Capstone Project - 3
Mobile Price Range Prediction

Name: Rudrajit Bhattacharyya
Cohort: Zanskar Pro

What’s inside?

1. Defining the problem statement
2. Defining the data
3. Data Cleaning/Pre-processing
4. EDA
   a. Distribution of all the features along with the target variable
   b. How the features vary with different price ranges
   c. Which features are influential in determining the price ranges
   d. Correlation analysis
5. Data Transformation/Preparation
6. Building Classification Models
7. Evaluating and Comparing all the Models
8. Conclusion

Defining the problem statement

In the competitive mobile phone market companies want to understand sales data of mobile phones and factors which drive the prices. Mobile phones come in all sorts of prices, features, specifications etc, and estimating the price of mobile phones is an important part of consumer strategy.

The objective of the project is to find out some relation between features of a mobile phone 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 dataset contains 2000 records with 21 features which is a mix of categorical and numerical features.

Data Summary

Categorical Features

- **Blue**: Has bluetooth or not
- **Dual_sim**: Has dual sim 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

For all features 0 means No, 1 means Yes

Numerical Features

- **Battery_power**: Capacity of the battery
- **Clock_speed:** Execution speed of microprocessor
- **Fc**: Front camera megapixels
- **Int_memory**: Internal memory storage
- **M_dep**: Mobile depth in cm
- **Mobile_wt**: Mobile weight
- **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
- **Sc_h**: Screen height in cm
- **Sc_w**: Screen width in cm
- **Talk_time**: Talk time in a single charge

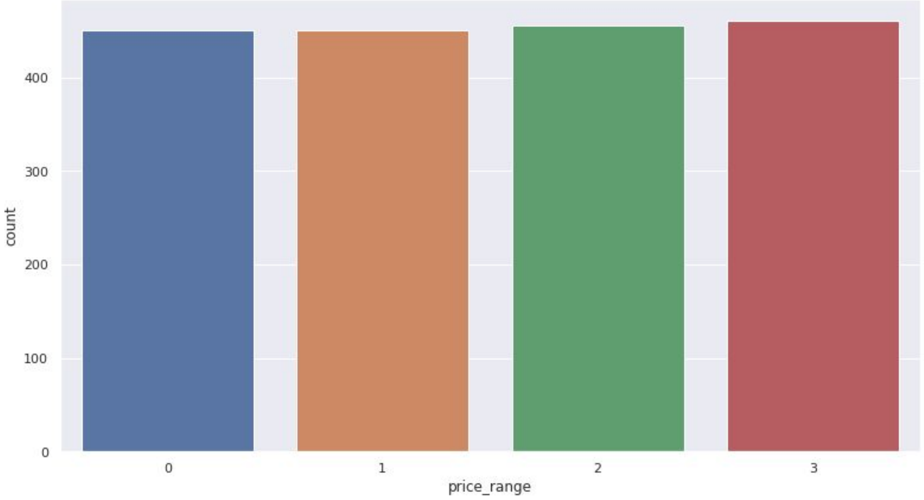
Data Cleaning

- The dataset was almost a cleaned one with no null values present or duplicate records found.
- The px_height and sc_w had some zero values which we had to remove before proceeding further as these values cannot be zero in real life.

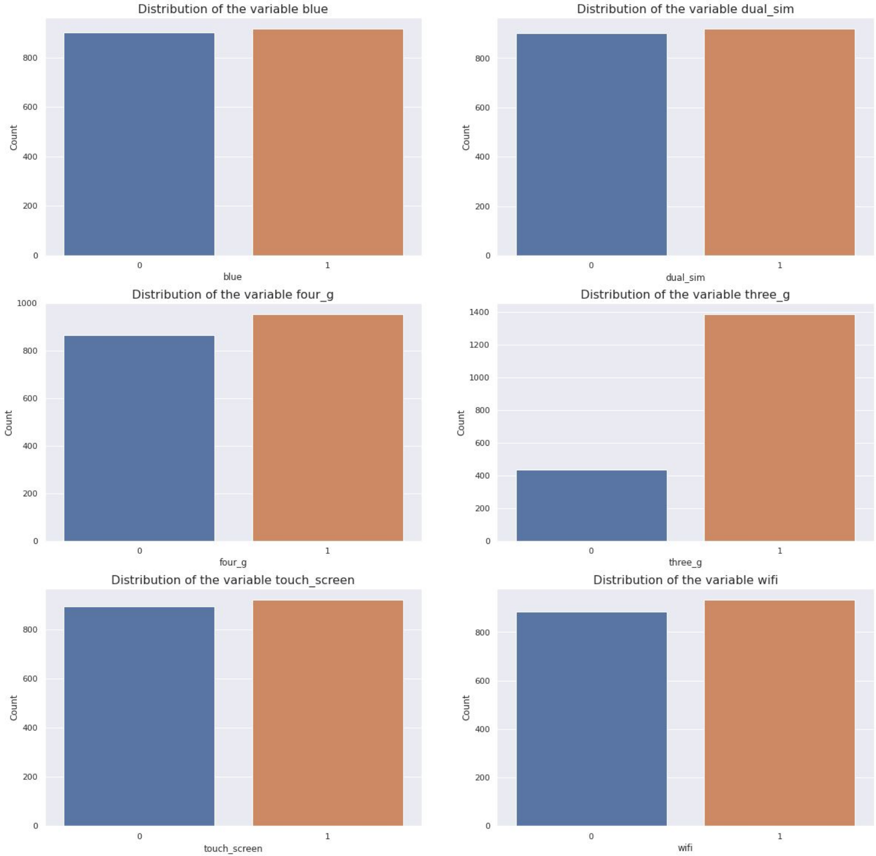
```
[ ] # remove zero values of pixel resolution height and screen width
    phone_df = phone_df[phone_df['sc_w'] != 0]
    phone_df = phone_df[phone_df['px_height'] != 0]
    phone_df.shape

    (1819, 21)
```

Exploratory Data Analysis

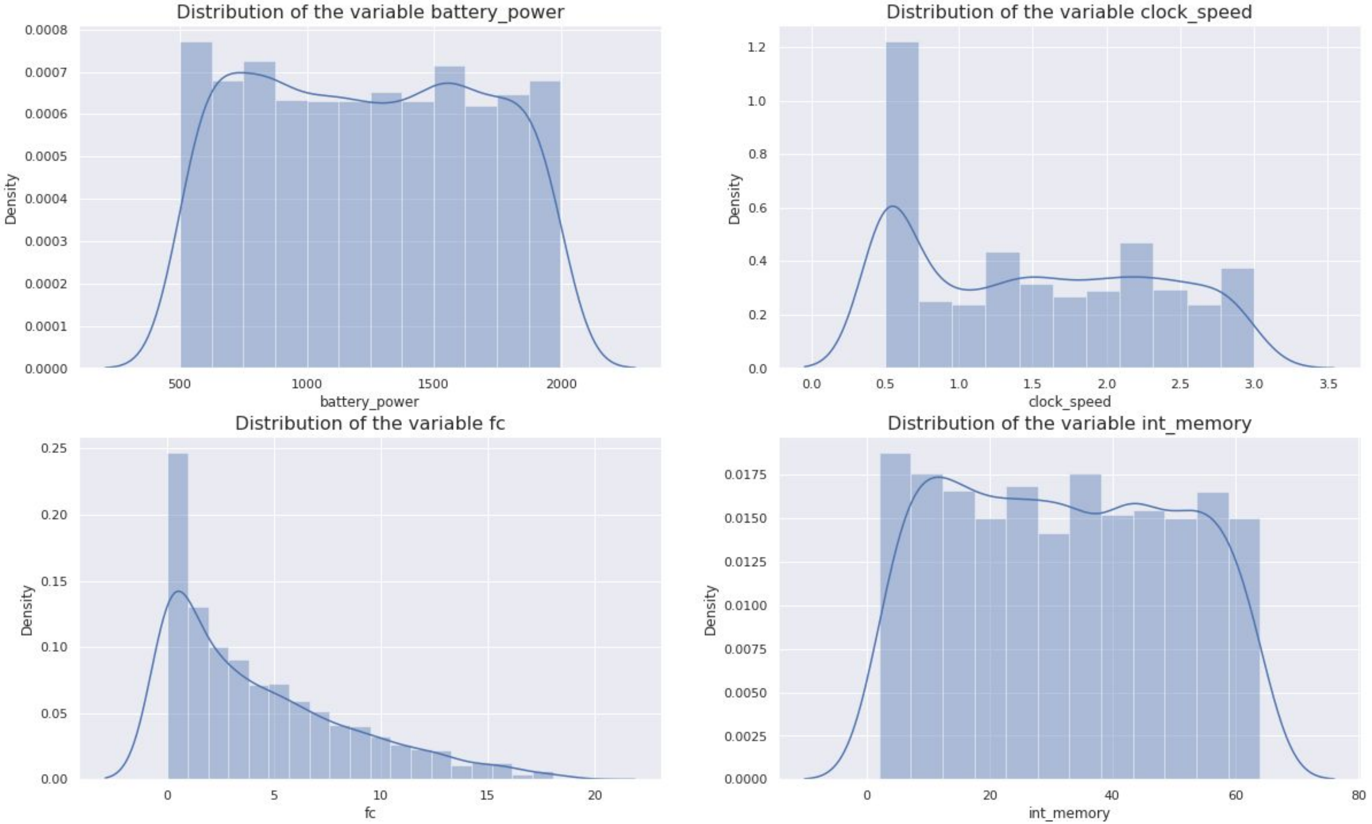
The target classes are almost balanced



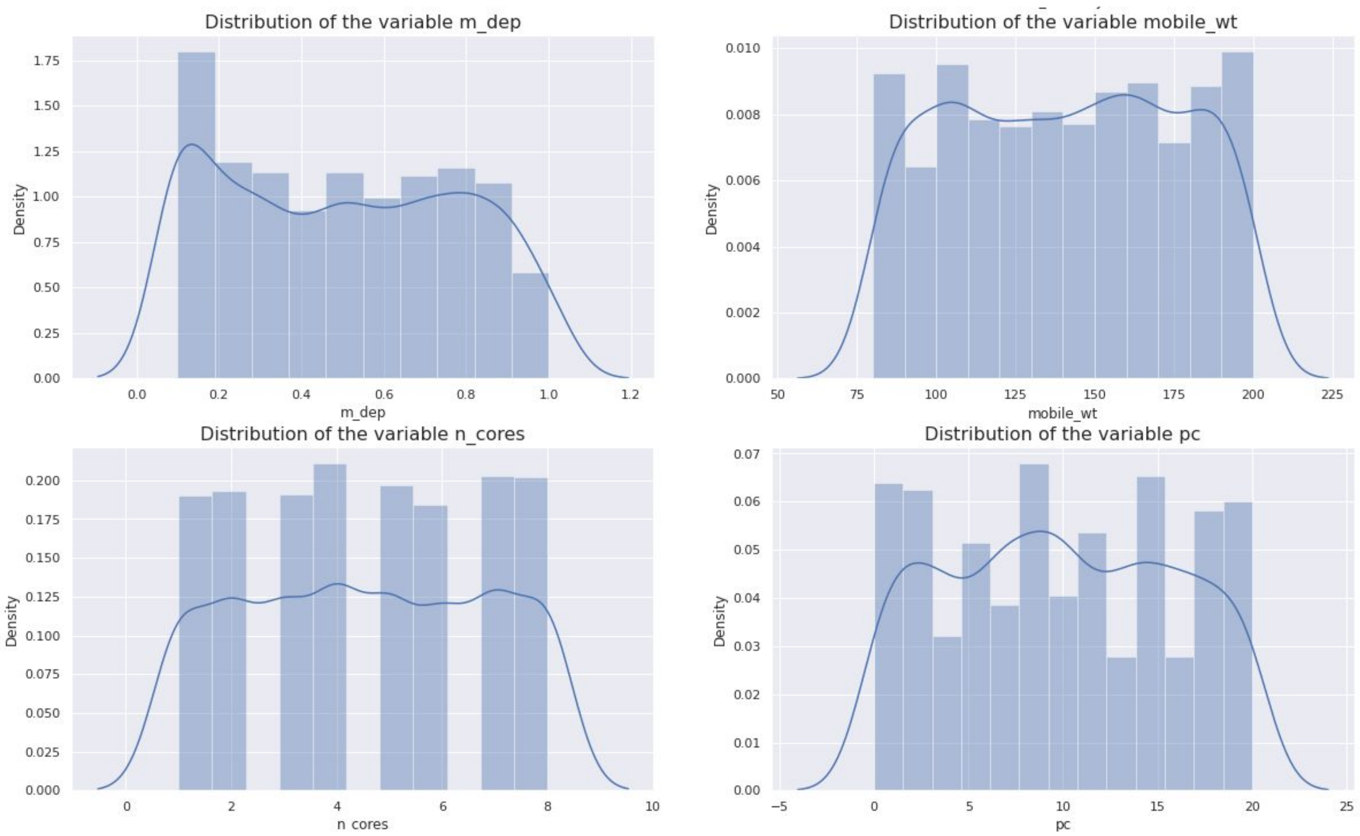
Low Medium High Very High



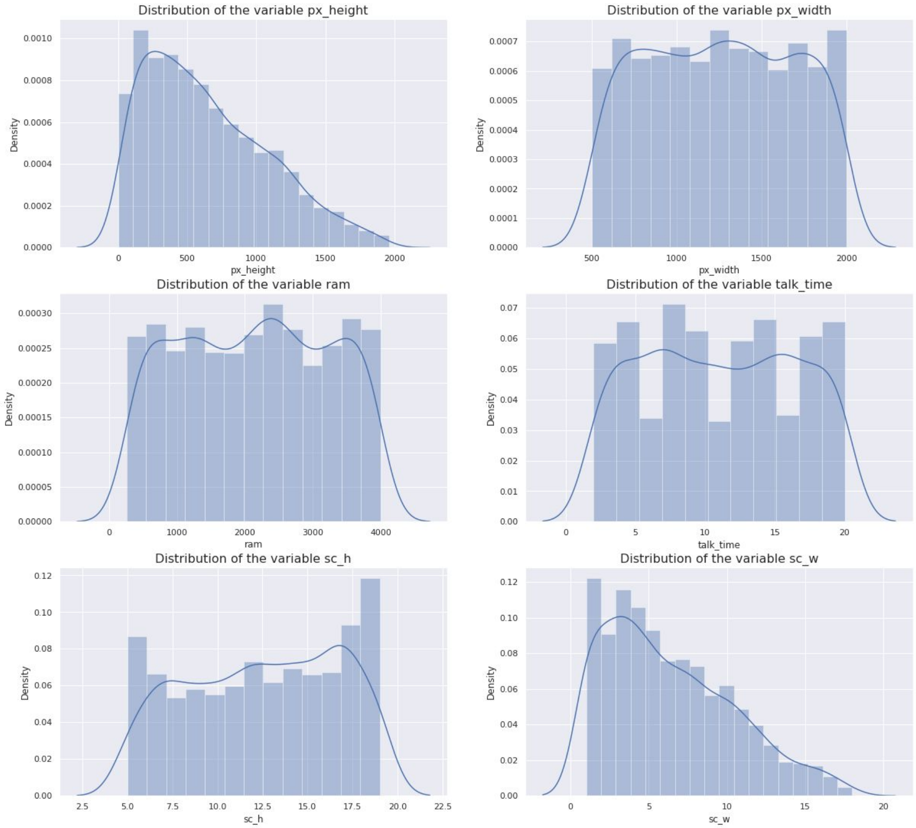
Exploratory Data Analysis



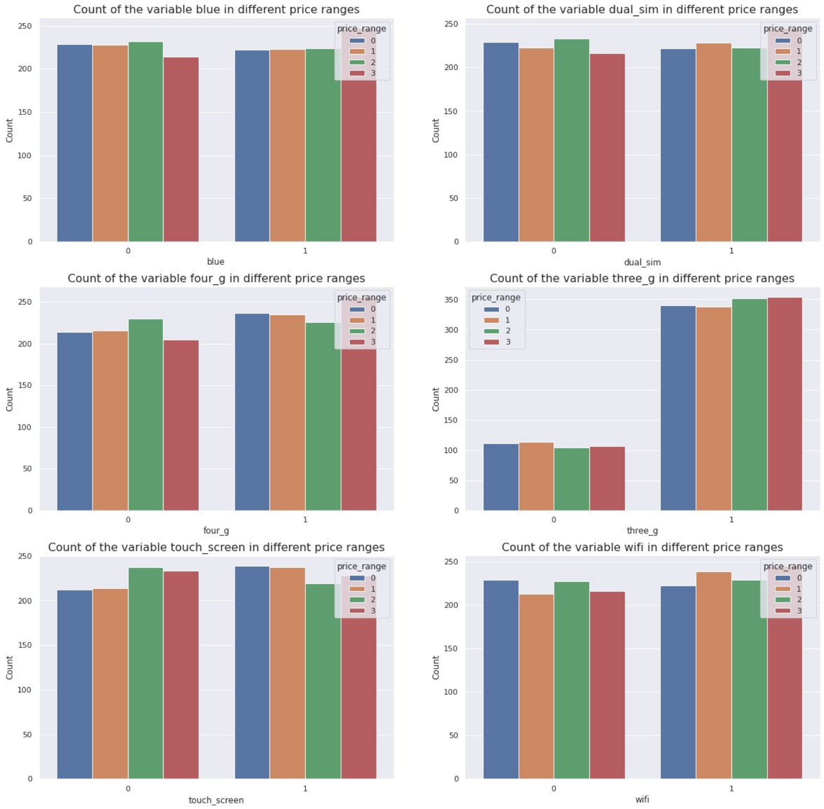
Exploratory Data Analysis



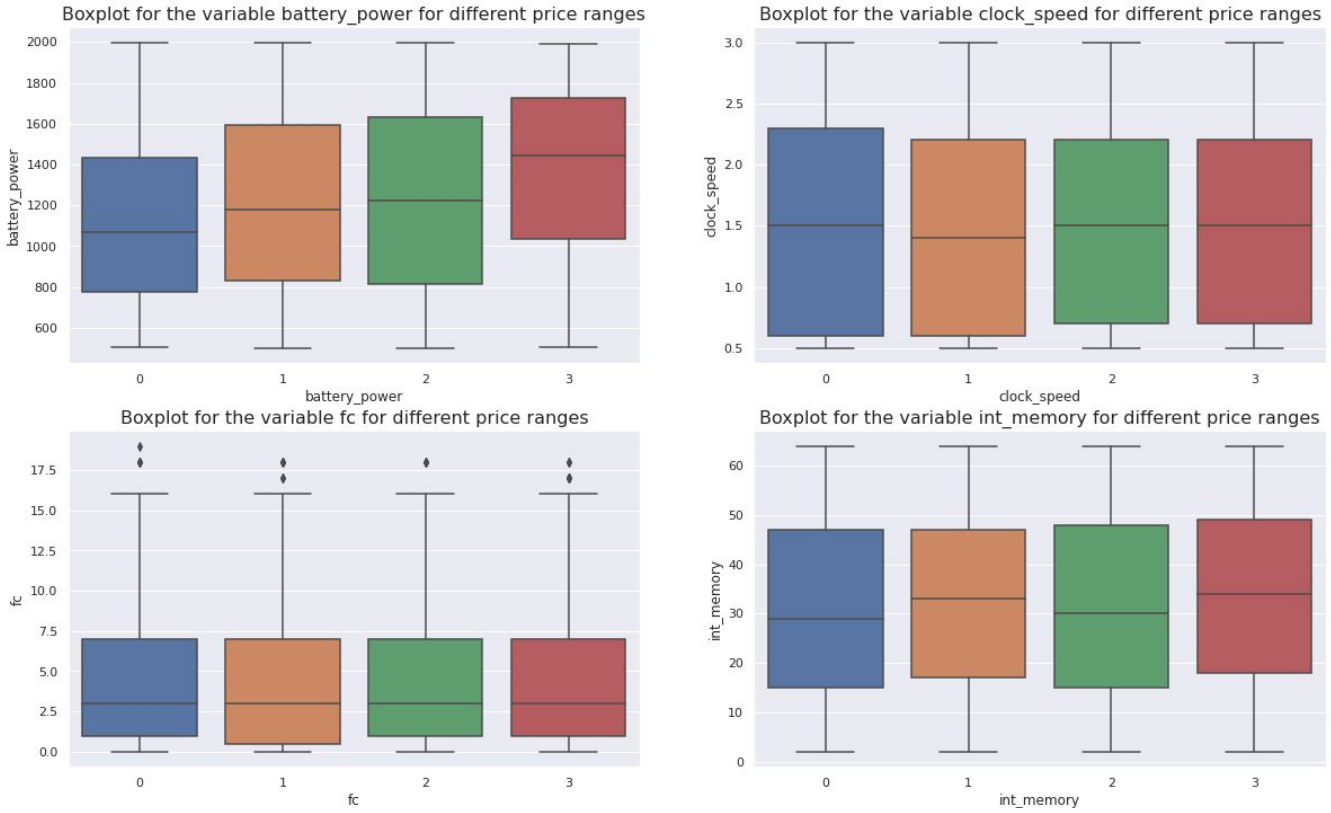
Exploratory Data Analysis



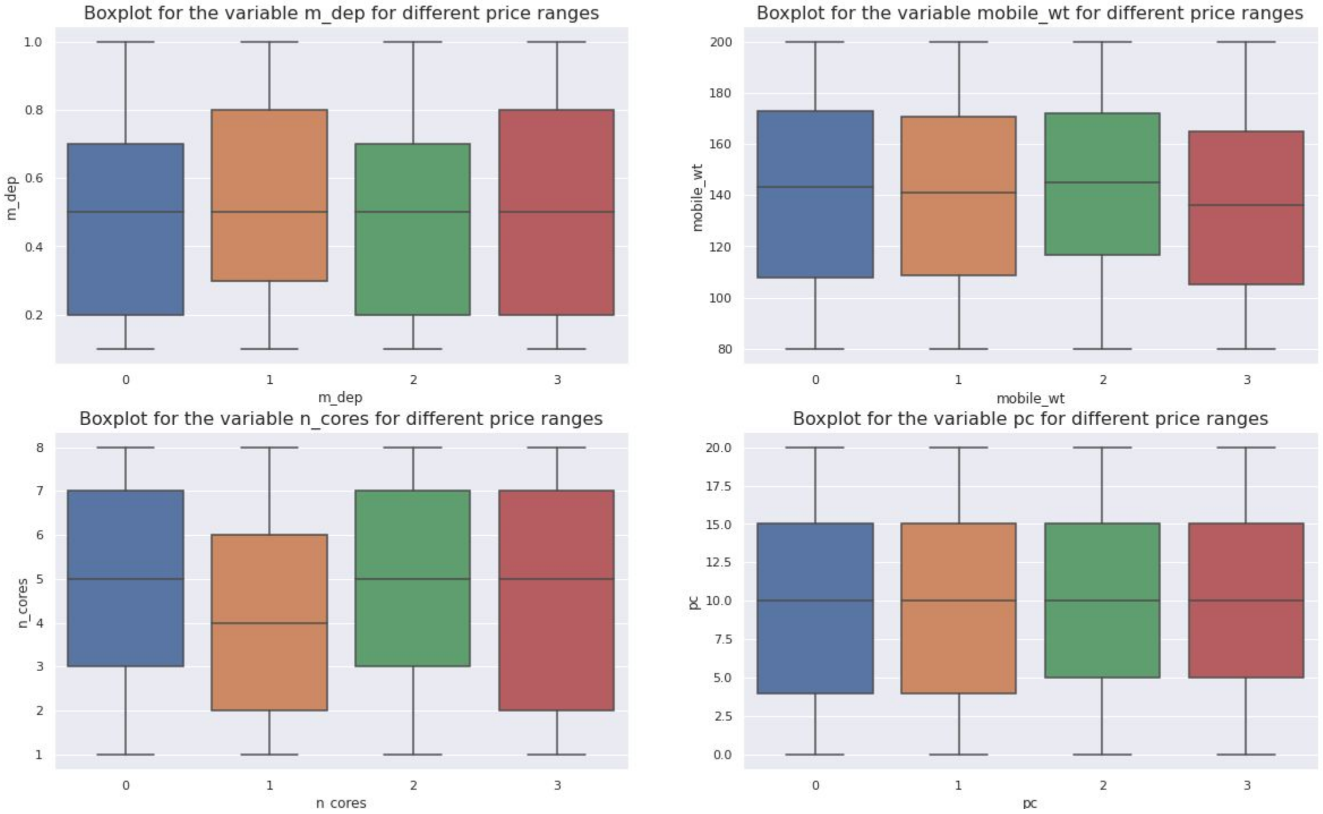
Exploratory Data Analysis



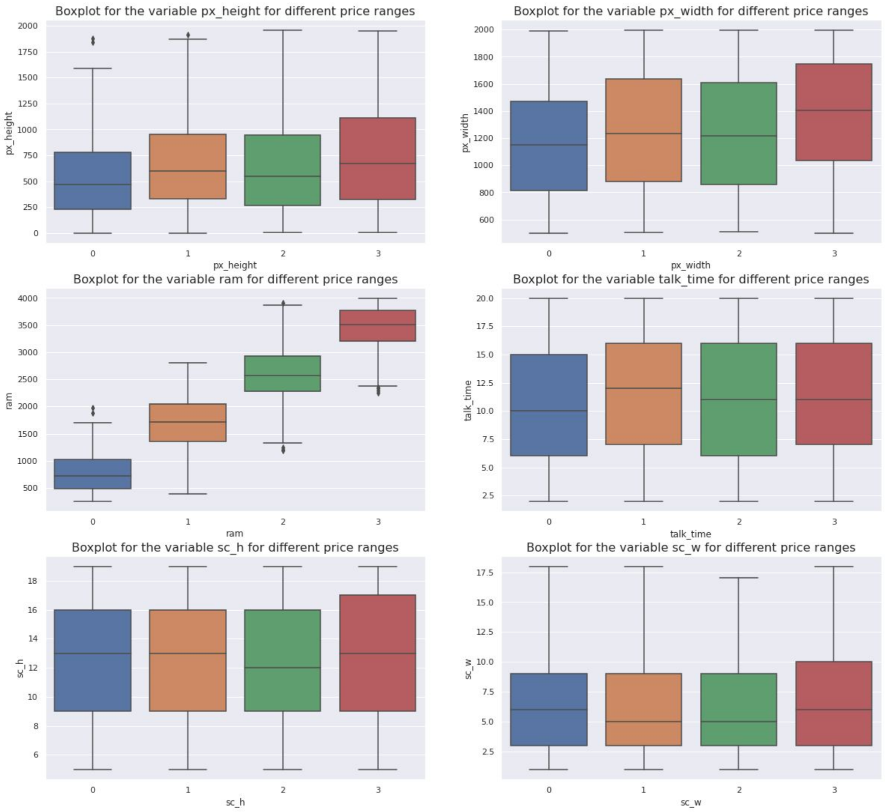
Exploratory Data Analysis



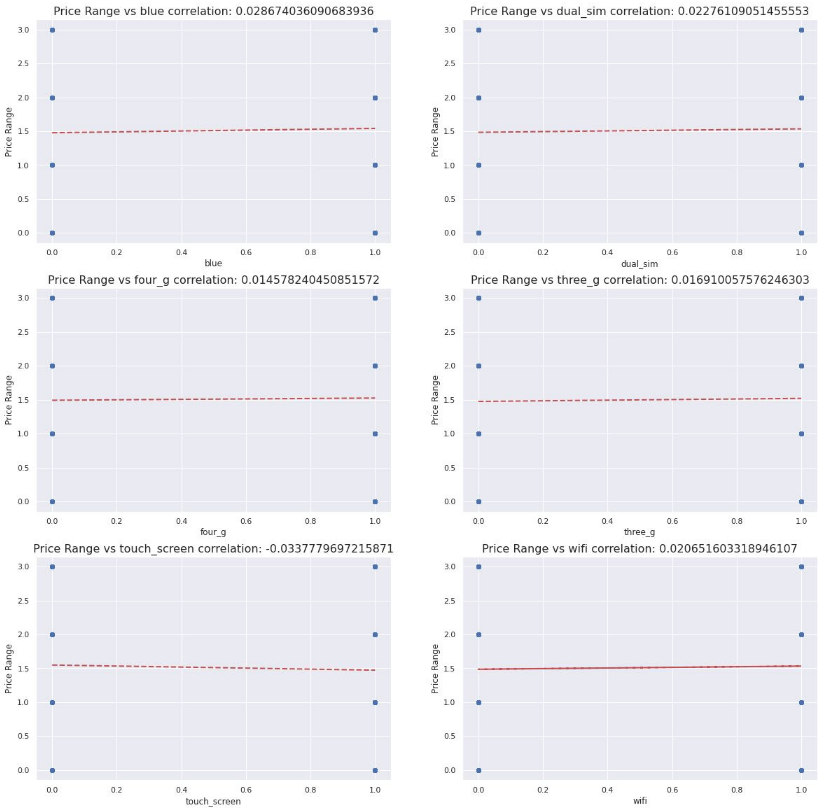
Exploratory Data Analysis



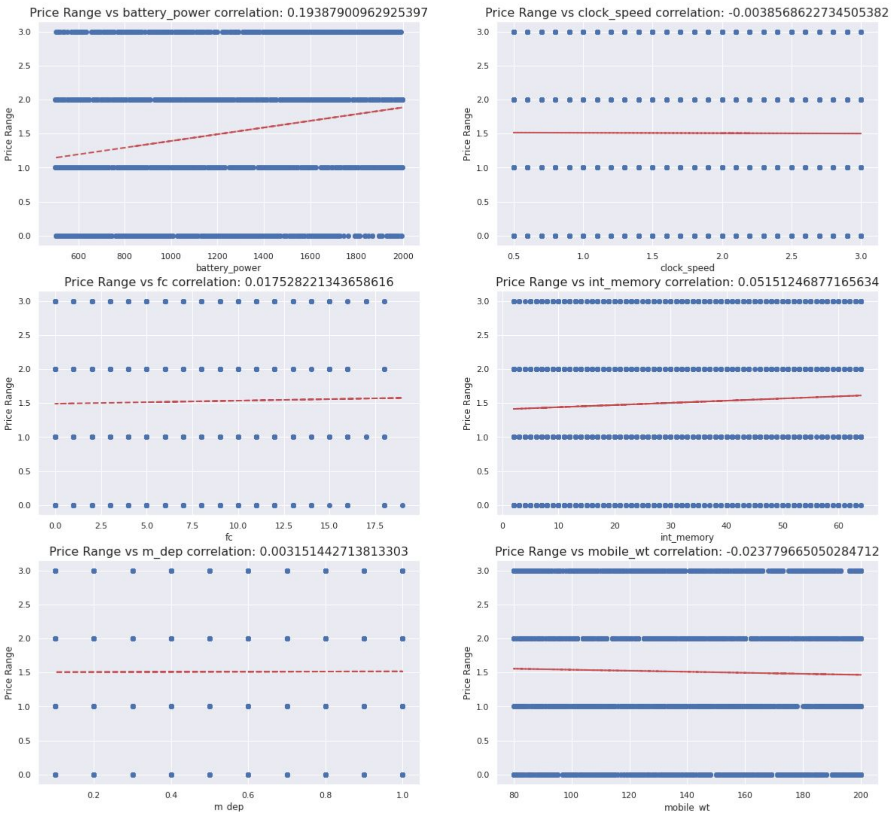
Exploratory Data Analysis



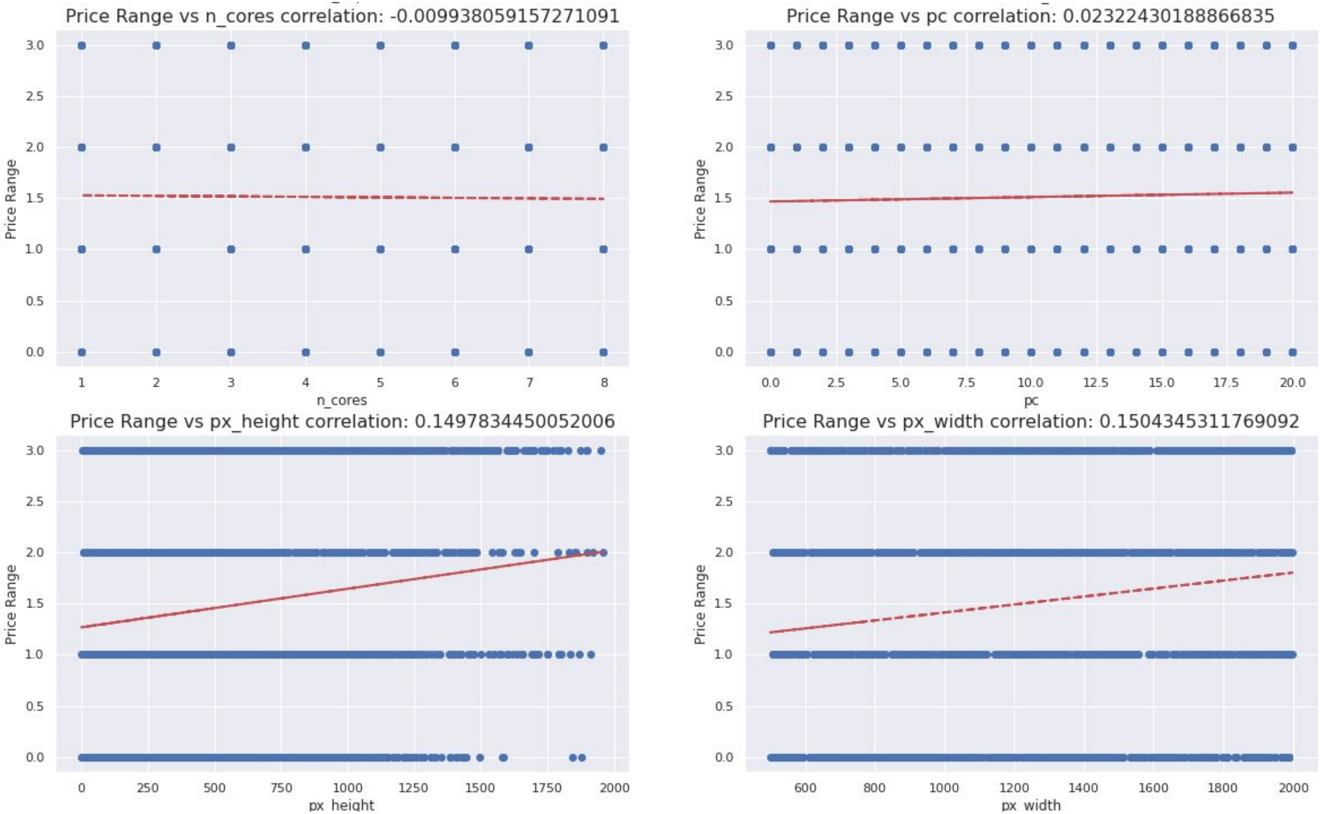
Correlation Analysis



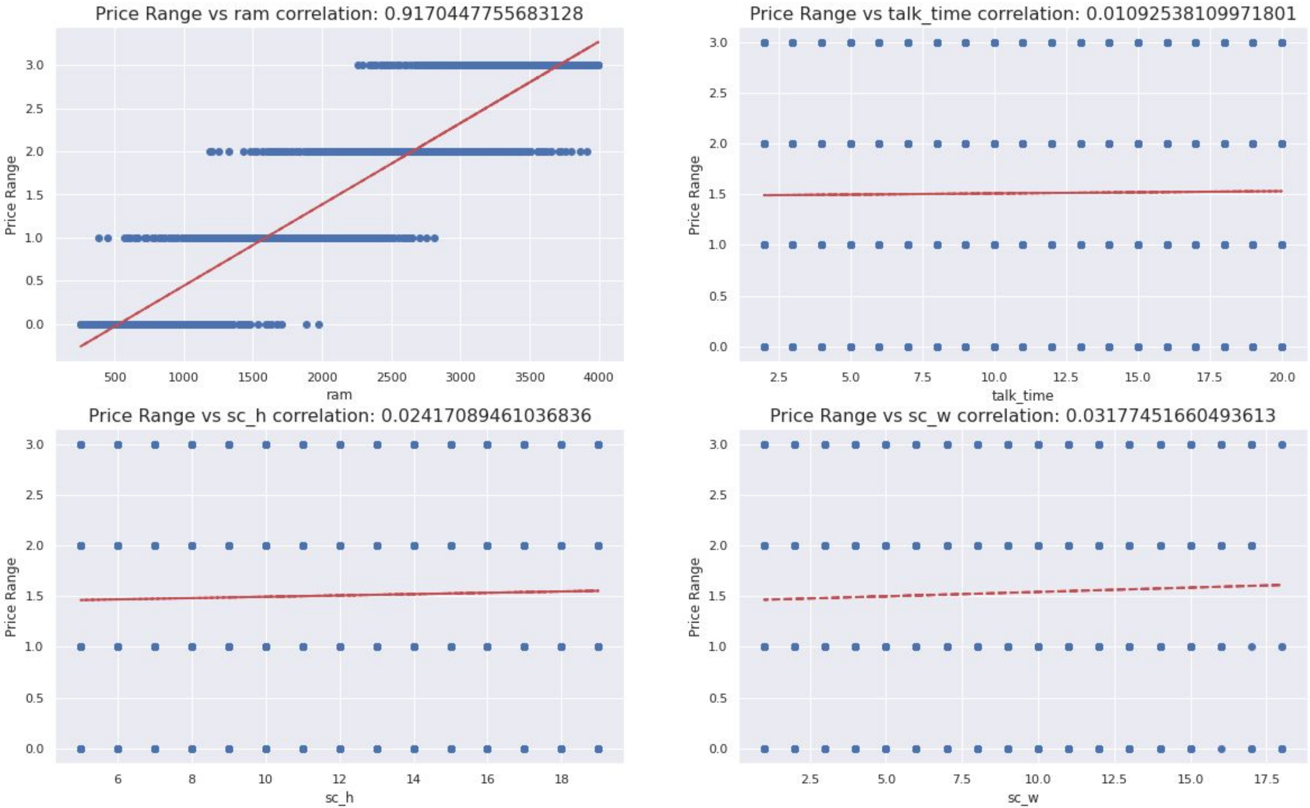
Correlation Analysis



Correlation Analysis



Correlation Analysis



Exploratory Data Analysis

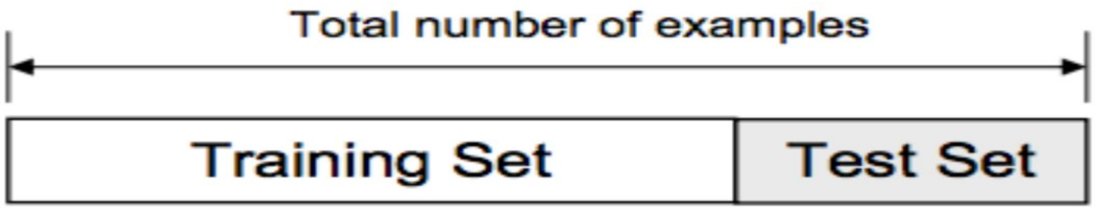
With the help of EDA, we can conclude that:

- The target classes are almost balanced, so there’s no class imbalance problem.
- Distribution of the categorical features is similar except three_g where there are very few records for mobiles which doesn’t have 3G access. The story remains the same when we break it down for different price ranges.
- Most of the numerical variables follow an uniform distribution except a few which are right skewed.
- RAM has the strongest correlation with the target variable followed by battery power, px_height and px_width.
- No categorical feature is strongly correlated with the target variable.
- There’s no pair of independent variables which are strongly correlated to each other, thus we don’t need to worry about multicollinearity.

Data Preparation

Train-Test Split:

- The train test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model.
- Cross validation has been used here while applying many algorithms. It is a resampling method that uses different portions of the data to test and train a model on different iterations.



Data Transformation

Feature Scaling:

- Machine learning algorithms like linear regression, logistic regression, etc that use gradient descent as an optimization technique require data to be scaled. The difference in ranges of features will cause different step sizes for each feature which will make it difficult for gradient descent to move towards minima.
- Standardization is used here where the values are centered around the mean with a unit standard deviation.

$$x_{\text{stand}} = \frac{x - \text{mean}(x)}{\text{standard deviation } (x)}$$

Evaluation Metrics

		Real Label	
		Positive	Negative
Predicted Label	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

$$\text{Precision} = \frac{\sum \text{TP}}{\sum \text{TP} + \text{FP}}$$

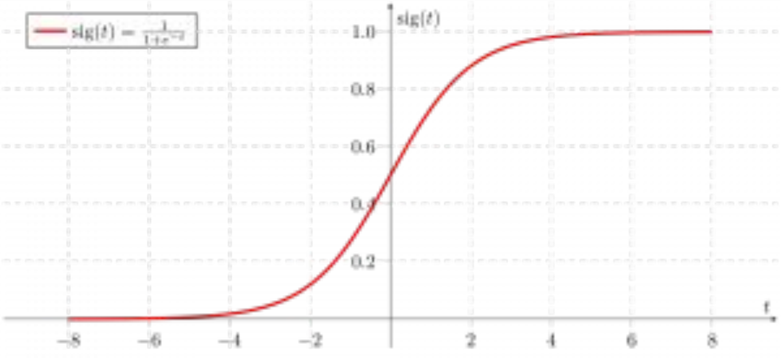
$$\text{Recall} = \frac{\sum \text{TP}}{\sum \text{TP} + \text{FN}}$$

$$\text{Accuracy} = \frac{\sum \text{TP} + \text{TN}}{\sum \text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Classification Models

Logistic Regression:

- It is a process of modeling the probability of a discrete outcome given an input variable.
- The most common logistic regression models a binary outcome, something that can take two values such as true/false, yes/no and so on.



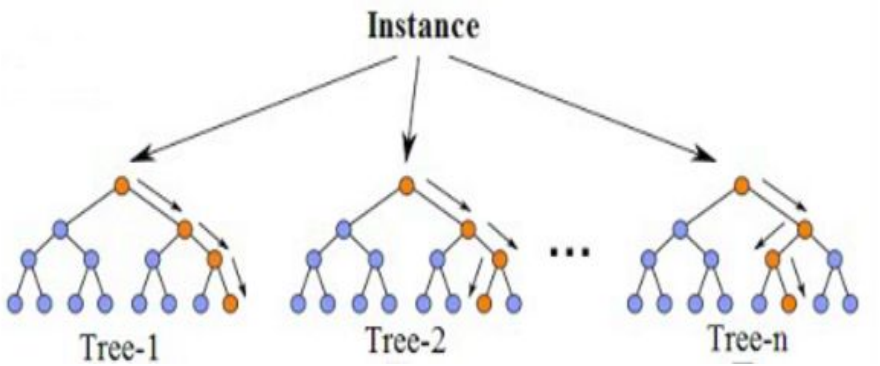
Accuracy	Precision	Recall	ROC AUC
0.9643	0.9645	0.9643	0.9975

Classification Models

Random Forest:

- Decision trees are great for obtaining non-linear relationships between input features and the target variable. The inner working of a decision tree can be thought of as a bunch of if-else conditions.
- Random forest is an ensemble of decision trees constructed in a certain random way.
- It randomly selects observations, builds a decision tree and the majority class is taken as output. It doesn't use any set of formulas.

Accuracy	Precision	Recall	ROC AUC
0.8956	0.8958	0.8956	0.9888

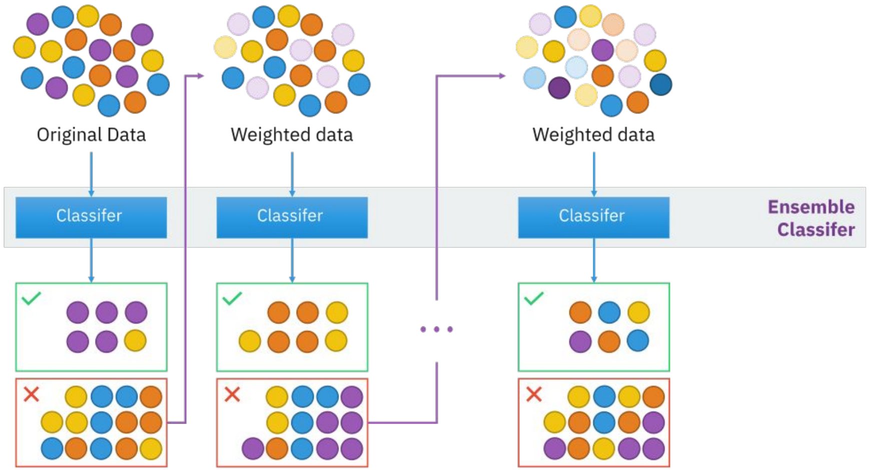


Classification Models

Gradient Boosting:

- Gradient boosting is one of the variants of ensemble methods where we create multiple weak models and combine them to get better performance as a whole.

Accuracy	Precision	Recall	ROC AUC
0.8928	0.8924	0.8928	0.9879

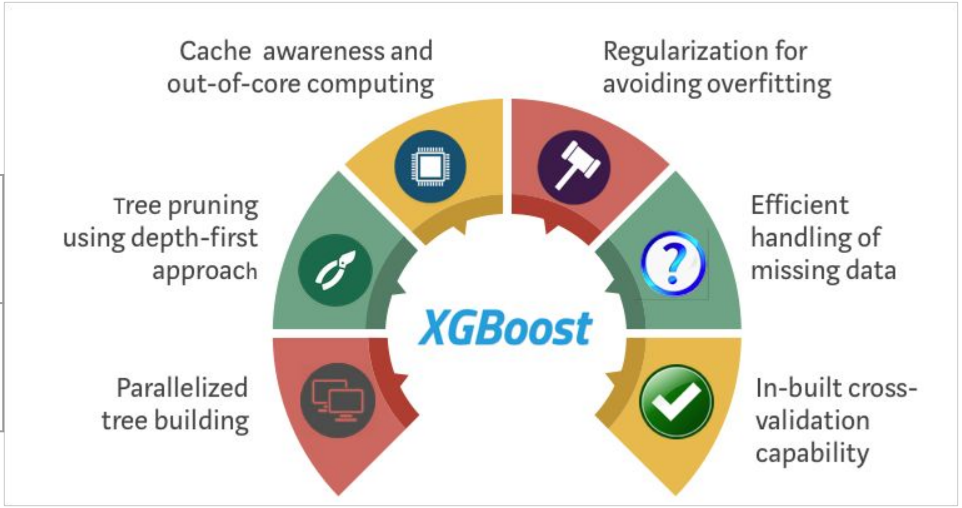


Classification Models

XGBoost:

- It is a scalable and highly accurate implementation of gradient boosting that pushes the limits of computing power for boosted tree algorithms.

Accuracy	Precision	Recall	ROC AUC
0.9066	0.9075	0.9066	0.9909

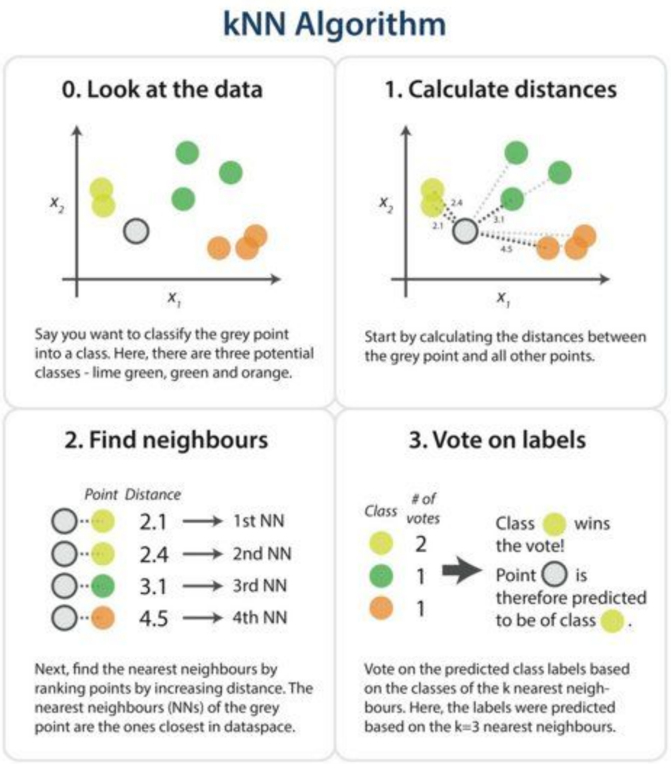


Classification Models

K-Nearest Neighbors:

- KNN assumes the similarity between the new data and available data and puts the new data into the category that is most similar to the available categories.
- It is a non-parametric and a lazy learning algorithm.

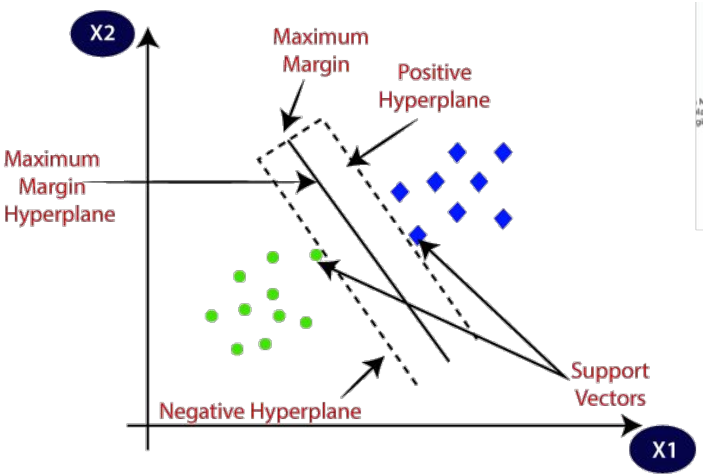
Accuracy	Precision	Recall	ROC AUC
0.9176	0.9176	0.9176	0.9775



Classification Models

Support Vector Machines:

- The objective of SVM is to find a hyperplane in an N-dimensional space that distinctly classifies the data points.
- There may be many possible hyperplanes but the objective is to find a plane that has the maximum margin.



Accuracy	Precision	Recall	ROC AUC
0.9615	0.9616	0.9615	0.9987

Model Performance & Comparison

Models	Accuracy	Precision	Recall	ROC AUC
Logistic Regression	0.9643	0.9645	0.9643	0.9975
SVM	0.9615	0.9616	0.9615	0.9987
KNN	0.9176	0.9176	0.9176	0.9775
XGBoost	0.9066	0.9075	0.9066	0.9909
Random Forest	0.8956	0.8958	0.8956	0.9888
Gradient Boosting	0.8928	0.8924	0.8928	0.9879

Logistic Regression has performed the best followed by SVM. The tree based methods have performed poorly in our case.

Conclusion

Let us end the presentation by summarizing few of the important insights we discovered from the project:

- The target classes are almost balanced thus we can use accuracy to compare our models.
- Distribution of all categorical features are similar except the feature ‘three_g’.
- Most of the numerical features follows an uniform distribution.
- RAM, battery power, px_height and px_width increase with price range, thus these features will be the most influential in determining or predicting the price ranges.
- No categorical feature is strongly correlated with price range.
- We have used 6 classification models and Logistic Regression has performed the best in terms of accuracy, precision, recall and roc auc score followed by SVM.
- All the tree based models has performed poorly in comparison with Logistic Regression, SVM and KNN.
- All the models have produced a good accuracy for predicting the price ranges.

Thank You !