ME4131D - Machine Learning For Data Science
&
Analytics

Course Project Report

Mobile price prediction using machine learning

A SAI MANOJ - B180161ME
M TEJONATH - B180129ME
K BHARGAV NAGA CHARAN - B180881EE
P BHARATH TEJA - B180953CS

Contents

1 Problem Statement

During purchase of a new mobile, it's often a hard task to decide whether a mobile with particular features is worth of it's price or not. The main theme of this project is to predict price range(not the actual price) of a mobile with given set of features such as battery power, clock speed, internal memory, ram, number of cores etc.,.

2 Literature Review

In a project publication by Arora.P et al [1]. they have collected data which considers factors like monitor size(Inches), the weight(g), the thickness(mm), the internal memory size(GB) etc. ZeroR algorithm in WEKA, Naive Bayes Algorithm, J48 Decision Tree in WEKA are used for predicting the price of mobile. Few research papers estimated the prices of used automobiles in Mauritius. They used a variety of approaches to forecast prices, including multiple linear regressions, k - nearest neighbors (KNN), Decision Tree.

Research paper by N. Priyadarshini et al. [2] has used chi squared based feature selection and considered 10 out of 21 features available are selected namely RAM, pixel height, battery power, pixel width, internal memory etc. Now different algorithms are used for the price predictions namely Support Vector Machine (SVM), Random Forest Classifier (RFC), Logistic Regression. The accuracy obtained using SVM, RFC and Logistic Regression is 95%, 83% and 76% respectively. After feature selection, the accuracy of SVM, RFC and Logistic Regression improved to 97%, 87% and 81% respectively

3 Description of Dataset

The dataset for the preceding problem statement is taken from kaggle[3]. It contains 2000 datapoints and 20 features along with price range as the target variable. The price range(target variable) is classified into 4 categories as shown in Table 1.

Output Class	Categorical representation
Low cost	0
Medium cost	1
High cost	2
Very high cost	3

Table 1: Output classes and their corresponding categorical representations

As mentioned, the dataset contains 20 input features of which 6 are categorical features and rest 14 are continuous features. Both categorical and continuous features are mentioned below in the Table 2

Name of the feature	type of the feature
Has bluetooth or not?	Categorical
Battery Power	Continuous
Clock Speed	Continuous
Has dual sim support or not?	Categorical
Front Camera mega pixels	Continuous
Has 4G or not	Categorical
Internal Memory in Gigabytes	Continuous
Mobile Depth in cm	Continuous
Weight of mobile phone	Continuous
Number of cores of processor	Continuous
Primary Camera mega pixels	Continuous
Pixel Resolution Height	Continuous
Pixel Resolution Width	Continuous
ram	Continuous
Screen Height of mobile in cm	Continuous
Screen Width of mobile in cm	Continuous
How long does battery last on single charge?	Continuous
Has 3G or not?	Categorical
Has touch screen or not?	Categorical
Has wifi or not?	Categorical

Table 2: Features available in Dataset

4 Algorithms and Justifications

Firstly, the above problem statement is purely a classification task. The model has to classify the mobile with given set of features into one of the four predefined classes which are shown in table 1. In this context, we have chosen and experimented with 4 machine learning algorithms:

- Logistic Regression - A basic algorithm used to solve classification tasks. So we started with logistic regression.
- Decision Trees - Decision Trees are usually better than logistic regression algorithm due to their interpretability. The final model can be viewed as a tree and can easily be interpreted. So, decision trees are also considered for experimentation.
- Random Forests - Due to the fact that, random forest is an ensemble of many decision trees, there is a higher chance for better performance. Hence, random forests are also considered for experimentation.

- TabNet [4](a deep neural network) - As we have significant number of datapoints(=2000), we have chosen TabNet[4](a neural network by google specially designed for tabular data) is considered. The advantage of training a deep learning algorithm is that they don't need any kind of feature extraction, rather they learn the important features by themselves. The architecture contains something called "attention mechanism" which makes it even more powerful. It does "self supervised pretraining" where we intentionally miss few values from each row of a table and ask the neural network to predict them back. Performing such a task will make the neural network understand the data distribution better. Cross entropy loss(given in below equation) is used as the criteria for optimizing the neural network.

$$L = \sum GT_i * logP_i$$

where GT_i is ground truth value for class 'i' taking 0 or 1 and P_i is the softmax probability for the i th class.

5 Analysis

In this section, the Exploratory Data Analysis performed on mobile price classification is discussed in detail.

5.1 Univariate Analysis

In univariate analysis each variable is analyzed separately i.e it doesn't involve with causes and relationships among features.

5.1.1 Distribution Plot

Distribution/density plots provide the summary of distributions. It's often useful to know how a particular variable is distributed in the dataset. The distribution plot is usually done for continuous variables. Distribution plots for all 14 continuous features in the dataset are shown in fig 1

5.1.2 Count plot

Count plots are provides number of occurence of the observation/class present in a categorical variable. The 6 count plots for 6 categorical features in the dataset are shown in fig 2

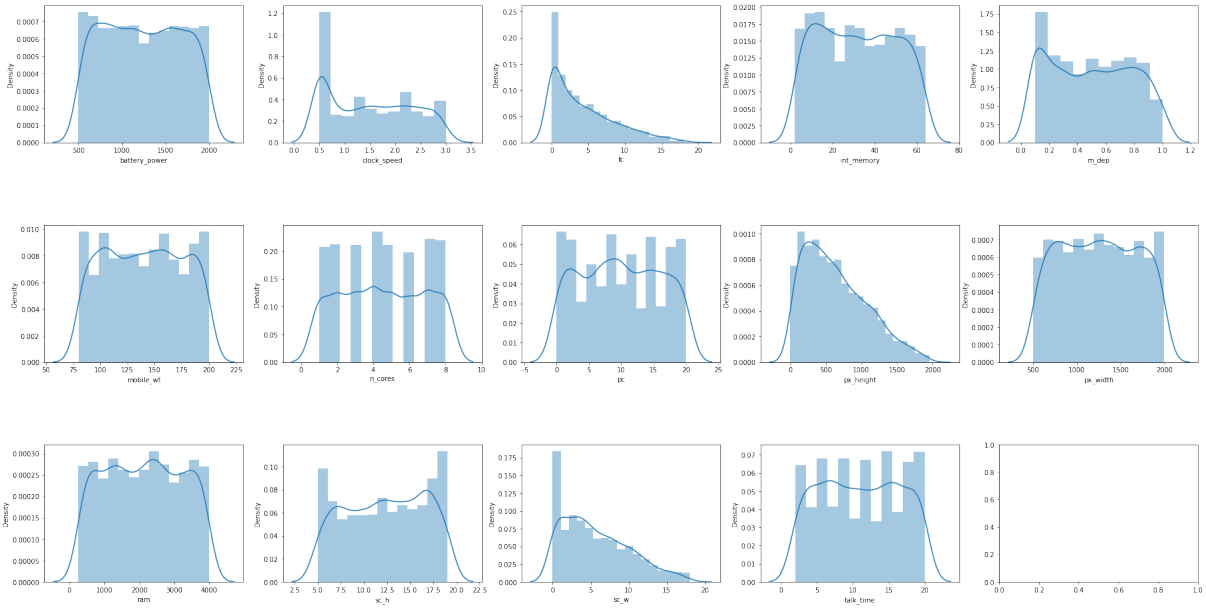

Figure 1: Distribution plots for continuous features in dataset

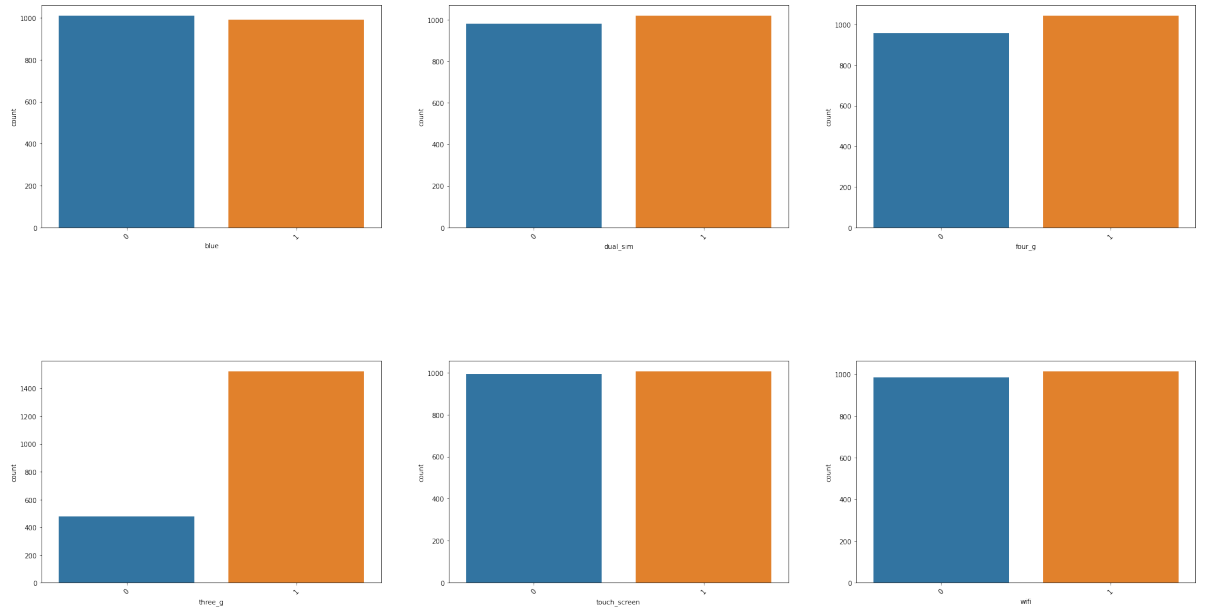

Figure 2: Count plot for categorical variables in the dataset

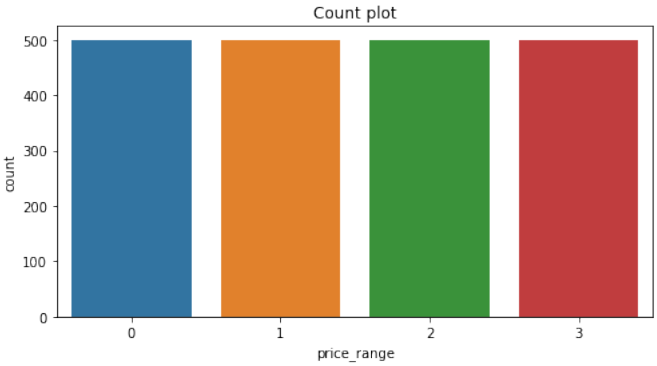

Figure 3: Count plot for target variable(price range)

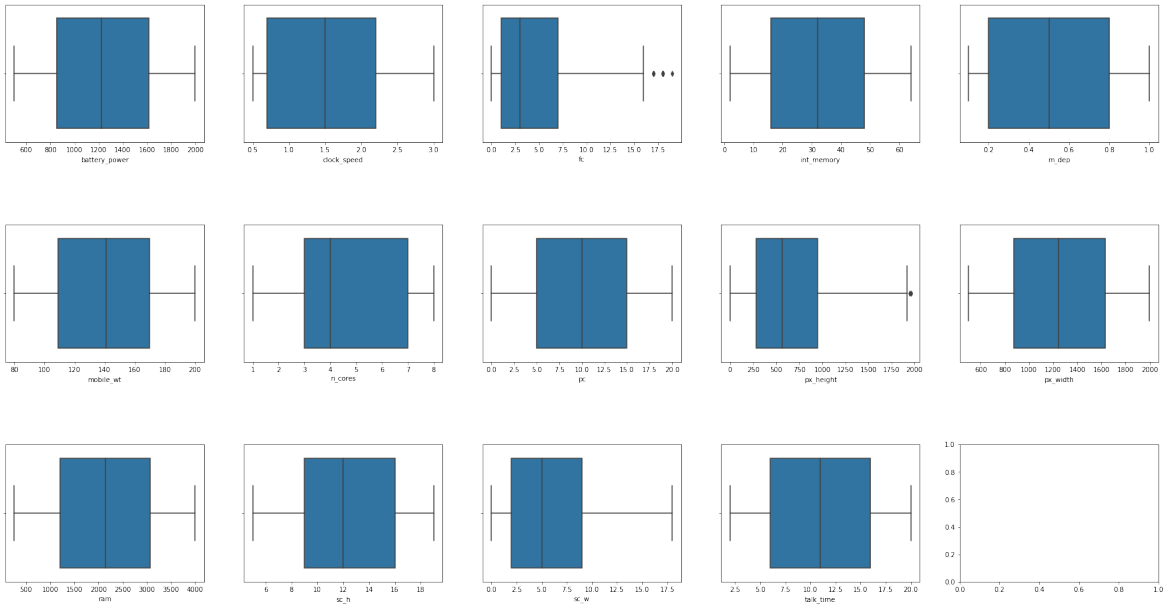

Figure 4: Box plot for continuous features

Since the target variable(price range) is also categorical, count plot is plotted for target variable as well and shown in fig 3. From the count plot, it is clear that the dataset is perfectly balanced i.e number of datapoints from each class is same.

5.1.3 Box plot

Boxplot is used for identifying outliers in the dataset. Typically the box represents the IQR (interquartile range), the center line represents the median, and the whiskers represent some extreme of the data. Box plots for 14 continuous variables are plotted and shown in fig 4

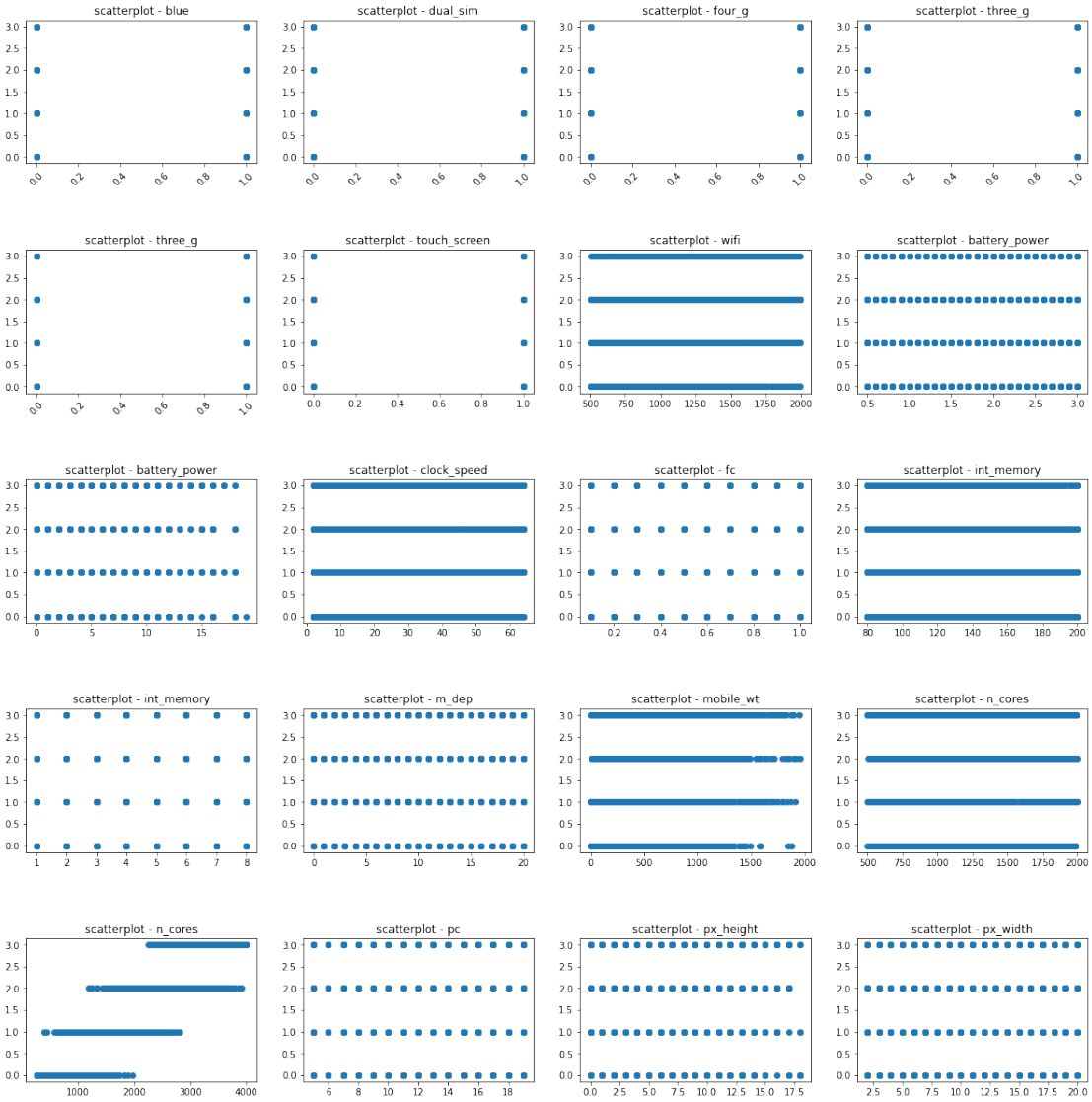

Figure 5: Scatter plot for features in dataset

5.2 Bivariate Analysis

Bivariate analysis carried out among any two variables of dataset to analyze the relationship between them.

5.2.1 Scatter plot

Scatter plots are the graphs that present the relationship between two variables in a data-set. It represents data points on a two-dimensional plane or on a Cartesian system. Scatter plots are plotted for the feature(one at a time) and target variable and shown in fig 5.

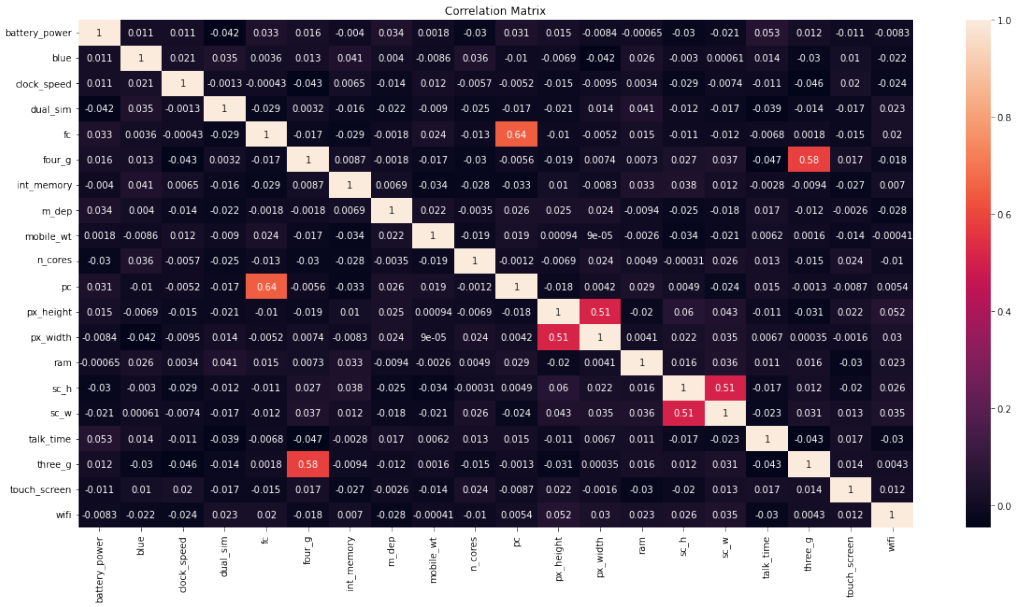

Figure 6: Correlation matrix

5.2.2 Correlation Matrix

Correlation matrix is a matrix which shows the correlation between every possible pair of features in the dataset. Correlation matrix for the features in dataset is plotted and shown in fig 6

6 Results

In this section, results produced using various algorithms are discussed in detail. As mentioned, we used 4 different supervised algorithms to solve the problem statement. The dataset is splitted into train and testsets in a stratified way. 1600 instances are used for training and 400 are reserved for testing.

6.1 Logistic Regression

Logisitic regression algorithm is imported using sklearn library in python. The algorithm trained on train dataset achieved 60% accuracy on testset. Rest of the metrics are shown in Table 4. F1 scores for class 2 and class 3 are very poor as show in Table 3.

6.2 Decision Trees

Sklearn library is used for importing decision trees. 'Gini' criterion is used for training the algorithm. Max depth is set to 100. The algorithm achieved 83% accuracy which is better than

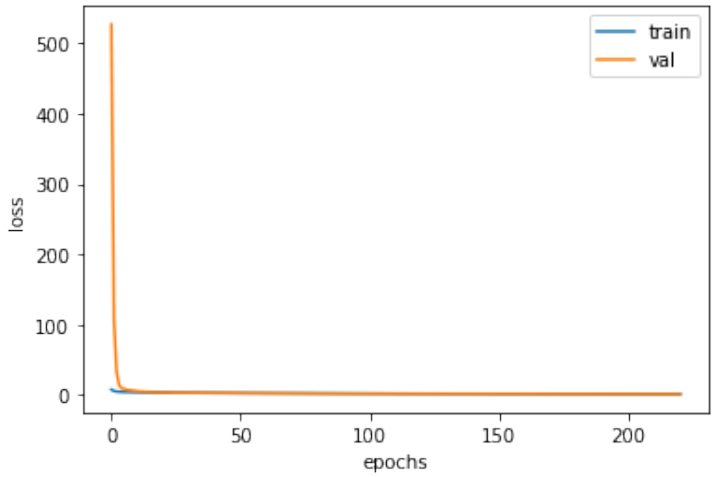

Figure 7: training curve for loss in pretraining phase(tabnet)

the previous logistic regression. F1 scores for class2 and class3 are improved when compared to logistic regression as shown in Table 3. All other metrics are shown in Table 4.

6.3 Random Forests

Random forest is an ensemble of many decision trees. Again, sklearn is used for importing Random forests Criterion is 'Gini' and max depth is set to 100. It achieved an accuracy of 88% which is superior than both decision trees and logistic regression algorithm. F1 scores of all classes have improved by a good margin when compared with logistic regression and decision trees as shown in Table 3. Also, other metrics like precision and recall had also improved as shown in Table 4.

6.4 TabNet

Pytorch framework is used for training the TabNet neural network. we used pytorch_tabnet library to import tabnet in pytorch.

6.4.1 Self supervised pretraining

As mentioned earlier, tabnet needs to be pretrained. The model is pretrained using "Adam" optimizer with learning rate set to 2e-2. It had taken around 220 epochs to converge as shown in the loss curves fig 7.

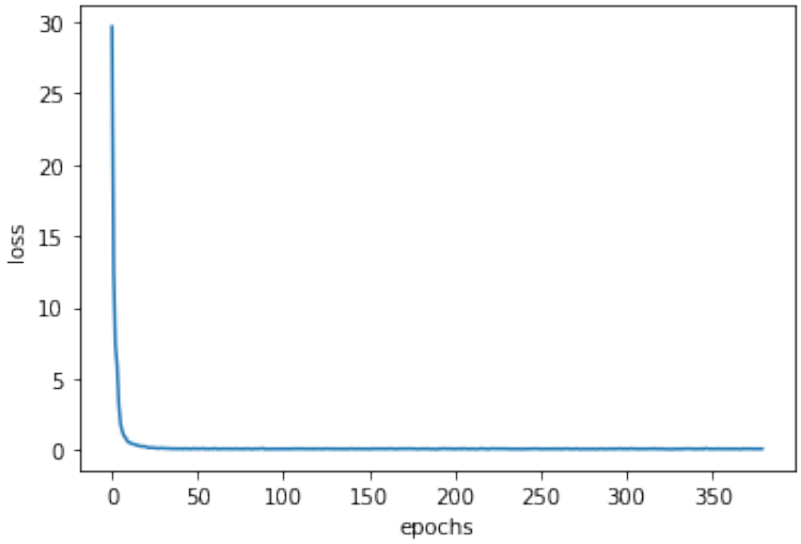

Figure 8: training loss curve in finetuning phase(tabnet)

6.4.2 Finetuning

The pretrained model is trained on 1600 labelled datapoints until it converges. It took 303 epochs to converge for the algorithm. The training curves for loss and accuracy are plotted and shown in fig 8 and fig 9 respectively. After trying out different combinations, we used Adam optimizer with learning rate 5e-2 is used. A learning rate decay scheme is also used where for every 10 epochs, learning rate will be halved as shown in the fig 10. The model has achieved an accuracy of 96% which is far better than all the previous algorithm. Other metrics are shown in Table 4. Individual F1 scores for each class are also increased by large margin as shown in Table 3.

Algorithm	class1	class2	class3	class4
Logistic regression	0.8308	0.5076	0.4174	0.6534
Decision Trees	0.8975	0.7589	0.7729	0.8912
Random forests	0.9552	0.8317	0.8020	0.93
TabNet	0.9644	0.9406	0.9552	0.98

Table 3: Comparison of F1 scores of each class across the algorithms

Algorithm	accuracy	precision	recall	F1 score
Logsitic Regression	60%	0.6052	0.6	0.6023
Decision Trees	83%	0.8319	0.83	0.8302
Random forests	88%	0.8796	0.88	0.8797
TabNet	96%	0.9603	0.96	0.9601

Table 4: Comparison of metrics across the algorithms

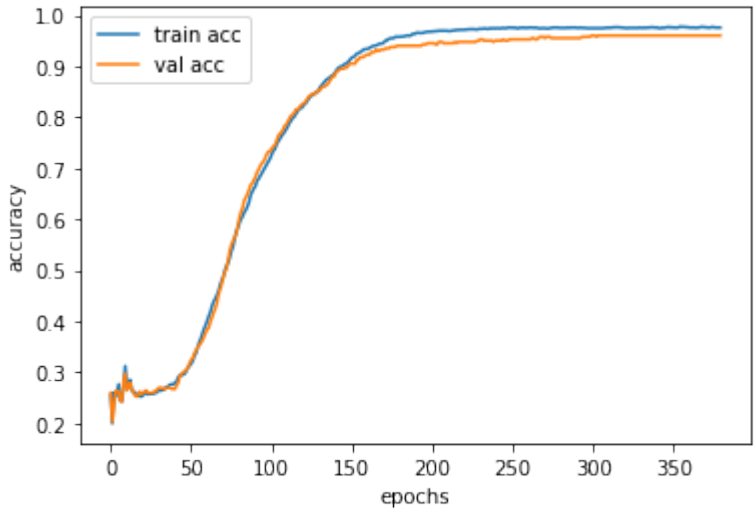

Figure 9: training accuracy curve in finetuning phase(tabnet)

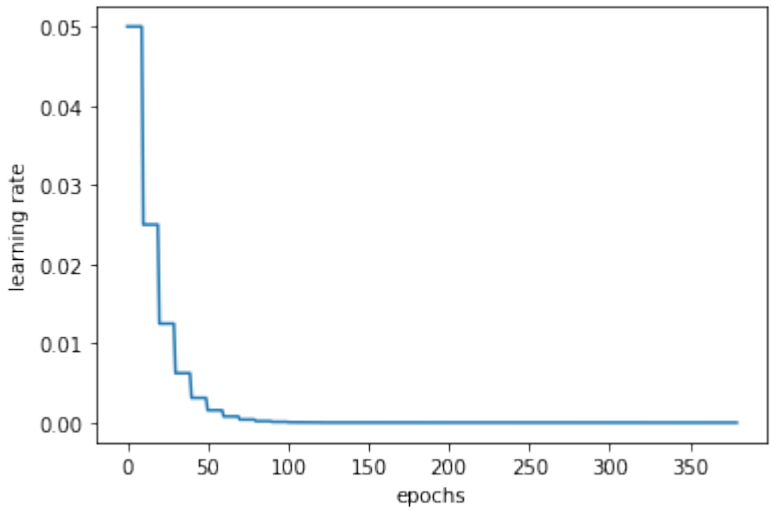

Figure 10: learning rate decay scheme

7 conclusion

Ambiguity in buying mobile is a frequently experienced problem by many users. By leveraging the recent machine learning techniques, we established models to decide the price range given the features of the mobile. Our best model, TabNet with cross entropy as the loss function achieved an accuracy of 96%.

References

[1] B. Arora.P., Srivastava.S., "Mobile price prediction using weka," 2022.

[2] K. S. Kalaivani, N. Priyadharshini, S. Nivedhashri, and R. Nandhini, "Predicting the price range of mobile phones using machine learning techniques," *AIP Conference Proceedings*, vol. 2387, no. 1, p. 140010, 2021. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/5.0068605

[3] "Mobile price classification," https://www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification.

[4] S. O. Arik and T. Pfister, "Tabnet: Attentive interpretable tabular learning," 2019. [Online]. Available: https://arxiv.org/abs/1908.07442