PROJECT REPORT ON

# ***MOBILE PRICE PREDICTION***

***RESEARCHER :*** AMIT KUMAR

***Presentation Title :*** Mobile Price Prediction

***School :*** Kalinga Institute of Industrial Technology

***Student Level :*** Bachelor's

***Presentation Type :*** Oral Presentation

Abstract : Classification model to predict whether price range of mobile based on certain specifications. The aim of this study is to develop a classification model to predict the price range of a mobile phone based on its specifications. The dataset used in this study contains information about mobile phones from various manufacturers, including features such as battery capacity, RAM, storage capacity, and camera quality, as well as their corresponding price ranges.

In order to predict the price range of a mobile phone, a machine learning approach is used, specifically a classification model. The data is first preprocessed, which includes handling missing values, encoding categorical variables, and scaling the numerical features. The preprocessed data is then split into training and testing sets, and a classification algorithm is trained on the training set.

Several classification algorithms are evaluated, including logistic regression, decision trees, and random forests, and their performance is compared using metrics such as accuracy, precision, recall, and F1 score. The best performing algorithm is then selected and used to make predictions on the testing set.

The results of the study show that a classification model can be successfully used to predict the price range of a mobile phone based on its specifications, achieving high accuracy and performance. This model could be useful for both consumers and manufacturers in the mobile phone industry to make informed decisions about pricing and purchasing.

## ***Introduction :***

Price always has a important impact factor in the product buying aspect and also in the mindset of the buyer who would consider "what is the worth and is it good to buy within this range". During any product launch into the market, there is a lot of variables and factors are considered and especially in mobiles many features and specification like memory is considered and also the impacting of the cost also may have impact with the

competition in the market place. In Mobile there are many specification and features like camera, video, quality of processor, quality of the material. There is many constraints in consideration of the price, as the product should be economical and There are many multiple variables to be considered to get the précised results of the price and other features. of the mobile dataset this will help the buyer and also the marketer and the developer to get precise information from historical data of mobile phones and help them to decide are fine and satisfactory reachable with overall consideration.

## PROBLEM STATEMENT

What are the things a potential phone buyer consider before purchasing  a phone - battery power , dual sim , primary mega pixels , ram ,  touchscreen or not , etc . But  the most important factor is the price of phone. Our project aims helps the buyer to get the estimate cost of the mobile phone she/he is looking for.

## TECHNOLOGY USED

- Python as Programming Language
- Implementation using Machine Learning
- Pandas for Data Cleaning
- Sklearn for Model Building

## FUNCTIONAL OVERVIEW

Below is the sample csv file screenshot.

| battery_power | blue | clock_speed | dual_sim | fc | four_g | int_memory | m_dep | mobile_wt | n_cores | pc | px_height | px_width | ram | sc_h | sc_w | talk_time | three_g | touch_screen | wifi | price_range |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 842 | 0 | 2.2 | 0 | 1 | 0 | 7 | 0.6 | 188 | 2 | 2 | 20 | 756 | 2549 | 9 | 7 | 19 | 0 | 0 | 1 | 1 |
| 1021 | 1 | 0.5 | 1 | 0 | 1 | 53 | 0.7 | 136 | 3 | 6 | 905 | 1988 | 2631 | 17 | 3 | 7 | 1 | 1 | 0 | 2 |
| 563 | 1 | 0.5 | 1 | 2 | 1 | 41 | 0.9 | 145 | 5 | 6 | 1263 | 1716 | 2603 | 11 | 2 | 9 | 1 | 1 | 0 | 2 |
| 615 | 1 | 2.5 | 0 | 0 | 0 | 10 | 0.8 | 131 | 6 | 9 | 1216 | 1786 | 2769 | 16 | 8 | 11 | 1 | 0 | 0 | 2 |
| 1821 | 1 | 1.2 | 0 | 13 | 1 | 44 | 0.6 | 141 | 2 | 14 | 1208 | 1212 | 1411 | 8 | 2 | 15 | 1 | 1 | 0 | 1 |
| 1859 | 0 | 0.5 | 1 | 3 | 0 | 22 | 0.7 | 164 | 1 | 7 | 1004 | 1654 | 1067 | 17 | 1 | 10 | 1 | 0 | 0 | 1 |
| 1821 | 0 | 1.7 | 0 | 4 | 1 | 10 | 0.8 | 139 | 8 | 10 | 381 | 1018 | 3220 | 13 | 8 | 18 | 1 | 0 | 1 | 3 |
| 1954 | 0 | 0.5 | 1 | 0 | 0 | 24 | 0.8 | 187 | 4 | 0 | 512 | 1149 | 700 | 16 | 3 | 5 | 1 | 1 | 1 | 0 |
| 1445 | 1 | 0.5 | 0 | 0 | 0 | 53 | 0.7 | 174 | 7 | 14 | 386 | 836 | 1099 | 17 | 1 | 20 | 1 | 0 | 0 | 0 |
| 509 | 1 | 0.6 | 1 | 2 | 1 | 9 | 0.1 | 93 | 5 | 15 | 1137 | 1224 | 513 | 19 | 10 | 12 | 1 | 0 | 0 | 0 |
| 769 | 1 | 2.9 | 1 | 0 | 0 | 9 | 0.1 | 182 | 5 | 1 | 248 | 874 | 3946 | 5 | 2 | 7 | 0 | 0 | 0 | 3 |
| 1520 | 1 | 2.2 | 0 | 5 | 1 | 33 | 0.5 | 177 | 8 | 18 | 151 | 1005 | 3826 | 14 | 9 | 13 | 1 | 1 | 1 | 3 |
| 1815 | 0 | 2.8 | 0 | 2 | 0 | 33 | 0.6 | 159 | 4 | 17 | 607 | 748 | 1482 | 18 | 0 | 2 | 1 | 0 | 0 | 1 |
| 803 | 1 | 2.1 | 0 | 7 | 0 | 17 | 1.0 | 198 | 4 | 11 | 344 | 1440 | 2680 | 7 | 1 | 4 | 1 | 0 | 1 | 2 |
| 1866 | 0 | 0.5 | 0 | 13 | 1 | 52 | 0.7 | 185 | 1 | 17 | 356 | 563 | 373 | 14 | 9 | 3 | 1 | 0 | 1 | 0 |
| 775 | 0 | 1.0 | 0 | 3 | 0 | 46 | 0.7 | 159 | 2 | 16 | 862 | 1864 | 568 | 17 | 15 | 11 | 1 | 1 | 1 | 0 |
| 838 | 0 | 0.5 | 0 | 1 | 1 | 13 | 0.1 | 196 | 8 | 4 | 984 | 1850 | 3554 | 10 | 9 | 19 | 1 | 0 | 1 | 3 |
| 595 | 0 | 0.9 | 1 | 7 | 1 | 23 | 0.1 | 121 | 3 | 17 | 441 | 810 | 3752 | 10 | 2 | 18 | 1 | 1 | 0 | 3 |
| 1131 | 1 | 0.5 | 1 | 11 | 0 | 49 | 0.6 | 101 | 5 | 18 | 658 | 878 | 1835 | 19 | 13 | 16 | 1 | 1 | 0 | 1 |
| 682 | 1 | 0.5 | 0 | 4 | 0 | 19 | 1.0 | 121 | 4 | 11 | 902 | 1064 | 2337 | 11 | 1 | 18 | 0 | 1 | 1 | 1 |

## *CHECK NULL VALUES OR NOT*

```
In [6]: df.info() # find out null values if any

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2000 entries, 0 to 1999
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   battery_power   2000 non-null   int64
 1   blue            2000 non-null   int64
 2   clock_speed     2000 non-null   float64
 3   dual_sim        2000 non-null   int64
 4   fc              2000 non-null   int64
 5   four_g          2000 non-null   int64
 6   int_memory      2000 non-null   int64
 7   m_dep           2000 non-null   float64
 8   mobile_wt       2000 non-null   int64
 9   n_cores         2000 non-null   int64
 10  pc              2000 non-null   int64
 11  px_height       2000 non-null   int64
 12  px_width        2000 non-null   int64
 13  ram             2000 non-null   int64
 14  sc_h            2000 non-null   int64
 15  sc_w            2000 non-null   int64
 16  talk_time       2000 non-null   int64
 17  three_g         2000 non-null   int64
 18  touch_screen    2000 non-null   int64
 19  wifi            2000 non-null   int64
 20  price_range     2000 non-null   int64
dtypes: float64(2), int64(19)
memory usage: 328.2 KB
```

## *RANDOM FOREST*

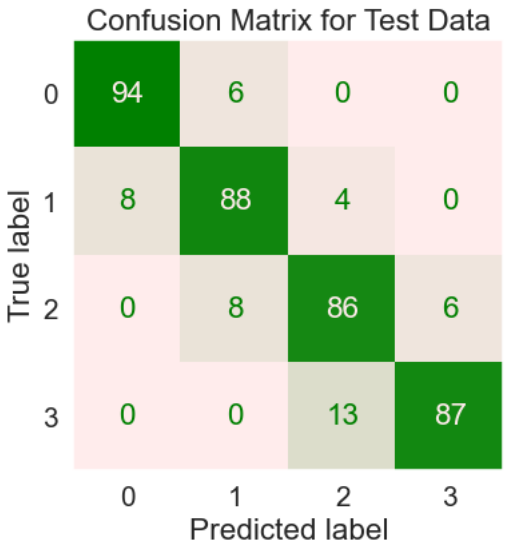
Random Forest Performance Summary on Test Data

| | Random Forest |
|---|---|
| Accuracy | 91.5% |
| Macro Precision | 91.57% |
| Macro Recall | 91.5% |
| Macro F1-score | 91.52% |

## *DECISION TREE*

Decision Tree Performance Summary on Test Data

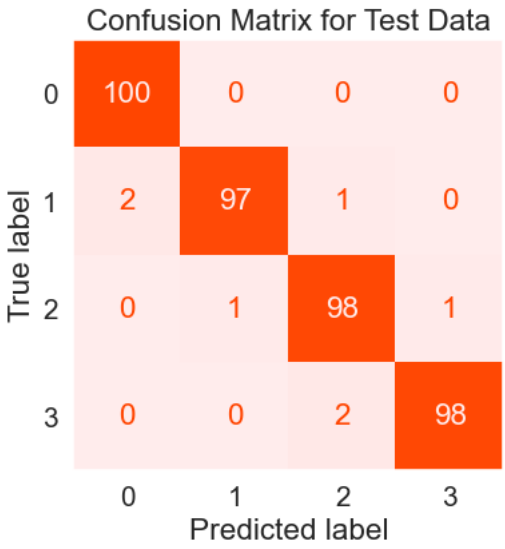
| | Decision Tree |
|---|---|
| Accuracy | 88.75% |
| Macro Precision | 88.87% |
| Macro Recall | 88.75% |
| Macro F1-score | 88.77% |

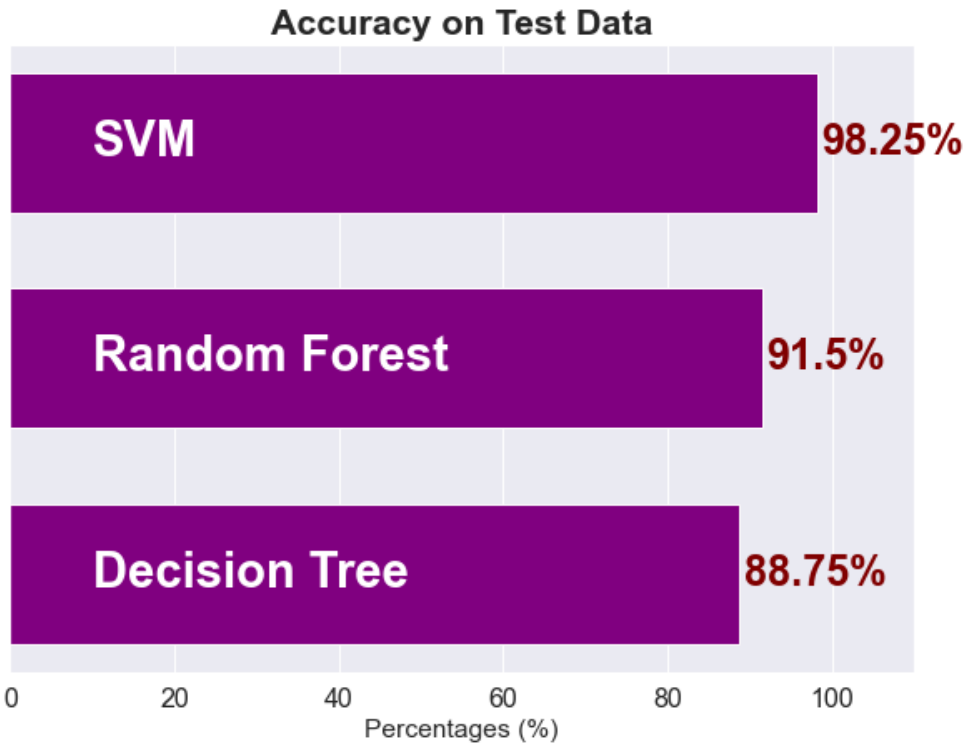



## *SVM PERFORMANCE*

SVM Performance Summary on Test Data

| | SVM |
|---|---|
| Accuracy | 98.25% |
| Macro Precision | 98.26% |
| Macro Recall | 98.25% |
| Macro F1-score | 98.25% |






**SVM has the best accuracy among the three.**

**CONCLUSION :**

This project will help people who plan to buy a mobile phone so that they can know the price range .
It will help the entrepreneur who has started his own mobile company to know about the price estimates.

**Future Work**

In order to improve the model performance certain changes can be made to our minor project such as changing the dataset used for training and testing as a larger dataset will help us consider a longer timeframe and analyse trends better, also it will help the model learn more about our

data and it will lead to better model performance.