**Documentation on Mini Project**

**Fuel Consumption Prediction**

**by**

**Tiyyaguara Naga Pavan Reddy**

1. **ABSTRACT**

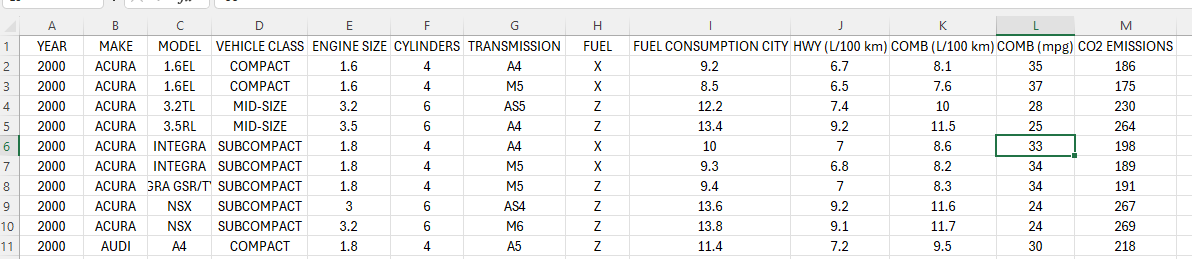
In this project, we use machine learning to analyze and predict how much fuel cars use (measured in liters per 100 kilometers) based on factors like **year, make, vehicle class, engine size, number of cylinders, fuel and transmission type**. The dataset we used contains records from the years **2000 to 2022**, and our aim is to find patterns that can help both car manufacturers and consumers make better decisions.

We tested four different methods—**Linear Regression, Decision Tree, Random Forest**, and **Gradient Boosting**—to see which one gives the best results. We evaluated the performance of each method using **R² score** and **Mean Squared Error** (MSE). Among all of them, **Random Forest** performed the best and gave the most accurate predictions. The project also involved a lot of data preparation, including label encoding for categorical features, and we used visual tools to better understand the models.

In the end, this project shows how machine learning can help solve real-world problems and how using data can help reduce fuel use and be better for the environment.

1. **DATASET INFORMATION**

The dataset used in this project is "Fuel Consumption 2000–2022". It includes information about vehicles tested between 2000 and 2022, with details on their specifications and performance. The dataset contains **22,556 records** and has **13 columns** in total. For this project, we chose 7 features as input and 1 target variable.

****

**2.1. Selected Features**

* **YEAR:** The year the vehicle was tested. It helps track changes in technology or regulations that could affect fuel efficiency.
* **MAKE:** The car's manufacturer (e.g., Ford, Toyota). Different manufacturers may design cars differently, which can impact fuel use.
* **VEHICLE CLASS:** The type or size of the vehicle (e.g., SUV, sedan, compact). Larger vehicles generally consume more fuel.
* **ENGINE SIZE:** The engine’s displacement in liters. Bigger engines usually burn more fuel.
* **CYLINDERS:** The number of cylinders in the engine. More cylinders typically mean more power, which can lead to higher fuel consumption.
* **TRANSMISSION:** The type of transmission (e.g., automatic, manual, CVT). It affects how efficiently the car uses fuel.
* **FUEL:** The type of fuel used (e.g., gasoline, diesel, hybrid). Different fuels have different efficiencies and emissions.

**2.2. Target Feature**

* **COMB (L/100 km):** The combined fuel consumption in liters per 100 kilometers. This tells us how fuel-efficient a vehicle is in normal driving conditions (both city and highway)

**2.3. Importance of Features**

Each feature is important for understanding fuel consumption:

* **ENGINE SIZE, CYLINDERS,** and **VEHICLE CLASS** directly influence how much **fuel** the car uses.
* **TRANSMISSION** and **FUEL TYPE** affect how efficiently the car runs.
* **MAKE** and **YEAR** help us understand **trends in car** manufacturing and changes over time.

1. **INTRODUCTION**

In today’s world, saving fuel and protecting the environment has become very important. As fuel prices are increasing and pollution is also becoming a big issue, it is necessary to find ways to make vehicles more fuel-efficient. In this project, I have used machine learning to predict the **combined fuel consumption** (in liters per 100 kilometers) of vehicles based on different features like **engine size, number of cylinders, vehicle class, fuel type, and transmission type.**

The dataset I used contains information about vehicles from the year 2000 to 2022. By training different machine learning models, I tried to find out how accurately we can predict how much fuel a vehicle will use. I have applied four algorithms—**Linear Regression, Decision Tree, Random Forest, and Gradient Boosting**—and compared their performance using **Mean Squared Error (MSE) and R² Score.**

The main aim of this project is to help people understand how different factors affect fuel consumption and to show how machine learning can be used to make useful predictions from real-world data. It can help customers choose better vehicles; manufacturers design more efficient cars and help in reducing environmental pollution.

* 1. **Models and Algorithms Used**
     1. **Linear Regression**
* The simplest form of regression, where the relationship between **the input and output is assumed to be linear (i.e., a straight line).**
* It tries to find **the best-fit line** that minimizes the difference between the predicted and actual values.
* It is one of the simplest forms of regression and is widely used for predicting **continuous outcomes** based on input features.
* The equation for Linear Regression is **y = m x +b.**
  + 1. **Decision Tree**
* A Decision Tree is a machine learning algorithm that makes decisions by **splitting data into branches** based on feature values, like a flowchart.
* A Decision Tree consists of a **Root Node** (starting point), **Decision Nodes** (splits based on features), **Leaf Nodes** (final predictions), and **Edges** (connections between nodes).
* How Decision Tree Regressor Works:

**1.Splitting**: The algorithm splits the dataset at each node based on a feature that minimizes a chosen splitting criterion (e.g., Mean Squared Error or MSE).

**2.Recursive Splitting:** The tree keeps splitting the data into smaller subsets until a stopping criterion is met, like maximum depth or minimum samples in a leaf.

**3.Prediction:** After training, the prediction is the average value of the target variable in the leaf node where the input data ends up.

* They handle both numerical and categorical data, capture non-linear relationships, and are easy to interpret.
  + 1. **Random Forest**
* A Random Forest is an **ensemble learning method** that combines multiple decision trees to improve model performance by reducing overfitting and increasing accuracy.
* How it Works:

**1.Bootstrap Sampling**: Randomly selects subsets of the training data (with replacement) to train each tree. Some data points might not appear in the training set of a tree (these are called "out-of-bag" samples).

**2.Tree Growth:** Each decision tree is trained on a different subset of the data. At each split, only a random subset of features is considered (not all features), ensuring trees are diverse.

**3.Prediction:**

**For classification:** The class that most trees vote for is selected as the final prediction.

**For regression:** The average of all trees' predictions is taken as the final output

* Random Forest **reduces overfitting**, handles large datasets and missing data, and can be used for both classification and regression tasks.
  + 1. **Gradient Boosting**
* Gradient Boosting is an **ensemble** learning method that builds a model by **combining multiple weak learners** (usually decision trees) to create a strong model.
* Unlike **Random Forest**, which builds trees in **parallel,** Gradient Boosting builds trees **sequentially**, where each tree tries to correct the mistakes made by the previous ones.
* How it Works:

**1.Start with a simple model**: Begin with a weak model like a small decision tree (a stump).

**2.Fit residuals:** Calculate the errors and train a new model to predict those errors.

**3.Add the new model:** Add the new model to the previous ones using a learning rate to control its impact.

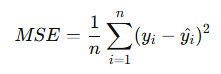
**4.Repeat:** Keep adding new models, each correcting the errors of the combined previous models.

**5.Final Prediction:** Combine all the models’ outputs to make the final prediction.

* 1. **Evaluation Metrics**

Evaluation metrics are used to measure how well a machine learning model performs. In this project I have used two evaluation metrics, and they are MSE and R2 Score.

* + 1. **Mean Squared Error (MSE):**
* Mean Squared Error (MSE) is a common metric used **to evaluate the accuracy** of a **regression model.**
* It calculates the **average of the squares of the differences** between the **actual** and **predicted values.**
* Formula



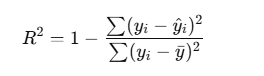
Where

n: number of observations

yi​: actual value

y^i: predicted value

* + 1. **R2 Score**
* The R² score measures how well the predicted values from a regression model approximate the actual data points.
* It indicates **the proportion of variance** in the dependent variable that is predictable from the independent variables.
* Formula:



Where:

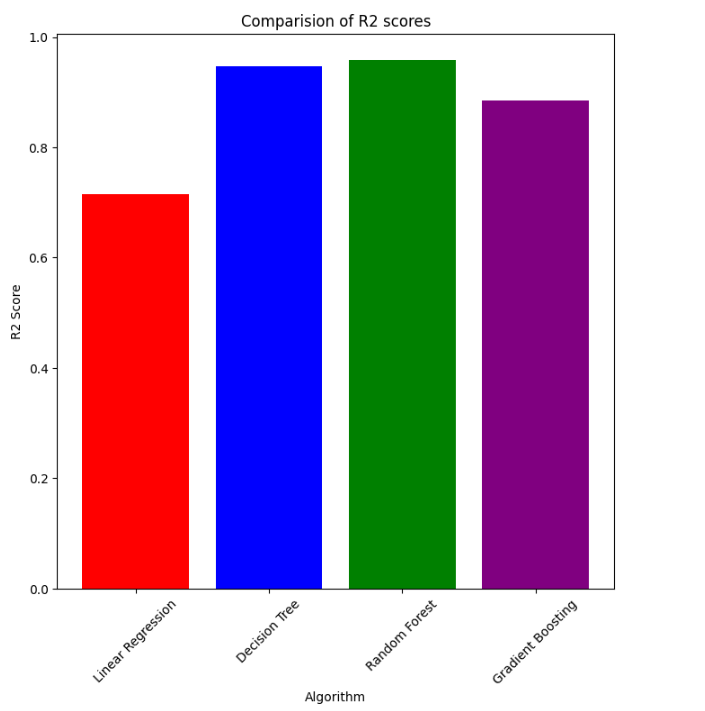
* yi: actual values
* y^i: predicted values
* y- : mean of actual values

1. **CLASS DIAGRAM**

The class diagram represents the overall workflow of the machine learning project, starting from importing the required libraries to visualizing the final results. It outlines each major step such as loading the dataset, preprocessing the data by handling categorical values and missing data, and splitting the dataset into training and testing sets. The diagram further shows how different regression algorithms are applied to the training data. It also includes the evaluation of each model using R² score and Mean Squared Error (MSE). Finally, the results are visualized through graphs for better understanding and comparison of model performances. This diagram provides a structured view of how each component is connected and how the entire project flows logically from start to end.

1. **DATA VISUALIZATION**

The bar graph presented above showcases a comparative analysis of the R² scores obtained from four different machine learning models: Linear Regression, Decision Tree Regressor, Random Forest Regressor, and Gradient Boosting Regressor. The R² score, or coefficient of determination, is a statistical metric that measures the proportion of variance in the dependent variable that can be predicted from the independent variables. In simpler terms, it tells us how well the model's predictions match the actual outcomes. A score closer to 1 indicates a more accurate and reliable model.

****

**Detailed Analysis:**

**1.Random Forest Regressor:**

* **R² Score:** 0.9579
* This model achieved the highest R² score among all models evaluated.
* Its ensemble-based architecture, which combines multiple decision trees, enables it to capture complex, non-linear relationships in the data.
* The high R² score indicates strong predictive power and generalization capability, making it a suitable choice for fuel consumption prediction

**2.Gradient Boosting Regressor:**

* **R² Score:** 0.8845
* This model ranked third in performance.
* It builds models sequentially, where each new model focuses on correcting the previous model’s errors.
* Although it didn’t outperform the Random Forest or Decision Tree, it still exhibited solid results and can capture intricate data patterns.

**3.Decision Tree Regressor:**

* **R² Score:** 0.9463
* Trailing closely behind Random Forest, the Decision Tree Regressor also demonstrated strong performance.
* It effectively handles non-linear data but may risk overfitting, especially with noisy or high-dimensional datasets.

**4.Linear Regression:**

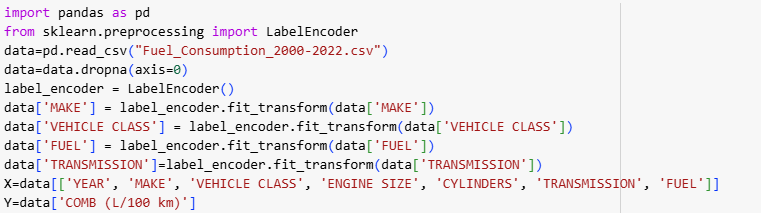
* R² Score: 0.7150
* This model recorded the lowest R² score among the four.
* Its assumption of linearity limits its ability to model complex relationships in the dataset.
* While simple, fast, and interpretable, its relatively poor performance suggests it's not the best fit for predicting fuel consumption in this scenario.

1. **IMPLEMANTATION**

In this section, we will go through the step-by-step process followed to build and evaluate machine learning models for predicting combined fuel consumption (in L/100 km) based on various vehicle characteristics.

**6.1. Data Preprocessing**

The implementation begins with preprocessing the data, encoding categorical values, and selecting relevant features.



* **pd.read\_csv()** loads the dataset.
* **dropna()** removes rows with missing values to prevent errors during training.
* **LabelEncode**r converts text categories (like car make or transmission type) into numeric labels so that they can be used by machine learning models.
* **X** contains the selected input features, while **Y** is the target variable (COMB (L/100 km)).

**6.2. Model Training**

We then split the data into training set and testing sets and then apply multiple regression algorithms—Linear Regression, Decision Tree, Random Forest, and Gradient Boosting—to train and test the model's performance.



* train\_test\_split() splits the data into training and testing sets (80% training, 20% testing) to evaluate model generalization.
* LinearRegression(), DecisionTreeRegressor(), RandomForestRegressor(), and GradientBoostingRegressor() are the four models used to predict fuel consumption.
* .fit() trains each model using the training data (x\_train, y\_train).

**6.3. Model Evaluation and Printing Evaluation Metrics**

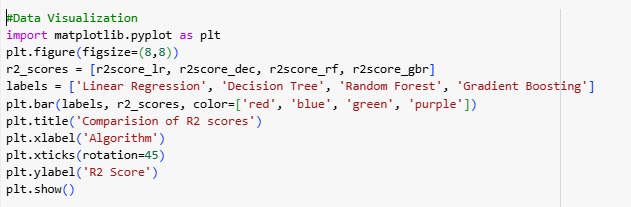
We then evaluate the model’s using metrics like Mean Squared Error and R² Score, visualize their accuracy, and identify the most effective model for our prediction task.



* Each model makes predictions on the x\_test data.
* mean\_squared\_error() calculates how much the predictions deviate from actual values (lower is better).
* r2\_score() measures how well the model explains the variation in the data (closer to 1 is better).
* These metrics are used to compare the performance of the models and decide the best one.

**6.4. Visualization of Results**

Visualizing the R² scores and MSE allows us to compare the effectiveness of each algorithm. It highlights which model best fits the data and which has the lowest prediction error.



* This bar plot provides a clear visual comparison of how well each model performed based on the R² score and Mean Squared Error (MSE).
* Different colors are used to represent each algorithm, making them easy to distinguish.
* The R² score plot helps identify which model gives the most accurate predictions (higher bar = better performance).
* The MSE plot highlights the prediction error of each model (lower bar = better performance).
* Together, these plots give a complete picture of each algorithm's effectiveness.

1. **CONCLUSION**

In this project, I worked on predicting how much fuel a car uses using machine learning. The data I used had details about different vehicles from the years **2000 to 2022**, including things like the year, brand, type of vehicle, engine size, number of cylinders, transmission type, and fuel type. I picked these features because they all affect how much fuel a car uses.

First, I cleaned the data by removing missing values. Then, I changed the text columns into numbers using **label encoding** so that machine learning algorithms could understand them. I chose **seven features** (year, vehicle class, make, engine size, cylinders, transmission and fuel) to use as inputs and set the column **COMB (L/100 km)** as the thing I wanted to predict. I split the data into training and testing sets and tried out four different models: **Linear Regression, Decision Tree, Random Forest, and Gradient Boosting.**

After training, I tested how well each model worked using **Mean Squared Error (MSE)** and **R² Score**. The **Random Forest** model gave the best results—it had the lowest error (MSE of 0.35) and the highest R² Score (0.95), which means it could explain **95%** of the changes in fuel usage. Decision Tree and Gradient Boosting also worked well, but not as good as Random Forest. Linear Regression did the worst, probably because it only works well when the data has a straight-line pattern, which wasn’t the case here.

**Random Forest** did the best because it uses **lots of decision trees** and **combines their results**, which helps **avoid overfitting** and makes the predictions more **accurate**. It’s also good at handling both simple and complex relationships in the data.

Overall, this project showed that machine learning can be helpful for predicting fuel usage. Random Forest turned out to be the best model for this, and it could be useful for people buying cars, car companies, or even the government to improve fuel efficiency and reduce pollution.