MOBILE PRICE RANGE PREDICTION

Rushikesh Arjun Mane

Data Science Trainee

Almabetter, Bangalore

**Absract:**

In this Modern Era, Smartphones are an integral part of the lives of human beings. When a smartphone is purchased, many factors like the Display, Processor, Memory, Camera, Thickness, Battery, Connectivity and others are taken into account. One factor that people do not consider is whether the product is worth the cost. As there are no resources to cross validate the price, people fail in taking the correct decision. This paper looks to solve the problem by taking the historical data pertaining to the key features of smartphones along with its cost and develop a model that will predict the approximate price of the new smartphone with a reasonable accuracy. The dataset used for this purpose has taken into consideration 21 different parameters for predicting the price of the phone . Random Forest Classifier, Support Vector Machine , Decision Tree, Gradient Boosting classifier, K-nearest Neighbour Classifier and XG Boost classifier have been used primarily. Based on the accuracy , the appropriate algorithm has been used to predict the prices of the smartphone. This not only helps the customers decide the right phone to purchase , it also helps the owners decide what should be the appropriate pricing of the phone for the features that they offer. This idea of predicting the price will help the people make informed choice when they are purchasing a phone in the future. Among the all classifiers chosen Support Vector Machine had highest accuracy of 98%.

**Problem Statement:**

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.

**Data Description -**

* **Battery\_power** - Total energy a battery can store in one time measured in mAh
* **Blue** - Has bluetooth or not
* ***Clock\_speed*** - speed at which microprocessor executes instructions
* ***Dual\_sim*** - Has dual sim support or not
* ***Fc*** - Front Camera mega pixels
* ***Four\_g*** - Has 4G or not
* ***Int\_memory*** - Internal Memory in Gigabytes
* ***M\_dep*** - Mobile Depth in cm
* ***Mobile\_wt*** - Weight of mobile phone
* ***N\_cores*** - Number of cores of processor
* ***Pc*** - Primary Camera mega pixels
* ***Px\_height*** - Pixel Resolution Height
* ***Px\_width*** - Pixel Resolution Width
* ***Ram*** - Random Access Memory in Mega Bytes
* ***Sc\_h*** - Screen Height of mobile in cm
* ***Sc\_w*** - Screen Width of mobile in cm
* ***Talk\_time*** - longest time that a single battery charge will last when you are
* ***Three\_g*** - Has 3G or not
* ***Touch\_screen*** - Has touch screen or not
* ***Wifi*** - Has wifi or not
* ***Price\_range*** - This is the target variable with value of
* 0(low cost),
* 1(medium cost),
* 2(high cost) and
* 3(very high cost).
* Thus our target variable has 4 categories so basically it is a Multiclass classification problem.

**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 and many more. Different type of feature selection algorithms are 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 mobile 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 todays 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 Low cost, Medium cost, High cost and Very High cost.

**Analysis Steps Involved:**

1. **EXAMINING NULL / MISSING VALUES:**

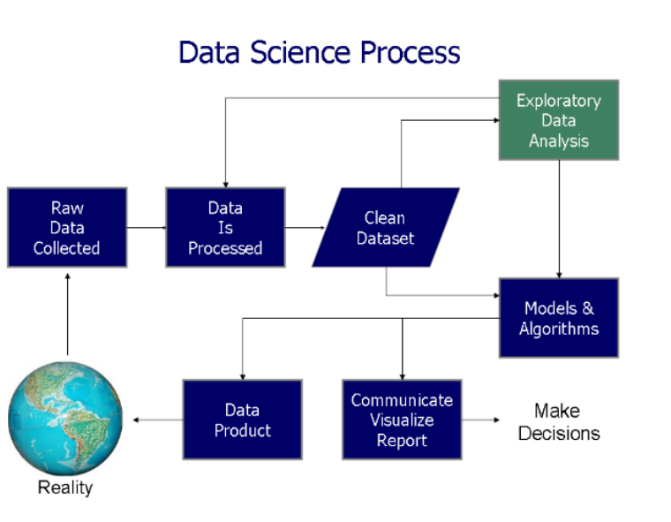
Null values are a big problem in machine learning and deep learning. If you are using sklearn, TensorFlow, or any other machine learning or deep learning packages, it is required to clean up null values before you pass your data to the machine learning or deep learning framework. Otherwise, it will give you a long and ugly error message. So we are checking for null/ missing values. There is no missing value and no null value in provided dataset.

1. **DATA CLEANING** :

Data cleaning is the foremost step in any data science project. No data is clean, but most is useful. Data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. To begin with our data cleaning, first we check for duplicate values and there is no duplicate values in given dataset. After doing so we are converting datatypes, and then we have done exploratory data analysis and find best fit model of dataset.

1. **EXPLORATORY DATA ANALYSIS :**

In statistics, exploratory data analysis (EDA) is an approach of analyzing data sets to summarize their main characteristics, often using statistical graphics and other data visualization methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling and thereby contrasts traditional hypothesis testing. EDA is helped us figuring out various aspects and relationships among the target and the independent variables.



1. **Feature Selection :**

In these steps we used algorithms like ExtraTree classifier to check the results of each feature i.e which feature is more important compared to our model and which is of less importance. Next we used Chi2 for categorical features and ANOVA for numerical features to select the best feature which we will be using further in our model.

1. **Fitting different models:**

For modelling we tried various classification algorithms like:

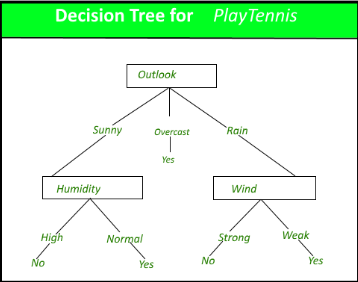
* **Decision Tree**
* **Random Forest Classifier**
* **Gradient Boosting Classifier**
* **K-nearest Neighbor Classifier**
* **XG Boost Classifier**
* **Support Vector Machine (SMV)**

1. **Tuning the hyperparameters for better accuracy:**

Tuning the hyperparameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting in case of tree based models like Random Forest Classifier and XGBoost classifier.

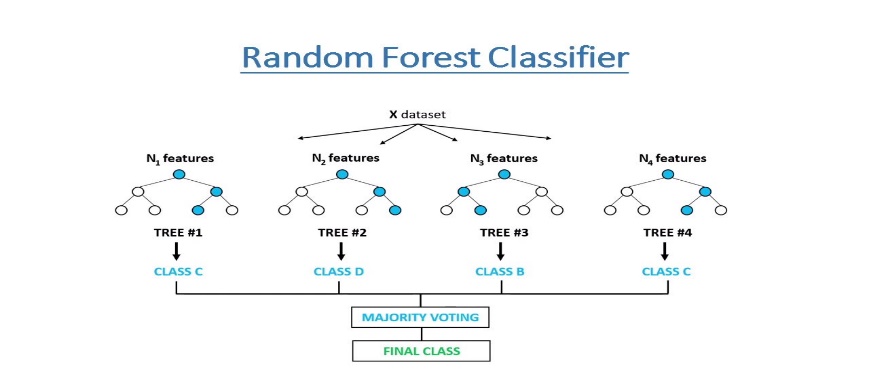
1. **Algorithms:**
2. **Decision Tree:**

**Decision Tree** is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.



1. **Random Forest Classifier:**

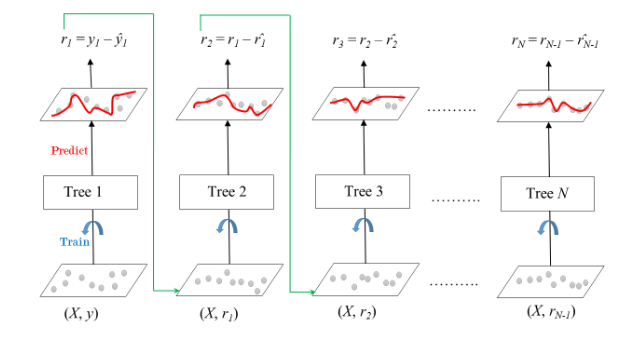
**Random Forest** is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of all.



1. **Gradient Boosting Classifier:**

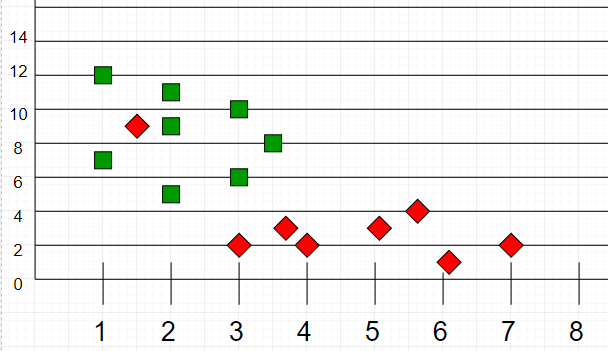
**Gradient Boosting** is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor’s error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

There is a technique called the **Gradient Boosted Trees** whose base learner is CART (Classification and Regression Trees).



1. **K-Nearest Neighbor Classifier:**

**K-Nearest Neighbours** is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.  
It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as [GMM](https://en.wikipedia.org/wiki/Mixture_model), which assume a Gaussian distribution of the given data).  
We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute.

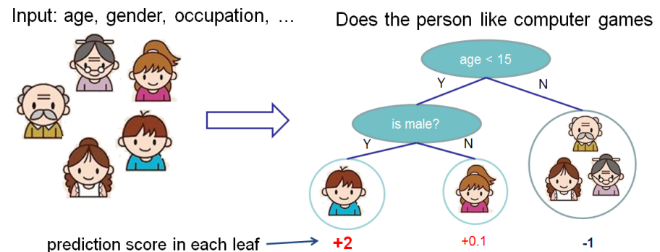


1. **XGBoost Classifier:**

To understand XGBoost we have to know gradient boosting beforehand.

* **Gradient Boosting-**

Gradient boosted trees consider the special case where the simple model is a decision tree

****

In this case, there are going to be 2 kinds of parameters P: the weights at each leaf, w, and the number of leaves T in each tree (so that in the above example, T=3 and w=[2, 0.1, -1]).

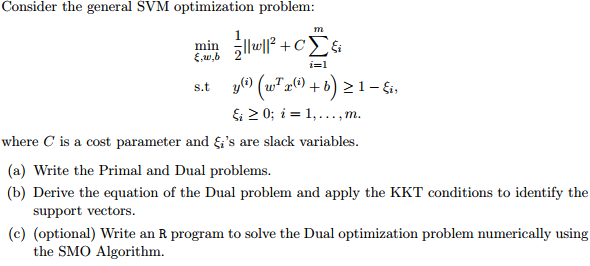
When building a decision tree, a challenge is to decide how to split a current leaf. For instance, in the above image, how could I add another layer to the (age > 15) leaf? A ‘greedy’ way to do this is to consider every possible split on the remaining features (so, gender and occupation), and calculate the new loss for each split; you could then pick the tree which most reduces your loss.

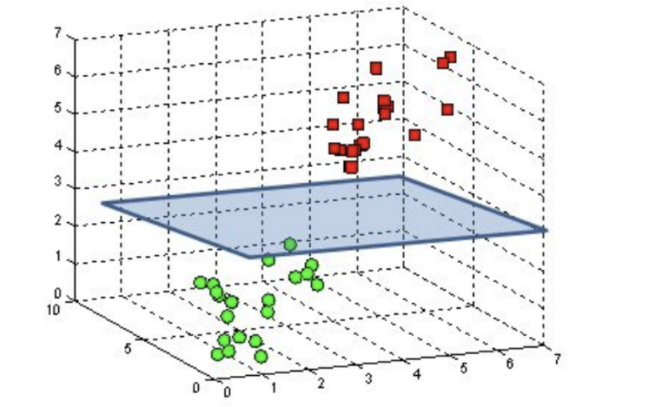
**XGBoost** is one of the fastest implementations of gradient boosting. trees. It does this by tackling one of the major inefficiencies of gradient boosted trees: considering the potential loss for all possible splits to create a new branch (especially if you consider the case where there are thousands of features, and therefore thousands of possible splits). XGBoost tackles this inefficiency by looking at the distribution of features across all data points in a leaf and using this information to reduce the search space of possible feature splits.

1. **Support Vector Machine Classifier:**

SVM is used mostly when the data cannot be linearly separated by logistic regression and the data has noise. This can be done by separating the data with a hyperplane at a higher order dimension.

In SVM we use the optimization algorithm as:





We use hinge loss to deal with the noise when the data isn’t linearly separable.

Kernel functions can be used to map data to higher dimensions when there is inherent non linearity.

**7.2. Model performance:**

Model can be evaluated by various metrics such as:

1. **Confusion Matrix-**

The confusion matrix is a table that summarizes how successful the classification modelis at predicting examples belonging to various classes. One axis of the confusion matrix is the label that the model predicted, and the other axis is the actual label.

1. **Precision/Recall-**

Precision is the ratio of correct positive predictions to the overall number of positive predictions : TP/TP+FP Recall is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

1. **Accuracy**-

Accuracy is given by the number of correctly classified examples divided by the total numberof classified examples. In terms of the confusion matrix, it is given by: TP+TN/TP+TN+FP+FN

1. **Area under ROC Curve(AUC)**-

ROC curves use a combination of the true positive rate (the proportion of positive examples predicted correctly, defined exactly as recall) and false positive rate (the proportion of negative examples predicted incorrectly) to build up a summary picture of the classification performance.

**7.3. Hyper parameter 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 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, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

1. **Grid Search CV-**Grid 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.
2. **Randomized Search CV-** In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control

**Conclusion:**

Huush….. Finally That’s it !! We got to the ending point. Its climax time for our story.

From the starting of data getting loading for analysis, finding mismatched values, handling null and duplicate values, Exploratory Data Analysis, Feature selection and the last but not the least building models. For all algorithms or models our accuracy stays between 80 to 90% and its improves after the hypermeter tunning.

So the accuracy for the best fitted model is 97.96 % which we can say it is good enough for this dataset.

**References:**

* Keggle.com
* <https://www.geeksforgeeks.org/boosting-in-machine-learning-boosting-and-adaboost/?ref=lbp>
* Analytics Vidhya
* https://online.stat.psu.edu/stat200/lesson/3/3.2#:~:text=We%20can%20use%20the%20IQR,add%20this%20value%20to%20Q3.