ENS 491-492 – Graduation Project

Final Report

Project Title: Data Based Vehicle Energy Consumption Modeling

Group Members:

Serhan Yorulmaz

Yunus Delipınar

Mert Kılıçaslan

Supervisor(s):

Tuğçe Yüksel

Öznur Taştan

Date: 04.06.2023

1. EXECUTIVE SUMMARY

The Increasing importance of sustainable transportation alternatives has led to significant development in the adoption of electric vehicles in various industries. One such sector is logistics and delivery where electric delivery trucks are gaining popularity due to their capability for reducing greenhouse gas emissions and decreasing expenses in conventional fuel costs (Morrissey, Weldon, & O'Mahony, 2016). As a result, it has become essential to investigate and analyze data related to electric vehicles to expand understanding of their energy consumption patterns. Within this context, Data Based Vehicle Energy Consumption Modeling aims to examine the data of electric delivery trucks to enhance the understanding of their attributes and energy consumption patterns to provide insights that can be used to optimize efficiency by predicting which features play a key role in their energy consumption through the application of machine learning algorithms. The dataset under examination in this project has been gathered from Smith Newton medium-duty electric delivery trucks, which are actively utilized by Frito-Lay North America, a US-based company involved in the distribution network of PepsiCo products.

There is a notable gap in existing literature concerning a thorough analysis of their energy consumption patterns and driving profiles. While many studies have investigated electric vehicles and their energy consumption, research specifically targeting medium-duty electric delivery trucks, such as those utilized by Frito-Lay North America, remains limited. Additionally, there is a lack of research that integrates data analysis, machine learning algorithms, and visualization techniques to offer a well-rounded understanding of the driving features that play a vital role in the energy consumption patterns of those electric delivery trucks (Lee, Kim, & Choi, 2022). Addressing this gap in the literature is crucial, as it would contribute to optimizing the efficiency of electric delivery trucks in the product distribution network of companies. A deeper understanding of these vehicles' energy consumption patterns and driving profiles can pave the way for more economy and naturally friendly transportation alternatives, which not only contribute to the reduction of air pollution but also lead to cost savings for many businesses in the logistics sector. Moreover, the project's findings could potentially stimulate innovation in the development of electric delivery trucks, enabling manufacturers to reconfigure their products to meet the specific requirements and challenges of the logistics sector. This could result in extended renewable energy utilization, energy conservation, and optimized performance under certain

driving conditions, further reinforcing the significance of electric delivery trucks within sustainable transportation solutions.

In the context of the project, the primary objectives and intended results include investigating and analyzing real-world data related to the driving profiles of Smith Newton medium-duty electric delivery trucks used by Frito-Lay North America. The project aims to earn a deeper understanding of the energy consumption patterns of these electric delivery trucks, considering factors such as trip planning, voltage-current relationship, driver behavior, and external temperature measurement. By constructing machine learning algorithms, the project seeks to predict the energy consumption of electric vehicles and identify trends that could contribute to the optimization of electric delivery truck performance; while also utilizing measurable criteria, such as prediction accuracy and relevance of insights, to evaluate its success. The project carefully assesses the effectiveness of these models by comparing predicted energy consumption patterns with manually calculated data and interprets its success by checking its objectives, criteria, and intended results, while also identifying areas for further improvement or research.

2. PROBLEM STATEMENT

The original complex problem addressed by this project is as accurate as the estimation of energy consumption for Smith Newton medium-duty electric delivery trucks during specific trips, considering various factors and constraints which are provided by the user. The motivation behind tackling this problem stems from the increasing importance of electric vehicles in commercial and personal use. By understanding the characteristics of electric vehicles and their energy consumption patterns, the project aims to contribute to the energy consumption calculation techniques, and machine learning models in this context by coming up with an acceptable energy consumption model.

In comparison to the existing literature, the project seeks to expand upon previous findings by incorporating machine learning algorithms into the energy consumption modeling process. While previous research, such as the study conducted by Burton (2013), explored the advantages of electric vehicles in fuel economy and carbon emissions, they primarily focused on visualizing data without incorporating predictive models. In the other research that was conducted by Pan (2023),

energy consumption prediction is based on real-world data just like our project. The similarity between that study and our project comes from the data collected from vehicles such as average speed, minimum temperature, maximum temperature, etc. Unlike that article, in our project, data is manipulated and new data variables such as traffic categories are derived from available ones to have a better prediction. The uniqueness of this project lies in its aim to develop a machine-learning model that can estimate energy consumption for specific trips, taking into account user-provided constraints.

By leveraging the dataset collected from Smith Newton medium-duty electric delivery trucks used by Frito-Lay North America's logistics, the project aims to provide an accurate estimation of energy consumption for that specific electric vehicle. This goes beyond the scope of previous studies, which mainly focused on visualization and comparative analysis of vehicle performance metrics with fuel-powered vehicles. Therefore, the project's goals encompass both investigating and analyzing the data and developing a machine-learning model capable of estimating energy consumption for electric vehicles with a high level of accuracy.

The project does not necessitate adherence to specific technical, scientific, or engineering norms, codes, protocols, or requirements since there is no strict methodology or steps to be followed in the project.

2.1. **Objectives/Tasks**

Researching previous findings beforehand the implementation of a solution: Understanding the problem, having overall knowledge about the problem, being advised by the previous research, and determining what to expect from the solution.

Data examination: examining the dataset, and determining what part of the data is useful for the solution. Studying the structure, format, and content of the data to gain more knowledge about its characteristics, limitations, and potential applications.

Data extraction: Preparing the data in a format suitable for future analysis and modeling by extracting unnecessary data, extracting problematic data from the dataset.

Data visualization: To gain a comprehensive understanding of the data, visualize the data by using some data visualization techniques. Uncovering patterns, relationships, and trends helps to develop better models.

Data analysis: Having a meaningful analysis that will be used in estimation. Applying statistical techniques and machine learning algorithms to identify crucial factors that have an impact on energy consumption in our vehicles.

Developing a machine learning model which estimates the energy consumption with certain constraints: Estimating energy consumption of a certain trip by specified constraints provided by the user. Taking advantage of various models with different parameters to get a better estimation.

2.2. Realistic Constraints

Economic, Available resources:

There is no monetary budget since every resource and tool needed is published online freely such as programming languages, libraries, data, IDE, articles, etc. Also, processing the data does not require massive processing power or time so it can be neglected. There is no cost to the project at all.

Time Frame:

The total time needed for the project to be completed is around 28 weeks. The time needed for every step of the project is different. Since the scope of the project is determined considering the time constraints, it is planned to be finished before the end of the time.

Environmental:

There is no environmental effect of the project. Since the project does not need any resources at all, it does not have any noticeable adverse effects on the environment.

Social:

Every tool and resource that will be used during the project is freely available online. There is no copyright for them which will cause legal problems. There is no chance of violating anyone's rights because of the nature of the project.

Health and Safety:

Since the model will not guarantee the prediction of energy consumption, people who will use the model should not rely on the model's results. So, the project has no chance to affect anybody's safety or health.

Ethical:

As mentioned above, every tool and resource is free to use. There is no violation of copyright. The project does not guarantee or claim anything. Also, the project has no direct effect on anybody. There are no ethical concerns about the project.

Sustainability:

The lifetime of the model ends whenever another model scientifically proves that it gives more accurate results. However, the model is open to development and can be trained with more data to give better results.

Inspectability:

The accuracy of the model can be tested with real-world trips. So, the project is fully inspectable but it would be costly.

Computational memory:

An important consideration throughout the project was the restricted amount of available memory resources. The volume and complexity of the dataset posed significant challenges, sometimes foreseen to exceed the memory capacity of our hardware. This called for careful management of memory usage during data processing, analysis, and modeling stages. We implemented strategies such as trip splitting and merging, allowing us to work with smaller subsets of the data at a time,

thereby decreasing memory usage. Additionally, we optimized the code by enhancing loops in the code to ensure that the project could be executed within the constraints of available memory resources.

Computational time:

Throughout the project, time complexity emerged as a critical consideration due to the large-scale dataset and the computational requirements of data processing, analysis, and modeling. With such big data transformations, feature engineering, and model training, it was essential to optimize the project's time complexity to ensure timely execution. We employed various techniques to minimize the computational time required for each step. Also, time management had a crucial role in the manner of computational time because pieces of code had to be executed on time and multiple times before meetings or the continuation of the project.

3. METHODOLOGY

3.1 Understanding the raw data

In this section, we first collected data from Smith Newton medium-duty electric delivery trucks which are used for a company called Frito-Lay North America's (FLNA) logistics in the United States. Collected data converted to python pandas object for further processing. Then, acquired data was analyzed and several features were extracted according to the domain knowledge we got from our instructors. After selecting relevant data, we started to process the data for our needs. One of the key features in our dataset was "BMU_Mode," which indicates whether the vehicle is turned on or off, We specifically focused on the data where the BMU_Mode was 2, indicating that the engine was turned on, and discarded the remaining data. In addition to that, we tackled the timestamp information, which was initially encrypted. We decrypted the timestamps by using the 'to_datetime' method of Python to get a human-readable date format facilitating further analysis. Additionally, we leveraged the existing data to create new features that could provide valuable insights. For example, we generated an acceleration value by utilizing the time and speed features.

This calculation allowed us to derive the rate of change in velocity, offering a measure of acceleration. Moreover, we created a power feature (kW) by combining the current and voltage data, enabling us to examine the power consumption of the vehicle. To gain further insights and enhance the interpretability of the data, we introduced a traffic category feature based on the time feature. By analyzing the timestamps, we categorized the data points into different traffic categories, providing us with a useful contextual variable for analysis. Finally, we ensured coherence by converting certain units to maintain consistency throughout the dataset such as mp/h to km/h. After completing the aforementioned data processing steps and extracting the necessary information, we proceeded to divide the dataset into individual trips. To accomplish this, we employed built-in Python methods that allowed us to separate the data based on the timestamp feature. By observing when the timestamps jumped 10 minutes ahead, we identified instances where the vehicle had stopped momentarily and a new trip had begun Python code we used can be found in Appendix A. Resulting raw data set can be seen in Table 1 below.

Table 1. Description of the data

Variables	*Descriptions*	*Data Types*	*Range*	*Unit*
GPS_Speed	Speed	Quantitative(float)	Min:0 Max:86.90	Km/h
Battery_Voltage_SYS	Voltage	Quantitative(float)	Min:0, Max:358	Volt
Battery_Current_SYS	Current	Quantitative(float)	Min:-400 Max:275	Ampere
RD_Ambient_Temp_degC	Temperature	Quantitative(float)	Min:-44 Max:60	Celsius
GPS_Altitude	Altitude	Quantitative(float)	Min: 0 Max:2932	Feet

GPS_Latitude	Latitude	Quantitative(float)	Min:-1553, Max: 2147	Degrees
GPS_Longitude	*Longitude*	Quantitative(float)	Min:–1324, Max:4221	Degrees
CT_Air_Con_Current_RD	Air conditioner current	Quantitative(float)	Min: 6, Max:3241	Was not Specified
GPS_Altitude_diff	Altitude difference	Quantitative(float)	Min: -100 Max:100	Feet
Acceleration	*Acceleration*	Quantitative(float)	Min: -24, Max:24	m/s2
Power(kW)	Power	Quantitative(float)	Kw, Min: -96 Max:134	KiloWatt
Traffic_Category	Traffic	Categorical	1:Low traffic 2:Medium Traffic 3:High Traffic	None
Trip_ID	Trip number and identifier	String	Min:0, Max:8800000	None

3.2 Processing the raw data

Following the completion of the data processing step, we consolidated all of our trip data into a single dataset by shrinking the data. This shrinkage process involved utilizing the group by method available in Python's DataFrame library. To achieve this, we grouped the rows belonging to the same trip together based on their trip identifiers. Subsequently, we applied aggregation functions

such as average, median, maximum, and minimum to these grouped rows. By doing so, we transformed the individual rows containing speed information into a single row that represented the trip. For example, if we had ten rows of data associated with trip one, each containing speed information, we converted them into a single row by calculating the average of these ten rows. This process ensured that each trip was represented by a single row in the resulting dataset, providing a more concise and condensed representation of the data. By utilizing aggregation functions, we effectively summarized the trip data, capturing key statistical measures such as the average, median, maximum, and minimum values. This approach reduced the overall size of the dataset while retaining essential information related to each trip. By shrinking the data in this manner, we achieved a more streamlined representation that facilitated further analysis and interpretation, enabling us to derive insights from the consolidated information on the trips. In addition to that, we also created new features that we will need during the machine learning model phase. For instance, we estimated the energy consumption of the trip which will be our dependent variable in the machine learning model. Then, we found in which state the observed trip took place using the latitude and longitude coordinates in the United States such as New York, Texas, or Ohio. This feature allowed us to analyze regional patterns and factors that might impact energy consumption or vehicle performance across different states. Then, we incorporated a categorical variable called 'season' that was derived from the trip date we have stored previously. This additional information enabled us to analyze and understand the impact of different seasons on energy consumption. Finally, we created a new feature called kinetic_intensity which is a measure of the aggressiveness of the driver while driving the vehicle. This measure was constructed according to the formulas presented in the article Duty Cycle Characterization and Evaluation Towards Heavy Hybrid Vehicle Applications (O'Keefe et al., 2007). This feature contributed a lot to our model as it detected the driving patterns of the trip.

Detailed Python codes for reducing the data based on trip metrics can be found in Appendix B. Furthermore, the processed dataset, which incorporates all the aforementioned transformations, is presented in Table 2 below.

Table 2. Description of the Main Dataset After Process

Variables	*Descriptions*	*Data Types*	*Range*	*Unit*
Average_Speed_kmh	Average speed	Quantitative (float)	Min:0 Max:68.21	Km/h
Duration_s	Duration of the trip	Quantitative (float)	Min:372 Max:78736	Seconds
Average_Ambient_Temperature	Temperature	Quantitative (float)	Min:-18 Max:58.4	Celsius
GPS_Altitude_Mean	Altitude average	Quantitative (float)	Min: 0 Max:465	Feet
Positive_Altitude-diff_Sum	Altitude difference sum between data	Quantitative (float)	Min:1, Max: 22620	Feet
Season	*Season of the trip*	Categorical	Fall Winter Spring Summer	None
Average_air	Air conditioner's current average	Quantitative (float)	Min: 2477, Max:2704	Was not Specified
GPS_Altitude_diff	Altitude difference	Quantitative (float)	Min: -100 Max:100	Feet

Positive_Acceleration_Mean	*Acceleration*	Quantitative (float)	Min: -24, Max:24	m/s2
Energy_Consumption_kWh	Energy consumption	Quantitative (float)	Min: -96 Max:134	KiloWatt
Traffic_Category_Mean	The traffic range of the day	Quantitative (float)	Min: 1 Max: 3	None
Kinetic Intensity	Kinetic Intensity	Quantitative (float)	Min: 0 Max: 0.5	1/mi
State Name	Name of the state during the trip	Categorical	Washington California new jersey New York Virginia Maryland Illinois	None
Total Distance_km	Distance made during the trip	Quantitative (float)	Min: 5 Max: 74.56	Kilometer

We also conducted several crucial visualizations on this newly created dataset. These visualizations included a correlation matrix, which allowed us to examine the relationships between variables and identify relevant features. Also, histograms, bar plots, and scatterplots were plotted to understand the data better. These visualizations were instrumental in uncovering key trends, distributions, and relationships among different variables. By examining histograms, we could analyze the frequency distribution of specific features, gaining insights into their patterns

and potential outliers. Bar plots allowed us to compare categorical variables, providing a clear visual representation of their distribution across different categories.

The scatter plots facilitated the exploration of relationships between two continuous variables, enabling us to identify any correlations or trends. The correlation matrix of all features is depicted in Figure 1

Figure 1. Correlation Matrix of processed data set

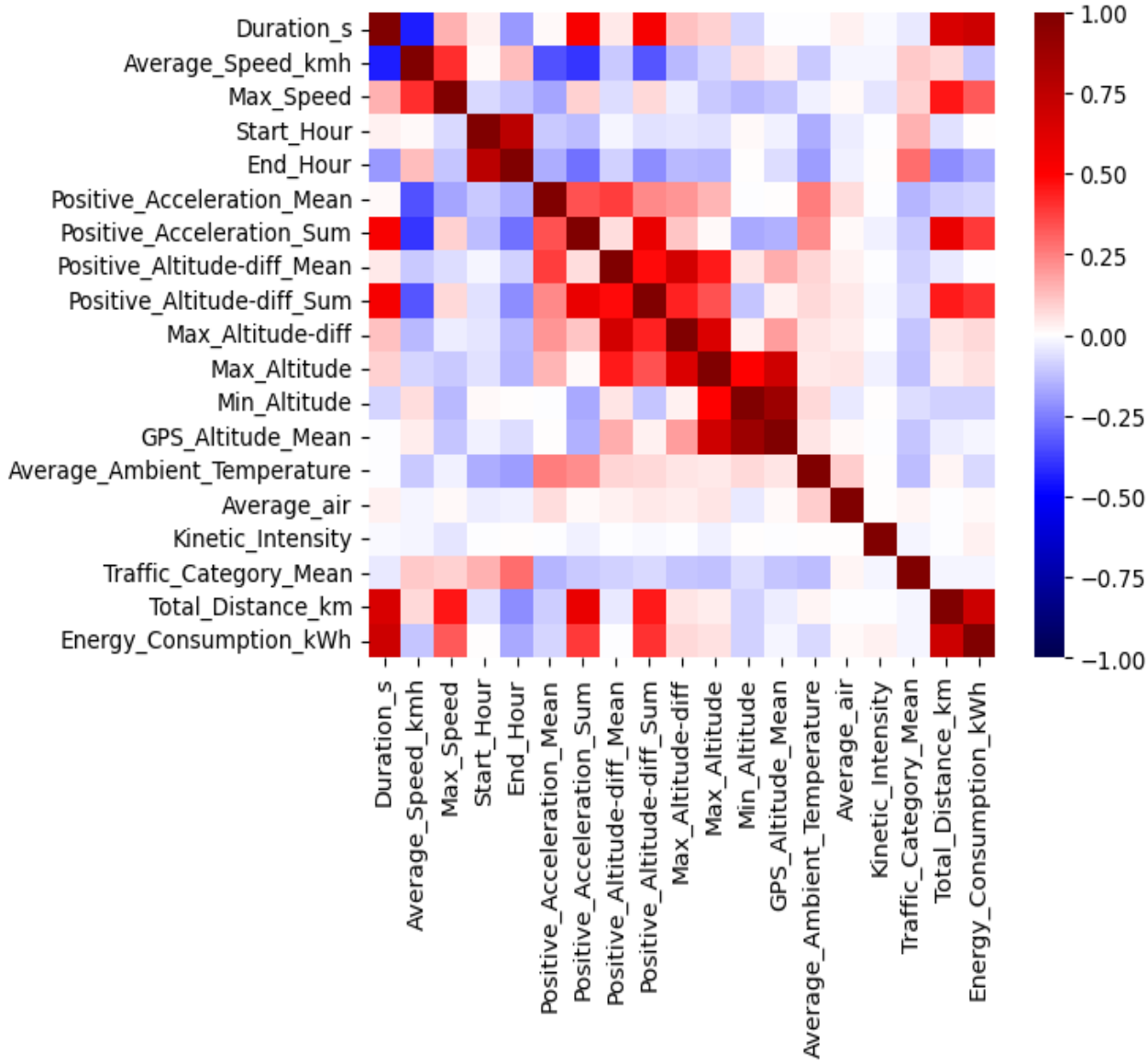


Figure 1 seems logical in terms of technical parts of our data as we see that most of the features that are related to energy consumption such as total distance, duration, and average speed seem to have a higher relationship compared to our other features such as average air and altitude diff mean. We also analyzed the distributions of our data in terms of categorical variables such as season and state names, the results of our analysis can be seen in Figure 2.

Figure 2. Distributions of our data by categorical variables

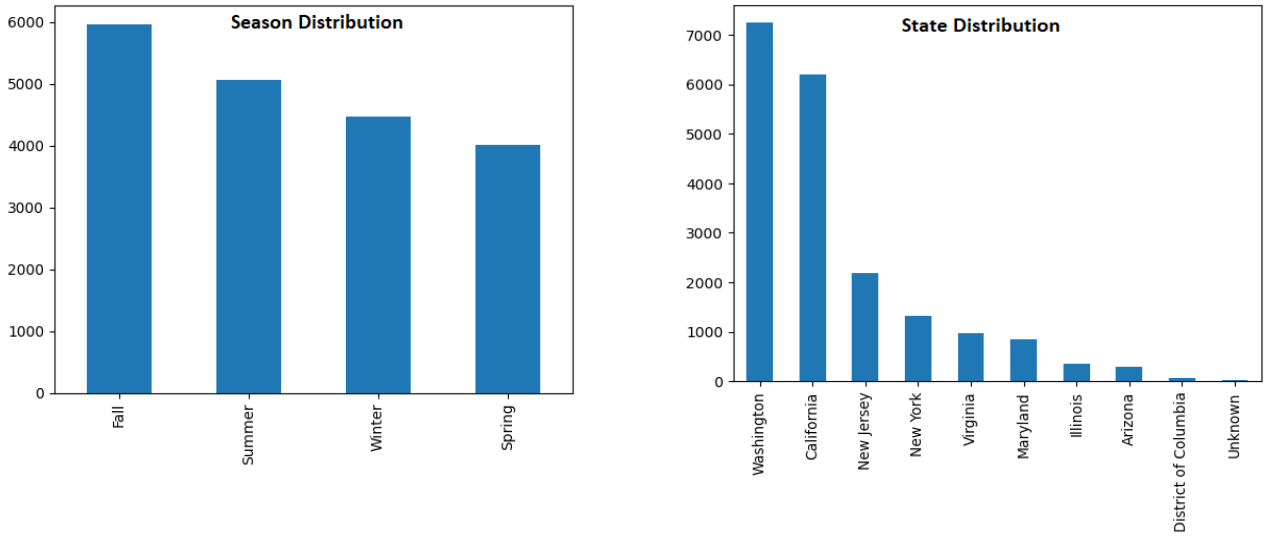
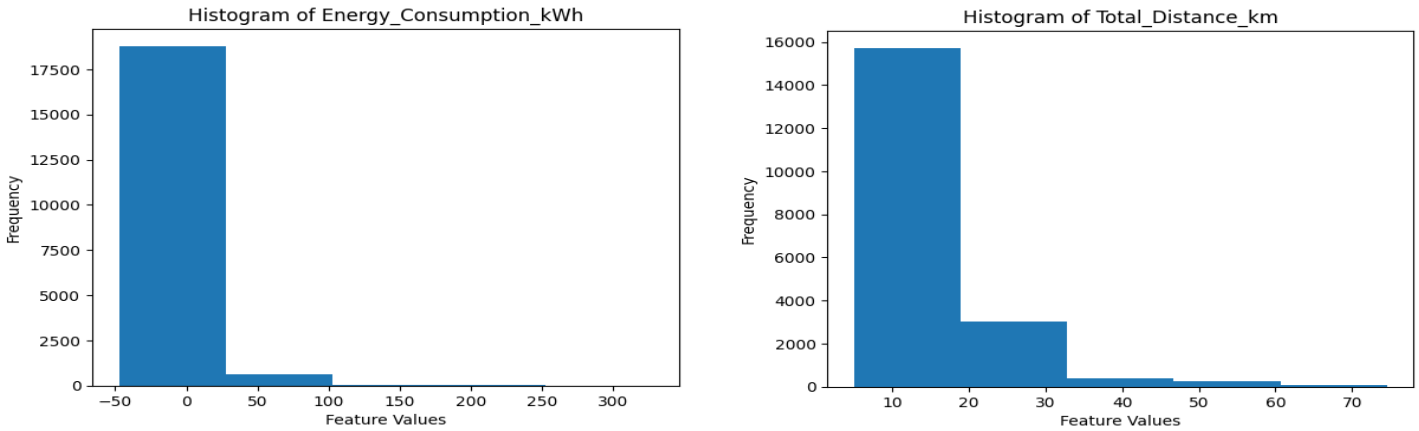


Figure 2 explains the overall distribution of our data of origin so that we can interpret our results accordingly. For instance, it seems that most of our data are collected from Washington and California which means that we may have a greater chance of predicting energy consumption accurately in these cities as our models see more data. We can also say that season distribution is pretty even so we can predict with equal accuracy for different seasons.

We also plotted the total distance taken in trip and total energy consumption histograms to see the distribution more accurately. Figure 3 shows the results.

Figure 3. Histogram showing total distance and Energy Consumption



In addition to the distribution visualizations, we also plotted scatter plots. You can see the two important plots in Figure 4 and 5

Figure 4. Scatter plot of Temperature and Energy Consumption

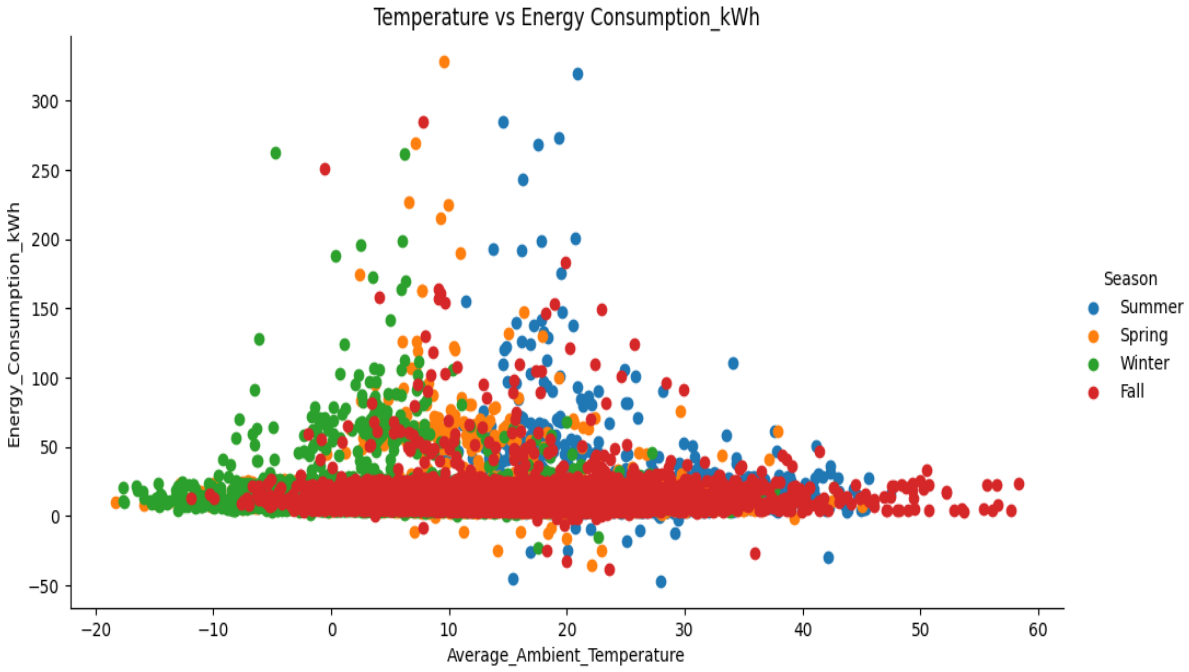
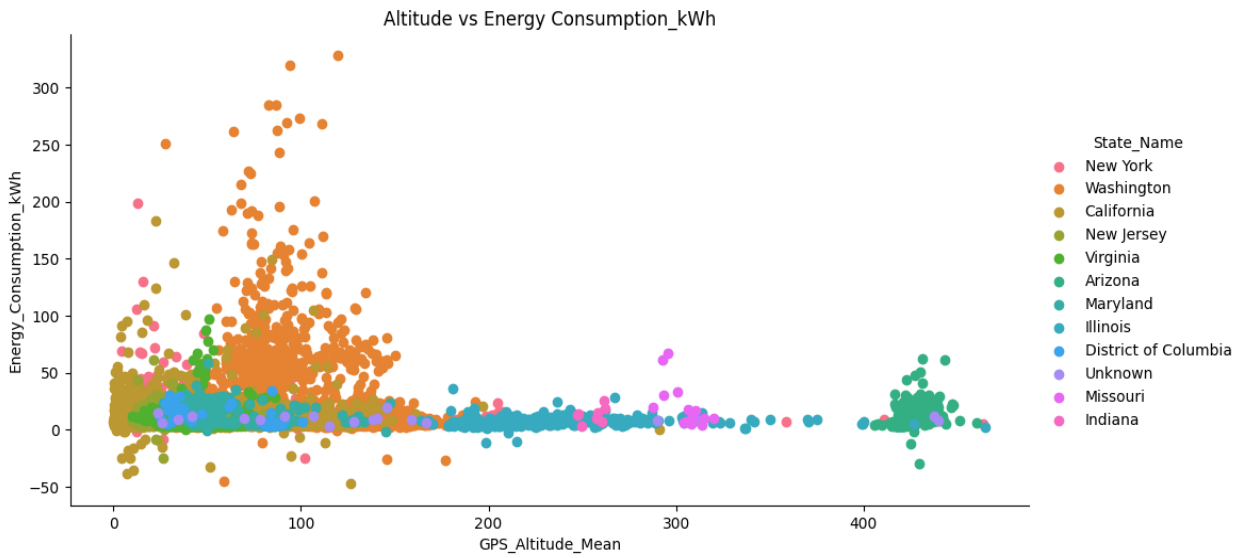


Figure 5. Scatter Plot of Altitude and Energy Consumption



The scatter plots have proven to be instrumental in gaining valuable insights into the distribution patterns of features, such as temperature and altitude, across various seasons and states. These visualizations have allowed us to analyze the intricate relationships between these factors and energy consumption more comprehensively. By analyzing these scatter plots, we were able to adjust the data and identify outliers more efficiently. This analysis helped us gain a deeper understanding of the data and its patterns.

3.3 Machine learning methods

After having the data set ready and the analysis made, we started applying machine learning models. Firstly we split our data into test and train and stratified it with state and season variables to make sure states and seasons are equally distributed. Then we trained different models with the training data using stratified kfold and cross-validation scores. We used different models in the training section such as XGBoost (Chen & Guestrin, 2016), RandomForest (*Ho, T. K., 1995)*, ExtraTrees (Geurts, Ernst, & Wehenkel, 2006), and LightGBM (Ke et al., 2017) to predict the energy consumption of the test data. To measure the performance of our models we developed a custom metric that results in 1 if our prediction is near the 10% error margin and result 0 if our

error is more than 10%. In addition, we also used the mean squared error metric to see our results with different metrics to measure the performance with higher detail. To get better scores we hyperparameter-tuned these models by determining the best parameters such as n_jobs and n_estimators.

4. RESULTS & DISCUSSION

The results of the projects were as good as we anticipated. We have nearly achieved the all milestones of the project and all of the initial objectives were realized. To elaborate it further, the raw data we have at the start of the project was complex and contained too much noise that made the data hard to understand. We analyzed and processed it to create clean and ready data for further analysis using various analysis tools. Our second objective was to implement high-end machine learning tools to process data to predict the energy consumption of the vehicle accurately. To realize that we used XGBoost (Chen & Guestrin, 2016), RandomForest (*Ho, T. K., 1995)*, ExtraTrees (Geurts, Ernst, & Wehenkel, 2006), and LightGBM (Ke et al., 2017) algorithms and tuned them with specific parameters to achieve higher accuracy. Among these models, we have decided to move forward with ExtraTreesRegressor as it gave the best results out of all models we have tried. We have calculated results with two target variables, energy consumption and energy consumption per km. You can see the results of different models for different dependent variables in Tables 3 and 4.

Table 3. Results of different models for energy consumption

Model	*Accuracy(%10 error)*	*Accuracy(%15 error)*	*MSE*
ExtraTrees	0.763	0.863	54.72
XGBoost	0.697	0.833	61.91
RandomForest	0.739	0.854	55.57
Baseline Model (Mean)	0.111	0.17	-

Table 4. Results of different models for energy consumption per km

Model	*Accuracy(%10 error)*	*Accuracy(%15 error)*	*MSE*
ExtraTrees	0.763	0.867	0.16
XGBoost	0.720	0.848	0.171
RandomForest	0.737	0.855	0.162
Baseline Model (Mean)	0.396	0.542	-

Table 5. Baseline accuracies for every state

State	*Baseline accuracy for specified state*	*Baseline Accuracy for all states*
New York	0.276	0.178
Washington	0.389	0.373
California	0.346	0.407
New Jersey	0.603	0.407
Virginia	0.555	0.396
Arizona	0.536	0.187
Maryland	0.553	0.397
Illinois	0.180	0.408

District of Columbia	0.4	0.391
Missiori	0.5	0.391

As it can be seen from our results ExtraTreesClassifier outperforms other models and we see a drastic increase in our accuracy when we increase our margin of error by 5%. In addition, we confidently assert that our model consistently outperforms the baseline model in all scenarios, showcasing superior accuracy, particularly when predicting total energy consumption per km. To delve deeper into this claim, our model achieves an impressive accuracy of 0.76, while the baseline model lags with a mere 0.46 accuracy for energy consumption per kilometer. This substantial difference underscores the significant advancements and improved precision offered by our model in the field of energy consumption prediction. Upon examining Table 5, we can deduce that certain states, such as New Jersey, exhibit a higher degree of predictability, whereas others, like New York, prove to be more challenging in terms of prediction accuracy. Notably, Table 5 also provides insights into the accuracies of state-specific baseline models had we employed them to predict the entire dataset.

We also plotted the feature importance for our best-performing models to see which features contributed the most and least. We used two different metrics to predict which are energy consumption per km and energy consumption. Investigating these two metrics helped us to find different patterns across the trips an their energy consumption. These two plots can be found in Figures 5 and 6.

Figure 5. Feature Importances for energy consumption per km

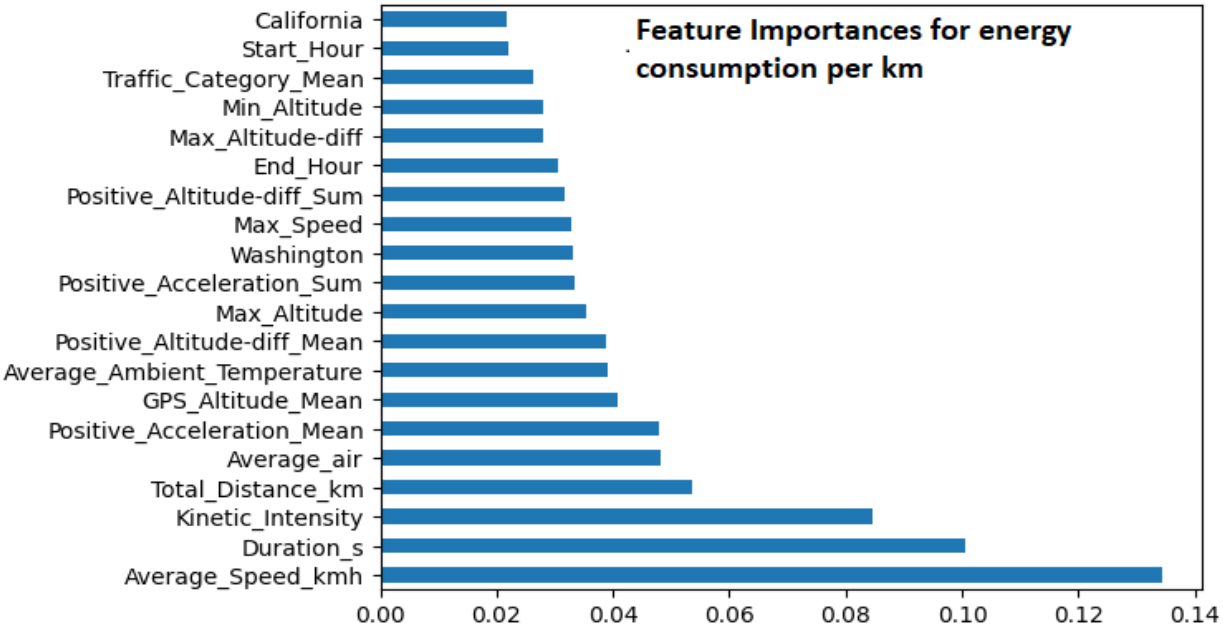
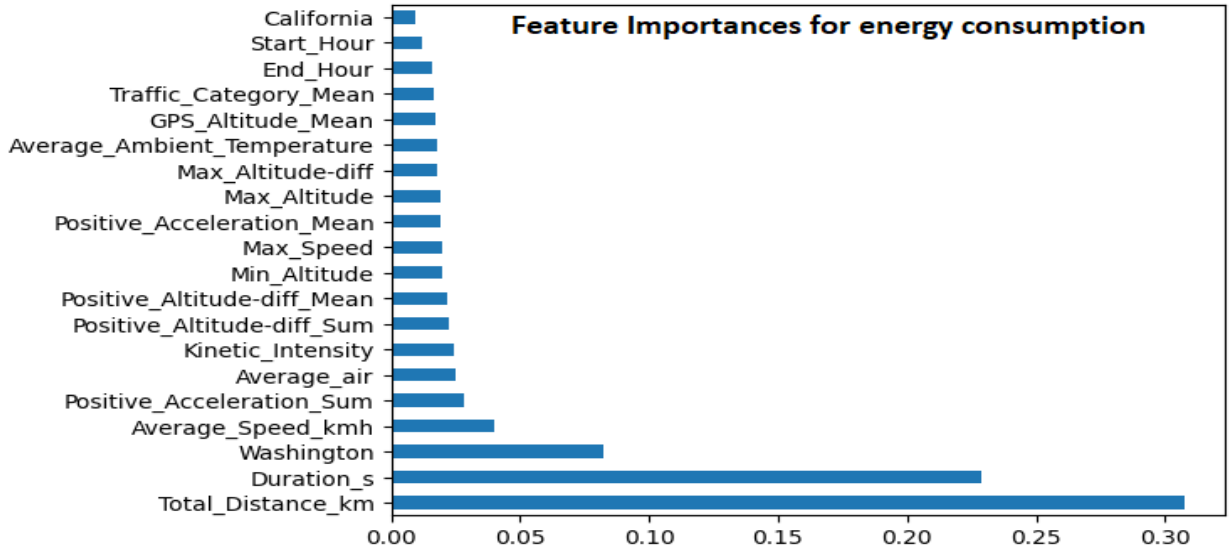


Figure 6. Feature Importances for energy consumption



These plots seem logical to us as they interpret the features according to our correlation matrix values. Also, using these plots helped us to determine the factors that are important for energy consumption apart from the common features such as distance and speed. We see that location of the trip, as well as ambient temperature and air conditioner level, is also important for predicting the energy consumption levels of electric vehicles. In light of our results, we can claim

that our initial objectives were realized in the end as we got a pretty high score of 76% accuracy in our best result for predicting the energy consumption of electric vehicles.

We believe that this project is completed as we achieved the initial objectives without changing them. Also, we met the deadlines that are specified at the start of the project without delaying the milestone steps. The results of this project contributed to the previous state of the art. Our project was based on the *Field Evaluation of Medium-Duty Plug-in Electric Delivery Trucks(Prohaska R.,2016)* in which they only analyzed, visualized, and interpreted the data without doing extensive data processing and modeling. We took the same data and processed it into a format that is easy to understand by machine learning models. Then we applied models to predict the energy consumption of these electrical trucks during their trips using other factors such as temperature, altitude, and speed. As a result, our contribution improved the perception of the energy consumption of electric vehicles.

5. IMPACT

Scientific Impact:

The findings from this project significantly contribute to the academic literature on electric vehicles, specifically medium-duty electric delivery trucks. By providing a comprehensive understanding of energy consumption patterns, driving profiles, and key factors influencing efficiency and energy consumption, the project enriches the related scientific knowledge in the field. The project doesn't stop at theory; it applies real-world data, turning it into actionable insights that fill the divide between industry practice and academic research. This approach might lead to a deeper, more practical understanding of how these electric trucks operate and how they can be optimized in different circumstances in terms of their energy consumption, in the research community. Moreover, this project helps in the growth of efficient and eco-friendly logistics solutions, pushing forward our understanding of sustainable transportation as it provides both the scientific community and the logistics industry with a lot of important data and insights. This knowledge can be used to better the way things are done, lessen the harm to the environment, and make smarter driving profiles. As a result, the project makes a big impact in the field of science,

as it adds to our shared knowledge of electric vehicles with their driving patterns and is a key player in the move toward a more sustainable future.

Technological Impact:

The application of machine learning algorithms to predict energy consumption patterns is an outstanding technological achievement. The innovative use of data analysis and visualization techniques for detecting energy consumption patterns again underlines this project's technological significance. This includes variations in energy usage based on speed, acceleration, braking, and the overall drive cycle which have successfully been interpreted through the implementation of these algorithms and techniques. The ability of the machine learning models to extract and learn from these complex patterns underscores their profound utility. This allows for a more precise prediction and understanding of energy usage in electric vehicles, and potentially a more effective and efficient use of energy. In the future, the technological techniques utilized in the project can be applied to optimize energy consumption, leading to improved performance of electric delivery trucks and potentially other electric vehicles as well.

Socio-Economic Impact:

The optimization of electric delivery truck performance based on the data feed can lead to significant cost savings for logistics companies like Frito-Lay North America. Reduced fuel costs and maintenance expenses can directly be passed onto consumers, potentially leading to more affordable and environment-friendly goods and services. Furthermore, as the appeal of sustainable transportation alternatives continues to grow, this project plays a crucial role by shedding light on the viability and benefits of electric vehicles. Modeling and predicting energy consumption patterns of electric delivery trucks, helps companies make more informed decisions, potentially accelerating the shift towards greener transport options. As the companies adopt these eco-friendly, green solutions, we see a decrease in greenhouse gas emissions and air pollution in the environment, both of which are important to combat climate change.

Innovative and Commercial/Entrepreneurial Aspects:

The findings from this project have significant commercial potential. Improved efficiency of electric delivery trucks can give logistics companies a competitive edge. Moreover, the data-driven practical insights found by the project have the potential to drive improvements and inventions in the electric vehicle industry. By revealing the key factors that affect the performance and energy consumption of electric delivery trucks, companies are given valuable information that can guide the development and improvement of their products. This goes beyond just making adjustments to existing models; it could lead to the design of entirely new vehicles specifically created to meet the unique requirements and challenges of the logistics sector, and even create a ripple effect of new business opportunities within the electric vehicle industry. For example, data analysis and machine learning models that specialize in predicting energy consumption patterns could create new innovative ideas for new startups. In addition, manufacturers offering solutions for improving driving behavior or managing the charging strategies for electric delivery trucks might lead to an increased demand for their services. These potential commercial applications underscore the far-reaching implications of this project and its potential to stimulate growth within the industry.

Freedom-to-Use (FTU) issues:

As of the current state, there are no Freedom-to-Use issues in the project. The data and research papers used in this project have been open-sourced and properly cited, adhering to all the necessary ethical guidelines and regulations. In a similar sense, the machine learning algorithms used for data analysis and predictive modeling are grounded on widely accepted open-source libraries that are broadly recognized and utilized by the software development community, which keeps us away from any potential licensing conflicts. Also, while we presently foresee no Freedom-to-Use issues in our project, we need to recognize that as the project evolves, new data, tools, methods, or algorithms may be incorporated. In such situations, ongoing evaluation of any potential Freedom-to-Use issues will be an integral part of the project's future development. We strongly believe that the commitment to ethical and legal compliance not only safeguards our project but also adds credibility and reliability.

6. ETHICAL ISSUES

Ethical considerations surrounding the project can be approached from a few different perspectives. Firstly, there are no copyright issues because tools used in the project such as Python programming language, data science libraries of it, and integrated development environments are publicly available. Also, the data is openly accessible online as well so there is no violation of private or sensitive information. Secondly, due to the nature of it, the project does not affect any individual directly so it cannot risk anybody's health or safety. Thirdly, the model developed as a part of the project does not guarantee or claim anything. It does not claim to give service. So, using this model and taking it as a reference is the user's own risk. It is essential to emphasize that the model should be used as a reference rather than a definitive source of information. Additionally, likewise, the project does not harm any individual's life, and it does not have detrimental effects on the environment in a bad way at all. Finally, it does not violate any kind of law or regulation. Hence, the project demonstrates a commitment to responsible research practices and upholding legal and ethical standards so there is no problem with the ethical aspect of the project throughout its execution.

7. PROJECT MANAGEMENT

As stated in the second section's "objectives/tasks" subsection, at the beginning of the project, the whole process starting from the very first day until the end of the project planned as follows:

- Researching previous findings beforehand for the implementation of a solution
- Examining the set of data
- Data extraction
- Data visualization
- Data analysis
- Developing a machine learning model which estimates energy consumption with certain constraints

Since the project is a comprehensive project which requires knowledge of fields such as physics, statistics, machine learning, and data science, learning by doing and researching was always an

essential concept for that project. Moreover, the project plan consists of multi-steps which might be addressed nonlinearly more than once due to the nature of the project. Depending on the outcome of milestones, data might have been altered to use it in future steps more efficiently and correctly. Thus, from time to time, going over steps more than once was a must.

Considering all the factors listed in the paragraph above, the initial plan of the project significantly changed through the development process. Sometimes, we had to take a step back to the previous stage. So, we decided to be more flexible in the plan. Furthermore, we have modified our dataset more than once before using it in the model. Last but not least, occasionally, we make our decisions based on our academic research rather than solely relying on our dataset.

During our graduation project, we learned many things about conducting a scientific project, reporting a project, working as a group, working together and being advised by supervisors, and much more. It is a long-lasting project which requires a well-designed plan to complete milestones and the project. So, time management was the key. Since it is a scientific project, some procedures must be followed, and a formal way of reporting the project. We can say we have learned an academic approach to a project while reporting our graduation project. Also, working as a group as well as working with our supervisors taught us how to work cooperatively and in an organized way. Working with our supervisors was good for us to internalize subordinate relationships.

Other than soft skills, we have learned many techniques and computer science-related skills. The most important learning outcome was the importance of time and space complexities. Using these resources efficiently might be the only option especially when the data is big. Other than that, we have developed our data science and machine learning knowledge.

8. CONCLUSION AND FUTURE WORK

The most important result of our project is the improvement of our prediction over the baseline model. The baseline model predicts the energy consumption of the vehicle with %40 accuracy while our prediction model using extra trees classifier predicts with 76% accuracy which nearly the double amount of accuracy over baseline's. Our accuracy result could be much higher if we had more time for this study as most of the parts of our project required deep research about both electrical vehicles and the implementation of Python code. Also, the complexity of the data limited us as it took a very long time to process and create something meaningful out of it. Furthermore, since we had huge chunks of data our hardware could not handle the all amount of data thus we

had to split the data into smaller parts and merge them after we shrink their sizes. This resulted in a huge amount of time lost as every time we had to run the code it took several hours. Apart from technical limitations, our conclusion is also limited in way that if a user wants to predict the energy consumption of the trip, he/she has to know several features such as altitude and acceleration of the vehicle which are sometimes hard to know. Another important limitation of our project is that we exclusively relied on data collected from Smith electric vehicles. Consequently, our model was specifically designed and trained to cater to this particular type of vehicle. As a result, it may encounter challenges when attempting to predict energy consumption for other electric vehicle models, as they may have different characteristics and specifications. To elaborate further, Electric vehicles show significant differences in terms of design, battery technologies, energy management systems, and driving behaviors, all of which influence energy consumption patterns. Therefore, since our model lacks exposure to data from diverse electric vehicle models, its ability to generalize predictions to different vehicles may be compromised.

For the next steps of the project, firstly we can expose our model with more data collected from different electric vehicles around the world which may help our model to generalize the problem better thus it can provide more accurate predictions for energy consumption. Secondly, we can try different machine learning models such as deep learning to further unveil the connections between independent variables to dependent variables. Lastly, when we improve the project we can present the detailed findings of our project to academia to get world recognition for our studies.

If we continue to work on the project, we believe that we can open up to the world. Firstly, when we improve the project after working more, we can put the resulting model into production by uploading it to our website. So that people from all around the world can use it and we can make a profit out of it. Then, we could talk with companies that use electric vehicles and make a deal with them to use our model to optimize their operations. We can then work more efficiently and improve our model to become a startup to sell our product across the world.

9. APPENDIX

Github link:

https://github.com/ydelipinar/Data_Based_Vehicle_Energy_Consumption_Modeling

A. Code used to process the data and incorporate it

```
df_list = []
STARTFOLDER = 25
ENDFOLDER = 30
#we have 30 folders
vehicle_folder_counter = 0
removed_file_counter = 0

# Iterate over the directories and files in the path_prefix
for path, currentDirectory, files in os.walk(path_prefix):
    # Iterate over the files in the current directory
    if(STARTFOLDER > vehicle_folder_counter):
        if(vehicle_folder_counter > 0):
            print(os.path.basename(path), "skipped")
        vehicle_folder_counter+=1
        continue

    elif(vehicle_folder_counter > ENDFOLDER):
        break

    print("Reading", os.path.basename(path) + "... ", end="")

    for file in files:
        if file.endswith('.mat'):
```

```
try:
    mat = loadmat(join(path, file))
except IOError:
    print(f"\nError reading file: {join(path, file)}")
    continue

mat = {k: v for k, v in mat.items() if k[0] != '_'}
df_temp = pd.DataFrame({k: np.array(v).flatten() for k, v in mat.items()})
# selecting columns to keep
df_temp = df_temp[cols_to_keep]
# Large NaN Detection
df_files_with_large_nan = df_temp[df_temp['BMU_Mode_SYS'] == 2.0]

if(df_files_with_large_nan['RD_Ambient_Temp_degC'].isnull().mean() >= 0.50 or
df_files_with_large_nan['GPS_Speed'].isnull().mean() >= 0.50 or
    df_files_with_large_nan['Battery_Voltage_SYS'].isnull().mean() >= 0.50 or
df_files_with_large_nan['Battery_Current_SYS'].isnull().mean() >= 0.50 or
    df_files_with_large_nan['GPS_Altitude'].isnull().mean()>= 0.50):
    removed_file_counter += 1
else:
    if (df_temp[df_temp['BMU_Mode_SYS'] == 2.0].shape[0] < 5):
        continue
    else:

        df_temp['Timestamp'] = pd.to_datetime(df_temp['Timestamp'], unit='s')
        df_temp['Seconds difference'] = df_temp['Timestamp'].diff().dt.total_seconds()
        df_temp['GPS_Altitude_diff'] = df_temp['GPS_Altitude'].diff()
        # df_temp.at[0, "GPS_Altitude_diff"] = 0

        # Calculate the cumulative sum of the time differences
        df_temp['Total Time(s)'] = df_temp['Seconds difference'].cumsum()
```

```
        # df_temp.at[0, "Total Time(s)"] = 0

        # Convert GPS speed from mph to km/h
        df_temp['GPS_Speed'] = df_temp['GPS_Speed'] * 1.60934

        # Calculate the speed difference between consecutive timestamps
        # Calculate acceleration (change in speed / change in time = m/s^2)
        df_temp['Acceleration'] = (df_temp['GPS_Speed'] / 3.6).diff() / df_temp['Seconds difference']

        # Calculate power
        df_temp['Power(kW)'] = (df_temp['Battery_Voltage_SYS'] * df_temp['Battery_Current_SYS'] * -1) / 1000

        # Select BMU
        df_temp = df_temp[df_temp['BMU_Mode_SYS'] == 2.0]

        # Assign traffic categories to a new column 'Traffic_Category'
        df_temp['Traffic_Category'] = df_temp['Timestamp'].dt.hour.apply(
            lambda hour: 1 if 0 <= hour < 5 # Early Morning (12AM - 5AM): Low traffic
            else 3 if 5 <= hour < 10       # Morning Rush Hour (5AM - 10AM): High traffic
            else 2 if 10 <= hour < 15      # Midday (10AM - 3PM): Moderate traffic
            else 3 if 15 <= hour < 20      # Evening Rush Hour (3PM - 8PM): High traffic
            else 1)                        # Late Evening (8PM - 12AM): Low traffic

        # Partition the data to trips by using Total Time(s)
        # Assign trip numbers based on the difference in 'Total Time(s)'
        df_temp['Trip_diff'] = df_temp["Total Time(s)"].diff()
        df_temp["Trip"] = (df_temp['Trip_diff'] > 1000).cumsum() + 1
```

```
        # Create a 'Trip_Seconds_Counter' column without adding intermediate columns to the DataFrame
        df_temp['Trip_Seconds_Counter'] = (
          df_temp['Trip'].ne(df_temp['Trip'].shift())
          .cumsum()
          .groupby(df_temp['Trip'])
          .cumcount()
        )

        # Finding the county name corresponding to the given average latitude and longitude values
        # df_temp['City_Name'] = get_city(df_temp['GPS_Latitude'].mean(), df_temp['GPS_Longitude'].mean())

        # DROPPING ANORMALIES First
        df_temp.drop(df_temp[df_temp['GPS_Speed'] >= 55].index, inplace=True)
        df_temp.drop(df_temp[df_temp['Battery_Voltage_SYS'] >= 360].index, inplace=True)
        df_temp.drop(df_temp[df_temp['Power(kW)'] >= 88].index, inplace=True)
        df_temp.drop(df_temp[df_temp['RD_Ambient_Temp_degC'] >= 36].index, inplace=True)
        df_temp.drop(df_temp[df_temp['RD_Ambient_Temp_degC'] < -15].index, inplace=True)

        # Create the vehicle_name extracting from path
        # Insert the 'ID' and 'File_Name' columns
        df_temp["ID"] = os.path.basename(path) + '_' + file[file.rfind("_")+1:file.rfind(".")] + "_trip_" + df_temp["Trip"].astype(str)
        df_temp["File_Name"] = os.path.basename(path) + '_' + file

        # Drop the 'Unnecessary' column
        df_temp = df_temp.dropna(subset=['Power(kW)'])
        df_temp = df_temp.drop(['Trip_diff'], axis=1)
```

```
            df_temp = df_temp.drop(['Seconds difference'], axis=1)
            df_temp = df_temp.drop(['BMU_Mode_SYS'], axis=1)
            df_temp = df_temp.drop(['Trip'], axis=1)
            df_temp = df_temp.drop(['Total Time(s)'], axis=1)
            df_temp = df_temp.drop(['File_Name'], axis=1)

            df_list.append(df_temp)
            # print(file,u'\u2713')
    print(u'\u2713')
    vehicle_folder_counter += 1
# Merge all dataframes in the list
print(removed_file_counter, 'files removed due to high NaN ratio')
print("Merging the files... ")
df_merged = pd.concat(df_list)
print("Files have been merged", u'\u2713')
```

B. Code used to shrink the data by groping them by trips

```
city_dict = {}
def calculate_trip_metrics(df):

    #latitude = df['GPS_Latitude'].median().round(3)
     longtitude = df['GPS_Longitude'].median().round(3)
     coordinate = str(latitude)+'-'+str(longtitude)
     if(coordinate in city_dict):
        city = city_dict[coordinate]
     else:
        city = get_city(latitude, longtitude)
        city_dict[coordinate] = city
```

```
energy_consumption = np.trapz(y=df["Power(kW)"], x=df["Trip_Seconds_Counter"])
energy_consumption_kwh = energy_consumption / 3600
avg_temperature = df["RD_Ambient_Temp_degC"].mean()
avg_speed_kmh = df["GPS_Speed"].mean(skipna=True)

avg_traffic_category = df["Traffic_Category"].mean()
avg_gps_altitude = df["GPS_Altitude"].mean()
avg_air = df["CT_Air_Con_Current_RD"].mean()

# Calculate separate means of positive and negative acceleration values, all of them in m/s^2
positive_acceleration_mean = df[df["Acceleration"] > 0]["Acceleration"].mean()

positive_acceleration_sum = df[df["Acceleration"] > 0]["Acceleration"].sum()
positive_altitude_diff_mean = df[df["GPS_Altitude_diff"] > 0]["GPS_Altitude_diff"].mean()
positive_altitude_diff_sum = df[df["GPS_Altitude_diff"] > 0]["GPS_Altitude_diff"].sum()

# Get start time and end time for each trip
start_time = df["Timestamp"].iloc[0]
end_time = df["Timestamp"].iloc[-1]

# Calculate max and min speed for each trip
max_speed = df["GPS_Speed"].max()
min_speed = df["GPS_Speed"].min()

# max_altitude_diff = gps_altitude_diff.max()
max_altitude_diff = df["GPS_Altitude_diff"].max()
max_altitude = df["GPS_Altitude"].max()

# min_altitude_diff = gps_altitude_diff.min()
```

```
    min_altitude_diff = df["GPS_Altitude_diff"].min()
    min_altitude = df["GPS_Altitude"].min()

    # Calculation of total distance in km
    duration = df["Trip_Seconds_Counter"].max()

    # gps_speed_kms = df["GPS_Speed"] / 3600
    # total_distance_km = np.trapz(y=gps_speed_kms, x=df["Trip_Seconds_Counter"])
    total_distance_km = (avg_speed_kmh / 3600) * duration
    total_distance_mph = total_distance_km / 1.60934

    df['Speed_mph'] = df["GPS_Speed"] / (1.60934)

    df["Trip_Seconds_Counter"] = df["Trip_Seconds_Counter"] / 3600

    characteristic_Acc = ( (df["Speed_mph"].shift(-1)**2  - df["Speed_mph"]**2) *0.5 + (9.8 * (df["GPS_Altitude"].shift(-1)-df["GPS_Altitude"]))).sum() / total_distance_mph

    aero = ((( df["Speed_mph"].shift(-1)**3 + df["Speed_mph"].shift(-1)**2 * df["Speed_mph"] + df["Speed_mph"].shift(-1) * df["Speed_mph"]**2 + df["Speed_mph"]**3 ) / 4 )
          *        (df["Trip_Seconds_Counter"].shift(-1)-df["Trip_Seconds_Counter"])).sum()       / total_distance_mph

    # Return the calculated metrics for the trip
    return {
        'Energy_Consumption_kWh': energy_consumption_kwh,
        'Start_Time': start_time,
        'End_Time': end_time,
        'Duration_s': duration,
        'Average_Speed_kmh' : avg_speed_kmh if not math.isnan(avg_speed_kmh) else 0,
        'Total_Distance_km': total_distance_km,
```

```
        # 'City_Name': city,

        'Max_Speed': max_speed,
        'Min_Speed': min_speed,

        'Positive_Acceleration_Mean': positive_acceleration_mean,
        'Positive_Acceleration_Sum': positive_acceleration_sum,

        'Positive_Altitude-diff_Mean': positive_altitude_diff_mean,
        'Positive_Altitude-diff_Sum': positive_altitude_diff_sum,

        'Max_Altitude-diff': max_altitude_diff,
        'Min_Altitude-diff': min_altitude_diff,
        'Max_Altitude': max_altitude,
        'Min_Altitude': min_altitude,
        'GPS_Altitude_Mean':avg_gps_altitude,

        'Average_Ambient_Temperature': avg_temperature,
        'Average_air': avg_air,
        'Average_Ambient_Temperature': avg_temperature,
        'Characteristic_Acceleration': characteristic_Acc,
        'Kinetic_Intensity': characteristic_Acc / aero,
        'Traffic_Category_Mean': avg_traffic_category
    }
# Group the DataFrame by 'ID', apply the custom function, and store the result in a Series
trip_metrics = df_merged.groupby('ID').apply(calculate_trip_metrics)
trip_metrics_df = trip_metrics.apply(pd.Series).reset_index()
```

10. REFERENCES

Burton, J., Walkowicz, K., Sindler, P., and Duran, A., "In-Use and Vehicle Dynamometer Evaluation and Comparison of Class 7 Hybrid Electric and Conventional Diesel Delivery Trucks," SAE Int. J. Commer. Veh. 6(2):2013, doi: 10.4271/2013-01-2468.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System (Version 1.4.1) [Software]. Retrieved from https://xgboost.readthedocs.io/

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. Machine Learning, 63(1), 3-42.

Ho, T. K. (1995). Random decision forests. In Proceedings of 3rd international conference on document analysis and recognition (Vol. 1, pp. 278–282).

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., & Ye, Q. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree (Version 3.4.1) [Software]. Retrieved from https://lightgbm.readthedocs.io/

Lee, J., Kim, S., & Choi, Y. (2022). Machine Learning in Electric Vehicle Energy Consumption: A Comprehensive Review. Journal of Electrical Engineering & Technology.

Morrissey, P., Weldon, P., & O'Mahony, M. (2016). Future Standard and Fast Charging Infrastructure Planning: An Analysis of Electric Vehicle Charging Behaviour. Energy Policy, 89 https://doi.org/10.1016/j.enpol.2015.11.008

O'Keefe, M., Simpson, A., Kelly, K., & Pedersen, D. S. (2007). Duty Cycle Characterization and Evaluation Towards Heavy Hybrid Vehicle Applications. In SAE technical paper series. https://doi.org/10.4271/2007-01-0302

Pan, Y., Fang, W., & Zhang, W. (2023). Development of an energy consumption prediction model for battery electric vehicles in real-world driving: A combined approach of short-trip segment division and deep learning. Journal of Cleaner Production, 400, 136742. https://doi.org/10.1016/j.jclepro.2023.136742

Prohaska, R., Simpson, M., Ragatz, A., Kelly, K., Smith, K., & Walkowicz, K. (2016). Field Evaluation of Medium-Duty Plug-in Electric Delivery Trucks. OSTI OAI (U.S. Department of Energy Office of Scientific and Technical Information). https://doi.org/10.2172/1337010