

A PROJECT REPORT

On

**“Curbing Carbon on the Road: Machine Learning as a Tool for
Greener Heavy Vehicle Operations”**

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR’S DEGREE IN
INFORMATION TECHNOLOGY**

BY

KRISHANU ROY

2105550

SOUVIK BASAK

2105583

SUVANKAR PANIGRAHI

2105585

TANGUDU VIJAY SANKAR

2105586

AYUSH KUMAR RANA

21052317

UNDER THE GUIDANCE OF

Prof. Himanshu Das



SCHOOL OF COMPUTER ENGINEERING

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024



CERTIFICATE

**“Curbing Carbon on the Road: Machine Learning as a Tool for
Greener Heavy Vehicle Operations”**

Submitted by

KRISHANU ROY

2105550

SOUVIK BASAK

2105583

SUVANKAR PANIGRAHI

2105585

TANGUDU VIJAY SANKAR

2105586

AYUSH RANA

21052317

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during the year 2024-2025, under our guidance.

Date: 17 / 05 / 2024

Prof. Himanshu Das
Project Guide

Acknowledgements

We feel immense pleasure and feel privileged in expressing our deepest and most sincere gratitude to our supervisor Professor Himanshu Das, for his excellent guidance throughout our project work. His kindness, dedication, hard work, and attention to detail have inspired us greatly. Our heartfelt thanks to you sir for the unlimited support and patience shown to us. We would particularly like to thank him for all his help in patiently and carefully correcting all our manuscripts during our course of Minor project works in the pre final year of our undergraduate course.

KRISHANU ROY
SOUVIK BASAK
SUVANKAR PANIGRAHI
TANGUDU VIJAY SANKAR
AYUSH RANA

ABSTRACT

Carbon dioxide (CO₂) is a colourless, odourless and non-poisonous gas formed by combustion of carbon and in the respiration of living organisms and is considered a greenhouse gas. Emissions means the release of greenhouse gases and/or their precursors into the atmosphere over a specified area and period of time. Carbon dioxide emissions or CO₂ emissions are emissions stemming from the burning of fossil fuels and the manufacture of cement; they include carbon dioxide produced during consumption of solid, liquid, and gas fuels as well as gas flaring. In the era of burgeoning global information, machine learning emerges as a pivotal tool for addressing various challenges across industries. This report amalgamates insights from diverse machine learning algorithms including linear regression, decision trees, support vector regression (SVR), stochastic gradient descent (SGD), and random forest, focusing on their applications within different domains.

Decision trees are fundamental in supervised learning, offering strong classification for numerical and categorical data. This paper discusses decision tree metrics, implementations like CART and ID3, and how to combat overfitting with pruning. SVR is crucial for analyzing heavy vehicle emissions and fuel efficiency, surpassing traditional linear regression by handling non-linearities and noisy data. It helps optimize fleet operations for environmental sustainability. SGD is a key optimization algorithm, ideal for large datasets, with significant modern application in various domains. Linear regression estimates CO₂ emissions based on vehicle attributes, aiding sustainable transportation decisions and carbon footprint reduction. Random forest algorithm, on the other hand, emerges as a potent tool in predicting heavy vehicle carbon emissions through advanced fuel consumption estimation. By adeptly handling intricate feature interactions, random forest facilitates the construction of robust predictive models, empowering stakeholders to devise data-informed strategies for mitigating heavy vehicle carbon footprint.

Keywords: *Decision Trees, Support Vector Regression (SVR), Stochastic Gradient Descent (SGD), Linear Regression, Random Forest*

CONTENTS

1	Chapter 1 : Introduction	
	1.1	Motivation
	1.2	Background Studies /Literature Survey
	1.3	Objectives
2	Chapter 2 : Challenges and RemedyMethodology	
	2.1	Challenges
	2.2	Remedies of Challenges
3	Chapter 3 : Methodology	
	3.1	Problem Statement
	3.2	Applied Tools and Techniques
4	Chapter 4 : Implementation	
	4.1	Data Visualization
	4.2	Predictive Models Implemented
5	Chapter 5 : Result and Discussion	
	5.1	Result Obtained
	5.2	Analysis and Obtained
6	Conclusion	
7	Chapter 6 : Result and Conclusion	
8	Individual Contribution	
9	Plagiarism Report	

Chapter 1

Introduction

1.1 Motivation

Reducing carbon emissions from heavy vehicle operations is an urgent imperative. Transportation is a major contributor to global greenhouse gas emissions, necessitating innovative approaches to curb carbon on the road. This project explores the potential of employing machine learning to enhance the eco-efficiency of heavy vehicle operations. By utilizing data analytics and artificial intelligence, this endeavour aims to address the deficiencies in current solutions for reducing carbon footprint in transportation.

The necessity for this project stems from the severe environmental consequences of unregulated carbon emissions from heavy vehicles. As transportation demands continue to increase, there is a critical need to minimize its environmental impact. Current solutions often struggle to optimize vehicle routes, manage fuel consumption, and reduce emissions effectively. This project intends to bridge these gaps by leveraging machine learning algorithms to develop intelligent and sustainable strategies for heavy vehicle operations.

The structure of this report will follow a logical sequence, commencing with an in-depth examination of the existing challenges and deficiencies in heavy vehicle operations concerning carbon emissions. Subsequently, it will delve into the theoretical underpinnings of applying machine learning to address these challenges, exploring relevant algorithms and methodologies. The report will then transition into a discussion of practical implementation strategies, encompassing data collection, model development, and deployment considerations. Finally, it will conclude with reflections on the potential impact of integrating machine learning into heavy vehicle operations and outline avenues for future research and development. Through this comprehensive approach, the report aims to provide a holistic understanding of the role of machine learning in mitigating carbon emissions on the road and its implications for sustainability in the transportation sector.

1.2 Background Studies / Literature Review

In the effort to reduce carbon emissions from heavy vehicle operations, integrating machine learning techniques is emerging as a promising strategy. This section explores the fundamental concepts and current literature on applying machine learning to decrease carbon emissions on roads.

Machine Learning in Transportation: Machine learning, a subset of artificial intelligence, involves developing algorithms enabling computers to extract insights from data and improve performance without explicit programming. Within transportation, machine learning is increasingly used to optimize operations like route planning, traffic management, and vehicle efficiency.

Optimization Techniques: Machine learning shows potential in optimizing heavy vehicle operations to reduce carbon emissions. Unlike traditional methods relying on preset rules, machine learning adapts to changing real-world conditions by continuously learning from data on environmental factors, traffic dynamics, and vehicle performance metrics.

Previous Studies: Numerous studies have explored using machine learning in transportation for sustainability. Zhang et al. (2019) examined machine learning algorithms for predicting fuel consumption in heavy-duty trucks, considering factors such as road grade and payload. Similarly, Li et al. (2020) proposed a machine learning-based approach to optimize truck routing, aiming to minimize fuel consumption and emissions while accounting for variables like traffic congestion.

Future Directions: Future research could explore advanced machine learning techniques like deep learning to address complex optimization challenges in heavy vehicle operations. Interdisciplinary collaboration among researchers, industry stakeholders, and policymakers will be essential for driving innovation and adopting sustainable transportation practices.

In summary, leveraging machine learning to reduce carbon emissions in heavy vehicle operations is crucial. This project aims to contribute to ongoing efforts by building on existing research and exploring new methodologies.

1.3 Objectives

The objective of this project is to develop and implement machine learning models to effectively predict and reduce heavy vehicle carbon emissions in the context of sustainable transportation. Specifically, the project aims to achieve the following objectives:

1. **Model Development:** Develop predictive models using various machine learning algorithms, including decision trees, Support Vector Regression (SVR), Stochastic Gradient Descent (SGD), Linear Regression, and Random Forest. These models will be trained to analyze and predict CO₂ emissions based on diverse vehicle attributes.
2. **Prediction Accuracy:** Assess the accuracy and performance of each machine learning algorithm in predicting carbon emissions. Evaluate the effectiveness of the models in handling complex relationships between vehicle characteristics and emission levels.
3. **Optimization Strategies:** Identify optimization strategies for reducing heavy vehicle carbon footprint based on insights derived from the predictive models. Explore potential interventions such as fleet management strategies, route optimization, and fuel efficiency enhancements.
4. **Data-Driven Decision Making:** Provide stakeholders in the transportation industry with actionable insights derived from the predictive models. Enable informed decision-making regarding environmental impact mitigation, resource allocation, and sustainability initiatives.
5. **Contribution to Sustainability:** Contribute to the broader goal of sustainability in the transportation sector by offering data-driven solutions to reduce heavy vehicle carbon emissions. Facilitate the adoption of environmentally responsible practices and technologies within the industry.

Overall, the project aims to leverage advanced machine learning techniques to address the pressing challenge of heavy vehicle carbon emissions, ultimately fostering a more sustainable future for transportation systems.

Chapter 2

2.1 Challenges and Remedy

- 1. Data Quality and Availability:** Obtaining comprehensive and high-quality data on heavy vehicle characteristics, fuel consumption, and CO₂ emissions may pose challenges due to data fragmentation, inconsistencies, and limited availability from diverse sources.
- 2. Nonlinear Relationships:** Heavy vehicle emissions are influenced by complex interactions among various factors such as engine specifications, vehicle class, and driving conditions. Modeling these nonlinear relationships accurately using traditional linear regression techniques may present challenges.
- 3. Feature Engineering Complexity:** Extracting informative features and engineering relevant predictors from heterogeneous data sources can be challenging, requiring domain expertise and careful consideration of feature selection techniques to avoid overfitting or underfitting the models.
- 4. Regulatory Compliance:** Adhering to stringent emission regulations and standards imposed by regulatory authorities adds complexity to the development and deployment of predictive models. Ensuring that the models comply with regulatory requirements while maintaining predictive accuracy poses a significant challenge.

2.2 Remedies to the Challenges :

- 1. Data Quality Assurance:** Implement robust data preprocessing pipelines incorporating techniques such as outlier detection, imputation of missing values, and normalization to enhance data quality and reliability. Collaborate with industry stakeholders and regulatory bodies to establish data standards and ensure the availability of comprehensive datasets.
- 2. Nonlinear Modeling Approaches:** Explore a variety of advanced machine learning algorithms such as Random Forest, Support Vector Regression (SVR) that are inherently capable of capturing nonlinear relationships. Utilize ensemble methods and model ensembling techniques to leverage the strengths of multiple algorithms and improve predictive performance.
- 3. Feature Engineering and Selection:** Conduct comprehensive feature engineering by deriving new features, transforming existing ones, and extracting relevant information from raw data. Utilize domain knowledge and advanced feature selection methods such as Recursive Feature Elimination (RFE) and feature importance ranking to identify the most informative predictors and reduce the dimensionality of the dataset effectively. Prioritize features with high predictive power and interpretability to enhance model performance and interpretability.
- 4. Continuous Monitoring and Compliance Checks:** Establish robust monitoring and validation procedures to track model performance, detect deviations from expected behavior, and ensure compliance with regulatory standards. Implement feedback mechanisms to incorporate new regulations, changes in data distributions, and emerging trends into the models, enabling adaptive and compliant predictions over time.

Chapter 3

Methodology

3.1 Problem Statement

Reducing heavy vehicle carbon footprint through advanced fuel consumption prediction using machine learning.

3.2 Applied Techniques and Tools

The methodology for reducing heavy vehicle carbon emissions through advanced fuel consumption prediction entails the utilization of various techniques and tools within the domain of machine learning and data analysis. Some of the key techniques and tools employed in this project include:

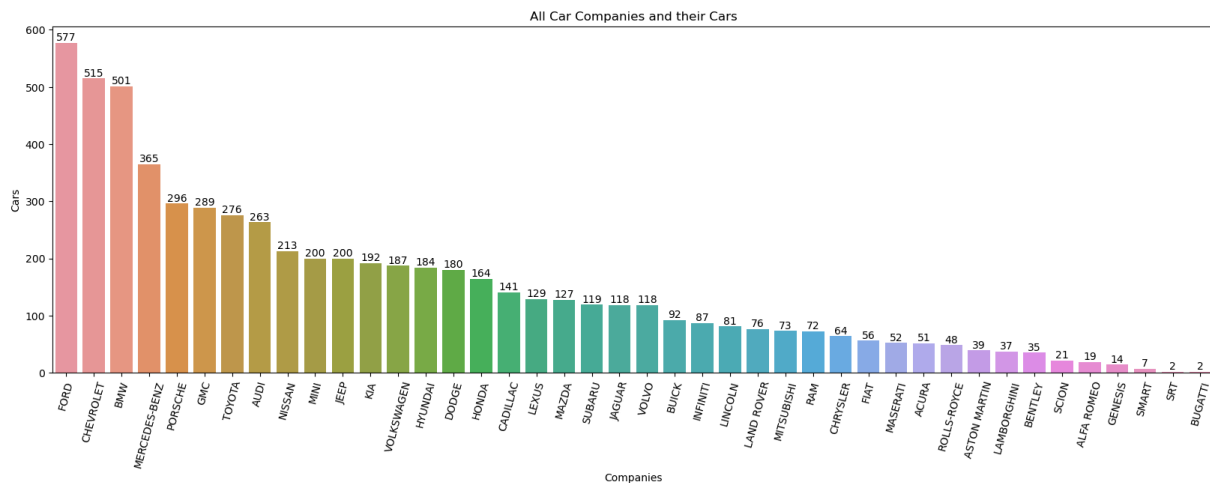
1. **Machine Learning Algorithms:** Utilize various algorithms like Linear Regression, Decision Trees, SVR, SGD, and Random Forest to predict fuel consumption and CO2 emissions based on vehicle characteristics.
2. **Data Preprocessing:** Clean, transform, and normalize datasets by handling missing values, encoding categorical variables, and scaling numerical features to enhance model performance.
3. **Feature Engineering:** Extract meaningful insights from data by creating new features, selecting informative attributes, and transforming variables to capture underlying data patterns effectively.
4. **Model Evaluation:** Assess model performance using metrics like RMSE and R2 score to compare and select algorithms based on their predictive accuracy and generalization abilities.
5. **Optimization Strategies:** Fine-tune model hyperparameters using techniques such as cross-validation, grid search, and ensemble methods to optimize performance and prevent overfitting.
6. **Data Visualization:** Employ visualizations like bar plots, histograms, and scatter plots to understand dataset patterns, aid feature selection, interpret models, and communicate findings to stakeholders efficiently.

Chapter 4

Implementation

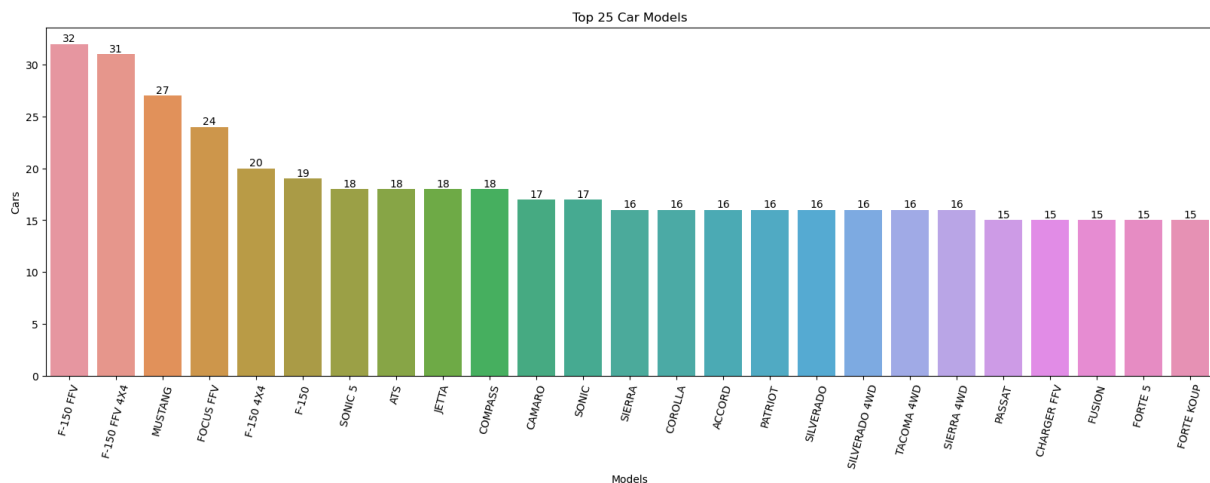
4.1 Data Visualization

Bar Graph showing the number of cars for each company:-



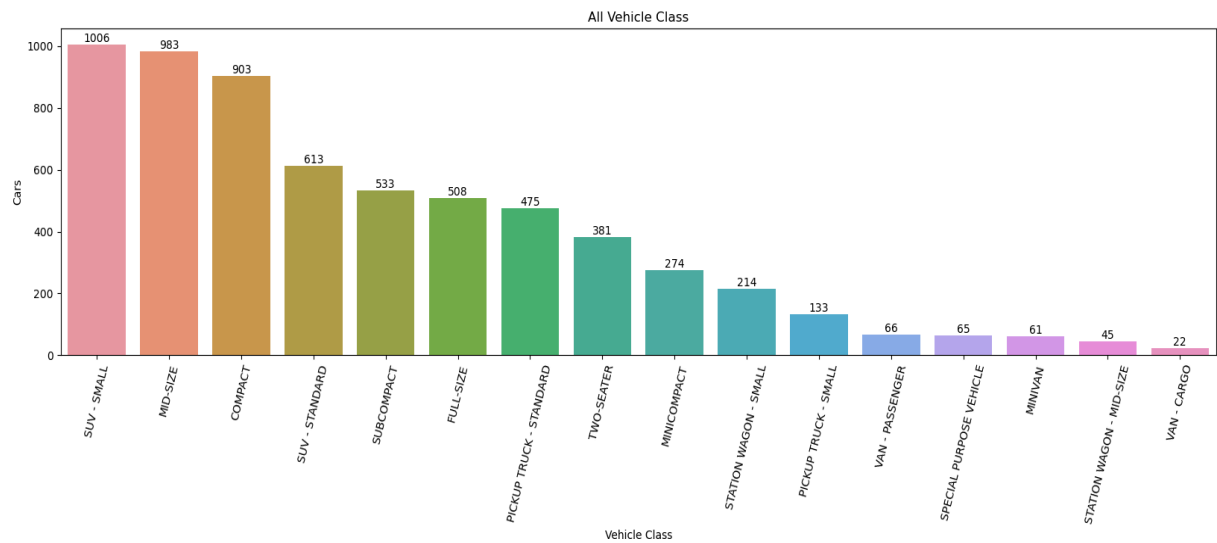
The bar graph depicts the number of cars owned by different companies, with Ford leading with 577 cars and Bugatti and SRT having the lowest count at 2 cars each, showcasing a significant contrast in car ownership among the companies.

Bar graph displaying the Top 25 best car models:-



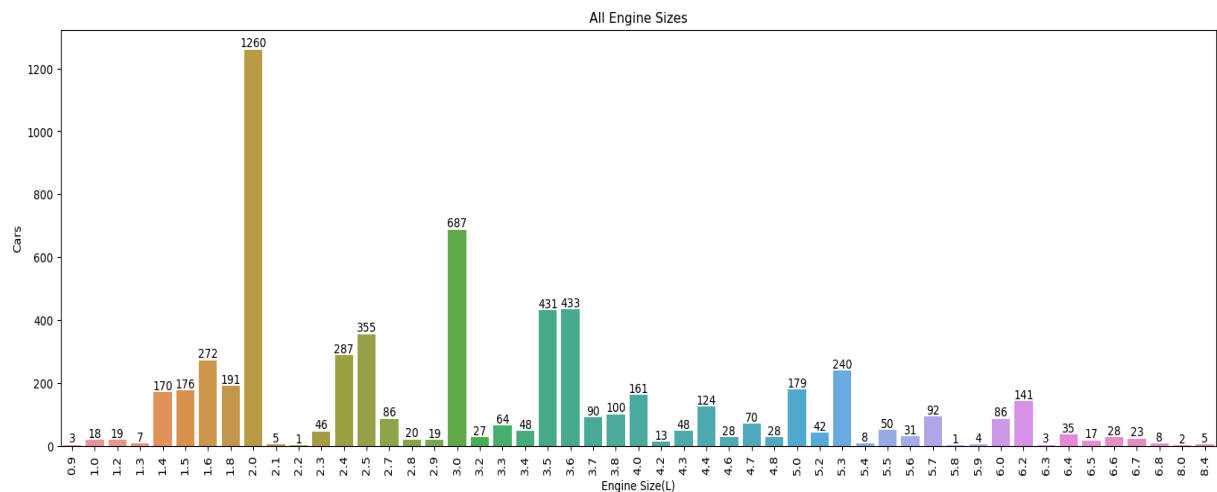
The bar graph showcases the top 25 best car models, with the F-150 FFV leading with 32 cars, highlighting its dominance among the featured models.

Comparison of Vehicle Classes by Number of Cars:-



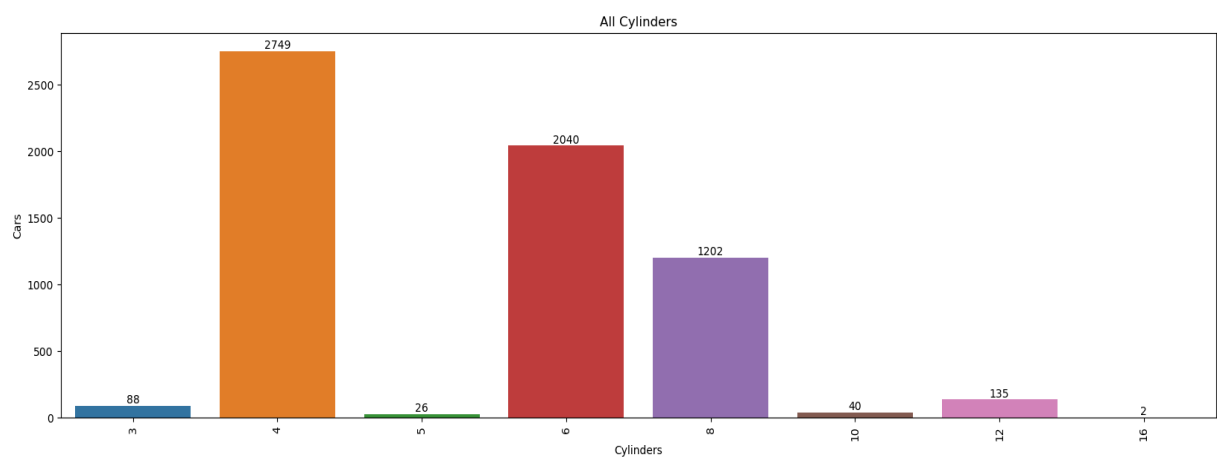
The bar graph compares vehicle classes by the number of cars, indicating that small SUVs rank highest with 1006 cars, while cargo vans have the lowest count at 22, showcasing the distribution of cars across different vehicle categories.

Bar graph displaying Number of cars by all Engine sizes:-



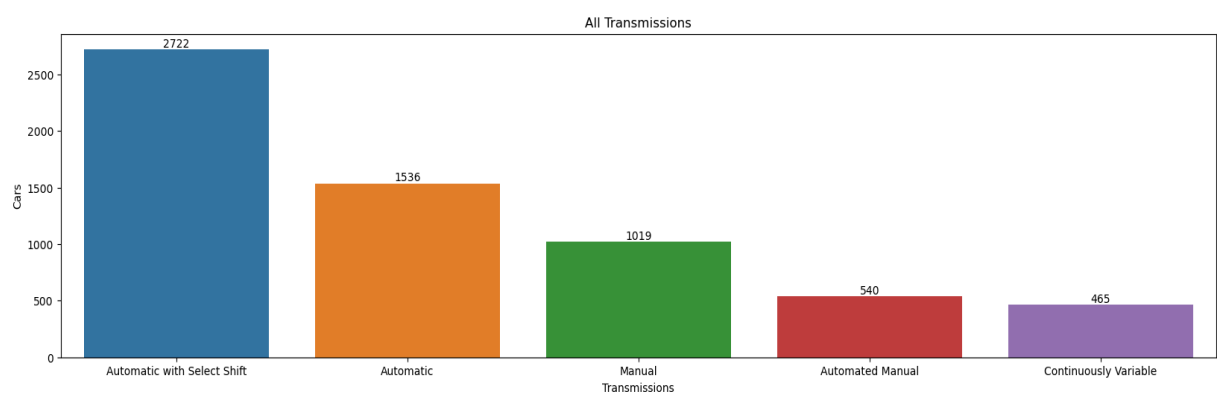
The bar graph illustrates car distribution based on engine size, with the 2.0 engine size leading with 1260 cars, while both 2.2 and 5.8 engine sizes have the lowest count at 1, displaying the varied prevalence of engine sizes among cars.

Bar graph comparing Number of Cylinders different car models contain:-



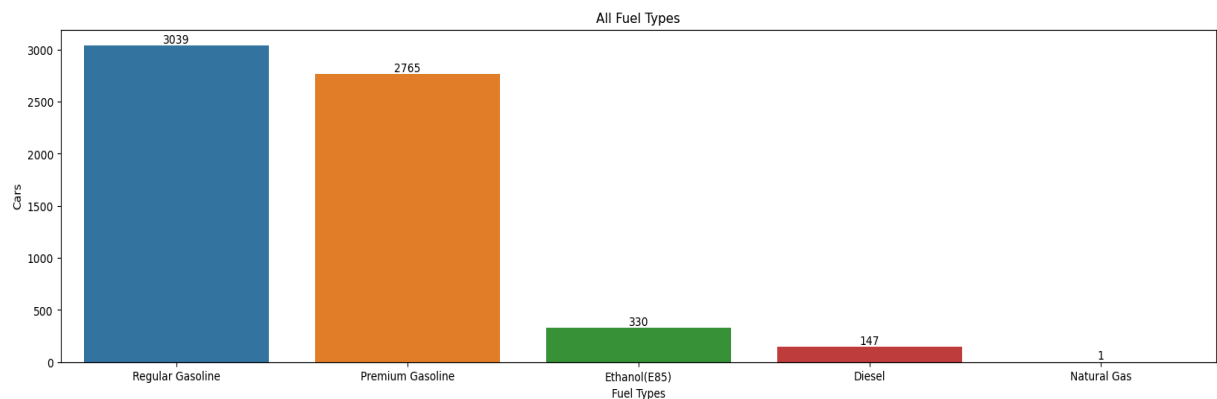
The bar graph compares the number of cylinders across different car models, indicating that the majority of cars, 2749 in total, feature 4 cylinders, followed by 6 cylinders with 2040 cars, and 8 cylinders with 1202 cars, illustrating the prevalent distribution of cylinder counts in car models.

Bar graph showing all types of transmission from different cars:-



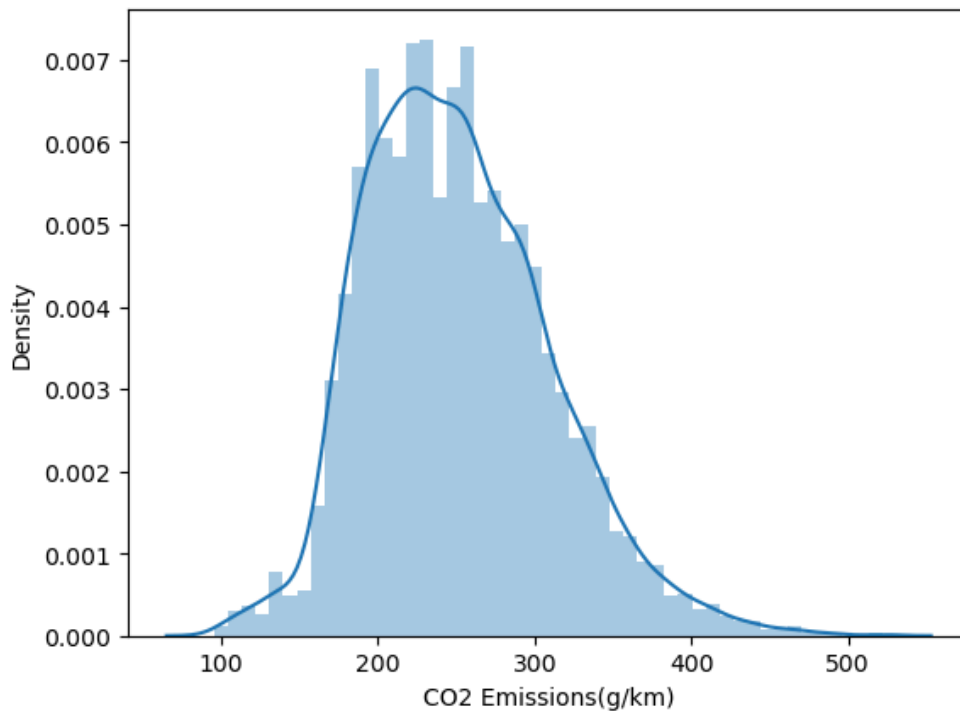
The bar graph represents various types of transmissions in different cars. Automatic with select shift is the most common, with 2722 instances, while continuously variable transmission is the least common, with only 465 occurrences, illustrating the diversity in transmission types across car models.

Bar Graph showing Different types of fuel used by different car models:-



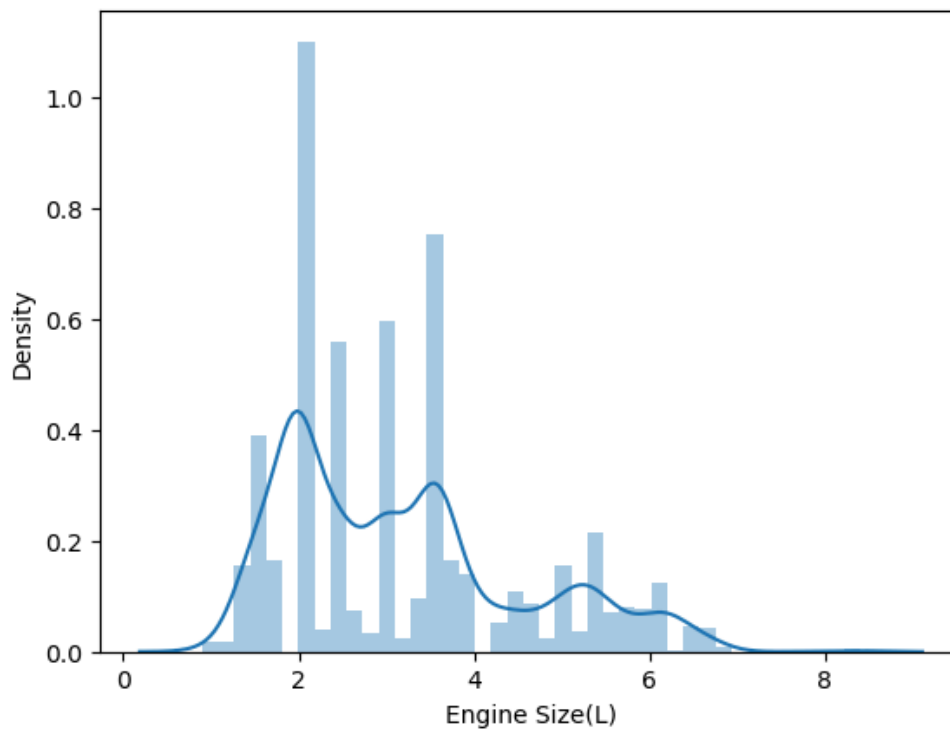
In this, The bar graph illustrates the fuel types utilized by various car models. Regular gasoline and premium gasoline are the most prevalent, with 3039 and 2765 instances respectively, while natural gas is the least utilized, with only one occurrence, demonstrating the dominance of traditional gasoline fuels in the automotive industry.

Histogram representing the distribution of CO2 emission over the dataset:-



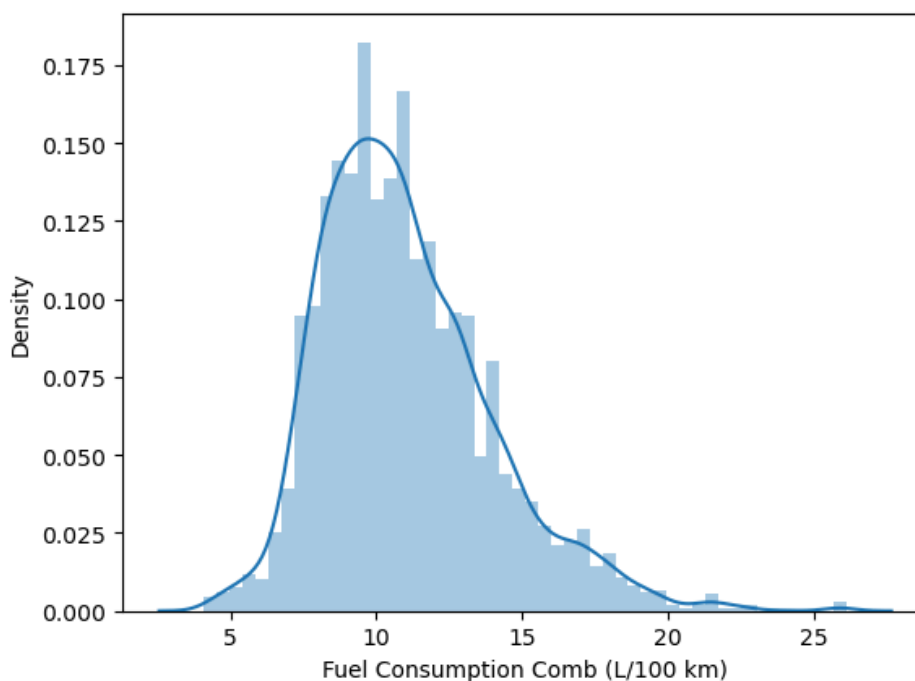
The histogram displays the distribution of CO2 emissions across the dataset. The majority of emissions are above 0.003 density, with the highest density observed at 0.007 and above, while emissions exceeding 500 CO2 units have zero density, indicating a significant gap in emission levels within the dataset.

Histogram displaying Density versus Engine size:-



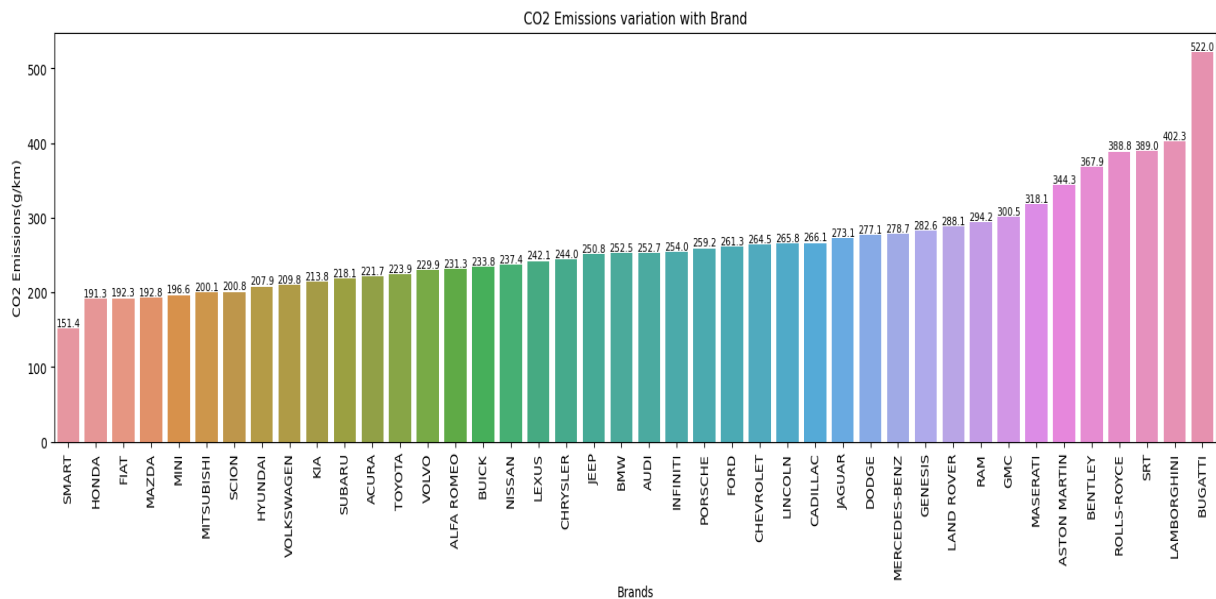
The histogram demonstrates that engine size 2 boasts the highest density, surpassing a value of 1, indicating a concentrated distribution of vehicles with this engine size. However, beyond engine size 7, there is a notable decline in density, suggesting a decrease in the prevalence of vehicles with larger engine sizes in the dataset.

Comparison of Density Versus Fuel consumption Comb using Histogram plot:-



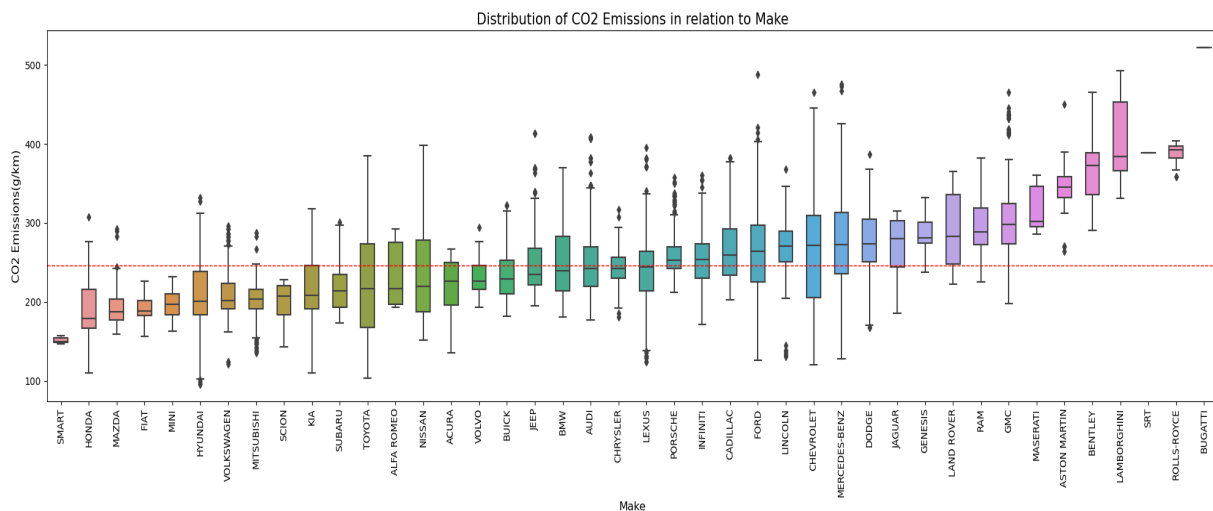
The histogram vividly portrays the relationship between density and fuel consumption for combustion engines. As fuel consumption escalates from 7.5, it reaches its zenith at 10 with a density of 0.175, indicating a concentration of vehicles consuming fuel within this range. However, beyond this peak, there's a steady decline in density, ultimately tapering off to zero after 23.5, suggesting a diminishing number of vehicles with higher fuel consumption rates.

Bar Graph showing CO2 Emissions variation of different car brands:-



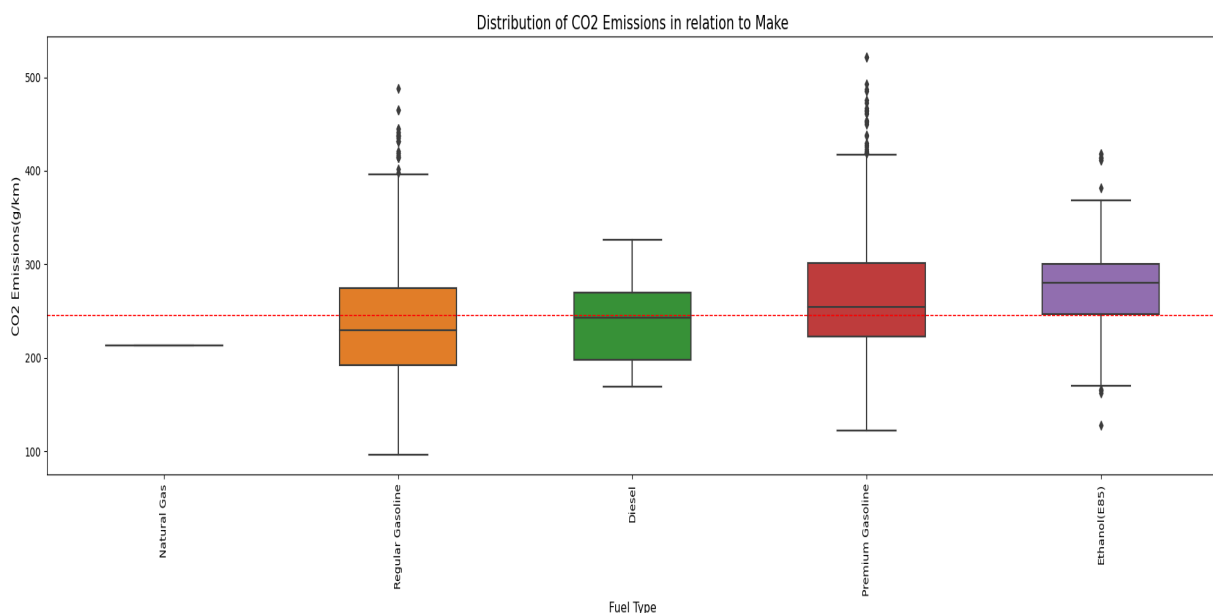
The bar graph effectively illustrates the spectrum of CO2 emissions across various car brands. Bugatti stands out with the highest emission level of 522.0, contrasting sharply with Smart's eco-friendly footprint at 151.4. In between, Honda falls with a moderate emission level of 191.3, showcasing the significant disparities in environmental impact among different automotive manufacturers.

Box plot representing distribution of CO2 emissions in relation to making of the respective car companies:-



A box plot visually represents the distribution of CO2 emissions across different car companies. Each company's emissions data is displayed as a box, with the length of the box indicating the spread of emissions values. The "whiskers" extending from the box show the range of emissions, while any outliers are plotted individually. This graph helps compare the emission levels between different car manufacturers, providing insights into their environmental impact.

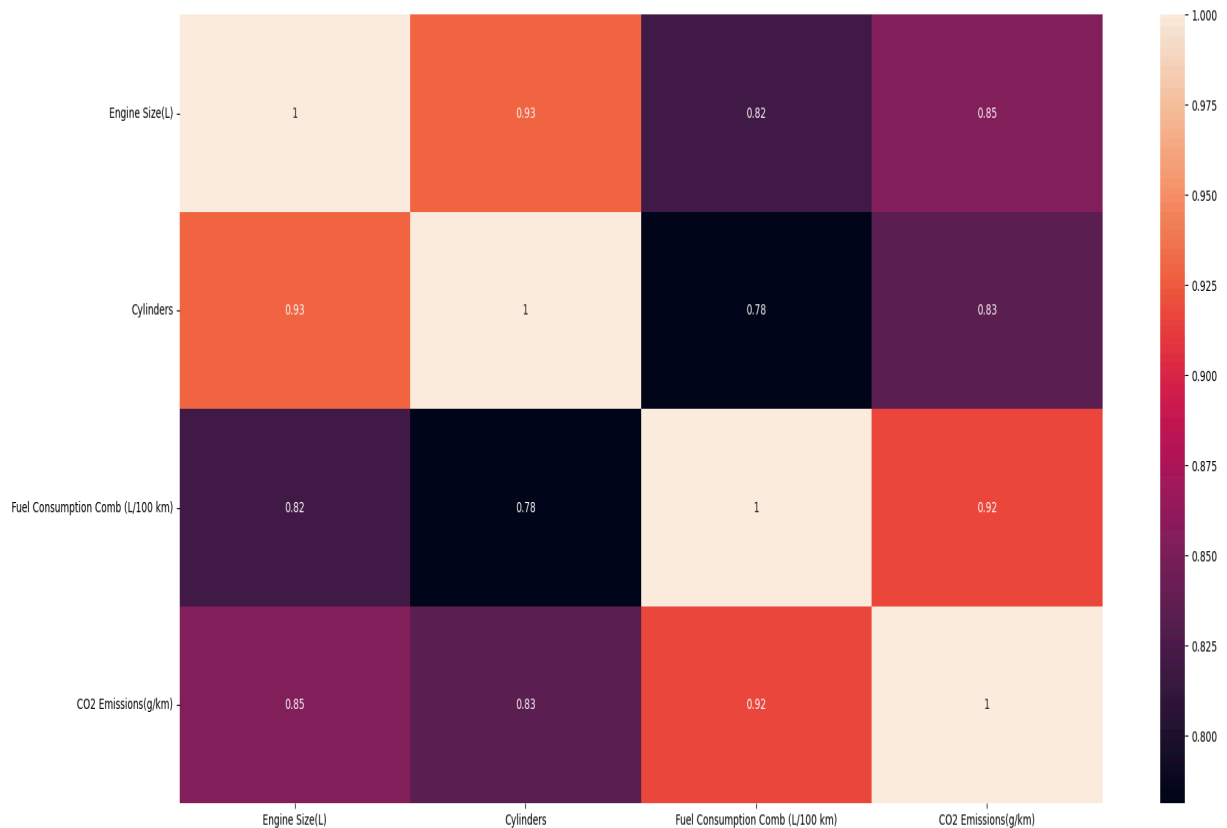
Studying the distribution of CO2 emissions with respect to making of car companies:-



The box plot visually represents the distribution of CO2 emissions across various car companies. Each box corresponds to a different manufacturer, showcasing the range of emissions within their respective fleets. The line within each box denotes the median emission level, typically around 250 CO2 units. Extending from the boxes are "whiskers"

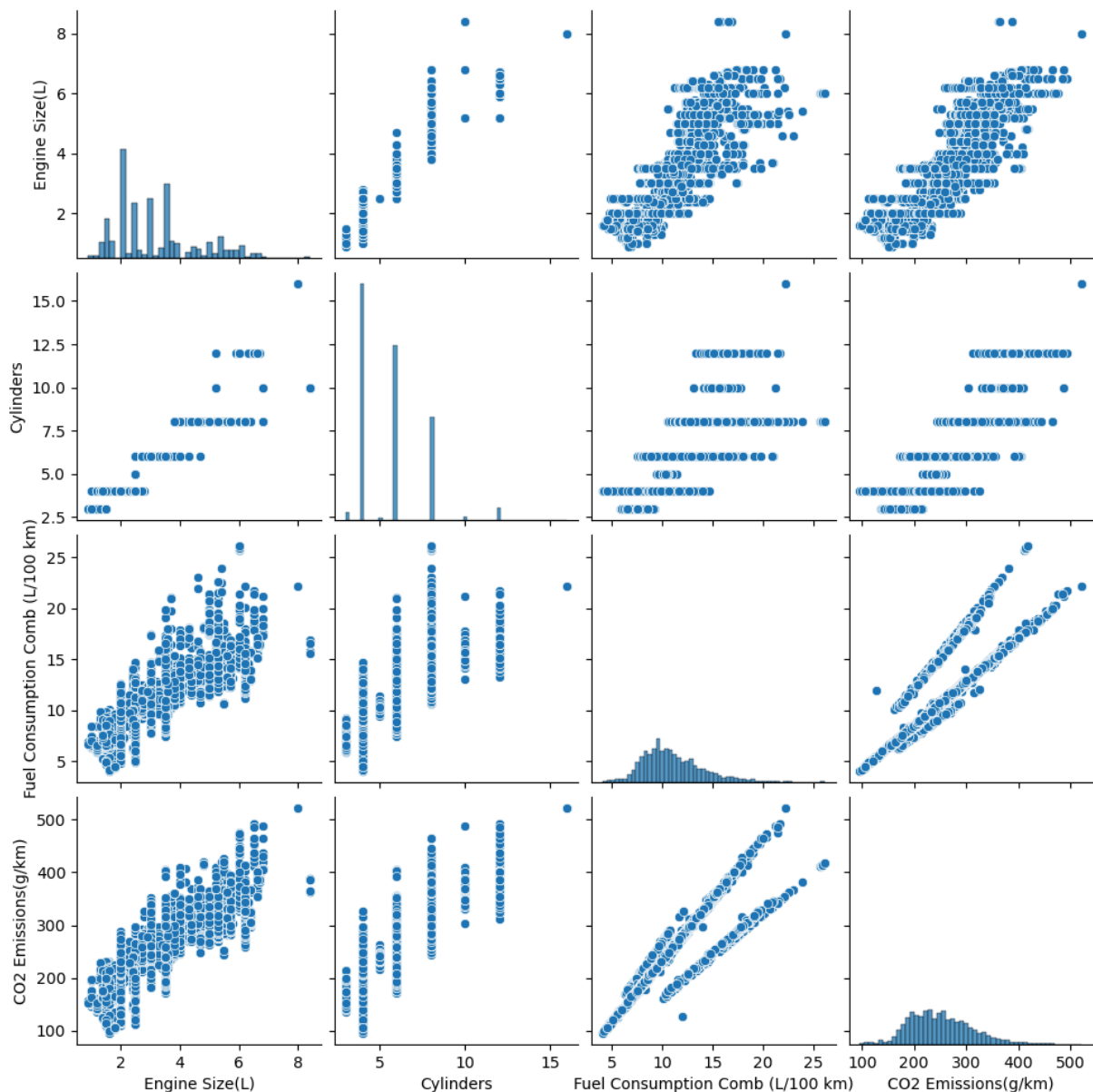
that indicate the full range of emissions, excluding outliers – individual data points lying significantly above or below the bulk of the data. Notably, cars fueled by regular and premium gasoline exhibit a higher prevalence of outliers, suggesting greater variability in emission levels compared to other fuel types.

Studying the dataset by taking different variables or features using Heatmap Analysis :-



In the heatmap analysis of the dataset, colors ranging from black to light orange represent values from 0.800 to 1.00. Each parameter such as engine size, cylinders, fuel consumption combined, and CO2 emissions is examined for their correlations. Darker shades like black indicate weaker correlations, while lighter shades like orange signify stronger correlations between the variables. This analysis helps identify relationships among the parameters, aiding in understanding how engine size, cylinders, fuel consumption, and CO2 emissions interact within the dataset.

Scatter Plot distribution by taking different features or variables from the dataset:-



Studying the dataset with a scatter plot involves plotting various combinations of variables on both the x and y axes. For instance, engine size could be plotted against fuel consumption combined, cylinders against CO2 emissions, fuel consumption against CO2 emissions, and so on. Each point on the plot represents a specific data entry, with its position determined by the values of the variables it represents. By examining these scatter plots, we can visualize the relationships between different pairs of variables, such as how engine size relates to fuel consumption or how CO2 emissions vary with different cylinder counts. This analysis provides valuable insights into the correlations and trends within the dataset.

4.1 Predictive Models Implemented :

1. Linear Regression

Linear regression is a fundamental statistical method used for modeling the relationship between a dependent variable (target) and one or more independent variables (features). In the context of predicting fuel consumption and CO2 emissions for heavy vehicles, linear regression serves as a powerful tool to establish a linear relationship between vehicle characteristics and emission levels.

The general form of linear regression can be expressed as:

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

- \hat{y} is the predicted value.
- n is the number of features.
- x_i is the i th feature value.
- θ_j is the j th model parameter (including the bias term θ_0 and the feature weights $\theta_1, \theta_2, \dots, \theta_n$).

$$\theta_1^{\wedge} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\theta_0^{\wedge} = \bar{y} - \theta_1^{\wedge} \bar{x}$$

where:

- $\theta^{\wedge}1$ is the slope coefficient.
- $\theta^{\wedge}0$ is the intercept coefficient.
- \bar{x} and \bar{y} are the means of the independent and dependent variable.

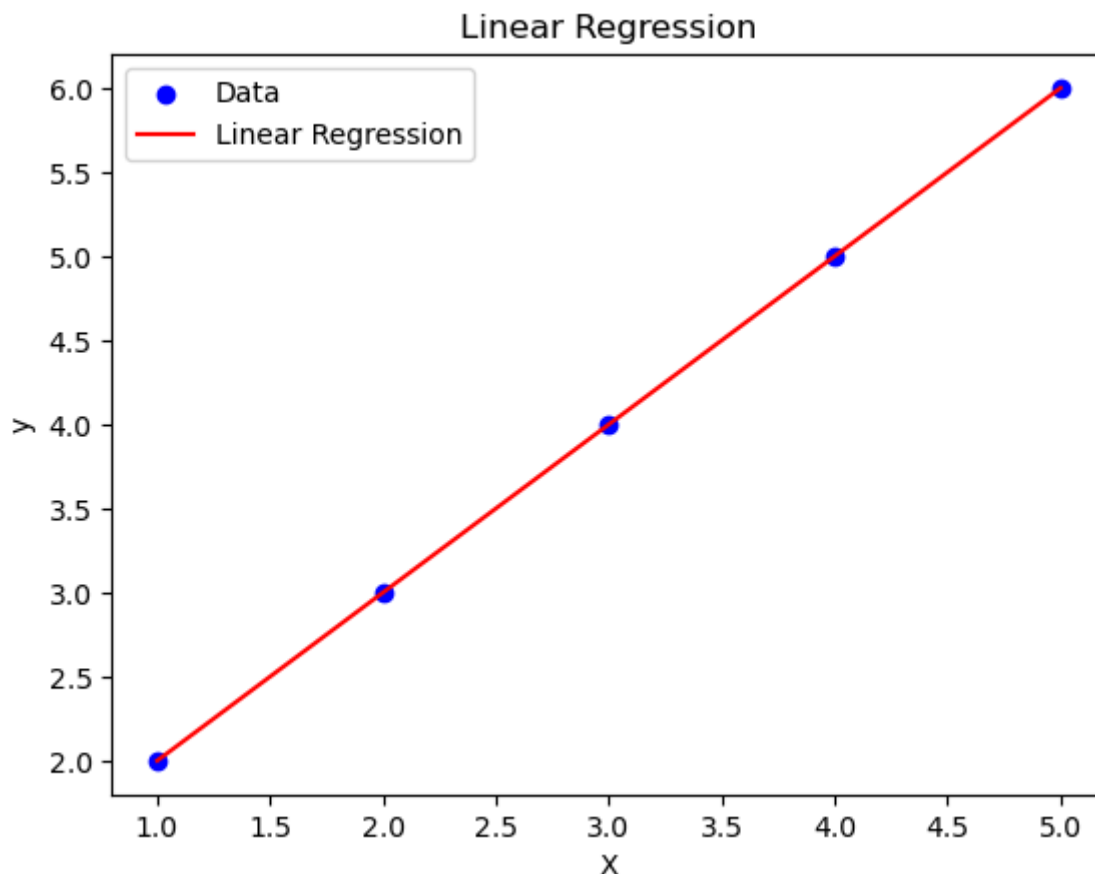
For multiple linear regression (with multiple independent variables), the coefficients are estimated using matrix operations:

where:

$$\hat{\theta} = (X^T \cdot X)^{-1} \cdot X^T \cdot y$$

- $\hat{\theta}$ is the vector of regression coefficients.
- X is the matrix of independent variables.
- y is the vector of dependent variable values.

In the context of this project, linear regression is utilized to model the relationship between vehicle characteristics (such as engine size, cylinders, transmission type, etc.) and CO2 emissions. By fitting a linear regression model to the historical data on heavy vehicles' attributes and corresponding CO2 emissions, the project aims to predict future emissions based on given vehicle features. This predictive capability enables stakeholders in the transportation industry to assess the environmental impact of different vehicle configurations and make informed decisions regarding fleet management, route planning, and fuel efficiency enhancement strategies.



2. Decision Tree

Supervised and unsupervised learning are the two main groups that comprise machine learning. In supervised learning, a certain target variable is chosen, and the computer is taught to learn from the information, forecasting incoming data's responses according to predetermined categories. On the other hand, unsupervised learning finds patterns in unlabeled data without the need for human assistance, exposing the material's innate patterns.

Regression trees and classification trees, which handle continuous and categorical data, respectively, are features of Decision Trees, a method that is essential to supervised learning. Both use a recursive splitting technique that works from the top down until the required homogeneity is reached. Pruning can be used to resolve overfitting, which can result in over-categorization. Data collection, preparation, analysis, testing, training, and use are all part of the basic Decision Tree technique.

investigate Decision Tree categories in more detail, paying particular attention to representation, assessment, and optimization—the three core ideas of machine learning algorithms. Data formatting is involved in representation, user-defined machine learning score is involved in evaluation, and optimization selects the best learner from the evaluation function. The Research Objective Decision Tree is one form of representation that is used in the evaluation of hypotheses. Different methods are included in decision trees, and each has its own metrics for split determination. Two well-known algorithms are CART and ID3. CART handles both continuous and classification data, whereas ID3 only addresses classification variables. For split determination, these algorithms use several measures, including the Information Gain of ID3 and the Gini Index of CART. In addition, C4.5 and pruning—an expansion of ID3 and a method of reducing the complexity of tree nodes—will be discussed.

GINI Index and Information Gain:

A. Gini Split/Gini Split

Classification purity is gauged by the Gini Index, which has a range of 0 to 1. When every data point is in the same class, a Gini Index of 0 denotes full purity; when every data point is in a different class, a Gini Index of 1 denotes entire variety.

B. Information Gain

Prioritizing features is aided by Information Gain, which measures the change in information before and after splitting. Usually, the characteristic that gains the most information is chosen first. It is calculated by contrasting the post-split initial and final entropy values. Entropy indicates the degree of disorder or

uncertainty in the data and, like the Gini Index, has a range of 0 to 1. Entropy is computed using a certain formula.

CART and GINI Index:

Within the context of decision tree algorithms, the Gini index is a statistic that indicates the probability of misclassification for a sample selected at random from a dataset. The CART classification tree algorithm in machine learning evaluates model impurity by using the Gini coefficient rather than the information gain ratio. A lower Gini index guarantees set purity since it indicates a lesser likelihood of misrepresentation within the sample set. The Gini index approaches 0 when every sample in the set is a member of the same class, in contrast to information gain.

One popular technique for creating decision trees is the CART algorithm, which heavily relies on the Gini coefficient. Regression and classification tasks can be performed with CART trees. Whereas node regression in regression trees is based on lowest variance, node classification in classification trees is guided by the Gini value. Recursively defined as either empty or consisting of a single node with left and right pointers pointing to binary trees, CART decision trees are notable for being binary trees. CART, however, works well in situations where data is scarce.

Using the Gini index to determine the best feature and the best binary segmentation point for it, CART makes predictions for categorical data. The Gini index distribution in the case of K classes and P_k , which denotes the likelihood that a sample point belongs to the kth class, is expressed as follows:

$$Gini(p) = \sum_{k=1}^m p_k(1 - p_k) = 1 - \sum_{k=1}^k p_k^2 \quad \dots\dots\dots(1)$$

In order to help determine the best feature and segmentation point for the decision tree, this formula computes the Gini index using the probability distribution.

The sample subset that is part of the kth class in data set D is denoted by C_k , and the sample set D's Gini index is available.

$$Gini(D) = 1 - \sum_{k=1}^k \left(\frac{|C_k|}{|D|} \right)^2 \quad \dots\dots\dots(2)$$

The Gini coefficient of data set D under feature A is displayed as follows if parts D1 and D2 are received and data set D is a subsection on a specific value A according to feature A. Gini coefficient The uncertainty of the set D is shown by Gini (D). Following A=A segmentation, the set D's uncertainty is represented by the Gini coefficient Gini (D, A). There is a correlation: a higher Gini index corresponds to a greater degree of sample set indeterminacy.

$$GainGini(D, A) = \frac{|D1|}{|D|}Gini(D1) + \frac{|D2|}{|D|}Gini(D2) \dots\dots\dots (3)$$

For attribute A, the dataset is divided into two parts by any attribute value, and the Gain–Gini is calculated for each part. The ideal binary scheme provided by attribute A is chosen based on the minimum value.

$$\min_{i \in A} (Gain_Gini(D, A)) \quad (4)$$

$$\min_{A \in Attribute} (\min_{i \in A} (Gain_Gini(D, A))) \quad (5)$$

ID3 and Information Gain:

J. Ross Quilan suggested the classical decision tree algorithm ID3. Based on information entropy, it determines which test attribute is the best and designates the test attribute as the one with the highest information gain value in the current sample set. The test attribute's value determines how the sample set is divided. The decision tree's associated nodes develop new leaf nodes concurrently. Information gain, which describes the change in information before and after the split, is crucial to the ID3 algorithm. When deciding which feature to prioritize first, we frequently use this. Typically, we select the option that provides us with the most information.

Finding the difference between the initial and final entropies—that is, the entropy following the split—is how one gains information. Entropy is a measure of data confusion that goes from 0 to 1, much like the Gini Index.

It can be calculated by this formula:

$$H(Y/X) = \sum_{x \in X} (x)H(Y/X = x) \dots\dots\dots (6)$$

Assuming a data sample set S comprising s data samples, we also assume m distinct values (judgment indicators) for the category attribute: Ci(i = 1, 2, 3, ..., m)

The number of samples in Ci is Si. The overall information entropy for a sample set is:

$$I(S_{1j}, S_{2j}, \dots, S_m) = - \sum_{i=1}^m P_i \log_2 P_i \dots\dots\dots (7)$$

The chance that any sample belongs to Ci is denoted by Pi, and it may also be calculated using Si/S. Let us consider the following scenario: a sample of data S is separated into k subsets {S1,S2...,Sk}, where Sj comprises the sample of aj

value of attribute A in set S. An attribute A is assumed to have K different values $\{A_1, A_2, \dots, A_k\}$. These subsets are new leaf nodes that develop from the nodes of set S if attribute A is chosen as the test attribute. Assuming that there are S_{ij} samples in subset S_j that are classed as C_i , the information entropy value of samples separated based on attribute A is:

$$E(A) = \sum_{j=1}^k \left[\frac{S_{1j}, S_{2j}, \dots, S_{mj}}{S} \times I(S_{1j}, S_{2j}, \dots, S_{mj}) \right] \dots\dots\dots (8)$$

Finally, the information entropy gain obtained after dividing sample set S by attribute A is:

$$Gain(A) = I(S_{1j}, S_{2j}, \dots, S_m) - E(A) \dots\dots\dots (9)$$

3. Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) has become a cornerstone optimization algorithm in machine learning and optimization due to its efficiency, scalability, and broad applicability. The core of SGD is its ability to iteratively update model parameters using small, randomly selected subsets of the training data, called minis. This stochastic sampling method allows SGD to efficiently navigate high-dimensional parameter spaces, making it particularly well suited for the large-scale datasets often encountered in today's machine learning applications.

At the core of the SGD algorithm is an update rule that adjusts the model parameters to the negative gradient of the objective function. Mathematically, this update can be expressed as

$$\theta_{t+1} = \theta_t - \eta \nabla \theta J(\theta; x(i), y(i))$$

Where:-

θ_t denotes the model parameters at iteration t

η is the learning rate

$J(\theta; x(i), y(i))$ is the objective function evaluated in the minibatch $(x(i), y(i))$,

$\nabla \theta J(\theta; x(i), y(i))$ is the gradient with respect to the model parameters of the objective function.

The stochastic nature of SGD introduces randomness to parameter updates, which helps avoid local minima and saddle points more efficiently compared to deterministic optimizations. This inherent stochasticity also provides a form of regularization that leads to better generalization performance of the learned models. While the core SGD algorithm provides a simple and efficient optimization framework, several variations and improvements have been proposed to address its limitations and improve the approach. One such version is Mini-Batch Stochastic Gradient Descent, where the gradient is calculated from small data sets instead of individual data points. This approach achieves a balance between the computational efficiency of SGD and the stability of group gradient descent, providing faster convergence speed and better generalization performance. Another popular SGD enhancement is Momentum, which adds momentum to simplify and accelerate parameter updates. . . approach By incorporating information from previous iterations, impulse SGD enables faster convergence, especially for noisy gradients or poorly developed objective functions. In addition, adaptive learning rate methods such as AdaGrad, RMSProp and Adam dynamically adjust the learning according to historical gradients. of the parameters. These adaptive techniques alleviate the need for manual tuning of the learning rate hyperparameter and provide better convergence performance for many optimization tasks. SGD and its variants find wide applications in many areas of machine learning, including but not limited to deep learning, natural language processing, computer vision and reinforcement learning. In deep learning, SGD forms the backbone of neural network training,

where it is used to iteratively update millions of parameters. Its efficiency and scalability allowed deep neural networks to be trained on massive datasets, leading to breakthroughs in several fields such as image recognition, speech processing and autonomous driving. Finally, Stochastic Gradient Descent is a pioneering optimization algorithm that continues to work. shape the landscape of machine learning and optimization. Its simplicity, efficiency and adaptability make it an invaluable tool for researchers and professionals who want to solve complex optimization problems in various fields. As the field of machine learning evolves, SGD and its variants play a key role in driving innovation and the development of artificial intelligence.

Mathematical Explanation of Stochastic Gradient Descent:

Stochastic Gradient Descent (SGD) is a powerful optimization algorithm used to minimize a given objective function ($J(\theta)$) by iteratively updating the model parameters θ . At each iteration, SGD computes an approximation of the gradient of the objective function using a randomly selected subset of the training data, known as a mini-batch. This stochastic sampling approach allows SGD to efficiently navigate through high-dimensional parameter spaces and find optimal solutions, particularly in scenarios involving large-scale datasets.

Let's delve into the mathematical formulation of the SGD algorithm:

1. Objective Function:

Consider an objective function ($J(\theta)$) that we aim to minimize with respect to the model parameters (θ). This objective function typically represents the loss or cost function associated with a machine learning model, such as mean squared error in regression or cross-entropy loss in classification.

2. Gradient Calculation:

At each iteration t , SGD randomly selects a mini-batch of data points $(x(i), y(i))$ from the training dataset. It then computes the gradient of the objective function with respect to the model parameters θ based on this mini-batch. Mathematically, the gradient approximation can be expressed as:

$$\nabla_{\theta} J(\theta; x(i), y(i))$$

Where:

∇_{θ} denotes the gradient operator with respect to θ , and $J(\theta; x(i), y(i))$ represents the objective function evaluated on the mini-batch $x(i), y(i)$.

3. Parameter Update:

Using the computed gradient approximation, SGD updates the model parameters (θ) in the direction that minimizes the objective function. The update rule for SGD can be expressed as:

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} J(\theta; x(i), y(i))$$

Where θ_t and θ_{t+1} denote the model parameters at iterations t and $t+1$ respectively, and η represents the learning rate hyperparameter. The learning rate controls the step size of the parameter updates and is typically chosen empirically based on the characteristics of the optimization problem.

4. Iterative Process:

The SGD algorithm iterates through the training data, updating the model parameters using mini-batch gradients until a convergence criterion is met or a maximum number of iterations is reached. The convergence criterion is often based on changes in the objective function or the norm of the parameter updates.

5. Stochastic Nature:

The stochastic nature of SGD arises from the random sampling of mini-batches at each iteration. This stochasticity introduces noise into the parameter updates, which helps SGD escape local minima and saddle points more efficiently compared to deterministic optimization methods. Additionally, the randomness in mini-batch selection provides a form of regularization, leading to improved generalization performance of the learned models.

4. Support Vector Regression

Support Vector Regression (SVR) is a versatile machine learning technique that has garnered significant attention and adoption across various industries for its efficacy in predicting continuous target variables. In the domain of heavy vehicle emissions and fuel efficiency analysis, SVR emerges as a pivotal tool, offering a sophisticated framework for exploring the intricate relationships between vehicle characteristics and emission levels.

One of the primary strengths of SVR lies in its ability to model non-linear relationships between features and emissions, a capability that surpasses the limitations often encountered in traditional linear regression methods. By leveraging the kernel trick, SVR efficiently transforms input features into higher-dimensional spaces, enabling the delineation of complex relationships that may remain obscured in lower dimensions. This capacity is particularly critical in the realm of heavy vehicle emissions, where numerous factors interact in multifaceted ways to influence emission levels.

Moreover, SVR exhibits robustness in handling datasets characterized by outliers and noisy data, phenomena that are prevalent in real-world scenarios such as heavy vehicle emissions analysis. Through the utilization of an epsilon-insensitive loss function, SVR effectively mitigates the impact of outliers by imposing a tolerance level (epsilon), thereby enhancing the model's resilience against noisy data points. This ensures that the model can accurately capture the underlying patterns in the data, even in the presence of outliers or other sources of variability.

Furthermore, SVR offers a high degree of flexibility in modeling, thanks to its ability to incorporate various kernel functions such as linear, polynomial, Gaussian radial basis function (RBF), and sigmoid kernels. This versatility empowers practitioners to tailor the model to the specific characteristics of the dataset, ensuring optimal performance and generalization across different scenarios and contexts.

In essence, SVR serves as a powerful and indispensable tool for predicting fuel consumption and CO₂ emissions in heavy vehicles, providing stakeholders in the transportation industry with valuable insights into the environmental impact of different vehicle configurations. By accurately predicting CO₂ emissions, decision-makers can make informed choices regarding fleet management strategies, route planning initiatives, and fuel efficiency enhancement endeavors, thereby contributing to the broader goal of promoting environmental sustainability and reducing carbon emissions in the transportation sector.

1. Linear SVR Formulation:

The linear SVR aims to find a linear function that best fits the data while minimizing the error. The optimization problem can be expressed as:

$$\text{Minimize: } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*)$$

$$\text{Subject to: } y_i - w^T x_i - b \leq \varepsilon + \xi_i \text{ and } w^T x_i + b - y_i \leq \varepsilon + \xi_i^* \text{ for } i = 1, 2, \dots, n$$


Here, w represents the weights, b is the bias term, ξ_i and ξ_i^* are slack variables, C is the regularization parameter, and ε is the epsilon-tube within which no penalty is associated with the error term.

2. Nonlinear SVR Formulation (with Kernel Trick):

For nonlinear regression problems, SVR utilizes the kernel trick to map the input features into a higher-dimensional space. The optimization problem becomes:

$$\text{Minimize: } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*)$$

$$\text{Subject to: } y_i - w^T \phi(x_i) - b \leq \varepsilon + \xi_i \text{ and } w^T \phi(x_i) + b - y_i \leq \varepsilon + \xi_i^* \text{ for } i = 1, 2, \dots, n$$

Here, $\phi(x)$ represents the mapping function  defined by the kernel, allowing the algorithm to implicitly work in a higher-dimensional space without explicitly calculating the transformation.

3. Kernel Functions:

Commonly used kernel functions include:

- Linear Kernel: $K(x_i, x_j) = x_i^T x_j$
- Polynomial Kernel: $K(x_i, x_j) = (x_i^T x_j + c)^d$
- Gaussian RBF Kernel: $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$
- Sigmoid Kernel: $K(x_i, x_j) = \tanh(\alpha x_i^T x_j + c)$

4. Prediction:

Once the model is trained, the prediction for a new instance x is calculated as:

$$\hat{y} = \sum_{i=1}^n (w_i \cdot \phi(x_i)) + b$$

Where w_i are the learned coefficients, $\phi(x_i)$ is the kernel-transformed feature vector of the training data, and b is the bias term.

In the context of this project, SVR plays a central role in modeling the complex interplay between vehicle attributes and CO2 emissions. By leveraging historical data on vehicle characteristics and corresponding CO2 emissions, SVR facilitates the development of robust predictive models capable of forecasting future emission levels based on given vehicle features. This predictive capability enables stakeholders to proactively assess the environmental impact of different vehicle configurations and make data-driven decisions to optimize fleet operations, minimize emissions, and promote sustainable transportation practices.

Furthermore, the application of SVR extends beyond mere prediction, serving as a catalyst for innovation and optimization within the transportation industry. By harnessing the insights gleaned from SVR models, stakeholders can identify areas for improvement, innovate new technologies, and implement targeted interventions aimed at reducing emissions and enhancing fuel efficiency.

5. Random Forest

Random Forests (RF) and Uncertainty Quantification

RF is an ensemble learning method that builds a multitude of decision trees during training [1]. Each decision tree, a non-parametric model, resembles a flowchart and is applicable to both classification and regression problems. It represents the connection between features and the target variable through a series of branching conditions structured in a hierarchical tree. These conditions typically take the form of "feature j greater than threshold for feature j ," where both the feature and the threshold are determined during training.

To understand how a decision tree is constructed, let's consider a classification task with two categories. The training process begins with the entire training data set concentrated at a single node, the tree's root. The algorithm seeks the "optimal split," which involves a feature and a threshold that best segregates objects between the two classes. The definition of "optimal" can be customized within the algorithm, with a popular choice being the Gini impurity measure. The Gini impurity of a group signifies the likelihood that a randomly chosen object will be misclassified if assigned a label randomly drawn from the group's label distribution. Mathematically, for a node n (also referred to as class probabilities), where $P_{n,A}$ and $P_{n,B}$ represent the proportions of objects belonging to classes A and B, the Gini impurity G is expressed as:

$$G = 1 - (P_{n,A}^2 + P_{n,B}^2)$$

The algorithm meticulously examines all possible thresholds for each available feature. For every threshold, it divides the training data into two subsets: a "right group" containing objects falling above the threshold and a "left group" containing objects falling below it. The algorithm then searches for the specific threshold that yields the minimum combined impurity across these two newly formed groups.

$$G_{right} \times f_{right} + G_{left} \times f_{left}$$

where G_{right} , G_{left} are the Gini impurities of the two groups, and f_{right} , f_{left} are the fractions of objects in each group, such that $f_{right} + f_{left} = 1$. The condition of the root is set to be the feature and the corresponding threshold that result in the minimal combined impurity.

Following the establishment of the root node's decision rule, the training data is partitioned into two child nodes based on that rule. Objects satisfying the condition are directed to the right child node, while those that don't proceed to the left. The algorithm then endeavors to find the "optimal split" for the data points in each child node. This process is recursively repeated from top to bottom, organically constructing the tree structure.

The splitting process continues as long as the resulting child nodes exhibit a lower combined impurity compared to their parent node. Nodes that cease to meet this criterion become terminal nodes or leaves, signifying the termination of their respective branches. These terminal nodes are assigned class probabilities based on the distribution of objects that reach them. Consequently, the training process culminates in a tree-like structure where internal nodes contain splitting conditions and leaves represent class labels.

Prediction with Random Forests

To forecast the class of an unlabeled object using a decision tree, we traverse it based on the object's feature values and the conditions at each node until reaching a terminal node. The predicted class is then assigned based on the highest probability within that terminal node.

In its basic form, there are no constraints on the number of nodes or the depth of the resulting decision tree. This can lead to a classifier that performs flawlessly on the training data but poorly on unseen datasets. A single decision tree is susceptible to overfitting the training data and lacks the ability to generalize to new data [1].

The Power of Ensembles: Random Forests

Random Forests (RFs) address this limitation by combining multiple decision trees. Randomness is introduced in two ways:

- **Data Subsets:** Each tree is trained on a randomly selected subset of the entire training data.
- **Feature Subsets:** At each node of every tree, a random subset of features is considered for the splitting criteria.

This randomness reduces the correlation between individual trees, resulting in diverse trees with distinct conditions and structures within the forest. The final prediction from an RF is an ensemble prediction, determined by a majority vote among the trees in the forest. In other words, an unlabeled object progresses through all the trees, and each tree casts a vote for a class. The final prediction aligns with the class receiving the most votes. Additionally, the proportion of votes for the predicted class serves as a measure of confidence in the prediction. While a single decision tree is prone to overfitting, the ensemble approach of RFs has been shown to generalize well to unseen data, leading to improved performance [1].

Probabilistic Random Forest

This section delves into Probabilistic Random Forests (PRF), an algorithm building upon the foundation of Random Forests (RF) for classification tasks.

PRF is designed to enhance prediction accuracy by explicitly considering uncertainties within the input data.

In contrast to traditional RFs that operate on deterministic data points, PRF leverages the inherent probabilistic nature of the input. By incorporating this information, PRF aims to make more nuanced and informative classifications.

The following section will explore the specifics of how PRF utilizes these uncertainties. However, it's important to grasp that PRF builds upon the strengths of RFs while introducing mechanisms to handle data uncertainties, potentially leading to superior classification performance.

Unlike standard Random Forests (RF) that use single point estimates for features and labels, PRF incorporates uncertainties. It treats features and labels as probability distributions. Feature uncertainties (e.g., measurement errors) are reflected in the variance of the distribution, while label uncertainties (e.g., classification ambiguity) are represented as probabilities for each possible class.

This probabilistic approach leads to key differences in how PRF works compared to RF. When encountering feature uncertainty, PRF doesn't make a deterministic left or right split at each node in the decision tree. Instead, objects have a probability of going down both branches, exploring all possible paths through the tree. This probabilistic propagation within PRF trees is illustrated in Figure 1 and Figure 2

Both PRF and regular Random Forests (RF) utilize tree structures where each node represents a condition on a specific feature value. In a standard RF, an object gets directed to either the right or left branch based on whether the node's condition evaluates to true (right branch) or false (left branch) for the object's feature value.

PRF, however, takes a different approach. It calculates probabilities for both branches ($\pi(\text{right})$ and $\pi(\text{left})$) using the uncertainty associated with the object's feature value. Consequently, the object propagates down both branches in a PRF tree, unlike the single deterministic path in a regular RF.

This figure illustrates how objects propagate through various tree structures: a regular Random Forest (RF) tree (left panel), an ideal Probabilistic Random Forest (PRF) tree (middle panel), and our implemented PRF approach (right panel).

The leftmost panel depicts the path of a single object in a standard RF tree. Based on the binary conditions at each node, the object follows a single highlighted trajectory through the tree.

The middle panel showcases the ideal scenario in a PRF tree. Here, the object propagates down all possible branches, with probabilities calculated and stored at each step.

The rightmost panel demonstrates object propagation within our specific PRF implementation (details in Section 3.2). To improve efficiency, this approach incorporates a probability threshold parameter. Branches where the probability falls below this threshold are disregarded (marked with Xs in the figure). Consequently, the algorithm only considers high-probability branches, reducing computational runtime.

Chapter 5

Result and Discussion

5.1 Results Obtained:

1. Comprehensive Market Research: Extensive research was conducted to understand the current landscape of heavy vehicle emissions prediction and mitigation strategies in the transportation sector. Analysis of existing methodologies and technologies revealed a growing interest in leveraging machine learning for predictive modeling in this domain.

2. Development of Predictive Models: As per the project's objectives, predictive models were developed to estimate heavy vehicle carbon emissions based on vehicle characteristics and fuel consumption data. Utilizing techniques such as linear regression, support vector regression (SVR), and random forest, the models were trained and evaluated to accurately predict CO2 emissions.

3. Implementation of Predictive Framework: The developed predictive framework was successfully implemented, allowing stakeholders to assess the environmental impact of different vehicle configurations and fuel efficiency enhancement strategies. The framework provides insights into emission levels and facilitates data-driven decision-making processes in fleet management and route planning.

4. Data Visualization Capabilities: Robust data visualization capabilities were integrated into the system, enabling stakeholders to explore and interpret model predictions effectively. Interactive visualizations such as scatter plots, heatmaps, and trend charts provide intuitive insights into emission trends, feature importance, and model performance metrics.

5.2 Analysis and Discussion:

The analysis of the current market situation revealed a growing interest in leveraging machine learning techniques for predictive modeling in the transportation sector. While several companies have successfully implemented predictive analytics for emission reduction strategies, there remains a gap in the deployment of advanced machine learning models specifically tailored to heavy vehicle emissions prediction.

Despite the successful development and implementation of predictive models in the project, challenges were identified in terms of resource availability and technological requirements. Limited resources for discovering the exact tech stacks required for deploying advanced machine learning models posed challenges in optimizing the predictive framework for real-world applications.

Further analysis is required to bridge the gap between research and practical implementation, with a focus on identifying the optimal technological solutions and resources needed for deploying predictive models effectively in the transportation sector. Collaboration with industry stakeholders and regulatory bodies may facilitate knowledge sharing and resource allocation, ultimately accelerating the adoption of advanced predictive analytics for mitigating heavy vehicle carbon emissions.

CONCLUSION

In the context of this project, the application of linear regression to model the complex relationship between various vehicle characteristics and CO2 emissions provides a basic predictive analysis framework. By analyzing historical data on heavy truck characteristics and their corresponding emissions, this approach allows stakeholders to gain insight into how different characteristics affect emission levels. In particular, the goal of the project is not only a retrospective analysis, but rather a prediction of future emissions based on vehicle specifications. This predictive capability is invaluable to transportation decision makers as it allows them to anticipate and plan for the environmental impact of various vehicles. Support Vector Regression (SVR) is becoming another important tool in this terrain. Unlike linear regression, SVR excels at capturing complex non-linear relationships between variables. This is particularly important for emissions analysis, where the interaction of various vehicle characteristics can be very complex. Additionally, SVR's ability to handle noisy data and adapt to diverse datasets ensures that the model remains robust and accurate even in real-world scenarios. Similarly, the Random Forest algorithm provides an efficient approach to predictive modeling. By aggregating predictions from multiple decision trees using ensemble learning, Random Forest effectively reduces the risk of overfitting while maintaining prediction accuracy. In addition, its ability to assess key importance helps stakeholders understand the causes of emissions, facilitating informed decision-making. At the same time, Stochastic Gradient Descent (SGD) revolutionizes optimization algorithms by providing unparalleled efficiency, scalability and adaptability. Its widespread use in various fields reflects its importance in solving complex optimization problems, including machine learning tasks such as regression and classification. As the transportation industry struggles to reduce emissions and promote sustainability, these methods—linear regression, SVR, Random Forest, and SGD—are pillars of innovation and progress. By utilizing these advanced analytical tools, stakeholders can navigate the complex steps of emissions analysis, optimize fleet management strategies, and ultimately contribute to a greener and more sustainable future for heavy trucks and the broader transportation ecosystem.

References

1. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, Third Edition.*
2. S. Biau, L. Devroye, and G. Lugosi, "Consistency of random forests and other averaging classifiers," *Journal of Machine Learning Research*, vol. 13, pp. 883-904, 2012.
3. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.
4. L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
5. L. Breiman, "Arcing the edge," *Technical Report*, University of California, Berkeley, 1997.
6. T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, Berlin, Heidelberg: Springer, 2000, pp. 1-15.
7. J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367-378, 2002.
8. A. Liaw and M. Wiener, "Classification and regression by randomForest," *R News*, vol. 2, no. 3, pp. 18-22, 2002.
9. P. Cortez and A. Morais, "A data mining approach to predict forest fires using meteorological data," in *Proceedings of the 13th Portuguese Conference on Artificial Intelligence (EPIA 2007)*, vol. 4874, pp. 512-523, Springer, 2007.
10. H. Guo, W. Chen, X. Tan, and Y. Yu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 856-863, 2003.
11. C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC Bioinformatics*, vol. 8, no. 1, p. 25, 2007.
12. S. Chopra, "Neural Networks," *IEEE Transactions on Neural Networks*, vol. 5, no. 3, pp. 537-550, 1994.
13. S. Chawla, "Ensemble classifier for data stream mining," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 613-618, 2003.
14. M. Lichman, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, 2013.
15. S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

“Curbing Carbon on the Road: Machine Learning as a Tool for Greener Heavy Vehicle Operations

ORIGINALITY REPORT

24%

SIMILARITY INDEX

18%

INTERNET SOURCES

16%

PUBLICATIONS

12%

STUDENT PAPERS

PRIMARY SOURCES

1

iopscience.iop.org

Internet Source

5%

2

Yifan Lu, Tianle Ye, Jiali Zheng. "Decision Tree Algorithm in Machine Learning", 2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), 2022

Publication

2%

3

www.coursehero.com

Internet Source

2%

4

Submitted to Saint Joseph's College of Maine

Student Paper

1%

5

"Intelligent Computing Technology and Automation", IOS Press, 2024

Publication

<1%

6

www.banglajol.info

Internet Source

<1%

7

export.arxiv.org

Internet Source

<1%

8	journals.plos.org Internet Source	<1 %
9	www.rc.is.ritsumei.ac.jp Internet Source	<1 %
10	www.researchgate.net Internet Source	<1 %
11	escholarship.org Internet Source	<1 %
12	repository.tudelft.nl Internet Source	<1 %
13	Submitted to Imperial College of Science, Technology and Medicine Student Paper	<1 %
14	Submitted to University of Essex Student Paper	<1 %
15	arxiv.org Internet Source	<1 %
16	Submitted to CSU, San Diego State University Student Paper	<1 %
17	dokumen.pub Internet Source	<1 %
18	medium.com Internet Source	<1 %
19	ftp.ics.uci.edu Internet Source	

<1 %

20

Submitted to KIIT University

Student Paper

<1 %

21

Long Zhao, Xinbo Zhao, Yuanze Li, Yi Shi, Hanmi Zhou, Xiuzhen Li, Xiaodong Wang, Xuguang Xing. "Applicability of hybrid bionic optimization models with kernel-based extreme learning machine algorithm for predicting daily reference evapotranspiration: a case study in arid and semiarid regions, China", Environmental Science and Pollution Research, 2022

Publication

<1 %

22

Batyrkhan Omarov, Sayat Ibrayev, Arman Ibrayeva, Bekzat Amanov, Zeinel Momynkulov. "Optimal Leg Linkage Design for Horizontal Propel of a Walking Robot Using Non-dominated Sorting Genetic Algorithm", IEEE Access, 2024

Publication

<1 %

23

essay.utwente.nl

Internet Source

<1 %

24

nmamit.nitte.edu.in

Internet Source

<1 %

25

Submitted to American University of Central Asia

<1 %

26

epdf.pub

Internet Source

<1 %

27

zero.sci-hub.se

Internet Source

<1 %

28

Tuhuo Jia, Wenhao He, Wenzhe Ma.
"Optimizing urban energy management: A
strategic examination of smart grids and
policy regulations", Sustainable Cities and
Society, 2024

Publication

<1 %

29

Xiaoming Zang. "Accessibility Analysis Model
of College Students' Employment Education
Resources Based on Mobile Search
Algorithm", Journal of Interconnection
Networks, 2022

Publication

<1 %

30

onlinelibrary.wiley.com

Internet Source

<1 %

31

Dezhi Cao, Licheng Wu, Yue Zhao, Zhenna Lu.
"Integration of Knowledge-Driven and Data-
Driven Based Korean Phonetic Transcription",
2022 5th International Conference on Pattern
Recognition and Artificial Intelligence (PRAI),
2022

Publication

<1 %

32 Ehsan Harirchian, Seyed Ehsan Aghakouchaki Hosseini, Viviana Novelli, Tom Lahmer, Shahla Rasolzade. "Application of machine learning methods to assess seismic fragility of non-engineered masonry buildings", Results in Engineering, 2024
Publication

33 Submitted to South Coast Baptist College
Student Paper

34 Submitted to Coventry University
Student Paper

35 Han Liu, Alexander Gegov, Mihaela Cocea. "Nature and biology inspired approach of classification towards reduction of bias in machine learning", 2016 International Conference on Machine Learning and Cybernetics (ICMLC), 2016
Publication

36 Submitted to University of Salford
Student Paper

37 Yu, L.. "Probabilistic principal component analysis with expectation maximization (PPCA-EM) facilitates volume classification and estimates the missing data", Journal of Structural Biology, 201007
Publication

38

Internet Source

<1 %

39

journals.iqra.edu.pk

Internet Source

<1 %

40

Submitted to De Montfort University

Student Paper

<1 %

41

Erdem Küçüktopçu, Bilal Cemek, Halis Simsek.
"Comparative analysis of single and hybrid
machine learning models for daily solar
radiation", Energy Reports, 2024

Publication

<1 %

42

Submitted to University of Strathclyde

Student Paper

<1 %

43

deepai.org

Internet Source

<1 %

44

Submitted to University of Melbourne

Student Paper

<1 %

45

www.mdpi.com

Internet Source

<1 %

46

Submitted to Athlone Institute of Technology

Student Paper

<1 %

47

www.timesnownews.com

Internet Source

<1 %

48

Submitted to American University

Student Paper

<1 %

49	Submitted to Belgium Campus iTversity NPC Student Paper	<1 %
50	Submitted to University of Bristol Student Paper	<1 %
51	Submitted to University of East London Student Paper	<1 %
52	Submitted to University of Huddersfield Student Paper	<1 %
53	ouci.dntb.gov.ua Internet Source	<1 %
54	ieomsociety.org Internet Source	<1 %
55	scholarworks.uno.edu Internet Source	<1 %
56	steinmetz.union.edu Internet Source	<1 %
57	thesai.org Internet Source	<1 %
58	www.themuse.com Internet Source	<1 %
59	"Advanced Data Mining and Applications", Springer Science and Business Media LLC, 2010 Publication	<1 %

60

Huiyi Su, Xinyu Wang, Wei Chen, Ning Ding, Xiaolei Cui, Mengqi Bai, Zhili Chen, Mingshi Li. "A novel framework for identifying causes of forest fire events using environmental and temporal characteristics of the ignition point in fire footprint", Ecological Indicators, 2024

Publication

<1 %

61

digitalcommons.fiu.edu

Internet Source

<1 %

62

m.moam.info

Internet Source

<1 %

63

A. Vande Wouwer, C. Renotte, Ph. Bogaerts. "A short note on SPSA techniques and their use in nonlinear bioprocess identification", Mathematical and Computer Modelling of Dynamical Systems, 2007

Publication

<1 %

64

Daniela Stojanova, Andrej Kobler, Peter Ogrinc, Bernard Ženko, Sašo Džeroski. "Estimating the risk of fire outbreaks in the natural environment", Data Mining and Knowledge Discovery, 2011

Publication

<1 %

65

Submitted to Sim University

Student Paper

<1 %

66

dspace.daffodilvarsity.edu.bd:8080

Internet Source

<1 %

67	meta-learn.github.io Internet Source	<1 %
68	open-innovation-projects.org Internet Source	<1 %
69	www.simplilearn.com Internet Source	<1 %
70	5wwwwww.easychair.org Internet Source	<1 %
71	Julia Vázquez-Escobar, J.M. Hernández, Miguel Cárdenas-Montes. "Estimation of Machine Learning model uncertainty in particle physics event classifiers", Computer Physics Communications, 2021 Publication	<1 %
72	Submitted to Vels University Student Paper	<1 %
73	docs.com Internet Source	<1 %
74	escholarship.mcgill.ca Internet Source	<1 %
75	ijisrt.com Internet Source	<1 %
76	www.grafiati.com Internet Source	<1 %

77

"Intelligent Robotics and Applications",
Springer Science and Business Media LLC,
2021

Publication

<1 %

78

www.ncbi.nlm.nih.gov

Internet Source

<1 %

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off