# Capstone Project-3 Mobile Price Range Prediction

**Team Members** 

Kedar Vasantrao Patil Rohit Rajendra Pawar

# **Contents:**

- 1. Introduction
- 2. Problem statement
- 3. EDA
- 4. Feature engineering
- 5. Machine learning models
- 6. Model validation
- 7. Model explainability
- 8. Conclusion



# **Need for Mobile Range Selection?**

- Mobile is a very competitive market segment that cannot be easily captured. Margin in the technology used is less, which is compensated by the volumes sold.
- Average mobile sold in a year are around 1500 millions, which will keep growing in the coming years as economic sector improves more and more people will be able to afford mobile phone
- Looking at the above two points we see the there is a need for segregating mobile phones into various price categories.
- Now for a company that want to develop a mobile with some specifications and want to know the budget for each unit that will be made.
- This is the place where this model will come into clutch.



# Introduction

- Mobile phones are now like an essential commodity for us to connect to the world or our loved ones, resulting in the huge amount of mobile phone manufactured, hence; huge amount of data being generated.
- Mobile phone prediction helps in deciding the range of a mobile phone depending upon its specifications, as the most expensive mobile phone will be loaded with a lot more and better features than the cheap ones.
- This insight can help deciding the specification for a mobile phone at industry level.

### JioPhone Next

finally launched in India!

### SPECIFICATIONS

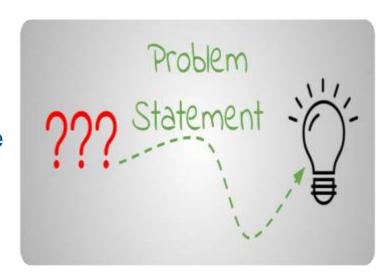
- 5.45-inch HD+ display
- Snapdragon 215 SoC
- 13MP (rear) + 8MP (front) cameras
- 3,500mAh battery
- 2GB RAM + 32GB storage lexpandable up to 512GB)





## **Problem Statement**

- In this 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.



# **Description of the Dataset**

- 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 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**

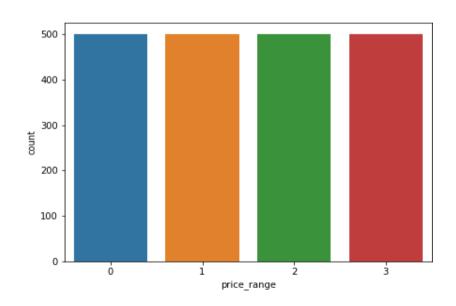
- Outlier Detection: No outlier found in categorical as well as numerical features.
- No Null values found
- No data Transformation needed.

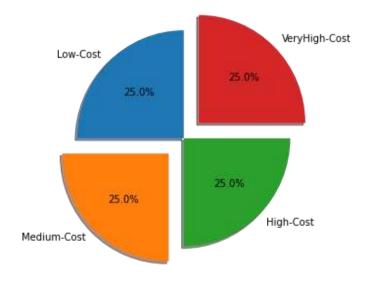


# **EXPLORATORY** DATA **ANALYSIS**

# **Analyzing Price Range**

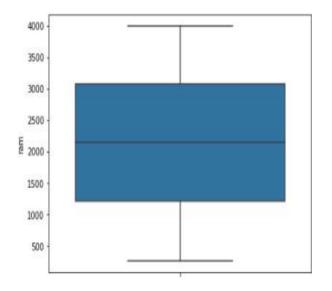
• We have a balanced dependent variable, where we have 500 mobiles lying in each category (0,1,2,3).

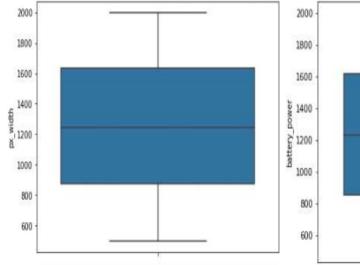


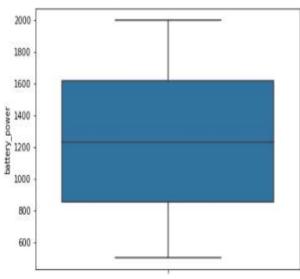


# **Check for Outliers**

 No problem related to outliers were faced, it was a very well balanced dataset. Below are the most important features and as we can clearly see, there are no outliers.





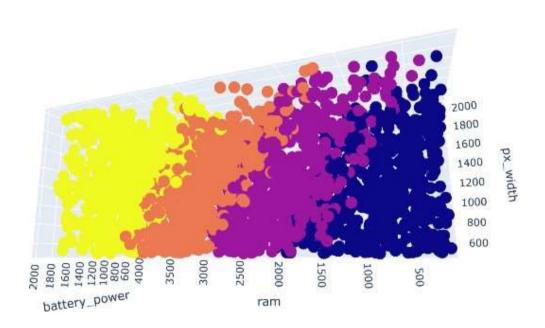


# **Multicollinearity**

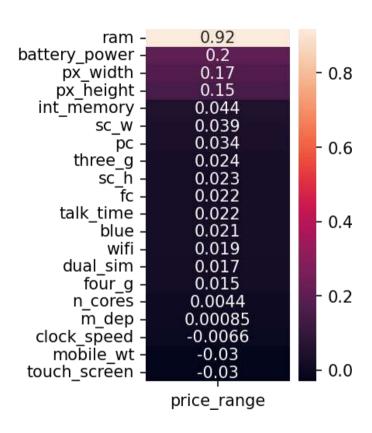
		3																			
nattery_power	1	0.011	0.011	-0.042	0.033	0.016	-0.004	0.034	0.0018	-0.03	0.031	0.015	-0.0084	0.00065	-0 03	-0.021	0.053	0.012	-0.011	-0.0083	0.2
blue	- 0.011	1	0.021	0.035	0.0036	0.013	0.041	0.004	-0.0086	0.036	-0.01	-0.0069	-0.042	0.026	-0.003	0.00061	0.014	-0.03	0.01	-0.022	0.021
dock_speed	- 0.011	0.021	1	-0.0013	-0.00043	-0.043	0.0065	0.014	0.012	0.0057	-0.0052	-0.015	-0.0095	0.0034	0.029	-0.0074	-0.011	-0.046	0.02	-0.024	-0.0064
dual_sim	0.042	0.035	-0.0013	1	-0.029	0.0032	-0.016	-0.022	-0.009	-0.025	-0.017	-0.021	0.014	0.041	-0.012	-0.017	-0.039	-0.014	-0.017	0.023	0.017
fc	- 0.033	0.0036	-0.00043	-0.029	1	-0.017	-0.029	-0.0018	0.024	-0.013	0.64	-0.01	-0.0052	0.015	-0.011	-0.012	-0.0068	0.0018	-0.015	0.02	0.022
four_g	- 0.016	0.013	-0.043	0.0032	-0.017	1	0.0087	-0.0018	-0.017	-0.03	-0.0056	-0.019	0.0074	0.0073	0.027	0.037	-0.047	0.58	0.017	-0.018	0.015
int_memory	0.004	0.041	0.0065	-0,016	-0.029	0.0087	1	0.0069	-0.034	-0,028	-0.033	0.01	-0.0083	0.033	0.038	0.012	-0.0028	-0.0094	-0.027	0.007	0.044
m_dep	- 0.034	0.004	-0.014	-0.022	-0.0018	0.0018	0.0069	1	0.022	-0.0035	0.026	0.025	0.024	-0.0094	-0.025	-0.018	0.017	-0.012	-0.0026	-0.028	0.0008
mobile_wt	- 0.0018	-0.0086	0.012	-0.009	0.024	-0.017	-0.034	0.022	1	-0.019	0.019	0.00094	9e-05	-0.0026	-0.034	-0.021	0.0062	0.0016	-0.014	-0:00041	-0.03
n_cores	0.03	0.036	-0.0057	-0.025	-0.013	-0.03	-0.028	-0.0035	-0.019	1	-0.0012	-0.0069	0.024	0.0049	0.00031	0.026	0.013	-0.015	0.024	-0.01	0.004
pc	- 0.031	-0.01	0.0052	-0.017	0.64	0.0056	-0.033	0.026	0.019	0.0012	1	-0.018	0.0042	0.029	0.0049	-0.024	0.015	-0.0013	0.0087	0.0054	0.034
px_height	- 0.015	-0.0069	-0.015	-0.021	-0.01	-0.019	0.01	0.025	0.00094	-0.0069	-0.018	1	0.51	-0.02	0.06	0.043	-0.011	-0.031	0.022	0.052	0.15
px_width	- 0.0084	0.042	-0.0095	0.014	0.0052	0.0074	-0.0083	0.024	9e-05	0.024	0.0042	0.51	1	0.0041	0.022	0.035	0.0067	0.00035	-0.0016	0.03	0.17
ram	-0.00065	0.026	0.0034	0.041	0.015	0.0073	0.033	-0.0094	-0.0026	0.0049	0.029	-0.02	0.0041	1	0.016	0.036	0.011	0.016	-0.03	0.023	0.92
sc_h	0.03	-0.003	-0.029	-0.012	-0.011	0.027	0.038	-0.025	-0.034	0.00031	0.0049	0.06	0.022	0.016	1	0.51	-0.017	0.012	-0.02	0.026	0.023
sc_w	0.021	0.00061	-0.0074	-0.017	-0.012	0.037	0.012	-0.018	-0.021	0.026	-0.024	0.043	0.035	0.036	0.51	1	-0.023	0.031	0.013	0.035	0.039
talk_time	- 0.053	0.014	-0.011	-0.039	0.0068	-0.047	-0.0028	0.017	0.0062	0.013	0.015	-0.011	0.0067	0.011	-0.017	-0.023	1	-0.043	0.017	-0.03	0.022
three_g	- 0.012	-0.03	-0.046	-0.014	0.0018	0.58	-0.0094	-0.012	0.0016	-0.015	-0.0013	-0.031	0.00035	0.016	0.012	0.031	-0,043	1	0.014	0.0043	0.024
touch_screen	0.011	0.01	0.02	-0.017	-0.015	0.017	-0.027	-0.0026	-0.014	0.024	-0.0087	0.022	-0.0016	-0.03	-0.02	0.013	0.017	0.014	1	0.012	-0.03
wifi	0.0083	-0.022	-0.024	0.023	0.02	0.018	0.007	-0.028	-0.00041	0.01	0.0054	0.052	0.03	0.023	0.026	0.035	-0.03	0.0043	0.012	1	0.019
price_range	0.2	0.021	-0.0066	0.017	0.022	0.015	0.044	0.00085	-0.03	0.0044	0.034	0.15	0.17	0.92	0.023	0.039	0.022	0.024	-0.03	0.019	1
	ttery power -	pho-	paads yoop	dual sim-	¥	four g	int memory -	m deb -	mobile wt -	n_cores -	ĸ.	px_height-	px_width	EE.	×,	W 25	talk time	three g -	ouch screen -	- unfi	price_range -

-0.2

# Multicollinearity (conti.)



3D correlation plot between ram, px\_width & battery\_power.



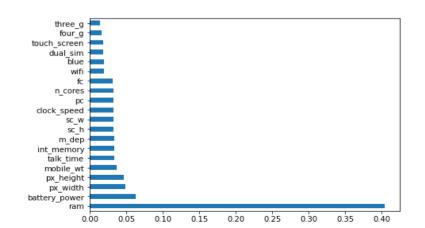
Ram is the most correlated to price i.e., it has the most impact on the price of a mobile

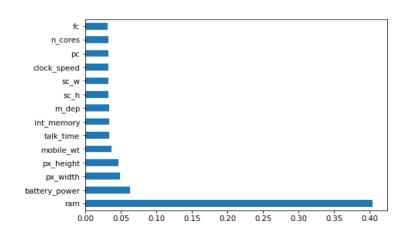
# **Feature Selection**

In Feature selection we removed non-informative features from the dataset. At beginning we have 2000 rows and 21 columns. After feature engineering we get 2000 rows and 14 columns.

### **Method for Feature Selection**

- We will use ExtraTreesClassifier from the sklearn library for Important Feature Selection.
- This class implements a meta estimator that fits a number of randomized decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over- fitting.
- RAM being the most influential and all the categorical columns being the least contributor to the price of a mobile.
- Thus; selecting top 14 features for further processes.



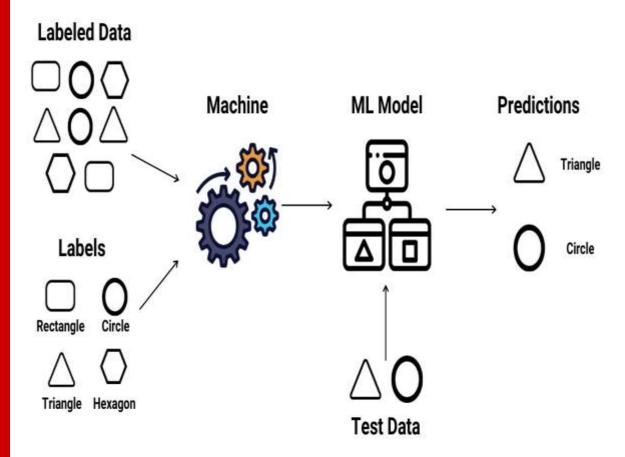




### Models Performed

- Logistic Regression
- Decision Tree
- Random Forest
- XGBoost
- K-Nearest Neighbour
- SVC
- Neural Network

# **Supervised Learning**



# **Applying Models - Validation & Selection**

	Model_Name	Training_accuracy	Test_accuracy	Train_roc_auc_score	Test_roc_auc_score
0	Logistic Regression	0.978	0.955	0.999	0.998
1	Decision Tree	0.941	0.855	0.996	0.944
2	KNN	0.731	0.635	0.907	0.868
3	Support Vector Machine	0.980	0.917	0.999	0.990
4	Neural-Network	0.983	0.975	0.999	0.998

We removed 'Random Forest' & 'XGBoost' from our model comparison because they were overfitting the data, also we had other models which working perfectly so that we don't need them in the comparison.

### **Confusion Matrix of the Models**

### **Logistic Regression**

### 

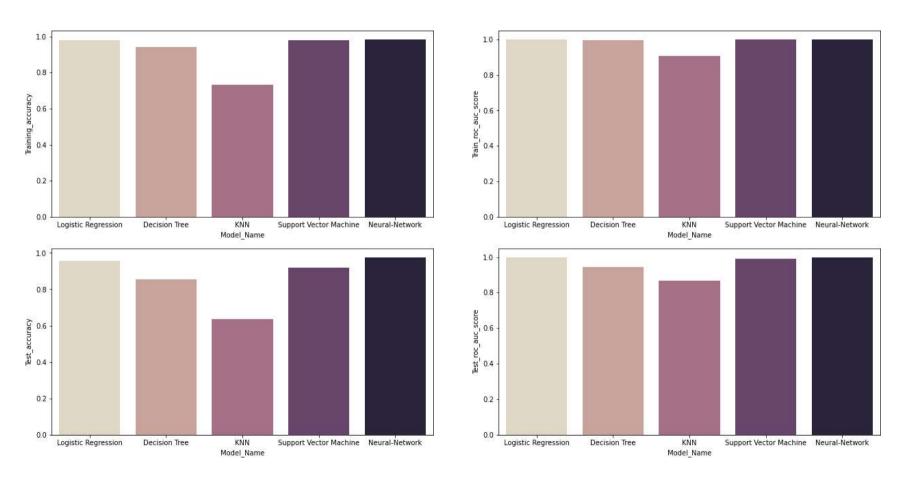
### **Decision Tree**

### **KNN**

### **SVC**

### **Neural Networks**

## **Model - Validation & Selection**



### **Model - Validation & Selection**

Observation 1: As observed from the table above Logistic Regression and decision tree model performs well along with support vector machine and neural network model.

Observation 2: Random forest & X-Gradient Boost model may overfit the model.

Observation 3: From the above observation we have come to a conclusion that we would choose our classification model from Neural Network.



# **Challenges**

- Being a balanced dataset, the difficulties were on a lower level.
- Some models overfitted the data.
- Feature selection.
- Model validation as most of the algorithm were performing almost similar.

# Conclusion

- As observed Logistic Regression and decision tree model performs well along with support vector machine and neural network model.
- Thus; for prediction we will go with our <u>Neural Network</u>
   <u>Model.</u>
- Ram, Battery power, Mobile weight, Screen size and Pixels are key features in predicting the mobile price range.

# THANK YOU