BRAC
UNIVERSITY

Inspiring Excellence

CSE422: Artificial Intelligence

Project Name: Mobile Price Prediction

Group: 07

Section: 11

Group Members :

No.	Name	ID
1	Rifai Rahman	19201013
2	Abrar Jamshed	19201002
3	Sajid Rashid	20101163
4	Nafisa Nawal	20101353

Introduction

Mobile phones are one of the most popular devices on the market. To keep up with the world, it is a necessity. However, because everyone has a different level of financial freedom, they all like to get a phone that fits within their price range. However, it can be challenging to focus on a price range while still getting the features that are necessary. Actually, a lot of people would find it useful to be able to create a budget and then save for it. Therefore, with the aid of machine learning, this system of mobile price prediction has been created to categorize the prices of a phone, which will offer customers an idea of how much money to set aside for a phone and an idea of how much a phone with the features they want will likely cost. This illustrates how far the phone can go based on several factors, such as the front camera, touch screen, centers, battery, clock speed, internal memory, battery capacity, etc.

Methodology

❖ **Dataset description**

The dataset consisted of 20 features with 2000 data points.

- **Source:**

  https://www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification?resource=download&select=train.csv

- **Features**

  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 megapixels

  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 megapixels

  px_height : Pixel Resolution Height

px_width : Pixel Resolution Width

ram : Random Access Memory in Megabytes

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

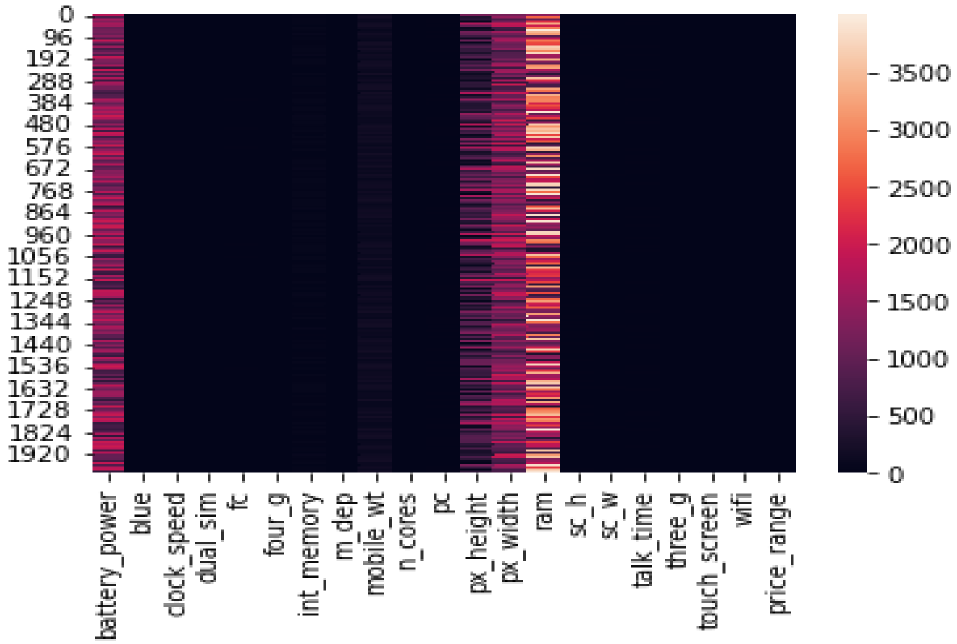
touch_screen : Has touch screen or not

wifi : Has wifi or not

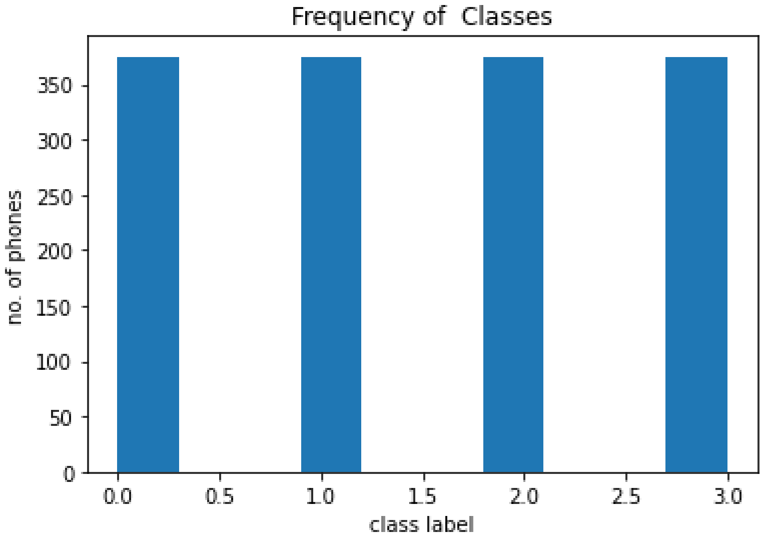
- **Label**

  price_range: This is the target variable with values of 0(low cost), 1(medium cost), 2(high cost), and 3(very high cost), thus making it categorical.

The features are both categorical and quantitative. Categorical data are statistical data that include categorical variables—variables that have been divided into categories—from the original data whereas data that expresses a definite quantity, amount, or range is known as quantitative data. Here, “blue”, “dual_sim”, “four_g”, “three_g”, “touch_screen” and “wifi” are all categorical data which are categorized as “0”s and “1”s which represents “no” and “yes” respectively. The rest of the features are quantitative because they are made up of a range of values.



The bar chart below shows that all the unique classes of the label used had equal number of instances.



Dataset Pre-processing

The dataset used was free of null values and outliers. Built-in methods like data_train.isnull().sum() and data_train.plot(kind='box') have been used to look out for null irreverent values to have them fixed before further working with the data.

Feature Scaling

Scaling the features helps to balance the gradient descent process and speeds up the algorithms' arrival at the cost function minima. Without scaling features, the algorithm might favor the feature with a larger magnitude. The two scaling methods that have been used for the project are illustrated below:

1. MinMax Scaler - It scales values to a range between 0 and 1 and if no negative values are present, and -1 to 1 if negative values are present.
2. Standard Scalar causes the values to be centered around the mean with a unit standard deviation.

MinMax Scalar gave an accuracy score of 0.41 which is less than the score of 0.49 of Standard Scalar. Thus, the Standard Scaling method has been selected for the scaling of the dataset.

Dataset Splitting

We split the dataset into two parts. We set test_size as 0.30 so that 30% of the data is used for testing while the remaining 70% is used for training. We also used a parameter called stratify that takes the target values of the data set. The purpose of stratify is to make sure that features of all kinds of target values/classes are being considered for both training and testing, eliminating the risk of not considering the data of any particular class. Stratify maintains the original dataset's class ratio.

Model Training & Testing

- Decision Tree:

  Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, and since this project is focused on categorical labels, *DecisionTreeCLassifier* from *sklearn* has been imported and calculated and generated using entropy. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

- Logistic regression:

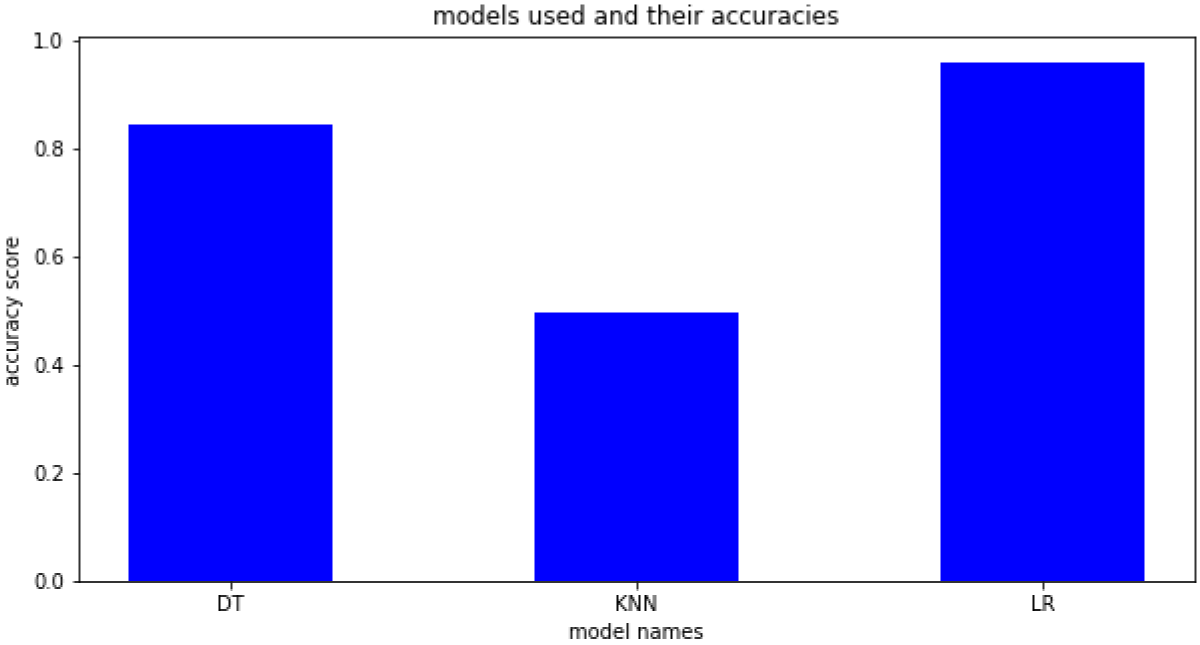
  One of the most often used Machine Learning algorithms, within the category of Supervised Learning, is logistic regression. Here, the categorical dependent variable is predicted using a set of independent variables based on probability. The output of a categorical dependent variable is predicted via logistic regression, representing it in discrete values. For our project, ordinal logistic regression has been used as we are dealing with four ordered dependent variables.

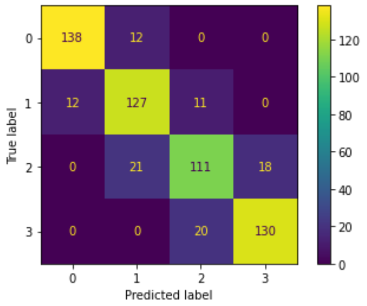
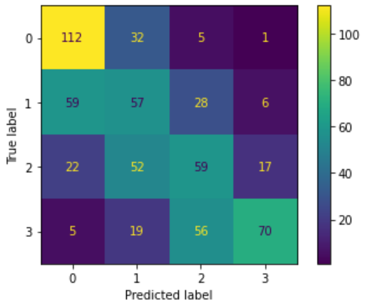
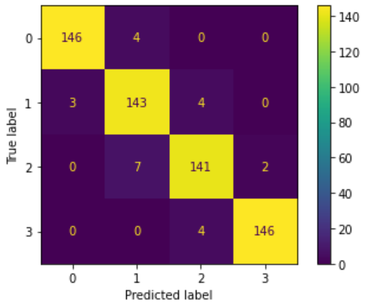
- K-Nearest Neighbours:

  This Machine Learning algorithm is used for classification problems only. It has an unknown parameter called neighbors which takes an input, n. During the testing phase, a testing sample would consider the closest n number of neighbors among the training samples to determine in which class it belongs. In this case, the neighbors of a certain class with maximum population would win, i.e The testing sample would be selected for that class. However, if there is a tie between the training samples, then the sample with the closest distance would win.

Model Training & Testing

The Bar Graph shows the accuracy in prediction of the results when compared to the test data set. From the graph, we can evaluate the accuracy of the models used. For precise evaluation, confusion matrices have been used to compare the actual cost with the predicted cost.



Model	**Accuracy Score**	**Confusion Matrix**
Decision Tree	**85.2%**	
K_Nearest Neighbours	**50%**	
Logistic Regression	**96%**	

Conclusion

With the above mentioned evaluation, we reached the conclusion that when compared to the two approaches indicated above, Logistic Regression would provide a more accurate evaluation as it attained a higher accuracy score.