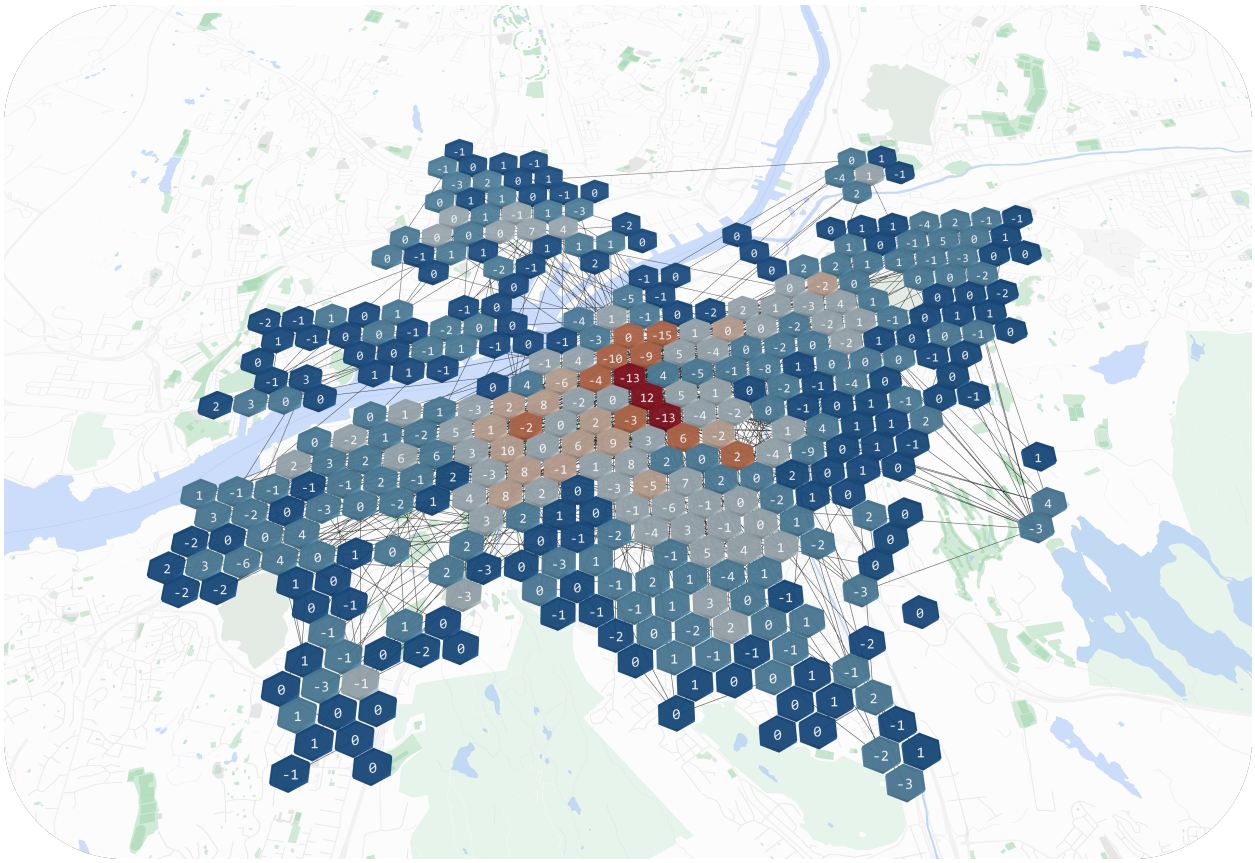
CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG



Fleet Management Optimisation with Spatio-Temporal Demand Forecasting in MaaS for Free-floating Micro-mobility

Implementing and evaluating data driven models to forecast future demand of micro-mobility vehicles in cities

Bachelor's thesis in Computer Science and Engineering

Oscar Almström, Erik Carlsson, Daniel Cronqvist, Max Karlsson, Fredrik Lilliecreutz, Alexander Viala Bellander

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021
www.chalmers.se

Bachelor's thesis 2021

Fleet Management Optimisation with Spatio-Temporal Demand Forecasting in MaaS for Free-floating Micro-mobility

Implementing and evaluating data driven models to forecast future demand of micro-mobility vehicles in cities

CHALMERS

Department of Computer Science and Engineering
Division of Data Science and AI
Research Group 59
Chalmers University of Technology
Gothenburg, Sweden 2021

Fleet Management Optimisation with Spatio-Temporal Demand Forecasting in MaaS for Free-floating Micro-mobility
Implementing and evaluating data driven models to forecast future demand of micro-mobility vehicles in cities

Supervisor: Emil Carlsson, Department of Computer Science and Engineering
Examiner: Devdatt Dubhashi, Department of Computer Science and Engineering

Bachelor's Thesis 2021
Department of Computer Science and Engineering
Division of Data Science and AI
Research Group 59
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Spatio-temporal delta in demand for Gothenburg in geospatial units Uber H3 calculated by a stationary Markov matrix with displayed traversals and k-means geospatial clusters.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2021

Abstract

In recent years, micro-mobility services have grown rapidly. Companies such as Voi, Bolt and Lime are front figures in this development. Utilising their resources efficiently through fleet management has been an essential factor in reaching profitability. Gathering data is becoming more critical for companies, enabling them to use data science and mathematical models to leverage their business. This report aims to examine the opportunity to use different data-driven models to predict future demand for micro-mobility services. Voi Technology provided real market data for this paper, limited to Gothenburg. All geospatial data was aggregated to geospatial units defined by Uber's H3 spatial index. Furthermore, the research focused on the following methods; Markov properties, Time Series Analysis and Poisson processes. These methods were implemented in the programming language Python. The Markov and Prophet models provided solid demand predictions. However, due to sparse data representation, for some areas of the city, the data was clustered to give more accurate forecasts at the cost of geospatial granularity.

Comparing the two, the Prophet models gave better results alone, while the Markov models instead gave insight into how the supply moves around in the market. With further work, the Markov model could prove favourable for a fleet operator during rebalancing due to its ability to track the vehicle flow and predict lack and surplus of supply. The data indicated significant seasonality effects and sporadic behaviour. FBprophet showed decent results in analysing those characteristics, using a sliding window technique as a significant contributor. Supporting FBprophet with several features improved the prediction, where rain stood out as the most impactful feature. The Poisson proccess model interprets demand as non-homogeneous and stochastic with inherent temporal randomness. While the theory behind the Poisson process model seems relevant, the results remain inconclusive. FBprophet performed the best demand prediction in contrast to the other models. However, further research could contradict this, and there is room for deeper exploration. Onwards, Neural Networks and Deep Learning are also exciting subjects for further research in demand forecasting.

Keywords: data analysis, micro-mobility, MaaS, demand forecasting, Markov chains, time series analysis, fleet optimisation, FBProphet, Poisson processes.

Sammandrag

Under de senaste åren har marknaden för mikromobilitetstjänster vuxit kraftigt och företag som Voi Technology, Bolt och Lime är ledande inom området. Att använda sina resurser effektivt genom att optimera sin fordonsflotta, är en viktig del för att uppnå lönsamhet. Insamling av data har blivit en vital del för företagen, då det möjliggör användande av data science och matematiska modeller för att förbättra sin verksamhet. Detta arbete grundar sig i att undersöka användandet av datadrivna modeller för att förutspå efterfrågan av mikromobilitetstjänster. Voi Technology har bidragit till studien genom att dela med sig marknadsdata från år 2020. All geospatiala datan används genom Uber H3s spatiala index, vilket delar in kartor i hexagoner. Studien fokuserade framförallt på tidsserieanalys, Markovegenskaper samt Poissonprocesser, vilka implementeras främst i programmeringsspråket Python. Man bör beakta att 2020 var ett unikt år vilket kan ha påverkat resultatet i studien. Trots detta resulterade Markovkedjor samt tidserieanalysen goda resultat i att förutspå efterfrågan. På somliga geografiska platser var glesheten av datan ett tydligt problem som hindrade efterfrågeprognosen. Åtgärden i att klustra datapunkterna resulterade i bättre prognoser men medförde en sämre geografisk precision.

När man jämför tidsserieanalysen med Markovkedjorna, gav tidserieanalysen genom FBprophet bättre efterfrågeprognoser. Markovkedjorna visade däremot tecken på goda användningsområden utanför efterfrågeprognos, så som i hur cykelflödet förväntas te sig i marknaden. Datan visade på signifikanta säsongseffekter samt sporadiskt beteende. För att anpassa prognosen för detta användes en metod med glidande medelvärden, vilket påverkade resultatet positivt. För att förbättra resultatet ytterligare togs andra mätvärden med så som väder, där regn visade sig ha störst effekt på efterfrågan. När dessa mätvärden nyttjades visade resultatet på en bättre prognos av efterfrågan. Ett försök gjordes med Poissonprocesser, men resultatet var bristfälligt. I sin helhet gav FBprophet bäst resultat bland de modeller som har undersöks. Det ska dock understrykas att vidare undersökning kan motbevisa detta och det finns behov av vidare forskning inom området. Vidare finns det andra modeller som ej tagits upp i denna studie som hade kunnat ge goda resultat, som till exempel djupinlärning och artificiella nätverk.

Nyckelord: dataanalys, mikromobilitet, MaaS, efterfrågeprognos, Markovkedjor, tidsserieanalys, optimering, FBProphet, Poissonprocesser.

Acknowledgements

The team would like to inform the reader that prior to this thesis, none of the authors had any experience of Machine Learning. Although inexperienced, yet now slight wiser, we are thankful for the deep inspiration this project has prompted. We would like to express our deep gratitude for all the help we have received from our supervisor Emil Carlsson and our contact person Filip Lindvall from Voi Technology. They have guided us through the process and given us invaluable feedback. The completion of the bachelor thesis would not have been possible without their help. Additionally, we would like to thank Jonas Rundberg at Voi Technology for entrusting the team with this challenge that would not have been possible without him.

Gothenburg, June 2021

Contents

List of Figures

List of Tables

1
Introduction

The following sections presents the background of the project, as well as what the purpose is and in which scope the research will narrowed down to. In addition to this, it also discusses relevant ethical aspects of the project, and how these will be considered.

1.1 Background

In recent years the transportation industry and general mobility have evolved significantly in response to electrification and, most recently, the COVID-19 pandemic [1][2]. Mobility-on-demand (MoD) has become increasingly popular with companies such as Uber, Bolt, and Voi Technology offering on-demand shared: multi-modal and or micro-mobility transportation methods. MoD in a Mobility-As-A-Service (MaaS) environment allows users to access, on-demand, a diverse menu of transport options through a single payment channel.

Light electric vehicles (LEVs) designed for a single person have become a considerably more conventional method of transportation as mobility sharing companies that provide such vehicles and services have emerged [3]. Common LEVs provided by shared micro-mobility companies in MoD in MaaS environments are electric bicycles, electric scooters, and electric mopeds. Expectations are that the growth of electric vehicles in transportation is yet to flourish [4]. However, the micro-mobility industry is already heavily electrified with large fleets of LEVs. Seemingly, MaaS could pave the way for a more sustainable and reliable mobility system helping cities and states become carbon neutral [5].

However, micro-mobility providers in this industry are yet to prove profitable [6]. Increased utilisation of shared LEVs through improved vehicle fleet management by accurately predicting customer demand could prove helpful in increasing company profit margins, in addition to providing users with improved vehicle accessibility and reliability. Predicting user demand requires finding spatial and temporal patterns in user mobility behaviour. Spatio-temporal demand forecasting in MoD in MaaS environments for free-floating (non-station based, park anywhere) micro-mobility providers could thus accelerate the progress towards carbon-neutral goals, in addition to stabilising the business models of fleet operators. Lastly, through improved vehicle accessibility, more people could see the service as a reliable and future proof method of transport.

1.2 Purpose

This paper aims to develop deeper insight into demand forecasting for free-floating micro-mobility services, focusing on applying time series analysis and Markov chain models on quantitative market data. These insights will be used to understand the challenges associated with demand forecasting in a MoD and MaaS environment, emphasizing the ability to forecast spatial and temporal demand. Moreover, this thesis will investigate solutions to improve the examined models and act as a basis for future work.

1.3 Delimitations

The scope includes evaluating different model types, evaluating data, defining the right features and validating the chosen models. The models are general and not suited for predicting extreme values, such as extreme weather conditions, but tries to account for normal weather conditions such as rainy and/or cold days. Other extreme events like concerts or other activities that might affect the prediction are not considered either. These extreme events could have a big impact on the predictions but were too complicated to consider given the project's time frame. The city's infrastructure was not taken into account either for the same reason. The infrastructure of a city could increase or decrease the demand significantly. For example, other public transportation and sufficient bicycle roads affect the demand for micro-mobility services.

The scope excludes Voi Technology's earnings, costs and profitability. This is because the focus should be on the demand forecast and the different technical aspects of the problem. A detailed demand forecast will not extended beyond 24 hours. The only time the thesis will forecast beyond 24 hours is to explore a pattern or test a model. The project only focuses on Gothenburg, Sweden, and the created model may only work for this particular city. However, we recognise that fleet operators most likely operate in multiple cities and the value of a model that could predict demand for any market might be very significant. Creating a model that could operate in multiple cities is considered too time-consuming for this project and very overwhelming when considering the team's lack of experience. Patterns in Gothenburg could be very different from the patterns in another city, which would make it hard to develop a viable multicity, generalised model.

1.4 Social and Ethical aspects

Following the guidelines for a bachelor thesis at the Chalmers University of Technology, four aspects regarding ethics are evaluated [7]: The relevance and value the project provides, whether the method can avoid ethical problems or not, what contribution or harm a successful outcome might provide and identify the expected affects by this project's outcome.

1.4.1 Relevance and Value

The scope of the project was to investigate and research the usage of a *Fleet Management Model* to predict the demand based on prior electrical scooter usage data. The concept has been researched in similar areas such as *Ride Sharing*[8]. However, free-floating mobility introduces new views that could require further assessment and investigation. Furthermore, there is no obligation to provide direct value for the partner company Voi Technology. This aligns well with our aim to investigate this area and hope to deepen the available knowledge for further research on this topic.

1.4.2 Method and Ethics

The main concern is privacy and personally identifiable information since personal data is collected using the mobility service. To avoid any breach of privacy, the data delivered by Voi Technology is cleansed from any direct personally identifiable information. Furthermore, no tracking data will be released in the report to avoid any indirect assembling of personally identifiable data in part or whole.

Since the movement of people is both tracked and predicted, it is of high importance that the data is handled with caution. The freedom of movement is vital, and by predicting demand, one can interpret that as predicting movement. However, this project intends to enable more rides and allow more people to travel and exercise their right of movement, not limit or control it.

1.4.3 Outcome and Ethics

As mentioned in Section 1.4.1, there are no expectations on the project to deliver a profitable model for the partner company. It can, however, be seen as a great goal to have in mind to push the project forward towards some concrete deliverable in the end. With research being our focus, the hope is that the outcome can indicate that *Fleet Optimisation with demand forecasting* can provide value for society by improving resource allocation (LEVs in this case) by predicting demand. Professors at the University of INSEAD published a special issue on the usage in humanitarian operations, where large fleets of aid vehicles were deployed in critical areas [9].

Another aspect is that any model created will be biased and tailored to fit the data provided. If a certain area lacks data, most of the models will either poorly or rarely predict any demand in that area. Similarly, areas with heavy traffic can be expected to be appointed a higher demand prediction, potentially resulting in self-fulfilling demand (people will travel with scooters where they are placed). As such, a model which grows over time and constantly evolves to assist the needs of people should be our aim, not a model that controls the need or demand.

1.4.4 Expected Area of Effect

Depending on how Voi Technology receives the project's outcome, they might revisit their current demand forecasting models. Once again, we as a group do not expect the resulting model to shift the landscape of fleet management, but it was a shared

interest between Voi Technology and the research group to explore the topic, and we hope this project may be of use. A hypothetical consequence for the residents of micro-mobility markets could be an increase in usage of their electric fleet since the demand is more efficiently meet. If this results in more or fewer LEVs present in certain cities or streets are left undetermined, the intent is that places where LEVs are appreciated (and used) are supplied to meet the demand and interest in the service.

1.4.5 Closing Thought on Ethics

This project has some ethical dilemmas, especially in what the received data contains and how it is handled. However, we feel that precautions can be made to ensure that the integrity of the data is held throughout the project. As the intent is to use *Machine Learning*, bias towards the data (such as stimulating existing behaviour) can not be avoided and should be considered when drawing conclusions from this thesis.

1.5 Thesis Outline

Chapter 1 introduces the background, purpose and scope of the project. Chapter 2 provides theory about the different main topics of the report, e.g time series models and the mathematical models that have been used. Chapter 3 gives a brief introduction to different terms that are used and gives them context. Chapter 4 presents the process and what was done to achieve the purpose. Chapter 5 presents the results of the report and briefly discusses the different findings under each subsection. Chapter 6 presents our final discussion and conclusions, discussing future works and how well the purpose of the project was met.

2
Theory

In the following sections, the necessary theory behind the project is introduced. This includes theory about fleet optimisation, mathematical models and the forecasting models that were used.

2.1 Supply and Demand

A market is where buyers and sellers can meet to trade [10]. The size of a market can vary from a minor store to a large geographic area. It could be both a psychical market where traders meet or an online market where they psychically never interact. What every market has in common is a supply-side and a demand side. The supply side represents what the seller offers and to what price, while demand represents the buyers and what they want to pay for and for which price. For a specific product, the quantity of demand may vary depending on the price. There could be 100 individuals interested in a product, but only 60 of them are customers that are willing to pay the price the suppliers are willing to sell for. This means there are 40 individuals lost in demand due to the price. However, the supplier considers modifying the price to reach the optimal quantity to maximise the total revenue. When the market prices are stable and the quantity of demand is equal to the supply quantity, it is called market equilibrium.

Moreover, there are also non-price determinants for the demand for products[11]. That could be seasonality, complementary goods, branding and accessibility, to name a few. Looking at Micro-Mobility Services, it could be a higher demand for LEVs (micro-mobility) during the summer than in the winter, even if the price would be the same. Complementary goods of a scooter could be other forms of transport such as taxi or bus. The accessibility can be seen as how much effort it requires to buy a product. This could mean the distance to a product, its user-friendliness, and availability. As mentioned earlier, it can be a loss in demand due to the price, but there can also be a loss in demand due to non-price determinants. Since the data in this report does not represent moving prices, the report will focus on non-price determinants for demand and its fluctuations.

2.2 Fleet optimisation

Optimisation in mathematics is about selecting the best alternative with regard to some criterion. Fleet optimisation is about finding the best outcome, from a

fleet operators perspective, for the fleet of vehicles based on a set of operational alternatives.

Fleet optimisation contains various challenges that big data analysis may solve since a majority of *established companies and startups in the on-demand mobility space are not performing high-end logistical work*[12]. By collecting data on customer and user behaviour with data points on the usage of vehicles and smartphone applications, inferences may be drawn to calculate, map, and/or predict, for example, future demand, maintenance, charging, efficient routing, or dynamic pricing [12] [8]. This may introduce new opportunities for optimising different operational tasks and may contain the answers for improved service accessibility and reliability.

- **Rebalance optimisation** - When people use free-floating vehicles, they will occasionally take them to areas with lower demand. Mobility service providers want their vehicles utilised as frequently as possible to maximise profits, and consumers want assurances that they can quickly and reliably access a vehicle [13]. With rebalancing, one may relocate the vehicles to higher demand areas to fulfil the aforementioned.
- **Predictive maintenance optimisation** - Vehicles can: break, get stolen, or through other means end up in an unusable state; being able to predict when a vehicle will require maintenance can be of help.
- **Battery Swap optimisation** - Predicting when and where to allocate vehicle re-charge/battery-swapping resources could increase profit margins. Especially for fleet operators with electric scooters with swappable battery technology.
- **Route optimisation** - Optimising routing with or without stations or points of interests for vehicle re-charge/battery-swapping operations could reduce CO_2 emissions and reduce the rebalancing trip time, consequently freeing up resources. Alternatively and additionally, routing for vehicle pick up and deployment could reduce costs and carbon emissions.
- **Task priority optimisation** - Finding a way to balance and prioritise the above-listed issues and other operational tasks could help companies reach operational excellence.

There may or may not exist an industry-specific solution to the above-written challenges that would fit all service providers. Different companies may require unique solutions, depending on a variety of factors. Examples of factors include, however, not limited to: the type of vehicles they provide, market-specific regulatory frameworks, and or operational business logic.

2.2.1 Rebalancing

Utilisation is the fraction of time vehicles are in use by customers, and to improve utilisation in a free-floating mobility system; fleet operators can rebalance their vehicles.

Rebalancing is the action of relocating vehicles from oversupplied to undersupplied areas [14]. Relocating vehicles from one area to another may seem trivial. However, deciding which vehicles to relocate and to where is a data scientific logistics problem.

Supplying consumers with easy access to an on-demand free-floating fleet of

vehicles means that, occasionally, consumers will end their trips in less desired areas from a fleet operators perspective. Ideally, fleet operators want their vehicles in areas where they are likely utilised. In lower-demand areas, a Vehicle's Turnaround Time (VTAT) could be longer than in higher-demand areas, thus lowering the utilisation of the vehicle and, in a larger perspective, the fleet.

Customers can organically and/or through incentives (incentivised customer rebalancing) rebalance vehicles enabled by, for example, a dynamic pricing model, offering cheaper fares when travelling from oversupplied areas to undersupplied areas with higher demand. Additionally, a logistics firm or in house logistics may also rebalance vehicles.

Deciding which vehicles to rebalance is, however, only half the problem. Since the purpose of rebalancing is to capture demand, one must know the higher demand areas in advance. Thus, successful and reliable rebalancing requires demand forecasting.

2.2.2 Demand forecasting

As the aforementioned describes, for successful vehicle rebalancing, a demand forecast may be preferential [15]. If the demand in a particular area was known in advance, fleet operators could attempt to satisfy said demand. This could be considered a classical supply/demand scenario where the fleet operators want to reach market equilibrium by adjusting the price or quantity supplied.

By analysing big data from micro-mobility service providers on user/customer interaction, one may predict and model demand in an area with some error. Thus, a fleet operator could use the demand forecast results to adjust the quantity supplied locally within a market, for example, through rebalancing or in its entirety by reducing the number of available vehicles to the customers in the market. Additionally, a fleet operator may use a demand forecast for vehicle deployment or dynamic pricing.

2.3 Time Series Models

When forecasting demand, many different types of time series models can be used. This section presents and explains the relevant models used in this project.

2.3.1 ARIMA

The autoregressive integrated moving average (ARIMA) model is a type of statistical model used to analyse data and predict the future (forecasting) [16]. It is one of the more understandable models that is used in the project. ARIMA is an acronym that describes the key aspects of the model, namely

- AR: Autoregression, A model that uses the relationship between present data and historical data points.
- I: Integration, Differencing data one or more times in order to make the time series stationary.
- MA: Moving Average, a model that uses the relationship between present data and a residual error from a moving average from historical data.

2.3.2 FBProphet

FBProphet is based on the ARIMA model but tries to make it easier for novice analysts to be in the loop and understand what they are doing [17]. FBProphet uses a composite time series model with three main components: trend, seasonality and holidays. Together they form the following equation:

$$y(t) = g(t) + s(t) + h(t) + \epsilon(t) \tag{2.1}$$

where $g(t)$ is trend, $s(t)$ is seasonality and $h(t)$ is the holiday component.

Compared to the standard ARIMA models, FBProphet uses the holiday component to better measure different special events that impact the time series. FBProphet can thus model different holidays and custom events that frequently occur that ARIMA usually has problems with. It is also possible to add custom made holidays in FBProphet.

FBProphet uses a piecewise linear model for forecasting problems without saturating growth. The trend model is

$$g(t) = (k + a(t)^T\Sigma)t\delta + (m + a(t)^T\gamma) \tag{2.2}$$

where k is the rate of growth, δ has the rate adjustments, m is the offset parameter and γ is set to $-\delta_j s_j$ to make the function continuous. s_j is the number of changepoints at time j. The flexibility of the trend is determined by τ, which is implemented into the equation by taking the prior $\delta_j \sim$ Laplace(0,τ). The analyst can change τ to control the flexibility. The command to change τ is *changepoint_prior_scale*.

When modelling business time series, there is seasonality as a result of human behaviour. For example, every week affects the time series that repeats every week. School breaks and vacations can produce effects that occur yearly. To forecast these effects, FBProphet specifies seasonality models that are periodic functions of time.

FBProphet uses the Fourier series to provide a flexible model of periodic effects. Let P be the expected period of the time series. For example $P = 365$ for yearly data or $P = 7$ for weekly data (when working in days). Arbitrary smooth seasonal effects can be approximated with

$$s(t) = \sum_{n=1}^{N} \left(a_n cos\left(\frac{2\pi nt}{P}\right) + b_n sin\left(\frac{2\pi nt}{P}\right)\right) \tag{2.3}$$

this standard Fourier series. Increasing N allows for fitting seasonal patterns that change more quickly, albeit with an increased risk of overfitting. In this project, most of the data is sub-daily (mostly hourly or groups of multiple hours). Finally, the analyst can change this N variable with the command *seasonality_prior_scale*.

Holidays and events is a unique feature for FBProphet that provides the analysts with the tools to forecast large, predictable shocks to the business time series which do not follow the typical seasonal periodic patterns mentioned before. Holidays like Christmas and Easter could affect a system in a shocking but somewhat predictable way. Other regularly occurring events can also be taken into consideration by FBProphet. The analyst can add events and holidays that they think might impact the forecast.

After providing the model with a list of all the events and holidays, along with the dates they occur, another standard Fourier series is used similarly as with seasonality. Days around the holiday itself can also be included using additional parameters to treat those days as holidays themselves.

Additionally, FBProphet also uses what they call *Analyst in the loop Modeling*. Analysts tend to have a lot of knowledge about the data but not enough statistical knowledge to model it. In FBProphet, there are a few ways that an analyst can influence the model with their own knowledge, without knowing much about the statistics behind it. It is easy to change certain parameters in the model to reflect reality better. For example, the analyst may know that certain holidays or regressors impact the forecast more than others and may fine-tune the smoothing parameters (N) to tell the model how much historical data should impact the future. FBProphet also provides good visualisation of the forecasts, making it apparent if there are data points skipped unintentionally, or if there are certain intervals of the data that should be left out.

2.4 Uber's H3 Spatial Index

The H3 spatial index introduced by Uber is a discrete global grid system that divides the entire earth into hexagonal shapes. These hexagons are all uniquely identifiable and can be of different sizes depending on which *resolution level* that is used. More information about this spatial index can be found in Appendix B.

2.5 Stochastic Models

This section describes which relevant stochastic models have been used throughout the project.

2.5.1 Markov Processes

When modelling complex stochastic systems, a Markov process is helpful as it encapsulates the often dynamic state transitions associated with complex systems. With its inherent traits of separating system states and isolating their influence over time, the Markov process $\{X(t), t \in T\}$ excels at describing transitions between states in the stochastic process. Its properties can be described as follows: Given any state $X(t)$ at time t, such that $s > t$, the state $X(s)$ will not be affected by any state prior to the time t, as a future state $X(s)$ will only be influenced by the current one $X(t)$. Following this, a transfer or traversal matrix P can represent the probability of different states switching each time period (see Equation 2.4). Each element p_{ij} is non-negative and represent the probability that the state i transfers over to state j, with each row summing to 1.

$$P = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1j} \\ p_{21} & p_{22} & \cdots & p_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i1} & p_{i2} & \cdots & p_{ij} \end{pmatrix} (0 \leq p_{ij} \leq 1) \tag{2.4}$$

The only knowledge of value is the current state, as past behaviour will not alter the outcome at hand. If this Markov process is repeated over a finite discrete time interval (such as $t = [0, 1, 2, ..., k]$) the sequence is called a Markov chain. Furthermore, if the transition matrix P is aperiodic and irreducible, meaning the process can reach any states regardless of its current state, the matrix will converge to a stationary distribution π once taken to the power of some k as follows:

$$\lim_{k \to \infty} P^k = \pi \tag{2.5}$$

The resulting property for π is that multiplying with P again will not affect the outcome: $\pi P = \pi$. However, creating a probability vector by multiplying π with a row vector results in a stationary distribution vector. An element in this vector gives the expected probability that the system is in that state once a balance is achieved, with the sum of all these elements being 1. This feature is used in the proposed demand forecasting model, which will consider the traversal matrix as H3 hexes or groups of H3 hexes that vehicles travel between, viewing them as system states.

2.5.2 Poisson

A discrete random variable X is said to have a Poisson distribution with parameter k if its density f is given by

$$f(x) = \frac{e^{-k} k^x}{x!}, \quad x \in \mathbb{N}_+ \quad k > 0 \tag{2.6}$$

Poisson random variables most commonly arise in Poisson processes which involve observing discrete events in a continuous interval of time, length, or space. The random variable X has an expected value of k which is defined as λs where λ is the average number of occurrences of the event per unit. s is the length or size of the observation period. Furthermore, the random variable X follows a Poisson distribution [18].

2.6 Model validation metrics

When validating models, it is important to use relevant metrics. In forecasting, both the *Mean Absolute Error* and *Mean Absolute Percentage Error* are commonly used [19]. When studying correlations between data, the *Coefficient of determination*, commonly referred to as R^2, is a metric to evaluate if and how much of a correlation there is. These metrics are further explained in this section.

2.6.1 Mean Absolute Error: MAE

Mean Absolute Error is the mean of all absolute errors in the data. The absolute error is the difference between the measured value and the actual value. The MAE is the mean of these values [19]. MAE is calculated by

$$M = \frac{1}{n}\sum_{i=n}^{n} |x_i - x| \tag{2.7}$$

where n is the number of errors, and $|x_i - x|$ is the absolute error.

2.6.2 Mean Absolute Percentage Error: MAPE

Mean absolute percentage error is a metric used to measure prediction accuracy in statistics [19]. The accuracy is measured as a percentage and is calculated as follows:

$$M = \frac{1}{n}\sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \tag{2.8}$$

where A_t are actual values and F_t are the predicted values. MAPE is one of the most standard measures of error and works best on data with no extremes or zeros.

2.6.3 Coefficient of determination : R^2

The coefficient of determination, also known as R^2, is used to determine if differences in one variable can explain differences in another variable. R^2 provides the percentage variation in y explained by x-variables [18]. The coefficient ranges from 0 to 1. The coefficient of determination is the square of the correlation coefficient R.

R^2 is calculated by calculating the regression coefficient squared.

$$R^2 = \left(\frac{n\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{\sqrt{\left(n\sum x^2 - \left(\sum x\right)^2\right)\left(n\sum y^2 - \left(\sum y\right)^2\right)}} \right)^2 \tag{2.9}$$

R^2 can be used to find the likelihood of future events falling within predicted outcomes for our machine learning models. The coefficient shows how likely it is that future samples would fall on the predicted line.

2.7 K-Means Clustering

K-mean clustering is a basic and straightforward unsupervised machine learning algorithm aimed to aggregate similar data points to identify underlying patterns [20]. The k parameter is the fixed number of clusters created from the dataset, often referred to as *centroids*, which represent the centre of each cluster. The algorithm often starts with randomly selecting these centroids and then assigning each data point to a centroid set depending on the least squared Euclidean distance [21]. Through iteration, the centroids are adjusted to minimize the distance to all data points assigned to it, converging on a local minimum, resulting in clusters consisting of similar data points emerging.

3
Definitions

This chapter describes and explains some of the terminology that is used throughout the report. Along with this terminology, it also presents the data that has been received from Voi Technology, and how different parts of this data is referred to as.

3.1 Demand

Demand can be seen in different ways depending on which perspective that is appropriate, and here the different types of demand that are referred to in this thesis, are further explained.

- **Captured Demand** - The quantity of demand willing to pay the suppliers' price and the non-price determinants are satisfied.
- **Uncaptured Demand** - Includes all the demand which has not led to a ride start, either due to the price or non-price determinants.
- **Demand** - The sum of both captured demand and uncaptured demand.

3.2 Supply

Below are the three different ways that supply is referred to in this thesis.

- **Organic supply** - Supply which comes from riders ending their vehicles in specific hexbins.
- **Non-organic supply** - Supply changes caused by the involvement of a fleet operator or all other causes.
- **Supply** - The sum of both organic supply and non-organic supply.

3.3 Data

The following section explains what some of the data contains and represents.

3.3.1 Rides

Rides data describes the vehicle movements; start position(H3) and end position(H3) of each ride a user have done. See A.1 for further reading.

3.3.2 App Open

This is the quantity of customers opening the smartphone application in a certain hexbin, and a timestamp in an hourly interval. See A.2 for further reading.

3.3.3 Vehicle status

Vehicle status gives the current status of a vehicle and its position(H3). If it is ready to be used by a user or if it is unavailable. See A.3 for further reading.

3.3.4 Idle Time

Idle time is the average amount of time vehicles have to wait between rides in a hexbin. See A.4 for further reading.

3.3.5 Weather

The Weather data contains historical rain, wind, sun and temperature data over Gotheburg during 2020. See A.5 for further reading.

4
Methodology

This chapter presents the applied procedure of the study. The research aim is quantitative, and the result is based upon quantitative data provided by Voi Technology. The given data has set up the possibility to model the research aim mathematically. Mathematical modelling has been put in effect through several programming packages in the programming language Python. The mathematical modelling of the study is divided into three central parts; the Time series analysis, Markov chain analysis and Poisson processes. Additionally, the possibility of syndicating the main models is evaluated and compared with using the models individually.

4.1 Data analysis

The data was delivered in a comma-separated values (CSV) file format which was primarily processed with Python-based libraries and packages. All the data was from the year 2020. It has been an ongoing dialogue with the contact person from Voi Technology, which has allowed requests of new data continously. Analysing and evaluating the requested data has been a substantial part of the project, and this will be described in the sections below.

Evaluation and examination by looking at the number of data points, the time interval between each data point and what each data point signifies were steps made early in the process. To find significant patterns in the different features, plots were created. Anything that seemed useful was plotted. This built the foundations of the forecasting model. After this, the following step was to consult with our supervisor, about using the data in the best possible way. Later, comprehensive analyses were performed to decide the features and evaluate if Voi Technology provided enough data.

4.1.1 App open

Services that provide transportation based on MaaS (Mobility-As-A-Service) usually come with a companion mobile application. This application often allows users to get information about the service, such as finding vehicles, tracking payments, seeing ride/trip history, etc. Software implementations like location tracking in the application can make it possible to track user behaviour. By tracking and analysing user behaviour and predicting if the user is looking for a vehicle, the app open feature (see Section 3.3.2) could be used to find both captured and uncaptured demand. All interactions in the companion mobile application will be referred to as sessions from

here on.

The number of sessions labelled as *looking for vehicle* which also correlate to started rides is equal to the captured demand. Sessions labelled as *looking for vehicle* which do not correlate to a started ride is the uncaptured demand. Correlating these data points is done by matching ride start and app open hexbins and their respective timestamp. These features could then be used in forecasting models, such as FBProphet (see Section 2.3.2), to make predictions for both captured and uncaptured demand.

4.2 Geospatial units & indexing

Working with geospatial data requires an interpretation of space. In the research, we have focused on two methods of defining and indexing spatial units. One method uses Uber's H3 geospatial index, whereas the other use a combination of H3 units, creating zone categorisation based on the ride start lag feature.

4.2.1 H3

All non-market wide analysis use Uber's H3 geospatial index due to its properties described in Section 2.4. The geospatial units have a fixed size throughout the research at an H3 resolution level of nine. Furthermore, a fundamental assumption is that users, regardless of their distance from the vehicle within the geospatial unit, are equally likely to convert to a ride. Thus any geographical or infrastructural constraints within a geospatial unit are disregarded.

$$H = \{\alpha : \alpha \text{ is a Uber H3 geospatial unit}\} \tag{4.1}$$

$$\alpha = \{l : l \text{ is a 2D position contained by geospatial unit } \alpha\} \tag{4.2}$$

$$\forall\, l,\ l \in \alpha,\ P(l) \sim \text{Uniform} \tag{4.3}$$

While the resolution level throughout the research is held constant, Uber's H3 API allows for scaling the geospatial units. Non clustered geospatial units, native H3 units, will hereafter be referred to as *hexbins*.

4.2.2 K-means clusters

In attempts of trying to classify similar performing hexbins, k-means is used to create n clusters. The clustering method k-means was selected due to its properties described in Section 2.7. The clusters are created from the hexbins cumulative ride starts over a time interval. Thus, clustering similarly performing hexbins with a spatial granularity of the total size of the combined hexbins and a temporal granularity defined by the size of the time interval. Furthermore, clustering can be beneficial due to a lack of data in a hexbin. By clustering similarly performing hexbins and treating the clusters as one geospatial unit, the underlying data in the individual hexbins is combined.

4.3 FBProphet

The following section describes how FBProphet was used during the development of the two different FBProphet models. Both models were developed using a similar manner. However, they handled geospatial information differently since the first model only considered a single hexbin, while the other could consider multiple hexbins simultaneously using clustering of hexbins. FBProphet provided an easy way to create time series models that were intuitive to understand and relatively easy to test.

4.3.1 Single H3-9 Demand Forecasting

The creation of the single hexbin forecasting model with FBProphet started with a straightforward model which considered the demand in a single hexbin as the amount of ride starts in it, which narrows the model to only being able to model captured demand, by predicting an amount of ride starts in a future point in time. The model's performance was hoped to improve by adding additional features to the model, such as weather information (temperature, wind speed, precipitation and sun time). Finding the best combination of these features was done using a brute force method, where every single combination was tested against each other during the same forecasting period. The MAPE was measured for each combination during testing. The combination whose average MAPE across all forecasting periods was the lowest was decided to be the best combination of features.

Additionally, another feature was introduced, which acts as an assist feature to tell the model that it should consider a point in time as a *weekend*. This essentially becomes a seasonality, but as a feature introduced into the model as a regressor instead, see Section 2.3.2 for more detailed information on how this works.

4.3.2 K-Means Zoned H3-9 Demand Forecasting

The k-means zoned model development started with clustering all hexbins into different zones using k-means. This model, too, only considered captured demand by recognising the ride starts as demand, which leads to the model predicting ride starts, given a future point in time. To find an optimal amount of zones to cluster into, a similar brute force method used in developing the single hexbin model (see Section 4.3.1) was used. By performing forecasts in many different time periods and measuring the forecasts' MAPE in all zones, different number of zones could be compared against each other to see which amount of zones provided a lower MAPE throughout all zones, while also granting a high enough geospatial granularity.

4.4 Demand as a Stochastic Process

Given that demand is dependant on supply and supply drives demand with the addition that demand and supply in one geospatial unit is dependant on the state of neighbour geospatial units. It seems trivial that a deterministic model would consist of multiple differential equations with heuristics far too complex to model.

Instead, can we design a stochastic process model that accurately captures the demand characteristics at fine temporal and spatial granularities?

A Poisson process can be used to describe demand by observing the number of occurrences of ride start events λ during an observation period with size or length s. The random variable X, the number of occurrences in the interval of size s, follows a Poisson distribution with parameter $k = \lambda s$. The interpretation, where demand is approximately Poisson, can be helpful to approximate demand.

4.4.1 Demand as Approximately Poisson

There are seemingly enough factors behind why someone would start a ride at a specific time and place for it to considered random.

$$\textit{A geospatial unit } \alpha \textit{ at time } t \textit{ with supply } S_t^\alpha,\ S_t^\alpha \in \mathbb{N} \tag{4.4}$$

$$\textit{Set of ride start events (departures)} \quad O_t^\alpha = min(Poi(\lambda), S_t^\alpha) \tag{4.5}$$

$$\textit{Set of ride end events (arrivals)} \quad I_t^\alpha \sim Poi(\lambda) \tag{4.6}$$

A ride start event belongs to a geospatial unit at a specific time, causing the supply in the geospatial unit to decrement. A ride end event causes the supply to increment. Furthermore, we will assume that the process has independent increments and that ride start events in O are independent of each other. The same is assumed for ride end events in I. A ride start event cannot exist when the supply is equal to zero and there cannot exist more ride starts than what there exists supply. Thus, the minimum of the two is the set of ride starts.

$$\textit{Ride (either start or end) events } o_1 \textit{ and } o_2 \quad P(b) = P(o_2|o_1) \tag{4.7}$$

With the above definitions, we can define the supply for a geospatial unit in time, given: the previous state of supply, the sets O, I, and N. The set N is the generalised set of change in supply due to fleet operator intervention and all other causes.

$$S_t^\alpha = S_{t-1}^\alpha + I_t^\alpha - O_t^\alpha - N_t^\alpha \tag{4.8}$$

Noteworthy is that the requirement of independence demonstrated in the Definition 4.7 does not always hold. An example of a violation of the independence requirement is when a ride start event causes the supply in a geospatial unit to fall to zero. It is not the event itself that violates the independence requirement; instead, it is the consequence of the event, causing the supply to become zero. This is described in the Definition 4.5

$$P(o|S_t^\alpha = 0, o \in O) = 0 \tag{4.9}$$

There cannot exist a new ride start event with zero available vehicles in the concerned geospatial unit. However, there are other assumptions required to ensure

independence and identical distribution. We will assume that the probability of a ride start event is equally distributed for any geospatial unit.

4.4.2 Measuring Demand

Assume a geospatial unit α with a constant oversupply with demand D at time t.

$$\forall t \quad D_t^\alpha < S_t^\alpha \tag{4.10}$$

Furthermore, assume we observe α with constant oversupply and measure the number of rides starts events for a duration of s minutes, starting at time t. The set of ride starts between the closed time interval $[t, t+s]$ fully describes the demand in α.

$$D_{[t,t+s]}^\alpha = \sum_t^s D_t^\alpha \tag{4.11}$$

With the assumption in Definition 4.10 there always exists enough vehicles to match all the demand. Thus, all rides that could have started between $[t, t+s]$ started and thus belongs to the set D. There are a few things to consider to oversupply a geospatial unit in practice. At what spatial granularity does ride conversion become approximately uniform? If we once more assume a uniform probability distribution for ride starts in a hexbin at resolution level nine (Definition 4.3), then, as long as there exists at least one available vehicle in α Equation 4.10 and subsequently 4.11 can be considered true.

$$S_{[t,t+s]}^\alpha \neq 0 \rightarrow \forall t,\ t = [t, t+s] \quad D_t^\alpha < S_t^\alpha \tag{4.12}$$

To measure demand when there is insufficient supply, we apply the scaling property of a Poisson process. A Poisson process, with rate $\lambda = 12$ and an observation duration $s = 1$ hour, has the same probability distribution as a Poisson process with rate $\lambda = 8$ and an observation duration $s = 2/3$ hour. We define a metric called *coverage* which describes the fraction of an hour a hexbin is oversupplied. For the above example, the coverage would be $2/3 \approx 0.67$.

$$Coverage_{[t,t+s]}^\alpha = \frac{samples\ between\ [t, t+s]\ with\ S_t^\alpha > 0}{s} \tag{4.13}$$

$$D_{[t,t+s]}^\alpha = \frac{A_{[t,t+s]}^\alpha}{Coverage_{[t,t+s]}^\alpha} \tag{4.14}$$

In conclusion, as long as there exists at least one available vehicle in the hexbin with a uniform spatial ride start probability distribution (Definition 4.3); the set of ride starts has a temporal Poisson Distribution, given the supply does not equal zero (Definition 4.5), with a rate λ fully representing the expected demand (Definition 4.12 and 4.11).

4.4.3 Predicting Demand

Assuming that arrival rates are dependant on time, enforced by rush hours or late at night when everybody is asleep, we can extend the Poisson interpretation of demand by interpreting demand as a Non-Homogeneous Poisson process, meaning the rate λ changes over time $\lambda = \lambda(t)$. We can thus split the hours of the day into appropriate intervals, such as morning, noon, afternoon, evening, and night, where the rates λ are notably different.

Identifying appropriate temporal intervals involve classifying similar performing time periods. In this research clustering method, k-means is used to find an appropriate λ for different t. The average cumulative demand for each hour of day and day of week is clustered, this may result in, for example, Fridays hours $\{$*17:00*, *18:00*, *20:00*, *23:00*$\}$ if the average demand in these periods are relatively equal. The Poisson distribution assumed for the future time period is thus the Poisson distribution described by the Poisson process for all clustered hours of day for each weekday over a defined time period. The time period can be four weeks, including four weekdays each, implying the four latest samples for each hour of day and week. Furthermore, the property of splitting and merging Poisson processes can be used to change the spatial granularity.

4.5 Application of Markov Models in Vehicle Demand Estimation

Combining Markov chain models and the other prediction-based models are heavily inspired by Zhou et al. [22], who looked at demand prediction on a bike-sharing system in Zhongshan City, China. By modelling the flow of organic vehicle rental traffic in the city, one could improve predictions during the day as these rentals had clear patterns depending on the time and day of the week [23]. However, as the vehicles are free floating rather than bound to stations, some adjustments are made to better model the MaaS demand problem at hand compared to most previous works which focused on bike rental instead.

The outlined steps described in the method by Zhou et al. still holds to a high extent, being the following:

- **Step 1** - Demand prediction on the whole market. Forecasting models based on time series, such as FBProphet, or probability models from a Poisson distribution.
- **Step 2** - Divide H-3 hexes into geographical zones by k-mean clustering based on their number of rides produced to reduce the size of future transfer matrices.
- **Step 3** - Create a transfer matrix between each zone, resulting in a $n \times n$ matrix which categorize all rides based on start and destination zone. An element a_{ij} represents the number of rides starting in zone i and ending in zone j.
- **Step 4** - Construct the transition probability matrix P for scooter rentals. An element P_{ij} indicates the probability of a start in zone i ending in zone j, described in Equation 2.4 and later referred to as the Flow Matrix (see

Equation 4.15).

$$P_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}} \tag{4.15}$$

- **Step 5** - Calculate a steady-state probability vector, explored in Equation 2.5.
- **Step 6** - Combine Flow Matrix (step 3), prediction (step 1) and supply/steady-state to simulate the system and improve the demand prediction.

Step 2 stemmed from assessing if the data provided can create these flow relations by limiting the number of unique hexes used. Since the data only held information on locations where rides had occurred, some discrepancies were found early. One such issue is that a ride is not limited to a predefined number of hexes/stations, something which introduced problems when in certain weeks over 650 unique hexes were visited while others had much fewer (closer to 300). To generate the traversal matrices discussed in Section 2.5.1 the received data was filtered to ensure all involved hexes had incoming and outgoing rides registered (as a hex with only incoming data can be seen as a dead-end for the vehicle supply). Furthermore, the large number of unique hexes proved difficult to handle. Most of them had very sparse ride data, meaning the resulting traversal matrix P (see Equation 2.4) contained mostly zeros and small fractions. Two months of ride data (Aug-Sep in Gothenburg) demonstrated that the 100 most popular ride starting hexes accounted for 64% of all rides. In contrast, the remaining 559 hexes were responsible for 36% of the market traffic (with over 230 of them having less than one ride per day on average).

The solution was to aggregate all hexes by k-mean clustering (see Section 2.7) to give each element in the traversal matrix meaningful values. These groups, later referred to as *Zones*, were also used in the time series analysis. Each zone tended to include enough ride data to make adequate predictions (more on this and the motivation behind the chosen $k = 6$ in Section 5.2.3). Each zone was given a letter as a name, with Zone A containing the most traffic-heavy hexes and each following letter representing hexes in ascending order of total ride starts during the time period. With the shape of the traversal matrix decided, the process of deriving a stationary distribution could begin.

As described in Equation 2.5, the steady-state vector could be computed by taking the matrix to the power of some k. The resulting vector was seen as a naive supply distribution once the system has followed the organic flow and used as a base. We wanted to try to simulate traffic flow in the city to predict shortage/oversupply.

To simulate the city traffic flow, we opted to divide the traversal matrix into separate matrices of flow depending on the hour and day of the week. This aimed to encapsulate seasonal commute during the week and also any increased weekend activity. The data received was bundled by the hour, which hindered any further granularity in allowing simulation of smaller increments. Nonetheless, each simulated step's expected number of ride starts in each zone was adjusted according to current and estimated incoming supply. If a zone lacks supply, the number of rides started is scaled down accordingly, with the logged discrepancies. The algorithm used to simulate each passing hour is in broad terms the following:

Algorithm 1 City Flow Simulation

Require: Prediction of starting rides for each zone
Input: 3D Flow Matrix for zones depending on time/day $FM[time/day][zone][zone]$ and supply for each zone as a vector S
Output: Supply vector over time SV, Missed rides over time MV

Initialization: Create matrices SV and MV to store each step in the simulation
while Simulate pre-decided number of days/hours **do**
    incomingSupply = ridesStarting*FM
    currentSupply = SV[x] - (ridesStarting - incomingSupply)
    if currentSupply contains negative supply **then**
        Calculate difference and reduce ridesStarting for that zone
        Repeat supply calculation
    else
        Save current supply in SV and any missed rides in MV
    end if
end while
Return SV and MV

To further extend the simulation, we tried to include a fundamental implementation of rebalancing (introduced in Section 2.2) by moving supply from different zones to others at the end of each day simulated. Onwards, real supply data was used in place of the stationary distribution and predictive results from the FBProphet model based on the same zones used for the flow matrices.

4.6 Ethical considerations

Most ethical dilemmas stemmed from the data received. On one hand, the data was essential for our research, with the continuous request for new data sets enabling the project to grow in directions we saw fit. However, the confidentiality of the data was always present as both the integrity of customers and the value the data had for the company is of high importance. By managing the received assets with caution, these conflicts were avoided.

4.7 Method discussion

Using data from unprecedented times like the one right now could lead to an inaccurate prediction for the future. The COVID-19 pandemic has forced people to work from home and changed their everyday movement pattern. There are several ways how the pandemic could have changed the demand for micro-mobility services. Firstly, the people who normally took a vehicle to work and is now working from home is no longer in need of a micro-mobility service since one does not have to go to work. Secondly, the government has encouraged all citizens to stay away from public transport, leading to more people using micro-mobility services. Thirdly, several pubs and nightclubs have closed, and there are fewer people in movement during the night, which could have reduced the demand for renting a scooter. There

are many ways one could analyze how COVID-19 has changed all citizens' lifestyles, and using the year 2020 as a foundation year for future demand prediction could lead to serious inaccuracy. To get a grasp of how the pandemic has affected the demand, a comparison between 2020 and 2019 would be needed.

Analyzing customer patterns for a start-up could also be misleading. Voi Technology was one of the first Swedish micro-mobility services companies when it launched in August of 2018 [3]. Since then, it has grown rapidly and expanded throughout Sweden. Training a machine learning model on data from 2019 might not represent the future market knowing current expansion of the company. This is also something that should be in mind when selecting a model for time series prediction. Further research is needed to find a suitable model for a recently launched product being in a growing stage.

5
Results and discussion

This chapter presents the correlation between different parts of the data and how the developed models perform. This chapter also discusses the different findings and why they are significant.

5.1 Data Analysis

The initial data analysis performed was very important in finding patterns present in the received data. Creating a variation plots eased the visualisation of trends and seasonality in the data while also highlighting aspects of the data that was easier to model. By having different window sizes in the model, the patterns could vary by a significant amount. The term *activity* is defined as the sum of ride starts and ends.

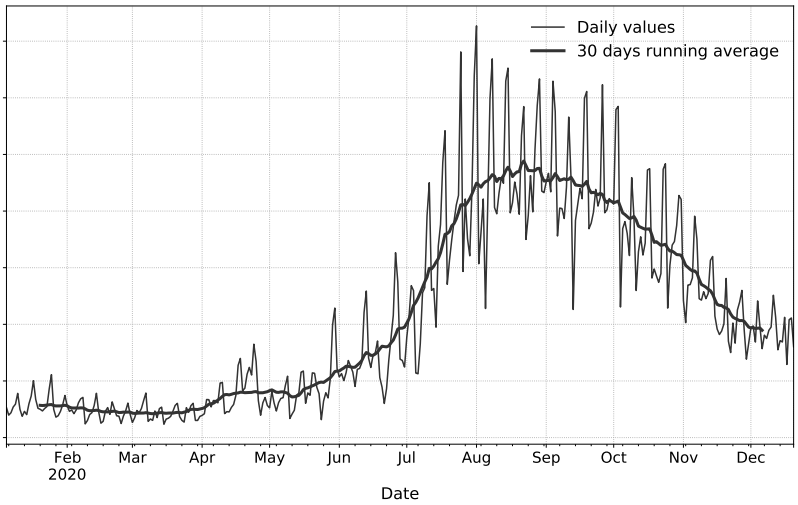
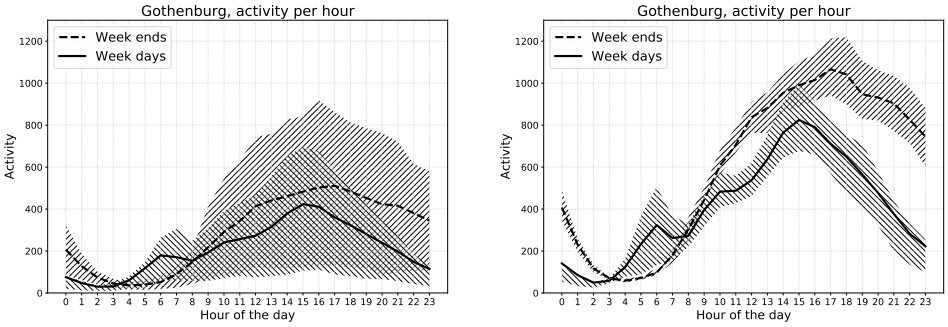


Figure 5.1: Daily averages and a 30 day rolling mean average of the activity in Gothenburg during most of the year. The average activity varies a lot during the year, making it significant to divide the year into different parts to get clearer views on the trends and seasonalities of the data.

In Figure 5.2, most of the year is considered when comparing the activity between weekdays and weekends. The plot shows the average activity during each of the hours of the day, where the dashed area includes 50% of the observed data. There is a clear pattern and a distinguishable difference between weekdays and weekends, but there is much overlap between the dashed areas.



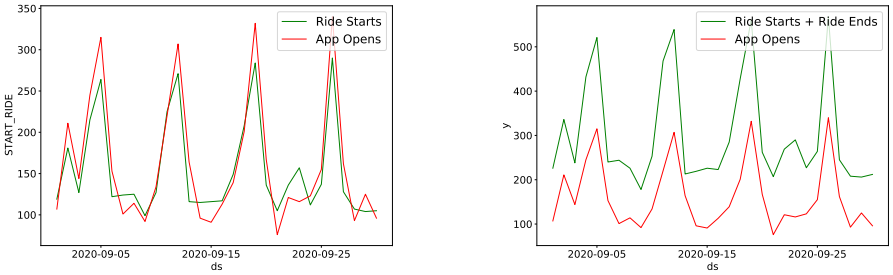
(a) Whole year **(b) Summer**

Figure 5.2: Comparison between activity on weekdays and weekends during the whole year, compared to the activity during weekdays and weekends during the summer. This figure shows how including data from different parts of the year can yield wildly different results.

The results are quite different by filtering away the high variance parts of the year 2020, the winter, spring and late autumn. 2020 was a special year, and the different effects of the pandemic are noticeable in the plots. Figure 5.2(b) shows the results of only plotting the high activity months of the summer. The overlap is gone, and the difference between weekend and weekdays is much clearer. On weekdays the activity follows the commuter traffic. People tend to use vehicles in the early mornings and after school or work, but there is a steepening curve after 16 in the afternoon. There is more activity on weekends, but the curve is also somewhat offset by a few hours. The activity both starts and steepens much later in the evening. The results of these plots supports that a sliding window would be beneficial.

5.1.1 App open

We quickly noticed discrepancies in the app open data by plotting the number of app opens and ride starts over a month. In Figure 5.3(a), the number of start rides is sometimes greater than the number of app opens.



(a) **(b)**

Figure 5.3: Rides and app opens in a popular hexbin during the month of September in 2020. (a) shows app opens (red line) and ride starts (green line). (b) compares ride start + ride ends and app opens. The red line should never be below the green line, since the application is used to start/end each ride.

The reason for this discrepancy seen in Figure 5.3 might lie in an issue with the implementation of the app open trigger, or data is being lost for another reason. A naive trigger for app opens would ideally follow the same trends as the sum of ride starts and ride ends. This is not the case with the received data, see Figure 5.3(b). There are two possible reasons for this: either the trigger tries to find customer sessions looking for vehicles and fails, or the trigger is for some other reason missing sessions. A theory could be that the application records a session when you first open it, but it does not end the session when you put your phone into sleep with the app still open. This might be what is causing the behaviour of ride ends to be seemingly excluded from the data. The app open approach with FBProphet was quickly abandoned because of the above-explained data discrepancies. However, App opens, if handled correctly with an advanced session trigger and data collecting, could be of great value for demand predictions and should be explored more. For instance, Woo Ham et al. [24] uses their equivalent app open data to find uncaptured demand in Konkuk University Station, South Korea, for their respective e-scooter market.

5.1.2 Weather effect on vehicle demand

There seems to exist a relation between the maximum number of observed rides in the observed data for any hexbin during an interval of one hour, over a time period of approximately a year. The fitted trend model suggests a significant relation with a p-value of $< 0,0001$. However, it does not explain much of the observed variation in the data given an R^2 value of $0,485824$. The relations appear to be logarithmic, where the max number of achievable rides decreases logarithmically with increased precipitation levels.

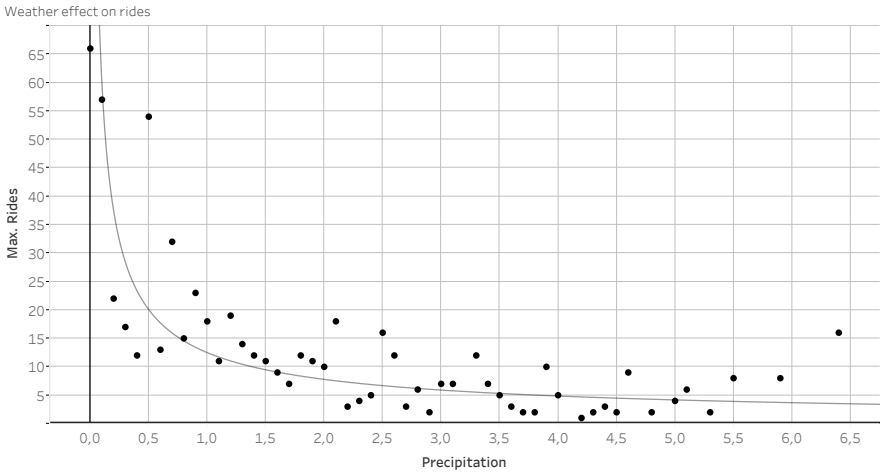


Figure 5.4: The plot of maximum of rides for precipitation. The data is filtered on Date and Time. The Date and Time filter ranges from 2020-01-01 01:00:00 to 2020-11-01 06:00:00. A Significant relation implying fewer rides with increased precipitation can be seen following the logarithmic curve.

It should, however, be noted that the distribution of samples for different levels

of precipitation is heavily skewed in favour of samples where the measured level of precipitation is lower. This is described in Figure 5.5. This directly implies that the observed data mostly consists of hours without precipitation. Another interesting observation is the effects of potential outliers since only one sample is selected for each level of precipitation during the time interval; irregular events unconcerned of weather are more likely to affect the results. An observed example could be the last sample in Figure 5.5, where 16 rides had an abnormal level of precipitation.

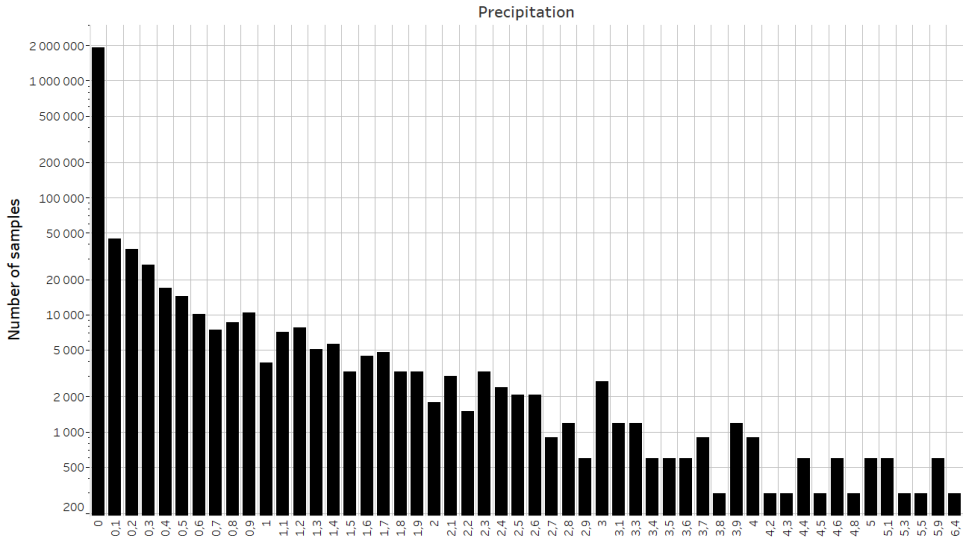


Figure 5.5: Count of Samples for each level of observed Precipitation. The data is filtered on Date and Time. The Date and Time filter ranges from 2020-01-01 01:00:00 to 2020-11-01 06:00:00, with a temporal granularity of one hour. There exists, a significant change, 40x more samples between the first level of precipitation (0) and the next (0.1).

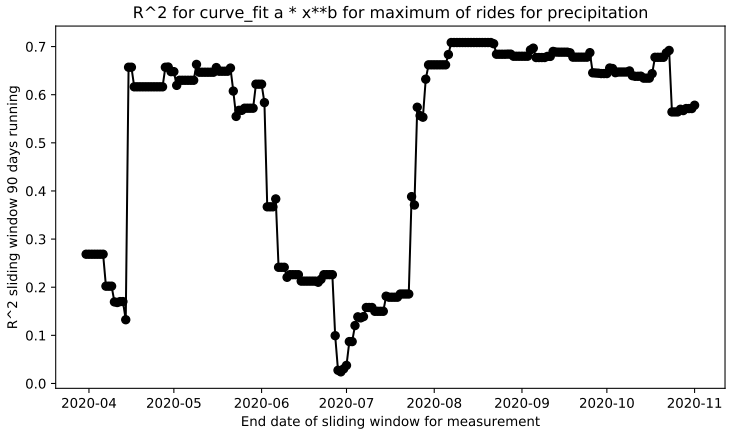


Figure 5.6: R^2 for curve fit function $a \cdot x^b$ for maximum of rides for precipitation within a time interval of 90 days running where the full time interval spans from 2020-01-01 01:00:00 to 2020-11-01 06:00:00. There exists an interval where the relations seems non existent and results as to why remain inconclusive.

The same method with 90-day intervals revealed that the relationship seems to exist except during a short period in January and larger portions of March and April. The reason for these results was not concluded.

5.2 FBProphet Models

This section aims to reveal the resulting models that were made using FBProphet. These models are all similar but have some differences that make their performance increasingly different.

5.2.1 Simple Single H3-9 Model

This model was built using only historical data on ride starts or captured demand. By modelling captured demand, the model could naively predict the number of ride starts in a single hexbin at a given point in time.

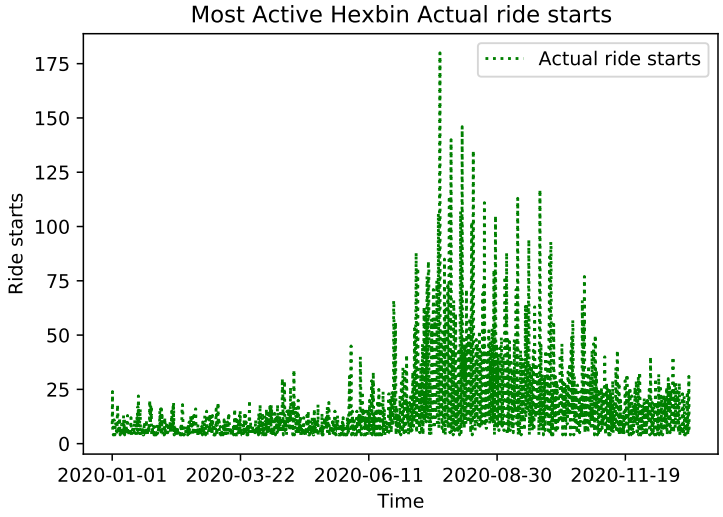


Figure 5.7: All ride start data present for the most active hexbin in Gothenburg during 2020. Here it is very obvious that the ride start pattern changes a lot throughout the year, with a more active period during the months of July-September.

By looking at the data, the single hexbin which had the most ride starts throughout multiple periods during the year 2020, whose total ride start data is shown in Figure 5.7. This hexbin was chosen during development since it had the most ride starts throughout multiple time periods. It quickly became apparent that hexbins that had fewer ride starts, or rather less frequent ride starts, were a lot harder to perform any modelling since their behaviour had very few rigorous patterns. Therefore it was decided to keep using this high traffic hexagon as a baseline for the top performance achievable with this model during development.

Using only historical data on ride starts to model demand resulted in the prediction only representing captured demand, which leaves out any possibility for finding

any uncaptured demand. However, the primary goal with this model was to predict ride starts, given a future point in time, which it was able to do.

By studying all rides in the hexbin displayed in Figure 5.7, it was apparent that the ride start behaviour shifts a lot during the year. There was a clear difference between the months January to June versus July to September, where the former has a lot less activity and a more regular pattern than the latter.

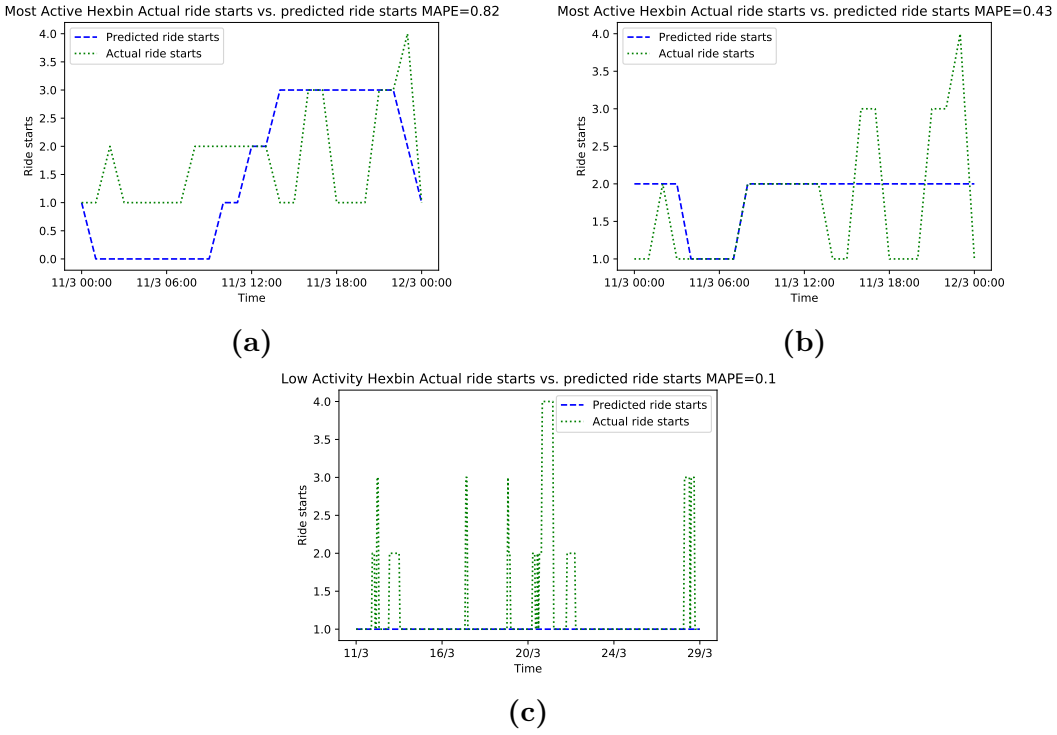


Figure 5.8: In all sub figures, the green line displays the true value at that point in time, while the blue line is the predicted value at that point in time. In (a), the model was trained on all available data which is shown in Figure 5.7(b), while (b) and (c) are only trained on data prior to their prediction intervals. This means that (a) is trained on the data it is trying to predict, while (b) and (c) are not. Comparing (a) and (b) shows that the model trained on data only prior to its prediction interval has better performance. The same model used on a less active hexbin gives a constant zero prediction, see (c). The less active hexbin is located very close to the most active hexbin, and it was still one of the most active hexbins throughout the city. Despite this, the model was unable to make realistic predictions.

A model was trained on all available ride start data, shown in Figure 5.7, which when predicting a 24-hour time frame in a low activity period, gave a MAPE of 82% (see Equation 2.8), visible in Figure 5.8(a). Important to note is that the predictions that were made actually follow a more smooth curve than what is shown. The prediction curve shown has been rounded to the closest integer number while negative predictions were replaced with zero. However, this model was extremely flawed, as it was trained on the data it was attempting to predict.

In Figure 5.8(b), the model was only allowed to be trained on data prior to 2020-03-11, which resulted in a prediction whose MAPE was 43%. The removal of future

values made the model both more realistic and increased its performance by around 50%. If applied to another hexbin with less activity than the most active hexbin, the resulting prediction would, for the most part, only consist of zeros, which can be seen in Figure 5.8(c). It was clearly visible that if a hexbin lacked activity, the prediction would most likely converge to always predicting no ride starts at all.

To further improve the model, it was decided that the data would be transformed from being hourly into being 4-hour intervals instead. This meant that in a 24-hour time span, there would be 6-points in time that would need to be predicted, and the spiky behaviour in the training data would be reduced since aggregating multiple hours reduces the frequency with which the data can change.

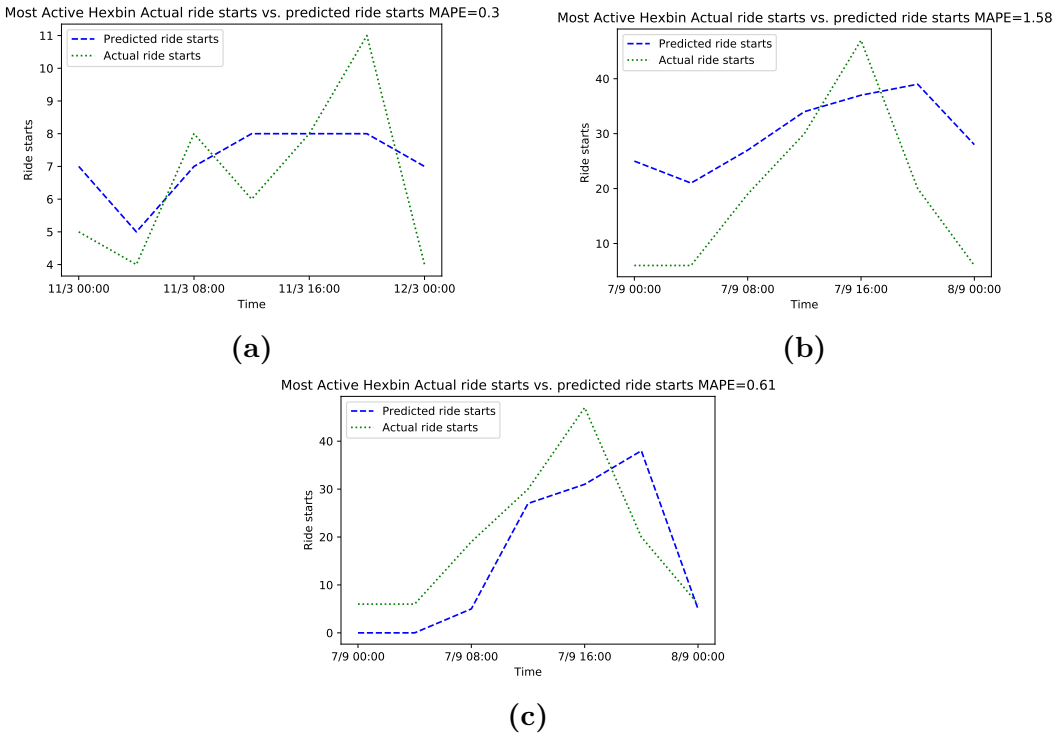


Figure 5.9: In all sub figures, the green line displays the true value at that point in time, while the blue line is the predicted value at that point in time. In (a), the first attempt at grouping together hours into 4 hour intervals is shown. It is evident that the grouping of hours has improved the predictions performance. However, when moving the prediction interval further into the future in (b), the performance escalated to very high numbers. This was because the model in (a) and (b) were both trained on all available data in the year 2020. By instead only allowing the model to be trained on a one month sliding window, in (c), the performance increased again.

It is evident that aggregating the hourly data into 4-hour intervals has increased its performance by around 30%, from Figure 5.9(a). However, it was found that further reducing the frequency in the dataset or grouping the data into larger aggregated groups of hours resulted in diminishing returns. To retain the most granularity, it was decided to stick with 4-hour groups.

Exercising the model in another time period which contained a lot more ride starts, like September in Figure 5.7, the result was much worse than first anticipated.

However, it is important to realise that this model was trained on all data before the 7th of September, meaning that the model has also been trained on low activity time periods. By instead training the model on data leading up to the prediction date, from 2020-08-07 to 2020-09-06 inclusive, the resulting MAPE was reduced by around 60%. During development, it was found that a sliding window, like the one in Figure 5.9(c), of 14-30 days gave better predictions in July-October, while a larger sliding window of 40-60 days improved predictions in November-June.

By only using a single hexbin and not relying on any other features than the historical ride starts, it remained difficult to get predictions with even lower MAPE than the one shown in Figure 5.9. Often it either overestimated or underestimated.

5.2.2 Single H3-9 Model with Additional Features

To further improve the single hexbin model, a feature which from now on will be called the *weekend feature* was introduced. This feature was simply a binary feature (1 or 0) which would serve as an assist feature to help better the model understand when it should consider a point in time as a *weekend*. The reason for this feature's introduction lies in an issue that the simple single hexbin model had.

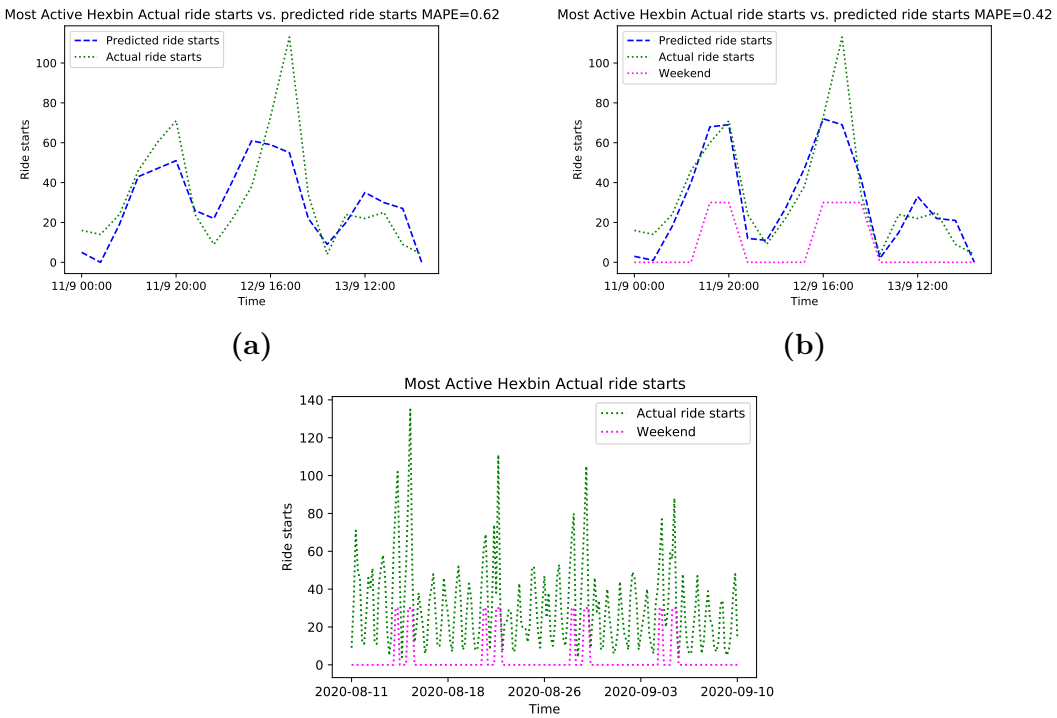
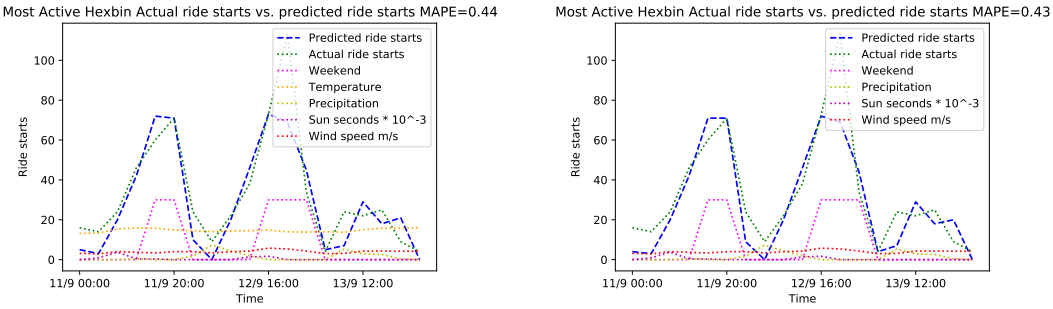


Figure 5.10: Using the same model described in Section 5.2.1, an attempt to predict the amount of ride starts in the most active hexbin was made, with a resulting MAPE of 62%, shown in (a), where the true amount of ride starts is in green and predicted in blue. By introducing the weekend feature into the model (magenta), its MAPE was reduced to 42%, shown in (b). In both (a) and (b), the models were trained using a one-month sliding window, and the training data is shown in (c).

The issue with the single hexbin model can be seen in Figure 5.10(a), where it is evident that the model was having trouble making accurate predictions during both 2020-09-11 (Friday) and 2020-09-12 (Saturday). The hope with the weekend feature was that its introduction would allow the model to interpret the feature as an undeniable pattern and assist in recognizing the weekend pattern.

With the introduction of the weekend feature, the model's performance increased by almost 35%, but there were flaws. In Figure 5.10(a), the model misses the spike on 2020-09-12 at 20:00. At first glance, this might seem strange, however, as shown in Figure 5.10(c), there is an apparent negative trend among the previous Saturdays that the model has been trained on. Just by looking at Figure 5.10(c), it seems reasonable that the next following Saturday might land on a spike at around 80 rides, which the model agrees on, in Figure 5.10(b). However, for some reason, the number of ride starts did not follow the trend and instead went up. These behaviours are challenging for the model to predict, giving a misleading result. However, the predicted process is somewhat stochastic, and therefore contains stochastic behaviours, like the one shown in Figure 5.10(c). If the model instead were trained on data from the entirety of 2019 and validated against data from 2020, a different result could have emerged.

In addition to the weekend feature, it was also tested to see if a similar *Saturday feature* could provide an additional increase in performance, however this made little to no difference in the predictions, which is why it was discarded.



(a) **(b)**

Figure 5.11: A comparison between the predictions performed by a model which uses the combination SPTW in weather features (a), and one which has the best performing SPW according to the tests in Table 5.1. Here the true amount of ride starts is in green, while the predicted amount is in blue. There is almost no difference between these two predictions, as their MAPE's are extremely similar, and in this specific time period it would make very little difference in which was used.

Together with the weekend feature, multiple weather features were introduced. These include temperature, precipitation, sun seconds, and wind speed. Simply adding these features as regressors (see Section 2.3.2 for information on how these work) into the model, using default parameters yielded the result shown in Figure 5.11(a). A quick comparison of MAPE's between the one in Figure 5.11(a) and Figure 5.10(c), shows that the resulting prediction actually performed worse with the new weather features, by a small margin. However, the intention was not to

use all of these weather features without checking which combination of those four would perform the best, instead all combinations were tested against each other in a brute force fashion.

Table 5.1: All possible weather feature combinations, together with their corresponding abbreviations, in order of ascending MAPE.

Abbrev.	Included weather features	MAPE
SPW	Sun time + Precipitation + Wind speed	0.5004
P	Precipitation	0.5019
PW	Precipitation + Wind speed	0.5022
SP	Sun time + Precipitation	0.5036
SPTW	Sun time + Precipitation + Temperature + Wind speed	0.5044
PTW	Precipitation + Temperature + Wind speed	0.5046
PT	Precipitation + Temperature	0.5065
SPT	Sun time + Precipitation + Temperature	0.5075
S	Sun time	0.5084
SW	Sun time + Wind speed	0.5110
W	Wind speed	0.5114
STW	Sun time + Temperature + Wind speed	0.5116
ST	Sun time + Temperature	0.5139
TW	Temperature + Wind speed	0.5164
T	Temperature	0.5170

To check which combination would perform the best, all possible combinations of weather features were made into their own models. These models were then trained on a one-month sliding window to predict 24 hours, for the first 7 days in each month between February and October, inclusive. Measuring the MAPE on all these predictions, the one with the lowest average across all predicted 7 day periods in each month was found to be SPW, with P by itself second best and PW being third, which can be seen in Table 5.1. The best performing combination SPW is shown in Figure 5.11(b), where its performance is only marginally better than the combination SPTW in Figure 5.11(a).

While this model was in development, it was also found that there was some correlation between the number of ride starts and precipitation (see Section 5.1.2). Since all combinations that include the precipitation feature outperform all combinations which do not include the precipitation feature, it is evident that its inclusion in the model increases its performance. This result only strengthens the argument that precipitation correlates with the number of ride starts.

An attempt to further improve the model was made by removing all data points in the time span 00:00-03:59, as it was thought that doing this would remove the possibility for the error at this point in time of having a large impact on the MAPE. When doing this and performing similar tests to the one described above, it was found that this removal increased the average performance by $11,9\%$ in time periods with much activity (July-September) while the average performance decreased by almost $27,8\%$ in time periods with lower activity (January-June). Therefore it

was decided that this removal of data points in the time span 00:00-03:59 would not be used in the model. However, for future research, it might be interesting to know that if the model is supposed to be used primarily on time periods with similar properties to the July-September time period, removal of nighttime hours may positively impact the model's prediction performance.

5.2.3 K-Means Zoned H3-9 Model

During the development of the single H3-9 model, it was quickly realised that the largest drawback was that it would not be suitable on any other hexbin than the ones that had the highest amount of traffic throughout the year. The model's inability to perform predictions on these types of hexbins can be seen in Figure 5.8(c).

For this exact reason, the development of the single hexbin FBProphet model carried on. Since the issue with the single hexbin model was that alone hexbins did not provide enough ride starts for there to be patterns that FBProphet could recognize, simply grouping together multiple hexbins would provide more obvious patterns that FBProphet could recognize. Using the total amount of ride starts that each hexbin had recorded throughout Aug-Sep 2020, the hexbins were clustered into k-means clusters, using differing amounts of clusters to evaluate which amount would perform best. From here on, these different clusters will be referred to as different zones.

Since it was previously found that the weather feature combination SPW performed the best in the single H3-9 model, this combination was also used to evaluate which amount of zones would be best. Performing a similar test to the one used when evaluating the best weather feature combination, which meant measuring the MAPE of the model's prediction when allowed to predict 24 hours ahead for 7 days using a one-month sliding window, in each month from February to and including October. This test was performed on all different zones created by the k-means clustering method.

From the performance tests results, which can be seen in Figure 5.12, any k between 2 and 7 results in at least $\frac{5}{7} \approx 71\%$ of all zones having a MAPE lower than the one achieved in the single H3-9 model, which was around 50%. However, with this model, it is important to understand that the geographical precision using zones is low since hundreds of hexbins across Gothenburg are divided into k zones instead. A fewer number of zones contribute to a less geographically precise model, whereas a higher number of zones provide a more geographically precise model. Depending on the model's situation, it is important to know which geographical resolution is required for the model to be useful. If a very geographically precise model is required, this model might not prove very useful since values of k higher than, e.g. 10 produce predictions with high MAPE for most clustered zones.

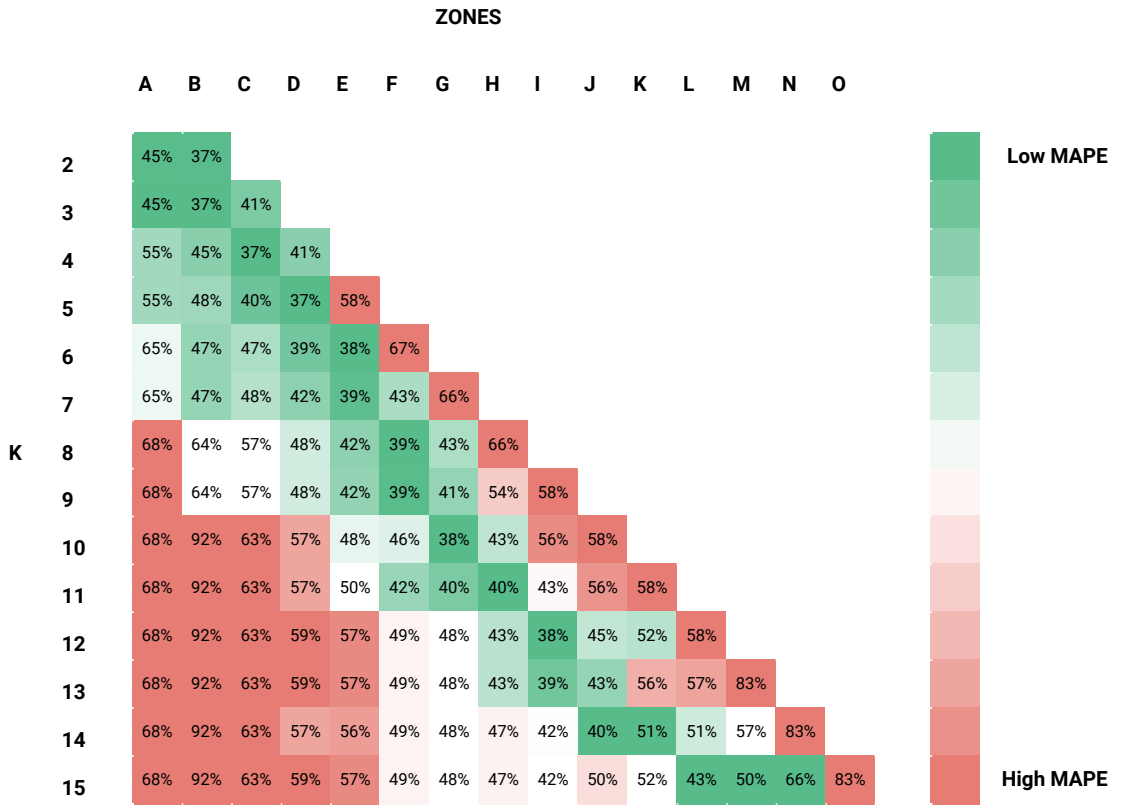


Figure 5.12: Comparison of different clustering's zone MAPE's. Inside each square is the median of the measured MAPE for that zone across 840 24 hour forecasts between February to and including October, using that clustering, where a lower MAPE is considered better. The colour of each square is decided by comparing the MAPE of the same zone in all other clusterings, where the maximum measured MAPE in that zone is mapped to red, and the lowest MAPE for that zone is mapped to green. With these measurements, it is evident that some zones perform better using certain k's. For example, the A zone performs quite well using k's between 2 and 7, however k's higher than 7 result in bad prediction performance compared to using lower k's for the A zone.

Another interesting result in Figure 5.12 is that k-values past 4 produce predictions in the kth zone which are, comparatively, one of the worst predictions in that specific zone, which is further explored and discussed in Section 5.3.1. It is found that at $k = 4$, the kth zone consists of hexes that contribute to almost 30% of all ride starts in total, whereas when $k = 5$, the kth zone suddenly only consists of around 10% of all ride starts. From before, it has been concluded that more ride starts provide more rigorous patterns which are easier to predict, and therefore give a lower MAPE during forecasting. Going from $k = 4$ to $k = 5$ results in the latter model's kth zone only having a third of the ride starts that the prior's kth zone had. This was most likely the issue since the very few ride starts used as training data for the kth zone, when $k = 5$, is not enough for FBProphet to recognize any patterns.

Furthermore, a similar pattern can be seen in the first zone (A), where higher k's result in higher MAPE's for that zone. However, the issue has seemed to be

of similar nature as the number of zones and the value of k increases, the number of hexbins distributed into the A zone decreases. Its been established before that zones containing fewer hexbins do not provide enough ride start data for FBProphet to work with, resulting in worse performance. With this in mind, its apparent why both ends of the zones have a higher measured MAPE than the other zones.

5.3 Markov Models

There were three core areas tackled when working with Markov chains. Demonstrating and confirming the existence of flow, simulating it based on old data and lastly combining the flow with predicted demand through simulation.

5.3.1 Flow Matrices

The first challenge in depicting the city flow with Markov chains was visualising and verifying the results. A first iteration in demonstrating clear flow patterns was achieved by using *Heatmaps*.

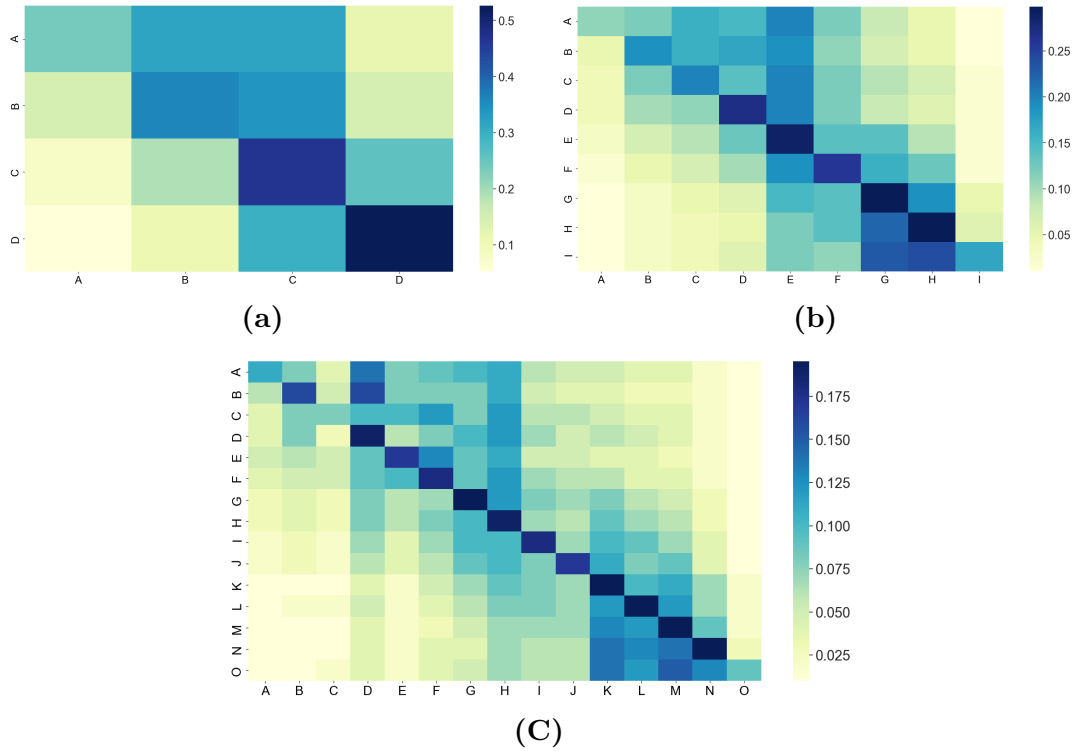


Figure 5.13: Heatmap depicting flow from two months of ride data, Y-axis is start location and X-axis is end with each cell being the probability of travel. (a) with $k = 4$ indicates circular flow between zones C and D while A and B moves supply towards B and C. (b) now has $k = 9$ and demonstrates that two blocks are forming, one above and one below the diagonal, which means that zones A-D and E-I can be seen as two smaller systems of circular flow within the market. At $k = 15$ in (c), a similar cutoff in the middle at zone H and along the diagonal.

With regards to Section 5.2.3 in which the usage of K-mean clustering was tested together with FBProphet, we in a similar manner deducted which k would produce adequate flow patterns when working with Markov chains. These two areas are later combined, and as such, we wanted to find a k that both minimize MAPE and maximize flow. In Figure 5.13, three values of k are compared.

Another aspect to keep in mind as the clustering parameter k changes is how many hexes and the percentage of rides each zone represent. In Figure 5.14 the three k:s respective zone attributes are displayed.

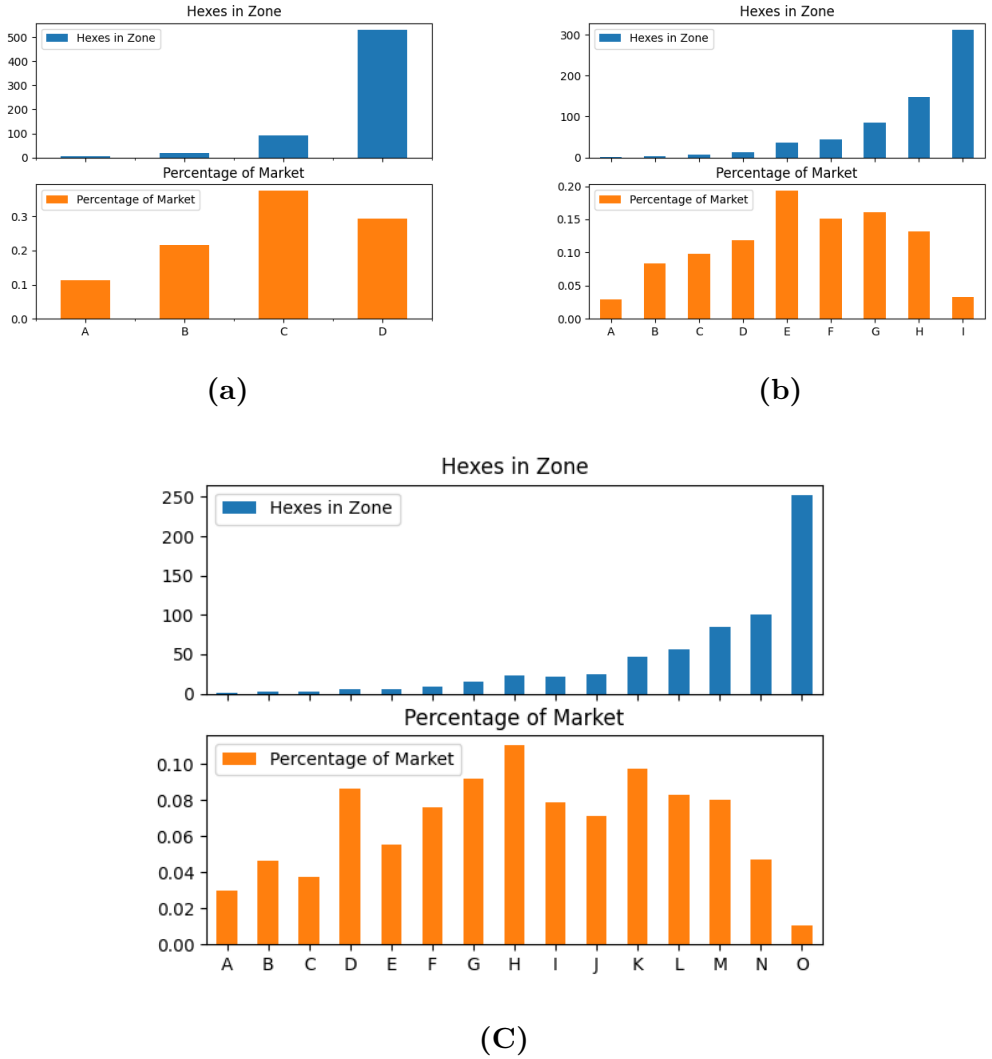


Figure 5.14: Three bar charts representing the distribution of hexes and rides between all zones depending on k. In (a), with only four zones, a clear outlier is zone D with around $\frac{5}{6}$ of the hexes but only 30% of the market. At the other end zones A and B produce the same amount of traffic but is concentrated to a vastly lower number of hexes. (b) instead depicts how more of the traffic is included in the middle zones while A and I are clear edge cases. This is taken to the extreme in (c) when zone O with 250 hexes only amounts to 1% of the traffic, while most central zones account for 4-10% each.

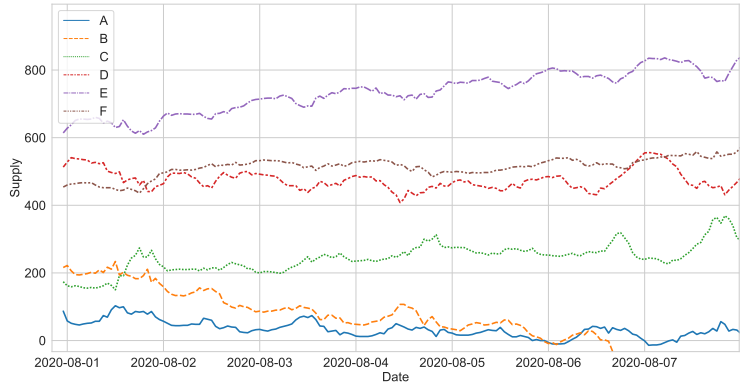
The conclusion drawn from Figure 5.13 and 5.14 is that a flow pattern is visible at all tested values of $k = [4, 15]$, with more granularity the higher it gets. Increasing k also separated the large number of low traffic hexes into the k:th zone, which explains the lower MAPE found in Section 5.2.3 since that zone has low traffic spread over many different geographical hexes. During the following work on simulating the market, the k-mean clustering parameter is set to $k = 6$, which demonstrated low MAPE and clear flow patterns. The comparison of flow between different values of k through mathematical methods was an area left unexplored. Utilising MAPE might have been possible here too, improving the selection of k based on flow throughout the day.

5.3.2 Simulation

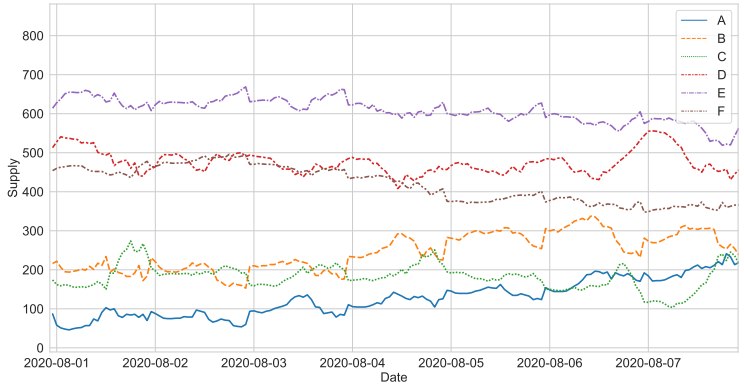
With the simulation of the system, we can start tracking how the supply is presumed to move around the market. This aims to help identify if certain zones require rebalancing (see Section 2.2.1) and prioritize the demand prediction to encourage the correct organic flow present in the market with regards to the current supply. However, supply is a difficult feature to pinpoint, and as such, a random snapshot of the actual supply from March 2021 is used to be able to estimate the flow.

By aggregating the given ride data used in Section 5.3.1 and simulating the exact recorded movement with a generic start distribution, the presumed supply in each zone over time can be tracked. In Figure 5.15 the first month of August 2020 is portrayed with some noteworthy characteristics.

Looking back at Figure 5.14 (a) and (b), zones A and B are presumed to have a higher traffic per hex ratio than the largest zone F. A fundamental action to be made is to move some supply from zones with increasing supply to those with decreasing one. Implementing a very naive static rebalancing in (b) succeeded in keeping the supply in all zones within reasonable levels. A realistic model would recalculate the rebalancing each night with regards to upcoming predictions. Furthermore, since the zones are decided based on the entire period, the grouping could be wrong after passing. The traffic in some hexes might change drastically between two time periods meaning we either should update the zones or change the rebalancing. In Figure 5.16, after the 22 of August, zone C for example switches from slow decline to rapid increase resulting in it being the zone with the most supply at the end of September.



(a)



(b)

Figure 5.15: Estimated supply in each zone during the first week of August 2020. In (a), zones F, E and C seems to have a general positive trend meaning they receive more supply than they export, whilst D looks more stable and A and B decrease below zero. Negative supply in this context indicates that non-organic supply (read rebalancing) has been introduced in meet the demand in these zones. A naive rebalancing at 23:00 each day is implemented in (b), which stabilize the market.

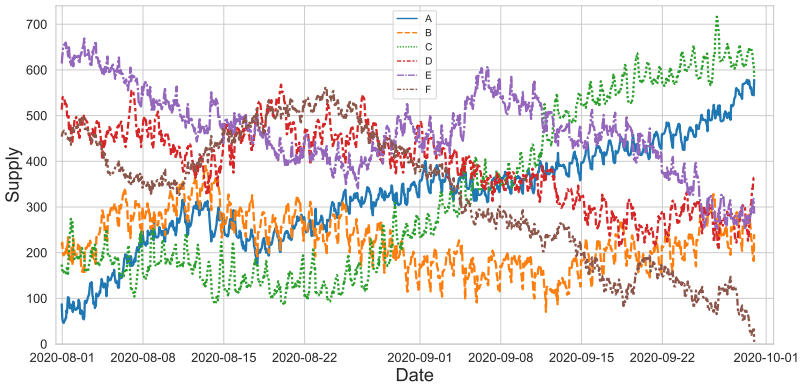
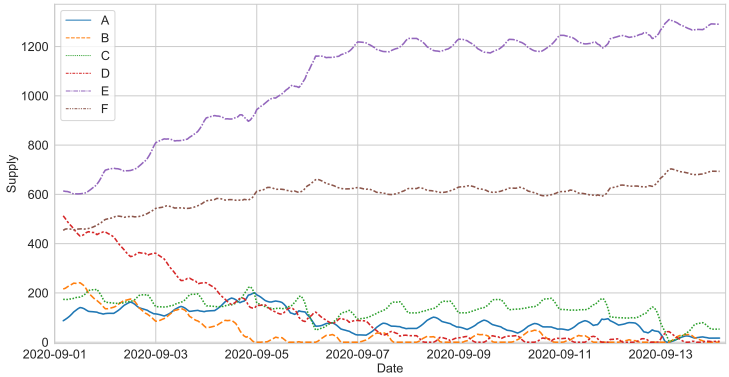
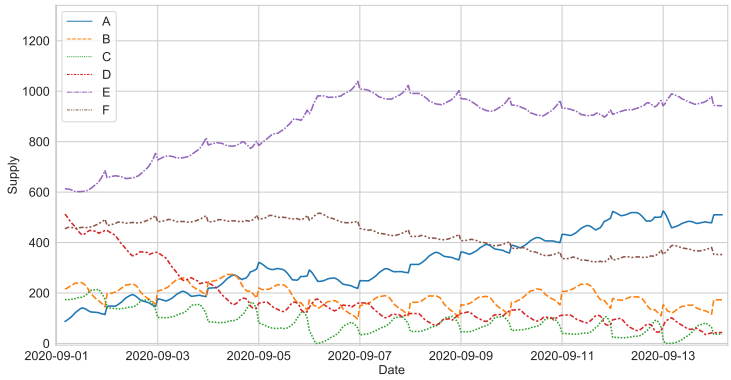


Figure 5.16: Estimated supply in each zone during August and September 2020 with naive static rebalancing at 23:00 each day. During most of August, zone C (the green dotted line) is in balance with the current flow, predicted ride starts and the rebalancing. However, in September it starts to absorb rides rapidly resulting in thrice the amount it started with in August.

Leaving simulating given data behind, combining the derived flow matrices and predicted ride starts allows a simulation to estimate how the supply will move in the future. With predictions on captured or uncaptured demand (see Section 3.1) the simulation can estimate the number of lost rides due to low supply. Making assumptions on lost rides is quite difficult however, as the knowledge that supply is present in a zone is not adequate to guarantee that a vehicle is in the correct hex (especially in zone E or F with over 100 hexes). As such, the simple solution is to, once supply is lower than estimated outgoing and incoming rides that hour, to reduce the outgoing rides from the zone in question (see Figure 5.17).



(a)



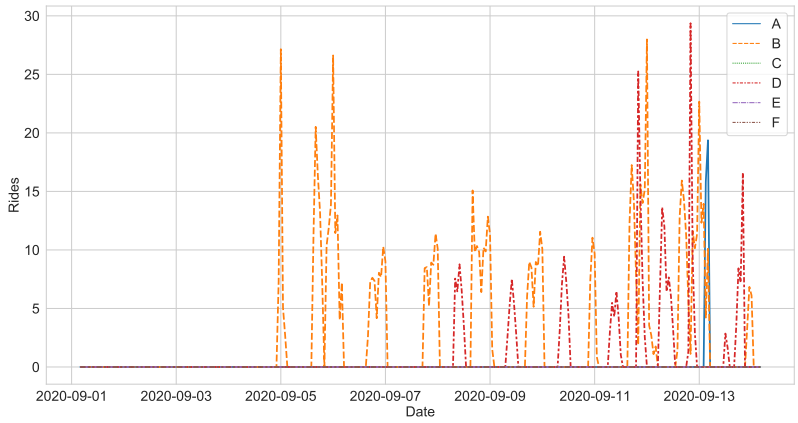
(b) - Rebalanced

Figure 5.17: Estimated supply in each zone during the first two weeks in September 2020 based on predicted demand, with (b) including naive rebalancing at 23:00 each day. In (a), only zones E and F grow in supply during the two weeks while the rest all eventually deplete. The relation between zone A and B is highlighted greatly in (a), at 2020-09-05 (when B has reached zero supply) zone A start to decline instead of grow. This can be tied to the flow perceived in Figure 5.13, where the zones in the upper block A-D demonstrate circular flow meaning that if one zone depletes, the others will follow soon after. The same static rebalancing from Figure 5.16 is applied to (b), which manages to hold almost all zones above zero while hindering the hoarding of supply in E and F to some extent.

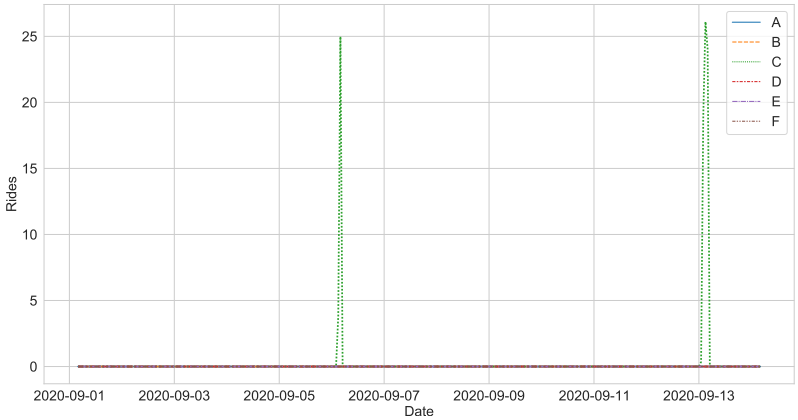
The demand prediction is from FBProphet with a sliding window of 20 days and precipitation accounted for on captured demand in August and September 2020. At two point in Figure 5.17, at 09-06 and 09-13, zone C is expected to lose some ride

(i.e. miss the captured demand) because of low supply. Both zones B and D are also on the verge of depleting, while zone E is a strong absorber. Looking back at Figure 5.15(b), two major differences are that zone A and F both hoard more supply now in Figure 5.17. This is connected to how the k:th zone and zone A have worse MAPE compared to the rest of the zones (see Figure 5.12), in this case resulting in lower expected ride starts in the FBProphet prediction.

Shifting the focus back to the zones which lack supply in Figure 5.17, the instances where the number of ride starts had to be decrease can be seen in Figure 5.18 as Lost Rides.



(a)



(b) - Rebalanced

Figure 5.18: Estimated number of lost rides due to low supply in same scenario as Figure 5.17. In (a), without any rebalancing active, zone B is losing out on rides after the fifth and continues like this all the way out. Both zones A and C lack supply too at the later period. In (b), only zone C lost rides throughout the entire period (and only at two times in total).

As mentioned before, the simulation has some glaring flaws. The prediction from FBProphet is for a 4-hour period, limiting the more delicate steps a simulation would benefit from. The effect of the low MAPE for zones A and F is also prominent when supply is tracked. Supply penalty for large zones with many hexes is not present, which should impact how many rides occurs given that a vehicle is in the correct hex within the zone. This ties to the lost rides seen in Figure 5.18, as the simulation lacks precision in moving supply in small increments resulting in overestimation in supply between each step. A use case for a working simulation would be to prioritise demand which moves supply to some other demand later on in the day. This requires high granular data with regards to time and geographical location. It is also reasonable to presume that rebalancing is a limited resource, meaning reaching all demand predicted might not be feasible nor economical.

5.4 Poisson Model

Most of the time spent on the Poisson model focused on validating the theory and method. Any performance validation in comparison to some ground truth was never made. As such, we consider this model relatively unfinished, however, with further potential. Henceforth, the results and discussion in this section will mostly contain ideas for improvement and model theory validation.

5.4.1 Calculated Demand

The calculated demand, described in figure 5.19, shows for a certain weekday over an observation period of 96 hours a calculated uncaptured demand of 2239 rides (22% of captured). The rate λ described in the figure is not representative of a future period due to a non-applied non-homogeneous interpretation of the process. As such, it is clear, that there exist trends and seasonality that violates the requirements of the process.

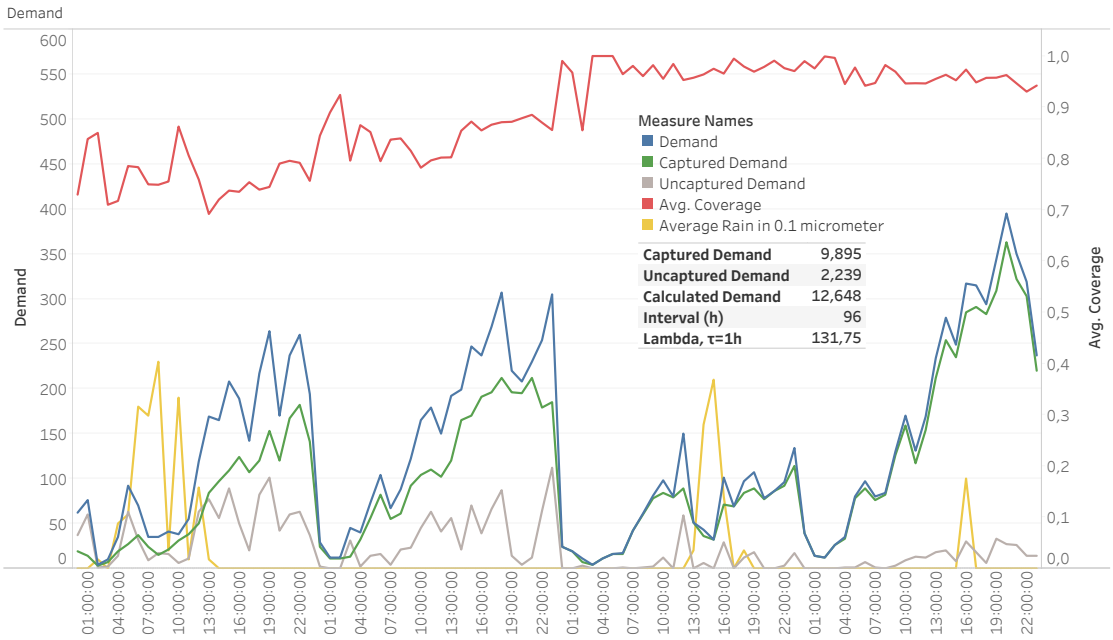
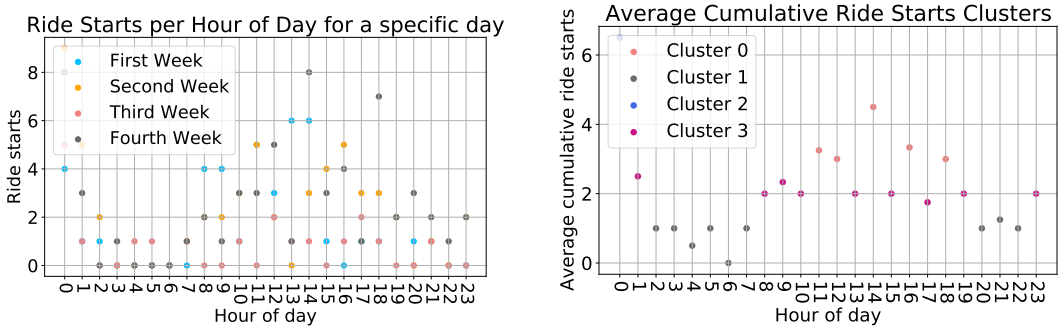


Figure 5.19: Displayed are calculated demand, captured demand, and uncaptured demand and average coverage and rain in micrometres. In the calculated fields, the figure assumes a homogeneous process where the rate is constant over the entire time interval while it is clearly not homogeneous.

5.4.2 Clustering

The reasoning behind clustering is probably lacking. The group remains inconclusive whether clustering is beneficial or not. While clustering similar performing time periods may improve the overall accuracy, it remains uncertain if it yields better results for time periods with significant gaps or specific time periods.



(a) **(b)**

Figure 5.20: The left (a) figure describes the ride starts for different hours of the day for a specific day while the right figure (b) describes the average of the samples in (a) clustered using k-means. The idea here is to capture a naive interpretation of some expected demand, the average for an hour of day, and cluster the naive interpretations with similar other hours during the day.

5.4.3 The assumption of uniform ride start probabilities

The research group believes that smaller hexbins are required to assume uniform ride start probability distribution (see Definition 4.2). A hexbin at resolution level nine has an apothem length of ∼175 meters, centre to vertex length of ∼200 meters, and vertex to vertex length of ∼400m. While other research suggests that the median walking distance is 244m [24], it does not describe whether or not the probability of an app open event or a desire to rent a vehicle is uniform in the geospatial unit.

Furthermore, the current method does not consider geographical or infrastructural constraints that could further increase the required walking distance to the desired vehicle. Although increasing spatial granularity would reduce the assumed required walking distance, it would increase the issue of sparse data representation. The usage of auto-encoders as a method for mitigating issues caused by zero-event-samples have shown promising results in other research [24].

5.4.4 Coverage and captured demand as sole parameters

A fundamental flaw with the method is how demand is calculated (definition 4.14). Although the theory behind stationary increments makes sense, in reality, the distribution is most likely far more non-homogeneous than the current model. The model caused severe outlier results when coverage was extremely small, such as there existed a vehicle in the hexbin only for the first few seconds of the hour, and it generated demand that would often spike far higher than any previously observed numbers in the hexbin or any hexbin in the market.

If the probability distribution is built using samples where coverage is at least some constant, the results may improve inaccuracy. Alternatively, one may look at the probability of said abnormal calculated demand using the Poisson distribution for the hex and set a probability constant where if the results are very improbable, scale the results down.

6
Conclusion

This research aimed to develop a deeper insight into demand forecasting for free-floating micro-mobility services, focusing on FBProphet and Markov Chains Models. The micro-mobility market is still growing quickly, which most likely can impact the movement patterns. The ongoing pandemic could also have impacted our research since the researched data was from 2020. There are still some patterns that can be concluded and further examined. Based on quantitative analysis of market data, it can be concluded that seasonality, precipitation and the overall data coverage are essential factors when predicting the demand. The results indicate that predictions through these means are heavily hindered by sporadic and sparse ride data, forcing the examined models to cluster locations and widen the period to capture meaningful patterns. Future work should explore other methods of aggregating the market, either by spatial or temporal behaviour. Additionally, the research explored the importance of using the correct definitions of metrics. Modifying the data acquisition to fit the desired metric could favour the forecasting, as App opens could have had an important role in uncaptured demand. Furthermore, many other forecasting areas were omitted, such as Neural Network and Deep Learning, which could manage the problem with sparse data better. Further work within these areas can be favourable for future research in demand forecasting for micro-mobility services.

Bibliography

[1] H. Ritchie. (2021). "Google mobility trends: How has the pandemic changed the movement of people around the world?" [Online]. Available: https://ourworldindata.org/covid-mobility-trends (visited on 02/05/2021).

[2] European Environment Agency. (2019). "Monitoring of co2 emissions from passenger cars – regulation (eu) 2019/631," [Online]. Available: https://www.eea.europa.eu/data-and-maps/data/co2-cars-emission-18 (visited on 02/08/2021).

[3] C. Tucker. (2020). "Stockholm-based voi technology lands more than €132 million to boost its geographic and fleet expansion," [Online]. Available: https://www.eu-startups.com/2020/12/stockholm-based-voi-technology-lands-more-than-e132-million-to-boost-its-geographic-and-fleet-expansion/ (visited on 02/05/2021).

[4] Directorate-General for Research and Innovation (European Commission). (2017). "Electrification of the transport system: Expert group report," (visited on 02/08/2021).

[5] T. Serafimova. (2020). "Covid-19: An opportunity to redesign mobility towards greater sustainability and resilience?" (Visited on 02/08/2021).

[6] J. Thorne. (2020). "Lime departs 12 cities, lays off 100 employees in a push for profits," [Online]. Available: https://pitchbook.com/news/articles/lime-departs-12-cities-lays-off-100-employees-in-a-push-for-profits (visited on 02/08/2021).

[7] Chalmers. (2018). "Riktlinjer för kandidatarbete på chalmers - bilaga 7," [Online]. Available: https://student.portal.chalmers.se/sv/chalmersstudier/kandidat-och-examensarbete/Documents/Samhalleliga-etiska-aspekter.pdf (visited on 02/03/2021).

[8] C. Wang, Y. Hou, and M. Barth. (2018). "Data-driven multi-step demand prediction for ride-hailing services using convolutional neural network," [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1911/1911.03441.pdf (visited on 02/03/2021).

[9] A. J. Martinez, O. Stapleton, and L. N. Wassenhove. (2010). "Field vehicle fleet management in humanitarian operations: A case-based approach," [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0272696310000987 (visited on 02/04/2021).

[10] K. Eklund, "Vår ekonomi : En introduktion till samhällsekonomin," in, 13th ed. Studentlitteratur AB, 2013.

[11] J. Fernando. (2021). "Law of supply and demand," [Online]. Available: https://www.investopedia.com/terms/l/law-of-supply-demand.asp (visited on 04/27/2021).

[12] F. Daso. (2020). "Zoba, a harvard spatial analytics startup, offers logistics-as-a-service to all," [Online]. Available: https://www.forbes.com/sites/frederickdaso/2020/02/04/zoba-a-harvard-spatial-analytics-startup-offers-logistics-as-a-service-to-all/?sh=1459a4d765d3 (visited on 02/12/2021).

[13] K. Spieser, S. Samaranayake, W. Gruel, and E. Frazzoli. (2015). "Shared-vehicle mobility-on-demand systems: A fleet operator's guide to rebalancing empty vehicles," [Online]. Available: http://cee.cornell.edu/samitha/Publications/TRB16.pdf (visited on 02/12/2021).

[14] G. Liebmann. (2019). "The art of fleet rebalancing: Our ai tool that increases the utilization of every single vehicle," [Online]. Available: https://medium.com/ubiq/the-art-of-fleet-rebalancing-our-ai-tool-to-increase-the-utilization-of-every-single-vehicle-c86731f98c39 (visited on 02/12/2021).

[15] ——, (2020). "Demand forecasting: Types, methods, and examples," [Online]. Available: https://redstagfulfillment.com/what-is-demand-forecasting/ (visited on 02/12/2021).

[16] R. J. Hyndman and G. Athanasopoulo, "Forecasting: Principles and practice," in, Second. Melbourne, Australia: OText, 2018, ch. 8. [Online]. Available: https://otexts.com/fpp2/ (visited on 04/21/2021).

[17] S. J. Taylor and B. Letham, "Forecasting at scale," 2017. [Online]. Available: https://peerj.com/preprints/3190/# (visited on 04/12/2021).

[18] J. S. Milton and J. C. Arnold, "Introduction to probability and statistics: Prinsicples and applications for engineering and the computing sciences," in, 4th ed. McGraw-Hill Companies Inc, 2003.

[19] R. J. Hyndman and G. Athanasopoulo, "Forecasting: Principles and practice," in, Second. Melbourne, Australia: OText, 2018, ch. 3.4. [Online]. Available: https://otexts.com/fpp2/ (visited on 04/24/2021).

[20] M. J. Garbade. (2018). "Understanding k-means clustering in machine learning," [Online]. Available: https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1 (visited on 05/12/2021).

[21] D. J. MacKay. (2003). "Information theory, inference, and learning algorithms," [Online]. Available: http://www.inference.org.uk/mackay/itprnn/ps/284.292.pdf (visited on 04/23/2021).

[22] Y. Zhou, L. Wang, R. Zhong, and Y. Tan, "A markov chain based demand prediction model for stations in bike sharing systems," *Mathematical Problems in Engineering*, vol. 2018, 2018, ISSN: 2071-1050. DOI: 10.1155/2018/8028714. [Online]. Available: https://doi.org/10.1155/2018/8028714 (visited on 04/08/2021).

[23] A. Faghih-Imani, N. Eluru, A. M. El-Geneidy, M. Rabbat, and U. Haq, "How land-use and urban form impact bicycle flows: Evidence from the bicycle-sharing system (bixi) in montreal," *Journal of Transport Geog-*

raphy, vol. 41, pp. 306–314, 2014, ISSN: 0966-6923. DOI: https://doi.org/10.1016/j.jtrangeo.2014.01.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0966692314000234 (visited on 04/23/2021).

[24] S. W. Ham, J.-H. Cho, S. Park, and D.-K. Kim, "Spatiotemporal demand prediction model for e-scooter sharing services with latent feature and deep learning," *Transportation Research Record*, vol. 0, no. 0, p. 03 611 981 211 003 896, 0. DOI: 10.1177/03611981211003896. eprint: https://doi.org/10.1177/03611981211003896. [Online]. Available: https://doi.org/10.1177/03611981211003896.

[25] I. Brodsky. (2018). "H3: Uber's hexagonal hierarchical spatial index," [Online]. Available: https://eng.uber.com/h3/ (visited on 04/23/2021).

[26] E. Weissten. (2021). "Icosahedron," [Online]. Available: https://mathworld.wolfram.com/Icosahedron.html (visited on 04/23/2021).

[27] K. Sahr, D. White, and A. J. Kimerling. (2003). "Geodesic discrete global grid systems," [Online]. Available: https://webpages.sou.edu/~sahrk/sqspc/pubs/gdggs03.pdf (visited on 04/23/2021).

A

Data Format

A.1 Rides

This data set contains hourly data on how many vehicles that were used in different H3 zoom level 9 hexbins in Gothenburg during 2020. The following points of data exists in the rides data set:

- **H3-9** - Which hexbin that the row of data is referring to. Throughout the report, this is synonymous to *hex* or *hexbin*.
- **Timestamp** - Which date and hour this data point refers to, in the ISO 8601 form.
- **Ride starts** - The amount of vehicles that have been unlocked using the app in this specific H3-9.
- **Ride ends** - The amount of vehicles that have been locked using the app in this specific H3-9.
- **Ride importance** - The total amount of ride starts in this H3-9 during the entirety of 2020. This value is the same for every row that has the same H3-9.

A.2 App Open

The following points of data exists in the app open data set:

- **H3-9** - Which hexbin that the row of data is referring to.
- **Timestamp** - Which date and hour this data point refers to, in the ISO 8601 form.
- **App opens** - The number of app opens that has been recorded in the Voi app.
- **App open importance** - The number of vehicles that have been locked using the app in this specific H3-9.

A.3 Vehicle status

Each vehicle reports in a ten second interval. A vehicle can be in three different states: available, riding and unavailable. When a vehicle is available, we assume that the vehicle is operational and ready to be used by a customer. When a customer is driving a vehicle, the vehicle is in a riding status. If a vehicle is not in a riding status and not in a operational condition, its status will be unavailable. A non

operational condition can be caused by numerous factors including: low battery, hardware failure, software problems etc.

The following points of data exists in the vehicle status data set:

- **Vehicle id** - The unique id of the reporting vehicle.
- **H3-9** - Which hexbin that the row of data is referring to.
- **Timestamp** - Which date and time this data point refers to, in the ISO 8601 form.
- **Status** - The current operational status of the given vehicle

A.4 Idle Time

Idle Time is grouped in hourly interval. The data is given in 50, 90, 95 and 99 percentiles to demonstrate the distribution of idle time recorded for all vehicles in a certain hex during the given hour.

The following points of data exists in the idle time data set:

- **Timestamp** - Which date and hour this data point refers to, in the ISO 8601 form.
- **H3-9** - Which hexbin that the row of data is referring to.
- **percentile 50** - The 50 percentile of the idle time data.
- **percentile 90** - The 90 percentile of the idle time data.
- **percentile 95** - The 95 percentile of the idle time data.
- **percentile 99** - The 99 percentile of the idle time data.

A.5 Weather

Weather data was provided by the Swedish Meteorological and Hydrological Institute (SMHI). The data was collected through multiple sources from SMHI and combined into a single dataframe. The weather stations track Gotheburg's weather in a hourly interval. The following points of data exists in the weather data set:

- **Timestamp** - Which date and hour this data point refers to, in the ISO 8601 form.
- **Temperature** - The temperature over Gothenburg in Celsius.
- **Wind** - The average wind speed at that time in meters per second.
- **Rain** - The amount of precipitation in millimeters.
- **Sun** - The amount of sun time in seconds.

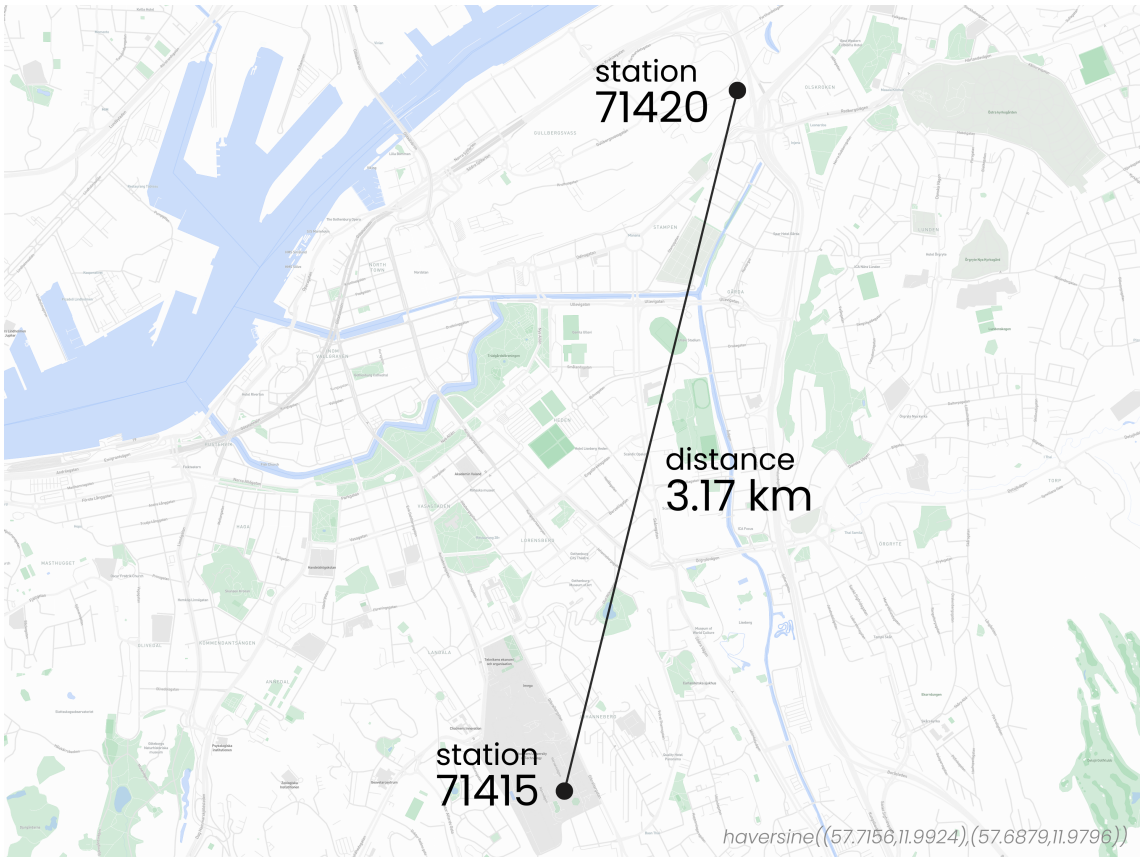


Figure A.1: A map over Gothenburg, Sweden, displaying where the weather stations 71415 and 71420 are located and the distance between them (3.17km). Map credit: ©Mapbox, ©OpenStreetMap

Station ID	Weather Measurement Type
71415	Sunlight
71420	Precipitation
71420	Temperature
71420	Wind speed (scalar)

A.5.1 Data Format

Precipitation data is captured using weather station 71420 (see Figure A.1). The precipitation is measured using automatic measuring stations, where the data represents the accumulated precipitation for one hour, 24 hours a day, in liquid form.

Example of data

Date & Time	Precipitation (mm)
2020-06-03 18:00:00	0.2

B

Uber's H3 Spatial Index

The H3 spatial index introduced by Uber is a discrete global grid system which divides the entire earth into hexagonal shapes [25]. Projecting an icosahedron [26] on the spherical earth results in twenty different two-dimensional planes rather than a single two-dimensional plane, which can be seen in Figure B.1.

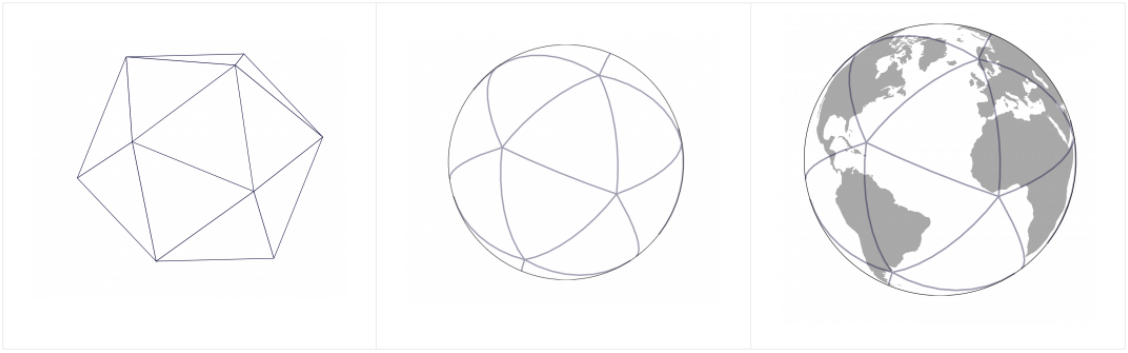


Figure B.1: Projecting earth as a spherical icosahedron. Image credit: Uber Engineering at https://eng.uber.com/h3/.

The icosahedron could be unfolded in many different ways to produce a two-dimensional map each time. However, H3's spatial indexing method does not include the need for unfolding the icosahedron, instead it lays out a grid on top of these icosahedron faces, forming a geodesic discrete global grid [27].

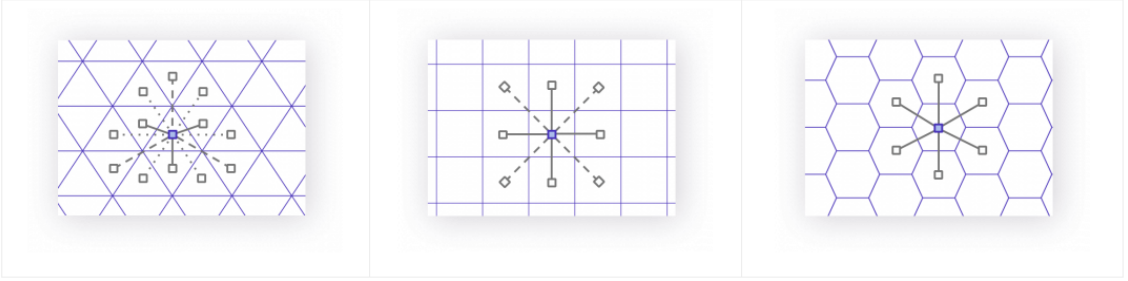


Figure B.2: Distances from the center of a triangle to another (left), a square to its neighbours (middle) and a hexagon to its neighbours (right). Image credit: Uber Engineering at https://eng.uber.com/h3/.

The use of hexagons is crucial for H3. As can be seen in Figure B.2, when using triangles for such a grid system, there would have to be three different distances that

need to be taken into account when performing calculations. When using squares, there are two different distances that need to be considered. However, when using hexagons, the distance between one and its neighbour is always the same, no matter the neighbours placement. This property of hexagons simplifies analysis greatly.

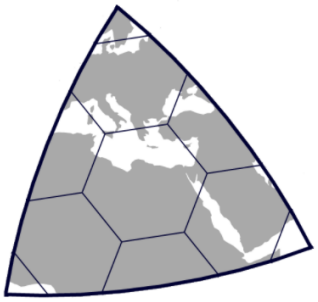


Figure B.3: Grid of hexagons on icosahedron face. Image credit: Uber Engineering at https://eng.uber.com/h3/.

By then dividing each of the icosahedron faces into hexagonal grids, shown in Figure B.3, the most zoomed out level of H3 is constructed. Important to note is that it is impossible to tile an icosahedron exactly using only hexagons, which has been solved by introducing some pentagons in the vertices of the icosahedron. These have all been placed in areas where they will not matter though, e.g. in the ocean, which means that the H3 spatial index still works as intended.

H3 supports different zoom levels, and the one shown in Figure B.3 is the most zoomed out level. Each finer resolution, or zoom level, results in cells which have one seventh the area of the coarser resolution. However, a single hexagon cannot be exactly subdivided into seven smaller hexagons, which means that the finer cells are only approximately inside a coarser cell. All of these cells are unique identifiable using a hexadecimal identifier.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

UNIVERSITY OF
GOTHENBURG

CHALMERS