**Machine learning assignment**

**Antoine Ashraf 211001666**

Aim:

The aim of this dataset of second-hand cars prediction is to develop machine learning models that can accurately predict the price or resale value of used cars based on their attributes such as brand, model, year, kilometers driven, fuel type, transmission, and others. By analyzing historical data of second-hand car sales, the models aim to identify patterns and relationships between these attributes and the sale price. This prediction can be valuable for both buyers and sellers in the used car market, helping sellers to set appropriate prices for their vehicles and aiding buyers in making informed purchasing decisions. Additionally, these models can assist in automating the valuation process, saving time and effort for both sellers and buyers.

Dataset:

The dataset includes parameters such as Brand, Model, Year, Kilometres Driven, Fuel Type, Transmission, Owner Type, Mileage, Engine, Power, Seats, and Price related to used automobiles. It includes a wide range of well-known automakers, including Volkswagen, Audi, BMW, Maruti, Hyundai, Toyota, Honda, and Ford. Every entry includes information on the features and cost of the vehicle, which is helpful in predicting the used automobile market value. Using this dataset, predictive models based on past data and the features of used cars might be developed to determine the cost of them.

**Task 1:**

* 1. First, we set up a thorough the environment for machine learning and data analysis. It imports matplotlib/seaborn for visualisation, numpy for numerical computations, and pandas for effective data handling. To create and evaluate the models, basic tools from scikit-learn are imported. These include logistic regression, decision trees, random forests, support vector machines, gradient boosting, AdaBoost, Gaussian naive Bayes, and multi-layer perceptrons. Regression assessment measures are covered, as are preprocessing methods like scaling and imputation. It also imports pipeline construction and grid search cross-validation utilities, offering a strong framework for investigating and refining machine learning algorithms for classification and regression applications. Then we load a dataset of cars from a CSV file into a pandas DataFrame. Then, it displays the first 10 rows of the DataFrame. It sets up everything needed to explore the data, prepare it for modeling, and build machine learning models to predict outcomes based on car attributes.
  2. Then we use codes to find out the number of missing values in each column of the DataFrame (`df`). Then providing information about the DataFrame, such as the number of non-null entries in each column and the data types. Then a descriptive statistic for numerical columns in the DataFrame, like count, mean, standard deviation, minimum, and maximum values. Finally, the shape of the DataFrame, which represents the number of rows and columns it contains.

**1.3,4** After that we generate a pairplot, a grid of scatterplots, showing the relationships between all pairs of numerical variables in the DataFrame (`df`). Secondly, we creates a pairplot for selected numerical variables with a hue representing different fuel types. It visualizes how these variables relate to each other while differentiating by fuel type. The third block selects only numeric columns from the DataFrame, then creates a heatmap showing the correlation between these variables. The intensity of colors indicates the strength and direction of correlations, aiding in identifying patterns and relationships in the data.

**Task 2:**

**1.1** This code block begins by encoding categorical variables into numerical values using one-hot encoding through `pd.get\_dummies()`. Then, it fills any missing values in the DataFrame (`df`) with the most frequent value for each column using a `SimpleImputer`. Afterward, it prints out the column names and information about the DataFrame.

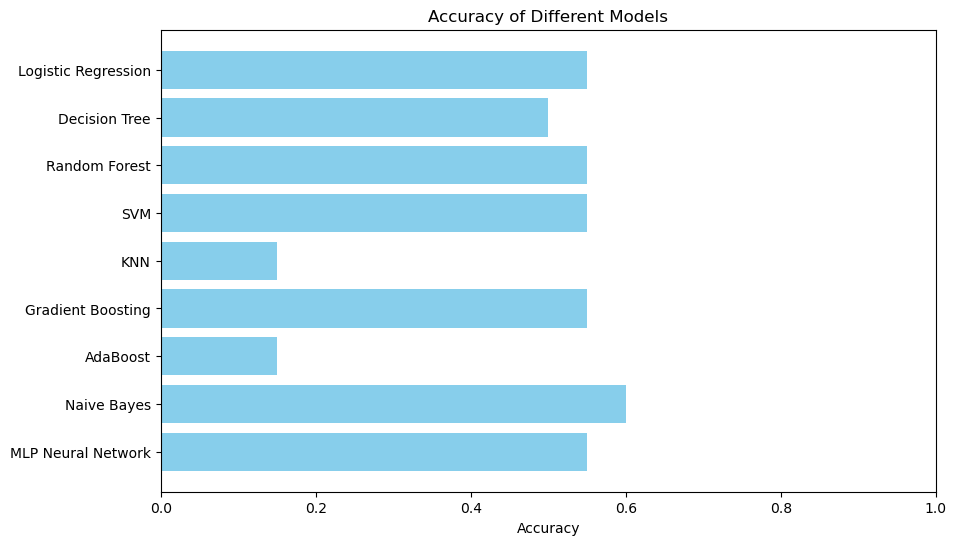
**1.2** Next, the code handles missing values in the training and testing datasets (`X\_train` and `X\_test`) by filling them with the mean value of each column using another `SimpleImputer`. Finally, it standardizes the features in both the training and testing datasets using `StandardScaler()`. This process ensures that all features have the same scale, which can improve the performance of certain machine learning algorithms.

**Task 3:**

The Mean Squared Error (MSE) is a measure of how far the predicted values are from the actual values, with a lower value indicating a better fit. In this case, the MSE is approximately 64 billion, suggesting that, on average, the squared difference between predicted and actual car prices is quite high. The Mean Absolute Error (MAE) is about 152,252, which gives an average of the absolute differences between the predicted and actual values, indicating the model's accuracy in predicting car prices. The accuracy scores for different machine learning models vary, with Logistic Regression, Decision Tree, Random Forest, and SVM achieving around 55%, indicating the proportion of correctly predicted car categories. KNN, however, only achieves 15% accuracy, suggesting lower performance. Gradient Boosting, AdaBoost, Naive Bayes, and MLP Neural Network achieve around 55% to 60% accuracy, demonstrating relatively better performance compared to other models. These accuracy scores help assess how well each model classifies car categories based on given features.

**Task 4:**

This code block conducts model evaluation and fine-tuning. Initially, it trains each model in the `models` dictionary on the training data and evaluates their performance using metrics like accuracy, precision, recall, and F1-score. It prints out the accuracy score and classification report for each model. Afterwards, it stores the accuracies of the models in a dictionary for comparison. Finally, it creates a horizontal bar plot displaying the accuracy of different models, providing a visual comparison of their performances. This evaluation helps in selecting the best-performing model and fine-tuning hyperparameters to optimize its performance further.



**Limitations:**

Our machine learning models' accuracy might be limited by the dataset's errors, such as weak or incomplete features. The ability of the models to produce accurate predictions may be impacted if the dataset has missing values or incomplete information. In this instance, the models' performance could not be ideal since the dataset might not fully capture every aspect of automotive qualities influencing prices. Furthermore, the models may find it difficult to generalize to fresh, unobserved data if the attributes in the dataset are not reliable predictors of automobile prices. Therefore, better model performance and higher accuracy may result from upgrading the dataset by correcting missing values, adding more pertinent variables, or boosting feature quality.  This highlights the importance of data quality and feature selection in machine learning tasks, as stronger features are crucial for building more effective predictive models.

**Model Deployment:**

This code makes a simple program using Streamlit, which helps predict car prices based on factors like brand, year, transmission type, and model. The program launches on a web page where dropdown menus allow you to select an automobile's brand, year, transmission, and model. Then, upon clicking a button, it estimates the potential cost of that car by using a model that was trained on a dataset of car pricing. The program accomplishes this by looking for patterns in the data to create predictions using a method known as linear regression. It's comparable to estimating a price based on prior prices for other similar autos.

**Conclusion:**

In summary, the study uses machine learning models trained on a dataset to forecast the selling price of secondhand automobiles. To create realistic pricing estimates, i have examined a number of factors, including fuel type, a transmission, year, model, and kilometres driven. Better feature selection and data quality could improve model performance. Our results highlight the significance of selecting strong characteristics and improving datasets for improved forecast accuracy in the used automobile market. Furthermore, the implementation of an accessible prediction tool using Streamlit showcases the usefulness of our research in real-world scenarios by assisting automobile dealers and purchasers in setting fair prices. All things considered, our research advances machine learning methods in the automotive sector, enabling more effective used car evaluation procedures.