Mobile Price Range Prediction

Name:	**Borade Sharayu Anil**
Registration No./Roll No.:	21080
Institute/University Name:	IISER Bhopal
Program/Stream:	e.g., DSE
Problem Release date:	August 17, 2023
Date of Submission:	November 19, 2023

1 Introduction

Machine learning models have proven effective in addressing classification tasks in the domain of mobile price prediction. The dataset provided for this project comprises 2000 instances in the training set and 1000 instances in the testing set. Each instance is characterized by 20 features, encompassing various aspects of mobile phone specifications. The goal is to classify mobile phones into one of four distinct price ranges: cheap (0), moderate (1), economical (2), and expensive (3). Notably, the dataset exhibits a balanced distribution of 500 instances across each price class, ensuring a representative training sample for each category. In contemporary research, machine learning models employed for classification tasks includes but are not limited to Decision Trees, Random Forest, Support Vector Machine (SVM), Logistic Regression and Gradient Boosting Methods such as XGBoost and LightGBM. Along with the machine learning models mentioned earlier it encomprise methods such as correlation analysis to find associations amongst the independent variables to avoid multicollinearity and other necessary measures to improve model performance and interpretability. In the project I used feature selection techniques such as SelectKBest to short-list features used in model, ensuring that the selected features are the most relevant, thus improving model performance.

Evaluation metrics which is used in this stage includes Macro-averaged Recall, Macro- averaged Precision and F1 Score.

2 Methods

1. Import Libraries: Importing the required libraries for data manipulation, machine learning, and evaluation.

2. Load Dataset: For reading the dataset into a DataFrame.

3. Separate Features and Target: To divide the dataset into features (X) and the target variable (y).

4. Split Dataset: For spliting the data into training and testing sets.

5. Standardize Features: To standardize the features by scaling them to have zero mean and unit variance.

6. Define SVM Model: Creating an instance of the Support Vector Machine (SVM) model.

7. Define Parameter Grid: For specifying a grid of hyperparameters to search over.

8. Grid Search: Using Grid Search with cross-validation to find the best hyperparameters.

9. Access Results: Retrieving the best parameters and the best model from the grid search.

10. Make Predictions: To use the best model to make predictions on the test set.

11. Evaluate Model: Calculating the accuracy of the model on the test set.

12. Print Results: Displaying the best parameters, accuracy, classification report, and confusion matrix. Similarly, Logistic regression is used for binary classification tasks. It models the probability of an event occurring based on input features, providing interpretable results. Decision trees operate

by recursively partitioning the input space to make categorical predictions being versatile and intuitive, suitable for both classification and regression tasks. LightGBM and XGBoost, belonging to the gradient boosting family, are stated an exceptional predictive performance it constructed an ensemble of weak learners (trees) sequentially, optimizing for predictive accuracy. LightGBM emphasizes efficiency through histogram-based techniques and XGBoost employed regularization and parallelization for enhanced speed and accuracy. These advanced models were particularly effective for large datasets and complex relationships, contributing significantly to predictive modeling in various domains of the project.

3 Experimental Analysis

The hyperparameter tuning resulted in a SVM model with the following best parameters: Best Parameters: 'C': 100, 'gamma': 'scale', 'kernel': 'linear' Accuracy: 97.5. Accuracy: The overall accuracy of the model is high at 97.5, indicating that the model performs well on the test data.

Precision: Precision measures the accuracy of the positive predictions. The model has high precision across all classes, indicating a low false-positive rate.

Recall: Recall measures the ability of the model to capture all the positive instances. The model has high recall, especially for class 1, indicating a low false-negative rate.

F1-Score: The F1-score is the harmonic mean of precision and recall. The model achieves high F1-scores for all classes, indicating a good balance between precision and recall.

Confusion Matrix: The confusion matrix provides a detailed breakdown of the model's predictions. It shows that the model performs well in correctly classifying instances for all classes.

The SVM model with the tuned hyperparameters demonstrates excellent performance on the given dataset, with high accuracy, precision, recall, and F1-score across different classes. The confusion matrix further supports the model's effectiveness in making accurate predictions. Logistic regression excelled in scenarios where linearity sufficed for accurate predictions. Decision trees showcased flexibility in handling complex feature interactions but were prone to overfitting. LightGBM and XGBoost, as gradient boosting algorithms, exhibited remarkable predictive power, with LightGBM emphasizing efficiency through histogram-based approaches and XGBoost leveraging regularization for enhanced accuracy. The experimental evaluation involved assessing accuracy, precision, recall, and F1-score metrics across different datasets, highlighting the trade-offs between model complexity and performance.

4 Discussion

Merits: The proposed SVM model with the tuned hyperparameters achieved a high accuracy of 97.5. This indicates that the model is effective in making correct predictions on the test data. The model exhibits high precision and recall across different classes, suggesting that it not only makes accurate positive predictions (precision) but also captures most of the actual positive instances (recall). The use of a linear kernel in the SVM model, as indicated by the best parameter 'kernel': 'linear', makes the model interpretable. This is beneficial for understanding the relationships between features and the target variable. The detailed analysis of the confusion matrix provides insights into the model's performance for each class, helping to identify specific areas where the model excels or may need improvement. The merits of the models vary, making logistic regression suitable for transparent insights, decision trees for interpretable structures, LightGBM for efficiency in large-scale data, and XGBoost for high predictive accuracy in diverse applications.

Limitations: SVMs, especially with non-linear kernels, can be computationally expensive, particularly as the size of the dataset increases. The chosen linear kernel helps mitigate this to some extent, but it might still be a limitation for very large datasets. SVMs can be sensitive to outliers, and their presence in the data can impact the model's performance. Robust preprocessing techniques or using kernel functions more resilient to outliers might be explored. The trade-offs among the algorithms highlight the importance of selecting models based on the specific characteristics of the data and the goals of the analysis.

Future Scopes: Exploring advanced feature engineering techniques may further enhance the model's performance. Feature selection or extraction methods can be employed to identify the most informative features. Investigating ensemble methods, such as combining multiple SVM models or integrating SVM with other classifiers, could be explored to improve overall predictive performance. Continuously refining the hyperparameter tuning process, potentially using more sophisticated optimization techniques, can be considered to find an even better set of hyperparameters. If the dataset is imbalanced, techniques to handle class imbalances (such as oversampling minority classes or using different class weights) can be employed to improve the model's ability to predict minority classes. The developed model can be deployed in real-world applications related to the dataset (assuming it represents a practical problem). Monitoring the model's performance in a real-world setting and adapting it to evolving data patterns would be crucial. Future work should focus on addressing computational complexity, refining the model through feature engineering, exploring ensemble methods, and adapting the model for deployment in real-world scenarios. The future scope of logistic regression, decision trees, LightGBM, and XGBoost lies in ongoing advancements and adaptability to emerging challenges in machine learning.