Topic: Smartphone price prediction

Author: Juanith Mathew Thomas

Institution: Karunya Institute of Technology and Sciences

Team name: Uno (1 member)

Author Note

This is a brief documentation for the Machine Learning models which were created for predicting the price of smartphones based on the specifications that accompany the device. Instructions on how to analyze the models are present in this document.

Approach

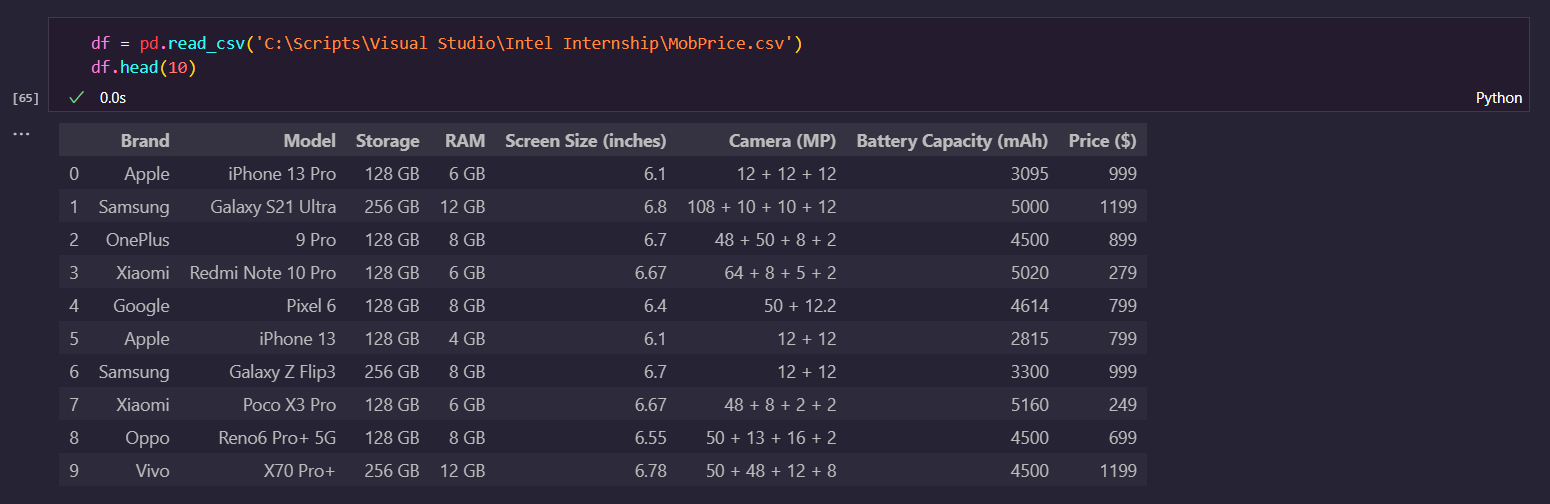
**Preprocessing ----------------------------------------------------------------------------------------------------------------------**

The smartphone price prediction problem can be solved by making use of Machine Learning techniques which can prove to be effective, accurate and suitable for such tasks of an order where patterns tend to be observed.

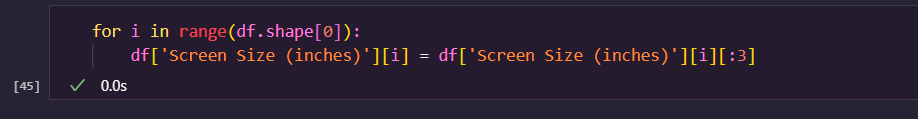
Some of the possible ML architectures that can be used here are:

1. Multiple Linear Regression
2. The decision tree Algorithm
3. The Random Forest Algorithm
4. K-Nearest Neighbors Algorithm

The dataset which was available is, at first glance, a very detailed and precise dataset with multiple factors accounting for the variable change in pricing of smartphones depending upon the various specifications of the handsets. A quick look at the dataset is as follows:



Some preprocessing approaches used were rounding off or minimizing the specificity of the data. For example, rounding off of the screen size of the smartphones to only consider two places after decimals when taking the size into consideration.



Transforming the values to a uniform format was also an instance of removing entropy from the data to provide evenness.

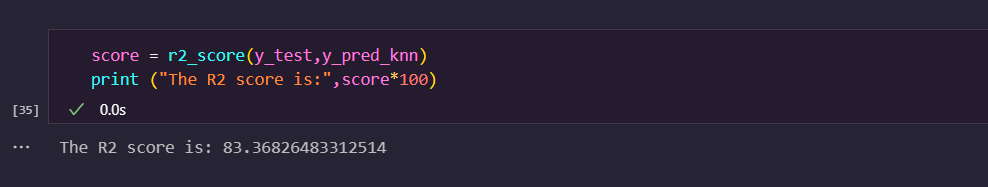


Techniques such as Label Encoding and Standard Scaling were also used to assist with smoother processing of the data. Conversion of different datatypes were done to further ensure uniformity.

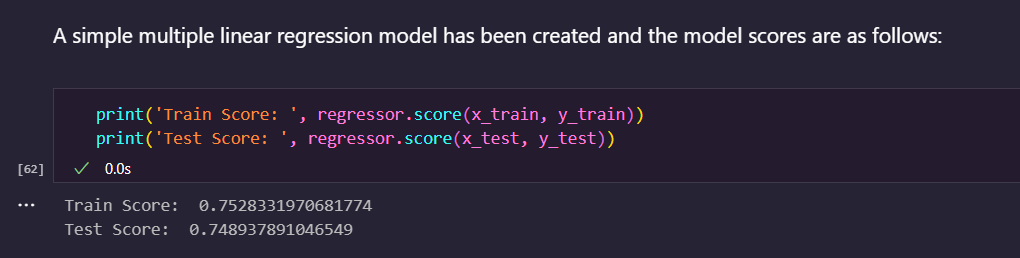
Report

The smartphone price prediction machine learning models are based off of some of the architecture models which include *Multiple Linear Regression (MLR), The Random Forest classifier and the K-Nearest Neighbors model (KNN)*. The following inferences can be made from the obtained results:

1. The preprocessing that is carried out is relatively heavy due to the presence of mixed values in the dataset for specific cells. There are multiple values of the screen size and attributes such as the foldability of the screen that cannot be taken into account by just using a single column.
2. The latter created model, the KNN model works the best among the 2, clearly denoted by the high R2 score of 83.3 percent as displayed in the code output.

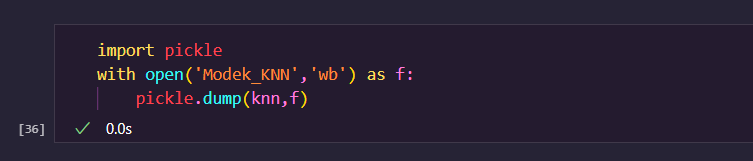


1. Although the simplest solution to this problems appears to be the use of Multiple Linear Regression, it does not grant the best result proven by the inferior result, which was approximately sliding between 75 - 80 percent for the training accuracy and between 65 – 75 percent for the testing accuracy.



1. Algorithms such as the Decision tree algorithm and the Random Forest algorithms would also yield feasible results but they have been omitted due to model sufficiency with MLR and KNN model results.
2. Pickle has been used to create models for easy evaluation of the models. The MLR model has been saved as ‘Model\_MLR’ and the KNN model has been saved as ‘Model\_KNN’ in the

‘Models’ folder.



**References ---------------------------------------------------------------------------------------------------------------------------**

*[1] Multiple linear regression, by Martin Krzywinski & Naomi Altman,* 01 December 2015.

*[2]* [*https://www.sciencedirect.com/science/article/pii/S1877042813046429*](https://www.sciencedirect.com/science/article/pii/S1877042813046429)

*[3]* [*https://youtu.be/KfnhNlD8WZI*](https://youtu.be/KfnhNlD8WZI) *(for using Pickle and Joblib)*

*[4] Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction (for analogical insights and ideas)*