Effects of data-driven forecasting models on controlling energy management systems

7S45M0 – Master thesis

Author:	W.P.A. (Ward) Somers	
Student number:	1387243	
Graduation supervision committee:	prof ir. W. (Wim) Zeiler	(TU/e & Kropman)
	ir. W. (Waqas) Khan	(TU/e)
	dr. ir. R. (Roel) Loonen	(TU/e)
	ir. K. (Kevin) de Bont	(Kropman)
	ir. J. (Joep) van der Velden	(Kropman)
University:	Eindhoven University of Technology	
Department:	Built Environment	
Master track:	Building Physics and Services	
Date:	13-04-2023	

Preface

This report represents my thesis for the Master of Building Physics and Services at the Eindhoven University of Technology. This thesis is a collaboration between the Building Services chair at the Built Environment faculty from TU/e and Kropman. The research focuses on data-driven forecasting models to enhance the controlling ability of energy management systems. My interest in data analytics, time series forecasting, energy flexibility, and energy management arose during my earlier master projects, which were also related to these topics, and the course “Intelligent buildings” where I learned more about Python and machine learning. Working on my thesis made me even more curious about the field of energy management systems.

Hereby, I would like to thank all the people who have assisted me during the graduation process. First of all, I would like to thank prof. ir. Wim Zeiler and dr. ir. Roel Loonen from TU/e and ir. Joep van der Velden from Kropman for facilitating my graduation project. I would like to thank the Ph.D. candidate ir. Waqas Khan from TU/e for his valuable support as the first supervisor. My learning curve in Python, data analytics, and machine learning would not have been possible without his narrow involvement. I would also like to thank dr. ir. Shalika Walker from TU/e for her continuous support, valuable feedback on the content and process, and for acting as a connector with other people from Kropman. I would also like to thank ir. Kevin de Bont from Kropman for helping me with questions related to the case studies and for the interesting conversations which triggered my thoughts and helped me in paving my way toward the end goal. Moreover, I would like to thank Bram Hermans from TU/e for sharing his knowledge about control engineering and for allowing me to use his MPC model as starting point. I would also like to thank Koen Reinders from TU/e for introducing me to the Python optimization package Pyomo and for all the valuable sparring sessions. Finally, I would like to thank my parents for their immense support throughout the years. Not only during my master's but during my entire educational journey. Without their help, I would not have been able to achieve this milestone.

Ward Somers
Uden, April 2023

List of Figures

List of Tables

Abbreviations

A	Amperes	NHTS	National Household Travel Survey
AC	Alternating current	NRMSE	Normalized Root Mean Squared Error
ACF	Auto Correlation Function	OBC	On-Board Charger
ATES	Aquifer Thermal Energy System	OCPI	Open Charge Point Interface
B4B	Brains for Buildings	OEM	Original Equipment Manufacturer
BESS	Battery Energy Storage System	PID	Proportional-Integral-Derivative
BMS	Battery management system	PPR	Power Peak Reduction
BPNN	Back Propagation Neural Network	PPRP	Power Peak Reduction Percentage
CPO	Charge Point Operator	PV	PhotoVoltaic
DC	Direct Current	RES	Renewable Energy Sources
DNO	Distribution Network Operator	RF	Random Forest
DR	Demand Response	RPC	Remote Procedure Call
DSM	Demand Side Management	SCSP	Smart Charging Service Provider
DT	Decision Tree	SHAP	SHapley Additive exPlanations
EMS	Energy Management System	SoC	State of Charge
EU	European Union	SoH	State of Health
EV	Electric Vehicle	TEMP	Temperature
EVSE	Electric Vehicle Supply Equipment	UNFCCC	United Nations Frame Convention on Climate Change
GHG	GreenHouse Gas	V	Volts
GHI	Global Horizontal Irradiance	V2G	Vehicle-to-Grid
HP	Heat Pump	V2X	Vehicle-to-X
HTML	HyperText Markup Language	VIF	Variance Inflation Factor
ICEV	Internal Combustion Engine Vehicle	XGB	eXtreme Gradient Boosting
KNN	K-Nearest Neighbor		
kWh	kiloWatt hour		
LSTM	Long Short-Term Memory		
MCS	Monte-Carlo Simulation		
ML	Machine Learning		
MLP-ANN	MultiLayer Perceptron Artificial Neural Network		
MPC	Model Predictive Control		

Abstract

EVs are recognized as valuable assets in microgrids since they have a significant energy flexibility potential. Smart charging of EVs can reduce the mismatch between energy demand and supply. Preferably, detailed user-specific information is utilized to optimally schedule EV charging while not compromising the user's comfort. However, such information is often not available in reality. Therefore, this study investigates if data-driven forecasting models can overcome the lack of user-specific EV information when applying MPC for smart charging. In doing so, spatial effects on the forecasting performance are evaluated in two case studies, and multiple data-driven forecasting methods – averaged-driven, XGB, and RF – are evaluated. The results show that case studies with larger EV populations are favorable to forecast since they are associated with less randomness and more repetitive patterns and that RF recursive multi-step forecasting is most suitable considering the trade-off performance versus complexity. It is found that the application of RF recursive multi-step forecasting in combination with MPC can forecast and control the aggregated EV fleets properly without user-specific information. Therefore, an aggregated approach significantly improves the applicability of the EMS because the system is independent of EV driver cooperation. As a result, the EMS becomes more robust, is not prone to privacy issues, and simplifies the optimization problem by creating a single-objective instead of a multi-objective optimization problem in which each EV would be considered separately. The promising results of this study pave the way to improve the forecasting models for EMSs in further research.

Table of Contents

1. Introduction

1.1. Research context

The awareness of the urgency to mitigate global warming has grown significantly in the last decade. In 2015, The Paris Agreement on climate change was adopted by 194 members of the UNFCCC aiming to limit global warming well below two degrees Celsius, preferably to 1.5 degrees Celsius, compared to pre-industrial levels [1]. To achieve this objective, the EU member states agreed to reduce GHG emissions by fifty-five percent in 2030 compared to 1990 and to become carbon neutral in 2050 [2]. This led to the rapid growth of RES world-wide. In the Netherlands, renewable energy generation grew between 2015 and 2021 by 116 percent of which solar photovoltaic and wind energy grew with respectively 920 percent and 138 percent as shown in Table 1.1. The rapid growth of these technologies is essential for the sustainability of the electricity mix, however, intermittency of several RES introduces challenges to maintaining the stability of the power grid. Especially, the generation of solar and wind energy fluctuates significantly due to its dependency on weather conditions and seasonality. In 2021, the aggregated solar and wind energy already represented thirty-one and twenty-six percent of the total renewable energy generation and the total primary electricity consumption respectively [3]. In the same year, a renewed agreement was reached by 197 members of the UNFCCC, referred to as the Glasgow Climate Pact, to sharpen targets from the Paris Agreement in the near future. A new target is set for the reduction of GHG emissions in 2030. Instead of reducing fifty-five percent in comparison to 1990, the Glasgow Climate Pact aims to reduce GHG emissions by forty-five percent in comparison to 2010 [4]. Therefore, the share of renewable energy will become even more dominant in the Dutch electricity mix, which will increase the variability of the energy supply.

On the demand side of the electricity network, the consumers, new developments arise to reduce GHG emissions. The built environment is electrifying by introducing HPs as a replacement for traditional gas-fired boilers [5]. Moreover, the mobility sector is electrifying by phasing out traditional ICEVs and introducing EVs [6]. In recent years, the adoption of HPs and EVs grew exponentially. It is expected that the growth of HPs and EVs will continue up to five million and nine million respectively in 2050 [7], [8]. Unfortunately, the energy demand of HPs and EVs also has a certain variability [9]. As a result, these developments cause a significant increase in power demand and demand variability whereupon the grid is not designed.

The increasing variability on both supply and demand sides makes it more complicated to match the energy supply and energy demand. Traditional power plants ideally adapt their energy supply, depending on the variable renewable energy generation, to prevent a mismatch between the total energy supply and the energy demand. However, slow-responding power plants, such as coal and nuclear plants, have limited flexibility [10]. Therefore, depending on the conditions, undesired measures are sometimes necessary to maintain the balance of the power grid. In times with a surplus of renewable energy generation, RES are curtailed to cap overproduction. This measure is not desirable since potential renewable energy is not harvested. In situations when the energy demand is larger than the capacity on the supply side, wholesale customers are compensated for lowering their energy demand - also known as DR. These measures are currently inevitable since electricity should be used instantly and because energy storage and DSM are still lacking nowadays [11].

On a smaller scale, office buildings can be seen as microgrids wherein similar mismatches occur between energy supply and demand due to the operation of the building, PV systems,

and EVs. In such a microgrid, electricity is supplied by the PV system and the power grid, where the office building and the EVs are the consumers. The most remarkable mismatch is notable between EV charging and PV generation. EV charging loads often peak in the morning due to uncoordinated charging and typical occupancy patterns, whereas PV generation mainly occurs in the afternoon. Therefore, the microgrid's power demand is significantly amplified in the morning and reduced in the afternoon. These occurrences increase the demand variability by increasing the difference in power consumption during on-peak and off-peak periods.

Table 1.1: Development of primary energy sources in the Netherlands [3].

Energy source	1990 [PJ]	2015 [PJ]	2021 [PJ]	Δ'90-'21 [%]	Δ'15-'21 [%]
Coal and coal products	366.9	463.6	235.4	- 35.8	- 49.2
Crude and petroleum products	1,047.5	1,143.3	1,101.9	+ 5.2	- 3.6%
Natural gas	1,325.0	1,211.8	1,255.6	- 5.2%	+ 3.6%
Renewable energy	31.2	158.0	341.4	+ 994.2	+ 116.1
Hydro power	0.3	0.3	0.3	N/A	N/A
Geothermal heat	0.0	2.4	6.4	N/A	+ 166.7
Ambient energy	0.0	5.7	15.9	N/A	+ 178.9
Solar thermal	0.1	1.1	1.2	+ 1,000.0	+ 9.1
Solar photovoltaic	0.0	4.0	40.8	N/A	+ 920.0
Wind energy	0.2	27.2	64.7	+ 32,250.0	+ 137.9
Biomass	30.6	117.3	212.2	+ 593.5	+ 80.9
Electricity and heat	31.3	33.1	2.6	- 91.7	- 92.1
Nuclear energy	38.0	39.2	37.3	- 1.8	- 4.8
Other energy sources	14.8	45.1	49.5	+ 234.5	+ 9.8
Total	2,854.7	3,094.1	3,023.7	+ 5.9	- 2.3

Fortunately, EVs can offer significant load flexibility with smart control and thereby regulate the power demand of a microgrid, aiming to avoid grid congestion by reducing power peaks and increasing the self-consumption of intermittent renewable energy [12], [13]. Ideally, EVs can act as BESS where they charge during off-peak periods and discharge during on-peak periods. Nowadays, most EVs are limited to unidirectional energy exchange

(charging) due to technical limitations. This limitation is caused by the widely used unidirectional OBC, which cannot convert energy from DC to AC. This conversion is necessary to enable bidirectional energy exchange (discharging) [14]. Nevertheless, smart control of EV charging, smart charging, can play a significant role in the field of DSM. Various DSM techniques are illustrated in Figure 1.1 of which load shifting is a commonly used technique with smart charging.

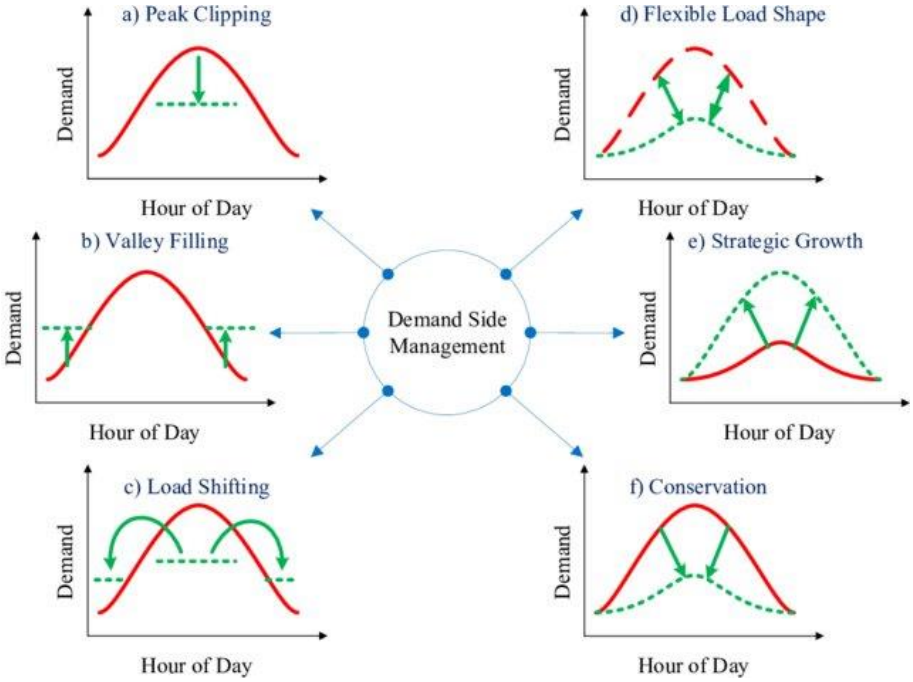

Figure 1.1: Overview of DSM techniques [15].

The load flexibility potential of EVs in microgrids results in an increasing interest in the application of MPC for smart charging. MPC is a control technique that is not only based on measurements, such as PID control but also considers forecasts to anticipate the future [16]. In discrete time, MPC considers a specific control and forecasting horizon – equally sized in Figure 1.2 – containing a specific number of instances that depend on the sample time. Based on the forecasts, the MPC determines an optimal control input trajectory by solving the cost function of a constrained optimization problem within the defined horizon. Only the first control input of the trajectory is implemented whereas the others are discarded. After implementing the first control input, the new state is measured and the optimization problem can be solved for the next instance. Not only the control inputs are repeatedly updated, but also the forecasts are updated every instance. The receding horizon principle is often applied with MPC. Hereby, the control and forecasting do not decrease in time but only shift in time [16].

1.2. Research problem

The goals of the Paris Agreement are not feasible by simply introducing and increasing sustainable technologies. Appropriate system integration is necessary for optimal interaction between subsystems in microgrids or energy hubs. Understanding the operations of these subsystems is key to system integration. In doing so, EMSs are a necessity because they provide insights into the operations by collecting and storing historical data. Furthermore, EMSs can monitor subsystems in real-time. The combination of historical databases and real-time monitoring allows for the application of smart control algorithms to optimize the

interaction between subsystems. EVs are recognized as controllable subsystems which are vital in the transformation of the existing power grid towards a smart grid. To implement sophisticated smart charging control strategies, it is essential to have detailed knowledge of EV charging behavior.

Ideally, user-specific information such as arrival time, initial SoC, minimal energy demand in kWh, battery capacity in kWh, maximum charging power in kW, and departure time is available prior to a charging session. With detailed user-specific information, a smart charging controller can optimize EV charging sessions optimally considering the power consumption of other subsystems within a microgrid and without compromising users' comfort. Additionally, it would be preferable to obtain this information thirty-six or forty-eight hours prior to the charging sessions, because it allows a controller to anticipate electricity prices from the day-ahead electricity market. However, in reality, such detailed information is not available, especially not so far in advance. Moreover, commonly used AC-charging protocols do not communicate SoC values. Other user-specific information is often not available due to a lack of data-collecting systems. Log data, EV current data, transaction data, and power consumption data are typically available. Transaction data are summaries of individual charging sessions, containing: EV ID, plug-in time, plug-out time, and charged energy. Power consumption data is monitored at the individual (charging pile) and at an aggregated level and is available in real-time or as historical data. The transaction data and power consumption data can be obtained from the CPO or are directly accessible via the relatively new OCPI protocol.

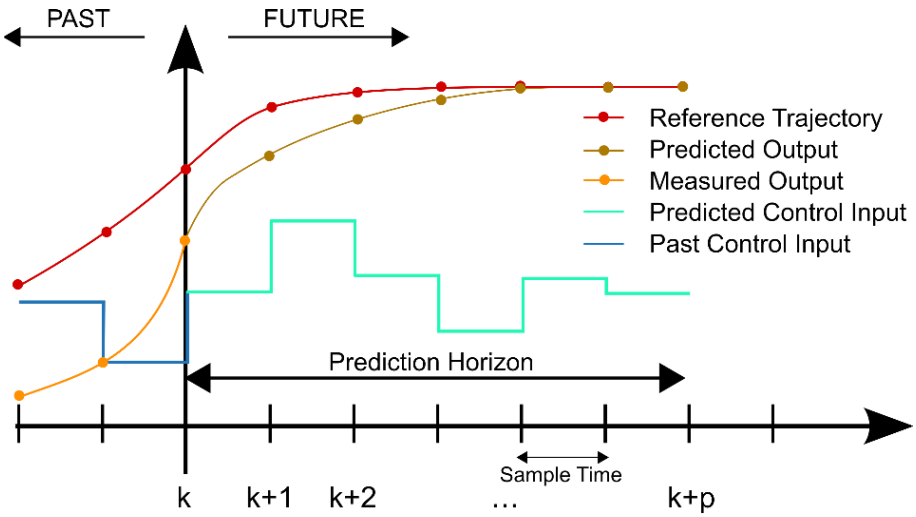

Figure 1.2: A discrete MPC scheme [17].

Therefore, earlier research done by the authors [18], [19] investigated various data-driven techniques, which do not require the cooperation of EV drivers by focusing on actual available data, aiming to overcome the lack of detailed user-specific information at workplaces. The findings from that research indicated the following:

- Load profiles are static, which means that they can capture general behavior. However, the main disadvantage is their inability to anticipate deviations in the presence and energy requirement of individual EVs. These deviations can have a significant impact on smaller case studies [18].
- Accurate individual EV forecasting is impossible without user-specific information [19].

- Individual EV forecasting is affected by disturbances such as arbitrary charging, the functioning of the BMS, and zero-inflated data [19].

It is expected that data-driven forecasting models should be able to forecast aggregated EV charging loads more accurately since less randomness is generally involved in larger populations. However, the threshold of the population size to enable accurate forecasting is currently unknown. Moreover, it is unknown what kind of data and features are necessary to accomplish accurate forecasts and how these forecasts influence controllers that are integrated into EMSs, especially the more advanced MPC.

1.3. Research objective

Since data-driven forecasting models are not able to accurately forecast individual EV charging loads due to lacking presence information and randomness, several topics remain unclear and require further investigation.

The first topic of interest is related to the forecasting performance of ML forecasting models at other spatial levels, aiming to forecast the aggregated EV charging load at a comparable case study, but with a significantly larger EV fleet. In such a case study, it is expected that the presence feature is redundant to forecast the aggregated EV charging load. Moreover, a significantly improved forecasting accuracy is expected due to the smaller impact of randomness from individual EVs. However, these expectations are not verified yet, thus, an investigation is necessary to accept or reject our hypothesis.

The second topic of interest is related to a comparison between averaged- and ML-driven forecasting models, focusing on the trade-off between complexity and performance. In this study, averaged-driven forecasting models are characterized as average load profiles per day type. Herein, a distinction is made between weekdays and holidays.

The final aim of this project is to create recommendations about required data, which features to use as input for EV charging load forecasting models, and how they affect EMSs with the main focus on MPC.

"What are the effects of data-driven forecasting models on the controlling abilities of energy management systems and what are the requirements for the realization of these models?"

To answer the main research question, the following subquestions are formulated to divide the problem into more manageable and specific parts:

- Which set of features and algorithms leads to the most accurate ML-driven EV forecasting model?
- How do ML-driven forecasting models perform in comparison to averaged forecasting models, considering the trade-off between performance and complexity?
- Are spatial effects noticeable when comparing the forecasting performance of small fleets with large fleets?
- What are the requirements to implement a forecasting model into an MPC and what are the different effects of the various forecasting models on the MPC?

Additionally, it is important to state that this research is part of the B4B consortium. B4B is a multi-year and multi-stakeholder project where educational institutions and companies from the industry collaborate to develop methods to harness big data from smart meters, building management systems, and the Internet of Things devices, to reduce energy consumption, increase comfort, respond flexibly to user behavior and local energy supply and

demand, and save on installation maintenance costs. This research will contribute to work package two: "Intelligent energy flexibility control strategies".

1.4. Research outline

The remainder of this thesis is presented in Figure 1.3 and is as follows. Chapter 2 presents the literature review about the application of forecasting models in MPC. Chapter 3 describes the methodology behind the constitution of the forecasting models and the MPC. Chapter 4 presents the exploratory analysis of the data, the performance of the various forecasting models, and the implementation of the forecasting models into the MPC. Chapter 5 discusses the limitations and assumptions of this study. Finally, chapter 6, concludes the findings of this thesis and provides recommendations for further research.

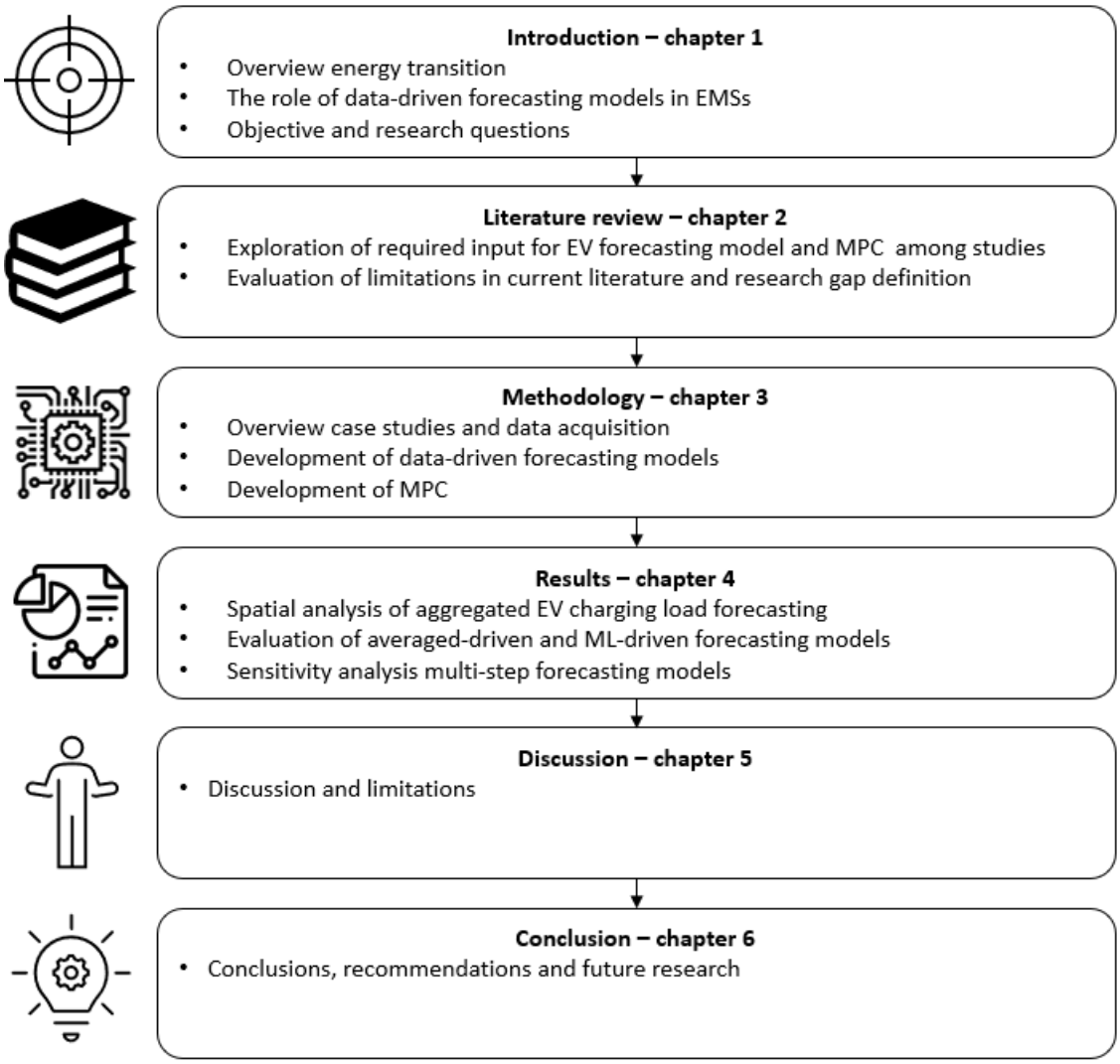

Figure 1.3: Research outline.

2. Literature review

In the literature, various objectives exist for the application of MPC within the field of smart charging. According to the review in [20], the most frequently used objectives are related to the maximization of profits, power quality, reliability, share of renewable energy sources, load factors, and resiliency and the minimization of operational costs, power losses, voltage deviations, and emissions. Despite the different objectives among studies, the similarity is the necessity of appropriate EV input data. Therefore, this review will mainly focus on the used data, features, and forecasting models that are used to generate input for the MPC.

In [21], they aim to minimize the charging costs of four EVs by applying load shifting with MPC. The arrival and departure times are 09:00 and 18:00 respectively and hold for each EV. The initial SoC values are assumed for each EV, whereas all four EVs should have an SoC of eighty percent at departure. Moreover, the battery capacities and charging powers are assumed to be twenty-four kWh and 6.6 kW respectively. The optimization problem can be perfectly solved with the assumed user-specific information. The availability of user-specific information prior to charging sessions is not realistic and therefore the main limitation of this study.

In [22], they aim to maximize operational revenue while handling high power loads in the grid with the application of MPC. Real-world EV charging data from a university campus with forty-four charging points is used for the charging demand forecasts. The data consist of the requested time, duration, and energy consumption of each charging session from the year 2019. Initially, the number of charging requests is determined based on week-ahead maximum and average values, which are obtained from the historical data. The forecasting horizon equals twenty-four hours with a sample time of fifteen minutes. As a result, the receding horizon of the MPC contains ninety-six instances. For the first instance, the maximum week-ahead number of requests is implemented, whereas the week-ahead average number of requests is taken for the other ninety-five instances. After an EV plugs-in, it is assumed to charge uncoordinated with a power of 7.2 kW. Using week-ahead maximum values is a limitation since it always assumes a worst-case scenario. With average values, trends are not immediately visible due to the weight of historical data. The larger the data set, the smaller the impact of trends on the average.

In [23], electricity bills and peak powers are minimized at a hospital with MPC. Real-world EV charging data used from the year 2019, containing information about ID, plug-in time, plug-out time, and energy consumption. SoC values and battery capacities are unknown, therefore, assumptions are made based on backward-engineering. The maximum charging power of an EV is based on the average power consumption, which is determined by dividing the energy consumption by the connection time (plug-out time minus plug-in time). The battery capacity is retrieved from the maximum registered energy consumption. Initial and final SoC values are calculated based on the consideration that plug-in times, plug-out times, charging powers, and battery capacities are known. With data analytics, it is possible to estimate the charging power and battery capacity of individual EVs. However, plug-in times, plug-out times, and SoC values are exposed to uncertainty and are difficult to estimate. Therefore, the chosen approach is suitable for simulation purposes, but not for real-life implementation.

In [24], the aim is to minimize operational costs at a public parking lot with MPC. Day-ahead aggregated EV charging loads are determined with the assumption that EVs should be

fully charged at departure. To do so, one by 'H' vectors are created, where 'H' represents the maximum number of charging piles. Each instance, the charging pile, contains four values that correspond to the plug-in time, initial SoC, battery capacity, and plug-out time. These values are determined with probabilistic distribution models. The vector represents only one hour of the day, thus, twenty-four one by 'H' vectors are created in total to generate day-ahead aggregated EV charging loads. When EVs are connected, battery state information such as Soc, SoH, voltage, and current is communicated as well as user preferences. The main limitations of this study are the large time interval of one hour, the application of probabilistic distribution models to create day-ahead charging load forecasts, and the use of SoC values in real-time. The utilization of probabilistic distribution models is not desired because they assume data sets to have a fixed and stationary underlying distribution. In many real-world events, data do not follow a specific distribution, are not stationary, and change over time due to seasonality, trends, and external factors. Moreover, probabilistic distribution models are not suitable for complex nonlinear forecasting problems. The aforementioned aspects in combination with a probabilistic distribution forecasting model can result in poor forecasting performance [25].

The study [26] aims to limit the impact on the power grid and the costs for customers by scheduling EV charging at residences. The optimization problem of individual EVs starts at arrival since this research assumes that arrival times cannot be predicted due to randomness. The communicated initial SoC, battery capacity, and departure time together with the assumption that EVs should be fully charged before departure are used as input for the optimization problem. The main limitation of this study is related to the exclusion of EVs that arrive in the near future. As a result, the optimization model provides sub-optimal outputs, because it only considers the charging loads of present EVs.

In [27], they aim to minimize the demand variability of a residential area with smart charging. To do so, MPC with a receding horizon is applied. The limitations of this study are threefold and relate to the data availability, the forecasts, and the considered periods. In this study, the initial SoC and final SoC are known at arrival. In a decentralized control system, where the EV driver is in control, it is plausible that SoC values are used. However, this study focuses on a centralized control system, where an aggregator collects data and controls it on an aggregated level. Herein, the availability of SoC values is not likely. Moreover, historical data with the addition of randomness are used as forecasts. This forecasting method is not viable for real-life implementation, because the added randomness might distort patterns in the historical data. Another limitation of this study is related to the inclusion of certain periods. Only weekdays are considered since weekend days and holidays are accompanied by more uncertainty.

In [28], a continuous-time MPC is deployed to minimize costs in a power system. Once an EV connects, it communicates information about its desired amount of charged energy, departure time, and maximum allowable time to delay charging. The availability of user-specific information prior to a charging session is not realistic since data-collecting systems are often lacking. Moreover, future arrivals within the control horizon are considered but are based on NHTS in combination with Poisson distributions. In [29], NHTS data with distribution models are also used to derive the arrival time, departure time, number of daily trips, and distance of each trip from individual EVs.

In [30], they aim to maximize the revenue for DNOs by charging EVs in low price periods without compromising EV drivers' requirements. The EV charging demand is modeled with MCS to deal with the stochastic characteristic of EV charging since real-world data are not

accessible. In the MCS, the arrival time, connection time, SoC values, and travel distances are determined for each discrete time instance within the horizon. The application of MCS is an appropriate method in simulations to proof the concept, but cannot be implemented in an EMS.

Study [31] aims to optimize residential EV charging. EV drivers are required to communicate their travel types for the next day. The MPC is obligated to provide a sufficient SoC at 06:00 the next day to ensure that the desired travels can be fulfilled. The proposed cooperation of EV drivers results in valuable information and might be viable in the residential domain, but is often not applicable in real scenarios at other places where data-collecting systems are lacking. Study [32] is similar to [31] because both studies aim to provide a sufficient SoC to individual EVs in the residential area before 06:00. In contrast to [31], study [32] uses probabilistic distribution models to determine the arrival times and initial SoC instead of data-collecting systems.

In [33], a multi-objective optimization model is established for minimizing the transmission losses, operating costs, and carbon emissions of multiple microgrid systems. In this study, deep learning algorithms are considered to forecast the day-ahead EV charging load. Initially, a BPNN-based forecasting model is trained on a data set containing fifteen days with an hourly time interval. Additionally, the forecasting performance is improved with the introduction of an LSTM-based forecasting model that is used to forecast the errors from the BPNN forecasting model. The main limitation of this study is the relatively low time resolution of one hour. A higher resolution, e.g. fifteen minutes, is preferred to capture more fluctuations in the EV power consumption.

In [34], they aim to maximize revenue by scheduling EV charging at a workplace parking lot that is powered by solar energy and the grid. Connected EVs must communicate their arrival time, departure time, and charging requirements to the central controller. Statistical distribution models are utilized to estimate the upcoming EVs. The main limitation of this study is the use of a probabilistic distribution model as a forecasting model. Additionally, the cooperation of EV drivers is required to collect user-specific information to solve the optimization problem.

Other studies [35]–[39], where MPC is not used as a control technique, apply similar methods to overcome the lack of user-specific information Studies [35]–[39] also rely on probabilistic forecasting models and/or the assumption that user-specific information is available to generate input for the controllers.

In the reviewed literature, several approaches related to EV charging load forecasting are observed. User-specific information such as plug-in time, plug-out time, energy demand, SoC, battery capacity, and charging power is often used as input for the MPC to deal with individual EV uncertainties in the optimization problem. They assume to have perfect forecasts by using historical data or collected user-specific information as input. In those cases, it is assumed that user-specific information is available prior to a charging session or at the moment an EV plugs-in. This allows the controller to optimally control the present EVs. Herein, upcoming EVs are not always considered. The upcoming EVs are preferably implemented to avoid sub-optimal control. To do so, probabilistic distribution models are frequently used to forecast the upcoming EVs. In reality, none of the user-specific variables are known in advance. It is impossible to deal with individual EV uncertainties without the cooperation of EV drivers and user-specific data-collecting systems. Hence, the approaches in existing literature are not realistic for real-life implementation Furthermore, studies tend to simplify their simulations by excluding periods that are exposed to more uncertainty, such as weekend days and

holidays. Another simplification is related to the applied time resolution. Several studies utilize hourly intervals, resulting in a decreasing complexity since fewer fluctuations are captured. Besides, EVs in residences or residential areas are often used as a case study, wherein overnight charging is the most common practice. Overnight charging is associated with less uncertainty due to significant idle times, less random behavior of EV drivers, and the absence of solar energy generation. Therefore, this study will contribute to the field of energy management by providing realistic ML-driven aggregated EV charging load forecasting models with a high resolution of fifteen minutes and a forecasting horizon of twenty-four hours by utilizing accessible data at workplaces. Complementary, guidelines about the necessary data, features, and forecasting algorithms to obtain optimal forecasting will be provided.

3. Methodology

The overall method followed in this project is demonstrated in Figure 3.1 and described in the following sections.

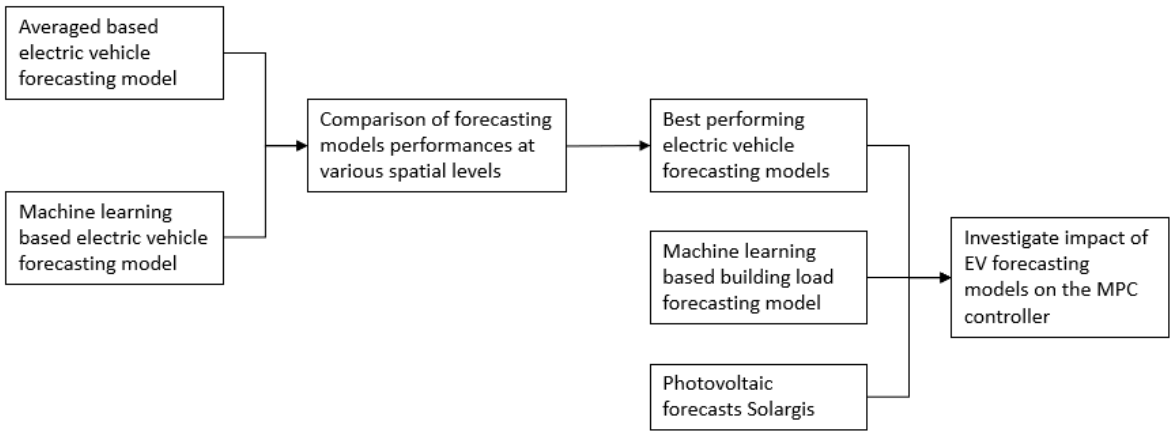

Figure 3.1: Stepwise overview of research.

3.1. Case studies

In this study, two case studies are considered. The characteristics of both case studies are briefly explained in this subchapter.

Building A

Building A is an office building in Breda, the Netherlands. It was built in the nineties and has an approximate floor area of 1500 square meters. The building can be described as a traditional Dutch office building whereby innovative systems are added for scientific purposes over the years. Initially, a PV system was introduced with 16.9 kWp and a BESS with a capacity of 57.4 kWh and a maximum (dis)charging power of 18 kW. Since 2020, two AC level II charging piles are operational with each two charging points. The EV chargers are connected to three-phase with an amperage of thirty-two and a voltage of 230 V. This results in a maximum charging power of 22 kW (3 * 32 * 230 = 22,080 W = 22 kW). The maximum charging power in the case study equals 44 kW. An overview of the present system with its associated data flows is presented in Overview case studies.

Building B

Building B is an office building in Utrecht, the Netherlands. It was built in the seventies and has an approximate floor area of 85,000 square meters. In the last fifteen years, the office building is completely electrified by introducing ATES, HPs, and PV systems. Furthermore, a green parking lot is realized with a local DC-grid. Currently, there are eighty-five AC level II charging piles with each two charging points. The charging piles have comparable characteristics to the charging piles from Building A. The 5,000 square meters of PV with 940 kWp and the BESS with a capacity of 173 kWh and maximum (dis)charging power of 180 kW are connected to the local DC-grid. The benefit of the local DC-grid is the increased efficiency by avoiding DC/AC transformation losses. The number of AC charging points will be extended from 170 to a total of approximately 450 in the future. From the 450 charging points, sixty-six DC level III charging points will be connected to the local DC grid. An overview of the present system with its associated information flows is presented in Overview case studies.

3.2. Data acquisition

The relatively new OCPI protocol enables bi-directional communication between the back-end of the CPO and the SCSP as illustrated in Figure 3.2. The SCSP, often a third party that is also in control of the EMS, can obtain data from the EMS with HTML or RPC calls. Moreover, the OCPI protocol allows the SCSP to send control inputs and predefined charging profiles toward the EVSE at individual or at busbar level. In this study, the OCPI is only used as a data-collecting mechanism. Aggregated EV power consumption data are used in both case studies for the constitution of the EV charging load forecasting models.

The aggregated EV power consumption of Building A is measured with an interval of eight minutes, whereas Building B has a higher resolution of one minute. The EV charging data from Building A are collected at individual charging pile levels, whereas the EV charging data from Building B are collected at the busbar level. To enable a fair comparison between the case studies, EV charging data from both buildings are considered from 01-03-2022 until 13-11-2022. Data before 01-03-2022 is not considered due to the influence of Covid pandemic measures The statements about the data acquisition are visually supported in Calendar plots. In addition to the power consumption data, transaction data is used, covering the same periods, to extract the number of connections. transaction data is a summary of charging events, containing information about the plug-in time, plug-out time, and charged energy. However, only the plug-in time and plug-out time are utilized to determine the number of connections for each fifteen minutes time interval. EV arrivals and departures are rounded towards the first quarter of the hour.

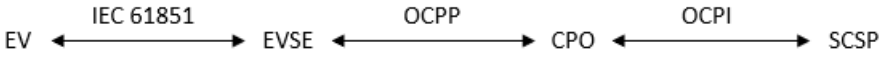

Figure 3.2: EV charger communication and data flow [40].

Building B is a significant office building, which makes it more complicated to determine the building load instantly. As shown in Overview case studies, no energy or power meter captures the total building load. The building load is divided over eight transformers. Therefore, the building load at each transformer can be derived by subtracting PV generation, EV charging, and BESS (dis)charging from the measurements at the transformer level. The total building load equals the summation of the building loads at the transformer level. Initially, power consumption data was used, however, negative building loads were sporadically noticeable. Buildings cannot have a negative power consumption, thus, the power consumption data is subjected to discrepancies. These discrepancies in the data are probably caused by the application of different measurement devices in combination with the variation in data resolutions among the measurement devices. Therefore, energy consumption data with an hourly time interval is used to avoid data discrepancies. The hourly resolution is not desirable since it captures fewer fluctuations, however, it is the best available alternative. Energy consumption data from the last five years are analyzed to find a suitable period for the development of the building load forecasting model. Significantly lower energy consumption is noticeable during weekends in the years 2018 and 2019 as illustrated in Calendar plots. Moreover, lower energy consumption is visible in 2020 and the first half of 2021, which can be explained by the Covid pandemic and malfunctioning of energy measurement devices. Therefore, energy consumption data used from the period 01-10-2022 until 31-12-2022 is suitable.

Additionally, weather data is obtained from KNMI and Solargis with time intervals of one hour and fifteen minutes respectively. Both data sets contain the following features: GHI and

TEMP [41], [42]. The unit for GHI differs between the KNMI and Solargis data. GHI values from the KNMI data are transformed to make them compatible with Solargis data. In general, Solargis data is preferred due to its higher time resolution and more representative unit for GHI. However, the use of KNMI data is inevitable since the historical Solargis data at Building B starts on 28-10-2021. Therefore, KNMI weather data is utilized to overcome the lack of Solargis data in October 2021.

3.3. Averaged-driven forecasting models

The averaged forecasting models are very simplistic in this study. This method can be seen as a quick-and-dirty approach to forecast future EV charging loads. The models solely use historical EV charging data for their development. The historical EV charging data set is initially checked and processed to handle discrepancies in the data sets. Missing values are located and replaced by the linear interpolation method. It should be noted that the replacement of missing values is limited to sixteen consecutive values. If there are more than sixteen consecutive missing values, interpolation is not performed due to high uncertainty. The threshold of sixteen is chosen arbitrarily. As a result, missing values could remain in the initial data set. The initial data set is resampled to fifteen minutes time intervals to ensure a complete index, similar time intervals, and similar measurement times. The applied resampling method cannot operate with missing values. Therefore, the remaining missing values from the initial data set are dropped by default. The resampled data set is checked again for missing values because the index from the initial data set could be incomplete. Missing values are replaced according to the same interpolation method, whereas the remaining missing values are ultimately dropped. The pre-processed data set is split into a train and test set with a size of eighty and twenty percent respectively. The day of the week and time of the day features are added as columns to the train data set and are based on the datetime index. Weekdays are represented with values from zero until six, where zero represents Mondays and six represents Sundays. Additionally, the number seven is incorporated to represent holidays. The identification of holidays is based on a Python package that contains dates from Dutch holidays. The train data is pivoted to obtain the required data structure that consists of ninety-six rows and eight columns. The rows represent the time of day with an interval of fifteen minutes and the columns represent the weekdays and holidays. Each value in the data frame represents the average value at a specific time of the day on a specific weekday or holiday. Therefore, each column in the data frame represents the average load profile for a specific weekday or holiday. The development of the forecasting model is completed with the constitution of load profiles per day type. Forecasts are generated by adding load profiles in the correct order. To do so, an additional datetime series is created with a daily resolution to allow iterations for day-type identification. Averaged-driven forecasts are generated by appending load profiles to an empty series in chronological order based on the corresponding day types.

3.4. Machine learning driven forecasting models

In this subchapter, the process and used methods for the constitution of ML-driven forecasting models are explained in detail. The process is illustrated in Figure 3.3.

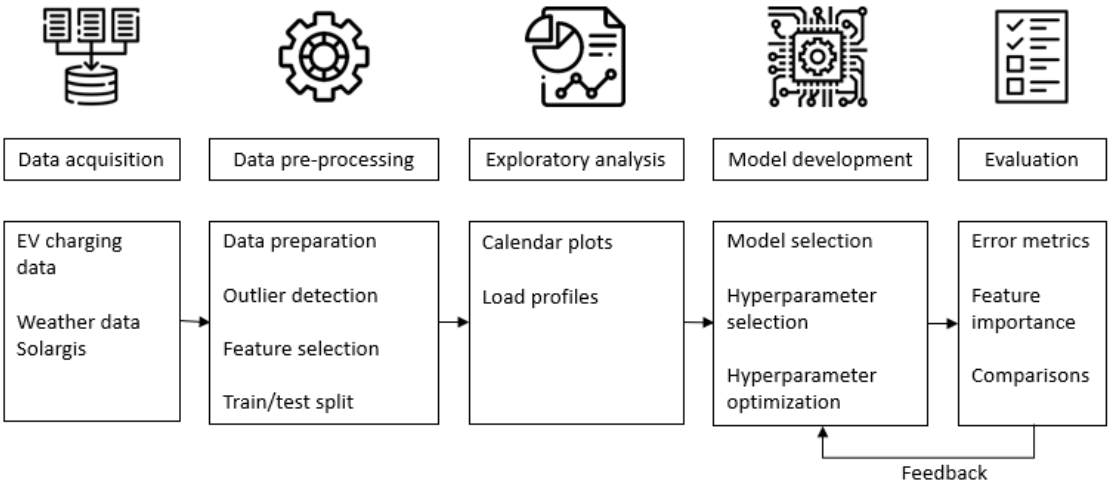

Figure 3.3: Stepwise process to constitute ML-driven forecasting models.

Data-preprocessing

Initially, data sets are checked for missing values and replaced with linear interpolation if present. A constraint is added to the interpolation procedure. When the number of consecutive missing values exceeds sixteen, interpolation is not performed to avoid unrealistic interpolation. Then, the data sets are resampled to fifteen minutes time intervals to equalize their time intervals and measurement times. Herein, missing values are dropped by default since the resampling method cannot handle missing values in data. After resampling, missing values are re-evaluated and processed according to the same method as described above. The cleaned data is enriched with features to enhance the forecasting performance. The feature selection is based on findings from earlier research [19] and contains four categories, namely temporal features, (holiday and weekend) dummies, weather features, and lag features. This study does not use weekend dummies, because temporal features, such as the week of the day, can distinguish work days from weekend days. Relevant lag features are initially unknown, thus, the ACF is used to find the most significant lag variables. The ACF compares the correlation between points 'i' and 'i-k', where 'k' indicates the number of lags. A value between zero and one indicates a positive correlation, whereas a value between zero and minus one indicates a negative correlation. The closer a correlation value approaches (minus) one, the stronger the correlation. Features with strong correlations are valuable for forecasting performance. The ACF results for the various data sets can be found in Evaluation of ACF and multicollinearity Furthermore, it is desirable to evaluate multicollinearity among features to eliminate similar performing features. Multicollinearity means that variables are not independent and influence each other which can affect the robustness of the forecasting model. The effect of multicollinearity can be described as follows: *''Multicollinearity practically inflates unnecessarily the standard errors of the coefficients. Whereas, increased standard errors in turn mean that coefficients for some independent variables may be found not to be significantly far from zero. In other words, by overinflating the standard errors, multicollinearity makes some variables statistically insignificant when they should be significant. Without multicollinearity, those coefficients might be significant''* [43]. To evaluate multicollinearity among variables, the VIF is introduced. The VIF measures the ratio between the variance for a given regression coefficient with only that variable in the model versus the variance for a given regression coefficient with all variables in the model. The VIF can be calculated according to Equation 1. A VIF value of one indicates no multicollinearity, a VIF between one and five indicates

moderate multicollinearity and a VIF above five indicates high multicollinearity that may be problematic. By removing the feature with the highest VIF, the VIF of other features can drop significantly [43], [44]. Therefore, the VIF analysis is an iterative process. Results of the VIF analysis can also be found in Evaluation of ACF and multicollinearity.

$$VIF_i = \frac{1}{1 - R_i^2} \qquad \text{Equation 1}$$

Where:

VIF_i = variance inflation factor of a specific feature

R^2_i = R^2 value of a specific feature

The final step of data-preprocessing is to split data sets into train and test data with a size of eighty and twenty percent respectively. The purpose of the train/test split is to use train data for model development, whereas the 'unseen' test data is used to validate the forecasting performance.

Model development

The forecasting algorithms XGB, RF, and MLP-ANN are evaluated in [19]. These three algorithms are categorized as black-box models. The drawback of black-box models is the less transparent interpretability. Feature importance is a method to explain the functioning of these models by assessing the contribution of individual features to the forecasting performance. Tree-based models, such as XGB and RF, have built-in feature importance modules, whereas MLP-ANN cannot intrinsically assess feature importance. Other methods should be used to evaluate the feature importance of deep learning models such as MLP-ANN [45]. Therefore, this study only focuses on XGB and RF.

XGB is a relatively new ML algorithm. XGB is an ensemble learning method by constructing a multitude of DTs. The DTs are trained sequentially, where DTs consider the forecasting error of the previously trained DT. This method is known as boosting and improves the forecast at each DT by reducing bias [46]. The advantages of this algorithm are its fast computations, the fact that feature scaling is not required, and the inclusion of regularization parameters to avoid overfitting. On the other hand, XGB has a lot of hyperparameters, as shown in Table 3.1, which results in an intensive and more complex hyperparameter optimization [47].

The RF algorithm is also an ensemble learning method by constructing a multitude of DTs. Each DT generates a forecast. In regression, the output of RF is determined by averaging the forecasted values from the DTs, also known as bagging [46]. The advantage of this algorithm is its generally great performance by combining weak individual DTs, its robustness for overfitting by reducing variance with bagging, its robustness to outliers, and the fact that feature scaling is not required. The main drawback is the relatively long computational time in comparison to DTs, where only one tree is trained [48], [49]. However, the computational time of RF is still fast in comparison to other ML algorithms. The tunable hyperparameters are shown in Table 3.2.

Table 3.1: XGB hyperparameter selection and description [47].

Feature	Default	Limit	Description
learning_rate (eta)	0.3	(0,1]	Shrinks feature weights to make the boosting process more conservative. Lower values result in a more conservative model.
n_estimators	100	(0,∞]	The number of gradient-boosted trees. Equivalent to the number of boosting rounds.
max_depth	6	(0,∞]	Maximum depth of a tree. Increasing this value will make the model more complex and more likely to overfit.
min_child_weight	1	[0,∞]	The minimum sum of instance weight needed in a child. The larger min_child_weight is, the more conservative the algorithm will be.
gamma	0	[0,∞]	Minimum loss reduction is required to make a further partition on a leaf node of the tree. The larger gamma is, the more conservative the algorithm will be.
subsample	1	(0,1]	The ratio of the training instances. It means that XGB would randomly sample the specified ratio of the training data before growing trees, which will prevent overfitting. Subsampling will occur once in every boosting iteration.
colsample_bytree	1	(0,1]	The subsample ratio of columns when constructing each tree. Subsampling occurs once for every tree constructed.
lambda	1	[0,∞]	L2 regularization term on weights. Increasing this value will make the model more conservative.
alpha	0	[0,∞]	L1 regularization term on weights. Increasing this value will make the model more conservative.

For each forecasting algorithm, hyperparameters are selected and optimized to enhance the forecasting performance. XGB and RF are widely applied forecasting models,

however, the use of default ML models or ML models from earlier research might not be representative in this study due to case-specific data sets. Thus, hyperparameters of the models should be optimized for our data sets to ensure appropriate and accurate forecasting models. An overview of the most important hyperparameters with their default values, limits, and descriptions is provided for each ML model in the tables below.

Table 3.2: RF hyperparameter selection and description [48].

Feature	Default value	Limit	Description
n_estimators	100	(0,∞]	The number of trees in the forest.
max_depth	None	(0,∞]	Maximum depth of a tree. Increasing this value will make the model more complex and more likely to overfit.
min_samples_split	2	[1,∞]	The minimum number of samples required to split an internal node.
min_samples_leaf	1	[1,∞]	The minimum number of samples required to be at a leaf node. This may have the effect of smoothing the model, especially in regression.
max_features	'auto'	'auto', 'sqrt'	The number of features to consider when looking for the best split.
Bootstrap	False	False, True	Whether bootstrap samples are used when building trees. If False, the whole dataset is used to build each tree.

After the hyperparameter selection, a range of values should be assigned to each hyperparameter. These initial ranges will be determined based on similar studies and will be adjusted iteratively with hyperparameter optimization. There are two basic methods for hyperparameter optimization: GridSearchCV and RandomizedSearchCV. The GridSearchCV method evaluates all possible combinations for the hyperparameters, resulting in an exponentially increasing number of combinations when more hyperparameters or values are added. RandomizedSearchCV can find an appropriate combination of hyperparameter values for the hyperparameters based on random selection and a predefined number of iterations. RandomizedSearchCV is preferred instead of GridsSearchCV, because of the significant reduction in computational time without compromising on performance [50]. Furthermore, cross-validation is incorporated in RandomizedSearchCV and is defined as five splits with five repeats as illustrated in Figure 3.4. This means that train data is split differently among the five splits with sizes of respectively eighty and twenty percent. Cross-validation aims to avoid overfitting by continuously using a different fold as test data. Therefore, the RandomizedSearchCV can find a suitable set of hyperparameter values based on twenty-five random combinations (five splits times five repeats).

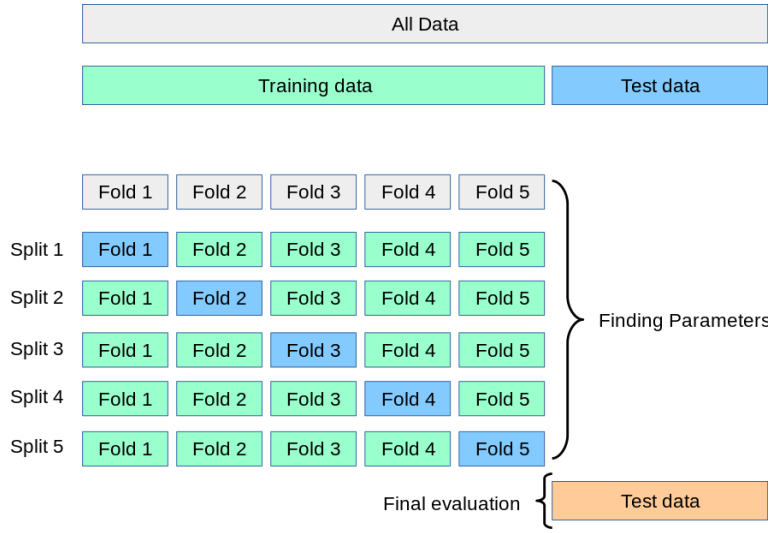

Figure 3.4: Cross-validation with RandomizedSearchCV [51].

Previous research [19] focused on single-step forecasting models. In real-time, single-step forecasting can only forecast the next instance. Therefore, this method is limited and not suitable for the generation of MPC input data since MPC considers a certain forecasting horizon. Multi-step forecasting is explored to overcome this issue. Two multi-step forecasting techniques are investigated, namely recursive multi-step forecasting and direct multi-step forecasting.

The principle of recursive multi-step forecasting is the use of lags to forecast future values from a specified horizon, with length 'H', by iterating over the data as shown in Figure 3.5. The number of lags is predefined and remains constant over time, also known as a receding horizon. The first step is forecasted with the use of historical data only. After the first step, the previously forecasted value is fed back to forecast the next value. Over time, the number of forecasted values increases in the set of lags in comparison to historical. The use of forecasted values seems to be a reasonable representation of unknown future values. However, the use of these approximated values results in an accumulation of errors [52].

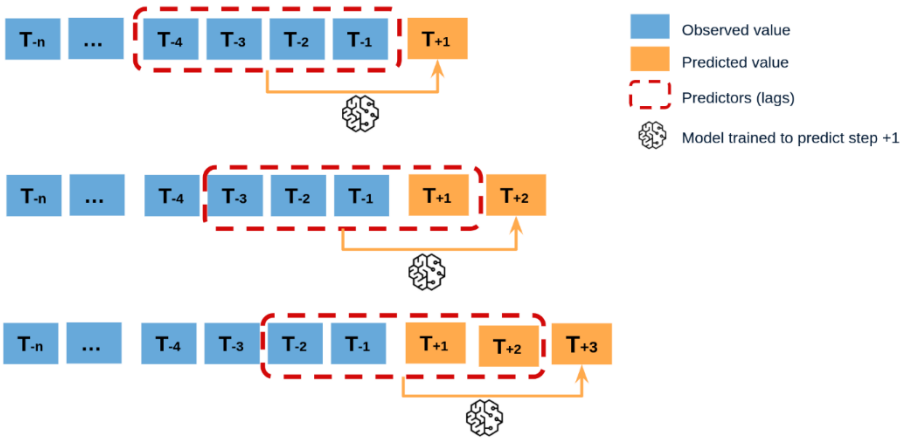

Figure 3.5: Recursive multi-step forecasting [53].

Direct multi-step forecasting also considers a predefined and constant forecasting horizon with a length of 'H'. In contrast to recursive forecasting, direct multi-step forecasting only uses historical data to forecast future values as illustrated in Figure 3.6. In each iteration, an 'H' number of separate forecasting models are trained, aiming to forecast one step from the forecasting horizon. Herein, the forecasting is receding and remains constant. The benefit

of this method is the avoidance of accumulation errors by only using historical data. A disadvantage of direct multi-step forecasting is related to the increasing complexity and its associated longer computational time [54].

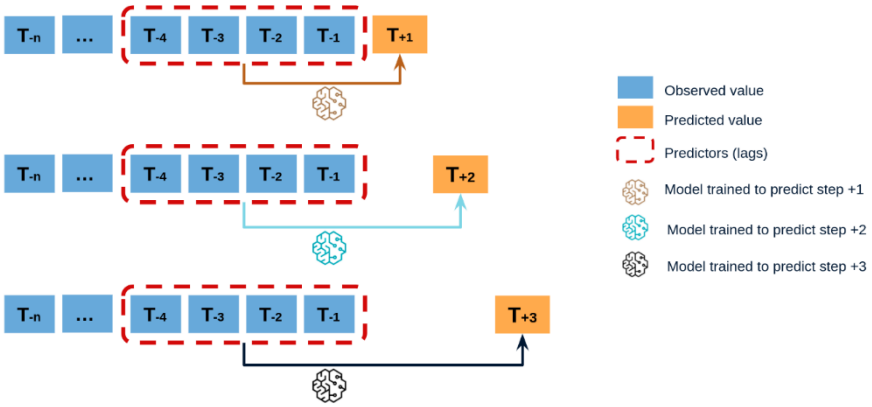

Figure 3.6: Direct multi-step forecasting [53].

The Skforecast Python library is utilized in this study to ease the use of scikit-learn regressors as multi-step forecasters [55]. The scikit RandomForestRegressor is used as regressor in this study since it outperformed XGB with single-step forecasting in previous research [19]. Initially, the RandomizedSearchCV method, as described earlier, is utilized for the hyperparameter optimization of the RandomForestRegressor from the multi-step forecasting models. Three levels of optimization are assessed for the recursive and direct multi-step forecasting methods as presented in Table 3.3. In addition, the lags ninety-six and 672 are assessed among the proposed forecasting models. In total, twelve model configurations are evaluated.

Table 3.3: Overview of forecasting methods and their level of optimization.

Model configuration	Description
REC1	Default recursive multi-step forecasting model
REC2	Recursive multi-step forecasting model with optimized hyperparameters
REC3	Recursive multi-step forecasting model with optimized hyperparameters and trained with exogenous variables.
DIR1	Default direct multi-step forecasting model
DIR2	Direct multi-step forecasting model with optimized hyperparameters
DIR3	Direct multi-step forecasting model with optimized hyperparameters and trained with exogenous variables.

3.5. Forecasting performance evaluation

An often-forgotten aspect in the evaluation of model performance is the selection of appropriate error metrics. Not all error metrics are suitable for specific situations. For example, EV charging data are intermittent, meaning that the charging power is not continuous. As a result, the data sets contain zero values. The presence of zero values has negative implications for relative error metrics, such as MAPE, because they are not able to quantify the error since zero division is not allowed. For that reason, this study uses MAE as shown in Equation 2. MAE is an absolute error metric that is useful to compare forecasting performances within the same study. Furthermore, the traditionally used RMSE is modified to the NRMSE. Normalization aims to enable comparisons among case studies with different magnitudes. Normalization is achieved by dividing the RMSE by the subtraction of the minimum actual value from the maximum actual value as shown in Equation 3.

$$MAE = \frac{1}{n}\sum |F_i - A_i| \qquad \text{Equation 2}$$

$$NRMSE = \frac{RMSE}{(A_{max} - A_{min})} \qquad \text{Equation 3}$$

Where:

A	=	actual value
A_{min}	=	minimum actual value
A_{max}	=	maximum actual value
F	=	forecasted value

As already mentioned, black-box forecasting models are more difficult to interpret. Interpretability is essential to improve the forecasting models and to justify decisions toward stakeholders. Beforehand, it is unknown how a set of features will affect the forecasting performance. Therefore, feature importance evaluation is necessary to find the best trade-off between model complexity and performance and to gain knowledge about the interpretability of the forecasting model. Several features might be irrelevant or detrimental to the learning process. Python has several packages for the evaluation of the feature importance, such as permutation feature importance and SHAP feature importance [56], [57]. However, this study uses built-in feature importance methods that are available for RF and XGB. It should be noted that this study primarily focuses on performance instead of complexity since the number of features is relatively low.

3.6. Model predictive controller

In this study, an MPC is created with the application of Pyomo. Pyomo is a Python-based open-source collection of software packages for formulating optimization models [58]. The fundamental mathematical equations of the MPC are based on research [59]. The objective is to minimize the interaction between the microgrid and the power grid. Herein, the power consumed by the microgrid or the power delivered to the grid can be defined as P_{grid}, which is the power summation of all transformers from Building B as shown in Equation 4. Building loads, PV generation, and EV charging are considered at each transformer as shown in Equation 5. An optimal situation is accomplished when $P_{grid} = 0$. This would imply that power is not extracted from the grid nor delivered to the grid, thus completely self-sufficient. $P_{grid} < 0$ implies an over-production of solar power and $P_{grid} > 0$ implies a larger power consumption

of the building and EVs than the PV generation. Therefore, a quadratic cost function is created to encounter negative power values at times of over-production and to penalize large peaks more severely. As a result, it becomes more desirable to have a lower power consumption during a longer period than a high power consumption during a short period. The defined cost function is mathematically formulated in Equation 6.

$$P_{grid} = \sum_{i \in S_{tr}} P_{tr_i} \quad \text{Equation 4}$$

$$P_{tr_i} = P_{building_i} + P_{pv_i} + P_{evs_i} \quad \text{Equation 5}$$

$$min\ J = \sum_{n=1}^{N} P_{grid}^2 \quad \text{Equation 6}$$

MPC cannot operate properly without specified constraints. The EV charging power (P_{evs}) is the controllable load and cannot exceed the hard constraints related to the physical limitations of the hardware capacity, defined as $P_{evs,min,hard,}$ and $P_{evs,max,hard1}$. The minimum EV charging power equals zero, because EV discharging is currently not possible, whereas the maximum EV charging power is limited to 1,200 kW as presented in Equation 7 and Equation 8. An additional hard constraint is introduced to set a realistic maximum charging power that corresponds to the number of connected EVs, which is defined as $P_{evs,max,hard2}$. This constraint can vary every time step - fifteen minutes - depending on changes regarding the connected EVs. The maximum charging power is determined by multiplying the forecasted number of connections with the median charging power as shown in Equation 9**.** As a result, the control input, u(t), should be larger than zero and smaller than the normative maximum power constraints (Equation 10). Since user-specific information, such as energy demand, is lacking, it is important to define another method to ensure that a minimum amount of energy is delivered to the aggregated EV fleet. It is assumed that a similar amount of energy should be provided as it would be with uncontrolled charging. Therefore, the actual supplied energy to the EVs should be equal to or larger than the forecasted energy demand considering a daily horizon (Equation 11). Herein, the forecasting model should be solely trained with data prior to the MPC implementation to mimic uncontrolled charging. The power demand forecast is executed once a day at midnight and should be fulfilled between midnight and tau. In this study, tau is considered as five o'clock in the afternoon. To ensure that the energy requirement is reached at time tau, the already supplied energy and the remaining energy demand are monitored in real-time as shown in Equation 12. The sum of supplied energy and remaining energy should be equal throughout the day. Only the proportions between the variables are changing. The remaining energy will decrease throughout the day, while the supplied energy will increase.

$$P_{evs,min,hard} \geq 0 \quad \text{Equation 7}$$

$$P_{evs,max,hard1} \leq 1{,}200 \quad \text{Equation 8}$$

$$P_{evs,max,hard2} \leq \widehat{N}_{evs} \cdot P_{ev,median} \quad \text{Equation 9}$$

$$P_{evs,min} \leq u(t) \leq P_{evs,max,hard2}(t) \leq P_{evs,max,hard1} \quad \text{Equation 10}$$

$$\sum_{T_{00:00}}^{\tau} \hat{P}_{evs} \cdot T_s \leq \sum_{T_{00:00}}^{\tau} P_{evs} \cdot T_s \quad \text{Equation 11}$$

$$\sum_{T_{00:00}}^{\tau} P_{evs} \cdot T_s = \sum_{T_{00:00}}^{T_0} P_{evs} \cdot T_s + \sum_{T_0}^{\tau} \hat{P}_{evs} \cdot T_s \quad \text{Equation 12}$$

Machine learning-driven forecasting models are used to forecast disturbances. The building load, PV generation, and the number of EV connections are considered disturbances. The PV generation forecasts are obtained from Solargis, an external party with advanced satellite technology to forecast cloud motion to enhance solar power forecasts. The forecasting model of the EV charging load is only trained on historical data containing uncontrolled charging events. Therefore, the model can forecast uncontrolled EV charging loads - even when the EV charging is already controlled - because the model is trained with a lag feature related to the number of connections and not the EV power. The forecasting models consistently provide day-ahead forecasts with a time interval of fifteen minutes. The day-ahead forecasts are updated every fifteen minutes according to the receding horizon principle to obtain optimal real-time forecasts.

The functioning of the MPC should be evaluated with appropriate KPIs. Two widely applied KPIs in the field of energy flexibility are selected [60]. PPR focuses on the absolute power peak reduction within a daily horizon. PPR is determined by subtracting the maximum power peak in the controlled scenario from the maximum power peak in the uncontrolled scenario as shown in Equation 13. PPRP also considers power peak reduction within a daily horizon but in relative terms instead of absolute as presented in Equation 14.

$$PPR = P_{peak,uncontrolled} - P_{peak,control} \quad \text{Equation 13}$$

$$PPRP = 1 - \frac{P_{peak,control}}{P_{peak,uncontrolled}} \quad \text{Equation 14}$$

4. Results

4.1. EV charging load forecasting models Building A

In previous research [19], ML-driven forecasting models are created to forecast the aggregated EV charging load at Building A. The obtained forecasting performance was not as well as desired. The poor forecasting performance might be enlarged by Covid because the utilized data in [19] cover the period from 16-11-2020 until 31-01-2022. Therefore, this study re-evaluates the aggregated EV charging load forecasting models with post-Covid data. Additionally, a more simplistic averaged-driven forecasting model is created to compare with ML-driven forecasting models. The comparison aims to find the best trade-off between performance and complexity.

The aggregated EV power consumption is presented in a heatmap to obtain insights about the charging loads throughout the days in the data set. According to the heatmap in Figure 4.1, EV charging regularly starts around eight o'clock in the morning. Despite the renewed data and regularity in start time, a significant diversity in power consumption and end times can be observed. The dominant randomness is the greatest challenge for the forecasting models to deal with.

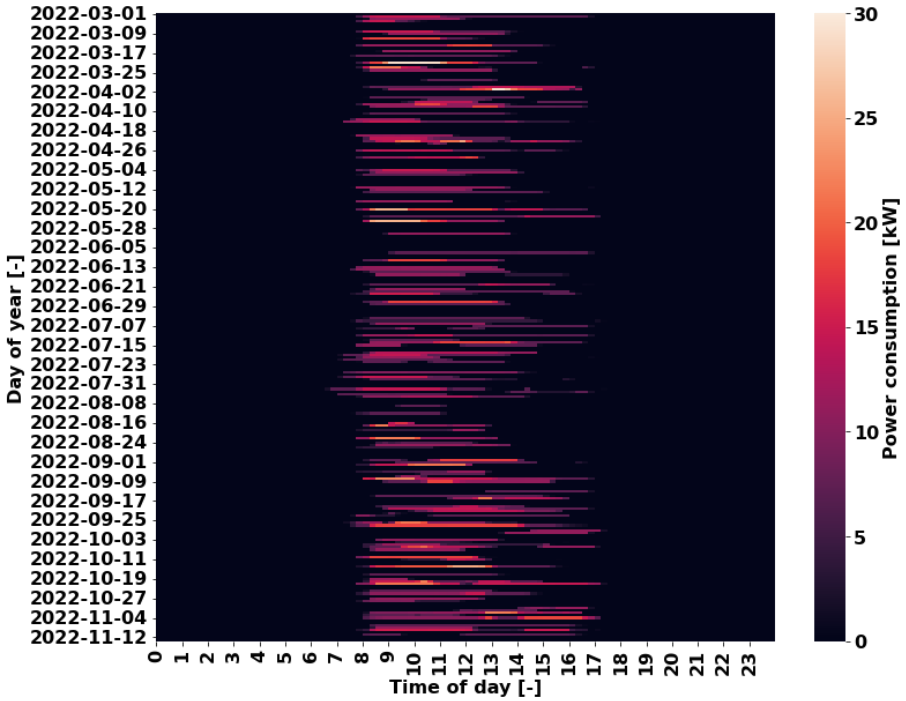

Figure 4.1: Heatmap of aggregated EV power consumption Building A.

The forecasting performances of the various models are visually presented in Figure 4.2. The inability of the forecasting models to forecast the largest power peaks is immediately noticeable. The underestimation of these power peaks is probably caused by their sporadic occurrences. The forecasting models do not expect these high charging loads due to misleading lag features. There is not only the tendency to underestimate the large power peaks but also to forecast a certain charging load, while in reality there is no consumption as illustrated in Figure 4.3. The XGB forecasting model even forecasts negative power consumption, which would indicate EV discharging. However, the bi-directional energy exchange is currently not possible at Building A. Nevertheless, the XGB forecasting model has learned this incorrect behavior by recognizing non-existing patterns between features and

the EV charging power. For that reason, the XGB forecasting model may need more data to train to avoid negative forecast values. It is difficult and inaccurate to visually determine which forecasting model performs best, therefore, numerical results are presented in Table 4.1. In addition, the optimized hyperparameters of the RF and XGB algorithms can be found in Hyperparameter optimization of forecasting models. Surprisingly, the averaged-driven forecasting model outperforms the more advanced ML-driven forecasting models. Unfortunately, the forecasting performance of the averaged model is still very poor as can be seen in Figure 4.2 and Figure 4.3. The application of forecasting models as a basis for smart charging seems to be inappropriate in workplaces where only a few charging piles are present. Hence, the collection of user-specific information might be inevitable.

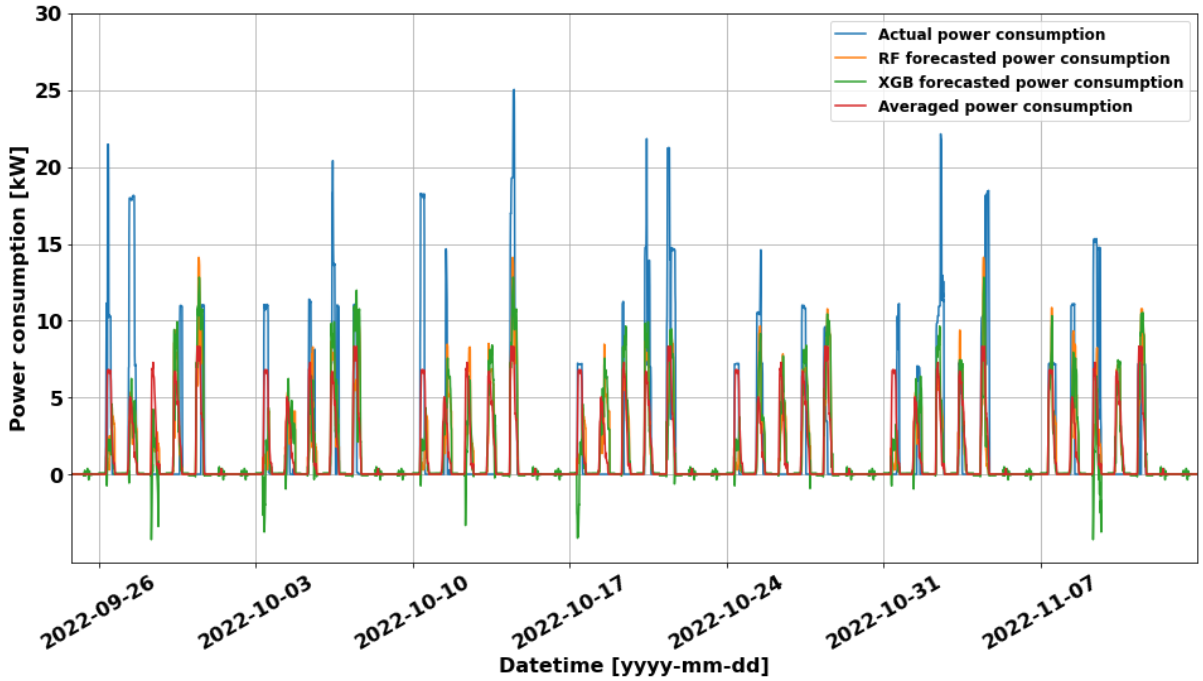

Figure 4.2: Visual evaluation of aggregated EV charging load forecasting models Building A.

Table 4.1: Evaluation of aggregated EV charging load forecasting models Building A.

Error metric	RF default	RF optimized	XGB default	XGB optimized	Averaged
MAE [kW]	1.528	1.426	1.560	1.456	1.246
NRMSE [-]	0.152	0.139	0.152	0.140	0.125

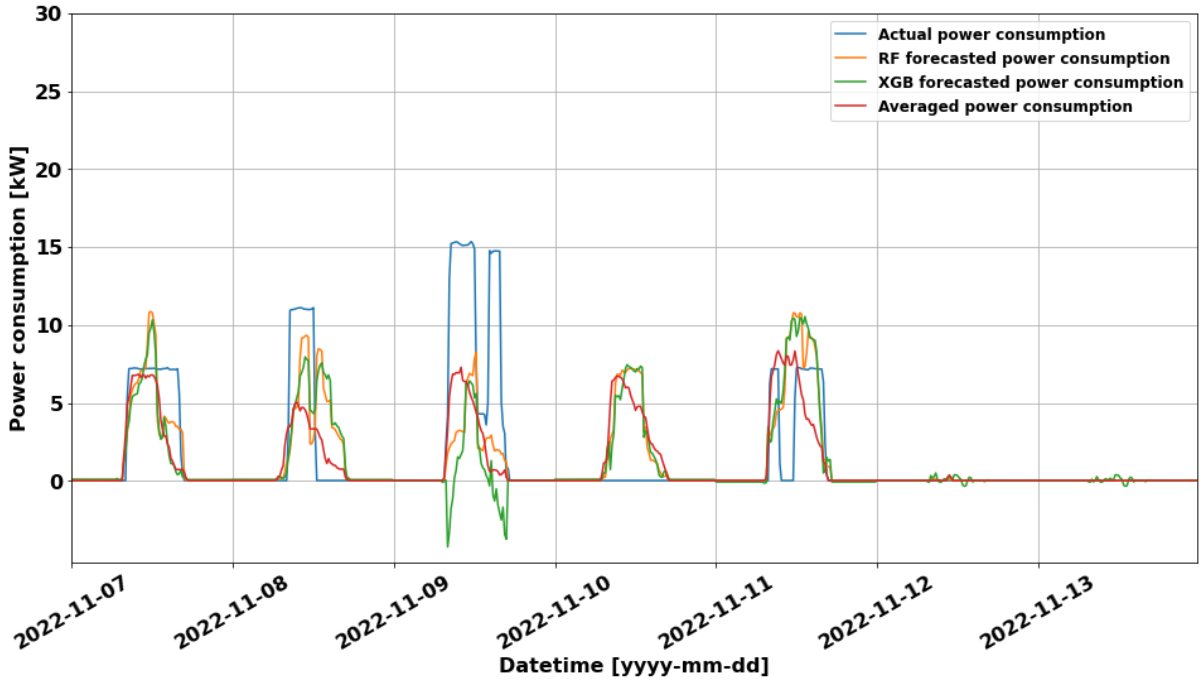

Figure 4.3: Visual evaluation of aggregated EV charging load forecasting models Building A during last week's test data.

The ML-driven forecasting models are optimized with temporal features, a holiday dummy, and a previous week lag. In addition to the ACF and VIF analysis, the feature importance is evaluated to further improve the set of input features as illustrated in Figure 4.4. In both algorithms, the number of connections the previous week is seen as an important feature, which explains the earlier hypothesis about the effect of the lag feature on the sporadic large power peaks. Instead of using the previous week's power data, the forecasting models consider the previous week's number of connections as a lag. This is because a controller can only control the charging load, not the number of connected EVs. By using the number of connections as a lag, the forecasting models can predict the uncontrolled charging load even when a controller is in operation. Since the charging load and the number of connections have a strong correlation, the application of the previous week's number of EVs is valid. Furthermore, the temporal features are indispensable because they enable the forecasting model to learn patterns between datetime and EV power consumption. Without temporal features, the forecasting model would not be able to forecast zero loads on the weekends. Initially, weather features were also utilized as features. However, it appeared that weather features have a limited contribution to the forecasting performance while increasing model complexity and decreasing the forecasting horizon. A day-ahead forecasting horizon is a maximum with weather features, whereas a week-ahead forecasting horizon is applicable without weather features. Moreover, the forecasting models are not exposed anymore to discrepancies in the weather data and become more robust since they do not rely on the availability of weather data. A thorough analysis of the feature selection can be found in Feature selection of EV charging load forecasting models.

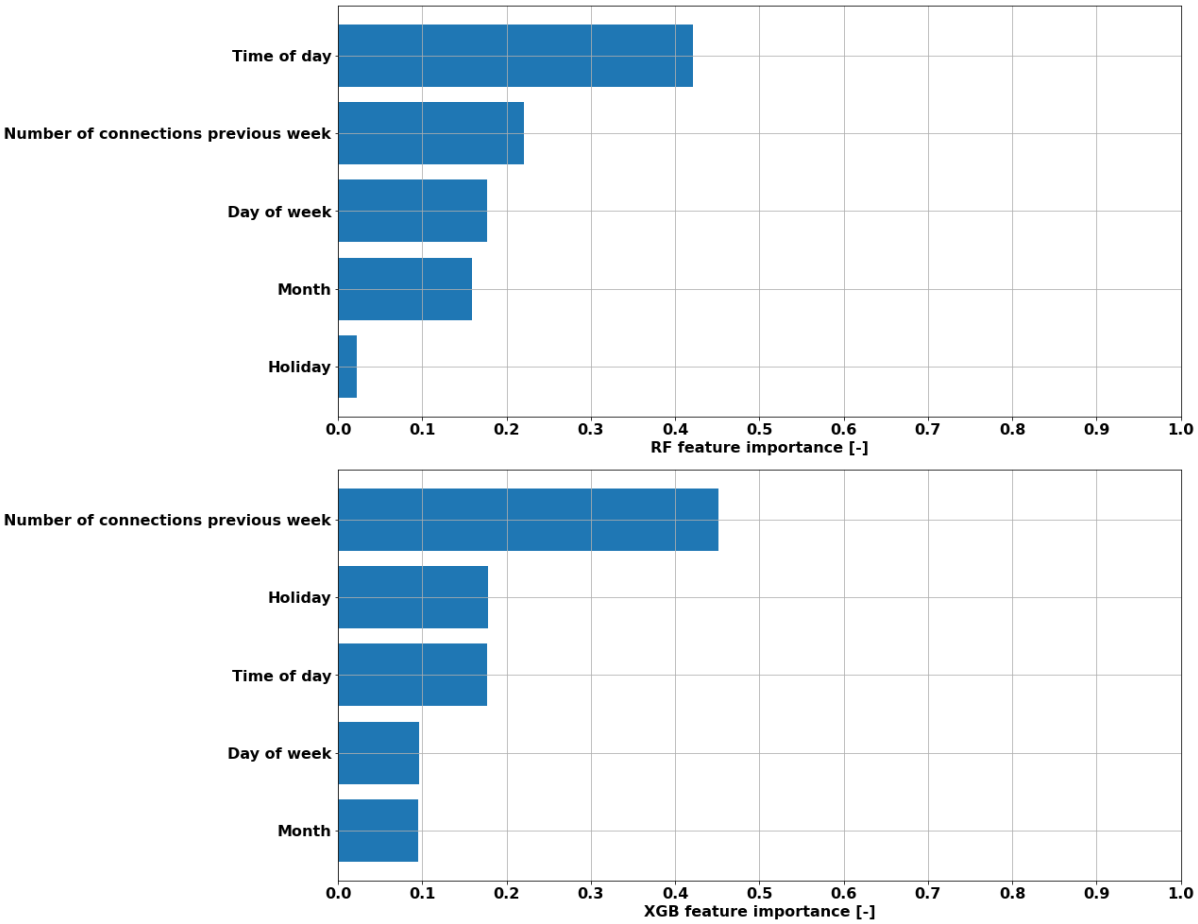

Figure 4.4: Feature importance RF and XGB EV charging load Building A.

4.2. Forecasting models Building B

In contrast to Building A, more patterns are visible in the heatmap of the Aggregated EV charging load from Building B as shown in Figure 4.5. EV charging often starts between seven and eight o'clock in the morning. Then, the charging load rapidly increases with the largest power peaks between nine o'clock in the morning and one o'clock in the afternoon. During the afternoon, the EV power consumption gradually decreases until charging stops between four and six o'clock in the afternoon. A clear daily pattern is observed in the data, which should be beneficial for the accuracy of the forecasting models.

The forecasting models seem to tend to underestimate the EV charging load, especially towards the end of the test data, according to Figure 4.6. To verify if the hypothesis about underestimation is correct, residuals of the ML-driven and averaged-driven forecasting models are thoroughly investigated in Figure 4.7. Herein, a residual value is defined as forecasted value minus actual value. The histogram on the y-axis shows the distribution of the residuals, which are. biased toward negative values for both the ML- and averaged-driven forecasting models. The underestimation of the charging load is caused by an increasing trend in the charging load as illustrated in Figure 4.8. Here, the weekly rolling mean is calculated to smoothen the data. A clear upward trend can be observed, where the average charging load starts at approximately thirty kW and increases to approximately forty kW. An increase of thirty-three percent. The increasing EV charging load is caused by an increasing EV population. The weekly rolling means of the EV charging load and the number of connections are matching almost perfectly as shown in Figure 4.8. Re-training of the forecasting models is the most obvious solution to deal with the increasing trend and is essential to maintain a desirable

forecasting performance. The XGB forecasting model still generates negative forecast values. As already mentioned, this is unrealistic since discharging is currently impossible. Despite the concerns about inaccurate forecast values, the forecasting performance increased significantly in comparison to Building A whereby the RF forecasting model performs best according to the numerical results in Table 4.2.

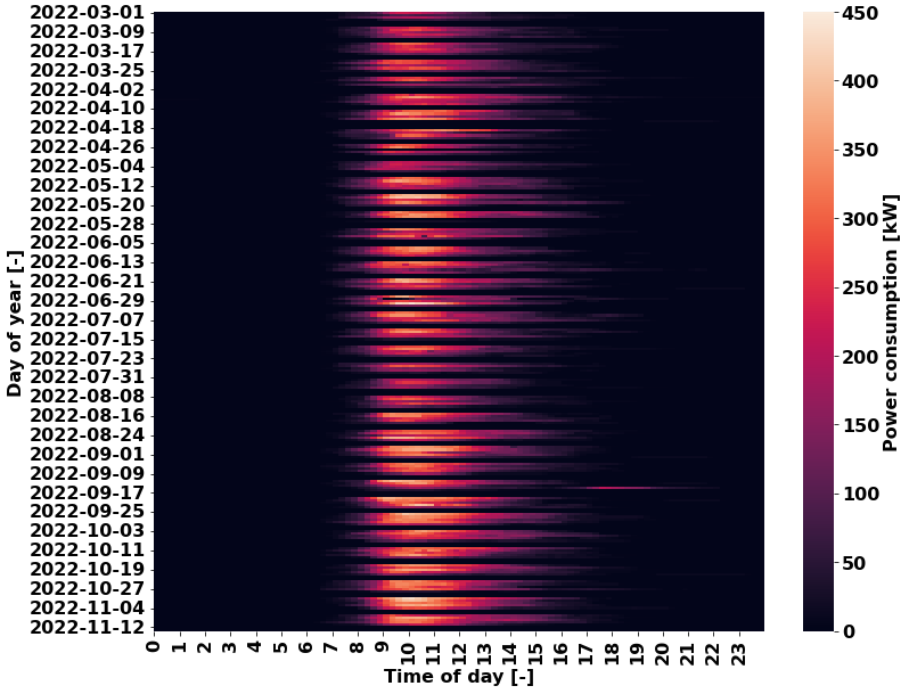

Figure 4.5: Heatmap of aggregated EV power consumption Building B.

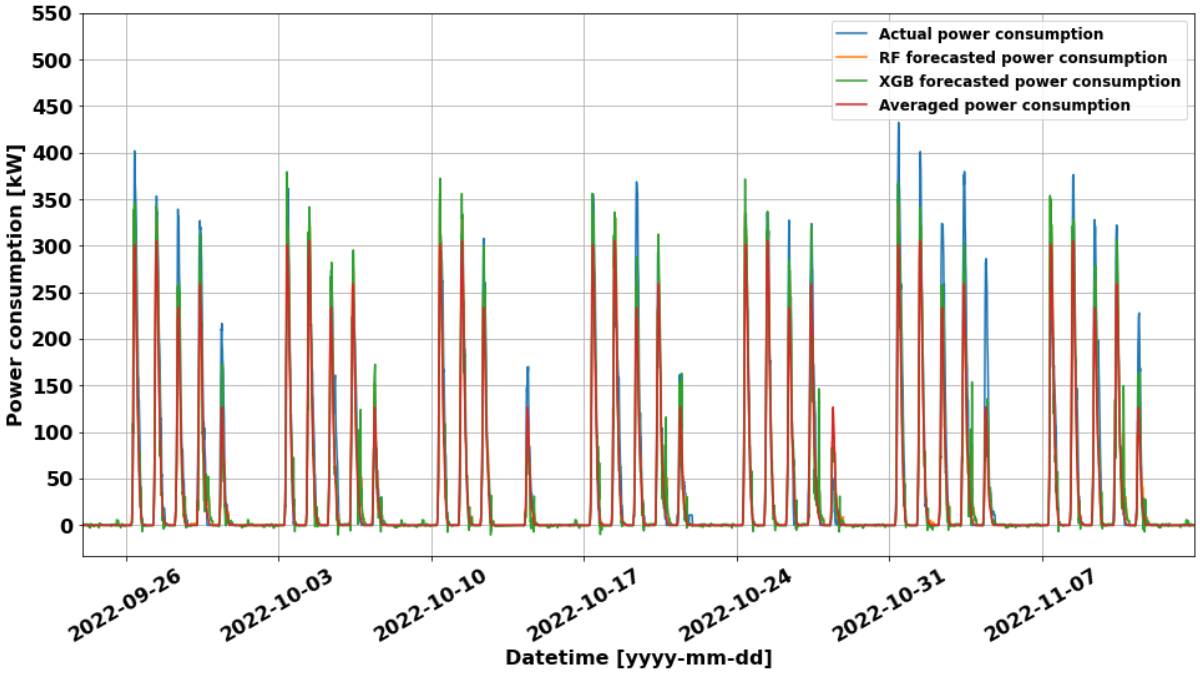

Figure 4.6: Visual evaluation of aggregated EV charging load forecasting models Building B.

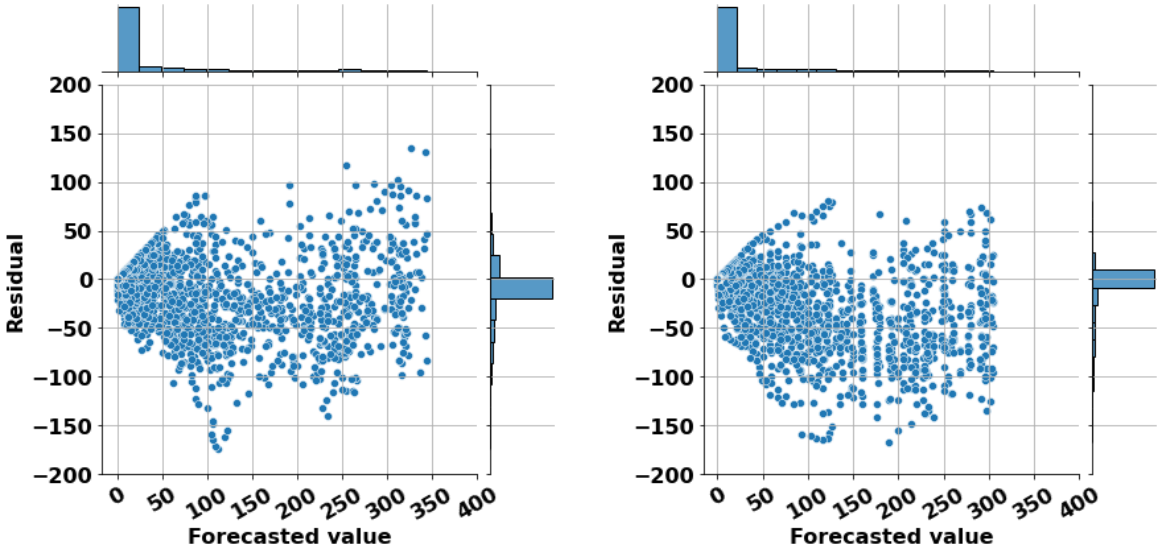

Figure 4.7: Residual plot based on RF and averaged forecasting model. RF on the left and averaged on the right.

Table 4.2: Evaluation of aggregated EV charging load forecasting models Building B.

Error metric	RF default	RF optimized	XGB default	XGB optimized	Averaged
MAE [kW]	11.34	11.17	12.20	11.92	11.84
NRMSE [-]	0.059	0.057	0.062	0.060	0.065

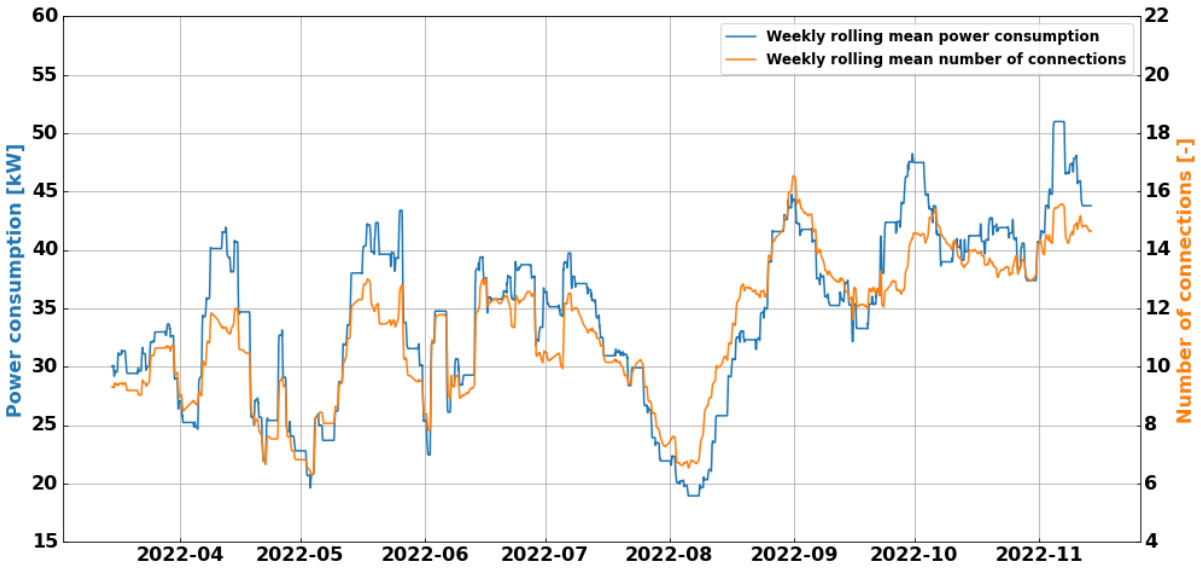

Figure 4.8: Weekly-based rolling means. Left: EV charging load on the left and right: number of connections.

The features 'Time of day' and 'Number of connections previous week' are relevant for both algorithms as illustrated in Figure 4.9. It is noticeable that the forecasting models do not rely as much on a single feature as observed in the feature importance evaluation from Building A. The robustness of a forecasting model increases when relying on multiple features since more underlying patterns are considered in comparison to a model with a high dependency on a single feature. The importance of weather features is also assessed for the EV charging load forecasting model from Building B. Again, the contribution of weather

features to the forecasting performance is limited. Therefore, weather features are not considered as input for the EV charging load forecasting models of Building B. A more detailed analysis of the feature's importance can be found in Feature selection of EV charging load forecasting models.

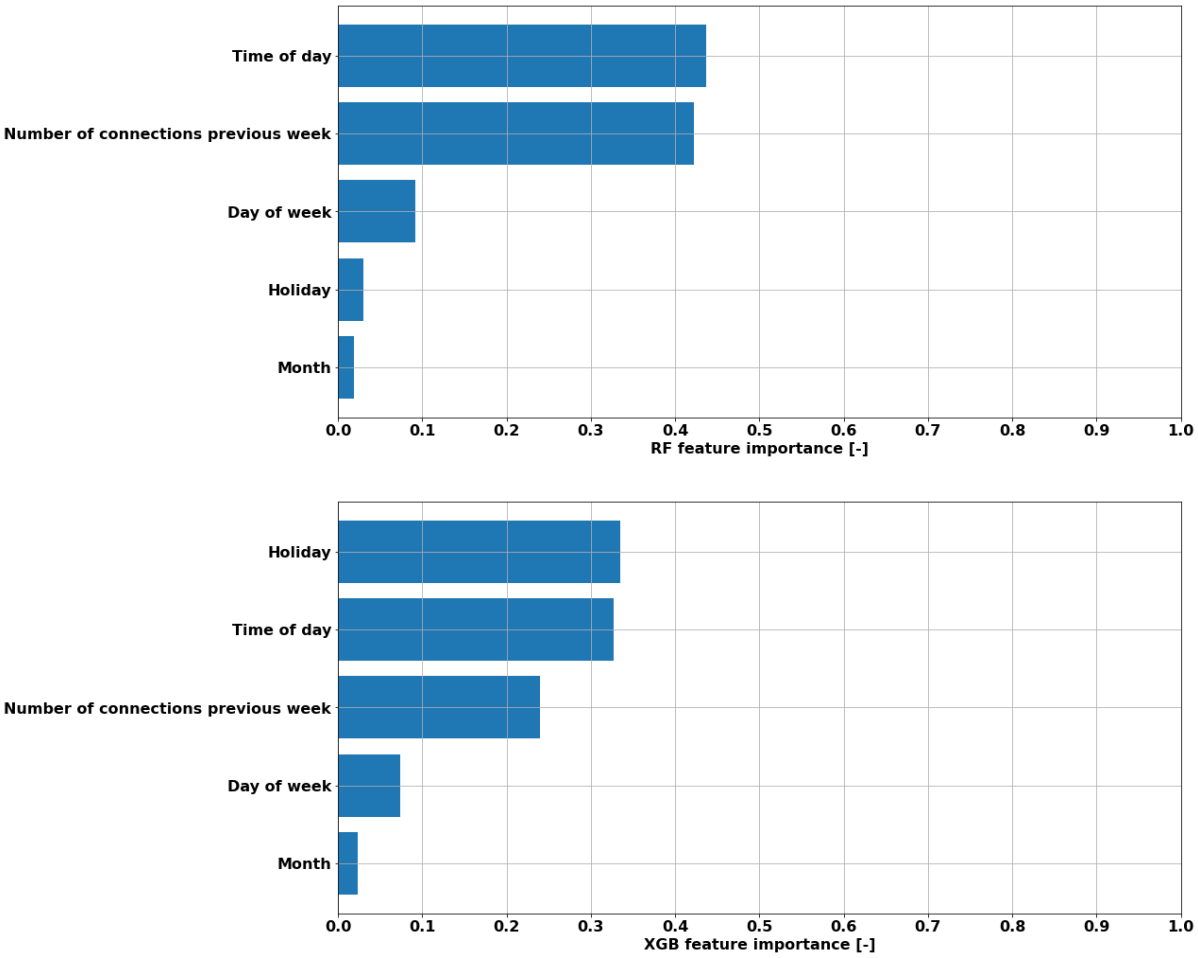

Figure 4.9: Feature importance RF and XGB EV charging load Building B

Significant spatial effects are observed between the aggregated EV charging load forecasts from Building A and Building B. The aggregated EV charging load at Building B can be forecasted more accurately in comparison to Building A as shown in Table 4.3. This confirms the hypothesis that aggregated EV charging loads can be forecasted more accurately with larger EV populations. Less randomness and more repetitive patterns are observed with larger populations, which decreases the complexity of the forecasting problem and thereby improves the forecasting performance. Despite the significantly improved forecasting performance, the forecasting models are generating less accurate forecasts at the end of the test data due to the increasing EV charging load and EV population. Therefore, it is essential to re-train forecasting models periodically to limit the impact of trends.

Table 4.3: Comparison of forecasting performances aggregated EV charging load Building A and B.

Error metric	RF A	RF B	XGB A	XGB B	Averaged A	Averaged B
NRMSE [-]	0.139	0.057	0.140	0.060	0.125	0.065

The previously evaluated ML-driven forecasting models can be characterized as single-step forecasters. As the name already suggests, single-step forecasters are limited since they can only forecast one time step ahead with the available historical data. In this study, the previous week's lag is used as an input feature. As a result, each historical data point enables the forecasting model to forecast the week-ahead value. For example, in real-time at time zero, the model can forecast the next week's value at time 671. At time one, the next data point, the model is capable of forecasting the value for time 672 et cetera. However, the advanced control technique MPC aims to optimize in real-time while considering a specific forecasting horizon. The consideration of a forecasting horizon, which is larger than one step, is not possible with a single-step forecaster. Therefore, multi-step forecasting methods are considered to enable the operation of MPC. Specifically, recursive multi-step and direct multi-step forecasting are evaluated. Several configurations of recursive and direct multi-step forecasting are assessed. In total, twelve model configurations are considered. The abbreviations REC and DIR represent the recursive and the direct multi-step forecasting methods respectively. For both recursive and direct forecasting, three levels of optimization were investigated. Herein, level one represents default models. In level two, the hyperparameters of the RandomForestRegressor are optimized. Level three represents a configuration in which the hyperparameters of the regressor are optimized and where exogenous variables are included for model development. These six model configurations are evaluated with ninety-six and 672 lags. To do so, data from 01-03-2022 until 06-11-2022 is used for training, whereas the week from 07-11-2022 until 13-11-2022 is used to test the forecasting performance. The train/test split method with sizes of eighty and twenty percent respectively is not used since it is desirable to have the most up-to-date forecasting models in the MPC. The best-performing model is DIR3-96 – DIR3 model configuration with ninety-six lags – according to the numerical results in Table 4.4. The error metrics scores in Table 4.4 are based on day-ahead forecasts because they are required for the determination of the daily aggregated EV energy demand. In comparison to the single-step week-ahead EV charging load forecasts, DIR3-96 has a better performance according to the MAE score but scores slightly poorer with NRSME. In other words, DIR3-96 performs better on average but has more difficulty handling variability in the forecasts. The higher NRSME score is probably caused by the use of ninety-six lags, which enlarges the weight of the previous day in the decision-making process of the forecasting model. A DIR3-672 model might be a solution to improve the ability to deal with variability by considering an entire week as a lag. Unfortunately, DIR3-672 is not evaluated due to the computational time for model development. Furthermore, it is observed that recursive and direct multi-step forecasting models with ninety-six lags cannot accurately forecast without the application of exogenous variables. The models without exogenous variables have difficulties with forecasting Mondays and weekend days accurately as illustrated in Visualization of data sensitivity analysis multi-step forecasting models. The temporal features, which are included in the exogenous variables, provide additional information about weekly patterns that allow for more accurate forecasts. The recursive forecasting models without exogenous variables can also be

improved significantly by increasing the number of lags from ninety-six to 672. In doing so, the forecasting performance increases since the EV charging loads are exposed to weekly patterns.

Table 4.4: Forecasting performances of multi-step forecasting models day ahead.

Metric	Lags	REC1	REC2	REC3	DIR1	DIR2	DIR3
MAE	96	27.35	25.35	15.34	25.86	25.76	9.75
	672	15.90	13.39	13.29	N/A	N/A	N/A
NRMSE	96	0.162	0.159	0.090	0.123	0.123	0.062
	672	0.101	0.083	0.076	N/A	N/A	N/A

Unfortunately, the current hyperparameter optimization of the recursive and direct multi-step forecasting models is very time-consuming. Since this study has a time limit, possibilities to decrease the model development time are investigated. Bayesian hyperparameter optimization is considered instead of RandomizedSearchCV to fasten the optimization process. However, this alternative cannot reduce the computational as much as desired. Therefore, a data sensitivity analysis is performed. The data sensitivity analysis aims to find the optimum amount of data whereby the computational time of model development is acceptable without compromising on the forecasting performance. A comparison is made between forecasting models that are trained on the initial eight months and models that are trained with only one month of data. The forecasting performances of the models trained with one month of data are numerically presented in Table 4.5. The forecasting model REC1-672 performs best. Moreover, recursive forecasting with ninety-six lags becomes better with further optimization of hyperparameters and the inclusion of exogenous variables. Surprisingly, the opposite happens with recursive forecasting models using 672 lags. The forecasting performance decreases when optimizing the hyperparameters and including exogenous variables. This phenomenon may be related to the lesser impact of temporal features. Seasonality might become negligible due to the relatively short train data set, whereas the other temporal features might be less significant since the weekly patterns are already captured by the considered lags. The REC1-672 model trained with one month of data has a better MAE score than the same model trained with eight months of data. However, REC1-672 with more train data performs better according to the NRMSE. Therefore, the forecasting model with more data seems to be more stable. Furthermore, direct multi-step forecasting models perform poorer with fewer data because they purely focus on historical data. Hence, training with more data is beneficial for both forecasting typologies but has a larger impact on the direct models than on the recursive models. In both cases, NRMSE scores are higher with a smaller train data set, indicating less stability in their forecasts. Despite the superior performance of DIR3-96 trained on eight months of data (Figure 4.10), the REC1-672 trained on one month of data is considered suitable and chosen for further development of the MPC in this study.

Table 4.5: Forecasting performances of multi-step forecasting models day-ahead with limited train data.

Metric	Lags	REC1	REC2	REC3	DIR1	DIR2	DIR3
MAE	96	28.32	27.59	16.52	29.13	28.98	13.56
	672	12.96	13.96	14.08	N/A	N/A	N/A
NRMSE	96	0.173	0.166	0.086	0.132	0.132	0.075
	672	0.081	0.086	0.088	N/A	N/A	N/A

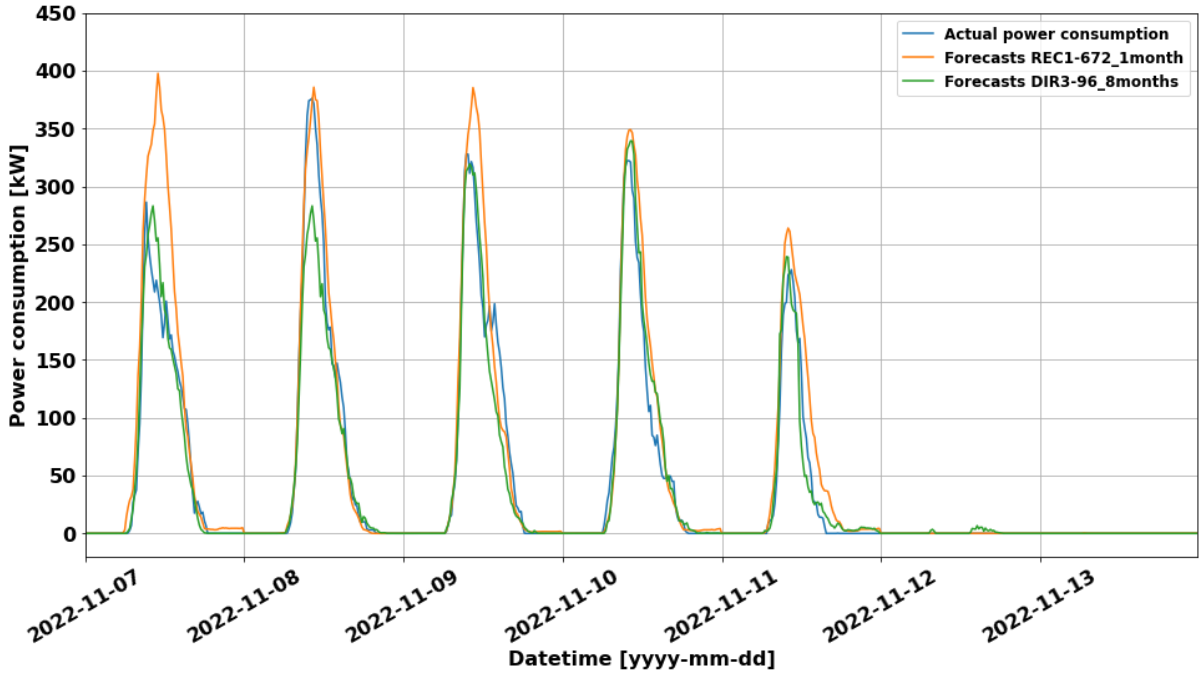

Figure 4.10: Visual evaluation of best-performing multi-step EV charging load forecasting models with different sizes of train data.

So far, this study focused on EV charging load forecasting. However, such a forecasting model cannot be utilized in real-time to determine control inputs of the MPC, because it is not possible to forecast and control a variable simultaneously. Therefore, to run an MPC that controls the EV charging load, variables that are independent of the EV charging load should be used as inputs Hence, the number of EV connections, building load, and PV generation are considered important disturbances on which the EV charging load should be anticipated. Multi-step forecasting models are created to forecast the number of connections and building load. The PV forecasts are provided by the external party Solargis, thus, PV forecasting models are out of scope in this study. The multi-step forecasting models are trained as REC1-672 forecasting models with one month of data.

The building load is trained on a similar set of data, namely from 01-03-2022 until 06-11-2022. In Figure 4.11, the day-ahead forecasting performance is evaluated with test data. The forecasting model can approach the actual building load on work days closely, whereas an underestimation of the building load on weekend days can be observed. The underestimation of the building load during weekend days is not an urgent issue since the MPC does not affect on weekends due to the absence of EV charging. In addition to day-ahead forecasts, MPC

optimizes forecasts in real-time with a receding horizon. Since only the first control input is forwarded to the controller, the first value from each iteration is the most crucial. Hence, it is interesting to evaluate the performance of the first value at each iteration in comparison to the actual values as shown in Figure 4.12. The real-time forecasts fit almost perfectly the actual building load, also during the weekend. The real-time one-step ahead forecasts perform significantly better than the day-ahead forecasts as shown in Table 4.6. Therefore, the building load forecasting model seems to be suitable for MPC applications.

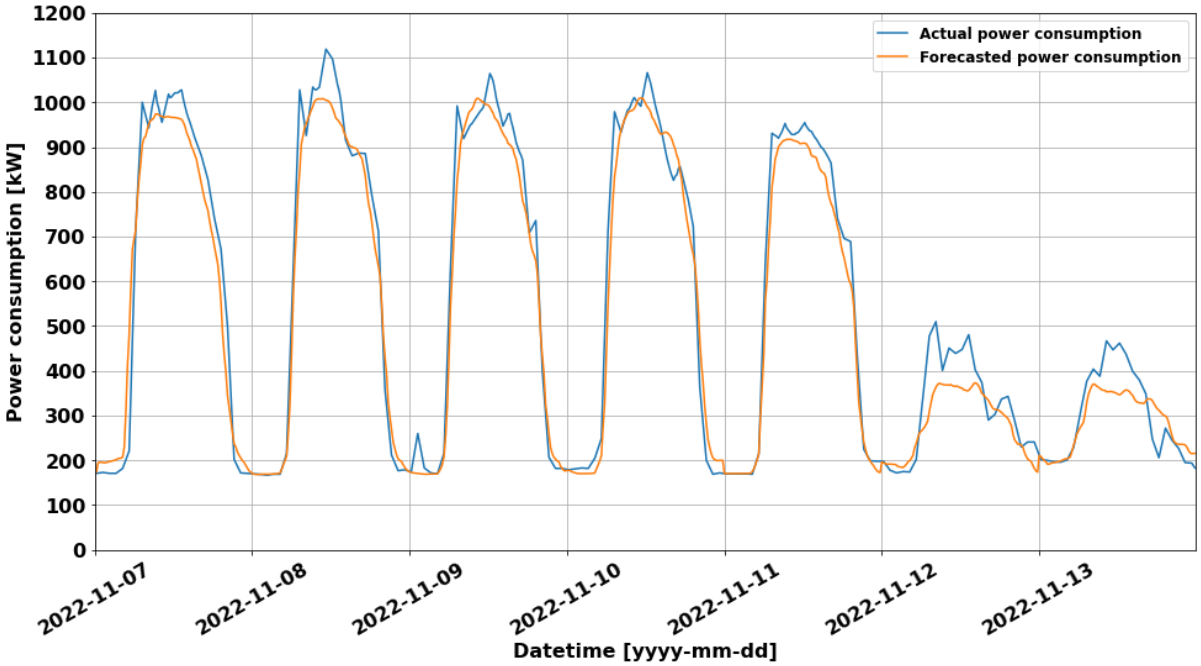

Figure 4.11: Visual evaluation of day-ahead forecasting performance building load.

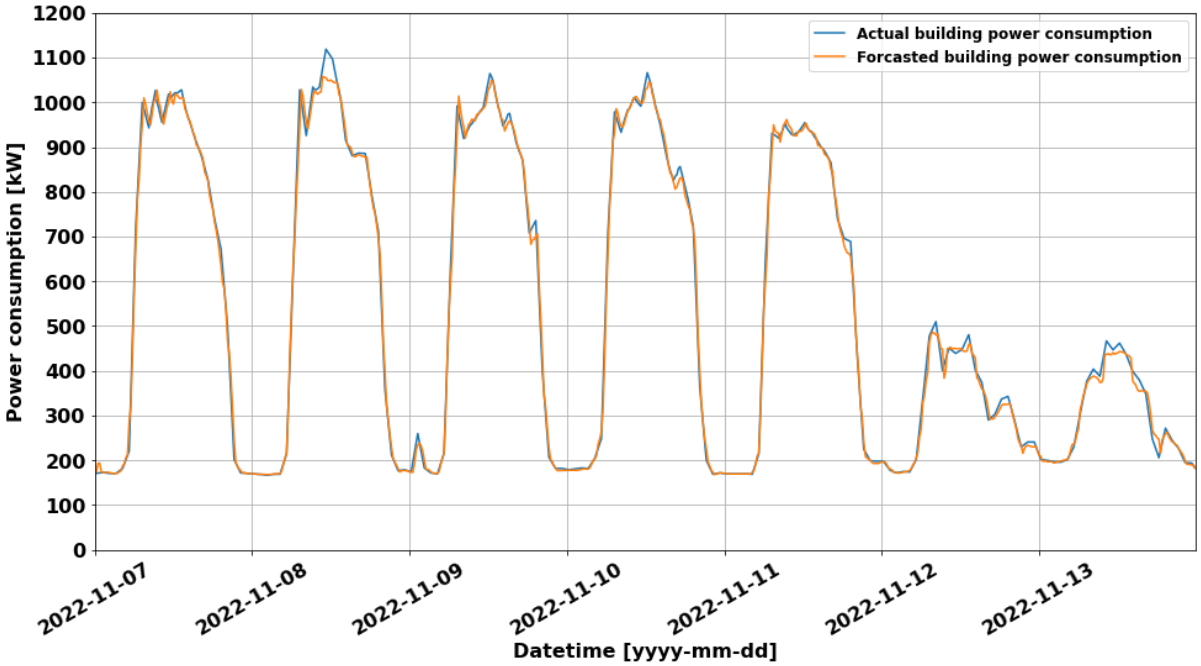

Figure 4.12: Visual evaluation of real-time forecasting performance building load.

Table 4.6: Numerical results of day-ahead and real-time forecasting accuracy building load.

Error metric	Day-ahead	Real-time
MAE	42.61	11.10
NRMSE	0.061	0.018

Moreover, the number of connections forecasting model is created similarly to the building load forecasting model. The day-ahead and one-step-ahead forecasting performances are illustrated in Figure 4.13 and Figure 4.14 respectively. Significant deviations exist in the day-ahead forecasts in comparison to the actual number of connections. These deviations in the day-ahead forecasts are not favorable because they influence the decision-making of the MPC. In a scenario where the forecasting model underestimates the number of EV connections, the maximum possible power that can be delivered to the EVs, according to Equation 9, is expected to be lower than in reality. As a result, the MPC will start EV charging earlier than necessary and thereby not fully use the potential to reduce power peaks. Similar issues arise when the forecasting model overestimates the number of connections. Then the model might want to provide more power than the EVs can charge or delays EV charging too long, which might lead to infeasibility in both cases. Fortunately, the one-step ahead performance significantly improves in comparison to the day-ahead forecasts as presented in Table 4.7.

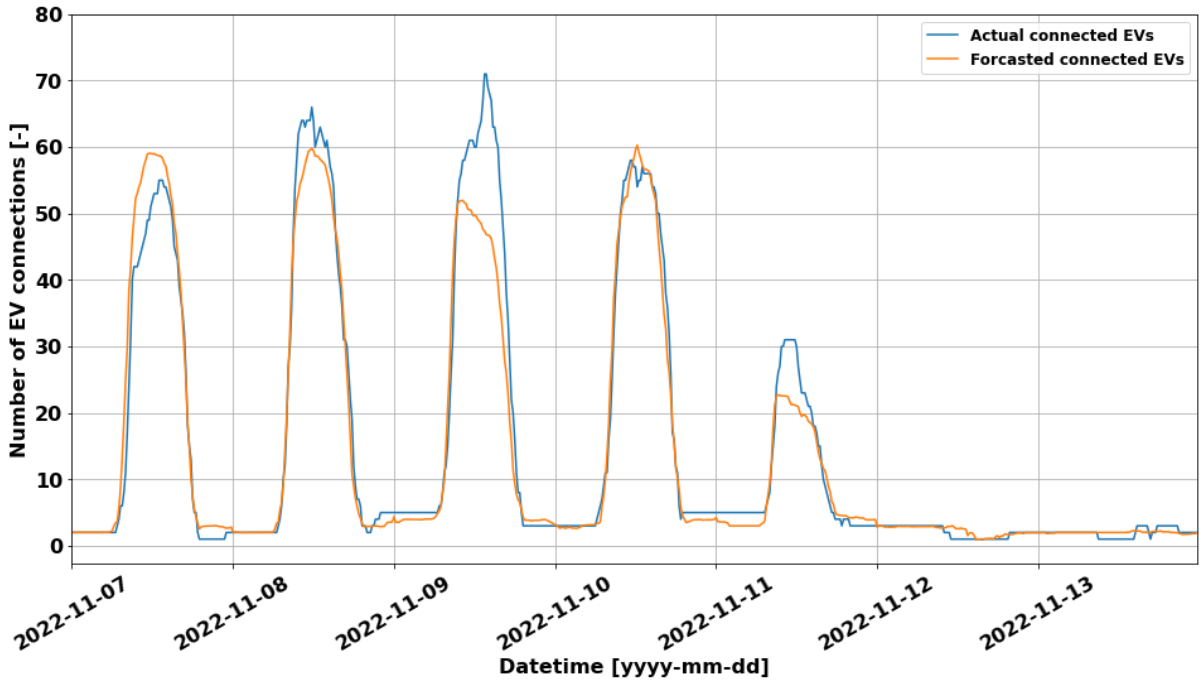

Figure 4.13: Visual evaluation of day-ahead forecasting performance number of EV connections.

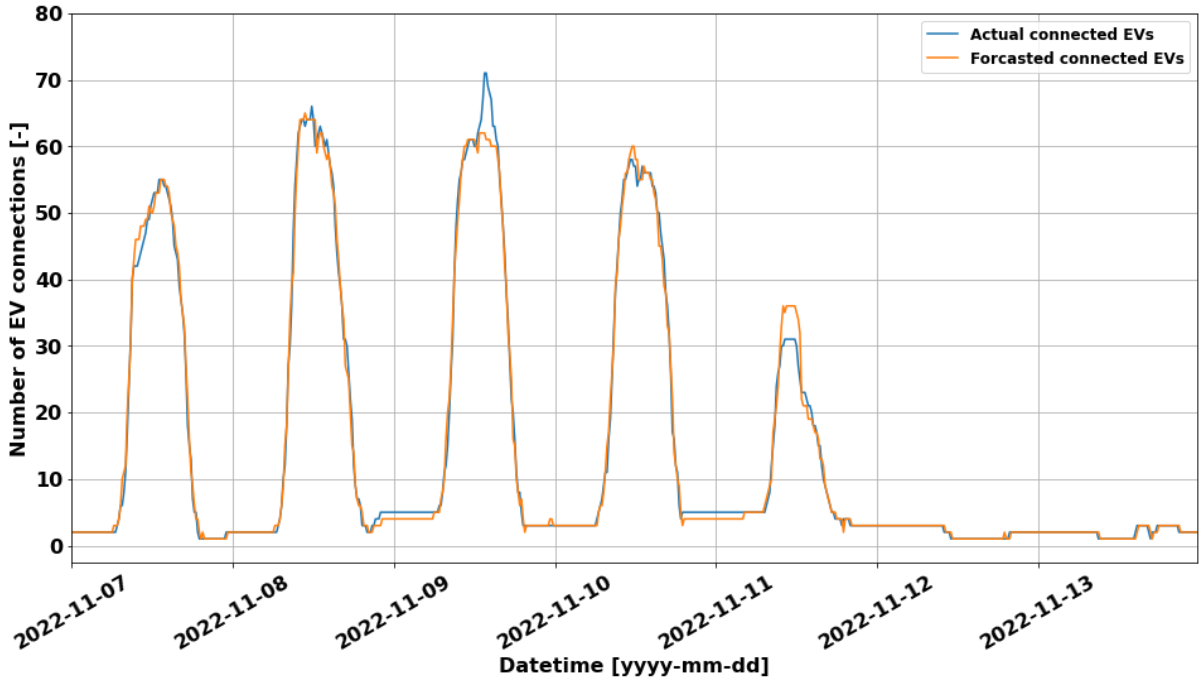

Figure 4.14: Visual evaluation of real-time forecasting performance number of EV connections.

Table 4.7: Numerical results of day-ahead and real-time forecasting accuracy of the number of EV connections.

Error metric	Day-ahead	Real-time
MAE	2.281	0.793
NRMSE	0.063	0.022

4.3. MPC

The developed MPC model aims to optimize the interaction between the microgrid and the power grid. To do so, the EV and building power consumption as well as the PV power generation are considered. The daily EV energy demand, building load, number of EV connections, and PV generation forecasting models are integrated into the MPC model to provide day-ahead forecasts in real-time. The integrated forecasting models are based on the REC1-672 configuration and are trained with one month of historical data. In the first and second plots from Figure 4.15, the effect of EV charging load shifting is visible. The EV charging load is shifted more towards the afternoon, where it matches with the PV generation. However, the effect on the microgrid remains relatively limited due to the significant base load from the building. The power peaks of the microgrid are reduced by up to 11.7 percent according to the numerical results in Table 4.8. The reductions are not very significant, especially in relative terms, due to the enormous uncontrollable base load. Potentially, the power peak reduction becomes more significant when the EV population grows.

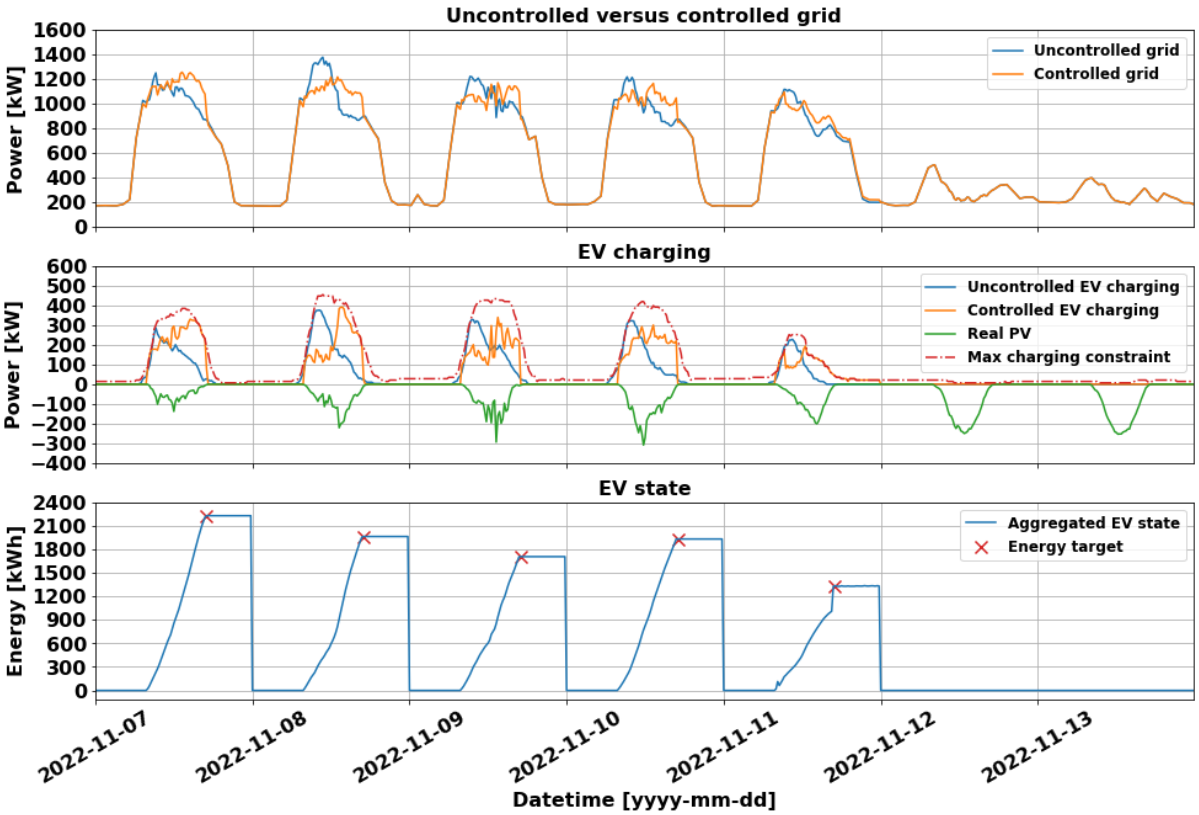

Figure 4.15: MPC results.

Table 4.8: Overview of KPI scores.

Power peaks	Mon	Tue	Wed	Thu	Fri	Avg
Uncontrolled [kW]	1,250	1,378	1,221	1,218	1,117	1,237
Controlled [kW]	1,256	1,217	1,170	1,165	1,096	1,181
PRP [kW]	-6	161	51	53	21	56
PRPP [%]	-0.48	11.7	4.2	4.4	1.9	4.5

Another reason for the limited reduction of the power peaks is the overestimation of the daily aggregated EV energy demand. In Table 4.9, the forecasted daily aggregated EV energy demands are compared with the actual energy demands. The energy demand is overestimated almost every day throughout the test week. On Monday and Friday, overestimations exceed fifty percent. For that reason, it is important to continuously work on the improvement of the forecasting models. A possible improvement might be the introduction of a direct multi-step forecasting model. However, more investigation and time is required to confirm this hypothesis.

Table 4.9: Evaluation of daily aggregated EV energy demands.

Energy consumption	Mon	Tue	Wed	Thu	Fri	Avg
Actual [kWh]	1,451	1,821	1,732	1,503	849	1,471
Forecasted [kWh]	2,229	1,964	1,707	1,932	1,336	1,834
Absolute deviation [kWh]	+ 778	+ 143	- 25	+ 429	+ 487	+ 363
Relative deviation [%]	+ 53.6	+ 7.9	- 1.4	+ 28.5	+ 57.4	+ 24.7

5. Discussion

Data

Only a limited amount of representative EV charging data is available in the case studies due to the interference of Covid. In this study, the period from 01-03-2022 until 13-11-2022, approximately eight months, is considered suitable for EV charging. Preferably, one year of data is utilized for model development because models become generally more robust since they can capture seasonality within a year if present. For example, the temperature has a significant effect on the performance of Li-ion batteries in EVs [61]. Therefore, seasonal differences might exist between the charging loads in summer and winter which are currently not captured due to the data limitation. Especially with the single-step and direct multi-step forecasting models, a sufficient data set is essential since these model configurations generate forecasts purely on historical data.

Additionally, a sub-optimal feature is utilized in the development of the aggregated EV charging load forecasting model. The previous week's number of EV connections is used as a lag feature. However, the number of connections also includes fully charged EVs that are not participating anymore. For that reason, it would be beneficial to only consider the number of connected EVs whose batteries are not fully charged. Such a correction would improve the second hard constraint of the maximum aggregated charging power.

Furthermore, energy consumption data with an hourly time interval are used in the model development of the building load forecasting model due to discrepancies in the power consumption data. Herein, mismatches are observed between the power consumption data of various subsystems. The mismatches are probably caused by the use of different measurement devices and different sampling times. The use of hourly energy consumption data is inevitable and seen as the best alternative for power consumption data with higher resolutions.

Model development

The number of iterations in the hyperparameter optimization of the single-step forecasting models is set to twenty-five. For the hyperparameter optimization of the initial multi-step forecasting models, the number of iterations is lowered to ten aiming to reduce computational time. With RandomizedSearchCV, ten randomly selected hyperparameter combinations are selected and evaluated. The best-performing combination of hyperparameter values can be seen as an optimal set. The disadvantage of RandomizedSearchCV is its inability to include the performance of the previously evaluated hyperparameter values. Therefore, the model might waste a significant amount of time with computations of irrelevant hyperparameter combinations. For the multi-step forecasting models trained on one month, Bayesian optimization is applied with ten iterations. The advantage of Bayesian optimization is its ability to consider the performance of previously used hyperparameter values. Hence, the optimization algorithm can further improve the set of hyperparameter values each iteration [62]. The hyperparameter optimization is already improved with the application of Bayesian optimization, however, the number of ten iterations remains critical. There might be a set of hyperparameter values that performs better but which are not evaluated due to the limited number of iterations. Therefore, the current set of hyperparameter values might be sub-optimal. Ideally, more iterations would be preferable since more hyperparameter combinations can be evaluated.

The RandomForestRegressor is assumed to be the optimal regressor for both the recursive and direct multi-step forecasting models. This assumption is purely based on the performances and interpretability of the XGB and RF single-step forecasting models. Other regressors might be more suitable.

The DIR3-672 model is not considered in this study due to its significant development time. In further research, it is worthwhile to investigate this model configuration because the DIR3-96 model trained on the full data set already outperforms the other models. It is expected that a DIR3-672 model can perform even better due to the utilization of more lags.

Furthermore, the single-step and multi-step forecasting models both generate point forecasts. The disadvantage of point forecasts is their inability to deal with their associated uncertainty. The addition of probabilistic prediction intervals might be a solution to deal with the uncertainty of point forecasts [63].

Model selection

In this study, the DIR3-96 model, trained on the fully available train data, performs best. However, the REC1-672 trained on one month of data is selected for implementation in the MPC. Herein, the trade-off between performance and complexity is leading in the decision-making. In total three forecasting models are required for the MPC simulation, therefore, a long development time is not desired. The disadvantage of the currently used REC1-672 model is its significant dependency on the previous week. As a result, problems may arise after the holidays. Therefore, the application of this method should be further investigated or enhanced with exogenous variables when implemented in reality. But for the considered test week, the REC1-672 model is sufficient to prove the concept.

MPC

The aggregated EV energy demand is based on day-ahead forecasts. Herein mismatches can occur on aggregated level by under- or overestimation. At the individual EV level, the mismatch between the actual charged and desired amount of energy can vary even more. The aggregated EV approach does not allow direct assessment of the individual EV driver satisfaction Therefore, with real-life implementation, it is wise to consider a priority zone where EV drivers can charge uncontrolled to avoid significant mismatches between individual energy demands. Such measure will reduce the smart charging potential but ensures the satisfaction and acceptance of EV drivers. Moreover, the used forecasting model for the determination of the daily aggregated EV load is limited since it can only be trained until the implementation of a controller. Afterward, re-training is not possible anymore due to the interference of the controller. Hence, it is questionable how long this method will remain viable considering the increasing trend of the EV population. Finally, the used MPC model is a simplification of the MPC model from [59]. As a result, the infeasibility of the optimization problem was noticeable during the test week. The infeasibility can be solved by improving the MPC model, however, that is not the scope of this study.

6. Conclusions

This study aims to investigate the effects of data-driven forecasting models on the controlling abilities of EMSs. More specifically, this study evaluates if data-driven forecasting models can overcome the lack of user-specific EV information in the application of MPC for smart charging. In doing so, aggregated EV charging load forecasting models are thoroughly explored at two similar workplaces with different spatial dimensions. Moreover, various data-driven forecasting models are explored in each case study. Averaged-driven and ML-driven forecasting models are considered. The goal of this research is threefold. First, it is desired to investigate the viability of forecasting models at different spatial levels. Second, it is investigated which forecasting method is most suitable considering the trade-off between model performance and complexity at the different spatial levels. Third, the requirements for the constitution of these forecasting models and the implementation into MPC are investigated.

Which set of features and algorithms leads to the most accurate ML-driven EV forecasting model?

Several forecasting model configurations are evaluated. Initially, single-step forecasting models based on the algorithms XGB and RF are evaluated. A set of features is considered, containing: temporal features, lags, a holiday dummy, and weather features (TEMP and GHI). The weather features appeared to be irrelevant and are ultimately dropped through multicollinearity and feature importance. The remaining set of features consists of the time of day, weekday, month, holiday dummy, and the previous week's number of connections lag. Besides, the exclusion of weather features allows the forecasting model to forecast week-ahead instead of day-ahead. The RF algorithm slightly outperformed the XGB algorithm with single-step forecasting in both case studies. If RF is combined with a multi-step forecasting method, the performance can increase even further, while also considering a forecasting horizon in real time. The DIR3-96 forecasting model trained on the full train data set performs best, but DIR3-672 is potentially more accurate.

Are spatial effects noticeable when comparing the forecasting performance of small fleets with large fleets?

There are significant spatial differences observed. The EV charging at larger populations is more consistent in comparison to smaller populations. The forecasting performances of two case studies with different spatial levels are compared with the error metric NRMSE. The RF model, XGB model, and averaged-driven forecasting model at Building B perform fifty-nine, fifty-seven, and forty-eight percent better on average in comparison to the same models at Building A. In addition, the major advantage of larger EV populations is the independency of user-specific information. Accurate forecasts can be generated without the cooperation of EV drivers. Thus, operating on aggregated level is more robust since a model is not dependent on the cooperation of EV drivers and is not sensitive to privacy-related issues, which reduces the burdens for real-life implementation.

How do ML-driven forecasting models perform in comparison to averaged forecasting models, considering the trade-off between performance and complexity?

At small EV populations, such as Building A, the averaged-driven forecasting model outperforms the more advanced RF and XGB ML-driven algorithms with NRMSE scores of 0.125, 0.139, and 0.140 respectively. Since small EV populations are prone to significant

randomness, models cannot find any useful pattern in the historical data or they are learning non-existing patterns based on randomness. Nevertheless, the forecasting performance of the averaged-driven model is still very poor and not suitable for implementation into a controller. Therefore, it can be concluded that the requirement of user-specific information in small case studies, an EV population of approximately five EVs in this case, is inevitable. At larger EV populations, such as building B, both ML- and averaged-driven forecasting performances increase significantly due to more repetitive patterns and less influence of individual EV randomness, which decreases the complexity of the forecasting problem. The RF algorithm outperforms the XGB algorithm and the averaged-driven model with NRSME scores of 0.057, 0.060, and 0.065 respectively. ML-driven models become favorable since they are fed with exogenous features that allow the models to identify and learn patterns to improve the understanding of underlying factors that influence the forecasted variable. In contrast, averaged-driven forecasting models are solely trained on historical EV charging data. Hereby all occurrences for specific weekdays or holidays are combined into a single average profile. The disadvantage of averaged-driven forecasting models is their inability to deal with trends. The weight of historical data in comparison to the relatively new data, that are exposed to a trend, results in under- or overestimation of the target. In other words, the size of historical data to train averaged-driven forecasting models delays the ability to capture trends.

What are the requirements to implement a forecasting model into an MPC and what are the different effects of the various forecasting models on the MPC?

The required data for a forecasting model depend on the objective. In the case of aggregated EV charging load forecasts, historical aggregated power consumption data and the number of connections are required. The same historical data about the number of connections is necessary to forecast the number of connections itself. The building load can be forecasted with historical building power consumption data. Hereby, data quality is an often forgotten aspect. In this study, various sampling times in combination with different measurement devices resulted in mismatches between power consumption data. Therefore, it is recommended to collect data with a high and similar time resolution to prevent mismatches. A high resolution also prevents the necessity of interpolation with upsampling. Furthermore, it is recommended to apply multi-step forecasting models in MPC instead of single-step forecasting models. Multi-step forecasting models can generate real-time forecasts while considering a specified forecasting horizon. The data sensitivity analysis shows that accurate multi-step forecasting models can already be generated after one month. As a result, real-life implementation is almost immediately feasible after the installation of EV charging infrastructure. In doing so, periodical re-training is crucial for forecasting models that are exposed to a trend, such as the increasing EV population. Without re-training, forecasting models may not be able to capture trends. Therefore, it is strongly advised to re-train forecasting on a weekly or monthly basis.

The main advantage of applying data-driven EV forecasting models on an aggregated scale (up to approximately a hundred EVs) is related to its smart charging applicability in real-life operations. User-specific information is not required to accurately forecast the number of connected EVs. Hence, forecasting models become more robust since they are not dependent on the willingness of EV drivers to cooperate. In addition, privacy issues are avoided because EMSs do not collect personal information. Finally, operating on aggregated scale allows MPC

to solve the optimization problem as a single-objective instead of a multi-objective problem where each EV is considered separately in the optimization problem.

Further research

Finally, some topics remain uninvestigated, which are interesting for further research. A summation of these uninvestigated topics is given below:

- The direct multi-step forecasting model with the application of 672 lags could be investigated. This model configuration is excluded from this study due to time constraints but has the potential to outperform the other model configurations.
- Other ML-driven algorithms could be evaluated in a multi-step forecasting setup. This study only focused on RF, whereas other ML algorithms might perform even better.
- Stacking forecasting models in a multi-step forecasting configuration with a meta-model could be interesting to further improve forecasting performances. Stacking leverages the strengths of each forecasting model and combines them to generate a final output.
- This study relies on point forecasts. The addition of probabilistic prediction intervals might be a solution to reduce the uncertainty of point forecasts. It would be interesting to investigate the effect of prediction intervals on forecasting accuracy and the performance of the MPC.

7. Bibliography

[1] “The Paris Agreement - Publication | UNFCCC.” https://unfccc.int/documents/184656 (accessed Nov. 07, 2022).

[2] “‘Fit for 55’: delivering the EU’s 2030 Climate Target on the way to climate neutrality,” European Commission, Brussels, Jul. 2021. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52021DC0550&from=EN

[3] “StatLine - Energiebalans; aanbod en verbruik, sector.” https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83989NED/table?ts=1676538514669 (accessed Feb. 16, 2023).

[4] “Report of the Conference of the Parties serving as the meeting of the Parties to the Paris Agreement on its third session, held in Glasgow from 31 October to 13 November 2021. Addendum. Part two: Action taken by the Conference of the Parties serving as the meeting of the Parties to the Paris Agreement at its third session | UNFCCC.” https://unfccc.int/documents/460950 (accessed Mar. 28, 2023).

[5] “StatLine - Warmtepompen; aantallen, thermisch vermogen en energiestromen.” https://opendata.cbs.nl/#/CBS/nl/dataset/82380NED/table?ts=1668001674668 (accessed Feb. 16, 2023).

[6] C. B. voor de Statistiek, “Groei aantal stekkerauto’s zet door,” *Centraal Bureau voor de Statistiek*, Oct. 14, 2021. https://www.cbs.nl/nl-nl/nieuws/2021/41/groei-aantal-stekkerauto-s-zet-door (accessed Feb. 16, 2023).

[7] N. Hoogervorst *et al.*, “Startanalyse aardgasvrije buurten (versie 2020, 24 september 2020); Gemeenterapport met toelichting bij tabellen met resultaten van de Startanalyse,” Planbureau voor de Leefomgeving (PBL), Den Haag.

[8] ElaadNL, “Outlook Q3 2021 Elektrisch rijden in stroomversnelling.” Oct. 2021.

[9] S. O’Connell, G. Reynders, and M. M. Keane, “Impact of source variability on flexibility for demand response,” *Energy*, vol. 237, p. 121612, Dec. 2021, doi: 10.1016/j.energy.2021.121612.

[10] K. Van den Bergh and E. Delarue, “Cycling of conventional power plants: Technical limits and actual costs,” *Energy Conversion and Management*, vol. 97, pp. 70–77, Jun. 2015, doi: 10.1016/j.enconman.2015.03.026.

[11] M. Z. Oskouei *et al.*, “A Critical Review on the Impacts of Energy Storage Systems and Demand-Side Management Strategies in the Economic Operation of Renewable-Based Distribution Network,” *Sustainability*, vol. 14, no. 4, Art. no. 4, Jan. 2022, doi: 10.3390/su14042110.

[12] P. Sharifi, A. Banerjee, and M. J. Feizollahi, “Leveraging owners’ flexibility in smart charge/discharge scheduling of electric vehicles to support renewable energy integration,” *Computers & Industrial Engineering*, vol. 149, Nov. 2020, doi: 10.1016/j.cie.2020.106762.

[13] C. S. Loakimidis, D. Thomas, P. Rycerski, and K. N. Genikomsakis, “Peak shaving and valley filling of power consumption profile in non-residential buildings using an electric vehicle parking lot,” *Energy*, vol. 148, pp. 148–158, Apr. 2018.

[14] S. Habib, M. Khan, F. Abbas, and H. Tang, “Assessment of electric vehicles concerning impacts, charging infrastructure with unidirectional and bidirectional chargers, and power flow comparisons,” *International Journal of Energy Research*, vol. 42, Oct. 2017, doi: 10.1002/er.4033.

[15] Q.-U. Ain, S. Iqbal, and N. Javaid, “User Comfort Enhancement in Home Energy Management Systems using Fuzzy Logic,” 2019.

[16] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, “Review on model predictive control: an engineering perspective,” *Int J Adv Manuf Technol*, vol. 117, no. 5, pp. 1327–1349, Nov. 2021, doi: 10.1007/s00170-021-07682-3.

[17] “Model predictive control,” *Wikipedia*. Feb. 17, 2023. Accessed: Feb. 27, 2023. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Model_predictive_control&oldid=1139919048

[18] W. Somers, W. Khan, K. de Bont, and W. Zeiler, “Individual EV load profiling and smart charging to flatten total electrical demand,” in *CLIMA 2022 conference*, May 2022. doi: 10.34641/clima.2022.164.

[19] W. Somers, W. Khan, K. de Bont, and W. Zeiler, “Forecasting individual and aggregated electric vehicle charging loads at offices with machine learning,” Eindhoven University of Technology, Eindhoven, Jul. 2022.

[20] O. Sadeghian, A. Oshnoei, B. Mohammadi-ivatloo, V. Vahidinasab, and A. Anvari-Moghaddam, “A comprehensive review on electric vehicles smart charging: Solutions, strategies, technologies, and challenges,” *Journal of Energy Storage*, vol. 54, p. 105241, Oct. 2022, doi: 10.1016/j.est.2022.105241.

[21] C.-J. Boo, H.-C. Kim, S.-W. Ji, S.-Y. Yang, M.-J. Kang, and K. Y. Lee, “Optimized EV Charging Method Using Model Predictive Control Algorithm”.

[22] Y. Yang, H.-G. Yeh, and R. Nguyen, “A Robust Model Predictive Control-Based Scheduling Approach for Electric Vehicle Charging With Photovoltaic Systems,” *IEEE Systems Journal*, vol. 17, no. 1, pp. 111–121, Mar. 2023, doi: 10.1109/JSYST.2022.3183626.

[23] G. Van Kriekinge, C. De Cauwer, N. Sapountzoglou, T. Coosemans, and M. Messagie, “Peak shaving and cost minimization using model predictive control for uni- and bi-directional charging of electric vehicles,” *Energy Reports*, vol. 7, pp. 8760–8771, Nov. 2021, doi: 10.1016/j.egyr.2021.11.207.

[24] W. Su, J. Wang, K. Zhang, and A. Q. Huang, “Model predictive control-based power dispatch for distribution system considering plug-in electric vehicle uncertainty,” *Electric Power Systems Research*, vol. 106, pp. 29–35, Jan. 2014, doi: 10.1016/j.epsr.2013.08.001.

[25] Y. Tang *et al.*, “A survey on machine learning models for financial time series forecasting,” *Neurocomputing*, vol. 512, pp. 363–380, Nov. 2022, doi: 10.1016/j.neucom.2022.09.003.

[26] Y. Shi, H. D. Tuan, A. V. Savkin, T. Q. Duong, and H. V. Poor, “Model Predictive Control for Smart Grids With Multiple Electric-Vehicle Charging Stations,” *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 2127–2136, Mar. 2019, doi: 10.1109/TSG.2017.2789333.

[27] Z. Yi *et al.*, “A highly efficient control framework for centralized residential charging coordination of large electric vehicle populations,” *International Journal of Electrical Power & Energy Systems*, vol. 117, p. 105661, May 2020, doi: 10.1016/j.ijepes.2019.105661.

[28] R. Khatami, M. Parvania, and A. Bagherinezhad, “Continuous-time Model Predictive Control for Real-time Flexibility Scheduling of Plugin Electric Vehicles,” *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 498–503, Jan. 2018, doi: 10.1016/j.ifacol.2018.11.752.

[29] “Integrated Energy Exchange Scheduling for Multimicrogrid System With Electric Vehicles | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/7154496 (accessed Mar. 03, 2023).

[30] J. Su, T. T. Lie, and R. Zamora, “A rolling horizon scheduling of aggregated electric vehicles charging under the electricity exchange market,” *Applied Energy*, vol. 275, p. 115406, Oct. 2020, doi: 10.1016/j.apenergy.2020.115406.

[31] C. Wu, S. Jiang, S. Gao, Y. Liu, and H. Han, “Event-triggered model predictive control for dynamic energy management of electric vehicles in microgrids,” *Journal of Cleaner Production*, vol. 368, p. 133175, Sep. 2022, doi: 10.1016/j.jclepro.2022.133175.

[32] C. Wu, S. Gao, Y. Liu, T. E. Song, and H. Han, “A model predictive control approach in microgrid considering multi-uncertainty of electric vehicles,” *Renewable Energy*, vol. 163, pp. 1385–1396, Jan. 2021, doi: 10.1016/j.renene.2020.08.137.

[33] B. Tan and H. Chen, “Multi-objective energy management of multiple microgrids under random electric vehicle charging,” *Energy*, vol. 208, p. 118360, Oct. 2020, doi: 10.1016/j.energy.2020.118360.

[34] Y. Zhang and L. Cai, “Dynamic Charging Scheduling for EV Parking Lots With Photovoltaic Power System,” *IEEE Access*, vol. 6, pp. 56995–57005, 2018, doi: 10.1109/ACCESS.2018.2873286.

[35] S. Limmer, “Evaluation of Optimization-Based EV Charging Scheduling with Load Limit in a Realistic Scenario,” *Energies*, vol. 12, no. 24, Art. no. 24, Jan. 2019, doi: 10.3390/en12244730.

[36] “Cost-Constrained Dynamic Optimal Electric Vehicle Charging | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/7586066 (accessed Mar. 03, 2023).

[37] K. Zhou, L. Cheng, L. Wen, X. Lu, and T. Ding, “A coordinated charging scheduling method for electric vehicles considering different charging demands,” *Energy*, vol. 213, p. 118882, Dec. 2020, doi: 10.1016/j.energy.2020.118882.

[38] “A Real-Time EV Charging Scheduling for Parking Lots With PV System and Energy Store System | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/8750840 (accessed Mar. 03, 2023).

[39] F. Wu and R. Sioshansi, “A two-stage stochastic optimization model for scheduling electric vehicle charging loads to relieve distribution-system constraints,” *Transportation Research Part B: Methodological*, vol. 102, pp. 55–82, Aug. 2017, doi: 10.1016/j.trb.2017.05.002.

[40] P. Klapwijk and L. Driessen, *EV Related Protocol Study*. 2017.

[41] “KNMI - Uurgegevens van het weer in Nederland.” https://www.knmi.nl/nederland-nu/klimatologie/uurgegevens (accessed Mar. 02, 2023).

[42] “Solar Irradiance data.” https://solargis.com/ (accessed Jun. 23, 2022).

[43] O. Akinwande, H. G. Dikko, and S. Agboola, “Variance Inflation Factor: As a Condition for the Inclusion of Suppressor Variable(s) in Regression Analysis,” *Open Journal of Statistics*, vol. 05, pp. 754–767, Jan. 2015, doi: 10.4236/ojs.2015.57075.

[44] J. I. Daoud, “Multicollinearity and Regression Analysis,” *J. Phys.: Conf. Ser.*, vol. 949, p. 012009, Dec. 2017, doi: 10.1088/1742-6596/949/1/012009.

[45] W. Samek, T. Wiegand, and K.-R. Müller, “Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models.” arXiv, Aug. 28, 2017. Accessed: Mar. 10, 2023. [Online]. Available: http://arxiv.org/abs/1708.08296

[46] S. Shahriar, A. Al-Ali, A. Osman, S. Dhou, and M. Nijim, “Machine Learning Approaches for EV Charging Behavior: A Review,” *IEEE Access*, vol. 8, pp. 168980–168993, Oct. 2020, doi: 10.1109/ACCESS.2020.3023388.

[47] “XGBoost Documentation — xgboost 1.6.0 documentation.” https://xgboost.readthedocs.io/en/stable/ (accessed Apr. 25, 2022).

[48] “sklearn.ensemble.RandomForestRegressor,” *scikit-learn*. https://scikit-learn/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html (accessed Apr. 25, 2022).

[49] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, pp. 5–32.

[50] J. Bergstra and Y. Bengio, “Random Search for Hyper-Parameter Optimization,” *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.

[51] “Scikit-Learn - Cross-Validation & Hyperparameter Tuning Using GridSearch by Sunny Solanki.” https://coderzcolumn.com/tutorials/machine-learning/scikit-learn-sklearn-cross-validation-and-hyperparameter-tuning-using-gridsearch (accessed Apr. 25, 2022).

[52] S. Ben Taieb and G. Bontempi, “Recursive Multi-step Time Series Forecasting by Perturbing Data,” in *2011 IEEE 11th International Conference on Data Mining*, Dec. 2011, pp. 695–704. doi: 10.1109/ICDM.2011.123.

[53] “Skforecast: time series forecasting with python and scikit learn.” https://www.cienciadedatos.net/documentos/py27-time-series-forecasting-python-scikitlearn.html (accessed Mar. 10, 2023).

[54] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, “Multiple-output modeling for multi-step-ahead time series forecasting,” *Neurocomputing*, vol. 73, no. 10, pp. 1950–1957, Jun. 2010, doi: 10.1016/j.neucom.2009.11.030.

[55] “Welcome to skforecast - Skforecast Docs.” https://joaquinamatrodrigo.github.io/skforecast/0.4.3/index.html (accessed Mar. 17, 2023).

[56] “4.2. Permutation feature importance,” *scikit-learn*. https://scikit-learn/stable/modules/permutation_importance.html (accessed Jun. 21, 2022).

[57] “API Reference — SHAP latest documentation.” https://shap.readthedocs.io/en/latest/api.html#explainers (accessed Jun. 21, 2022).

[58] M. L. Bynum *et al.*, *Pyomo — Optimization Modeling in Python*, Third., vol. 67. in Springer Optimization and Its Applications, vol. 67. Cham, Switzerland: Springer Nature Switzerland AG, 2021.

[59] B. Hermans, “Model Predictive Control of Vehicle Charging Stations in Grid-Connected Microgrids,” Eindhoven University of Technology, Eindhoven, 2023.

[60] K. Reinders, “Model Predictive Controller for a Battery Energy Storage System to reshape the energy demand curve of an office,” Eindhoven University of Technology, Eindhoven, 2022.

[61] S. Ma *et al.*, “Temperature effect and thermal impact in lithium-ion batteries: A review,” *Progress in Natural Science: Materials International*, vol. 28, no. 6, pp. 653–666, Dec. 2018, doi: 10.1016/j.pnsc.2018.11.002.

[62] S. Shekhar, A. Bansode, and A. Salim, “A Comparative study of Hyper-Parameter Optimization Tools.” arXiv, Jan. 17, 2022. Accessed: Apr. 04, 2023. [Online]. Available: http://arxiv.org/abs/2201.06433

[63] O. F. Eikeland, F. D. Hovem, T. E. Olsen, M. Chiesa, and F. M. Bianchi, “Probabilistic forecasts of wind power generation in regions with complex topography using deep learning methods: An Arctic case,” *Energy Conversion and Management: X*, vol. 15, p. 100239, Aug. 2022, doi: 10.1016/j.ecmx.2022.100239.

A. Overview case studies

Building A

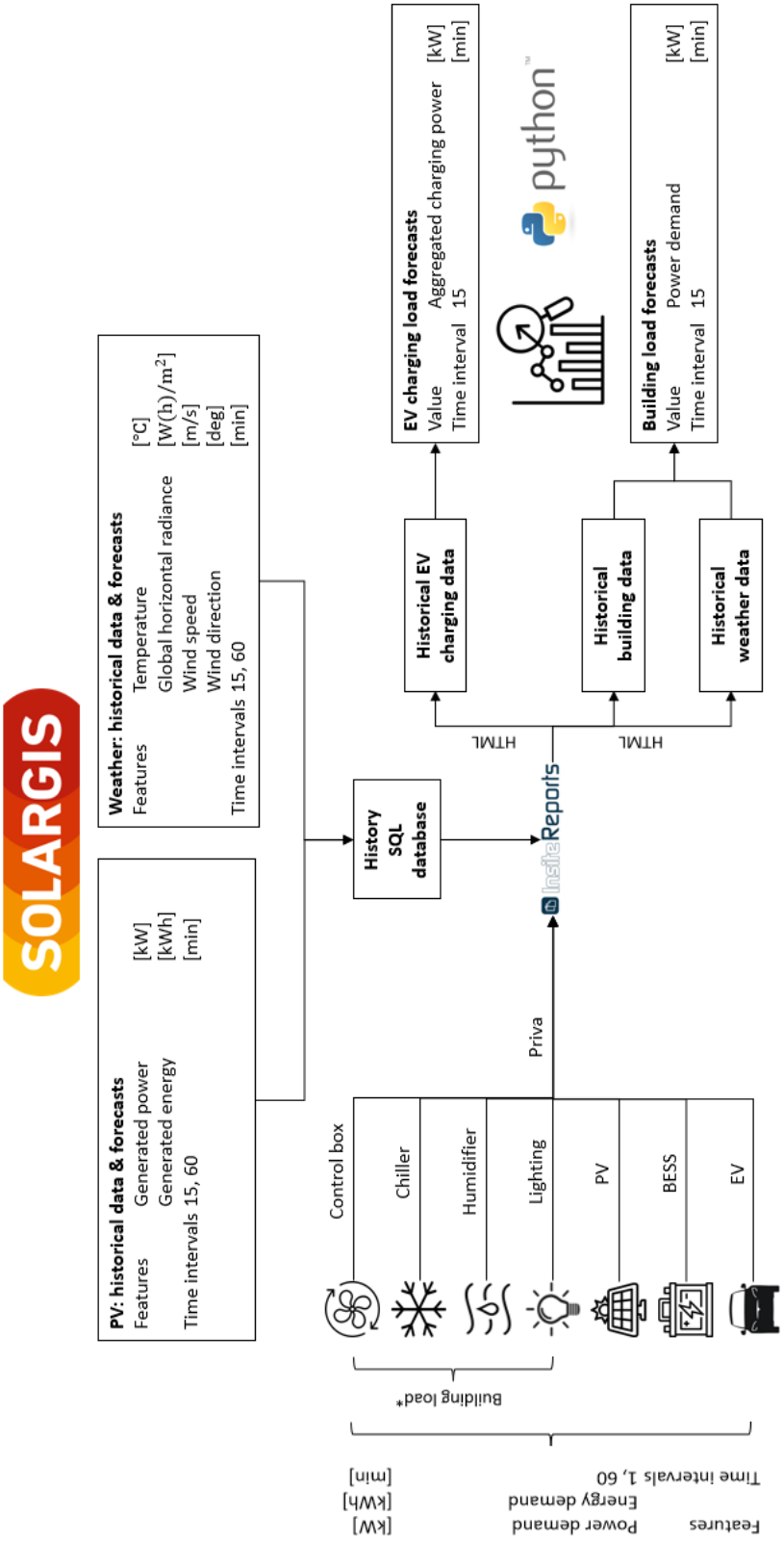

Figure A.1: Network map Building A.

Building B

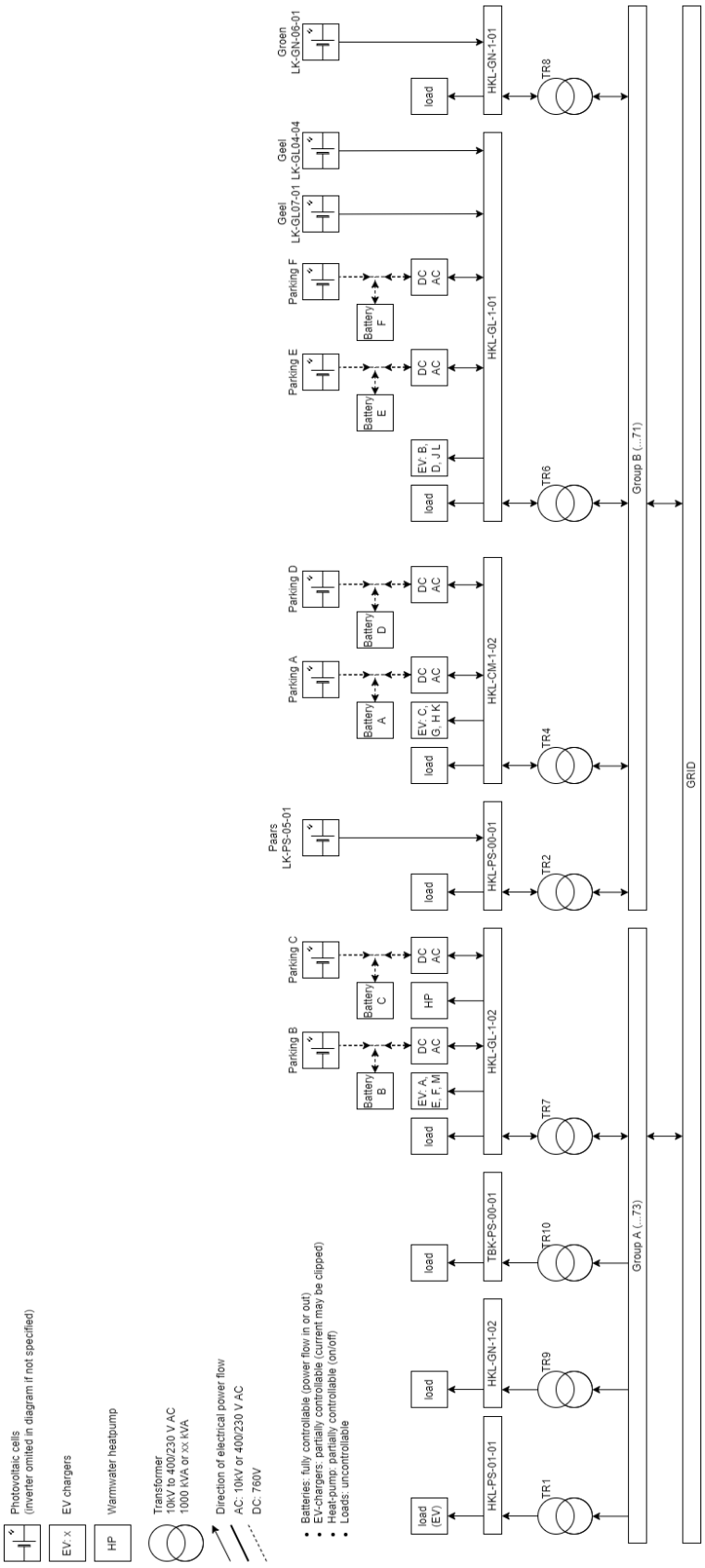

Figure A.2: Overview of electrical loads Building B.

B. Calendar plots

Calendar plots provide insights about seasonality, trends, and deviations in data throughout the year(s) with a daily resolution. In this study, calendar plots are valuable in the early stages to select appropriate data for model development.

Aggregated EV charging load Building A

Significant differences are observed between 2021 and 2022. The EV charging load increased in 2022 compared to 2021. The increasing EV charging load is probably caused by an increase in the EV population (trend) in combination with the Covid pandemic (deviation).

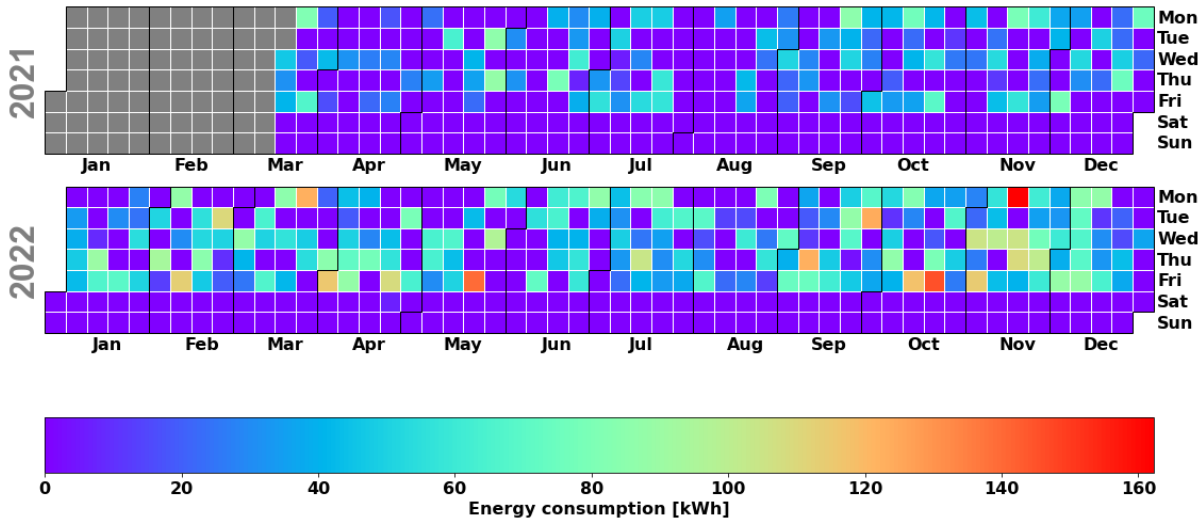

Figure B.1: Calendar plot aggregated EV charging load Building A.

Aggregated EV charging load Building B

The influence of the Covid pandemic and the increasing EV charging load is also noticeable in Building B. In 2021, the low EV charging loads are caused by Covid, whereas the increasing EV charging load is caused by a growing EV population in 2022.

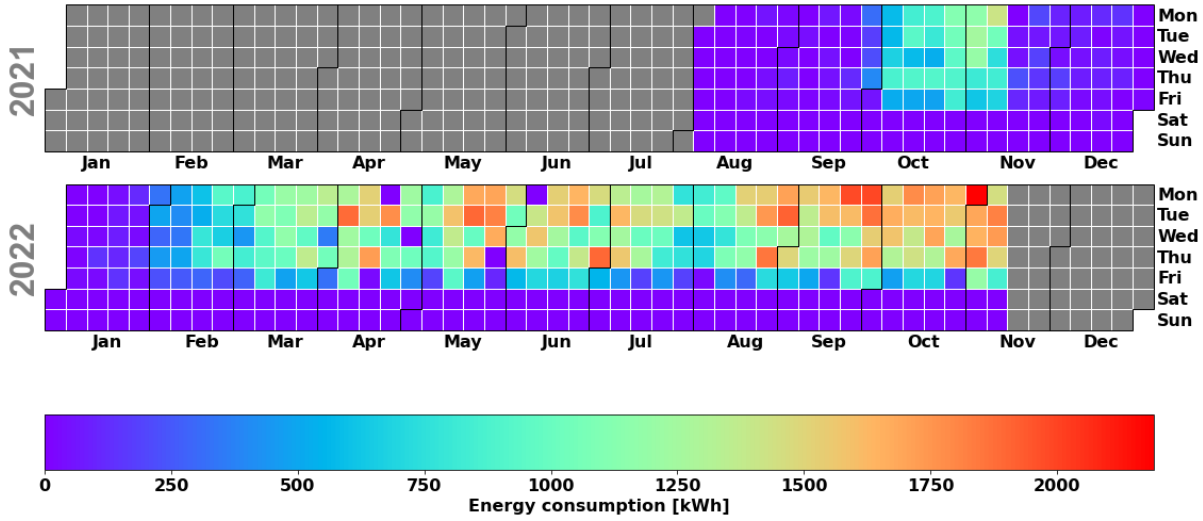

Figure B.2: Calendar plot aggregated EV charging load Building B.

Building load Building B

The years 2018, 2019, and 2020 deviate from 2022. In the first two years, the building load is significantly lower on the weekends in comparison to the load nowadays. The unexpected Covid period influenced the building load in 2020. In 2021, Covid is not the only factor that influences the building load. Malfunctioning of measurement devices causes zero building load in May and June. The building load data seems to be representative since October 2021.

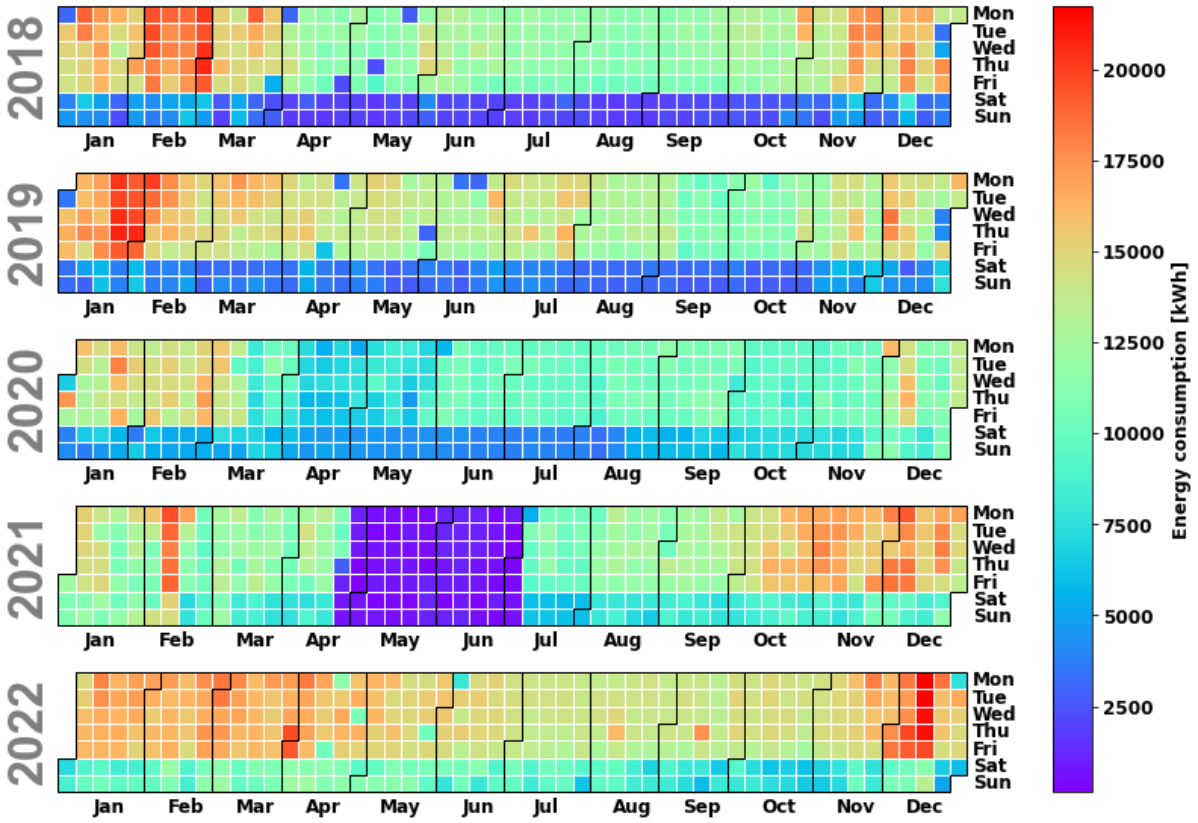

Figure B.3: Calendar plot aggregated building load Building B.

C. Evaluation of ACF and multicollinearity

Evaluation Building A

Temporal features, a holiday dummy, lag features, and weather features are initially used as input for the forecasting models. From this set of features, lags are the only category that can be changed and optimized. Therefore, ACF is applied to find the correlation of lags. The higher the correlation, the more it will enhance the forecasting performance. The first lags are correlated the most but are considered irrelevant since they limit the forecasting horizon. In the Figure below, a weekly pattern is visible. Therefore, the previous week's lag is included in the initial set of features.

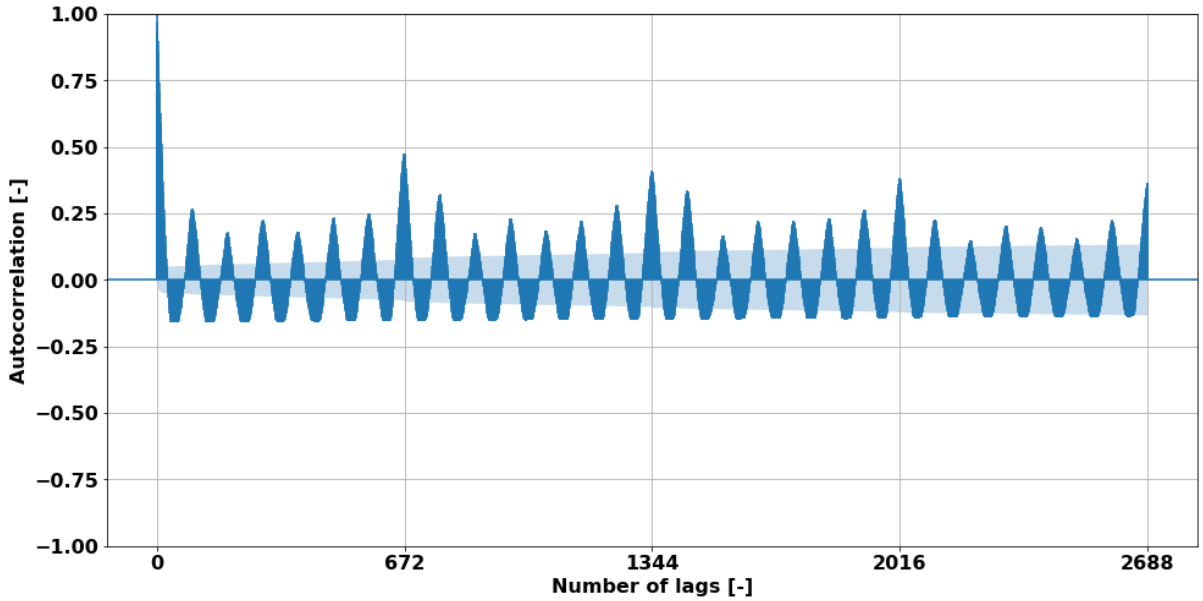

Figure C.1: ACF evaluation of the number of connected EVs Building A.

To enhance the interpretability of the forecasting models, multicollinearity is evaluated among the initial set of features. The features TEMP and month exceed the threshold of five in the first iteration. Thus, TEMP is removed since it has the highest VIF value. In the second iteration, the VIF value of 'Month' is significantly dropped. As a result, the threshold is not violated. The remaining features are used as input for the initial model development.

Table C.1: VIF analysis Building A

Features	Iteration 1	Iteration 2
TEMP	11.24	Removed
GHI	2.66	1.76
Number of connections previous week	1.29	1.28
Time of day	3.98	3.48
Day of week	2.89	2.89
Month	8.09	4.18
Holiday	1.03	1.03

Evaluation Building B

More significant correlations among the lags are noticeable. Again, weekly patterns are present in the data regarding the number of EV connections. Therefore, the previous week's lag is selected as the lag feature.

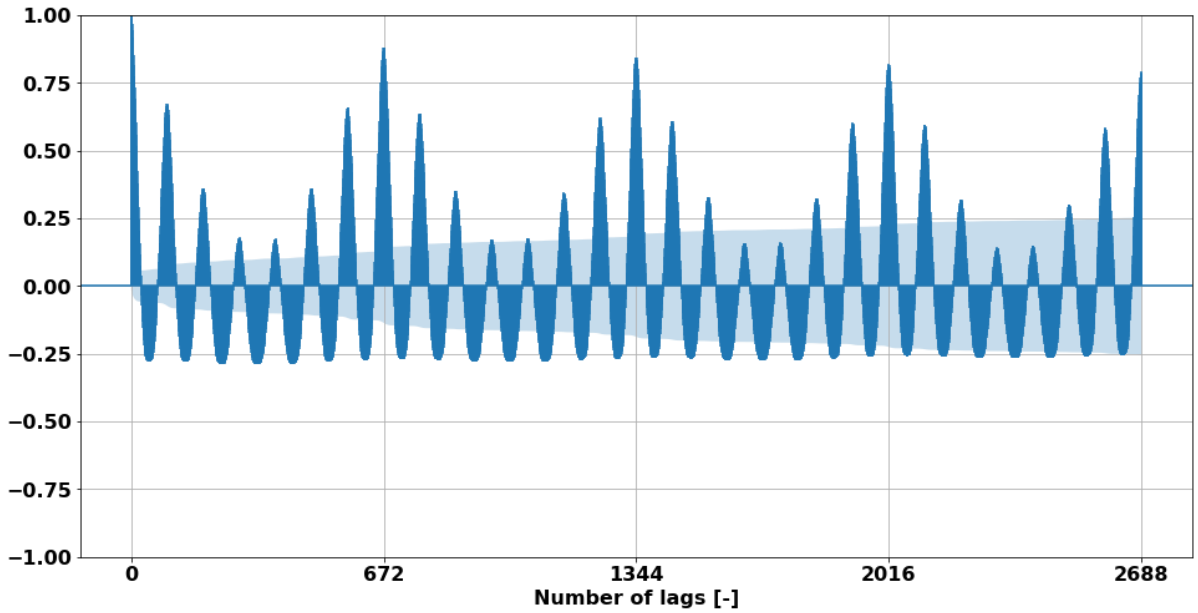

Figure C.2: ACF evaluation of the number of connected EVs Building B.

Similar results regarding multicollinearity are noticeable in Building B. The feature TEMP is removed after the first iteration, whereas the VIF value of the month feature drops below the threshold of five after the second iteration. The same initial set of features is used at Building B as is used at Building A.

Table C.2: VIF analysis Building B

Features	Iteration 1	Iteration 2
TEMP	12.68	Removed
GHI	3.36	2.31
Number of connections previous week	2.25	2.24
Time of day	3.87	3.47
Day of week	3.52	3.51
Month	10.17	4.94
Holiday	1.03	1.03

D. Feature selection of EV charging load forecasting models

Building A

A significant variation in feature importance is noticeable among the RF and XGB algorithm. The feature GHI seems to be important for the RF algorithm, whereas GHI has the lowest feature importance with the application of XGB. Therefore, the feature importance of GHI becomes arguable. In addition, GHI values are generated twenty-four hours ahead, which limits the EV charging load forecasting horizon. A week-ahead forecasting horizon can be achieved without the application of weather features. Despite the contradictions between RF and XGB, it is worthwhile to investigate the exclusion of GHI to lengthen the forecasting horizon and decrease model complexity. In the Table, a comparison is made between the forecasting performances with and without weather features. Astonishingly, the forecasting performance of both algorithms increases slightly. An increase in the XGB forecasting model is not unexpected due to the low GHI feature importance. However, GHI has a high feature importance for the RF forecasting model.

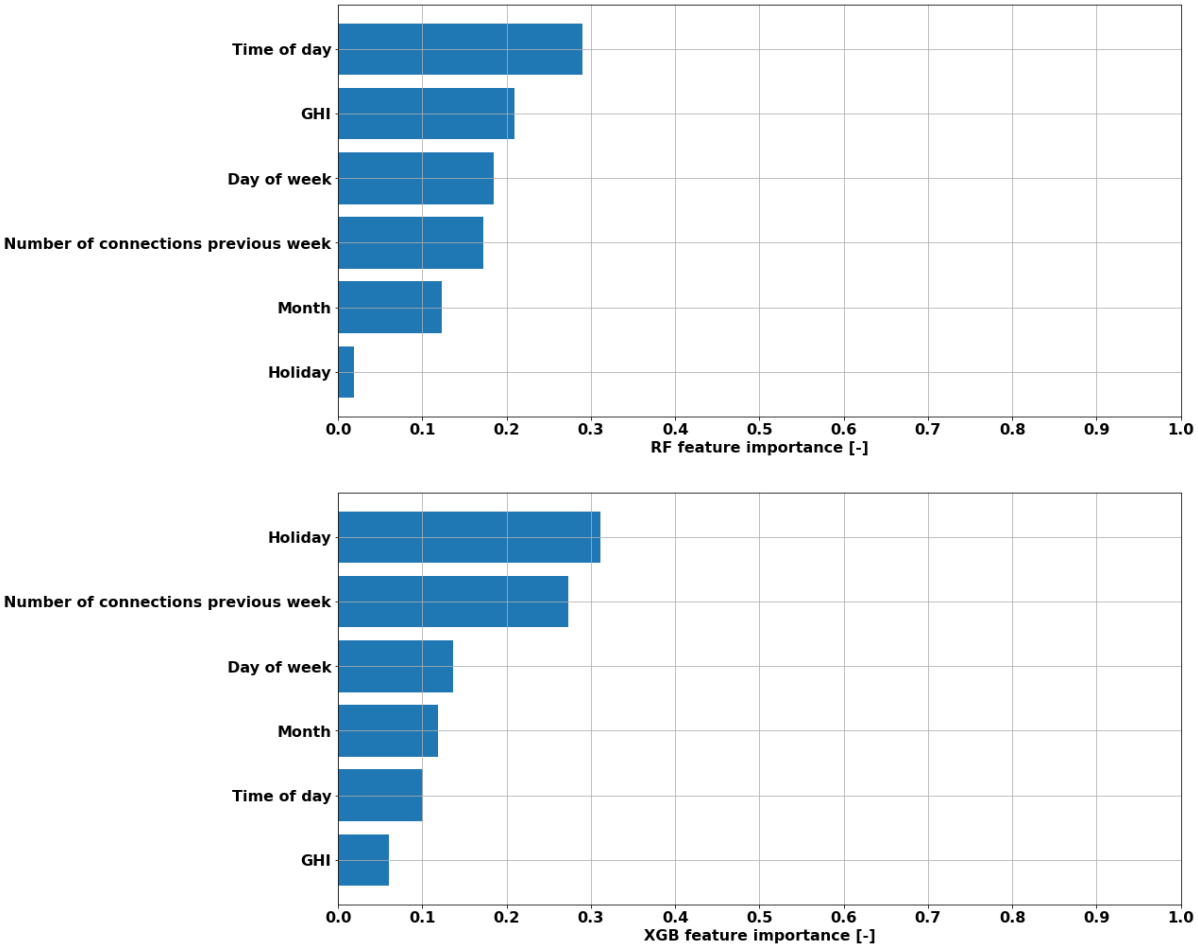

Figure D.1: Feature importance evaluation Building A.

Table D.1: Comparison of forecasting performances with or without weather features Building A.

Model	MAE [kW]	NRSME [-]
RF	1.426	0.139
RF + weather	1.500	0.143
XGB	1.456	0.140
XGB + weather	1.583	0.148

Building B

Similar results are obtained from the feature importance evaluation in comparison to Building A. The RF forecasting performance improved slightly, whereas the XGB forecasting performance decreased slightly. Therefore, GHI is removed from the set of input features to decrease the model complexity and increase the forecasting horizon.

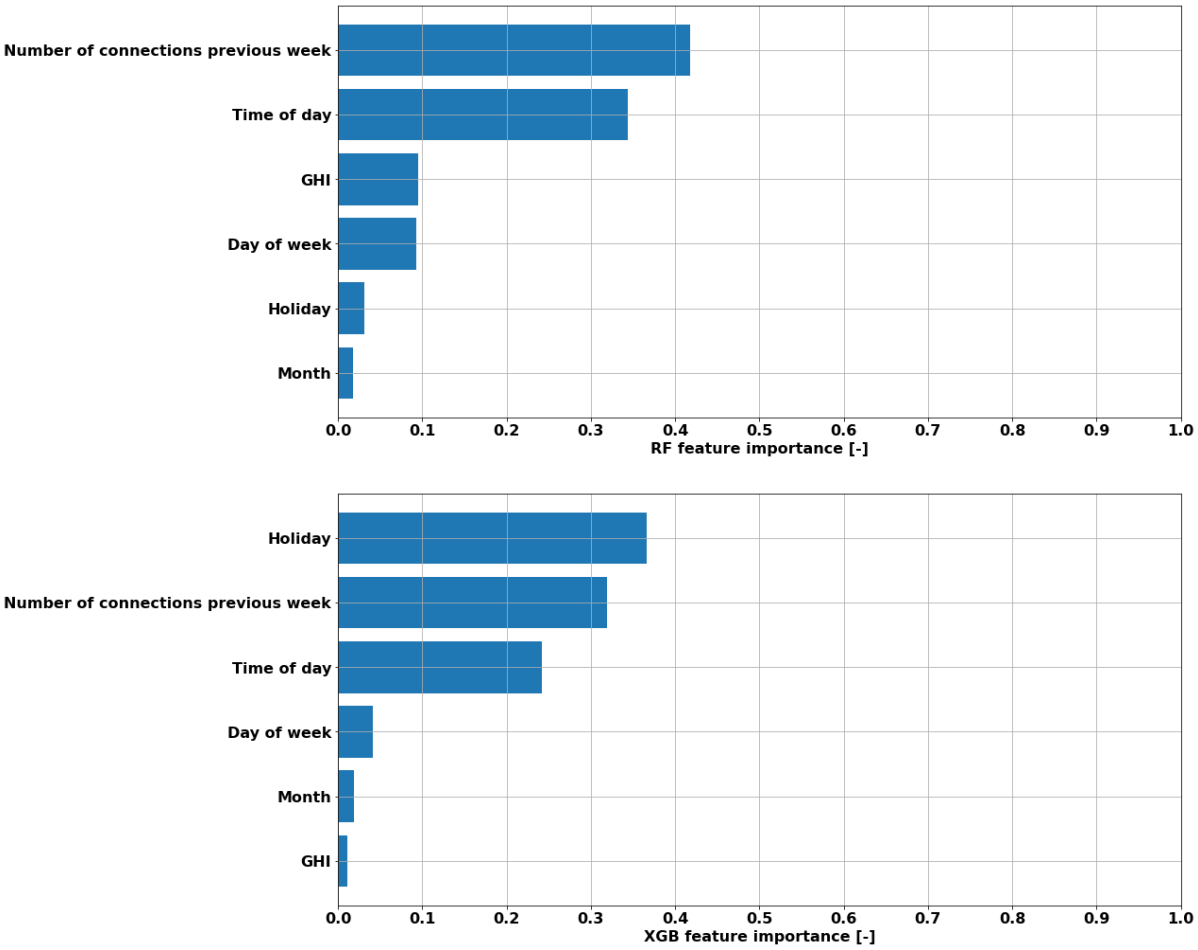

Figure D.2: Feature importance evaluation Building B.

Table D.2: Comparison of forecasting performances with or without weather features Building B.

Model	MAE [kW]	NRSME [-]
RF	11.17	0.057
RF + weather	11.76	0.065
XGB	11.92	0.060
XGB + weather	11.36	0.062

E. Hyperparameter optimization of forecasting models

Table E.1: Results of hyperparameter optimization single-step forecasting models.

Model	Hyperparameter	Default	Building A	Building B
XGB	learning_rate (eta)	0.30	0.11	0.16
	n_estimators	100	60	80
	max_depth	6	9	9
	min_child_weight	1	9	3
	gamma	0	1	10
	subsample	1	0.9	0.8
	colsample_bytree	1	0.9	0.9
	reg_lambda	1	0.3	0.3
	reg_alpha	0	0.3	0.7
RF	n_estimators	100	140	150
	max_depth	None	60	90
	min_samples_split	2	8	9
	min_samples_leaf	1	7	4
	max_features	‘auto’	‘sqrt’	‘sqrt’
	bootstrap	False	False	False

Table E.2: Results hyperparameter optimization multi-step forecasting models Building B.

Train data	Model configuration	n_estimators	max_depth	min_samples_split	min_samples_leaf
8 months	REC1-96	100	None	2	1
	REC2-96	90	80	7	3
	REC3-96	80	90	8	4
	DIR1-96	100	None	2	1
	DIR2-96	80	90	8	4
	DIR3-96	80	90	8	4
	REC1-672	100	None	2	1
	REC2-672	90	110	5	8
	REC3-672	100	140	5	9
1 month	REC1-96	100	None	2	1
	REC2-96	72	89	9	5
	REC3-96	72	89	9	5
	DIR1-96	100	None	2	1
	DIR2-96	112	88	4	2
	DIR3-96	112	88	4	2
	REC1-672	100	None	2	1
	REC2-672	112	88	4	2
	REC3-672	112	88	4	2

F. Visualization of data sensitivity analysis multi-step forecasting models

Visualization of performances forecasting models trained with eight months of data

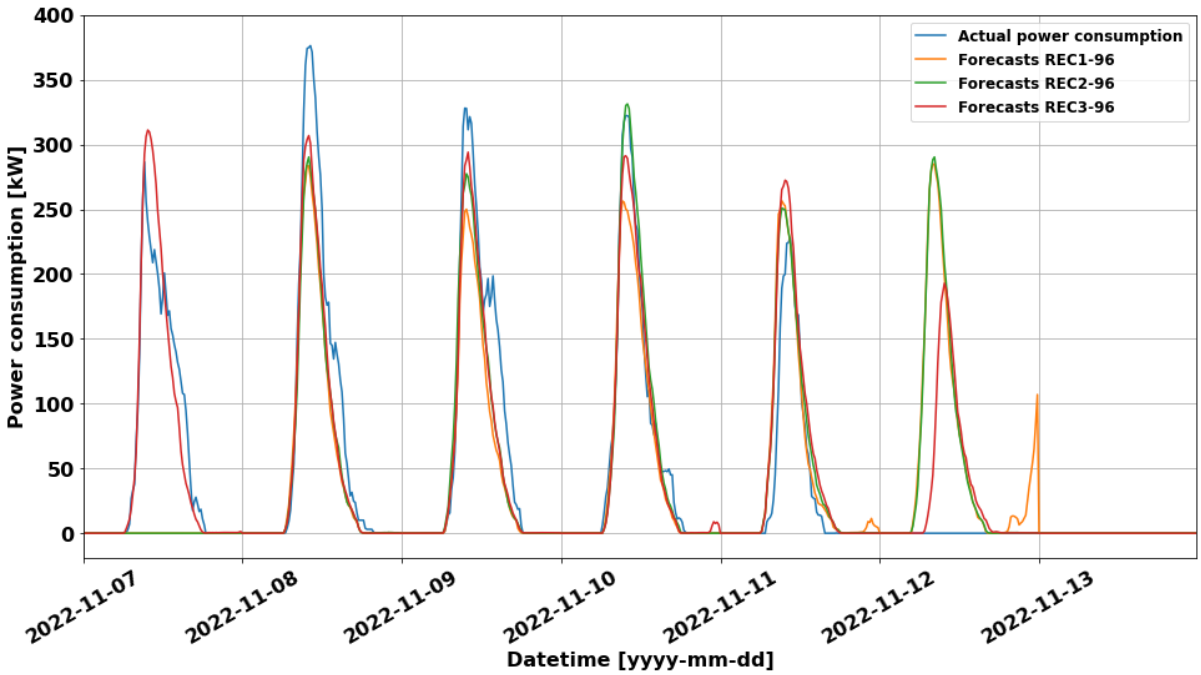

Figure F.1: REC-96 models trained with eight months of data.

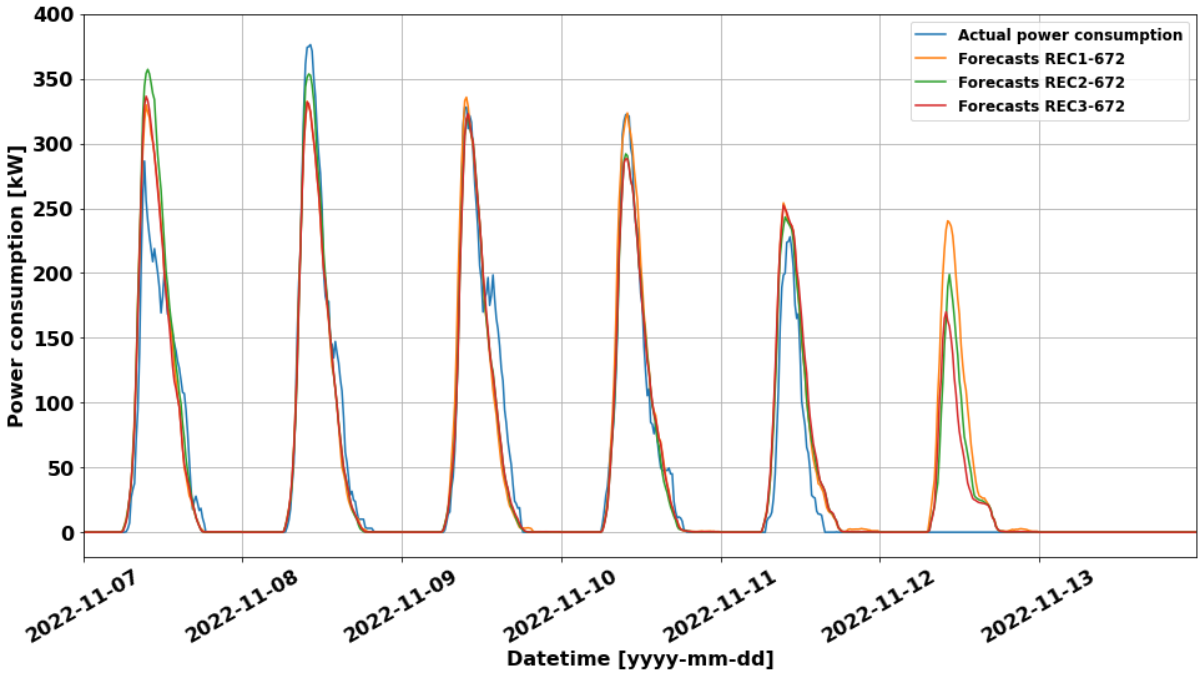

Figure F.2: REC-672 models trained with eight months of data.

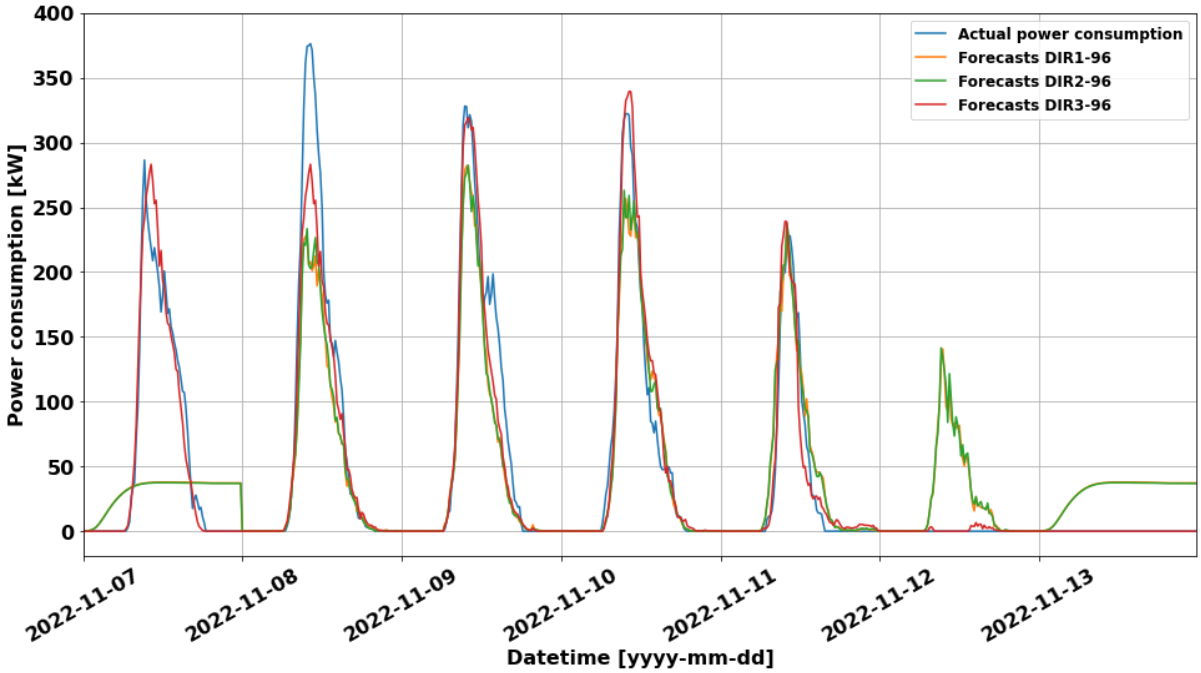

Figure F.3: DIR-96 models trained with eight months of data.

Visualization of performances forecasting models trained with one month of data

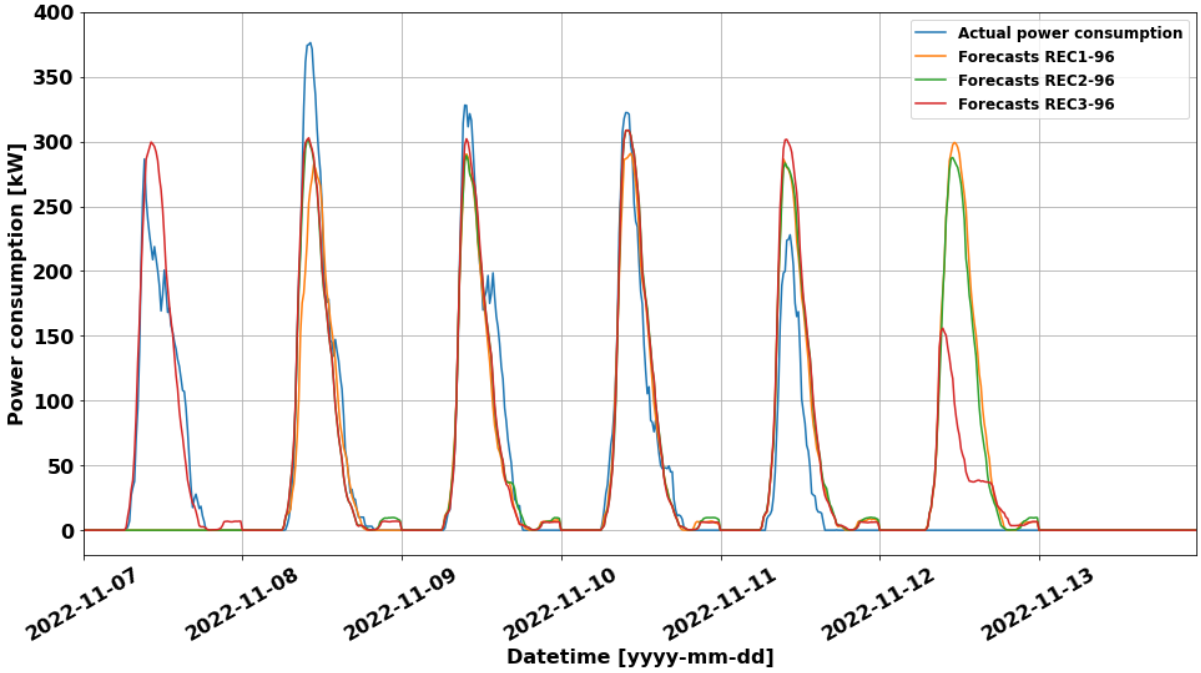

Figure F.4: REC-96 models trained with one month of data.

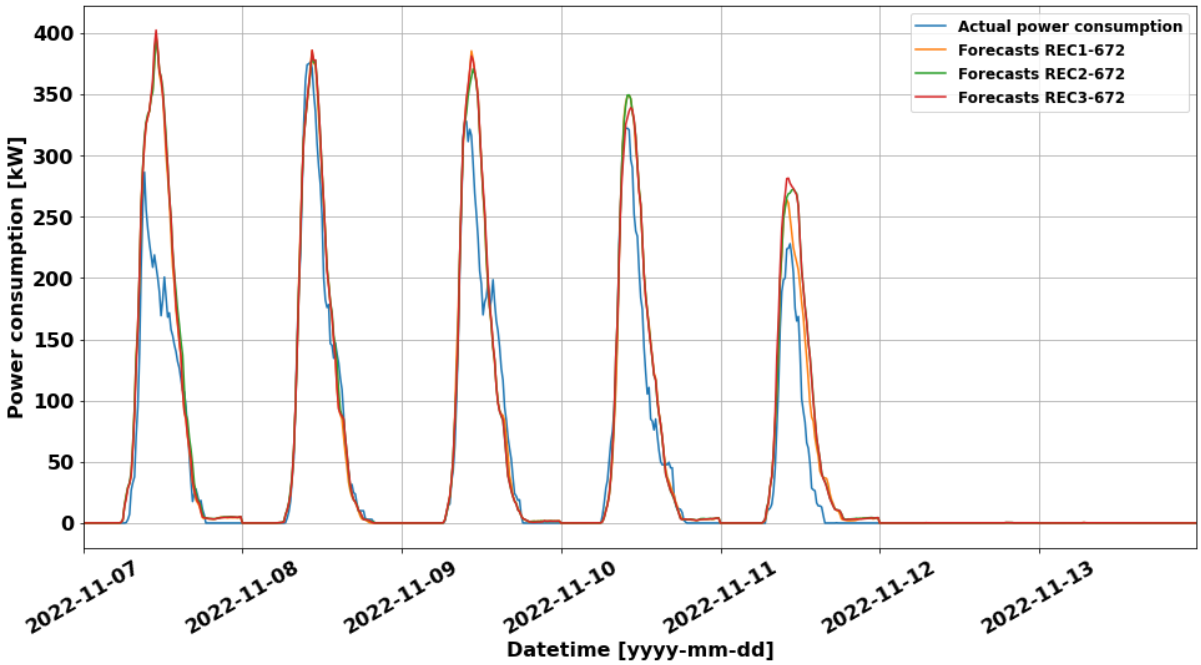

Figure F.5: REC-672 models trained with one month of data.

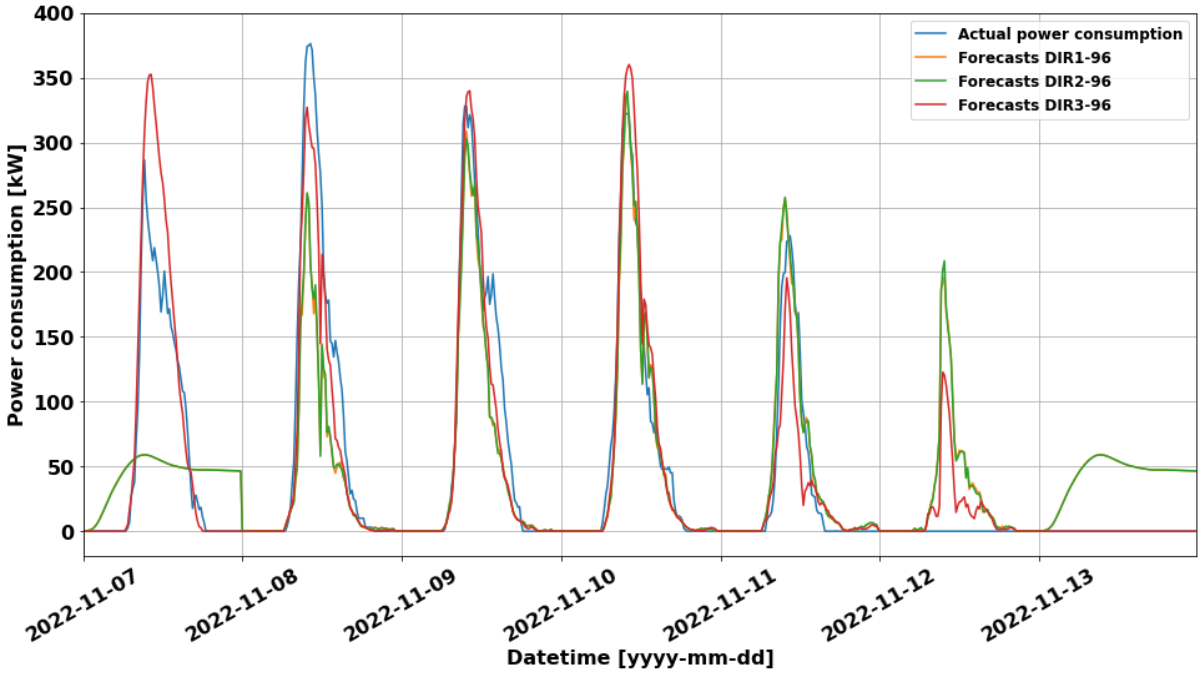

Figure F.6: DIR-96 models trained with one month of data.