

Analyzing Day-ahead System Load Forecasts with SHAP

Bachelor Thesis of

Lena Richau

At the KIT Department of Informatics
Institute for Automation and Applied Informatics (IAI)

First examiner: TT-Prof. Dr. Benjamin Schäfer

Second examiner: Prof. Dr. Veit Hagenmeyer

First advisor: M.Sc. Matthias Hertel

Second advisor: M.Sc. Alexandra Nikoltchovska

Matriculation Number 
December 1st, 2025 – March 29th, 2026

Karlsruhe Institute of Technology (KIT)
Department of Informatics
P.O. Box 6980
76128 Karlsruhe

On the Use of AI Tools

During the preparation of this work the authors used AI tools in order to improve efficiency, clarity and fluency. Their outputs were always critically examined and they only served as aids to thoughtful analysis and judgment.

The following AI Tools were used:

Experimental work:

- **Github Copilot:** used for writing comments, identifying errors and improving plots
- **Claude Sonnet 4.5:** used for further literature research

Thesis Writing:

- **Claude Sonnet 4.4 and 4.5 as well as Haiku 4.5:** used for summarizing, refining academic language and for helping with LaTeX formatting
- **DeepL Write:** Used for sentence paraphrasing and stylistic improvements

After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

I declare that I have developed and written the enclosed thesis completely by myself. I have not used any other than the aids that I have mentioned. I have marked all parts of the thesis that I have included from referenced literature, either in their original wording or paraphrasing their contents. I have followed the by-laws to implement scientific integrity at KIT.

Karlsruhe, March 29, 2026



.....
(Lena Richau)

Abstract

Accurate day-ahead electricity load forecasting is becoming increasingly challenging as power systems transition towards higher shares of variable renewable energy sources. Solar and wind generation are inherently weather-dependent, making the electricity mix more sensitive to meteorological conditions. Simultaneously, climate change is shifting temperature extremes and weather patterns in ways that alter electricity demand.

Day-ahead electricity load forecasts published on the ENTSO-E Transparency Platform contain systematic, learnable errors — yet their structure and drivers remain poorly understood. This thesis investigates whether the day-ahead load forecast issued by TransnetBW, which is one of four German Transmission Service Operator (TSO)s, contains predictable residual patterns, and quantifies which features drive those patterns using SHapley Additive exPlanations.

We model the residuals between the official ENTSO-E day-ahead forecast and the actual load data provided directly by TransnetBW using XGBoost, covering the period January 2021 to October 2025. The feature set spans four categories: the day-ahead forecast itself, calendar features, meteorological features from the ERA5 reanalysis, and lagged load and error features.

We evaluated different feature set combinations and trained different models with them. The best model outperforms the accuracy of the day-ahead forecast significantly.

A subsequent SHapley Additive exPlanations (SHAP) analysis on the held-out test set identifies four physically interpretable sources of systematic error: Firstly, insufficient differentiation between weekdays and weekend leads to consistent underestimation on workdays and overestimation on weekends. Secondly, extreme temperatures produce a distinct error pattern, attributable to inadequately modeled electrically driven heating and cooling demand and insufficient capturing of concept drift patterns based on climate change leading to more extreme temperature overall. Thirdly, high solar irradiance causes systematic overprediction being possibly caused by behind-the-meter photovoltaic generation which reduces metered load in a way not captured by the TSO model. Fourthly, the forecast overestimates demand on public holidays and bridge days, reflecting a limited representation of reduced industrial activity.

The results demonstrate that the ENTSO-E day-ahead forecast contains temporally stable, physically driven systematic errors that are partially predictable from observable features. These findings suggest that a post-processing correction approach could be a promising direction for further improving day-ahead forecast accuracy in the TransnetBW control area.

Acknowledgments

I would like to thank everyone who contributed to the success of this thesis. In particular, I would like to thank Alexandra Nikoltchovska and Matthias Hertel for their excellent guidance. I would also like to thank my friends and family for their support during the writing of this thesis and throughout my studies.

Contents

Abstract	i
Acknowledgments	iii
1. Introduction	1
1.1. Related Work	2
1.1.1. Systematic Errors of ENTSO-E Day-Ahead Forecasts	2
1.1.2. ML-Based Forecast Post-Processing and Error Decomposition . .	2
1.1.3. Research Gap	2
2. Fundamentals	5
2.1. Time Series	5
2.1.1. General Definition	5
2.1.2. Concept Drift	5
2.2. ENTSO-E day-ahead forecast	6
2.2.1. ENTSO-E Transparency Platform	6
2.2.2. TransnetBW	6
2.3. Gradient Boosted Decision Trees	6
2.3.1. XGBoost	7
2.4. SHapley Additive exPlanations	8
2.4.1. Historical Background	8
2.4.2. Exact Calculation of Shapley Values	9
2.4.3. Estimation of Shapley Values	9
2.4.4. SHAP Feature Interaction	10
3. Data and Experimental Setup	13
3.1. Dataset	13
3.1.1. Load Data	13
3.1.2. Dataset Components	15
3.2. Correlation of features	16
3.3. Dataset Split	17
3.3.1. Group Shuffle Split	17
3.3.2. Train-Validation-Test Split	18
3.4. Error Metrics	18
3.4.1. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE)	18
3.4.2. Mean Absolute Percentage Error (MAPE)	19
3.4.3. symmetric Mean Absolute Percentage Error (sMAPE)	19

3.4.4.	Mean Absolute Scaled Error	20
4.	Methodology	21
4.1.	Model Setup	21
4.1.1.	Model Parameters and Hyperparameters	21
4.1.2.	Model Evaluation	22
4.2.	Selection of most accurate Data Subset	22
4.3.	SHAP	23
4.3.1.	SHAP Analysis on test set	23
4.3.2.	SHAP Analysis on most accurate Data Subset	23
5.	Evaluation	25
5.1.	Analysis of different trained Models	25
5.2.	Prediction and Error Analysis of best model	27
5.2.1.	MAE and RMSE	28
5.2.2.	Mean Absolute Scaled Error (MASE)	29
5.3.	SHAP Analysis of Best Model	30
5.3.1.	SHAP Bar and Beeswarm plot analysis	30
5.3.2.	SHAP Feature interaction analysis	32
5.3.3.	SHAP Dependence plot analysis	33
5.4.	SHAP Analysis of Best Model's most accurate Dataset Part	40
5.4.1.	SHAP Analysis of most accurate Data Subset	40
6.	Conclusion	43
7.	Limitations and Outlook	45
7.1.	Limitations	45
7.1.1.	Concept Drift and Temporal Generalization	45
7.1.2.	Feature Set	45
7.2.	Outlook	46
7.2.1.	Concept Drift Analysis	46
7.2.2.	Comparing Online Load and European Network of Transmission System Operators for Electricity (ENTSO-E) Load	46
7.2.3.	Feature Set improvements	46
7.2.4.	Alternative Data Splitting Strategies	47
7.2.5.	Forecast Correction	47
	Bibliography	49
A.	Appendix	53
A.1.	MAPE and sMApe for all models	54
A.1.1.	MAPE	54
A.1.2.	sMAPE	54
A.2.	All Dependence Plots	55

List of Figures

3.1.	Feature correlation matrix displaying pairwise Pearson correlations between all input features, ranging from -1 (dark purple) to +1 (bright yellow), with the diagonal equal to 1.0 by definition.	16
5.1.	Residual of day-ahead forecast and predicted day-ahead residual	27
5.2.	Bar Plot for the best model with the top ten mean absolute SHAP features	30
5.3.	SHAP beeswarm plot of the top ten features	31
5.4.	Interaction matrix of the different SHAP features. SHAP values which show high interaction patterns are shown in lighter colors, whereas SHAP values which interact less are shown in darker colors.	32
5.5.	SHAP Dependence Plot of the feature day-ahead total load forecast.	33
5.6.	SHAP Dependence Plot of the feature load lagged by 24 hours.	34
5.7.	SHAP Dependence Plot of the feature day of week.	34
5.8.	SHAP Dependence Plot of the feature air temperature.	35
5.9.	SHAP Dependence Plot of the feature precipitation.	36
5.10.	SHAP Dependence Plot of the feature week of year.	36
5.11.	SHAP Dependence Plot of the feature hour.	37
5.12.	SHAP Dependence Plot of the feature irradiance.	38
5.13.	SHAP Dependence Plot of the feature is holiday.	39
5.14.	SHAP Dependence Plot of the feature is long weekend.	39
5.15.	SHAP Barplot for SHAP analysis on most accurate subset in terms of MASE of the data	40
5.16.	SHAP Beeswarm plot for SHAP analysis on most accurate subset in terms of MASE of the data	41
5.17.	SHAP Dependence Plot of the feature week of year	42
A.1.	SHAP dependence plots for all features of the best model, grouped by category: lagged load (row 1), lagged error (row 2), weather (rows 3–4), and calendar (row 5). Each plot shows the SHAP value of the feature on the y -axis against the feature value on the x -axis, with point colour indicating the value of the feature with the strongest SHAP interaction.	55

List of Tables

3.1.	Comparison of errors between the two load datasets for the day-ahead total load forecast.	14
3.2.	Overview of all input features grouped by category. Lag offsets are given in hours; binary features take the value 1 (true) or 0 (false).	15
5.1.	Performance comparison of models trained on different feature subsets. CF = calendar features, WF = weather features, LF = lag features, CWF = calendar + weather features, LCF = lag + calendar features, WLF = weather + lag features, ALL = all features combined.	25
5.2.	Comparison of forecasting errors between the best model and the day-ahead (DA) total load forecast baseline.	28
A.1.	Performance comparison of models trained on different feature subsets. CF = calendar features, WF = weather features, LF = lag features, CWF = calendar + weather features, LCF = lag + calendar features, WLF = weather + lag features, ALL = all features combined.	54

1. Introduction

Ensuring secure and reliable energy supply requires accurate short-term load forecasting. The energy landscape is undergoing a structural transformation, which is also visible in Germany: renewable energy shares covered a record 62.7% of net public electricity generation in 2024, with wind contributing 33 % and solar 14 %, and photovoltaic expansion [Fra25]. As electricity networks integrate ever-higher shares of variable renewable sources, load fluctuations become more volatile and increasingly dependent on meteorological conditions [GPB19]. At the same time, climate change is altering weather patterns and shifting temperature extremes in ways that are more difficult to capture by forecasts [CF14]. As the electricity mix grows more reliant on weather-driven generation, demand and supply become jointly sensitive to the same meteorological variables, amplifying forecast uncertainty. This growing complexity makes accurate forecasting more challenging and simultaneously more essential for preventing bottlenecks, supply shortages and costly imbalances.

The Transmission System Operators (TSOs) are required to provide a day-ahead load forecast to the ENTSO-E transparency platform [Eur13]. According to the ENTSO-E Detailed Data Descriptions [ENT23], these forecasts rely primarily on historical load profiles of similar days, adjusted for relevant influencing factors such as meteorological conditions, climatic patterns and socioeconomic factors. Additionally, forecasts must be updated whenever deviations from the actual total load exceed 10%. Despite these procedures, significant forecast errors are frequently observed, generating financial risk due to curtailment actions or the procurement of balancing energy.

The objective of this thesis is to evaluate the accuracy of ENTSO-E day-ahead load forecast by modeling their residuals using machine learning, specifically gradient-boosted regression trees (e.g. XGBoost). The central research question is whether the TSO's forecast exhibits systematic, learnable errors driven by daily, weekly and annual calendar patterns, meteorological influence, special-day behavior and concept drift. To answer this question, the residual prediction model is interpreted using SHAP (SHapley Additive exPlanations) [LL17; Lun+20] to quantify the influence of individual and pairwise interactions of features on the prediction error. The SHAP analysis is cross-checked against a second analysis restricted to the subset of predictions with the highest relative accuracy.

1.1. Related Work

1.1.1. Systematic Errors of ENTSO-E Day-Ahead Forecasts

Kazmi and Tao (2022) [KT22] provide the most comprehensive pan-European assessment of TSO forecast quality. They analyse five years of day-ahead load and renewable generation forecasts across 16 EU countries. They found that absolute forecast errors increased approximately linearly with demand, and that the rapid expansion of wind and solar capacity amplified these errors over time. Their work establishes that ENTSO-E residuals are not purely random and that there is substantial room for improvement across most TSOs, including those in Germany. Möbius et al. (2023) [Möb+25] showed in their work that the German ENTSO-E load forecast can be corrected using a model only trained on the historical forecast errors. Together, these two papers suggest that there are systematic, learnable error patterns relating to solar, wind, and lagged error features.

1.1.2. ML-Based Forecast Post-Processing and Error Decomposition

Xie et al. (2025) [Xie+25] propose a theoretically motivated decomposition of "forecast errors into three components: model bias from historical inertia, errors in dependent variables, and random noise." They develop targeted correction mechanisms for each component and demonstrate substantial error reductions in the PJM Interconnection, an American Regional Transmission Organization, where data centers and industrial growth has caused rapid structural shifts in the load profile. This paper shows, that it is possible to use systematic error patterns to improve day-ahead forecasts and that it is possible to capture systematic error pattern shifts over time by using a rolling window approach. While this work also analyses the feature-based error, it is focused on the US energy market (PJM), which makes the findings not directly applicable to the EU or German energy market. Moreover, the error is not further analyzed to provide interpretable, feature-level explanations of which contextual conditions produce which error patterns.

1.1.3. Research Gap

The reviewed literature shows, that there is a research gap at the intersection of explainable error analysis of the ENTSO-E day-ahead forecast and residual prediction. The reviewed literature establishes that there are several shortcomings of the day-ahead forecast. Studies show, that the residual contains a learnable error structure in terms of solar and wind energy generation [KT22] and other existing error or feature based patterns [Möb+25; Xie+25]. The studies further show that this structure can be used to improve the day-ahead forecast [Xie+25; Möb+25]. No prior work has applied SHAP-based feature-level error attribution to analyze systematic errors in the German ENTSO-E forecast.

This thesis addresses this gap by training an XGBoost model to predict the ENTSO-E day-ahead residual using the same ground truth load data as the TSO and applying SHAP

to provide a granular, feature-level explanation of which contextual conditions drive systematic forecast errors.

The remainder of this thesis is structured as follows:

Chapter 2 introduces the theoretical background, covering time series concepts including concept drift, the ENTSO-E day-ahead forecasting framework, gradient-boosted decision trees, and the SHAP methodology.

Chapter 3 describes the dataset, feature construction, data splitting strategy, and error metrics used throughout the study.

Chapter 4 details the model setup, hyperparameter configuration, and the definition of the high-accuracy data subset used for the secondary SHAP analysis.

Chapter 5 presents the experimental results, beginning with a comparison of models trained on different feature subsets, followed by an error analysis of the best model and the full SHAP interpretation on the test set and the followed by the SHAP analysis on the high-accuracy data subset.

Chapter 6 summarizes the findings and answers the research question, and Chapter 7 discusses limitations and directions for future work.

2. Fundamentals

This chapter on fundamentals gives an overview of all necessary concepts.

2.1. Time Series

2.1.1. General Definition

A time series is a sequence of observations recorded sequentially in time [HA21]. Formally a time series is defined as $(y_1, \dots, y_T)$. A model to predict this series based on different feature variables $j^1, \dots, j^n$ is defined as

$$\hat{Y} = f(j^1, \dots, j^n)$$

In our case of predicting the residual y_t of the day-ahead forecast at a given time t we define:

$$\hat{e}_t = f(j_t^1, \dots, j_t^n)$$

When providing lagged features j_{t-l}^i we include them into the model as:

$$\hat{e}_t = f(j_t^1, \dots, j_t^n, j_{t-l}^{n+1}, \dots, j_{t-l}^m)$$

with $1 \leq n < m$ and l being the lag in hours.

We can then decompose the day-ahead forecast error time series into two components:

$$Y_t = \hat{e}_t + e_t$$

with $\hat{e}$ being our prediction for the day-ahead forecast residual and e being the error component of our prediction model at time t . This error component e_t is heavily influenced by factors like accuracy of the dataset, limitations of the model itself and Concept Drift.

2.1.2. Concept Drift

Concept Drift in time series is an overall huge contributor to the inaccuracies of prediction and is hardly capturable. In our case of electricity load data and the day-ahead forecast this is driven by several parameters that are changing continuously. The energy mix is shifting towards greater reliance on renewable energy, making it highly dependent on weather

variables, which are becoming increasingly difficult to predict as climate change advances. Additionally, load patterns are changing due to shifts in industrial and residential patterns, driven by increased behind-the-meter PV expansion [Erd+22] and growing heat pump adoption [Sem+24]. At the same time, it is unclear whether or when the TSO adapts or switches the day-ahead forecast model.

2.2. ENTSO-E day-ahead forecast

2.2.1. ENTSO-E Transparency Platform

ENTSO-E is the association of European TSOs, currently comprising 40 members from 36 countries [ENT26]. Its mandate includes providing the data necessary to coordinate cross-border grid operations and facilitating the functioning of the European internal electricity market [ENT26]. Under EU Regulation No 543/2013 [Eur13] on submission and publication of data in electricity markets ENTSO-E operates the Transparency Platform to provide a centralized, publicly accessible repository of pan-European electricity data. It covers electricity generation, load, transmission, balancing and outages [ENT26].

TSOs are legally required under this regulation to submit day-ahead total load forecasts for their control areas to the Transparency Platform by a defined deadline before the delivery day. According to the platform documentation, the day-ahead load forecast is calculated on the basis of the historic load profile on similar days, taking into account variables that affect electricity demand including weather conditions, climate factors, and socioeconomic variables [ENT23]. The forecast methodology and model used differ across TSOs, as each member submits forecasts independently using their own internal systems.

2.2.2. TransnetBW

TransnetBW GmbH is one of four TSOs in Germany alongside TenneT, Amprion, and 50Hertz. It was founded in 2012 as a legally and operationally independent entity separated from EnBW (Energie Baden-Württemberg) under EU reform for the liberalization of the wholesale electricity industry [Tra26].

As the TSO responsible for the Baden-Württemberg control area, TransnetBW is legally required to submit day-ahead total load forecasts to the ENTSO-E Transparency Platform.

2.3. Gradient Boosted Decision Trees

Gradient tree boosting is an ensemble learning technique that builds a strong predictive model by sequentially combining a series of weak learners in the form of shallow decision trees. The core idea, introduced by Friedman [Fri01], is to interpret the fitting of each

successive tree as gradient descent in function space. At each step, a new tree is trained to approximate the negative gradient of the loss function with respect to the current ensemble prediction, effectively correcting the residual errors of all previous trees.

Following [CG16], given a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, the model builds an additive prediction:

$$\hat{y}_i = \sum_{k=1}^K f_k(\mathbf{x}_i), \quad (2.1)$$

where $\hat{y}_i$ is the predicted value for data point i , $\mathbf{x}_i$ is its feature vector, K is the total number of trees, and $f_k(\mathbf{x}_i)$ is the output of the k -th tree. At each training step t , a new tree f_t is added to minimize the total loss over all n training points [CG16]:

$$\mathcal{L}^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)), \quad (2.2)$$

where l is a differentiable convex loss function, y_i is the true value, and $\hat{y}_i^{(t-1)}$ is the prediction of the ensemble before adding tree t . To make this optimization tractable, Chen and Guestrin (2016) [CG16] approximate the loss via a second-order Taylor expansion:

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^n \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right], \quad (2.3)$$

where $g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$ and $h_i = \partial_{\hat{y}_i^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)})$ are the first- and second-order gradient statistics of the loss. Since g_i and h_i are computed from the previous ensemble, they are fixed when optimizing f_t , making the problem analytically tractable. An additional technique to prevent overfitting is shrinkage, introduced by Friedman (2001) [Fri01], which scales each newly added tree by a learning rate $\eta \in (0, 1]$, reducing the influence of any individual tree and leaving capacity for subsequent trees to refine the model.

2.3.1. XGBoost

XGBoost (eXtreme Gradient Boosting), introduced by Chen and Guestrin (2016) [CG16], is a scalable, open-source implementation of gradient tree boosting that has become the state-of-the-art choice for tabular datasets such as ours, which is also confirmed by Borisov et al. [Bor+24].

The key contribution of XGBoost over standard gradient tree boosting is the addition of an explicit regularization term to the learning objective:

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k), \quad \text{where} \quad \Omega(f) = \gamma T + \frac{1}{2} \lambda \|\mathbf{w}\|^2. \quad (2.4)$$

Here T is the number of leaves in tree f , $\mathbf{w}$ are the leaf weights, and γ and λ are regularization hyperparameters controlling tree complexity and weight magnitude respectively. This regularized objective penalizes overly complex trees and smooths the learned leaf weights, thereby improving generalization over unseen data.

2.4. SHapley Additive exPlanations

Machine learning models such as gradient boosted trees achieve high predictive performance, but their internal mechanics are largely opaque. To bridge the gap between model accuracy and human understanding, a family of methods known as Explainable Artificial Intelligence (XAI) has emerged. Some of these methods, especially SHAP are model-agnostic in nature: they treat the trained model as a black box with inputs and outputs, manipulate those inputs, and draw conclusions from the resulting changes in output [Mol23]. SHAP is one such method and has become a de facto standard for post-hoc model explanation. This section introduces the theoretical foundations of Shapley values, explains how they are estimated in practice, and describes the Tree Explainer algorithm used in this thesis.

2.4.1. Historical Background

The concept of Shapley values originates from cooperative game theory introduced by Lloyd Shapley in 1952 [Sha52]. The central problem which this paper addresses is how to fairly distribute the total payout of a cooperative game among its individual players, given that players contribute differently to this payout.

The application of Shapley values in machine learning was further developed by Strumbelj and Kononenko [ŠK14]. In 2016, Ribeiro et al. introduced Local Interpretable Model-Agnostic Explanations (LIME), a related local explanation method. Building on these developments, Lundberg and Lee [LL17] introduce the acronym SHAP and showing that Shapley values provide a principled unification of several existing explanation techniques such as LIME and DeepLift. The latter provided SHAP python package, which is maintained by Lundberg and had a growing number of contributors, played a decisive role in the widespread adoption. A further advancement was made when Lundberg et al. [Lun+20] proposed an algorithm for efficient SHAP value computation for tree-based models.

2.4.2. Exact Calculation of Shapley Values

To apply Shapley values to a machine learning model f , each input feature j of a feature input vector to a data point $\mathbf{x}^{(i)}$ is treated as a player. As a player, the feature can be part of a coalition. A coalition S consists of the features whose values are “known”, while the complementary set $C = \{1, \dots, p\} \setminus S$ contains features that are treated as absent. Because a machine learning model typically requires a value for every feature, absent features are marginalized out by sampling from a background dataset, which is typically a subset of the training data. The dataset on which the SHAP analysis itself is conducted - in this thesis the held-out test set - is kept separate from the background data. The SHAP value function is then defined as:

$$v_{f, \mathbf{x}^{(i)}}(S) = \int f(\mathbf{x}_S^{(i)} \cup \mathbf{X}_C) d\mathbb{P}_{\mathbf{X}_C} - \mathbb{E}[f(\mathbf{X})], \quad (2.5)$$

where $\mathbf{x}_S^{(i)}$ denotes the observed feature values in coalition S , $\mathbf{X}_C$ are random variables representing the absent features, and $\mathbb{E}[f(\mathbf{X})]$ is the expected model output over the background data. Subtracting the expected output ensures that $v(\emptyset) = 0$, so that the total payout equals $f(\mathbf{x}^{(i)}) - \mathbb{E}[f(\mathbf{X})]$.

Substituting (2.5) into the Shapley formula yields the SHAP value $\phi_j^{(i)}$ of feature j for instance i :

$$\phi_j^{(i)} = \sum_{S \subseteq \{1, \dots, p\} \setminus \{j\}} \frac{|S|! (p - |S| - 1)!}{p!} \left(v_{f, \mathbf{x}^{(i)}}(S \cup \{j\}) - v_{f, \mathbf{x}^{(i)}}(S) \right). \quad (2.6)$$

The weight $\frac{|S|! (p - |S| - 1)!}{p!}$ corresponds to the probability of encountering coalition S when drawing uniformly at random from all $p!$ permutations of the features.

According to Molnar [Mol23], a key property of SHAP values is that they are centered, meaning that each $\phi_j^{(i)}$ represents the contribution of feature j to the deviation of the prediction from the average. A positive value indicates that feature j pushed the prediction above the mean, while a negative value indicates the opposite.

2.4.3. Estimation of Shapley Values

Exact computation of SHAP values faces two practical obstacles. First, the integral in (2.5) requires knowledge of the joint distribution of the absent features $\mathbf{X}_C$, which is generally unavailable. Second, the number of coalitions grows exponentially with the number of features (2^p), making exhaustive enumeration infeasible for models with many inputs.

The core idea is to replace the integral over the feature distribution with a sample average over a background dataset, typically the training data or a representative subsample. Absent feature values are substituted with values randomly drawn from this background

data, and the model is evaluated on the resulting hybrid inputs. The value function for a coalition S is then estimated as

$$\hat{v}(S) = \frac{1}{n} \sum_{k=1}^n \left(f(\mathbf{x}_S^{(i)} \cup \mathbf{x}_C^{(k)}) - f(\mathbf{x}^{(k)}) \right), \quad (2.7)$$

where $\mathbf{x}_C^{(k)}$ denotes the absent feature values drawn from the k -th background sample. The marginal contribution of feature j when added to coalition S can then be calculated by the following formula:

$$\hat{\Delta}_{S,j} = \hat{v}(S \cup \{j\}) - \hat{v}(S) = \frac{1}{n} \sum_{k=1}^n \left(f(\mathbf{x}_{S \cup \{j\}}^{(i)} \cup \mathbf{x}_{C \setminus \{j\}}^{(k)}) - f(\mathbf{x}_S^{(i)} \cup \mathbf{x}_C^{(k)}) \right). \quad (2.8)$$

When p is large and iterating over all coalitions remains intractable, the permutation estimator samples random orderings of the features and computes marginal contributions in a forward and backward sweep through each permutation. Because adjacent coalitions in a sweep share intermediate model evaluations, the method is computationally efficient [Mol23].

2.4.4. SHAP Feature Interaction

Two features interact when the prediction cannot be decomposed into a sum of their individual effects. That is, when the effect of one feature changes depending on the value of another [Mol23]. Standard SHAP values already absorb these interactions implicitly as $\phi_j^{(i)}$ is averaged over all coalitions, it captures both the isolated contribution of feature j and any joint effects with other features.

To decompose interactions explicitly, Lundberg and Lee [Lun+20] introduced SHAP interaction values, extending the standard SHAP framework using the Shapley interaction index from cooperative game theory.

To decompose interactions explicitly, Lundberg et al. [Lun+20] proposed SHAP interaction values, extending the standard SHAP framework by decomposing each feature contribution into a main effect and pairwise interaction terms. The approach is grounded in the Shapley interaction index from cooperative game theory [FKM06], which measures simultaneous interaction among players through differences of their marginal contributions as:

$$\Delta_{jk} v_{f, \mathbf{x}^{(i)}}(T) = v(T \cup \{j, k\}) - v(T \cup \{j\}) - v(T \cup \{k\}) + f(T), \quad (2.9)$$

where $T \subseteq \{1, \dots, p\} \setminus \{j, k\}$ is the set of other present features, distinct from the coalition S in the SHAP value function 2.5. This quantity measures how much the marginal contribution of feature k changes when feature j is also present in T . Substituting into

the Shapley interaction index [FKM06] yields the pairwise SHAP interaction value for features $j \neq k$:

$$\Phi_{jk}^{(i)} = \sum_{T \subseteq \{1, \dots, p\} \setminus \{j, k\}} \frac{1}{p-1} \binom{p-2}{|T|}^{-1} \Delta_{jk} v_{f, \mathbf{x}^{(i)}}(T). \quad (2.10)$$

The result is a symmetric $p \times p$ matrix per prediction, where $\Phi_{jk}^{(i)} = \Phi_{kj}^{(i)}$. A positive off-diagonal entry reflects a complementarity effect between features j and k , while a negative entry reflects redundancy [FKM06]. Lundberg et al. [Lun+20] additionally define diagonal entries as $\Phi_{jj}^{(i)} = \phi_j^{(i)} - \sum_{k \neq j} \Phi_{jk}^{(i)}$, absorbing each feature's remaining main effect such that summing any row recovers $\phi_j^{(i)}$. They also extended the Tree Explainer algorithm to compute this full interaction matrix exactly in polynomial time by exploiting the recursive structure of decision trees.

The interaction values as well have a direct application in SHAP dependence plots. When the full SHAP explanation object is passed, the plot automatically selects the feature with the strongest approximate interaction to color the points [SHA26]. This makes it possible to visually identify which feature drives the interaction directly from the plot.

3. Data and Experimental Setup

The code for this thesis which is referenced in this section can be found at: https://gitlab.kit.edu/usfwo/load_forecasting_with_shap.git

3.1. Dataset

The main goal of this thesis is to evaluate if there is a systematic error in the ENTSO-E day-ahead forecast. Therefore we only use data that was available at the time of the day-ahead forecast to have the same ground-truth as the day-ahead forecast at the time.

We used a dataset consisting of the forecasting goal of the day ahead residual and different feature variables. The residual was calculated using load data provided by TransnetBW and the day-ahead forecast from ENTSO-E. The features consisted of weather features, calendar features and lagged load and error features. They are listed in Chapter 3.1.2 in detail with their exact sources.

We used the residual of the day-ahead forecast as target, consisting of the ENTSO-E day-ahead forecast and the actual load from TransnetBW. As feature variables we used weather data, calendar data and lagged load and residual features. All data was provided in quarter-hourly format. In order to prevent double datapoints or gaps that result from the switching between summer and winter time in Germany, it is necessary to estimate the data for some timestamps. In the case of switching from summer to winter time in october, when there are two values, the mean of those is taken for each quarter hourly data point. For the switch from winter to summer time in march, the missing values are linear interpolated.

Overall, the dataset contained data from 01-01-2021 to 01-10-2025.

3.1.1. Load Data

TransnetBW provided us with a quarter-hourly load dataset which was closer to the one on which their day-ahead forecast model was trained. This dataset differs from the load data which is published to the ENTSO-E Transparency Platform as the data there is retrospectively overwritten with more accurate metered values within approximately two months of the delivery day.

Table 3.1 quantifies this divergence by comparing the residual of the day-ahead total load forecast with each load dataset over the full study period. The residual is defined as $load - forecast$.

Table 3.1.: Comparison of errors between the two load datasets for the day-ahead total load forecast.

Metric	ENTSO-E load	TransnetBW load
MAE	444.37	238.22
RMSE	526.57	322.03

The TransnetBW load reduces the MAE by 46% and the RMSE by 39% relative to the ENTSO-E load, suggesting that this load dataset is closer to the ground truth of the day-ahead forecast model.

In this thesis we therefore used the load data from TransnetBW, as we deemed the significance and outcome of this thesis to be more valuable if we use the same ground truth as the model's residual we want to predict.

3.1.2. Dataset Components

The ENTSO-E Transparency report states the following for their model's features dataset: "The day-ahead forecast is calculated (estimated) on the historic load profile on similar days, taking into account the variables that affect electricity demand, such as weather conditions, climate and socioeconomic factors" [ENT23]. On that basis we chose the dataset feature components out of four categories:

Table 3.2.: Overview of all input features grouped by category. Lag offsets are given in hours; binary features take the value 1 (true) or 0 (false).

Category	Feature	Unit/Range	Source
Target	Day-ahead total load forecast	MW	ENTSO-E
	Actual load	MW	TransnetBW
Weather	Air temperature	°C	ERA5 [Her+17]
	Irradiance	W/m ²	
	Precipitation	m	
	Wind speed (10 m)	m/s	
	Wind speed (100 m)	m/s	
	Sea-level pressure	Pa	
Historic Load	Load, lagged 24 h	MW	Derived
	Load, lagged 168 h	MW	
	Load, lagged 336 h	MW	
	Forecast residual, lagged 24 h	MW	
	Forecast residual, lagged 168 h	MW	
	Forecast residual, lagged 336 h	MW	
Calendar	Hour	0–23	Python datetime [Pyt25]
	Day of week	0–6	
	Week of year	1–53	
	Is holiday	binary	Python holidays [DRM+25]
	Is school holiday	binary	BW school holidays [sch26]
	Is long weekend	binary	Derived

Table 3.2 lists all input features grouped into four categories, following the factors named in the ENTSO-E Detailed Data Descriptions [ENT23]: climate factors, historic load profiles (estimated by lagged load and error features), and socioeconomic factors (estimated by calendar features). Weather features were taken from the ERA5 global reanalysis [Her+17]. Calendar features were derived from the timestamp using the Python datetime [Pyt25] and holidays [DRM+25] libraries with Baden-Württemberg set as the province parameter. The is long weekend flag is set to 1 for any working day directly adjacent to a public holiday or school holiday period. School holidays were taken from [sch26].

3.2. Correlation of features

To assess the linear relationships between input features, pairwise Pearson correlations are computed across the full dataset. For two features j^a and j^b with n observations, the Pearson correlation coefficient is defined as

$$r(j^a, j^b) = \frac{\sum_{t=1}^n (j_t^a - \bar{j}^a) (j_t^b - \bar{j}^b)}{\sqrt{\sum_{t=1}^n (j_t^a - \bar{j}^a)^2} \cdot \sqrt{\sum_{t=1}^n (j_t^b - \bar{j}^b)^2}}, \quad (3.1)$$

where $\bar{j}^a$ and $\bar{j}^b$ denote the respective feature means. The coefficient r ranges from -1 to $+1$, where $+1$ indicates a perfect positive linear relationship, -1 a perfect negative linear relationship, and 0 the absence of any linear relationship. High absolute values between two input features indicate multicollinearity, which can make it difficult to isolate the individual contribution of each feature to the model prediction. The resulting correlation matrix is shown in Figure 3.1:

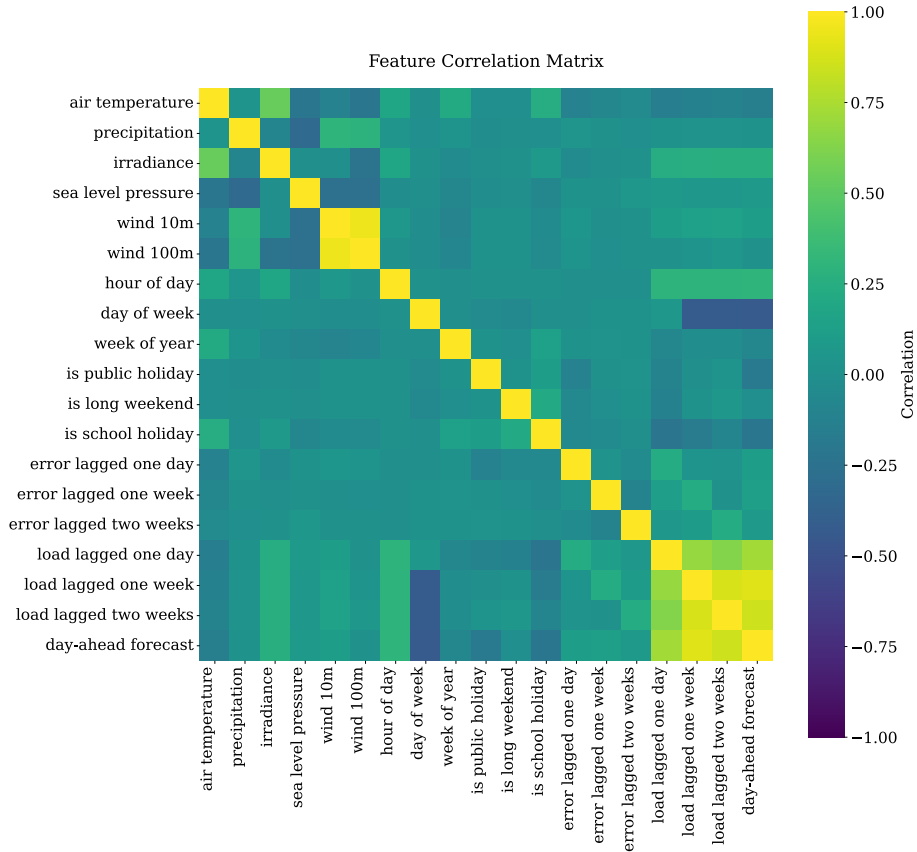


Figure 3.1.: Feature correlation matrix displaying pairwise Pearson correlations between all input features, ranging from -1 (dark purple) to $+1$ (bright yellow), with the diagonal equal to 1.0 by definition.

The most structurally notable pattern is the strong negative correlation between the day-ahead forecast and day of week. This reflects the well-known weekly seasonality of electricity demand, where consumption is systematically higher on workdays and lower on weekends. The relationship between the lagged load features 168 and 336 hours and day of week can be explained because they are exactly the one and two week lagged features and therefore always on the same day.

The three lagged load features (by one day, one week and two weeks) show moderate to strong mutual correlations, which is expected given the high autocorrelation inherent in electricity demand time series. Despite this, each lag captures a distinct temporal scale justifying their joint inclusion. The lagged error features (residual lagged by 24, 168 and 336 hours), by contrast, exhibit low mutual correlation, indicating that forecast residuals at different temporal offsets carry largely independent information about the systematic bias in the ENTSO-E forecast.

Among the meteorological features, the two wind speed variables at 10m and 100m are strongly correlated with each other, which is physically expected as both measure the same atmospheric wind field at adjacent heights. Despite this collinearity, both variables are retained in the feature set: wind speed at 10 m is more representative of surface-level weather effects, while wind speed at 100 m is the standard reference height for utility-scale wind turbine generation. The remaining weather features show only weak inter-correlations. Also the calendar features which are indicating holidays (holiday, school holiday and bridge day) features are weakly correlated with most other features.

Overall, the matrix reveals no critical multicollinearity requiring feature removal. The diverse correlation structure across feature groups confirms that each feature contributes independent predictive signal, supporting the decision to retain the full feature set in the final model.

3.3. Dataset Split

In load forecasting, the state-of-the-art approach is to use blocked cross-validation as dataset split, where the temporal order of the dataset is strictly respected and train, validation and test data always follow each other chronologically and multiple models are trained on different folds. [CTM20] [BB12] [BHK18].

However, our research question is fundamentally different as we are not developing a model for future time series forecasting but rather want to analyze the existing systematic errors of the ENTSO-E day-ahead forecast model.

3.3.1. Group Shuffle Split

To enable a robust evaluation of the systematic errors in the day-ahead forecast, we employed a three-way data partition using scikit-learn's GroupShuffleSplit [sci25]. Cal-

endar weeks were used as natural temporal blocks to ensure that week-level temporal integrity was preserved and no group was split across training, validation, and test sets. This methodological choice reflects both the specific research objective and principles of group-aware data splitting for time series data.

Cerqueira et al. (2020) [CTM20] conducted an extensive empirical study comparing 13 different performance estimation methods for time series forecasting across 174 real-world datasets. Their key finding that block-based approaches outperform random partitioning—is relevant to justifying our choice of group-based splitting. Though Cerqueira et al. focus on iterated block CV with multiple models, the underlying principle applies: partitioning data by meaningful temporal blocks (weeks) is superior to naive random shuffling, even in a single-split context. Random shuffling is particularly problematic for time series data because adjacent timestamps are often highly similar, meaning that randomly assigned train and test sets share substantial mutual information and can yield optimistically biased performance estimates.

3.3.2. Train-Validation-Test Split

The grouped weeks were randomly allocated into three disjoint sets:

- 70% Training Set (172 weeks)
- 20% Validation Set (49 weeks)
- 10% Test Set (24 weeks)

This allocation provides a sufficiently large training dataset while reserving adequate validation data for hyperparameter tuning. The test set contains completely independent holdout data, used exclusively for the final SHAP analysis and model evaluation.

3.4. Error Metrics

To evaluate our model in its capabilities to forecast the actual residual, we looked at different classical error metrics commonly used in load forecasting and decided on some of them to take a closer look at:

3.4.1. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE)

$$MAE = \frac{1}{n} \sum_{t=1}^n |Y_t - F_t|$$

with Y_t being the actual value and F_t being the forecast.

For electricity market applications, the Mean Absolute Error is particularly relevant as balancing costs are typically proportional to absolute deviations from scheduled positions. The model's prediction is therefore relevant as the forecast errors are not purely random but contain exploitable structure.

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (Y_t - F_t)^2}$$

with Y_t being the actual value and F_t being the forecast.

The RMSE penalizes larger errors more heavily than the MAE due to the squaring of residuals, making it particularly sensitive to outliers and extreme forecast deviations. In electricity markets, this property is especially relevant during periods of high price volatility or unexpected demand spikes, where large individual errors can have disproportionate consequences for grid balancing and market participants.

3.4.2. Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{Y_t - F_t}{Y_t} \right|$$

with Y_t being the actual value and F_t being the forecast.

As Hyndman and Koehler (2006) [HK06] note, the greatest limitation of the MAPE is, that it "has the disadvantage of being infinite or undefined if $Y_t = 0$ for any t in the period of interest, and having extreme values if any Y_t is close to zero". In the context of day-ahead residuals this happens when the forecast is highly accurate, so the actual residual approaches zero. Then the denominator of the MAPE becomes close to or zero resulting in disproportionately large percentage errors for small absolute predictions. As Hyndman and Koehler (2006) further note, MAPE "puts a heavier penalty on negative errors than on positive errors" [HK06], creating asymmetric penalties that can bias model optimization additionally and lead to a disbalanced metric overall.

Given these limitations, MAPE is not a suitable metric for comparing model performance in this application and will not be used for further analysis.

3.4.3. symmetric Mean Absolute Percentage Error (sMAPE)

$$sMAPE = \frac{1}{n} \sum_{t=1}^n \frac{|F_t - Y_t|}{(|Y_t| + |F_t|)/2}$$

with Y_t being the actual value and F_t being the forecast.

The symmetric Mean Absolute Percentage Error (sMAPE) was designed to address the asymmetry of MAPE by averaging the absolute actual and forecast values in the denominator. However, it suffers from its own fundamental shortcomings. When both actual and forecast values are close to zero, the denominator remains small, leading to numerical instability analogous to that of MAPE. Moreover Hyndmann and Koehler (2006) note that "the value of sMAPE can be negative, so it is not really a measure of 'absolute percentage errors' at all" and explicitly recommend "the sMAPE not be used" [HK06].

Following Hyndman and Koehler's recommendation, sMAPE is likewise not used as a primary evaluation metric in this analysis.

3.4.4. Mean Absolute Scaled Error

The Mean Absolute Scaled Error (MASE) was proposed by Hyndman and Koehler (2006) [HK06] specifically to overcome the limitations of percentage-based error metrics. The metric scales forecast errors by the mean absolute error of a naive seasonal forecast:

$$MASE = \frac{MAE}{\frac{1}{n-s} \sum_{t=s}^n |Y_t - Y_{t-s}|}$$

where s denotes the seasonal period, Y_t being the actual value and F_t being the forecast.

For this quarter-hourly dataset, we evaluate two seasonality benchmarks: a daily and weekly seasonal component with $s_{daily} = 96$ or $s_{weekly} = 672$. This daily and weekly seasonal components reflect a good balance between capturing the most relevant seasonal structure and preserving the maximum number of usable observations, as choosing a large s results in a larger part of the dataset being excluded as s data points are needed before the first evaluated data point to calculate the naive forecast.

The MASE moreover has several advantages: it is scale independent, easy to interpret and robust to outliers. When interpreting the MASE a score of 1.0 and less indicates a smaller error than the naive seasonal forecast.

4. Methodology

4.1. Model Setup

We select eXtreme Gradient Boosting (XGBoost) as our modeling framework for analyzing systematic errors in the ENTSO-E day-ahead electricity forecast.

4.1.1. Model Parameters and Hyperparameters

We implemented an XGBoost regressor with the following hyperparameter configuration:

Listing 4.1: model setup

```
model = xgb.XGBRegressor(  
    max_depth=5,  
    n_estimators=240,  
    learning_rate=0.2,  
    random_state=42  
)
```

The exact parameters were evaluated by training multiple models and evaluating their results based on RMSE on the validation set.

- **max_depth = 5** [default=6]: This parameter defined the depth of one weak learner decision tree. A depth of 5 provides a balance between model complexity and generalization. In multiple model trainings with values ranging from 3 to 7 this values yielded the best results even when changing the other parameters at the same time.
- **n_estimators = 240** [default=100]: This parameter specifies the total number of decision trees to sequentially add to the ensemble. We tested multiple settings in the range of 50-300 for the different model settings and most yielded the best results around 200-250 for the different models. This is why we chose a middle value best performing for most of our models.
- **learning_rate = 0.2** [default=0.3]: : The learning rate controls the contribution of each new tree to the ensemble's prediction. The parameter was tested using ranges from 0.1 to 0.3 and was tuned together with n_estimators to achieve the best result.

- **random_state = 42**: this parameter ensures reproducibility of our results.

4.1.2. Model Evaluation

Model performance was evaluated exclusively on the held-out test set, which was not used during training or validation. This ensures an unbiased assessment of the model's generalization capability to unseen data.

Furthermore, multiple model configurations were evaluated on different feature sets separating the different feature groups of weather, calendar and lag features and then systematically recombining them to evaluate the interaction and accuracy of the groups and determine a final most exact model.

4.2. Selection of most accurate Data Subset

To further evaluate the most influential errors and compare them with the ones from the SHAP analysis on the test set, a data subset was defined with all data points, where we especially believe the models prediction and deem it most relevant. This subset represents predictions where the model successfully explains the ENTSO-E forecast residual sources with minimal error itself. We then used this subset to perform a SHAP analysis. The overall goal is to determine whether certain features or feature interactions are particularly important when the model performs well, providing insights into conditions under which the systematic forecast residual is most predictable.

Following the recommendations of Hyndman and Koehler (2006) [HK06], we use MASE as the splitting criterion for relative exact predictions due to its favorable statistical properties and robustness to near-zero values. For each prediction t in the dataset, we calculate the per-sample scaled error:

$$ScaledError_t = \frac{|Y_t - F_t|}{MAE_{naive}}$$

with

$$MAE_{naive} = \frac{1}{n - 168} \sum_{t=168}^n |Y_t - Y_{t-168}|$$

as the mean absolute error of a naive weekly seasonal forecast computed over the time period. We chose the weekly seasonal period as it captured the patterns of the dataset more accurately.

We define the high-accuracy subset as predictions where:

- $ScaledError_t < 0.5$: model error is less than 50% of naive forecast error
- Sign prediction is correct: model prediction and actual residual have the same sign

This dual criterion ensures we analyze predictions that are both quantitatively accurate and qualitatively correct in terms of forecast deviation direction.

This leaves us with a dataset of 60.22% (147 weeks) of the whole data points.

4.3. SHAP

Understanding which features most strongly influence the systematic errors (residuals) of the ENTSO-E forecast is critical for error characterization. To accomplish this, we employed SHAP analysis, a theoretically grounded and widely-adopted method for explaining machine learning model predictions [Lun+20].

4.3.1. SHAP Analysis on test set

For computing SHAP values, we used the TreeExplainer, which is specifically optimized for tree-based models like XGBoost.

The explainer was initialized with the full feature matrix X , containing all training, validation, and test features as background data. This ensures the SHAP values are computed with respect to the true feature distributions.

Providing a background dataset configures the Tree Explainer to compute SHAP values via exact tree traversal, where the background data defines the reference distribution used to calculate expected model outputs at each node [Lun+20]. This ensures that the resulting SHAP values reflect the true marginal feature contributions with respect to the empirical feature distribution of the background data, rather than relying on the tree's native path-dependent weighting.

The SHAP explanation and SHAP values were computed exclusively on the held-out test set (X_{test}), which had not been used during model training. This maintains the independence of our error analysis and prevents bias from training-induced feature correlations.

4.3.2. SHAP Analysis on most accurate Data Subset

To understand how feature importance patterns differ between the test dataset and the data subset with the lowest error, we conducted a separate SHAP analysis on a subset of predictions with minimal errors we defined in 4.2.

The same explainer configuration is used, so the SHAP values of both analyses are directly comparable. By comparing SHAP values between the test dataset and the subset of most

accurate predictions, we can identify which features are crucial for explaining systematic errors and are therefore present in both sets.

5. Evaluation

5.1. Analysis of different trained Models

To systematically analyze the day-ahead forecast residuals, multiple model configurations were trained, each designed to isolate the contribution of different feature categories as defined in Chapter 3.1.2:

- only calendar features + forecast (**CF**)
- only weather features + forecast (**WF**)
- only lag features + forecast (**LF**)
- calendar + weather features (**CWF**)
- weather + lag features (**WLF**)
- lag + calendar features(**LCF**)
- all features (**ALL**)

Model performance was evaluated on different error metrics as described in Section 3.4:

Table 5.1.: Performance comparison of models trained on different feature subsets. CF = calendar features, WF = weather features, LF = lag features, CWF = calendar + weather features, LCF = lag + calendar features, WLF = weather + lag features, ALL = all features combined.

Metric	CF	WF	LF	CWF	WLF	LCF	ALL
MAE	256.77	192.42	183.69	161.18	159.45	237.77	139.34
RMSE	241.42	258.54	246.34	218.94	215.26	317.03	188.69
MASE (daily)	0.74	0.73	0.69	0.61	0.60	0.60	0.53
MASE (weekly)	0.56	0.55	0.52	0.46	0.45	0.45	0.40
Accuracy (%)	67.13	68.38	70.31	74.64	74.88	74.97	78.62

As the results in 5.1 show, among the single feature model group the lag features only model (LF) performs best, indicating that historical load patterns provide strongest predictive signal.

Nevertheless, the results demonstrate clear performance gains from feature integration. The ALL model achieves the lowest error across all metrics, with particularly notable improvements in MASE (weekly): 0.40 compared to 0.52-0.56 for individual feature groups and 0.45-0.46 for dual feature groups. This comes down to a 23-29% improvement over the next-best single feature model (LF) and 9-11% improvement over dual feature groups.

The substantial improvement obtained by combining all feature categories as well as the improvements made by combining the features to dual feature groups demonstrates that calendar, weather and lag features contribute complementary, non-redundant information to residual prediction.

5.2. Prediction and Error Analysis of best model

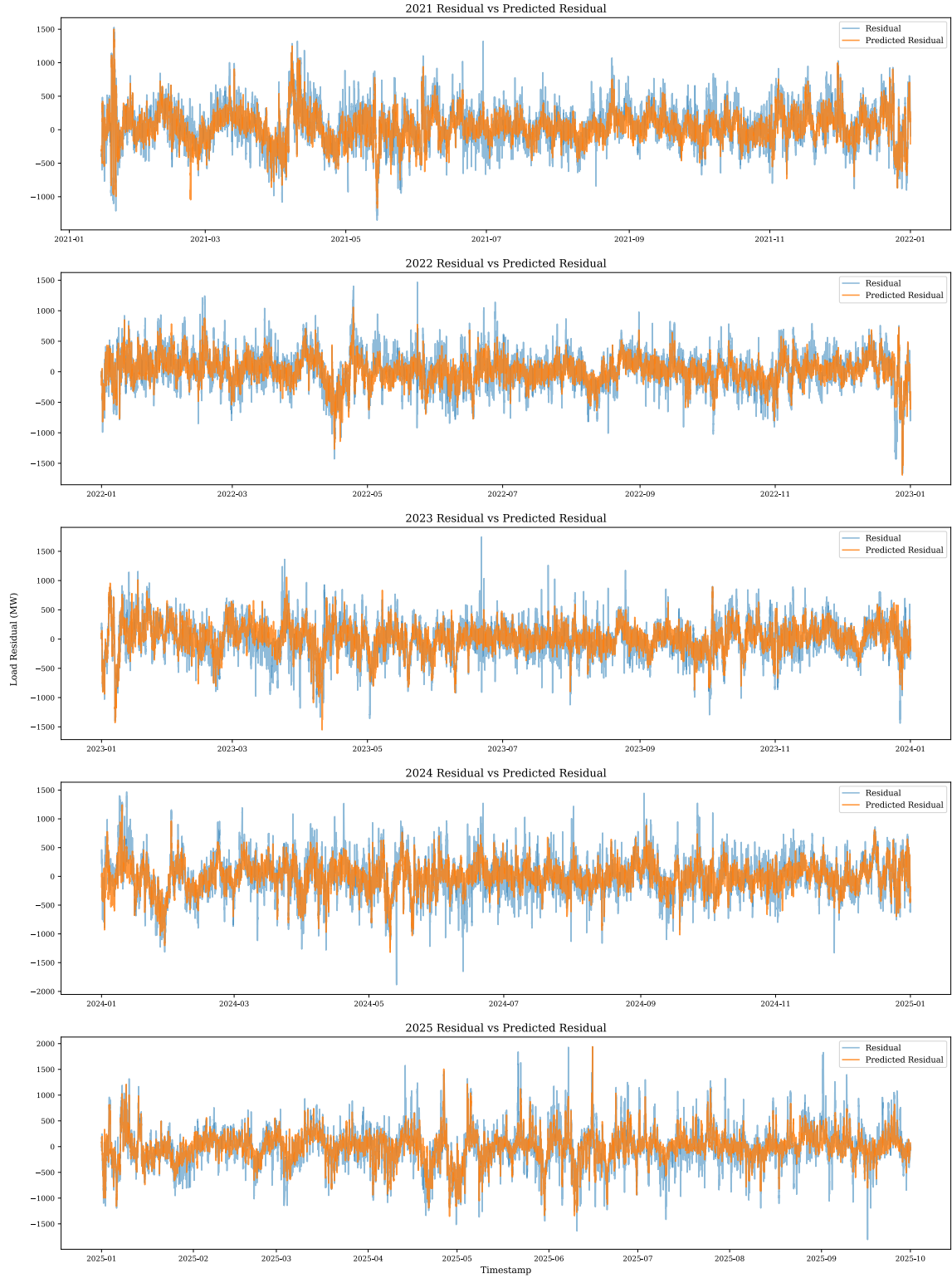


Figure 5.1.: Residual of day-ahead forecast and predicted day-ahead residual

Based on the analysis before, the model with the whole feature set as input is identified as the best-performing configuration. When taking a closer look at this model and its predictions, one can see most of the day-ahead residuals peaks are captured with higher accuracy.

In the following sections, we examine different error metrics in detail to comprehensively evaluate the model's performance.

5.2.1. MAE and RMSE

The following Table compares the residual prediction model's performance against the magnitude of the original day-ahead forecast errors, providing context for the achievable improvement.

Table 5.2.: Comparison of forecasting errors between the best model and the day-ahead (DA) total load forecast baseline.

Metric	Best Model	DA Total Load Forecast
MAE	139.34 MW	238.22 MW
RMSE	188.69 MW	322.03 MW

The residual prediction model exhibits substantially lower MAE and RMSE values than the original day-ahead forecast errors. This indicates that systematic components of the forecast error are predictable based on the available feature set. It can furthermore be deduced that the magnitude of the predictable residual is smaller and more structured than the total error of the baseline forecast.

While the model itself does not modify the day-ahead forecast, the accuracy of the residual predictions suggests that a correction mechanism or post-processing step could, in principle, reduce forecast uncertainty. But even if the model outperforms the day-ahead forecast in terms of MAE and RMSE by 41%, this does not mean that the actual error of the day-ahead forecast can be improved at the same rate. First, the ratio of RMSE and MAE of 1.35 indicates that occasionally large prediction errors occur. Moreover, also the sign prediction accuracy of 78.62% as in Table 5.1, even if meaningfully above the 50% random baseline, means that approximately in one of 5 cases the model predicts a correction in the wrong direction. This overall means that, in some cases, the day-ahead forecast residual is not captured well enough, resulting in an increased predicted residual if it were to be applied directly as a correction.

The ratio of $RMSE/MAE = 1.35$ indicates a moderate spread in the error distribution, suggesting that while most predictions are reasonably accurate, occasional larger errors occur.

5.2.2. MASE

The model achieves a MASE of 0.40 when benchmarked against weekly seasonal naive forecasts and 0.53 against daily seasonal naive forecasts. Both values being substantially below 1 demonstrate that the XGBoost model significantly outperforms naive baseline forecasts. The stronger performance against the weekly baseline (lower MASE) suggests that the model effectively captures day-of-week patterns, which are prominent in electricity load data due to differences between weekdays and weekends.

5.3. SHAP Analysis of Best Model

5.3.1. SHAP Bar and Beeswarm plot analysis

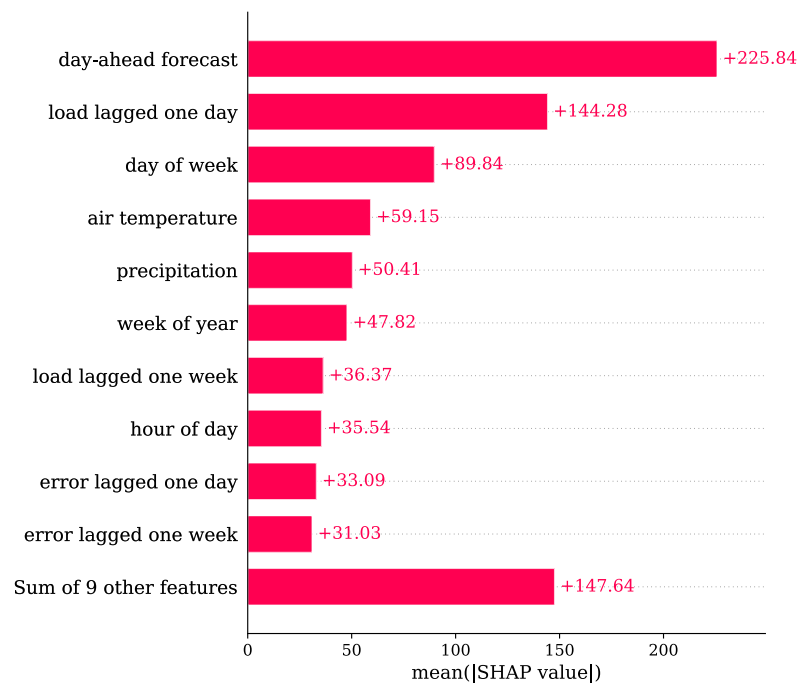


Figure 5.2.: Bar Plot for the best model with the top ten mean absolute SHAP features

As illustrated by the SHAP Bar Plot in Figure 5.2, the most important features are the day-ahead load forecast, the load from 24 hours before, the day of the week, the air temperature as well as the precipitation. When looking at the beeswarm plot 5.3, most features exhibit both positive and negative SHAP values and are centered around zero. This indicates that they contribute to both over- and underprediction depending on context. Among the top features, the day of the week and air temperature display noticeable asymmetry in their contributions, suggesting systematic directional effects. These patterns are examined further using SHAP interaction and dependence plots.

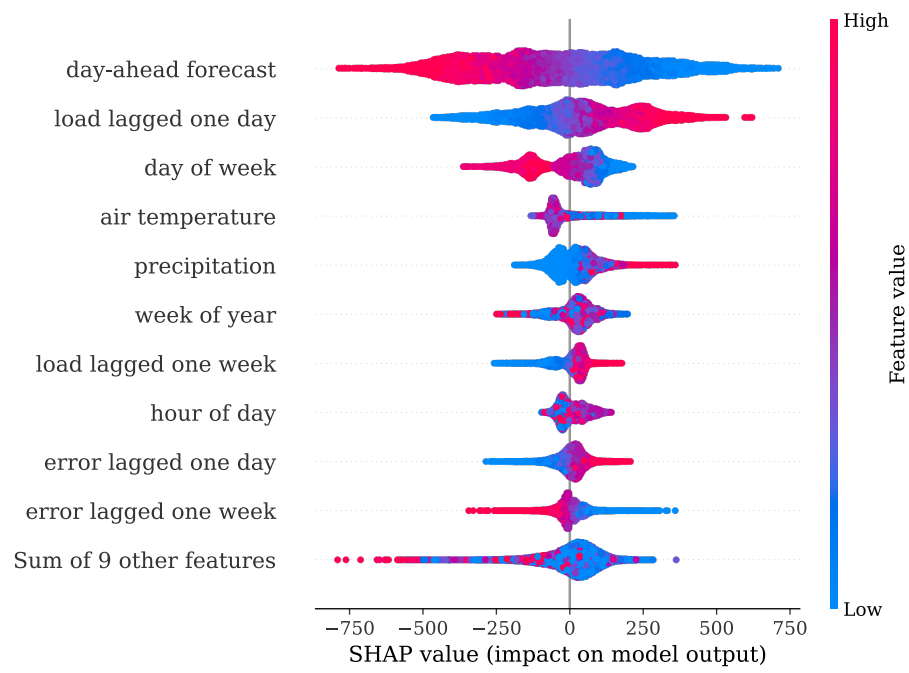


Figure 5.3.: SHAP beeswarm plot of the top ten features

5.3.2. SHAP Feature interaction analysis

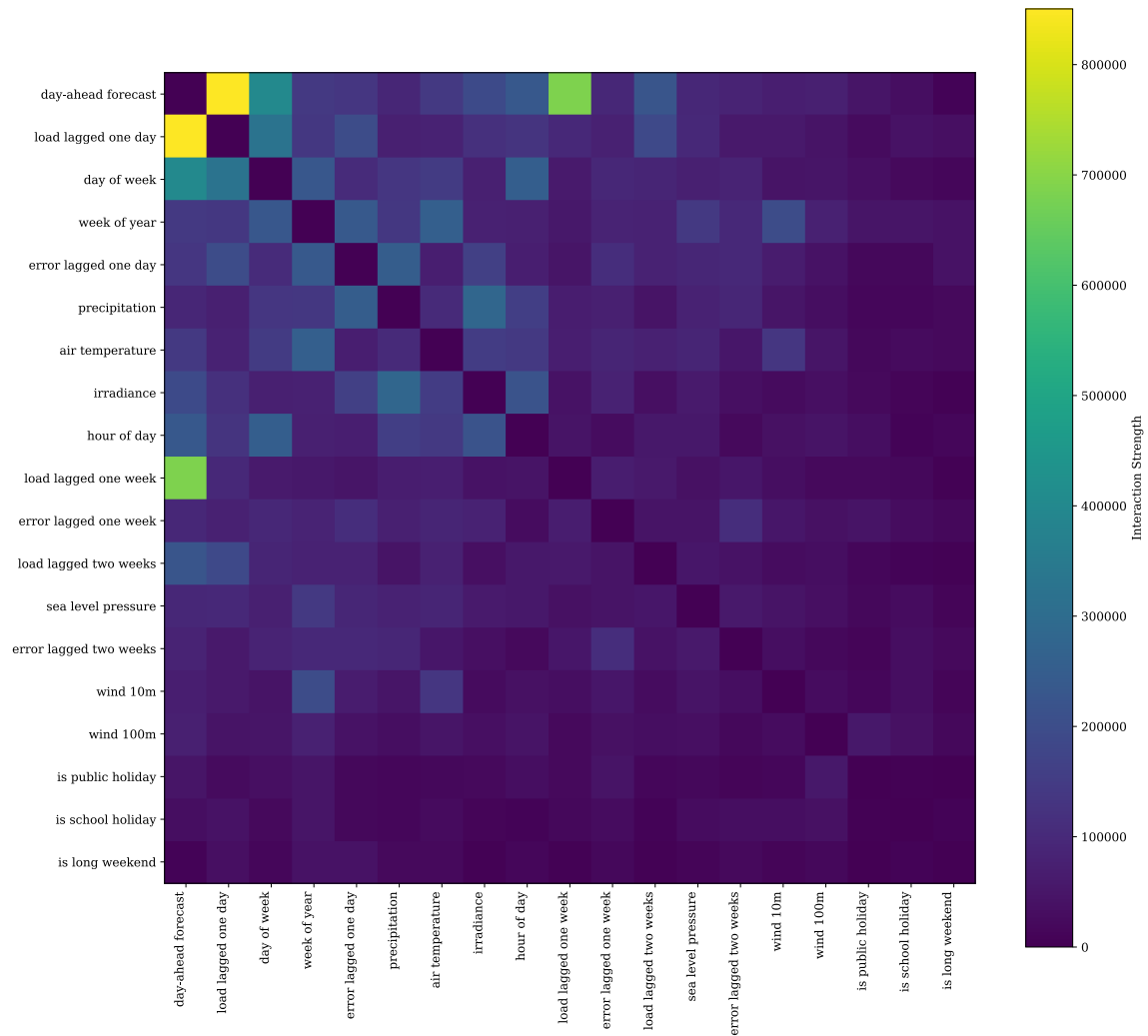


Figure 5.4.: Interaction matrix of the different SHAP features. SHAP values which show high interaction patterns are shown in lighter colors, whereas SHAP values which interact less are shown in darker colors.

The interaction matrix reveals strong interaction effects between the day-ahead forecast and different lagged load features, with interaction strength decreasing as the temporal distance of the lag increases. This behavior is expected, as the residual to be predicted is defined as $load - forecast$ and recent historical load provides context for systematic forecast bias. Additionally, on similar days or days which are directly following each other, the load as well follows similar patterns.

The SHAP value of the day of the week shows particularly strong interaction effects with the day-ahead forecast as well as with the lagged load 24 hours prior. This is plausible

because the day-ahead forecast itself encodes weekly demand patterns, as on most weekdays, industrial load profiles are structurally similar. So the marginal contribution of the day-of-week feature depends on whether the forecast has already accounted for this weekly structure, producing a measurable interaction between the two.

For the SHAP value of irradiance the interaction matrix highlights two dominant interaction partners: precipitation and hour. As a higher irradiance strongly correlates with clear-sky conditions and therefore also lesser precipitation, it is obvious that those features interact with each other and influence the model's output. Also as irradiance is naturally concentrated during daylight hours, which leads the model to interpret irradiance differently depending on the time of day. Overall this indicates that the model interprets irradiance in a context-dependent manner, conditioned on both weather consistency and time of day.

5.3.3. SHAP Dependence plot analysis

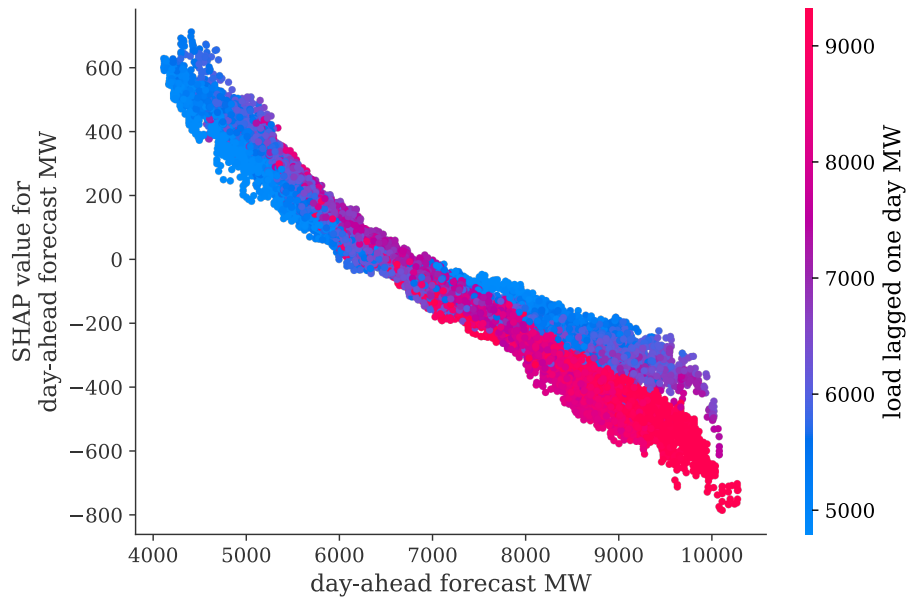


Figure 5.5.: SHAP Dependence Plot of the feature day-ahead total load forecast.

The dependence plot of the day-ahead forecast feature in Figure 5.5 shows a linear, negative relationship between feature value and SHAP value: higher forecast values are associated with more negative SHAP values, while lower forecast values are associated with more positive ones. Since the residual is defined as $actual - forecast$, a negative SHAP value indicates that the model expects actual load to fall short of the forecast, while a positive SHAP value indicates the opposite. The plot therefore suggests a tendency toward overprediction at higher load levels and underprediction at lower load levels. The feature with the strongest interaction is the 24-hour lagged load, suggesting that this systematic pattern persists across consecutive days with similar load profiles.

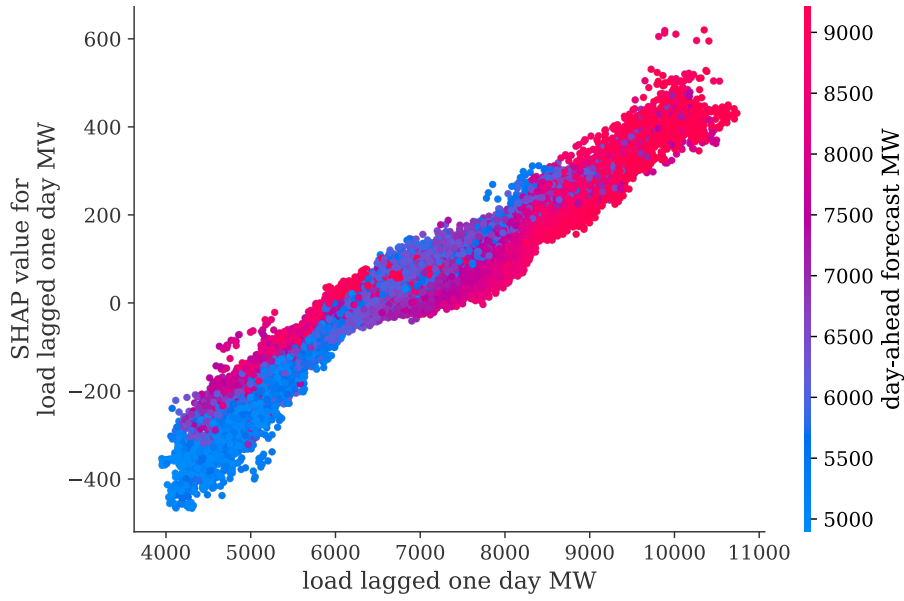


Figure 5.6.: SHAP Dependence Plot of the feature load lagged by 24 hours.

Inspecting the dependence plot for the 24-hour lagged load feature in Figure 5.6, it is noticeable that it also displays a near-linear relationship: higher lagged load values are associated with more positive SHAP values, while lower lagged load values are associated with more negative ones. This mirrors the behavior of the day-ahead forecast feature with inverted directionality, which follows naturally from the residual definition as *actual* – *forecast*. The near-linear pattern across the full range of lagged load values suggests that the systematic relationship between historical load levels and forecast error is consistent and not limited to extreme cases.

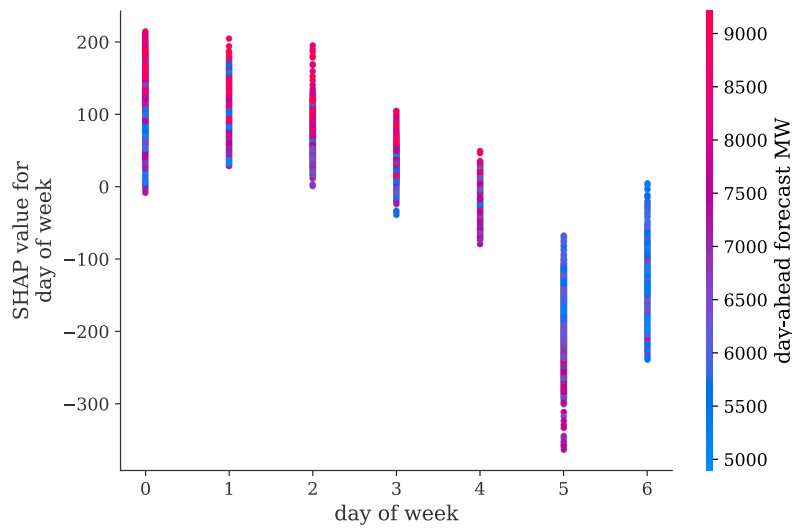


Figure 5.7.: SHAP Dependence Plot of the feature day of week.

For the feature day of week, the SHAP dependence plot 5.7 shows, that on most week-days the model underpredicts the load whereas on the weekend, especially on Saturdays the load is more likely to be overpredicted. This pattern reflects structural differences between industrial and residential load profiles and indicates insufficient differentiation in the day-ahead forecast. When looking at the strongest interaction feature of the day-ahead forecast, we can see the same pattern as before in the analysis of its dependence plot: the higher the load and forecast levels the higher the over- or underprediction on weekdays and weekends.

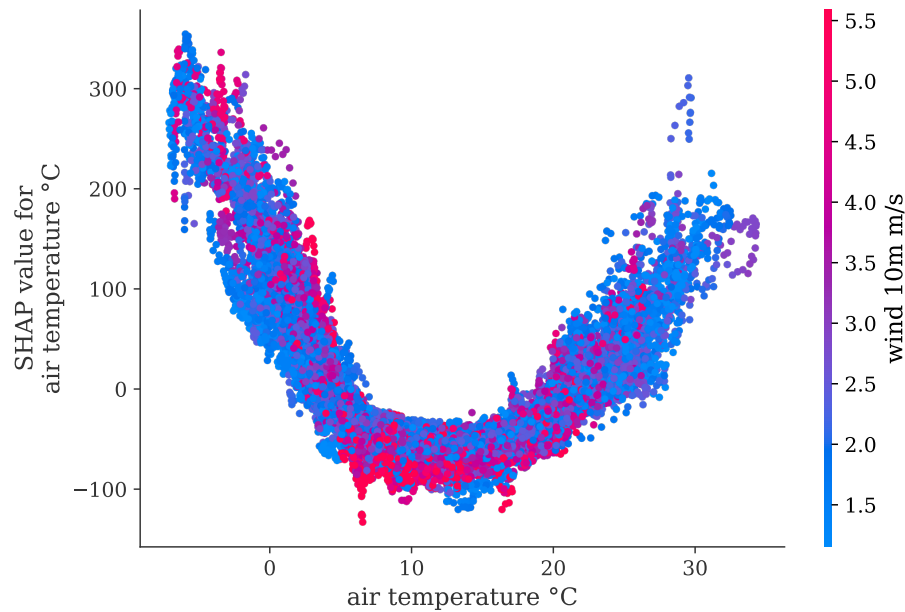


Figure 5.8.: SHAP Dependence Plot of the feature air temperature.

The SHAP dependence plot for air temperature 5.8 reveals a U-shaped pattern: extreme temperatures (below 0°C and above 25°C) are associated with positive SHAP values, indicating underprediction at temperature extremes due to increased heating and cooling demand. At moderate temperatures, slight overprediction is observed (negative SHAP values). The strongest SHAP interaction is with wind speed at 10 meters, reflecting coupled atmospheric dynamics between temperature and wind patterns. The underestimation at low temperatures is consistent with findings by [Sem+24], which shows that electrically driven heating technologies like heating pumps are not yet adequately captured in many load forecasting models.

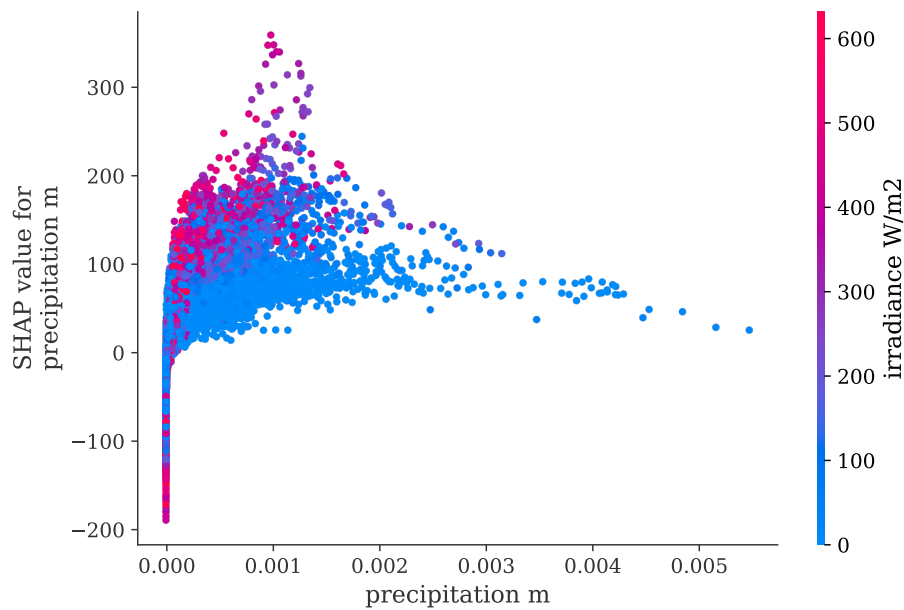


Figure 5.9.: SHAP Dependence Plot of the feature precipitation.

Looking at the dependence plot of the feature precipitation 5.9, it can be observed that precipitation predominantly contributes to positive SHAP values, indicating underprediction during rainy conditions. It interacts most with the feature irradiance which is expected given their inverse physical relationship. The plot also shows that on data points where both features are high, indicative of unstable or rapidly changing weather conditions, the residual is also highly influenced.

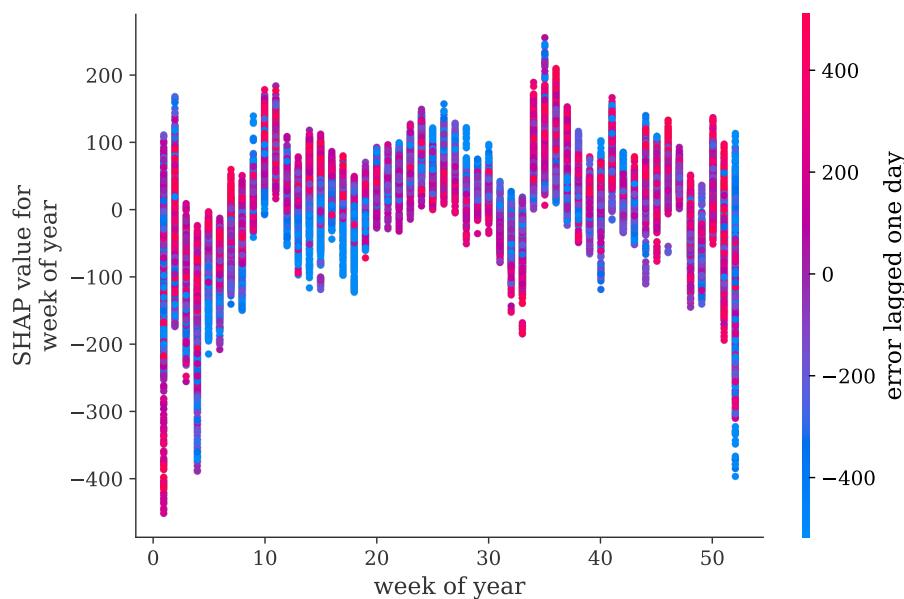


Figure 5.10.: SHAP Dependence Plot of the feature week of year.

The SHAP dependence plot for the feature week of year 5.10 highlights systematic over-prediction at the beginning and end of the year. These periods coincide with extended holiday seasons, during which industrial activity is reduced and overall electricity demand decreases. For the rest of the weeks the SHAP values show comparatively low variance and are mostly centered around zero with small outliers mid of the year and around easter period, where the load is rather underpredicted.

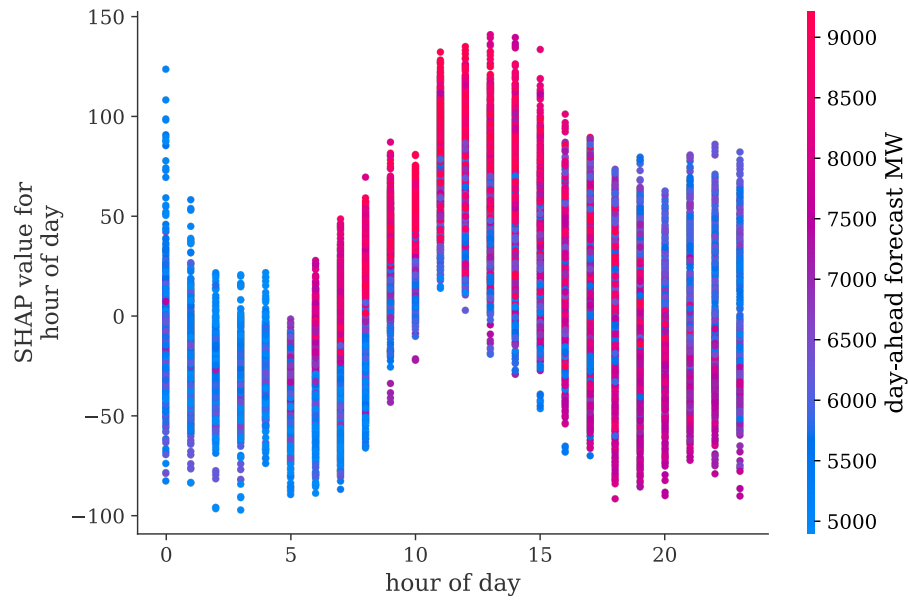


Figure 5.11.: SHAP Dependence Plot of the feature hour.

When looking at the SHAP dependence plot for the feature hour in Figure 5.11, it shows a strong interaction of an increased SHAP value of the feature during daytime hours indicating underprediction during periods of elevated demand between 8am and 6pm. The strongest interaction is with the day-ahead forecast itself, which is naturally high during these same daytime periods. This suggests that the hour feature captures a within-day underprediction pattern that is not fully explained by the forecast level alone, and that the two features jointly characterize this systematic daytime error.

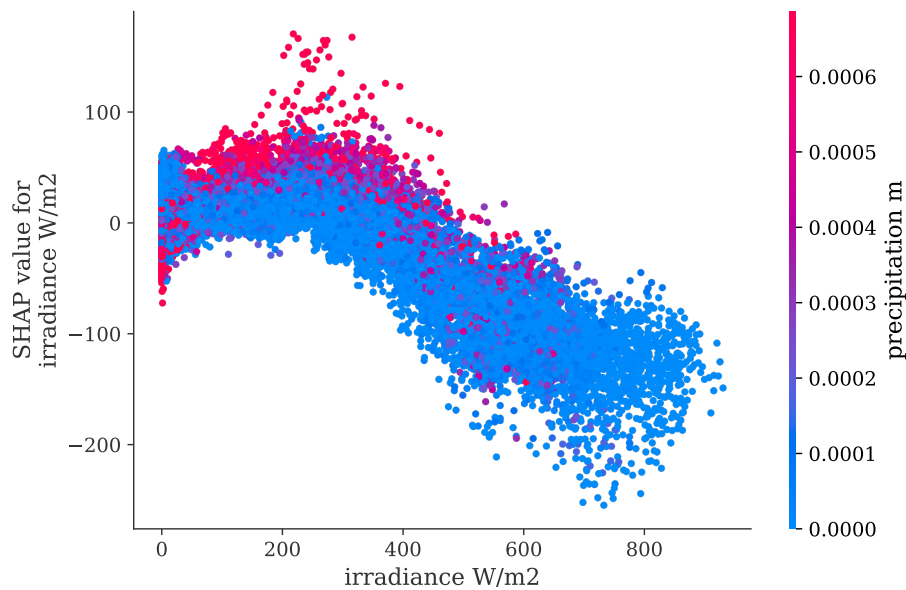


Figure 5.12.: SHAP Dependence Plot of the feature irradiance.

The SHAP dependence plot of the feature irradiance, shows that increasing irradiance is associated with increasingly negative SHAP values after irradiance exceeds a value of 400 W/m². This reflects the known impact of behind-the-meter (BTM) solar PV generation, which Cakir Erdener et al. (2022) [Erd+22] identify as a major challenge: "solar photovoltaic systems largely integrated within the distribution grid are operated 'behind-the-meter' and power generation cannot be directly monitored by most utilities" [Erd+22]. During periods of high irradiance, BTM PV reduces visible load without being explicitly modeled in the day-ahead forecast, creating systematic overprediction (negative residuals).

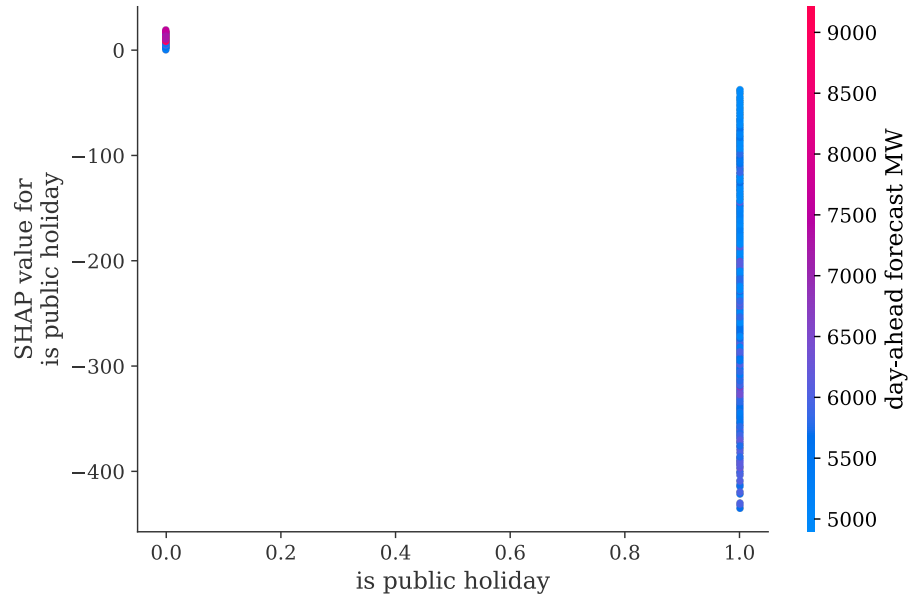


Figure 5.13.: SHAP Dependence Plot of the feature is holiday.

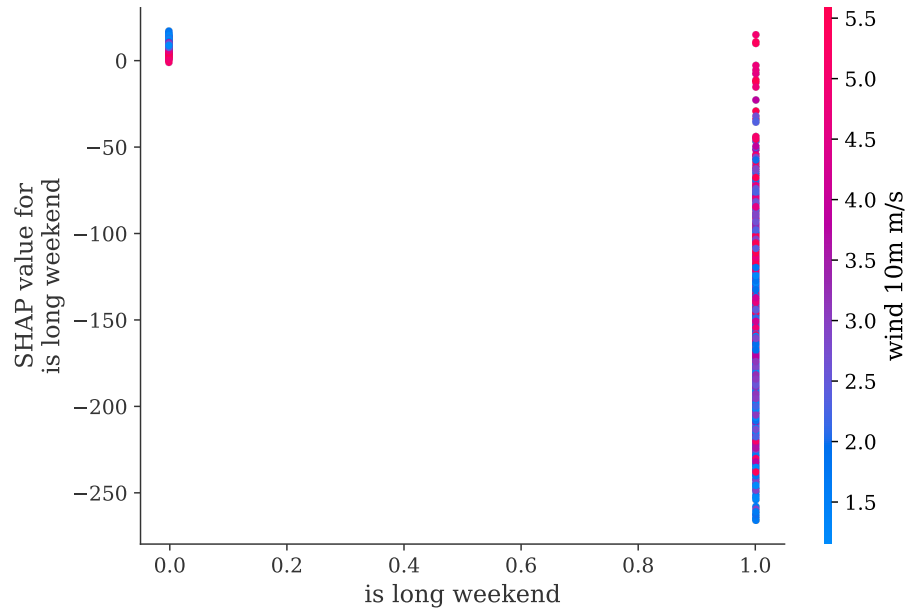


Figure 5.14.: SHAP Dependence Plot of the feature is long weekend.

The dependence plots of the holiday related features holiday and long weekend in Figure 5.14 and 5.13 show, that the day-ahead forecast does not take these features into account properly, thus leading to an overprediction on those days. This reflects reduced industrial demand on holidays and bridge days, during which consumption patterns deviate from typical weekday behavior, dominated by commercial load patterns rather than domestic load patterns.

5.4. SHAP Analysis of Best Model's most accurate Dataset Part

To identify systematic patterns in the model's most accurate predictions, we perform a SHAP analysis on the subset of predictions with lowest relative errors. This analysis aims to reveal whether certain features or feature interactions are particularly important when the model performs well, providing insights into conditions under which the systematic forecast error is most predictable.

5.4.1. SHAP Analysis of most accurate Data Subset

Performing a SHAP Analysis on the data subset with least relative error, leads to the following SHAP feature importance values:

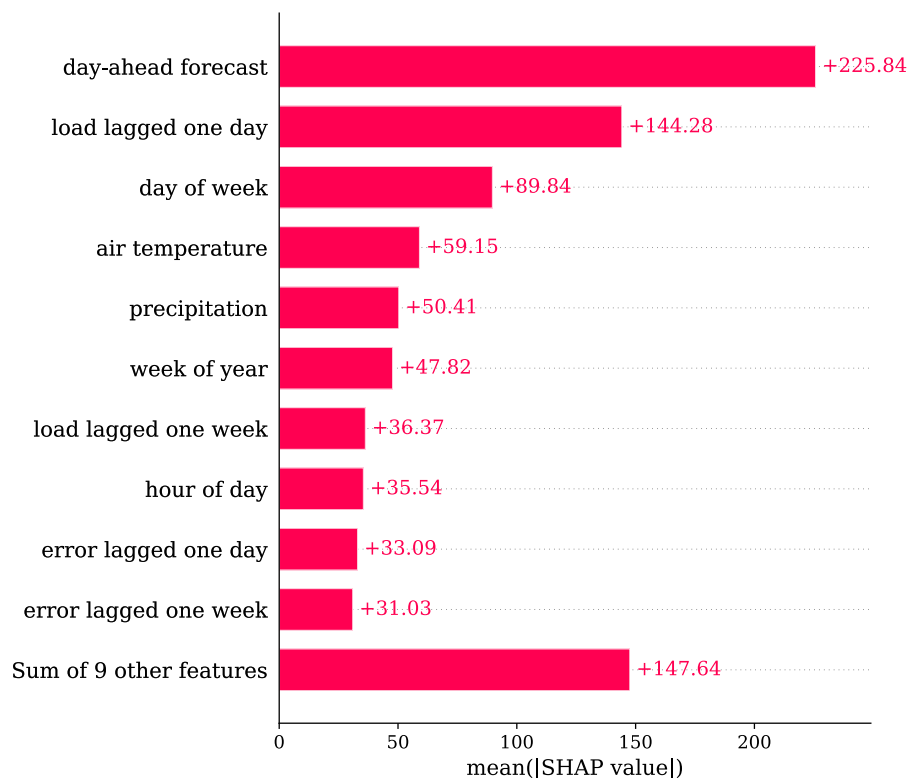


Figure 5.15.: SHAP Barplot for SHAP analysis on most accurate subset in terms of MASE of the data

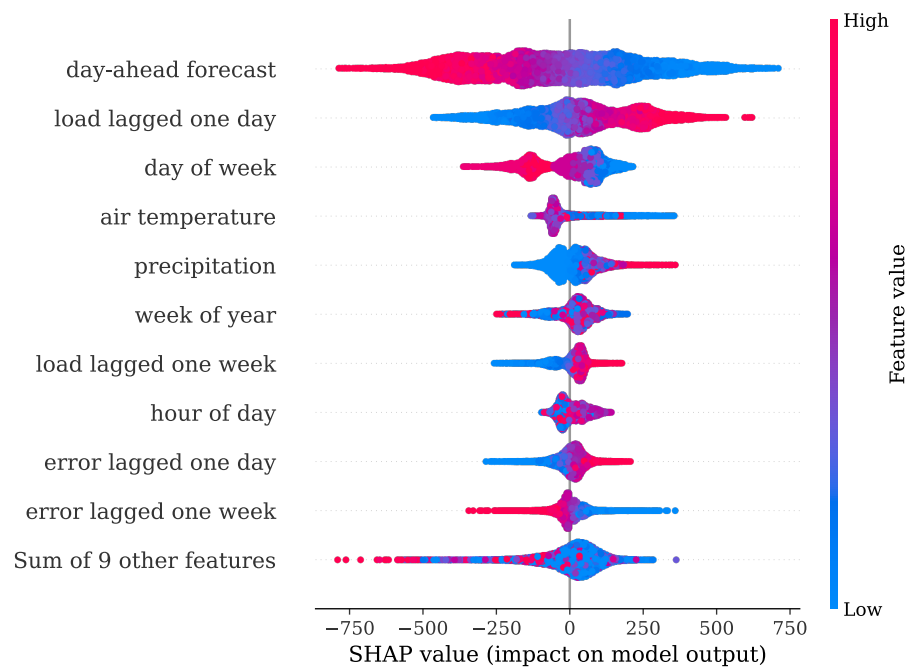


Figure 5.16.: SHAP Beeswarm plot for SHAP analysis on most accurate subset in terms of MASE of the data

Comparing the high-accuracy subset as described in Section 4.2 SHAP values (Figure 5.15) with the overall SHAP analysis (Figure 5.2) reveals both similarities and notable differences:

Increased importance of week of year: The most striking difference is the elevated importance of the week of year feature in the high-accuracy subset. This suggests that seasonal patterns have a particular high influence on the residuals of the day-ahead forecast.

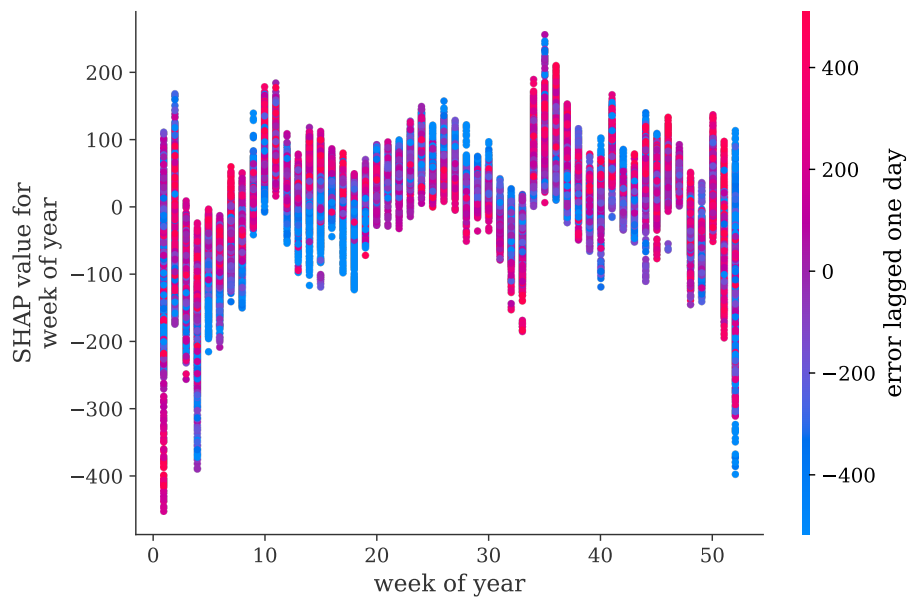


Figure 5.17.: SHAP Dependence Plot of the feature week of year

When taking a closer look at the SHAP dependence plot for week of year, it can be observed that weeks at the beginning and end of the year exhibit especially strong effects, consistent with the earlier SHAP analysis.

Consistent top features: The day-ahead forecast, load lagged by 24 hours, air temperature, and precipitation remain among the most important features in both analyses. Also, the feature influence variance as can be seen in the beeswarm plot, follows the same patterns as before. This consistency suggests these features provide reliable signal for residual prediction regardless of error magnitude.

6. Conclusion

This thesis aims to answer the research question of whether the ENTSO-E’s day-ahead forecast contains systematic, learnable error patterns.

To answer this research question, an XGBoost model was trained to predict the residuals of the day-ahead forecast. The feature set was constructed following the data categories described in the ENTSO-E Transparency Report [ENT23], comprising meteorological variables, historical load proxies, and calendar indicators, as listed in Table 3.2. This model was then analyzed with SHAP, first on the held out test dataset and then compared to the most accurate data subset.

The trained XGBoost model achieved a weekly MASE of 0.40 and an MAE of 139.34 MW on the held-out test set, outperforming the day-ahead forecast baseline by 41% in terms of MAE, which confirms that the residuals contain learnable structure beyond noise. The SHAP analysis revealed several distinct and physically interpretable sources of systematic error patterns. Firstly, the day-ahead forecast does not differentiate sufficiently between weekdays and weekends, thus underestimating the load on weekdays and overestimating it on weekends. Secondly, the day-ahead forecast underestimates load at extreme temperatures, likely due to insufficient representation of electrically driven heating and cooling technologies [Sem+24]. More broadly, ongoing structural changes during the study period including the expansion of behind-the-meter PV generation [Erd+22], growing heat pump adoption [Sem+24], and demand-side shifts driven by rising energy prices continuously alter load profiles and introduce concept drift that static forecast models may not adequately capture. Thirdly, high solar irradiance leads to the overprediction of load due to the unmonitored expansion of behind-the-meter solar PV generation. Fourthly, the forecast overestimates demand at the start and end of a year and on public holidays and bridge days, reflecting a limited representation of the reduction in demand associated with reduced industrial activity on these days.

The SHAP analysis on the high-accuracy data subset further confirms the consistency of these patterns. Here, the feature week of year shows elevated importance, suggesting that especially the holiday related features are substantially underestimated by the day-ahead forecast.

Together, these findings demonstrate that the ENTSO-E day-ahead forecast contains temporally stable and physically motivated systematic errors that are partially predictable from observable features. The trained XGBoost residual model achieved a weekly MASE of 0.40 and reduced the MAE of the raw forecast by 41%, from 238.22, MW to 139.34 MW. This quantitative improvement, combined with the physically interpretable error sources

6. Conclusion

identified through the SHAP analysis, suggests that a post-processing correction targeting the four identified error patterns which are weekday/weekend demand differentiation, holiday and year-end effects, extreme temperature response, and behind-the-meter PV influence, could be a promising direction for improving day-ahead forecast accuracy in the TransnetBW control area.

7. Limitations and Outlook

7.1. Limitations

7.1.1. Concept Drift and Temporal Generalization

A fundamental challenge for any model trained on historical time series data is concept drift, particularly with regard to load data, given that many parameters are continuously changing as also analyzed in Chapter 2.1.2. The ongoing electrification of heating through heat pump adoption [Sem+24], the continued expansion of behind-the-meter PV generation [Erd+22], and demand-side behavioral shifts partly driven by rising energy prices - all of which alter the load profile in ways that historical lag, calendar, and weather features cannot fully anticipate. For this thesis, this means that error patterns learned from data spanning 2021-2025 may not remain stable over time. Features that were informative predictors of the residual in earlier years may lose predictive power, or new systematic patterns may emerge that the model was not trained to capture. As a consequence, the model cannot be directly applied as a correction mechanism for future day-ahead forecasts without revalidation, since a correction term derived from historically learned error patterns may itself become a source of error if the underlying load dynamics have shifted since the training period.

Also, the model's significance is limited to the study period of 2021–2025, and it does not evaluate how well it would perform with future, unseen data. This is partly due to the group shuffle split, whereby calendar weeks are randomly assigned to the training, validation and test sets, regardless of their chronological position in the dataset. Although this approach is well-suited to analyzing the systematic error over the whole time period in response to our research question, it has only limited significance for future data points, as it neither captures nor adapts to concept drift.

7.1.2. Feature Set

The feature set used in this thesis only captures the categories of the ENTSO-E's Transparency Report [ENT23] in a limited way. Rather than using load profiles, lagged load and error features were employed, which do not distinguish between residential and industrial load patterns. Furthermore, we used high-quality retrospective weather data from the ERA5 dataset [Her+17], instead of historical weather forecasts, which would have been closer to the data available at the time of the day-ahead forecast. The actual energy mix

composition is expected to be partly inferred from the weather data and is therefore also only an estimation.

7.2. Outlook

Building on the findings of this thesis, several directions for future research can be identified that could deepen the understanding of systematic error patterns of the day-ahead forecast and improve the practical applicability of the residual prediction model.

7.2.1. Concept Drift Analysis

One key limitation of the present work is its limited ability to capture temporal structural changes across the period from 2021 to 2025, such as the gradual electrification of heating through growing heat pump adoption [Sem+24], the expansion of behind-the-meter solar PV [Erd+22], or demand-side behavioral changes driven by energy price volatility. A model trained on a static historical window may not adapt to these evolving patterns. Potential remedies include adding trend-sensitive proxy features for these structural shifts, or adopting a rolling-window training approach that continuously updates the model as new data becomes available.

Additional features that could be added include energy generation data, especially the share of wind, solar and hydroelectric power plants, across periods.

Training the model with a rolling window approach would also allow us to track how feature importance and error metrics evolve over time.

7.2.2. Comparing Online Load and ENTSO-E Load

In this thesis we used the load data provided by Transnet BW. Analyzing the discrepancy between the two datasets could be interesting for evaluating systematic error. Such an analysis would reveal temporal errors affecting published load statistics, which could be relevant for other researchers relying on ENTSO-E data.

7.2.3. Feature Set improvements

The feature set in this thesis was grounded in the categories mentioned by ENTSO-E Transparency Report [ENT23]. Currently, the model shows potential for improvement in some of the report's categories, including load profiles, socioeconomic factors and weather features.

Including actual load profile data in model training, rather than lagged load features, could further improve the model's accuracy. Currently, the model relies on the holiday

features (is holiday, is school holiday and is bridge day) to capture socioeconomic factors. Differentiated load profile data for industrial and private load could be included, as well as other factors such as population data or an estimation of behind-the-meter PV or heat pumps. Additional improvements could be made if, instead of the high-quality retrospective weather data from ERA5 [Her+17], historical weather forecasts at the time of the day ahead forecast are used.

Overall, the model configuration in terms of feature set combinations could be analyzed further. Currently, we have only compared the configurations and systematic combinations based on calendar, weather and lag feature segregations. A systematic approach for feature selection could yield interesting results and further improvements. This could be achieved using a Dijkstra-inspired approach, incrementally adding features to the model that yield the greatest marginal improvement.

7.2.4. Alternative Data Splitting Strategies

The group shuffle split used in this thesis is appropriate for characterizing systematic error patterns across the full historical period, as stated in the research objective. However, to evaluate the practical utility of the model for correcting the future day-ahead forecast, a temporally ordered cross-validation scheme would be more appropriate, as this is consistent with state-of-the-art practices in load forecasting [CTM20] [BB12] [BHK18]. Comparing the results obtained by both splitting strategies would further consolidate the systematic error patterns evaluated in this thesis.

7.2.5. Forecast Correction

The most practically impactful direction for future work is to evaluate whether the residual prediction and evaluated error patterns could be used to improve the ENTSO-E day-ahead forecast directly. This could be achieved by adding the predicted residual directly to the day-ahead forecast as a post-processing correction step and evaluating the corrected forecast against the original. Alternatively, methods could be developed to incorporate additional features or corrections during the day-ahead forecast training process to account for the most significant error patterns identified in this thesis.

Bibliography

- [BB12] Christoph Bergmeir and José M. Benítez. “On the use of cross-validation for time series predictor evaluation”. In: *Information Sciences*. Data Mining for Software Trustworthiness 191 (May 2012), pp. 192–213. ISSN: 0020-0255. DOI: 10.1016/j.ins.2011.12.028. URL: <https://www.sciencedirect.com/science/article/pii/S0020025511006773> (visited on 12/08/2025).
- [BHK18] Christoph Bergmeir, Rob J. Hyndman, and Bonsoo Koo. “A note on the validity of cross-validation for evaluating autoregressive time series prediction”. In: *Computational Statistics & Data Analysis* 120 (Apr. 2018), pp. 70–83. ISSN: 0167-9473. DOI: 10.1016/j.csda.2017.11.003. URL: <https://www.sciencedirect.com/science/article/pii/S0167947317302384> (visited on 02/17/2026).
- [Bor+24] Vadim Borisov et al. “Deep Neural Networks and Tabular Data: A Survey”. In: *IEEE Transactions on Neural Networks and Learning Systems* 35.6 (June 2024), pp. 7499–7519. ISSN: 2162-2388. DOI: 10.1109/TNNLS.2022.3229161. URL: <https://ieeexplore.ieee.org/document/9998482> (visited on 02/19/2026).
- [CF14] Shankar N. Chandramowli and Frank A. Felder. “Impact of climate change on electricity systems and markets – A review of models and forecasts”. In: *Sustainable Energy Technologies and Assessments* 5 (Mar. 2014), pp. 62–74. ISSN: 2213-1388. DOI: 10.1016/j.seta.2013.11.003. URL: <https://www.sciencedirect.com/science/article/pii/S2213138813000805> (visited on 02/27/2026).
- [CG16] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’16. New York, NY, USA: Association for Computing Machinery, Aug. 2016, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. URL: <https://dl.acm.org/doi/10.1145/2939672.2939785> (visited on 02/25/2026).
- [CTM20] Vitor Cerqueira, Luis Torgo, and Igor Mozetič. “Evaluating time series forecasting models: an empirical study on performance estimation methods”. In: *Machine Learning* 109.11 (Nov. 2020), pp. 1997–2028. ISSN: 1573-0565. DOI: 10.1007/s10994-020-05910-7. URL: <https://doi.org/10.1007/s10994-020-05910-7> (visited on 12/08/2025).
- [DRM+25] DRMacIver et al. *holidays: Python Library for Holiday Dates*. <https://pypi.org/project/holidays/>. Version 0.24. Python package; accessed on 12-Feb-2026. 2025.

- [ENT23] ENTSO-E. *Detailed Data Descriptions, Version 3, Release 4*. Manual of Procedures Reference (MoP Ref2 DDD v3r4). 2023. URL: https://eepublicdownloads.entsoe.eu/clean-documents/Transparency/MoP_Ref2_DDD_v3r4.pdf#page=19.
- [ENT26] ENTSO-E. *About ENTSO-E: Mission, Vision and Values*. Accessed on 12-Feb-2026. 2026. URL: <https://www.entsoe.eu/about/>.
- [Erd+22] Burcin Cakir Erdener et al. “A review of behind-the-meter solar forecasting”. In: *Renewable and Sustainable Energy Reviews* 160 (May 2022), p. 112224. ISSN: 1364-0321. DOI: 10.1016/j.rser.2022.112224. URL: <https://www.sciencedirect.com/science/article/pii/S1364032122001472> (visited on 02/11/2026).
- [Eur13] European Union. *Commission Regulation (EU) No 543/2013 of 14 June 2013 on submission and publication of data in electricity markets*. <https://eur-lex.europa.eu/eli/reg/2013/543/oj/eng>. Official Journal of the European Union (Accessed on 12-Feb-2026). 2013.
- [FKM06] Katsushige Fujimoto, Ivan Kojadinovic, and Jean-Luc Marichal. “Axiomatic characterizations of probabilistic and cardinal-probabilistic interaction indices”. In: *Games and Economic Behavior* 55.1 (Apr. 2006), pp. 72–99. ISSN: 0899-8256. DOI: 10.1016/j.geb.2005.03.002. URL: <https://www.sciencedirect.com/science/article/pii/S0899825605000278> (visited on 02/27/2026).
- [Fra25] Fraunhofer Institute for Solar Energy Systems ISE. *Public Electricity Generation 2024: Renewable Energies Cover More Than 60 Percent of German Electricity Consumption for the First Time*. Accessed on 13-Feb-2026; press release reporting German electricity generation statistics for 2024. Jan. 7, 2025. URL: <https://www.ise.fraunhofer.de/en/press-media/press-releases/2025/public-electricity-generation-2024-renewable-energies-cover-more-than-60-percent-of-german-electricity-consumption-for-the-first-time.html>.
- [Fri01] Jerome H. Friedman. “Greedy function approximation: A gradient boosting machine.” en. In: *The Annals of Statistics* 29.5 (Oct. 2001), pp. 1189–1232. ISSN: 0090-5364, 2168-8966. DOI: 10.1214/aos/1013203451. URL: <https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.full> (visited on 02/25/2026).
- [GPB19] Shadi Goodarzi, H. Niles Perera, and Derek Bunn. “The impact of renewable energy forecast errors on imbalance volumes and electricity spot prices”. In: *Energy Policy* 134 (Nov. 2019), p. 110827. ISSN: 0301-4215. DOI: 10.1016/j.enpol.2019.06.035. URL: <https://www.sciencedirect.com/science/article/pii/S0301421519304057> (visited on 02/27/2026).
- [HA21] Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice (3rd ed.)* Online textbook, accessed on 13-Feb-2026. 2021. URL: <https://otexts.com/fpp3/>.

-
- [Her+17] Hans Hersbach et al. *Complete ERA5 from 1940: Fifth generation of ECMWF atmospheric reanalyses of the global climate*. Accessed on 28-10-2025. 2017. DOI: 10.24381/cds.143582cf. URL: <https://cds.climate.copernicus.eu/datasets/reanalysis-era5-complete>.
- [HK06] Rob J. Hyndman and Anne B. Koehler. “Another look at measures of forecast accuracy”. In: *International Journal of Forecasting* 22.4 (Oct. 2006), pp. 679–688. ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2006.03.001. URL: <https://www.sciencedirect.com/science/article/pii/S0169207006000239> (visited on 01/25/2026).
- [KT22] Hussain Kazmi and Zhenmin Tao. “How good are TSO load and renewable generation forecasts: Learning curves, challenges, and the road ahead”. In: *Applied Energy* 323 (Oct. 2022), p. 119565. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2022.119565. URL: <https://www.sciencedirect.com/science/article/pii/S0306261922008753> (visited on 12/10/2025).
- [LL17] Scott M Lundberg and Su-In Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html (visited on 02/24/2026).
- [Lun+20] Scott M. Lundberg et al. “From local explanations to global understanding with explainable AI for trees”. en. In: *Nature Machine Intelligence* 2.1 (Jan. 2020), pp. 56–67. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0138-9. URL: <https://www.nature.com/articles/s42256-019-0138-9> (visited on 02/24/2026).
- [Möb+25] Thomas Möbius et al. “Enhancing energy system models using better load forecasts”. en. In: *Energy Systems* 16.2 (May 2025), pp. 573–602. ISSN: 1868-3975. DOI: 10.1007/s12667-023-00590-3. URL: <https://doi.org/10.1007/s12667-023-00590-3> (visited on 12/10/2025).
- [Mol23] Christoph Molnar. *Interpreting Machine Learning Models with SHAP*. Bookversion 2023-08-01. Aug. 1, 2023. URL: <https://christophmolnar.com/books/shap>.
- [Pyt25] Python Software Foundation. *datetime — Basic date and time types*. <https://docs.python.org/3/library/datetime.html>. Python 3 Standard Library Documentation (Accessed on 12-Feb-2026). 2025.
- [sch26] schulferien.org. *Schulferien Baden-Württemberg – Ferien in Baden-Württemberg, Deutschland*. Accessed on 12-Feb-2026; overview of school holiday dates and durations for Baden-Württemberg, Germany. 2026. URL: <https://www.schulferien.org/deutschland/ferien/baden-wuerttemberg/>.

- [sci25] scikit-learn developers. *sklearn.model_selection.GroupShuffleSplit* – scikit – learndocumentation. Accessed on 12-Feb-2026; API reference for GroupShuffleSplit cross-validation iterator in scikit-learn. 2025. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GroupShuffleSplit.html.
- [Sem+24] Leo Semmelmann et al. “The impact of heat pumps on day-ahead energy community load forecasting”. In: *Applied Energy* 368 (Aug. 2024), p. 123364. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2024.123364. URL: <https://www.sciencedirect.com/science/article/pii/S0306261924007475> (visited on 02/13/2026).
- [SHA26] SHAP Developers. *NHANES I Survival Model — SHAP example notebook*. Accessed on 13-Feb-2026; example notebook demonstrating SHAP on NHANES I survival model. 2026. URL: https://shap.readthedocs.io/en/latest/example_notebooks/tabular_examples/tree_based_models/NHANES%20I%20Survival%20Model.html.
- [Sha52] Lloyd S. Shapley. *A Value for N-Person Games*. en. Tech. rep. Mar. 1952. URL: <https://www.rand.org/pubs/papers/P295.html> (visited on 02/24/2026).
- [ŠK14] Erik Štrumbelj and Igor Kononenko. “Explaining prediction models and individual predictions with feature contributions”. en. In: *Knowledge and Information Systems* 41.3 (Dec. 2014), pp. 647–665. ISSN: 0219-3116. DOI: 10.1007/s10115-013-0679-x. URL: <https://doi.org/10.1007/s10115-013-0679-x> (visited on 02/24/2026).
- [Tra26] TransnetBW GmbH. *Who we are – Portrait of TransnetBW*. Accessed on 12-Feb-2026. 2026. URL: <https://www.transnetbw.de/en/company/portrait/who-we-are>.
- [Xie+25] Kexin Xie et al. “Bias calibration and error propagation adjustment for ML-based time series forecasting: A systematic study for PJM’s electricity load forecast amid Virginia’s data center surge”. In: *Energy* 336 (Nov. 2025), p. 138411. ISSN: 0360-5442. DOI: 10.1016/j.energy.2025.138411. URL: <https://www.sciencedirect.com/science/article/pii/S0360544225040538> (visited on 12/08/2025).

A. Appendix

RMSE Root Mean Squared Error

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

sMAPE symmetric Mean Absolute Percentage Error

MASE Mean Absolute Scaled Error

SHAP SHapley Additive exPlanations

XGBoost eXtreme Gradient Boosting

ENTSO-E European Network of Transmission System Operators for Electricity

TSO Transmission Service Operator

XAI Explainable Artificial Intelligence

LIME Local Interpretable Model-Agnostic Explanations

A.1. MAPE and sMAPE for all models

Table A.1.: Performance comparison of models trained on different feature subsets. CF = calendar features, WF = weather features, LF = lag features, CWF = calendar + weather features, LCF = lag + calendar features, WLF = weather + lag features, ALL = all features combined.

Metric	CF	WF	LF	CWF	WLF	LCF	ALL
MAPE	429.89	482.44	454.57	410.11	515.25	515.60	399.87
sMAPE	1.23	1.23	1.16	1.05	1.04	1.04	0.95

A.1.1. MAPE

The exceptionally high MAPE value of 399.87% requires careful interpretation. As Hyndman and Koehler (2006) [HK06] discussed, the greatest limitation of the MAPE is, that it "has the disadvantage of being infinite or undefined if $Y_t = 0$ for any t in the period of interest, and having extreme values if any Y_t is close to zero". In the context of day-ahead residuals this happens when the forecast is highly accurate, so the actual residual approaches zero. Then the denominator of the MAPE becomes below zero resulting in disproportionately large percentage errors for small absolute predictions. As Hyndman and Koehler (2006) further note, MAPE "puts a heavier penalty on negative errors than on positive errors" [HK06], creating asymmetric penalties that can bias model optimization additionally and lead to a disbalanced metric overall.

Given these limitations, MAPE is not a suitable metric for comparing model performance in this application and will not be used for further analysis.

A.1.2. sMAPE

While the sMAPE value of 0.95 appears more reasonable than MAPE, this metric also suffers from fundamental shortcomings. When both actual and forecast values are close to zero, the denominator remains small, resulting in instability when the denominator gets close to zero. Moreover they note that "the value of sMAPE can be negative, so it is not really a measure of 'absolute percentage errors' at all" and explicitly recommend "the sMAPE not be used" [HK06].

Following Hyndman and Koehler's recommendation, sMAPE is not used as a primary evaluation metric in this analysis.

A.2. All Dependence Plots

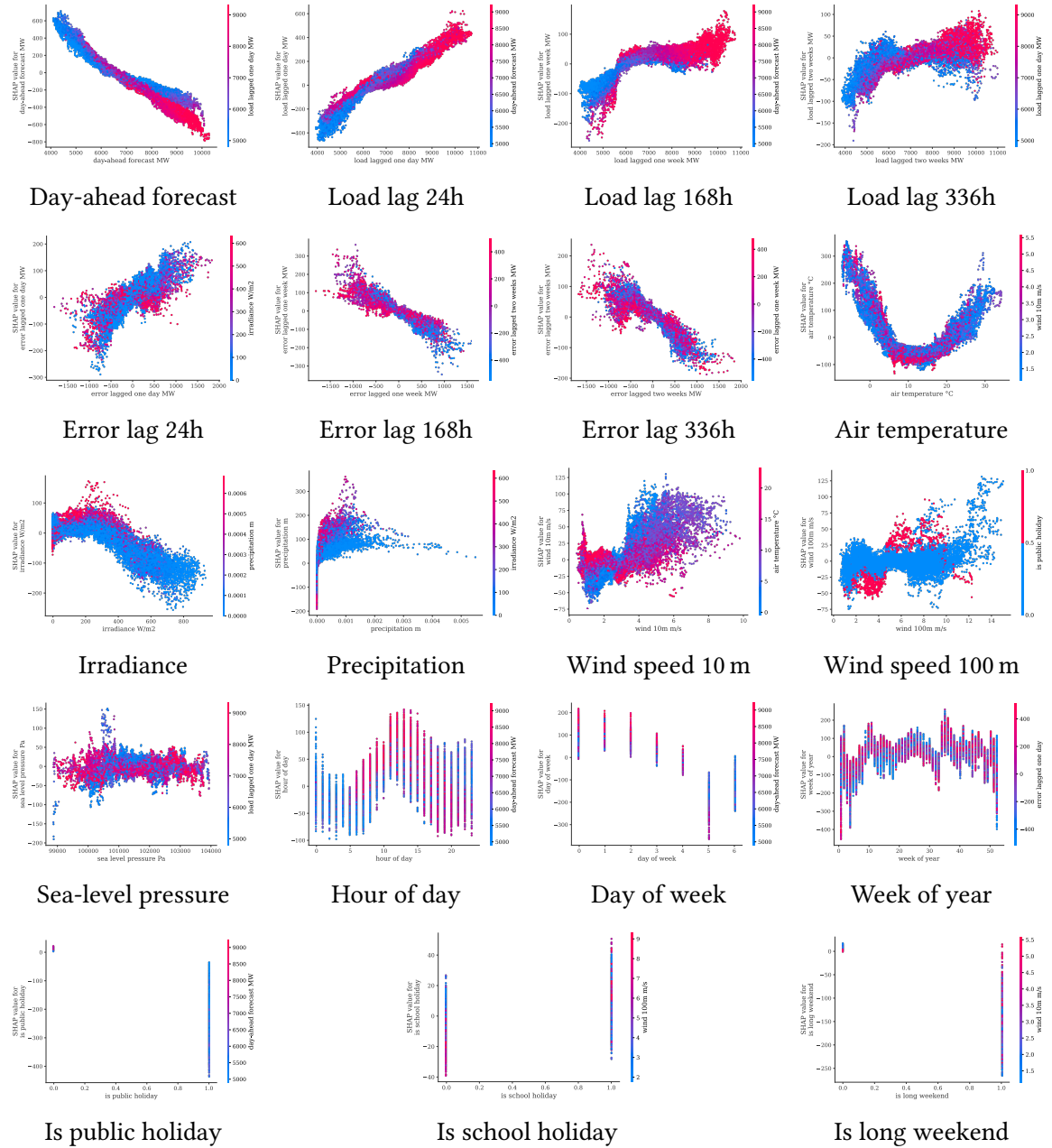


Figure A.1.: SHAP dependence plots for all features of the best model, grouped by category: lagged load (row 1), lagged error (row 2), weather (rows 3–4), and calendar (row 5). Each plot shows the SHAP value of the feature on the y -axis against the feature value on the x -axis, with point colour indicating the value of the feature with the strongest SHAP interaction.