**PROJECT REPORT**

**Project Title:** Predictive Vehicle Maintenance

**Student Name:** Sahil Khan

**Student ID:** 202482066

1. **Introduction**

Industries are undergoing a revolution thanks to predictive maintenance, which makes cost reductions, increased operational effectiveness, and safety possible. In order to solve the particular issue of determining whether a vehicle requires maintenance or not, this project focuses on creating a machine learning model to forecast maintenance requirements. This feature minimizes downtime and maximizes resource allocation by enabling proactive maintenance scheduling. In order to give fleet managers and car owners useful information, this study also seeks to pinpoint the main characteristics influencing these forecasts.

1. **Project Goals**

The project has two primary goals:

* **Develop a high-performing classification model:** Achieve a minimum of 90% accuracy on a held-out test set using machine learning algorithms such as logistic regression, decision trees, and neural networks.
* **Identify key features:** Determine the most influential features for predicting maintenance needs, offering insights into factors affecting vehicle health and informing preventative strategies.

1. **Data Source and Exploration**

The "Vehicles Maintenance Dataset," a synthetic dataset of 50,000 entries, is used in this study. A wide variety of characteristics pertaining to auto maintenance are included in this dataset. Vehicle details (model, mileage, age, fuel type, transmission type, engine size, odometer reading), maintenance history (history, last service date, repair history count), operational data (reported problems, fuel efficiency), component condition (tire, brake, battery), and other pertinent details (owner type, insurance premium, accident history, warranty expiration date) are all examples of feature categories. Need\_maintenance is the goal variable (1 = Yes, 0 = No). Controlled testing and model validation under various conditions are made possible by the use of a synthetic dataset.

Important ideas were uncovered using exploratory data analysis, or EDA. The majority of the cars in the sample had moderate mileage and usage and were very new. There was a reasonably even distribution of car models. "Poor" was the most common category in maintenance\_history, indicating a large class disparity. This draws attention to certain areas where maintenance procedures should be strengthened. The most common fuel type was "Petrol," which was followed by "Electric" and "Diesel." Balanced distributions of tire, brake, and battery conditions suggested routine component repair. Crucially, 81% of the cars needed repair, highlighting how crucial predictive models are. According to a correlation study, need\_maintenance was more strongly correlated with reported\_issues, service\_history, accident\_history, and odometer\_reading.

1. **Data Preprocessing and Feature Engineering**

Several preprocessing steps were implemented:

Date Conversion: Date columns were converted to datetime objects.

Missing Values: The dataset contained no missing values.

Duplicates: No duplicate records were found.

Feature Engineering: New features were engineered to potentially improve model performance. These include:

time\_since\_last\_service (days): Time elapsed since the last service.

warranty\_duration (days): Remaining warranty period.

mileage\_per\_year: Annual mileage.

service\_frequency: Services per year.

accident\_rate: Accidents per year.

Encoding: Categorical features (vehicle\_model, maintenance\_history, fuel\_type, transmission\_type, owner\_type, tire\_condition, brake\_condition, battery\_status) were encoded using one-hot encoding. Numerical features were scaled using StandardScaler.

1. **Model Training and Evaluation**

Logistic Regression, Random Forest, SGD Classifier, Decision Tree, XGB Classifier, and a Neural Network were among the classification models that were trained and assessed. The decision tree-based model's hyperparameters were tuned using a grid search with five-fold cross-validation to maximize performance. On the test set, the top-performing Decision Tree Classifier had an accuracy of 96%. Even though L2 regularization and dropout were used to try to reduce overfitting, a basic neural network was nonetheless built. In the end, the decision tree-based approach offered the optimal trade-off between computational efficiency and performance.

Root mean squared error (RMSE) was also used as the main metric for training and assessing a number of regression models. Linear Regression, Ridge, Random Forest Regressor, Gradient Boosting Regressor, and XGB Regressor were among these models. R-squared values near 0 indicate inadequate explanatory ability for the goal variable fuel\_efficiency given the features at hand, indicating fairly consistent performance across different models.

1. **Feature Importance**

The best performing model, the tuned Decision Tree Classifier, has been integrated into a Streamlit application for easy access and prediction. The application allows users to input vehicle characteristics and receive predictions regarding the need for maintenance. The application code (stApp.py) is available for deployment using Streamlit. A separate script (app.py) is also provided for a complete demonstration of the data pipeline, including data ingestion, validation, transformation, model training, evaluation, and prediction.

1. **Conclusion**

Through this project, a high-performing Decision Tree Classifier with 96% accuracy in predicting vehicle maintenance needs was successfully constructed. The identification of important features offers helpful direction for maintenance plans. The model's implementation through a Streamlit application provides an intuitive user interface for real-world application. To increase the robustness and adaptability of the model, future research might investigate more intricate model structures or gather more real-world data.

1. **Project Link: https://github.com/sahil82764/Vehicle\_Maintenance\_Prediction**

[Additional Files apart from Submission Files can be accessed]