**Mobile Price prediction using phone Specifications - Regression**

**Group Name: Team Danilo**

**Group Members:**

|  |  |  |
| --- | --- | --- |
| **First name** | **Last Name** | **Student number** |
| Aanal | Patel | C0910376 |
| Bimal Kumar | Shrestha | C0919385 |
| Danilo | Diaz | C0889539 |
| Ernie | Sumoso | C0881591 |
| Jayachandhran | Saravanan | C010392 |

**Submission date: 13-Apr-2024**

Contents

[Abstract 2](#_Toc163899718)

[Introduction 3](#_Toc163899719)

[Dataset 4](#_Toc163899720)

[Methods Exploratory Data Analysis: 5](#_Toc163899721)

[Visualization 6](#_Toc163899722)

[Preprocessing 7](#_Toc163899723)

[Training and model testing 8](#_Toc163899724)

[Model table (baseline and hyper-parameter tuning) 8](#_Toc163899725)

[Conclusions 10](#_Toc163899726)

[Annexures 11](#_Toc163899727)

[References 13](#_Toc163899728)

# **Abstract**

This project will discuss the various steps involved, from fetching the raw data to comparing the performance of the models. In this activity, the data speaks about mobile phone prices and the respective important specifications (features) of that particular mobile phone. Standard data operations are applied throughout the assignment to address the distribution of the numeric data and handle the outliers. Using feature selection, the top five features are listed, and their importance is compared with each trained model to verify the impact of the selected feature on the predictions. The linear regression model shows decent performance as it is considered the baseline model in this activity, further, the residuals are moderate among other models. The regularization and hyperparameter tunings are applied, and the results are discussed and elaborated. Lasso outperformed the used models, with good performance on both the testing and training datasets and a mean square error of 0.045. random forest and gradient-based boosting models are showing slight overfitting characteristics The following report shows a complete overview of the regression-based approach to predicting the mobile phone(smartphone) price.

**Key Words: regression, mobile phone, feature selection, linear regression, Lasso regularization (L1 Lorm), Mean Square Error, random forest, gradient-based boosting, baseline model and smartphone**

# **Introduction**

The usage of mobile phones in this modern era is considered to be one of the basic necessities along with the other basic needs. The smartphone brings the whole world into the user’s palm, and choosing it requires more detailed research comparisons. Each year, technology grows rapidly and also determines the price of the smartphone with the presence of specifications which include but are not limited to touchscreen, dual SIM, global positioning sensor (GPS), Bluetooth capabilities, processing chip, internal storage and processing memory. In this report, a detailed analysis of smartphone prices is carried out, and the regression model to predict the upcoming mobile phone price is attained. There is still room for improvements in tuning the model’s performance and considering more important features which add more weight to the model during the training phone. The following is the task distribution of this project,

|  |  |
| --- | --- |
| **TASK** | **PURPOSE/DESCRIPTION** |
| Data source | - Finalizing an informative dataset to carry out the mentioned Machine Learning Steps to gain more hands-on experience |
| Feature set understanding | - 360-degree examination of the dataset to articulate the further analysis |
| Exploratory Data Analysis | - Handling the data instabilities and uncovering the characteristics of the dataset |
| Categorical and Numerical Feature processing | - Special attention to the feature types as they impact the performance of the model |
| Pre-Processing | - Standardization and normalization of the data points to create some meaningful pattern for the model to learn |
| Model Pipeline | - Significant steps to experiment with various models and their predictions on different types of data combination |
| Tuning Model parameters | - tweaking and checking the performance of the models based on their potential parameters (e.g. learning rate, iterations ) |
| Results Comparison | - Observation and inference of the performance with the known and unknown dataset |

# **Dataset**

The data is fetched from the famous Gadgets 360 website, where the ultimate data describes the Indian demographics. The total size of the data is around 29898 (~30K approx.), which consists of 22 features as columns. Further, the dataset terminology is explained below,

|  |  |
| --- | --- |
| **Feature Name** | **Representation** |
| Name | Mobile (smart) phone name |
| Brand | The brand which entitles the phone |
| Model | Model name specifies the version of the particular mobile phone (e.g., iPhone13 has mini, Pro, and Promax as model names) |
| Battery | Power Capacity of the phone mentioned in milli Ampere (hour) |
| Screen size | diagonal measurement of the screen in inches |
| Resolution x and y | Represents the screen pixel rate with respect to the X and Y axis |
| RAM | Processing Power (memory) in Megabytes |
| Processor | Specification of the processor chip |
| Internal Storage | The storage capability of the phone is measured in Gigabytes |
| OS | Operating System details |
| Rear and Front Camera | Camera resolution capturing specifications |
| Number of sims | Sim slots available on the phone |
| Feature set | * Bluetooth * Wi-Fi * 3G and 4G * GPS * Touch screen |
| Price | Selling price (in Rupees) |

A screenshot of a data overview

Description automatically generatedA white background with black lines

Description automatically generated

*Figure 2:*

# **Methods** Exploratory Data Analysis:

A screenshot of a computer screen

Description automatically generatedThis section encloses the required data analysis carried out. The raw data is identified with the shape of (1359, 22). The primary data validation of checking the missing and duplicate values is checked, and the data is more likely processed before fetching, so a very negligible amount of values is seen. The extensive analysis of the data, like statistical analysis of the numeric columns using describe() provides more meaningful insights into the data distribution and the important findings present in the data. Further, the data is checked for outlier values. In this activity, the numeric columns are validated using more than one method to check the presence of the outliers. The reason is to use more than one technique to understand the outliers and develop a proper prevention or handling pipeline. After looking at the descriptive table, the numeric values are plotted using a line plot along with the target variable to check the spikes and longevity of the tail or extreme values.

A graph of a graph of a graph

Description automatically generated with medium confidenceThe five-number summary statistics are used to get the interquartile range (IQR), which is then visualized using boxplots to capture the potential data points. The QQ (prob\_plot) method is used to substantiate the theoretical representation of the data points against the actual values.

It clearly explains the distribution and provides a comparative analysis of the projected values. Similarly, the binary data types are analyzed using the pie chart and the boolean representation ratios are clearly noted. A close-up of a pie chart

Description automatically generatedThe Frequency chart for the categorical values is displayed for the potential features to interpret its values. The frequency count also provided the variance of the column values, which are highly required for feature selections. Finally, the informative analytics of the features are printed, and the data types are verified

## Visualization

Numerous types of visual representation tools are utilized in this project for various analytical reasons. The following table gives an oversight view of the visual graphs used and their purpose.

|  |  |
| --- | --- |
| Boxplots | Capturing potential outliers |
| Distribution and QQ plot | Detailed analysis of the outliers w.r.t columns & the target variables |
| Frequency chart | Understand the unique values present in the categorical columns |
| Pie Chart | Ratio presentation of the binary values |
| Heatmap | Correlation of features |

## Preprocessing

A comparison of a graph

Description automatically generatedThe pre-processing steps include handling outliers present in the numeric columns, validation of the categorical and numeric values, target variable correlations and potential multi-colinearity handlings. The categorical features are processed and converted to numeric representation using the label encoding technique. With experimentation of all the available approaches to handle outliers, this dataset is more suitable for applying logarithmic transformation to preserve the originality of the data and avoid unwanted bias in the cleaned data. After the numeric values are processed, the outlier is quite less significant in impacting the model’s performance. The visual representation of the numeric values and their distribution is plotted using a histogram method with density estimation. It emphasizes the distribution is normalized with a standard deviation of 1 and mean value of 0.

A screenshot of a chart

Description automatically generatedIn addition to handling the outliers, feature removal is applied to ignore the less significant columns like Names, models and brands from training. The multilinear data features like the resolution of the Y axis, screen size and the front camera are avoided. Finally, feature engineering is applied to the five categorical columns, creating the latest tech flag column. It is defined as a boolean data type with “1” representing the presence of all the features and “0” resembling the absence of even one feature. Initial log transformation was applied to the target variable, and the results were not good, so the transformation was applied to all the numeric values.

**Results**

## Training and model testing

A screenshot of a computer code

Description automatically generatedThe processing pipeline is constructed with one transformer step and the regression model step. Then, the experiment showed better results on the combination of pre-processed categorical value is allowed through the “passthrough” flag, thereby reducing the steps involved in the transformation. The numeric values are scaled using the standard scaler method with a mean value of 0 and a standard deviation of unit 1. Prior to this, the features are mapped to the SelectKBest method to get the best 5 features of the given dataset. The linear regression model is considered as the baseline model, and the train\_test set values are configured in a 70:30 ratio with selected features. The training pipeline is optimized in a way that fits the data to the model and gets the prediction values for both the training and testing datasets. The model shows decent performance on both the known and unknown datasets.

## Model table (baseline and hyper-parameter tuning)

|  |  |
| --- | --- |
| **Model** | **Remarks** |
| Linear Regression | Decent Performance on both testing and training data |
| Random forest regressosr | High overfitting |
| Gradient Boosting regressor | Slight overfitting |
| Random forest (hyper\_parameters) | Slight overfitting |
| Gradient Boosting (hyper\_paramters) | Overfitting is seen |
| Ridge Regularisation | Near moderate performance |
| Lasso Regularisation | Best performance on both testing and training |

The model is evaluated using standard metric systems like R2, Mean Square Error and Mean Absolute Error scores. The values are saved and formulated as a comparison table to investigate the results of various combinations of datasets. Post-training, the importance of the features is carried out to get the most important features for each model. The top 5 weighed features are compared with the Kbest 5 A graph with blue dots

Description automatically generatedfeatures selected during the pre-training module. Surprisingly, the feature sets are closely associated, but the weighting is varied, with collinearity detected. The scatter plot is implied here to show the actual and predicted data points visually, and the regression lines are plotted along to give a better understanding.

A graph with a blue line

Description automatically generated with medium confidence The residuals of each model are plotted with a density curve, and the histogram representation is displayed for various references. Using Plotly, an interactive graphical method, the evaluation metrics are shown, and the feature of closely examining the values is made possible. Thus, the project comprises a detailed approach toward a regression-based prediction problem.

**Business Case and Future Scope:**

In the future, this project will potentially support numerous price predictions, and further improvements will be helpful in creating a trend analysis of various smart gadgets. It also brings more value to the distributors and the consumers by allowing them to compare the price authentically and effectively instead of paying wholesome, unverified system amounts in the local region. In addition, it will be useful for all kinds of people abiding by the DEI principles

# **Conclusions**

The project analysis and the detailed notebook can be viewed in the hosted remote repository, which contains the dataset and the analysis that was carried out. It also comprises of the final version of the notebook, which discusses the performance of the models along with the pandas profiling. It also provided a clear picture to the team on how to proceed with real-world data and further apply this project in data mining aspects. A neural network-based model and incorporation of live data are planned for the next phase.

A table with numbers and letters

Description automatically generated

# Annexures

**Annexure I: Pandas Profiling**

A screenshot of a computer

Description automatically generated

**Annexure 2: Evaluation Metrics**

A blue and white striped graph

Description automatically generated

A graph of blue and red bars

Description automatically generated

**Annexure 3: Hyper Parameter Selection and Feature importance**

Parameters selected for tuning and finding optimized model:

|  |  |
| --- | --- |
| **Model** | **Parameters** |
| Random Forest | Number of decision trees and the data points split are considered here to find the best estimations.  “n\_estimators”,”min\_sample\_split”, “max\_depth” |
| Gradient Boosting | Similar to the random forest, the learning\_rate is consider as additional component to find the best prediction factors.  “n\_estimators”,”learning\_rate”, “max\_depth” |
| Regularization | “alpha” – the regularization constant (strength) |

A screenshot of a computer

Description automatically generated

# **References**

*Mobile phone Specifications and Prices*. (2022, August 14). Kaggle. <https://www.kaggle.com/datasets/pratikgarai/mobile-phone-specifications-and-prices>

Gadgets 360. (n.d.). *Tech News, Latest Technology, Mobiles, Laptops - Gadgets 360*. <https://www.gadgets360.com>

Sruthi, L. (2022, January 7). Hyperparameter tuning in linear Regression. - Analytics Vidhya - medium. *Medium*. <https://medium.com/analytics-vidhya/hyperparameter-tuning-in-linear-regression-e0e0f1f968a1>

Lewinson, E. (2023, July 11). *A comprehensive overview of regression evaluation metrics | NVIDIA Technical blog*. NVIDIA Technical Blog. <https://developer.nvidia.com/blog/a-comprehensive-overview-of-regression-evaluation-metrics/>

*Feature selection*. (n.d.). Scikit-learn. <https://scikit-learn.org/stable/modules/feature_selection.html>

Daython. (2023, May 14). Mastering the Art of Feature selection: Python techniques for visualizing feature importance. *Medium*. <https://medium.com/@daython3/mastering-the-art-of-feature-selection-python-techniques-for-visualizing-feature-importance-cacf406e6b71>