Capstone Project-3

Mobile-Price-Range-Prediction

By
Usha Rani Bolimera
Data Science Trainee, Almabetter

Content :

- Problem statement
- EDA and feature engineering
- Feature Selection
- Applying Model
- Model Validation and Selection
- Conclusion

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.

Data Summary :

Independent variables :

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

Data Summary contd...

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

Three_g - Has 3G or not

Touch_screen - Has touch screen or not

Wifi - Has wifi or not

Data Summary contd...

Dependent variables :

Price_range - This is the target variable with value of

- 0 - Low cost Phones
- 1 - Medium cost phones
- 2 - High cost phones
- 3 - Very High cost phones

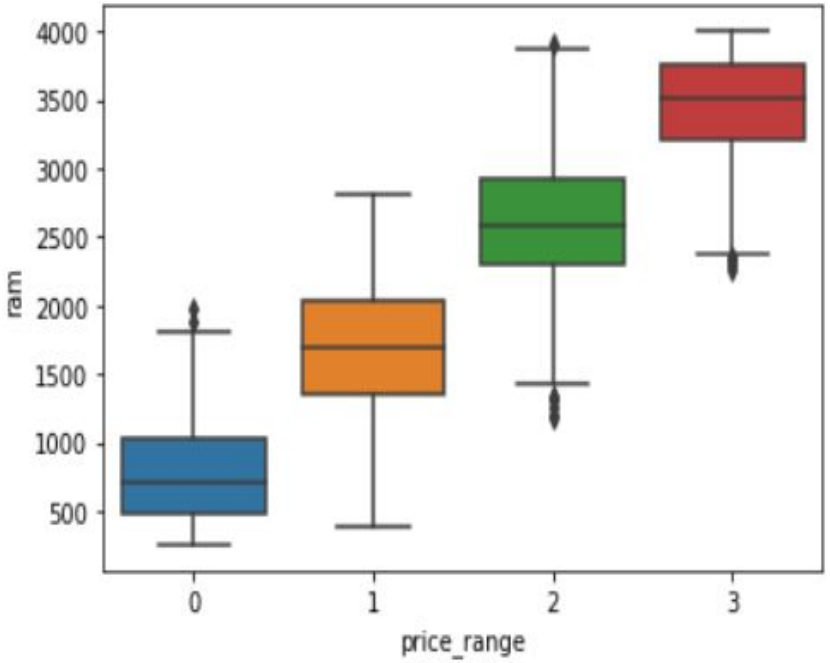
EDA

Relation Between Price Range & Ram

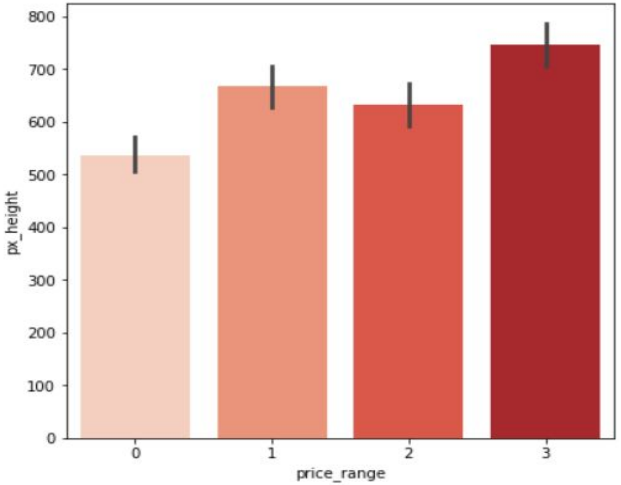
This is a positive relationship, with increase in RAM, price too increases. There are 4 types of price range

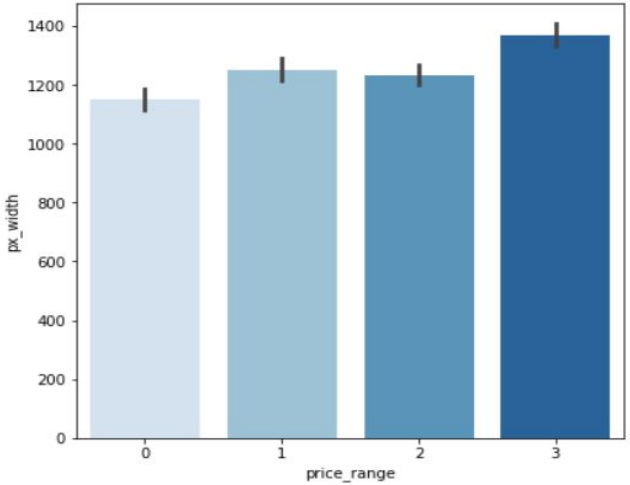
- Type 1(low cost): RAM ranges between 216 to 1974 megabytes
- Type 2(medium cost): RAM ranges between 387 to 2811 megabytes
- Type 3(high cost): RAM ranges between 1185 to 3916 megabytes
- Type 4(very high cost): RAM ranges between 2255 to 4000 megabytes

Price Range Vs Ram



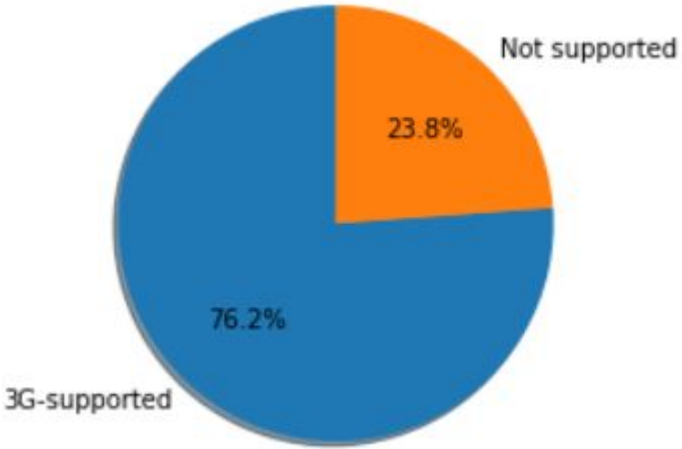
Relationship between the Price Range and Pixel Height / Width

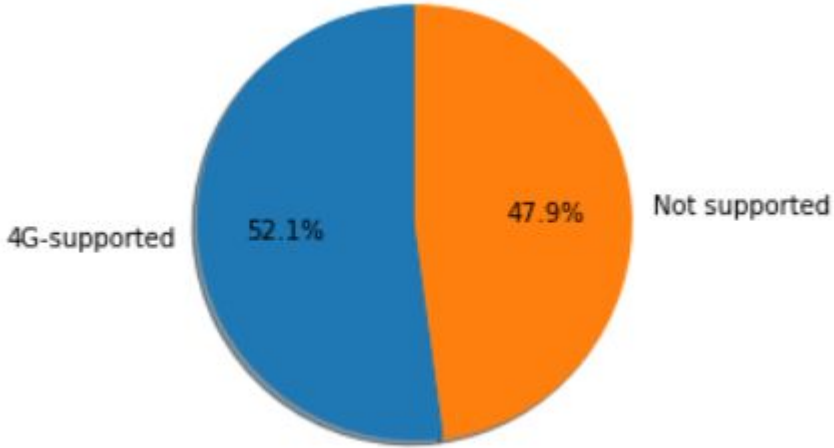




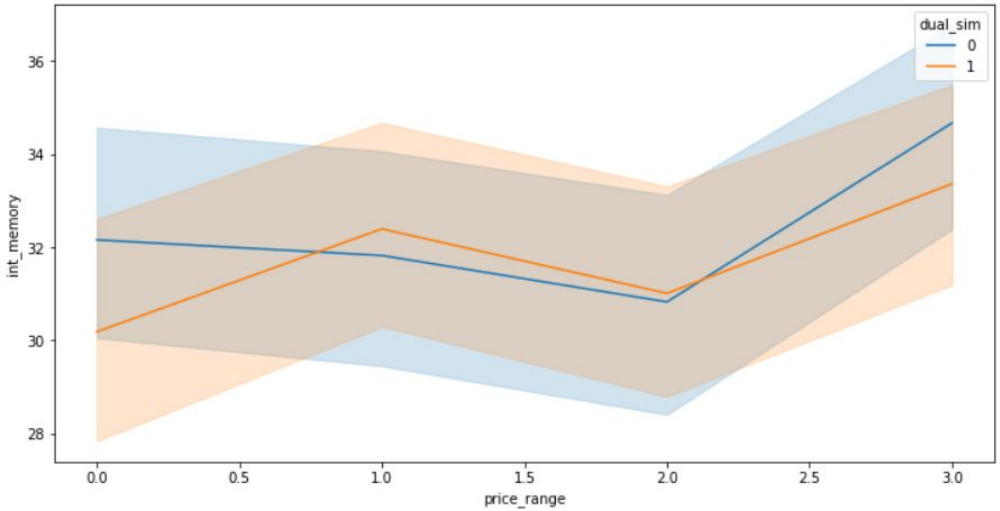
- From the above bar plot, we can see that the average pixel height and width are highest for the price range 3(very high cost).
- Low-cost phones have smaller average pixel width and pixel height.
- We can observe from this Bar plot that pixel height and pixel width are roughly equal in relevance when it comes to model development for prediction.

EDA contd..


Not supported
23.8%
76.2%
3G-supported


4G-supported
52.1%
47.9%
Not supported

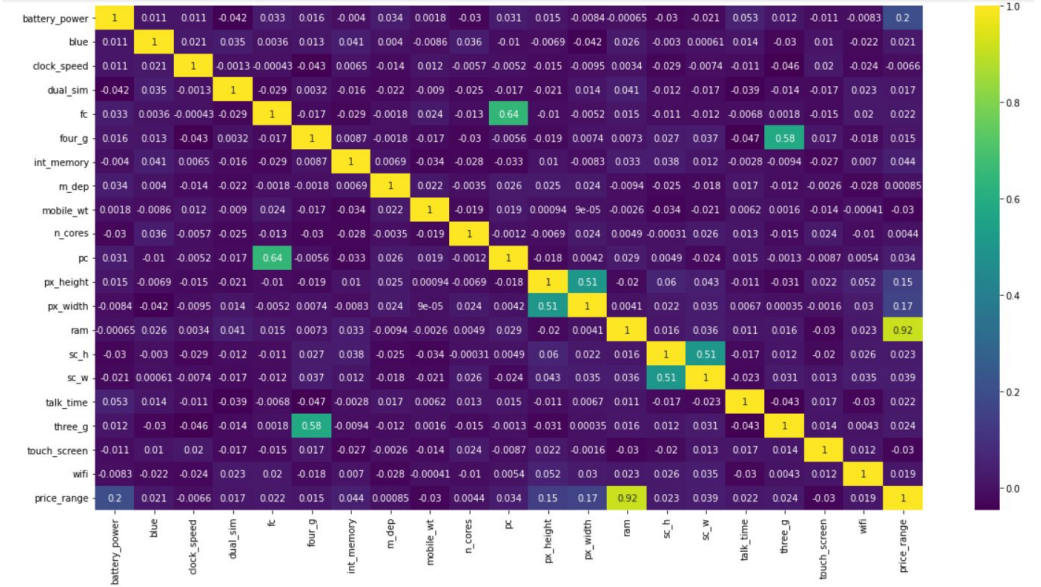
Multivariate analysis - int_memory, mobile_wt



- There is drastic increase in internal memory for vey high prices.
- Also there is drastic Decrease in mobile weight for very high price.

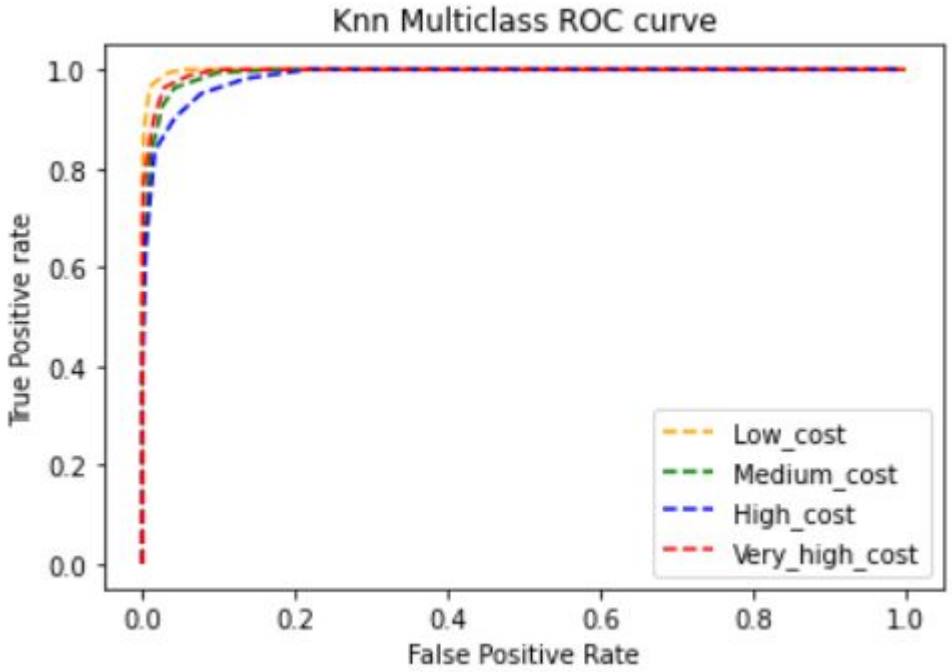
Multivariate analysis

- Pc is correlated with Fc .
- Px_height and px_width are moderately correlated.
- Sc_h and sc_w are moderately correlated.
- Ram is highly correlated with price_range.



Implementing K Neighbours Classifier

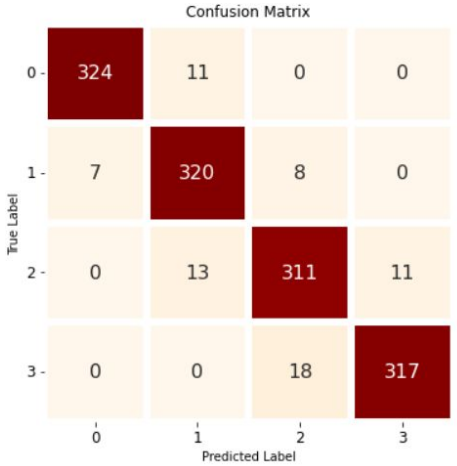
TPR = TP / (TP+FN)
FPR = FP / (FP+TN)



Implementing K Neighbours Classifier contd.

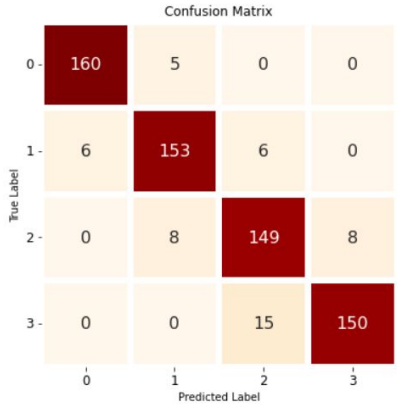
Train metrics

	precision	recall	f1-score	support
0	0.98	0.97	0.97	335
1	0.93	0.96	0.94	335
2	0.92	0.93	0.93	335
3	0.97	0.95	0.96	335
accuracy			0.95	1340
macro avg	0.95	0.95	0.95	1340
weighted avg	0.95	0.95	0.95	1340



Test metrics

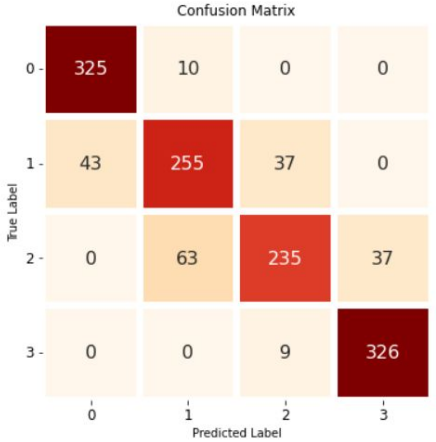
	precision	recall	f1-score	support
0	0.96	0.97	0.97	165
1	0.92	0.93	0.92	165
2	0.88	0.90	0.89	165
3	0.95	0.91	0.93	165
accuracy			0.93	660
macro avg	0.93	0.93	0.93	660
weighted avg	0.93	0.93	0.93	660



Implementing Random Forest Classifier

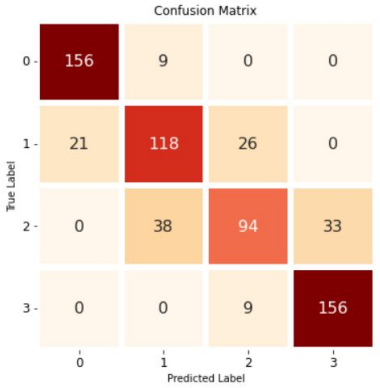
Train metrics

	precision	recall	f1-score	support
0	0.88	0.97	0.92	335
1	0.78	0.76	0.77	335
2	0.84	0.70	0.76	335
3	0.90	0.97	0.93	335
accuracy			0.85	1340
macro avg	0.85	0.85	0.85	1340
weighted avg	0.85	0.85	0.85	1340



Test metrics

	precision	recall	f1-score	support
0	0.88	0.95	0.91	165
1	0.72	0.72	0.72	165
2	0.73	0.57	0.64	165
3	0.83	0.95	0.88	165
accuracy			0.79	660
macro avg	0.79	0.79	0.79	660
weighted avg	0.79	0.79	0.79	660

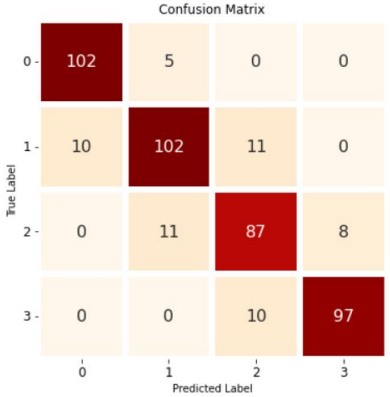


Implementing Gradient Boosting Classifier

Train metrics

Classification Report

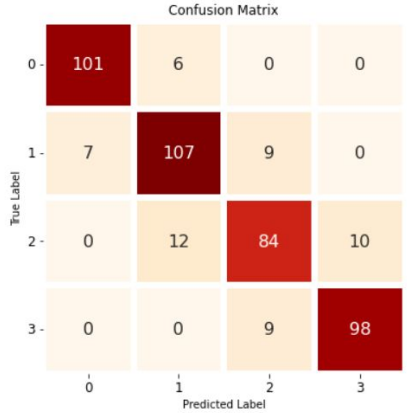
	precision	recall	f1-score	support
0	0.91	0.95	0.93	107
1	0.86	0.83	0.85	123
2	0.81	0.82	0.81	106
3	0.92	0.91	0.92	107
accuracy			0.88	443
macro avg	0.88	0.88	0.88	443
weighted avg	0.88	0.88	0.88	443



Test metrics

Classification Report

	precision	recall	f1-score	support
0	0.94	0.94	0.94	107
1	0.86	0.87	0.86	123
2	0.82	0.79	0.81	106
3	0.91	0.92	0.91	107
accuracy			0.88	443
macro avg	0.88	0.88	0.88	443
weighted avg	0.88	0.88	0.88	443

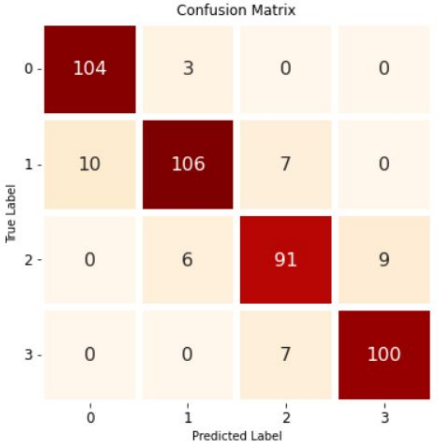


Implementing XGB Classifier

Train metrics

Classification Report

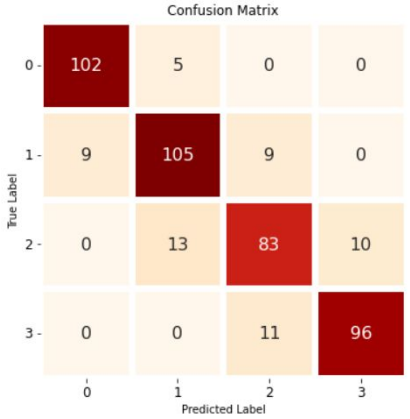
	precision	recall	f1-score	support
0	0.91	0.97	0.94	107
1	0.92	0.86	0.89	123
2	0.87	0.86	0.86	106
3	0.92	0.93	0.93	107
accuracy			0.91	443
macro avg	0.90	0.91	0.91	443
weighted avg	0.91	0.91	0.90	443



Test metrics

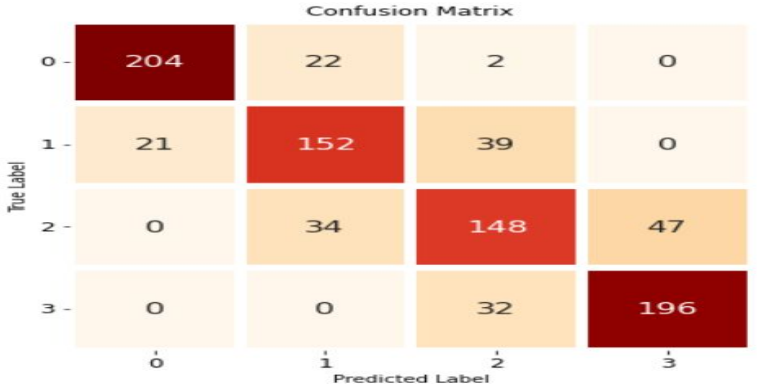
Classification Report

	precision	recall	f1-score	support
0	0.92	0.95	0.94	107
1	0.85	0.85	0.85	123
2	0.81	0.78	0.79	106
3	0.91	0.90	0.90	107
accuracy			0.87	443
macro avg	0.87	0.87	0.87	443
weighted avg	0.87	0.87	0.87	443

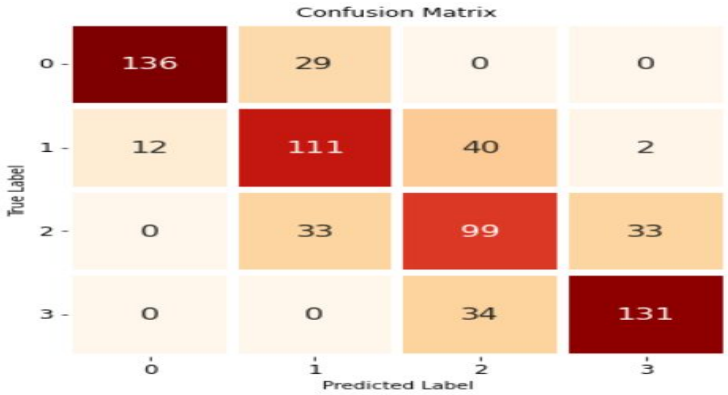


Implementing Logistic regression

	precision	recall	f1-score	support
0	0.91	0.89	0.90	228
1	0.73	0.72	0.72	212
2	0.67	0.65	0.66	229
3	0.81	0.86	0.83	228
accuracy			0.78	897
macro avg	0.78	0.78	0.78	897
weighted avg	0.78	0.78	0.78	897



	precision	recall	f1-score	support
0	0.92	0.82	0.87	165
1	0.64	0.67	0.66	165
2	0.57	0.60	0.59	165
3	0.79	0.79	0.79	165
accuracy			0.72	660
macro avg	0.73	0.72	0.73	660
weighted avg	0.73	0.72	0.73	660



Model Validation & Selection contd …

Observations:

- As seen in the above slides Random forest classifier is not giving great results , GradientBoostingClassifier is bit better than Random forest in recall and precision
- XGboost classifier is giving the better results than GB but the recall of random forest classifier is somewhat similar
- KNeighbors is giving the best results among all of the algorithms
- Logistic regression is giving low results among all of them

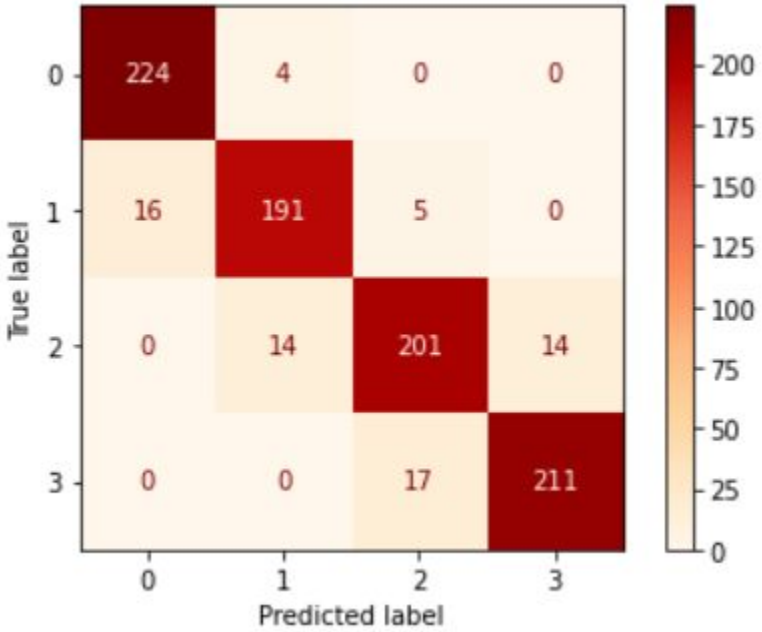
Model Validation & Selection contd …

So we had chosen Kneighbors classifier for the prediction and the best hyperparameters obtained are as below

Best hyperparameters :
Train : (algorithm='auto', leaf_size=30, metric='Euclidean', metric_params=None, n_jobs=None, n_neighbors=11, p=2, weights='distance')

Test : (algorithm='auto', leaf_size=30, metric='euclidean', metric_params=None, n_jobs=None, n_neighbors=17, p=2, weights='distance')

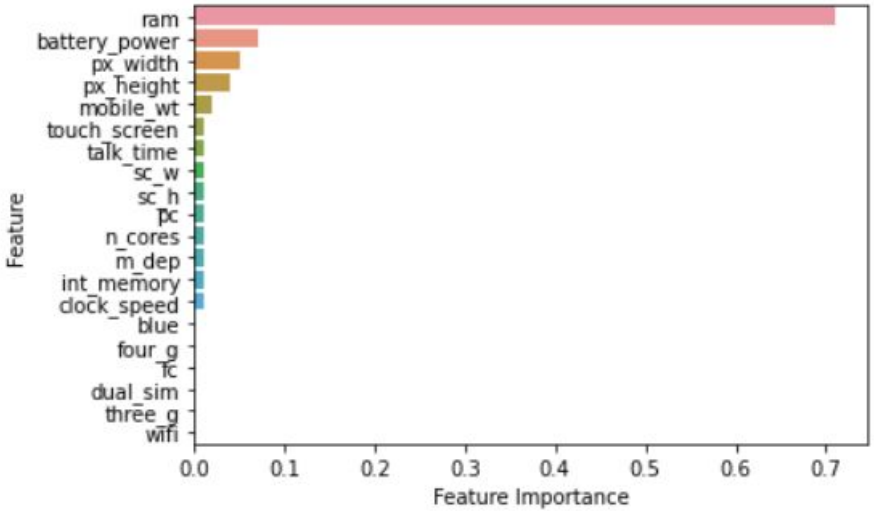
Model Validation & Selection(Hyper paramter tuned)



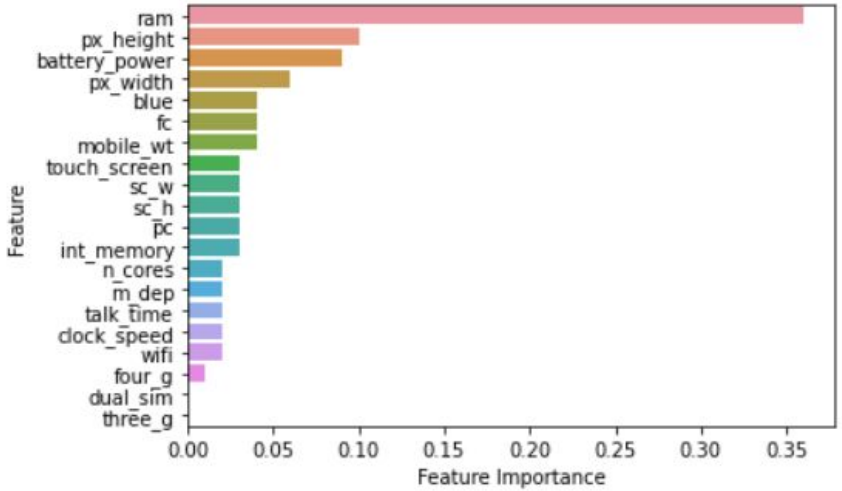
	precision	recall	f1-score	support
0	0.93	0.98	0.96	228
1	0.91	0.90	0.91	212
2	0.90	0.88	0.89	229
3	0.94	0.93	0.93	228
accuracy			0.92	897
macro avg	0.92	0.92	0.92	897
weighted avg	0.92	0.92	0.92	897

Feature Importance

Random Forest Classifier



XGBoost Classifier



Conclusion

- Ram , Battery_power features were found to be the most relevant features for predicting price range of mobiles and dropping negative correlation features which are clock speed , mobile_wt , touch_screen
- Kneighbors and Xgboost are given best accuracy score 93% test ,95% train and 91% train , 88% test respectively and roc_auc score for kneighbors is 99%
- Tuning the hyperparameters by GridSearchCV on kneighbors but not getting much difference in results but the best parameters n_neighbors for train and test are 11and 17
- So we conclude that kneighbors classifier is giving the best results for these dataset
- So we can say that in the price range prediction as the ram and battery_power increases the price range will increase for sure.