Content

- Problem statement
- Data summary
- EDA
- Modelling
- Metrics
- Challenges
- Conclusions

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:

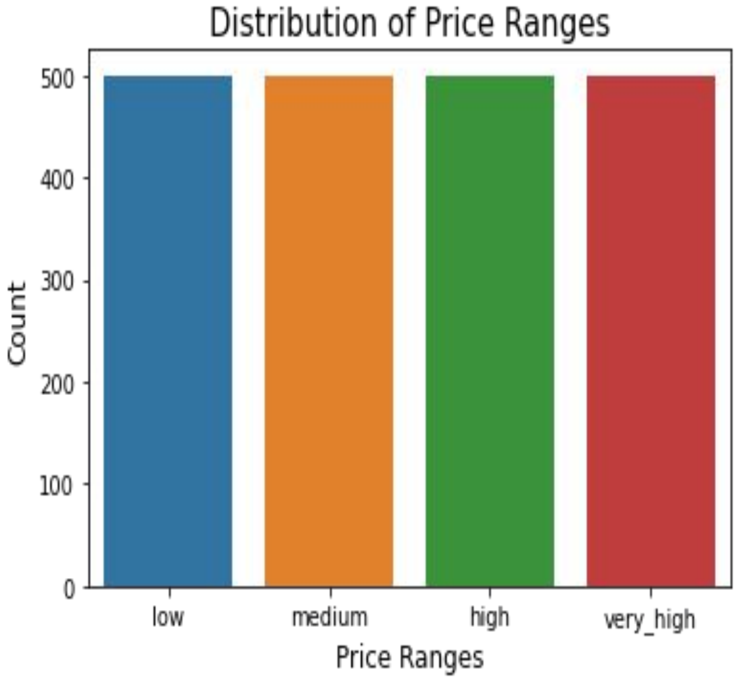
- 0 - Low cost Phones
- 1 - Medium cost phones
- 2 - High cost phones
- 3 - Very High cost phones

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

- We have records of 2000 mobile phones with 20 columns/features.
- We have perfectly balanced dataset with 500 observations for each class.
- Each column represents the feature of the mobile.
- Interestingly , we had zero null values.



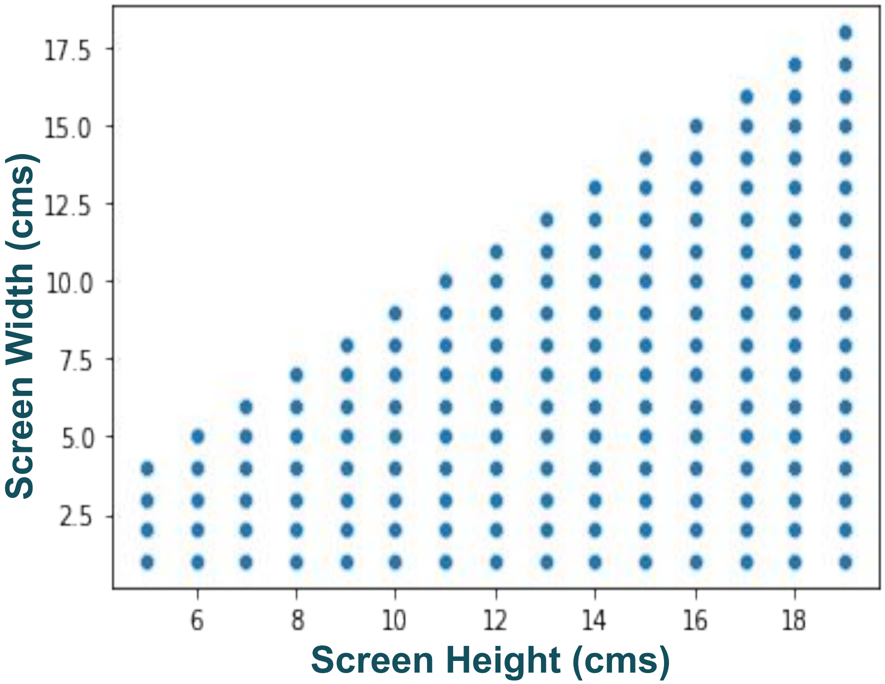
Data Overview

- **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
- **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).

Handling Discrepancies

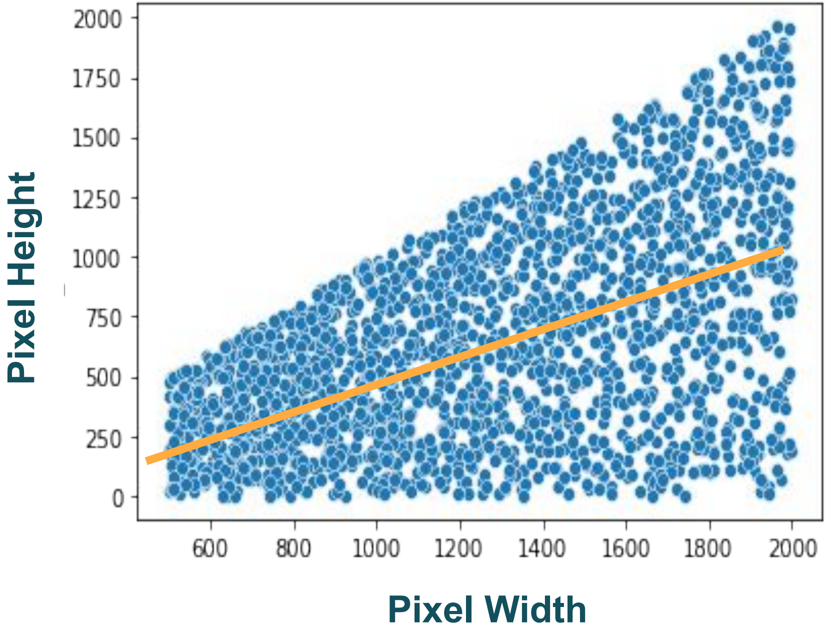
In the data we observed that in 9% of rows the value for columns 'sc_w' (screen width) is 0, which is not possible in real life.

As we can see in the plot, for each value of 'sc_h' there are multiple values of 'sc_w', so to handle zero values, we replaced them with mean of all available values 'sc_w' for all values of 'sc_h'.



Handling Discrepancies

There are also discrepancies in ‘Px_height’ column.
To handle those discrepancies we replaced those values by using linear regression.

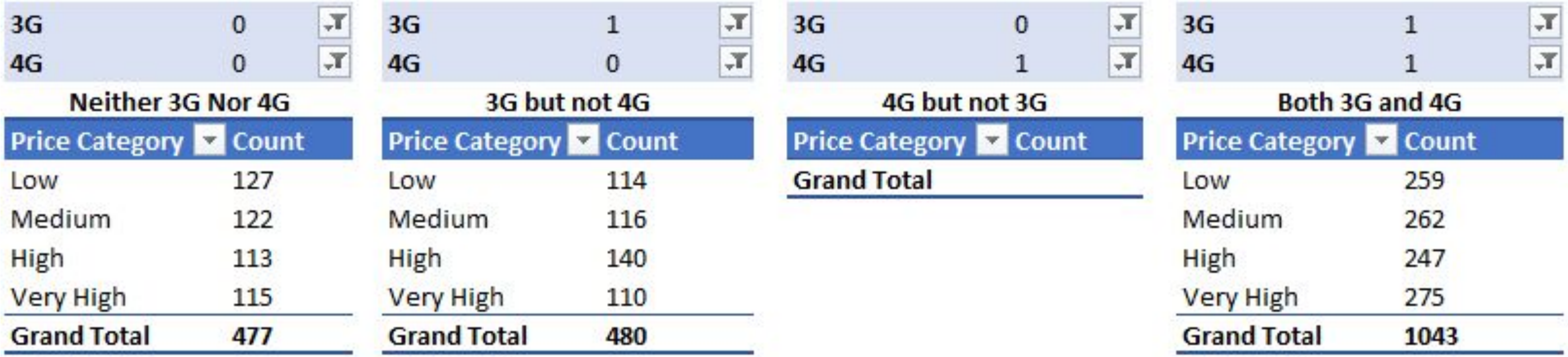


Feature Engineering

4,7"
5,8"
6,1"
5,5"
6,5"

- Generally the screen size of the phone is expressed in Inches.
- We have columns **‘sc_h’** and **‘sc_w’** out of which we have created a new feature **‘Screen_size’** which is diagonal length of the screen.

Feature Engineering



3G	0
4G	0

Neither 3G Nor 4G

Price Category	Count
Low	127
Medium	122
High	113
Very High	115
Grand Total	477

3G	1
4G	0

3G but not 4G

Price Category	Count
Low	114
Medium	116
High	140
Very High	110
Grand Total	480

3G	0
4G	1

4G but not 3G

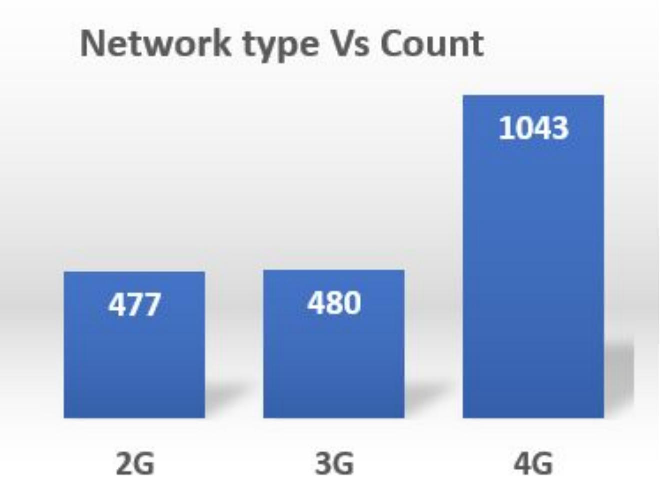
Price Category	Count
Grand Total	

3G	1
4G	1

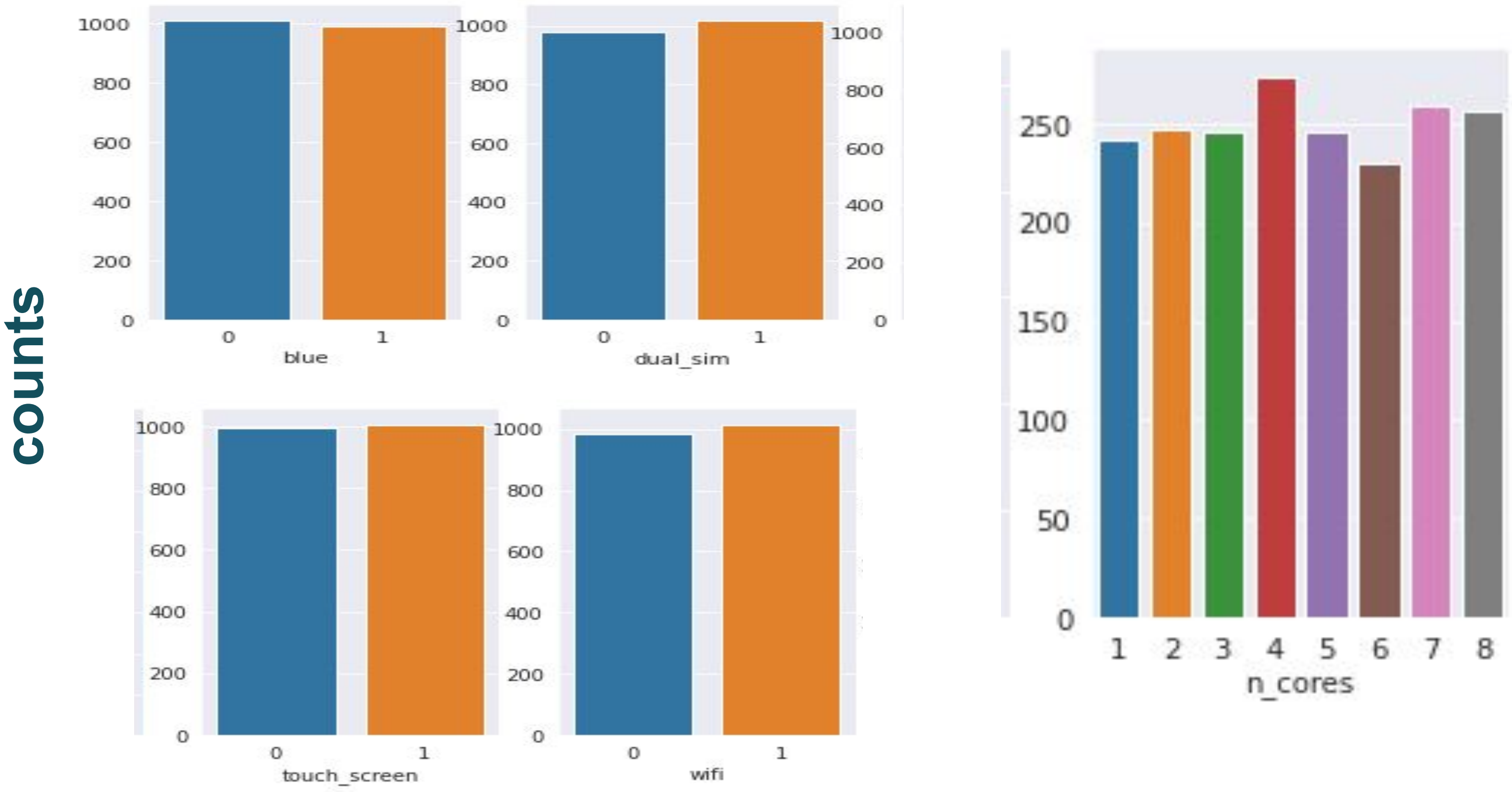
Both 3G and 4G

Price Category	Count
Low	259
Medium	262
High	247
Very High	275
Grand Total	1043

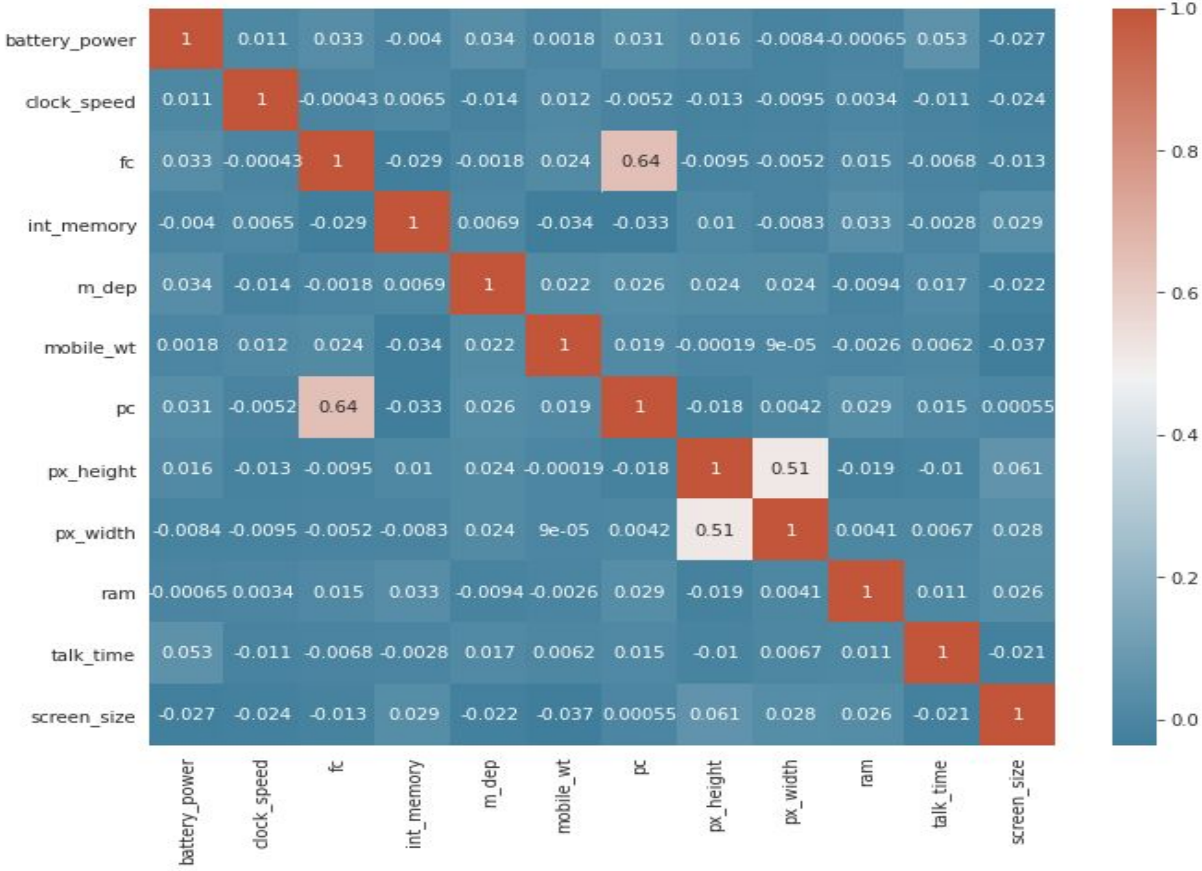
- We observed that if a phone supports 4G, it by default has 3G as well. So we don’t really need two columns for this.
- We created a single column ‘network’ by the addition of 3G and 4G. Where:
- 0 - 2G, 1 - 3G, 2 - 4G.



Categorical Analysis



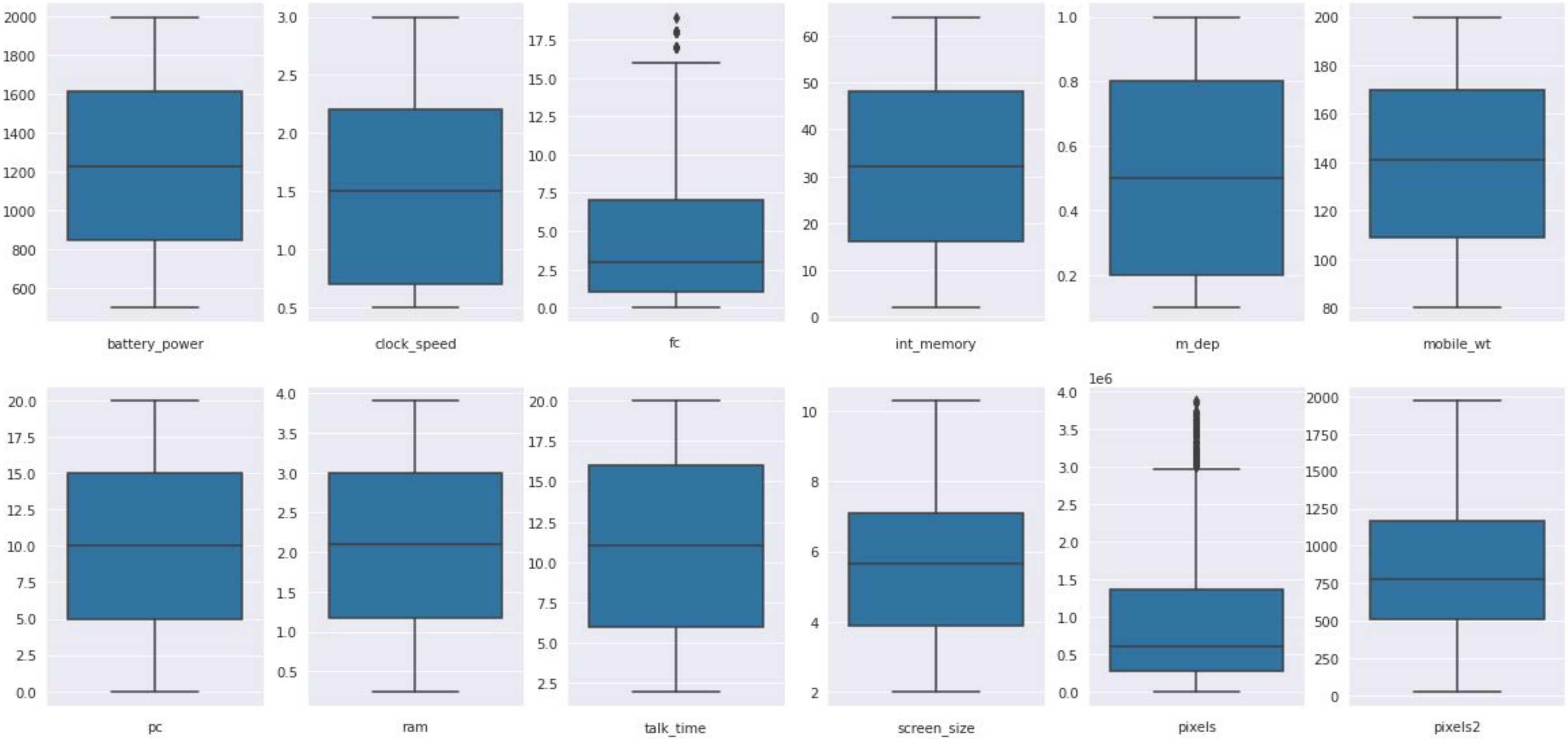
Collinearity



	battery_power	clock_speed	fc	int_memory	m_dep	mobile_wt	pc	px_height	px_width	ram	talk_time	screen_size
battery_power	1	0.011	0.033	-0.004	0.034	0.0018	0.031	0.016	-0.0084	-0.00065	0.053	-0.027
clock_speed	0.011	1	-0.00043	0.0065	-0.014	0.012	-0.0052	-0.013	-0.0095	0.0034	-0.011	-0.024
fc	0.033	-0.00043	1	-0.029	-0.0018	0.024	0.64	-0.0095	-0.0052	0.015	-0.0068	-0.013
int_memory	-0.004	0.0065	-0.029	1	0.0069	-0.034	-0.033	0.01	-0.0083	0.033	-0.0028	0.029
m_dep	0.034	-0.014	-0.0018	0.0069	1	0.022	0.026	0.024	0.024	-0.0094	0.017	-0.022
mobile_wt	0.0018	0.012	0.024	-0.034	0.022	1	0.019	-0.00019	9e-05	-0.0026	0.0062	-0.037
pc	0.031	-0.0052	0.64	-0.033	0.026	0.019	1	-0.018	0.0042	0.029	0.015	0.00055
px_height	0.016	-0.013	-0.0095	0.01	0.024	-0.00019	-0.018	1	0.51	-0.019	-0.01	0.061
px_width	-0.0084	-0.0095	-0.0052	-0.0083	0.024	9e-05	0.0042	0.51	1	0.0041	0.0067	0.028
ram	-0.00065	0.0034	0.015	0.033	-0.0094	-0.0026	0.029	-0.019	0.0041	1	0.011	0.026
talk_time	0.053	-0.011	-0.0068	-0.0028	0.017	0.0062	0.015	-0.01	0.0067	0.011	1	-0.021
screen_size	-0.027	-0.024	-0.013	0.029	-0.022	-0.037	0.00055	0.061	0.028	0.026	-0.021	1

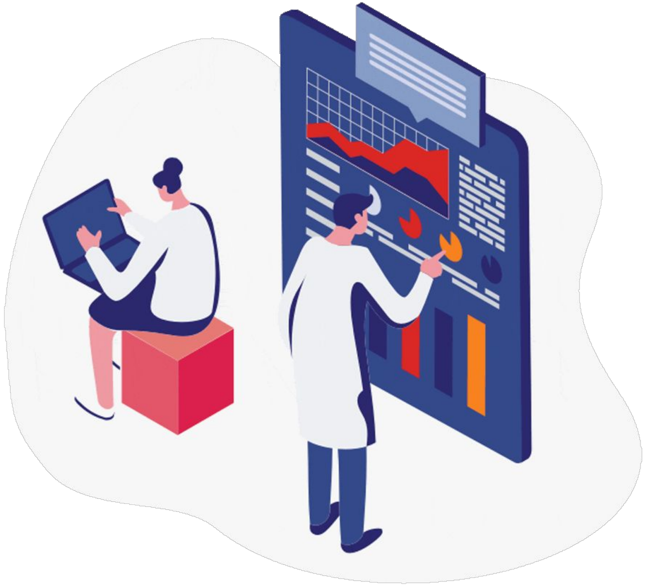
We created a new column ‘Pixels’ by taking the product of ‘px_height’ and ‘px_width’.

Outlier Analysis in continuous features



There were few outliers in ‘pixels’ column, so to handle outliers we replaced the values of ‘Pixels’ column with the square root.

Predictive Modelling

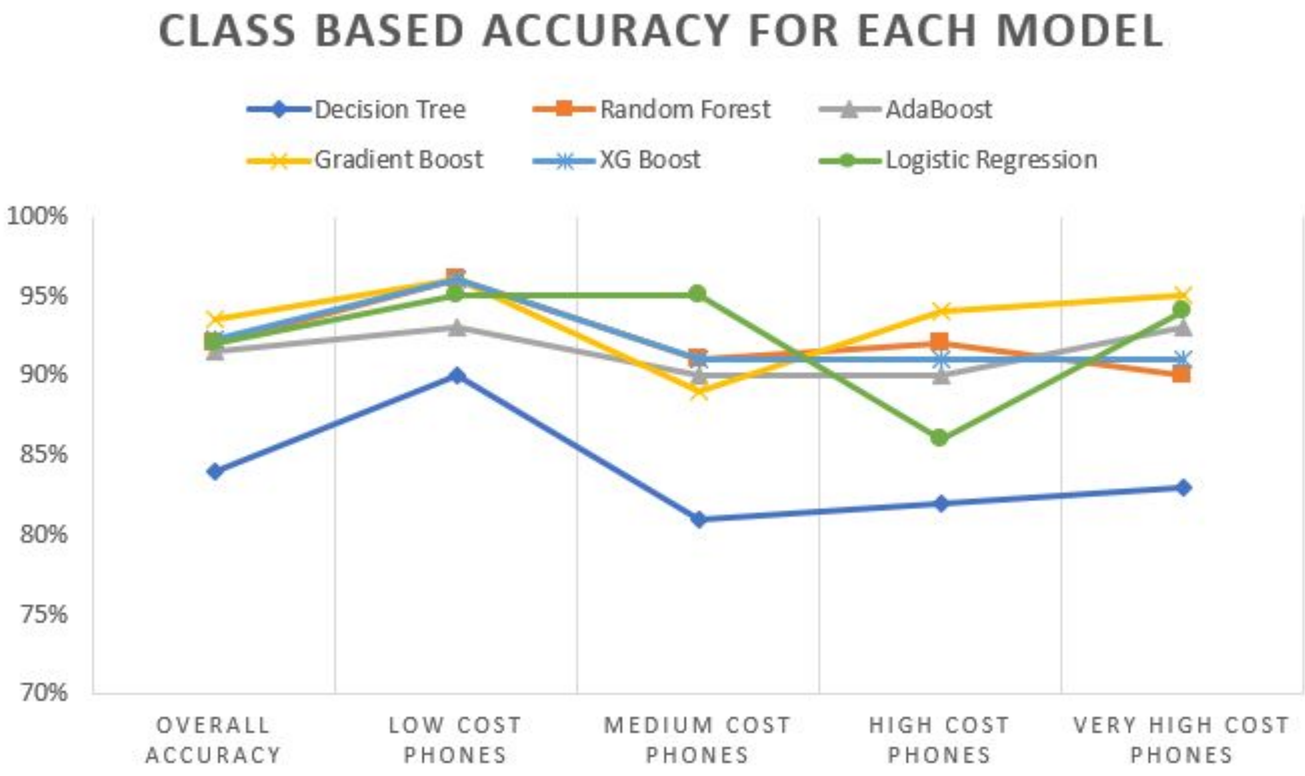


Hyperparameter Tuning - Grid Search - Cross Validation

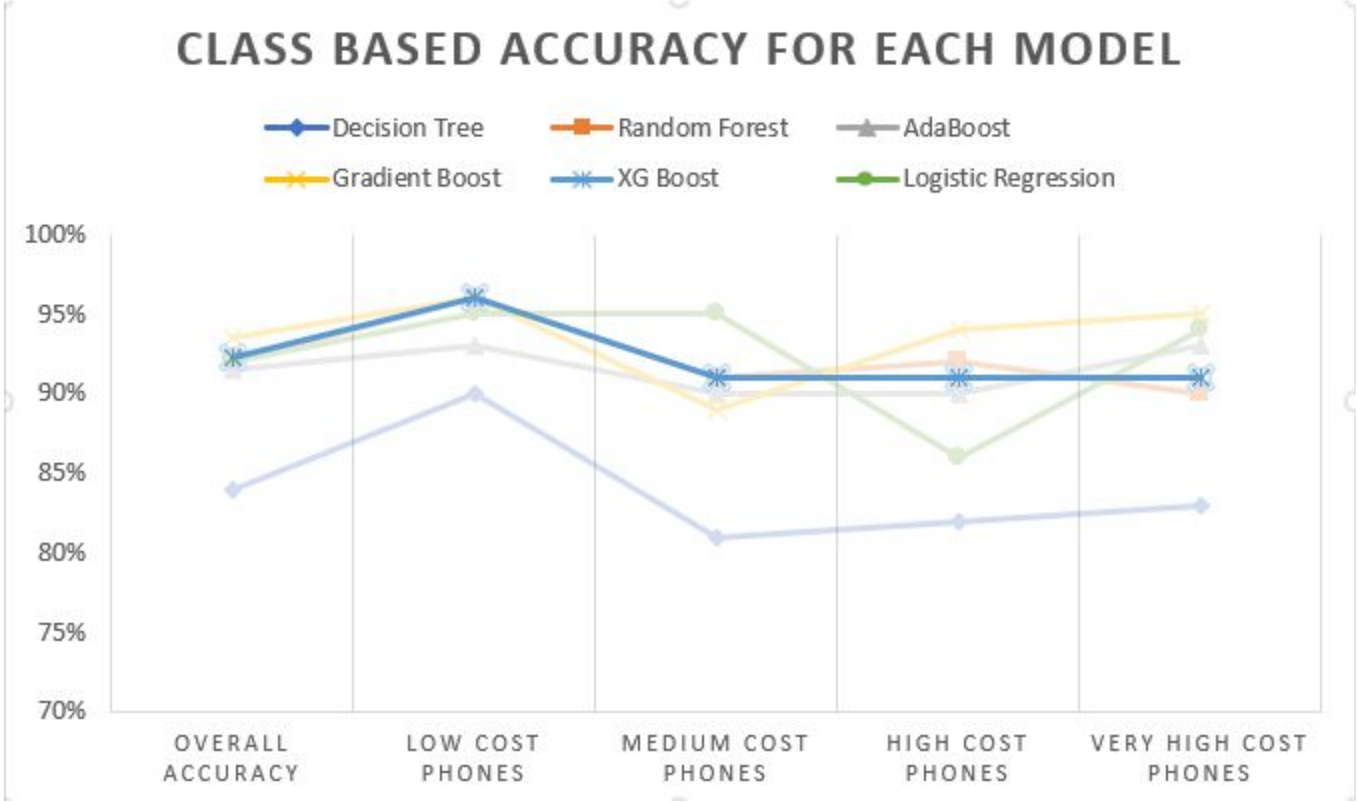
We compared 6 classifiers and evaluated them based on overall accuracy & class based accuracy as well.

- Decision Trees
- Random Forest
- Ada Boost
- Gradient Boosting
- XGBoost
- Logistic Regression

Comparison of Models



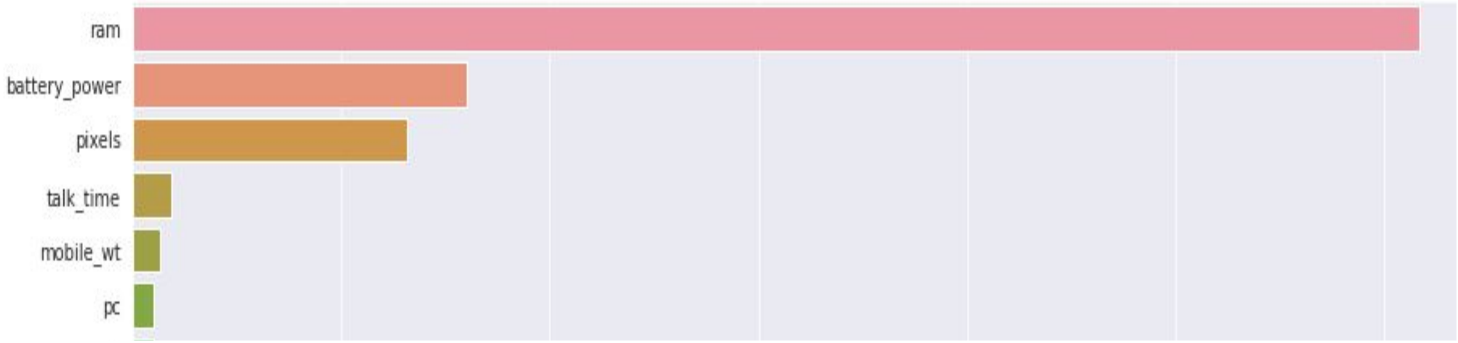
Comparison of Models



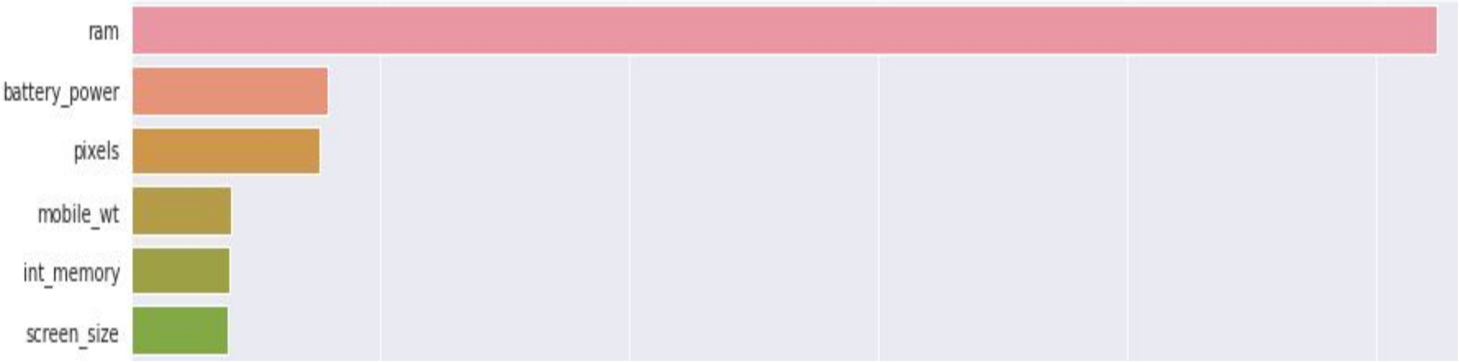
XG Boost is the best performing model on the given dataset

Feature Importance

Decision Tree

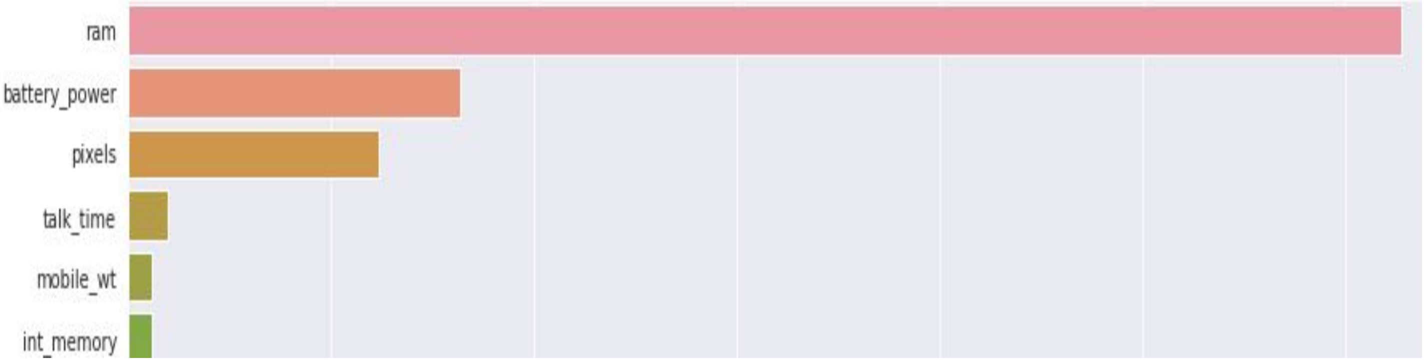


Random forest

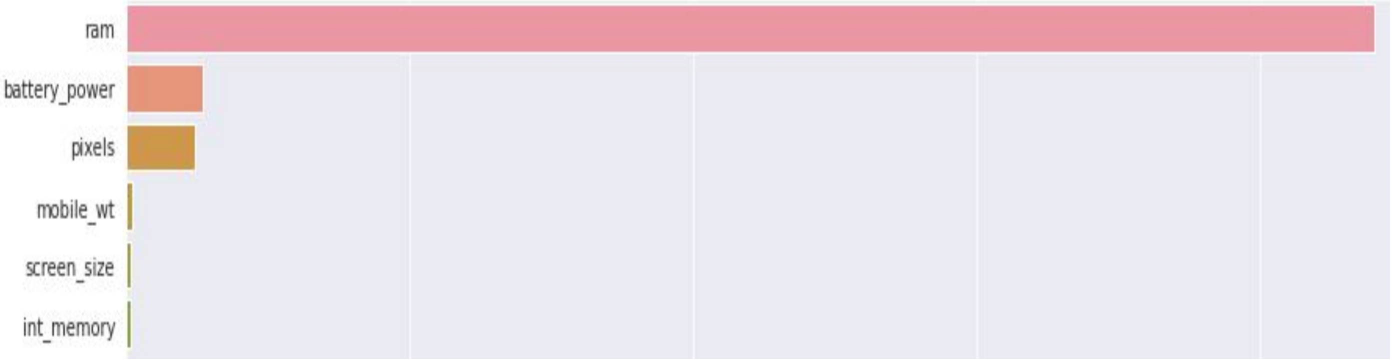


Feature Importance Contd..

Ada Boost



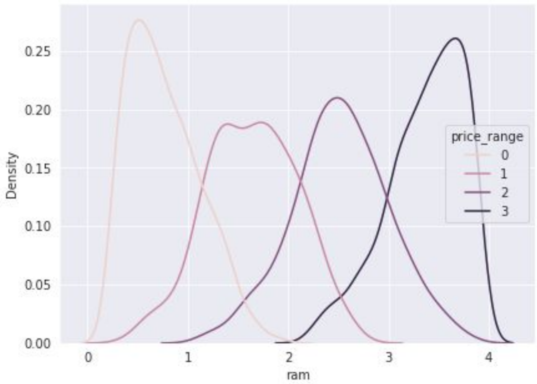
Gradient Boost

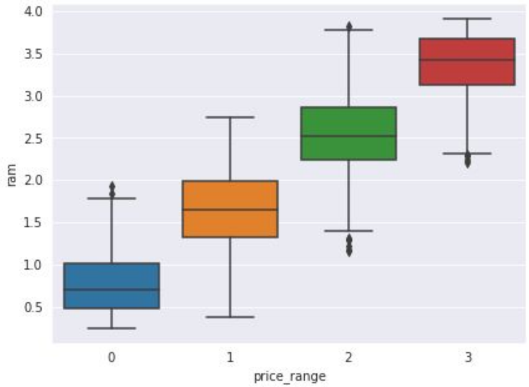


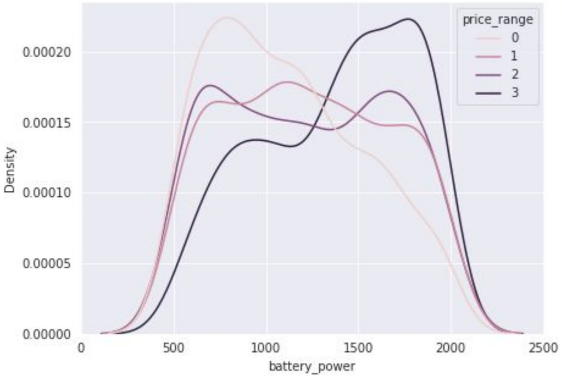
Feature Importance Contd..

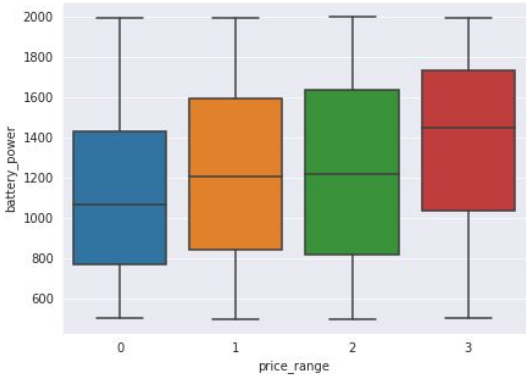
- **RAM and battery power are two most important features for the models.**

- **RAM and Battery power show the most variation along the different price ranges.**





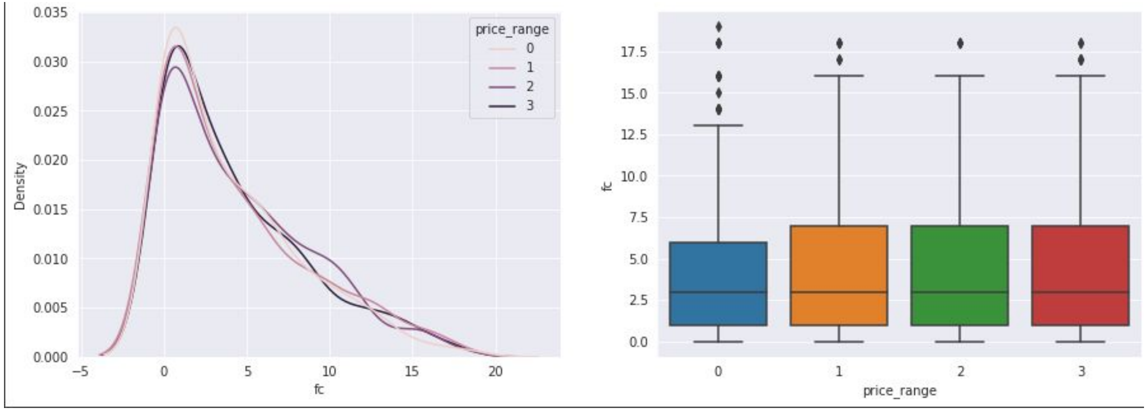


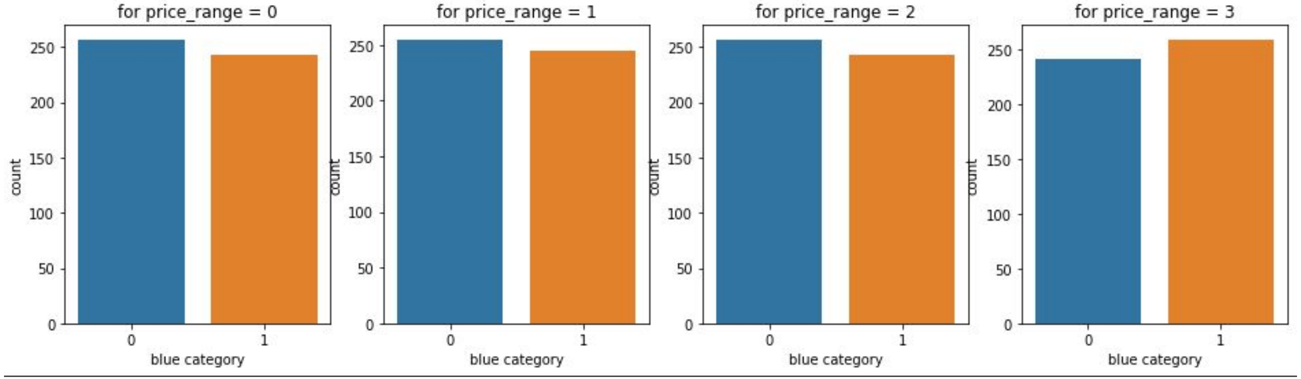


Feature Importance Contd..

- **Apart from selected important features any feature doesn’t show variation along the different price ranges.**

- **Here is example of front camera and bluetooth.**





Challenges

- We performed “Hypothesis driven EDA” based on domain, but unluckily most of our hypothesis got rejected by our data.

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

- We hit a ceiling at 94% accuracy using a single model.

Conclusion

- Gradient Boost, Random forest and ADABoost Models are also giving us good overall accuracy but they didn't perform well on Individual classes.

- Out of all the model we have tried XG Boost is performing well on Overall as well as Individual classes.

- Ram, Battery power, Mobile weight, Screen size and pixels are key features in predicting the mobile price range.

- 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.