

Prognose der Netzabgabemenge im Stromnetz der Vorarlberger Energienetze GmbH

Mit Bewertung der Prognosen durch die
Ausgleichsenergiepreise

Masterarbeit
zur Erlangung des akademischen Grades

Master of Science

Fachhochschule Vorarlberg
Studiengang: Nachhaltige Energiesysteme

Betreut von
Dr. Klaus Rheinberger

Vorgelegt von
Thümmerer Philipp, BSc

Dornbirn, 02.08.2023

Kurzreferat

Prognose der Netzabgabemenge im Stromnetz der Vorarlberger Energienetze GmbH

Die Anforderungen an gute Prognosen im Energiehandel der illwerke vkw AG steigen kontinuierlich an, da diese dazu verwendet werden, die richtigen Energiemengen auf dem Day-Ahead-Markt zu kaufen oder zu verkaufen. Zusätzliche Herausforderungen wie Wittereinflüsse und der steigende Energiebedarf von E-Autos erschweren die Prognose. Darüber hinaus gestaltet es sich mit den herkömmlichen Prognosebewertungen schwierig, die Kostenaspekte angemessen abzubilden. Aus diesem Grund wird in dieser Masterarbeit eine neue Methode zur Bewertung der Prognosequalität implementiert und mit bewährten Kennzahlen wie R^2 , MAE, MAPE und RMSE verglichen. Diese Methode nutzt einen Preisverlauf, der die durchschnittlichen Marktpreise der Ausgleichsenergiepreise als Bewertungsmaßstab widerspiegelt. Die neue Bewertungsmethode wird mit Algorithmen wie dem Artificial Neural Network von Tensorflow, dem Decision Tree Regressor, der Linearen Regression, dem Multi-Layer Perception und dem Random Forest Regressor von Scikit-Learn verglichen. Zudem wird ein eigens erstelltes Gurobi-Modell, welches die Ausgleichsenergiepreise für die Gewichtung der absoluten Prognosefehler heranzieht, als Lineare Regression implementiert. Die genannten Modelle werden mithilfe von Python implementiert. Es werden tägliche Vorhersagen basierend auf viertelstündlichen Daten für Zeiträume von bis zu einem Jahr unter Verwendung eines Sliding-Window-Verfahrens mit unterschiedlichen Trainingsdaten erstellt. Die neue Bewertungsmethode mit der Einbeziehung von Kostenfaktoren, konnte sich im Vergleich zu MAPE, MAE, RMSE und R^2 nicht deutlich von den anderen unterscheiden. Die besten Ergebnisse wurden mit der Linearen Regression von Scikit-Learn und dem Gurobi-Modell mit der preisgewichteten Minimierung erzielt. Dabei sind die anderen Algorithmen nicht weit von den Resultaten der LR und GP entfernt. Unter dem Aspekt, dass das eigene Modell die Kosten minimieren soll, konnte es aber nicht als die beste Methode bewertet werden. Deswegen müssen das Modell und die gewählten Ausgleichsenergiepreise optimiert werden. Hierzu erzielte unter den sieben verwendeten Methoden die Lineare Regression das beste Ergebnis.

Schlagwörter: Load Forecasting, Artificial Neural Network, Decision Tree, Multi-Layer Perception, Linear Regression, Random Forest, Support Vector Regression, Sliding-Window

Abstract

Forecast of the net energy volume in the electricity grid of Vorarlberger Energienetze GmbH

The requirements for good forecasts in energy trading at illwerke vkw AG are continuously increasing, as these are used to buy or sell the right amounts of energy on the day-ahead market. Additional challenges such as weather influences and the increasing energy demand of e-cars make forecasting more difficult. In addition, traditional forecasting evaluations are proving difficult to adequately represent the cost aspects. For this reason, this master thesis implements a new method for evaluating forecast quality and compares it with proven metrics such as R^2 , MAE, MAPE, and RMSE. This method uses a price history that reflects the average market prices of the balancing energy prices as an evaluation metric. The new valuation method is compared with algorithms such as Tensorflow's Artificial Neural Network, Decision Tree Regressor, Linear Regression, Multi-Layer Perception, and Random Forest Regressor from Scikit-Learn. In addition, a custom-built Gurobi model that uses balancing energy prices to weight absolute forecast errors is implemented as a Linear Regression. The above models are implemented using Python. Daily forecasts based on quarter-hourly data for periods up to one year are produced using a sliding window method with different training data. The new evaluation method with the inclusion of cost factors, could not clearly distinguish itself from the others when compared to MAPE, MAE, RMSE and R^2 . The best results were obtained with the Scikit-Learn linear regression and the Gurobi model with the price-weighted minimization. Thereby, the other algorithms are not far from the LR and GP results. However, from the point of view that the own model should minimize the costs, it could not score as the best method. As a result, the model and the selected balancing energy prices have to be optimized. For this purpose, Linear Regression achieved the best result among the seven methods used.

Keywords: Load Forecasting, Artificial Neural Network, Decision Tree, Multi-Layer Perception, Random Forest, Linear Regression, Support Vector Regression, Sliding-Window

Vorwort

In dieser Masterarbeit wird das Thema Energieprognosen und Kennzahlen zur Prognosegüte behandelt, das folgende Fragestellung beinhaltet: Kann eine Prognose, deren Güte mit Ausgleichsenergiepreisen bewertet wird, durch die Verwendung der Ausgleichsenergiepreise im Vergleich zu Standardmethoden in ihrer Güte verbessert werden?

Durch die Tätigkeit in meinem Arbeitsumfeld bei der Vorarlberger Energietze GmbH hatte ich die Möglichkeit, meinem persönlichem Interesse an der Erarbeitung dieses Themas nachzugehen. Es hat mir einen direkteren Kontakt zum Energiehandel der illwerke vkw AG ermöglicht, was von Extern in dieser Form sicher nicht möglich gewesen wäre. Diesen Vorteil habe ich genutzt um in engem Kontakt mit dem Handel meine Vorgehensweise zu besprechen, weswegen ich mich an dieser Stelle auch für die gute Zusammenarbeit bedanken möchte.

Die größte Herausforderung bei der Erstellung dieser Arbeit war das Simulieren der Methoden mit realen Messdaten und die anschließende Beurteilung durch die neue Bewertungsmethode. Hierzu gilt mein persönlicher Dank meinem Betreuer Herrn Dr. Klaus Rheinberger, der mir im Verlauf dieser Arbeit eine große Hilfe war und mich tatkräftig unterstützte. Auch der Vorarlberger Energienetze GmbH möchte ich danken, die mir die für diese Arbeit notwendigen Energiedaten zur Verfügung gestellt haben.

Viel Spaß beim Lesen meiner Masterarbeit.

Thümmerer Philipp

Dornbirn, 02.08.2023

Inhaltsverzeichnis

Abbildungsverzeichnis	8
Tabellenverzeichnis	10
Abkürzungsverzeichnis	11
1 Einleitung	12
1.1 Literatur und Qualitätsgrößen	13
1.2 Zielsetzung	15
2 Aufbau und Methoden	17
2.1 Daten	17
2.1.1 Ausgleichsenergiepreise der APG	17
2.1.2 Energiedaten der Vorarlberger Energienetze GmbH	19
2.1.3 Wetterdaten	21
2.2 Prognosen der illwerke vkw AG	24
2.3 Verwendete Methoden	25
2.3.1 ANN - Artificial Neural Network und MLP - Multi-Layer Perception	26
2.3.2 DT - Decision Tree Regressor	28
2.3.3 RF - Random Forest Regressor	29
2.3.4 SVR - Support Vector Regressor	30
2.3.5 LR - Linear Regression	31
2.3.6 Gurobi - LP	32
3 Anwendung der Methoden	35
3.1 Gewichtung und Bewertung der absoluten Prognosefehler	40
3.2 Ergebnisse ANN und MLP	42
3.3 Ergebnisse DT	44
3.4 Ergebnisse RF	47
3.5 Ergebnisse SVR	48
3.6 Ergebnisse LR	50
3.7 Ergebnisse Gurobi	51
4 Ergebnisbewertung und -vergleich	53
4.1 Ergebnisinterpretation	60

5	Fazit und Diskussion	64
	Literaturverzeichnis	66
	Anhang	68
	Eidesstattliche Erklärung	75

Abbildungsverzeichnis

Abbildung 1: Methodenaufbau	17
Abbildung 2: Ausgleichsenergiepreise zwischen 01.01.2022 und 01.04.2023.....	18
Abbildung 3: Viertelstundendaten Netzabgabemenge	19
Abbildung 4: Gesamte Netzabgabemenge berechnete Summe	20
Abbildung 5: Mittlere Netzabgabemenge je Wochentag.....	21
Abbildung 6: Karte mit Messpunkten, Wetterstationen und Umspannwerke.....	22
Abbildung 7: Scatter-Matrix Energieverbrauch und a) Temperatur, b) Niederschlagsmenge, und c) Sonnenscheindauer und d) Windgeschwindigkeit	23
Abbildung 8: Meist genutzte Algorithmen für Vorhersagemodelle; Quelle: (Nti u.a. 2020, S. 12).....	25
Abbildung 9: Top 7, meist genutzte Algorithmen für Vorhersagemodelle; Quelle: (Rostum u.a. 2020, S. 855)	25
Abbildung 10: Aufbau Neuronales Netzwerk; Quelle: (Haben; Voss; Holderbaum 2023, S. 176)	27
Abbildung 11: Beispiel Auswirkung Sample-Weights bei DT.....	29
Abbildung 12: Beispiel Auswirkung der Max. Tiefe bei a) DT und b) RF	36
Abbildung 13: Beispiel Auswirkung maximale Iterationen bei MLP	37
Abbildung 14: Beispiel Auswirkung C bei SVR	38
Abbildung 15: Glättung und Vergleich Ausgleichsenergiepreise	41
Abbildung 16: Kostendarstellung anhand Prognosefehler.....	42
Abbildung 17: Ergebnisse ANN und MLP mit Vorhersagewerte und Test-Scores	44
Abbildung 18: Boxplot der Grid-Search Ergebnisse DT	45
Abbildung 19: Ergebnis DT mit Vorhersagewerte und Test-Scores	46
Abbildung 20: BEC DT pro Tag mit 8 Wochen Trainingsdaten	46
Abbildung 21: BEC RF pro Tag mit 8 Woche Trainingsdaten	48
Abbildung 22: Grid-Search Support Vector Regression, Parameter C.....	49
Abbildung 23: Ausschnitt Prognosedaten SVR mit 8 Wochen Trainingsdaten	49
Abbildung 24: Prognosefehler LR	50
Abbildung 25: Prognosefehler Vergleich GP und LR.....	51
Abbildung 26: BEC GP pro Tag mit 8 Woche Trainingsdaten	52

Abbildung 27: Prognosefehler je Tageszeit und Trainingstage	53
Abbildung 28: BEC je Tageszeit und Trainingsdaten	54
Abbildung 29: Vergleich Prognosegüte Kennzahlen Gesamt.....	55
Abbildung 30: Saisonaler Vergleich MBEC.....	56
Abbildung 31: Wochentag-Vergleich MBEC	57
Abbildung 32: Vergleich Gewichtung durch BEP.....	58
Abbildung 33: Vergleich Gurobi-Modell mit Linearer Regression	59
Abbildung 34: Prognosefehler - Bester und schlechtester Fall.....	59
Abbildung 35: BEC Jahresverlauf Methodenvergleich 8 Wochen Trainingsdaten	60

Tabellenverzeichnis

Tabelle 1: Kennzahlen Ausgleichsenergiepreise	19
Tabelle 2: Liste Wetterstationen mit Koordinaten	21
Tabelle 3: Korrelation Saisonzeiten zur Netzabgabemenge.....	23
Tabelle 4: Beispieeltabelle - Ausschnitt Features aus X-Matrix.....	35
Tabelle 5: Iterationsprozess der Prognose mit 7 Trainingstagen	38
Tabelle 6: Savitzky-Golay-Filter Parameter	40
Tabelle 7: Ergebnisse ANN und MLP, BEC und MBEC	43
Tabelle 8: Grid-Search DT, Top 10 Parameter-Häufigkeit.....	45
Tabelle 9: Ergebnisse DT, BEC und MBEC.....	47
Tabelle 10: Grid-Search RF, Top 10 Parameter-Häufigkeit.....	47
Tabelle 11: Ergebnisse RF, BEC und MBEC.....	48
Tabelle 12: Grid-Search SVR, Top 8 Parameter-Häufigkeit	48
Tabelle 13: Ergebnisse SVR, BEC und MBEC	49
Tabelle 14: Ergebnisse LR, BEC und MBEC.....	50
Tabelle 15: Ergebnisse GP BEC und MBEC	51
Tabelle 16: BEC prozentuale Ergebnisunterschiede bei 8 Wochen Trainingsmenge	61
Tabelle 17: Gesamtergebnisse der unterschiedlichen Trainingsmengen.....	68
Tabelle 18: Gesamtergebnisse MAE Saison der unterschiedlichen Trainingsmengen.....	69
Tabelle 19: Gesamtergebnisse MAPE Saison der unterschiedlichen Trainingsmengen	69
Tabelle 20: Gesamtergebnisse RMSE Saison der unterschiedlichen Trainingsmengen	70
Tabelle 21: Gesamtergebnisse MBEC Saison der unterschiedlichen Trainingsmengen	70
Tabelle 22: Gesamtergebnisse MAE Wochentag der unterschiedlichen Trainingsmengen	71
Tabelle 23: Gesamtergebnisse MAPE Wochentag der unterschiedlichen Trainingsmengen	72
Tabelle 24: Gesamtergebnisse RMSE Wochentag der unterschiedlichen Trainingsmengen	73
Tabelle 25: Gesamtergebnisse MBEC Wochentag der unterschiedlichen Trainingsmengen	74

Abkürzungsverzeichnis

ANN	Artificial Neural Network
API	Application Programming Interface
APG	Austrian Power Grid
ARIMA	Auto-Regressive Integrated Moving Average
BEP	Balancing Energy Prices
BEC	Balancing Energy Costs
CV	Cross-Validation
DL	Deep Learning
DT	Decision Tree
ES	Exponential Smoothing
GP	Gurobi
GWh	Gigawattstunde
kWh	Kilowattstunde
LR	Linear Regression
MAPE	Mean Absolute Percentage Error
MBEC	Mean Balancing Energy Costs
MLP	Multi-Layer Perception
MWh	Megawattstunde
MSE	Mean Squared Error
RF	Random Forest
MLP	Multi-Layer Perception
RMSE	Root Mean Squared Error
SVM	Support Vector Machine
SVR	Support Vector Regression
€	Euro

1 Einleitung

Mit dem stetigen Ausbau von erneuerbaren Energien, E-Mobilität und Klein-Energieerzeugern, wird das Prognostizieren von Energiemengen in einem Stromnetz immer herausfordernder. Und in Kombination mit den sich immer stärker verändernden Wetterverhältnissen, nicht zuletzt aufgrund des stetigen Klimawandels, werden historische Vergleiche schwerer. Diese Umstände beschäftigen den Energiehandel der illwerke vkw AG, der aktuell nach neuen Methoden und Möglichkeiten sucht, bessere Prognosen zu erstellen. Hierzu sind Vorhersagen der Netzabgabemenge, also wieviel Energie an die Endkunden geliefert werden muss, von hoher Wichtigkeit und je genauer diese sind, umso mehr Vorteile bieten sich auf dem Energiemarkt. Durch mein persönliches Interesse an diesem Thema widme ich mich dieser Problemstellung und versuche, in dieser Arbeit bestehende Methoden aufzuzeigen. Diese sollen sich vorallem durch eine andere Vorgehensweise unterscheiden, welche die Ausgleichsenergiepreise der Austrian Power Grid als Bewertungsmethode der absoluten Prognosefehler heranzieht.

Der Vorteil der sich durch eine solche Bewertung bieten soll besteht darin, dass nicht alle Prognosefehler gleich behandelt werden, sondern sich zu den verschiedenen Tageszeiten unterscheiden. Grund dafür sind die Preise selbst, denn diese sind über den Verlauf des Tages nicht gleichbleibend, sondern ändern sich pro Viertelstunde und je nach Höhe des Preises hat ein Prognosefehler einen größeren oder kleineren Einfluss auf die Ausgleichsenergiekosten.

Für den programmierteil dieser Arbeit wird Python genutzt, um mit Methoden wie Artificial Neural Network (Multi-Layer Perception), Decision Tree Regressor, Linear Regression, Random Forest Regressor, Support Vector Regression und ein Gurobi-Modell Prognosen zu erzielen, sowie Vergleiche zu erstellen. Diese Algorithmen kommen im Bereich der Energie- oder Lastvorhersage in Stromnetzen oft zum Einsatz. Ob Kurz-, Mittel- oder Langzeitprognosen, diese Methoden können vielseitig verwendet werden. Dabei wird in dieser Arbeit mit Kurzzeitprognosen gearbeitet, wobei als Prognoseziel immer ein vollständiger Tag mit 96 Viertelstundenwerten herangezogen wird, was sich auf den Day-Ahead Handel am Strommarkt zurückführen lässt. Denn am Day-Ahead-Markt muss die benötigte oder überschüssige Energiemenge am Vortag zum Verkauf oder Kauf angeboten werden. Dieser Wert/Energiepreis kann dann zwar am Tag des Handels am Intraday-Markt schwanken, was in Kapitel 2.1.1 näher erläutert wird, doch es wird in dieser Arbeit nur der Energiehandel am Vortag behandelt und der Handel am Intraday-Markt wird außer acht gelassen, sodass nur die Ausgleichsenergiepreise schlagend werden.

Die Prognosen werden in diesem Zuge der Berechnungen zusätzlich mit Wetterdaten als Features angereichert, um die Prognosequalität zu verbessern. Temperatur, Sonnenminuten, Wind und Regenmenge stellen zu unterschiedlichen Jahreszeiten eine nicht zu

vernachlässigende Korrelation zu den Netzabgabemengen dar und können das Ergebnis der Prognose verbessern.

Über diese Punkte ergibt sich somit folgende Frage: Kann eine Prognose, deren Güte mit Ausgleichsenergiepreisen bewertet wird, durch die Verwendung der Ausgleichsenergiepreise im Vergleich zu Standardmethoden in ihrer Güte verbessert werden?

1.1 Literatur und Qualitätsgrößen

In diesem Kapitel wird aufgezeigt, welche Ergebnisse in der Literatur mit den zuvor erwähnten Algorithmen erzielt werden konnten. Der Vergleich mit dieser Arbeit ist aber aufgrund unterschiedlicher Datensätze und Parameter mit Vorsicht zu betrachten. Denn wie Anfangs erwähnt soll das Hauptaugenmerk darin liegen, die Bewertung der Methoden durch den Ausgleichsenergiepreis durchzuführen. Dabei werden bewährte Kennzahlen wie R^2 , MAPE, MAE oder RMSE als Vergleich herangezogen. Die folgenden Ausschnitte aus der Literatur sollen ein allgemeines Bild über die verwendeten Methoden im weiteren Verlauf dieser Arbeit geben. Wie erwähnt, wird in den meisten Fällen die Prognosegüte mit Kennzahlen, wie den mittleren absoluten Fehler (MAE), den mittleren absoluten prozentualen Fehler (MAPE) oder die Wurzel des mittleren quadratischen Fehlers (RMSE) definiert. Diese Größen werden in (Rostum u.a. 2020, S. 843) definiert als:

$$MAPE = \frac{\sum_{t=1}^S \left| \frac{A_t - P_t}{A_t} \right|}{S} * 100,$$

$$MAE = \frac{\sum_{t=1}^S |A_t - P_t|}{S},$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^S (A_t - P_t)^2}{S}},$$

dabei sind A_t die Echtdata, P_t prognostizierte Werte und S die Länge des verwendeten Datensatzes. Der R^2 -Wert nimmt ebenfalls eine entscheidende Rolle ein. Denn er wird hauptsächlich bei den in dieser Arbeit verwendeten Algorithmen von Scikit-Learn, wie z.B. LR („Sklearn - LR“ o. J.) oder SVR („Sklearn - SVR“ o. J.), als Kennzahl für die Prognosegüte verwendet und in (Wang u.a. 2021, S. 8) wie folgt definiert:

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (\bar{y}_i - \hat{y}_i)^2},$$

wobei y_i und \hat{y}_i die tatsächlichen bzw. vorhergesagten Werte darstellen und N die Größe des Testdatensatzes ist. \bar{y}_i ist der Mittelwert der tatsächlichen Werte. „Der bestmögliche zu erreichende Wert ist 1. Der R^2 kann auch negativ sein weil das Modell beliebig schlecht sein kann. Ein konstantes Modell, das immer den erwarteten Wert \bar{y}_i von y_i vorhersagt und dabei die Eingangsmerkmale außer Acht lässt, würde immer einen Wert von 0 erhalten.“ („Sklearn - SVR“ o. J.) Im folgenden Literaturüberblick wird zumindest eine der oben erwähnten Kennzahlen als Bewertungsmethode verwendet.

Fan erwähnt, dass sich der SVR „aufgrund der Vorteile der statistischen Lernfähigkeit bei der Verarbeitung hochdimensionaler Daten, besonders für das Lernen mit kleinen Stichproben eignet und zu einem beliebten Algorithmus für viele Prognoseprobleme geworden ist“ (Fan u.a. 2013, S. 1888). Dabei konnte er in seinen Methoden mit der SVR Ergebnisse mit einem MAPE von 11,69% für einen kleinen Datensatz und 12,87% für einen großen Datensatz erzielen. Dabei versuchte er eine Lastprognose für New South Wales (Australien) zu erstellen. In (Fan u.a. 2016, S. 968) erzielte ein ARIMA(4,1,4) Modell mit denselben Daten einen MAPE von 45,33% bzw. 60,65%.

Desai und Patel untersuchten an einem Standard-Datensatz die Algorithmen Lineare Regression, Support Vector Regression, Decision Trees und Random Forest im Vergleich zu einem eigens erstellten Kombimodell, welches DT und LR beinhaltet. Für die in dieser Arbeit verwendeten Methoden erzielten sie in den Prognosen einen MAE bei SVR von 3,02, DT: 4,48 und RF: 3,42. (Desai; Patel o. J., S. 5)

In dem Review-Artikel von Jahan, Snasel und Misak wurden mehrere Artikel mit Lastvorhersage und unterschiedliche Methoden untersucht. Ziel dabei war es, „das Wissen über die Stromnachfrage und frühere Studien über Stromlastprognosesysteme zu überprüfen, die Wissenschaftlern und Experten auf diesem Gebiet helfen können“ (Jahan; Snasel; Misak 2020, S. 8). Beinhaltet sind unter anderem ANN mit einem MAPE von 0,268%, DT und RF mit 1,166% und SVM mit 0,196%.

Agyeman erwähnt, dass der Einsatz von ANN in den Prognoseverfahren sehr wichtig ist, da in der eigenen Fallstudie eine klassische Vorhersagemethode wie SVR nicht gut mit Schwankungen im Datensatz umgehen konnte. „Das ANN ist bereits für seinen dominanten Einsatz in Energieprognoseverfahren bekannt, doch aufgrund seiner inhärenten Komplexität führt es jedoch meist zu Overfitting“ (Agyeman u.a. 2020, S. 17) und konnte mit verschiedensten Datensätzen und Zeiträumen im Schnitt einen MAPE von 15,96% erreichen.

Wang konnte in seiner Studie herausfinden, dass „kurzfristige Lastprognosen (STLF) entscheidend für die tägliche Planung, den wirtschaftlichen und sicheren Betrieb von Stromsystemen sind, die in den letzten Jahrzehnten intensiv untersucht wurden.“ (Wang u.a. 2021, S. 1) Er verwendet mehrere Methoden im Bereich der Mittelzeitprognosen, indem ein reales industrielles Lastprognoseproblem in China abgebildet wird und untersucht dabei auch

SVR und ANN Modelle. Das Resultat des MAE für seine Lastprognose der Monate zwischen Jänner 2019 und Dezember 2022 betragen beim SVR 9,38 TWh (etwa 7%) und bei ANN 8,67 TWh (etwa 6%). Bei der Jahresprognose lagen die Werten von SVR bei 96,79 TWh (etwa 7%). und ANN bei 108,11 TWh (etwa 8%).

Bouktif hingegen verwendete Lastprognosedaten einer französischen Stadt, bei der mit den Algorithmen KNN, Ridge, RF, GB, ANN und Extra Trees ein Forecast von bis zu 4 Monaten in die Zukunft erstellt wurde. Wobei die Performance durch den MAE für RF (DT) bei 370 MWh und ANN bei 2180 MWh lag. Für das schlechte Ergebnis des ANN kommentierten sie, dass „das Ergebnissen von ANN nicht gut abschneidet, was daran liegt, dass ein ANN-Modell eine optimale Netzwerkstruktur und aktualisierte Gewichte haben muss, um eine gute Genauigkeit zu erreichen, was mehrere Epochen und Netzwerkkonfigurationen erfordern kann“ (Bouktif u.a. 2018, S. 10).

1.2 Zielsetzung

Ein Ziel dieser Masterarbeit ist es, die Methoden unter einem neuen Bewertungsschema zu beurteilen. Dazu werden die Ausgleichsenergiepreise, welche in Kapitel 3.1 beschrieben werden, für eine Bewertung der verwendeten Methoden herangezogen. Das Preisschema benennen wir als „Balancing Energy Prices“, kurz BEP, und die Summe der Kosten BEC „Balancing Energy Costs“. Die mittleren Kosten je Viertelstunde werden dabei als MBEC „Mean Balancing Energy Costs“ dargestellt. Für beide gilt folgende neue Definition einer Bewertung:

$$BEC = \sum_{t=1}^S |A_t - P_t| * BEP_t,$$

$$MBEC = \frac{\sum_{t=1}^S |A_t - P_t| * BEP_t}{S},$$

worin die absoluten Prognosefehler $|A_t - P_t|$ in MWh, durch die positiven Ausgleichsenergiepreise BEP_t eine zeitabhängige Fehlergewichtung von €/MWh bekommen. BEC und MBEC stellen die daraus resultierenden Kosten in € dar, welche keine direkt vergleichbare Kosten sind, sondern als Qualitätsparameter gelten.

Ein weiteres Ziel ist es eine neue Prognosemethode, welche die Ausgleichsenergie verwendet, mit den bekannten Methoden anhand der Gütemaße BEC und MBEC zu vergleichen. Die neue Prognosemethode stellt ein Lineares-Regressionsproblem dar, welches mit Gurobi implementiert wird. Das Ziel der Optimierung besteht darin, die Koeffizienten so zu

finden, dass die Vorhersagewerte möglichst nah an den tatsächlichen Werten liegen. Die Abweichung zwischen den Vorhersagen und den tatsächlichen Werten wird durch eine absolute Variable repräsentiert. Da die Lineare Regression verwendet wird, wird die Methode der kleinsten Quadrate angewendet, um die Koeffizienten zu optimieren. Das Ziel ist es, die Summe der quadrierten Abweichungen zwischen den Vorhersagen und den tatsächlichen Werten zu minimieren. Ziel dabei ist es nicht die absoluten Prognosefehler selbst zu minimieren, sondern die durch die BEP gewichteten absoluten Prognosefehler. Dadurch werden die BEC und die sich im Modell befindenden Variablen optimiert. Die genaue Vorgehensweise wird in Kapitel 2.3.6 erläutert.

2 Aufbau und Methoden

Der Aufbau der angewendeten Methoden, kann in einfacher und konzeptioneller Form in Abbildung 1 dargestellt werden. Hierbei sind die ersten Schritte die Beschaffung von Daten in Kapitel 2.1, die dann mit den Methoden in Kapitel 2.3 zu einer Vorhersage der Netzabgabemenge für 96 Viertelstundenwerte führen.

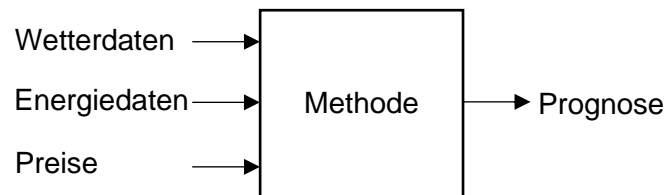


Abbildung 1: Methodenaufbau

Dabei werden Wetterdaten und Ausgleichsenergiepreise aus öffentlichen Quellen und die Energiedaten von der Vorarlberger Energienetze GmbH bezogen. Für die Methoden erfolgte die Programmierung in der Programmiersprache Python, wobei JupiterLab als kostenfreie Benutzeroberfläche genutzt wurde. Die Methoden wurden mit den technischen Ressourcen eines herkömmlichen Notebooks ausgeführt, welches einen Prozessor mit 2 Kernen und 2,6 GHz (Intel(R) Core(TM) i5-7300U) und einen Arbeitsspeicher von 8 GB RAM besaß. Aufgrund dieser Hardware-Konfiguration benötigten einige Methoden-Durchläufe mehrere Stunden Rechenzeit.

2.1 Daten

Eine fundierte Datenlage ist ein wichtiger Teil und zwingend notwendig, um eine plausible Vorhersage durchführen zu können, weil damit die Effizienz und Zuverlässigkeit einer Vorhersage stark verbessert werden kann. Eine gute Effizienz kann in diesem Fall den Teilen zugeschrieben werden, welche aufgrund einer guten Vorhersage zu niedrigeren BEC oder MBEC führen. Die Zuverlässigkeit der Daten spiegelt eine konstant gute Prognose wider. Damit können fortlaufend plausible Ergebnisse erzielt werden.

2.1.1 Ausgleichsenergiepreise der APG

Die Ausgleichsenergiepreise werden als Bewertungsfaktor für die Prognosefehler aller Algorithmen verwendet. Die Verarbeitung dieser und wie sie schlussendlich mit in die Prognosegüte kalkuliert werden, wird in Kapitel 3.1 näher erläutert. Die Berechnung dieser Preise werden von der European Power Exchange (EPEX SPOT), welche für den

zentraleuropäischen Spotmarkt für Energie verantwortlich ist, durchgeführt und von der Austrian Power Grid für österreichische Energieversorgungsunternehmen zur Verfügung gestellt. „Der Ausgleichsenergiepreis berechnet sich für negative Bilanzkreisabweichungen als das Maximum aus dem gewichteten Durchschnittspreis positiver aktivierter Regelarbeit, dem Börsenpreisindikator und dem Knappheitselement. Für positive Bilanzkreisabweichungen als das Minimum aus dem gewichteten Durchschnittspreis negativer aktivierter Regelarbeit dem Börsenpreisindikator und dem Knappheitselement.“ („Ausgleichsenergiepreise | APG“ o. J.)

Doch wie wirken sich die Ausgleichsenergiepreise bei einem Energieversorgungsunternehmen aus? In einem Interview mit dem Energiehandel der illwerke vkw AG wurde ermittelt, dass in der Börsenauktion (Day-Ahead-Markt) am Vortag die Energie für den Folgetag gehandelt wird, welche laut Prognose im Energiehandel zu viel oder zu wenig in der Energiebilanz erscheint. Hierbei kann sich der Preis bis zum tatsächlichen Handel aber stündlich ändern. Jegliche Abweichung, die vor dem Zeitpunkt des Handels aufgrund von unerwarteten Energieflüssen noch offen ist, kann durch Mitarbeiter im Handel am Intraday-Markt nachgehandelt werden: also Energie einkaufen oder verkaufen. Dabei können auch nach dem Handel am Intraday-Markt noch Abweichungen in der Bilanz vorhanden sein, welche dann mit dem Ausgleichsenergiepreis verrechnet werden und hierzu können je nach Position auf der Deltaregelzone Kosten oder Einnahmen für den Energielieferanten entstehen. Wird keine Ausgleichsenergie benötigt, so müssen auch keine Ausgleichsenergiepreise von der APG verrechnet werden. Die Position in der Deltaregelzone bestimmt in diesem Fall nur, ob gerade verkauft oder eingekauft werden muss, demnach ob es einen Überschuss oder Mangel an Energie in der eigenen Bilanz gibt.

Für die in Abbildung 2 dargestellten Ausgleichsenergiepreise, für den Zeitraum der Datenverfügbarkeit zwischen 01.01.2022 und 01.04.2023, ergeben sich folgende Kennzahlen in Tabelle 1.

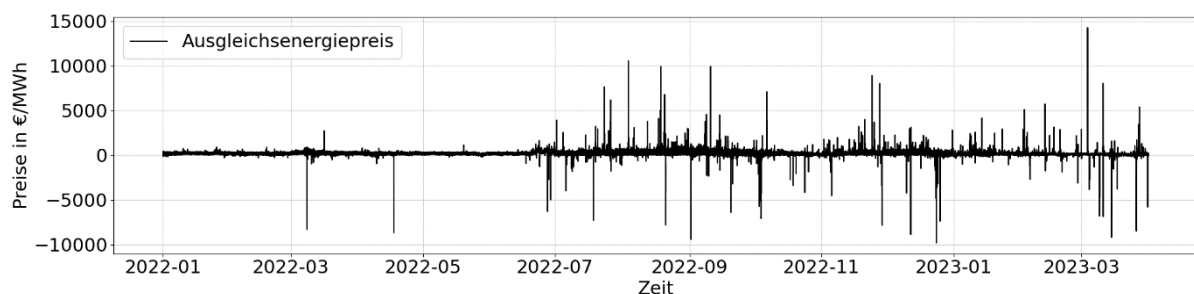


Abbildung 2: Ausgleichsenergiepreise zwischen 01.01.2022 und 01.04.2023

Minimum in €/MWh	Maximum in €/MWh	Median in €/MWh	Mittelwert in €/MWh
-9820.92	14244.38	210.93	170.52

Tabelle 1: Kennzahlen Ausgleichsenergiepreise

Es ist zu erkennen, dass kurzzeitig stark positive und stark negative Preise gebildet werden, hingegen der Median oder Mittelwert über die gesamte Datenbasis deutlich niedriger und positiv ist. Je nach Preislage auf dem Energiemarkt, sind die Ausgleichsenergiepreise stark volatil oder nicht. Zu sehen ist dies ab Juni 2022, wo die Energiepreise stark gestiegen und bis zum Ende der hier abgebildeten Datenreihe hoch geblieben sind.

Die Ausgleichsenergiepreise unterscheiden sich im Vergleich zu den Day-Ahead- und Intraday-Preisen im Tagesverlauf kaum. In Abbildung 15 ist dargestellt, wie die Preise an unterschiedlichen Tagen denselben Tagesverlauf aufweisen. Daraus lässt sich ableiten, dass die Preise an unterschiedlichen Tagen zu den selben Tageszeiten korrelieren und ähnlich steigen oder fallen. Ob in diesem Fall die Ausgleichsenergie-, Day-Ahead- oder Intraday-Preise verwendet werden, macht keinen großen Unterschied.

2.1.2 Energiedaten der Vorarlberger Energienetze GmbH

Für die Prognosen werden historische Messdaten des Energienetzbetreibers Vorarlberger Energienetze GmbH verwendet. Dieser ist als Tochterunternehmen der illwerke vkw AG für die Aufrechterhaltung des Netzbetriebs bzw. für den Ausbau der Netzinfrastruktur verantwortlich und erhebt bzw. speichert die Messdaten in dessen Backend-Systemen, aus welchen die Daten für diese Arbeit zur Verfügung gestellt wurden.

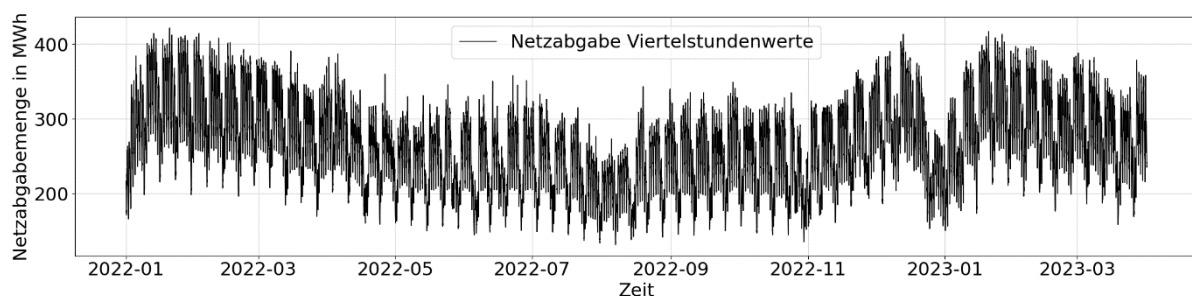


Abbildung 3: Viertelstundendaten Netzabgabemenge

Als Datengrundlage für die Energiemengen im Stromnetz wurde die berechnete Summe der Netzabgabemenge in kWh bzw. die Menge der einzelnen Umspannwerke zur Verfügung gestellt, wobei diese für eine einfachere grafische Darstellung in den Abbildungen, auf die Einheit MWh skaliert wurde. Zu der Summe der einzelnen Umspannwerke, welche die

Netzabgabemenge im Mittelspannungsnetz abbilden, kommen noch die berechneten Netzverluste und einzelne Einspeisemengen im Hochspannungsnetz dazu. Mit diesen berücksichtigten Faktoren werden die zur Verfügung gestellten Daten, welche vom 01.01.2022 bis zum 01.04.2023 reichen, in einer Granularität von Viertelstundenwerten in Abbildung 3 visualisiert.

Der abgebildete Jahresverlauf stellt eine höhere Netzabgabemenge in den kälteren Monaten aufgrund des erhöhten Wärmebedarfs dar. Der Einbruch des Energieverbrauchs Ende Dezember bzw. Anfang Jänner kann auf die Weihnachts- und Neujahresfeiertage zurückgeführt werden. In dieser Zeit sind viele Unternehmen geschlossen, was auch die niedrige Abgabe widerspiegelt. Die noch niedrigere Menge zwischen Juli und September ist dabei den Sommerferien zuzuschreiben. In Abbildung 4 ist der Jahresverlauf der Monatsmengen in GWh abgebildet, dabei wurden die Viertelstundenwerte der einzelnen Monate summiert. Hierbei sind die Mengen vom Januar, Februar und März des Jahres 2023 deutlich geringer als die des Vorjahres.

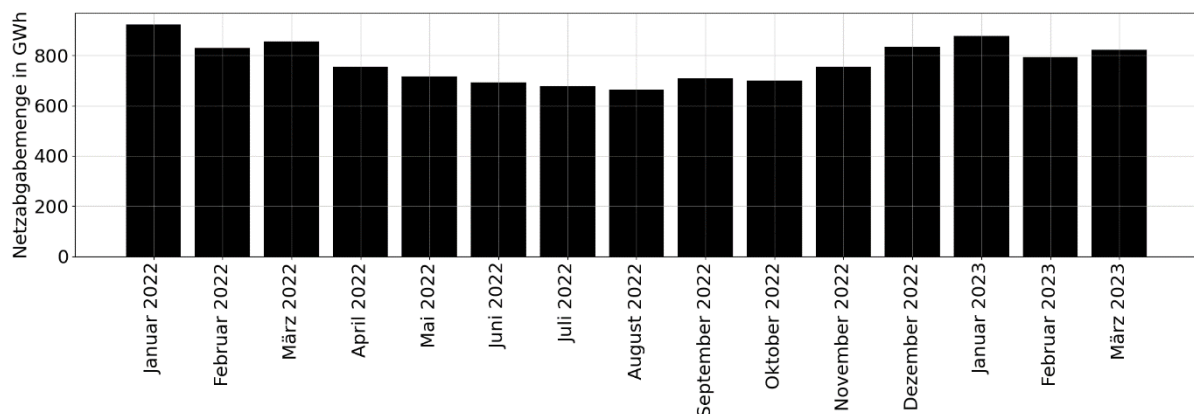


Abbildung 4: Gesamte Netzabgabemenge berechnete Summe

In Abbildung 5 ist visualisiert, wie sich die Netzabgabemenge zwischen den Wochentagen unterscheidet. Von Montag bis Freitag gibt es keine großen Unterschiede im Vergleich zu den Wochenendtagen. Der Sonntag weist den niedrigsten Durchschnittsverbrauch auf. Da dieser ein überwiegend arbeitsfreier Tag ist (geschlossene Betriebe), wird an diesem auch am wenigsten Energie benötigt. Gefolgt von Samstag, der eine Spitze zwischen 15:00Uhr und 16:00Uhr aufweist, was auf den Start der Warmwasser-Boileraufladung im Stromnetz zurückzuführen ist.

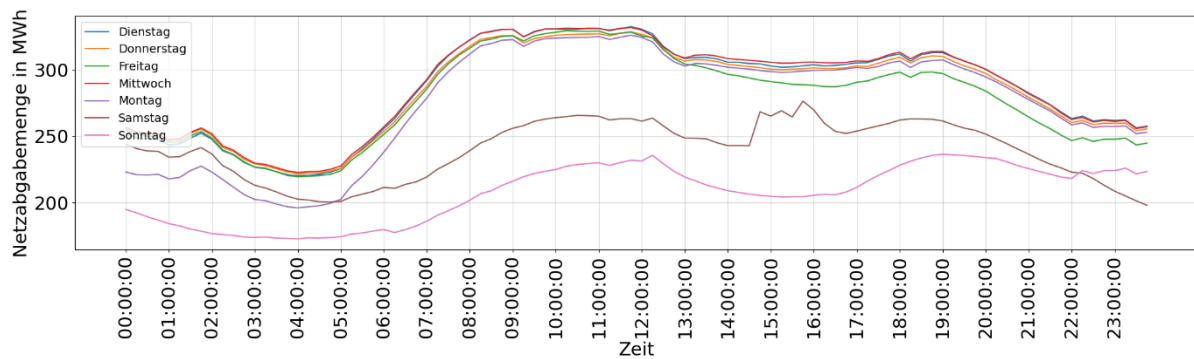


Abbildung 5: Mittlere Netzabgabemenge je Wochentag

2.1.3 Wetterdaten

Der Energieverbrauch steht oft in Zusammenhang mit dem Wetter, weshalb für Prognosen in diesen Bereichen mit Wetterdaten gearbeitet wird, um damit die Genauigkeit einer Prognose zu erhöhen. Die Wetterdaten sind von der Open-Source-Quelle Meteostat („Python Library | Meteostat Developers“ o. J.) mittels API, welche in Python über eine URL implementiert wird, abgerufen worden. Für die Erhebung der Daten sind 36 Messpunkte mit Breitengrad und Längengrad im Raum Vorarlberg erstellt worden und diese dienen als verschiedene Referenzpunkte, mit denen ebenfalls ein besseres Prognoseergebnis erzielt werden sollte. Von jedem dieser Messpunkte wurden Messdaten wie Temperatur in C°, Sonnenminuten je Stunde, Durchschnittliche Niederschlagsmenge in Millimeter und Windgeschwindigkeit in km/h in einer Granularität von Stundenwerten abgerufen. Diese Messpunkte beziehen die Wetterdaten nicht aus der direkten Umgebung, sondern von den nächstgelegenen Wetterstationen, die in der Tabelle 2 aufgelistet sind.

Wetterstation	Breitengrad	Längengrad
Balderschwang	47.4650	10.1043
Bregenz	47.5000	9.7333
Feldkirch	47.2667	9.6000
Galtuer / Tschafein	46.9678	10.1856
Galzig	47.1333	10.2333
Hohenems / Dornbirn Flugfeld	47.3833	9.7000
Oberreute	47.5538	9.9407
Sigmarszell-Zeisertsweiler	47.5776	9.7404
Vaduz	47.1333	9.5167
Warth	47.2500	10.1833
Weissfluhjoch / Schatzalp	46.8333	9.8064

Tabelle 2: Liste Wetterstationen mit Koordinaten

Nach der Erstellung der Messpunkte, wurden im nächsten Schritt über die API die Wetterdaten einmalig abgerufen und gespeichert, um diese nicht immer wieder neu beziehen zu müssen. Damit konnte die Durchlaufzeit bei der Auswahl von unterschiedlichen Zeiträumen verkürzt werden. Die Laufzeit des einmaligen Bezugs von der API betrug in etwa drei Tage, die dann für 36 Messpunkte die Wetterdaten im Zeitraum von 01.01.2022 bis 01.04.2023 lieferten. Da Meteostat und die von den Wetterstationen bezogenen Daten keine Viertelstundenwerte zur Verfügung stellen, wurden diese zwischen den Stundenwerten interpoliert, um sie für die Berechnungen verwenden zu können. Zudem muss beachtet werden, dass die Wetterdaten keine historischen Prognosedaten, sondern reale Messdaten der Wetterstationen sind. Das bedeutet, dass in dieser Arbeit nicht eruiert wird, wie die Berechnung der Netzabgabemenge im Zusammenhang zu Wetter-Prognosewerten steht.

In Abbildung 6 ist das Bundesland Vorarlberg abgebildet, in welchem der Sitz der illwerke vkw AG und des Netzbetreibers Vorarlberger Energienetze GmbH liegt. Darin sind die 36 Messpunkte, die Umspannwerke und die 11 Wetterstationen zu sehen. Mit den grünen Linien werden die Verbindungen zwischen den Messpunkten und Wetterstationen angezeigt:

- Wetterstationen für Messpunkte
- Umspannwerke der Vorarlberger Energienetze GmbH
- Messpunkte für Wetterdaten (Gitter)

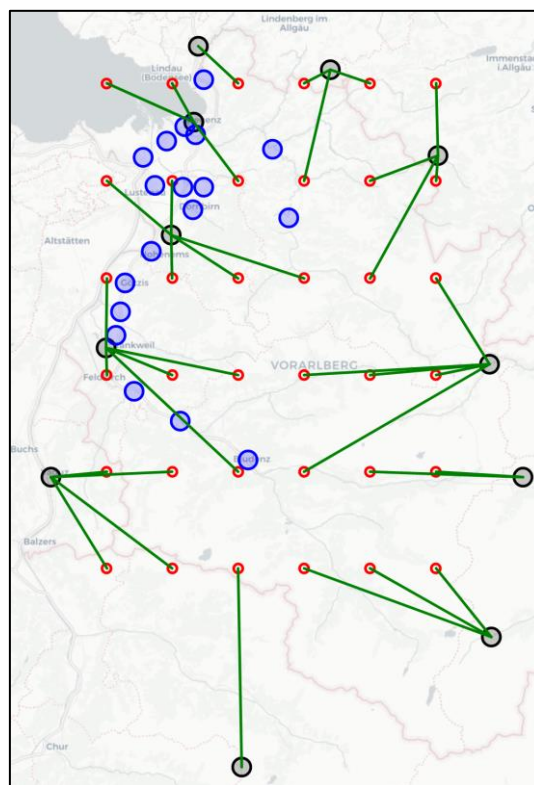


Abbildung 6: Karte mit Messpunkten, Wetterstationen und Umspannwerke

Um den Zusammenhang zwischen Wetterdaten und der Netzabgabemenge darzustellen wird eine Scatter-Matrix erstellt. Wie in (Bouktif u.a. 2018, S. 9) werden Wetterdaten wie z.B. Temperatur, Luftfeuchtigkeit und Wind in Korrelation zu dem Energieverbrauch gestellt. Dabei konnte festgestellt werden, dass die Temperatur wie in Abbildung 7 eine negative Korrelation aufweist. D.h. je niedriger die Temperatur ist, desto höher ist der Energiebedarf.

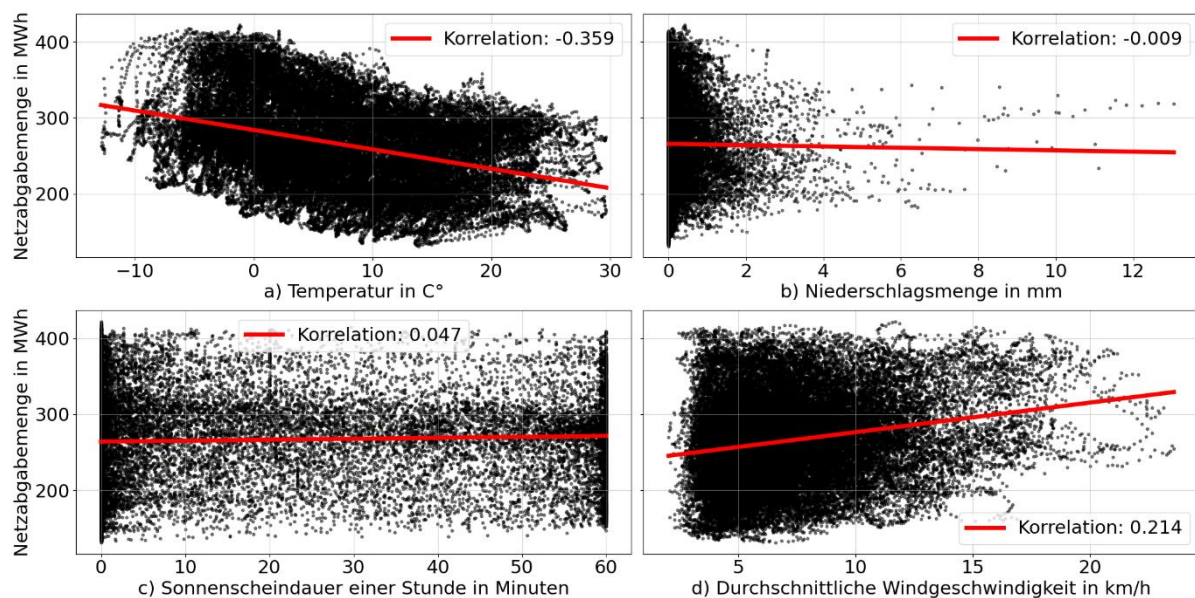


Abbildung 7: Scatter-Matrix Energieverbrauch und a) Temperatur, b) Niederschlagsmenge, und c) Sonnenscheindauer und d) Windgeschwindigkeit

Relativ gesehen zu den Gesamtdaten haben im Verhältnis zur Temperatur die Niederschlagsmenge, die Windgeschwindigkeit und die Sonnenscheindauer über den gesamten Zeitraum eine vergleichsweise geringe Korrelation in Bezug zum Energieverbrauch. Dabei darf nicht vernachlässigt werden, dass sich diese Korrelationen innerhalb der einzelnen vier Jahreszeiten verändern können, was in Tabelle 3 dargestellt ist. Dabei gelten für den Winter die Monate Dezember bis Februar, für den Frühling die Monate März bis Mai, für den Sommer die Monate Juni bis August und für den Herbst die Monate September bis November. Zusätzlich gelten Frühling und Herbst als Übergangszeiten und Winter und Sommer als Hochsaisonzeiten.

Saison	Temperatur	Regen	Sonne	Wind
Frühling	0,253	0,151	0,201	0,326
Sommer	0,027	0,027	0,190	0,089
Herbst	-0,240	-0,106	0,123	0,091
Winter	-0,089	-0,027	0,215	0,121

Tabelle 3: Korrelation Saisonzeiten zur Netzabgabemenge

In den Übergangszeiten sind die Korrelationen stärker vorhanden als in den Hochsaisonzeiten. Somit sollte eine Vorhersage mit diesen Features zu diesen Zeiten besser möglich sein, da auf Basis der Daten bessere Rückschlüsse auf die Netzabgabemenge gezogen werden können.

2.2 Prognosen der illwerke vkw AG

Die illwerke vkw AG hat für ihre aktuellen Prognosemethoden keine Referenzdaten zur Verfügung gestellt. Aufgrund der Tatsache, dass der Energiehandel und die dahinterstehenden Methoden sich in einem kritischen Geschäftsumfeld befinden, werden diese Daten aus Informationsicherheitsgründen nicht ausgegeben. Somit gibt es zwar keine Vergleichsdaten, doch es konnten größtenteils die Interessen des Energiehandels in Interviews abgefragt und die in dieser Masterarbeit verwendeten Methoden mit dem Energiehandel besprochen werden. In mehreren Abstimmungsrunden wurde bestätigt, dass zum einen die Bewertungsmethode durch die Ausgleichsenergiepreise passend ist und zum anderen der zu berechnende Zeitraum eines Tages, also die 96 Viertelstundenwerte, für einen Forecast Sinn macht.

In Bezug auf die Wetterdaten konnte festgestellt werden, dass im Vergleich zu dieser Arbeit, die illwerke vkw AG für ihre Prognose meist nur vergleichbare Tage heranzieht. Somit sprechen sie von Tagen, die z.B. dieselben Außenverhältnisse aufweisen, die dann als Referenz dienen. Auch die menschliche Erfahrung spielt im Energieeinkauf eine Rolle. Z.B. wird mehr Energie eingeplant, wenn in den Schigebieten die Schneekanonen eingesetzt werden. Das wird mit den Energiebedarfsträgern nicht abgesprochen. So kann es auch zu Missinterpretationen kommen und am Day-Ahead-Markt zu viel Energie für den Folgetag eingekauft werden, welche dann am Intraday-Markt wieder verkauft werden muss. Diese Punkte reichen aus, um einen Vergleich der Ergebnisse anhand von möglichen Referenzdaten der illwerke vkw AG und den in dieser Arbeit verwendeten Methoden nicht sinnvoll erscheinen zu lassen. Denn die verwendeten Methoden werden diese Faktoren des Energieeinkauf nie berücksichtigen oder abbilden können.

Der Energiehandel der illwerke vkw AG befürwortet die Untersuchungen der Bewertungs- und Optimierungsmethoden mittels Ausgleichsenergiepreise, da der Energiehandel selbst auch an anderen Methoden zur Prognose arbeitet. Jedoch wird diese Masterarbeit nicht im Auftrag des Energiehandels erstellt, sondern beruht vollständig auf dem Eigeninteresse des Autors an diesem Thema.

2.3 Verwendete Methoden

Für Prognoseverfahren gibt es eine Vielzahl an frei verfügbaren Algorithmen im Internet. In dieser Arbeit werden unter anderem die in „Electricity load forecasting: a systematic review (Nti u.a. 2020, S. 12)“ meist verwendeten Algorithmen verwendet. Dabei wurden 67 Studien zu Prognosemethoden von Energiedaten verglichen. Die in dieser Arbeit verwendeten Standardmethoden sind ANN, DT, LR, MLP, RF und SVR.

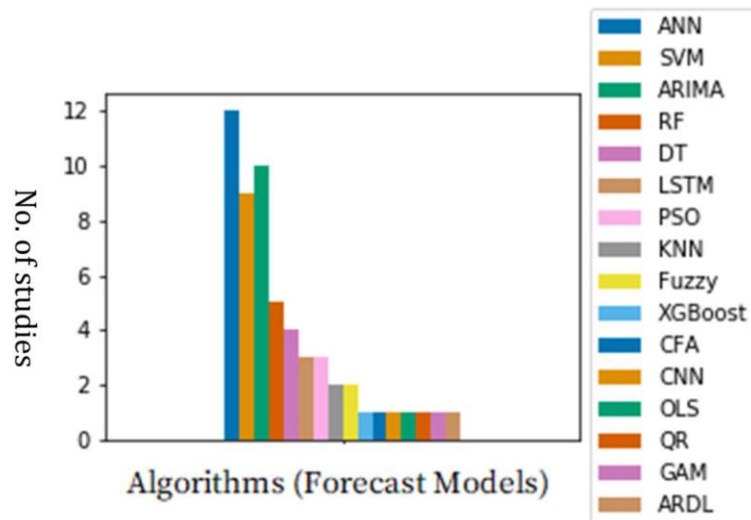


Abbildung 8: Meist genutzte Algorithmen für Vorhersagemodelle; Quelle: (Nti u.a. 2020, S. 12)

Rostum hatte in seinem Review-Artikel ebenfalls herausgefunden, dass „mehr als 42 % der Arbeiten ANN verwendeten. Die am zweithäufigsten verwendete Modellierungstechnik ist SVM mit 20,55 %. Die restlichen Arbeiten verteilen sich auf die anderen Methoden wie DL, ARIMA, Fuzzy, ES und LR“ (Rostum u.a. 2020, S. 855). Hierbei wurden insgesamt 70 Arbeiten in den Artikel miteinbezogen und verglichen und auch hier waren in den Top 3 die Algorithmen ANN und SVR (SVM).

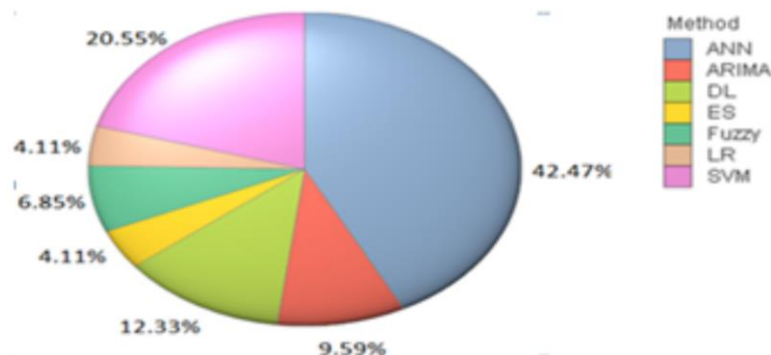


Abbildung 9: Top 7, meist genutzte Algorithmen für Vorhersagemodelle; Quelle: (Rostum u.a. 2020, S. 855)

Grund für die Verwendung der genannten Methoden in dieser Arbeit ist die einfache Implementierung in Python und die Tatsache, dass sich diese in der Vergangenheit lt. Literatur bewährt haben. Die fertigen Algorithmen ermöglichen nach einer erfolgreichen Datenvorverarbeitung einen schnellen Start, solange die zu setzenden Parameter klar sind. Zudem macht es Sinn, auf bewährte Methoden zu setzen, die auch in der Literatur gute Ergebnisse erzielen konnten.

2.3.1 ANN - Artificial Neural Network und MLP - Multi-Layer Perception

Künstliche Neuronale Netzwerke werden in den Vorhersageverfahren laut (Nti u.a. 2020, S. 12) am häufigsten verwendet. Auch in der eigenen Literaturrecherche wurden in mehr als 15 von etwa 40 Quellen ANNs angewandt und sie bestätigt somit die Wichtigkeit dieser Methode. „ANNs bestehen aus einer Ansammlung miteinander verbundener künstlicher Neuronen, und wie die Synapsen im Gehirn kann jedes künstliche Neuron über Dendriten ein Signal an benachbarte, verbundene Neuronen senden. Wie in biologischen neuronalen Netzen wird das "Lernen" durch die Anpassung der Verbindung zwischen den Neuronen erreicht.“ (Haben; Voss; Holderbaum 2023, S. 171) In diesem wird auch die mathematische Definition für ein vollständiges Neuronales Netzwerk wie folgt ausgedrückt:

$$\hat{L}_{N+j} = g_0 \left(\sum_{i=1}^l \beta_{i,j}^{(2)} h_i(X) \right)$$

und für jeden „Hidden Layer“:

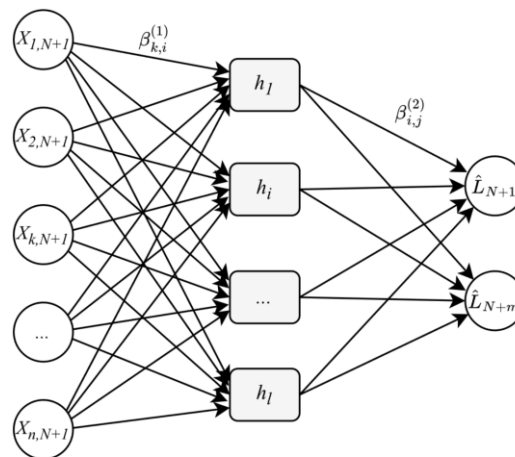
$$h_i(X) = g \left(\sum_{k=1}^n \beta_{k,i}^{(1)} X_{k,N+1} \right).$$

Für die oberen zwei Definitionen gilt, $\beta_{k,i}^{(1)}$ bildet die Verbindung zwischen Eingabe-Layer $X_{k,N+1}$ und Verstecktem-Layer h_i und $\beta_{i,j}^{(2)}$ die Funktion zwischen Verstecktem-Layer h_i und Ausgangs-Layer \hat{L}_{N+j} . g_0 ist in diesem Fall die Aktivierungsfunktion, welche vor der Ausführung definiert werden muss. N stellt dabei immer den Zeitschritt der Testdaten und l stellt die Anzahl an verwendeten Knoten im versteckten Layer dar, was in Abbildung 10 zu sehen ist.

Hierbei gibt es mehrere Auswahlmöglichkeiten aber in dieser Arbeit wird die ReLU-Funktion verwendet. „Die gleichgerichtete lineare Einheit (ReLU) ist sehr effizient zu berechnen und führt nicht zu Stabilitätsproblemen wie die Sigmoid- und tanh-Funktionen“ (Haben; Voss; Holderbaum 2023, S. 173), welche sich dabei als Standard etabliert hat und sich mathematisch definieren lässt durch:

$$\text{ReLU}(z) = \max(0, z).$$

„Sie ordnet Werte unter null der Null zu, ist aber gleich der Eingabe selbst, wenn diese größer oder gleich Null ist, d.h. eine lineare Aktivierung“. (Haben; Voss; Holderbaum 2023, S. 173) Die oben genannten Definitionen finden dabei in der Abbildung 10 eine visuelle Darstellung des ganzen Prozesses. Der Eingabe-Layer wird mit den Features der X Matrix befüllt, die dann über $\beta_{k,i}^{(1)}$ gewichtet und an die versteckten Layer weitergegeben werden. „Während die Ein- und Ausgabeschicht lediglich aus einer Ebene bestehen, können beliebig viele Ebenen an Neuronen in der verborgenen Schicht vorhanden sein. Hier werden die empfangenen Informationen erneut gewichtet und von Neuron zu Neuron bis zur Ausgabeschicht weitergereicht.“ (Wuttke o. J.) Im Ausgang-Layer werden dann die zu benötigenden Prognosewerte ausgegeben.



Input Layer $\in \mathbb{R}^n$ Hidden Layer $\in \mathbb{R}^l$ Output Layer $\in \mathbb{R}^m$

Abbildung 10: Aufbau Neuronales Netzwerk; Quelle: (Haben; Voss; Holderbaum 2023, S. 176)

Dabei gibt es zum einen die Möglichkeit den Algorithmus über Tensorflow und zum anderen über Scikit-Learn, wo er den Namen MLP - „Multi Layer Perception“ trägt, zu erstellen. Die Unterschiede liegen dabei in der Komplexität der Implementierung in Python. Für das

Tensorflow-Modell ist eine bestimmte Grundkenntnis für den Aufbau der Layer notwendig. Hingegen wird über Scikit-Learn der vollständige Algorithmus bereitgestellt und es müssen nur die Parameter in Relation zur Datenbasis gesetzt werden. Sowie in den folgenden Erklärungen von DT und RF wird hier keine Gewichtung über die BEP angewendet. Aufgrund der hohen Komplexität und den unbekanntenen Auswirkungen darauf, werden die Gewichte nur auf die performanteren und weniger komplexen Algorithmen angewendet.

2.3.2 DT - Decision Tree Regressor

Der Decision Tree Regressor ist ein Algorithmus, der, wie die namensgebende Bezeichnung andeutet, einen Entscheidungsbaum bildet. „Ein Entscheidungsbaum definiert eine Funktion, indem er die Trainingsbeobachtungen im Bereich X in disjunkte Teilmengen aufteilt, die sich unterscheiden und nicht überlappen. Die Funktion ist einfach der Durchschnittswert der historischen Beobachtungen der abhängigen Variable innerhalb jeder disjunkten Teilmenge.“ (Haben; Voss; Holderbaum 2023, S. 162) Der Decision Tree Regressor minimiert in seiner Standardform von Scikit-Learn den quadratischen Fehler, „der der Varianzreduktion als Kriterium für die Auswahl der Merkmale entspricht und den L2-Verlust minimiert, indem der Mittelwert jedes Endknotens verwendet wird“ („Sklearn - DT“ o. J.). Der L2-Verlust wird minimiert, indem man die Vorhersagen des Modells so anpasst, dass die quadratische Abweichung zwischen den Vorhersagen und den tatsächlichen Werten möglichst gering wird. Dies bedeutet, dass das Modell versucht, eine Linie oder Kurve zu finden, welche die Datenpunkte möglichst gut repräsentiert, indem es die Abstände zwischen den tatsächlichen Werten und den Vorhersagen minimiert. Dabei wird die Funktion in (Rostum u.a. 2020, S. 843) wie folgt ausgedrückt:

$$MSE = \frac{\sum_{t=1}^S (A_t - P_t)^2}{S}$$

Die Gewichtung über BEP erfolgt indem die Entscheidungsregel des Decision Trees beeinflusst wird. Ein Entscheidungsbaum besteht aus einer Sequenz von Entscheidungsregeln, die in diesem Fall verwendet werden, um aus Datenpunkten Werte zu prognostizieren. Jeder Entscheidungsknoten im Baum stellt eine Funktion dar, die auf einem bestimmten Merkmal und Schwellenwert basiert. Die Gewichtungsoption in Entscheidungsbäumen kann verwendet werden, um das Gewicht jedes Datenpunkts in der Trainingsphase anzupassen. Mathematisch gesehen wird die Gewichtung der Datenpunkte in Entscheidungsbäumen normalerweise durch die Minimierung einer Verlustfunktion, also die L2 bzw. MSE Funktion erreicht. Das Ziel besteht darin, die Gesamtverluste der Entscheidungsregeln zu minimieren, während die Gewichte der Datenpunkte berücksichtigt

werden. Indem man den Gewichten unterschiedliche Werte zuweist, können bestimmte Datenpunkte mehr oder weniger Einfluss auf die Modellanpassung haben. In Abbildung 11 ist ein Beispiel dargestellt, welches zufällig generierte Trainingsdaten beinhaltet und in dem die ersten fünf Datenpunkte mit dem Wert 20 gewichtet wurden. Die Methode ohne Gewichtung hat einen R^2 von 0,933 und jene mit Gewichtung einen Wert von 0,953. Der Wert ist nur gering höher, aber es ist zu erkennen, dass der Graph durch die Gewichtung versucht näher an den ersten fünf Datenpunkten zu sein. Die Funktion der Gewichtung mit BEP ergibt sich aus der vorherigen Erklärung wie folgt:

$$MSE_{BEP} = \frac{\sum_{t=1}^S (A_t - P_t)^2 * BEP_t}{S}$$

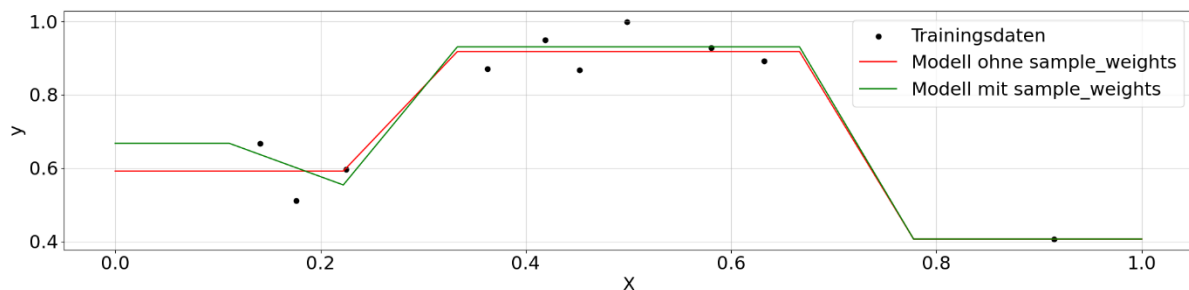


Abbildung 11: Beispiel Auswirkung Sample-Weights bei DT

Weitere Gründe für die Verwendung des DT sind, dass dieser die Entscheidungsregeln basierend auf den Eingabedaten erstellt und die Interpretation der Vorhersagen, da die Entscheidungspfade im Baum leicht nachvollziehbar sind. Zudem kann der DT gut mit fehlenden Daten umgehen, da nur Entscheidungen basierend auf den vorhandenen Daten getroffen werden. Dadurch ist das Modell weniger anfällig für fehlende Datenpunkte, die in realen Datensätzen häufig vorkommen können. DTs sind in der Regel robust gegenüber Ausreißern in den Daten. Sie teilen den Datensatz in immer kleinere Untergruppen auf und berücksichtigen dabei die Mehrheit der Datenpunkte. Aufgrund dessen haben Ausreißer weniger Einfluss auf die Vorhersagen des Modells. Ein zusätzlich wichtiger Punkt ist die effiziente Verarbeitung von großen Datenmengen, ohne dass komplexe Berechnungen notwendig sind und dass Echtzeitberechnungen relativ einfach durchgeführt werden können.

2.3.3 RF - Random Forest Regressor

„RF ist eine Gruppierungsmethode, die viele Entscheidungsbaumvorhersagen zusammenfasst, dabei besteht keine Korrelation zwischen den einzelnen Entscheidungsbäumen. Die Darstellung von RF wird hauptsächlich durch die Zufallsauswahl

von Merkmalen reflektiert, wenn die Probe in eine zufällige Anzahl von Stichproben (Bootstraps) zurückgelegt und ein Entscheidungsbaum erstellt wird.“ (Liu u.a. 2018, S. 3) In (Haben; Voss; Holderbaum 2023, S. 167) werden zudem die Vorteile der einfachen Implementierung, der gute Kompromiss zwischen Verzerrung und Varianz und dass trotz Einfügen von tausenden Eingangsvariablen es zu keiner Überanpassung („Overfitting“) kommt, beschrieben. Der Nachteil dieses Algorithmus liegt aber darin, dass wenn sich Daten außerhalb der Grenzen befinden, keine gute Prognose erstellt werden kann. Hierbei kann dieselbe mathematische Definition wie für den Decision Tree Regressor in Kapitel 2.3.2 verwendet werden, da ein RF aus mehreren Entscheidungsbäumen besteht und sich der Aufbau gleich verhält. Diese Methode minimiert ebenfalls den quadratischen Fehler.

2.3.4 SVR - Support Vector Regressor

Die Methode des SVR ist sehr beliebt, denn sie wird häufig auch in Finanzprognosen, Zeitreihenanalysen oder im Maschinenbau verwendet. „Die Idee der Support-Vektor-Regression, die eine Erweiterung der Support-Vektor-Maschine ist, besteht darin, eine Funktion zu finden, bei der jede Vorhersage y höchstens ϵ weit vom Zielwert entfernt ist.“ (Groß u.a. 2021, S. 5) Welche wie folgt definiert ist:

$$y = w^T * \phi(x) + b$$

Der Term w stellt in diesem Fall den Gewichtsvektor für die Entscheidungsfunktion y dar, $\phi(x)$ ist die Kernel-Funktion und x dabei der verwendete Feature-Datensatz. Der Gewichtsvektor w enthält die Gewichte für die einzelnen Merkmale, die angeben, wie stark jedes Merkmal zur Vorhersage beiträgt. Wobei die Minimierungsfunktion lautet:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \xi_i^*),$$

$$\text{s. t. } y_i - w^T * \phi(x) - b \leq \epsilon + \xi_i^*, i = 1, \dots, N,$$

$$w^T * \phi(x) - y_i + b \leq \epsilon + \xi_i, i = 1, \dots, N,$$

$$\xi_i, \xi_i^* \geq 0, i = 1, \dots, N.$$

„ N ist die Anzahl der Stichproben im Trainingsatz und b ist die Verzerrung. Ziel ist es, eine Regressionsgerade zu erhalten, die zum einen flach ist und zum anderen den

Vorhersagefehler minimiert. Um eine flache Regressionsfunktion zu erhalten, möchte man die Norm minimieren. Abweichungen von der ϵ -Röhre werden durch die Schlupfvariablen ξ_i und ξ_i^* toleriert. Die Konstante C stellt einen Kompromiss zwischen der Flachheit der Funktionen und der Anzahl der Vorhersagen dar, die außerhalb der ϵ -Röhre toleriert werden können.“ (Groß u.a. 2021, S. 5) Diese Arbeit beschränkt sich aber lediglich auf den Parameter C und dessen Ermittlung, was in Kapitel 3.5 erläutert wird. Wie auch das ANN, besitzt die SVR eine Vielzahl an Aktivierungs- bzw. Abbildungsfunktionen. Dabei wird standardmäßig im Scikit-Learn Paket die „RBF“, besser bekannt als Radial Basis Funktion, gesetzt. Die RBF-Kernel-Funktion verwendet eine Gauß'sche Funktion des Abstands zwischen den Eingabevektoren. Diese Funktion ermöglicht die Modellierung komplexer nichtlinearer Zusammenhänge und ist die am häufigsten verwendete Kernel-Funktion in der SVR, die sich in („Sklearn - SVM“ o. J.) wie folgt definiert:

$$\phi(x) = \exp(-\gamma \|x - x'\|^2).$$

Der Parameter γ ist dabei spezifiziert als Gamma und muss Größer als 0 sein. Doch wieso wird SVR so oft in der Literatur erwähnt? SVR ist in der Lage, nichtlineare Zusammenhänge zwischen den Eingabevariablen (wie Wetterdaten, Uhrzeit, saisonale Trends, usw.) und der Lastprognose abzubilden. Das ist wichtig, da die Netzabgabemenge von einer Vielzahl von Faktoren beeinflusst wird, die nicht immer linear miteinander zusammenhängen. Zudem ist die Methode robust gegenüber Ausreißern in den Daten. Ausreißer können in Lastprognosen beispielsweise aufgrund unvorhergesehener Ereignisse oder Fehlern in den Daten auftreten. Der SVR berücksichtigt diese Ausreißer nicht sehr stark bei der Modellbildung, was zu robusteren Prognosen führen kann.

2.3.5 LR - Linear Regression

Die Lineare-Regression beschreibt eine lineare Beziehung zwischen einer abhängigen Zielvariable und einer oder mehreren Features. Ziel der linearen Regression ist es, anhand der gegebenen Daten ein Modell zu erstellen, das die bestmögliche Vorhersage für die Zielvariable auf Basis der unabhängigen Variablen macht. „Diese Methode passt ein lineares Modell mit Koeffizienten an, um die Summe der quadrierten Fehler zwischen den tatsächlichen Werten im Datensatz und den Vorhersagewerten zu minimieren.“ („Sklearn - LR“ o. J.) Die Minimierungsfunktion wird dabei wie folgt definiert:

$$\min_w \|Xw - y\|_2^2$$

Wobei w die Koeffizienten, X die Features und y die tatsächlichen Werte darstellt. Das Ziel dabei ist es, die optimalen Werte für die Koeffizienten w zu finden, um die Summe der quadrierten Fehler zu minimieren. Dieser Prozess wird normalerweise unter Verwendung von Methoden wie der Methode der kleinsten Quadrate durchgeführt, um das beste Modell zu finden, das die Daten am besten beschreibt.

$$\hat{y}(w, x) = w_0 + w_1x_1 + \dots + w_px_p$$

In diesem Fall wird w_0 als Intercept-Wert hinzugefügt und x_p spiegelt die einzelnen Features wider.

2.3.6 Gurobi - LP

Das Gurobi-Modell bietet eine optimale Lösung für das gegebene lineare Regressionsproblem, welches die Optimierung nicht auf Basis z.B. des quadratischen Fehlers, sondern auf Basis der Gewichtung der absoluten Prognosefehler über die Ausgleichsenergiepreise, also eine Minimierung der Ausgleichsenergiekosten durchführt. Dabei wird die L1-Regulierung verwendet, welche den Effekt hat, dass sie einige der Koeffizienten auf Null setzen kann, was bedeutet, dass sie diejenigen Variablen auswählt, die am stärksten mit der Zielvariable korrelieren und andere Variablen eliminiert. Dies kann zu einem Modell mit höherer Vorhersagekraft und interpretierbaren Ergebnissen führen. Mit dieser Methode soll vor allem erzielt werden, dass die Ausgleichsenergiekosten im Vergleich zu den anderen Methoden geringer ausfallen. Denn komplexere Methoden wie ANN, MLP oder SVR sind in der Literatur weit verbreitet im Gegensatz zu Modellen die eine Lineare Regression abbilden.

Das Modell setzt sich aus Entscheidungsvariablen, Nebenbedingungen und einer Zielfunktion zusammen. Entscheidungsvariablen dienen dazu, die Entscheidungen zu repräsentieren, die in einem Optimierungsproblem getroffen werden müssen und jede steht für eine zu optimierende Größe, die einen bestimmten Wert annehmen kann. Diese sind in der Arbeit als kontinuierliche Entscheidungsvariablen verwendet worden, welche jeden beliebigen Wert innerhalb eines bestimmten Wertebereichs annehmen können. Dazu zählt die Variable c als Vektor, der die Regressionskoeffizienten für jedes Merkmal in X enthält. Diese Koeffizienten werden im Modell optimiert, um die beste Vorhersage \hat{y}_t für die Zielvariable y_t zu erzielen. c ist dabei eine Variable ohne Schranken. In diesem Modell wurden die Entscheidungsvariablen mit dem Variablentyp „MVar (Multiple Variables)“ implementiert, was es möglich macht, mehrere Variablen, die in einem Vektor oder einer Matrix beinhaltet sind, dem Modell gleichzeitig hinzuzufügen.

Nebenbedingungen stellen hingegen die Einschränkungen einer Lösung des Optimierungsproblems dar, indem sie bestimmte Beziehungen zwischen den Entscheidungsvariablen festlegen. Diese Beziehungen können mathematische Gleichungen oder Ungleichungen sein und repräsentieren typischerweise die Beschränkungen. Für a_t , welcher den Absolutwert aus der Differenz zwischen den tatsächlichen Werten y_t und den Vorhersagewerten \hat{y}_t repräsentiert, gilt:

$$a_t = |y_t - \hat{y}_t|.$$

Für die obere mathematische Definition des Absolutwertes a_t in diesem Modell, musste die Umsetzung in Gurobi in anderer Weise erfolgen, da das Modell keine Möglichkeit bietet direkt einen Absolutwert aus der Differenz zwischen y_t und \hat{y}_t zu bilden. Dies ist wie folgt implementiert worden:

$$-a_t \leq y_t - \hat{y}_t \leq a_t.$$

In diesem Fall muss $y_t - \hat{y}_t$ größer gleich $-a_t$ und $y_t - \hat{y}_t$ kleiner gleich a_t sein und somit mussten zwei Nebenbedingungen für die Definition $a_t = |y_t - \hat{y}_t|$ im Modell hinzugefügt werden. Der Vorhersagewert \hat{y}_t wird dabei als Nebenbedingung und Regression definiert als:

$$\hat{y}_t = X_t * c_t,$$

Die Vorhersagewerte \hat{y}_t müssen gleich der Matrixmultiplikation der Merkmale in X_t und dem Konstantenvektor c_t sein. Bei dieser Methode besteht die erste Spalte X , welche die Feature-Matrix darstellt, nur aus Einsen. Durch das Hinzufügen einer Spalte von Einsen, wird eine zusätzliche Dimension zum Merkmalsraum hinzugefügt und somit sichergestellt, dass der Term in der linearen Regression berücksichtigt wird und die Vorhersage somit genauer wird, auch genannt als Intercept. \hat{y}_t kann in diesem Fall nur durch den Koeffizienten c optimiert werden, welcher selbst eine Entscheidungsvariable darstellt.

Die folgende Zielfunktion dieser Methode minimiert die Summe, der durch BEP_t gewichteten Absolutwerte a_t , indem die Koeffizienten c optimiert werden:

$$\min \sum_{t=1}^T a_t * BEP_t, \forall t = 1, \dots, T$$

Nach der Optimierung des Modells, können die optimierten Koeffizienten c ausgegeben werden. Die Koeffizienten c dienen nun dazu, um für die in X_t enthaltenen Features des Prognosezeitraums eine Matrixmultiplikation durchzuführen, um im Anschluss die neuen Vorhersagewerte \hat{y}_t zu berechnen. Somit wird im ersten Schritt mit den unterschiedlichen Trainingszeiträumen für X_t , y_t und BEP_t , wie in Kapitel 3 dargestellt, das Modell optimiert und die Koeffizienten c dazu genutzt, um eine Prognose mit X_t des Testzeitraums von 96 Viertelstundenwerten zu erstellen. Dabei werden für jeden Testzeitraum neue Koeffizienten c berechnet.

Es soll der spezielle Vergleich zu den anderen Algorithmen entstehen und dass diese Methode die Ausgleichenergiekosten minimiert und zusätzlich mit den BEP bewertet wird. Die anderen Methoden minimieren die Ausgleichenergiekosten nicht, werden aber trotzdem mit der neuen Bewertungsmethode bewertet. Dabei ist zu erwarten, dass die Kosten bei diesem Modell am geringsten ausfallen. Da dieses Modell aber auch stark der LR Methode ähnelt bzw. demselben Prinzip entspricht, welche jedoch die Ausgleichenergiekosten nicht minimiert, wird der Vergleich hier umso interessanter.

3 Anwendung der Methoden

Eine effiziente Anwendung der Algorithmen, mit denen in Kapitel 2.1 erwähnten Daten, erfordert eine sorgfältige Datenvorverarbeitung, welche hier jetzt näher beschrieben wird. Nach der Beschaffung der Daten, auch von jenen, die in Kapitel 3.1 nachbearbeitet wurden, wurde ein „Haupt“-Datenframe erstellt. Darin befinden sich die Features X , das Target y und die Ausgleichsenergiepreise als Term BEP . Der Grund für die Erstellung eines solchen Datenframes ist, dass alle Daten in Zusammenhang mit dem Index als Zeitstempel vorhanden sind und jeder Wert danach zugeordnet werden kann. Das vereinfacht z.B. die Zuordnung von Gewichtungen oder auch die Bewertung durch BEP in Abhängigkeit der Zeit, womit ein Prognosefehler um 00:00 Uhr auch mit dessen Preis bewertet wird. Nach dieser Verarbeitung der Daten können aus einem „Single-Point-of-Truth“ die Daten bezogen werden. Die Feature-Matrix wird wie folgt dargestellt, wobei die Features, die in X verwendet werden, die Temperatur T , die Regenmenge R , die Sonnenminuten S , die Windgeschwindigkeit W und die Wochentage WD (Montag bis Sonntag) sind. Jeder Wochentag wird als eigenes Feature hinzugefügt und besitzt den Wert 0 oder 1. Zusätzlich werden Minuten M und Stunden H als separate Features berücksichtigt, was insgesamt 13 Features ergibt. Dabei wird die Feature-Matrix in einem Ausschnitt in Tabelle 4 dargestellt und folgend definiert als:

$$X = \begin{pmatrix} T_0 & R_0 & S_0 & W_0 & WD0_0 & \dots & WD6_0 & H_0 & M_0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ T_t & R_t & S_t & W_t & WD0_t & \dots & WD6_t & H_t & M_t \end{pmatrix}$$

t	T	R	S	W	$WD0$...	$WD6$	H	M
190	7,42	0	0	9,51	0	...	1	23	15
191	7,36	0	0	9,77	0	...	1	23	30
192	7,31	0	0	10,03	0	...	1	23	45
193	7,25	0	0	10,29	1	...	0	00	00
194	7,05	0	0	10,28	1	...	0	00	15
195	6,84	0	0	10,27	1	...	0	00	30

Tabelle 4: Beispieltabelle - Ausschnitt Features aus X-Matrix

Und der Zielvektor y bildet sich aus der Netzabgabemenge P :

$$y = \begin{pmatrix} P_0 \\ \vdots \\ P_t \end{pmatrix}$$

Nach der Auswahl der Features und dem Target wird ein geeigneter Parameter für die angewendeten Scikit-Learn Algorithmen gesucht. Für diesen Schritt wurde GridSearchCV zur Ermittlung des besten Parameters verwendet. Dabei wurde eine Aufteilung von 5% der Testdaten zu 95% der Trainingsdaten gewählt. Das entspricht einer Menge von Viertelstundenwerte von einem Tag zu Viertelstundenwerte von drei Wochen. Hierbei wurde wie für die Prognosen selbst ein Sliding-Window-Verfahren angewendet, dass die Berechnung über den vollständigen Datensatz ausgeführt hat. Für den CV-Wert, der bestimmt, wie oft die Daten aufgeteilt werden und wie oft das Modell trainiert und validiert wird, wurde der Wert 3 definiert. Dazu wurde für jeden Algorithmus ein Durchlauf gestartet, der über den gesamten vom Datensatz beschriebenen Zeitraum den optimalen Parameter für die verschiedenen Testszenarien ermittelte. Wegen der hohen Rechenintensität war es aufgrund technischer Ressourceneinschränkungen nicht möglich, mehrere Parameter parallel in die Grid-Search-Auswahl einzubeziehen. Daher wurde für jeden Algorithmus jeweils ein Parameter ausgewählt, der Grid-Search angewendet und der beste Parameter ausgegeben.

Dabei wurden für den DT und RF die maximale Tiefe der Entscheidungsbäume gewählt. Die Veranschaulichung der Auswirkung auf die Prognosegenauigkeit in Abbildung 12 zeigt, dass, je höher der Parameter gesetzt wird, sich dieser umso mehr an die Trainingsdaten nähert. Da dieser Parameter das Prognoseergebnis stark beeinflussen kann, wurde er für den Grid-Search und auch für die Verwendung im Algorithmus verwendet.

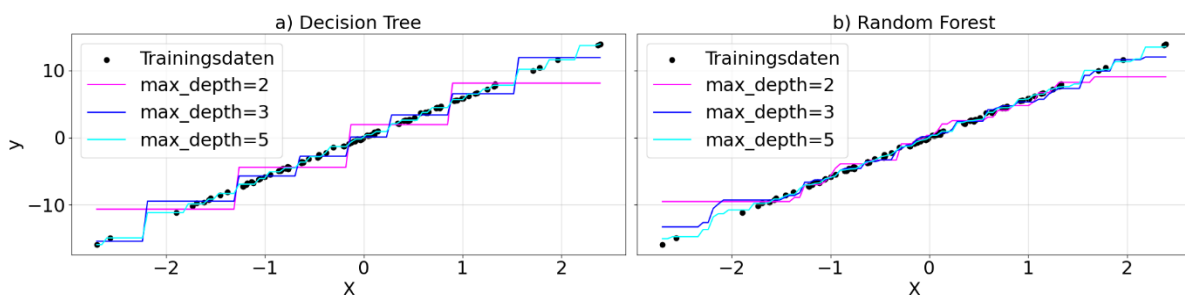


Abbildung 12: Beispiel Auswirkung der Max. Tiefe bei a) DT und b) RF

Für ANN von Tensorflow gibt es keine Möglichkeit über Grid-Search die besten Parameter herauszufinden. Dabei wird sich auf den Code in der Fallstudie über Lastprognose auf Github von (Voss o. J.) berufen, welcher darin eine Lastprognose mit ANN von Tensorflow ausführt. Dieser Teil in der Fallstudie wird dazu verwendet, um dieselben Parameter zu nutzen wie bei Voss. Beim MLP kann zwar über den Grid-Search die Anzahl an Iterationen bestimmt werden, jedoch ist dieser Schritt für einen Computer mit wenig Rechenleistung nicht performant genug, sodass durch schrittweises testen die Anzahl an Iterationen gefunden wurde bis das Modell konvergiert ist. Es steuert nämlich, wie oft das Netzwerk die Trainingsdaten durchläuft, um die

Gewichte und das Modell anzupassen. Während des Trainings eines MLP-Modells werden die Gewichte des Netzwerks schrittweise angepasst, um die Fehler zwischen den vorhergesagten und den tatsächlichen Werten zu minimieren. Jede Iteration durchläuft das Netzwerk und die Trainingsdaten einmal und aktualisiert die Gewichte entsprechend den Gradientenabstiegsregeln. Werden dabei zu wenig Iterationen verwendet, so sind noch immer starke Veränderungen des Modells möglich und es ist somit noch nicht konvergiert. Das bedeutet, dass das MLP-Modell nicht in der Lage war, die Gewichte so anzupassen, dass es eine optimale Lösung findet. In Abbildung 13 wird gezeigt: je höher die Iteration, desto besser passt sich der Algorithmus an die Trainingsdaten an. Es ist aber auch zu erkennen, dass der Schritt zwischen 300 und 10000 keine gravierenden Verbesserungen mehr bringt. Daraus lässt sich ableiten, dass mit 10000 Iterationen nur unnötig Ressourcen verbraucht werden, weil das Modell mit 300 Iterationen auch schon ausreichend gute Ergebnisse erzielt.

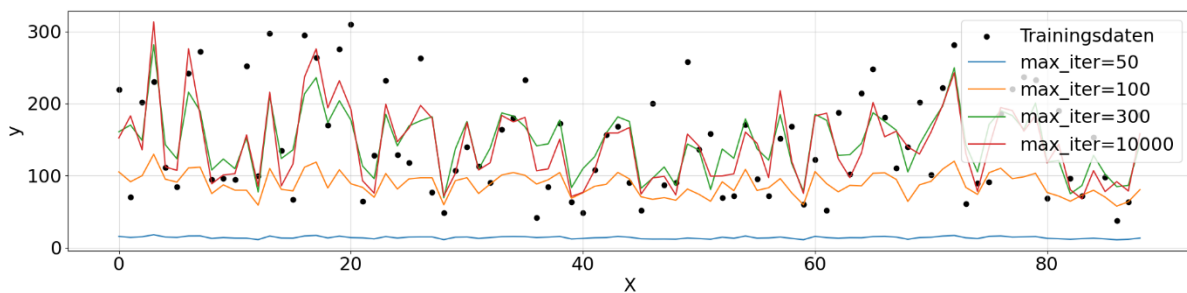


Abbildung 13: Beispiel Auswirkung maximale Iterationen bei MLP

Für die SVR ist der Regulierungsparameter C die einflussreichste Einzelkomponente im Algorithmus. Dieser steuert das Gleichgewicht zwischen der Minimierung des Trainingsfehlers und der Aufrechterhaltung eines einfachen Modells. Ein höherer Wert von C führt in diesem Fall zu einer strengeren Anpassung an die Trainingsdaten, während ein niedrigerer Wert von C zu einer glatteren Kurve führt. Mathematisch gesehen zielt SVR darauf ab, eine Funktion zu finden, die die Trainingsdaten optimal anpasst, während der Fehler innerhalb einer bestimmten Grenze bleibt. Der Parameter C spielt dabei eine entscheidende Rolle, da er den Kompromiss zwischen der Minimierung des Trainingsfehlers und der Zulassung von mehr Flexibilität im Modell bestimmt. Ein großer Wert von C führt dazu, dass SVR versucht, die Trainingsdaten so genau wie möglich anzupassen, was zu einem möglicherweise komplexeren Modell mit einem geringeren Trainingsfehler führen kann. Ein kleinerer Wert von C hingegen ermöglicht mehr Flexibilität, kann aber zu einem größeren Trainingsfehler, aber auch zu einem einfacheren Modell führen. In Abbildung 14 ist beispielhaft dargestellt wie sich der Parameter bei unterschiedlichen C Werten verhält. Wie zuvor erwähnt, je höher der Wert C ist, umso mehr passt sich der Algorithmus an die Trainingsdaten an. Für C gibt es keine

Grenze und der Wert für C kann theoretisch als fast unendlich bzw. eine sehr hohe Zahl definiert sein.

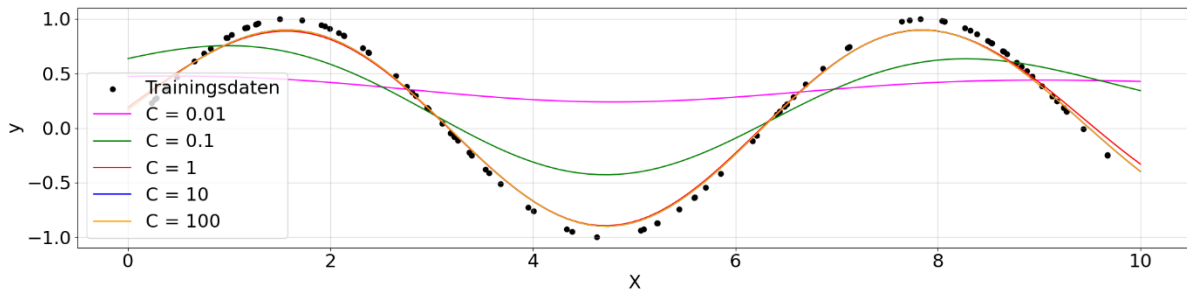


Abbildung 14: Beispiel Auswirkung C bei SVR

Im Anschluss an die Parametersuche erfolgt die Auswahl der Anzahl an Test- und Trainingsdaten für das Trainieren der Algorithmen auf die Datenbasis. Wie in der Einleitung erwähnt, wird immer ein vollständiger Tag mit 96 Viertelstundenwerten prognostiziert. Dabei wird jedoch mit unterschiedlich großen Trainingsdatensätzen gearbeitet, die wie folgt ausgewählt wurden:

- 1 Woche (7 Tage)
- 3 Wochen (21 Tage)
- 4 Wochen (28 Tage)
- 8 Wochen (56 Tage)

Es wird geprüft, worin die Unterschiede zwischen den Ausführungen der unterschiedlichen Trainingsdaten liegen und ob das die Ergebnisse des BEC und MBEC letztenendes verbessert. Dabei kann die Methode der unterschiedlichen Trainingslängen grafisch vereinfacht auch folgendermaßen wie in Tabelle 5 dargestellt werden:

Verwendung an Trainingstagen:											
Prognose für Datum:	27.03.	28.03.	29.03.	30.03.	31.03.	01.04.	02.04.	03.04.	04.04.	05.04.	06.04.
01.04.2023	Grey	Grey	Grey	Grey	Grey	Red					
02.04.2023	Grey	Grey	Grey	Grey	Grey	Red	Red				
03.04.2023	Grey	Grey	Grey	Grey	Grey	Red	Red	Red			
04.04.2023	White	Grey	Grey	Grey	Grey	Red	Red	Red	Red		
05.04.2023	White	White	Grey	Grey	Grey	Red	Red	Red	Red	Red	
06.04.2023	White	White	White	Grey	Grey	Red	Red	Red	Red	Red	Red

Tabelle 5: Iterationsprozess der Prognose mit 7 Trainingstagen

Rot visualisiert den zu prognostizierenden Tag und der graue Bereich die Menge an Trainingsdaten. Dabei wurde beispielsweise die Vorhersage mit Trainingsdaten von einer Woche dargestellt, wodurch ersichtlich ist, wie die Schleife jeden Tag durchläuft. Im Code wird die Schleife dann je Algorithmus wie folgt ausgeführt:

1. Der Startpunkt der Trainingsdaten wird auf den Index des ersten Datums im Datensatz gesetzt.
2. Der Endpunkt der Trainingsdaten wird als Startpunkt der Trainingsdaten plus die Größe des Trainingsfensters festgelegt.
3. Der Endpunkt der Testdaten wird als Endpunkt der Trainingsdaten plus die Größe des Testfensters festgelegt.
4. Es wird eine Schleife gestartet, die solange läuft, bis der Endpunkt der Testdaten den Index des letzten Datums im Datensatz erreicht.
5. Innerhalb der Schleife werden die Trainingsdaten und Testdaten aus den entsprechenden Bereichen des Gesamtdatensatzes extrahiert.
6. Ein Algorithmus-Modell wird mit den Trainingsdaten erstellt und angepasst.
7. Eine Vorhersage für den Testzeitpunkt wird mit dem Modell durchgeführt.
8. Die Start-, End- und Testpunkte werden um einen Tag verschoben, um den nächsten Schritt des Sliding-Windows zu ermöglichen.
9. Die Schleife wird fortgesetzt, bis alle Testzeitpunkte abgedeckt sind.

Dieser Ansatz ermöglicht es, den Modelltrainings- und Testprozess über eine feste Anzahl von Schritten fortzuführen und die Leistung des Modells für verschiedene Zeiträume zu bewerten. Und mit diesem Modell werden die Prognosen in den folgenden Kapiteln durchgeführt. Dabei liegt das Ziel darin, mit den vorhergesagten Werten die Kosten durch die Ausgleichsenergiepreise so gering wie möglich zu halten.

Dabei muss noch berücksichtigt werden, dass aufgrund einer unterschiedlich großen Anzahl von Trainingstagen und trotz der Ausführung über den Gesamtzeitraum, die Ergebnismenge unterschiedlich ist. Es wird somit jener Zeitraum verglichen, in dem auch die Prognosen mit 8 Wochen Trainingsmenge vorhanden sind, da logischerweise bei der Ausführung die Startmenge vom Gesamtzeitraum abgezogen werden muss und diese dann die geringste Prognosemenge aufweist. Der Zeitraum reicht in diesem Fall vom 22.02.2022 bis Ende der Datenverfügbarkeit zum 01.04.2023.

3.1 Gewichtung und Bewertung der absoluten Prognosefehler

Der Versuch für eine bessere Prognose mit einer Gewichtung durch BEP wird an drei (DT, GP und RF) der sieben Methoden getestet. Ziel dabei ist es, bei den Methoden von Scikit-Learn die Funktion für das Gewichten über Samples zu verwenden. Diese wird von Scikit-Learn als „Sample-Weights“ bezeichnet und wird als Vektor bei allen Algorithmen hinzugefügt, um den quadratischen Fehler bei der Erstellung eines Entscheidungsbaums zu gewichten. Diese Methode unterscheidet sich aber wesentlich von der eigens erstellten Methode durch Gurobi, da diese nicht den absoluten Prognosefehler durch die Ausgleichsenergiepreise gewichtet und somit keine Optimierung der BEC oder MBEC durchführt. Trotzdem soll der Vergleich der Methoden mit unterschiedlichen herangehensweisen einer Gewichtung aufzeigen, ob diese ein besseres Resultat erzielen, als jene Methoden ohne Gewichtung.

Die Ausgleichsenergiepreise in 0 werden dazu nicht eins zu eins auf den berechneten Zeitraum angewendet, sondern müssen vor der Anwendung bearbeitet werden. Denn für eine Bewertung oder Gewichtung soll ermittelt werden, wann eine Abweichung vom realen Messwert innerhalb eines Tages schlechter bewertet oder ein Feature höher gewichtet wird als sonst. Dabei wurde über den Gesamtzeitraum der Ausgleichsenergiepreise zwischen 01.01.2022 und 01.04.2023 für jede Viertelstunde der Mittelwert gebildet und somit entstanden 96 Viertelstundenmittelwerte für einen vollständigen Tag, welche in Abbildung 15 a) dargestellt sind. Nach diesem Schritt ist zu erkennen, dass zu bestimmten Tageszeiten unterschiedliche Preise für die Ausgleichsenergie bezahlt werden müsste. Um dann noch Spitzen und Täler der Mittelwerte zu bereinigen, wird eine Glättung mittels Savitzky-Golay-Filter angewendet, der sich wie in (Liu u.a. 2016, S. 103) mathematisch beschreiben lässt als:

$$f_k(i) = \sum_{n=0}^k b_n * i^n, i \leq 2m.$$

Der Term b_n bildet den Koeffizienten, k den Grad des Polynoms und m den halben Wert der ausgewählten Fenstergröße. Der Funktionswert f_k gibt dann den geglätteten Wert aus. Für den Savitzky-Golay-Filter wurden im Python-Code folgende Parameter verwendet:

Parameter	Wert
k	6
m	86

Tabelle 6: Savitzky-Golay-Filter Parameter

Das Ergebnis des Savitzky-Golay-Filters kann in der Abbildung 15 als rote Linie erkannt werden, welche im Vergleich zu den Wochentagen und Saisonszeiten dargestellt ist. Zudem sind im Vergleich dazu die geglätteten Preise von Day-Ahead und Intraday abgebildet. Um aufzuzeigen, dass die Glättung über den Mittelwert ein plausibler Weg zur Ermittlung eines Bewertungs- oder Gewichtungsschema ist, sind die Vergleiche in der Abbildung 15 unter Punkt b) Wochentagsvergleich und c) Saisonsvergleich dargestellt. Dabei ist in b) zu erkennen, dass der Samstag und Sonntag zwischen 06:00-09:00Uhr, stark von den restlichen Wochentagen abweichen aber die geglätteten Werte in der Regel stimmen. Betrachtet man nun die saisonalen Verhältnisse unter c), dann ist offensichtlich, dass die Sommermonate einen deutlich höheren Durchschnittspreis haben als die anderen drei Saisonen.

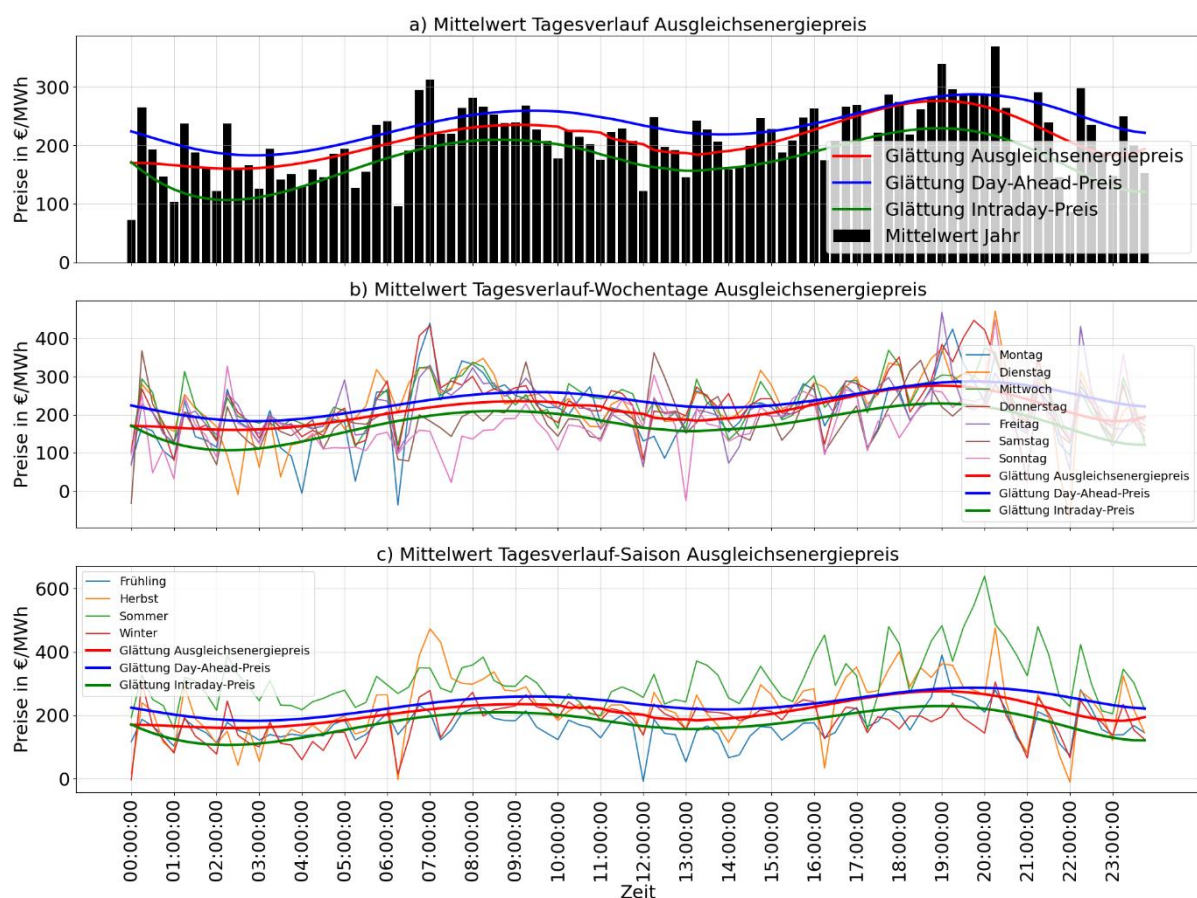


Abbildung 15: Glättung und Vergleich Ausgleichsenergiepreise

Auf Basis dieser Grundlage wird für jeden Tag die gleiche Gewichtung angewendet und diese bestimmt damit, dass eine Abweichung vom realen Messwert etwa von 06:00Uhr bis 11:00Uhr und von 16:00Uhr bis 21:00Uhr deutlich höher bewertet wird, als die Abweichungen zu den anderen Zeitpunkten am Tag. Hier sollte eine größere Abweichung in der Prognose vermieden werden, da die BEP und die Schwere der Fehler in diesen Zeitfenstern viel höhere

Auswirkungen auf BEC und MBEC haben. Für die rote Linie, also den geglätteten Mittelwert, gilt damit:

$$BEP = \begin{pmatrix} f_0 \\ \vdots \\ f_k \end{pmatrix},$$

wobei BEP das Preisschema eines Tages abbildet und f die Preise einer Viertelstunde. Wird dieses Preisschema nun mit den absoluten Prognosefehlern multipliziert, so entstehen die BEC oder MBEC. In Abbildung 16 wird dabei in einem Beispiel visualisiert, welche absoluten Prognosefehler mit dem BEP multipliziert werden.

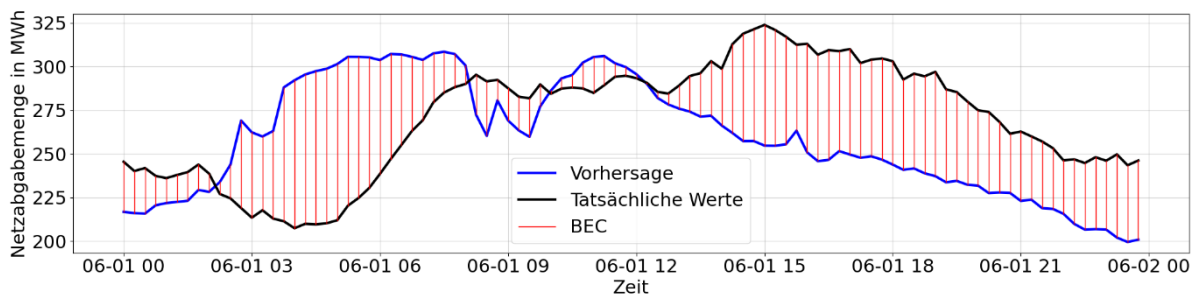


Abbildung 16: Kostendarstellung anhand Prognosefehler

Stellen wir nochmal den Vergleich zu den illwerke vkw AG her, so nutzen sie für ihre Prognosen keine Gewichtung über Ausgleichsenergiepreise bzw. werden keine Kennzahlen wie z.B. BEC und MBEC verwendet. Im Interview mit dem Energiehandel wurde der Einsatz einer solchen Methode aber positiv aufgenommen.

3.2 Ergebnisse ANN und MLP

Wie zu Beginn des Kapitel 3 erwähnt, wurden für die Methode über Tensorflow die Parameter einer Fallstudie für die Ausführung verwendet. Die dabei gesetzten Parameter wurden ohne Veränderung übernommen und bei der Prognose angewendet. In dem Modell der Fallstudie wurden zwei Layer verwendet, die jeweils die Größe von 50 Neuronen besitzen und für beide wurde die Aktivierungsfunktion ReLU angewendet. Eine größere Anzahl von versteckten Schichten und Neuronen kann zu einem komplexeren Modell führen, welches in der Lage ist, komplizierte Muster in den Daten zu erfassen. Dies kann jedoch zu Overfitting führen, insbesondere wenn die Daten begrenzt sind. Ein kleineres Netzwerk mit weniger Schichten

und Neuronen kann zu einer besseren Generalisierung führen, ist jedoch möglicherweise nicht in der Lage, die volle Komplexität des Problems zu erfassen.

Zudem wurde eine „Batch“-Größe von 16 verwendet, welche eine Auswirkung auf die Parallelisierung von gleichzeitig verarbeiteten Trainingsdaten hat, was zu einer effizienteren Nutzung der Hardware führen kann. Das Training kann schneller ablaufen, da die Berechnungen parallelisiert werden können. Allerdings kann ein zu großer Batch auch zu einem erhöhten Speicherbedarf führen, insbesondere wenn die Datenmenge sehr groß ist. Ebenfalls kann ein größerer Batch zu einer ungenauen Schätzung des Gradienten führen, da die Gewichtsaktualisierungen auf Basis des durchschnittlichen Gradienten über den Batch erfolgen.

Um eine korrekte Vergleichsbasis zu schaffen, werden dieselben Parameter auch für die Methode über Scikit-Learn verwendet. Der Unterschied zwischen den beiden liegt nur darin, dass für die MLP die Anzahl an Iterationen definiert werden muss. Die Vorgehensweise, um den korrekten Parameter zu finden, wurde durch einzelne Tests mit unterschiedlichen Parametergrößen ausgeführt. Dabei wurde in 500er Schritten die Iterationen bis 14000 getestet, bis das Modell konvergierte. Hierbei hätte der Grid-Search die optimalen Parameter suchen können, dieser wäre jedoch zu zeitintensiv gewesen und nur Rechner mit massiv höherer Leistung wären für die Ausführung in Frage gekommen. Das war auch der Grund für die testweise Erhöhung des Parameters.

Die Ergebnisse der BEC und MBEC konnten mit der Erhöhung an Trainingsdaten fortlaufend verbessert werden. Vergleicht man beide Modelle untereinander, so sind die Resultate trotz der gleichen Parameter unterschiedlich und somit konnte die ANN Variante einen BEC Wert von 393,88 Mio.€ erreichen. Performanter war dabei die MLP, denn diese erzielte einen BEC Wert von 384,51€. Wieso dieses Ergebnis und diese Differenz zustande gekommen sein könnte, wird in der Ergebnisinterpretation in Kapitel 4 näher erläutert.

Trainingsdaten	BEC ANN in Mio.€	MBEC ANN in €	BEC MLP in Mio.€	MBEC MLP in €
1 Woche	572,93	14.842,00	513,79	13.309,91
3 Wochen	504,04	13.057,25	408,63	10.585,74
4 Wochen	481,19	12.465,34	395,50	10.245,47
8 Wochen	393,88	10.203,72	384,51	9.960,88

Tabelle 7: Ergebnisse ANN und MLP, BEC und MBEC

Wie die in vorangegangener Literatur und zu Anfang erwähnten Methoden, konnte auch die Methode der Artificial Neural Networks nur teilweise überzeugen. Dabei weicht das beste

Resultat von ANN nur wenig von denen von DT, LR, RF, SVR und dem Gurobi-Modell ab. In der Darstellung in Abbildung 17 kann der Vergleich der von beiden Algorithmen prognostizierten Daten in Relation zu den tatsächlichen Werten und die Anzahl der erreichten R^2 -Scores betrachtet werden. Hierbei scheint es, als ob beide Algorithmen mit einzelnen Peaks oft Werte unter den realen Messdaten prognostiziert haben, wodurch die erhöhten BEC und MBEC zustande kamen. Dafür waren die Prognosen um den Jahreswechsel, wo auch eine starke Veränderung der Netzabgabemenge vorherrscht, durchaus besser als im Sommer.

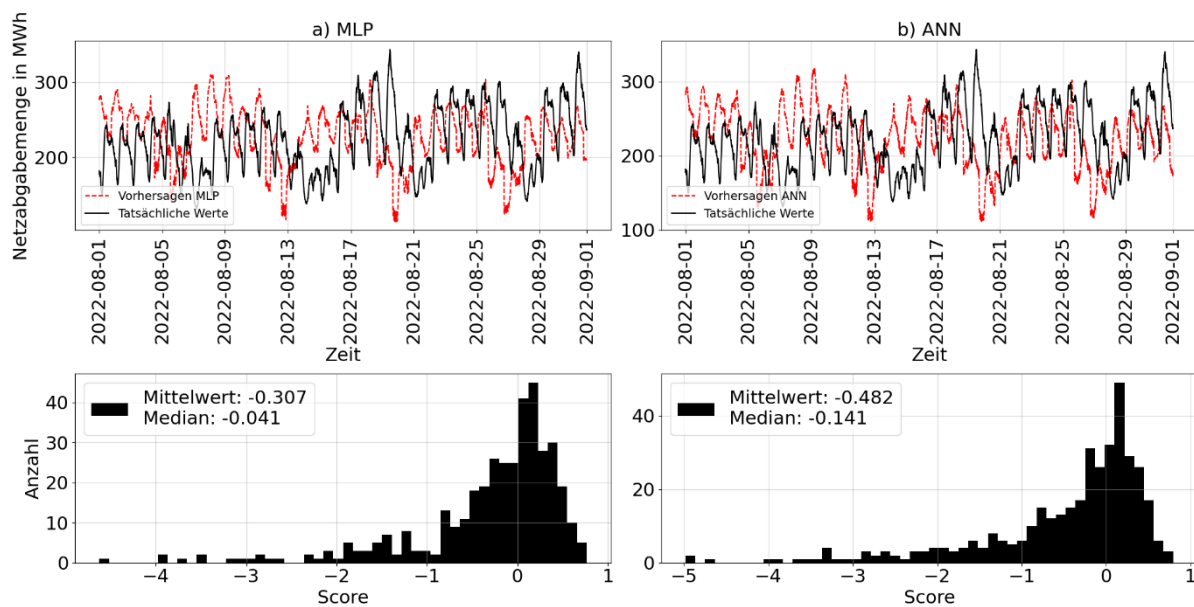


Abbildung 17: Ergebnisse ANN und MLP mit Vorhersagewerte und Test-Scores

Wie auch bei den folgenden Methoden, wurden durch die größeren Ausreißer in den Prognosen tiefe negative R^2 Werte ausgegeben. Man sieht aber, dass sich bei der Verteilung die meisten Werte nahe der Null befindet. Dabei wurden für eine bessere Visualisierung alle Werte, die sich unterhalb von -5 befinden, ausgegrenzt.

3.3 Ergebnisse DT

Bei der Auswertung der Grid-Search Ergebnisse für den Decision Tree Regressor zeigte sich, dass der Parameter für die maximale Baumtiefe am häufigsten den Wert von 7 erreichte. Dabei wurde für jeden Durchlauf der Bereich von 1 bis 30 eingeschränkt. In der Tabelle 8 sind die besten 10 Parameterwerte nach der Häufigkeit gelistet, wie oft der höchste Test-Score erzielt wurde und auch wie nahe sich die Werte 4 und 3 dem Wert 7 befinden. Demzufolge kann nicht eindeutig ausgesagt werden, dass der Parameter mit dem Wert 7 über einen größeren Zeitraum gute Ergebnissen bringen würde.

Im Boxplot der Abbildung 18 ist zudem zu erkennen, dass beim Parameterwert 7 der höchste Punkt erreicht werden konnte und bei den Werten danach die Streuung der Testergebnisse wieder zunimmt. Somit wäre mit mehr schlechteren Resultaten durch eine erhöhte Baumtiefe zu rechnen.

Max. Tiefe	7	4	3	5	6	8	9	10	12	2
Häufigkeit	55	48	44	39	38	37	34	20	17	10

Tabelle 8: Grid-Search DT, Top 10 Parameter-Häufigkeit

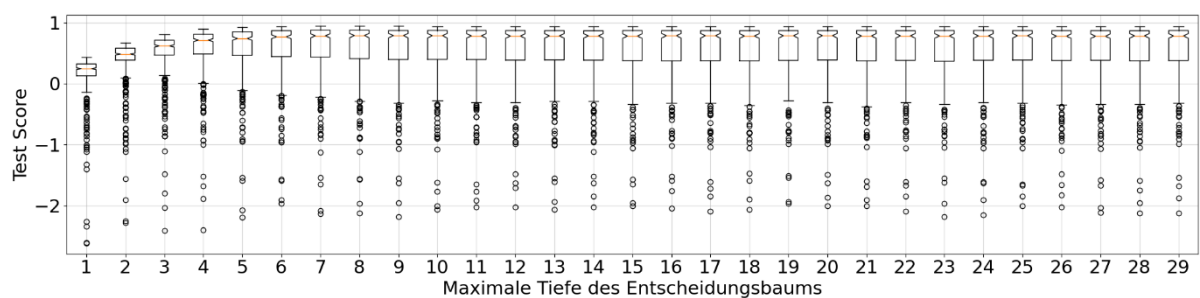


Abbildung 18: Boxplot der Grid-Search Ergebnisse DT

Der Decision Tree Regressor wurde nach der Grid-Search Auswertung und trotz des nicht eindeutigen Ergebnisses mit dem Parameterwert 7 ausgeführt. Die besten Ergebnisse konnten in der Ausführung mit acht Wochen an Testdaten erzielt werden. Die Methode kam auf einen Test-Score im Median von 0,638 aber der Mittelwert betrug -0,095. Der schlechte Mittelwert begründet sich auf die stark negativen Einzelergebnisse, welche in Abbildung 19 b) im linken Bereich des Graphen ersichtlich sind, weshalb der Median hier ein plausibleres Ergebnis widerspiegelt. In Abbildung 19 a) sind zudem immer starke Abweichungen in den Wochenendtagen vorhanden, was daran zu sehen ist, dass die roten Vorhersagen deutlich von den schwarzen realen Werten abweichen. Ab August bis zum Ende des Prognosezeitraums 01.04.2023 sind die Prognosefehler wesentlich geringer aber die Abweichungen werden in den Winterferien bzw. zum Jahreswechsel wieder größer, was auch die stark negativen Einzelergebnisse der Scores verursacht hat.

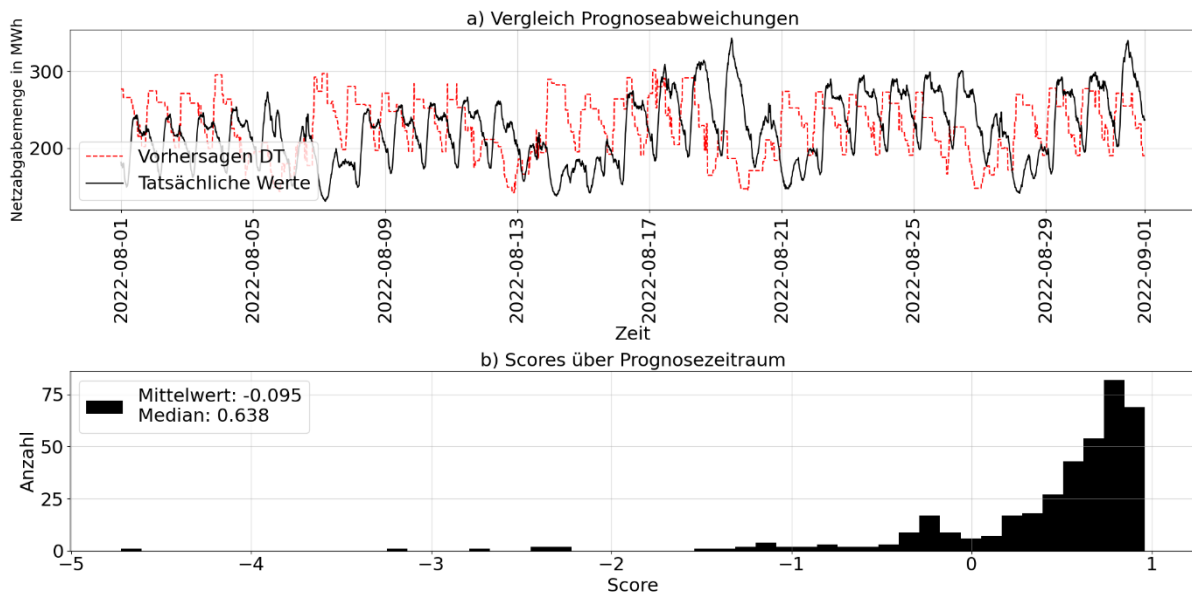


Abbildung 19: Ergebnis DT mit Vorhersagewerte und Test-Scores

Die kumulierten BEC in der Abbildung 20 aus den produzierten Prognosefehlern belaufen sich für Zeitraum vom 22.02.2022 bis zum 01.04.2023 auf 398,64 Mio.€, dabei sind im Durchschnitt je Viertelstunde MBEC in der Höhe von 10.326,97 € angefallen. Zu Beginn des Jahres sind die Ausreißer der BEC sehr stark aber zwischen September 2022 und Ende März 2023 sind dabei weniger Ausreißer von BEC zu verzeichnen gewesen.

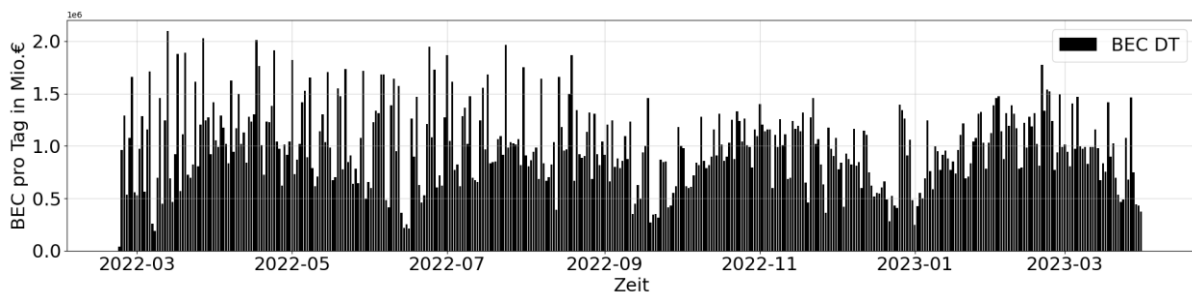


Abbildung 20: BEC DT pro Tag mit 8 Wochen Trainingsdaten

In der Tabelle 9 sind alle BEC und MBEC aufgelistet und es ist zu erkennen, dass beim Decision Tree durch die höhere Anzahl an Trainingsdaten immer bessere Resultate erzielt wurden und jene mit 8 Wochen am besten abgeschnitten haben. Eventuell müssten ab hier die Parameter erneut untersucht und angepasst werden, um mit einer größeren Trainingsmenge gute Ergebnisse erzielen zu können.

Trainingsdaten	BEC in Mio.€	MBEC in €
1 Woche	401,83 (400,22)	10.409,47 (10.367,79)
3 Wochen	402,26 (402,62)	10.420,59 (10.430,09)
4 Wochen	401,58	10.403,15
8 Wochen	398,64	10.326,97

Tabelle 9: Ergebnisse DT, BEC und MBEC

Die Ergebnisse die mit den Gewichtungen erzielt worden sind, sind in der oberen Tabelle in Klammer dargestellt. Die Ergebnisse mit drei Wochen an Trainingsdaten waren hierbei schlechter als in der Ausführung mit einer Woche. Der Gesamtvergleich folgt in Kapitel 4.

3.4 Ergebnisse RF

Die Grid-Search beim Random Forest haben den besten Wert bei einer maximalen Baumtiefe von 4 ermittelt. Der Parameter "max_depth" steuert die maximale Anzahl von Ebenen oder Entscheidungsknoten in jedem einzelnen Entscheidungsbaum im Random Forest. Wenn "max_depth" beispielsweise auf den Wert 4 gesetzt wird, kann jeder Baum im Random Forest höchstens vier Ebenen haben. Im Vergleich zu dem Decision Tree Regressor fällt hier das Ergebnis für eine höhere Parameter-Häufigkeit eindeutiger aus, was in der Tabelle 10 mit dem Wert 4 als bestes Ergebnis zu sehen ist. 42 Mal konnte in diesem Fall mit dem Parameterwert 4 der beste Score erzielt werden, wobei der mögliche Bereich der Auswahl wie beim DT von 1 bis 30 eingeschränkt war.

Max. Tiefe	4	3	7	5	6	11	13	9	8	16
Häufigkeit	42	29	29	24	19	17	16	16	16	16

Tabelle 10: Grid-Search RF, Top 10 Parameter-Häufigkeit

Die Ergebnisse sind ab einer Entscheidungsbauktiefe von 5 ziemlich identisch und unverändert. Was in erster Betrachtung nicht eindeutig erscheint, hat aber im Bereich der Streuung und des Maximalwertes einen entscheidenden Einfluss. Dabei ist mit dem Parameter vermutlich ein guter Score mit einer geringen Streuung nach unten zu erreichen.

Mit dem gesetzten Parameter von 4 konnte beim RF das beste Ergebnis in der Prognose mit einer Trainingsanzahl von einer Woche erzielt werden. Das Resultat in Abbildung 21 ähnelt dabei sehr dem Graphen in Abbildung 20 vom Decision Tree Regressor, was nicht überrascht, da der Random Forest ebenfalls aus Entscheidungsbäumen besteht. Die Prognose lag auch im Zeitraum des Jahreswechsels 2022/2023, wo die BEC am sinken waren bzw. einen

niedrigeren Stand erreichen sollten, deutlich über dem realen Verbrauch wodurch hohe BEC entstanden.

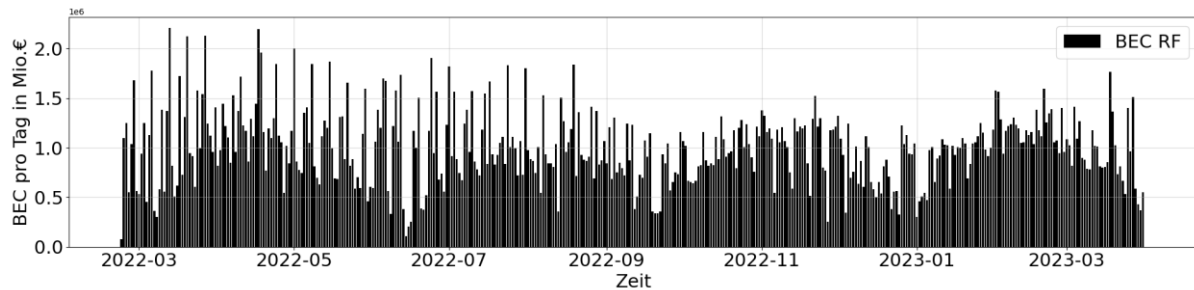


Abbildung 21: BEC RF pro Tag mit 8 Woche Trainingsdaten

Trainingsdaten	BEC RF in Mio.€	MBEC RF in €
1 Woche	388,52 (388,24)	10.064,88 (10.057,40)
3 Wochen	392,99 (392,83)	10.180,52 (10.176,36)
4 Wochen	397,75	10.303,74
8 Wochen	405,06	10.493,16

Tabelle 11: Ergebnisse RF, BEC und MBEC

Was in diesem Fall unerwartet ist, sind die schlechter werdenden Resultate mit steigender Anzahl an Trainingsmengen. RF ist somit die einzige Methode, bei der mit einer geringeren Menge an Trainingsdaten bessere Ergebnisse erzielt werden konnten. Dasselbe Schema ist bei den Ausführungen mit der gewichteten Ausführung zu erkennen. Diese haben zwar ein gering besseres Ergebnis erzielt aber als einziger Algorithmus wurden hier die Resultate mit steigenden Trainingsmengen schlechter.

3.5 Ergebnisse SVR

Für die folgenden erzielten Resultate der Support Vector Regression wurde für den Parameter $C = 1 \times 10^7$ verwendet und die Auswahl erfolgte, wie zuvor bei DT und RF, auf Basis des Grid-Search. Mit diesem Parameter konnten mit Abstand die besten Resultate erzielt werden, doch wie in Abbildung 22 zu sehen, gibt es einzelne Ergebnisse die weit in den negativen Score nach unten streuen. Nichtsdestotrotz sind 264 beste Scores mit $C = 1 \times 10^7$ erzielt worden.

C	1×10^1	1×10^2	1×10^3	1×10^4	1×10^5	1×10^6	1×10^7	1×10^8
Häufigkeit	0	0	1	2	15	79	264	72

Tabelle 12: Grid-Search SVR, Top 8 Parameter-Häufigkeit

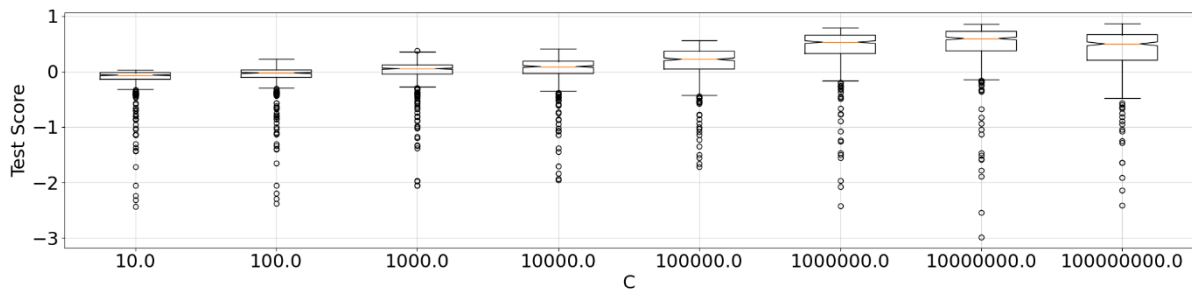


Abbildung 22: Grid-Search Support Vector Regression, Parameter C

Mit dem Parameter C sind mit den genannten Spezifikationen des Computers, für eine Berechnung mit Sliding-Window-Verfahren über einen Zeitraum von mehr als einem Jahr, mit Durchlaufzeit von bis zu neun Stunden zu rechnen. Durch die Schleife über alle Tage soll aber aufgezeigt werden, wie performant der Algorithmus mit den gegebenen Features ist. Wird die Berechnung nur für einen einzelnen Tag ausgeführt, wie es in der Regel auch gemacht wird, so minimiert sich die Laufzeit drastisch. Trotz der langen Laufzeiten wurden Ergebnisse ausgewertet, welche mit steigender Anzahl an Trainingstagen immer besser wurden und somit konnte mit 8 Wochen, einem BEC von 393,92 Mio.€ und einem R^2 Median von 0,626 (Mittelwert -0,163) beste Ergebnis erzielt werden.

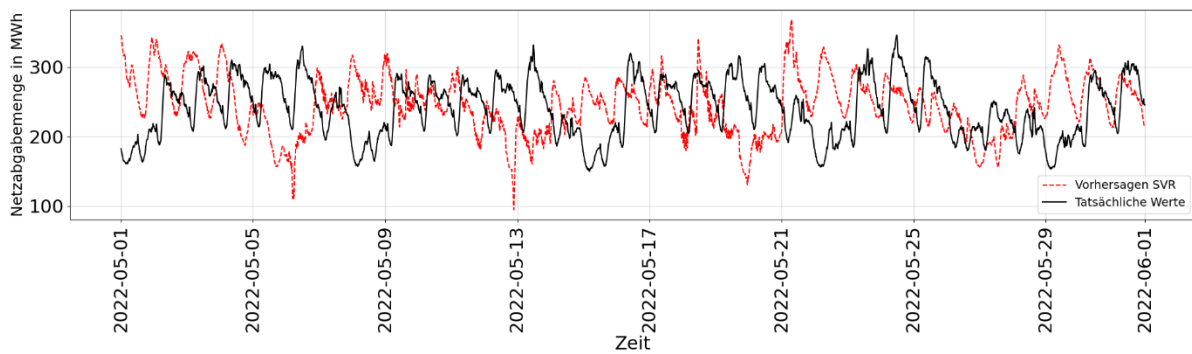


Abbildung 23: Ausschnitt Prognosedaten SVR mit 8 Wochen Trainingsdaten

Trainingsdaten	BEC SVR in Mio.€	MBEC SVR in €
1 Woche	425,63	11.026,03
3 Wochen	401,52	10.402,42
4 Wochen	403,66	10.457,05
8 Wochen	393,92	10.204,59

Tabelle 13: Ergebnisse SVR, BEC und MBEC

Bei den Vorhersagen des SVR gibt es in den Monaten Mai, November und Dezember 2022, sowie Jänner 2023 mit einer Trainingsmenge von acht Wochen außerordentlich starke Abweichungen. In Abbildung 23 sind diese Abweichungen, sowohl zu niedrig als auch zu hoch prognostizierte Daten, ersichtlich.

3.6 Ergebnisse LR

Für die Lineare Regression von Scikit-Learn wurde kein Grid-Search ausgeführt, da es dafür keine Parameter gibt, die durch ein Grid-Search zu finden wären. Somit wurden mit der LR in ihrer Standardform Prognosen erstellt. Die Ergebnisse sind dabei sehr vielversprechend ausgefallen und konnten im Vergleich zu den anderen Methoden den niedrigsten BEC (377,56€) erzielen. Dabei ist der Unterschied zwischen einer Woche und drei Wochen an Trainingsdaten gravierend. Die LR erzielte einen Median beim R^2 von -0,047 und einen Mittelwert von -0,763.

Trainingsdaten	BEC LR in Mio.€	MBEC LR in €
1 Woche	462,56	11.982,73
3 Wochen	383,22	9.927,58
4 Wochen	380,36	9.853,38
8 Wochen	377,56	9.780,73

Tabelle 14: Ergebnisse LR, BEC und MBEC

Grund für diesen enormen Unterschied ist eine große Abweichung, die in Abbildung 24 visualisiert ist, die zwischen 18.07.2022 21:00 Uhr und 19.07.2022 09:00 Uhr zu verzeichnen war. Was diese Abweichung erzeugt hat, kann nicht erklärt werden und ist vermutlich eine Anomalie, die der Linearen Regression zu verschulden ist.

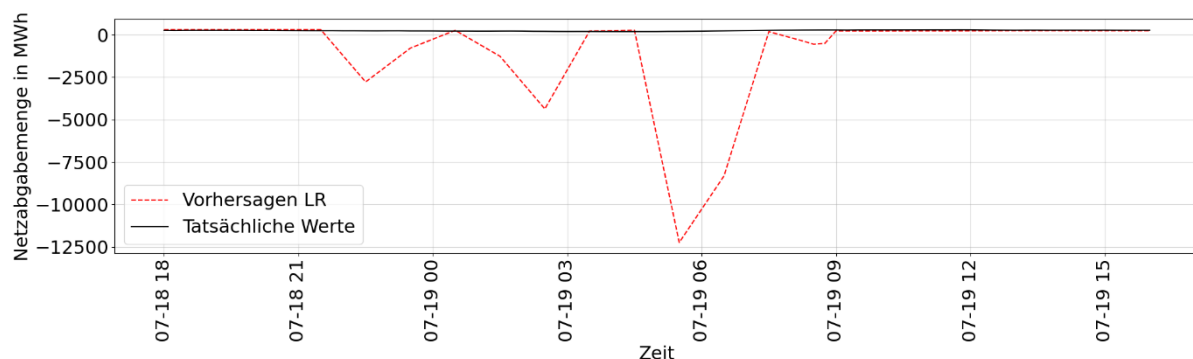


Abbildung 24: Prognosefehler LR

3.7 Ergebnisse Gurobi

Mit den gegebenen Entscheidungsvariablen und Nebenbedingungen konnte die Methode einen Bestwert von 383,47 Mio.€ bei 8 Wochen an Trainingstagen erzielen. Die Durchlaufzeit betrug dabei etwa fünf Minuten. Es trat zudem dieselbe Anomalie zum selben Zeitpunkt, bei einer Trainingsmenge von einer Woche, wie bei der LR auf aber mit einem größeren Fehler, was in Abbildung 25 dargestellt ist.

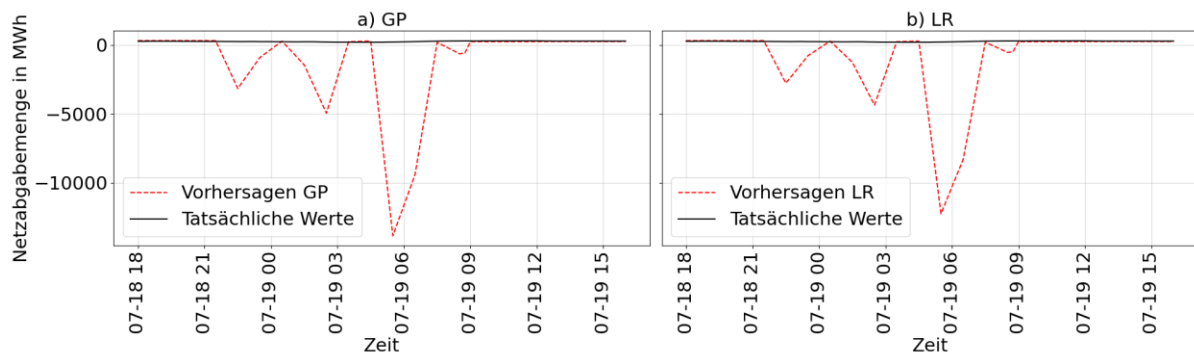


Abbildung 25: Prognosefehler Vergleich GP und LR

Bei der Überprüfung der Features konnten keine Ausreißer oder außergewöhnliche Ereignisse in diesem Zeitraum festgestellt werden, sodass wie bei der LR keine Rückschlüsse gezogen werden können, wie dieser enorme Prognosefehler zustande kam. Auch bei mehreren Durchläufen einer Prognose mit einer Trainingsmenge von einer Woche, sowohl bei der LR als auch beim GP, wurde immer derselbe Fehler erzeugt. Die darauffolgenden Ergebnisse, mit einer höheren Menge an Trainingsdaten, wiesen keine derartigen Ausreißer mehr auf.

Trainingsdaten	BEC GP in Mio.€	MBEC GP in €
1 Woche	495,94	12.847,47
3 Wochen	389,25	10.083,56
4 Wochen	386,81	10.020,55
8 Wochen	383,47	9.933,98

Tabelle 15: Ergebnisse GP BEC und MBEC

Das eigens erstellte Gurobi-Modell konnte keine besseren Ergebnisse als das LR-Modell erzielen, wobei die Resultate sehr nahe beieinander liegen. Die BEC pro Tag in Abbildung 26 mit einer Trainingsmenge von acht Wochen sind dabei konstant, bis auf einzelnen Ausreißer

bis in das zweite Drittel des Jahres 2022. Werden diese summiert, so ergibt sich das Ergebnis in Tabelle 15.

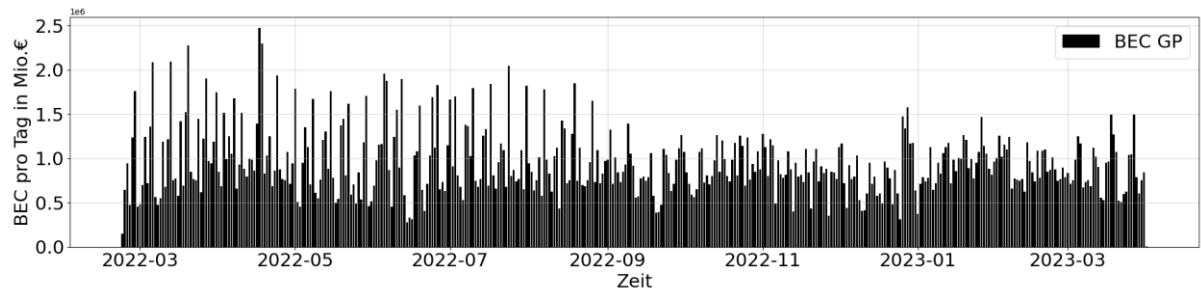


Abbildung 26: BEC GP pro Tag mit 8 Woche Trainingsdaten

4 Ergebnisbewertung und -vergleich

Hier erfolgt die Gesamtbetrachtung aller Algorithmen und deren Ergebnisse mit den unterschiedlichen Mengen an Trainingsdaten. Bisher wurden dagegen in den Einzelergebnissen der Methoden nur jeweils einzelne Ereignisse betrachtet. Hier wird beschrieben, wie sich die neu implementierte Bewertungsmethode im Vergleich zu den bewährten Methoden verhält und wie sich die preisgewichtete Optimierung durch Gurobi im Vergleich zu den anderen Algorithmen positioniert hat. Zudem werden die Kennzahlen der Prognosen zu den verwendeten Trainingstagen dargestellt. Zusammengefasst wurden nun, mit den Methoden ANN, DT, GP, LR, MLP, RF und SVR, Prognosen mit jeweils 1, 3, 4 und 8 Wochen an Trainingsdaten für über 400 Tage erstellt. Somit wurden insgesamt über 1,075 Millionen Viertelstundenwerte berechnet. Da die Prognosen über einen langen Datenzeitraum ausgeführt wurden, ist eine Visualisierung der Daten eines jeden Algorithmus unübersichtlich. Die Darstellung in Abbildung 27 soll hingegen mit dem MAPE für jede Viertelstunde eines Tages verdeutlichen, wie groß die prozentuale Schwankung der Prognoseabweichungen je Methode ist. Der Unterschied der verwendeten Trainingsdaten wird hierbei umso ersichtlicher, denn mit steigender Anzahl, nimmt die Differenz der Prognosefehler zwischen den Methoden stark ab.

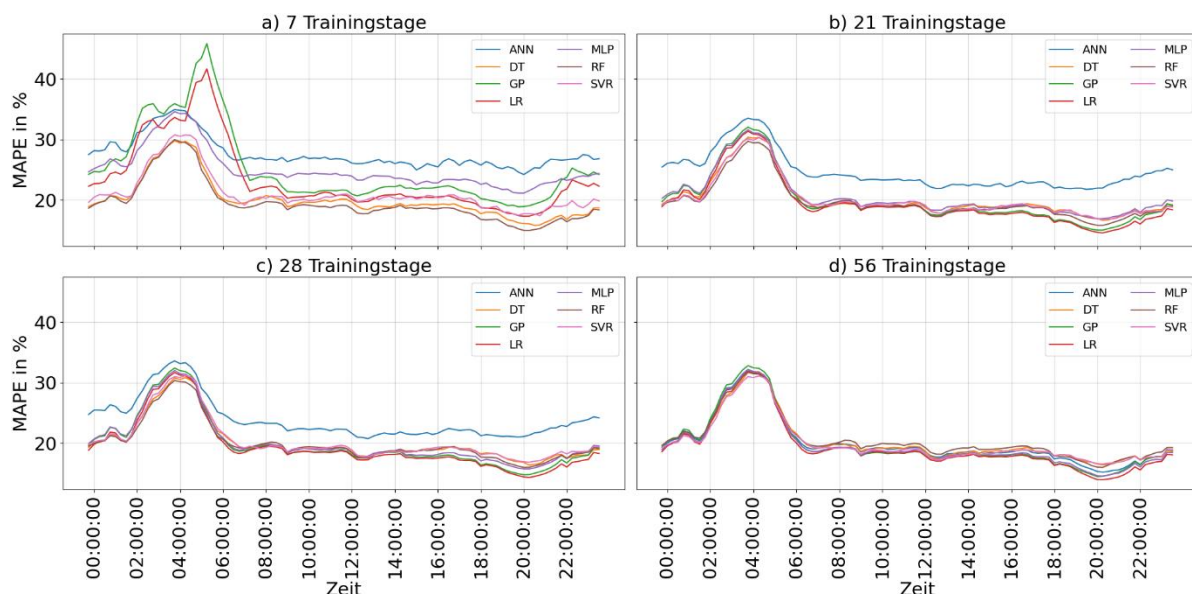


Abbildung 27: Prognosefehler je Tageszeit und Trainingstage

Die Verwendung von mehr Trainingsdaten zeigt somit die Wichtigkeit einer sorgfältigen Auswahl auf. Hier ist die größte Veränderung der erzeugten Prognosefehler von einer Woche zu drei Wochen an Trainingsdaten zu bemerken und je mehr Daten verwendet werden, umso schmaler wird der Bereich, indem sich Fehler befinden. Im Tagesverlauf zwischen 02:00 Uhr

und 06:00 Uhr ist ein deutlicher Ausreißer zu verzeichnen, welcher sich durch die zuerst fallenden Netzabgabemengen in Abbildung 5 in dieser Zeitspanne erklären lässt. Somit ist zwischen diesen Tageszeiten mit einem höheren BEC zu rechnen, als zu den Zeiten mit niedrigerem MAPE, was in Abbildung 28 ersichtlich ist. Stellt man den Vergleich mit den MBEC her, so ist die Kostenverteilung des Tages erkennbar. Dabei zeichnet sich der BEP-Verlauf des Tages ab und es ist zu sehen, dass zu den höherpreisigen Tageszeiten höhere MBEC entstanden sind. Resultierend aus den oberen Prognosefehlern sind die MBEC mit den geringsten Trainingsmengen am höchsten. Bis auf die Random Forest Regression, welche mit einer Trainingsmenge von 7 Tagen das beste BEC und MBEC Ergebnis erzielen konnte. Durch die hohen Prognosefehler zwischen 02:00 Uhr und 06:00 Uhr sind auch im Falle von niedrigeren Ausgleichsenergiepreisen erhöhte MBEC entstanden.

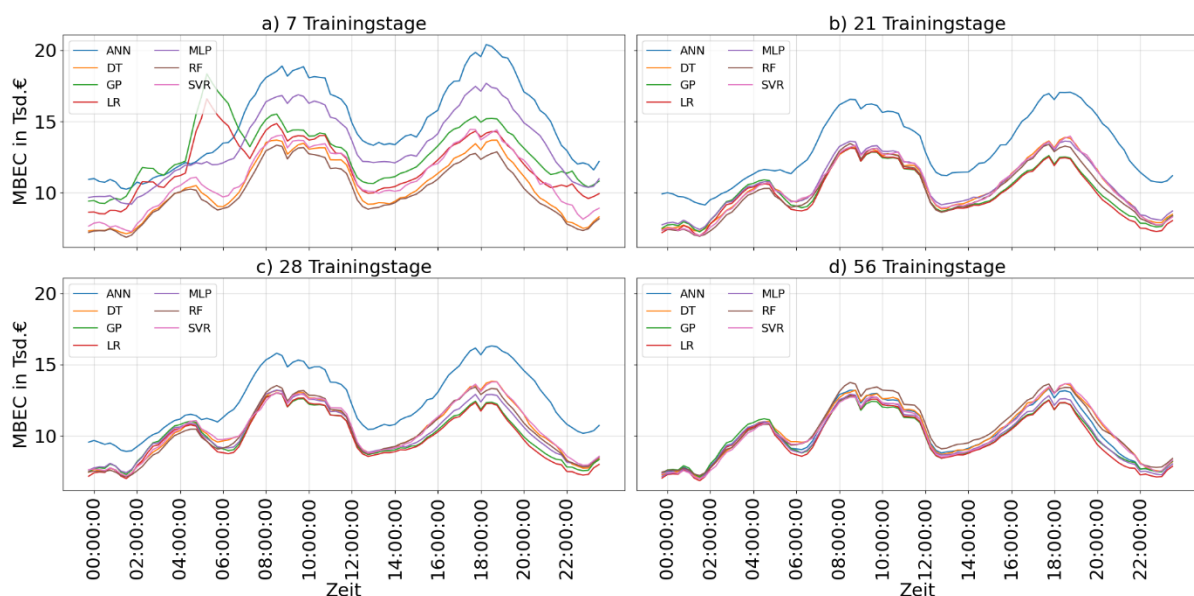


Abbildung 28: BEC je Tageszeit und Trainingsdaten

In der Analyse der Algorithmen haben der Decision Tree Regressor, Random Forest und Support Vector Regression, im Segment mit der niedrigsten Anzahl an Trainingsdaten, im Vergleich zu ANN und MLP ein ebenfalls gutes Ergebnis erzielen können. Das liegt daran, dass Entscheidungsbäume auch bei fehlenden Daten stark performen können und die SVR robuster gegenüber Ausreißern ist, obwohl bei einer geringeren Trainingsmenge Ausreißer normalerweise schwerer ins Gewicht fallen. Die ANN und MLP konnten aber mit steigender Anzahl an Trainingsdaten ihre Prognosen stetig verbessern, wobei DT sich in der Verteilung nur bedingt verbessert hat. Es stach vor allem die Methode der ANN heraus, die auch mit 4 Wochen Trainingsdaten starke Abweichungen zwischen über den Tagesverlauf aufwies und konstant von den Resultaten der anderen Methoden abwich. Schlussendlich konnten mit 8

Wochen Trainingsdaten die Prognosefehler stetig verkleinert werden, doch die Ergebnisse haben noch Verbesserungspotenzial.

Die Betrachtung der Prognosefehler gilt aber nicht nur im Falle der BEC und MBEC, denn diese müssen nun neben den bewährten Bewertungsmethoden beweisen, dass auch diese plausibel und aussagekräftig sind. Dabei soll die Darstellung in Abbildung 29 den Vergleich der Prognosegüte darstellen. Für die R^2 Kennzahl wurde hierbei der Median verwendet, denn wie im Kapitel 3 die Ergebnisse zeigten, sind einzelne starke negative Abweichungen bei jeder Methode vorhanden, welche den Mittelwert beeinflussen. Nichtsdestotrotz haben ANN, LR und MLP keine positiven Werte erreichen können, was darauf hindeutet, dass die Menge an Daten, die den Algorithmen für das Training zur Verfügung gestellt wurden, zu wenig waren oder die Netzabgabemengen zu wenig mit den Features korrelierten. Es ist aber zu erkennen, dass diese vermutlich mit mehr Trainingsdaten ein positives R^2 Resultat erzielen hätten können, da der Trend stetig nach oben ging. Im Vergleich zu BEC und MBEC schlugen sie sich wiederum gut und sind mit 8 Wochen bzw. mit 56 Trainingstagen auf dem Niveau von DT, GP, RF und SVR.

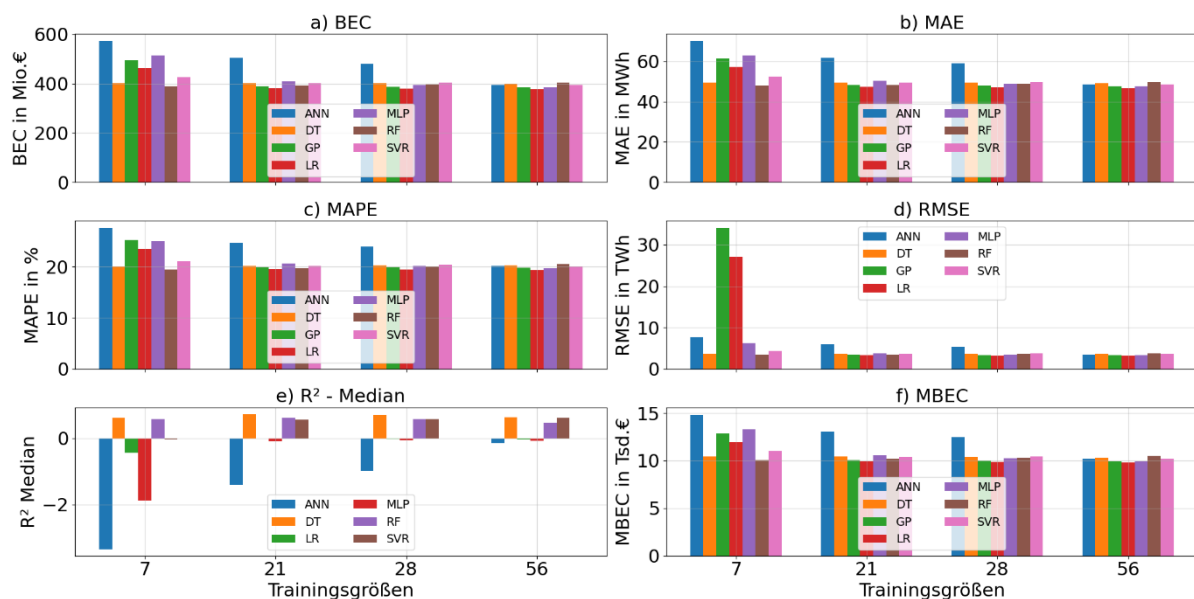


Abbildung 29: Vergleich Prognosegüte Kennzahlen Gesamt

Werden die Kennzahlen untereinander verglichen, speziell zwischen den bewährten Methoden und den neu implementierten BEC und MBEC, so stellt keine dieser einen deutlichen Unterschied zwischen den Trainingsgrößen dar. Diese Verdeutlichung ist im Falle des RSME und der Algorithmen GP und LR von einer Woche Trainingsdaten umso hilfreicher, denn dieser verzeichnete klar die großen Ausreißer beider Methoden. Beim R^2 schneiden LR, MLP und ANN weitaus schlechter ab als bei den anderen Kennzahlen. Trotzdem können im

Falle eines Kostenschemas, wie durch die BEP, mit den anderen Kennzahlen keine direkten Rückschlüsse auf eine Kostenauswirkung getroffen werden.

Neben der Betrachtung des gesamten Prognosezeitraums werden die Ergebnisse nach ihren Jahreszeiten verglichen. Im Kapitel 2.1.3 der Wetterdaten wird aufgezeigt, dass die Korrelationen zu den Netzabgabemengen im Frühling, Sommer, Herbst und Winter unterschiedlich sind. Für diese Betrachtung dient die Abbildung 30 mit den MBEC, welche die Durchschnittskosten je Viertelstunde darstellen. Zu Hochsaisonzeiten sowie zu Übergangszeiten sind doch Unterschiede zu erkennen. Dabei sind mit mehr Trainingsdaten, wie bei den zuvor genannten Gesamtergebnissen, weniger Kosten entstanden. Dabei sind in den kälteren Monaten des Herbstes und des Winters bessere Ergebnisse erzielt worden, als in denen im Frühling und im Sommer.

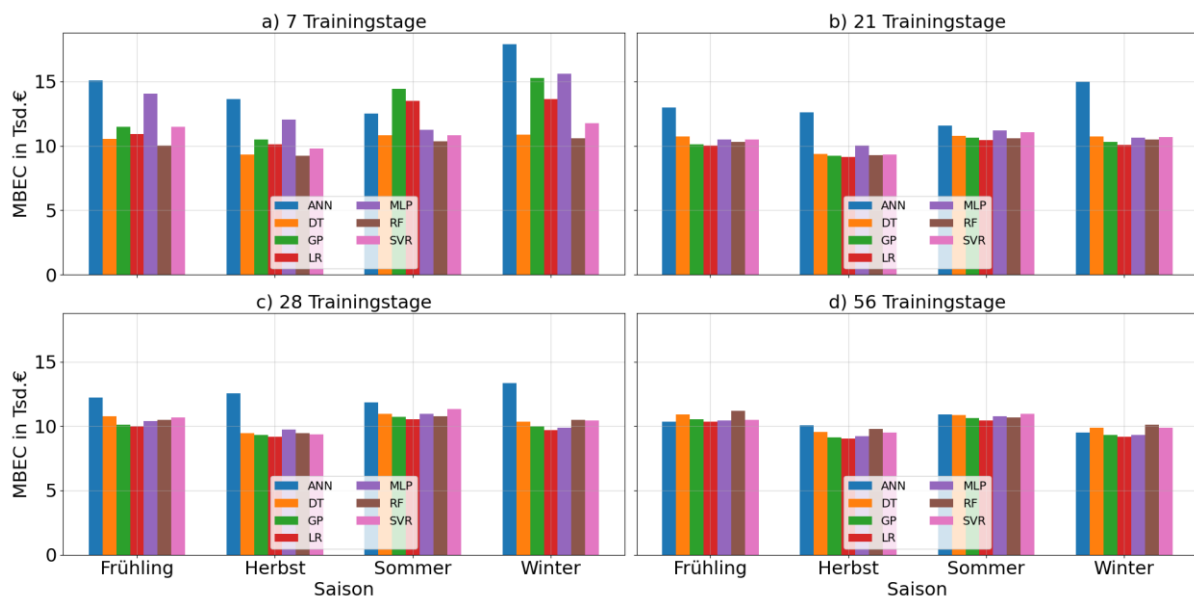


Abbildung 30: Saisonaler Vergleich MBEC

Das Verhalten jedoch ist bei allen Algorithmen im saisonalen Zusammenhang überwiegend gleich und sie sind somit je Saison und Methode positiver oder negativer. Doch wieso ist das so und können diese Ergebnisse mit den Wetterdaten zusammenhängen? Vergleicht man die Korrelation zu den Wetterdaten in Tabelle 3, so wiesen der Herbst und Winter mehr negative Korrelationen auf, wie der Frühling und der Sommer. Der RMSE zeigt bei den 8 Wochen an Trainingsdaten die Zusammenhänge umso deutlicher. Denn je positiver die Korrelation zu den Wetterdaten, umso schlechter ist das Ergebnis. Die anderen Kennzahlen, wo sich die tabellarischen Darstellungen im Anhang befinden, weisen ebenfalls derartige Differenzen zwischen den Saisonzeiten auf.

Da sich die Netzaufgabenmengen auch an den Wochentagen unterscheiden und in den Features berücksichtigt wurden, ist auch hier der Ergebnisvergleich in Abbildung 31 visualisiert, um etwaige Differenzen aufzeigen zu können. Die Ergebnisse der MBEC weisen starke Unterschiede zwischen den einzelnen Wochentagen auf (0 = Montag, 1 = Dienstag, 3 = Mittwoch, 4 = Donnerstag, 5 = Freitag, 6 = Samstag, 7 = Sonntag), dabei wich der Sonntag generell von den erwarteten Ergebnissen ab. Hierzu wurde in einem Interview mit dem Energiehandel der illwerke vkw AG erwähnt, dass die Herausforderung einer korrekten Vorhersage von Wochenenden höher ist, da sich diese Netzaufgabenmengen von den anderen Tagen stark unterscheiden. Die Algorithmen können somit nur schlecht von den Wochentagen unterscheiden und weisen mehrheitlich den Tagen Montag bis Samstag die größere Relevanz zu als dem Sonntag. Scheinbar hatte die Berücksichtigung der Wochentage in den Features nicht die gewünschte Wirkung erzielt. Wieso der Sonntag aber vermehrt höhere MBEC oder BEC als Ergebnis hatte, war, weil die Netzaufgabenmengen an den Wochentagen höher sind als an den Wochenenden und somit das Problem entstand, dass größtenteils zu viel Energie prognostiziert wurde.

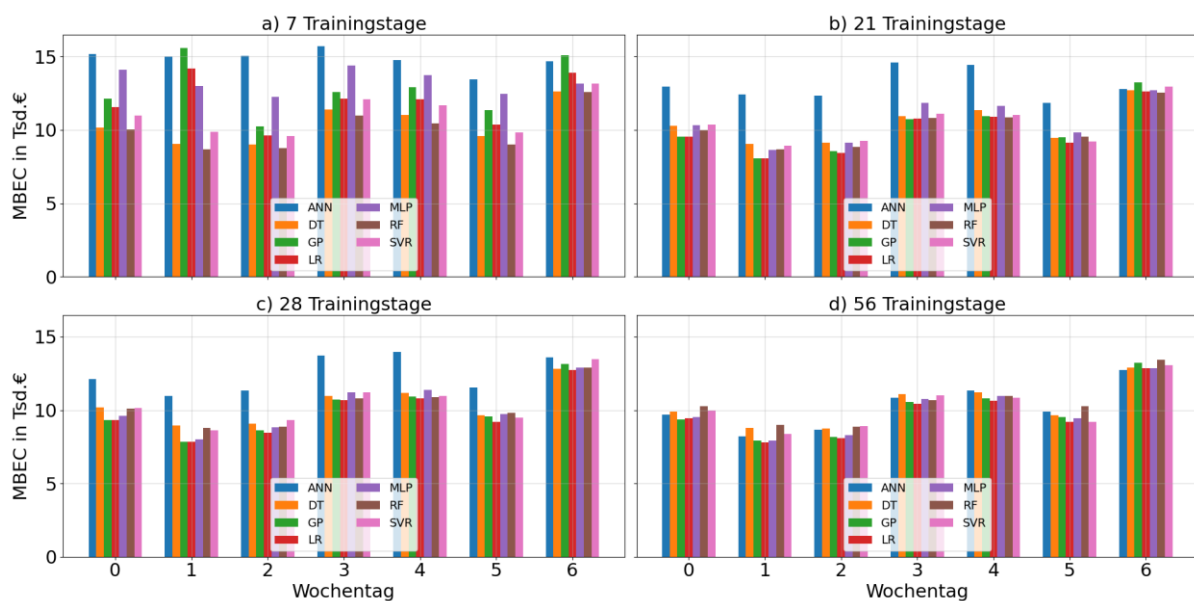


Abbildung 31: Wochentag-Vergleich MBEC

Vergleicht man die Ergebnisse nun mit MAE oder RMSE, so sind diese nicht viel unterschiedlicher als erwartet. Sowohl beim MAE als auch beim RMSE unterscheidet sich der Sonntag wie das auch bei BEC oder MBEC der Fall war. Der RMSE zeigt dabei noch deutlicher die Prognosefehler auf. Auf alle Fälle kann nach den Beurteilungen festgestellt werden, dass die Prognosen für den Sonntag schlechtere Ergebnisse erzielen, als wie für die anderen Tage. Wieso aber der Samstag ebenfalls gute Resultate erzielen konnte und nicht

wie der Sonntag sich von den anderen Tagen unterscheidet, kann nur damit erklärt werden, dass sich die Netzabgabemengen zu Beginn dieses Tages gleich verhalten, wie die Wochentage Montag bis Freitag. Zu den restlichen Stunden des Tages ist im Durchschnitt, wie beim Sonntag, viel weniger Netzabgabemenge zu verzeichnen und so wäre auch hier ein ähnliches Resultat wie jenes vom Sonntag zu erwarten gewesen.

Auch die Ausführung mit der Gewichtung auf DT und RF hatte einen Einfluss auf den BEC bzw. MBEC. Vergleicht man diese mit den zwei Algorithmen in Abbildung 32, bei denen die Gewichtung mit BEP hinzugekommen ist, so sind keine signifikanten Unterschiede zu erkennen. Durch die Gewichtung sank der BEC beim DT mit der Ausführung von 7 Trainingstagen um 0,4% und im Vergleich bei 21 Trainingstagen stieg das Ergebnis nur um 0,08% an. Beim dem RF konnten mit der Gewichtung immer bessere Resultate erzielt werden. Diese lagen aber ebenfalls bei einer geringen Differenz von 0,07% bei 7 Trainingstagen und 0,04% mit einer Anzahl von 21 Trainingstagen als Trainingsmenge. Zudem muss beachtet werden, dass diese Gewichtung nicht derselben entspricht, wie sie in der gewichteten Optimierung bei GP angewendet wurde. Denn das GP-Modell minimiert den gewichteten absoluten Prognosefehler, wogegen die Gewichtung über DT und RF über den quadratischen Fehler ausgeführt wird.

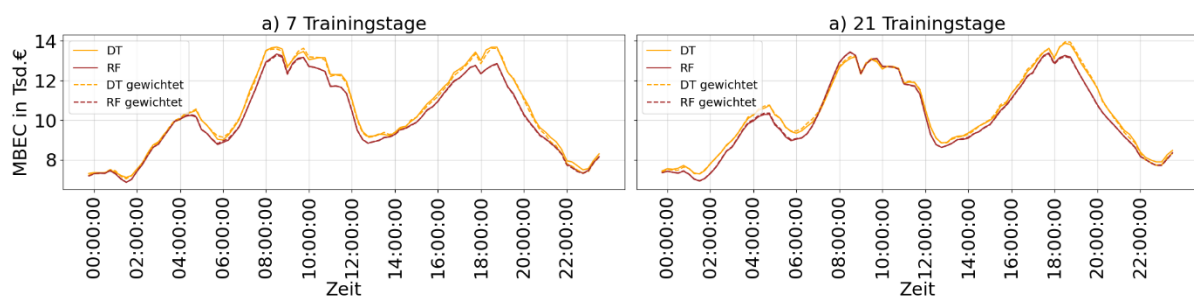


Abbildung 32: Vergleich Gewichtung durch BEP

Da es sich bei GP und LR bei beiden um eine Lineare Regression handelt, wurde auch hier ein Vergleich, in Abbildung 33, erstellt. Dabei schnitt die LR Methode mit geringem Abstand besser ab als das GP-Modell. Da das GP-Modell eine preisgewichtete Optimierung darstellt, wäre zu erwarten gewesen, dass der GP bessere Ergebnisse erzielt, als die Vorhersage mit LR, zumal die Minimierung des BEC bzw. MBEC über GP das Ziel war. Die LR hat mit 8 Wochen Trainingsdaten einen BEC von 377,56 Mio.€ erzielt und GP einen BEC von 383,47 Mio.€, also eine Differenz von 5,91 Mio.€ bzw. 1,54 %, womit GP das zweitbeste Ergebnis erzielt hat.

In Abbildung 33 ist ersichtlich, dass die LR zu den niederpreisigen Tageszeiten den Abstand zum GP-Modell erhöht hat aber zwischen 08:00 Uhr und 12:00 Uhr die MBEC von GP leicht niedriger sind, als die von der Linearen Regression von Scikit-Learn.

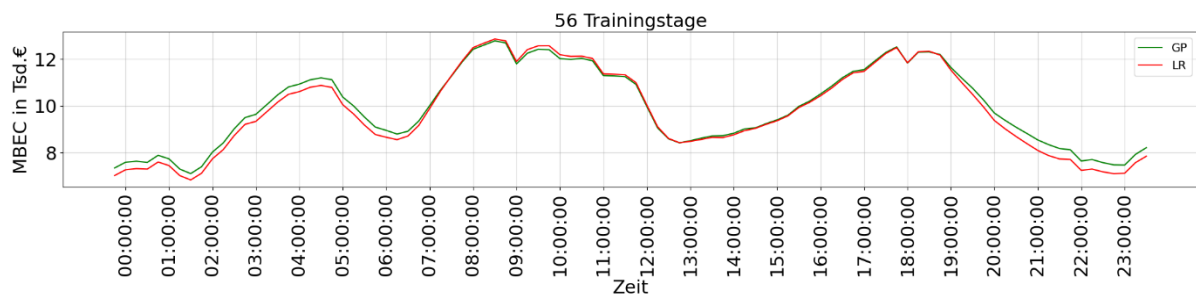


Abbildung 33: Vergleich Gurobi-Modell mit Linearer Regression

Zuletzt wird noch der beste und schlechteste Prognosetag, also der Tag an dem die Summe aller Prognosefehler von den Algorithmen am niedrigsten war, in Abbildung 34 verglichen. Dabei ist zu erkennen, dass die Methoden in diesen zwei Darstellungen nicht weit auseinander liegen, sowohl am Besten als auch am schlechtesten Prognosetag. Der Unterschied zwischen a) 14.06.2022 und b) 17.04.2022 liegt bei etwa 700MWh bzw. 89%, was sich letztenendes bei den BEC auswirkt.

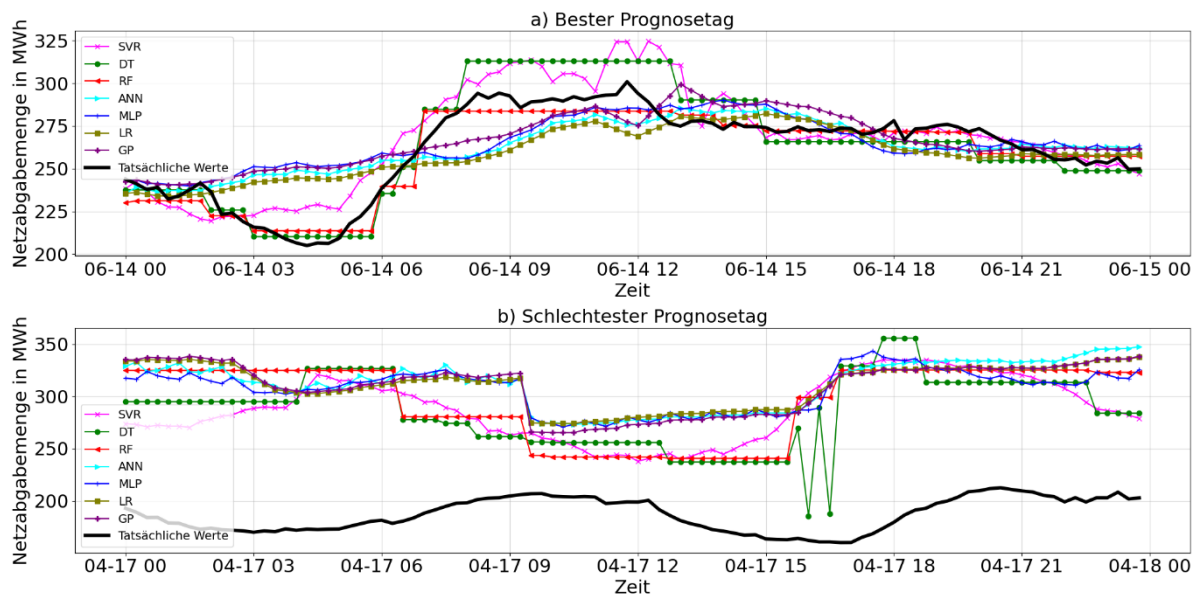


Abbildung 34: Prognosefehler - Bester und schlechtester Fall

Dabei überraschte es nicht, dass das schlechteste Ergebnis am 17.04.2022, an einem Sonntag, erzielt wurde, bei dem in Abbildung 31 die größten MBEC an diesem Wochentag nachgewiesen werden konnten. Das beste Ergebnis aus Sicht der Prognosefehler konnte dabei am Dienstag, den 14.06.2022, erzielt werden.

Alle Ergebnisse der Kennzahlen sind in Tabellenform im Anhang zur Verfügung gestellt.

4.1 Ergebnisinterpretation

Alle BEC und MBEC Ergebnisse, bis auf jene des RF, konnten mit der Erhöhung der Trainingsdaten stetig verbessert bzw. reduziert werden. Das Resultat sollte dabei so gering wie möglich sein, um als gut quantifiziert werden zu können. Zu hohe Ergebnisse weisen darauf hin, dass in der Regel zu wenig oder zu viel Netzabgabemenge prognostiziert wurde. In Abbildung 35 ist dies noch einmal für alle Algorithmen verdeutlicht. Zu sehen sind zu Beginn der Prognose im Jahr 2022 stark schwankende BEC pro Tag, welche kumuliert die Gesamtergebnisse wiedergeben.

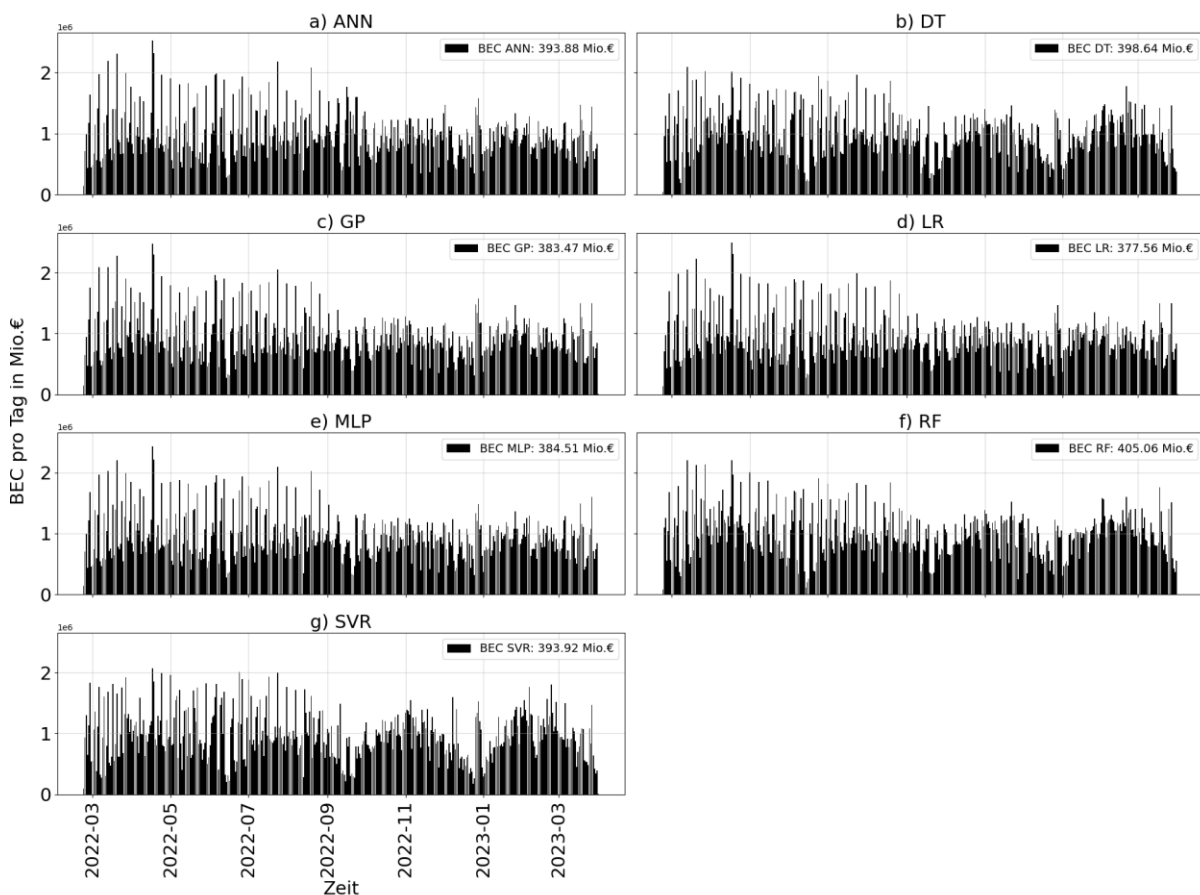


Abbildung 35: BEC Jahresverlauf Methodenvergleich 8 Wochen Trainingsdaten

Es ist zu erkennen, dass die BEC über den Prognosezeitraum Erhebungen abbildet, welche der Netzagbabemengen in Abbildung 3 gleich kommt. Bei allen Algorithmen sind bis Juli des Jahres 2022 etwa gleiche Muster zu erkennen, doch ab Mitte August werden die Unterschiede zwischen den Methoden immer größer und dabei schneidet die LR und GP am besten ab. Die vergleichsweise schlechten Ergebnisse von ANN zu MLP sind in der Abbildung 35 a) und e) wiederzufinden. Der Grund sind die hohen BEC zwischen September und Dezember 2022. Doch warum diese nicht an die Resultate von MLP herankommen liegt vermutlich an der noch möglichen Parametererweiterung des ANN, wobei der Unterschied dieser zwei Algorithmen beim BEC bei 2,38 % liegt. Denn es ist zu beachten, dass Tensorflow ein flexibles Framework ist und verschiedene Möglichkeiten bietet, das Training zu steuern und anzupassen. Je nach Modellarchitektur und spezifischer Implementierung können auch weitere Parameter verwendet werden, um das Training zu regulieren und zu steuern. Die einfache Implementierung des MLP ist dabei vorteilhaft. In der Abbildung 34 ist erkennbar, dass die beiden verwandten Algorithmen ANN und MLP zwar nahe beieinander liegen, doch die Gesamtergebnisse des BEC und MBEC beim ANN höher sind als beim MLP. Zudem hat das ANN Modell von Tensorflow bei allen Durchführungen am schlechtesten abgeschnitten. Trotz der Verwendung von den Parametern aus der Fallstudie, welche ebenfalls für eine Lastprognose verwendet wurden, funktionierte dies scheinbar nicht auf der bestehenden Datenbasis dieser Arbeit. Da der Aufbau beim ANN komplexer als wie beim MLP Modell ist, sind die Resultate dieses Algorithmus besser. Es müssten die Parameter bei dem Modell von Tensorflow noch nachjustiert und weiter getestet werden. In der Betrachtung des MAPE, RMSE oder MAE konnten die zwei Methoden durchaus akzeptable Ergebnisse erzielen und hatten im Vergleich zu (Agyeman u.a. 2020, S. 17) nur eine Differenz von etwa 5% bei der Verwendung von ANN bzw. MLP.

BEC	ANN	DT	GP	LR	MLP	RF	SVR
ANN	x	+1,21 %	-2,64 %	-4,14 %	-2,38 %	+2,84 %	+0,01 %
DT	-1,19 %	x	-3,81 %	-5,29 %	-3,54 %	+1,61 %	-1,19 %
GP	+2,71 %	+3,96 %	x	-1,54 %	+0,27 %	+5,63 %	+2,73 %
LR	+4,32 %	+5,58 %	+1,57 %	x	+1,84 %	+7,28 %	+4,33 %
MLP	+2,44 %	+3,67 %	-0,27 %	-1,81 %	x	+5.34 %	+2.45 %
RF	-2,76 %	-1,58 %	-5,33 %	-6,79 %	-5.07 %	x	-2,75 %
SVR	-0,01 %	+1,20 %	-2,65 %	-4,15 %	-2.39 %	+2,83 %	x

Tabelle 16: BEC prozentuale Ergebnisunterschiede bei 8 Wochen Trainingsmenge

Die Ergebnisse in Tabelle 16 zeigen auf, dass sich die Unterschiede etwa zwischen 1% und 8% befinden. Dabei wurde der prozentuale Unterschied zwischen der Methode aus den

Spalten zu denen aus den Zeilen berechnet. Wird dabei von einem BEC von 400 Mio.€ ausgegangen, so macht das bei einem prozentualen Unterschied von 8%, 32 Mio.€ aus.

Mit dem preisgewichteten Optimierungsmodell des GP konnte leider nicht das Ergebnis erzielt werden, welches zu erwarten war. Dabei konnte die verwandte Methode LR von Scikit-Learn die besseren Ergebnisse erzielen. Speziell bei der Betrachtung des Tagesverlaufs in Abbildung 33 sollte der Unterschied zu den höherpreisigen Tageszeiten größer bzw. besser sein. Wie erwähnt waren die Kosten zwischen 08:00 Uhr und 12:00 Uhr zwar leicht geringer, dies lässt jedoch nicht deutlich darauf zurückschließen, ob nun die preisgewichtete Methode zu den Tageszeiten mit höheren BEP besser ist. Grund dafür könnten die Preise selbst sein, da sie nach der Glättung in Relation zu nahe beieinander liegen. Dies könnte verhindert werden indem die Preise künstlich hoch oder runter gesetzt bzw. geringe BEP weiter gesenkt und höhere BEP weiter erhöht werden. Ziel dabei sollte sein, die Preise logisch voneinander zu entfernen und ihre Wirkung dadurch zu erhöhen.

Die Ausführung der Methoden mit Gewichtung für den Decision Tree Regressor und Random Forest Regressor konnten keine signifikanten Verbesserungen, mit Testdaten aus einer Menge von 1 und 3 Wochen und im Zeitraum von 22.02.2022 bis 01.04.2023, erzielen. Die Idee mit der Gewichtung durch die Ausgleichsenergiepreise BEP war, dass durch ein Preisschema die Prognosefehler geringer werden und sich mehr an die tatsächlichen Werte annähern, was in dieser Ausführung nicht erreicht werden konnte. Grund dafür könnte die Funktionsweise der „Sample-Weights“ selbst sein, da diese nicht die absoluten Prognosefehler gewichtet.

Durch die Betrachtung der einzelnen Wochentage fiel der starke Unterschied zueinander auf. Dabei wurde der Sonntag im Ergebnisvergleich thematisiert. Warum sich aber die Tage Montag bis Freitag auch so stark unterscheiden kann bis dato nicht begründet werden, auch wenn die Netzabgabemenge an diesen Tagen nahezu ident ist. Dies ließe sich eventuell damit lösen, die Wochentage separat zu betrachten. Denn in dieser Arbeit wurden alle Daten bzw. alle Tage gesamt in den Features herangezogen und für eine zukünftige Lösung könnte der richtige Schritt in der Verwendung von verschiedenen Feature-Files je Wochentag liegen. Dabei müsste die Implementierung so angepasst werden, dass z.B. bei der Prognose eines Donnerstags, auch nur Donnerstage in den Features verwendet werden. Dasselbe gilt für die Hochsaison- und Übergangszeiten. Auch wenn die Differenzen nicht sehr groß waren, könnten die Ergebnisse durch die Entkoppelung der Trainingsdaten der unterschiedlichen Saisonzeiten eventuell verbessert werden. Durch die saisonale Trennung wäre die Anpassung der Ausgleichsenergiepreise BEP ebenfalls ein wichtiger Schritt, welcher durch die Erstellung des Mittelwertes der Saisonpreise aus Abbildung 15 c) und eine daraus resultierende Glättung entsteht. Somit würden die Preise nach den Anpassungen der Features gerecht verwendet werden. Die Anpassung der Preise auf Basis der Wochentage

würde in diesem Fall keinen Sinn machen, da sich diese, wie in Abbildung 15 b) zu sehen, ähnlich verhalten und keine großen Unterschiede zueinander vorhanden sind.

Versucht man nun den Vergleich der Kennzahlen der Prognosegüte zu interpretieren, so ist der MAE, RMSE oder MAPE zwar aussagekräftig, wenn man die Prognosefehler selbst betrachtet, aber nicht, wenn diese in Relation zu Kosten gesetzt werden müssen. In Abbildung 29 ist bei diesen vor allem keine signifikante Verbesserung auf Basis der Trainingsmengen zu erkennen und als Aussage müsste hier getroffen werden, dass die in dieser Arbeit verwendete Menge der Trainingsdaten keinen Einfluss auf die Entwicklung der Prognoseverbesserung haben. Eine Ausnahme ist ANN, GP und LR, welche sich bei steigender Trainingsmenge signifikant verbessern konnte. Auch der RF wies ein gegenteiliges Verhalten auf, denn bei MAE, MLP, RMSE, BEC und MBEC wurden die Resultate mit steigender Trainingsmenge immer schlechter. Für die Gesamtbetrachtung waren bei BEC und MBEC im Vergleich zu R^2 , MAE, RMSE und MAPE nur geringe Unterschiede zwischen den verwendeten Trainingsmengen zu erkennen. Ausserdem waren bei vier Algorithmen, trotz der Betrachtung durch den Median, negative R^2 Werte zu verzeichnen. Diese sind, wie schon erwähnt, auf Ausreißer in den Prognosen zurückzuführen. Doch wie sind die Ergebnisse der neu implementierten BEC und MBEC nun zu interpretieren? Aufgrund der sich zu wenig differenzierenden BEP kann eine direkte Verbindung oder Korrelation zu den bewährten Methoden erstellt werden, denn das Verhalten und die Unterschiede weichen nur wenig voneinander ab und es kann theoretisch von MBEC auf MAPE oder umgekehrt zurückgeschlossen werden. Somit ist es folglich möglich z.B. von einem sinkenden MAPE auf sinkende MBEC oder BEC zu schließen. Dies ließe sich, wie bei der zuvor erwähnten Anpassung des GP-Modells, durch eine neue Interpretierung der Ausgleichsenergiepreise vermutlich lösen.

5 Fazit und Diskussion

Mit dem Ziel, der Einführung der BEC und MBEC als Kennzahl einer Prognosegüte, sind neue Betrachtungsweisen entstanden. Diese Kennzahlen könnten Unternehmen wie die illwerke vkw AG dazu nutzen, um Prognosemodelle unter Berücksichtigung des Kostenfaktors fortlaufen zu monitoren. Denn anderen Kennzahlen sind zwar für die generelle Bewertung einer Prognose gut, spiegeln jedoch die Situation im Energiehandel und -markt nicht ansatzweise wider. Dabei muss es sich bei den Prognosen nicht zwingend um eine Last- oder Verbrauchsprognose im Energiesektor handeln, sondern kann überall da eingesetzt werden, wo Prognosen in Verbindung mit Kosten und einem von dieser Arbeit substituierbaren Preisschema verglichen werden sollen.

Ein zusätzlicher Punkt sind auch die realen Ausgleichsenergiepreise in Abbildung 2. Denn diese sind zwar im Mittel oder Median positiv, doch bei negativen Preisen führt zu viel prognostizierte Energie zu Mehrkosten, da die Abgabe nun bezahlt werden muss und nicht mehr zu Einnahmen führt. Grund dafür ist die hohe Menge an Energie im Markt und jede dazukommende Einheit wird nun mit einem negativen Preis beaufschlagt. Andererseits bietet es einen Vorteil, wenn in so einem Fall zu wenig Energie prognostiziert wurde, da die benötigte Energie günstig am Markt verfügbar ist. Deshalb muss diese Tatsache bei der Betrachtung des Preisschemas BEP berücksichtigt werden und vor allem, dass diese nicht als reale Preise angenommen werden dürfen und sich der Markt sehr viel komplexer gestaltet. Trotzdem kann durch die Kennzahl BEC oder MBEC als Prognosegüte, speziell wenn sich das Preis-Zeit-Verhalten beim Day-Ahead, Intraday und bei den Ausgleichsenergiepreisen gleich verhält, eine Aussage über einen Regressions-Algorithmus getätigt werden.

Andere Algorithmen wie der KNeighbors Regression, die lineare Regression oder die Ridge Regression wurden in dieser Arbeit nicht berücksichtigt, da sie sich im Bereich der Lastvorhersage nicht bewährt und bei der Ermittlung der zu verwenden Methoden mit kurzen Tests nicht die gewünschten Ergebnisse erzielt haben. Mit einem der meist verwendeten Algorithmen wurde aber auch das ARIMA Modell nicht angewandt. Diese ist in der Implementierung sehr komplex und die zu wählenden Parameter sind stark von der verwendeten Datenmenge bzw. den Features abhängig. Diese Parameter müssen in so einem Fall für jede Ausführung der Trainingsmengen erneut ermittelt werden. Hinzu kommt, dass sich die Implementierung in ein Sliding-Window-Verfahren im Vergleich zu ANN, DT, GP, LR, MLP, RF oder SVR als sehr schwer herausgestellt hat und die Methode dadurch nicht anwendbar war. Aber mit jenen, die verwendet wurden, konnte eine Vielzahl an Vorhersagewerten berechnet werden, die als Referenz zueinander ausreichten. Kritisch zu hinterfragen ist aber neben der Auswahl der Algorithmen auch die Entscheidung der Verwendung eines einzelnen Parameters. Theoretisch stellen z.B. Scikit-Learn eine Vielzahl an möglichen Parametern je Modell zur Verfügung. Doch diese zu finden und auszuwerten war mit den zur Verfügung stehenden Ressourcen bei der Erarbeitung dieses Themas nicht

möglich. Die Durchlaufzeiten eines Grid-Search würden mit steigender Anzahl an Parameter und steigendem CV-Wert zu groß werden. Nimmt man ein Beispiel mit drei Parametern und einem CV-Wert von fünf, so gäbe es für jeden Prognosewert 15 Varianten, welche nun für einen größeren Trainingszeitraum angewendet werden müssten, um eine plausible Ausgabe von optimalen Parameterwerten bekommen zu können. Da ein Algorithmus nicht nur durch einen einzelnen Parameter optimiert, gibt es hier das Verbesserungspotenzial, mehrere Parameter zu verwenden, um damit auch die Streuung der Fehler zu minimieren.

Grund für die Entscheidung in dieser Arbeit nicht mehr als 8 Wochen an Trainingsdaten zu verwenden, sind die saisonalen Bedingungen. Denn je weiter der Betrachtungszeitraum in die Vergangenheit reicht, umso öfter würden Trainingsdaten in das Raster fallen, welche nicht zum eigentlich Prognosetag korrelieren. Das hätte zur Folge, dass die Prognosen eventuell mit steigender Anzahl wieder schlechter werden würden. Speziell im Zuge dieser Arbeit und den zur Verfügung stehenden Daten, welche nicht über mehrere Jahre reichen, wäre das ein Risiko in der Berechnung von Vorhersagewerten. Würde der Datenbestand über mehr als ein Jahr reichen, so könnten die Trainingsdaten für den Zeitraum gleich bleiben, doch über die Mehrzahl an Jahren angereichert werden.

Das zweite erklärte Ziel, die Erstellung einer gewichteten Optimierung durch die Ausgleichsenergiepreise, konnte durch Gurobi erfolgreich implementiert werden. Wie in dieser Arbeit aber ersichtlich, konnte sich dieser Algorithmus nicht wie gewünscht von den anderen Algorithmen abheben. Trotzdem erzielte das Modell bessere Prognosen wie ANN, DT, RF, MLP und SVR. Da wäre ein Denkanstoß in die Richtung, zukünftig zu versuchen, die genannten Methoden durch eine Gewichtung der absoluten Prognosefehler zu verbessern. Möglicherweise könnte dazu in den bewährten Methoden ANN, DT, RF, MLP und SVR eine kostenoptimierte Funktion implementiert werden, indem die gewünschten Preise als Vektor hinzugefügt werden können.

Nicht zuletzt war aufgrund der langen Durchlaufzeit die Ausführung der Methoden ein sehr zeitaufwändiger Prozess. Prognosen, wie die Support Vector Regression mit einer Trainingszahl von 8 Wochen, dauerten etwa neun Stunden bis das Ergebnis vorlag. Für ein Projekt dieser Art, oder auch einer Fortführung, wäre es somit vorteilhaft, leistungsfähigere Rechner zu verwenden oder nur einen schmalen Betrachtungszeitraum zu wählen. Aber in dieser Arbeit machte es aufgrund der Wetter- und Energiedaten Sinn, die Prognose auf diese Art zu erstellen. Letztendlich war es sehr zeitintensiv aber zielführend. Vorallem konnten die BEC und MBEC aber eine neue Sichtweise in eine neue Bewertungsmethode bzw. eine Kennzahl als Prognosegüte aufzeigen. Wünschenswert wäre, dass zukünftige Arbeiten dieses Thema weiter verfolgen und diese Thesis als Basis dient, auf der darauf aufgebaut werden kann.

Literaturverzeichnis

- Agyeman, Kofi Afrifa u.a. (2020): „An Ensemble Stochastic Forecasting Framework for Variable Distributed Demand Loads.“ In: *Energies*, 13 (2020), 10, S. 2658. Online im Internet: DOI: 10.3390/en13102658
- „Ausgleichsenergiepreise | APG“ (o. J.): Ausgleichsenergiepreise | APG. Online im Internet: URL: <https://markttransparenz.apg.at/de/markt/Markttransparenz/Netzregelung/Ausgleichsenergiepreise> (Zugriff am: 03.06.2023).
- Bouktif, Salah u.a. (2018): „Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches †.“ In: *Energies*, 11 (2018), 7, S. 1636. Online im Internet: DOI: 10.3390/en11071636
- Desai, Nihal; Patel, Vatsal (o. J.): „Linear Decision Tree Regressor: Decision Tree Regressor Combined with Linear Regressor.“ In:
- Fan, Guo-Feng u.a. (2013): „Support Vector Regression Model Based on Empirical Mode Decomposition and Auto Regression for Electric Load Forecasting.“ In: *Energies*, 6 (2013), 4, S. 1887–1901. Online im Internet: DOI: 10.3390/en6041887
- Fan, Guo-Feng u.a. (2016): „Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression.“ In: *Neurocomputing*, 173 (2016), S. 958–970. Online im Internet: DOI: 10.1016/j.neucom.2015.08.051
- Groß, Arne u.a. (2021): „Comparison of short-term electrical load forecasting methods for different building types.“ In: *Energy Informatics*, 4 (2021), S3, S. 13. Online im Internet: DOI: 10.1186/s42162-021-00172-6
- Haben, Stephen; Voss, Marcus; Holderbaum, William (2023): *Core Concepts and Methods in Load Forecasting: With Applications in Distribution Networks*. Cham: Springer International Publishing. Online im Internet: DOI: 10.1007/978-3-031-27852-5 (Zugriff am: 11.06.2023).
- Jahan, Ibrahim Salem; Snasel, Vaclav; Misak, Stanislav (2020): „Intelligent Systems for Power Load Forecasting: A Study Review.“ In: *Energies*, 13 (2020), 22, S. 6105. Online im Internet: DOI: 10.3390/en13226105
- Liu, Da u.a. (2018): „Monthly Load Forecasting Based on Economic Data by Decomposition Integration Theory.“ In: *Sustainability*, 10 (2018), 9, S. 3282. Online im Internet: DOI: 10.3390/su10093282
- Liu, Yanping u.a. (2016): „Applications of Savitzky-Golay Filter for Seismic Random Noise Reduction.“ In: *Acta Geophysica*, 64 (2016), 1, S. 101–124. Online im Internet: DOI: 10.1515/acgeo-2015-0062
- Nti, Isaac Kofi u.a. (2020): „Electricity load forecasting: a systematic review.“ In: *Journal of Electrical Systems and Information Technology*, 7 (2020), 1, S. 13. Online im Internet: DOI: 10.1186/s43067-020-00021-8
- „Python Library | Meteostat Developers“ (o. J.): Python Library | Meteostat Developers. Online im Internet: URL: <https://dev.meteostat.net/python/> (Zugriff am: 07.06.2023).

- Rostum, Medhat u.a. (2020): „Electrical Load Forecasting: A methodological overview.“ In: International Journal of Engineering & Technology, 9 (2020), 3, S. 842. Online im Internet: DOI: 10.14419/ijet.v9i3.30706
- „Sklearn - DT“ (o. J.): Sklearn - DT. Online im Internet: URL: <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html> (Zugriff am: 28.07.2023).
- „Sklearn - LR“ (o. J.): Sklearn - LR. Online im Internet: URL: https://scikit-learn.org/stable/modules/linear_model.html (Zugriff am: 26.07.2023).
- „Sklearn - SVM“ (o. J.): Sklearn - SVM. Online im Internet: URL: <https://scikit-learn.org/stable/modules/svm.html> (Zugriff am: 27.07.2023).
- „Sklearn - SVR“ (o. J.): Sklearn - SVR. Online im Internet: URL: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html> (Zugriff am: 02.07.2023).
- Voss, Marcus (o. J.): book-case-study/Case_Study_LV_Load_Forecasting_with_ML.ipynb at main · low-voltage-loadforecasting/book-case-study · GitHub. Online im Internet: URL: https://github.com/low-voltage-loadforecasting/book-case-study/blob/main/Case_Study_LV_Load_Forecasting_with_ML.ipynb (Zugriff am: 29.06.2023).
- Wang, Zeyu u.a. (2021): „Hierarchical parameter optimization based support vector regression for power load forecasting.“ In: Sustainable Cities and Society, 71 (2021), S. 102937. Online im Internet: DOI: 10.1016/j.scs.2021.102937
- Wuttke, Laurenz (o. J.): Künstliche neuronale Netze (KNN): Grundlagen und Aufbau. Online im Internet: URL: <https://datasolut.com/neuronale-netzwerke-einfuehrung/> (Zugriff am: 30.06.2023).

Anhang

Im Anhang befinden sich die gesamten Ergebnisse, welche in Kapitel 4 grafisch aufbereitet wurden. Der MAE wird in MWh, der MAPE in %, der RMSE in TWh, der BEC in Mio.€ und der MBEC in €. R^2 besitzt in diesem Fall keine Einheit. Die Ergebnisse können als Vergleiche zu anderen Arbeiten dienen und stehen frei zur Verfügung.

Modell	Training	MAE	MAPE	RMSE	R^2	BEC	MBEC
ANN	7	70,08	27,58	7.694,59	-3,348	572,93	14.842,00
	21	61,82	24,63	5.978,42	-1,402	504,04	13.057,25
	28	59,11	23,91	5.392,32	-0,982	481,19	12.465,34
	56	48,62	20,08	3.460,07	-0,155	393,88	10.203,72
DT	7	49,40	20,06	3.666,98	0,624	401,83	10.409,47
	21	49,48	20,16	3.684,31	0,728	402,26	10.420,59
	28	49,42	20,20	3.690,73	0,707	401,58	10.403,15
	56	49,08	20,20	3.649,83	0,638	398,64	10.326,97
GP	7	61,52	25,21	34.107,55	-	495,94	12.847,47
	21	48,18	19,91	3.433,67	-	389,25	10.083,56
	28	47,94	19,88	3.389,09	-	386,25	10.020,55
	56	47,52	19,77	3.310,11	-	383,47	9.933,98
LR	7	57,32	23,44	27.084,58	-0,442	462,56	11.982,73
	21	47,37	19,50	3.306,29	0,004	383,22	9.927,58
	28	47,08	19,45	3.265,21	-0,024	380,36	9.853,38
	56	46,72	19,37	3.198,32	-0,047	377,56	9.780,73
MLP	7	63,04	25,01	6.251,10	-1,874	513,79	13.309,91
	21	50,37	20,61	3.776,03	-0,100	408,63	10.585,74
	28	48,90	20,18	3.551,36	-0,064	395,50	10.245,47
	56	47,54	19,69	3.323,61	-0,076	384,51	9.960,88
RF	7	47,86	19,48	3.453,75	0,574	388,52	10.064,88
	21	48,32	19,71	3.472,78	0,622	392,99	10.180,52
	28	48,94	20,02	3.572,53	0,588	397,75	10.303,74
	56	49,88	20,51	3.720,51	0,465	405,06	10.493,16
SVR	7	52,31	21,09	4.356,85	-0,046	425,63	11.026,03
	21	49,35	20,12	3.692,69	0,571	401,52	10.401,42
	28	49,70	20,38	3.753,56	0,578	403,66	10.457,05
	56	48,47	19,99	3.593,13	0,626	393,92	10.204,59

Tabelle 17: Gesamtergebnisse der unterschiedlichen Trainingsmengen

Training	Saison	MAE in MWh						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Frühling	71,53	50,09	54,96	52,33	66,52	47,84	54,55
	Herbst	64,18	44,33	49,87	47,86	56,71	43,92	46,66
	Sommer	59,08	51,12	69,80	65,38	53,27	48,94	51,02
	Winter	84,35	51,69	73,02	64,97	73,94	50,59	56,01
21	Frühling	61,84	50,87	48,51	47,89	50,18	48,92	49,62
	Herbst	59,20	44,26	43,63	43,11	47,34	43,86	44,37
	Sommer	54,81	50,95	50,45	49,74	52,97	50,03	52,34
	Winter	70,98	51,24	49,89	48,49	50,97	50,13	50,84
28	Frühling	58,14	51,25	48,38	47,85	49,76	49,83	50,66
	Herbst	58,84	44,52	43,99	43,34	45,88	44,45	44,30
	Sommer	56,08	51,79	50,95	50,04	51,95	50,87	53,69
	Winter	63,50	49,42	48,25	46,81	47,74	50,19	49,79
56	Frühling	49,74	52,02	50,63	49,62	50,05	53,70	49,73
	Herbst	46,81	45,00	42,83	42,28	43,25	46,03	45,00
	Sommer	52,21	51,49	50,80	49,98	51,48	50,75	51,98
	Winter	45,46	46,90	44,85	44,09	44,64	47,80	46,77

Tabelle 18: Gesamtergebnisse MAE Saison der unterschiedlichen Trainingsmengen

Training	Saison	MAPE in %						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Winter	27,42	19,98	21,89	20,77	25,49	19,10	21,52
	Frühling	26,54	18,75	21,09	20,11	23,40	18,57	19,66
	Sommer	26,65	23,46	32,43	30,32	24,36	22,53	23,38
	Herbst	29,67	18,16	26,46	23,43	26,53	17,92	19,70
21	Winter	24,19	20,36	19,71	19,42	20,25	19,60	19,83
	Frühling	24,22	18,69	18,54	18,14	19,76	18,47	18,68
	Sommer	25,03	23,54	23,39	23,03	24,31	23,09	24,20
	Herbst	25,18	18,08	18,15	17,50	18,33	17,77	17,95
28	Winter	23,18	20,61	19,72	19,50	20,22	20,10	20,38
	Frühling	24,28	18,82	18,67	18,21	19,20	18,67	18,73
	Sommer	25,78	23,98	23,67	23,26	24,09	23,57	24,93
	Herbst	22,71	17,39	17,58	16,92	17,29	17,82	17,57
56	Winter	20,44	21,07	20,72	20,37	20,47	21,96	20,09
	Frühling	19,08	18,94	17,90	17,50	17,83	19,15	18,94
	Sommer	24,57	24,18	24,07	23,62	24,31	23,88	24,58
	Herbst	16,27	16,46	16,24	15,78	16,04	16,73	16,44

Tabelle 19: Gesamtergebnisse MAPE Saison der unterschiedlichen Trainingsmengen

Training	Saison	RMSE in TWh						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Winter	7,98	3,78	4,63	4,22	6,99	3,47	4,64
	Frühling	6,49	3,00	3,82	3,75	4,87	2,93	3,35
	Sommer	5,06	3,83	129,41	101,90	4,18	3,55	3,86
	Herbst	10,99	4,00	9,28	6,83	8,60	3,84	5,42
21	Winter	5,98	3,91	3,51	3,44	3,82	3,57	3,74
	Frühling	5,15	2,99	2,73	2,63	3,25	2,92	3,05
	Sommer	4,51	3,80	3,70	3,63	4,12	3,61	3,95
	Herbst	8,17	3,94	3,75	3,47	3,90	3,74	3,99
28	Winter	5,22	3,95	3,49	3,43	3,72	3,69	3,92
	Frühling	5,09	3,02	2,77	2,66	3,05	2,98	3,03
	Sommer	4,76	3,90	3,78	3,70	3,99	3,75	4,16
	Herbst	6,50	3,75	3,47	3,22	3,39	3,82	3,83
56	Winter	3,68	4,07	3,77	3,66	3,72	4,33	3,80
	Frühling	3,09	3,02	2,57	2,50	2,66	3,11	3,06
	Sommer	4,05	3,96	3,82	3,71	3,93	3,84	4,06
	Herbst	2,95	3,41	2,93	2,77	2,87	3,41	3,39

Tabelle 20: Gesamtergebnisse RMSE Saison der unterschiedlichen Trainingsmengen

Training	Saison	MBEC in Tsd.€						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Winter	15,10	10,52	11,48	10,94	14,04	10,05	11,50
	Frühling	13,66	9,35	10,50	10,12	12,02	9,23	9,80
	Sommer	12,49	10,84	14,44	13,50	11,22	10,35	10,82
	Herbst	17,88	10,86	15,29	13,63	15,60	10,60	11,7
21	Winter	12,98	10,71	10,13	10,01	10,51	10,30	10,48
	Frühling	12,61	9,36	9,21	9,13	10,03	9,29	9,34
	Sommer	11,58	10,76	10,62	10,46	11,19	10,60	11,07
	Herbst	14,98	10,73	10,33	10,07	10,62	10,48	10,67
28	Winter	12,21	10,76	10,09	9,98	10,39	10,47	10,65
	Frühling	12,53	9,44	9,29	9,19	9,74	9,43	9,35
	Sommer	11,82	10,95	10,73	10,52	10,94	10,76	11,34
	Herbst	13,35	10,34	9,95	9,69	9,88	10,49	10,41
56	Winter	10,35	10,89	10,54	10,32	10,43	11,21	10,46
	Frühling	10,04	9,53	9,10	9,01	9,23	9,78	9,50
	Sommer	10,92	10,84	10,62	10,44	10,76	10,66	10,93
	Herbst	9,49	9,87	9,29	9,19	9,29	10,09	9,85

Tabelle 21: Gesamtergebnisse MBEC Saison der unterschiedlichen Trainingsmengen

Training	Wochentag	MAE in MWh						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Mo	71,56	48,44	58,63	55,65	66,44	47,92	52,51
	Di	70,66	43,11	77,08	69,91	61,52	41,37	47,19
	Mi	70,92	42,68	48,90	45,83	58,11	41,60	45,77
	Do	73,75	53,34	58,99	56,92	67,77	51,44	56,75
	Fr	69,45	51,81	60,54	56,69	64,77	49,08	54,67
	Sa	63,73	45,72	54,07	49,40	59,15	43,09	46,56
	So	70,43	60,77	72,68	67,06	63,52	60,58	62,70
21	Mo	61,16	49,15	46,19	45,99	49,35	47,81	49,45
	Di	59,18	43,39	39,11	38,98	41,73	41,54	42,54
	Mi	58,21	43,35	40,86	40,30	43,34	42,01	44,15
	Do	68,38	51,42	50,35	50,59	55,54	50,66	51,88
	Fr	67,77	53,16	51,58	51,26	54,54	50,72	51,84
	Sa	56,17	45,00	45,27	43,49	46,69	45,37	43,71
	So	61,72	60,95	63,94	61,04	61,37	60,17	61,89
28	Mo	57,58	48,69	45,25	45,03	46,45	48,30	48,82
	Di	52,61	43,02	38,30	38,07	39,04	42,00	41,25
	Mi	53,64	43,20	41,19	40,43	42,15	42,15	44,49
	Do	64,30	51,40	50,32	50,03	52,62	50,59	52,53
	Fr	65,33	52,30	51,31	50,67	53,39	50,91	51,64
	Sa	54,72	45,86	45,58	43,75	46,29	46,72	45,00
	So	65,48	61,48	63,70	61,62	62,36	61,97	64,23
56	Mo	46,84	47,51	45,58	45,77	46,10	49,25	47,93
	Di	39,89	42,01	38,65	38,04	38,45	43,07	40,15
	Mi	41,32	41,88	39,29	38,80	39,64	42,49	42,53
	Do	50,75	51,98	49,49	48,79	50,35	49,90	51,60
	Fr	53,10	52,23	50,54	49,93	51,30	51,09	50,69
	Sa	46,98	45,95	45,18	43,55	44,86	48,66	43,65
	So	61,49	62,07	64,02	62,22	62,17	64,84	52,76

Tabelle 22: Gesamtergebnisse MAE Wochentag der unterschiedlichen Trainingsmengen

Training	Wochentag	MAPE in %						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Mo	27,19	19,07	23,19	21,99	25,35	18,91	20,49
	Di	25,14	15,76	30,34	27,52	22,21	15,14	17,29
	Mi	25,07	15,39	17,63	16,50	20,79	14,97	16,60
	Do	26,26	19,07	21,00	20,25	24,25	18,36	20,38
	Fr	25,43	19,04	22,28	20,80	23,82	18,03	20,04
	Sa	27,70	19,91	23,82	21,68	25,49	18,86	20,27
	So	36,39	32,34	38,48	35,57	33,24	32,23	32,67
21	Mo	23,57	19,40	18,66	18,45	19,66	18,91	19,56
	Di	21,62	16,09	14,75	14,63	15,64	15,36	15,86
	Mi	20,93	15,83	14,93	14,71	15,81	15,24	16,16
	Do	24,46	18,40	18,07	18,13	19,96	18,03	18,62
	Fr	24,88	19,59	19,21	19,06	20,35	18,65	19,10
	Sa	24,51	19,58	19,81	18,98	20,39	19,82	19,13
	So	32,50	32,39	34,10	32,65	32,58	32,10	32,56
28	Mo	22,64	19,37	18,46	18,21	18,75	19,21	19,50
	Di	19,53	16,02	14,55	14,37	14,75	15,59	15,52
	Mi	19,43	15,80	15,10	14,80	15,42	15,31	16,33
	Do	23,15	18,46	18,07	17,97	18,92	18,04	18,89
	Fr	24,09	19,25	19,05	18,80	19,92	18,72	19,02
	Sa	24,07	19,97	20,00	19,14	20,29	20,41	19,68
	So	34,58	32,71	34,05	33,02	33,31	33,03	33,83
56	Mo	19,04	19,07	18,73	18,65	18,78	19,75	19,30
	Di	15,09	15,75	14,74	14,39	14,54	16,10	15,19
	Mi	15,18	15,39	14,49	14,25	14,57	15,51	15,69
	Do	18,33	18,65	17,85	17,58	18,15	17,81	18,48
	Fr	19,70	19,19	18,70	18,47	19,04	18,72	18,72
	Sa	20,52	20,17	19,81	19,01	19,61	21,28	19,12
	So	32,83	33,32	34,28	33,37	33,29	34,62	33,54

Tabelle 23: Gesamtergebnisse MAPE Wochentag der unterschiedlichen Trainingsmengen

Training	Wochentag	RMSE in TWh						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Mo	7,93	3,66	12,25	10,15	6,82	3,56	4,25
	Di	7,83	2,86	198,93	155,97	6,04	2,64	3,79
	Mi	7,75	2,73	3,79	3,32	5,02	2,60	3,37
	Do	8,70	4,07	5,57	5,46	7,23	3,81	5,11
	Fr	7,90	4,00	6,26	5,21	6,60	3,63	4,64
	Sa	6,19	3,07	4,96	4,03	5,84	2,78	3,46
	So	7,53	5,27	8,10	6,31	6,21	5,17	5,87
21	Mo	5,95	3,74	3,32	3,22	3,76	3,50	3,75
	Di	5,55	2,87	2,35	2,31	2,73	2,64	2,70
	Mi	4,99	2,83	2,46	2,37	2,75	2,63	2,95
	Do	6,97	3,83	3,65	3,67	4,47	3,72	3,99
	Fr	7,46	4,18	4,03	3,93	4,44	3,83	4,06
	Sa	5,21	3,05	2,83	2,64	3,09	3,01	2,92
	So	5,69	5,29	5,41	5,01	5,19	5,00	5,48
28	Mo	5,39	3,63	3,14	3,08	3,32	3,53	3,64
	Di	4,38	2,77	2,24	2,19	2,34	2,69	2,51
	Mi	4,32	2,80	2,49	2,37	2,59	2,66	3,03
	Do	6,08	3,88	3,63	3,57	3,95	3,71	4,09
	Fr	6,73	4,10	3,92	3,81	4,26	3,86	4,07
	Sa	4,67	3,14	2,86	2,67	3,03	3,20	3,07
	So	6,16	5,45	5,44	5,17	5,37	5,36	5,87
56	Mo	3,33	3,48	3,12	3,14	3,23	3,65	3,58
	Di	2,36	2,67	2,25	2,16	2,20	2,81	2,44
	Mi	2,47	2,67	2,27	2,19	2,28	2,76	2,69
	Do	3,57	3,94	3,46	3,33	3,55	3,61	3,99
	Fr	4,08	4,07	3,68	3,60	3,81	3,83	3,87
	Sa	3,08	3,13	2,85	2,65	2,85	3,43	2,90
	So	5,34	5,59	5,55	5,32	5,35	5,97	5,69

Tabelle 24: Gesamtergebnisse RMSE Wochentag der unterschiedlichen Trainingsmengen

Trai- ning	Wo- chen- tag	MBEC in Tsd.€						
		ANN	DT	GP	LR	MLP	RF	SVR
7	Mo	15,17	10,17	12,15	11,55	14,13	10,02	10,98
	Di	15,02	9,05	15,60	14,20	12,99	8,66	9,88
	Mi	15,06	9,00	10,26	9,62	12,26	8,77	9,59
	Do	15,70	11,38	12,57	12,14	14,41	10,97	12,08
	Fr	14,78	11,04	12,90	12,11	13,73	10,46	11,67
	Sa	13,45	9,60	11,37	10,37	12,47	9,00	9,84
	So	14,70	12,63	15,12	13,92	13,18	12,57	13,14
21	Mo	12,96	10,30	9,55	9,55	10,31	10,02	10,35
	Di	12,41	9,05	8,05	8,05	8,64	8,69	8,91
	Mi	12,33	9,13	8,57	8,45	9,12	8,83	9,27
	Do	14,59	10,96	10,73	10,78	11,84	10,81	11,09
	Fr	14,46	11,35	10,96	10,91	11,63	10,85	11,03
	Sa	11,83	9,46	9,50	9,14	9,83	9,53	9,21
	So	12,79	12,71	13,23	12,61	12,71	12,54	12,95
28	Mo	12,12	10,18	9,30	9,31	9,63	10,09	10,16
	Di	10,96	8,97	7,85	7,82	8,02	8,78	8,61
	Mi	11,34	9,08	8,62	8,47	8,84	8,86	9,31
	Do	13,71	10,96	10,73	10,68	11,23	10,81	11,22
	Fr	13,95	11,17	10,92	10,79	11,37	10,90	10,96
	Sa	11,53	9,64	9,56	9,19	9,73	9,81	9,48
	So	13,61	12,82	13,17	12,73	12,90	12,89	13,46
56	Mo	9,70	9,90	9,36	9,44	9,51	10,25	9,99
	Di	8,23	8,77	7,91	7,81	7,91	8,97	8,39
	Mi	8,66	8,76	8,18	8,10	8,28	8,89	8,91
	Do	10,84	11,08	10,56	10,42	10,75	10,67	11,02
	Fr	11,35	11,20	10,79	10,65	10,95	10,99	10,83
	Sa	9,91	9,67	9,51	9,18	9,46	10,25	9,22
	So	12,74	12,90	13,24	12,87	12,86	13,45	13,07

Tabelle 25: Gesamtergebnisse MBEC Wochentag der unterschiedlichen Trainingsmengen

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich vorliegende Masterarbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Stellen sind als solche kenntlich gemacht.

Die Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Dornbirn, am 02.08.2023

Unterschrift Verfasser*in