MAHMOUD ABUBAKAR

ML-BASED ECO-FRIENDLY MOBILITY: PREDICTING FUEL, EMISSIONS AND SAVINGS IN CONVENTIONAL AND ELECTRIC VEHICLES

MASTER THESIS

2024

NEAR EAST UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

DEPARTMENT OF ARTIFICIAL INTELLIGENCE ENGINEERING

ML-BASED ECO-FRIENDLY MOBILITY: PREDICTING FUEL, EMISSIONS AND SAVINGS IN CONVENTIONAL AND ELECTRIC VEHICLES

M.Sc. THESIS

MAHMOUD ABDUSWAMAD OMAR ABUBAKAR

Nicosia

January, 2024

NEAR EAST UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

DEPARTMENT OF ARTIFICIAL INTELLIGENCE ENGINEERING

ML-BASED ECO-FRIENDLY MOBILITY: PREDICTING FUEL, EMISSIONS AND SAVINGS IN CONVENTIONAL AND ELECTRIC VEHICLES

M.Sc. THESIS

MAHMOUD ABDUSWAMAD OMAR ABUBAKAR

Supervisor

Prof. Dr. FADI AL-TURJMAN

Nicosia

January, 2024

Approval

We certify that we have read the thesis submitted by MAHMOUD ABUBAKAR titled "**ML-Based Eco-Friendly Mobility: Predicting Fuel, Emissions and Savings in Conventional and Electric Vehicles**" and that in our combined opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Educational Sciences.

Examining Committee	Name-Surname	Signature
Head of the Committee:	Prof. Dr. Fadi AL-Turjman	....
Committee Member:	Asst.Prof.Dr. Abdullahi Umar İbrahim	....
Committee Member:	Asst.Prof.Dr.Mubarak Auwalu Saleh	....A. Mubarak....
Committee Member:	Yard. Assoc. Dr. Cengiz Mesut Bükeç	....
Supervisor:	Prof. Dr. Fadi AL-Turjman	....

Approved by the Head of the Department

..1../.2./2024

Prof. Dr. Fadi AL-Turjman

Head of Department

Approved by the Institute of Graduate Studies

NEAR EAST UNIVERSITY INSTITUTE OF GRADUATE STUDIES 1988

..1../..2./2024

Prof. Dr. Kemal Hüsnü Can Başer

Head of the Institute

Declaration

I hereby certify that all of the material, papers, analyses, and results given in this thesis are in accordance with the academic norms and ethical principles established by the Institute of Graduate Studies at Near East University. Furthermore, I swear that I have rigorously credited and referenced information and data not originating from this study in accordance with these requirements.

Mahmoud Abubakar

10/1/2024

Acknowledgments

All gratitude be to Allah for allowing me to go on the journey for my masters. It has been a transforming experience distinguished by knowledge acquired, obstacles overcome, and continuous improvement. The support and encouragement of numerous people has made this journey possible overcoming all the obstacles. My deepest appreciation goes to Prof. Dr. Fadi Al-Turjman, my supervisor. His tremendous advice, steadfast support, and encouragement all helped greatly to the successful completion of my research.

I would also like to thank my family for being with me through thick and thin showering their love and support throughout my scholarly journey, and I am eternally grateful for their unrelenting encouragement. My dad, Mr. Abduswamad Omar, mom Mrs Zeina Said and brothers have been the supporting pillars through all my ups and downs.

My Special thanks are due to my course mate, Mr. Ibrahim Ame for his collaborative efforts and knowledge sharing. I am also particularly grateful to my childhood friends, Mr. Hemed Fernandes, Mr. Ahmed Zakwan and Mr. Lukman Fihri for their continuous encouragement and valuable inputs.

Mahmoud Abubakar

ABSTRACT

ML-BASED ECO-FRIENDLY MOBILITY: PREDICTING FUEL, EMISSIONS AND SAVINGS IN CONVENTIONAL AND ELECTRIC VEHICLES

Mahmoud Abduswamad Omar Abubakar

MSc Department of Artificial Intelligence Engineering

12.27.2023. 83 pages

The transport industry is among the top contributors to human-made Greenhouse Gases (GHG). One of the (GHGs) is carbon emissions, which contribute to an increase in global temperature. Moreover, airborne pollutants lead to human health risks. That being said, a need for simplified and effective techniques arises to assist in controlling emissions and reducing fuel costs by forecasting fuel consumption and CO_2 emissions for Internal Combustion Engine (ICE) cars. Driven by these objectives, this research explores machine learning techniques to create a new era of environmentally responsible transportation. This study aims to use machine learning techniques to predict fuel consumption and CO_2 emissions of different conventional vehicles. It also compares Internal Combustion Engine (ICE) cars with a specified Electric Vehicle (EV) efficiency in fuel and charging costs. Regression analysis and machine learning methodologies were used to explore a large dataset with diverse ICE vehicle characteristics from 2016 to 2023. The data was pre-processed, analysed using Exploratory Data Analysis (EDA), and then trained using machine learning models such as the Support Vector Regression (SVR), Passive Aggressive Regressor (PAR), Extra Tree Regressor (ETR), Extreme Gradient Boosting Regressor (XGB), Bagging Regressor (BR), Tweedie Regressor (TR), and k-Nearest Neighbours Regressor (KNN). The ETR model fared the best among the models, proving its capacity to estimate fuel consumption with an R2 of 0.97 and an R2 of 0.94 in predicting CO_2 emissions for conventional vehicles. The project also includes creating a user-friendly online application that provides users with real-time forecasts which will be fetched from the deployed trained model. The incorporation of geographical information via Map box improves forecast accuracy by taking driving distance into account. The research intends to assist sustainable transportation by offering valuable insights into environmentally friendly mobility alternatives and aiding automobile decision-making. The results showed that car users will save more if they use EVs. The study also recommends more use of predictive methods to be applied in other modes of transport in order to reduce carbon emissions.

Key Words: Machine learning, electronic vehicles, internal combustion engine, predictions, carbon emission.

Table of Contents

List of figures

List of tables

List of Abbreviations

Abbreviation	Full Form
GHG	Greenhouse Gas
IEA	International Energy Agency
ICE	Internal Combustion Engine
EV	Electric Vehicle
MAE	Mean Absolute Error
MSE	Mean Squared Error
AI	Artificial Intelligence
IoT	Internet of Things
CO_2	Carbon Dioxide
RF	Random Forest
LR	Linear Regression
GB	Gradient Boosting
PLS	Partial Least Squares
GPR	Gaussian Process Regression
QR	Quantile Regression
OLS	Ordinary Least Squares
SVR	Support Vector Regression
SVM	Support Vector Machine
NB	Naïve Bayes
UPR	Univariate Polynomial Regression
MLR	Multiple Linear Regression
MPL	Multivariate Polynomial Regression
CNN	Convolutional Neural Network
MBE	Mean Bias Error
RMSE	Root Mean Squared Error
API	Application Programming Interface
PHP	Hypertext Pre-processor
WSGI	Web Server Gateway Interface
kWh	Kilowatt-hour
Tl	Turkish Lira

JSON	JavaScript Object Notation
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
TRNC	Turkish Republic of Northern Cyprus
GBDT	Gradient Boosting Decision Tree
DT	Decision Tree
PEMS	Portable Emissions Measurement System

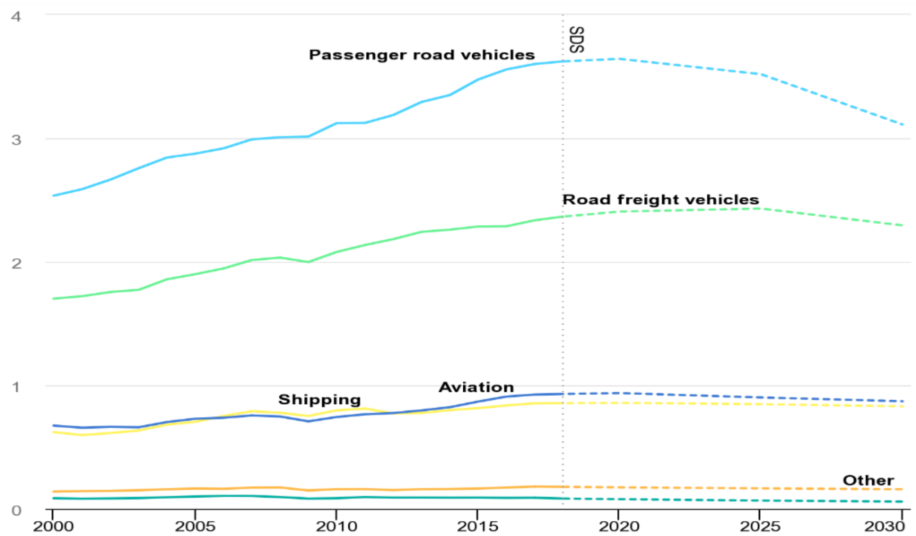
Chapter 1

Introduction

Throughout the decades, the earth has been witnessing fluctuations in temperature, experiencing warmer and colder phases than average. Human activities have been primarily attributed to rapid global warming in the new era. One of the human activities is transport. Several studies have illustrated the significant contribution of the transportation sector to emissions (Grazi et al., 2008; Y. H. Liu et al., 2019; Michaelis, 1993; Mustapa & Bekhet, 2016; Ross Morrow et al., 2010).

Figure 1.1

Emissions in transport industry (IEA, 2019)



Land transportation is the most commonly used mode of transportation and shares three-quarters of total emissions, as explained in the Intergovernmental Panel on Climate Change (2012). The emissions are shown in the accompanying chart in Figure 1.1. Shipping and aviation sectors emit the least GHG, followed by road freight and passenger vehicles. The increasing number of vehicle users influences this hierarchy as cars become more affordable.

Goals twelve and thirteen of the Sustainable Development Goals (SDGs) are specifically relevant to this project. The twelfth aim is to guarantee that consumption and production patterns are sustainable enough to support the livelihoods of current and future generations. Climate change is addressed in the twelfth objective, where urgent action is required to handle climate change and its consequences. In response, the advent of new technologies, such as advanced computers, signifies a watershed moment in human activities, especially in the transport sector. The capabilities of machine learning significantly influence how we predict and manage fuel consumption, carbon emissions, and costs in various vehicles, including both conventional models and electric vehicles. As we move into an era where technology intertwines with ecological responsibility, we need a trailblazer, empowering users with granular insights into the complex interplay of vehicle specifications and their broader ecological footprint. The machine learning underpinning this endeavor extends its scope beyond predictive analytics, marking the beginning of a new era characterized by informed decision-making within the world of transportation.

This research was conducted for more than one year at the Near East university's innovation center. Originating as an investigation into predicting fuel consumption and carbon emissions in air transportation, the focus shifted to cars due to data accessibility challenges. Air transportation emits a lot of carbon emissions too but accessing data from different flights is hard as it is very sensitive. This project was done with an aim to help reduce carbon emissions and bring about fuel efficiency. This will be accomplished by demonstrating the difference in fuel usage, fuel cost, and CO_2 emissions between conventional and electrical vehicles.

Data from conventional vehicles that was used in this research was accessed online containing different car classes, their specifications, fuel consumption and emissions. Machine learning models were then trained using Google Colab. The model with the best R2 score, MAE and MSE was chosen for deployment in a web app. The EV data was simulated from the vehicle's specifications and added to the web app for comparison purposes. The webapp developed, provides users with a multifaceted lens to evaluate their choices, this initiative seeks to catalyze a paradigm shift, steering individuals towards more responsible and ecologically sensitive vehicular preferences. By comparing the fuel costs in ICE and EVs, users will have a first-hand point of view on which vehicle is the better option. The research problem, significance, limitations, motivation and problem statements are explained in the following sections.

Research problem

The transport industry has influenced greenhouse gas emissions, especially carbon emissions. This leads to a rise in the need for new techniques to forecast fuel consumption and emissions in conventional cars to have control. Addressing this problem requires leveraging machine learning techniques to provide accurate and timely predictions, promote environmentally responsible transportation choices, and inform decision-making within the automotive sector. Beyond the focus on fuel consumption, the research widens its scope to explore the broader environmental implications of different ICE vehicle specifications, aiming to establish a thorough understanding of the ecological footprint associated with these variations. There is also an increase in demand for comparative assessments of conventional and electric vehicles (EVs), focusing on estimating fuel and charging costs. The central aim is to investigate whether long-term economic and environmental benefits balance the fuel cost difference between refueling ICE and EV vehicles. The research aims to offer nuanced insights into the dynamic nature of vehicle performance without the need for specific driving scenarios.

Significance

The significance of this study transcends conventional boundaries, positioning itself as a potentially transformative force in sustainable transportation. By harnessing the formidable power of machine learning, this research furnishes users with meticulously detailed and real-time estimates of their vehicles' environmental impact. It actively cultivates a heightened awareness of eco-friendly driving practices. The implications extend beyond individual users, making noteworthy contributions to broader ecological conservation endeavors. In a global context where sustainable transportation is pivotal in various initiatives, this research proffers a practical and accessible tool. The web app developed will allow users to actively participate in reducing carbon emissions, contributing to a greener and more sustainable future.

Limitations

Transitioning from the exploration of vehicle specifications and their impact on fuel efficiency, it's crucial to acknowledge the inherent limitations of the proposed model. The precision of predictions intricately links to the quality and representativeness of the training data. Some assumptions were made during the modelling process where there was disregard of unforeseen changes in driving conditions, variations in road infrastructure, or the fluid nature of fuel prices. Additionally, the model's applicability may exhibit subtle regional variations, and its precision might fluctuate based on individual driving behaviours. Despite these acknowledged limitations, the model stands as an innovative and invaluable tool. It serves as a pioneering force, steering users towards more sustainable travel practices while actively contributing to ongoing advancements in the field.

Motivation

The motivation propelling this research is deeply rooted in the urgency to address the far-reaching environmental consequences of conventional transportation practices. The consequences of climate change are no longer a distant future; therefore, more eco-friendly practices are needed. The environment study also seeks to inspire individuals to make informed decisions by synergizing machine learning capabilities with real-time data and mapping features.

Problem statement

A notable hole remains in today's world of multidimensional transportation challenges, and that is a comprehensive and user-centric tool that analyses both conventional and electric cars. This research, therefore, addresses this problem, filling the gap and giving users first-hand information to make informed choices. The optimization of fuel efficiency in ICE vehicles stands as a paramount challenge with significant implications for environmental sustainability. This research explores the intricate dynamics of vehicle specifications, including engine sizes, fuel injection systems, and transmission mechanisms, and their profound impact on fuel consumption. That being said, the research questions are as follows:

a) How are machine learning techniques effective in forecasting fuel consumption and carbon emissions in Internal Combustion Engine (ICE) cars?
b) How do ICE vehicle specifications affect fuel consumption and carbon emissions?
c) How do the Internal Combustion Engine (ICE) vehicles compare with Electric Vehicles (EVs) in terms of fuel and charging costs and efficiency?

The consequent sections of the thesis include Chapter 2, where we look into the work done by other researchers on the study topic. Chapter 3 gives the methodology of this study, and the results are explained and discussed in chapter 4. Lastly, chapter 5 presents the conclusion and recommendations for future work.

Chapter 2

Literature review

Banking, healthcare, manufacturing, agriculture, transportation, automobile and many more industries have achieved success using new technologies. The potential of artificial intelligence is having a significant influence on the automotive industry environment. Customers' preferences for enhanced and novel features, driver assistance, self-driving, and so on have accelerated the implementation of artificial intelligence in the automobile sector. Artificial intelligence is being employed in many phases of the automobile industries, ranging from autonomous driving (Kumari & Bhat, 2021; Othman, 2022; Tewari et al., 2021) to manufacturing, production, supply chain and driver safety. There are several studies from Internal Combustion Engine (ICE) vehicles and some EVs literature on fuel consumption and emissions presented. They present the use of various AI techniques in making predictions.

Related work

Reducing CO_2 emissions in road transport is crucial. Bappon et al. (2022) utilized machine learning for predicting the CO_2 rating of fuel-consuming vehicles in Canada, employing eight techniques. The Random Forest (RF) model excelled with a 96% accuracy, while the Naïve Bayes (NB) classifier performed least with 73%. Additionally, Bousonville et al. (2019) emphasized factors like weight, speed, driving style, and weather impacting fuel usage. They utilized RF, Linear Regression (LR), and Gradient Boosting (GB) models and incorporated weather data for improved accuracy. The GB model performed exceptionally well, achieving an R-squared value of 0.84 with weather data. Furthermore, temperature and wind speed were identified as significant influencers on fuel consumption. To assess the correlation between fuel efficiency and other features, Yin et al. (2016) suggested employing a Mutual Information Index (MII)-based technique. The authors analysed and predicted vehicle fuel efficiency using over 10,000 real-world vehicle samples. Employing regression methods such as Partial Least Squares (PLS), Gaussian Process Regression (GPR), Quantile Regression (QR), Ordinary Least Squares (OLS), and Support Vector Regression (SVR), they found that QR demonstrated the highest prediction accuracy. In another study, Hamed et al. (2021) developed a fuel consumption prediction model using the SVM algorithm. The dataset included On-Board Diagnostics (OBD) data, predicting fuel consumption based on Mass Air Flow (MAF), vehicle speed, Revolutions Per Minute (RPM), and throttle position sensor features. RF and decision tree (DT) algorithms were used for feature weighting, revealing that the Vehicle Speed Mass Air Flow (VS_MAF)-based equation achieved a higher accuracy with an R-Squared value of 0.97 compared to the original equations.

It is very important as an EV user that you know the battery life. This will really come in handy in preventing unforeseen problems such as getting stuck on the road with a low charge battery in the vehicle. Mahesh Prasanna et al. (2022) used SVM, RF and Gradient Boosting Decision Tree (GBDT) to predict the battery life in electrical vehicles. The authors aimed to address the issue of battery degradation. The effectiveness of the models was evaluated based on accuracy and loss values, and it was concluded that the SVM method performed the best with a high accuracy of 97.3%. Cabani et al. (2021) argued that the main factors that influence energy consumption in electric vehicles are the environment, driver behaviour and the vehicle. On this basis, the authors used k-NN algorithm to predict the energy consumption in EVs. They also proposed an algorithm for finding the optimal path in terms of energy consumption using a heuristic estimation function based on the energy consumption model. The results from the simulation demonstrate the potential of the proposed model for accurately estimating energy consumption and finding optimal paths for EVs having an accuracy of 96.5%.

Katreddi et al. (2023) developed a predictive model for heavy-duty vehicle maintenance costs, encompassing diesel and alternative fuels. The mixed-effects random forest model demonstrated exceptional accuracy, achieving a 98.96% R2 score for the training dataset and 94.31% for the validation dataset. The model's efficacy was consistently proven across diverse heavy-duty vehicles and fuel types, spanning school buses, delivery trucks, vocational trucks, refuse trucks, and goods movement trucks. Moreover, Li et al. (2019) used a Multilayer Perceptron (MLP) method to estimate the real-world fuel consumption rate of light-duty vehicles. The authors used a large dataset from the Bear Oil database in China, consisting of over 2 million samples for model optimization. The MLP algorithm regression had 417-dimensional input vectors representing various features such as city, brand, vehicle type, engine parameters, and transmission type. The results showed that differences exist between real-world fuel consumption and standard fuel consumption, with actual fuel consumption generally higher than the standard. The MLP model performed well in predicting fuel consumption, and the sensitivity analysis provides insights into the factors affecting the predictions. Still on the light duty vehicles, Hien and Kor (2022) focused on analysing and predicting fuel consumption and carbon dioxide emissions with the aim of delivering a comparative view of different vehicle brands and models. The study used a dataset containing fuel consumption and CO_2 for 4974 samples of light duty vehicles. The authors used Univariate Polynomial Regression (UPR) model which acquired an accuracy of 98.6%. There are other models used including Multiple Linear Regression (MLR) and Multivariate Polynomial Regression (MPL) which had an accuracy of approximately 75%. Moreover, a deep learning model, Convolutional Neural Network was used attaining an accuracy of 70%.

In a separate investigation, Hassan et al. (2023) employed Random Forest (RF) ensemble models to anticipate fuel consumption and emission rates in the urban areas of Greater Cairo, Egypt. Assessment using indicators like mean bias error (MBE), root mean squared error (RMSE), and coefficient of determination (R2) showcased the RF models' accuracy and robustness, particularly for CO_2 and CO emissions, with testing R2 values ranging from 0.814 to 0.935. Another study by Zhang et al. (2023) tackled key issues related to CO_2, CO, and HC emissions from gasoline vehicles under various fuel detergency conditions. Conducted in Zhengzhou, China, the study involved testing a passenger car under different fuel detergency conditions, monitoring emissions using a Portable Emissions Measurement System (PEMS), and creating a deep learning prediction model. Results included the identification of emission characteristics with different fuel detergency, assessment of synergistic emission reduction potential, and development of an accurate prediction model. Gasoline detergent significantly impacted emissions, and the deep learning prediction model exhibited high accuracy. Al-Nefaie & Aldhyani (2023) concentrated on predicting CO_2 emissions from vehicle traffic using deep learning models. Utilizing a dataset with official records of carbon emission data from cars with diverse features, the authors proposed a framework employing methodologies like rough k-means clustering, statistical analysis, and deep learning models such as Long- and BiLSTM. The deep learning models, especially the BiLSTM model, displayed robust performance in predicting CO_2 emissions, as evidenced by high R% and R2 values and low MSE and RMSE values in training and testing results.

Table 2.1

Summary of selected studies

Variable	**Predicted**	**Models Used**	**Result**	**Reference**
CO_2 rating of fuel-consuming vehicles	CO_2 rating	NB, RF	NB: 73%, RF: 96%	(Bappon et al. 2022)
Factors impacting fuel usage	Fuel usage	RF, LR, GB	GB with weather data: R-squared	(Bousonville et al. 2019)

			0.84 with weather data	
Vehicle fuel efficiency	Fuel efficiency	PLS, GPR, QR, OLS, SVR	QR highest accuracy	(Yin et al. 2016)
Fuel consumption	Fuel consumption	SVM, RF, DT	VS_MAF-based equation: R-Squared 0.97	(Hamed et al. 2021)
Battery life in electrical vehicles	Battery life	SVM, RF, GBDT	SVM: 97.3% accuracy	(Mahesh Prasanna et al. 2022)
Energy consumption in electric vehicles	Energy consumption	k-NN	Accuracy 96.5%	(Cabani et al. (2021
Heavy-duty vehicle maintenance costs	Maintenance costs	RF	Training: 98.96% R^2, Validation: 94.31% R^2	(Katreddi et al., 2023)
Real-world fuel consumption rate of vehicles	Real-world fuel consumption rate	MLP	MLP model performed well	(Li et al., 2019)
Fuel consumption and CO_2 emissions	Fuel consumption, CO_2 emissions	UPR, MLR, MPL, CNN	UPR: 98.6%, MLR/MPL: ~75%, CNN: 70%	(Hien and Kor 2022)
Fuel consumption and emission rates	Fuel consumption, CO, CO_2 emissions	RF, DL	RF models' accuracy: R2 0.814 to 0.935	(Hassan et al. 2023)
CO_2, CO, and HC emissions	CO_2 emissions	DL models (Long- and BiLSTM)	BiLSTM displayed robust performance	Al-Nefaie and Aldhyani (2023)

As evident in table 2.1, RF model appears to frequently demonstrate high accuracy in different studies in fuel consumption, CO_2 ratings and maintenance costs. On the other hand, SVM, GBDT and GB models show effectiveness in predicting battery life and fuel consumption. The recurrent of these models can be attributed to their strengths. RF's ensemble learning approach, known for building robust decision trees and handling diverse features effectively, provides stability and accurate predictions, making it versatile for different scenarios. Gradient Boosting models, including GBDT on the other hand excel in handling nonlinear relationships and boosting model accuracy with weak learners, proving valuable in complex prediction scenarios. Their successful application across domains, versatility, and robustness in dealing with real-world datasets contribute to their continued adoption.

Limitations of existing studies

From the reviewed papers, it is evident that many studies focused on just predictive modelling and there is a relative scarcity of research that goes into integration of artificial intelligence technologies. These can help make real-time decisions in critical areas such as energy management. Moreover, since the main aim of most of the studies is to bring about fuel efficiency, there are very few studies that show comparison between conventional and electrical vehicles.

Chapter 3

Methodology

Theoretical framework

In the ever-changing landscape of technological evolution, the advent of Industry 4.0 has brought about a transformative era which is characterized by intelligent automation and cutting-edge technologies. This shift, driven by innovations such as the Internet of Things, cloud computing, blockchain, simulation, and artificial intelligence, has revolutionized efficiency, manufacturing processes, and overall productivity (Michael Rüßmann et al., 2015). As we navigate this industrial renaissance, the focus sharpens on the profound implications of Industry 4.0 on fuel-related predictions. The intersection of these technologies not only enhances predictive capabilities but also propels us into a realm where machine learning and deep learning become pivotal in understanding and optimizing fuel consumption and CO_2 emissions. In this section, we delve into the intricacies of machine learning and deep learning, exploring their roles as primary data sources and methodologies in the pursuit of more sustainable and efficient transportation solutions.

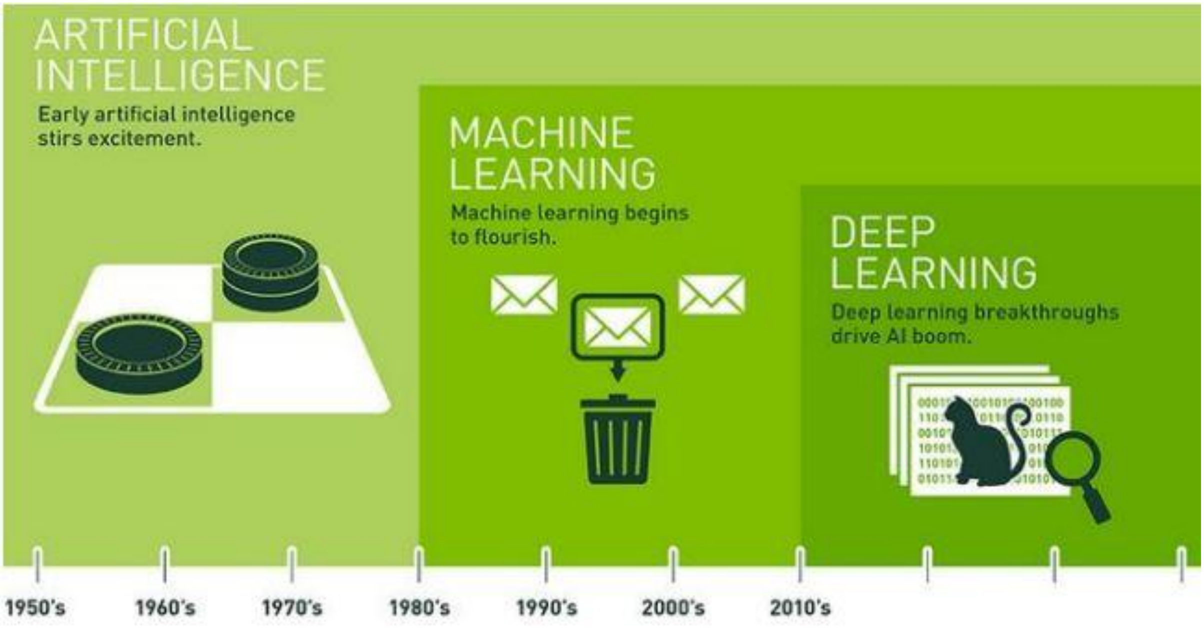
The following sections discuss and contextualize key study themes. Machine learning identified as the primary data source crucial for statistical predictions of the fuel consumption and CO_2 emissions is the main focus. Moving on to the next section the focus shifts to deep learning.

Machine learning

Machine learning is often synonymous with artificial intelligence, attributed to its capacity for learning and decision-making. Machine learning gives the computers the ability to learn without being programmed explicitly as explained by Samuel (2000). Until the 1970s, machine learning was a component of AI evolution where it then split on its own as depicted in figure 3.1. It has evolved into a critical response tool for cloud computing and e-commerce, and it is now being employed in a wide range of technologies.

Figure 3.1

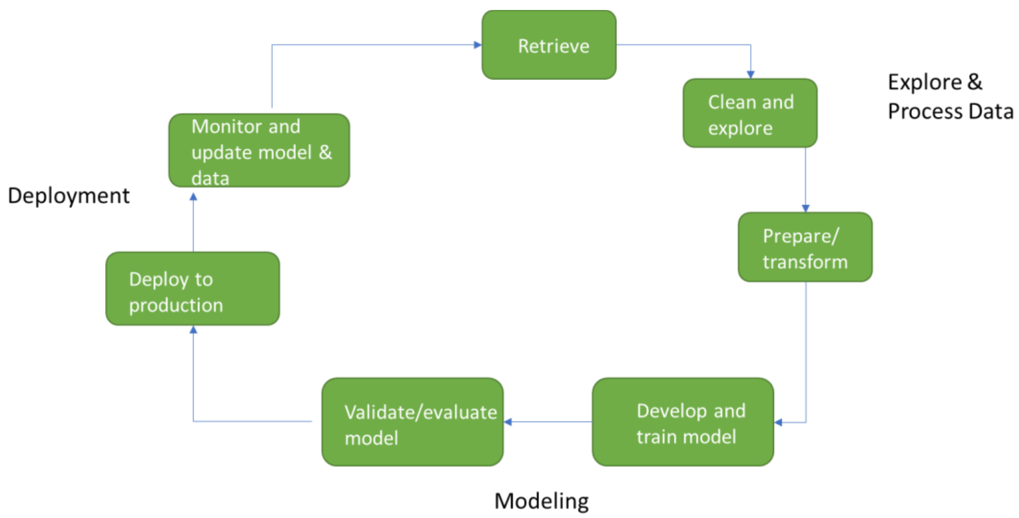
Artificial intelligence and its subsets (Tiwari et al., 2018)



Algorithms may develop prediction models by recognising patterns in data and learning from it. The purpose of machine learning is to make excellent enough predictions to be helpful but not flawless forecasts (Medar et al., 2017). However, the performance of the machine learning techniques is determined by the data's qualities and complexity (Sarker, 2021). Figure 3.2 shows a typical machine learning workflow.

Figure 3.2

A typical workflow in machine learning

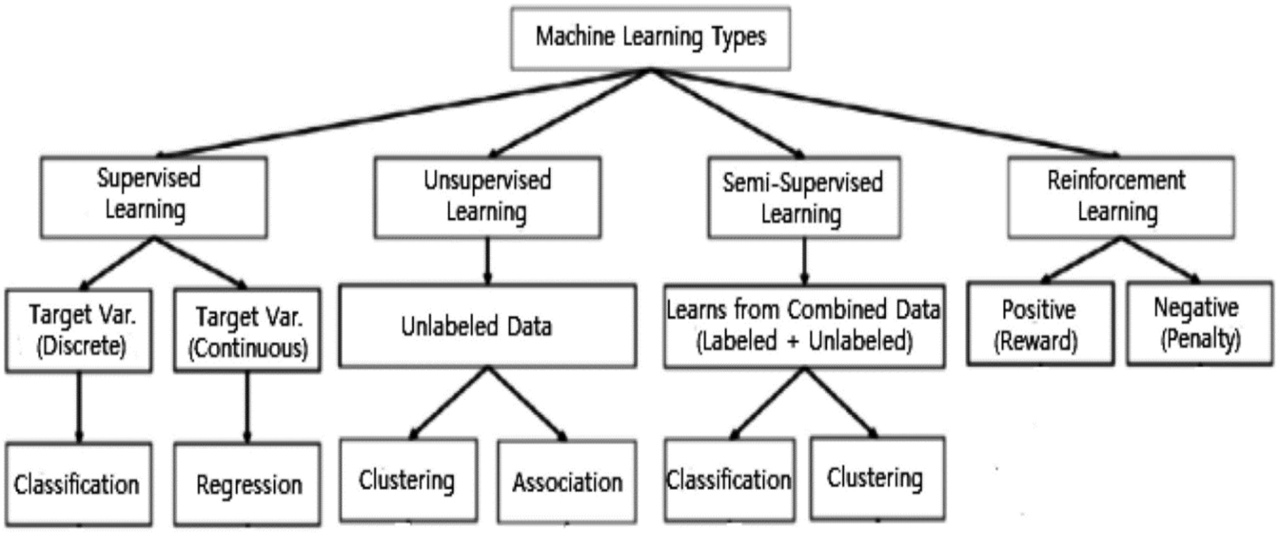


Data is retrieved, cleaned and pre-processed in the initial phase of the workflow. The processed data is then trained using a developed model where the model learns the patterns and relationships within the data to makes predictions. The model's performance is then evaluated using a testing data set. Once one is satisfied with the performance, the model is deployed to make predictions on new unseen data. Deployment can involve integrating the model into a web application, server, or any other system. There is continuous monitoring of the model's performance in real-world scenarios. It is retrained periodically with updated data to ensure it remains effective as patterns in the data evolve.

Machine learning is made up other subsets namely, supervised, unsupervised, semi supervised and reinforcement learning which are determined by the type of data at hand (Chitralekha & Roogi, 2021). Data is the main key to machine learning whereby its quantity dictates how many times it should be trained moreover; the type of algorithm dictates the amount of training the system should undergo. Figure 3.3 gives a visual representation of the four categories discussed in the sections that follow.

Figure 3.3

Machine learning techniques (Sarker, 2021)



Techniques in machine learning

1. Supervised leaning: The purpose of this technique is for the algorithm to "learn" by comparing its actual output to the "taught" outputs offered, identifying flaws, and updating the model as needed (Q. Liu & Wu, 2012). In supervised learning, patterns are utilised to predict label values for further unlabelled data. Throughout the learning process, complex mathematical approaches are used to optimize this function. In figure 3.3 discrete variable refers to the target variable when it has discrete values, meaning it can only take specific, predefined categories. The continuous variable on the other hand,

means that the values in the target variable are continuous and any value within a range can be taken.

2. Unsupervised learning: It uncovers patterns in unlabelled data without extensive human guidance, contrasting with supervised learning that relies on labelled data. It explores probability densities over inputs, learning from input data to identify patterns and create new features (Saman Siadati, 2020). This approach finds applications in clustering, dimensionality reduction, feature extraction, and anomaly detection. Without predefined answers, unsupervised learning excels in organizing complex and seemingly unrelated data, making it suitable for anomaly detection and recommender systems. Its strength lies in exploring diverse datasets, revealing valuable insights that structured approaches may miss.
3. Semi-supervised learning is a hybrid strategy that incorporates characteristics of both supervised and unsupervised approaches. In contrast to closely supervised learning, it functions with less strict control, decreasing its load. The dataset is purposefully built with the bulk of unlabelled data and a smaller collection of labelled data. This arrangement is useful in situations where labelled data is scarce.
4. Reinforcement learning: It operates primarily through a trial-and-error approach(Barto, 1994), where algorithms eschew labelled data in favour of positive and negative feedback to reinforce their learning. This environment-driven method relies on a system of rewards and penalties, akin to insights from environmental activists seeking to enhance rewards and minimize risks. Reinforcement learning is employed to train models aimed at increasing automation or optimizing performance, particularly in domains such as autonomous driving, robotics, manufacturing, and supply chain management.

Advantages of machine learning

1. Quick adaptability to fluctuations in data.
2. Prioritization of the decision-making process leading to swift decisions.
3. Immediate analysis of data patterns, allowing for prompt action.

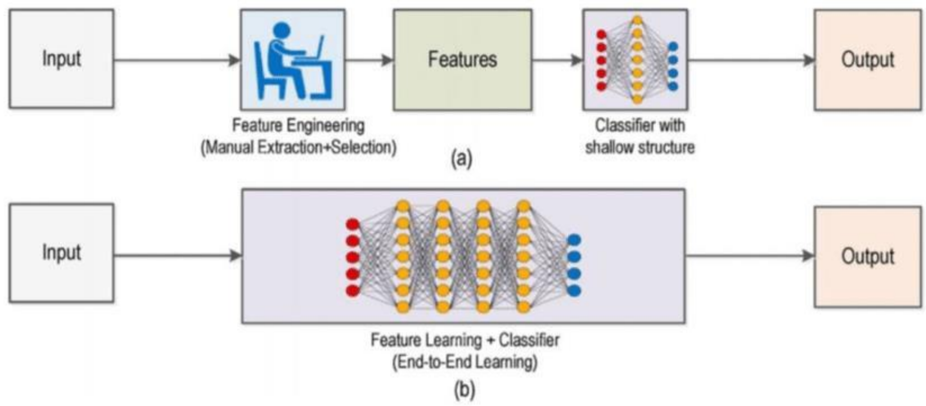
Since its commencement, machine learning has evolved to include a diverse array of approaches, algorithms, and applications, with ongoing room for enhancement. The latest advancements in computer technology have facilitated parallel data processing, elevating the relevance of machine learning methodologies. The introduction of deep learning has particularly emerged as a consequence of advancements in memory capabilities.

Deep learning

Deep learning constitutes a machine learning methodology wherein inputs traverse multiple layers interconnected by neuron-like structures, enabling the system to learn from data. Due to its flexible architecture, neural networks can learn from raw data. Neural networks possess the ability to learn directly from raw data with minimal pre-processing, and their prediction accuracy is heightened by their flexible architectures (Prabhudesai et al., 2019). The number of hidden layers that build connections for increased learning determines the depth of machine learning. The best results are obtained by minimizing loss through weight and bias adjustments. While the conceptual foundations of artificial neurons date back to 1943, as envisioned by Walter Pitts and Warren McCulloch (Piccinini, 2004), it's crucial to note that the modern manifestation of deep learning has gained prominence more recently. The surge in computational power, especially in recent years, has been a catalyst, enabling the scalability and complexity required for deep learning networks. Widespread applications of deep learning in industries such as health (Al Turjman et al., 2024; Ibrahim et al., 2021) have become increasingly prevalent with advancements in technology. This is what shapes the landscape of artificial intelligence in contemporary time. Deep Learning specializes in managing complex mappings from input to output, relying on extensive datasets and high computational capabilities.

Figure 3.4

Machine and deep learning(Del Real et al., 2020)



In figure 3.4, we can see how different machine learning is to deep learning. For instance, machine learning uses simple algorithms like regression which may require human intervention for feature engineering. On the other hand, deep learning employs neural networks with multiple layers enabling it to learn directly from the data. These networks can automatically extract features, eliminating the need for manual feature engineering.

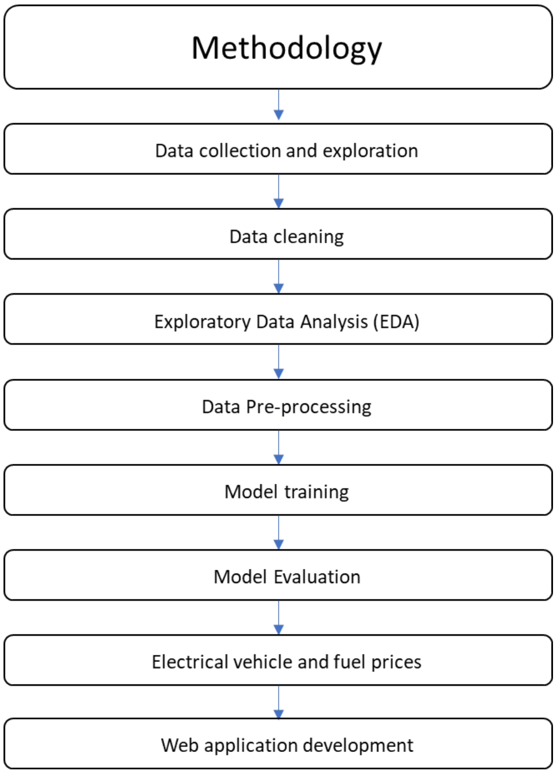
This study employs regression analysis through machine learning methodologies to give predictions on the fuel consumed and carbon emissions of different ICE vehicles. The reason behind employing these models stems from the complex relationships between the variables influencing fuel consumption and CO_2 emissions. The main advantages of the regression analysis are (Saleh et al., 2022):

1. They allow for the prediction of outcomes based on relationship between variables. This in turn provides valuable insight for decision-making and planning.
2. There is clear understanding of relationships between the dependent and independent variables which in turn allow interpretation of the impact of each predictor variable on the outcome.
3. They are simple and easy to interpret.
4. Regression models are compatible to a wide range of problems.

The model input comprises a pre-processed data set linked to an individual car, encompassing 8058 rows of the actual vehicle specifications. For the EV, the specifications of one car namely, B9 from Gunsel company in the Turkish Republic of Northern Cyprus were used later on. This particular car was chosen as a representative for most EVs and because it will give the engineers at Gunsel a real time comparison between their vehicle and other conventional vehicles. The methodology of this project includes data pre-processing, Exploratory Data Analysis (EDA) and machine learning techniques implementations as shown in figure 3.5. Data is first acquired, processed, split and then trained using seven machine learning models. The regression models were first used for fuel consumption predictions and later on the same models were used to predict the emissions. Since the data set has input-output pairs with the correct output known, all the models used fall under supervised machine learning technique and were used to make predictions for unseen data.

Figure 3.5

Methodology flowchart



Data collection and exploration

The dataset used for this study is sourced from an open source (Government of Canada, 2023), containing comprehensive information on vehicle characteristics and emissions. The dataset covers the years 2016 to 2023 and serves as the foundation for model training and evaluation. I uploaded the data to Google Drive and used Google Colab to execute the Python code. Colab, or Google Colaboratory, is a cloud-based service based on the Jupyter Notebook design. Google Colaboratory provides researchers with free access to GPUs for deep learning applications, allowing them to execute high-end machine learning concepts (J. & V., 2021). After importing the dataset, an initial investigation is performed to determine its structure. Table 3.1 shows what each column in the data represents.

Table 3.1

Data key

Vehicle class	It represents the category system used in the automotive industry grouping vehicles based on size, body style and use
Model	Refers to a distinct variation of a vehicle created by a manufacturer under a specific make or brand.
Make	Denotes the brand or manufacturer of a vehicle.

Model Year	Indicates the production year of a specific vehicle model
Engine size (L)	Typically measured in cubic centimeters (cc) or liters (L), it tells the total volume of air and fuel that can be drawn into the engine cylinders of a vehicle in one cycle.
Cylinders	Refers to the number of chambers in vehicles' engine. It is where combustion takes place.
Transition	This system facilitates power transfer ideally from the engine to the wheels, enabling the vehicle to make either forward or backward direction movements. Different designations are used to denote various transmission types: A stands for automatic, AM for automated manual, AS for automatic with select shift, AV for continuously variable, and M for manual.
Fuel type	It provides information about the vehicle's fuel type. Z stands for premium petrol, D stands for diesel, E stands for ethanol and X stands for normal petrol.
Fuel Consumption	Gives the average fuel consumed in highway and city roads
CO_2 Emissions(g/km)	It refers to the amount of CO_2 that is emitted by a vehicle for every kilometer that it is driven.
CO_2 Rating	It measures a vehicle's carbon dioxide (CO_2) emissions, typically expressed in grams of CO_2 per kilometer(g/km)

There are different vehicle classes ranging from compact to full size vehicles in the dataset. The different models in the dataset have different engine sizes which range from 1 to 10 where the number of cylinders have the lowest value at 1 and highest at 16. The CO_2 rating values are from 1 to 10. If the rating is high and is getting closer to 10, it means that the vehicle has lower CO_2 emissions and lower fuel consumption. On the other hand, a lower rating closer to 1 suggests higher CO_2 emissions and a potentially greater environmental impact.

Data Cleaning

Since we are predicting both fuel consumption and CO_2, the data went through two phases. For the first phase, a new data frame is generated with a focus on pertinent columns essential for the regression model for predicting fuel consumption. Columns such as model year, make, and model are excluded, emphasizing key attributes as illustrated in table 3.2. The columns were dropped as they are not significant for the model training. The model is going to be deployed and used for all kind of conventional cars. In the second phase, all the columns were dropped leaving only two, fuel consumption and CO_2 emissions as we are going to see in the following sections.

Table 3.2

Data outlook after dropping unwanted columns

	Vehicle Class	Engine Size (L)	Cylinders	Transmission	Fuel Type	Fuel Consumption Comb (L/100km)	CO2 Rating
0	Compact	2.4	4	AM8	Z	8.1	7
1	SUV: Small	3.5	6	AS9	Z	11.1	5
2	SUV: Small	3.5	6	AS9	Z	10.7	5
3	SUV: Small	3.5	6	AS6	Z	10.7	5
4	Mid-size	3.5	6	AS6	Z	10.0	5
...	...	...	...	...	...	...	...
8053	SUV: Small	2.0	4	AS8	Z	9.2	5
8054	SUV: Small	2.0	4	AS8	Z	9.4	5
8055	SUV: Small	2.0	4	AS8	Z	10.0	5
8056	SUV: Standard	2.0	4	AS8	Z	9.6	5
8057	SUV: Standard	2.0	4	AS8	Z	10.6	5

8058 rows × 7 columns

It is very important to also check for null values in the data. These values can affect the statistical measures such as the mean, median and the standard deviation. That being said, if we understand the presence of missing values, we can then choose the appropriate way to handle them so as they do not affect our analysis. Some of the columns were also renamed to make them more concise for ease of use.

Concerning CO_2 ratings, new ratings are assigned based on fuel consumption values in instances where the original ratings are absent. To improve the model's interpretability, this transformation is implemented. The new values of the CO_2 ratings are assigned based on different ranges of fuel consumption where in this case a higher rating means the consumption is low leading to lower emissions. This approach allows the model to capture the impact of fuel efficiency on emissions more explicitly. Standardization of numeric values within the transmission column is also performed, wherein detailed values like 'AM8' are replaced with simplified representations such as 'AM.' This was done so as to enable the models to understand and generalize the data.

Exploratory Data Analysis (EDA)

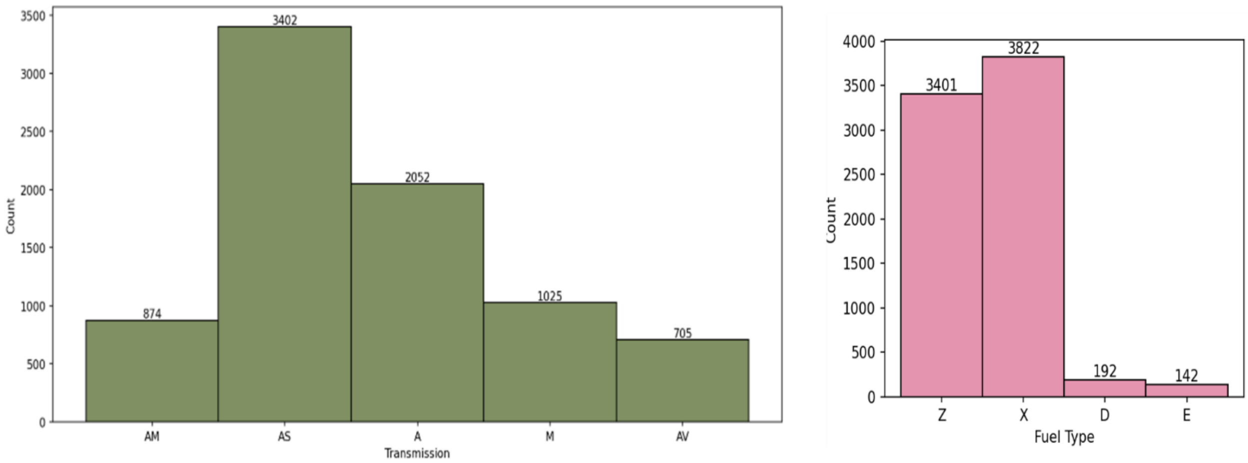
It is very important to understand the underlying patterns and the characteristics in the data. This is done by using EDA, which gives a valuable insight into the dataset's structure(Morgenthaler, 2009). This project applied both univariate and bivariate analysis.

Univariate Analysis

Univariate analysis, a statistical methodology focused on the examination of a single variable within a dataset, plays a pivotal role in scrutinizing and elucidating the distribution, characteristics, and patterns inherent in that specific variable. Its significance lies in its ability to provide a comprehensive understanding of the variable's distribution, including the identification of range, central tendency, and variability measures(Cleff, 2014). Univariate analysis aids in outlier detection, crucial for maintaining the integrity of statistical properties. That being said, histograms were used to show variables' distribution. Figures 3.6 to 3.10 show the distribution of all the variables.

Figure 3.6

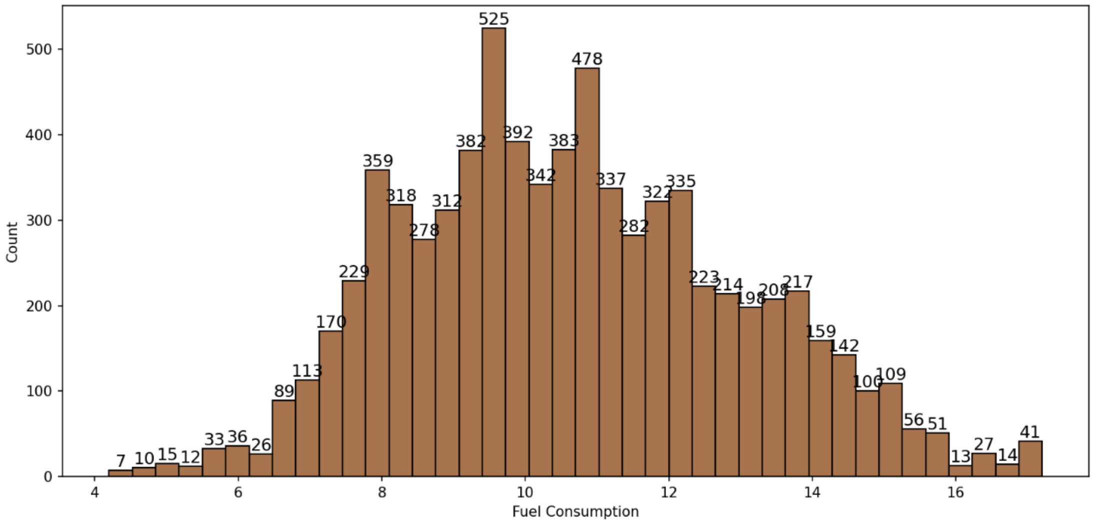
Transmission and Fuel type distribution



In figure 3.6 above, it is evident that there are many vehicles in the dataset which use petrol fuel whether normal 'X' or premium 'Z'. Ethanol 'E' and diesel 'D' are the least used types of fuel. This might be because of the widespread availability of petrol and it being affordable. Moreover, petrol cars are generally cheaper compared to other cars. Most of the cars also use semi-automatic mode of transmission followed by automatic then manual cars.

Figure 3.7

Fuel consumption distribution



Most vehicles consume an average of ten litres for every 100 km as depicted in figure 3.7. Very few vehicles consume four to seven litres and fifteen to seventeen litres of fuel for every 100 km. This could be influenced mainly by the preferences of the consumers to use efficient vehicles driving manufacturers to produce vehicles with a consumption of rate of between eight to fourteen litres per 100 km. The next figures 3.8 to 3.10 illustrate engine size, number of cylinders and the CO_2 rating distributions.

Figure 3.8

Engine size contribution

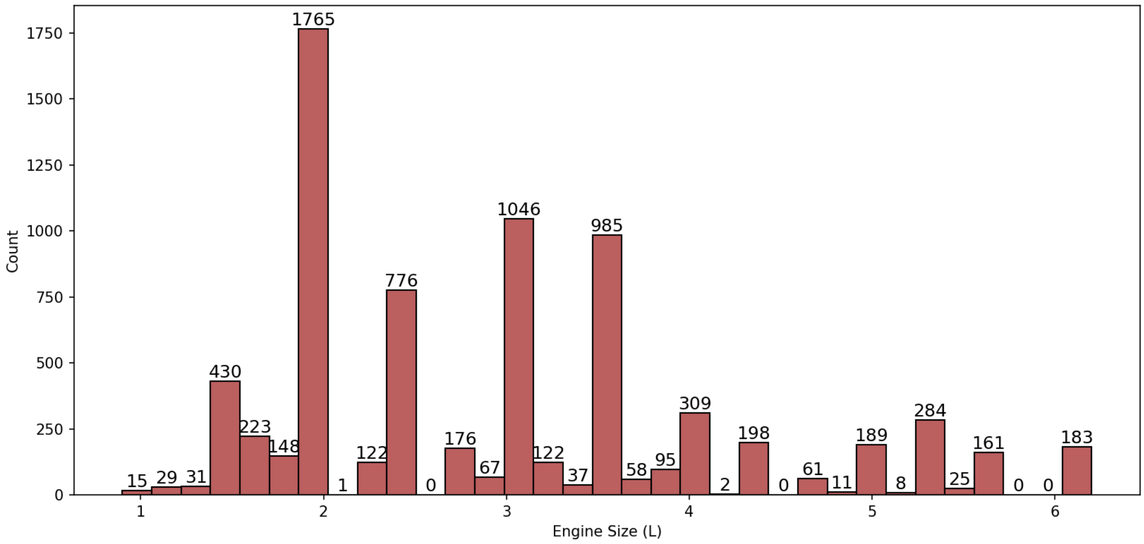


Figure 3.9

Number of cylinders distribution

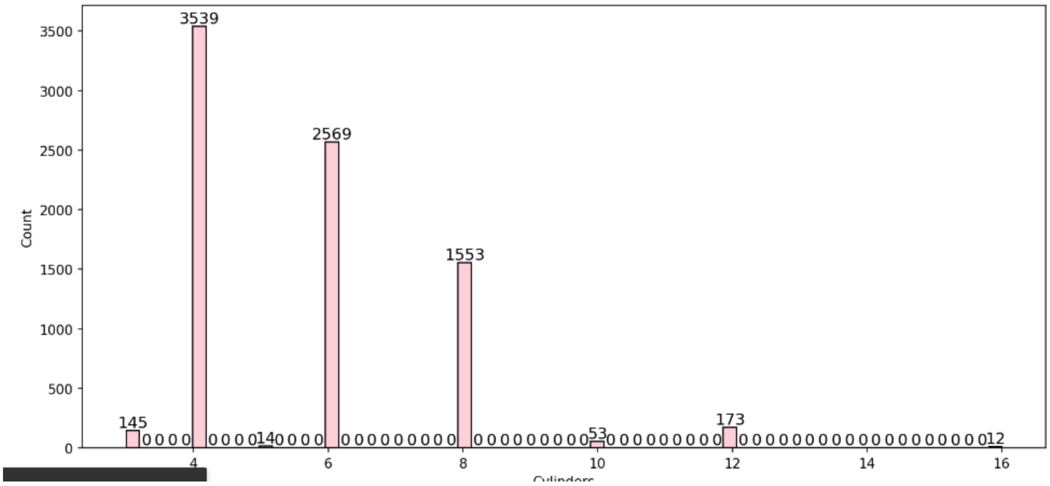
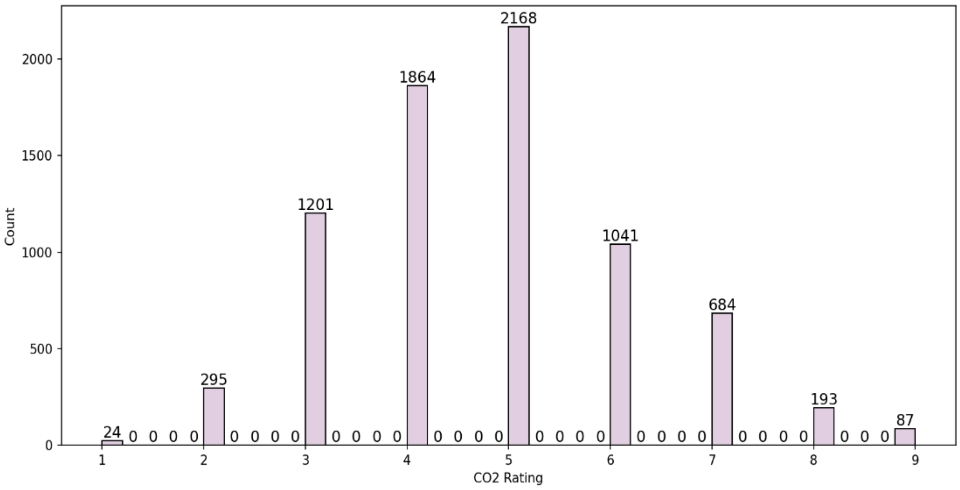


Figure 3.10

CO_2 rating distribution



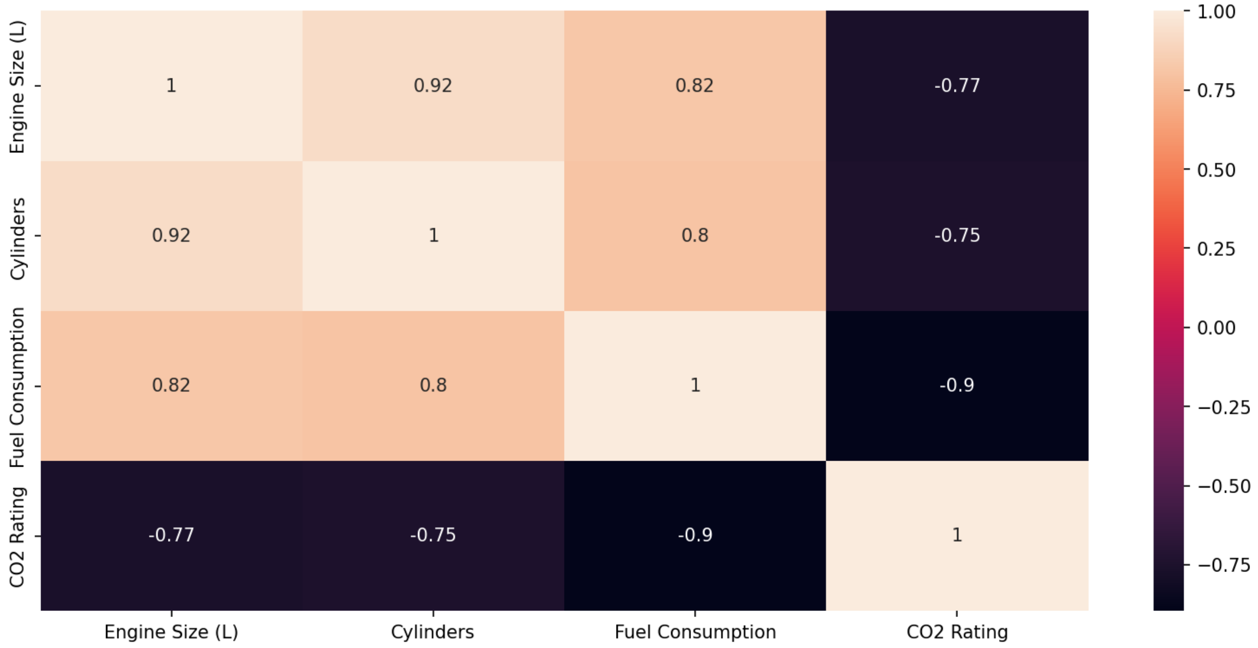
In figure 3.8 it is evident that most vehicles have 2-liter engine sizes with some ranging from 3 to 4. Also, most of the vehicles have four-cylinder engine configuration in the dataset as shown in figure 3.9. This implies that most of the cars are designed with a balance of power and fuel efficiency. On the other hand, the CO_2 rating is at an average of 5 meaning the carbon emissions from most vehicles is neither good nor bad.

Bivariate Analysis

Bivariate analysis involves the examination and analysis of relationships between two variables(Bertani et al., 2018). For this project, a heatmap was generated to visualize the Pearson correlation between various attributes, aiding in identifying potential multicollinearity. The value of correlation ranges between -1 and 1 where the value closest to either of them shows that here is high correlation. Negative values show negative correlation meaning if x increases, y decreases and a positive value shows positive correlation meaning an increase in x causes an increase in y.

Figure 3.11

Pearson correlation heatmap



As shown in figure 3.11, the number of cylinders, CO_2 rating and the engine size have very high correlation with the fuel consumption. The CO_2 rating has a correlation value of -0.9 meaning it negatively affects the fuel consumption. If its value increases, then the value of fuel consumption decreases an vice versa. The number of cylinders and engine size have a positive correlation value of 0.8 and 0.82 respectively, an increase in one leads to an increase in the other variable.

Data Pre-processing

Ordinal encoding and One-Hot encoding

The dataset, as shown in Table 3.2, has three categorical variables: vehicle class, transmission, and fuel type. Machine learning techniques cannot process these categories as they are not numeric. Therefore, two encoding techniques, ordinal and one-hot encoding, were used to convert the data into numerical form. Ordinal encoding assigns each value an integer from the categories with a natural ranking between them. For the dataset in this project, ordinal encoding is applied to transform categorical variables like transmission and vehicle class into numeric representations, maintaining their ordinal relationships.

On the other hand, one hot encoding uses absolute values in a binary matrix where each category will be represented as a column. The fuel type column was chosen for this, creating binary columns for each fuel type, as shown in Table 3.3. Each categorical variable was changed into a numerical value, and the fuel types for each variable were binary digits.

Table 3.3

Dataset after ordinal and One-hot encoding

	Vehicle Class	Engine Size (L)	Cylinders	Transmission	Fuel Type	Fuel Consumption	CO2 Rating	Transmission_X	Vehicle Class_X	D	E	X	Z
0	Compact	2.4	4	AM	Z	8.1	7	1.0	2.0	0	0	0	1
1	SUV: Small	3.5	6	AS	Z	11.1	5	3.0	6.0	0	0	0	1
2	SUV: Small	3.5	6	AS	Z	10.7	5	3.0	6.0	0	0	0	1
3	SUV: Small	3.5	6	AS	Z	10.7	5	3.0	6.0	0	0	0	1
4	Mid-size	3.5	6	AS	Z	10.0	5	3.0	4.0	0	0	0	1

Splitting the data

The x and y features were chosen where the y feature here is the fuel consumption for the first part and the x features are as shown in table 3.4. For predicting the CO_2 emissions, the x feature was chosen as the fuel consumption. The training and testing sets were set to 90% and 10 % respectively for both predictions. This was done so as to enable the models to have a high performance. The more the training dataset is, the better the performance.

Table 3.4 Input features for predicting fuel consumption

	Engine Size (L)	Cylinders	CO2 Rating	Transmission_X	Vehicle Class_X	D	E	X	Z
0	2.4	4	7	1.0	2.0	0	0	0	1
1	3.5	6	5	3.0	6.0	0	0	0	1
2	3.5	6	5	3.0	6.0	0	0	0	1
3	3.5	6	5	3.0	6.0	0	0	0	1
4	3.5	6	5	3.0	4.0	0	0	0	1

The original vehicle class, transmission and fuel type columns were dropped as shown in table 3.4. The remaining columns contained the encoded columns, engine size, cylinders and the CO_2 columns. It is also important scale the data to avoid bias outcome of predictions. Standardization is therefore applied to scale numerical features, ensuring that all variables contribute equally to the model and preventing dominance by variables with larger magnitudes.

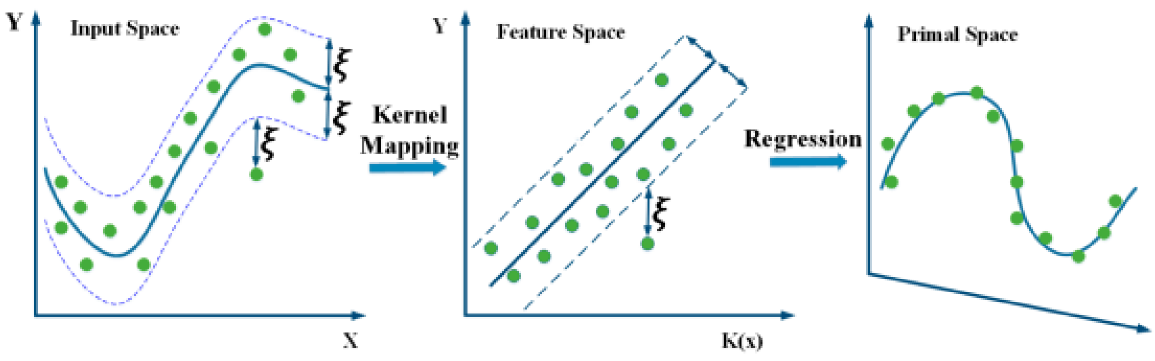
Model training

Support Vector Regression (SVR)

The model is a supervised learning machine used for pattern recognition and regression. It was developed from the support vector machine (SVM). It constructs models complex enough to handle a large class of problems, yet simple enough to be analysed mathematically (Busuttil, n.d.). The model is based on the concept of hyperplane classifiers and uses feature spaces and kernels to create a nonlinear decision surface in the input space as depicted in figure 3.12. It is useful since it simplifies to an optimization issue with a unique solution and has been used effectively to a variety of real-world challenges. The decision function in the SVR model is a key component used for classifying input vectors. In this research, the SVR model employed a linear kernel function to simulate linear correlations between input variables and the target variable. This is due to the fact that both the input and output characteristics are linear.

Figure 3.12

Support Vector Regression principle (Moradzadeh et al., 2020)

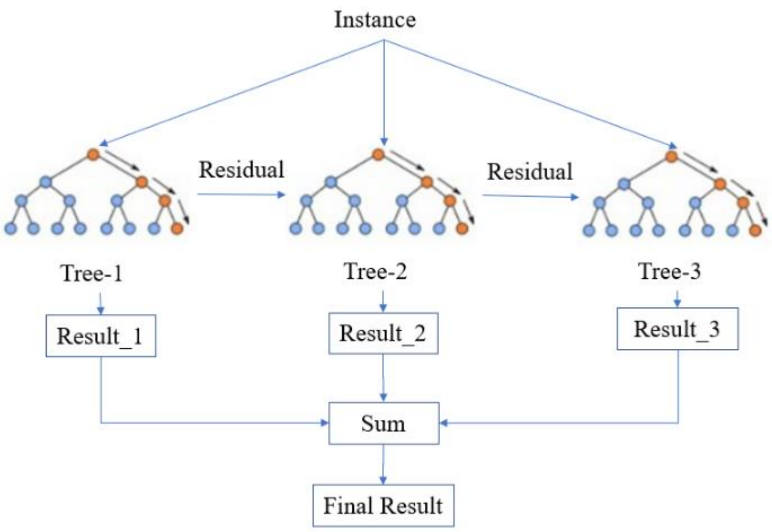


Extreme Gradient Boosting Model (XGB)

XGB is a powerful and scalable ensemble machine learning algorithm based on decision trees and gradient boosting as shown in figure 3.13. Introduced by Tianqi Chen and Carlos Guestrin (2016), it has achieved success in Kaggle tournaments and is widely used for diverse data science problems. With features like cross-validation, distributed weighted quantile sketch, and parallelization, XGB excels in modelling attributes, classification, prediction, and system optimization. Its efficiency and accuracy make it a popular choice in various applications. The XGB model used in this project is a specialized class within the XGB library designed specifically for regression. The objective parameter was set to “reg: squared error” meaning the model is being used for regression, and the objective function is the mean squared error (MSE). The random state parameter was set to 51 ensuring that the random processes (like random initialization of weights) in the model are reproducible.

Figure 3.13

XGB model principle (Wang et al., 2021)



Passive Aggressive Regressor (PAR)

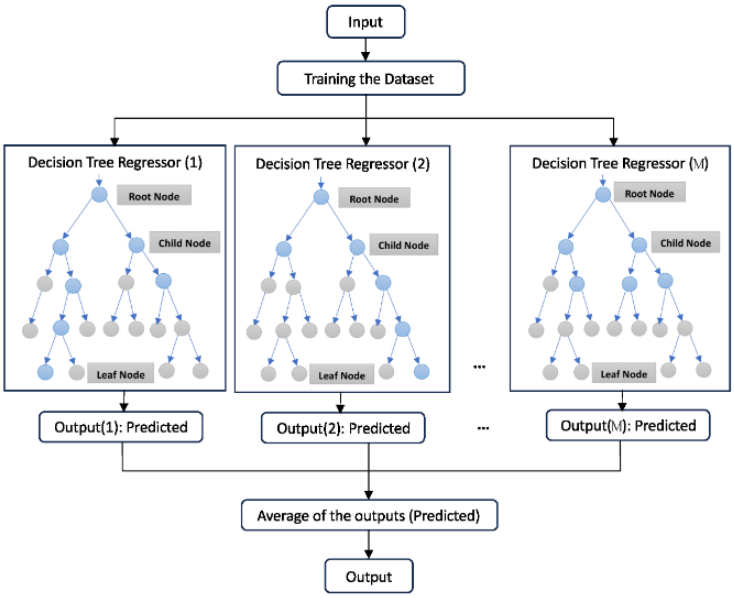
Another model that was used is the PAR which falls under online learning in machine learning(Crammer et al., 2006; Herbster, 2001). The passive component is where minimal updates are made to the model when the predictions are correct while the aggressive component makes updates when they are incorrect predictions. The passive-aggressive behaviour allows for quick adaptability to changes in the data distribution. The model is created with hyperparameter c set to 1 in order to control the regularization strength and the random state parameter is set to 51.

Extra Tree Regressor (ETR)

The ETR model, originates from the Random Forest (RF) model (Geurts et al., 2006). It works in an advanced approach that constructs an ensemble of unpruned decision or regression trees as depicted in figure 3.14. Employing two critical parameters k and n_{min} the ETR algorithm governs the splitting process to enhance precision while concurrently mitigating overfitting (Mishra et al., 2017). This model strategically utilizes all cutting points, selecting randomly from these points to divide nodes, and employs the entire learning samples for tree cultivation. By doing so, it minimizes bias and significantly amplifies the model's precision. The ETR model serves the purpose of predicting a target variable based on input features, consistently achieving remarkable accuracy in its predictions. The ensemble configuration for this project includes 10 decision trees, with a designated random state set at 51.

Figure 3.14

Regression trees in ETR model (Aziz et al., 2023)

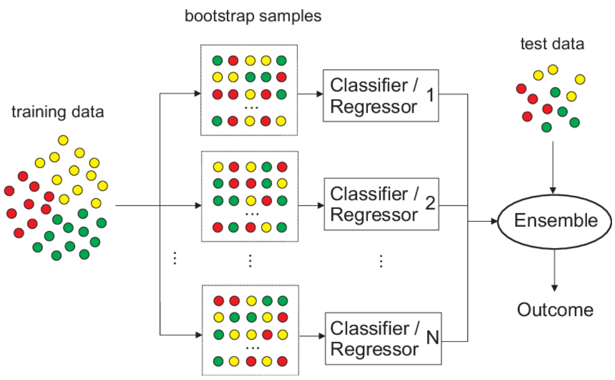


Bagging Regressor model (BR)

It is also an ensemble model categorized within the family of bagging algorithms. Bagging, short for bootstrap aggregating, involves training numerous instances of a base model on distinct subsets of the training data and combining their predictions to formulate a final prediction as is evidence in figure 3.15. This approach, enhancing performance, is recognized as one of the models employed in this project for predicting fuel consumption and CO_2 emissions.

Figure 3.15

Working principle of bagging regression model (Salcedo-Sanz et al., 2022)



Tweedie Regressor

Tweedie Regressor (TR) (Kokonendji et al., 2021) is a regression model designed for predicting non-negative target variables and can handle semi-continuous data with excess zero. It is a compound distribution that encompasses both the Poisson and gamma distributions. The TR model used in this study was set at a power parameter value of 1 which corresponds to the Poisson distribution. Strength in the model is controlled by the alpha parameter which was added to the loss function to prevent overfitting and set to 0.5 implying a moderate level of regularization. Moreover, link parameter used in this model is log which represents a logarithmic link. The choice of the log-link is suitable for predicting positive and right-skewed data, as it allows the linear predictor to be related to the log of the expected value of the response variable.

k-Neighbours Regressor

Another model that was used is the k-Neighbors Regressor. This is another type of regression algorithm which belongs to the family of k-Nearest Neighbors (k-NN) methods. KNN is a non-parametric, supervised learning classifier that employs proximity to classify or predict the grouping of a single data point(IBM, 2022). It is used for predicting a continuous target variable based on the values of its k-nearest neighbors in the feature space.

All the models used in this project were used to train the data in predicting the fuel consumption and carbon emissions. The parameter settings in the models were the same for training the x features fuel consumption and CO_2 emissions.

Model Evaluation

All the models were quantitively assessed using three metrics. The most commonly used metrics were made use of including the R2 score, the Mean Absolute Error (MAE) and Mean Squared Error (MSE).

Mean Squared Error (MSE)

As mentioned earlier, it is one of the common metrics used for evaluation of regression models to see how they performed. A lower MSE value shows that the model performed well because the squared differences between predicted and actual values are smaller. The MSE calculates the average squared difference between the target variable's expected and true values. It is as shown in equation 3.1

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2 \tag{3.1}$$

The number of observations is represented by n, the real value of the target variable is y_i, and the predicted value of the target variable is $\widehat{y_i}$. It is a positive number, and a lower MSE indicates that the model worked well.

R-squared (R2)

The R2 score, also referred to as the coefficient of determination, gauges the proportion of variance in the dependent variable that the independent variables in the regression model can account for. This metric is pivotal for assessing the model's fitting quality. The R2 score is calculated through the following formula as depicted in equation 3.2

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y_i})^2}{\sum_{i=1}^{n}(y_i - \overline{y_i})^2} \tag{3.2}$$

Here, n denotes the number of observations, y_i signifies the actual value of the dependent variable, $\widehat{y_i}$ represents the predicted value of the dependent variable, and $\overline{y_i}$ denotes the mean of the true values of the dependent variable. R2 values range from 0 to 1. A score of 1 indicates a perfect fit, explaining all variability in the dependent variable. Conversely, a score of 0 signifies that the model fails to explain variability.

Mean Absolute Error (MAE)

This type of metric measures the average absolute difference between the predicted values and the true values of the target variable. A smaller MAE value is more desirable in evaluating the models. This indicates that the predicted values are closer to the actual values. Equation 3.3 shows how MAE is calculated

3.3

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}|$$

Here, n is the number of observations, y_i depicts the true value of the target variable for the observation and $\hat{y_i}$ depicts the predicted value for the observation.

Electrical vehicle and fuel prices

As mentioned earlier, one electric vehicle was chosen to compare fuel consumption and energy consumed for a journey of the same distance. The Gunsel company manufactures the car named the B9. The main feature of the vehicle used in this project is the energy consumption, which is 14 kilowatt-hour/100km (kWh/100km). The car is a four-wheel drive two-seater powered by a 140-kW lithium-ion battery. The fuel and charging prices were accessed online, and these prices specifically for that particular day.. They were set to Turkish Lira (Tl) based on the geographic area in which this research was done: the Turkish Republic of Northern Cyprus (TRNC).

Web application development

A Python Application Programming Interface (API) was created in the backend. This is because the model was developed using python language. The application is first run locally and its backend is implemented using Flask, a micro web framework for Python. Flask is used to spin up a webserver and here it handles parameter inputs, passes them to the deployed model, and returns the predicted fuel consumption and CO_2 emissions as JavaScript Object Notation (JSON) object. Upon confirmation that the web app runs, Waitress module was used to host the web app online. The module is a production quality pure -Python Web Server Gateway Interface (WSGI) with no dependencies(Chris & Waitress, 2023). This means that it does not rely on external libraries other than Python libraries, simplifying the installation and deployment process.

Moreover, the web app is integrated with Mapbox to provide distance and navigation information. Mapbox is a platform that provides various tools and services for creating custom maps and location-based applications. It offers mapping and location-based services, including APIs (Application Programming Interfaces), SDKs (Software Development Kits), and a cloud-based platform for designing and hosting custom maps(Amy Lee Walton, n.d.). This enhances the accuracy of predictions, considering that fuel efficiency can be influenced by factors such as driving distance. The fuel consumption was then calculated based on the distance the vehicle might travel and this also goes for the energy consumption of the EV. The formula for the fuel consumed and energy consumed are given by:

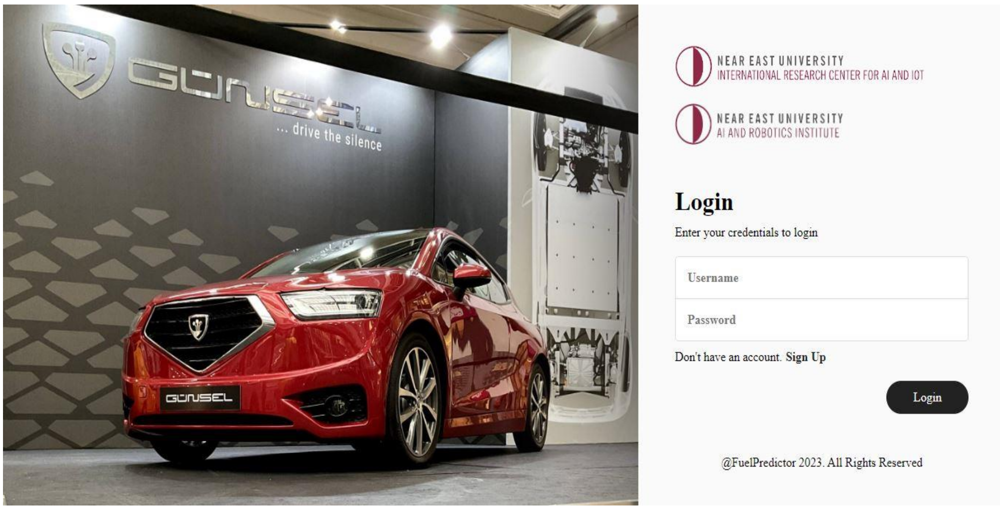
$$\text{Fuel consumption (litres)} = \frac{\text{Fuel consumption rate (litres per 100km)} * \text{Distance travelled (km)}}{100} \tag{3.4}$$

$$\text{Energy consumption (Wh)} = \frac{\text{Energy consumption rate (Wh per 100km)} * \text{Distance travelled (km)}}{100} \tag{3.5}$$

The development of the web application's frontend utilized Visual Studio Code, a robust and lightweight source code editor designed for desktop use. It comes equipped with built-in support for various languages, including JavaScript and Python. The frontend of the web application is created using Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript (JS). The backend for the web user interface was created using Hypertext Pre-processor (PHP).

Figure 3.16

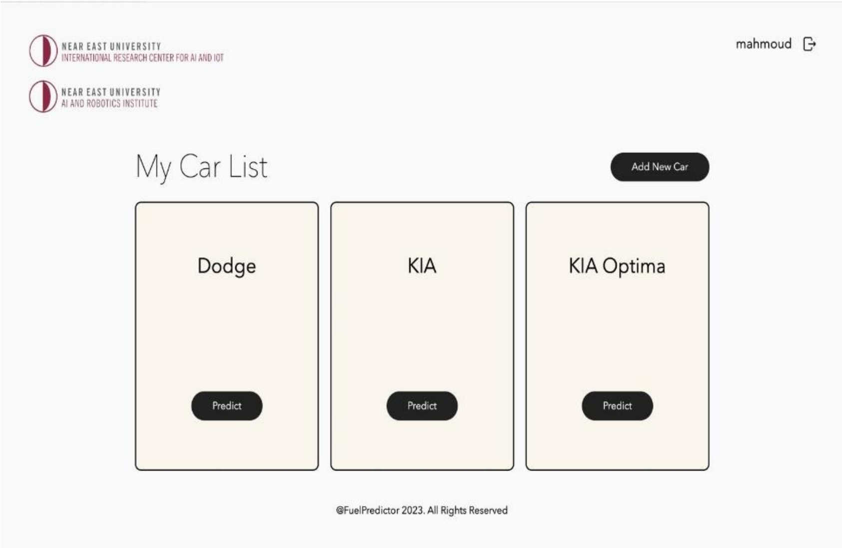
Web app login page



The login page in figure 3.16 gives the user an opportunity to either create an account or login to an existing account. After logging in the user is presented with the dashboard with a list of added cars or an option to add a new car.

Figure 3.17

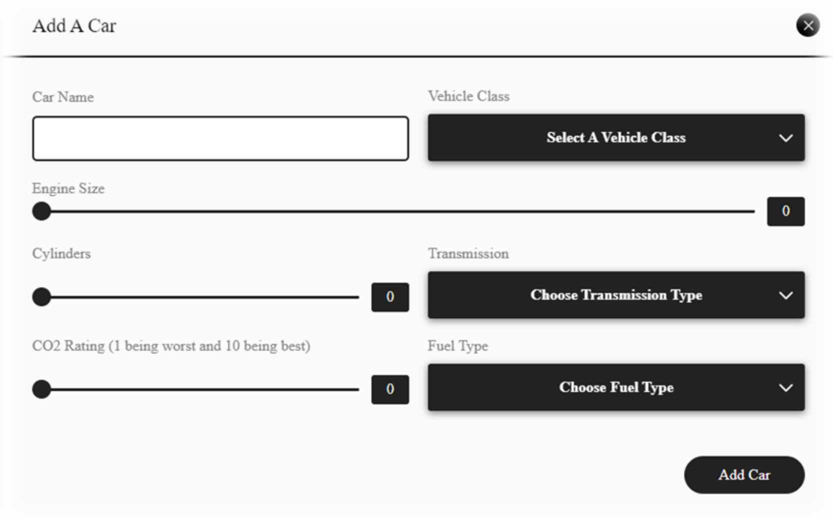
Dashboard view



In figure 3.17, the user has an option to either add a new car or use the previously added cars for prediction.

Figure 3.18

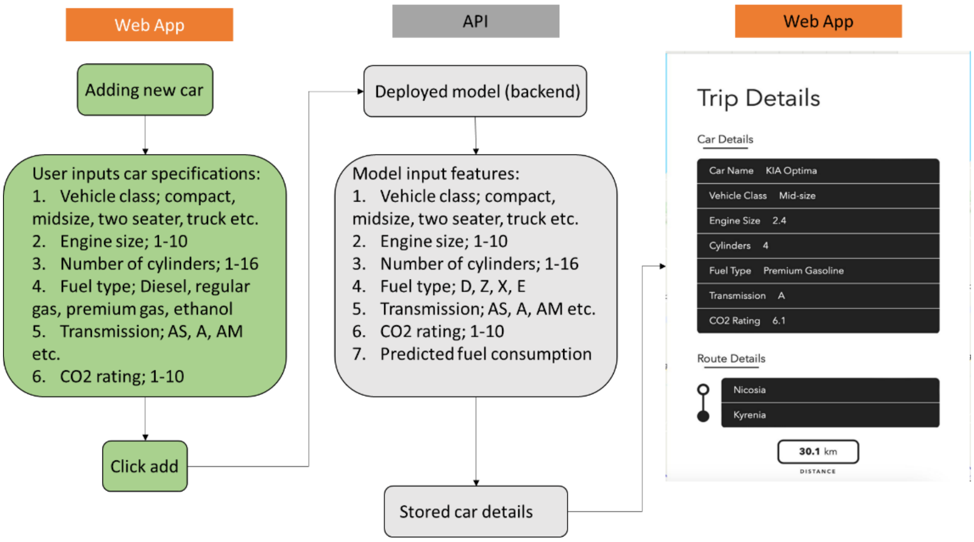
Adding car details



On clicking add new car, the user adds the car name and its details and clicks on add car as shown in figure 3.18. The user inputs, vehicle class, engine size, number of cylinders, fuel type, transmission and the CO_2 rating are sent to the backend where they are read as input features for the trained model as shown in figure 3.19.

Figure 3.19

User parameter inputs

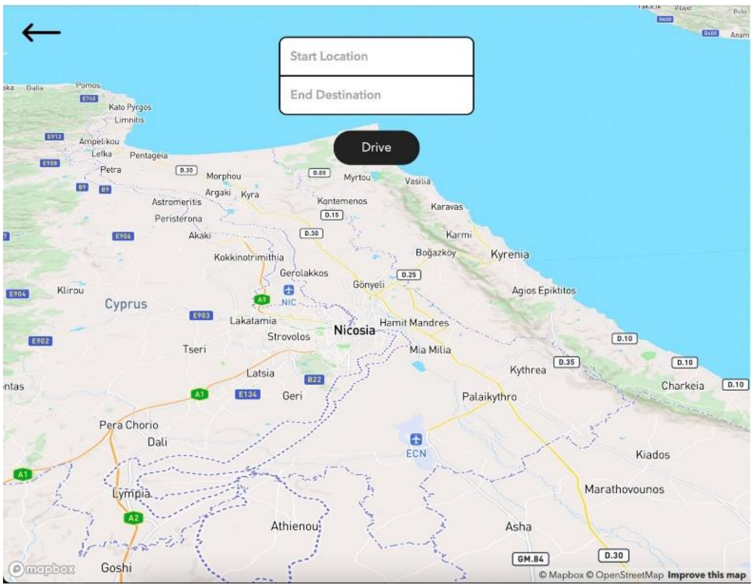


The car details are stored in the back end while the prediction is given after the user inputs the start and destination of his drive.

Upon clicking the 'predict' button for his chosen car, the user is redirected to the next page where he can input the start and destination as shown in figure 3.20. The user can input locations of his choice and he will get the distance and route, moreover, the fuel consumption, CO_2 emissions, energy consumption results of the vehicles are given. The fuel and energy charging prices are also displayed for comparison. This is elaborated more in the results section.

Figure 3.20

Integrated map with start location and destination inputs



Chapter 4

Results and Discussion

The study represents fuel consumption and CO_2 emissions in ICE vehicles. For the fuel consumption, the vehicle class, engine size, cylinders, transmission, fuel type and CO_2 rating were used as the inputs. As for the CO_2 rating, fuel consumption was use as the input and in this instance, the predicted fuel consumption was taken as the input to predict the CO_2 emissions. All the models were trained and the performances were taken note of for the fuel consumption and emissions respectively. Scatter plots were used to give a visual presentation of the models' performances as shown in figures 4.1and 4.2 for fuel consumption prediction and figures 4.3 and 4.4. for CO_2 emission predictions.

Figure 4.1

Fuel consumption scatter plots for ETR, BR and XGB Regressor models

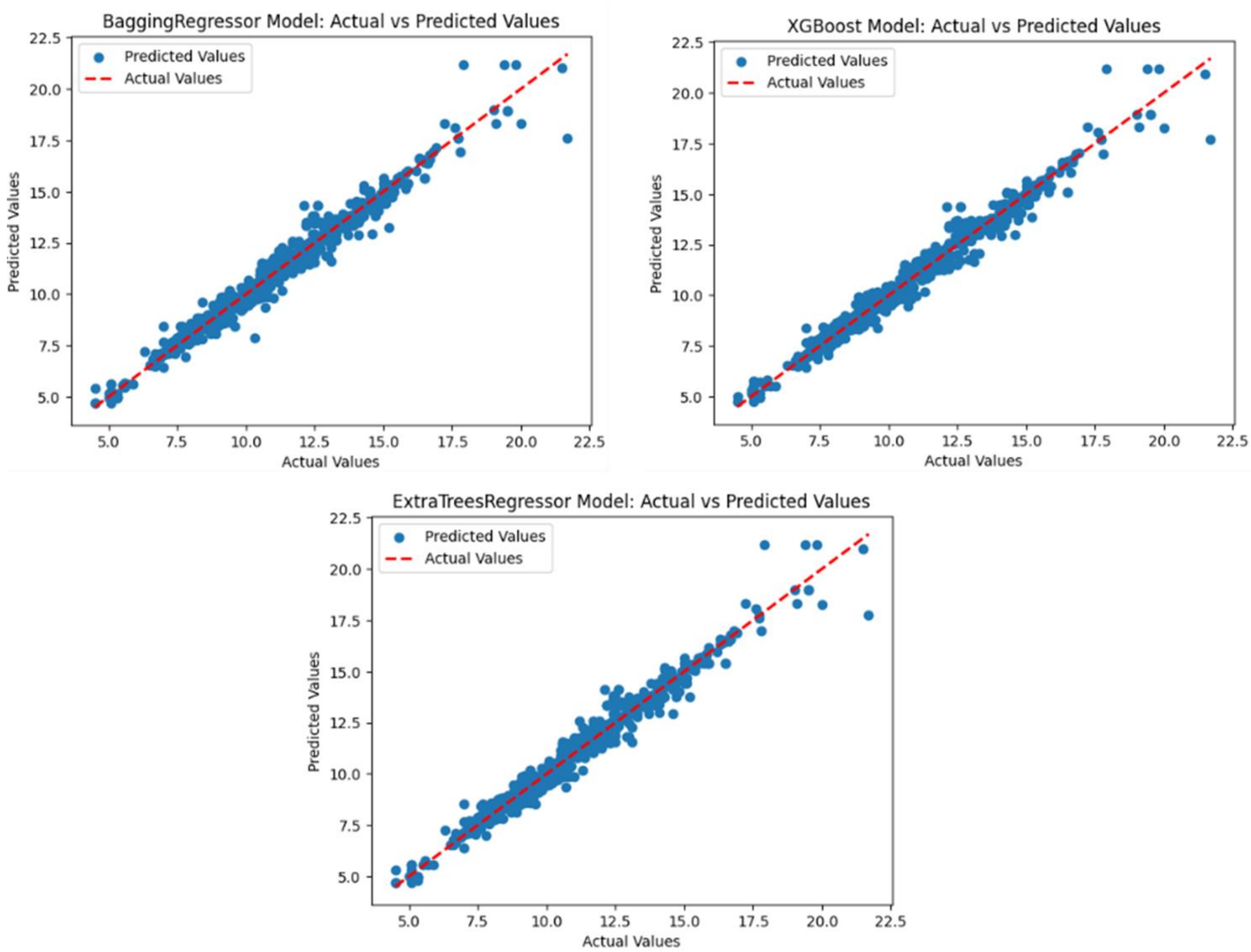
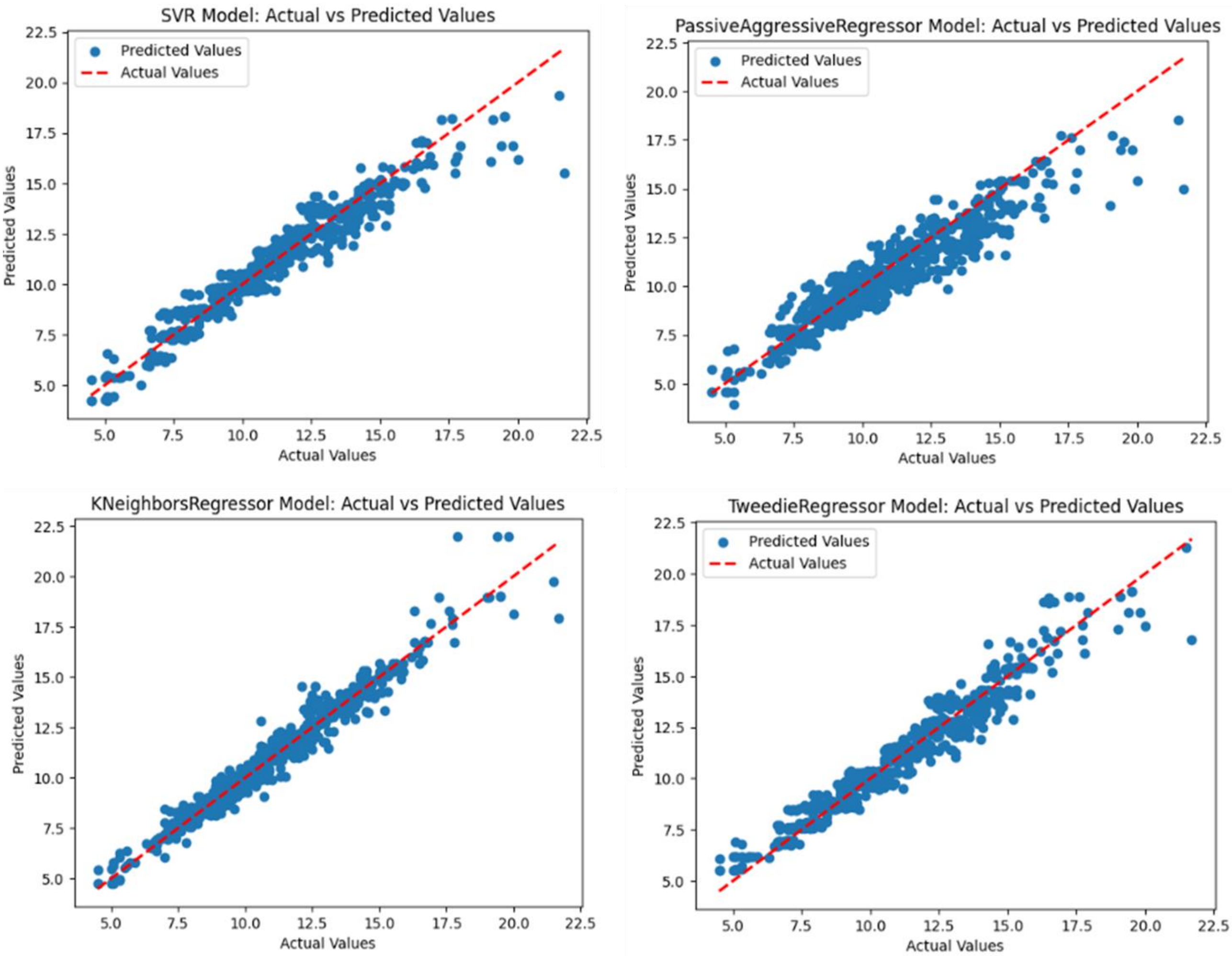


Figure 4.2

Fuel consumption scatter plots for SVR, PAR, KNN and TR models



The scatter plots are used to show the relationship between two quantitative variables and in our case would be the predicted and actual or test values(Mindrila & Phoebe, n.d.). The scatter plots allow for a quick identification of trends, clusters or potential outliers. From figure 4.1, we can see that the ETR, XGB regressor and the BR models show tight clusters around the diagonal line which indicates accurate predictions of the fuel consumption even though there are deviations. The SVR and TR models shown in are not quite impressive as most of the points stray away from the diagonal shown in figure 4.2.

Figure 4.3

CO_2 emissions scatter plots for ETR, BR and XGB Regressor models

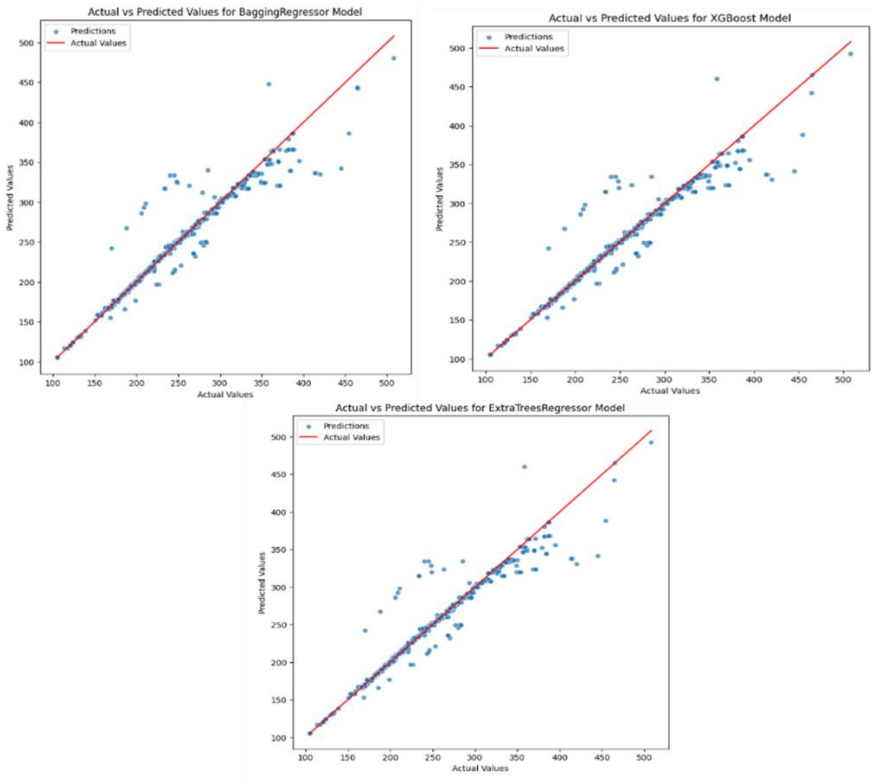
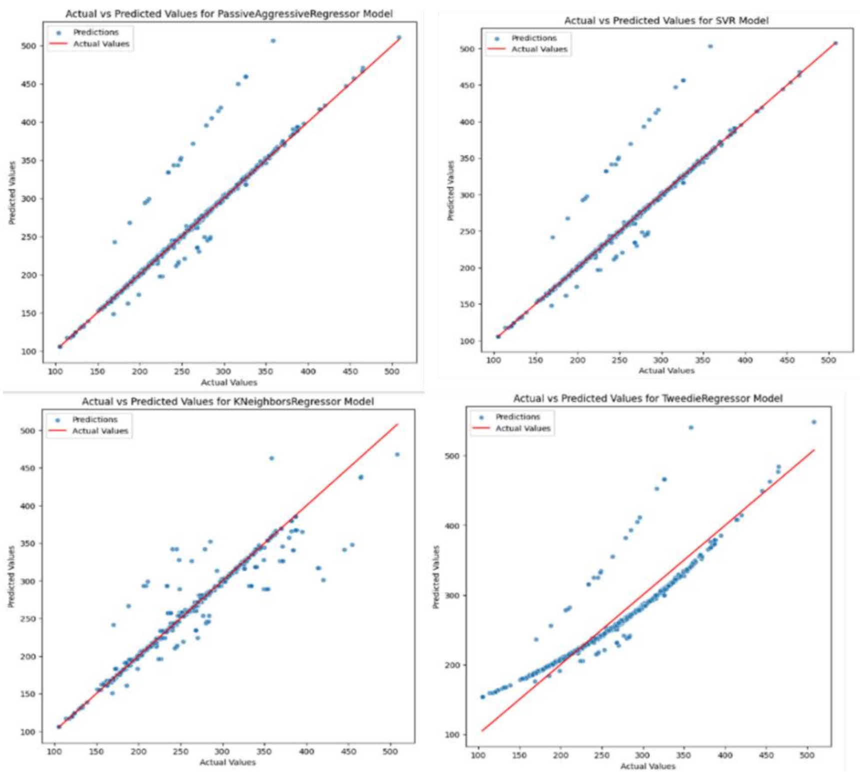


Figure 4.4

CO_2 emissions scatter plots for SVR, PAR, KNN and TR models



As observed in figure 4.3, XGB regressor model has tightly clustered point around the diagonal line although it has outliers which suggest underestimation of some values. The points in the TR model are the most scattered forming sort of a curve along the diagonal. This means that there is a non-linear relationship between the predicted and actual values leading to higher degree of uncertainty in the model. There is a tight cluster in around the diagonal for the ETR model outperforming the other models in terms of prediction accuracy. SVR and PAR models have a lot of points away from the diagonal but with some clusters at the diagonal too making their performance moderate.

The models were further evaluated using R2 score, MSE and MAE metrics as shown in table 4.1 where the model with the best performance was selected for deployment.

Table 4.1.

Model performance for fuel prediction and CO_2 emissions

	Fuel consumption			CO_2 emissions		
Model	MSE	R2	MAE	MSE	R2	MAE
SVR	0.49	0.93	0.52	314.19	0.92	4.37
PAR	1.09	0.85	0.79	325.31	0.91	4.72
BR	0.23	0.97	0.32	211.68	0.94	5.37
ETR	0.21	0.97	0.31	211.15	0.94	5.33
TR	0.43	0.94	0.50	471.33	0.87	14.12
k-Neighbours Regressor	0.29	0.96	0.36	275.14	0.93	6.21
XGB Regressor	0.23	0.97	0.34	211.22	0.94	5.33

In fuel consumption predictions, the Extra Tree Regression (ETR) model had the highest R2 score of 0.97, lowest MSE and MAE values of 0.21 and 0.31 respectively as shown in table 4.1 making it the best performing model. The next best performing models were the XGB Regressor and the BR models with an R2 score of 0.97 for both but had a relatively higher values for the MSE and MAE. The Moreover, in predicting the CO_2 emissions the (ETR) model had the highest R2 score of 0.94 although closely followed by XGB Regressor and BR models. The model also scored lowest MSE score of 211.15. The model with the lowest MAE value is the SVR model which scored 4.37. That being said, the ETR model emerged as the best model for predicting both variables.

The ETR model was then selected for deployment to the web application for real-time predictions. The model was also serialized using joblib library ensuring efficient integration and seamless user experience. The developed web app was tried with real world data and it did not disappoint. Ten vehicles were chosen for experimental purposes for real life testing where the user would have his inputs of different place he wants to go. The results are as shown in table 4.2.

Table 4.2

Example inputs from real world data

		Fuel consumption (L)			CO_2 emissions (kg)		
Car name	Distance (km)	Real	Predicted	Error	Real	Predicted	Error
Dodge challenger SXT AWD	29.36	3.73	2.22	-1.51	8.87	5.96	-2.91
Acura MDX Hybrid AWD	10.71	0.96	0.93	-0.04	2.25	2.17	-0.07
Aston Martin DB11 V12	45.53	4.46	4.19	-0.27	12.25	12.11	-0.14
AUDI A3	12	1.16	1.26	0.09	2.72	2.94	0.22
Audi Q3	25.7	2.13	2.80	0.67	4.47	6.56	2.08
BMW 328d drive Touring	34.6	2.33	2.76	0.43	6.30	6.46	0.17
Hyundai Elantra	50.2	3.51	5.84	2.32	9.49	13.45	3.97
GMC Canyon	36.6	3.51	4.00	0.49	9.41	9.37	-0.04
Honda Civic Hatchback Sport	67.4	4.92	8.00	3.08	11.46	18.60	7.14
Infiniti Q50 Hybrid AWD	7.7	0.65	0.69	0.04	1.54	1.63	0.09

The results show that the model gives accurate predictions especially with the fuel predictions. The carbon emissions predictions fluctuate in accuracy and this might be because of the model's performance whereby the accuracy was not too high. The distance column from table 4.2 is auto generated from the app when the user inputs his start location and destination. The same cars were also used to show the fuel and charging price differences as illustrated in table 4.3 below

Table 4.3

Fuel and charging costs comparison between ICE and EV

	ICE vehicles		Gunsel B9 (EV)		
Distance (km)	Fuel consumed (L)	Fuel price (TL)	Energy consumed (kWh)	Charging cost	Money saved
29.36	2.22	76.59	4.1104	38.925488	37.66
10.71	0.93	31.96	1.4994	14.199318	17.76
45.53	4.19	144.34	6.3742	60.363674	83.98
12	1.26	43.30	1.68	15.9096	27.39

25.7	2.80	96.44	3.598	34.07306	62.37
34.6	2.76	95.03	4.844	45.87268	49.15
50.2	5.84	201.19	7.028	66.55516	134.63
36.6	4.00	148.41	5.124	48.52428	99.89
67.4	8.00	275.69	9.436	89.35892	186.33
7.7	0.69	23.93	1.078	10.20866	13.73

The amount of money a road user will save is significantly a lot when he uses an electric vehicle as is evident in table 4.3. The fuel prices keep on fluctuating and for the table above, the price of petrol cars was set at 34.46 Turkish Liras per litre and for diesel at 37.1 per litre. The charging price was 9.47 for every kWh and all the prices were accessed online.

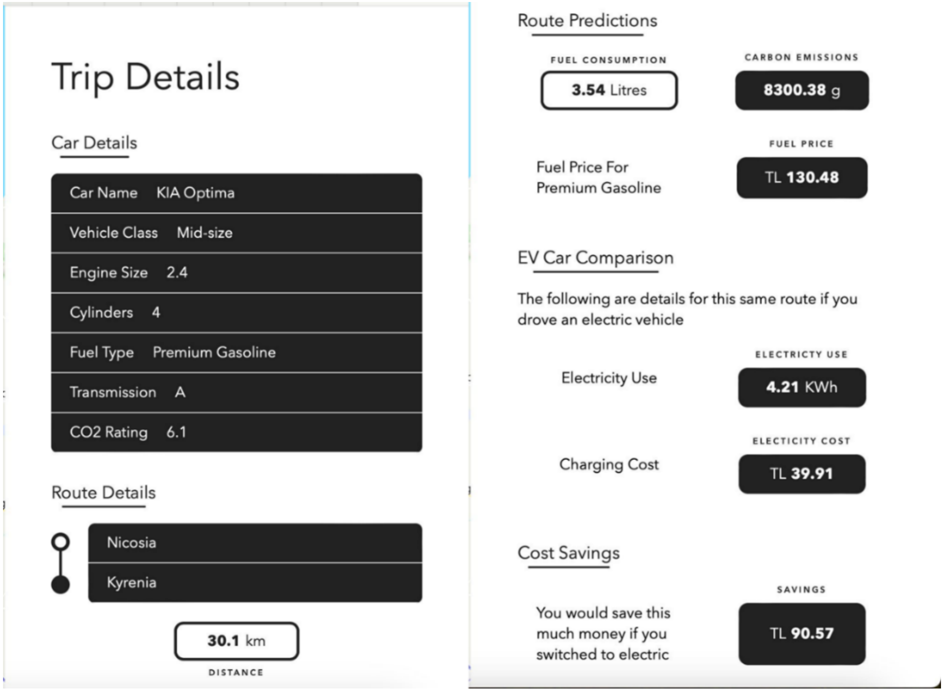
Figure 4.5

Route suggestion from the app

The route is also suggested for the user as shown in figure 4.5 with the details. In the web app there is a side bar which shows the saved car details, start and destination, distance, the predicted fuel consumption and carbon emissions, energy consumption of the EV and the fuel and charging prices as shown in figure 4.6. In the figure an example of a car input, KIA Optima, was used by inputting start location and destination as Nicosia and Kyrenia respectively located in the TRNC. The distance between then was determined to be 30.1 km with the fuel consumption of 3.54 litres and 8300 g of CO_2 emissions. The vehicle uses premium gasoline and it was calculated to be 130.48 Turkish lira. The amount of energy that would be consumed for the same distance for the electric vehicle was 4.21 kWh and for this amount of energy, the price for charging was determined to be 39.91 Turkish lira. This meant that the user would have saved 90 Turkish lira with zero carbon emissions if he used the EV instead of the conventional vehicle.

Figure 4.6.

Web app side pane showing car details, route predictions and car comparisons



The results show that it is ideal to use electronic cars because of two main reasons, cost efficiency and environment safety. Since EVs run on battery, they have no emissions compared to their counterparts ICE vehicles which contribute a lot to the global warming. There have been so many studies on using machine learning in predicting either fuel consumption or carbon emissions (Azeez et al., 2019; Nyhan et al., 2016; Syahputra, 2016; Zargarnezhad et al., 2019; Ziółkowski et al., 2021) which goes to show that the advancements in technology can really help in solving world problems such as global warming. Machine learning algorithms can be used to efficiently describe dynamic relationships in the processed data, enable multi-level modelling of key performance indices, and facilitate sustainability performance evaluation and prediction(Ge et al., 2017).

The findings of various studies advocate for adoption EVs as an ideal choice. In his study, (Hawkins et al., 2013) found out that EVs powered by the European electricity mix offer 10% to 20% decrease in Global Warming Potential (GWP) compared to conventional vehicles if they were to assume a lifetimes of 150,000 km. Even though EVs sound promising, there is need for a comprehensive approach to environmental policy and decision-making in the adoption of electric transportation technologies (Ma et al., 2012). The vehicle life cycle emissions of electrical vehicles are fairly higher than in the conventional vehicles due to the GHG emissions associated with battery manufacture. With the advent of inflation, the fuel prices are already sky rocketing, therefore using electronic vehicles becomes more economically efficient. As shown from the results, the price of charging electrical cars is way less than when one uses fuel. A study conducted in Portugal Borges et al. (2010) found out that fast charging an EV is 43% cheaper than refuelling a diesel ICE vehicle and 70% less expensive than refuelling a gasoline vehicle. The initial cost of purchasing the EV may be higher than the conventional vehicle.

The specifications in conventional vehicles also affect their performance and the amount of CO_2 gases they emit. The results show that vehicles with larger engines have more cylinders and consume more fuel compared to smaller engine vehicles with less cylinders. The more the fuel is consumed the higher the CO_2 emissions. Automatic transmission vehicles also have higher fuel consumption rates compared to manual transmission. The vehicle class of different vehicles also influence the amount of fuel consumed and CO_2 emitted. The research found that SUVs, pickup trucks and sports cars consume a lot of fuel and emit the most gases.

Chapter 5

Conclusion

This research showcases the application of machine learning models to forecast fuel consumption and CO_2 emissions in Internal Combustion Engine (ICE) cars. Utilizing a dataset encompassing vehicles from 2016 to 2023, seven machine learning models were trained, with the Extra Trees Regressor emerging as the top performer, achieving an impressive R2 score of 0.97 for fuel predictions and 0.94 for CO_2 emissions. The model was then used for deployment in a web app. An electronic vehicle's specifications were added to the backend of the web app, together with the charging and fuel prices. The web app was also integrated with map box to enable real-time visualization of the distance between start and end point. The results show that EVs are more efficient in that they are cheaper to charge with zero emissions. On the other hand, the use of predictive models can help journey planning for drivers who use ICE vehicles, also bringing about efficiency and cost savings.

The research did not consider the driving conditions, driving behaviour, and the road infrastructure; therefore, application in real life might not be as accurate. Comparisons were also made between conventional vehicles and only one electric vehicle. That being said, it would be better to include these factors for future research and add the different types of electric cars for a more extensive comparison spectrum. Future work should also have more comparisons between the use of ICE vehicles and EVs concerning emissions, initial and maintenance costs, and life span. This will help car users get a good point of view on the better option.

There is also room for more research to be done, especially in the other modes of transport such as sea and air transport. Significant emissions are produced by planes and airports. Using machine learning and artificial intelligence in flight planning will help companies save fuel costs and reduce emissions. In sea transport artificial intelligence can also be used to predict the weather conditions and suggest alternative routes to save fuel costs. There is also a need to acknowledge that the manufacture of electric vehicles leads to a lot of emissions. If the manufacturing industries considered using renewable energy sources such as solar in their plants for production, it would effectively reduce emissions. Manufacturers can also recycle used batteries instead of producing more batteries for EVs. Some recycled materials can also be added to some parts of the EVs for a fully carbon-neutral vehicle. If all this is done, then electric vehicles are the future cars.

References

Al Turjman, F., Ameen, Z. S., Serte, S., & Mubarak, A. (2024). Data Augmentation and Denoising of Computed Tomography (CT) Scan Images in Training Deep Learning Models for Rapid COVID-19 Detection. *International Journal of Business Intelligence and Data Mining*, *24*(1). https://doi.org/10.1504/ijbidm.2024.10055417

Al-Nefaie, A. H., & Aldhyani, T. H. H. (2023). Predicting CO2 Emissions from Traffic Vehicles for Sustainable and Smart Environment Using a Deep Learning Model. *Sustainability (Switzerland)*, *15*(9). https://doi.org/10.3390/su15097615

Amy Lee Walton. (n.d.). *the-guide-to-map-design*.

Azeez, O. S., Pradhan, B., Shafri, H. Z. M., Shukla, N., Lee, C. W., & Rizeei, H. M. (2019). Modelling of CO emissions from traffic vehicles using artificial neural networks. *Applied Sciences (Switzerland)*, *9*(2). https://doi.org/10.3390/app9020313

Aziz, T., Camana, M. R., Garcia, C. E., Hwang, T., & Koo, I. (2023). REM-Based Indoor Localization with an Extra-Trees Regressor. *Electronics*, *12*(20), 4350. https://doi.org/10.3390/electronics12204350

Bappon, S. D., Dey, A., Sabuj, S. M., & Das, A. (2022). Toward a Machine Learning Approach to Predict the CO2Rating of Fuel-Consuming Vehicles in Canada. *Proceedings of 2022 25th International Conference on Computer and Information Technology, ICCIT 2022*. https://doi.org/10.1109/ICCIT57492.2022.10054732

Barto, A. G. (1994). Reinforcement learning control. *Current Opinion in Neurobiology*, *4*(6), 888–893. https://doi.org/10.1016/0959-4388(94)90138-4

Bertani, A., Di Paola, G., Russo, E., & Tuzzolino, F. (2018). How to describe bivariate data. *Journal of Thoracic Disease*, *10*(2). https://doi.org/10.21037/jtd.2018.01.134

Borges, J., Ioakimidis, C. S., & Ferrão, P. (2010). Fast charging stations for electric vehicles infrastructure. *WIT Transactions on Ecology and the Environment*, *130*, 275–284. https://doi.org/10.2495/ISLANDS100241

Bousonville, T., Dirichs, M., & Kruger, T. (2019). Estimating truck fuel consumption with machine learning using telematics, topology and weather data. *Proceedings of the 2019 International Conference on Industrial Engineering and Systems Management, IESM 2019*. https://doi.org/10.1109/IESM45758.2019.8948175

Busuttil, S. (n.d.). *Support Vector Machines*.

Cabani, A., Zhang, P., Khemmar, R., & Xu, J. (2021). Enhancement of energy consumption estimation for electric vehicles by using machine learning. *IAES International Journal of Artificial Intelligence*, *10*(1). https://doi.org/10.11591/ijai.v10i1.pp215-223

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 13-17-August-2016*. https://doi.org/10.1145/2939672.2939785

Chitralekha, G., & Roogi, J. M. (2021). A Quick Review of ML Algorithms. *Proceedings of the 6th International Conference on Communication and Electronics Systems, ICCES 2021*. https://doi.org/10.1109/ICCES51350.2021.9488982

Chris, D., & Waitress, C. (2023). *waitress Documentation Release 3.0.0b0 Pylons Project Developers*.

Cleff, T. (2014). Univariate Data Analysis. In *Exploratory Data Analysis in Business and Economics*. https://doi.org/10.1007/978-3-319-01517-0_3

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, *7*.

Del Real, A. J., Dorado, F., & Durán, J. (2020). Energy demand forecasting using deep learning: Applications for the French grid. *Energies*, *13*(9). https://doi.org/10.3390/en13092242

Ge, Z., Song, Z., Ding, S. X., & Huang, B. (2017). Data Mining and Analytics in the Process Industry: The Role of Machine Learning. *IEEE Access*, *5*. https://doi.org/10.1109/ACCESS.2017.2756872

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Government of Canada. (2023). *Fuel consumption ratings*. https://open.canada.ca/data/en/dataset/98f1a129-f628-4ce4-b24d-6f16bf24dd64

Grazi, F., van den Bergh, J. C. J. M., & van Ommeren, J. N. (2008). An empirical analysis of urban form, transport, and global warming. *Energy Journal*, *29*(4). https://doi.org/10.5547/ISSN0195-6574-EJ-Vol29-No4-5

Hamed, M. A., Khafagy, M. H., & Badry, R. M. (2021). Fuel Consumption Prediction Model using Machine Learning. *International Journal of Advanced Computer Science and Applications*, *12*(11). https://doi.org/10.14569/IJACSA.2021.0121146

Hassan, M. A., Salem, H., Bailek, N., & Kisi, O. (2023). Random Forest Ensemble-Based Predictions of On-Road Vehicular Emissions and Fuel Consumption in Developing Urban Areas. *Sustainability*, *15*(2). https://doi.org/10.3390/su15021503

Hawkins, T. R., Singh, B., Majeau-Bettez, G., & Strømman, A. H. (2013). Comparative Environmental Life Cycle Assessment of Conventional and Electric Vehicles. *Journal of Industrial Ecology*, *17*(1). https://doi.org/10.1111/j.1530-9290.2012.00532.x

Herbster, M. (2001). Learning additive models online with fast evaluating kernels. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2111*. https://doi.org/10.1007/3-540-44581-1_29

Hien, N. L. H., & Kor, A. L. (2022). Analysis and Prediction Model of Fuel Consumption and Carbon Dioxide Emissions of Light-Duty Vehicles. *Applied Sciences (Switzerland)*, *12*(2). https://doi.org/10.3390/app12020803

IBM. (2022). What is the k-nearest neighbors algorithm? | IBM. *Https://Www.Ibm.Com/Topics/Knn*.

Ibrahim, A. U., Ozsoz, M., Serte, S., Al-Turjman, F., & Yakoi, P. S. (2021). Pneumonia Classification Using Deep Learning from Chest X-ray Images During COVID-19. *Cognitive Computation*. https://doi.org/10.1007/s12559-020-09787-5

IEA. (2019). *Transport sector CO2 emissions by mode in the Sustainable Development Scenario, 2000-2030*. https://www.iea.org/data-and-statistics/charts/transport-sector-co2-emissions-by-mode-in-the-sustainable-development-scenario-2000-2030

Intergovernmental Panel on Climate Change. (2012). Transport and its infrastructure. In *Climate Change 2007*. https://doi.org/10.1017/cbo9780511546013.009

Katreddi, S., Thiruvengadam, A., Thompson, G. J., & Schmid, N. A. (2023). Mixed Effects Random Forest Model for Maintenance Cost Estimation in Heavy-Duty Vehicles Using Diesel and Alternative Fuels. *IEEE Access, 11*. https://doi.org/10.1109/ACCESS.2023.3290994

Kokonendji, C. C., Bonat, W. H., & Abid, R. (2021). Tweedie regression models and its geometric sums for (semi-)continuous data. In *Wiley Interdisciplinary Reviews: Computational Statistics* (Vol. 13, Issue 1). https://doi.org/10.1002/wics.1496

Kumari, D., & Bhat, S. (2021). Accelerating the Race to Autonomous Cars – A Case Study. *International Journal of Applied Engineering and Management Letters*. https://doi.org/10.47992/ijaeml.2581.7000.0114

Li, Y., Tang, G., Du, J., Zhou, N., Zhao, Y., & Wu, T. (2019). Multilayer Perceptron Method to Estimate Real-World Fuel Consumption Rate of Light Duty Vehicles. *IEEE Access*, *7*, 63395–63402. https://doi.org/10.1109/ACCESS.2019.2914378

Liu, Q., & Wu, Y. (2012). Supervised Learning. In *Encyclopaedia of the Sciences of Learning*. https://doi.org/10.1007/978-1-4419-1428-6_451

Liu, Y. H., Liao, W. Y., Li, L., Huang, Y. T., Xu, W. J., & Zeng, X. L. (2019). Reduction measures for air pollutants and greenhouse gas in the transportation sector: A costbenefit analysis. *Journal of Cleaner Production*, *207*. https://doi.org/10.1016/j.jclepro.2018.10.094

Ma, H., Balthasar, F., Tait, N., Riera-Palou, X., & Harrison, A. (2012). A new comparison between the life cycle greenhouse gas emissions of battery electric vehicles and internal combustion vehicles. *Energy Policy*, *44*. https://doi.org/10.1016/j.enpol.2012.01.034

Mahesh Prasanna, K., Karthic, R. S., Babu, B. H., Patil, P. R., Chandan, R. R., & Hemavathi. (2022). Machine Learning Model for the Prediction of an E-Vehicle's Battery Life Cycle. *International Conference on Edge Computing and Applications, ICECAA 2022 - Proceedings*. https://doi.org/10.1109/ICECAA55415.2022.9936343

Medar, R., Rajpurohit, V. S., & Rashmi, B. (2017). Impact of Training and Testing Data Splits on Accuracy of Time Series Forecasting in Machine Learning. *2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017*. https://doi.org/10.1109/ICCUBEA.2017.8463779

Michael Rüßmann, Markus Lorenz, Philipp Gerbert, Manuela Waldner, Pascal Engel, Michael Harnisch, & Jan Justus. (2015). Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries. *Boston Consulting Group*, *9*(1).

Michaelis, L. (1993). Global warming impacts of transport. *Science of the Total Environment, The*, *134*(1–3). https://doi.org/10.1016/0048-9697(93)90344-6

Mindrila, D., & Phoebe, M. E. (n.d.). *Scatterplots and Correlation*.

Mishra, G., Sehgal, D., & Valadi, J. K. (2017). Quantitative Structure Activity Relationship study of the Anti-Hepatitis Peptides employing Random Forest and Extra Tree regressors. *Bioinformation*, *13*(03), 60–62. https://doi.org/10.6026/97320630013060

Moradzadeh, A., Mansour-Saatloo, A., Mohammadi-Ivatloo, B., & Anvari-Moghaddam, A. (2020). Performance evaluation of two machine learning techniques in heating and cooling loads forecasting of residential buildings. *Applied Sciences (Switzerland)*, *10*(11). https://doi.org/10.3390/app10113829

Morgenthaler, S. (2009). Exploratory data analysis. *WIREs Computational Statistics*, *1*(1), 33–44. https://doi.org/10.1002/wics.2

Mustapa, S. I., & Bekhet, H. A. (2016). Analysis of CO2 emissions reduction in the Malaysian transportation sector: An optimisation approach. *Energy Policy*, *89*. https://doi.org/10.1016/j.enpol.2015.11.016

Nyhan, M., Sobolevsky, S., Kang, C., Robinson, P., Corti, A., Szell, M., Streets, D., Lu, Z., Britter, R., Barrett, S. R. H., & Ratti, C. (2016). Predicting vehicular emissions in high spatial resolution using pervasively measured transportation data and microscopic emissions model. *Atmospheric Environment*, *140*. https://doi.org/10.1016/j.atmosenv.2016.06.018

Othman, K. (2022). Exploring the implications of autonomous vehicles: a comprehensive review. In *Innovative Infrastructure Solutions* (Vol. 7, Issue 2). https://doi.org/10.1007/s41062-022-00763-6

Piccinini, G. (2004). The first computational theory of mind and brain: A close look at McCulloch and Pitts's "logical calculus of ideas immanent in nervous activity." *Synthese*, *141*(2). https://doi.org/10.1023/B:SYNT.0000043018.52445.3e

Prabhudesai, K. S., Collins, L. M., & Mainsah, B. O. (2019). Automated feature learning using deep convolutional auto-encoder neural network for clustering electroencephalograms into sleep stages. *International IEEE/EMBS Conference on Neural Engineering, NER*, *2019-March*. https://doi.org/10.1109/NER.2019.8716996

Ross Morrow, W., Gallagher, K. S., Collantes, G., & Lee, H. (2010). Analysis of policies to reduce oil consumption and greenhouse-gas emissions from the US transportation sector. *Energy Policy*, *38*(3). https://doi.org/10.1016/j.enpol.2009.11.006

Salcedo-Sanz, S., Pérez-Aracil, J., Ascenso, G., & ... (2022). Analysis, Characterization, Prediction and Attribution of Extreme Atmospheric Events with Machine Learning: A Review. *ArXiv Preprint ArXiv …*.

Saleh, H., Layous, J. A., & Republic, S. A. (2022). *Machine Learning-Regression*. https://doi.org/10.13140/RG.2.2.35768.67842

Saman Siadati. (2020). What is Unsupervised Learning. *Ibm*, *August 2018*.

Samuel, A. L. (2000). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, *44*(1–2). https://doi.org/10.1147/rd.441.0206

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. In *SN Computer Science* (Vol. 2, Issue 3). https://doi.org/10.1007/s42979-021-00592-x

Syahputra, R. (2016). Application of neuro-fuzzy method for prediction of vehicle fuel consumption. *Journal of Theoretical and Applied Information Technology*, *86*(1).

Tewari, A., Sarguroh, N., Kingrani, P., Shetty, T., & Motwani, R. (2021). AI-Based Autonomous Driving Assistance System. *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*. https://doi.org/10.1109/ICCMC51019.2021.9418403

Tiwari, T., Tiwari, T., & Tiwari, S. (2018). How Artificial Intelligence, Machine Learning and Deep Learning are Radically Different? *International Journal of Advanced*

Research in Computer Science and Software Engineering, *8*(2). https://doi.org/10.23956/ijarcsse.v8i2.569

Wang, W., Chakraborty, G., & Chakraborty, B. (2021). Predicting the risk of chronic kidney disease (Ckd) using machine learning algorithm. *Applied Sciences (Switzerland)*, *11*(1). https://doi.org/10.3390/app11010202

Yin, X., Li, Z., Shah, S. L., Zhang, L., & Wang, C. (2016). Fuel Efficiency Modeling and Prediction for Automotive Vehicles: A Data-Driven Approach. *Proceedings - 2015 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2015*. https://doi.org/10.1109/SMC.2015.442

Zargarnezhad, S., Dashti, R., & Ahmadi, R. (2019). Predicting vehicle fuel consumption in energy distribution companies using ANNs. *Transportation Research Part D: Transport and Environment*, *74*. https://doi.org/10.1016/j.trd.2019.07.020

Zhang, R., Chen, H., Xie, P., Zu, L., Wei, Y., Wang, M., Wang, Y., & Zhu, R. (2023). Exhaust Emissions from Gasoline Vehicles with Different Fuel Detergency and the Prediction Model Using Deep Learning. *Sensors*, *23*(17). https://doi.org/10.3390/s23177655

Ziółkowski, J., Oszczypała, M., Małachowski, J., & Szkutnik-Rogoż, J. (2021). Use of artificial neural networks to predict fuel consumption on the basis of technical parameters of vehicles. *Energies*, *14*(9). https://doi.org/10.3390/en14092639

Appendices

Appendix A. JavaScript project controller code

```
let viewContainer = document.querySelector(".user-credentials-view");
let imageView = document.querySelector(".image-view");
let dashboard = document.querySelector(".dashboard");
let dashboardLoader = document.querySelector(".dashboard-loader");
let userActions =document.querySelector(".user-actions");
let loginContainer = document.querySelector(".login-container")
let signupContainer = document.querySelector(".signup-container")
let mapView = document.querySelector(".map-view-container")
let carOverlay = document.querySelector(".car-overlay");
let carListContainer = document.querySelector(".car-list-container");
let carNameInput = carOverlay.querySelector("#car-name-input");
let vehicleClassInput = carOverlay.querySelector("#vehicle-class-input");
let engineSizeInput = carOverlay.querySelector("#engine-size-input");
let cylindersInput = carOverlay.querySelector("#cylinders-input");
let transmissionInput = carOverlay.querySelector("#transmission-input");
let CO2RatingInput = carOverlay.querySelector("#co2-rating-input");
let fuelTypeInput = carOverlay.querySelector("#fuel-type-input");
let APIResults = { };
if(USERNAME != ""){
   showUsername(USERNAME);
   maximize();
   getAvailableCars();
}
function AJAXCall(callObject){
   let {
      phpFilePath,
      rejectMessage,
      params,
      type,
   } = callObject;
   return new Promise((resolve,reject) => {
      let xhr = new XMLHttpRequest();
      xhr.open("POST", phpFilePath, true);
      xhr.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
      xhr.onload = function(){
         if( this.status == 200 ){
            let result = type == "fetch" ?
            JSON.parse(this.responseText) : this.responseText ;
```

```
                //TODO: Take a look one more time
                if(result.length < 1 && type != "fetch") reject(rejectMessage || "SQLError");
                else { resolve(result) }
            }
            else{
                reject("Error With PHP Script");
            }
        }
        xhr.send(params);
    });
}
function showUsername(username){
    let placeholder = document.querySelector(".username");
    placeholder.textContent = username;
}
function showLogin(){
    loginContainer.style.display = "grid";
    signupContainer.style.display = "none";
}
function showSignup(){
    signupContainer.style.display = "grid";
    loginContainer.style.display = "none";
}
async function logout(){
    let callObject = {
        phpFilePath: "include/logout-script.php",
        rejectMessage: "oooops",
        params: "",
        type: "post",
    }
    try {
        let result = await AJAXCall(callObject);
        console.log(result);
    }
    catch(error){
        console.log(error);
    }
    minimize();
}
function minimize(){
    viewContainer.style.width = "500px";
```

```
    viewContainer.style.boxShadow = "0px 0px 200px var(--accent)";
    dashboard.style.display = "none";
    userActions.style.display = "none";
    imageView.style.left = "0px";
    viewContainer.style.top = "0px";
    viewContainer.style.left = "calc(100vw - 500px)";
    loginContainer.style.display = "grid";
    loginContainer.style.transform = "scale(0)";
    signupContainer.style.display = "none";
    setTimeout(() => {
        loginContainer.style.transform = "scale(1)";
    }, 200)
}
function maximize(){
    viewContainer.style.width = "100vw";
    viewContainer.style.boxShadow = "unset";
    // viewContainer.style.position = "absolute";
    imageView.style.left = "calc(-100vw + 500px)";
    viewContainer.style.top = "0px";
    viewContainer.style.left = "0px";
    loginContainer.style.display = "none";
    signupContainer.style.display = "none";
    // show loader 2s
    dashboardLoader.style.display = "flex";
    setTimeout(() => {
        dashboardLoader.style.display = "none";
        dashboard.style.display = "grid";
        userActions.style.display = "grid";
    }, 2000)
}
async function login() {
    let loginButton = document.querySelector(".login-button");
    let loginLoader = loginButton.querySelector(".button-loader");
    let buttonText = loginButton.querySelector("p");

    buttonText.style.display = "none";
    loginLoader.style.display = "flex";
    // REGEX Inputs
    let username = document.querySelector(".login-username-field").value;
    let password = document.querySelector(".login-password-field").value;
    let params = username=${username}&&+password=${password};
```

```
    let callObject = {
        phpFilePath: "include/login-script.php",
        rejectMessage: "shoot",
        params,
        type: "post",
    }
    try {
        let result = await AJAXCall(callObject)
        console.log(result);
        if( result == "success" ){
            setTimeout(() => {
                maximize();
                getAvailableCars();
                showUsername(username);
                buttonText.style.display = "block";
                loginLoader.style.display = "none";
            }, 2000);
        }
        else {
            // showWrongCredentialsWarning();
            setTimeout(() => {
                buttonText.style.display = "block";
                loginLoader.style.display = "none";
            }, 1300);
        }
    }
    catch(error){
        console.log((error));
    }
}
function slideOutMapView(){
    mapView.style.left = "100vw";
    viewContainer.style.left = "0vw";
}
function showAddCarOverlay(){
    carOverlay.style.display = "grid";
}

function closeAddCarOverlay(){
    carOverlay.style.display = "none";
    resetCarForm();
```

```
}

async function addCar(element){
    let text = element.querySelector("p");
    let buttonLoader = element.querySelector(".button-loader");
    text.style.display = "none";
    buttonLoader.style.display = "flex";
    let details = getCarDetails();
    try {
        console.log(details);
        let result = await sendCarDetails(details);
        console.log("result: ", result);
        //TODO: resetFormAndLoadingButton()
    }
    catch(error){
        console.log(error);
    }
    setTimeout(() => {
        text.style.display = "none";
        buttonLoader.style.display = "flex";
        closeAddCarOverlay();
        getAvailableCars();
        // refreshDashboard();
    }, 3000);
}

function resetCarForm() {
    carNameInput.value = "";

    vehicleClassInput.setAttribute("data-value", "Select Vehicle Class");
    vehicleClassInput.setAttribute("data-empty", "true");
    resetRangeElement(engineSizeInput);
    resetRangeElement(cylindersInput);
    resetRangeElement(CO2RatingInput);
    transmissionInput.setAttribute("data-value", "Select Transmission");
    transmissionInput.setAttribute("data-empty", "true");
    fuelTypeInput.setAttribute("data-value", "Select Fuel Type");
    fuelTypeInput.setAttribute("data-empty", "true");
    let button = document.querySelector(".add-car-button");
    button.querySelector(".button-loader").style.display = "none";
    button.querySelector("p").style.display = "block";
```

```
}
function getCarDetails() {
  let carName = carNameInput.value;
  let vehicleClass = vehicleClassInput.getAttribute("data-value");
  let engineSize = engineSizeInput.value;
  let cylinders = cylindersInput.value;
  let transmission = transmissionInput.getAttribute("data-value");
  let CO2Rating = CO2RatingInput.value;
  let fuelType = fuelTypeInput.getAttribute("data-value");

  switch (fuelType) {
    case "Diesel":
      fuelType = "D";
    break;
    case "Ethanol":
      fuelType = "E";
    break;
    case "Regular Gasoline":
      fuelType = "X";
    break;
    case "Premium Gasoline":
      fuelType = "Z";
    break;
  }


  // checkEmptyInputs();

  return {
    carName,
    vehicleClass,
    engineSize,
    cylinders,
    transmission,
    CO2Rating,
    fuelType
  }

}

async function sendCarDetails(details){
```

```
    let {
        carName,
        vehicleClass,
        engineSize,
        cylinders,
        transmission,
        CO2Rating,
        fuelType
    } = details;

    let params =
    userID=${userID}&&+
    carName=${carName}&&+
    vehicleClass=${vehicleClass}&&+
    engineSize=${engineSize}&&+
    cylinders=${cylinders}&&+
    transmission=${transmission}&&+
    CO2Rating=${CO2Rating}&&+
    fuelType=${fuelType};

    let callObject = {
        phpFilePath: "include/add-car.php",
        rejectMessage: "car not added",
        params,
        type: "post",
    }

    return await AJAXCall(callObject);
}

async function getAvailableCars(){

    let params = userID=${userID};

    let callObject = {
        phpFilePath: "include/cars.fetch.php",
        rejectMessage: "cars not fetched",
        params,
        type: "fetch",
    }
```

```
    try {
        let cars = await AJAXCall(callObject);

        if (cars.length > 0) {

            let innerHTML = cars.map( car =>
                `<div class="car-item">
                    <h1>${car.carName}</h1>
                    <div class="button" onclick="predictWithID(${car.id})">Predict</div>
                </div>`
            )

            carListContainer.innerHTML = innerHTML.join("");
        }
        else {
            carListContainer.innerHTML =
            <div class="span-all-directions">There are no cars yet, add a new car.</div>
        }
    }
    catch(error){
        console.log(error);
    }
async function predictWithID(givenID){

    mapView.style.left = "0vw";
    viewContainer.style.left = "-100vw";

    try{
        let carArray = await fetchDetailsFor(givenID);

        carArray = carArray[0]



        // let carArray = {
        //     carName: "Mercedes",
        //     vehicleClass: "Two-seater",
        //     engineSize: 5,
        //     cylinders: 6,
        //     transmission : "AV",
        //     CO2Rating: 6,
```

```
        //     fuelType: "X"
        // }
        let indexesArray = convertToArrayOfIndexes(carArray);
        console.log("indexesArray: ", indexesArray)
        let results = await predictWithObject(indexesArray);
        setSidePaneValues(carArray, results);
        console.log("results: ", results);
        // Make sure server is on...
        // setResultsGlobally
        // <--- Wait for start and destination
        // <-- displayDetailsOnScreen()
    }
    catch(error){
        console.log(error)
    }
}
async function fetchDetailsFor(givenID){
    let params = carID=${givenID};
    let callObject = {
        phpFilePath: "include/car.fetch.php",
        rejectMessage: "car details not fetched",
        params,
        type: "fetch",
    }

    try {
        return await AJAXCall(callObject);
    }
    catch(error){
        console.log(error);
    }
}
function setSidePaneValues(carArray, results){
        let {
            carName,
            vehicleClass,
            engineSize,
            cylinders,
            transmission,
            CO2Rating,
            fuelType,
```

```
        } = carArray;
        APIResults = results;
        let carNameBox = document.querySelector(".car-name-box");
        let carVehicleClassBox = document.querySelector(".car-vehicle-class-box");
        let carEngineBox = document.querySelector(".car-engine-box");
        let carCylinderBox = document.querySelector(".car-cylinders-box");
        let carFuelTypeBox = document.querySelector(".car-fuel-type-box");
        let carTransmissionBox = document.querySelector(".car-transmission-box");
        let carCO2RatingBox = document.querySelector(".car-co2-rating-box");
        carNameBox.querySelector("div").textContent = carName;
        carVehicleClassBox.querySelector("div").textContent = vehicleClass;
        carEngineBox.querySelector("div").textContent = engineSize;
        carCylinderBox.querySelector("div").textContent = cylinders;

        let _fuelType;

        switch (fuelType) {
            case "D":
                _fuelType = "Diesel";
            break;
            case "E":
                _fuelType = "Ethanol";
            break;
            case "X":
                _fuelType = "Regular Gasoline";
            break;
            case "Z":
                _fuelType = "Premium Gasoline";
            break;
        }
        carFuelTypeBox.querySelector("div").textContent = _fuelType;
        carTransmissionBox.querySelector("div").textContent = transmission;
        carCO2RatingBox.querySelector("div").textContent = CO2Rating;
}
```

Appendix B. Map view controller code

```
let suggestionsView = document.querySelector(".suggestions");
let startLocationName = ""
let endLocationName = ""
let startInputBox = document.querySelector(".start-input-location")
let endInputBox = document.querySelector(".end-input-location")
let newID = generateUUID();
```

```
mapboxgl.accessToken =
'pk.eyJ1IjoiaWJyYWhpbWFtZTEzIiwiYSI6ImNsb2xlaDUxbDJlcXYya3A5bzZoZWc5MzkifQ.YyKfquv1mvX
7xrUj5oG1Ow';
const map = new mapboxgl.Map({
   container: 'map', // container ID
   style: 'mapbox://styles/mapbox/streets-v12', // style URL
   center: [33.321258,35.212448], // starting position [lng, lat]
   pitch: 60,
   bearing: -60,
   zoom: 10
});
let source;
let destination;
let startMarker;
let endMarker;
async function getRoute(start, end) {
   if(startMarker) startMarker.remove()
   if(endMarker) endMarker.remove()
   // Set marker options.
endMarker = new mapboxgl.Marker({
   color: "#FFFFFF",
   draggable: true
   })
   .setLngLat(end)
   .setPopup(new mapboxgl.Popup().setHTML("<h1>Destination</h1>"))
   .addTo(map);
startMarker = new mapboxgl.Marker({
   color: "var(--accent)",
   draggable: true
   })
   .setLngLat(start)
   .setPopup(new mapboxgl.Popup().setHTML("<h1>Destination</h1>"))
   .addTo(map);
   const bbox = [start, end];
   map.fitBounds(bbox, {
   padding: {top: 200, bottom: 300, left: 150, right: 150},
   duration: 2000,
   });
const query = await fetch(
https://api.mapbox.com/directions/v5/mapbox/driving/${start[0]},${start[1]};${end[0]},${end[1]}?steps=true&
geometries=geojson&access_token=${mapboxgl.accessToken},
```

```
{ method: 'GET' }
);
   const json = await query.json();
   const routes = json.routes;
   console.log("routes: ",routes);
   const data = json.routes[0];
   console.log("data: ", data.duration /* in seconds */);
   console.log("distance: ", data.distance);
   const route = data.geometry.coordinates;
   setDistance(data.distance);
const geojson = {
   type: 'Feature',
   properties: {},
   geometry: {
   type: 'LineString',
   coordinates: route
   }
};
console.log(geojson);
// if the route already exists on the map, we'll reset it using setData
if (map.getSource('route')) {
map.getSource('route').setData(geojson);
}
// otherwise, we'll make a new request
else {
map.addLayer({
id: 'route',
type: 'line',
source: {
   type: 'geojson',
   data: geojson
},
layout: {
   'line-join': 'round',
   'line-cap': 'round'
},
paint: {
   'line-color': '#ff0000',
   'line-width': 5,
   'line-opacity': 0.75
}
```

```
});
}
// add turn instructions here at the end
}
async function giveSuggestions(element) {
    let value = element.value
    showSuggestionsView();

    const query = await fetch(
https://api.mapbox.com/search/searchbox/v1/suggest?q=${value}&limit=3&session_token=${newID}&access_
token=${mapboxgl.accessToken},
        { method: 'GET' }
    );
    const json = await query.json();
    suggestionsView.innerHTML = "";
    if (element.className == "start-input-location"){
        if(json.suggestions){
            json.suggestions.forEach( result => {
                populateList(result.name, setSourceTo);
            });
        }
        // store source lang lat
    }
    else if(element.className == "end-input-location"){
        if(json.suggestions){
            json.suggestions.forEach( result => {
                populateList(result.name, setEndTo);
            });
        }
    }
    function populateList(value, evenListenerCallback) {
        let element = document.createElement('div');
        console.log("value: ",value);
        element.textContent = value;
        element.className = "suggestion-row";
        element.addEventListener('click', () => { evenListenerCallback(value); hideSuggestionsView() });
        suggestionsView.appendChild(element);
    }
    async function setSourceTo(value){
        startLocationName = value;
        console.log("babe: ", value);
```

```
        startInputBox.value = value;
        document.querySelector(".start-route-box").textContent = value;
        const query = await fetch(
  https://api.mapbox.com/geocoding/v5/mapbox.places/${value}.json?access_token=${mapboxgl.accessToken},
            { method: 'GET' }
        );
        const json = await query.json();
        if(json){
            console.log(json)
            console.log("[lat, long]: ", json.features[0].center);
            let result = json.features[0].center;
            source = result;
            // getRoute(source, result);
        }
        // hideSuggestionsView();
    }
    async function setEndTo(value){
        endLocationName = value;
        endInputBox.value = value;
        document.querySelector(".end-route-box").textContent = value;
        const query = await fetch(
  https://api.mapbox.com/geocoding/v5/mapbox.places/${value}.json?access_token=${mapboxgl.accessToken},
            { method: 'GET' }
        );
        const json = await query.json();
        if(json){
            console.log(json)
            console.log("[lat, long]: ", json.features[0].center);
            let result = json.features[0].center;
            destination = result;
            // getRoute(source, result);
        }
        // hideSuggestionsView();
    }
    function showSuggestionsView(){
        suggestionsView.style.display = "grid"
    }
    function hideSuggestionsView(){
        suggestionsView.style.display = "none"
    }
}
```

```
function setDistance(distance){
   let distanceBox = document.querySelector(".distance-box");
   let roundedDistance = Math.round(distance / 1000 * 100) / 100 ;
   distanceBox.querySelector("b").textContent = roundedDistance;
   let {
      fuel,
      co2
   } = APIResults;
   let fuelResult = (fuel * roundedDistance) / 100;
   fuelResult = Math.round(fuelResult * 100) / 100;
   let co2Result = co2 * roundedDistance;
   co2Result = Math.round(roundedDistance * 100);
   let litresBox = document.querySelector(".litres-box");
   let gramsBox = document.querySelector(".grams-box");
   litresBox.querySelector("b").textContent = fuelResult;
   gramsBox.querySelector("b").textContent = co2Result;
   let fuelPrice = getFuelPrice(fuelResult);
   let electicityCost = setEVDetails(roundedDistance);
   let fuelPriceBox = document.querySelector(".fuel-price-box");
   fuelPriceBox.querySelector("b").textContent = fuelPrice;

   let savingsResult = fuelPrice - electicityCost;
   savingsResult = Math.round((fuelPrice - electicityCost) * 100) / 100;
   let savingsBox = document.querySelector(".savings-box");
   savingsBox.querySelector("b").textContent = savingsResult;
   // What if the fuelPrice is less than the electic cost.
   // What if it comes back negative
}
function getFuelPrice(fuelInLitres) {
   let fuelPriceDifferentiator = document.querySelector(".fuel-price-differentiator");
   let carFuelTypeBox = document.querySelector(".car-fuel-type-box");
   let fuelType = carFuelTypeBox.querySelector("div").textContent;
   let costPrice;
   fuelPriceDifferentiator.textContent = Fuel Price For ${fuelType};
   switch (fuelType){
      case "Diesel":
         costPrice = 39.06;
      break;
      case "Ethanol":
         costPrice = 34.91;
      break;
```

```
        case "Regular Gasoline":
            costPrice = 34.55;
        break;
        case "Premium Gasoline":
            costPrice = 36.86;
        break;
    }
    return Math.round(fuelInLitres * costPrice * 100) / 100;
}
function getElectricityCost(electicityUse) {
    return Math.round(electicityUse * 9.48 * 100) / 100;
}
function calculateElectricityUse(distance){
    return Math.round(distance * 0.14 * 100) / 100;
}
function setEVDetails(distance){
    let electicityUse = calculateElectricityUse(distance);
    let electricityCost = getElectricityCost(electicityUse);
    let electicityUseBox = document.querySelector(".electicity-use-box");
    let electicityCostBox = document.querySelector(".electricity-cost-box");
    electicityUseBox.querySelector("b").textContent = electicityUse;
    electicityCostBox.querySelector("b").textContent = electricityCost;

    return electricityCost;
}
function generateUUID() { // Public Domain/MIT
    var d = new Date().getTime();//Timestamp
    var d2 = ((typeof performance !== 'undefined') && performance.now && (performance.now()*1000)) ||
0;//Time in microseconds since page-load or 0 if unsupported
    return 'xxxxxxxx-xxxx-4xxx-yxxx-xxxxxxxxxxxx'.replace(/[xy]/g, function(c) {
        var r = Math.random() * 16;//random number between 0 and 16
        if(d > 0){//Use timestamp until depleted
            r = (d + r)%16 | 0;
            d = Math.floor(d/16);
        } else {//Use microseconds since page-load if supported
            r = (d2 + r)%16 | 0;
            d2 = Math.floor(d2/16);
        }
        return (c === 'x' ? r : (r & 0x3 | 0x8)).toString(16);
    });
}
```

Appendix X. Turnitin Similarity Report

turnitin

All Classes | Join Account (TA) | Quick Submit

NOW VIEWING: HOME > QUICK SUBMIT

As of January 23rd, approved paper deletion requests will be immediately and permanently deleted. Once approved for deletion, papers will no longer be recoverable.

About this page

This is your assignment inbox. To view a paper, select the paper's title. To view a Similarity Report, select the paper's Similarity Report icon in the similarity column. A ghosted icon indicates that the Similarity Report has not yet been generated.

Yakın Doğu Üniversitesi

QUICK SUBMIT | NOW VIEWING: ALL PAPERS ▼

Submit

AUTHOR	TITLE	SIMILARITY	FILE	PAPER ID	DATE
Abubakar M	ABSTRACT	0%		2282821020	31-Jan-2024
Abubakar M	Chapter 5	0%		2282821035	31-Jan-2024
Abubakar M	Chapter 1	1%		2282821022	31-Jan-2024
Abubakar M	Chapter 4	5%		2282821033	31-Jan-2024
Abubakar M	Chapter 3	8%		2282821030	31-Jan-2024
Abubakar M	Full Thesis	8%		2282824086	31-Jan-2024
Abubakar M	Chapter 2	14%		2282821025	31-Jan-2024