01	-0.02	0.64	-0.01	-0.03	0.03	0.02	-0.00	1.00	-0.02	0.00	0.03	0.00	-0.02	0.01	-0.00	-0.01	0.01	0.03
px_height	0.01	-0.01	-0.01	-0.02	-0.01	-0.02	0.01	0.03	0.00	-0.01	-0.02	1.00	0.51	-0.02	0.06	0.04	-0.01	-0.03	0.02	0.05	0.15
px_width	-0.01	-0.04	-0.01	0.01	-0.01	0.01	-0.01	0.02	0.00	0.02	0.00	0.51	1.00	0.00	0.02	0.03	0.01	0.00	-0.00	0.03	0.17
ram	-0.00	0.03	0.00	0.04	0.02	0.01	0.03	-0.01	-0.00	0.00	0.03	-0.02	0.00	1.00	0.02	0.04	0.01	0.02	-0.03	0.02	0.92
sc_h	-0.03	-0.00	-0.03	-0.01	-0.01	0.03	0.04	-0.03	-0.03	-0.00	0.00	0.06	0.02	0.02	1.00	0.51	-0.02	0.01	-0.02	0.03	0.02
sc_w	-0.02	0.00	-0.01	-0.02	-0.01	0.04	0.01	-0.02	-0.02	0.03	-0.02	0.04	0.03	0.04	0.51	1.00	-0.02	0.03	0.01	0.04	0.04
talk_time	0.05	0.01	-0.01	-0.04	-0.01	-0.05	-0.00	0.02	0.01	0.01	0.01	-0.01	0.01	0.01	-0.02	-0.02	1.00	-0.04	0.02	-0.03	0.02
three_g	0.01	-0.03	-0.05	-0.01	0.00	0.58	-0.01	-0.01	0.00	-0.01	-0.00	-0.03	0.00	0.02	0.01	0.03	-0.04	1.00	0.01	0.00	0.02
touch_screen	-0.01	0.01	0.02	-0.02	-0.01	0.02	-0.03	-0.00	-0.01	0.02	-0.01	0.02	-0.00	-0.03	-0.02	0.01	0.02	0.01	1.00	0.01	-0.03
wifi	-0.01	-0.02	-0.02	0.02	0.02	-0.02	0.01	-0.03	-0.00	-0.01	0.01	0.05	0.03	0.02	0.03	0.04	-0.03	0.00	0.01	1.00	0.02
price_range	0.20	0.02	-0.01	0.02	0.02	0.01	0.04	0.00	-0.03	0.00	0.03	0.15	0.17	0.92	0.02	0.04	0.02	0.02	-0.03	0.02	1.00

FEATURE SELECTION





TRAIN TEST SPLIT

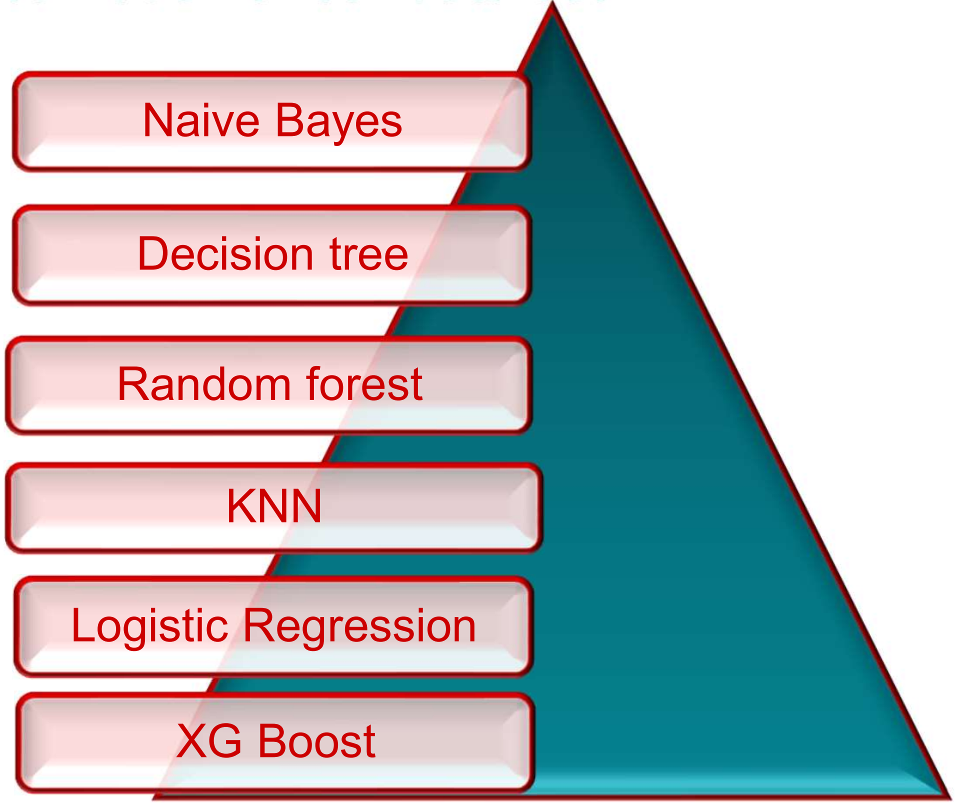
- Before, fitting any model it is a rule of thumb to split the dataset into a training and test set.

- This means some proportions of the data will go into training the model and some portion will be used to evaluate how our model is performing on any unseen data.

- The proportions may vary from 60:40, 70:30, 80:20 depending on the person. But mostly used is 80:20 for training and testing respectively. In this step we will split our data into training and testing set using Sickit learn library.

IMPLEMENTATION OF ML ALGORITHMS

To predict the mobile phone prices, we are going to apply below algorithms respectively on the training and validation dataset. After that, we are going to choose the best model for our data set.



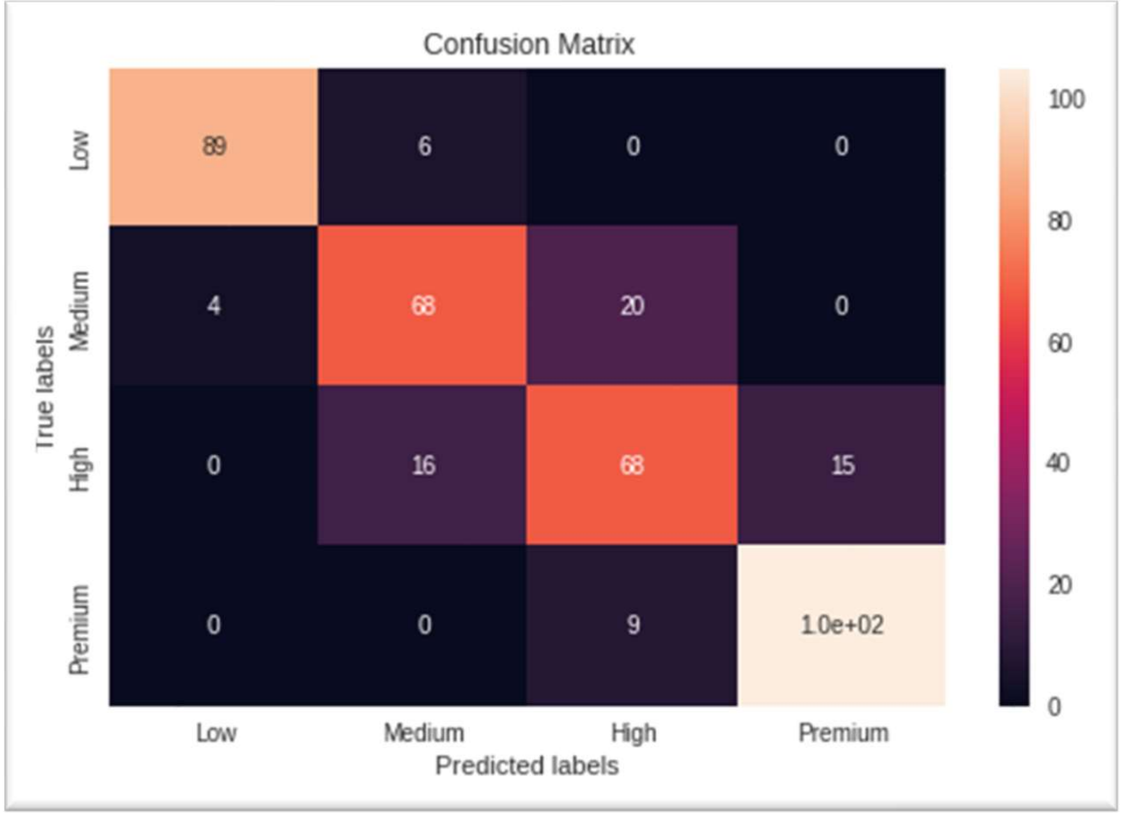
- Naive Bayes
- Decision tree
- Random forest
- KNN
- Logistic Regression
- XG Boost

Naive Bayes

Evaluation metrices on the train data Accuracy : 0.816875 ,Recall : 0.816875 Precision : 0.8177958342933045, F1 : 0.817203795640038 [0.816875, 0.816875, 0.8177958342933045, 0.817203795640038

Naive Bayes is a supervised machine learning model majorly used in solving classification problems. Supervised machine learning models are those where we use in text classification that includes a high-dimensional training dataset.

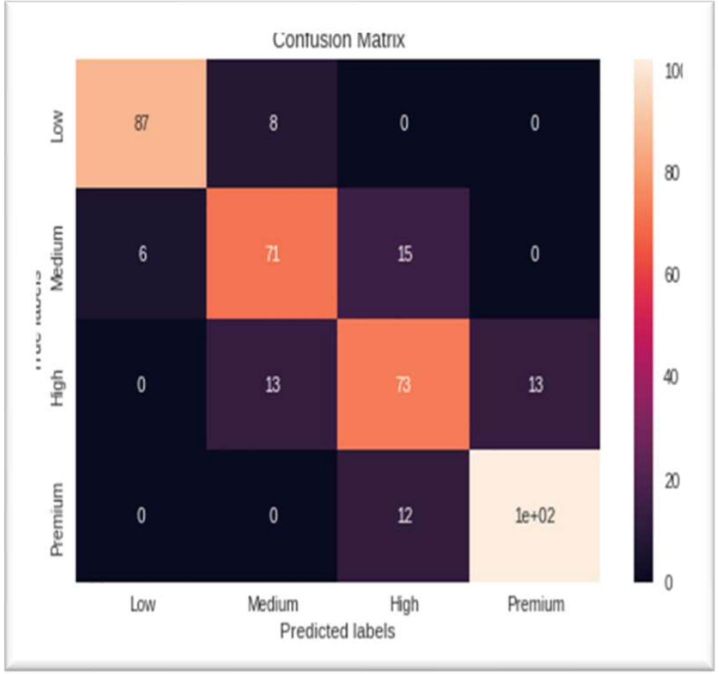
Evaluation metrices on the test data Accuracy : 0.825 ,Recall : 0.825, Precision : 0.8239428786535121 ,F1 : [0.825, 0.825, 0.8239428786535121, 0.8242390777915095]



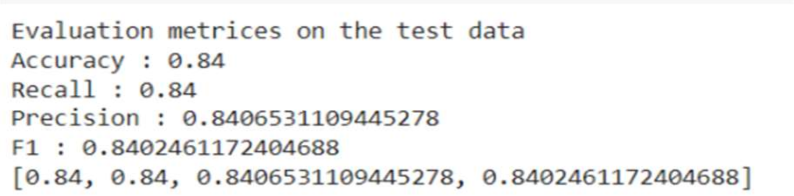
Confusion Matrix

True labels \ Predicted labels	Low	Medium	High	Premium
Low	89	6	0	0
Medium	4	68	20	0
High	0	16	68	15
Premium	0	0	9	1.0e+02

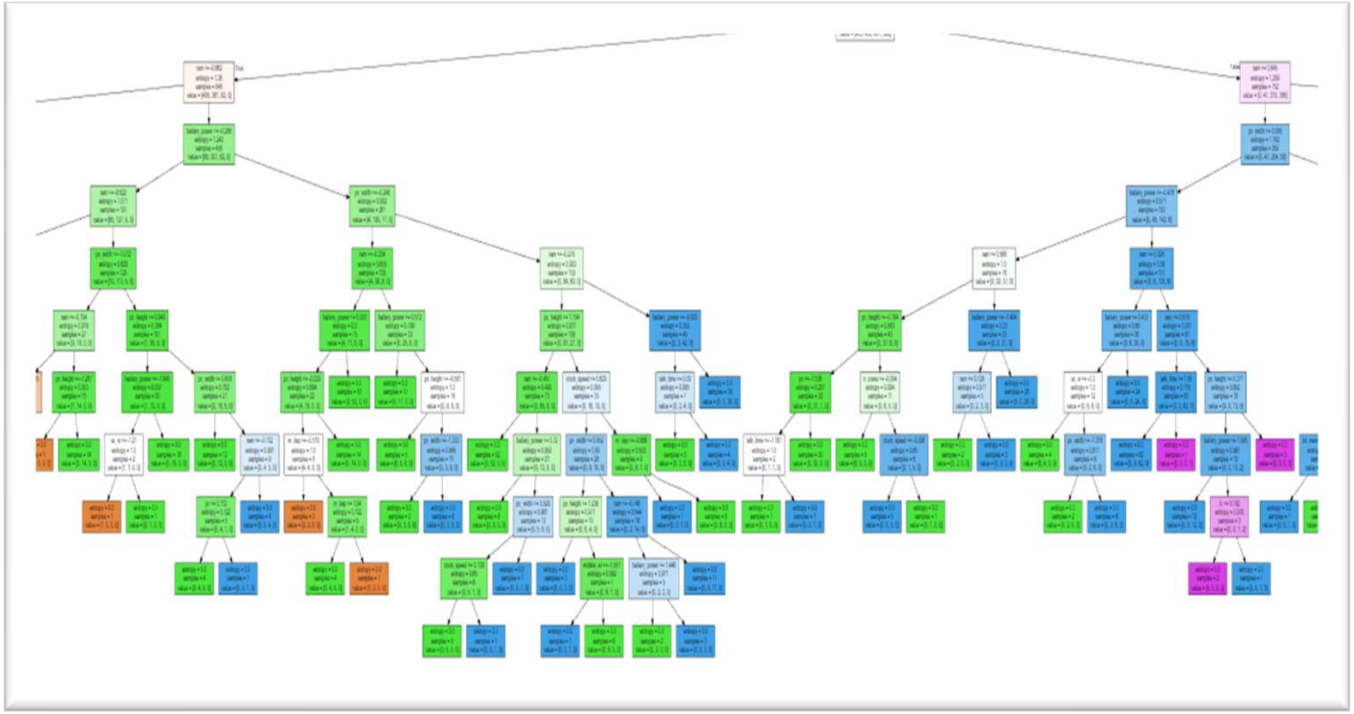
DECISION TREE



Decision tree is the most powerful and popular tool to deal with classification problem. A Decision tree is a flowchart like tree structure, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.



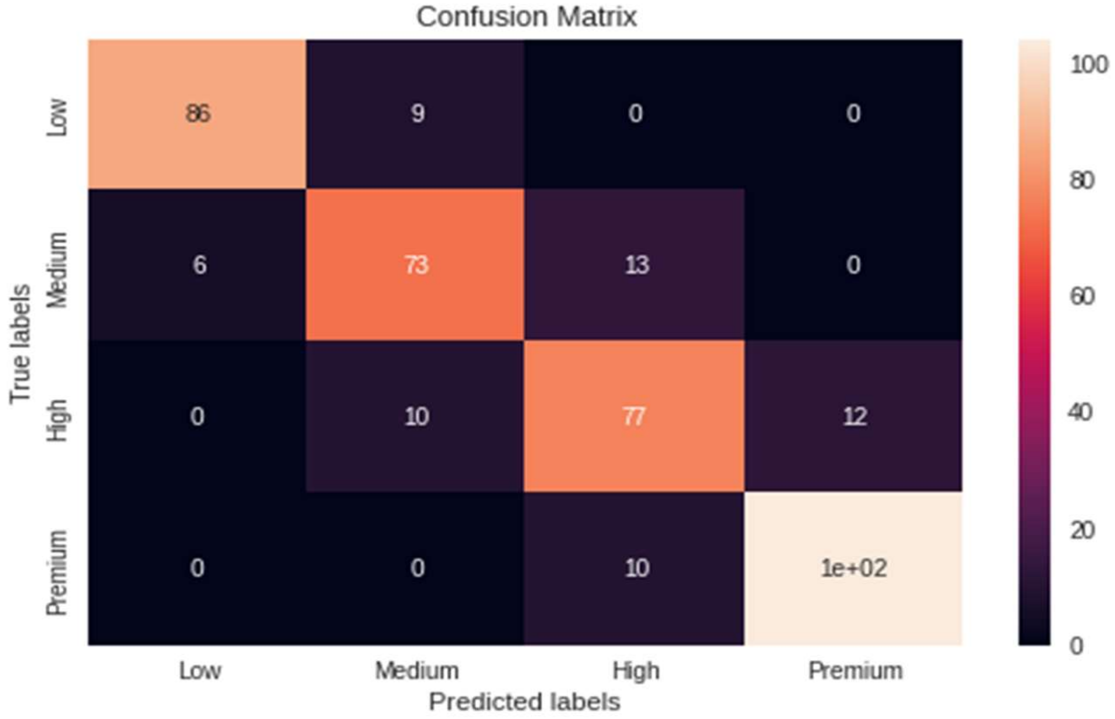
```
Evaluation metrices on the test data
Accuracy : 0.84
Recall : 0.84
Precision : 0.8406531109445278
F1 : 0.8402461172404688
[0.84, 0.84, 0.8406531109445278, 0.8402461172404688]
```



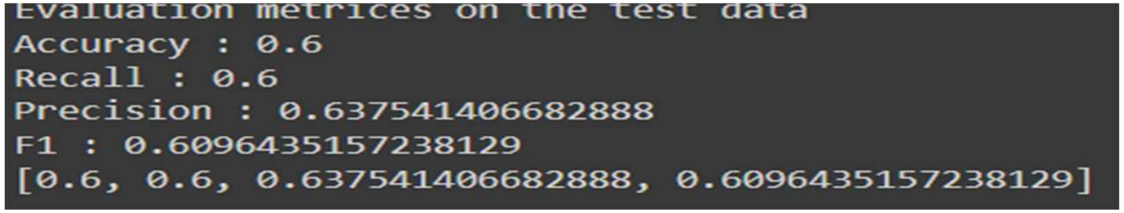
RANDOM FOREST

Random Forest is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of all.

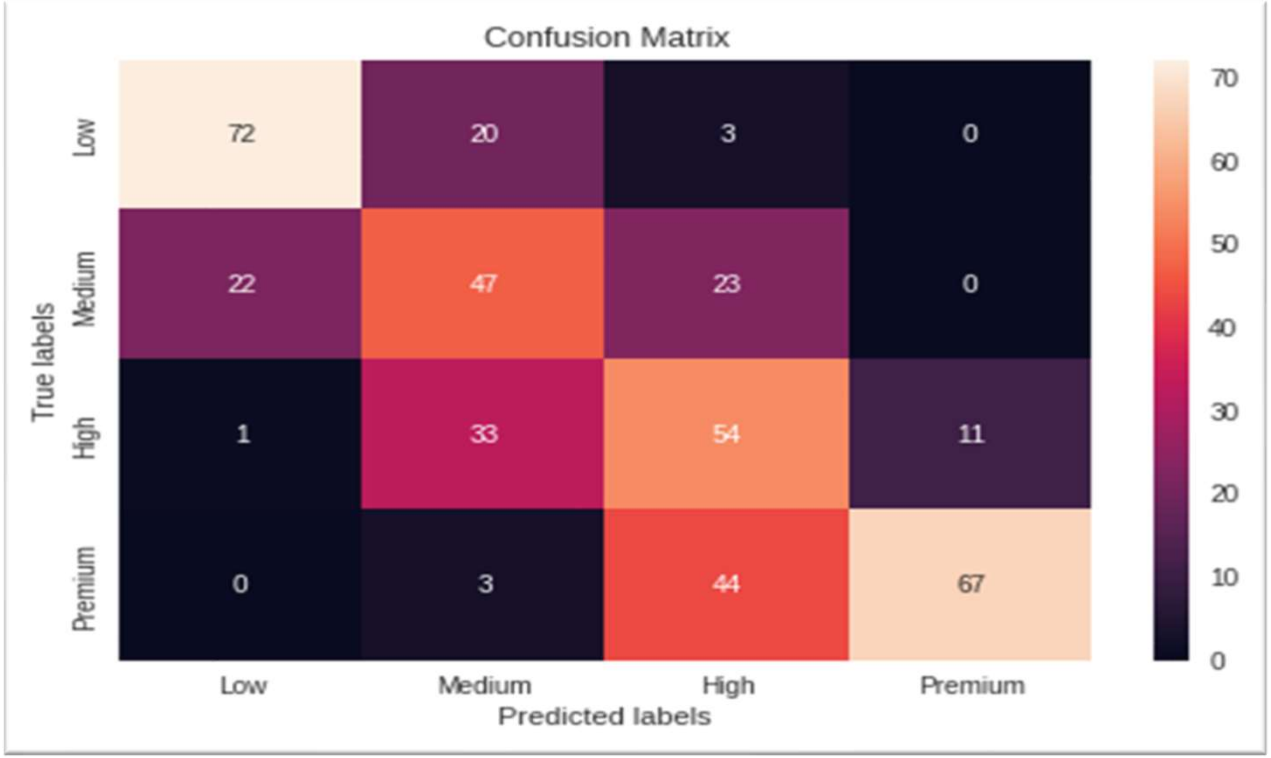
```
Evaluation metrices on the test data
Accuracy : 0.85 Recall : 0.85 Precision :
0.8506031109445277 F1 :
0.850220991612328
[0.85, 0.85, 0.8506031109445277,
0.850220991612328]
```



KNN

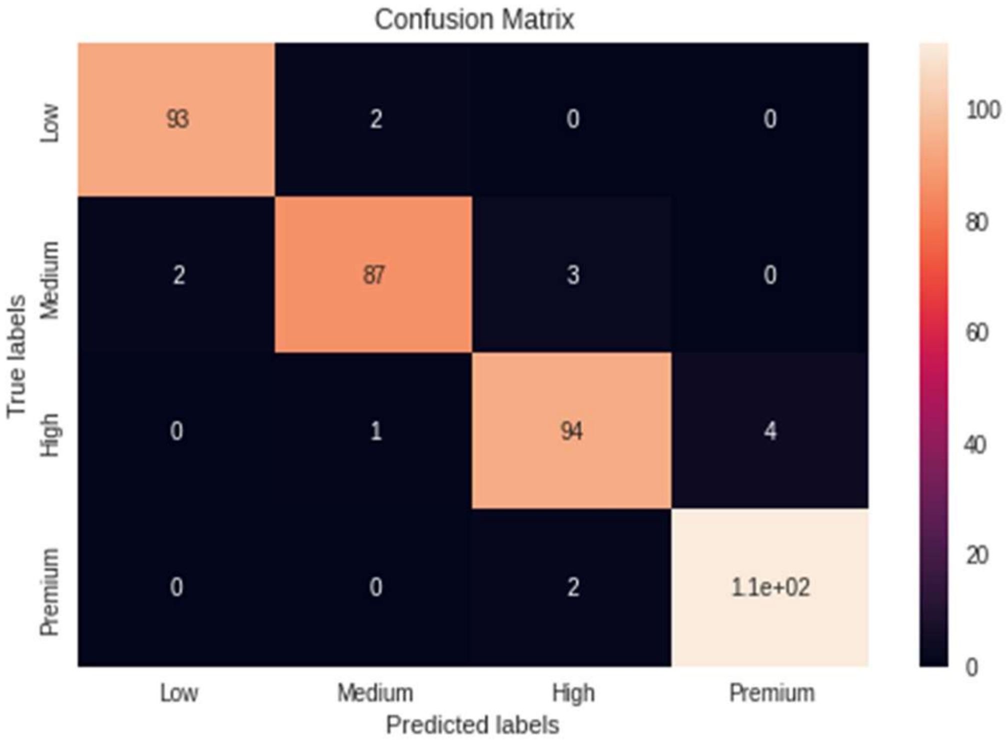


```
Evaluation metrices on the test data
Accuracy : 0.6
Recall : 0.6
Precision : 0.637541406682888
F1 : 0.6096435157238129
[0.6, 0.6, 0.637541406682888, 0.6096435157238129]
```

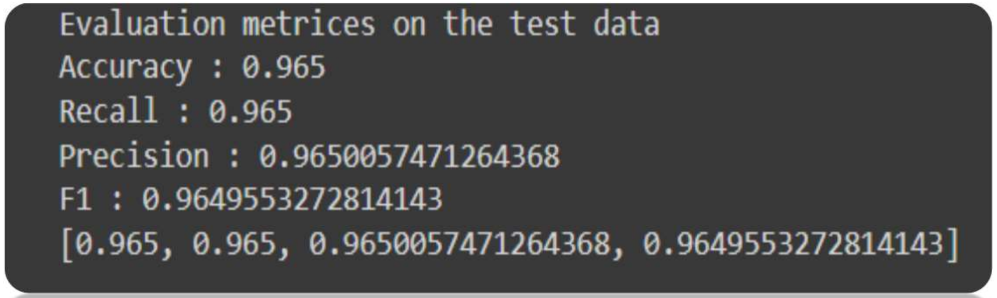


KNN is a method which is used for classifying objects based on closest training examples in the feature space. KNN is the most basic type of instance based learning or lazy learning. It assumes all instances are points in n dimensional space. A distance measure is needed to determine the “closeness” of instances. It classifies an instance by finding its nearest neighbours and picking the most popular class among the neighbours.

LOGISTIC REGRESSION



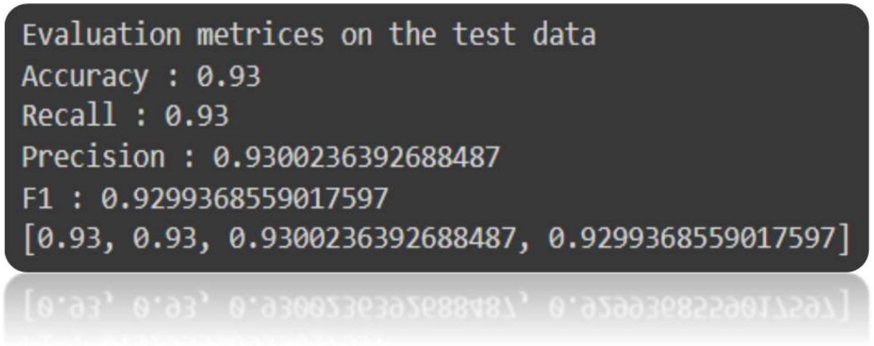
Logistic regression estimates the probability of an event occurring. Since the outcome is a probability, the dependent variable is bounded between 0 and 1



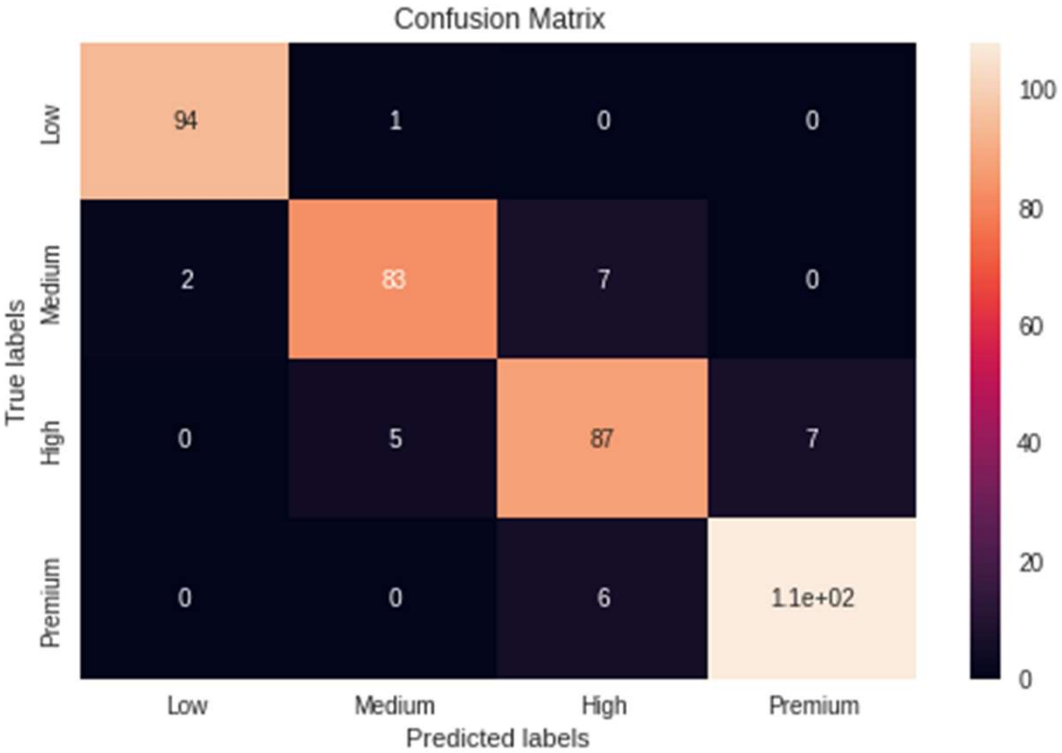
```
Evaluation metrices on the test data
Accuracy : 0.965
Recall : 0.965
Precision : 0.9650057471264368
F1 : 0.9649553272814143
[0.965, 0.965, 0.9650057471264368, 0.9649553272814143]
```

XG BOOST

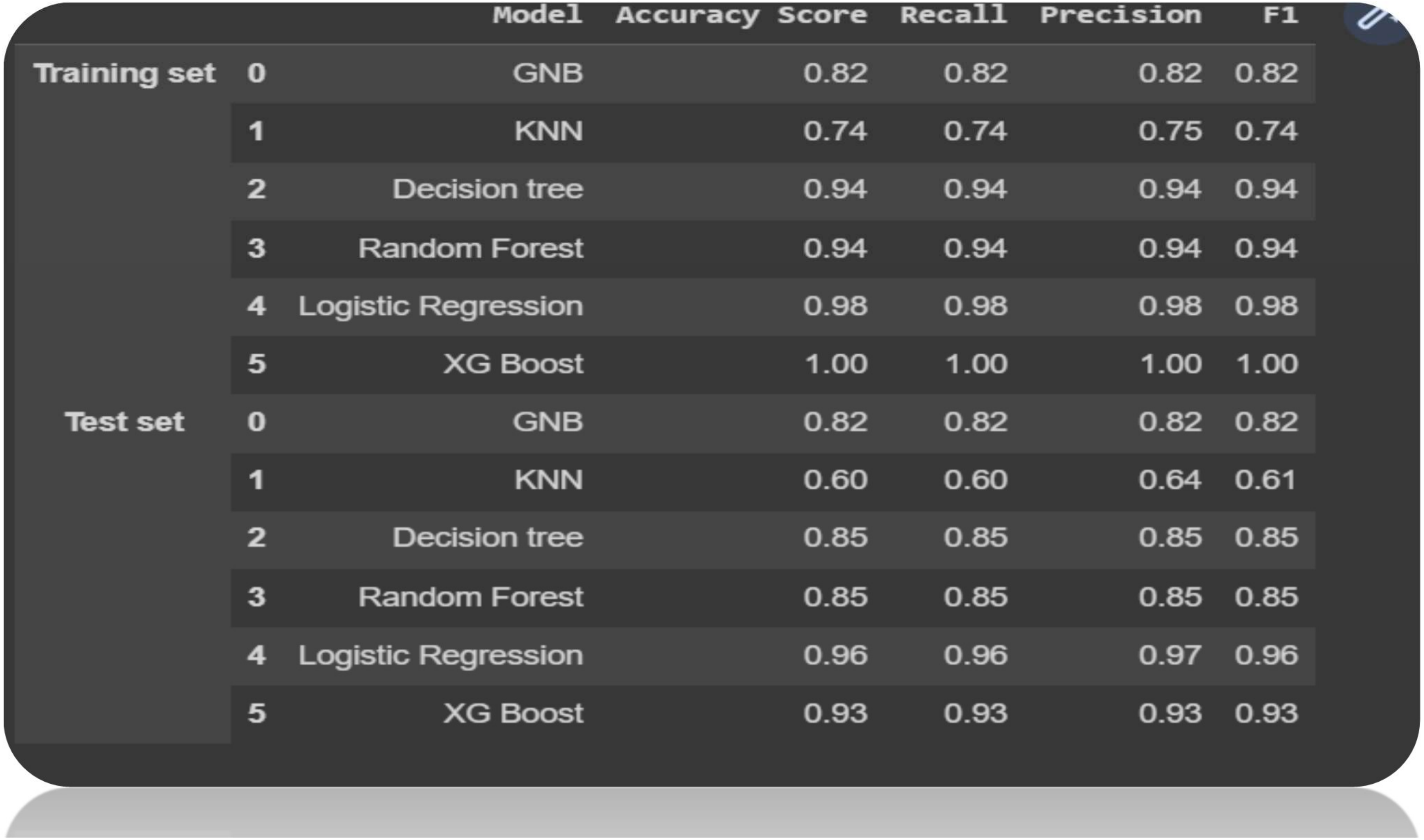
XG Boost stands for Extreme Gradient Boosting. The term gradient boosting consists of two sub-terms gradient and boosting.



```
Evaluation metrices on the test data
Accuracy : 0.93
Recall : 0.93
Precision : 0.9300236392688487
F1 : 0.9299368559017597
[0.93, 0.93, 0.9300236392688487, 0.9299368559017597]
```



MODEL PERFORMANCE



		Model	Accuracy Score	Recall	Precision	F1
Training set	0	GNB	0.82	0.82	0.82	0.82
	1	KNN	0.74	0.74	0.75	0.74
	2	Decision tree	0.94	0.94	0.94	0.94
	3	Random Forest	0.94	0.94	0.94	0.94
	4	Logistic Regression	0.98	0.98	0.98	0.98
	5	XG Boost	1.00	1.00	1.00	1.00
Test set	0	GNB	0.82	0.82	0.82	0.82
	1	KNN	0.60	0.60	0.64	0.61
	2	Decision tree	0.85	0.85	0.85	0.85
	3	Random Forest	0.85	0.85	0.85	0.85
	4	Logistic Regression	0.96	0.96	0.97	0.96
	5	XG Boost	0.93	0.93	0.93	0.93

CHALLENGES

Most of the models are not able to get good accuracy for each class of target variable.

With hyperparameter tuning, even after assigning different parameters values XG boost performed not so good on test data but It works really well on training set.

CONCLUSION

- XG Boost is giving us good overall accuracy but they didn’t perform well on Individual classes.
- Ram has continuous increase with price range while moving from Low cost to Very high cost .
- Costly phones are lighter .
- RAM, battery power, pixels played more significant role in deciding the price range of mobile phone.
- Out of all the model we have tried logistic regression is performing well on overall as well as Individual classes.
- Most of the mis-classifications were encountered between Medium range phones and high range phones.
- To counter that we can train a specific model for these two classes and can reclassify the cases when base model predicts the result as Medium range or High range

thank you