Mobile Price Range Prediction

Smitesh M Jadhav

mr.smiteshjadhav@gmail.com

Problem statement

- In the competitive mobile phone market companies wantto 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 theactual price but a price range indicating how high the price is.

Points to discuss

- Data description and summary
- Exploratory data analysis
- Heat map
- Machine learning algorithms
    1. Logistic regression
    2. Decision tree
    3. Random forest classifier
    4. Xgboost classifier
- conclusion

Data description

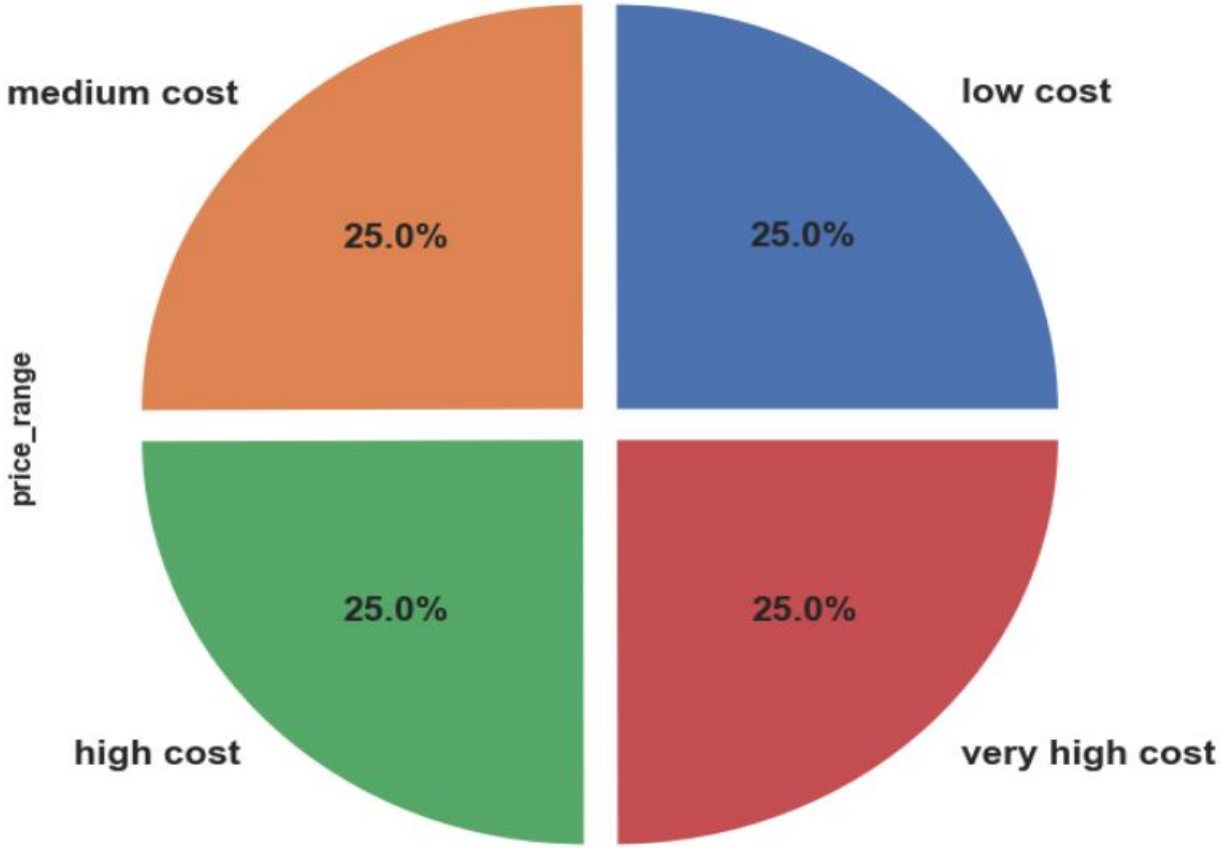
The data contains information regarding mobile phone features, specifications etc and their price range. The various features and information can be used to predict the price range of a mobile phone.

- 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

Data description(cont,.)

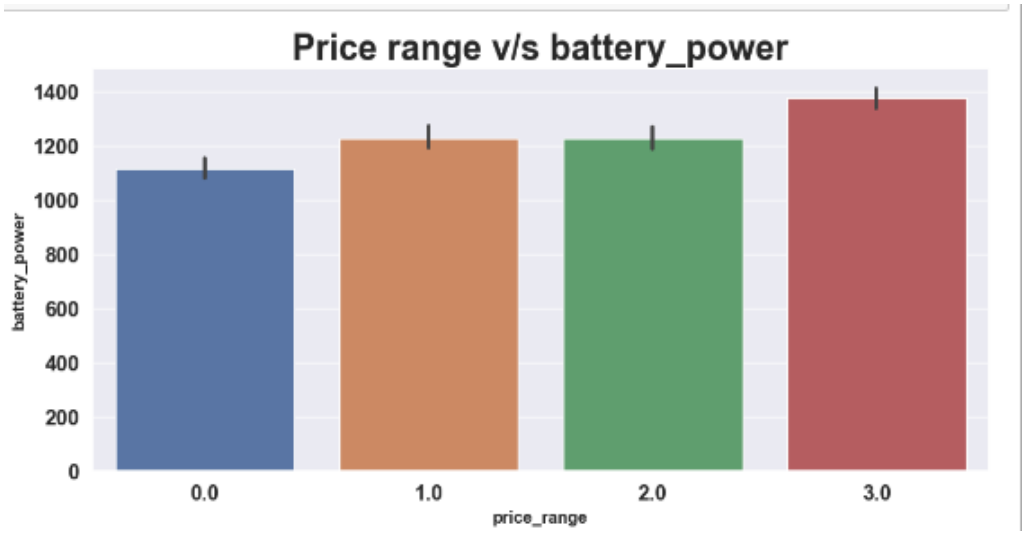
- N_cores - Number of cores of processor
- Pc - Primary Camera mega pixels
- Px_height - Pixel Resolution Height
- Px_width - Pixel Resolution Width
- Ram - Random Access Memory in Mega Bytes
- Sc_h - Screen Height of mobile in cm
- Sc_w - Screen Width of mobile in cm
- Talk_time - longest time that a single battery charge will last when you are
- 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).

Price



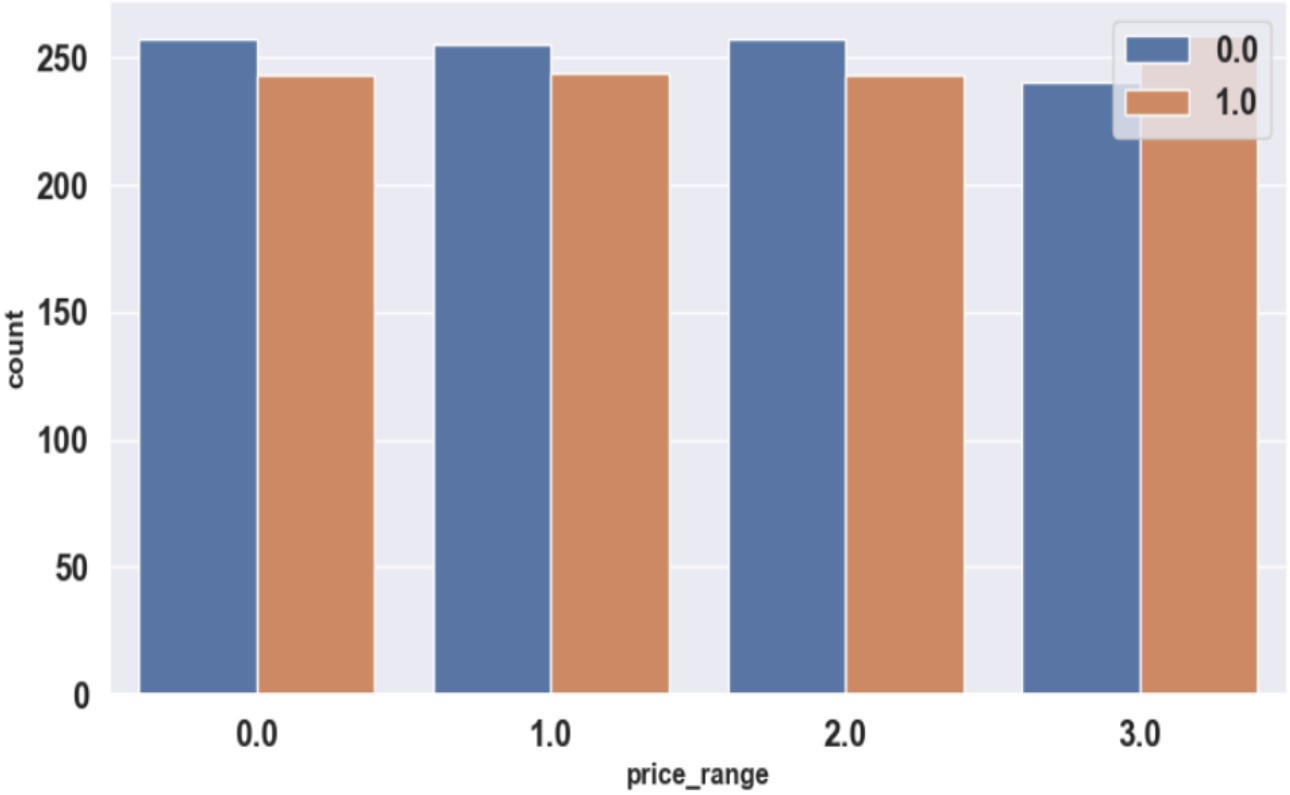
- There are mobile phones in 4 price ranges. the number of elements is almost similar

Battery



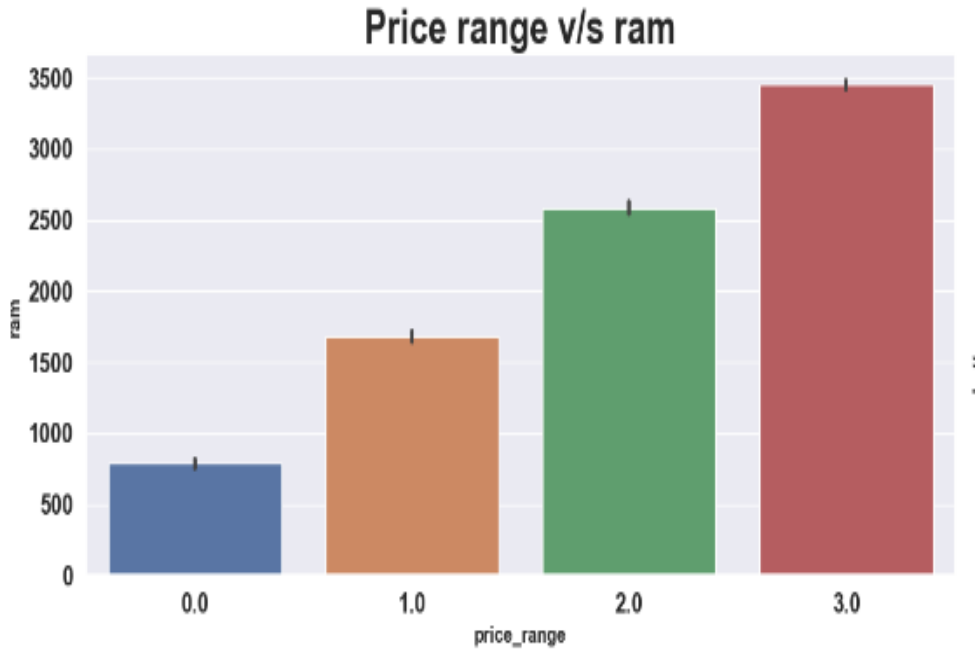
- This plot shows how the battery mAh increases as the price range increases.

Bluetooth



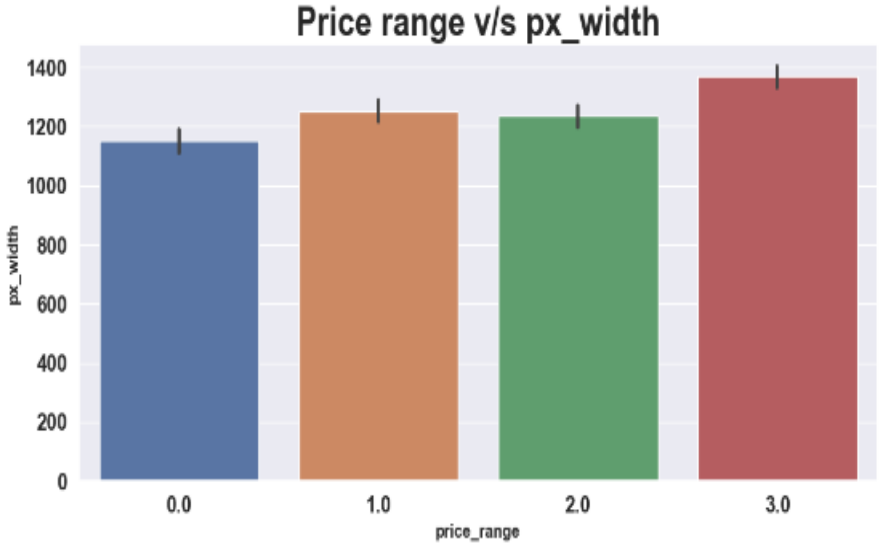
Majority of phones of price range from 0 to 2 don't have Bluetooth on other hand price range of 3 have Bluetooth service.

RAM



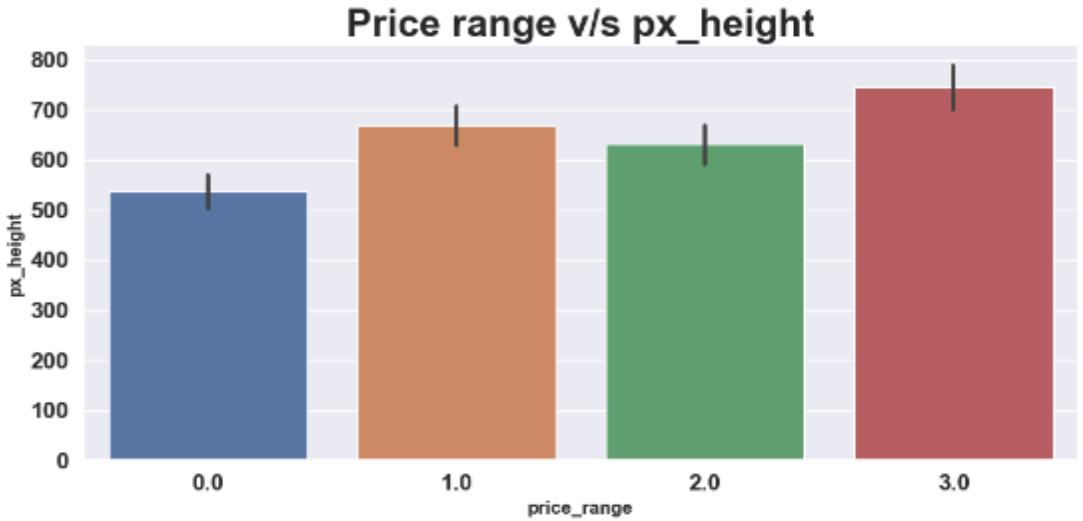
Ram has continuous increase with price range while moving from Low cost to Very high cost

Px_width



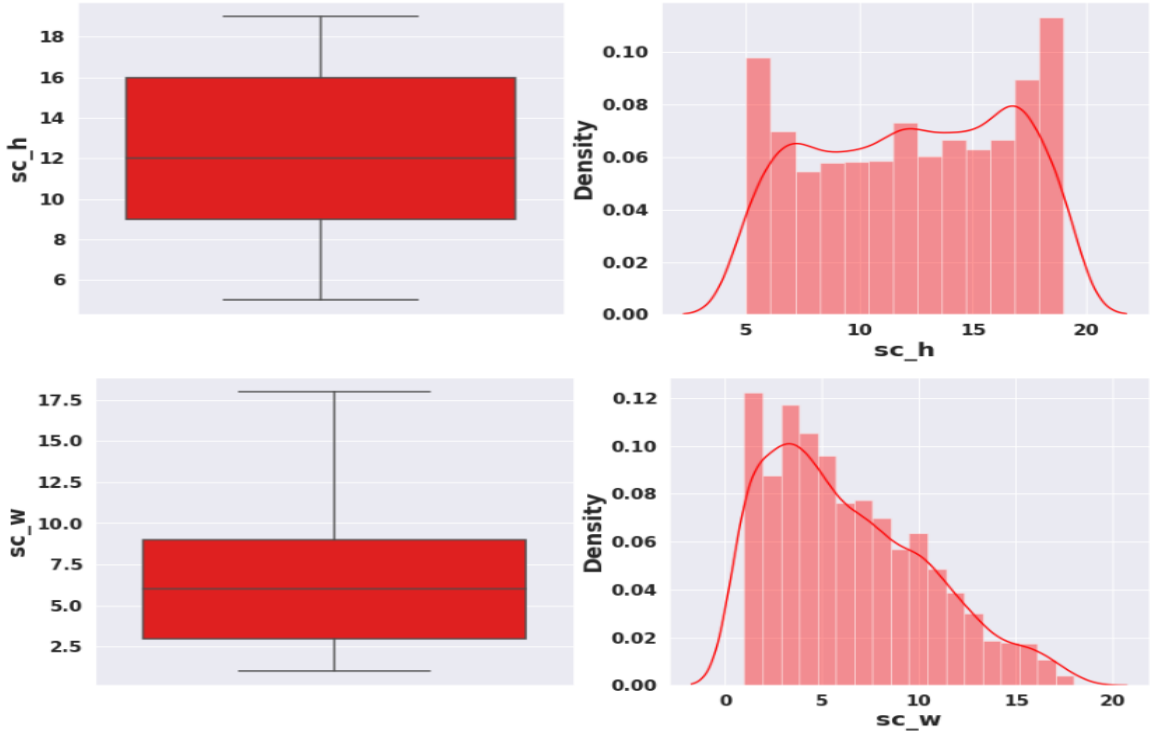
There is not a continuous increase in pixel width as we move from Low cost to Very high cost. Mobiles with 'Medium cost' and 'High cost' has almost equal pixel width. so we can say that it would be a driving factor in deciding price_range.

Px_height



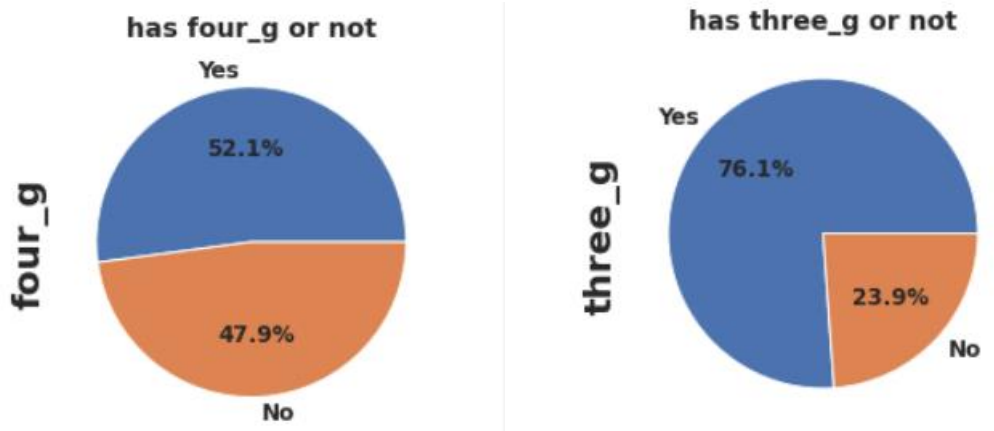
Pixel height is almost similar as we move from Low cost to Very high cost.little variation in pixel_height

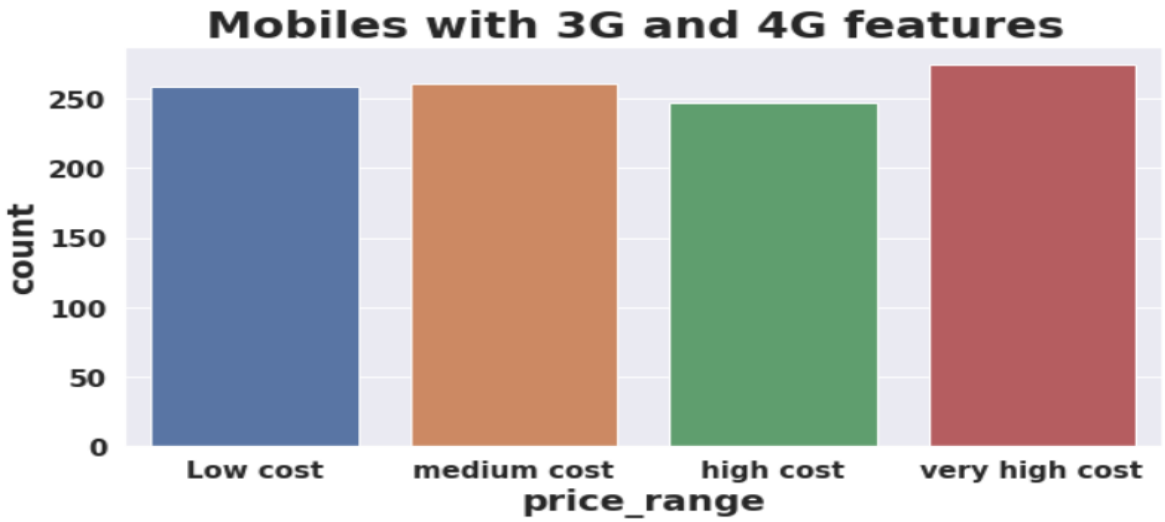
Screen_size



The sc_height and sc_width shows little variation along the target variables. This can be helpful in predicting the target categories.

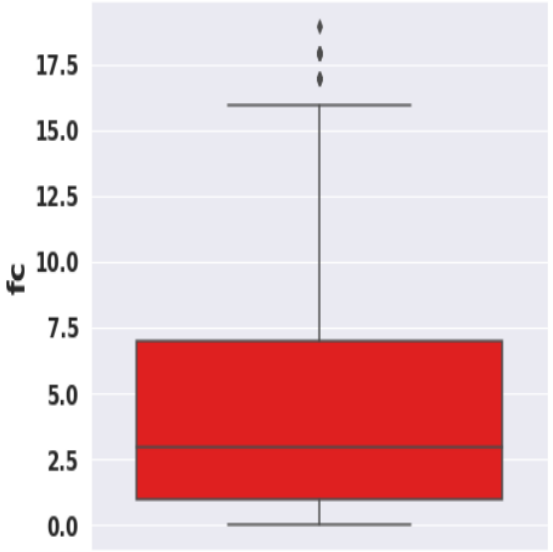
4g and 3g





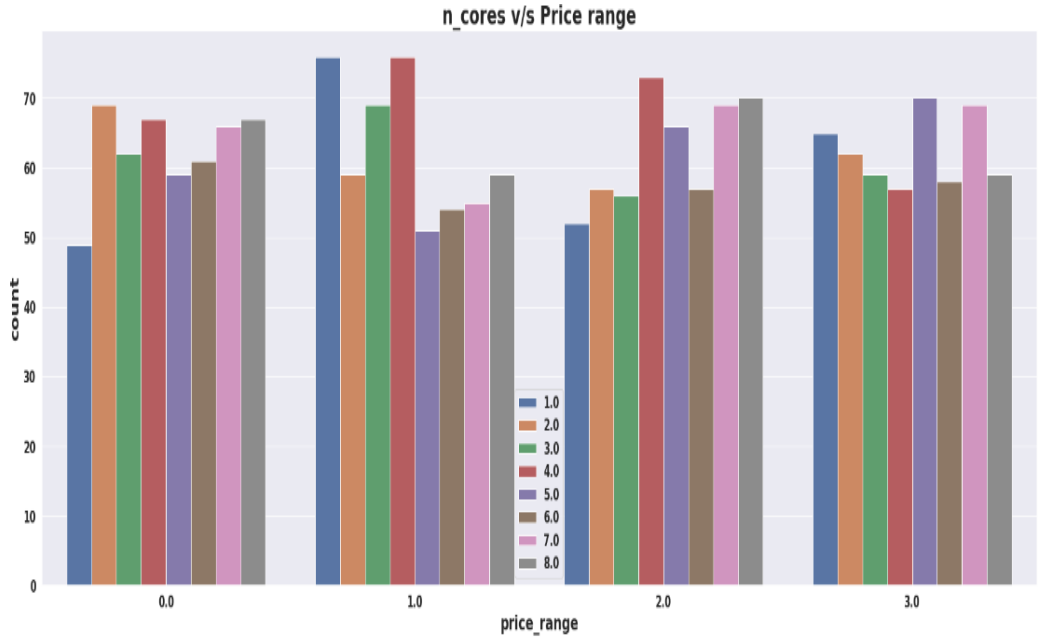
50% of the phones support 4_g and 76% of phones support 3_g,feature 'three_g' play an important feature in prediction

FC (front camera megapixels)



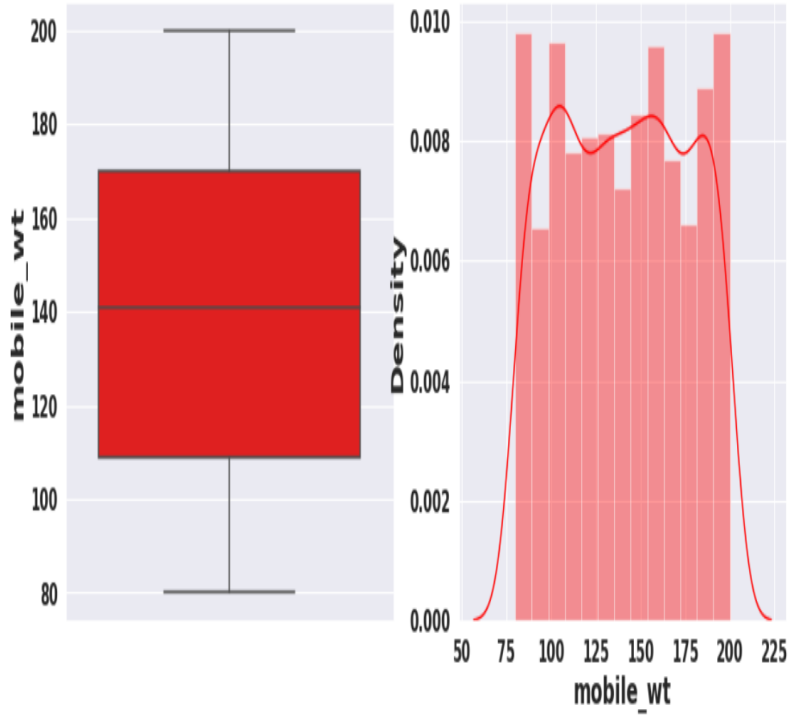
- This features distribution is almost similar along all the price ranges variable, it may not be helpful in making predictions

PC (Primary camera Megapixels)



- Primary camera megapixels are showing a little variation along the target categories, which is a good sign for prediction.

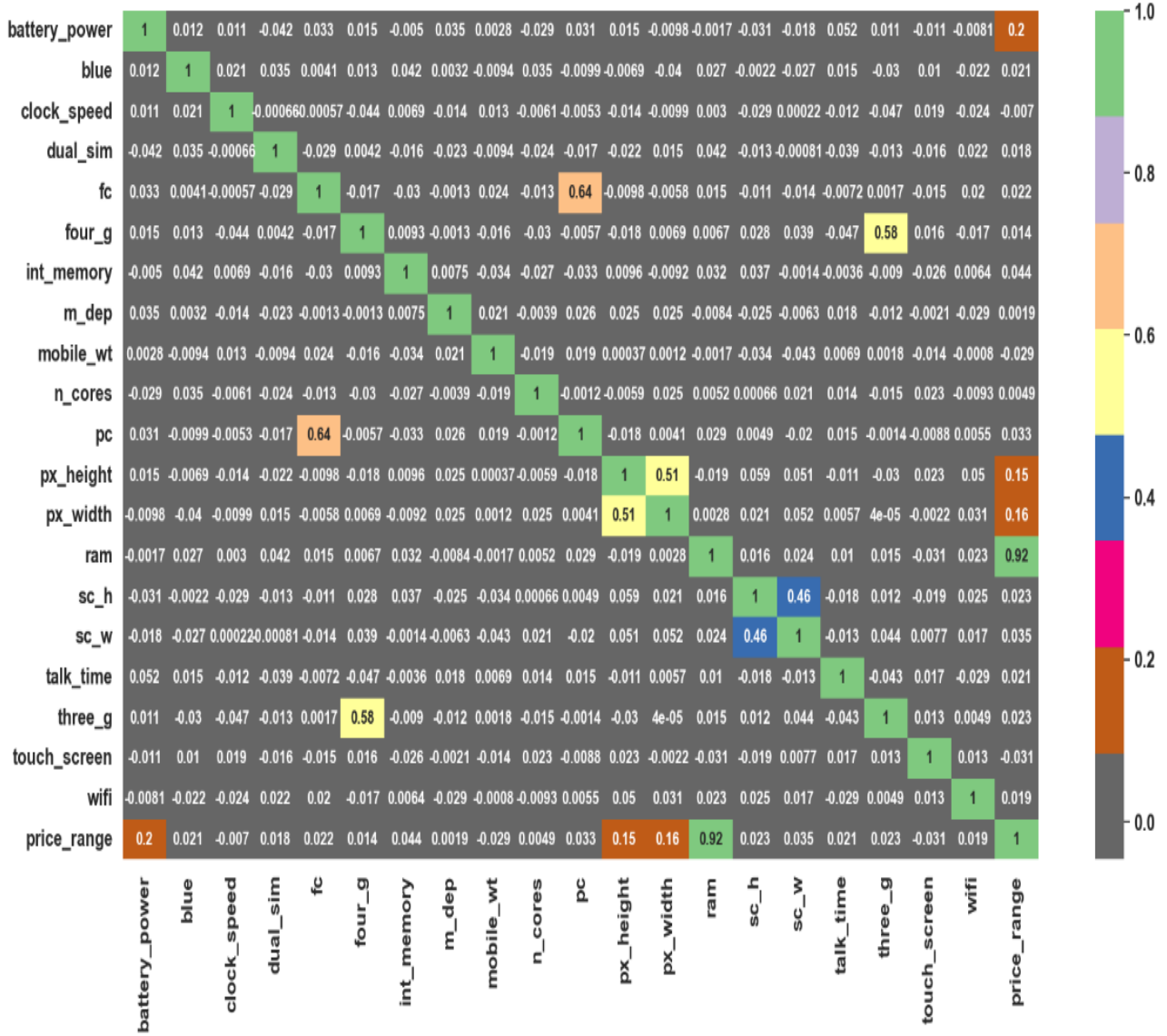
Mobile weight



- costly phones are lighter

Heat map

- RAM and price_range shows high correlation which is a good sign, it signifies that RAM will play major deciding factor in estimating the price range.
- There is some collinearity in feature pairs ('pc', 'fc') and ('px_width', 'px_height'). Both correlations are justified since there are good chances that if front camera of a phone is good, the back camera would also be good.
- Also, if px_height increases, pixel width also increases, that means the overall pixels in the screen. We can replace these two features with one feature. Front Camera megapixels and Primary camera megapixels are different entities despite of showing colinearity. So we'll be keeping them as they are.



ML algorithms

1. Decision tree
2. Random Forest classification
3. Xgboost Classifier.
4. Gradient Boosting Classifier
5. K Nearest Neighbours
6. Support Vector Machine(SVM)

Support Vector Machine

Train_accuracy : 99%

```
#printing the classifiaction report of train set.
print(classification_report(y_train,y_train_pred))
```

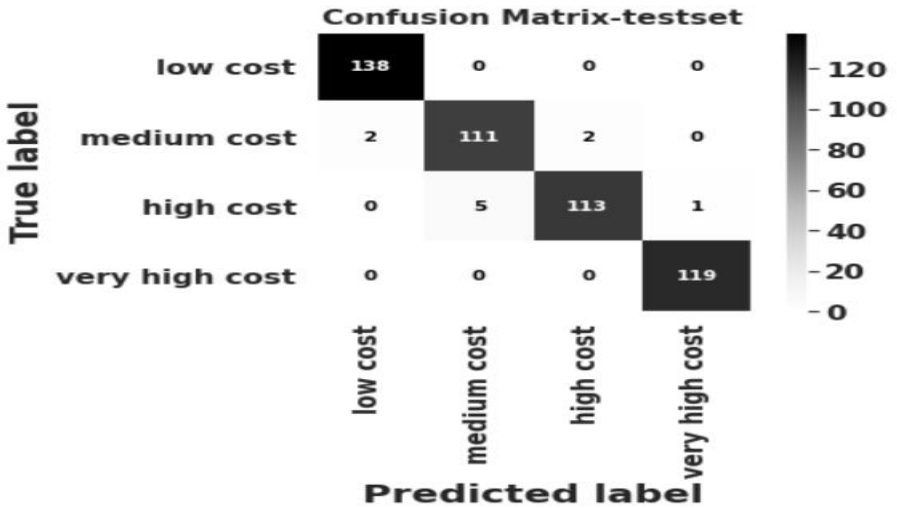
	precision	recall	f1-score	support
0.0	0.99	0.99	0.99	356
1.0	0.99	0.97	0.98	376
2.0	0.97	0.99	0.98	372
3.0	1.00	0.99	0.99	367
accuracy			0.99	1471
macro avg	0.99	0.99	0.99	1471
weighted avg	0.99	0.99	0.99	1471

Decision tree with hyperparameter tuning

Test_accuracy : 98%

```
#printing the classifiaction report of train set.
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0.0	0.99	1.00	0.99	138
1.0	0.96	0.97	0.96	115
2.0	0.98	0.95	0.97	119
3.0	0.99	1.00	1.00	119
accuracy			0.98	491
macro avg	0.98	0.98	0.98	491
weighted avg	0.98	0.98	0.98	491



XGboost

Train_accuracy :
100%

```
#printing the classifiaction report of train set.
print(classification_report(y_train,y_train_pred))
```

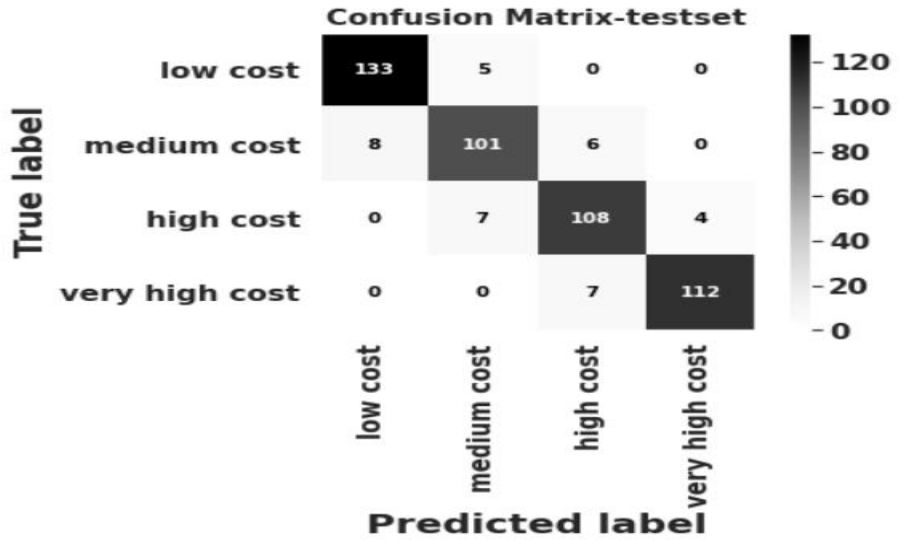
	precision	recall	f1-score	support
0.0	1.00	1.00	1.00	356
1.0	1.00	1.00	1.00	376
2.0	1.00	1.00	1.00	372
3.0	1.00	1.00	1.00	367
accuracy			1.00	1471
macro avg	1.00	1.00	1.00	1471
weighted avg	1.00	1.00	1.00	1471

XGboost with hyperparameter tuning

Test_accuracy : 92%

```
#printing the classifiaction report of test set.
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0.0	0.94	0.96	0.95	138
1.0	0.89	0.88	0.89	115
2.0	0.89	0.91	0.90	119
3.0	0.97	0.94	0.95	119
accuracy			0.92	491
macro avg	0.92	0.92	0.92	491
weighted avg	0.92	0.92	0.92	491



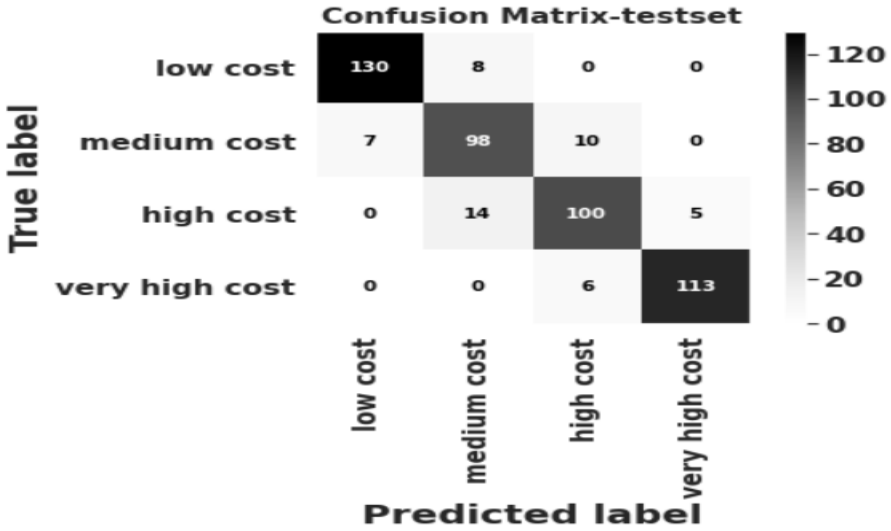
Random forest classifier with hyper parameter tuning

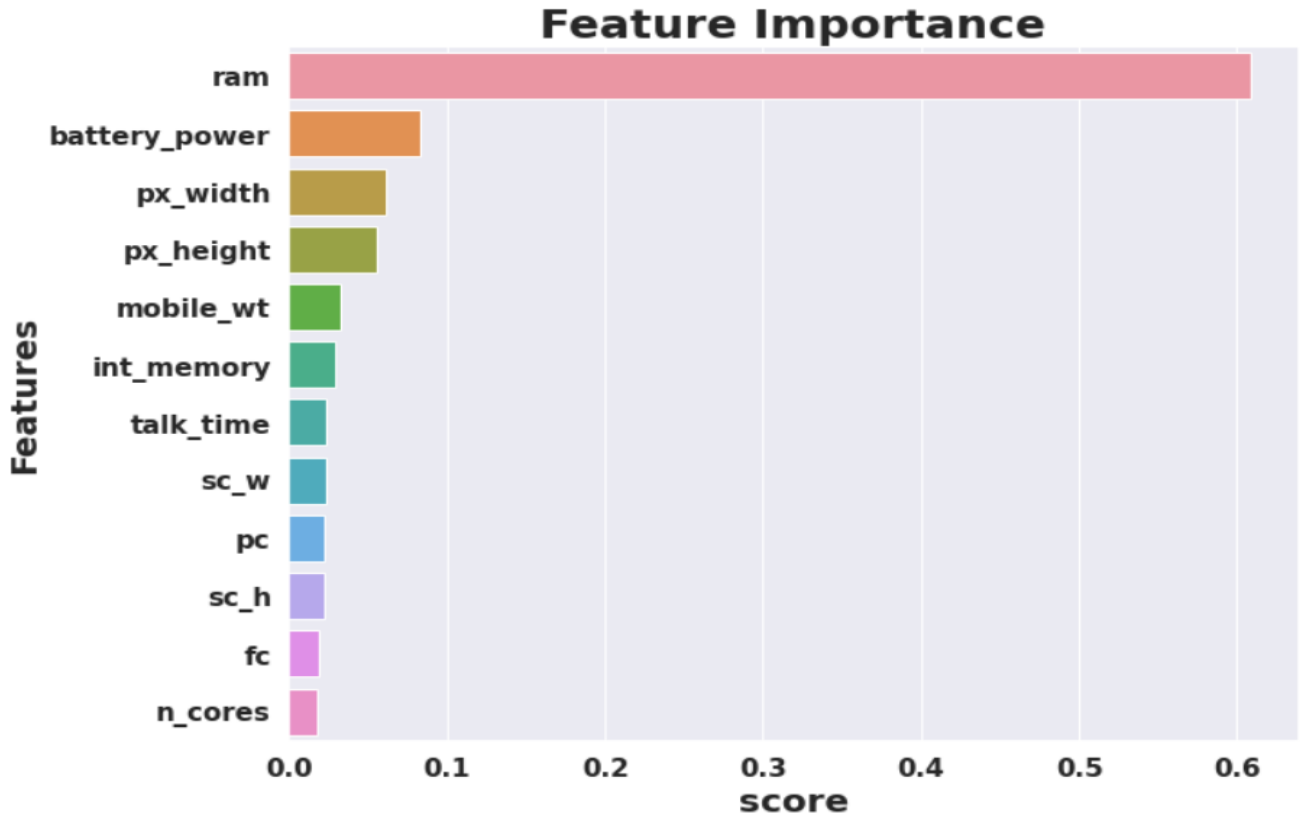
Train_accuracy : 1 %

Test_accuracy : 90 %

```
# printing the classification report for train set
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0.0	0.95	0.94	0.95	138
1.0	0.82	0.85	0.83	115
2.0	0.86	0.84	0.85	119
3.0	0.96	0.95	0.95	119
accuracy			0.90	491
macro avg	0.90	0.90	0.90	491
weighted avg	0.90	0.90	0.90	491





As we can see the top 3 important features of our dataset are: RAM, battery_power ,pixels

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 K best feature selection method using Chi square statistic.
- 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.
- XG boost is the second best good model which gave good performance after hyper-parameter tuning with 100% train accuracy and 92.25% test accuracy score.
- KNN gave very worst model performance.
- We checked for the feature importance's of each model. RAM, Battery Power, Px_height and px_width contributed the most while predicting the price range.