Prediction of Mobile Phone Price Dataset Using Machine Learning Algorithms

Suyash Gupta

B. Tech (IT)

Manipal Institute of Technology

Suyash1567@gmail.com

ABSTRACT: Mobile phones play a crucial role in our daily lives, prompting careful consideration when making a purchase. This research project focuses on predicting mobile phone prices by analyzing key factors such as brand reputation, camera quality, processor speed, and 4G band availability. A comprehensive dataset was collected, cleaned, and transformed to ensure accuracy. Feature engineering techniques were employed to extract meaningful insights and optimize predictive models. Machine learning algorithms, including Linear Regression, ADB Regressor, Random Forest Regressor, Gradient Boosting Regressor, and XGB Regressor, accurately predicted prices based on phone features. This research provides valuable insights for consumers, aiding in informed decision-making based on preferences and budget. Similarly, businesses can utilize the predictive models to set competitive prices in the market.

Keywords- Mean Absolute Error (MAE), Gradient Boosting (GR), Random Forest (RF), R-squared Error, Explained Variance Score.

SECTION A: LITERATURE REVIEW

Several research papers have contributed significantly to smartphone price prediction [1][2][3][4][5]. They utilize diverse datasets encompassing various features, including display, processor, memory, camera, thickness, battery, and connectivity. Machine learning algorithms such as random forest classifier, support vector machine, logistic regression, decision tree, K-nearest neighbours, and naive Bayes have been employed to evaluate their effectiveness in predicting smartphone prices [1][2]. Notably, logistic regression, support vector machine, and support vector classifier (SVC) demonstrate high accuracy rates ranging from 81% to 94.7% [1][2][4]. These findings aid consumers in making informed purchasing decisions and assist manufacturers in setting appropriate prices [1][2][4]. Additionally, studies focus on recycled mobile phones, proposing models based on fuzzy neural networks, principal component analysis, and momentum optimization to accurately determine their value [5]. These models improve decision-making in the recycling market by considering nonlinear mapping relationships and fuzzy concepts [5]. Overall, these papers contribute valuable insights to the dynamic smartphone market and highlight the importance of machine learning techniques, diverse datasets, and feature analysis in price prediction research.

SECTION B: CORE CONTENT

I. INTRODUCTION

In the competitive world of mobile phones, pricing is a major concern for consumers. With Apple holding a 21% market share, Samsung with 22%, and Xiaomi capturing 11%, understanding how different features impact prices is crucial. This study employs machine learning techniques to predict mobile phone prices based on specific smartphone features. By analyzing a comprehensive dataset, we meticulously clean and transform the data to ensure accuracy. Leveraging advanced models and algorithms, including Linear Regression, AdaBoost Regressor, Random Forest Regressor, Gradient Boosting Regressor, and XGB Regressor, we accurately predict prices. This research empowers consumers to make informed decisions, aligning their needs and budgets with appropriate feature sets and price ranges. By utilizing these predictive models, businesses can set competitive prices in the market.

II. METHODS AND MATERIALS

This section aims at providing a brief overview of the methods and materials associated with the use of different machine learning algorithms. All the analysis in this study was performed in Kaggle integrated Jupyter Notebook.

A. DESCRIPTION OF DATASET

The dataset used is ‘Smart phone price prediction’ by Arnold Anand uploaded on Kaggle. It consists of mobile phone specifications and prices. It provides information on various features such as hardware components, operating system, camera details, battery capacity and pricing. The dataset underwent cleaning and preprocessing steps to handle missing values and ensure data quality.

B. DATA CLEANING AND PREPROCESSING

The dataset underwent several cleaning and preprocessing steps to ensure data quality and consistency. The following steps were applied:

1. Removal of irrelevant columns: Irrelevant columns were dropped from the dataset to reduce the complexity of the dataset and also improve the performance and accuracy of the models.
The dropped columns either are not relevant for our prediction task or contain a large number of same values which do not affect the decision making of the model.
The following columns were dropped from the dataset.

'Name', *'Brightness'*, *'Screen to Body Ratio'*, *'Chipset'*, *'Bluetooth'*, *'Capacity'*, *'Wi-Fi'*, *'GPS'*, *'SIM Slots(s)'*

2. Handling missing values: Handling missing values involves addressing the issue of data points that are incomplete or unavailable. Various approaches can be employed to handle missing values, ensuring that they do not adversely affect the analysis or modelling process.
The rows with missing values for the following were either filled with 'other' for the respective column or removed from the dataset:

'RAM', *'Processor'*, *'Battery'*, *'Front Camera'*, *'Operating System'*

3. Data transformation: Data transformation involves modifying existing data to meet analysis requirements, improve quality, and enable better interpretation. The dataset underwent several transformations, including numeric column extraction and conversion, launch date transformation, operating system standardization, graphics modification, display type categorization, handling missing values, binary conversion of quick charging, USB type-C filling based on launch date, audio jack standardization, SIM slot transformation, fingerprint sensor filling based on RAM, price cleaning and rounding. These transformations enhanced data quality and suitability for analysis.

The dataset is now cleaned and pre-processed, ready for further analysis and modelling.
Below is a preview of the dataset after cleaning and preprocessing.

	Brand	Name	RAM	Processor	Battery	Rear Camera	Front Camera	Display	Launch Date
0	OnePlus	OnePlus Nord CE 3 Lite 5G	8.0	Qualcomm	5000	108	16.0	6.72	2023
1	realme	realme 10 Pro Plus 5G	6.0	MediaTek	5000	108	16.0	6.70	2022
2	realme	realme Narzo N53	4.0	Unisoc	5000	50	8.0	6.74	2023
3	OnePlus	OnePlus 11R	8.0	Qualcomm	5000	50	16.0	6.74	2023
4	POCO	POCO F5	8.0	Qualcomm	5000	64	16.0	6.67	2023

Table 1: A snippet of the dataset

C. LABEL ENCODING

Label encoding is a technique used to convert categorical variables into numerical representations. It assigns a unique numerical value to each category in a categorical feature, allowing machine learning models to interpret and utilize these features during the prediction process. In this study, label encoding was applied to various features of smartphones to convert them into numerical labels.

The following features were label encoded:

Brand, Processor, Operating System, Graphics, Display Type, Quick Charging, USB Type-C, Expandable Memory, Audio Jack, Fingerprint Sensor, Wi-Fi, GPS

By label encoding these features, each unique category is assigned a unique integer label. This numeric representation enables machine learning algorithms to process and analyze the categorical information effectively.

D. DETECTING AND REMOVING OUTLIERS

Outliers are data points that significantly deviate from the normal distribution or expected patterns within a dataset. They can have a substantial impact on the analysis and modelling process, leading to misleading results. Thus, it is crucial to detect and minimize the number of Outliers in the dataset.

The following steps were performed for outlier detection and removal:

1. Duplicate Rows: Duplicate rows were identified and removed from the dataset to eliminate any redundant data entries that could bias the analysis.

2. Brand Filtering: Smartphones from the brand 'Apple' were filtered out to focus the analysis on non-Apple smartphones specifically. This decision was made due to limited data availability for Apple smartphones and their premium pricing, which could impact the overall model results.

3. Histogram Plots: Histogram plots were used to visualize the distribution of various features in the dataset, such as 'RAM', 'Processor', 'GPS', 'Pixel Density', 'Price', 'Rear Camera', 'Refresh Rate', 'SIM Slots', 'Expandable Memory', 'Fabrication', 'Wi-Fi', 'Internal Memory', 'USB-Type C', 'Quick Charging', and 'Operating System'. By analyzing these histograms, rows with values below a certain percentage for a specific feature were removed. This step helped refine the dataset and focus on relevant data points.

The outlier detection and removal steps aim to improve the data quality and ensure that the dataset is suitable for analysis and modelling.

E. FEATURE SELECTION

Feature selection is a crucial step in machine learning where the most informative features are chosen to improve model performance. By selecting relevant features and eliminating irrelevant ones, it reduces overfitting, enhances generalization, and simplifies model interpretation. Additionally, feature selection helps optimize computational resources by reducing dimensionality. Overall, it plays a significant role in creating accurate and efficient predictive models.

Following is a summary of the process:

Data Preparation: The dataset is divided into input features (X) and the target variable (y). A train-test split is performed, with 20% of the data reserved for testing. A pipeline is created, consisting of two steps: feature scaling using MinMaxScaler and applying the Lasso regression model.

Grid Search: GridSearchCV is employed to search for the optimal hyperparameter value (alpha) for the Lasso model. The alpha values are explored within the range of 0.1 to 10 with a step size of 0.1. Cross-validation with 6 folds is employed to assess the model's performance. The negative mean squared error (neg_mean_squared_error) is used as the evaluation metric.

Best Model Selection: Following the grid search, the hyperparameter value (alpha) yielding the best performance is identified. The most important features and their corresponding coefficients are extracted from the best Lasso model. The absolute values of the coefficients are utilized to determine feature importance. Only the features with non-zero coefficients are considered as important features.

Dataset Transformation: The dataset is transformed to include only the selected features and the target variable (Price). Any rows with missing values are removed from the dataset. By applying the Lasso regression model and considering the identified important features, the feature selection process aims to enhance the predictive capability of the model by focusing on the most influential attributes.

III. MODELS

Model selection is a crucial step in machine learning, involving the evaluation and comparison of different models. It includes training and testing various models with different algorithms and configurations, using metrics like mean squared error, mean absolute error, R-squared, and explained variance score. Cross-validation is commonly used to ensure reliable performance estimates. The goal is to choose the model that performs best on unseen data, avoiding overfitting or underfitting. The selected model can then be refined and optimized for accurate predictions on new data. A brief description of each of the models selected is given below:

Linear Regression: A simple algorithm that models the relationship between a dependent variable and independent variables using a linear equation.

AdaBoost Regressor: An ensemble learning algorithm that combines weak models iteratively, assigning higher weights to instances with larger errors for improved prediction.

Random Forest Regressor: An ensemble learning method that constructs multiple decision trees and combines their predictions, reducing overfitting and improving generalization.

XGBoost Regressor: An optimized gradient boosting algorithm that builds an ensemble of weak models, correcting mistakes made by previous models and controlling overfitting.

Gradient Boosting Regressor: Another gradient boosting algorithm that builds a strong model by sequentially adding weak models to minimize the loss function, handling complex relationships, and producing accurate predictions.

IV. MEASURING PERFORMANCE

Each of the models selected gave varying results for the evaluation metrics chosen, namely, Random State, Mean Absolute Error (MAE), R-Squared Score (R2) and Root Mean Squared Error (RMSE). The results of the various models are given below:

Model	RMSE	MAE	R2	Random State
Linear Regressor	4379.6472	3128.212	0.84628	41
AdaBoost Regressor	5994.8918	5042.077	0.75097	9
RF Regressor	3157.2845	2176.108	0.92393	72
XGB Regressor	3434.5165	2256.315	0.90999	72
GB Regressor	3412.2993	2116.833	0.91115	72

Table 2: Performance metrics of different models

It may be observed that RF Regressor and GB Regressor exhibit superior performance, with lower MAE values, as well as higher R-squared Score. These models demonstrate better accuracy, fit, and predictive power in estimating smartphone prices.

GB Regressor:

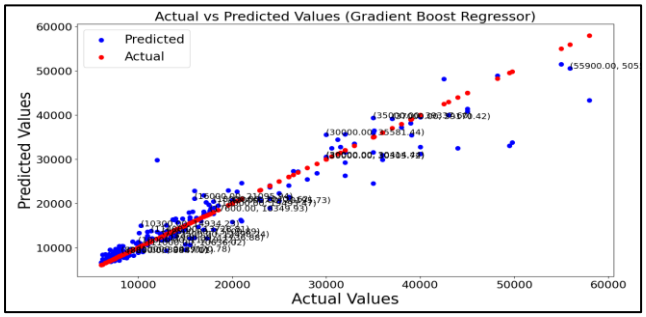

Figure 1: A scatter plot of actual vs predicted values obtained by using GB Regressor.

The Gradient Boosting Regressor model is well fitted to lower-priced smartphones, exhibiting a strong correlation with their actual prices. However, as smartphone prices increase, deviations from the predicted values become more noticeable. This discrepancy may be attributed to the limited number of samples available for training higher-priced smartphones compared to lower-priced ones.

AdaBoost Regressor:

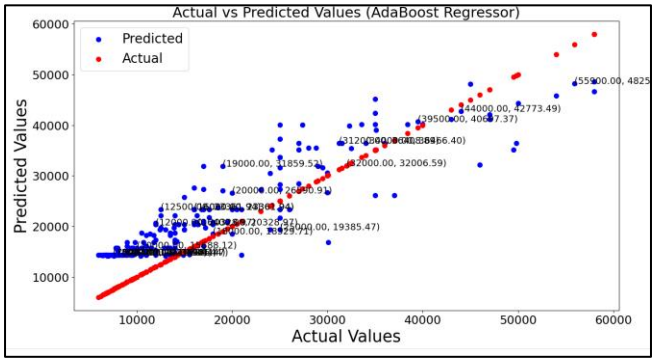

Figure 2: A scatter plot of actual vs predicted values obtained by using AdaBoost Regressor.

An observation may be drawn that AdaBoost Regressor does not fit well for our task of smartphone price. The model deviates significantly from the actual prices, even for smartphones at a lower price point. This result is consistent with our performance metrics for AdaBoost Regressor.

Linear Regressor:

We might be able to observe that Linear regressor performs better than AdaBoost Regressor, but it's fit for the prices is still not satisfactory. Even for smartphones with lower prices a noticeable deviation can be observed in the predicted values.

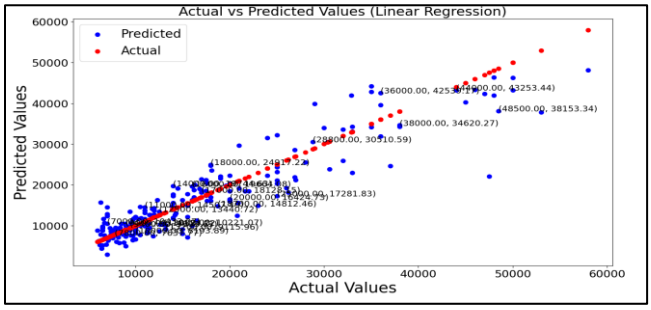

Figure 3: A scatter plot of actual vs predicted values obtained by using Linear Regression.

RF Regressor:

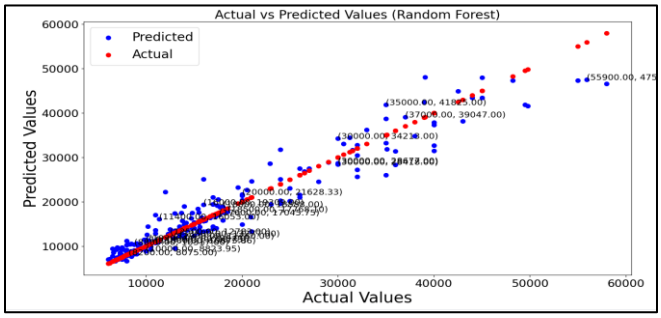

Figure 4: A scatter plot of actual vs predicted values obtained by using RF Regressor.

The obtained plot for the RF Regressor closely resembled the results for the GB Regressor. These findings are supported by their respective performance metrics. In some runs, the RF Regressor slightly outperformed the GB Regressor due to the random splitting and sampling of the data, which leads to slight variations in the models. Given the close relationship between the results of the two models, either one could prevail in a specific run.

XGB Regressor:

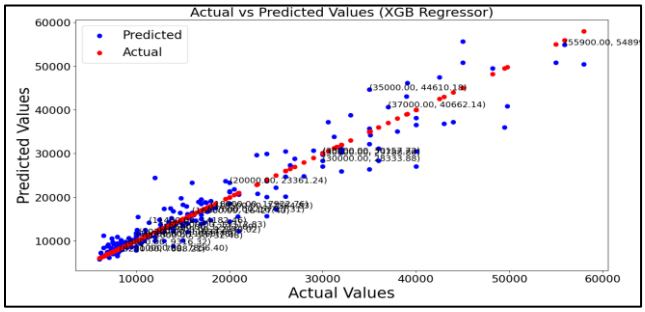

Figure 5: A scatter plot of actual vs predicted values obtained by using XGB Regressor

The scatter plot comparison reveals that the GB Regressor performs better than the XGB Regressor in predicting smartphone prices, particularly for premium devices. The GB Regressor demonstrates a more accurate and consistent fit across different price ranges, while the XGB Regressor exhibits a higher degree of deviation and less accuracy for

higher-priced smartphones. These findings are supported by the model's performance metrics.

V. CONCLUSION

In this study, we explored the predictive modelling of smartphone prices using machine learning algorithms. The findings highlight the importance of accurate price estimation in the competitive smartphone market. The Gradient Boosting Regressor emerged as the top performing model, exhibiting strong predictive capabilities for smartphones with lower price points. However, the models' accuracy diminished for high-end devices, possibly due to the limited availability of data for premium smartphones. To improve future predictions, it is recommended to expand the dataset to encompass a wider range of high-end smartphones and explore advanced feature engineering techniques. Nonetheless, this study contributes valuable insights for industry stakeholders, aiding in pricing strategies, market analysis, and decision-making processes.

VI. REFERENCES

1. Subhiksha, S., Thota, S., Sangeetha, J. (2020). Prediction of Phone Prices Using Machine Learning Techniques. In: Raju, K., Senkerik, R., Lanka, S., Rajagopal, V. (eds) Data Engineering and Communication Technology. Advances in Intelligent Systems and Computing, vol 1079. Springer, Singapore. https://doi.org/10.1007/978-981-15-1097-7_65

2. N. Hu, "Classification of Mobile Phone Price Dataset Using Machine Learning Algorithms," *2022 3rd International Conference on Pattern Recognition and Machine Learning (PRML)*, Chengdu, China, 2022, pp. 438-443, doi: 10.1109/PRML56267.2022.9882236.

3. Asim, Muhammad & Khan, Zafar. (2018). Mobile Price Class prediction using Machine Learning Techniques. International Journal of Computer Applications. 179. 6-11. 10.5120/ijca2018916555.

4. A. Kalmaz and O. Akin, "Estimation of Mobile Phone Prices with Machine Learning," *2022 International Conference on Engineering and Emerging Technologies (ICEET)*, Kuala Lumpur, Malaysia, 2022, pp. 1-7, doi: 10.1109/ICEET56468.2022.10007128.

5. H. Liu, J. Huang, H. Han and H. Yang, "An Improved Intelligent Pricing Model for Recycled Mobile Phones," *2020 Chinese Automation Congress (CAC)*, Shanghai, China, 2020, pp. 3724-3731, doi: 10.1109/CAC51589.2020.9327611.