Capstone Project - 3

Mobile Price Range Prediction

(Supervised Machine Learning Classification)

Submitted By

Bhavesh Amol Amre

Points to be Discuss :

- Introduction and Problem statement
- Workflow
- Data Description
- Data Cleaning
- Exploratory Data Analysis
- Feature Engineering
- Model formulation
- Conclusion

Problem Statement :

- Mobile Phone has become necessity for everyone in today's world. People's want more features, best specification and latest updates in their Mobile Phone with cheapest price.

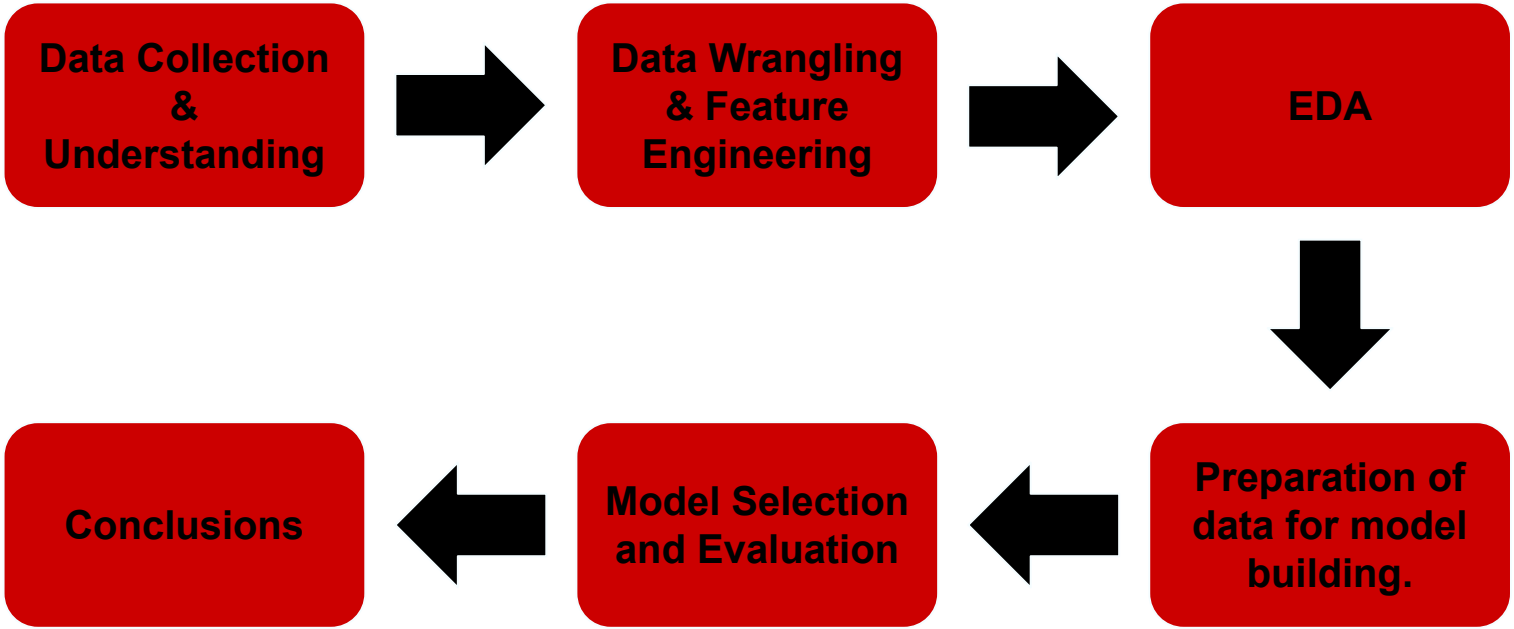
- Mobile Phones comes with all sorts of prices, features, specification and many more updates. So, price estimation with the given features, specification and other things in a mobile phone is a important part for the success of the product. The product must be developed according to the consumer market so, that consumer will find it appropriate to buy the product and it will lead to success.

- In the competitive mobile phone market companies want to understand sales data of mobile phones and factors which drive the prices. The objective is to find out some relation between features of a mobile phone(eg:- 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.

- The main objective of this project is to build a model which will classify the price range of mobile phone bases on the different feature of mobile for effective Selling strategy in consumer market.

Work Flow :

- So we will divide our work flow into following steps.



Data Collection and Understanding :

- In this dataset we have total 2000 observations and 21 features including target variable.

Data Description :

- **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 megapixels
- ***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
- ***Pc*** - Primary Camera megapixels
- ***Px_height*** - Pixel Resolution Height
- ***Px_width*** - Pixel Resolution Width
- ***Ram*** - Random Access Memory in MegaBytes
- ***Sc_h*** - Screen Height of mobile in cm

Data Collection and Understanding :

- ***Sc_w*** - Screen Width of mobile in cm
- ***Talk_time*** - longest time that a single battery charge will last when you are
- ***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).

Data Wrangling and Feature Engineering :

Handling the miss match values in the dataset :

- We can see that sc_width and px_height has minimum value 0. which is not possible in any mobile. We need to handle this mismatch.
- Missing values are im-muted using the K-Nearest Neighbour approach where Euclidean distance is used to find the nearest neighbour.

	count	mean	std	min	25%	50%	75%	max
px_height	2000.0	645.10800	443.780811	0.0	282.75	564.0	947.25	1960.0
sc_w	2000.0	5.76700	4.356398	0.0	2.00	5.0	9.00	18.0

```
# Checking How many observations having screen width value as 0.
print(mobile_data[mobile_data['sc_w']==0].shape[0])
```

```
180
```

```
mobile_data[mobile_data['px_height']==0].shape[0]
```

```
2
```

```
# As there are only 2 observations having px_height=0. so we will drop it.
mobile_data=mobile_data[mobile_data['px_height']!=0]
```

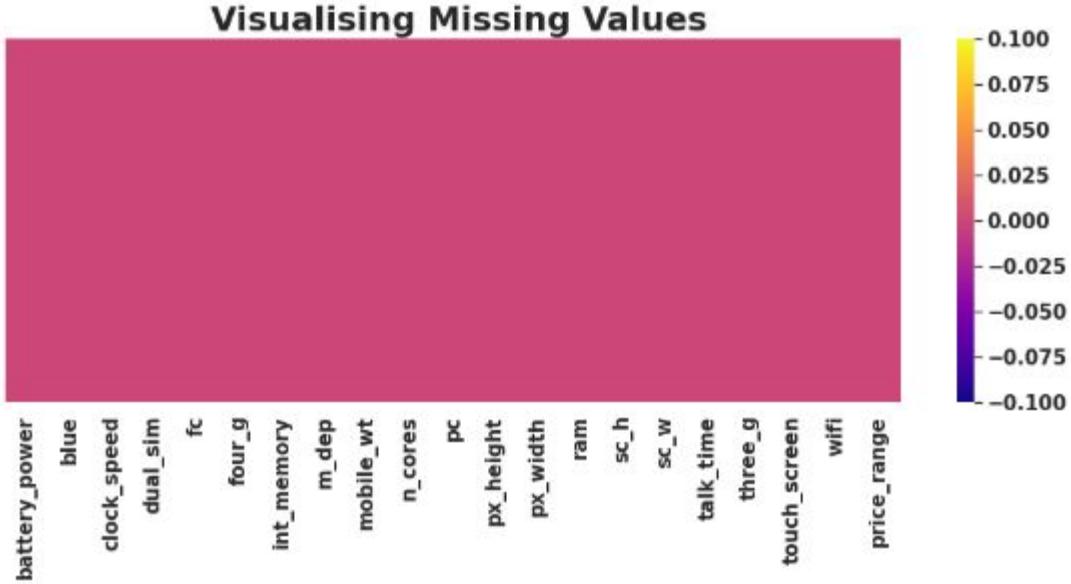
```
# Replacing 0 with NAN so that we can implement KNN Imputer.
mobile_data['sc_w']=mobile_data['sc_w'].replace(0,np.nan)
```

```
# Checking How many observations having sc_w value as 0.
mobile_data[mobile_data['sc_w']==0].shape[0]
```

```
0
```

Data Wrangling and Feature Engineering :

- We had zero null values in our dataset.
- Zero Duplicate entries found.



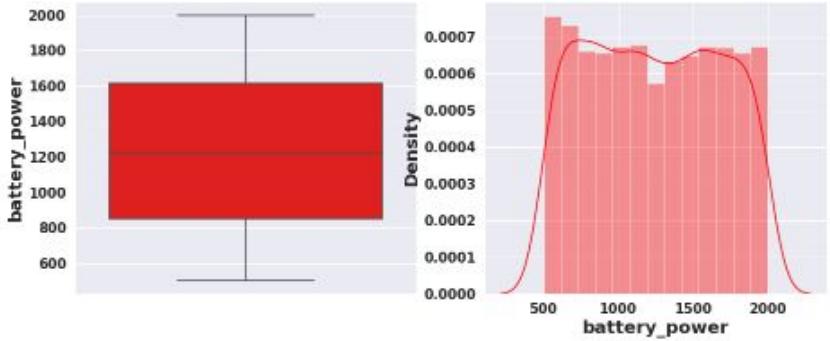

```
# Checking Duplicate values in data set.
print(f' We have {mobile_data.duplicated().sum()} duplicate values in dataset.')
```

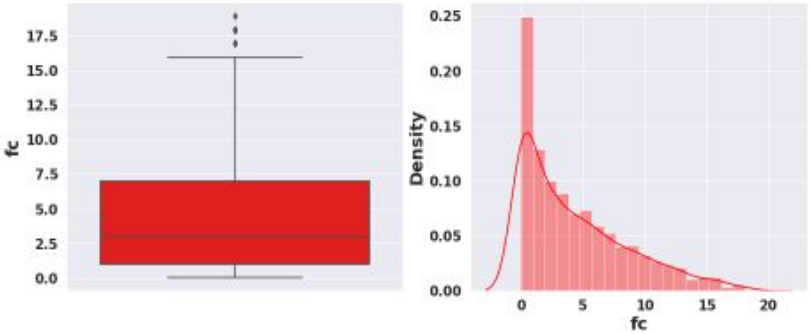
```
 We have 0 duplicate values in dataset.
```

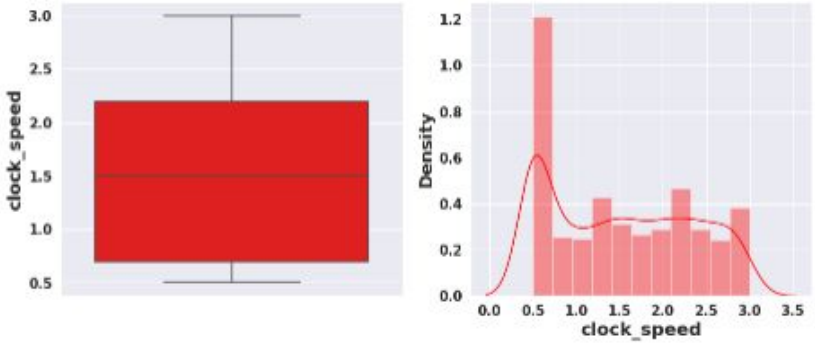
Data Wrangling and Feature Engineering :

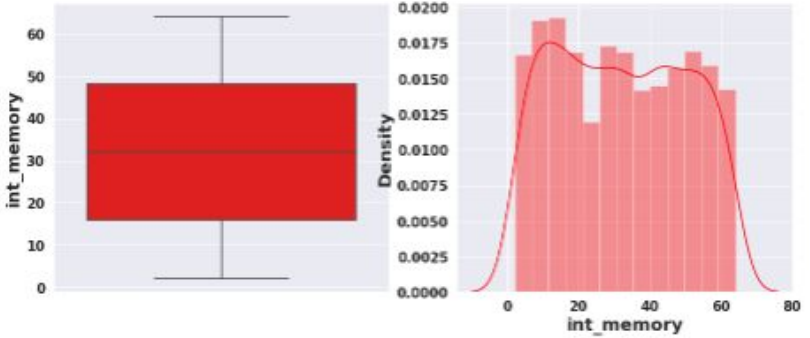
Check the distribution of numerical columns and Outliers :

- Data is well distributed.
- fc and px_height has some outliers.

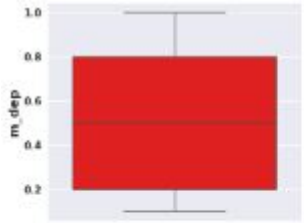


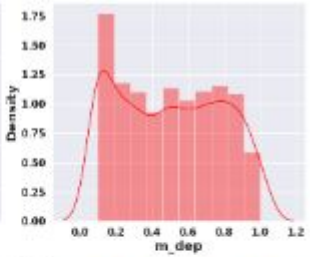


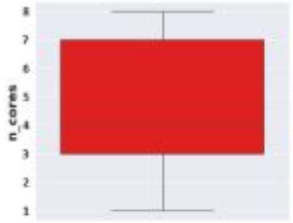


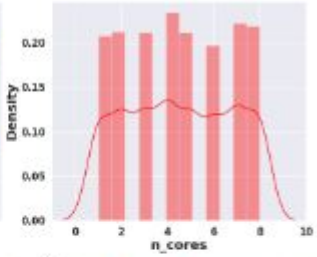


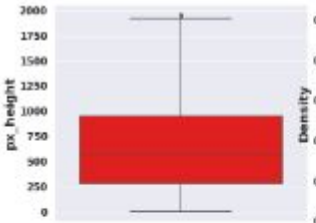
❖ Data Wrangling and Feature Engineering :

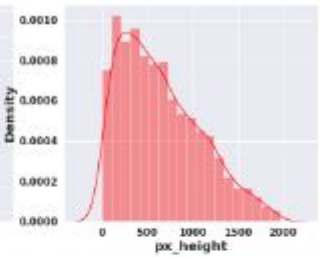


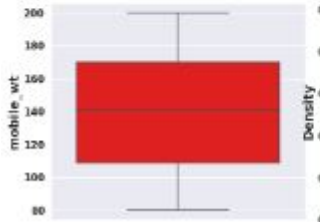


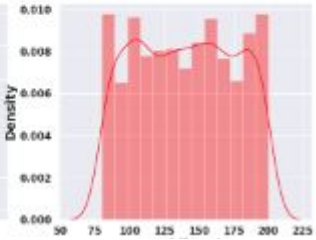


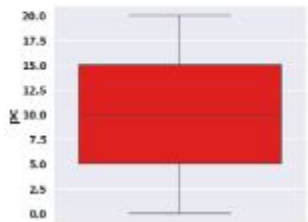


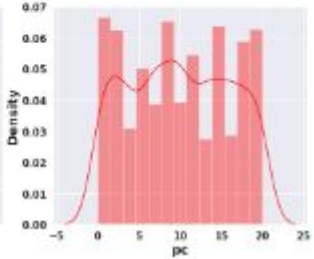


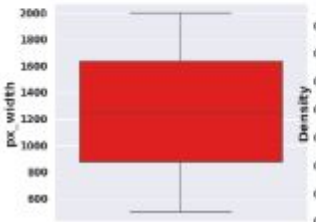


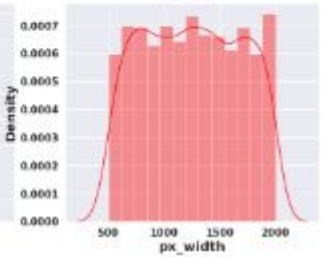


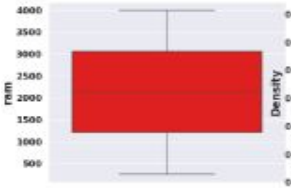


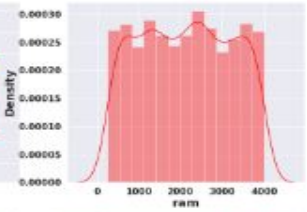


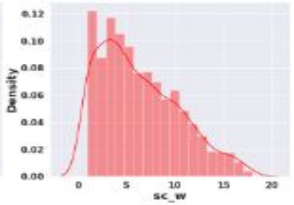


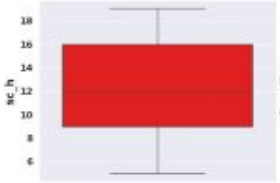


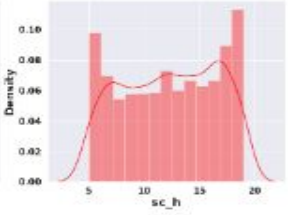


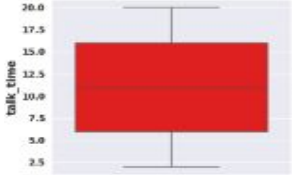


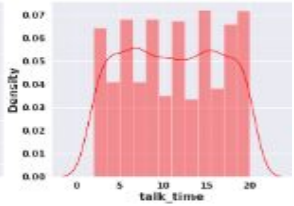






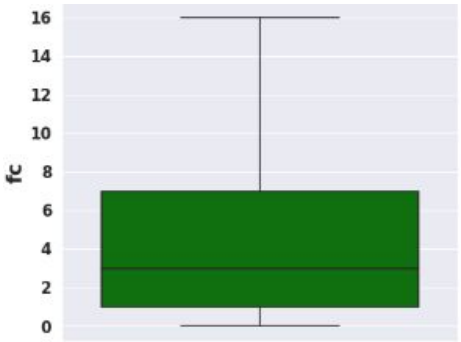


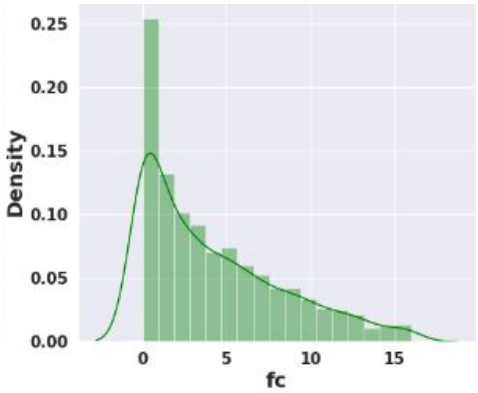


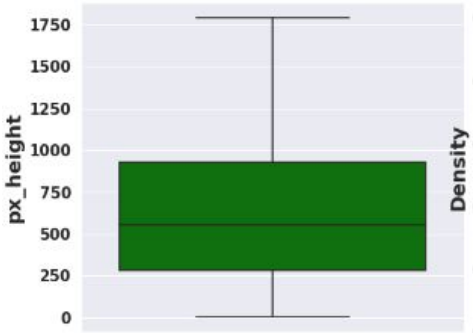


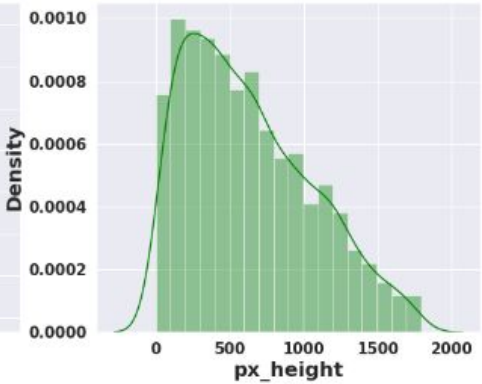
Data Wrangling and Feature Engineering :

- After removal of Outlier



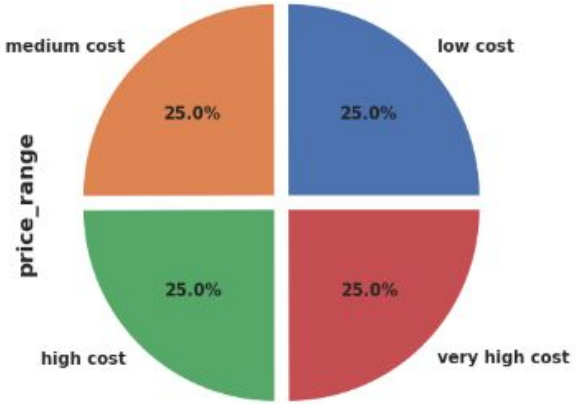


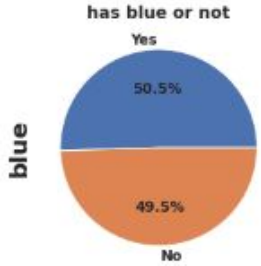


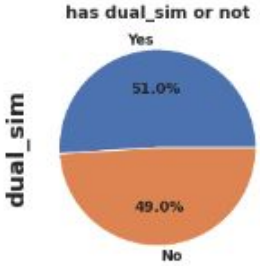


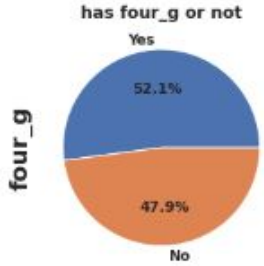
EDA(Exploratory Data Analysis) :

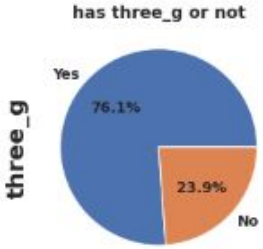
Univariate Analysis

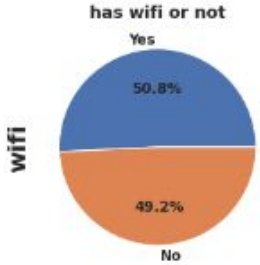


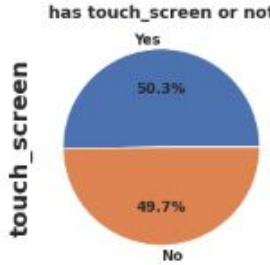










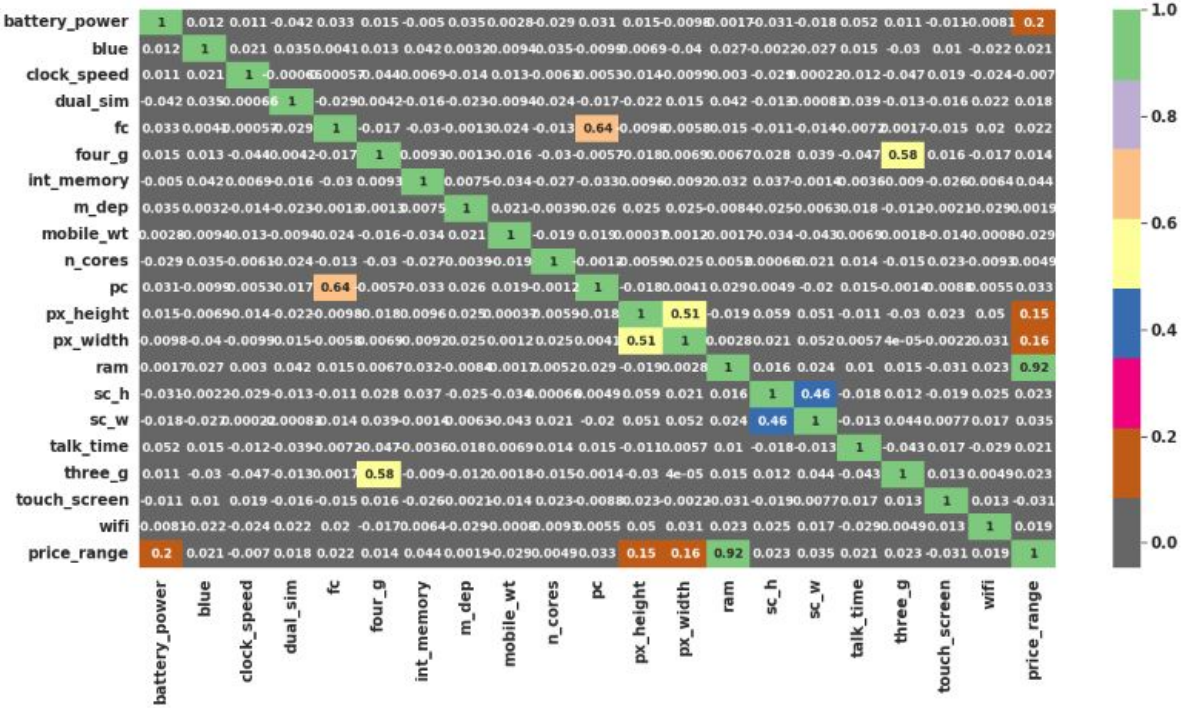


- Our target variable has equal number of observation in each category. Target variable is equally distributed.
- Percentage distribution of mobile phone having bluetooth, dual sim, 4G, wifi and touch screen are almost about 50 %.
- While very few (23.8 %) mobile phones do not have 3G.

EDA(Exploratory Data Analysis) :

Bi-Variate and Multivariate Analysis :

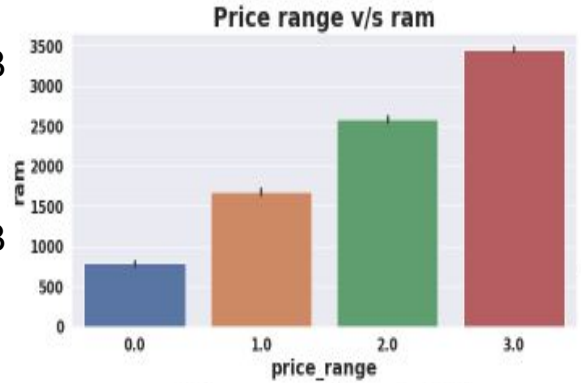
- Ram has Positive correlation with the price range and we know that we know that mobile with high ram are costly. Thus ram increase price range also increase.
- Battery power also has positive correlation with the price range. Generally mobile having high prices comes with good battery power.
- px_height and px_width are positively correlated.Generally high price high price mobiles have high resolution.
- 4G and 3G are highly positively correlated. now days, most of mobile have both features.
- Primary camera (pc) and Front camera (fc)Are positively correlated.
- Sc_h and sc_w are also positively correlated.

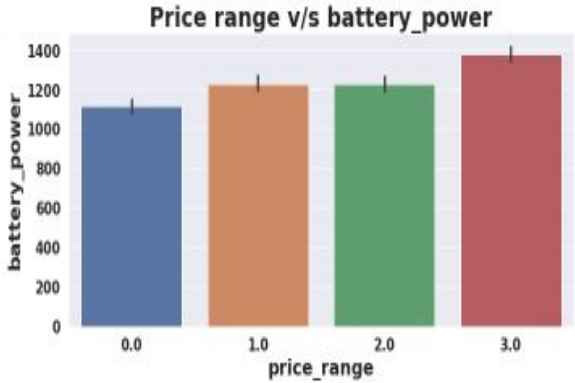


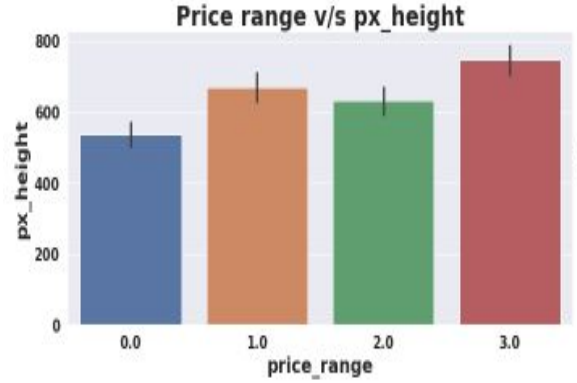
EDA(Exploratory Data Analysis) :

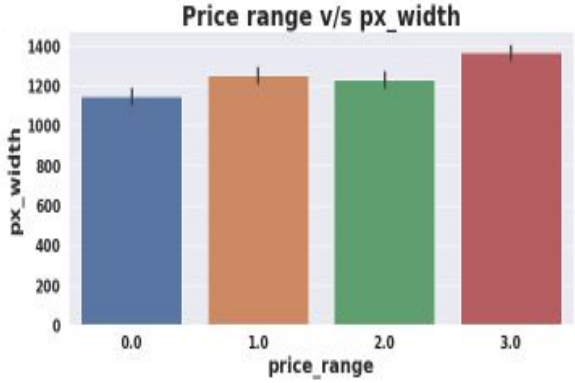
Bi-Variate and Multivariate Analysis :

- Mobiles having Ram less than 1000 MB falls under low cost category.
- Mobile having Ram more than 3000 MB falls under very high cost category.
- Mobile with battery power more than 1300 mAh has very high cost while battery power between 1200 and 1300 mAh falls under medium and high cost category.
- Mobile with more than 700 pixel height and more than 1300 width has very high cost.









EDA(Exploratory Data Analysis) :

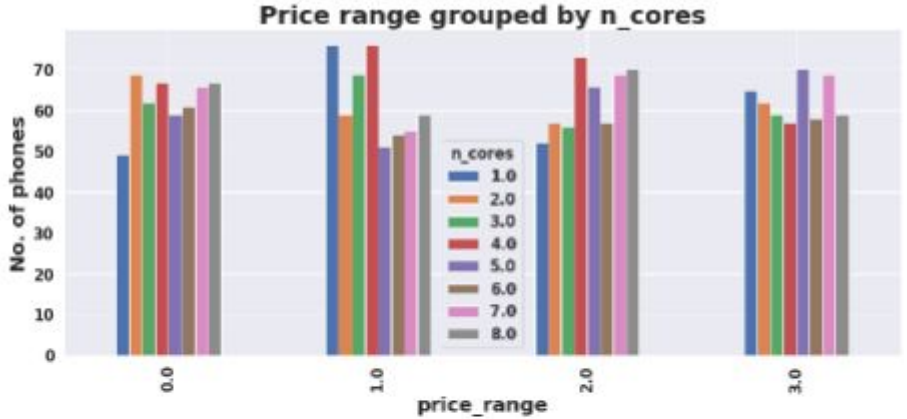
Bi-Variate and Multivariate Analysis :

- Each price range category has equal number of mobile phones having both supporting and non-supporting specification.

EDA(Exploratory Data Analysis) :

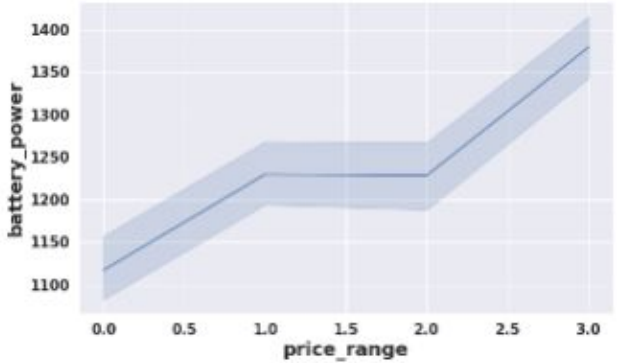
➢ **Bi-Variate and Multivariate Analysis :**

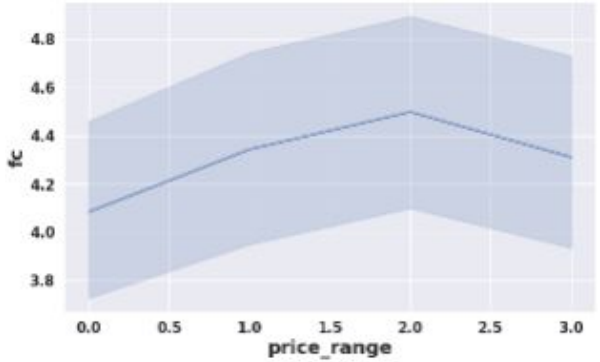
- ❑ Their are very few mobiles in price range 0 to 1 lesser number of cores.
- ❑ Most of the mobile in price range 2 and 3 are with high no. of cores.
- ❑ Number of phones with less thickness is high while with high thickness is low.

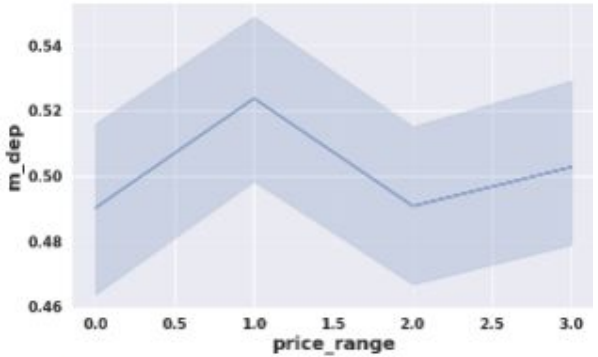


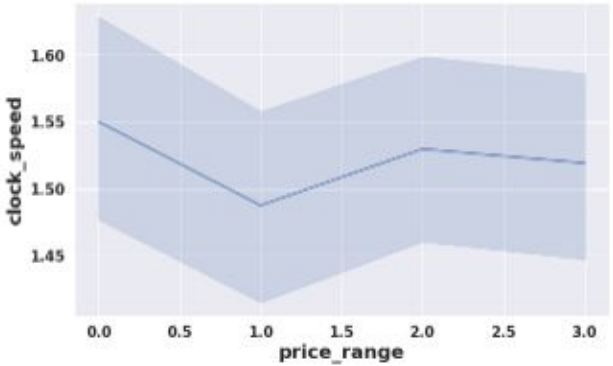
EDA(Exploratory Data Analysis) :

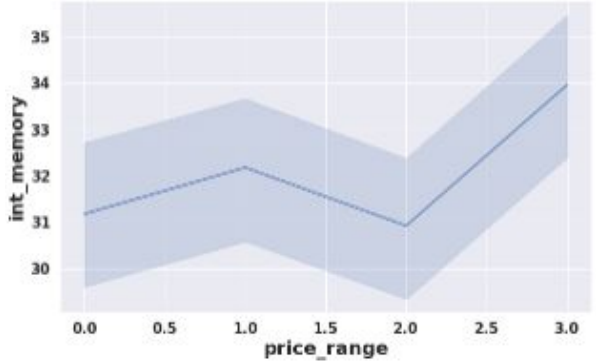
- **Bi-Variate and Multivariate Analysis :** Different trend of Price range V/S Other Features.

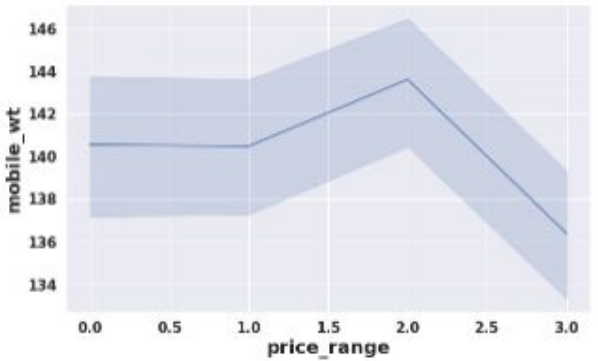








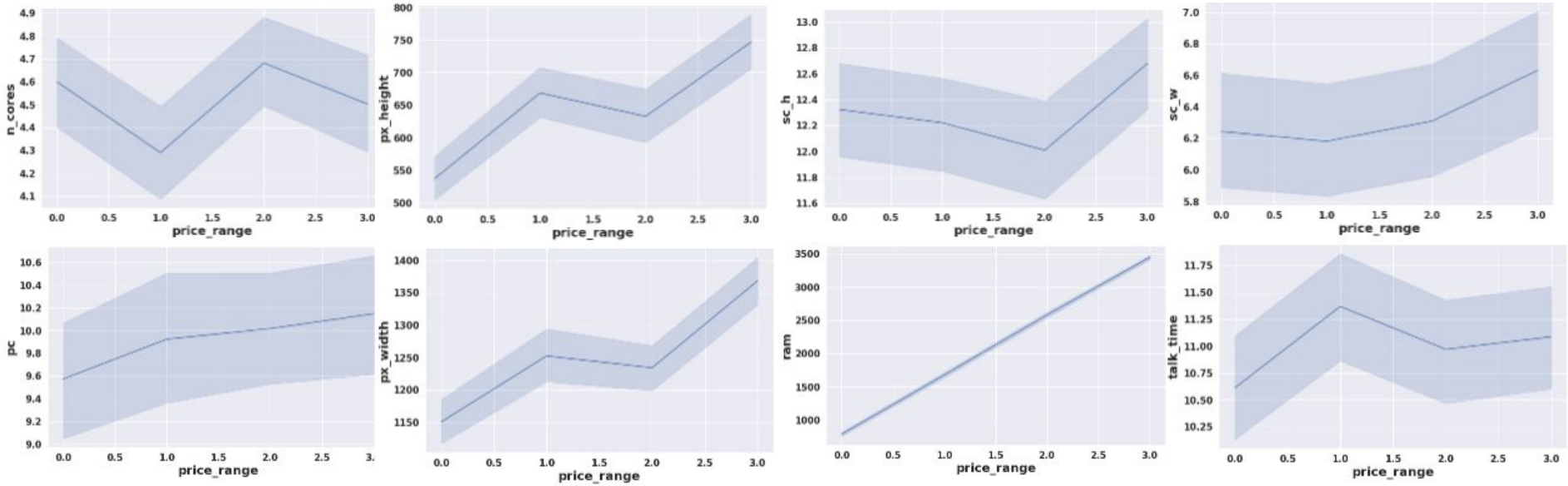




- **For class 1 and class2 battery power range is almost similar. As battery power increases price also increases which is quite obvious..**

EDA(Exploratory Data Analysis) :

➢ **Bi-Variate and Multivariate Analysis :** Different trend of Price range V/S Other Features.

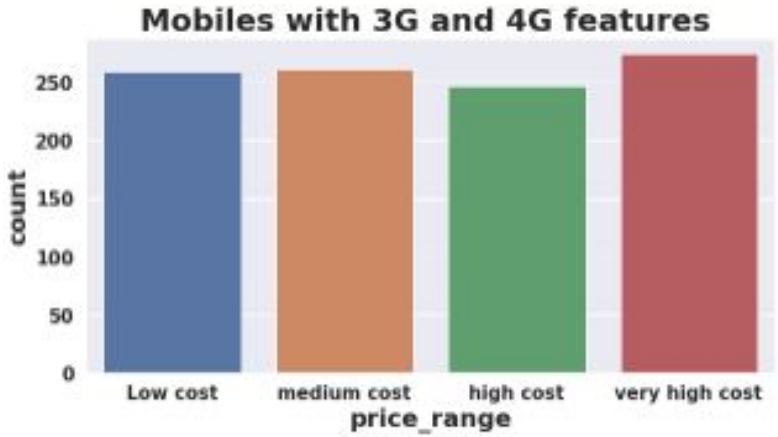


- ***Mobiles in very high price range(Class 3) has less weight compared to other classes.That means as weight of mobiles decrease price increases.***
- ***Mobiles having max screen height and width falls in very high price category. We can see in linechart of sc_width and sc_height from class 2 screen width and height starts increasing with price. Similar case is with px_height and px_width. As resolution of screen increases the price also increases***
- **RAM has clear relationship with price range we saw that in correlation matrix also.**

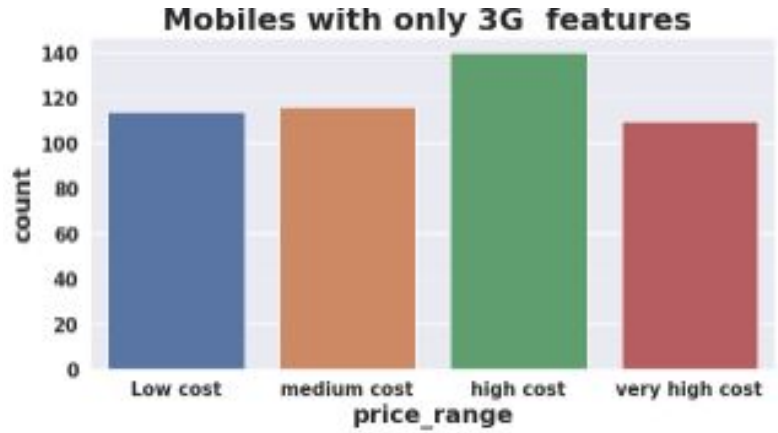
EDA(Exploratory Data Analysis) :

➢ **Bi-Variate and Multivariate Analysis :**

- Mobile with 3G and 4G is highest in very high cost and lowest in high cost.



- Mobile with 3G only has highest count in high cost.



Preparation of data for model building :

- We are separating the independent variable and dependent variable for model building.
- We have selected the top 12 important features by applying Chi Square Test.
- Now we will go ahead with the top 12 independent variable for model building.

```
# 12 features with highest chi squared statistic
print(featureScores.nlargest(12,'Score'))
```

	Specs	Score
13	ram	914971.362532
11	px_height	15629.508974
0	battery_power	13276.863289
12	px_width	9172.791036
8	mobile_wt	87.027556
6	int_memory	78.357703
14	sc_h	11.815783
16	talk_time	11.460771
15	sc_w	11.079469
4	fc	10.867193
9	n_cores	8.395937
10	pc	7.639203

```
# 12 features with highest chi squared statistic are selected as independent variables.
X=mobile_data[['ram','px_height','battery_power','px_width','mobile_wt','int_memory','sc_h','talk_time','sc_w','fc','n_cores','pc']]

# dependent varaible
y=mobile_data['price_range']
```

```
# Check dataframe
featureScores
```

	Specs	Score
0	battery_power	13276.863289
1	blue	0.625168
2	clock_speed	0.830213
3	dual_sim	0.736762
4	fc	10.867193
5	four_g	1.319605
6	int_memory	78.357703
7	m_dep	0.797351
8	mobile_wt	87.027556
9	n_cores	8.395937
10	pc	7.639203
11	px_height	15629.508974
12	px_width	9172.791036
13	ram	914971.362532
14	sc_h	11.815783
15	sc_w	11.079469
16	talk_time	11.460771
17	three_g	0.377184
18	touch_screen	2.344057

Model Selection and Evaluation:

Before building the models we performed the train test split. We kept 25% of data for the test and the remaining 75% of the data for training the model.

We compared 6 algorithms and evaluated them based on the overall accuracy score recall of the individual classes.

- **Accuracy is the ratio of total number correct predictions and the total number of predictions.**
- **The recall is the measure of our model correctly identifying True Positives.**

1) Decision Tree
2) Random Forest classifier
3) Gradient Boosting Classifier
4) K-nearest Neighbor Classifier
5) XGBoost Classifier
6) Support Vector Machine(SVM)

Model Selection and Evaluation:

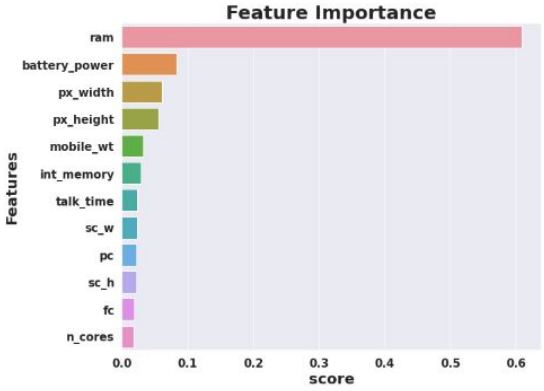
Algorithm	Training Set		Test Set	
	Accuracy	Recall(%)	Accuracy	Recall(%)
Decision Tree	100	100	84	83.91
Decision Tree(HyperParameter Tuning)	97.62	98	85.13	85
Random Forest classifier	100	100	88.95	89
Random Forest classifier(HyperParameter Tuning)	100	100	89.81	90
Gradient Boosting Classifier	100	100	90.02	90
Gradient Boosting Classifier(HyperParameter Tuning)	100	100	90.42	90
K-nearest Neighbor Classifier	75.86	76	53.47	59
K-nearest Neighbor Classifier(HyperParameter Tuning)	76.6	77	70.26	70
XGBoost Classifier	98.98	99	90.22	90
XGBoost Classifier(HyperParameter Tuning)	100	100	92.486	92
Support Vector Machine	98.57	99	89.81	90
Support Vector Machine(HyperParameter Tuning)	98.3	98	97.96	98

- Best model is Support Vector Machine after HyperParameter Tuning which has Accuracy of 98.3% on Training Set and 97.96% on Test Set.
- The Second best model is XGBoost Classifier after HyperParameter Tuning in terms of both Accuracy and Recall.
- K-nearest Neighbor Classifier is the Worst Model with Accuracy of 75.86% on Training Set and 53.47% on Test Set on Recall it gavea Score of 76% on Training Set and 59% on Test Set.

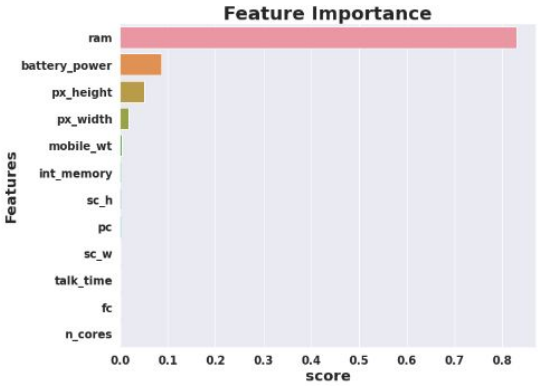
Feature Importances :

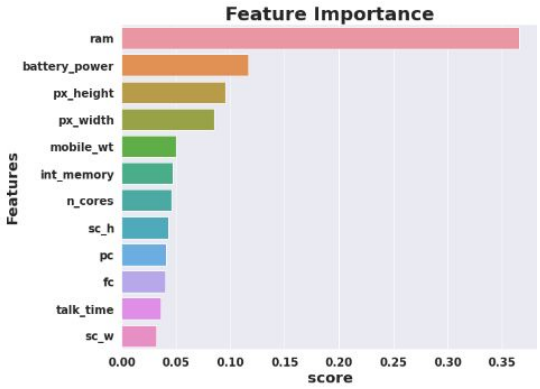
Decision Tree

Random Forest

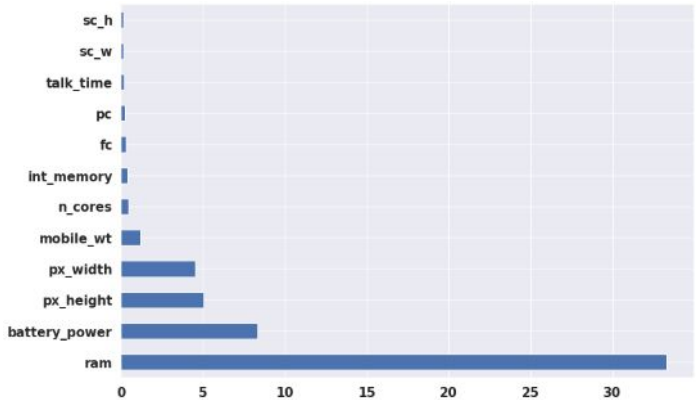


Gradient Boost





XG boost

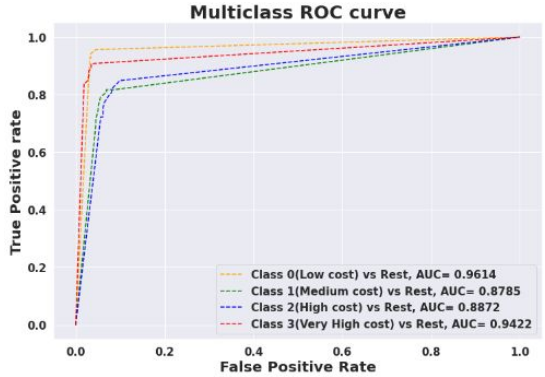


SVM

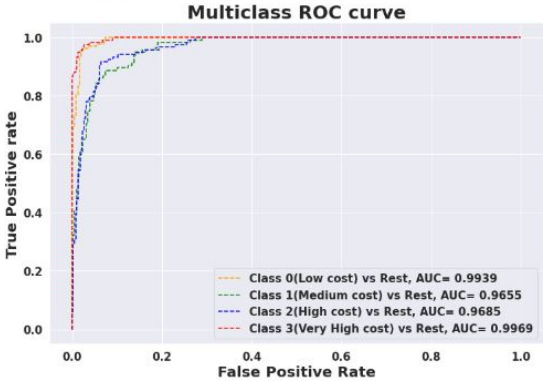
- **RAM,Battery power,px_height and px_weight are the important features in predicting the price range.**

AUC ROC Curves :

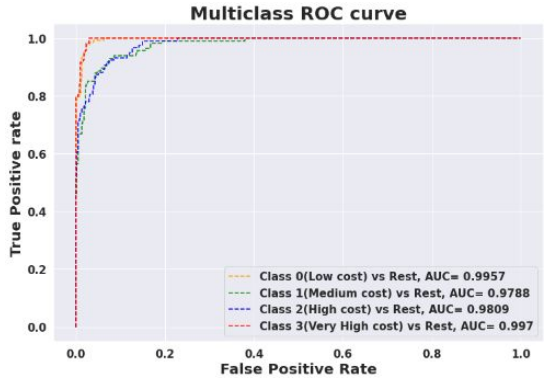
Decision Tree

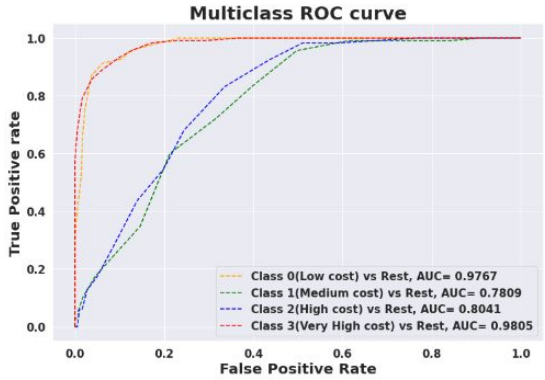


Random Forest

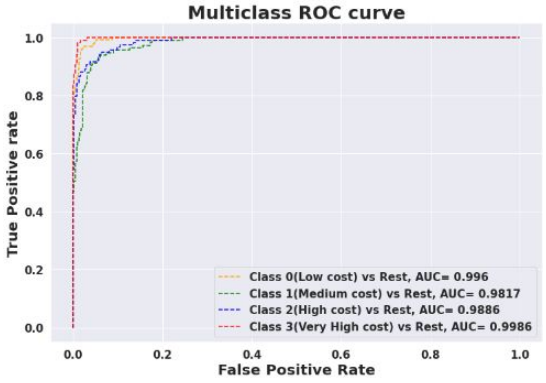


Gradient Boost

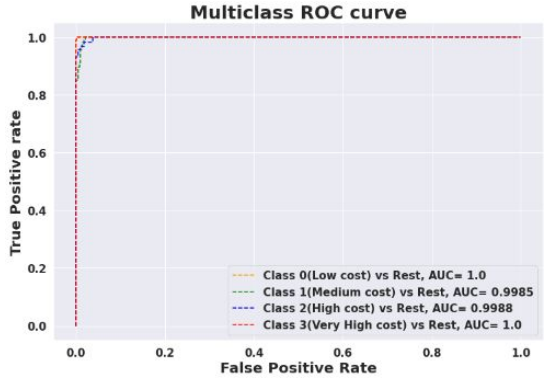




KNN



XG boost



SVM

Conclusion :

- We Started with Data understanding, data wrangling, basic EDA where we found the relationships, trends between price range and other independent variables.
- We selected the best features for predictive modeling by using the K best feature selection method using Chi square statistics.
- Implemented various classification algorithms, out of which the SVM(Support vector machine) algorithm gave the best performance after hyper-parameter tuning with 98.3% train accuracy and 97 % test accuracy.
- XGboost is the second best model which gave good performance after hyper-parameter tuning with 100% train accuracy and 92.25% test accuracy score.
- KNN gave the very worst model performance.
- We checked for the feature importances of each model. RAM, Battery Power, Px_height and px_width contributed the most while predicting the price range.

THANK YOU