Mobile Price Prediction Using Maching Learning

PROJECT REPORT

by

Kailas Nath S (MGP20UCS082)

Unni Krishnan G (MGP20UCS140)

Susan Jacob (MGP20UCS137)

Supervised

by

Prof. Arun Sebastian

Mobile Price Prediction Using Machine Learning

1 Introduction

Predicting mobile phone prices using machine learning is a relevant and valuable problem statement in today's world due to the increasing reliance on smartphones and their wide availability. Accurate price prediction can assist consumers in making informed purchasing decisions and help manufacturers optimize pricing strategies. Furthermore, this approach can be extended to other industries where pricing plays a crucial role, enabling businesses to maximize profitability and enhance market competitiveness.

2 Literature Review

We referred to many research papers in the field of 'price prediction using machine learning'. We found that machine learning was used to predict prices in various domains, such as second-hand cars and houses. Sameerchand-Pudaruth [1] applied techniques including Multiple Linear Regression, KNN, Decision Tree, and Naïve Bayes to predict car prices in Mauritius, while Shonda Kuiper [2] focused on multivariate regression for General Motors cars. Mariana Listiani [3] explored SVM for leasing car prices. These studies contribute insights into machine learning-based price prediction methods.

3 Methodology

3.1 Data Collection

For this project, we utilized a publicly available data-set. This particular data-set was chosen because it had continuous price values.Once the data-set was chosen, it was cleaned by removing duplicate rows and columns, separating data in each column to form continuous values,also the categorical data were converted using One-hot encoding to improve prediction accuracy.

Features: In order to predict the price of the mobile phones we considered some features for estimation such as sim count, display size(inches), cores, clock speed, front and rear camera information, RAM, ROM, 4G, 5G, battery and OS.

Flow Chart:

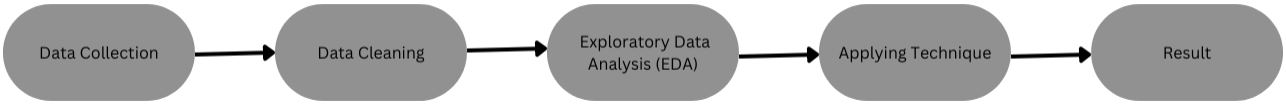


Figure 1

Use: Helps users, manufacturers, and retailers in various ways.By leveraging historical data and features of

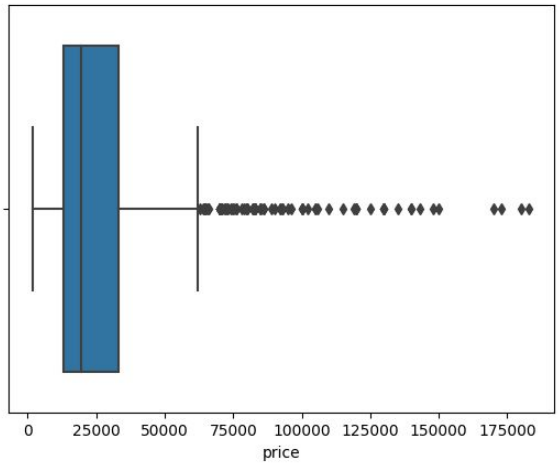
mobile devices, machine learning algorithms can analyze patterns and make predictions about the price of a mobile phone.

4 Exploratory Data Analysis

Comparing price range and mobile features using different types of Graphs.

Box Plot

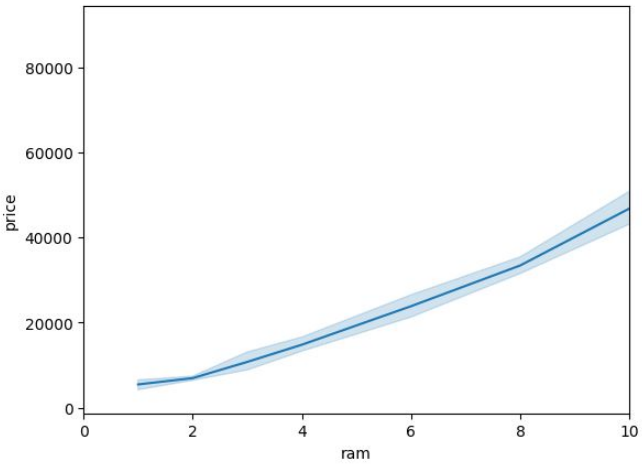
sns.boxplot(x="price",data=data)



To detect outliers in data.

Line Plot

sns.lineplot(x=data["ram"],y=data["price"]



As the size of ram increases,price of the Phone also increases.

5 Training the Model

Splitting the data into testing and training sets

```
y = data["price"]
data.drop(["price"],axis=1,inplace=True)
x = data
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.1,random_state=42)
```

✓ 0.0s

Model creation and training

```
lr = LinearRegression().fit(x_train,y_train)
rf = RandomForestRegressor(max_depth=10,n_estimators=100,random_state=1).fit(x_train,y_train)
dt = DecisionTreeRegressor().fit(x_train,y_train)
```

✓ 0.6s

6 Testing and Accuracy Prediction

Testing Linear Regression model with test data

```
print("Linear Regression\n")
print("score: ", lr.score(x_train,y_train))
lr_pred = lr.predict(x_test)
print("r2 score: " , r2_score(y_test,lr_pred))
```

✓ 0.0s

```
Linear Regression

score:  0.701065139742171
r2 score:  -39.26283868118931
```

We can see that the model's initial predictions are not accurate from the testing training score.So the model is under fitted.

Testing the Random Forest model with test data

```
print("Random Forest\n")
print("score: ", rf.score(x_train,y_train))
rf_pred = rf.predict(x_test)
print("r2 score: " , r2_score(y_test,rf_pred))
```

✓ 0.0s

```
Random Forest

score:  0.9721126158373805
r2 score:  0.8658217900727673
```

This models initial predictions and test scores are pretty accurate.

Testing the Decision Tree model with test data

```
print("Decision Tree\n")
print("score: ", dt.score(x_train,y_train))
dt_pred = dt.predict(x_test)
print("r2 score: " , r2_score(y_test,dt_pred))
```

✓ 0.0s

```
Decision Tree

score:  0.9981244520875332
r2 score:  0.7929693552983781
```

We have a high training score and low test score. So this model is slightly over-fitted and not accurate as compared to random forest model

Checking accuracy of Random Forest using real world data

Since the Random Forest algorithm gave the best score compared to other algorithms,We decided to use the Random Forest algorithm.

```
# original price = 18490
a = [2,6.6,8,2.4,13,50,1,1,128,6,6000,0]
a = np.array(a)
a = np.expand_dims(a, 0)
print(round(rf.predict(a)[0],ndigits=2))
```

✓ 0.0s

```
18546.76
```

Specifications of a real phone was entered and an accurate output was obtained.

7 Conclusion

The conclusion of the project is, a machine learning approach was applied to predict the prices of mobile phones. Three models were evaluated: Linear Regression, Random Forest, and Decision Tree. The Random Forest model outperformed the other two models, achieving the highest accuracy and the lowest mean squared error. Cross-validation scores indicated the stability of the models. Overall, the Random Forest model can be considered as a reliable approach for predicting mobile phone prices based on the given dataset.

8 References

[1] Sameer Chand Pudaruth. "Predicting the Price of Used Cars using Machine Learning Techniques", International Journal of Information Computation Technology. ISSN 0974-2239 Volume 4, Number 7 (2014), pp. 753764.

[2] Shonda Kuiper, "Introduction to Multiple Regression: How Much Is Your Car Worth? " , Journal of Statistics Education · November 2008.

[3] Mariana Listiani, 2009. "Support Vector Regression Analysis for Price Prediction in a Car Leasing Application". Master Thesis. Hamburg University of Technology.