Prediction of Mobile Model Price Using Machine Learning Techniques

Kunal R Kodarlikar

Cohort-Boston

# Abstract: - Mobile phone has become a common commodity and usually the most common purchased item. Thousands of types of mobiles are released every year with new features and new specification and new designs. So, the real question is prediction is that what is the real price of the mobile and to estimate the price of the mobile within the market for optimal marketing and successful launch of the product. Price has become a major factor for development of any product and its sustainability in the market. Mobile prices also impact the marketing of the mobile and also its popularity with other competitors. With the available specifications and desired designs, money is also an important factor to survive within the market. Customer usually sees that they are able to buy with the specification with the given estimated price or not. So, to estimating the price is an important factor before releasing the mobile and also to know about the market and competitors. In this Prediction, Dataset is collected from the existing market and different algorithms are applied to reduce the complexity and also identify the major selection features and get the best comparison within the data. This Tool is used to find the best price with maximum specifications.

***Keywords: -*** *Data Collection, Correlation Analysis, Mobile price range Prediction, Machine Learning.*

# INTRODUCTION

Price is the most effective attribute of marketing and business. The very first question of costumer is about the price of items. All the costumers are first worried and thinks “If he would be able to purchase something with given specifications or not”. So to estimate price at home is the basic purpose of the work. This paper is only the first step toward the above mentioned destination. Artificial Intelligence-which makes machine capable to answer the questions intelligently- now a days is very vast engineering field. Machine learning provides us best techniques for artificial intelligence like classification, regression, supervised learning and unsupervised learning and many more. Different tools are available for machine learning tasks like MATLAB, Python, Cygwin, WEKA etc. We can use any of classifiers like Decision tree, Naïve Bayes and many more. Different type of feature selection algorithms is

available to select only best features and minimize dataset.

This will reduce computational complexity of the problem. As this is optimization problem so many optimization techniques are also used to reduce dimensionality of the dataset. Mobile now a days is one of the most selling and purchasing device. Every day new mobiles with new version and more features are launched. Hundreds and thousands of mobiles are sold and purchased on daily basis. So here the mobile price class prediction is a case study for the given type of problem i.e. finding optimal product. The same work can be done to estimate real price of all products like cars, bikes, generators, motors, food items, medicine etc.

Many features are very important to be considered to estimate price of mobile. For example, Processor of the mobile. Battery timing is also very important in today’s busy schedule of human being. Size and thickness of the mobile are also important decision factors.

Internal memory, Camera pixels, and video quality must be under consideration. Internet browsing is also one of the most important constraints in this technological era of 21st century. And so is the list of many features based upon those, mobile price is decided. So, we will use many of above-mentioned features to classify whether the mobile would be very economical, economical, expensive or very\_ expensive.

# RESEARCH METHODOLOGY

# 

The research was carried out in Google Colab’s Python kernel.

The general workflow diagram of supervised ML tasks

is as follows

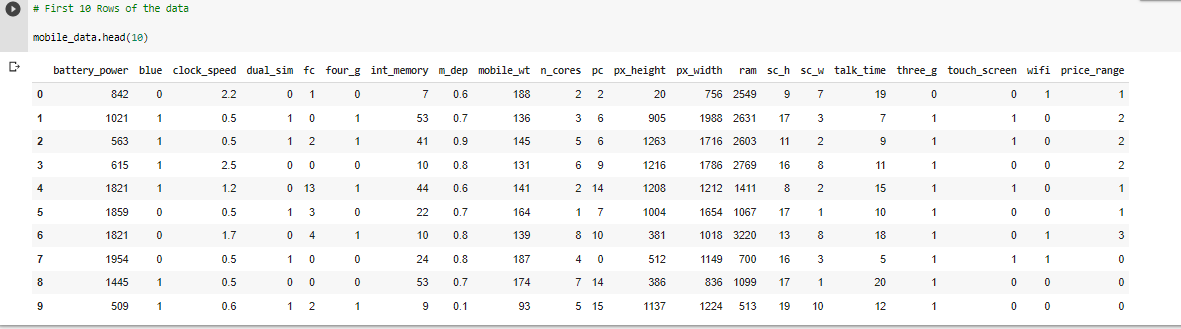


# The dataset is portioned into two – train for training the model and test for its evaluation. The computer tries to comprehend the logic behind the pricing of a mobile based on its features and uses it to forecast future instances as correctly as possible.

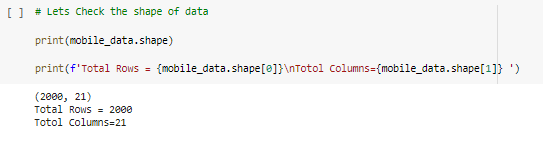
# UNDERSTANDING THE DATASET

The dataset contains 21 attributes in total – 20 features and a class label which is the price range. The features include battery capacity, RAM, weight, camera pixels, etc. The class label is the price range. It has 4 kinds of values – 0,1,2 and 3 which are of ordinal data type representing the increasing degree of price. Higher the value, higher is the price range the mobile falls under.

These 4 values can be interpreted as economical, mid-range, flagship and premium. So, despite price traditionally being a numeric problem, the type of ML is classification (not regression) since there are discrete values in the class label. This is advantageous when using algorithms like Naïve Bayes and Decision Tree as they normally don’t work well with numeric data.



The dataset contains 2000 records in total.





This is the numerical breakdown of the dataset:



# TRAINING THE PREDICATION MODEL

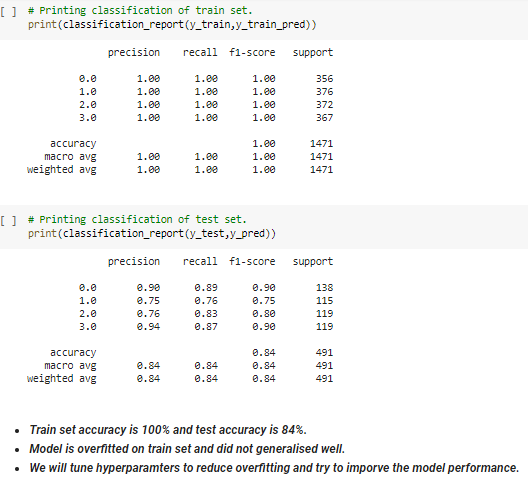
***Algorithms used for predictive modelling*:**

1) Decision Tree  
2) Random Forest classifier  
3) Gradient Boosting Classifier  
4) K-nearest Neighbors classifier  
5) XG Boost Classifier  
6) Support Vector Machine (SVM)

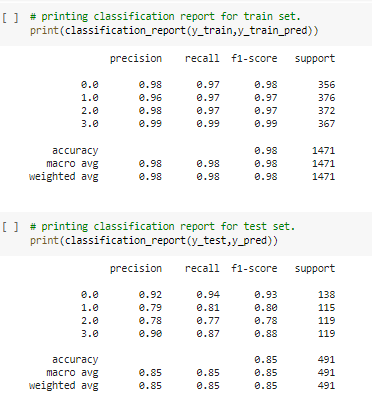
***Decision Tree:***

Decision trees and ensemble methods do not require feature scaling to be performed as they are not sensitive to the variance in the data. So here we will use X\_train,X\_test,y\_test and Y\_train which are not scaled.

**With default hyperparameters**:



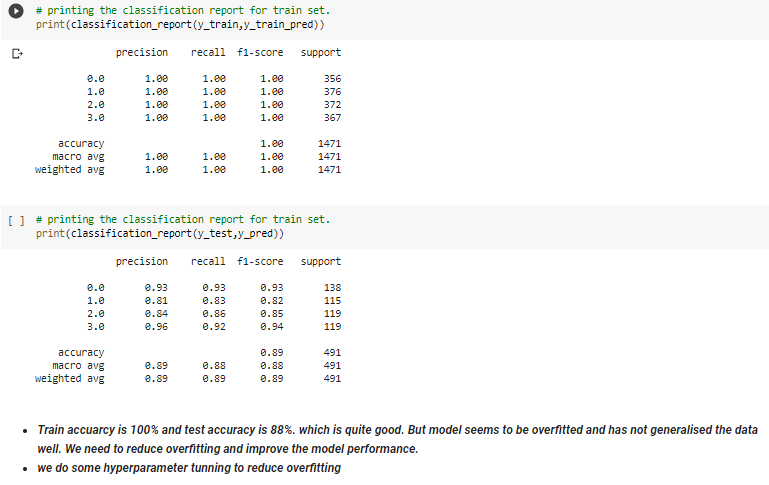
**Let's tune some hyperparameters of Decision Tree classifier:**



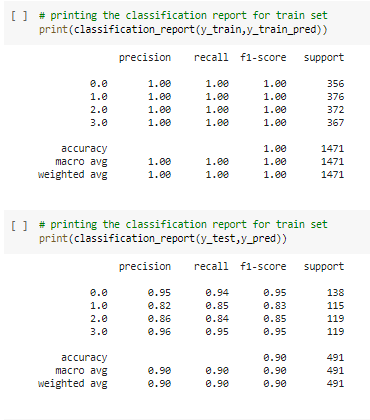
* Decision Tree Classifier-Observations:
* Train accuracy has been reduced to 98% from 100% and test accuracy is increased by 1% . Thus we somewhat reduced the overfitting by reducing the training accuracy . However this will not be good model for us.
* RAM, battery power, px\_height and width came out to be the most important features
* This model classified the class 0 and class 3 very nicely as we can see the AUC is almost 0.96 for both classes, whereas for class 1 and class 2 it is 0.88.

***Random Forest classifier:***

**With default hyperparameters**:



### **Let's do some Hyperparameter Tunning of the Random Forest model**

******

Before Tuning

* training accuracy = 100%
* test accuracy = 88%

Model is overfitted the data and does not generalized well. So we tuned the hyperparameters.

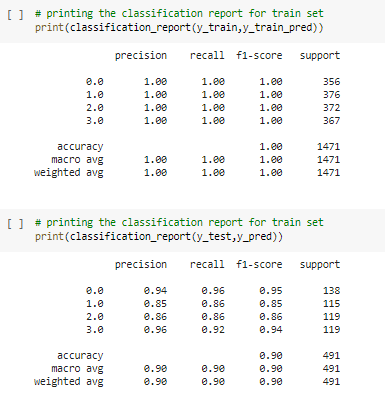
After tuning:

* Training accuracy = 100%
* Test accuracy = 90%

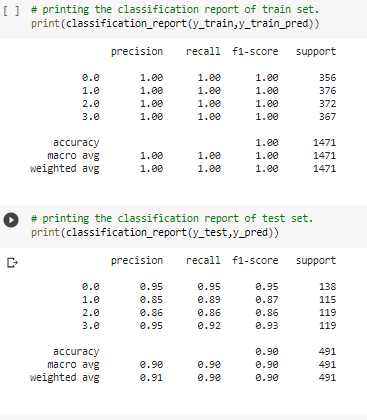
we have slightly improved the model and overfitting is reduced slightly. From roc curve it’s clear that model has poorly performed to classify class 1 and class 2.

***Gradient Boosting Classifier:***

**With default hyperparameters**:

****

**Let's tune some hyperparameters of *Gradient Boosting Classifier :***

****

Before tunning:

* Train accuracy score= 100%.
* Test accuracy score= 89%

Model did not generalised well and overfitted the training data. so we tuned hyperparameters of model.

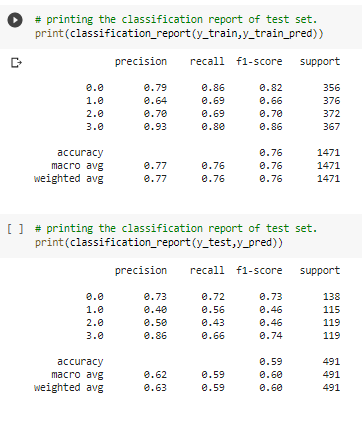
After Hyperparameter Tuning

* Train accuracy score= 100%
* Test accuracy score=90%

Thus we slightly improved the model performance. However, the model is not best.

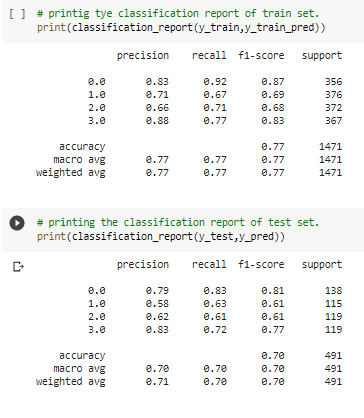
From ROC curve it's clear that model was good to classify the class 0 and class 3.From the classification report its clear that recall for class 0 and class 3 is also good which is 96% and 90% respectively.

# *K Nearest Neighbors:* With default hyperparameters:

****

**Let's tune some hyperparameters of *Gradient Boosting***

***Classifier :***

****

Before hyperparameters tuning:

* Train Accuracy:75 %
* Test Accuracy :59 %

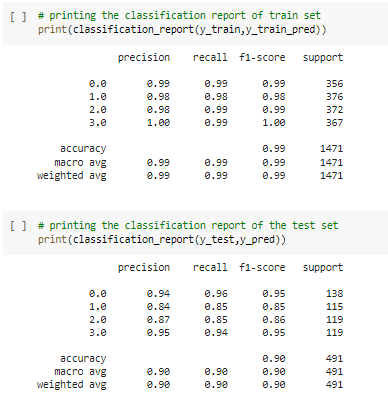
Clearly Model has performed very worst. We did hyperparameter tuning

After Hyperparameter Tuning:

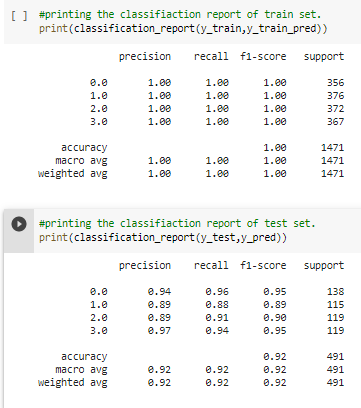
* Train Accuracy : 77%
* Test Accuracy : 70%

Surely we improved the model performance and reduced overfitting but however this is not good model for us***.***

# *XG Boost Classifier:* With default hyperparameters:

****

**Let's tune some hyperparameters of *XG Boost Classifier:***

****

Before hyperparameter Tuning

* Train Accuracy = 98%
* Test Accuracy = 90%

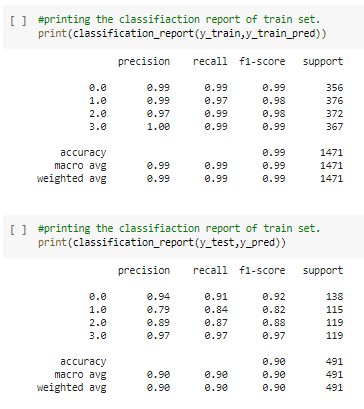
After hyperparameter Tuning

* Train Accuracy = 1%
* Test Accuracy = 92%

we have improved the model performance by Hyperparameter tuning. Test accuracy is increased to 92%.But still the difference of accuracy score between train and test is more than 5%.We can say model is very slightly overfitted

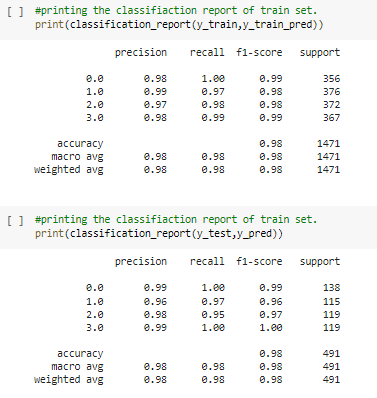
From AUC-ROC curve its clear that model has almost correctly predicted the class 0 and class 3.

# *Support Vector Machine (SVM):* With default hyperparameters:

****

**Let's tune some hyperparameters of *Gradient Boosting***

***Classifier:***

****

* Accuracy score on train set is 98.5% and Test score is 89%.
* Model seems to be overfitted as the difference between train and test accuracy score is almost 10%.\*
* After Hyperparameter tuning train accuracy remained almost same 98.3% and test accuracy score increased to 97%.
* SVM performed very well as compared to other algorithms.
* In terms of feature importance RAM, Battery power,px\_height and px\_weight are the important features.
* f1 score for individual classes is also very good. Area under curve for each class prediction is also almost 1.

# CHALLANGES

Below mentioned challenges were considered while predicting the mobile price range model:

1. **Data preparation:** One of the most frequently overlooked challenges of predictive modeling is acquiring the correct amount of data and sorting out the right data to use when developing algorithms. Once the data has been sorted, one must be careful to avoid overfitting. Over-testing on training data can result in a model that appears very accurate but has memorized the key points in the data set rather than learned how to generalize.

2. **Technical and cultural barriers**: While predictive modeling is often considered to be primarily a mathematical problem, users must plan for the technical and organizational barriers that might prevent them from getting the data they need. Often, systems that store useful data are not connected directly to centralized data warehouses. Also, some lines of business may feel that the data they manage is their asset, and they may not share it freely with data science teams.

3. **Choosing the right business case:** Another potential obstacle for predictive modeling initiatives is making sure projects address significant business challenges. Predictive modeling initiatives need to have a solid foundation of business relevance.

Hence, we Implemented various classification algorithms, out of which the SVM (Support vector machine) algorithm gave the best performance after hyper-parameter tuning with 98.3% train accuracy and 97 % test accuracy.

# CONCLUSION

The model trained using LDA was found to predict mobile price classes most accurately (95%). The accuracy of the models can be improved by doing some data preprocessing steps like normalization and standardization. Feature selection and extraction algorithms can be used to remove unsuitable and duplicative features to get better results. The same procedure used in this paper can be applied to predict the prices of other products like cars, bikes, houses, etc. using the archival data containing features like cost, specifications, etc. This would help organizations and consumers alike to make more educated decisions when it comes to price.