Capstone Project-3

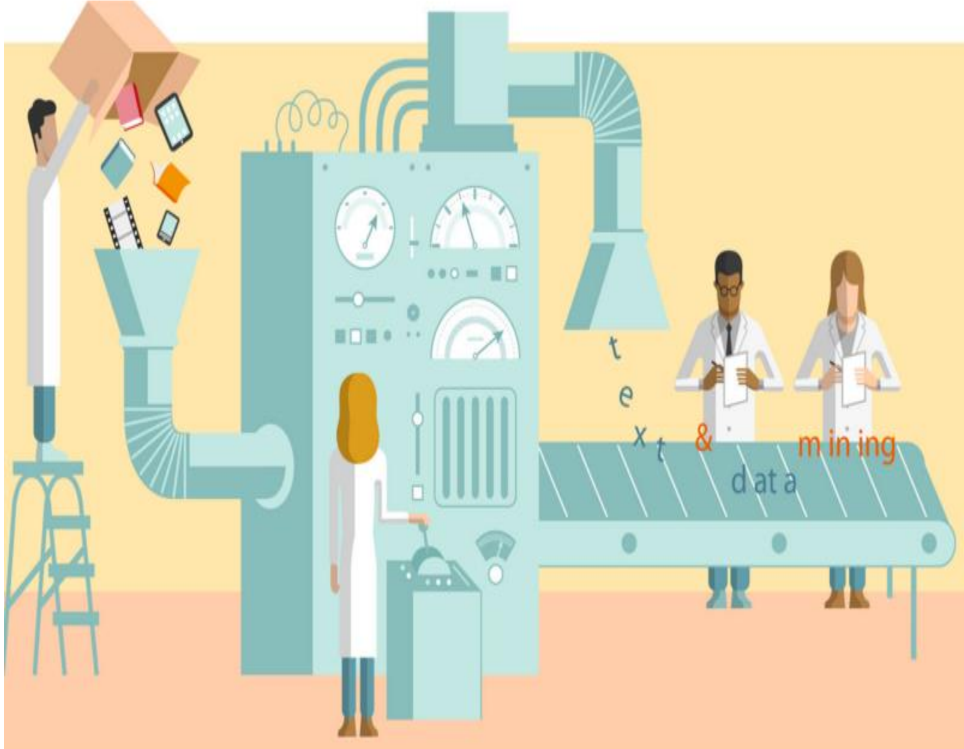
Presentation on

Mobile Price Range Prediction

Presented By: Ankit Rai

TABLE OF CONTENT

- Defining Problem Statement
- Data Pipeline
- Data Summary
- EDA
- Data preparation for Modelling
- Applying Model
- Model Validation and Selection
- Conclusion



PROBLEM STATEMENT

- Today, phone market is considered as highly competitive, and one must do adequate research before entering or launching their new product. Cell phone market companies want to understand sales data of mobile phones and factors which drive the prices.

- Our objective is to find out some relation between given features of a mobile phone(e.g.:- RAM, Internal Memory, etc.) and its selling price. Further , we will create a predictive model which could predict the range indicating how high the price is.

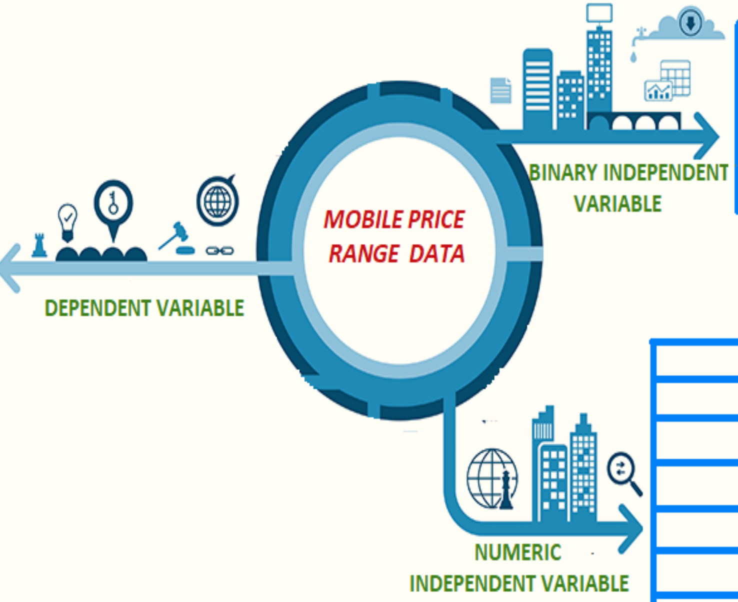
DATA PIPELINE

- **Data Processing :** First we checked for null values, duplicate values and outliers in our data. Then we derived new variables from the existing variables and also dropped the variables which were not further required. We also performed standardisation of our variable values in order to generalise them .

- **EDA :** In this part we performed exploratory data analysis on the given data. We started with univariate analysis of our dependent feature and then performed multivariate analysis of binary and numeric independent variables.

- **Create a model :** Finally after splitting our data and then rescaling training data values, we started creating classifier models. After training our data, we also evaluated our different models on the basis of their accuracy.
- **Hyperparameter tuning :** After selecting our best model on the basis of their accuracy and CV score, we optimized our models tuning its hyperparameter.

DATA SUMMARY



MOBILE PRICE RANGE DATA

DEPENDENT VARIABLE

BINARY INDEPENDENT VARIABLE

NUMERIC INDEPENDENT VARIABLE

Price_range - This is the target variable with value of 0(low cost), 1(medium cost), 2(high cost) and 3(very high cost).

Blue - Has bluetooth or not
Dual_sim - Has dual sim support or not
Four_g - Has 4G or not
Three_g - Has 3G or not
Touch_screen - Has touch screen or not
Wifi - Has wifi or not

Battery_power - Total energy a battery can store in one time measured in mAh
Clock_speed - speed at which microprocessor executes instructions
Fc - Front Camera mega pixels
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
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 when you are

Data summary

Binary Independent Variable

- **Blue –** Tells whether given phone Has Bluetooth or not
- **Dual_sim -** Has dual sim support or not
- **Four_g –** The feature shows that a particular mobile has 4G support or not.
- **Touch_screen –** Tells whether given phone has touch screen feature or not
- **Wifi –** Tells whether the given phone has Wifi or not
- **Three_g -**The feature shows that a particular mobile has 3G support or not.

Data summary

Numeric Independent Variables

- **Battery_power -** Total energy a battery can store in one time measured in mAh
- **Clock_speed -** speed at which microprocessor executes instructions
- **RAM –** It is the Random Access Memory measured in Mega Bytes.
- **Int_memory –** This feature gives the Internal Memory of mobiles in Gigabytes.
- **Mobile_wt –** This variable tells about the Weight of mobile phones.
- **Fc –** Tells number of front camera megapixels
- **Pc –** Tells megapixels of Primary Camera
- **N_cores –** Shows the number of cores present in a given phone
- **M_dep -** Mobile Depth in cm
- **Px_height -** Pixel Resolution Height

Data summary

Numeric Independent Variables

- **Sc_h -** Screen Height of mobile in cm
- **Sc_w -** Screen Width of mobile in cm
- **Px_width -** Pixel Resolution Width
- **Talk Time-** longest time that a single battery charge will last

Dependent Variable

- **Price_range –** This is our dependent/target variable , which tells about the price range to which the given phone belongs.

New Derived Features

sc_area (Screen Area)

This area tells about the screen size or it’s area to be precise. We have derived this new variable from two other variables :

sc_h (Screen Height)
sc_w(Screen Width)

sc_area = sc_h * sc_w

px_size (Pixel Size)

This new variables talks about display pixels size. We derived this new variable from other two variables :

px_height (Pixel Height)
px_width (Pixel Width

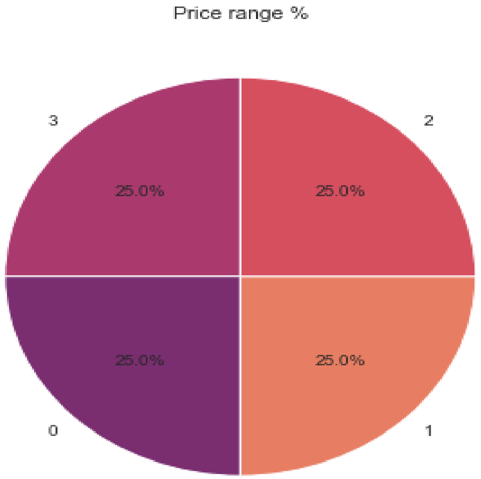
px_size = px_height * px_width

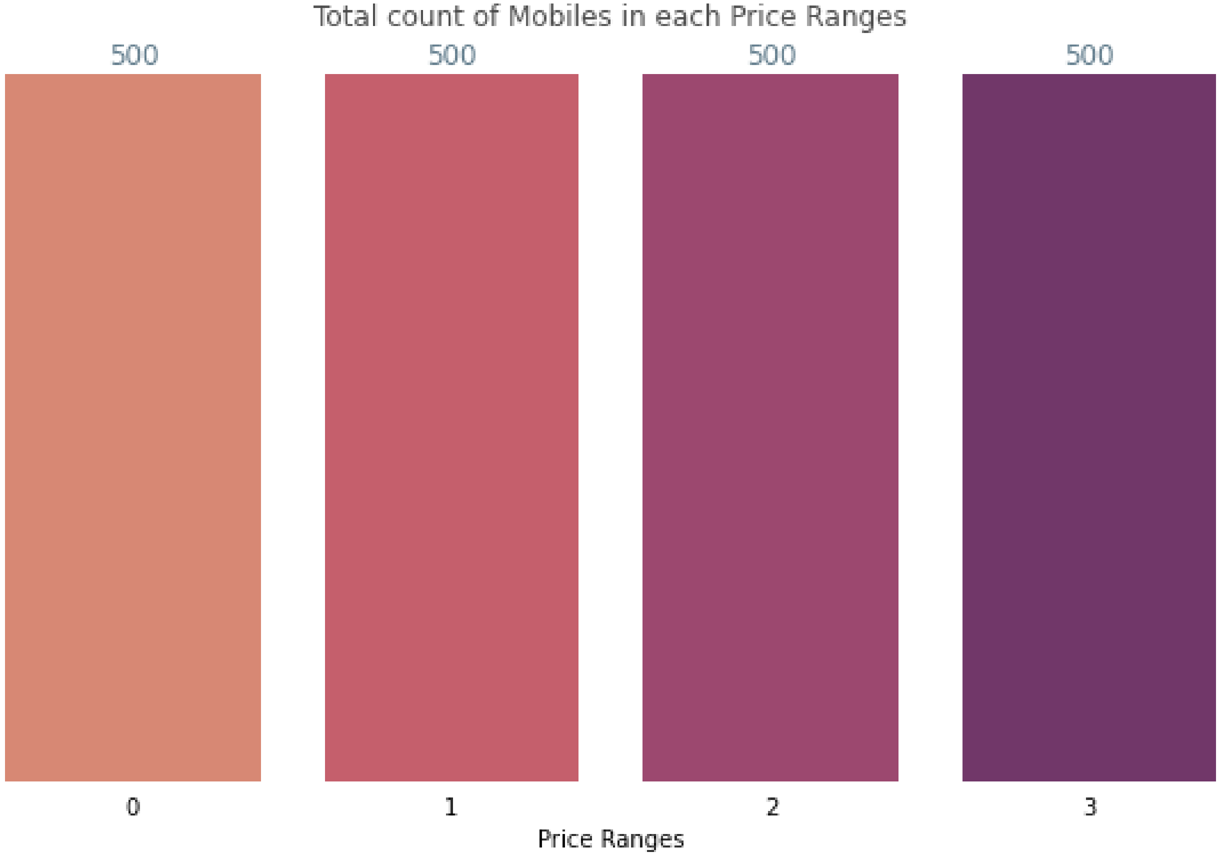
Define Dependent Variable

There are four types of price ranges :

0 : Low Cost range
1 : Medium Cost Range
2 : High Cost Range
3 : Very High Cost Range

All four classes are divided equally in our data.



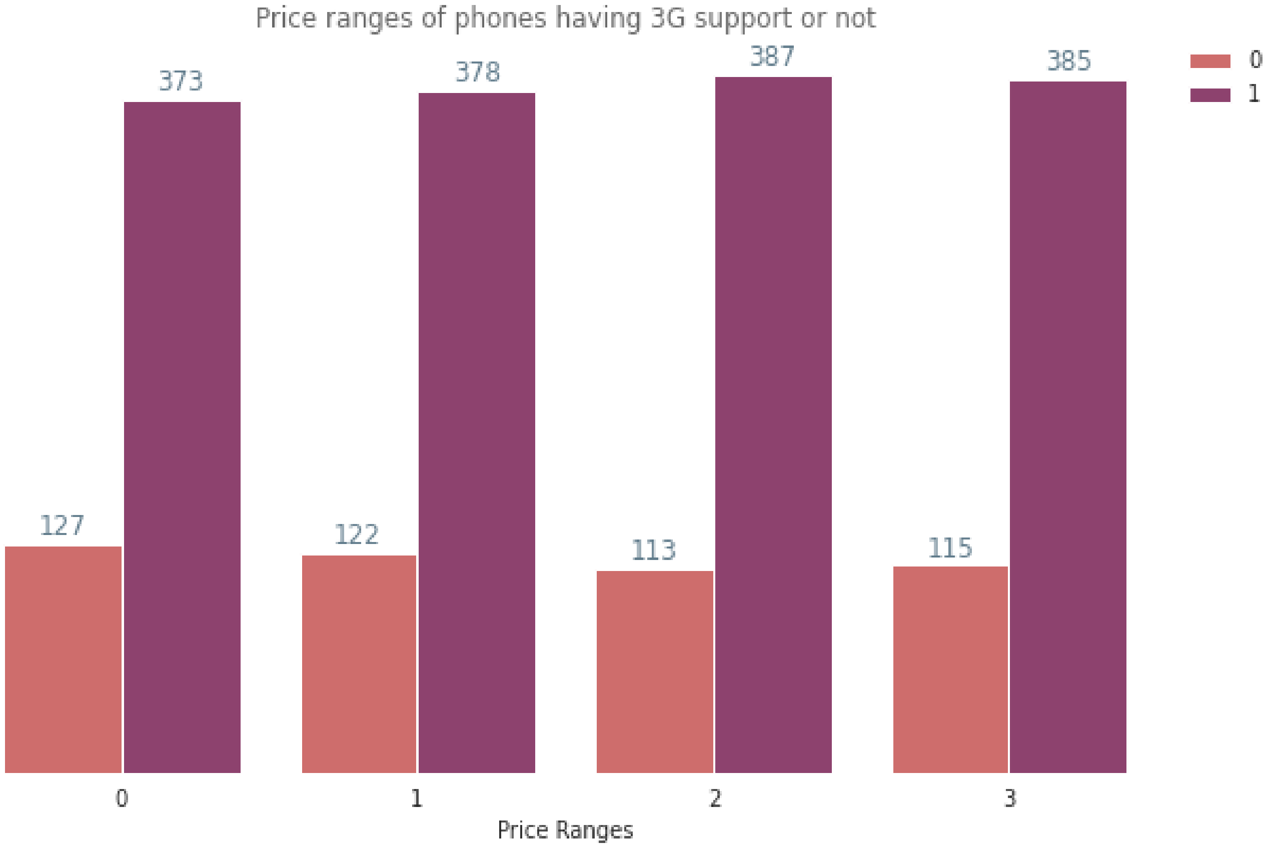


Exploratory Data Analysis

Three-G Support of mobiles

- Almost 66% of the phones in the market support 3g connectivity.

- For every price range, the ratio of phones having 3g connectivity to those which do not support 3g, are almost same.

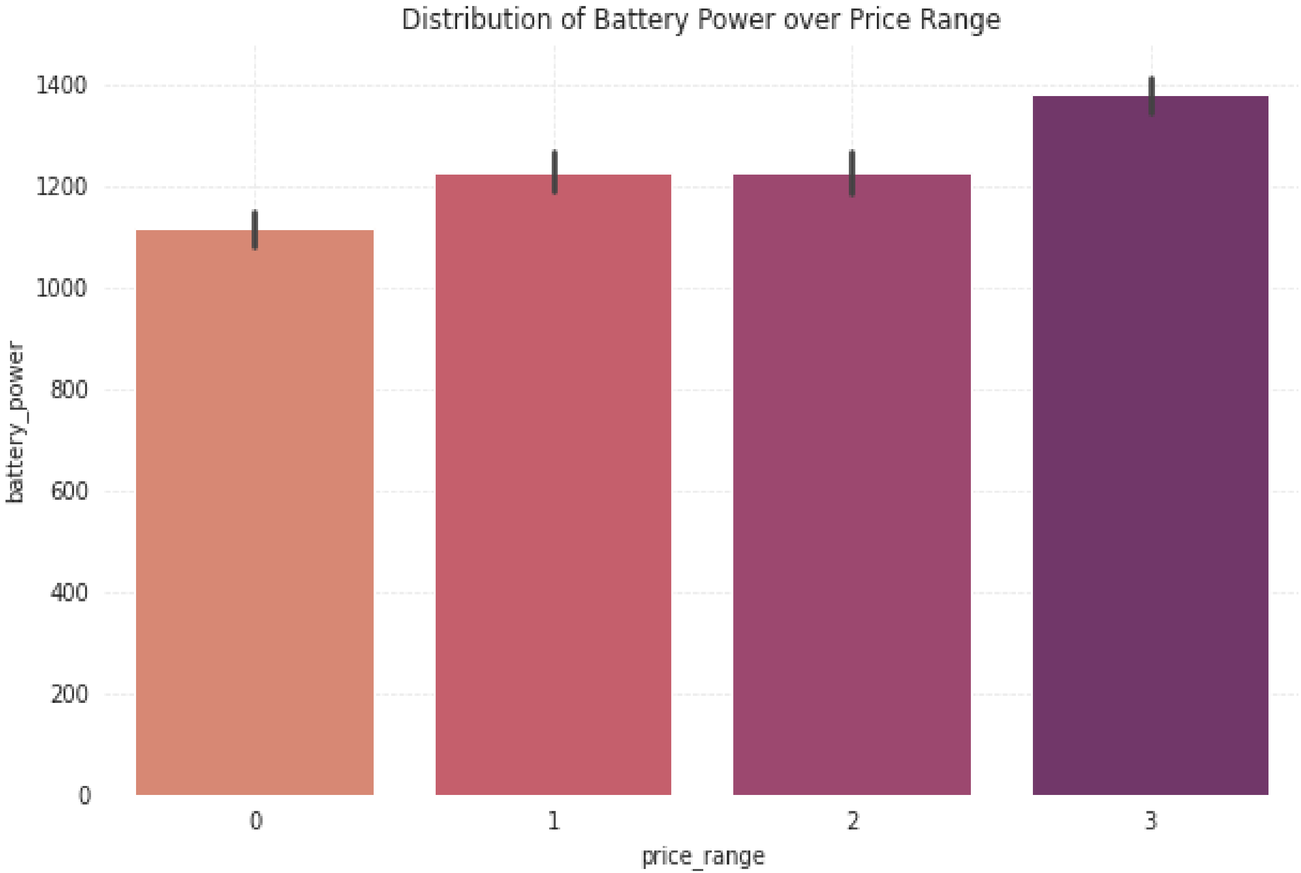


- Low battery power phones are of low cost, while very high battery powered phones are of high cost.

- It is interesting to note that medium and high cost phones have almost similar range of battery power.

Exploratory Data Analysis

Battery power with price ranges of mobiles



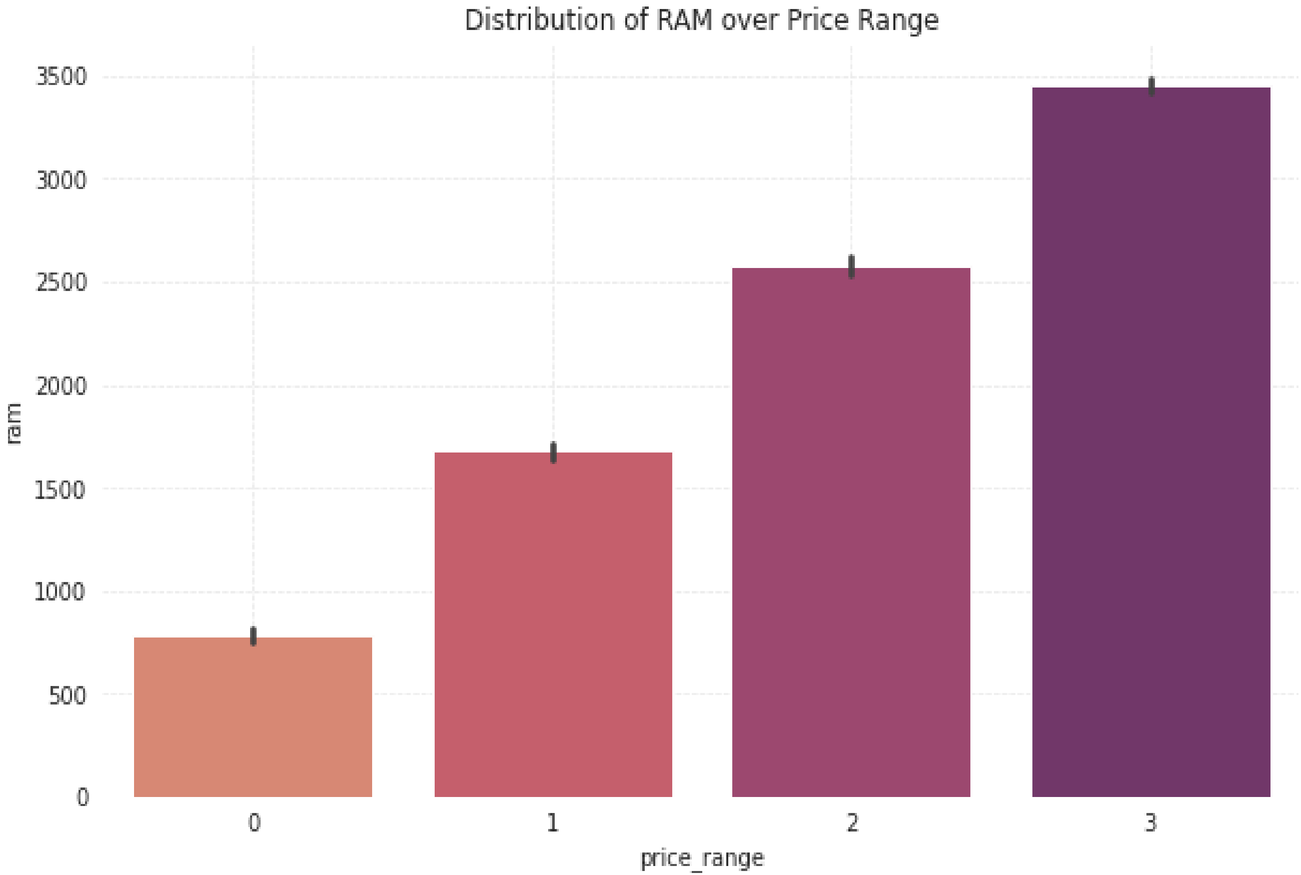
- The mobile phones having high RAM are of high price range and the phones having low RAM are of low price range.

- As RAM increases Price Range also increases and as RAM decreases Price Range also decreases.

- From this we can say that in cell phone market RAM is game changing feature which decides the price range of mobiles.

Exploratory Data Analysis

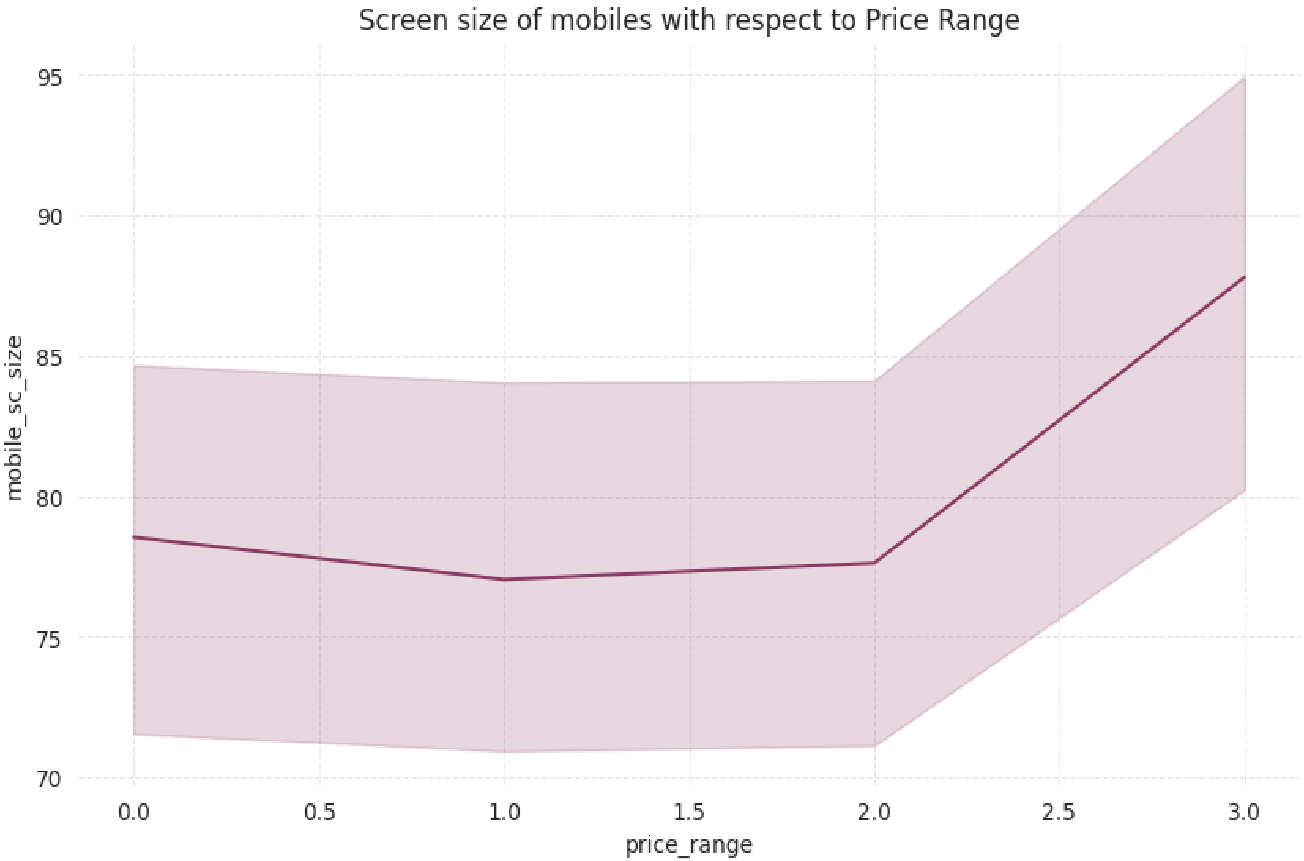
RAM of Mobiles with each Price Range



- Low , medium and high cost phones have nearly same screen size.
- Mobile phones with larger screen area have very high cost.

Exploratory Data Analysis

Screen Size of mobiles with respect to Price Range



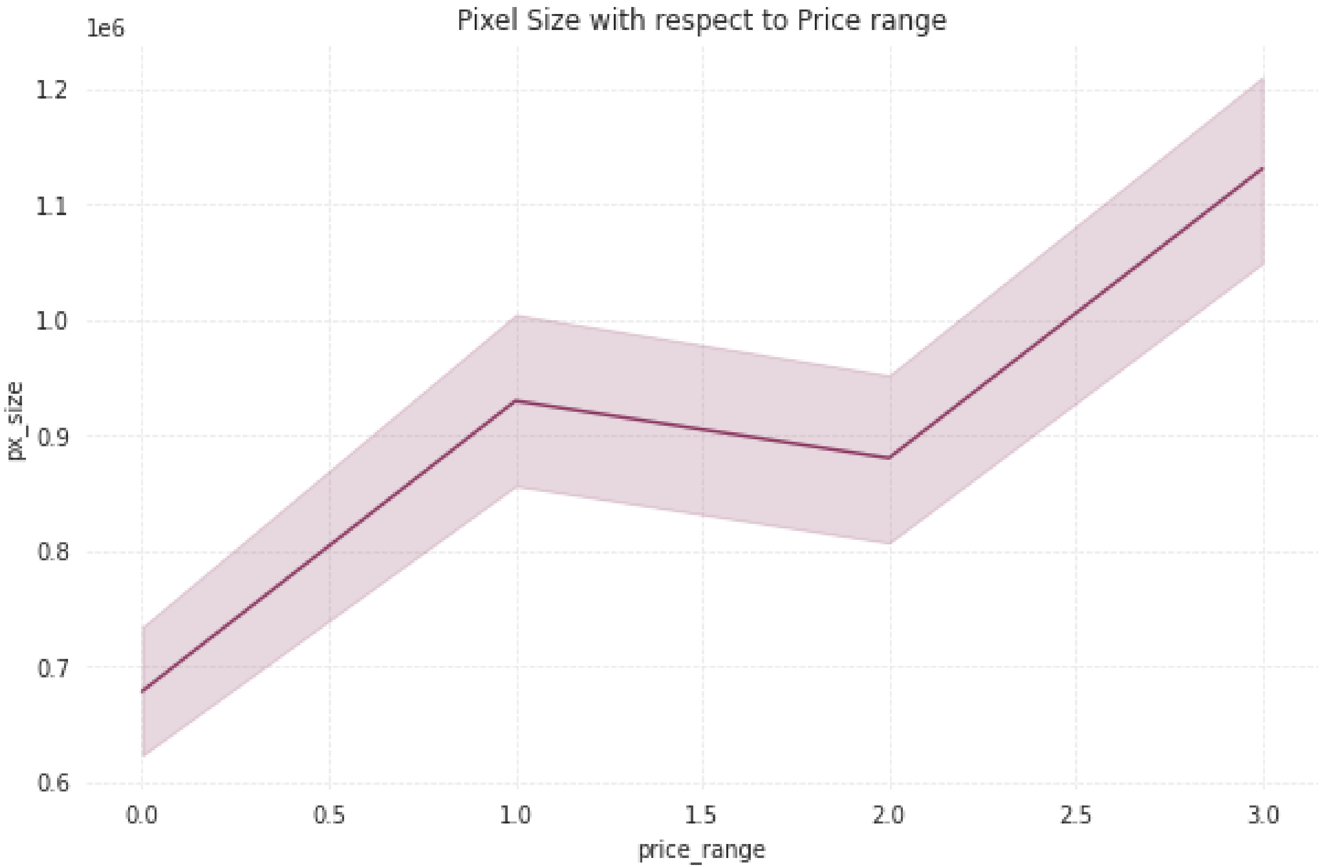
- For low price range phones pixel size does matter that means higher pixel size gives more price range.

- For medium price range it is not like that, for some medium price ranges pixel size does not matter that means some of medium range phones the pixel size is low compared to low price range.

- But in expensive price range the pixel is also high with respect to price range.

Exploratory Data Analysis

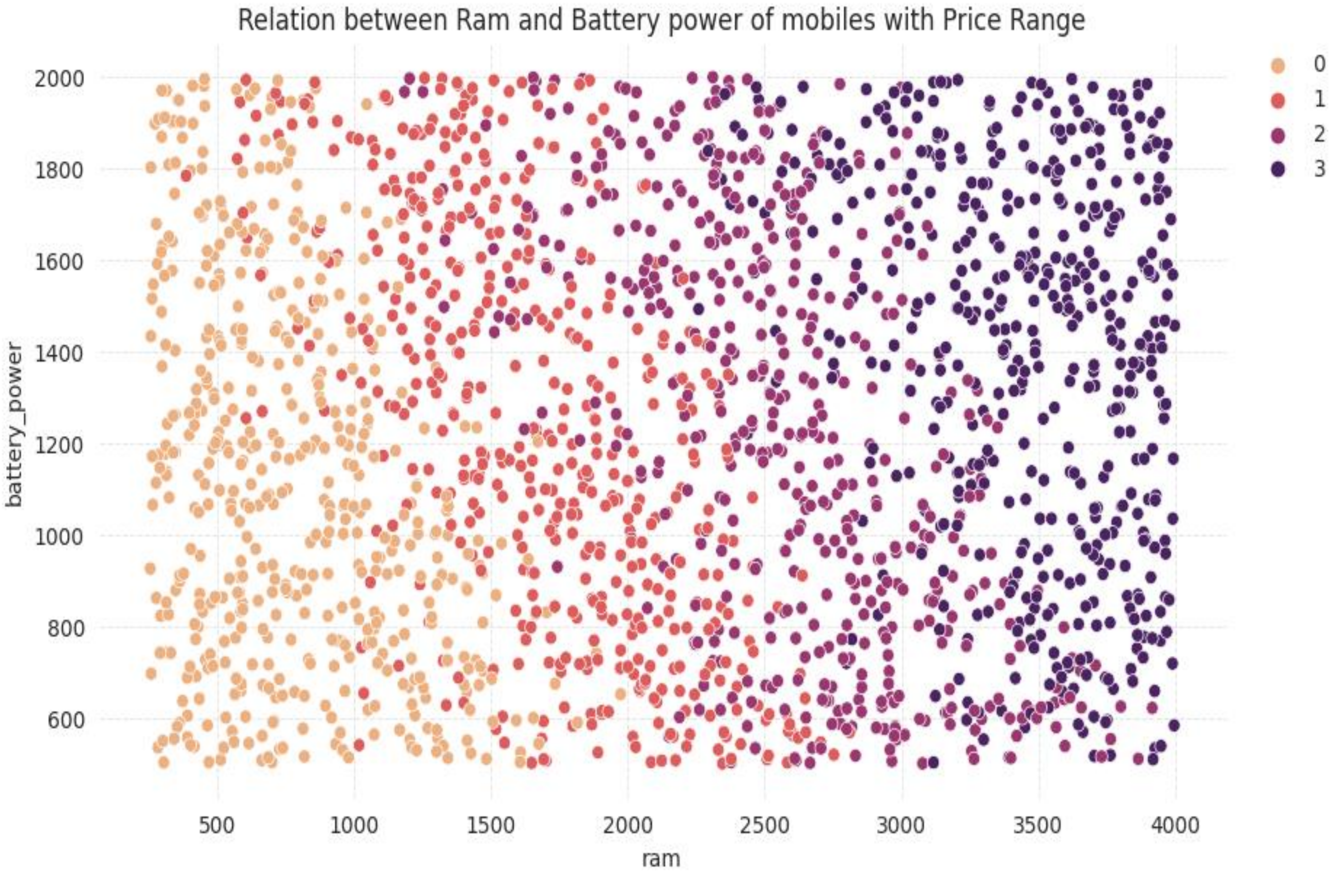
PIXEL SIZE OVER DIFFERENT PRICE RANGE



Exploratory Data Analysis

RAM and Battery Power of mobiles

- The scatter plot shows the relation of ram and battery power with respect to price range of mobiles.
- As we can see when the ram and battery power are less at that time price of that particular mobile is also low.
- And when battery power and ram are high the price range is also high.



Data Preparation for Modelling

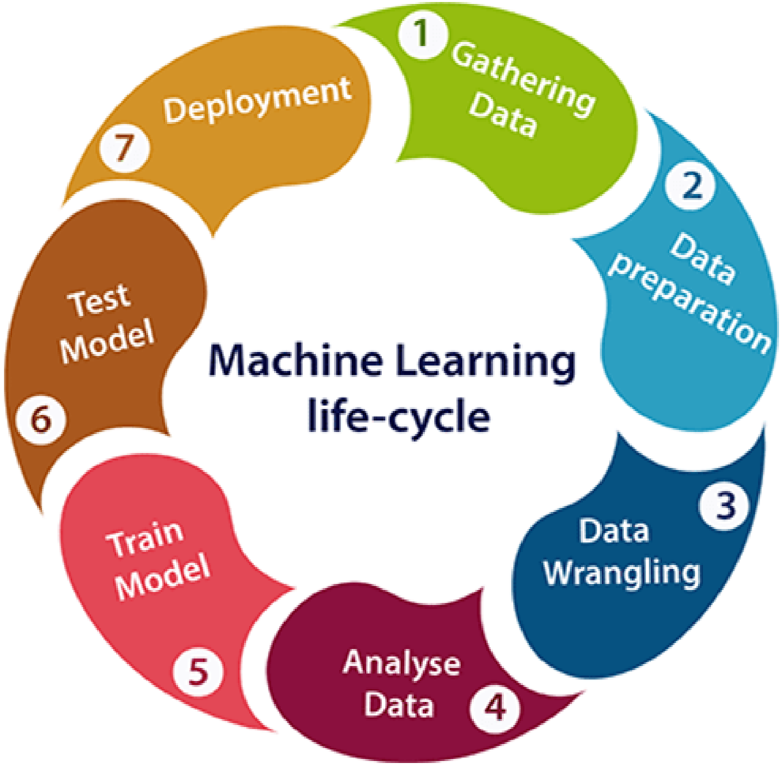
Splitting data for Training and Testing :

- First we assigned our dependent and independent variable values to new variables X and y respectively.

- We split 20% data for testing and rest for training.

Train Set: (1600,21)

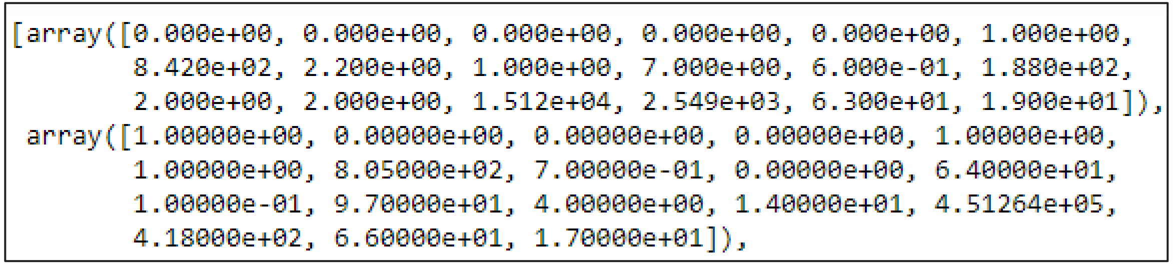
Test Set: (400,21)



Data Preparation for Modelling

Rescaling Values– Standardization

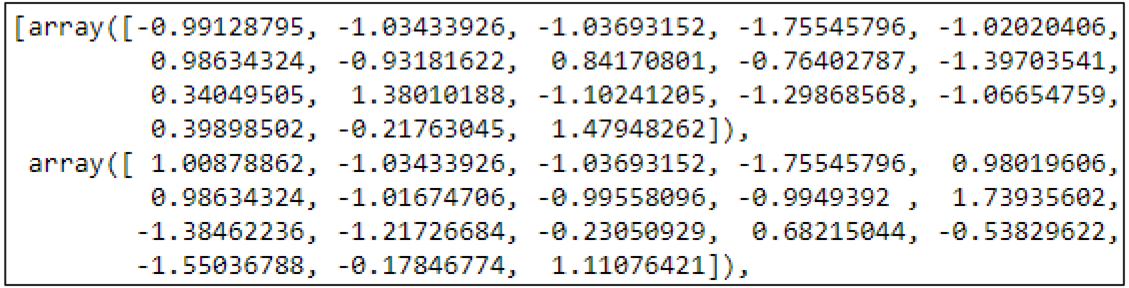
Before applying models, it is important to generalize our data. For this we will be using standardization method. Standardization gives all features the same influence on the distance metric. If one feature has very large values, it will dominate over other features when calculating the distance.



```
[array([0.000e+00, 0.000e+00, 0.000e+00, 0.000e+00, 0.000e+00, 1.000e+00,
        8.420e+02, 2.200e+00, 1.000e+00, 7.000e+00, 6.000e-01, 1.880e+02,
        2.000e+00, 2.000e+00, 1.512e+04, 2.549e+03, 6.300e+01, 1.900e+01]),
 array([1.00000e+00, 0.00000e+00, 0.00000e+00, 0.00000e+00, 1.00000e+00,
        1.00000e+00, 8.05000e+02, 7.00000e-01, 0.00000e+00, 6.40000e+01,
        1.00000e-01, 9.70000e+01, 4.00000e+00, 1.40000e+01, 4.51264e+05,
        4.18000e+02, 6.60000e+01, 1.70000e+01]),
```

Before Standardization ‘X_test’

After Standardization ‘X_test’

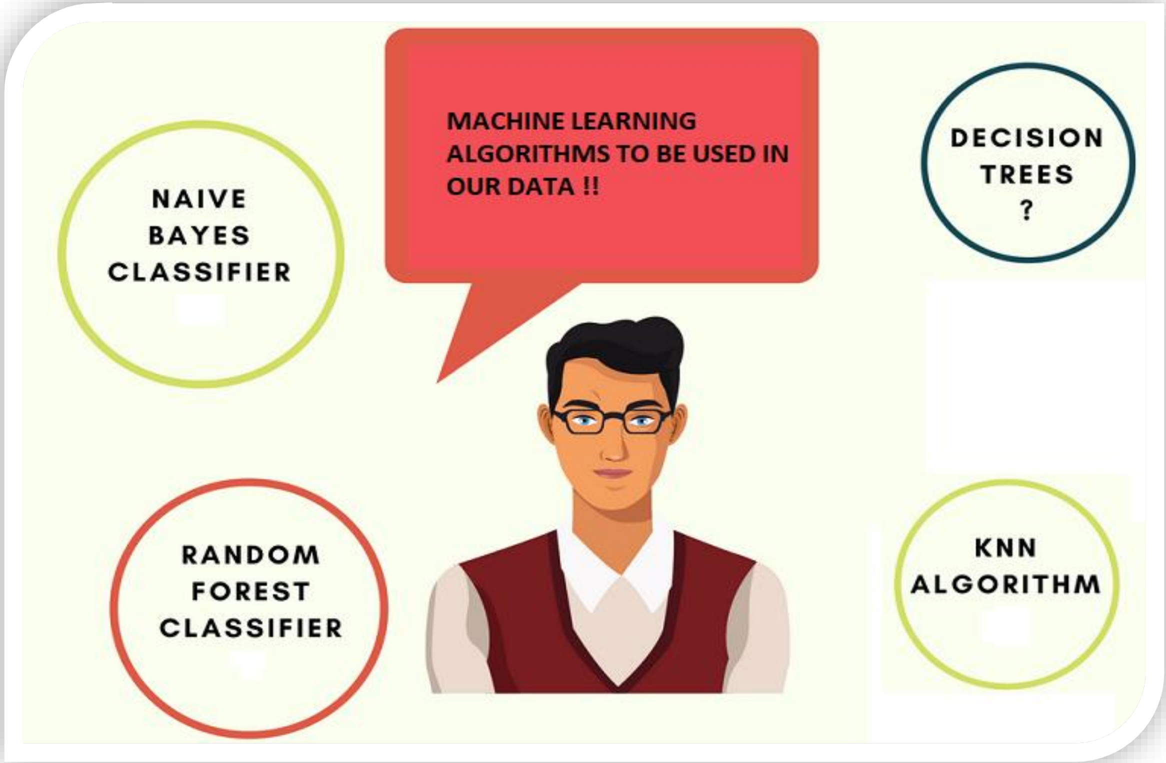


```
[array([-0.99128795, -1.03433926, -1.03693152, -1.75545796, -1.02020406,
         0.98634324, -0.93181622,  0.84170801, -0.76402787, -1.39703541,
         0.34049505,  1.38010188, -1.10241205, -1.29868568, -1.06654759,
         0.39898502, -0.21763045,  1.47948262]),
 array([ 1.00878862, -1.03433926, -1.03693152, -1.75545796,  0.98019606,
         0.98634324, -1.01674706, -0.99558096, -0.9949392 ,  1.73935602,
        -1.38462236, -1.21726684, -0.23050929,  0.68215044, -0.53829622,
        -1.55036788, -0.17846774,  1.11076421]),
```

Applying Model

Since our dependent variable is a multiclass variable, we preferred to go and try following models :

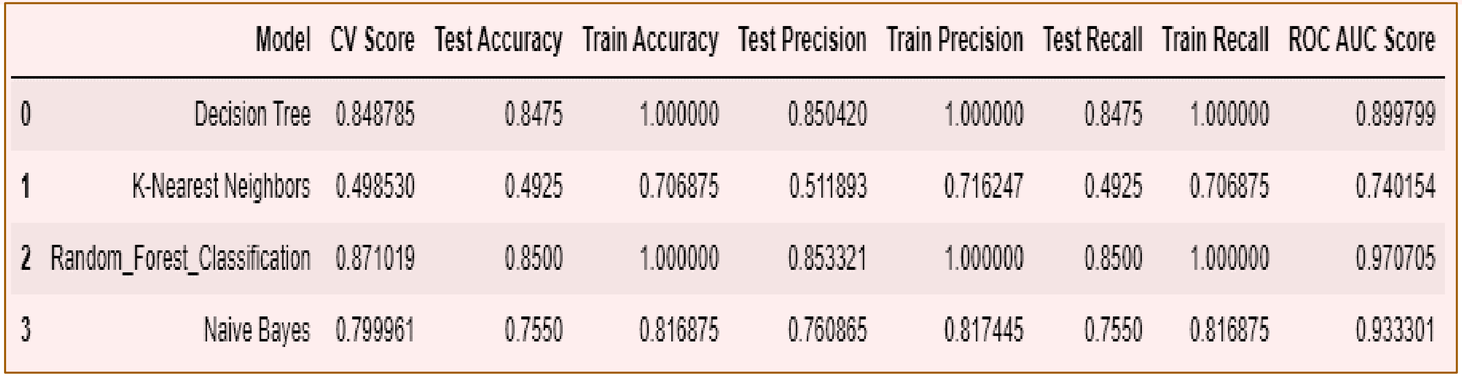
- Naive Bayes Classifier
- Decision Tree Algorithm
- Random Forest Classifier
- KNN Algorithm



Model Validation and Selection

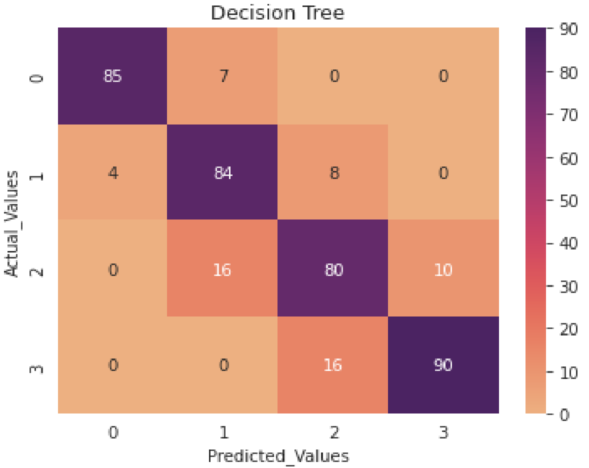
Observation : Decision Tree and Random Forest are giving best results.

However, Random Forest has given slightly better results in ROC AUC score.



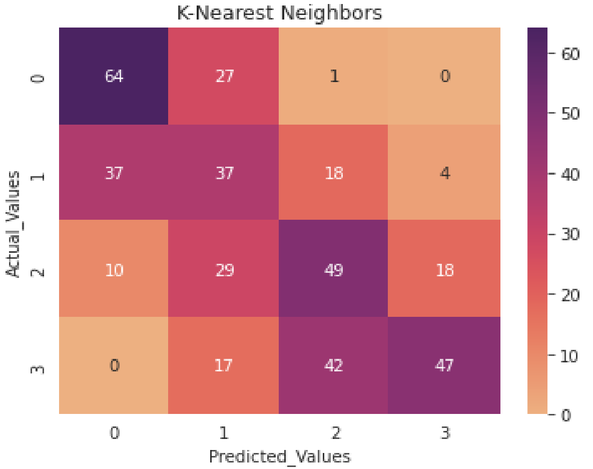
	Model	CV Score	Test Accuracy	Train Accuracy	Test Precision	Train Precision	Test Recall	Train Recall	ROC AUC Score
0	Decision Tree	0.848785	0.8475	1.000000	0.850420	1.000000	0.8475	1.000000	0.899799
1	K-Nearest Neighbors	0.498530	0.4925	0.706875	0.511893	0.716247	0.4925	0.706875	0.740154
2	Random_Forest_Classification	0.871019	0.8500	1.000000	0.853321	1.000000	0.8500	1.000000	0.970705
3	Naive Bayes	0.799961	0.7550	0.816875	0.760865	0.817445	0.7550	0.816875	0.933301

Confusion Matrix



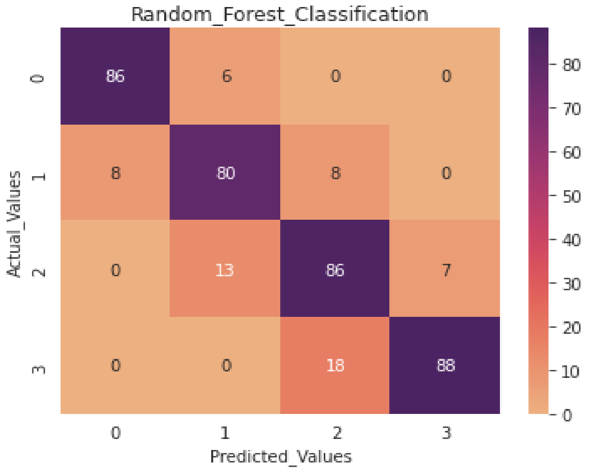
Decision Tree

Actual_Values \ Predicted_Values	0	1	2	3
0	85	7	0	0
1	4	84	8	0
2	0	16	80	10
3	0	0	16	90



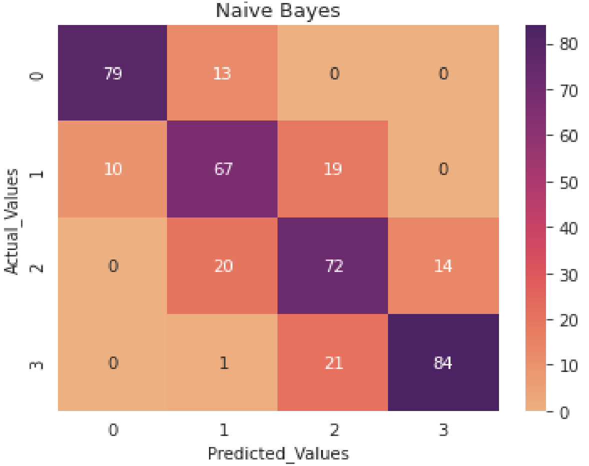
K-Nearest Neighbors

Actual_Values \ Predicted_Values	0	1	2	3
0	64	27	1	0
1	37	37	18	4
2	10	29	49	18
3	0	17	42	47



Random_Forest_Classification

Actual_Values \ Predicted_Values	0	1	2	3
0	86	6	0	0
1	8	80	8	0
2	0	13	86	7
3	0	0	18	88



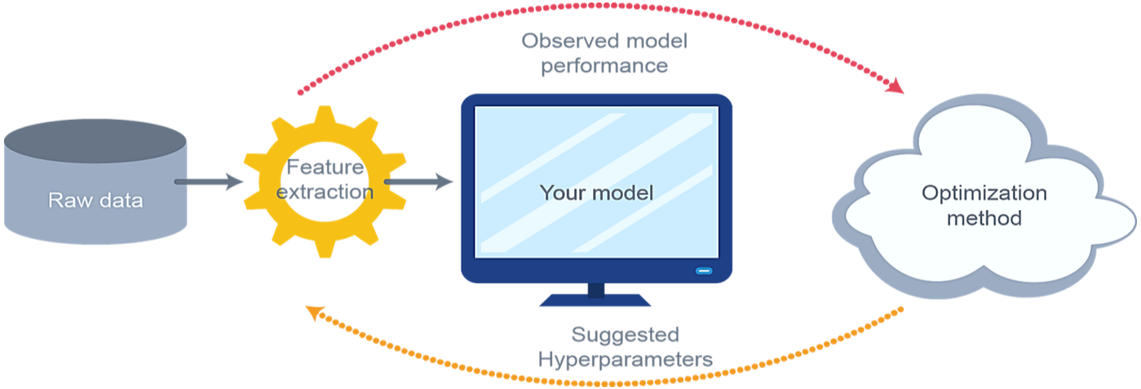
Naive Bayes

Actual_Values \ Predicted_Values	0	1	2	3
0	79	13	0	0
1	10	67	19	0
2	0	20	72	14
3	0	1	21	84

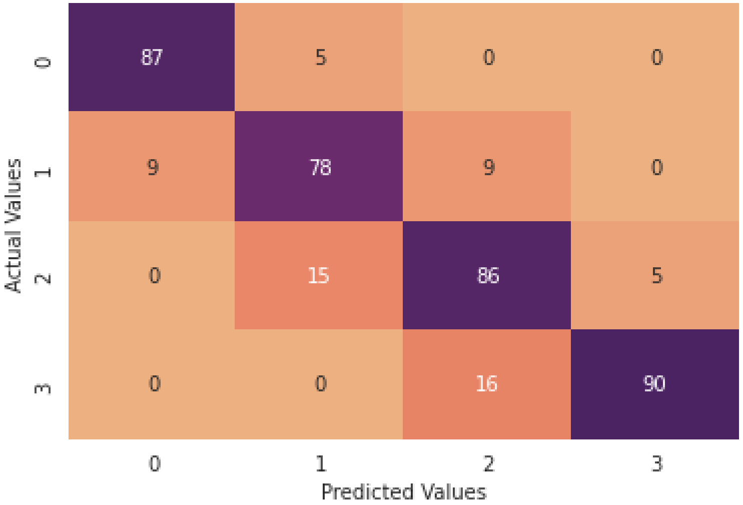
Hyperparameter Tuning

We had chosen Random Forest Classifier for our prediction and the best Hyperparameters obtained are as below.

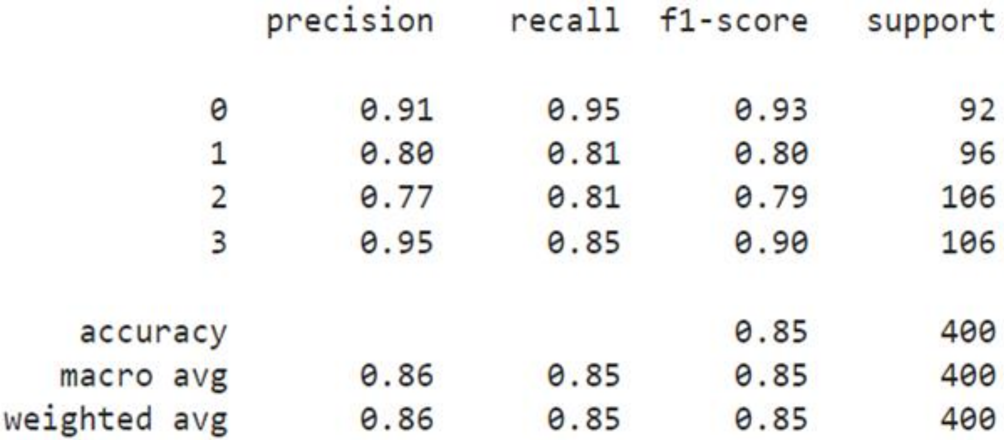
- criterion = ‘entropy’
- max_depth = 8
- max_features = ‘log2’
- N_estimators = 200



Random Forest tuned Confusion Matrix



	0	1	2	3
0	87	5	0	0
1	9	78	9	0
2	0	15	86	5
3	0	0	16	90

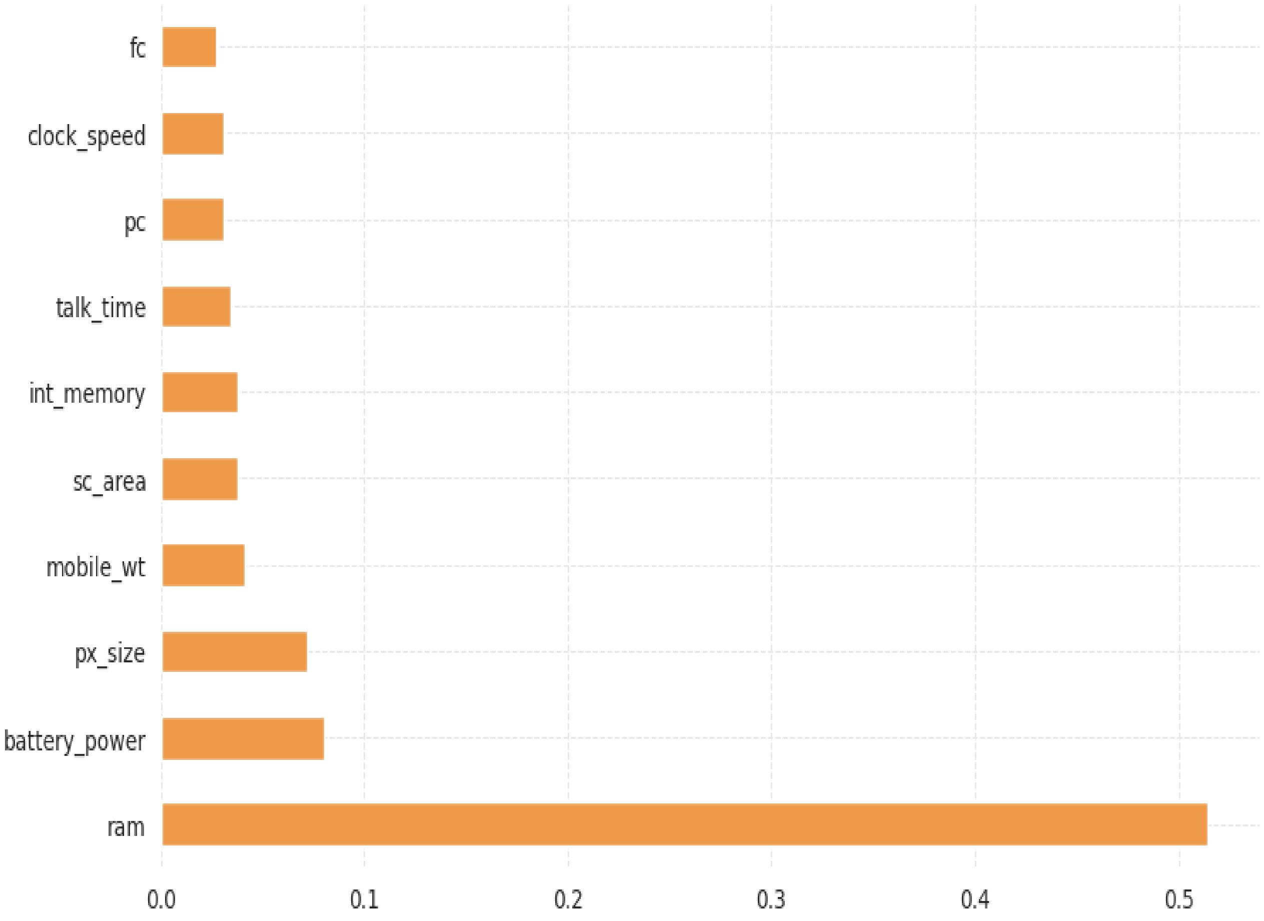


```
              precision    recall  f1-score   support

           0       0.91      0.95      0.93        92
           1       0.80      0.81      0.80        96
           2       0.77      0.81      0.79       106
           3       0.95      0.85      0.90       106

    accuracy                           0.85       400
   macro avg       0.86      0.85      0.85       400
weighted avg       0.86      0.85      0.85       400
```

Feature Importance

- The feature importance (variable importance) describes which features are relevant.

- It can help with better understanding of the solved problem and sometimes lead to model improvements by employing the feature selection.



Conclusion

- We started this project with the intention to identify the useful variables/factors which drives phone price and to build a predictive model which can give phone price range depending on its feature.
- For this, we performed exploratory data analysis on our data after cleaning and making it easy to analyze. This analysis helped us to identify variables which directly impacts Mobile Phone Prices.
- We found that 'RAM' of a phone linearly affects the phone prices. Other variables like battery life and px_size also shows linearity (up to some extent only) with the phone prices.
- We found that most very high-cost phones have low handset weight, high Internal Memory, high pixel and screen size.
- Our next job was to make a price range predictive model. For this, we processed our data, split it for training and testing and finally applied four different models.
- Both Random Forest and Decision tree classifiers gave some good accuracy scores, however on cross validation scores Random Forest performed much better.
- This gave us confidence to perform hyperparameter tuning on our Random Forest model.
- Finally, we optimized our model, which increased the accuracy score of our model to 0.86,which is pretty decent.

THANK YOU