Mobile Price Range Prediction

Kajal Dhun, Navinkumar Sambari, Tanu Rajput,
Data Science Trainees, Almabetter, Bangalore.

Abstract

During the purchase of mobile phones, various features like memory, display, battery, camera, etc., are considered. People fail to make correct decisions, due to the non-availability of necessary resources to cross-validate the price. To address this issue, a machine learning model is developed using the data related to the key features of the mobile phone. The developed model is then used to predict the price range of the new mobile phone. Five machine learning algorithms viz., Support Vector Machine, Random Forest Classifier, Logistic Regression, K Nearest Neighbor Classifier, Decision Tree Classifier are used to train the model and predict the output as low, medium, high or very high. In order to improve the classification accuracy, Chi-squared based feature viz., RAM, pixel height, battery power, pixel width, mobile weight, internal memory, screen width, talk time, front camera and screen height are selected and used to train the model. Before applying feature selection, the accuracy obtained using SVM, RFC, LR, DT, and KNN is 96%, 87%, 86%, 84%, 63% respectively. From the experiments conducted. It is found that SVM and LogisticRegression gave superior performance. And we chose SVM for price range prediction

Keywords: ***memory, mobile phones, SuperVectorMachine, Logistic Regression, KNN, accuracy, prediction.***

Problem Statement

As we all know, mobile phones are a very essential part of our life,But how to make mobile prices for higher demand.so we are making a price range predictor by which we categorize the price in four ranges which are-low,medium,high,very high based on their features. In the competitive mobile phone market companies want to understand sales data of mobile phones and factors which drive the prices. The objective is to find out some relation between features of a mobile phone(eg:- RAM, Internal Memory, etc) and its selling price. In this problem, we do not have to predict the actual price but a price range indicating how high the price is.

Introduction

Mobile phones come in all sorts of prices, features, specifications and all. Price estimation and prediction is an important part of consumer strategy. Deciding on the correct price of a product is very important for the market success of a product. A new product that has to be launched, must have the correct price so that consumers find it appropriate to buy the product.The various features and information can be used to predict the price range of a mobile phone,which are – Battery Power in mAh, BlueTooth or not, Microprocessor clock speed,The phone has dual sim support or not, Front Camera Megapixels, Has 4G support or not.

Project Steps Involved

Preprocessing

Data preprocessing is a data mining technique which consists in transforming the data in order to make it understandable. In machine learning, the data preprocessing step is critical because it involves cleaning, integration, transformation, scaling, standardizing data and many other tasks, in order to have a good preparation for the application of models.

In this part we do the following tasks

1. Dropping px_height(pixel resolution height) having value 0.
2. Replacing all nan values in sc_w(screen width) columns to 0.

Exploratory Data Analysis

After preparing the data set we got our data frame so we perform EDA for price range based on many features.

After that we see some percentage distribution on features supported or not in mobile with respect to price range to see how the price affected.

Standardizing the feature

We used the Standard Scaler technique to standardize the features. Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.

Fitting the models using classification algorithms

We have a classification problem because our target is the price range of mobiles. So the goal of this part is to apply many algorithms in order to find the algorithm with the best predictor.

For modeling we tried various classification algorithms like:

1. **Logistic Regression**
2. **DecisionTreeClassifier**
3. **KNearestNeighborClassifier**
4. **SuperVectorMachine**
5. **RandomForestClassifier**

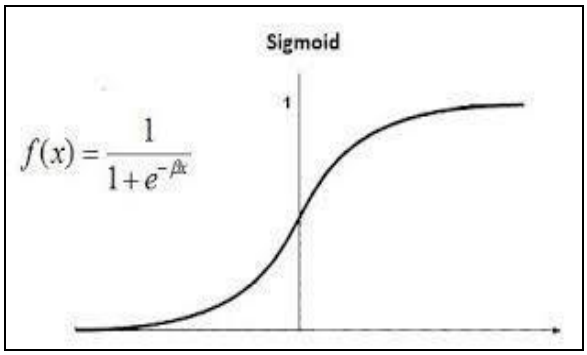
Hyperparameter tuning for better accuracy

Tuning the hyper parameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting in the case of tree-based models like DecisionTree, Random Forest Regression and SupervectorMachine.

Classification Algorithms used

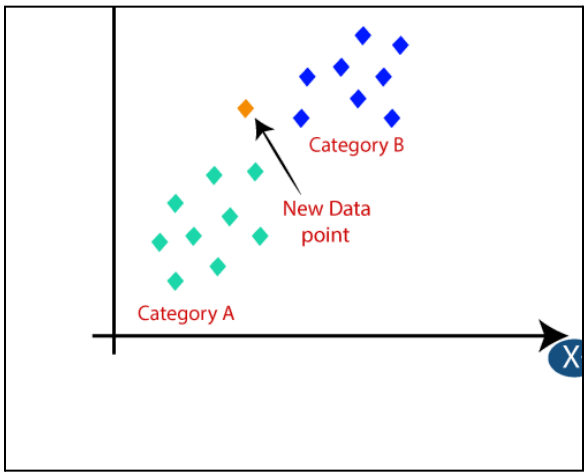
1. Logistic Regression:

Logistic regression is **a statistical model that** in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).



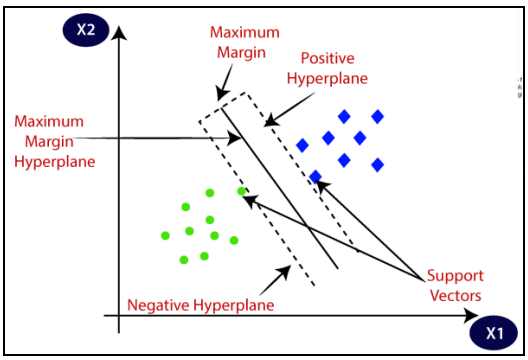
2. K-Nearest Neighbor:

The K-Nearest Neighbors (KNN) algorithm is a data classification method for estimating the likelihood that a data point will become a member of one group or another based on what group the data points nearest to it belong to. The k-nearest neighbors algorithm is a type of supervised machine learning algorithm used to solve classification and regression problems. However, it's mainly used for classification problems.



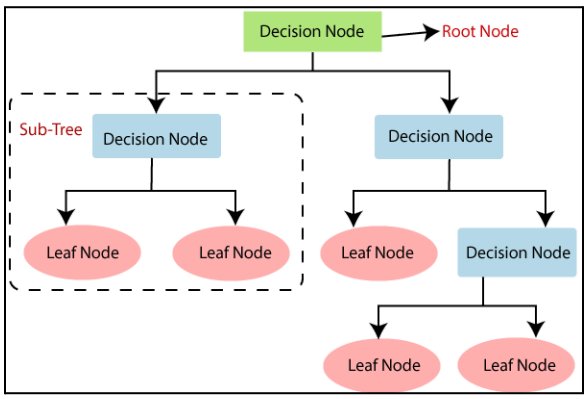
3. SuperVectorMachine:

SVM is a supervised machine learning algorithm which can be used **for classification or regression problems**. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.



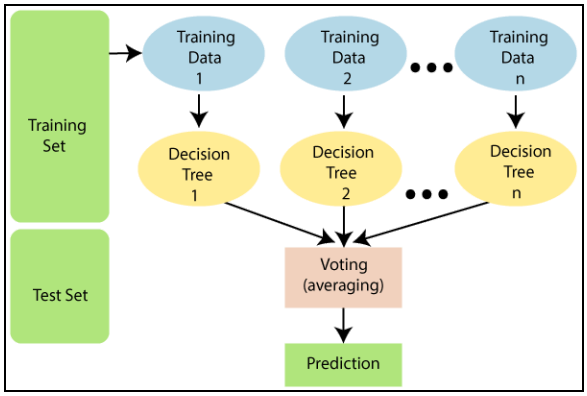
4. DecisionTree Classifier:

A decision tree is a graphical representation of possible solutions to a decision based on certain conditions. It's called a decision tree because it starts with a single box (or root), which then branches off into a number of solutions, just like a tree.



5. Random Forest:

Random Forest is a supervised learning algorithm that uses ensemble learning methods for regression. ... A Random Forest operates by constructing several decision trees during training time and outputting the mean of the classes as the prediction of all the trees.



Hyperparameter Tuning

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects the performance, stability, and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV, for hyperparameter tuning. This also results in cross-validation.

- **Grid Search CV:**

  Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model's performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

Model Performance

Model can be evaluated by various metrics such as:

1. **Accuracy Score**
2. **Precision**

3. **Recall**
4. **F1 Score**
5. **Confusion Matrix**
6. **Training and testing score**

Optimization

Before moving onto performance metrics, let's discuss optimization. What metric exactly are we optimizing? In this case, we are optimizing recall.

Ideally, we do not want to allow any defaults to fall through the cracks, so our optimal model will minimize False Negatives (So RecallScore is as high as possible).

Challenges Faced

- Faced some challenge while handling mismatch values in few columns
- We felt a little challenging when we started working on different algorithms and its metrics and also choosing quite a number of algorithms to work upon.
- Deciding about the best model for prediction among LogisticRegression and SVM wrt their evaluation metrics.

Conclusion

That's it! We reached the end of our documentation

➢ Starting with loading the data so far we have done EDA, Outliers treatment, encoding of categorical columns, feature selection, and then model building.

➢ In all of these models, our recall scores revolve in the range of 90% to 96%.

➢ And there is some amount of improvement in the recall score after hyper parameter tuning.

- So the recall score of our best model (SVM) is 95% which can be said to be good for this large dataset. So, SVM is the algorithm we chose for a predictive model to predict the mobile price range.