Mobile Price Classification Using Machine Learning

My name is Md. Siddiqur Rahman, from Dhaka, Bangladesh. I am a Machine Learning Intern at Mentorness. I have a BSc in Economics from Jahangirnagar University. Now, I have to classify the price of mobile. This is the second task for the Mentorness Internship Program.

Introduction:

In today's market, mobile phones come in a wide range of prices, each offering different features and specifications. For consumers, selecting a mobile phone that aligns with their budget and requirements can be challenging. To assist consumers in making informed decisions, this project aims to develop a machine learning model that can classify mobile phones into different price ranges based on their features.

Problem Statement:

The task is to build a predictive model that can accurately classify mobile phones into predefined price ranges based on various attributes such as battery power, camera features, memory, connectivity options, and more. The dataset provided contains information about several mobile phones, including their specifications and corresponding price ranges.

Dataset Description:

The dataset comprises the following columns:

- battery_power: Total energy a battery can store in mAh.
- blue: Bluetooth enabled (1 if yes, 0 if no).
- clock_speed: Speed at which microprocessor executes instructions.
- dual_sim: Dual SIM support (1 if yes, 0 if no).
- fc: Front Camera mega pixels.
- four_g: 4G network support (1 if yes, 0 if no).
- int_memory: Internal Memory (in gigabytes).
- m_dep: Mobile Depth in cm.
- mobile_wt: Weight of mobile phone.
- n_cores: Number of cores of the processor.
- pc: Primary Camera mega pixels.
- 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 talking.
- three_g: 3G network support (1 if yes, 0 if no).

- touch_screen: Touch screen support (1 if yes, 0 if no).
- wifi: Wifi connectivity (1 if yes, 0 if no).
- price_range: Price range of the mobile phone (0 - low cost, 1 - medium cost, 2 - high cost, 3 - very high cost).

Objectives:

- Explore and preprocess the dataset to handle missing values, outliers, and any other data inconsistencies.
- Perform exploratory data analysis (EDA) to gain insights into the relationships between different features and the target variable (price range).
- Select appropriate machine learning algorithms for classification and evaluate their performance using suitable metrics.
- Fine-tune the chosen model to improve its predictive accuracy.
- Validate the final model using cross-validation techniques to ensure its robustness.
- Deploy the model for real-time predictions if applicable.

Step-by-step process to classify mobile price

To classify mobile price, I have used Python language and Jupyter Notebook in Visual Studio Code.

1. All required libraries have been imported.
2. Read the dataset using pandas.
3. Check basic information (i.e., number of rows and columns, missing values, datatype of all columns etc.) of the dataset. The training dataset contains 2000 observations. It has 21 variables: 20 independent variables and 1 outcome variable. There are 7 categorical variables: price_range, blue, dual_sim, four_g, three_g, touch_screen, and wifi. We have 14 numeric variables: battery_power, clock_speed, fc, int_memory, m_dep, mobile_wt, n_cores, pc, px_height, px_width, ram, talk_time, sc_h, and sc_w. The dataset has no missing values.
4. Check mean, median, standard deviation, frequency distribution, skewness, and outliers of the numerical variables. I have used histogram and boxplot here to visualize. Some features, including fc (Front Camera mega pixels), and px_height (Pixel Resolution Height) is affected by outlier or noise!
5. Check if the categorical variables are in balance. All the categorical variables are in balance.
6. Detect outliers from the dataset. Concluding below 5% and above 95% data as outliers, I have found 85 observations from fc and 200 observations from px_height as noise. Here is an insight that mobile phone with the earliest technology and with the latest modern technology might fall into the outliers. I did not drop any of them.
7. Check Pearson Correlation among all variables. I have found that price_range is highly correlated with ram ($\rho = 0.92$). price_range has a low correlation value with the rest of the

features, but this cannot be used as a criterion to remove these features since the pearson correlation only expresses the linear relationship between two variables. Among the features, two features pc and fc have the highest correlation with each other (ρ = 0.64). Another two features three_g and four_g is also correlated with each other (ρ = 0.58).

8. During analysis the relationship between numerical variables and price_range, I came to see that ram and battery_power have the most impact on the price_range.
9. During analysis the relationship between numerical variables and price_range, I came to see that three_g has a greater impact on the price_range because by changing the category here, the percentage of samples belonging to each class of the price_range has changed more significantly.
10. Then I have defined 'price_range' as the target vector and other variables as feature matrix.
11. To evaluate after training, I have split them into X_train, X_test, y_train, and y_test. Test data contains 20% of the dataset.
12. This is the time to build machine learning model. I have trained logistic regression, support vector classifier, decision tree, random forest, and gradient boosting. Each model is built via a pipeline. In pipeline, I have scaled data using standard scaler as a data preprocessing step. After training each model with default hyperparameter, I have tuned and cross-validated them using grid-search.
13. To evaluate each model, I used accuracy score, classification report, and confusion matrix.
14. For decision tree, random forest, and gradient boosting, I looked at the feature importances.
15. Among the 5 machine learning models, logistic regression model showed best accuracy score of 0.98 after tuning and cross-validation. I took it as the final model. For hyperparameters, 1000 to C (to control the strength of regularization), newton-cg to solver (the optimization algorithm), and multinomial to multi_class (strategy to handle multiclass classification problems) has been chosen.
16. According to classification report of the tuned logistic regression model: 100% of the instances predicted as Class 0, 95% of the instances predicted as Class 1, 99% of the instances predicted as Class 2, and 97% of the instances predicted as Class 3 were correct (precision). The over-all accuracy for test data is 98%.
17. Then I have deployed the model in streamlit. This is the link to run: Mobile Price Classification.