Smart Mobile Phone Price Prediction Using Machine Learning Algorithms

Shreyansh Rai
3rd Year BTech Undergraduate, Dept of Information and Communication Technology
Manipal Institute of Technology, Udupi, India
shreyansh.rai@learner.manipal.edu https://www.linkedin.com/in/shreyansh-rai/

***Abstract*— This research paper addresses the problem of predicting mobile phone prices using machine learning techniques. The study utilizes a dataset obtained through web scraping using Python's Beautiful Soup and Selenium libraries. The dataset includes features such as Brand, Launch Date, Memory, Camera and many more. Various machine learning models, including Linear Regression, Support Vector Regression, Decision Tree, Random Forest, Gradient Boosting, LightGBM, XGBoost, CatBoost, HistGradientBoost, Bagging, and K-nearest neighbors, are trained and evaluated using GridSearch and RandomSearch for hyperparameter tuning.**

The best performing model is XGBoost, trained on a log-transformed and scaled dataframe. The evaluation metrics used include Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared (R2). The XGBoost model achieves an MSE of 0.1139, RMSE of 0.3375, MAE of 0.2518, and R2 of 0.885, indicating high predictive accuracy.

The findings of this study provide valuable insights for both mobile phone buyers and sellers. Buyers can make informed decisions by leveraging the predictive model to estimate future mobile phone prices, while sellers can appropriately set prices based on the features of the smartphones. The research contributes to the fields of information technology and machine learning, demonstrating the effectiveness of XGBoost for mobile phone price prediction.

I. LITERATURE REVIEW

The researchers in [1] employed 4 ML models namely, DT, SVM, NB, KNN to predict the classes of smart phones and they have found that SVM worked the best among these 4 using Accuracy and F1 score as evaluation metrics. They also found that the utilization of feature selection has increased both Accuracy and F1 score by almost 1 to 2% in comparison to when they have not used feature selection. This seems to be a running theme in other research papers related to this problem statement as well. So, feature selection plays an important part in obtaining better results from the models. They have observed that brand loyalty plays an important role in customer behavior meaning that customers sometimes will be willing to buy a smart phone from a brand they like even though it costs a bit more compared to other brands they aren't loyal to.

This regression problem is converted into a classification problem in [2] where the authors have first assigned classes to all the rows in the dataset based on the price and then trained Decision Tree and Naïve Bayes classifiers to classify the price based on the features. They have used many techniques of dimensionality reduction such as forward selection, backward selection, feature selection to extract only useful features. They used only Accuracy alone as an evaluation metric which might be problematic. F1 score should've also been considered.

The paper [3] almost entirely uses the same techniques and methodology as [2] with the exception that the authors have used a few new models like ZeroR and J48 Decision Tree. Interestingly, both [2] and [3] used the same software called WEKA for implementing the models. According to their study, J48 DT gave the highest accuracy.

The authors in [4] have also converted this regression problem into a classification problem and used KNN and logistic regression models to classify the dataset. They observed that logistic regression gave higher accuracy. They haven't used any other evaluation metric. One important conclusion that they mentioned in their paper is that they realized that conversion of the regression problem into classification problem have introduced more error into their work.

The researchers in [5] have followed along in the footsteps of [2] and have used similar models and evaluation metrics therefore yielding similar results. One important addition however is that they have developed an interactive web application that takes in the values of various features as input and gives the price class as output. They suggested future enhancements like the usage of better ML and DL algorithms and the picking of more relevant characteristics to improve accuracy.

II. DATA PREPROCESSING AND EXPLORATORY DATA ANALYSIS

A. Description of the dataset

The dataset used for this research project comprises scraped data from the website Pricebaba.com. The data was obtained using web scraping techniques with Beautiful Soup and Selenium. The purpose of web scraping was to gather the most up-to-date data with detailed features not readily available in existing online datasets. Specifically, data was collected for mobile phones from popular brands, ranging from Rs. 10,000 to Rs. 1,00,000. A total of 710 phones' data was scraped and stored in a CSV format.

The dataset consists of 30 columns, including key information such as Brand, Model, Price, Launch Date, Operating System (OpSys), Weight, Screen Size, Resolution, Pixels Per Inch (PPI), Screen Type, Refresh Rate, Processor, CPU, RAM, Memory, Number of Cameras (NoofCam), Main Camera Megapixels (MainCamMP), Main Camera Video capabilities (MainCamVideo), Front Camera Megapixels (FrontCamMP), Battery Capacity, Wireless Charging support, Charging Type, Fingerprint Sensor availability, Network compatibility, and SIM type. The data was informative but noisy and unclean and needed preprocessing.

B. Data Cleaning

The dataset is very noisy containing a lot of missing values and outliers. So, we will first clean and preprocess the data.

- The 'Model' column is dropped.
- The numeric value in Ruppes from column 'Price' is extracted.
- The 'Launch Date' column is converted to 'Age' in months calculating as per June 2023.
- From 'OpSys' column, the version number is extracted as 'OpSys_version' as the Operating System will be "iOS" if Brand is "Apple"
- The numeric value in grams is extracted from column 'Weight'.
- The numeric value in inches is extracted from column 'Screen Size'

- The ‘Resolution’ column is split into two columns ‘Resolution X’ and ‘Resolution Y’ with numerical values in pixels.
- The numerical value in pixels per inch is extracted from column ‘PPI’
- The values in ‘Screen Type’ column are converted into either AMOLED, LCD or OLED .
- The numerical values in Hertz is extracted from column ‘Refresh Rate’.
- The ‘Processor’ column having names of processor is dropped and from ‘CPU’ column the numeric value of maximum speed in Gigahertz is extracted and renamed as ‘MaxSpeed’.
- The numeric values in Gigabytes is extracted from ‘RAM’ column.
- The numeric values in Gigabytes is extracted from ‘Memory’ column.
- Categorical values like “Single”, “Double”, “Triple” etc. are converted to numerical values in ‘NoOfCam’ column
- The maximum numeric value in Megapixels is extracted from the Camera information in column ‘MainCamMP’ and ‘FrontCamMP’ denoting the maximum resolution of the cameras.
- The numeric value in mAh is extracted from the ‘Battery’ column.
- In ‘Wireless Ch’ column the categorical values of “Yes” and “No” are converted to binary format.
- ‘Ch Type’ column which tells whether fast charging is available or not is converted to binary column ‘Fast Charging’
- Similarly ‘Fingerprint’ column is converted to binary column showing the availability of fingerprint sensor.
- From the ‘Network’ column, as all phones had atleast 4G, two columns binary columns ‘4G’ and ‘5G’ were extracted.
- From the column ‘MainCamVideo’ column, having video camera capability information three numeric columns were extracted, ‘MaxVideoResolutionWidth’, ‘MaxVideoResolutionHeight’ and ‘MaxFrameRate’.

C. Exploratory Data Analysis

- Now we will see the basic shape of the data after preprocessing and cleaning in Fig. 1.

```
Number of rows and columns in the dataset: (656, 28)
Total number of missing values in the dataset: 0
```

Fig. 1 Dataset shape and missing values

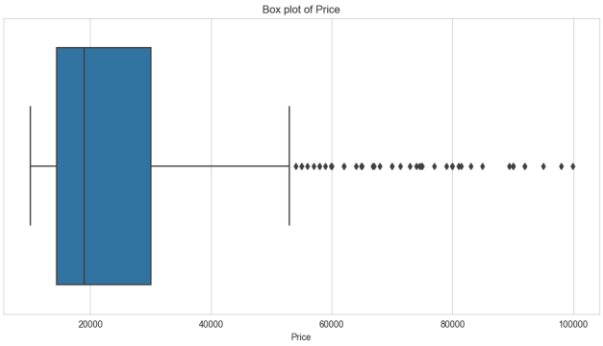


Fig. Boxplot for target value ‘Price’

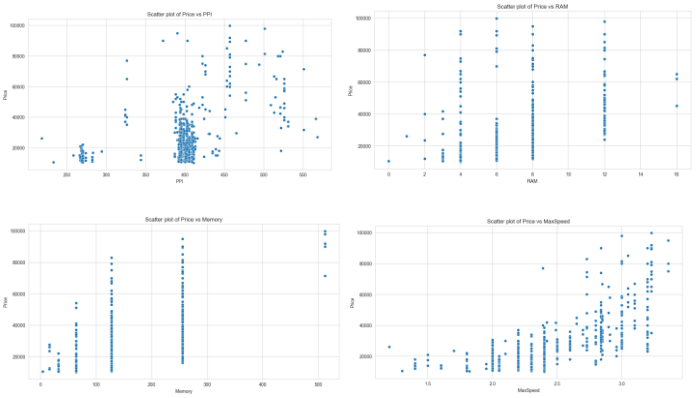


Fig. 2. Variation of price with various features

- As we can see from Fig. 2., as these features values increase, the price of the smart phone also increases. Hence, these features are important for our analysis,

- As we can see from the following scatter plot, as the Age increases, the price seems to be decreasing.

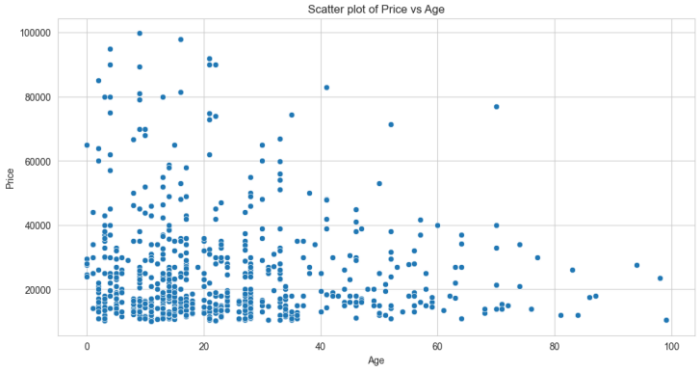


Fig. 4. Age vs Price

- Smart phones having Apple processors, Apple graphics and iOS operating system appear to have the highest prices. We can conclude that apple products cost higher. Indian local brand Infinix and Xiaomi produces value priced mobiles.

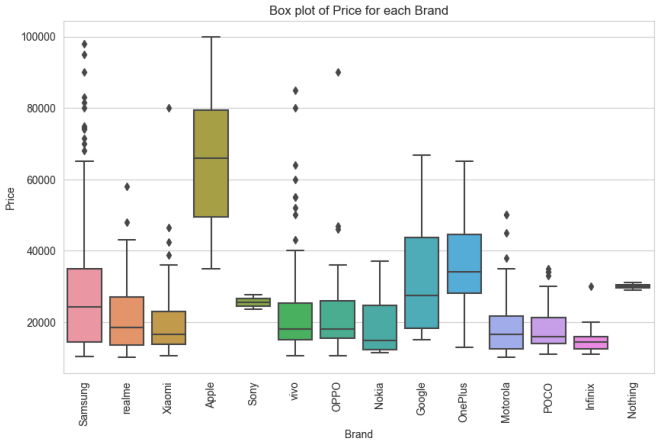
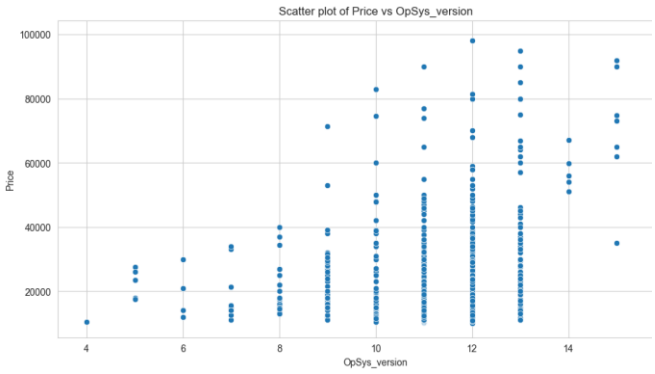


Fig. 5. Brand vs Price



- Finally, to study the correlation between various features, we constructed a heat map.

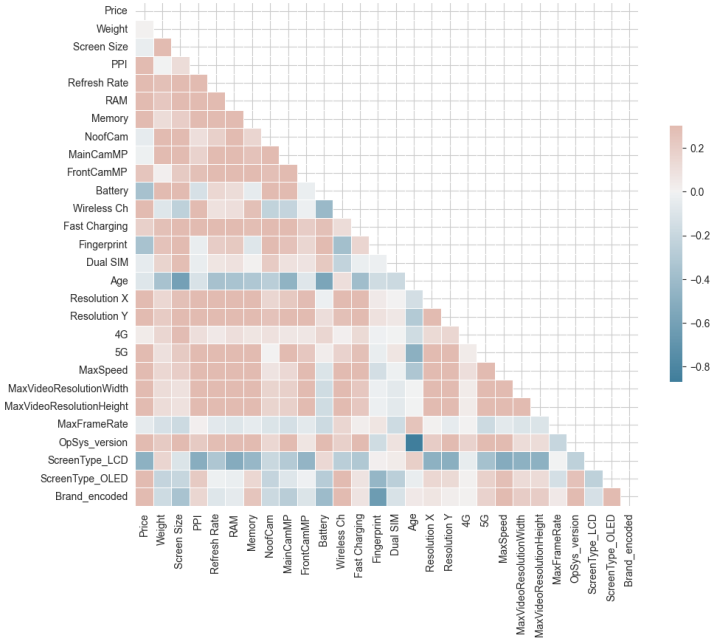


Fig. 10. Heatmap

D. Encoding

- The Categorical values of 'Screen Type' were one-hot encoded for OLED and LCD, the rest being AMOLED.
- Target coding (mean) was done to handle the 'Brand' column. The brand names were replaced by the average price of a phone of the brand. This helped capture the brand value in determining costs.

  E.g. "Apple" was encoded as 65523 and "Samsung" as 22850

E. Feature Scaling and Hyperparameter Tuning

In the preprocessing and modeling phase, a pipeline was implemented using scikit-learn. This pipeline incorporated several key steps: feature scaling with Standard Scaler, Winsorizing to handle extremes at the 5th percentile, and normalization using the Box-Cox method. To optimize the model's performance, an extensive Random Search was conducted over 50 iterations to identify the best hyperparameters. The data was then split into training and testing sets using StratifiedShuffleSplit, and the pipeline was fitted to the training data, ensuring that the model was trained and evaluated in a robust manner.

F. Feature Transformation

The dataset underwent transformations to handle skewness and preserve valuable outlier information. Right-skewed columns, including 'Price', 'Weight', 'PPI', among others, were log-transformed, while the left-skewed 'Screen Size' was squared. The use of only Winsorization was done as minimal manipulation maintained the integrity of the data, reflecting the inherent volatility of the mobile phone market. The Scaling, normalization and transformation helped handle extreme data better.

III. MODELS AND EVALUATION METRICS USED

A Total of 11 models were used:

- *Linear Regression*
- *K-Neighbors Regression*
- *Decision Tree*
- *Random Forest*
- *Gradient Boosting*
- *Extreme Gradient Boosting (XGBoosting)*
- *Light Gradient Boosting Machine (LightGBM)*
- *Categorical Boosting (CatBoost)*
- *Histogram Gradient Boosting*
- *Bagging*

A total of 4 evaluation metrics were used. Here is a summary about them:

- *Mean Absolute Error (MAE) is a metric that quantifies the average absolute difference between predicted and actual values. It is less sensitive to outliers as it doesn't square the errors. A lower MAE signifies superior model performance.*
- *Mean Squared Error (MSE) calculates the average squared difference between predicted and actual values. It assigns more weight to larger errors, making it more sensitive to outliers. A lower MSE indicates a better model in terms of minimizing prediction errors.*
- *R-squared, or the coefficient of determination, measures how well the model explains the variance in the dependent variable. It ranges from 0 to 1, with higher values indicating a better model fit. However, it can be biased towards more complex models.*

IV. MODEL SELECTION

- A StratifiedShuffleSplit was used to divide the dataset into training and testing sets, ensuring a balanced representation of outliers in both sets. Outliers were identified using the Interquartile Range (IQR) method.

- A variety of regression models were considered, including Linear Regression, Support Vector Regression, Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM, CatBoost, HistGradientBoosting, Bagging, and K-Neighbors. Each model was subjected to hyperparameter tuning using RandomizedSearchCV, with specific parameter ranges defined for each model.

- The models were evaluated based on three metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2). These metrics were calculated for each model's predictions on the test set.

- After performing the model selection, Extreme Gradient Boosting Regressor came out to be the best model.

- Learning curves were plotted for selected models to visualize their performance as the number of training examples increases. The learning curves plot the training score and the cross-validation score as functions of the training set size. This helps in understanding the model's learning process and diagnosing issues such as underfitting or overfitting.

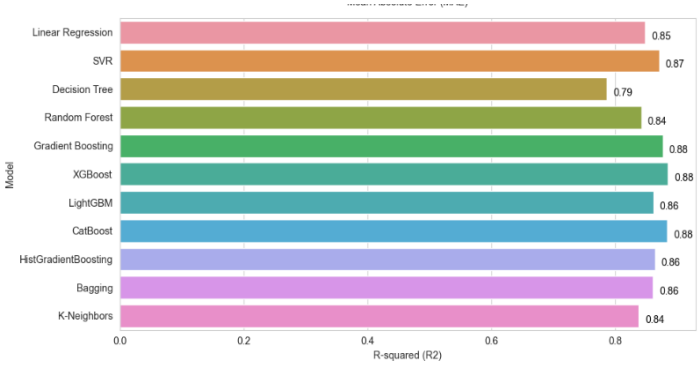


Fig. 13. Best model selection

V. MODEL EVALUATION

- Since we obtained Extreme Gradient Boosting Regressor as the best model among the ones we used in Random Search, we evaluated it. Here are the results in Fig. 14.

```
XGBoost Metrics:
Mean Squared Error (MSE): 0.11388243917609256
Root Mean Squared Error (RMSE): 0.3374647228616534
Mean Absolute Error (MAE): 0.2517909646289533
R-squared (R2 ): 0.8849992344688793
```

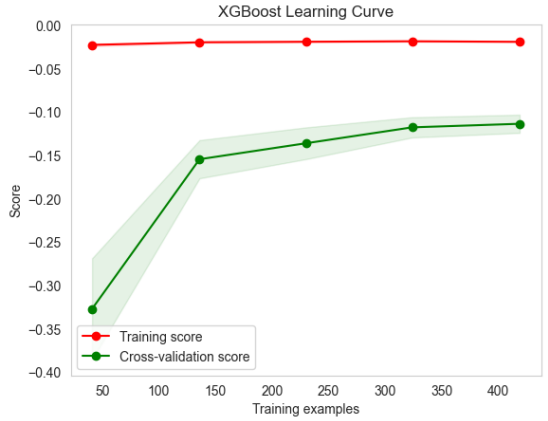


Fig. 14. Results of Extreme Gradient Boosting Regressor

```
Random Forest Metrics:
Mean Squared Error (MSE): 0.15679181906287928
Root Mean Squared Error (RMSE): 0.39596946733666133
Mean Absolute Error (MAE): 0.29528015093086746
R-squared (R2 ): 0.8416684841693014
```

Fig. 15. Results for Random Forest Regressor

```
Linear Regression Metrics:
Mean Squared Error (MSE): 0.15026370929542557
Root Mean Squared Error (RMSE): 0.38763863235676804
Mean Absolute Error (MAE): 0.3063758844216364
R-squared (R2 ): 0.848260700020663
```

Fig. 16. Results for Linear Regression

```
K-Neighbors Metrics:
Mean Squared Error (MSE): 0.16074657186286767
Root Mean Squared Error (RMSE): 0.4009321287485797
Mean Absolute Error (MAE): 0.2945090675167726
R-squared (R2 ): 0.8376748956689553
```

Fig. 17. Results for K-Neighbors Regressor

```
CatBoost Metrics:
Mean Squared Error (MSE): 0.11521314890705703
Root Mean Squared Error (RMSE): 0.3394306245863167
Mean Absolute Error (MAE): 0.25436176852789777
R-squared (R2 ): 0.8836554571589817
```

Fig. 18. Results for Random Forest Regressor

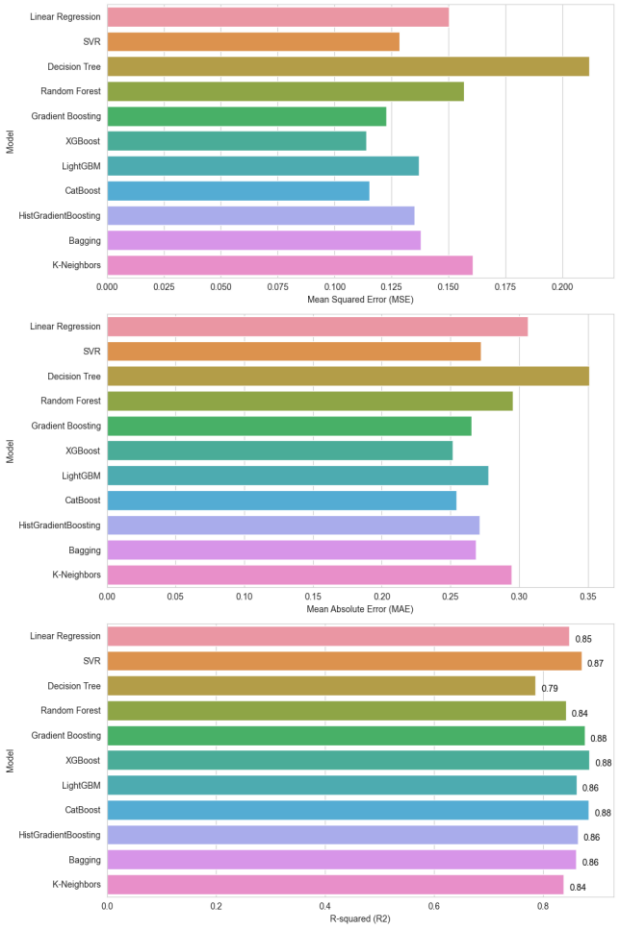


Fig. 18. Comparison of various models

VI. CONCLUSION

In conclusion, the XGBoost model demonstrated the highest performance in predicting mobile phone prices, achieving the best R2 score and the lowest MAE and MSE. However, it's noteworthy that other models such as CatBoost, Gradient Boosting, SVR, Bagging, and LightGBM also exhibited competitive performance, indicating the robustness of the feature set used and the effectiveness of the preprocessing and hyperparameter tuning steps.

As for future work, there are several avenues to explore to further improve the model's performance. One could consider incorporating more features that could potentially influence mobile phone prices, such as brand reputation, user reviews, or technological trends. Additionally, more sophisticated model architectures, such as deep learning models, could be explored. Furthermore, ensemble methods that combine the predictions of multiple models could be employed to potentially achieve better performance. Lastly, the model could be retrained and updated regularly with new data to ensure its relevance and accuracy in the ever-evolving mobile phone market.

VII. REFERENCES

[1] Ningyuan Hu, University College London, “Classification of Mobile Phone Price Dataset Using Machine Learning

Algorithms", International Conference on Pattern Recognition and Machine Learning, 2022

[2] Muhammad Asim, Zafar Khan, UET Lahore, "Mobile Price Class prediction using Machine Learning Techniques" International Journal of Computer Applications (0975 – 8887) Volume 179 – No.29, March 2018

[3] Pritish Arora, Sudhanshu Srivastava, Bindu Garg, Bharati Vidyapeeth (Deemed to be University) College of Engineering, "MOBILE PRICE PREDICTION USING WEKA", International Journal of Scientific Development and Research, Volume 5, April 2020

[4] Sireesha Mitta, U. Arul, Madanapalle Institute of Technology & Science, "Mobile Price Prediction Using Feature Selection and Classifier Algorithms of Machine Learning", e-ISSN: 2582-5208, International Research Journal of Modernization in Engineering Technology and Science Volume:03 Issue:06 June-2021

[5] A. Renuka, Veer Sudheer Goud, Holy Mary Institute of Technology & Science, "Online Mobile Price Prediction using Machine Learning", Turkish Online Journal of Qualitative Inquiry (TOJQI), Volume 13, Issue 1, January 2022: 681-688