UNIVERSITY of STIRLING

Division of Computing Science and Mathematics
Faculty of Natural Sciences
University of Stirling

Predicting the Impact of Road Conditions on Electric Vehicle Usage Using AI Algorithms

Author: Michael Abebaw Bogale

Dissertation submitted in partial fulfilment for the degree of
Master of Science in Artificial Intelligence

September 2023

Abstract

Problem:

The evolution of Electric Vehicles (EVs) has revolutionized sustainable transport, by reducing the carbon footprint and offering an environmentally friendly alternative to combustion engines. However, the adoption of EVs, despite their potential, has been affected by various challenges. One concern, embedded deeply within the EV discourse, is 'range anxiety'. This term encapsulates the unease experienced by EV users about battery depletion before reaching their destination or the next available charging station.

Moreover, as EVs venture into diverse routes, the impact of varying road conditions on battery consumption becomes an intricate variable in this equation. Precisely, understanding how specific road conditions influence EV battery discharge rates is pivotal. Variabilities in ambient temperature, driving speed and other environmental factors play a significant role in dictating the effective range of an EV. Thus, there's a dire need to comprehensively examine and predict the implications of these road conditions on EV usage to alleviate the anxiety and provide a seamless driving experience.

Objectives:

To alleviate the concerns of 'range anxiety', this dissertation endeavours to:

1. Formulate an optimization methodology, specifically tailored for the Tesla Model 3, to mitigate range anxiety.
2. Understand and quantify the effect of specific road conditions on the EV's battery discharge rate specifically tailored for BMW i3.
3. Create a harmonious blueprint between theory and application by integrating real-world data for more practical outcomes.

Methodology:

Genetic Algorithm Optimization: Chapter 3 details an approach that uses genetic algorithms to minimize travel duration by considering charging locations, intervals, and durations for the Tesla Model 3.

Data Synthesis and Predictive Analysis: Chapter 4 examines the incorporation of empirical data through Google's API and crafting a predictive machine learning model. The model, constructed using a refined dataset from 72 distinct trips, forecasts the state of charge based on factors like ambient temperature, speed, and distance.

Achievements:

Comprehensive Model: A model has been constructed which, while considering variables like driving time and charging time, furnishes EV drivers with an efficient and effective travel strategy.

Data-Driven Insights: By merging our theoretical algorithms with realistic, real-world data from Google's API, the research not only maps out the journey but pinpoints strategic charging locations, thus maximizing efficiency.

Mitigating Range Anxiety: Through the blending of advanced computational techniques, empirical data, and analysis, this dissertation offers a solution to address the prevalent concern of range anxiety among EV users.

Attestation

I understand the nature of plagiarism, and I am aware of the University's policy on this.

I certify that this dissertation reports original work by me during my university project except for the following (*adjust the list below according to the circumstances*):

- The data in Section 2.4.1 and 3.2 was predominantly sourced from[6].
- The data in Section 2.4.2 and 4.2 was predominantly sourced from [7].
- The Code discussed in section 3.7 is largely taken from DEAP documentation [11] and public repository https://github.com/DEAP/deap
- Code from Section 4.2.4 has been predominantly sourced from [14]
- The code discussed in section 4.3 is mostly taken from [13]
- Section 4.4.2 code incorporates elements from [8] and [16]

Signature *Michael Abebaw Bogale* **Date** 07 September 2023

Acknowledgements

First and foremost, I would like to express my profound gratitude to Dr. Andrew Hoyle. His guidance has been invaluable throughout this journey. He has not only played a pivotal role in shaping the trajectory of my dissertation but also inspired me to explore concepts I had never previously considered. Dr. Hoyle's unwavering support and his continuous encouragement, often encapsulated in his words "try it," have instilled in me a belief that I can make significant contributions to the realm of Artificial Intelligence in my future career. His mentorship has been transformative, broadening my horizons and refining my research capabilities.

To my family, words seem insufficient to convey the depth of my gratitude. Their unwavering belief in my potential has been my anchor. They have consistently prioritized my ambitions and aspirations, often putting my needs and dreams ahead of their own. Their devoted support, sacrifices, and encouragement have been instrumental in bringing me to this pivotal moment in my academic journey. I am deeply thankful to them.

I also wish to acknowledge the contributions of those, whose work I have referenced in this dissertation, reinforcing the academic integrity and collaborative spirit of scholarly research.

Table of Contents

Abstract i
Attestation ii
Acknowledgements iii
Table of Contents iv
List of Figures vi
1 Introduction 1
1.1 Background and Context 1
1.2 Scope and Objectives 1
1.2.1 Algorithm Implementation and Optimization: 1
1.2.2 Data-Driven Insights on EV Performance: 1
1.2.3 Integration of Real-World Data: 1
1.3 Achievements 2
1.4 Overview of Dissertation 2
2 State-of-The-Art 3
2.1 EV Range Anxiety and its Implications 3
2.2 Existing Solutions and Challenges 3
2.3 The Need for an Innovative Approach 3
2.3.1 Introduction to Search Algorithms 4
2.3.2 Rationale for Choosing Genetic Algorithms 4
2.4 Sources of Data 5
2.4.1 Tesla Model 3: Real-world Charging Data Insights 5
2.4.2 BMW i3 (60 Ah): Real-world Driving Data Insights 5
3 A Genetic Algorithm Approach to Efficient Charging Strategies of Tesla Model 3 on a Simulated Route 6
3.1 Introduction 6
3.1.1 Aim of the Chapter 6
3.1.2 Towards a Smooth Journey: Addressing the Core Challenge 6
3.2 Gathering the Right Data 7
3.2.1 Core Parameters for the Charging Data of Tesla Model 3 7
3.2.2 Charge Curve Insights: The Art Behind the Numbers 8
3.2.3 Deeper Insight 9
3.3 Data Extraction and Interpretation 9
3.3.1 State of Charge (SoC) and Time Extraction 9
3.4 Designing a Simulated Journey 11
3.4.1 Charging Station Sample Data - A Closer Look: 12
3.5 Journey Dynamics and Parameters: Steering our Tesla Model 3 Through Mathematical Rules 12
3.5.1 Optimization Objective: Minimizing Charging Time: 13
3.6 The Rationale Behind the Algorithm 14
3.6.1 What are Genetic Algorithms? 14
3.6.2 Why Genetic Algorithm? 14
3.7 Core code points on the algorithm 15
3.7.1 Initial Parameters: 15
3.7.2 Interpolation Function (interpolate_charge_time): 15
3.7.3 Genetic Algorithm Configuration: 16
3.7.4 Importing the DEAP Library: 18
3.7.5 Definition of Fitness and Individual Classes: 18
3.8 Results and Discussion 18
3.8.1 Result Challenges and Solutions: 18

3.8.2 Result Analysis 18
4 Enhancing Realism: Leveraging Google's API and Predictive Machine Learning 21
4.1 Introduction 21
4.2 Section 1: Gathering the Right Data for Machine learning. 22
4.2.1 Source of the Data 22
4.2.2 Structure of the Dataset 22
4.2.3 Key Data Categories and Their Significance 22
4.2.4 Data Pre-processing 23
4.2.5 Machine Learning Data Preprocessing 24
4.3 Model Construction and Assessment 24
4.4 Section 2: Real-world Data Integration and Analysis 27
4.4.1 Objective 27
4.4.2 Harnessing Google API for Route and Charging Stations Data 27
4.4.3 Visualization: 28
5 Conclusion 29
5.1 Summary 29
5.2 Evaluation 29
5.3 Future Work 30
References 31
Appendix 1 33
Appendix 2 – User guide 38
Appendix 3 – Installation guide 39

List of Figures

Figure 1. Charge curve Tesla Model 3 SR+ RTM75 Charger. [6] ... 8
Figure 2. Charging Curve Extracted from Manually Traced Data ... 10
Figure 3. Plentiful Charging stations (500 miles, 125 charging stations) ... 12
Figure 4. Interpolation Equation [12] ... 16
Figure 5. Charging strategy with 2-minute penalty ... 19
Figure 6. Charging strategy with 5-minute penalty ... 20
Figure 7. Extracted EV Charging Stations from Glasgow to Wolverhampton ... 21
Figure 8. Sample Row data of Trip A04 with primary interest [7] ... 22
Figure 9. Trip B20 vehicle speed throughout the journey (Km/hr) ... 23
Figure 10. Trip B20 Ambient temperature throughout the journey(^{0}c) ... 23
Figure 11. Data after Feature Engineering. ... 24
Figure 12. Model generated Final SOC'S by varying Temperature. ... 25
Figure 13. Model generated Final SOC'S varying Speed. ... 26
Figure 14. Location charging stations from Stirling to Birmingham ... 27
Figure 15. Selected charging stations by the algorithm from Stirling to Birmingham ... 28

1 Introduction

1.1 Background and Context

The transportation sector, undergoing a significant transformation, is steadily shifting from fossil fuel dependency to the embrace of Electric Vehicles (EVs), representing a paradigm shift towards sustainability and cleaner transportation alternatives. While the appeal of EVs primarily revolves around their environmental friendliness, technological innovation, and operational cost-effectiveness, they aren't without their challenges.[2]

A significant obstacle, often emphasized in both scholarly and public discourse, is 'range anxiety'. This term encapsulates the inherent apprehension EV drivers feel about their vehicle's battery depleting before reaching a charging station or destination. The phenomenon is like the fear of running out of fuel in conventional vehicles but magnified due to the relative scarcity of EV charging infrastructures when compared to abundant petrol stations.[1]

The study at its core, the research acknowledges the challenge and actively seeks computational solutions to mitigate it. By leveraging advanced algorithms tailored for the specificities of EVs. The project undertakes a journey from theoretical formulations to practical, actionable insights.

1.2 Scope and Objectives

In this research, we navigate the intersection of cutting-edge electric vehicle (EV) advancements and the practical realities confronting EV users. Our exploration revolves around the journey dynamics of an EV—focusing not just on reaching the destination but doing so optimally. We delve into the details of the EV charging curve and the complexities it introduces to travel planning. Further, we attempt to forecast how external factors, such as driving speed and ambient temperature, shape the performance of an EV.

The central objectives of this research are:

1.2.1 Algorithm Implementation and Optimization:

To design, implement, and refine an Algorithm tailored for determining an efficient charging strategy for EVs. The strategy seeks to minimize the travel time for a given distance, considering variables such as the state of charge (SOC), locations of charging stations, and respective charging rates.

1.2.2 Data-Driven Insights on EV Performance:

Dive deep into understanding how specific road conditions, including driving speed and ambient temperatures, influence the discharge rates of EVs. This involved curating a dataset of 72 distinct trips, refining this dataset, and subsequently constructing a predictive model. The model, armed with data like initial SOC, ambient temperature, and journey dynamics, predicts the final state of charge.

1.2.3 Integration of Real-World Data:

Enhance the theoretical constructs of our GA by integrating it with real-world data procured via Google's API. This not only includes basic route details but extends to detailed insights like total route distance, location of charging stations, and their relative proximities from the journey's start.

1.3 Achievements

- **Genetic Algorithm (GA) Framework:** Over several iterations for the journey, the Genetic Algorithm demonstrated notable progress. It's worth highlighting that travel time was optimized remarkably, decreasing from an initial estimate of 15 hours to a more efficient duration of 11 hours and 7 minutes. This underscores the algorithm's potential in adaptively navigating the complexities of EV charging strategies.
- **Development of a Predictive Framework:** Through diligent data collection and processing, we instituted a predictive model that holds considerable promise. This model, when furnished with specific parameters, can proficiently forecast an EV's final state of charge. Such a capability is crucial, offering a forward-looking perspective that aids users in better navigating their EV journeys.
- **Harmonizing Theory with Practicality:** Our research went beyond mere theoretical formulations by embracing real-world data through the integration of Google's API. This integration allowed us to map out comprehensive routes and extract essential travel details.

1.4 Overview of Dissertation

The forthcoming chapters are structured to a coherent exploration of the conducted project:

Chapter 3: Optimizing Travel Using Genetic Algorithm

This chapter places emphasis on the development and application of an optimization methodology specific to the Tesla Model 3. The objective is clear: provide drivers with a designed travel strategy, highlighting precise charging stops and durations.

By analysing and integrating various computational elements, this chapter provides an innovative solution for one of the most pressing concerns for EV drivers: optimizing the balance between driving and charging durations. We explore, in detail, the key factors that play a role in ensuring the efficiency of an EV journey, with a primary focus on minimizing charging times without compromising on the journey's effectiveness.

Chapter 4: Real-world Application and Predictive Modelling

Transitioning from the theoretical constructs of the previous chapter, this section dives into the practical implications of the research.

The initial segment emphasizes the role of specific road conditions, such as driving speed and ambient temperature, and how they affect an EV's discharge rate. Through an intensive dataset of 72 trips, this segment sheds light on the data cleaning processes adopted and the resulting predictive model's capabilities.[7]

In contrast, the subsequent segment magnifies the significance of integrating practical, real-world data, highlighting the utilization of Google's API. This integration offers readers an insight into the blend of theoretical algorithmic concepts developed in Chapter 3 with tangible, real-world applications. The chapter encapsulates the successful integration of essential data like route details, charging station locations, and relative distances, all curated to optimize the EV travel experience.[8]

2 State-of-The-Art

2.1 EV Range Anxiety and its Implications

Range Anxiety refers to a drivers concern that a vehicle may not possess adequate energy storage, either in terms of fuel or battery capacity, to traverse the required distance to reach the intended endpoint, potentially leaving the passengers stranded. While this concept can be applied broadly, in contemporary times, it's chiefly associated with the functional driving range of battery electric vehicles (BEVs).[1] Range anxiety has been extensively studied, with researchers keen on understanding its psychological and practical roots. Early studies, emphasized the challenges of sparse charging infrastructure, leading to range concerns. This concern has been pinpointed as one of the paramount psychological impediments hindering the widespread acceptance and adoption of electric vehicles by the general population.[2][3]

2.2 Existing Solutions and Challenges

Range anxiety, as previously delineated, is a concern amongst potential and current EV users. Addressing this challenge has been the core of numerous research initiatives and industry efforts. A survey of the existing solutions reveals diverse approaches:

- **Grid-to-Vehicle (G2V) & Vehicle-to-Grid (V2G) Solutions**: These approaches primarily focus on the interplay between the vehicle and the electrical grid. While they do provide avenues to optimize charging and alleviate range concerns, they largely hinge on infrastructure upgrades and grid dynamics, rather than optimizing the EV's inherent performance or offering drivers actionable insights for on-the-go decisions.[17]
- **Mathematical Models**: Some researchers have turned to mathematical equations, formulating models to predict and optimize the charge rate and consumption patterns. Though these models can be precise, they often operate under a set of assumptions, and they are for charging and discharging dynamics, this approach might not hold true in real-world, dynamic conditions. [18]
- **Proprietary Industry Solutions**: Notable is the fact that several companies have ventured into devising solutions for range anxiety. However, a significant portion of these corporate undertakings remain hid in secrecy due to proprietary rights, making it challenging for the broader community to assess, validate, or build upon them.[19]

Given this landscape, there's a noticeable gap. An optimal solution would not only effectively address range anxiety but would also be adaptable, user-friendly, and not overly reliant on large-scale infrastructural changes.

2.3 The Need for an Innovative Approach

Given the limitations and constraints of current solutions, it became evident that a different trajectory was required to address range anxiety holistically. Our preliminary research explored various available options, gravitating towards search algorithms.

The universe of computing has witnessed a wide array of search algorithms designed to find solutions in vast and complex spaces. These algorithms offer methods to comb through these vast spaces efficiently, uncovering solutions that meet specific criteria or objectives. This chapter examines into the of search algorithms, focusing predominantly on their significance in the context of the Electric Vehicle (EV) domain.

2.3.1 Introduction to Search Algorithms

Search algorithms, at their core, are procedures or formulas used to search for solutions within a space. This 'space' could be anything from a database for simple look-up tasks, to more complex domains such as combinatorial configurations, route planning, or optimization problems. Two broad classifications can encapsulate most search algorithms: deterministic and stochastic.[20]

Deterministic Search Algorithms primarily follow a set path or rule. They are predictable in their approach and usually guarantee an optimal solution if one exists. Examples include A* search algorithm, commonly used in pathfinding on a map based on known data.[5]

Stochastic Search Algorithms are more probabilistic. They incorporate random processes and might not always produce the same outcome for multiple runs on the same problem. However, their strength lies in potentially uncovering solutions in vast spaces where deterministic methods might falter. Venturing into stochastic algorithms, we initially evaluated simpler methods like random search and hill-climbing. A random search would indiscriminately explore the solution space, lacking the strategy and precision we're aiming for. Meanwhile, hill-climbing, although more strategic, could easily get trapped in local optima given the complex charging curve insights and the variable nature of EV charging stations.

With these considerations, our sights turned to the promising realm of Genetic Algorithms. Genetic Algorithms (GA) are a subset of stochastic search algorithms inspired by the process of natural evolution. Leveraging principles from genetics and natural selection, GAs are adepted at searching through large and complex spaces, making them particularly suited for optimization problems.[5] Although, my exposure to genetic algorithms, especially as part of my coursework on stochastic processes and optimization, provided me with a unique lens to view the problem.

2.3.2 Rationale for Choosing Genetic Algorithms

Several compelling factors made Genetic Algorithms the logical choice for addressing the challenges posited by our EV domain:

1. **Complexity of Problem**: The multifaceted nature of the EV challenge — optimizing routes considering charging stations, distance, battery consumption rates under various conditions — requires a robust search mechanism. Unlike hill-climbing, which risks settling for suboptimal solutions, GAs strikes a balance between exploring new solution areas and exploiting promising regions.

2. **Incorporating Complex Data Patterns:** The unique charging curve insights, characterized by a rapid initial charge rate that decelerates as it approaches full capacity, fits naturally into GA's adaptive model. By simulating multiple generations of potential solutions, GAs can inherently recognize and capitalize on this non-linear charging profile.

3. **Robustness against Unpredictability:** Our simulated journey, with its dense distribution of charging stations, presents a lot of potential charging strategies. GAs thrives in such environments by sampling a wide array of strategies, selecting the best among them, and iteratively refining these choices over generations.

4. **Flexibility with Parameters:** Given the various initial parameters and values, the GA can be tailored for specific scenarios by adjusting factors like mutation probability, crossover rate, and population size. This adaptability ensures the algorithm remains relevant even with evolving data or different EV models.

5. **Prior Success Stories**: GAs have a proven track record in transport and routing problems. Their success in these domains augured well for their potential applicability to the EV context.[5]
6. **Convergence to Near-Optimal Solutions**: While GAs don't always guarantee the absolute best solution, they often converge to near-optimal solutions. In real-world scenarios like EV routing, 'near-optimal' is frequently 'good enough', especially when considering the dynamic variables in play. [5]

2.4 Sources of Data

2.4.1 Tesla Model 3: Real-world Charging Data Insights

Analytical tasks, especially those steeped in advanced algorithmic studies, are only as robust as the data underpinning them. Acquiring precise and relevant data is critical. Our investigation into the Tesla Model 3, an embodiment of state-of-the-art EV technology, was driven by this principle. Conducted in Broadbeach, Queensland, Australia, this study leveraged the reliable and modern RTM75 charger to collect charging data from the vehicle. VedaPrime, a recognized expert in the EV realm, supervised the entire process to ensure data reliability and comprehensiveness [6].

The research considered various parameters like the State of Charge (SoC), environmental conditions, and the vehicle's internal climate control. Noteworthy findings include the vehicle's ability to quickly charge from 9% to 70% SoC in 25 minutes, with a subsequent slowing in the rate as the charge approached 100%.[6]

2.4.2 BMW i3 (60 Ah): Real-world Driving Data Insights

The core of our research was dependent on collating a detailed dataset showcasing the BMW i3's behaviour and characteristics. Spread across 72 real driving trips, this dataset offered granular insights into load fluctuations and various attributes like environmental data, vehicle metrics, battery information, and heating circuit readings.[7]

Data Source and Structure

The database, garnered from these driving trips, ranged between 10,000 to 20,000 rows of data for each trip, detailing about 38 columns per dataset. This depth of data ensured an exhaustive representation of the BMW i3's performance under different scenarios.[7]

Key Data Categories and Implications

The measurement data was bifurcated into two main categories: summer and winter recordings. While the summer data faced some inconsistencies, the winter dataset was hailed for its detailed and holistic representation.[7]

3 A Genetic Algorithm Approach to Efficient Charging Strategies of Tesla Model 3 on a Simulated Route

3.1 Introduction

As the 21st century advances, the focus on sustainability has become vital. The era of internal combustion engine vehicles, which contribute a significant portion of greenhouse gas emissions, is gradually diminishing, giving rise to the adoption of cleaner and more sustainable Electric Vehicles (EVs). Pioneered by industry leaders such as Tesla, these EVs are revolutionising our transportation infrastructure, indicating a novel era in transportation.[9]

Nevertheless, this transformation brings its own set of challenges. Among the problems faced by EVs, the phenomenon termed as 'range anxiety' is notably significant. It refers to the apprehension experienced by electric vehicle drivers regarding the depletion of the battery before reaching a designated destination or charging facility. This apprehension has emerged as a significant obstacle for potential EV adopters, frequently eclipsing the various advantages of electric vehicles. The prospect of being immobilised in a remote area due to an exhausted battery is a genuine concern for many, particularly for those familiar with the widespread availability of fuel stations.[1]

This chapter is dedicated to addressing this challenge. The primary objective is to conceptualise and formulate an optimization methodology to mitigate range anxiety, with a specific focus on the Tesla Model 3. By leveraging a harmonious integration of science, empirical data, and computational algorithms, the intention is to provide Tesla Model 3 drivers with a comprehensive and efficient travel strategy, presenting precise charging intervals and durations. Through this endeavour, the aspiration is to render extended EV journeys more streamlined and predictable.

3.1.1 Aim of the Chapter

The primary goal of this chapter is twofold: to determine *when to stop* and ascertain *how long to stop* during a journey to optimally recharge a Tesla Model 3. By optimally, we refer to a strategy that minimises the total journey time.

This is a composite of two variables:

Total Journey Time = Total Driving Time + Total Charging Time

Assuming the Total Driving Time remains consistent, driven by factors such as distance and driving conditions, the key to an efficient journey lies in minimising the Total Charging Time. It's pivotal to understand that the Total Charging Time isn't just about how long the vehicle is plugged in; it's also about the strategy – knowing where to charge and for how long, to continue the journey with confidence. To this end, a robust model for the Total Charging Time becomes indispensable, which this chapter endeavours to construct.

3.1.2 Towards a Smooth Journey: Addressing the Core Challenge

For a successful model aimed at optimizing EV charging strategies and mitigating range anxiety, it's imperative to feed the model with comprehensive and accurate data. Our endeavour was to bridge this data need by encompassing several data relevant to our study. The data required to go through our journey is expressed as follows.

1. Vehicle-Specific Data:

- **Battery Capacity:** Information on how much energy the EV's battery can store.
- **Battery Discharge Rate:** This entails how fast the battery drains at various speeds and conditions.
- **Charging Rate:** A detailed curve showing how fast the EV's battery charges at different State of Charges (SoC).

2. Journey-Specific Data:

- **Distance:** The total length of the journey.
- **Driving Speed:** Assumed constant for this study, it affects the battery's discharge rate.
- **Charging Stations:** Locations of charging stations on the route and their specific charging capabilities.

3.2 Gathering the Right Data

Importance of Reliable Data: In any analytical work, especially in advanced algorithmic studies like ours, the bedrock upon which the entire analysis stands is data. Acquiring data that is precise, reliable, and relevant to the subject at hand is pivotal. For our Algorithm model, it wasn't just about obtaining any data; it was about obtaining the *right* data.

Source of Our Data: Our primary data source was from an in-depth experiment based on real-world charging data stemming from a Tesla Model 3. Why Tesla Model 3? Because it represents a blend of cutting-edge technology, popularity, and is an epitome of electric vehicle advancements. Furthermore, this vehicle was paired with an RTM75 charger, ensuring that the charging process was as efficient and modern as the vehicle itself.[6]

Location Significance: The trial was carried out in the sunny locales of Broadbeach, Queensland, Australia. Broadbeach, with its relatively consistent climatic conditions, especially on sunny days, reduces external weather-induced variables that might skew charging data. Moreover, Australia's electric vehicle infrastructure is rapidly evolving, making it an apt location for such studies.[6]

Supervision and Authenticity: Ensuring the reliability and authenticity of the data, the experiment was diligently supervised by VedaPrime, a recognized expert EV owner. Their profound understanding of electric vehicle intricacies ensured that the data extracted was holistic, capturing all the nuanced aspects of the charging process. [6]

3.2.1 Core Parameters for the Charging Data of Tesla Model 3

1. Starting State of Charge (SoC): The journey initiated with the Tesla Model 3's battery at an SoC of 10%. This relatively low starting point was chosen to simulate a common scenario faced by many EV drivers, where the vehicle is used for a considerable period without recharging.

2. Objective of the Charging: While the overarching goal was to attain a 100% SoC, or fully charge the battery, this not only provides a comprehensive test of the charging infrastructure and the vehicle's battery health but also simulates the necessity of prepping the vehicle for a long journey.

3. Environmental Conditions:

- **Ambient Temperature:** The study was conducted on a summer morning, where the average ambient temperature at 8 am was 26°C. Such a temperature is considered moderate and can have specific impacts on the battery's charging performance. Typically, batteries have an optimal temperature range for charging, and our selected condition lies well within this range for the Tesla Model 3.[6]
- **Time of Day:** Conducting the study in the morning ensures that the solar radiation and heat buildup from the previous day have minimal influence on the vehicle's charging performance.[6]

4. Vehicle's Internal Air conditioning: For the entirety of the charging period, the Tesla Model 3's internal air conditioning system was uniformly maintained at 21°C. This is a crucial parameter as the internal climate control can draw power from the battery, affecting its SoC. Keeping the internal climate at a consistent temperature ensured that this variable's effect on the battery's charging rate was constant and could be factored into our results.[6]

Figure 1. Charge curve Tesla Model 3 SR+ RTM75 Charger. [6]

3.2.2 Charge Curve Insights: The Art Behind the Numbers

1. Overall Charge Time: From the initial inspection of the data, one can discern that transitioning from a 9% state of charge (SoC) to a full 100% capacity required precisely 60 minutes. This is an impressive feat given the advancements in charging technologies.

2. Delving into the Charge Time Discrepancy:

- **Initial Burst of Energy:** The initial phase of charging, from 9% to roughly 70% SoC, was remarkably swift, consuming a mere 25 minutes. This rapid charging can be attributed to the battery's ability to accept higher currents when at a lower state of charge without risking damage or overheating.

- **The Slowing Down:** However, after crossing the 70% threshold, we begin to observe a marked deceleration in the charging rate. This section, from 70% to 90%, accounted for 15 minutes, and the final stretch from 90% to 100% took an additional 20 minutes.

3.2.3 Deeper Insight

Understanding the Shift

This shift in charging pace post the 70% mark isn't an anomaly but a deliberate design choice in lithium-ion batteries. Often termed as the 'Constant Voltage' phase, the battery management system begins to regulate the charging current to avoid potential hazards.

- **Why Does This Happen?** As a lithium-ion battery charges, the potential difference between the battery's current voltage and its maximum voltage decreases. As this gap narrows, the charging current must be reduced to prevent damage to the cell structures. This is crucial to ensure safety, battery longevity, and performance.[10]

A Broader Perspective

The charging behaviour witnessed, especially the gradual decline in charging speed as the battery nears its full potential, is not unique to the Tesla Model 3 SR+ but is characteristic of most modern-day lithium-ion batteries.[10]

Charging Curve's Role in Optimization:

The charging curve isn't just a passive observation; it's the linchpin of our optimization problem.

- **Predictive Analysis:** The non-linear nature of the charging curve provides valuable predictive insights. Recognizing the thresholds and the varying rates of charge acceptance allows us to forecast the charging time at different intervals accurately.
- **Optimal Stop Decisioning:** With the knowledge of how a battery charges, decisions about when to stop and how long to pause become more strategic. Instead of arbitrary halts, stops can be coordinated at SoC levels where charging is faster, thereby minimizing overall journey time.

3.3 Data Extraction and Interpretation

Understanding and drawing insights from data is the foundation of any research, and our project is no exception. The primary focus of this section is to dissect the processes we've undertaken to extract, transform, and interpret the data, particularly with regard to the State of Charge (SoC) and its temporal progression.

3.3.1 State of Charge (SoC) and Time Extraction

Origins of the Data:

Before diving into the extraction process, it's pivotal to understand the genesis of our data. The data roots itself in a graphical representation from a picture of the interplay between the SoC and the charging time for a Tesla Model 3. [6] This graph is our source, tracing the minute-by-minute progression of the charging process.

Extraction Techniques:

Our methodology embraced a hands-on approach:

- **Manual Sourcing:** Given the granularity and precision we sought; manual extraction was employed. Through the aid of specialized tools, specific points on the graph were selected, enabling us to capture exact SoC values and their respective time stamps.
- **Quality Assurance:** Post extraction, a secondary review was conducted to ensure the accuracy and fidelity of the data. Any discrepancies were promptly rectified, assuring the integrity of our dataset.

Transformation for Computational Compatibility:

Upon securing a robust dataset, our next endeavour was its transformation:

- **Format Transformation:** With the cleaned data in hand, the next stride was format transformation. This involved converting the raw data into a structure conducive to computational analysis. Tools and libraries, intrinsic to Python such as Pandas, Numpy were employed in this pursuit.

Data Structuring and Storage:

The transformation is manifested in the **charging_rate_df** DataFrame, a structured and organized representation of our dataset using the Pandas library:

- **DataFrame Dynamics:** In this Pandas DataFrame, each row encapsulates a unique timestamp, while the columns detail attributes like SoC, charging Time.
- **Visualization Prowess:** Armed with this DataFrame, we were able to derive visually informative graphs, plots, and charts, augmenting our analytical capabilities.

Insights from the Derived Data:

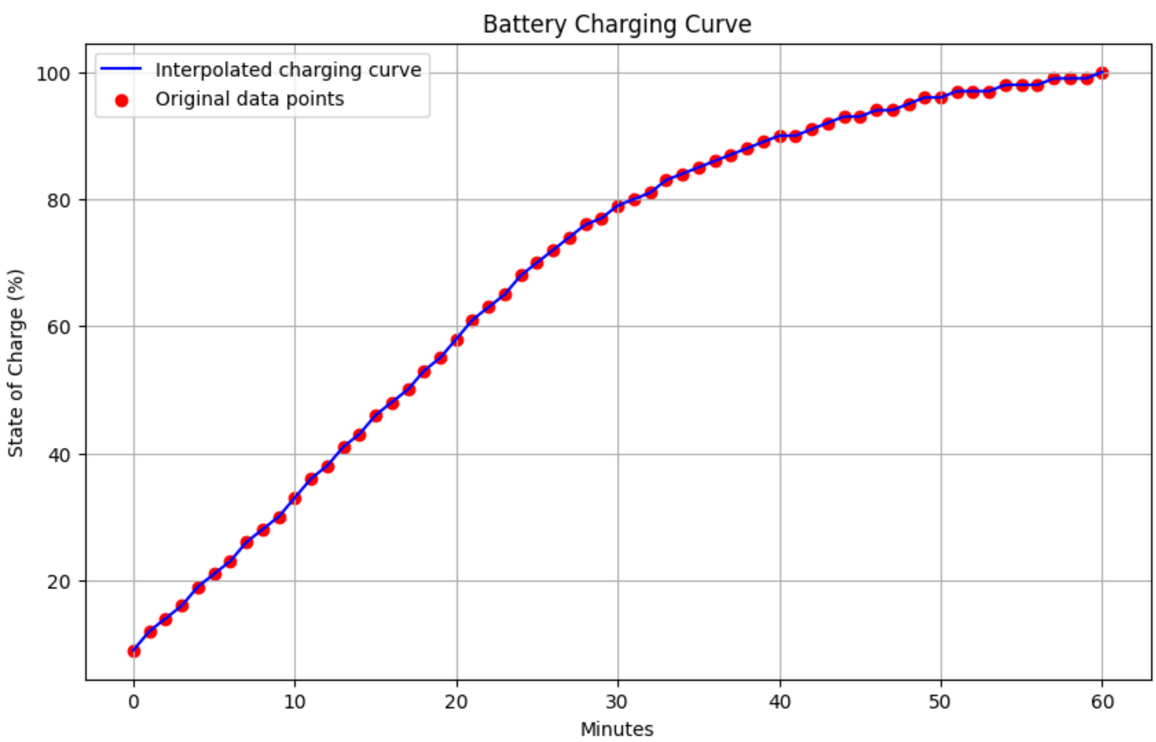

Figure 2. Charging Curve Extracted from Manually Traced Data

3.4 Designing a Simulated Journey

Given the expansive nature of contemporary highways, there exists a persistent challenge of unpredictable charging infrastructure for Electric Vehicles (EVs). For the purposes of this experiment, we have conceptualized a scenario characterized by abundant charging availability – envision a highway generously equipped with charging stations at regular intervals.

Journey Overview: Consider a hypothetical 500-mile journey. To rigorously evaluate the competence and adaptability of our genetic algorithm, the designated route has been densely populated with charging stations.

Charging Station Distribution: For this exercise, a remarkable total of 125 charging stations have been postulated along the 500-mile journey, with a station situated every 4 miles.

Why So Many Stations?

- **Flexibility:** The augmented availability eliminates the traditional range anxiety. This enriched infrastructure provides our algorithm with an extensive scope to exercise its decision-making capabilities.
- **Testing our Algorithm's Decision-Making:** Confronted with too many choices, the genetic algorithm is tasked with discerning the optimal route, harmonizing charging duration, and driving time. Inherent in this is the algorithm's ongoing evaluation of the practicality of each potential stop.
- **The Penalty Framework:**

Incorporation of the Penalty Framework: In real-world scenarios, every decision has its consequences. Particularly, when considering an electric vehicle stopping for charging isn't just about the duration spent charging.

In the initial version of our algorithm, where the penalty wasn't considered, the optimization produced frequent and impractical stops, sometimes for durations as brief as a minute. Though this seemed optimal in a purely computational sense, it failed to emulate real-world practicalities: diverting to a charging station, initiating the charging process, and payment formalities all take time.

Upon integrating a time penalty to account for these practical processes, the algorithm's results transformed significantly. The number of stops was reduced, presenting a more pragmatic and efficient charging route for users. This adjustment underscores the importance of melding computational optimization with real-world practicalities, highlighting the depth of thought invested in this project.

- **Algorithmic Dynamics – The Decision Evolution:**

With an array of charging stations available, the algorithm's main focus becomes determining the optimal value of each potential stop. This involves considering:

1. The current battery charge level of the electric vehicle.
2. The distance to the subsequent charging station.
3. The time penalty associated with making a stop.

These critical factors play a pivotal role in shaping the algorithm's decisions. With each iteration, the algorithm enhances its understanding, continuously seeking the best compromise between the number of charging stops and the overall duration of the journey.

Figure 3. Plentiful Charging stations (500 miles, 125 charging stations)

3.4.1 Charging Station Sample Data - A Closer Look:

For an illustrative breakdown, let's zoom into our **station_df** DataFrame. This dataframe is our charging station ledger, detailing each station's specifics:

index	station_number	location(mile)
0	1	4
1	2	8
2	3	12
3	4	16
4	5	20
...	...	...

3.5 Journey Dynamics and Parameters: Steering our Tesla Model 3 Through Mathematical Rules

The heart of any analytical model is the constellation of parameters that drive it. When contemplating an extended journey in a Tesla Model 3, it's imperative to lay down the analytical framework that will anchor our journey. This involves establishing a detailed set of rules and assumptions, each playing a pivotal role in the unfolding of our simulation.

These are the vehicle parameters:

Initial State of Charge (initial_soc):

Our journey begins with the car's battery at a certain level of charge. For the purpose of this simulation, we've chosen an arbitrary starting point of 40%. This isn't a fixed rule, and in real-life scenarios, the initial_soc can vary based on previous use, where you last charged, and other variables.

Journey Length (total_distance):

While any distance could have been chosen for this simulation, we've selected a lengthy 500 miles. This allows us to evaluate the efficiency and reliability of our algorithms across a considerable distance, approximating a more strenuous real-life journey scenario.

Battery Usage Rate (discharge_rate):

This might be the most crucial part of our discussion it will be elaborated in chapter 4. Discharge rate represents the rate at which the battery depletes as the car is driven. Here's how it's calculated for our Tesla Model 3:

Deriving the Discharge Rate: To find out how much percentage of the battery is consumed for every mile driven, we use the following calculation: Given that the car covers 250 miles on a full 100% charge:

For 1 mile: Discharge Percentage = $\frac{100}{250} = 0.4\%$

Thus, the battery discharges at a rate of 0.4% for every mile driven.

Constant Speed & Implications for Journey Time:

Speed: We've chosen to maintain a constant speed of 50 miles/hr throughout our journey. This is a critical assumption, as changing speeds can influence battery discharge rates differently. The implications of changing speeds on battery discharge rates will be explored in Chapter 4.

Driving Time: With the constant speed of 50 miles/hr and a total journey length of 500 miles:

Total Driving Time = $\frac{\text{Journey Length}}{Speed}$

= $\frac{500\ miles}{50\ miles/hr} = 10\ hours$ travel time

The Constituents of Total Journey Time:

The cumulative time taken for any EV journey can be dissected into two primary segments:

- **Driving Time:** The sheer duration the vehicle spends on the road, covering the distance.
- **Charging Time:** The time invested at charging stations to replenish the battery.

Total Journey time = Total Driving Time + Total Charging Time

Formally, this relationship can be expressed as:

Total Journey Time = Driving Time + Charging Time

Given Driving Time is fixed, to minimise the Total Journey Time requires minimising the Total Charging Time.

Considering the above assumptions and parameters, we can better model and predict the battery consumption for our journey in the Tesla Model 3.

3.5.1 Optimization Objective: Minimizing Charging Time:

Our primary objective is crystalline: Minimize the **Total Journey Time**. Since the driving time remains a constant in our model, our real challenge lies in diminishing the charging time. By doing so, we effectively reduce the overall journey duration.

Fitness or Objective Function:

The effectiveness of our genetic algorithm is evaluated through a fitness (or objective) function. This function gauges the quality of the solutions (in our case, journey strategies) the algorithm produces.

Fitness Function = − (*Charging Time*)

Here, the negative sign signifies our intent to minimize this variable. The lesser the charging time, the higher the fitness of the solution. Our genetic algorithm, through iterations, refines its strategies, constantly striving to maximize this fitness function. In simpler terms, the algorithm's goal is to keep the charging time as brief as possible.

Minimum and Maximum State of Charge (min_soc & max_soc):

The health and longevity of any battery, including that of our Tesla Model 3, are influenced by how much we discharge and charge it.

- **Minimum State of Charge (min_soc):** We've set a floor at 9% to prevent complete battery drainage, safeguarding against potential irreversible damages to the battery structure.
- **Maximum State of Charge (max_soc):** Charging beyond capacity can be hazardous. Thus, our ceiling is a natural 100%, ensuring we don't overcharge the battery.

3.6 The Rationale Behind the Algorithm

3.6.1 What are Genetic Algorithms?

Think of Genetic Algorithms as a method inspired by nature. Genetic Algorithms simulate natural selection processes. Just as in nature where the fittest individuals are selected for reproduction, GAs evolve potential solutions over generations to derive the optimal solution. [5]

3.6.2 Why Genetic Algorithm?

In conceptualizing a strategy to optimize the total driving time for an Electric Vehicle (EV), one is confronted with a multi-dimensional optimization problem, influenced by a plethora of decision points, encompassing charging duration and the selection of charging stations. The Genetic Algorithms (GAs) present themselves as a particularly suited mechanism for addressing such challenges for the following reasons:[4]. Refer to Appendix 1[A]

1. **Specific Objective Handling**: Contrary to certain optimization methodologies that may struggle with the intricacies of multi-objective problems, GAs exhibit proficiency in addressing distinct objectives. In this context, the predominant objective is the minimization of total driving time, which indirectly necessitates the optimization of charging intervals and specific station selections.[4]
2. **Binary Decision Complexity**: The decision to stop or not stop at each station is binary in nature. Mathematically, the number of potential combinations grows exponentially mathematically expressed as $f(n) = 2^n$, where n is the number of stations. Genetic Algorithms (GAs) work seamlessly with binary decision problems, evolving and refining strategies over generations. The function below provides an example of making such binary decisions based on a given solution:[4]
3. **Adaptive to Non-Linear Constraints**: The charge rate of batteries can exhibit non-linear characteristics, particularly towards the tail end of their capacity. This, combined with the variability in charging rates and the dynamic journey constraints, can complicate the optimization landscape. GAs, not being bound by the need for continuity or derivative information, can easily navigate these non-linear terrains.[4]
4. **Handling Large Decision Spaces**: Given the many charging stations in the journey, the potential charging strategies are vast. Traditional techniques might stagnate in local op-

tima or get overwhelmed by the solution space's size. GAs, however, adopt a more exploratory approach, diversifying their search and iteratively refining solutions.[4]

5. **Adaptive Evolution**: Parameters such as the state of charge (SOC), station locations, and the distance between them can influence the EV's journey. The dynamic nature of these parameters necessitates an algorithm that can adapt and evolve. GAs, through their crossover and mutation operations, offer this adaptability, recalibrating solutions as the journey's dynamics change.[4]

The GA iteratively refines the decision of which stations to stop at and for how long, by evolving a population of possible strategies over generations. By focusing on the objective of minimizing total driving time, the GA inherently considers the charging duration and whether stopping at a particular station is beneficial. Moreover, by integrating real-world constraints, such as the state of charge linear interpolation, the code ensures solutions are optimal and feasible.

3.7 Core code points on the algorithm

In this piece of work, an advanced solution was developed to optimize the travel time to determine *when to stop* and ascertain *how long to stop* during a journey to optimally recharge a Tesla Model 3. Further details can be found in Appendix 1[A].

Here's a partial detailed breakdown of the steps taken:

3.7.1 Initial Parameters:

Values:

- Initial State of Charge (SOC): 40% (**initial_soc = 0.4**)
- Total intended distance: 500 miles (**total_distance = 500**)
- Average Speed: 50 mph (**average_speed = 50**)
- Discharge rate: 0.004 SOC per mile (**discharge_rate = 0.004**)
- Minimum acceptable SOC: 9% (**min_soc = 0.09**)
- Time penalty for each station stop: 0 minutes (**STOP_PENALTY = 0**)

3.7.2 Interpolation Function (interpolate_charge_time):

Why Use Interpolation?

Interpolation is employed to predict values within a given range, utilizing a known set of data points. Its purpose is especially for situations where obtaining precise data for every possible input is challenging, time-consuming, or expensive. For the charging strategy of an electric vehicle, this becomes necessary because of point based data. [12]

Linear interpolation is used because of simplicity, computational efficiency, and resource.

Linear Interpolation Formula Used:

- # Linear interpolation formula: y = y1 + ((y2 - y1) / (x2 - x1)) * (x - x1)

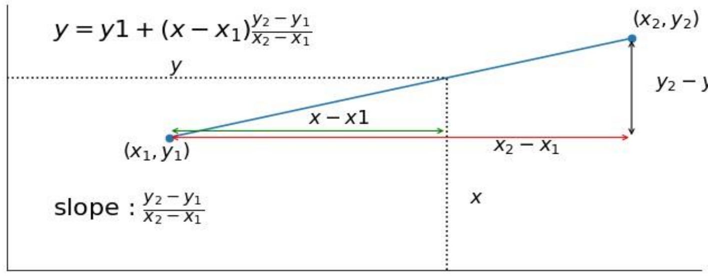

Figure 4. Interpolation Equation [12]

3.7.3 Genetic Algorithm Configuration:

The optimal charging strategy for the electric vehicle, covering the designated route, is computed using a genetic algorithm. This computational method mimics the process of natural selection to find the best solution among a vast number of possibilities. Here's a detailed breakdown of the hyperparameters that drive the genetic algorithm and their implications in the context of the EV travel optimization:[11] Further details can be found in Appendix 1A.

Values:

- Population Size: 500 (**POPULATION_SIZE = 500**)
- Mutation Probability: 20% (**MUTATION_PROB = 0.2**)
- Crossover Probability: 50% (**CROSSOVER_PROB = 0.5**)
- Number of Generations: 100 (**NUMBER_OF_GENERATIONS = 150**)
- Individual Mutation Chance: 5% (**MUTATION_CHANCE = 0.05**)
- Tournament Size: 3 (**TOURNAMENT_SIZE = 3**)

1. **POPULATION_SIZE (500)**:
    - **Overview**: This parameter determines the number of potential charging strategies or travel plans to be considered during the optimization process.
    - **Significance**: With a population size of 500, the algorithm starts its search with 500 diverse itineraries. Each of these represents a unique combination of charging stops aimed at minimizing the total journey duration while ensuring the EV doesn't run out of charge.

2. **MUTATION_PROB (0.2)**:
    - **Overview**: Mutation introduces slight random modifications in the travel plans, ensuring diversity in the pool.
    - **Significance**: A mutation probability of 20% indicates that, on average, 1 out of 5 travel plans will undergo some alteration in each generation. These changes might include modifying decisions related to stopping at specific charging stations, ensuring that the algorithm doesn't get trapped in suboptimal solutions.

3. **CROSSOVER_PROB (0.5)**:

- **Overview**: This parameter guides the crossover operation, where traits from two parent itineraries are combined to produce one or more offspring.
- **Significance**: A crossover probability of 50% ensures that half of the travel plans in the population will engage in this recombination process, allowing the algorithm to blend and experiment with successful features from multiple plans.

4. **NUMBER_OF_GENERATIONS (150)**:
   - **Overview**: This hyperparameter sets the total number of evolutionary cycles the algorithm will undertake.
   - **Significance**: Over 150 generations, the algorithm persistently refines the travel plans through selection, crossover, and mutation. This iterative refinement aspires to the most efficient charging strategy for the vehicle.

5. **MUTATION_CHANCE (0.05)**:
   - **Overview**: This parameter delves deeper into the mutation process, determining how frequently individual components of a travel plan might change.
   - **Significance**: For each possible charging decision within a travel plan, there's a 5% probability it will mutate. This nuanced approach to mutation ensures a thorough yet targeted exploration of potential strategies, preventing drastic and unbeneficial changes that might disrupt an already efficient plan.
6. **TOURNAMENT_SIZE (3):**
   - **Overview:** Tournament selection is a method for selecting individuals based on direct competition. A tournament size determines how many individuals are randomly chosen from the population to compete in each tournament.
   - **Significance:** With a tournament size of 3, for each selection, three individuals are randomly chosen from the population, and the best among them (based on their fitness) is selected. This introduces a balance between randomness and direct competition. By allowing weaker individuals to sometimes be selected, it ensures genetic diversity, yet by typically favouring the stronger competitors, it promotes a steady improvement in the solutions over generations.

7. **STOP_PENALTY (0)**:
   - **Overview**: The efficiency of a travel plan isn't solely about the distance covered or the battery's state of charge. The time spent at each charging stop plays a vital role.
   - **Significance**: The stop penalty hyperparameter quantifies the time penalty associated with each charging stop. In this case, it's set to 0, meaning there's no additional delay considered beyond the charging time itself. However, this parameter can be adjusted to account for real-world considerations like waiting times or non-charging activities.

3.7.4 Importing the DEAP Library:

Upon commencing this project, we opted to utilize the DEAP library, an acronym for Distributed Evolutionary Algorithms in Python. After installing it via pip install deap, we integrated its modules into our work. We chose DEAP for the following reasons:[11]

1. **Efficiency**: Simplifies rapid development without building a genetic algorithm from scratch.
2. **Robustness**: Widely adopted, ensuring consistent results.

3.7.5 Definition of Fitness and Individual Classes:

1. **FitnessMin Class**:[11]
    - **FitnessMin** is a custom fitness class derived from **base.Fitness**.
    - The **weights** parameter is set to **(-1.0,)**, indicating the algorithm seeks to minimize the value of the fitness function. This is typical for problems where we want to minimize a certain metric, For our algorithm Total Travel time.

creator.create('FitnessMin', base.Fitness, weights=(-1.0,))

3.8 Results and Discussion

The primary objective of this study was to implement and optimize a Genetic Algorithm (GA) to determine an effective charging strategy for electric vehicles. This strategy aimed at minimizing the total travel time across a given distance, taking into account parameters such as state of charge (SOC), charging station locations, and charging rates.

3.8.1 Result Challenges and Solutions:

1. **Negative Charging Times**: Our initial algorithm iterations sometimes attempted to charge the vehicle at an already lower state of charge leading to negative charging times. This is physically impractical.
    - **Solution**: To address this, we implemented a penalty mechanism. If the algorithm suggests charging to an SOC lower than the current SOC, we assign it an infinite fitness value, effectively discouraging such decisions.
2. **Low State of Charge upon Arrival**: We noticed that, on occasions, vehicles were arriving at charging stations with critically low states of charge, sometimes even less than 9%.
    - **Solution**: We set such states not good for battery and risk of stop without charge and, as a preventive measure, assigned an infinite fitness value to those instances as well. This ensures the vehicle always maintains a safe buffer of charge.

3.8.2 Result Analysis

Over multiple generations, the GA exhibited continuous improvement in its optimization goal. Notably, there was a marked decrease in total travel time from approximately 15 hours in the initial generations to 11 hours and 7 minutes in the final generation.

Results for different stop penalties were also examined. The relationship between penalties and the number of stops is directly proportional: as penalties increase, the number of stops decreases, revealing the algorithm's endeavour to balance between travel time and charging considerations.

Without Stop Penalty (0-minute penalty)

Total Travel Time: The GA settled on an optimal travel time of 11 hours and 7 minutes.

Charging Stops:

- **Total Stops:** 18. It's notable that some of these stops were only for a fraction of seconds, showcasing the GA's strategy to make frequent but quick stops without any penalty.
- **State of Charge upon Reaching Final Destination:** 9.40%.

Without introducing a penalty, the GA seems to prioritize minimizing the travel time, even if that means making frequent stops.

State of Charge (SOC) Analysis:

- **SOC Range during Stops:** Throughout these stops, the SOC values fluctuated between 9.2% and 60%.
- **Rationale Behind SOC Selection:** The choice of SOC values, specifically ranging from 9.2% to 60%, is not arbitrary. A meticulous examination of the charging curve elucidates this decision. The relationship between the state of charge and charging duration, especially within the bracket of 9% to 70%, approximates a linear relationship. Thus, whether one opts for a range of 9% to 20% or 20% to 30%, the difference in charging time is insubstantial.

With a 2-minute penalty:

- **Total Travel Time:** 11 hours 19 minutes.
- **Number of Charging Stops:** 5.
- **SOC upon Reaching Destination:** 9.60%.

Detailed Stop Analysis: the following is the output of the GA

```
Total travel time: 11 hours 19 minutes
Charging times:  [12.800000000000008, 18.333333333333343, 16.96666666666668, 14.966666666666683, 16.000000000000007]
Charging stations:  [10, 25, 59, 80, 90]
State of Charge upon reaching final destination: 9.60%
```

	Charging Stations	Charging Time(minute)	SOCs Before Charging	SOCs After Charging	Distance (mile)
0	10	12.800000	0.224	0.49	44
1	25	18.333333	0.250	0.65	104
2	59	16.966667	0.106	0.47	240
3	80	14.966667	0.134	0.45	324
4	90	16.000000	0.290	0.64	364

Figure 5. Charging strategy with 2-minute penalty

Comparing the SOCs before and after charging provides insight into the amount of charge added during each stop. On average, the vehicle increases its SOC by around 33.8% at each stop. Introducing a 2-minute penalty leads the GA to prefer longer charging sessions to ensure the vehicle can cover more distance between stops.

Introducing a 2-minute penalty for each stop also shifts the strategy. The GA now balances between fewer stops and achieving a shorter travel time.

With a 5-minute Stop Penalty:

- Travel Time: 11 hours 30 minutes
- Stops: 3 significant charging stops were advised
- Ending SOC: 12.06%

Let's examine the SOC increase for this scenario:

```
Total travel time: 11 hours 30 minutes
Charging times:  [24.800000000000008, 36.60000000000002, 29.10000000000002]
Charging stations:  [10, 48, 80]
State of Charge upon reaching final destination: 12.60%
```

	Charging Stations	Charging Time(minute)	SOCs Before Charging	SOCs After Charging	Distance (mile)
0	10	24.8	0.224	0.71	44
1	48	36.6	0.102	0.81	196
2	80	29.1	0.298	0.83	324

Figure 6. Charging strategy with 5-minute penalty

On average, the vehicle increases its SOC by around 57.2% at each stop. This substantial jump in the average SOC gain per stop indicates that the vehicle is charging more at each stop, again confirming that the GA is trying to maximize distance between stops.

Insightful Observations:

1. **Frequency vs. Duration:** As the penalty for stops increases, the GA not only reduces the number of stops but also increases the amount of charge added during each stop. This trade-off between frequency and duration is crucial for maintaining optimal travel time.
2. **Optimal Charging Range:** In the absence of a penalty, the GA seems to operate the vehicle within a wide SOC range. However, as the penalty increases, the algorithm focuses on maintaining a higher SOC, charging the vehicle more at each stop. This change in strategy indicates that the GA, when penalized, finds it optimal to run the vehicle within a higher SOC range.

Varying the penalty values offers a clear insight into how the genetic algorithm strategizes charging stops and manages travel time. It becomes evident that while raw optimization leads to shortest travel times, practical considerations such as the number of stops and vehicle SOC become pivotal when penalties are introduced. This adaptability is an example of a robust and effective optimization algorithm.

The presented GA offers a dynamic and versatile solution to optimize electric vehicle charging strategies. Its strength lies in its adaptability. While we used a particular charging curve for this analysis, the algorithm can handle a variety of curves corresponding to different vehicles or charging equipment. Its flexibility is further demonstrated by its ability to optimize for varying travel distances and charging station locations.

In conclusion, this GA represents a robust framework for optimizing EV charging strategies across diverse scenarios and constraints. The success in minimizing travel times and its adaptability to various parameters underline its potential for real-world application.

4 Enhancing Realism: Leveraging Google's API and Predictive Machine Learning

4.1 Introduction

The advancement of Electric Vehicles (EVs) in modern transportation systems requires detailed analysis of various factors affecting their performance. In this chapter, our investigation is primarily partitioned into two distinct but interconnected sections.

The first subsection examines deep into the evaluation of specific road conditions and their influence on the discharge rate of EVs. Notably, factors such as driving speed and ambient temperature emerge as pivotal elements in this analysis. To facilitate a comprehensive understanding, we assembled a substantial dataset encompassing 72 distinct trips, sourced directly from data source here [7]. Ensuring the reliability of our data, a cleaning process was instituted. This process was imperative to address missing values, incomplete records, and, in some instances, led to the exclusion of certain trips due to considerable data discrepancies. Our refined dataset then formed the cornerstone for our predictive model. This model, equipped with inputs like initial state of charge, ambient temperature, and speed or distance, can determine the final state of charge with heightened accuracy.

Our exploration in the second subsection takes a more holistic approach, integrating real-world data to enhance the applicability of our findings. We leveraged the robust Google API to procure essential data tailored to the needs of our Genetic Algorithm. By simply inputting initial and destination postcodes, our system can now ascertain the complete route details. But the competence of our methodology doesn't end there. We further enriched our data by extracting additional information such as total route distance and the strategic locations of charging stations throughout the journey. Moreover, the relative distance of these stations from the commencement of the journey was also computed. This synthesis of real-world data, when integrated with our algorithm crafted in Chapter 3, shows the harmony between theoretical concepts and practical applications.[8] Further details can be found in Appendix 1[C].

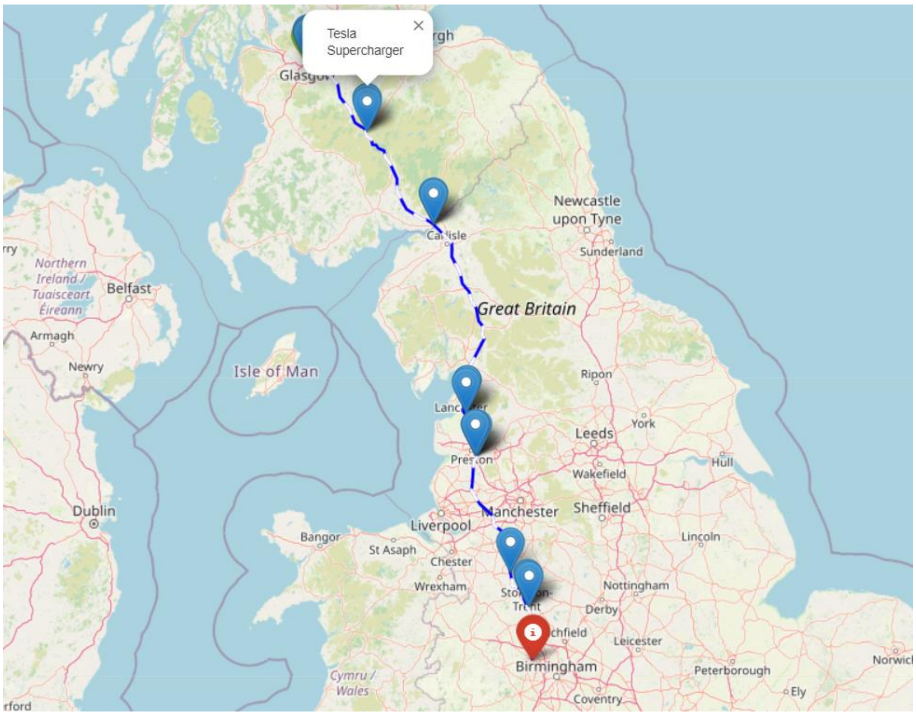

Figure 7. Extracted EV Charging Stations from Glasgow to Wolverhampton

4.2 Section 1: Gathering the Right Data for Machine learning.

The research's fundamental foundation rests on the collection of an extensive dataset representing the behaviour and characteristics of battery electric vehicles, particularly the BMW i3 (60 Ah.[7] This chapter outlines the methodology adopted in gathering this dataset, the nature of the data, and the critical attributes identified as essential for subsequent analysis.

4.2.1 Source of the Data

The data was primarily sourced from a series of 72 real driving trips using a BMW i3 (60 Ah). These trips provided valuable insights into the load fluctuations faced by high-voltage batteries in battery electric vehicles. The dataset contains a variety of attributes such as environmental data, vehicle metrics, battery information, and heating circuit readings. For comprehensive details, the source of the data can be referenced here. [7]

4.2.2 Structure of the Dataset

Each trip's dataset contains a robust structure with approximately 10,000 to 20,000 rows of data, averaging around 38 columns per dataset. Given this structure, the database is substantial, capturing detailed granularity of the vehicle's operation across different conditions. [7]

4.2.3 Key Data Categories and Their Significance

The measurement data are categorized into two distinct categories:

- **Category A:** Data from this category was recorded during the summer months. However, due to certain complications with the measurement system, it lacks a comprehensive dataset.
- **Category B:** This category encompasses data recorded during the winter months and is commended for its consistency and holistic representation of the metrics.

	A	B	C	D	E	F	G	H	I
1	Time [s]	Velocity [km/h]	Elevation [m]	Throttle [%]	Motor Torque [N	Longitudinal Acc	Regenerative Br	Battery Voltage [	Battery Current [
2	0	0	582	0	0	0.19	0	386.8	-1.7
3	0.1	0	582	0	0	0.2	0	386.8	-1.7
4	0.2	0	582	0	0	0.24	0	386.8	-1.7
5	0.3	0	582	0	0	0.25	0	386.8	-1.7
6	0.4	0	582	0	0	0.22	0	386.8	-1.74
7	0.5	0	582	0	0	0.25	0	386.8	-1.79
8	0.6	0	582	0	0	0.21	0	386.8	-1.76
9	0.7	0	582	0	0	0.19	0	386.8	-1.71

J	K	L
[Battery Tempera	max. Battery Ter	SoC [%]
25	26	75.1
25	26	75.1
25	26	75.1
25	26	75.1
25	26	75.1

S	T	U	V
Heater Signal	Ambient Temper	Requested Cool	Heat Exchanger (
1	24	85	14
1	24	85	14
1	24	85	14
1	24	85	14
1	24	85	14
1	24	85	14

Figure 8. Sample Row data of Trip A04 with primary interest [7]

The diagram below depicts the speed variations during Trip B20, which lasted for 23.4 minutes. This data is extracted from the dataset [7], where the x-axis represents time, and the y-axis indicates speed.

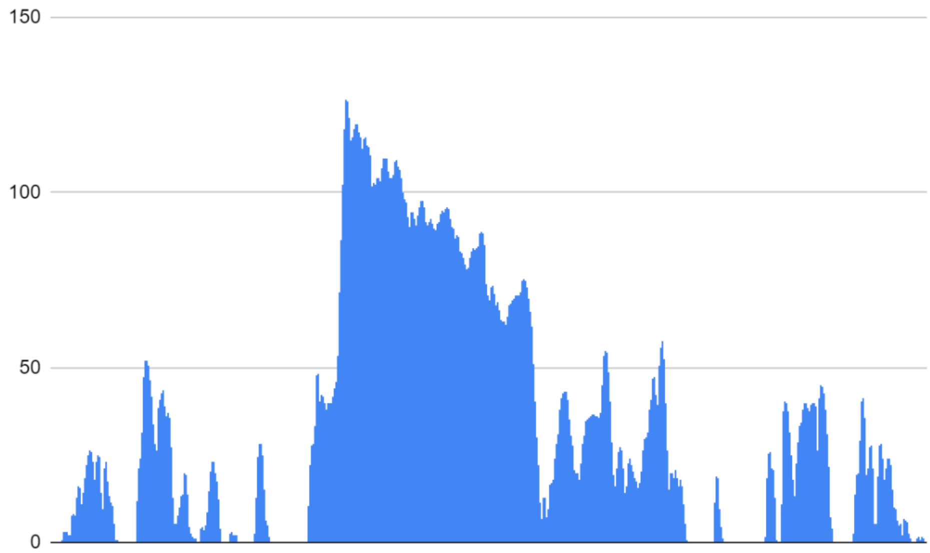

Figure 9. Trip B20 vehicle speed throughout the journey (Km/hr)

The diagram below illustrates the variation in ambient temperature during Trip B20, which spans a duration of 23.4 minutes. This data is extracted from the dataset [7], with the x-axis representing time and the y-axis denoting ambient temperature in degree Celsius.

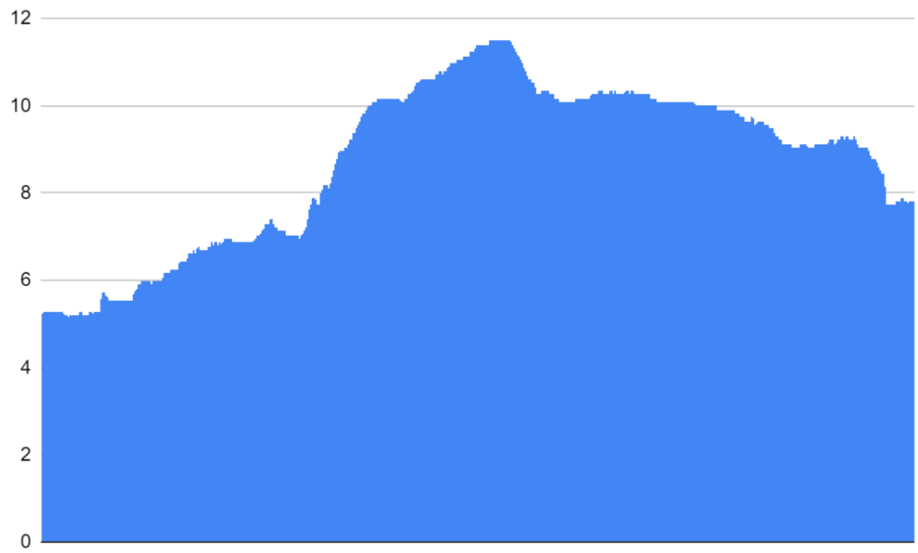

Figure 10. Trip B20 Ambient temperature throughout the journey(^{0}c)

4.2.4 Data Pre-processing

Data Reading and Primary Column Filtering

The raw data, stored as **.csv** files, was read using the Python library, pandas. The key columns of interest include:

- Time [s]
- Velocity [km/h]

- SoC [%]
- Ambient Temperature [°C]

These columns are the primary focus as they have direct relevance to the problem at hand.

Handling Missing Values

Rows containing NaN values in the columns were dropped to maintain data integrity. The DataFrame index was reset afterward.

Feature Engineering

Several new features were engineered from the raw data to aid the machine learning model: [14]

- Trip Time: Derived by converting the time from seconds to minutes.
- Average Velocity: Calculated by taking the mean velocity throughout the trip.
- Initial & Final SoC: Representing the state of charge at the start and end of the trip.
- Average Ambient Temperature: Average temperature value during the entire trip.
- Distance: Estimated using the trip time and the average velocity.
- Discharge Rate: Computed as the difference between the final and initial SoC, divided by the distance.
- Delta SoC: The difference between the final and initial SoC.

	A	B	C	D	E	F	G	H	I	J
1		Trip	Trip Time [min]	Average Velocity [km/h]	Initial SoC [%]	Final SoC [%]	Average Ambient Temperature [Â°C]	Distance (Km)	Discharge Rate (%/km)	delta_SoC
2	0	TripA01	16.815	26.50217047	86.9	81.5	30.76997225	7.427233273	-0.727054046	-5.4
3	1	TripA02	23.54833333	59.90926681	80.3	67.3	31.12757325	23.51272308	-0.552892149	-13
4	2	TripA03	11.175	68.82649269	83.5	75.1	23.33491649	12.81893426	-0.655280683	-8.4
5	3	TripA04	6.871666667	93.64541465	75.1	66.7	24.42395732	10.72500124	-0.783216693	-8.4
6	4	TripA05	22.77666667	32.64633716	66.7	60.2	24.58744275	12.39291232	-0.524493342	-6.5
7	5	TripA06	52.74	52.32310254	85.1	63.4	30.18323432	45.99200714	-0.471821113	-21.7
8	6	TripA07	34.88666667	76.24785506	63.4	34.2	29.1681264	44.33389173	-0.658638321	-29.2
9	7	TripA08	46.76333333	23.38753626	86.6	76.1	27.53847999	18.2279859	-0.576037312	-10.5
10	8	TripA09	30.57166667	22.88518262	74.7	68.6	18.11106138	11.66063624	-0.523127544	-6.1
11	9	TripA10	23.625	41.25678964	69.6	60.9	23.99351651	16.24486092	-0.535553985	-8.7
12	10	TripA11	23.74	44.8118259	87	75.6	27.78451808	17.73054578	-0.642958211	-11.4
13	11	TripA12	27.305	40.98765198	75.5	64.3	27.89970398	18.65279729	-0.600446133	-11.2
14	12	TripA13	11.93166667	22.9926648	85.8	82.9	21.81993436	4.57234687	-0.634247594	-2.9
15	13	TripA14	11.575	25.55016844	82.8	79.9	22.90294846	4.929053329	-0.58834827	-2.9

Figure 11. **Data after Feature Engineering.**

4.2.5 Machine Learning Data Preprocessing

Data Splitting

The entire dataset was divided into:

- Training Set: 80%
- Testing Set: 20%

This facilitates model training on a larger subset and validation on the remaining data.

4.3 Model Construction and Assessment

In the multifaceted realm of artificial intelligence, the architecture and design of a model can significantly utter its efficacy. With the objective of understanding electric vehicle (EV) discharge rates under varying conditions, a neural network model was architected utilizing the Keras library.[13] Further details can be found in Appendix 1B.

Data Training and Validation

The foundation phase involved training our model on a designated training dataset, subsequently subjected to validation processes using a distinct test set. This cyclical process of training and validation ensured our model's robustness, training it for subsequent tests under diverse conditions. Further details can be found in Appendix 1[B].

Effect of Ambient Temperature

To unravel the relationship between ambient temperature and discharge rate, a series of tests were executed. The model's input parameters were modulated, where ambient temperature ranged from 0 to 40 degrees Celsius, while other factors were held constant:

- Trip Duration: 23.55 minutes
- Mean Velocity: 59.9 km/h
- Initial State of Charge (SoC): 80.3%
- Journey Length: 23.51272308 km

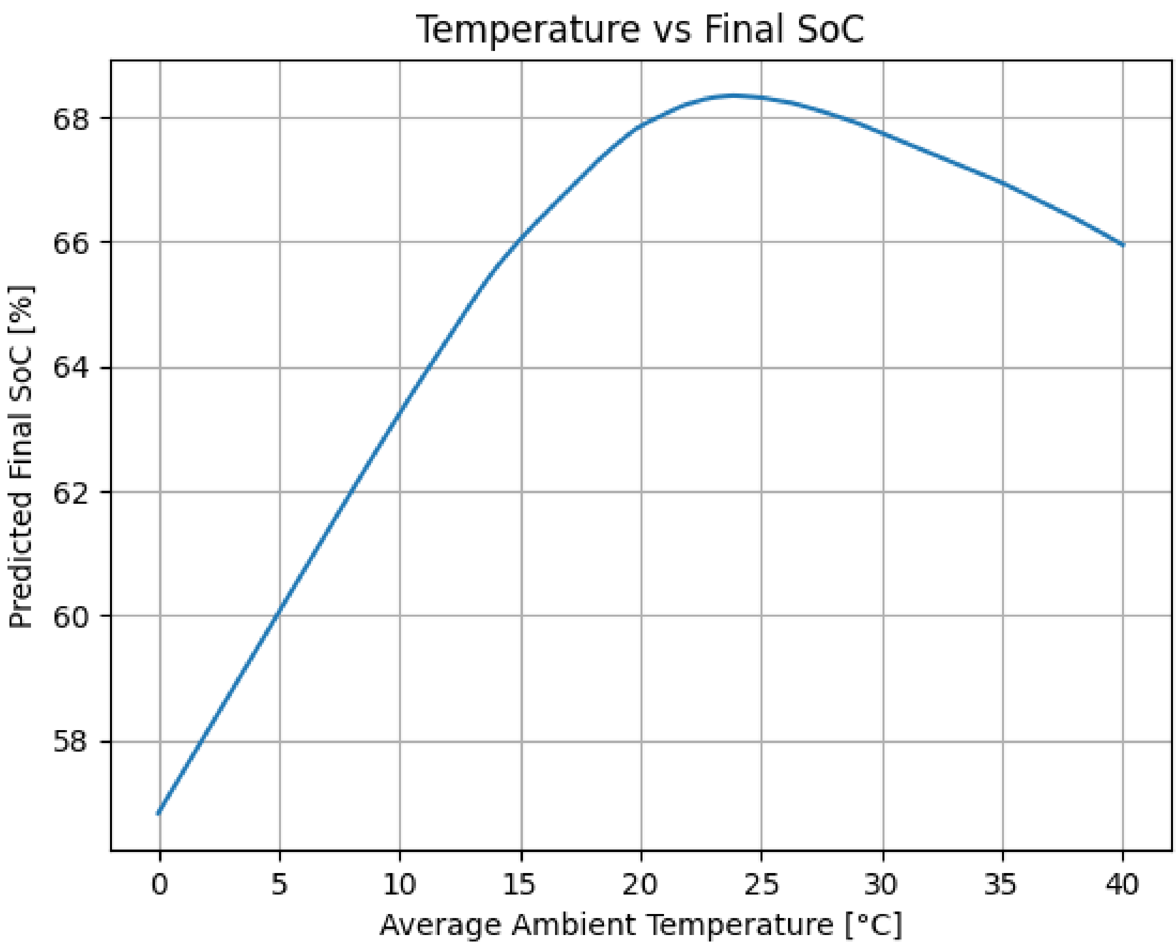

Figure 12. Model generated Final SOC'S by varying Temperature.

The results were interesting. A graphical representation derived from our model distinctly exhibited an inverse correlation between temperature and final state of charge. Specifically, at 20°C, the final SoC hovered around 68%, whereas at 3°C, it dropped to approximately 59%. A pivotal observation emerged post the 25°C mark, where discharge rates observed an upswing. A reasonable rationale for this phenomenon could be attributed to the augmented use of vehicular air conditioning at extreme temperatures.

Effect of Velocity

The next frontier explored was the relationship between vehicular speed and its consequent discharge rate. Here, speed parameters were varied from 20 to 120 km/hr. The following were maintained constant:

- Initial SoC: 80.3%
- Journey Length: 23.51272308 km

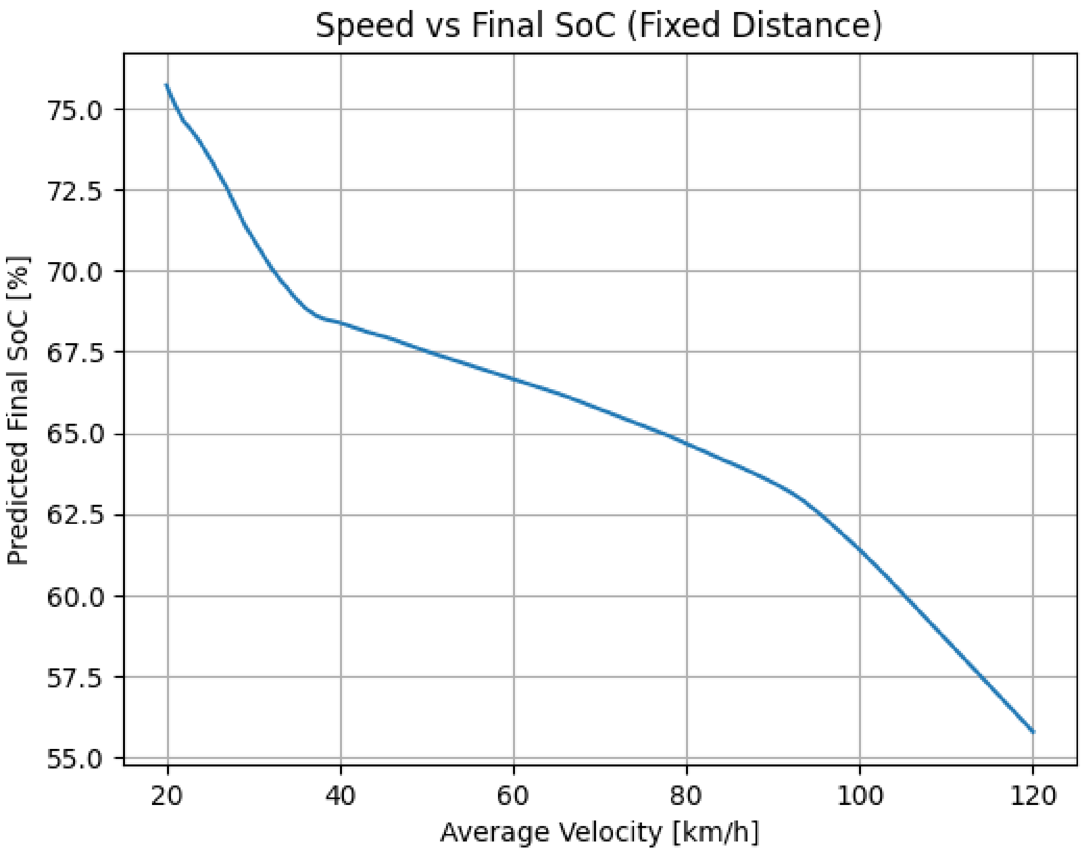

Figure 13. **Model generated Final SOC'S varying Speed.**

With distance as a fixed entity and speed as the variable, our analysis revealed that the EV's discharge rate was minimal at lower velocities but surged with escalating speeds.

Implications and Applications

We can conclude that this data underwent an intensive cleaning process to ensure reliability and accuracy. Our approach incorporated a neural network model to utilize trip characteristics (like average speed, ambient temperature, and initial SoC) to *predict the SoC at the trip's conclusion. Through our model, we were able to visually demonstrate the relationship between ambient temperature, driving speed, and the final state of charge.*

The overarching goal of these rigorous experiments was two-pronged. Firstly, to shed light on how varying road conditions could potentially affect EVs. Secondly, to extrapolate these findings to real-world scenarios by integrating it to our Genetic algorithm, thereby allowing our model to predict EV performance with heightened precision.

4.4 Section 2: Real-world Data Integration and Analysis

4.4.1 Objective

The effort to seamlessly integrate theoretical algorithms with real-world data is the essence of computational research. Our primary goal in this chapter is to supercharge the capabilities of our Genetic Algorithm (Chapter 3) by infusing it with genuine, on-ground data. Such integration not only gives the algorithm with practical relevance but also gives a multitude of real-world applications. Further details can be found in Appendix 1[C].

4.4.2 Harnessing Google API for Route and Charging Stations Data

Our approach to enriching the Genetic Algorithm commenced with the utilization of the powerful Google API. This platform facilitated:[8]

1. Acquiring comprehensive route details between the postcodes FK9 5AL and B4 7XG. Located in Stirling and Birmingham respectively.
2. Extracting pivotal data points such as total route distance, which is 494 km for the trip.
3. Identifying strategic locations of electric vehicle (EV) charging stations along the entire route. Specifically, charging stations were searched every 2 km from the starting point, ensuring a diversity of charging options rather than relying solely on super-chargers. We also refined the chargers by selecting the charger closer to the route i.e., if there are two chargers close to each other through 2 km search then the one closer to the search is chosen.

This technique, captured using the **googlemaps** Python library, allows our system to outline the entire route into manageable segments of 2 km each. Through this, we strategically identify EV charging stations close to the route, fostering the notion of charger diversity.[8]

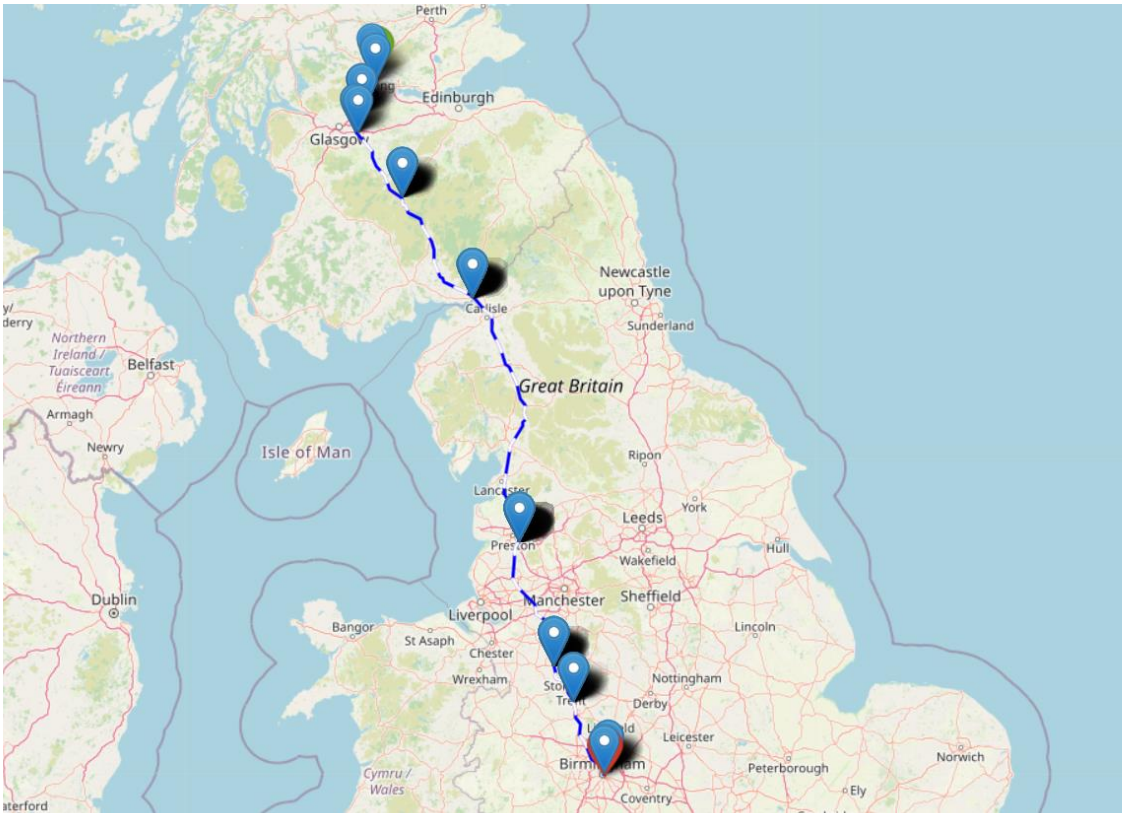

Figure 14. **Location charging stations from Stirling to Birmingham**

Driving Specifications and Charging Dynamics achieved by the Genetic Algorithm

The simulation was premised on a driving speed of 70 km/hr. Commencing the journey from Stirling with an initial state of charge (SoC) of 40%, the EV encounters 11 strategically located charging stations along its route. Our Genetic Algorithm, aiming for optimal charging decisions, selected two of these stations:

1. The first station facilitated charging from 29% to 82%.
2. The subsequent station enabled a charge from 12% to 49%.

Thus, with a total driving time of 6 hours and 59 minutes and an additional charging duration of 49 minutes, the entire journey concluded in 7 hours and 49 minutes. The vehicle's state of charge upon reaching the destination was 8.13%.

4.4.3 Visualization:

We further enriched our analysis with spatial visualization. Utilizing the **folium** library, our selected charging stations are vividly portrayed on an interactive HTML map that is provided with dissertation data. Each marker, representing a station, can be clicked to reveal its specific name. This interactive approach serves dual purposes: enhancing user experience and illustrating the strategic selection by our Genetic Algorithm when infused with real-world data.

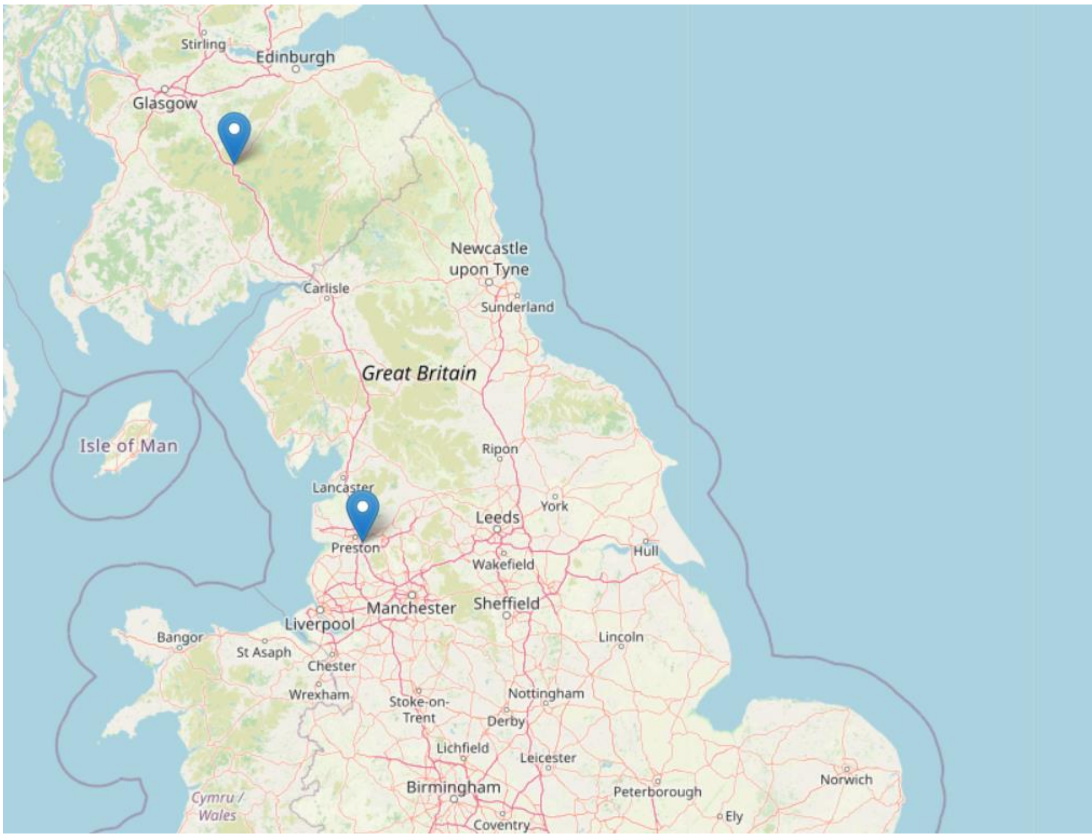

Figure 15. Selected charging stations by the algorithm from Stirling to Birmingham

In conclusion this chapter is a testament to the synergy between theory and practice. By intertwining real-world data, advanced programming techniques, and intelligent algorithmic designs, we've showcased the potential of theoretical constructs when applied practically.

5 Conclusion

5.1 Summary

The transportation sector is witnessing a shift towards Electric Vehicles (EVs), operational cost-effectiveness, and technological innovations. However, a significant challenge faced by EV users is 'range anxiety'—the fear of their vehicle's battery running out before reaching a charging station. The research aims to address this challenge through python computational solutions. The primary objectives include designing an algorithm for efficient EV charging, understanding the impact of external factors like driving speed and ambient temperature on EV performance, and integrating real-world data through Google's API.

The research achievements highlight the development of a Genetic Algorithm (GA) framework that optimized travel time for EVs, a predictive model to forecast an EV's final state of charge, and the integration of real-world data using Google's API. The dissertation's structure involves chapters dedicated to the optimization of EV travel using the GA, the application of this theoretical approach in real-world scenarios, and the role of external factors in influencing EV performance.

The GA's success is evident through its ability to optimize EV travel time, with significant improvements seen over multiple iterations. By introducing varying penalties for charging stops, the GA showcases its adaptability to balance between travel time and charging considerations. Without penalties, the GA minimized travel time by making frequent but quick stops. However, when penalties were applied, the algorithm made fewer stops but charged the vehicle more at each station. This adaptability and robustness make the GA a promising tool for real-world EV charging optimization.

To ensure the research's practical applicability, the study integrates real-world data through Google's API and machine learning techniques. By analysing the impact of road conditions on EV performance, a predictive model was developed using a dataset of 72 trips. The integration of Google's API enabled the extraction of comprehensive route details, charging station locations, and distances, bridging the gap between theoretical constructs and real-world applications.

5.2 Evaluation

In a methodical approach to understanding and addressing the prevailing challenges faced by Electric Vehicle (EV) users, we researched deep into a journey to explore the algorithm development, specifically by using Python programming language and real-world datasets.

Alignment with Objectives:

- **Algorithm Implementation and Optimization:** Objective section 1.2.1
  We go on board on the task of designing an advanced algorithm tailored to determine an efficient charging strategy for EVs. Our Genetic Algorithm (GA) not only showcased promising results by decreasing travel time significantly but also exhibited adaptability by adjusting to penalties introduced during testing. The optimization from an estimated 15 hours to 11 hours and 7 minutes, combined with its ability to harmoniously juggle between charging times and travel efficiency, unequivocally proves the accomplishment of this objective.
- **Data-Driven Insights on EV Performance:** Objective section 1.2.2
  Our commitment to understanding the granular aspects of EV performance led to the creation of a meticulous dataset of 72 distinct trips. Through rigorous data cleaning and refinement processes, we sculpted a predictive model that accurately forecasts an

EV's final state of charge. With factors like driving speed and ambient temperature playing pivotal roles, our model's prowess in predicting the discharge rates cements our success in achieving this objective.

- **Integration of Real-World Data:** Objective section 1.2.3
  Our vision of integrating theoretical constructs with actionable real-world insights found its culmination in our collaboration with Google's API. By fetching details like route distances, charging station locations, and their relative proximities, we've bridged the gap between theory and practice. This seamless integration not only substantiates our achievements in this realm but also fortifies our algorithm's applicability in real-world scenarios.

5.3 Future Work

The research undertaken in this dissertation offers a promising approach to optimising electric vehicle charging using genetic algorithms, as well as integrating real-time data. While the results achieved were fascinating, there remain various avenues for further enhancement and expansion of the work.

1. Using Google API to Extract Real-time Travel Data: One of the limitations faced in the current study was the static nature of travel time used. Leveraging the Google API to extract real-time travel data can make the algorithm much more dynamic. This would account for unexpected travel delays, potentially affecting the travel plan of an electric vehicle, thus offering a more realistic simulation and solution.

2. Integrating Discharge Rate to GA: A crucial component to accurately predict the state of charge for an EV is the discharge rate, we have created a model to predict discharge based on time and real traffic data. By integrating this with the genetic algorithm, along with the extracted time from the Google API, we can enhance the predictability of our model on chapter 4 section 1. This ensures the EV's state of charge predictions are not just based on travel but also on the vehicle's consumption patterns.

3. Data Variety: The research was possibly limited by the variety of data available. Gathering data from different models of vehicles can provide a more comprehensive picture, accounting for variations in vehicle efficiencies, and battery capacities. Moreover, including data from different types of chargers can ensure the model is robust across various charging infrastructures.

4. Real-time Waiting Time: As the adoption of EVs increases, waiting times at charging stations can become a significant factor in route and charging planning. Incorporating real-time data on charger availability and wait times will make the solutions more practical for everyday EV users.

5. Real-time Vehicle Data: Vehicles today come equipped with advanced telemetry that can provide detailed real-time data. By extracting the vehicle's discharge rate in real-time and feeding it into the model, predictions can be continually refined, ensuring the most accurate state of charge prediction at any given point in time.

6. User-friendly Application: Lastly, the real-world implementation and acceptance of such an advanced system will hinge on its user interface. A user-friendly app that can relay complex algorithmic decisions in an intuitive manner will be crucial. The app could include features like real-time updates, predictive analytics on best charging points, and perhaps even cost-saving recommendations based on current energy prices.

In relation to other works in the field, this dissertation stands out in its approach to integrating genetic algorithms with real-time data for EV charging. However, as with any research, there's a continuum of improvements. By building on the above-mentioned future work avenues, this research has the potential to set a new benchmark in the realm of intelligent EV charging systems.

References

[1]. B. Schott, "Range Anxiety," *The New York Times*, 15-Jan-2009. [Online]. Available: https://archive.nytimes.com/schott.blogs.nytimes.com/2009/01/15/range-anxiety/

[2]. U. Eberle and R. von Helmolt, "Sustainable transportation based on electric vehicle concepts: a brief overview," *Royal Society of Chemistry*, 2010. [Online]. Available: https://www.researchgate.net/publication/224880220_Sustainable_transportation_based_on_electric_vehicle_concepts_A_brief_overview

[3]. M. Backstrom, "Comments of Southern California Edison Company on the California Public Utilities Commission Staff's White Paper, Light-Duty V," 2009, p. 4. [Online]. Available: https://web.archive.org/web/20100629060851/http://www.cpuc.ca.gov/NR/rdonlyres/8B8F7624-1DF1-426B-847A-74255D3484D6/0/SCE_Comments_on_LDV_Electrification_White_Paper.pdf

[4]. D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994. [Online]. Available: http://cobweb.cs.uga.edu/~potter/CompIntell/ga_tutorial.pdf.

[5]. M. Mitchell, *An Introduction to Genetic Algorithms*, Cambridge, MA: MIT Press, 1996. ISBN: 9780585030944. Available: https://www.boente.eti.br/fuzzy/ebook-fuzzy-mitchell.pdf

[6]. "Tesla Model 3 puts Tritium's new 75kW fast charger to the test," The Driven, 28 January 2021. [Online]. Available: https://thedriven.io/2021/01/28/tesla-model-3-puts-tritiums-new-75kw-fast-charger-to-the-test/.

[7]. M. Steinstraeter, J. Buberger, and D. Trifonov, "Battery and Heating Data in Real Driving Cycles," IEEE Dataport, 19 October 2020.Available: https://dx.doi.org/10.21227/6jr9-5235.

[8]. "Google Maps Platform | Google Developers," Google Developers. [Online]. Available: https://developers.google.com/maps.

[9]. J. Holland, M. Mansur, N. Muller, and A. Yates, "Are there environmental benefits from driving electric vehicles? The importance of local factors," *American Economic Review*, vol. 106, no. 12, pp. 3700-3729, 2016. [Online]. Available: https://www.aeaweb.org/articles?id=10.1257/aer.20150897.

[10]. Battery University, "BU-409: Charging Lithium-ion." [Online]. Available: https://batteryuniversity.com/article/bu-409-charging-lithium-ion.

[11]. "DEAP documentation," DEAP: Distributed Evolutionary Algorithms in Python. [Online]. Available: https://deap.readthedocs.io/en/master/.

[12]. M. Brett, "Linear interpolation — Tutorials on imaging, computing and mathematics," [Online]. Available: https://matthew-brett.github.io/teaching/linear_interpolation.html.

[13]. "Keras guide," TensorFlow. [Online]. Available: https://www.tensorflow.org/guide/keras.

[14]. T. Gupta, "Data Preprocessing in Python for Machine Learning with working code example," Towards Data Science, [Online]. Available: https://towardsdatascience.com/data-preprocessing-in-python-b52b652e37d5.

[15]. B. M. "haversine 2.8.0," PyPI. [Online]. Available: https://pypi.org/project/haversine/.

[16]. M. Breuss, "Python Folium: Create Web Maps From Your Data," Real Python, [Online]. Available: https://realpython.com/python-folium-web-maps-from-data/.

[17]. J. Jordán, J. Palanca, P. Martí, and V. Julian, "Electric vehicle charging stations emplacement using genetic algorithms and agent-based simulation," Expert Systems with Applications, vol. 197, pp. 116739, 2022. [Online]. Available: https://doi.org/10.1016/j.eswa.2022.116739

[18]. Z. Qu, J. Song, Y. Liu, et al., "Optimization Model of EV Charging and Discharging Price Considering Vehicle Owner Response and Power Grid Cost," J. Electr. Eng. Technol., vol. 14, pp. 2251–2261, 2019. [Online]. Available: https://doi.org/10.1007/s42835-019-00264-0.

[19]. Tesla, "Go Anywhere," [Online]. Available: https://www.tesla.com/trips.

[20]. V. Fulber-Garcia, "Deterministic and Stochastic Optimization Methods," Baeldung, November 4, 2022. [Online]. Available: https://www.baeldung.com/cs/deterministic-stochastic-optimization.

Appendix 1

[A]. Genetic Algorithm (GA)

A genetic algorithm is a heuristic optimization technique inspired by the process of natural selection. It involves populations of candidate solutions (called individuals) evolving towards better solutions. The evolution starts from a population of completely random individuals and occurs in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected (based on their fitness) and modified to form a new population. The new population is then used in the next iteration of the algorithm.

EV Charge Strategy Problem: we are developing a charging strategy for an electric vehicle (EV) that plans to travel a certain distance. I used a Genetic Algorithm (GA) to find the most efficient way to charge the vehicle at multiple stations along its route, so as to to ensure an EV can travel a specific distance with minimal total time (driving + charging).

Code explanation

1. **Initialization of Variables**:

```
# Assigning given values

initial_soc = 0.4  # initial state of charge

total_distance = 500  # total travel distance in miles

average_speed = 50  # average speed in mph

discharge_rate = 0.004  # discharge rate in SOC per mile

min_soc = 0.09  # minimum SOC
```

We start by setting up some basic parameters: initial state of charge, total distance to travel, average speed, discharge rate per mile, and minimum state of charge.

2. **GA Hyperparameters**: These are parameters that guide the evolution of the genetic algorithm, such as population size, mutation probability, and crossover probability.
3. **Interpolate Charging Time Function**:

```
def interpolate_charge_time(start_soc, end_soc):

   ...
```

This function takes a starting and ending state of charge and calculates the charging time using interpolation. If it's impossible to reach the desired SOC, a high penalty is returned. The primary logic involves locating nearest indices for the provided SOCs and linear interpolation.

4. **Fitness and Individual Class Definitions**: Two classes are defined: 'FitnessMin' which aims to minimize a given objective, and 'Individual' which represents a potential solution.
5. **Toolbox**: **toolbox** is an object that holds functions and parameters that the genetic algorithm uses. Various methods are registered in this toolbox such as the initialization of individuals, mutation, mating, and evaluation functions.
6. **Fitness Evaluation Function**:

```
def evaluate(individual):
    ...
```

This function calculates the total travel time based on an individual's charging plan. It factors in driving time, charging time, and penalties for suboptimal states of charge.

7. **Mutation and Crossover**: **mutate** function introduces randomness by occasionally flipping charging decisions, while **mate** combines attributes of two individuals.
8. **Main Genetic Algorithm Function**:

```
def main():
    ...
```

This function initializes the population, defines statistics for the evolution process, and runs the evolution using the library algorithm, which is a basic evolutionary algorithm. It then prints the results of the best individual.

9. **Execution**:

```
pop, log, hof = main()
```

The main function is executed to run the genetic algorithm and obtain the best individual's charging strategy.

10. **Output Formatting**:

    The results, such as charging stations, charging times, and final state of charge, are presented in a structured DataFrame for clarity.

[B]. Machine Learning:

We created a model to undertake a project, aimed at predicting how varying road conditions, such as speed and ambient temperature, influenced the state of charge (SoC) of electric vehicles (EVs). We created a model based on a rich dataset containing 72 different trips, seeking to discern patterns that would allow us to make accurate predictions.

This data underwent an intensive cleaning process to ensure reliability and accuracy. Our approach incorporated a neural network model to utilize trip characteristics (like average speed, ambient temperature, and initial SoC) to *predict the SoC at the trip's conclusion. Through our model, we were able to visually demonstrate the relationship between ambient temperature, driving speed, and the final state of charge.*

Code Explanation:

1. **Dependencies**:
    - Essential Python libraries and frameworks were imported to assist in the data processing, analysis, and modeling.
2. **File Acquisition**:
    - A list of all CSV files from a specified directory is created. These files contain the trip data.[7]
3. **Feature Extraction Function (extract_features)**:
    - The **extract_features** function processes individual trip CSV files.
    - It focuses on specific columns of interest, such as 'Time', 'Velocity', 'SoC', and 'Ambient Temperature'.
    - It calculates various metrics for the trip: average velocity, the initial state of charge, the final state of charge, and average ambient temperature.
4. **Creating Master Dataset**:
    - Every file from the directory is processed using the **extract_features** function.
    - A master dataframe (**master_df**) is compiled, which aggregates the processed data from all trips.
5. **Additional Feature Engineering**:
    - The distance for each trip is computed using the formula: **Distance = Time * Velocity**.
    - A discharge rate is calculated to understand the rate at which the vehicle's battery gets depleted concerning the distance traveled.
    - The difference between the final and initial SoC (**delta_SoC**) is also computed.
6. **Data Preparation for Modeling**:
    - The required features ('Trip Time', 'Average Velocity', 'Initial SoC', 'Ambient Temperature', 'Distance') are extracted from the master dataset.
    - The target variable (change in SoC, **delta_SoC**) is isolated.
    - The dataset is split into training and test sets. 20% of the data is reserved for testing the model.

- Data scaling is performed to normalize feature values, essential for neural network performance.

7. **Neural Network Model**:
    - A neural network model is defined using the Keras framework. The model comprises an input layer, two hidden layers, and an output layer.
    - The model is then compiled using the Adam optimizer and Mean Squared Error as the loss function.
    - The model is trained using the training data.
8. **Model Prediction**:
    - The model is used to predict the change in SoC (**delta_SoC**) for the test set. This prediction is then adjusted with the initial SoC to get the predicted final SoC.
9. **Visualization**:
    - Two visual analyses are presented:
        1. **Temperature vs. Final SoC**: The predicted SoC is plotted against a range of ambient temperatures to visualize its relationship.
        2. **Speed vs. Final SoC**: Similarly, the predicted SoC is plotted against varying speeds to deduce its relationship.

[C]. Bridging Real-world Data and Genetic Algorithms

The quest for practicality in scientific inquiry necessitates real-world data integration. In the domain of artificial intelligence, algorithms benefit immensely from accurate, real-world inputs, enriching their decision-making process. Our project strives for an innovative solution that optimizes Tesla Model 3's travel time, leveraging real-world route and charging station data.[8]

Code Description:

1. **Initialization & Importing Modules:** We start by importing necessary modules: **googlemaps** to interface with Google's vast geographic and place data, **pandas** for data structuring and analysis, **folium** for map visualizations, and **polyline** for route decoding. Additionally, mathematical operations are facilitated using the **math** module.[8]
2. **Google Maps Setup:** We initialize our connection to Google Maps API by establishing a client using our API key.[8]
3. **Route Segmentation:** The function **get_route_segments** splits a given route (from start to end) into smaller segments of a specified length (default is 2 km). These segments are particularly useful in locating charging stations at regular intervals.[8]
4. **Distance Calculation:** The **haversine_distance** function calculates the distance between two geographical points (given their latitudes and longitudes) using the Haversine formula, which is crucial for pinpointing the nearest charging stations.[15]
5. **Locating EV Charging Stations:** In **get_ev_charging_stations**, we retrieve the charging stations near the segmented endpoints of our journey. Leveraging Google Places API, we find the closest station to each segment endpoint and filter out repetitive stations (those already considered in a previous segment).[8]
6. **Testing & Data Preparation:** We initiate a start and end location using postcodes and extract charging stations along the route. Next, the charging rate data is provided, which is converted into a DataFrame for easy manipulation.
7. **Map Visualization:** We decode the retrieved route polyline to derive the geographical coordinates. Using Folium, a comprehensive map is generated displaying the journey, marking both the start and end points, and populating it with the detected charging stations.[16]
8. **Setting Constants & Hyperparameters:** Lastly, various constants like **initial_soc**, **average_speed**, and **discharge_rate** are set up, followed by genetic algorithm hyperparameters like **POPULATION_SIZE**, **MUTATION_PROB**, etc. These are crucial for the subsequent genetic algorithm's operation, enabling the fine-tuning of its performance.[11]

Appendix 2 – User guide

Using the Software

[A] Code Format:

- **File Type:** The software is encapsulated within a **.ipynb** (Jupyter Notebook), or py(Python) file for user-friendly execution and visualization.
- **Access:** This file can be directly downloaded from the University of Stirling repository.

[B] Input Specifications:

- **Start and End Postcodes:** Input these to define the start and end points of your trip. It allows the software to understand your journey's commencement and conclusion points.
- **Initial State of Charge (SoC):** This essential parameter helps the algorithm understand the initial state of charge of your electric vehicle.
- **Average Speed:** Clarifying this parameter assists in making predictions regarding energy consumption based on your driving pattern.
- **Penalty for Stops:** By setting a penalty for stops, users can influence the software's decision on how frequently it should recommend charging. Default penalty values have been pre-defined but can be tailored for customized results.

[C] Expected Outputs from the Software:

- **Optimal Journey Plan:** This plan provides users with a detailed travel plan of the journey, suggesting optimal charging stations.
- **Charging Duration:** The software informs about the exact duration required for charging at each station.
- **State of charge on Arrival:** Gives information about the state of charge upon reaching destination point.
- **Interactive Route Visualization:** Using Python libraries, the software offers an interactive map detailing all charging stations and the proposed route.

[D] Data Preparation & Machine Learning Integration

- **Dataset Requirement:** To achieve outputs, the correct dataset must be fed into the machine learning model.
- **Dataset Source Configuration:** Users need to modify the source folder in the code to correctly link the dataset.
- **Cleaning & Preparation:** Once sourced, rest easy! All data cleaning, preprocessing, and machine learning preparation steps are embedded within the code itself. Thus, ensuring a seamless and hassle-free experience

[E] Troubleshooting & Tips:

- **API Key:** Chapter 4, Section 2 is reliant on the Google Maps API. You'll need to provide your unique key due to privacy protocols. But the integration is straightforward and documented below.
- **Running Code:** GA and GA with maps codes are designed for seamless execution without necessitating third-party APIs. However, always ensure you have all the necessary Python libraries installed.

Appendix 3 – Installation guide

[A] Sourcing Codes & Data from University Repository:

- The required Python scripts and datasets have been uploaded and stored at the University of Stirling repository.

[B] Setting up Google Maps API: A Step-by-Step Guide

1. **Access Google Cloud Console:** Begin by navigating to **https://console.cloud.google.com/**.
2. **Project Creation:** After signing in, select or create a new project tailored for this software.
3. **Activating the API:** Within the console, proceed to "APIs & Services > Library". Under the Maps segment, initiate the "Maps JavaScript API".
4. **Securely Generating API Key:** To maintain privacy and security, produce your unique API key. Navigate to "APIs & Services > Credentials", select "+ CREATE CREDENTIALS", and opt for "API Key". Safeguard this key
5. **Seamless Integration:** Integrate this key within the **.ipynb** notebook. Ensure its proper placement, encapsulated within quotation marks.

[C] GitHub Repository:

For additional information, datasets, or any clarifications, visit my GitHub repository at https://github.com/workingbetter. Feel free to raise issues or send direct messages for further details.