

Feature selection for system imbalance forecasting

Jannes De Sutter

Thesis submitted for the degree of
Master of Science in Mechanical
Engineering

Supervisors:

Prof. dr. ir. Erik Delarue
Prof. dr. ir. Hussain Syed Kazmi

Assessors:

Prof. dr. ir. Konstantinos Gryllias
Dr. ir. Ruben Smets

Assistant-supervisor:

Ir. Yucun Lu

© Copyright KU Leuven

Without written permission of the supervisors and the author it is forbidden to reproduce or adapt in any form or by any means any part of this publication. Requests for obtaining the right to reproduce or utilize parts of this publication should be addressed to Faculteit Ingenieurswetenschappen, Kasteelpark Arenberg 1 bus 2200, B-3001 Leuven, +32-16-321350.

A written permission of the supervisors is also required to use the methods, products, schematics and programmes described in this work for industrial or commercial use, and for submitting this publication in scientific contests.

Contents

Abstract	iii
List of Figures and Tables	iv
List of Abbreviations and Symbols	vii
List of abbreviations	vii
1 Introduction	1
2 Background	3
2.1 Regulatory framework overview	3
2.2 Balancing mechanisms in the Belgian market	4
2.3 System imbalance and imbalance price	6
3 Literature review	9
3.1 Introduction to system imbalance forecasting	9
3.2 Machine learning models	10
3.3 Feature selection methods	14
4 Methodology	21
4.1 Forecasting target	21
4.2 Data collection	21
4.3 Exploratory data analysis	22
4.4 Feature selection methods	25
4.5 Modelling	31
4.6 Validation Strategy	34
4.7 Evaluation Metrics	36
5 Exploratory data analysis	37
5.1 One-dimensional analysis of system imbalance	37
5.2 Cross-correlation	41
6 Results and discussion	45
6.1 Forecasting target and model setup	45
6.2 Computational Considerations	46
6.3 Implementation of FS methods	46
6.4 Results	47
6.5 Similarity between feature selection methods	52
6.6 Feature interpretability	54
6.7 Expansion	57

CONTENTS

6.8	Final Feature Set	58
7	Conclusion	59
A	Selected features	63
A.1	Full Feature list	63
A.2	List of selected input features	65
A.3	Feature-Lag Selection Heatmap	66
A.4	Final feature set	68
B	FS methods in depth	71
B.1	Random forest	71
B.2	Principle component analysis	72
C	Model architecture	73
C.1	Theoretical background of feedforward neural networks	73
D	Supplementary analyses	77
D.1	Probabilistic Forecast Metrics	77
D.2	STL: Seasonal and Trend decomposition using Loess	77
D.3	Extended exploratory results	79
D.4	Feature selection overlap matrix	80
D.5	Cross-validation performance details	81
D.6	Cross-border feature importance analysis	81
D.7	PCA	82
D.8	Imbalance price	82
	Bibliography	83

Abstract

In the context of liberalized electricity markets and an increasing share of intermittent renewable energy sources, accurate forecasting of system imbalance has become critical for maintaining grid stability and supporting operational decisions. This thesis presents a structured evaluation of feature selection methods, addressing a gap in the literature on short-term imbalance forecasting in Belgium.

The methodology relies on constructing lagged versions of input variables across multiple time resolutions, followed by a comparative analysis of filter, wrapper, and embedded feature selection techniques. Two forecasting models are considered: a multivariate linear regression (LR) and a shallow multilayer perceptron (MLP), both evaluated using a time-series-aware cross-validation approach. The results indicate that appropriate feature selection can improve forecasting performance by up to 6% relative to the benchmark model currently used by the Belgian transmission system operator, when applied to the same predictive model. In addition, feature selection contributes to a significant reduction in model complexity by identifying compact and informative input subsets, while also highlighting the computational costs associated with the selection process.

Beyond quantitative improvements, this work provides insight into the temporal and contextual relevance of individual features. Several variables are consistently selected across methods and models, showing high importance when included. Wind and solar generation features stand out, reinforcing the link between renewable production and system imbalance. Furthermore, recurring patterns in cross-border nominations, particularly with France and the Netherlands, suggest their notable impact on imbalance dynamics.

Two final feature sets are proposed for operational use, one for each model, and the full implementation is made publicly available, offering a foundation for future adaptation and extension as the electricity landscape evolves and additional data sources become available.

List of Figures and Tables

List of Figures

2.1	Timeline of European electricity markets, from [99].	4
2.2	Scheduled Activation profile, from [33].	6
2.3	Single imbalance price settlement scheme, from [9].	8
2.4	BRPs respond to the imbalance price, from [26].	8
4.1	SFFS flow chart, from [15].	28
4.2	Feedforward neural network consisting of six input units in the input layer, one hidden layer with 3 neurons and one output unit in the output layer.	33
4.3	Rolling-window cross-validation: each line represents a model with training (blue) and test (red) periods.	35
5.1	Statistical evaluation of the system imbalance signal.	39
5.2	Evolution of 10-month centered moving average of SI volumes.	39
5.3	Distributional profiles of the SI. The box represents the interquartile range (IQR). The black line inside the box indicates the median value. .	40
5.4	Autocorrelation and partial autocorrelation of the system imbalance signal. .	41
5.5	Comparison of Partial Autocorrelation and Mutual Information for lagged SI features. PACF values (full bars) are shown on the left axis, MI values (hatched bars) on the right.	42
5.6	Cross-correlations of 300 to 1 lagged input features with system imbalance (SI). Each subplot shows the correlation structure for a specific feature category.	43
5.7	Correlation matrix of Germany in absolute values.	44
6.1	MAE score for sequential selection methods in function of number of features.	49
6.2	Marginal gain and computation time per step as a function of the number of selected features. The top row shows Sequential Backward Selection (SBS) and Sequential Backward Floating Selection, both for LR. The bottom row shows SBS for MLP.	50

6.3	Jaccard similarity matrix comparing feature subsets selected by different FS methods. Higher values indicate greater agreement between selected subsets, normalized for subset size.	53
6.4	Weighted selection frequency of the top 20 feature-lag pairs across all models and feature selection methods.	55
6.5	Permutation importance of cross-border features (left) and observed electricity transmission volumes over time [TWh] (right) [42].	57
A.1	Comprehensive heatmap showing the weighted frequency of feature-lag selections across all models and feature selection methods. The intensity and label values reflect how often each combination was chosen, weighted by model performance. (top half)	66
A.2	Heatmap (bottom half) of weighted feature-lag selection frequencies.	67
D.1	Additive seasonal decomposition of the weekly average system imbalance for 2022.	78
D.2	Additive seasonal decomposition of the monthly average system imbalance from January 2020 to December 2023.	78
D.3	Absolute Pearson correlation of candidate features with system imbalance (SI).	79
D.4	Maximum cross-correlation of exogenous variables with SI across multiple lags	79
D.5	Overlap matrix showing the number of shared features between different selection methods or models. Higher values indicate greater consistency in the selected feature subsets.	80
D.6	Cross-validation performance across time-aware folds.	81
D.7	Feature importance of cross-border nominations using SHAP and permutation metrics.	81
D.8	Effect of applying PCA to the least important features: impact on MAE and runtime across different component settings.	82
D.9	Jaccard similarity matrix for feature subsets selected by SBS and SBFS for imbalance price and system imbalance.	82

List of Tables

3.1	Strengths, weaknesses and examples of the three main feature selection categories. Adapted from [89]	18
4.1	Overview of key features and their availability timing at qh+0.	25
5.1	Key statistics on the Belgian system imbalance (MW).	38
6.1	Model training setup and hyperparameters	46

LIST OF FIGURES AND TABLES

6.2	Forecast performance with different feature selection methods (Linear Regression vs. MLP). Lower MAE/RMSE/SMAPE indicate better accuracy.	48
6.3	Computational cost of feature selection methods. Model fits include training across cross-validation folds. HPC: 64-core parallel execution.	51
A.1	Overview of minute-wise and hourly parameters and their availability timing at qh+0.	63
A.2	Overview of quarter-hour parameters and their availability timing at qh+0.	64
A.3	Overview of input the 355 features and their lag/lead structure after preliminary analysis	65
A.4	Final selected features with corresponding lags and leads for the Multi-Layer Perceptron (MLP) model MAE = 95.07	68
A.5	Final selected features with corresponding lags and leads for the Linear Regression (LR) model MAE = 91.72	69

List of Abbreviations and Symbols

Abbreviations

List of abbreviations

ACF	Auto Correlation Function
aFRR	automatic Frequency Restoration Reserve
BESS	Battery Energy Storage System
BRP	Balance Responsible Party
BSP	Balancing Service Provider
CNN	Convolutional Neural Network
CRPS	Continuous Rank Probability Score
CV	Cross-Validation
DM	Diebold-Mariano
FCR	Frequency Containment Reserve
FNN	Feedforward Neural Network
FRR	Frequency Restoration Reserve
FS	Feature Selection
GA	Genetic Algorithm
HP	HyperParameter
LA	Look-Ahead
LASSO	Least Absolute Shrinkage and Selection Operator
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MARI	Manually Activated Reserves Initiative
MASE	Mean Absolute Scaled Error
MIP	Marginal price for Upward Regulation
mFRR	manual Frequency Restoration Reserves

LIST OF ABBREVIATIONS AND SYMBOLS

ML	Machine Learning
MLP	Multi-Layer Perceptron
NN	Neural Network
NRV	Net Regulation Volume
PCA	Principal Component Analysis
PSO	Particle Swarm Optimization
RF	Random Forest
RFE	Recursive Feature Elimination
RMSE	Root Mean Squared Error
SB(F)S	Sequential Backward (Floating) Selection
SF(F)S	Sequential Forward (Floating) Selection
SI	System Imbalance
SMAPE	Symmetric Mean Absolute Percentage Error
SVM	Support Vector Machine
TSO	Transmission System Operator
VSC	Vlaams Supercomputer Centrum (Flemish Supercomputer Center)

Symbols

$\hat{\beta}$	Solution to the LASSO optimization problem
$cov(\cdot, \cdot)$	Covariance
$CRPS$	Continuous Ranked Probability Score
$FRCE_j$	Frequency Recovery Control Error for settlement period j
NRV_j	Net Regulation Volume for settlement period j
$OOB_b()$	Out-of-bag error for bootstrap sample b
SI_j	System Imbalance for settlement period j
$var(\cdot)$	Variance
$v(t)$	Current velocity at time t
\mathbf{x}_t	Input vector at time t
y_i	Observed system imbalance at forecast point i
\hat{y}_{t+h}	Predicted output at time $t + h$
σ	Activation function

Chapter 1

Introduction

The European electricity landscape has undergone a fundamental transformation. Formerly dominated by a vertically integrated monopoly structure, it has now shifted towards liberalized, competitive markets [72]. Simultaneously, the increasing integration of weather-dependent renewable energy sources and rising electricity demand due to electrification have introduced significant volatility on both the supply and demand sides of the power system [60].

Balancing mechanisms are used to compensate for mismatches between scheduled and actual injections and withdrawal, with the net sum of all individual deviations by Balancing Responsible Parties (BRPs) within a balancing area referred to as the *system imbalance* [17]. Under the single imbalance pricing scheme, market participants are no longer disadvantaged for system-supportive deviations when out of balance [5].

As electricity markets shift towards shorter timeframes due to rising uncertainty, the importance of near real-time forecasting grows [47, 103]. In this context, accurate short-term system imbalance forecasts are essential [53, 87], not only for Transmission System Operators (TSOs) to proactively manage reserves [32], but also for market participants to develop profitable strategies. Many authors have studied the strategic behaviour of Energy Storage System (ESS) operators in real-time balancing and reserve capacity markets, demonstrating how forecast-driven strategies are used to maximize profits by optimizing charging and discharging in response to real-time imbalance signals [20, 101, 105, 120].

Machine Learning (ML) has shown great promise across a range of forecasting domains, including economic, weather and energy forecasting [55]. Electricity market data is particularly well suited for data-driven approaches, as it is high-frequency, time-aligned and rich in system indicators, ranging from generation and load values to weather variables and market signals. Both linear and nonlinear models have been applied to system imbalance [6, 18, 94]. While neural networks are valued for their ability to model complex, nonlinear relationships in time-series data, linear models

1. INTRODUCTION

remain relevant due to their simplicity and competitive performance [28].

However, the performance of neural networks is highly sensitive to the input feature set. In many studies, models are presented in their final form with little explanation of how features were selected. This ad-hoc or trial-and-error process limits reproducibility and may lead to suboptimal models [70]. The problem is amplified in high-dimensional settings, where an abundance of features often introduces redundancy or irrelevant information, ultimately degrading model performance and increasing computational demands [107]. This is particularly relevant in electricity forecasting, where a wide array of potentially useful features exists [109].

In such high-dimensional settings, feature selection becomes a critical step. It addresses these challenges by identifying a subset of informative features that enhance model accuracy and efficiency. Feature selection plays a key role in improving interpretability, reducing complexity and mitigating overfitting [71, 109].

This thesis is, to the best of the author's knowledge, the first academic study to directly address *feature selection for system imbalance forecasting*, filling a clear gap in the literature where the topic has typically been treated in a limited or indirect manner [6, 28, 67, 87, 107]. Moreover, system imbalance forecasting is often incorporated as a secondary component within larger research objectives, such as energy storage system optimization [101, 105, 120].

It extends the existing literature on forecasting in the electricity domain, which is primarily focused on model architectures and forecasting performance, often overlooking the critical role of input feature selection. This thesis systematically evaluates and benchmarks a range of feature selection methods, including filter, wrapper and embedded approaches, within both linear regression and neural network-based forecasting models, trained on real data from the Belgian electricity market. In doing so, it identifies which methods yield the most compact and informative feature sets, while balancing forecasting performance and computational cost. All crucial considerations in settings were forecasts must be delivered rapidly and reliably. This work also includes a modest exploration of feature interpretability.

The remainder of the thesis is structured as follows. Chapter 2 provides the necessary background by outlining the regulatory framework and describing the balancing mechanisms of the Belgian electricity market. Chapter 3 reviews the relevant academic literature. It focuses on forecasting methods in the electricity sector, feature selection techniques and situates the work within the broader academic context. Chapter 4 outlines the methodological approach. Chapter 5 introduces the dataset, key characteristics of selected features and provides details on feature selection choices. Chapter 6 presents the results and discussion, comparing the performance of linear and neural network models under different feature selection methods. Finally, Chapter 7 concludes the thesis by summarizing the main findings and suggested directions for future research.

Chapter 2

Background

To properly frame the context of this study, a brief overview of the regulatory structure of the electricity market is provided in Section 2.1. This sets the stage for understanding the mechanisms behind balancing and reserve provision, discussed in Section 2.2 and highlights the importance of this issue within the broader context of the ongoing energy transition. Section 2.3 further defines system imbalance, including the principles of imbalance settlement and pricing.

2.1 Regulatory framework overview

Energy policy in Belgium is a shared responsibility between the federal government and the regional governments: Flanders, Wallonia and the Brussels-Capital Region, based on EU liberalization directives¹ [41]. The electricity market is now fully liberalized, separating competitive sectors (generation, retail) from regulated natural monopolies (transmission, distribution). Elia, the Belgian Transmission System Operator (TSO), holds the legal monopoly over the high-voltage transmission grid (70 kV - 380 kV) in Belgium [36].

As part of the Central Western Europe (CWE) market coupling zone, Belgium participates in the Single Day-Ahead and Intraday Coupling mechanisms (SDAC and SIDC), enabling cross-border electricity trading and price convergence. The day-ahead and intraday markets are operated by multiple Nominated Electricity Market Operators (NEMOs), such as EPEX SPOT and Nord Pool, under the Belgian Multiple NEMO Arrangement (MNA). The official day-ahead price is determined through the Market Coupling Operation (MCO), while Elia also publishes a volume-weighted Belgian Reference Price for informational and fallback purposes [1].

¹Electricity Act, which implemented EU Directives 96/92/EC, 2003/54/EC and 2009/72/EC

2. BACKGROUND

2.2 Balancing mechanisms in the Belgian market

System imbalance occurs when Balance Responsible Parties (BRPs) fail to keep their injection and offtake positions balanced. Elia continuously monitors and compensates for these deviations by activating balancing energy in real-time, using reserves procured from Balancing Service Providers (BSPs) in accordance with market rules [34]. Some reserves are already integrated with neighbouring TSO systems; others are in transition. Further details on how system imbalance is calculated as well as on how the resulting imbalance volumes and prices are used for settlement and incentivization, are provided in Section 2.3.2.

System imbalance is shaped by price signals, platform interactions and regulatory design. It results from deviations between forecasted and actual consumption or generation in a coupled electricity market, where BRPs are responsible for maintaining balance. Market coupling, renewables and differing NEMO dynamics increase forecasting complexity and shape the financial incentives related to system balancing.

To improve efficiency, many European countries separated balancing capacity from balancing energy markets. Capacity costs are covered via grid fees (shared by all users), while the balancing energy costs are paid by those responsible for imbalances [38].

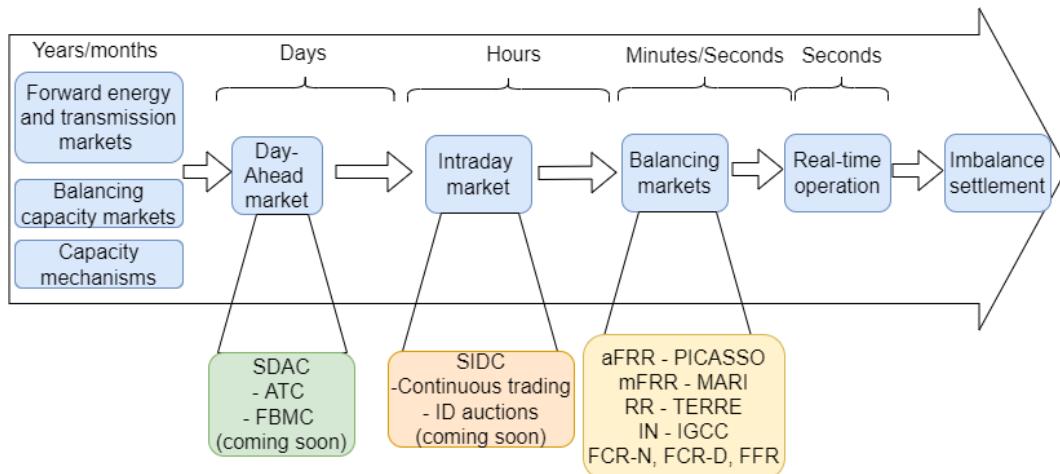


FIGURE 2.1: Timeline of European electricity markets, from [99].

To maintain system balance, Elia procures three main types of reserves:

Frequency Containment Reserve (FCR)

FCR is activated automatically within seconds to stabilize the frequency at a steady-state value after a disturbance across the European synchronous area [37]. It offsets imbalances that cause frequency variations and is the first line of defense in preventing

system instability or blackouts. Grid users connected to Elia or a distribution system may directly or indirectly participate as BSPs [33, 92].

automatic Frequency Restoration Reserve (aFRR)

The aFRR is the second fastest control reserve market and is used to restore the system frequency back to its nominal value (50 Hz) after FCR activation. Elia activates aFRR automatically by sending a control signal every four seconds, with delivery expected within the Full Activation Time (FAT). The FAT was 7.5 minutes until 3 December 2024 and is now reduced to 5 minutes [33]. With the progressive increase in aFRR provision, FCR resources are freed up and may be redirected to manage subsequent imbalances [85].

manual Frequency Restoration Reserve (mFRR)

The mFRR is manually activated by Elia, typically in response to larger or prolonged system imbalances, in order to release activated aFRR capacity or to provide additional frequency restoration power. The full activation time (FAT) for mFRR is 12.5 minutes, meaning that BSPs must fully deliver the requested balancing energy within 12.5 minutes after the activation signal is sent.

Balancing energy bids for both aFRR and mFRR are ranked using a Local Merit Order List (LMOL) based on activation price. These are shared with European platforms, such as MARI, to support coordinated cross-border activation, forming a Common Merit Order List (CMOL) [33]. For mFRR, Elia may issue an activation request as late as 7.5 minutes before the start of the relevant quarter-hour [29]. Bids submitted by BSPs can come from reserved or non-reserved capacity, allowing access to a broader pool of balancing resources with a market granularity of 15 minutes.

In practice, SI forecasts must be available no later than minute 7 of the preceding quarter hour, and preferably by minute 3, to enable timely activation decisions, bid evaluation and coordination with the MARI platform [33]. For instance, to prepare for the 16:15-16:30 quarter-hour, a SI estimate is required by 16:08 at the latest. The operational timeline for this process is illustrated in Figure 2.2.

2.2.1 Cross-border integration

Balancing markets in Europe are undergoing increasing integration through cross-border balancing platforms, which enable TSOs to access the most cost-efficient balancing energy available across national borders.

Regarding aFRR, Elia has participated in the International Grid Control Cooperation (IGCC) since 2012 and joined the PICASSO platform, the official European aFRR platform, on 26 November 2024. Early results indicate that the PICASSO

2. BACKGROUND

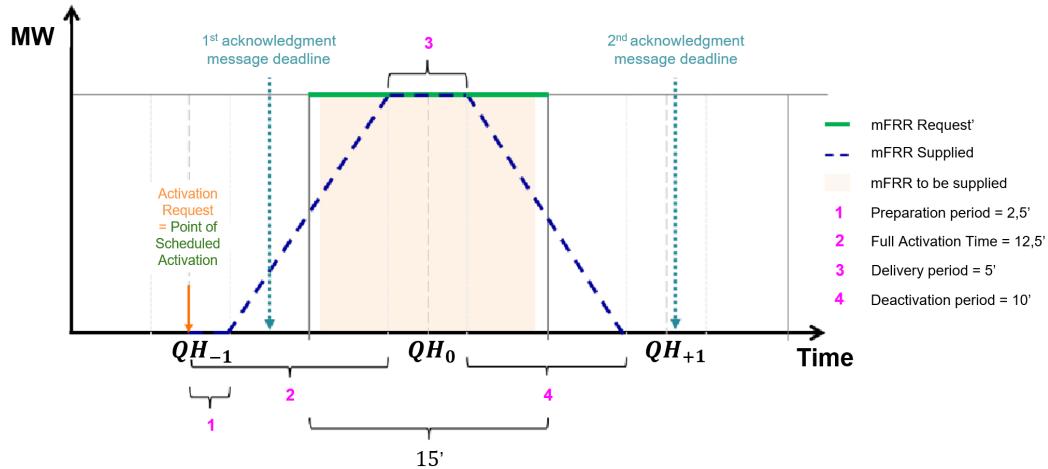


FIGURE 2.2: Scheduled Activation profile, from [33].

platform has increased the share of Elia's aFRR activations via cross-border exchanges, often contributing to lower imbalance prices. However, given the recency of Elia's accession, long-term impacts remain to be evaluated [31, 33].

Regarding mFRR, Elia formally acceded to the Manually Activated Reserves Initiative (MARI) platform on *25 May 2025*, following a local go-live and interoperability testing in May 2024 [39]. MARI enables the cross-border exchange of manually activated frequency restoration reserves among European TSOs and allows Belgian BSPs to participate in the integrated European balancing market, improving access to flexible capacity and harmonizing practices across countries [17, 33].

As post-accession data is still limited, this thesis primarily focuses on the pre-MARI and pre-PICASSO periods. Nonetheless, an exploratory feature analysis is included to evaluate whether the relevance of input variables on the post-accession period, potentially indicating shifts in system behaviour.

2.3 System imbalance and imbalance price

2.3.1 System imbalance

System Imbalance (SI) is a fundamental indicator of the real-time equilibrium between electricity supply and demand, and hence of system stability. Conceptually, it is defined as the net mismatch between the total generated and consumed electrical power on the grid, measured in megawatts [MW]. In practice, SI arises due to deviations between forecasted and actual consumption or generation, particularly in the context of a liberalized and increasingly coupled European electricity market.

Commission Regulation (EU) 2017/2195, the Electricity Balancing Guideline, es-

2.3. System imbalance and imbalance price

tablishes the EU-wide framework for balancing market design, including the roles of TSOs and BRPs as well as the rules for activating and settling balancing energy [30, 32], following the activation of balancing reserves, as discussed in Section 2.2.

The system imbalance in Belgium is calculated as described in article 30.7 of the T&C BRP [34], that closely aligns with those used in the rest of Europe. For each Imbalance Settlement Period j , the system imbalance (SI_j) is defined as the difference between the Frequency Recovery Control Error ($FRCE_j$) and the Net Regulation Volume (NRV_j). This calculation applies to the period prior to Elia's accession to the PICASSO platform and before the technical go-live of the mFRR platform (MARI). Formally, this is expressed as:

$$SI_j = FRCE_j - NRV_j \quad [\text{MW}] \quad (2.1)$$

In the Belgian context, the FRCE is equivalent to the Area Control Error (ACE), which quantifies the unintended mismatch between expected and actual frequency support from balancing mechanisms. The Net Regulation Volume (NRV) is calculated as the difference between the Gross Upward Volume (GUV) and the Gross Downward Volume (GDV), i.e.,

$$NRV = GUV - GDV \quad [\text{MW}] \quad (2.2)$$

GUV and GDV represent, respectively, the total activated volumes of upward and downward balancing energy. They are composed of activated aFRR and mFRR energy bids, as well as cross-border exchanges through platforms such as IGCC. The exact formula definitions of calculation have evolved slightly due to platform changes, but the underlying principle remains unchanged.

2.3.2 Imbalance price

Imbalance prices are determined in accordance with regulatory guidelines and applied through a single-price settlement mechanism designed to penalize deviations. They can be viewed as the financial consequence of forecast errors, serving both as settlement rates and as economic incentives for BRPs to maintain or to restore the balance. Deviations are charged with imbalance tariffs that are proportional to both the volume and direction of the imbalance [32].

The pricing system rewards BRPs that contribute to restoring system balance (indicated in green in Figure 2.3) and to penalize those that aggravate the imbalance (indicated in red) [9]. In Figure 2.3, MIP and MDP refer to the Marginal Incremental and Decremental Prices used for the activation of balancing energy. This approach is in line with the EU Electricity Balancing Guideline and the Belgian Terms and Conditions for BRPs [32, 34]. This motivates the use of direct forecasting the imbalance price, as implemented in Section 6.7.2.

The current design of the Belgian balancing market allows BRPs to generate revenue

2. BACKGROUND

		Transmission System Operator (TSO) control area	
		Excess ($P>C$) Positive SI	Shortage ($P<C$) Negative SI
BRP area	Excess	MDP (low price) TSO pays BRP	MIP (high price) TSO pays BRP
	Shortage	MDP (low price) BRP pays TSO	MIP (high price) BRP pays TSO

FIGURE 2.3: Single imbalance price settlement scheme, from [9].

by strategically participating in the real-time market through implicit balancing. In this strategy, a BRP deliberately operates out of balance to benefit from real-time imbalance price fluctuations, without requiring day-ahead commitments in reserve capacity markets, as illustrated conceptually in Figure 2.4. Such behaviour is permitted under current Belgian regulation [35], provided the BRP is able to restore balance when requested by the system operator.

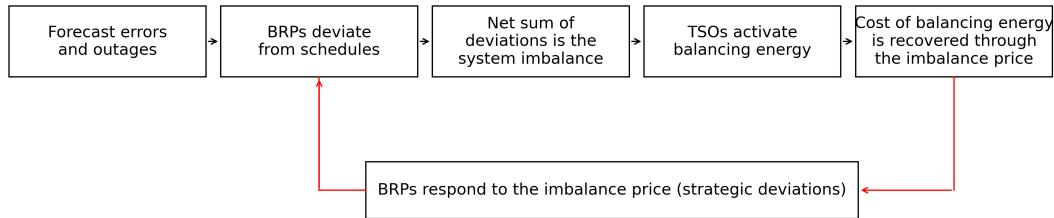


FIGURE 2.4: BRPs respond to the imbalance price, from [26].

Smets et al. explore this concept further in the context of Battery Energy Storage Systems (BESS), demonstrating how imbalance price volatility can be exploited by forecasting system imbalance using an attention-based recurrent neural network. Their approach enables the prediction of SI quantiles and facilitates a risk-based revenue assessment in the balancing market [101], addressing the importance of SI forecast.

This view is echoed by Hu et al. [58], who emphasize the growing role of BESS in European imbalance settlement and highlight their suitability to participate in high-resolution real-time markets due to their fast response capabilities.

Unlike day-ahead and intraday markets, the balancing energy market features inelastic demand from the TSOs. While energy providers (BSPs) can submit freely priced bids for balancing energy, TSOs are required to activate whatever quantity is needed to maintain system stability, regardless of the price. This lack of price sensitivity on the demand side necessitates the use of harmonised minimum and maximum price limits in order to avoid extreme costs spikes [32].

Chapter 3

Literature review

This chapter presents an overview of state-of-the-art research in energy forecasting, as introduced in Section 3.1. Section 3.2 reviews forecasting methods, with a particular focus on machine learning models commonly applied in the electricity domain and emphasizes recent developments in short-term system imbalance forecasting. Section 3.3 explores feature selection techniques, which constitute the *methodological foundation* of this thesis.

3.1 Introduction to system imbalance forecasting

Forecasting system imbalance is a challenging time-series prediction problem that remains relatively underexplored in academic electricity market research. While extensive literature exists on load and electricity price forecasting [13, 25, 55, 56, 110], less attention has been devoted to imbalance forecasting. Nonetheless, research on forecasting methods in the electricity sector is expanding rapidly, reflecting the growing importance of accurate predictions for real-time balancing and market incentives. As electricity markets become increasingly integrated and volatile, accurate short-term imbalance forecasts are essential for TSOs and market participants alike, as discussed in Section 2.2. This is not to say that market participants do not use such forecasting models; rather, the limited academic attention is likely due to fewer incentives to publish proprietary approaches used in practice.

Several studies have nevertheless highlighted the economic implications of forecast accuracy. Cui et al. [22] quantify how forecast error affects the benefits of participants in the imbalance market by modeling the error as a zero-mean Gaussian distribution. Similarly, Smets et al. [101] explore how Battery Energy Storage Systems (BESS) can exploit imbalance price volatility using short-term forecasts. Zhao et al. [118] extend this perspective by examining strategic bidding behaviours under uncertainty in both local demand and renewable generation, highlighting the role of forecast-informed decision-making in competitive settings. Hirth et al. [53] additionally examine forecast errors in variable renewable generation (VRG) and their impact on balancing reserve requirements.

3. LITERATURE REVIEW

Forecasting methods have evolved from simple statistical models to more advanced machine learning and deep learning approaches. Time series analysis aims to identify patterns and dependencies in historical data to predict future values [100]. Early methods, such as ARIMA and Exponential Smoothing, have long been applied in this domain [11, 19, 110]. While these models offer interpretability and remain useful in specific contexts, their underlying assumptions often do not hold for system imbalance data, which is typically noisy, unstructured and non-stationary. As noted by Garcia et al. [44], such characteristics limit the predictive power of traditional models in electricity markets. Nonetheless, their simplicity and transparency continue to offer value in certain applications, such as feature selection. A more detailed analysis on the characteristics of system imbalance data is provided in Section 5.1.

Both Garcia et al. [44] and Dumas et al. [25] conclude that multilayer perceptrons (MLPs) are better suited to capture nonlinear patterns present in imbalance data. These limitations of traditional models have thus driven the shift towards machine learning techniques.

3.2 Machine learning models

In recent years, machine learning (ML) has gained increasing traction in the context of forecasting and optimization problems in electricity markets. Two key categories of ML methods are particularly relevant: supervised learning and reinforcement learning. Both of these make use of neural networks (NNs).

This section discusses key ML models used for system imbalance forecasting, grouped by architecture and growing in complexity, from linear models to ensemble methods, recurrent networks and probabilistic approaches.

Linear models

Despite or maybe because of their simplicity, linear regression remains widely used due to their interpretability and low computational cost. For example, Belgium's TSO Elia relies on multivariate linear regression to forecast quarter-hourly system imbalance [28, 27]. Their approach uses a separate model for each minute of the quarter-hour, assuming that input-output relationships evolve over the course of the quarter-hour interval.

However, comparative studies have shown mixed results. A greek study [87] compared linear regression, random forest and LSTM for 15-min and 1-hour ahead system imbalance forecasts. While Elia reported better performance with linear regression (by MAE of 14%), the Greek study found random forest to be superior (by 18%). Such discrepancies highlight the importance of model configuration and feature selection. Even when using the same dataset, different users can arrive at different models due to model tuning strategies [100].

Multivariate linear models also face limitations due to spurious interactions, particularly when two input features, X and Y , are strongly correlated. This phenomenon is known as multicollinearity [21, 65]. In such cases, it becomes difficult to interpret individual feature effects, as multicollinearity can lead to unreliable coefficient estimates. This underscores the need for thoughtful feature selection.

Feedforward neural networks

Feedforward Neural Networks (FNNs), also known as Multi-Layer Perceptrons (MLPs), have shown promise for time-series forecasting tasks when combined with lagged inputs. Their advantages include relatively fast training, computational efficient, easy to interpret and the ability to model nonlinear relationships, striking a good balance between predictive performance and architectural complexity [9].

A study by Amjadi et al. [3] applied 24 independent FNNs in order to perform day-ahead electricity price forecasting, using one network per hour. While the model delivered promising results, the feature selection process was not fully disclosed, limiting transparency. Other works, such as Dumas et al. [25] and Pozzetti and Cartlidge [88], suggested that further investigation into FNNs with appropriate feature selection could be valuable, particularly as a lightweight alternative to more complex architectures. Bottieau et al. [9] also compared various MLP architectures for probabilistic forecasting, finding that their performance is inferior to that of encoder-decoder models, but similar to tree-based ensemble methods.

Direct comparisons between FNNs and linear models yield mixed results. For example, Elia [28] reported only a marginal improvement (0.1%) in favor of linear regression over a single-hidden-layer neural network. Similarly, Urdiales [107] found that a regularized linear regression model, incorporating L2 penalties to address coefficient instability, performs comparably to a MLP. In contrast, Garcia et al. [44] reported that a FNN with three hidden layers significantly outperforms linear regression, by at least 16%, in predicting weekly system imbalance volumes. These differences in model performance likely reflect not only distinctions in the forecasting target (imbalance vs. imbalance volume), but also differences in forecasting granularity, feature sets and network architectures used.

Ensemble models and random forest

Ensemble methods integrate multiple machine learning techniques into a single predictive model to reduce variance or to improve predictions. In time-series forecasting, they have shown strong performance, as evidenced by the M4 competition, where ensemble-based models ranked among the top performers [4]. Random Forests (RF), a bagging-based ensemble method, that constructs multiple decision trees and aggregates their outputs [12], have been successfully applied to imbalance forecasting across several countries. This approach shows robustness against noise and overfitting.

3. LITERATURE REVIEW

Konstantinos et al. [87] compared RF, MLP and Long Short-Term Memory (LSTM) networks for 15-minute-ahead forecasting, identifying RF as the most accurate and computationally efficient among the three. Balazs et al. [6] reached similar conclusions using a RF variant, showing that it outperformed LSTM models in capturing auto-correlated imbalance patterns with multiple lagged inputs. In Spain, Contreras [18] also adopted a RF approach for short-term imbalance prediction, noting the benefit of ensemble learning, because of its robustness, stability and competitive accuracy.

Beyond forecasting, RF can be interesting for feature selection, it provides measures such as permutation importance, which allow for evaluating the relevance of input variables [59]. Niu et al. [84] also demonstrated the effectiveness of this RF-based feature selection in the context of photovoltaic power forecasting, improving model performance by selecting informative features in high-dimensional datasets.

Other ensemble methods, not RF, used in energy context can be found in [45, 109]

Recurrent neural networks

Recurrent Neural Networks (RNNs) are well-suited for capturing temporal dependencies in sequential data and have therefore been widely applied in time-series forecasting tasks. A notable extension of RNNs is the Long Short-Term Memory (LSTM) architecture, developed by Hochreiter and Schmidhuber [54], which addresses the vanishing gradient problem through memory cells and gating mechanisms.

LSTM models have been applied to system imbalance forecasting with promising results. For instance, Konstantinos et al. [87] evaluated LSTM for 15-minute ahead system imbalance prediction and found its performance to be superior to that of linear regression and comparable to that of random forests. In the broader energy forecasting literature, LSTMs have also shown competitive performance. Bottieau et al. [10], in the context of day-ahead wind forecasting, reported higher accuracy for LSTM models compared to MLPs. The authors attribute this to the LSTMs ability to capture temporal dependencies, although details regarding the selection of input features, particularly the use of lagged variables, were not specified. Gompel et al. [45] implemented an LSTM architecture in combination with a Variable Selection Network (VSN), yielding a model that performed only 6% worse than their proposed model. While being informative, these results may not directly translate to the short-term system imbalance point forecasting tasks examined in this thesis.

LSTM models also appear frequently as baseline or reference architectures in related studies [6, 9, 72], further underscoring their role in the current forecasting landscape. Nevertheless, as with other deep neural network architectures, the application of LSTMs to high-dimensional datasets may introduce challenges. In particular, the risk of overfitting increases with the number of input features.

Probabilistic forecasting

Due to the inherent uncertainty in imbalance data, probabilistic forecasting is gaining importance as a complement to traditional point prediction methods. Rather than producing a single point estimate, these methods output prediction intervals that reflect underlying uncertainty. This uncertainty stems from multiple sources, including variability in explanatory variables (e.g. weather-driven renewables) and model misspecification [9].

TSOs have started integrating such probabilistic tools into their operations. For example, Norway's TSO, Statnett, developed a model based on Quantile Regression Forests (QRF) in order to forecast intra-hour imbalance distributions. QRF extends the traditional random forest algorithm to output prediction intervals, such as 95% confidence bounds, offering operators a probabilistic view of future imbalances. Similarly, a Belgian study introduced an ensemble of neural networks (C-VSNs) that yielded a 23% improvement in Continuous Ranked Probability Score (CRPS) for large imbalance events compared to earlier methods [9, 45].

Hybrid and transformer-based forecasting approaches

Recent advances in time-series forecasting have introduced architectures such as transformer models, which leverage multi-head self-attention mechanisms to capture long-range dependencies. These models, initially developed for natural language processing, have demonstrated strong performance in various forecasting tasks. However, recent studies have shown that carefully tuned linear models can match or even outperform transformer-based approaches in certain time-series contexts [115].

In parallel, hybrid and ensemble methods, combining multiple model types or learning paradigms, have demonstrated improvements over basic neural network models [51, 119].

Given the focus of this thesis on evaluating feature selection techniques using interpretable, efficient forecasting models, the implementation of transformer or hybrid architectures is not pursued. Nonetheless, these models remain relevant for future research, particularly in applications requiring probabilistic or long-horizon forecasting.

3.2.1 Operational integration

Machine learning-based forecasting methods are increasingly being adopted by European TSOs as decision-support tools for real-time imbalance management. In Belgium, Elia combines linear and logistic regression models for minute-level updates and recent collaboration with Ghent University could lead to improved probabilistic forecasts using ensemble neural networks [28, 45]. Norway's Statnett employs Quantile Regression Forests (QRF) to produce intra-hour forecasts with prediction intervals, supplementing rather than replacing its manual Planning Table tool [94].

3. LITERATURE REVIEW

France’s RTE publishes public forecasts with probabilistic indicators such as 1% risk margins [93], while the UK’s NESO is gradually deploying deep learning models through its Platform for Energy Forecasting (PEF) [81, 82, 119]. Other TSOs, such as those in Greece and Italy, have explored random forest and neural network models through academic partnerships. These developments demonstrate a broader trend in which TSOs cautiously integrate machine learning into operational workflows, using them to complement expert-based systems while improving responsiveness and forecast accuracy.

Across Europe, platforms, such as ENTSO-E, facilitate knowledge exchange and provide access to standardized historical data, enabling benchmarking and research. Moreover, harmonization efforts like MARI and PICASSO further motivate TSOs to refine local imbalance forecasts to ensure efficient cross-border balancing participation.

3.2.2 Model selection for this thesis

The primary objective of this thesis is to investigate feature selection techniques for short-term system imbalance forecasting. As such, model selection prioritizes interpretability, training efficiency and compatibility with feature evaluation frameworks over architectural complexity or state-of-the-art accuracy.

To this end, two forecasting models are selected as baselines: a multivariate linear regression model and a feedforward neural network (FNN) with a single hidden layer and ReLU activation. Temporal structure is captured via lagged input features derived from past imbalance values and related time-series signals. These models offer a favorable trade-off between simplicity, computational cost and predictive performance and are widely used in imbalance forecasting literature.

Many forecasting studies report only final model performance, without detailing the feature selection process, covariate construction, or model tuning strategies. This opacity makes it difficult to benchmark or reproduce existing results. The present work therefore emphasizes transparency in input selection and cross-validation, offering a structured approach in comparing feature selection techniques.

3.3 Feature selection methods

Feature selection (FS) plays a crucial role in time series forecasting, especially in domains where large volumes of data are recorded at high frequency. In many real-world applications, ranging from finance and healthcare to energy systems, models are required to process high-dimensional input data that may include redundant, irrelevant, or noisy features. Including all available inputs in such settings can lead to what is commonly referred to as the *curse of dimensionality* [74, 98], where the volume of the feature space grows so rapidly with the number of features that the available data becomes sparse. This sparsity makes it difficult for models to learn meaningful patterns, ultimately harming forecasting performance.

In the Belgian electricity market, for example, system imbalance is affected by a wide range of factors, including electricity load, renewable generation, cross-border exchanges, market trades, weather conditions and temporal effects. In this thesis, over 50 base features are considered. When lagged versions are added to capture time dependencies, the feature space can easily exceed several hundred variables. While the increasing availability of high-resolution data offers rich input for machine learning models, it also raises the risk of overfitting and leads to greater computational cost. Feature selection helps mitigate this risk by identifying the most informative predictors and discarding irrelevant ones, thereby improving both forecast accuracy and model interpretability [114].

While improvements in load forecasting have been shown to yield substantial savings, as an example, in the United Kingdom, a 1% reduction in load forecast error has been estimated to save approximately 10 million pound in operational costs [59]. The economic value of accurate system imbalance forecasting lies in its ability to minimize imbalance costs for BRPs and to enable profit opportunities for flexible assets, like energy storage system operators [101]. Although exact cost savings are harder to quantify, the underlying incentive is clear [68, 111].

Despite its importance, feature selection is often underreported in forecasting literature. For example, in [3], 14 features were selected for training 24 feedforward neural networks, but no rationale for feature choice was provided, limiting reproducibility. This lack of transparency is common and poses challenges for comparative research and industrial deployment.

To address high dimensionality, two broad approaches are typically used: feature selection and feature extraction. Feature selection methods aim to retain a subset of the original features, whereas feature extraction transforms inputs into a new set of variables. FS is particularly valuable in contexts where data is continuously generated, as it helps eliminate irrelevant or redundant inputs [114].

Feature selection techniques are commonly categorized as filter methods, wrapper methods, embedded methods (Table 3.1), or hybrids thereof. While the ideal solution would involve an exhaustive search over all possible feature subsets, this is computationally infeasible except for very small feature spaces.

3.3.1 Filter methods

Filter methods perform feature selection by ranking features based on a performance measure derived often from statistical measures, without involving a predictive model. These methods assess the intrinsic properties of the data and are typically used as a preprocessing step to eliminate irrelevant or redundant features before model training. Because filter methods do not require training separate models for each feature subset, they are computationally efficient and less susceptible to overfitting

3. LITERATURE REVIEW

to a particular model’s structure [89]. However, a key limitation is that they do not account for feature interactions or model-specific learning dynamics [71, 98, 109].

In energy forecasting, simple filter techniques, like correlation analysis, are commonly used as a first step. For instance, researchers often compute Pearson correlation coefficients between candidate inputs and the target load or price (cross-correlation), removing those with negligible correlation [72]. Xie et al. applied this approach to electric vehicles charging load data, using Pearson correlation to identify key influential factors (such as new COVID case counts and weather variables) before modeling [112]. Similarly, mutual information (MI) has been employed to capture nonlinear associations. Amjadi & Keynia [3] proposed a MI-based feature selection algorithm for electricity price forecasting, which was later adopted by other studies. Sun et al. [57] used a MI-based metric to evaluate how different features contribute to load at various times. An advanced filter criterion, Maximum Relevance and Minimum Redundancy (mRMR), builds on MI by rewarding features that are highly relevant to the target but minimally redundant with each other. mRMR has seen successful application in load forecasting. For example, Liang et al. [75] optimized short term load forecasting by first selecting features via mRMR. The appeal of mRMR lies in its ability to penalize redundant inputs (e.g. multiple highly correlated sensors) while keeping the most predictive ones. Other filter methods referenced in energy contexts include statistical tests and variants of RelieF, though these are more common in broader machine learning literature [40].

3.3.2 Wrapper methods

Wrapper methods select features by directly evaluating forecasting model performance. In a wrapper approach, one trains a candidate model (e.g. a linear regression or neural network) on various subsets of features and uses the model’s error as feedback to choose the best subset. This approach can consider feature dependencies and tailor the selection to a specific predictive algorithm, often yielding higher performance than filters for that model. However, exhaustively testing all feature combinations is computationally infeasible, so wrappers rely on heuristic strategies [71, 89, 109].

Despite their cost, wrappers are frequently used in energy forecasting research, often in simplified forms such as stepwise regression. Early applications in electricity price forecasting include the work of Karakatsani and Bunn [63] and Misiorek [80], who applied stepwise elimination of statistically insignificant variables in autoregressive models, guided by t-tests.

Two widely adopted wrapper techniques are *Recursive Feature Elimination* (RFE) [15, 72] and *Sequential Backward Selection* (SBS) [59, 114]. Both begin with the full feature set and iteratively remove one feature at a time. However, their selection strategies differ. RFE ranks features based on model-derived importance, such as absolute coefficient magnitudes in linear models or feature importance scores in tree-based models and removes the least important feature at each iteration, re-

training the model throughout. On the contrary, SBS evaluates all possible feature subsets formed by removing one feature and selecting the best-performing subset at each step. Such sequential strategies strike a balance between simplicity and effectiveness, making them attractive for identifying high-performing feature sets [15].

More advanced wrapper strategies use metaheuristic search algorithms, such as *Genetic algorithms* (GA) and *particle swarm optimization* (PSO), in order to explore the feature space more flexibly [15]. For instance, Li et al. implemented a wrapper-based hybrid model combining PSO and GA to select inputs for an LSTM-based day-ahead electricity price forecaster [72, 114].

A known risk with wrappers is overfitting [89]. This can be mitigated by restricting the search (stepwise selection or two-stage approaches) or by employing cross-validation to evaluate model generalization [69, 72]. When done carefully, wrappers have yielded substantial accuracy gains. For example, Keles's hybrid wrapper approach and similar studies demonstrated improved price forecast accuracy by custom-selecting inputs for their models [64].

3.3.3 Embedded methods

Embedded feature selection methods integrate the selection process into the model training itself. In other words, the model naturally selects or emphasizes important features as part of the learning. Classic examples include regularization techniques and decision-tree-based algorithms: these models have built-in mechanisms that perform feature selection or produce feature importance metrics as they fit the data [71, 89, 109]. Embedded methods strike a balance between filters and wrappers by leveraging model feedback without requiring multiple training runs.

One prominent embedded approach in energy forecasting is penalized regression. The Least Absolute Shrinkage and Selection Operator (LASSO) regression adds an L_1 penalty that drives the coefficients of less useful features to zero, effectively removing them. Recent electricity price forecasting studies have found LASSO-based models to outperform traditional expert-selected models. Uniejewski et al. [106] conducted a thorough comparison of automated selection methods in EPF, showing that LASSO and its extension (elastic net) can yield significant accuracy gains over unregularized models. In practice, using a complex linear model with hundreds of candidate features, LASSO can shrink most coefficients to zero and retain a parsimonious set that generalizes well. For example, Barnes and Balda [7] were among the first to apply regularization in EPF, using ridge regression (an L^2 -penalty variant) to predict New York electricity prices with over 50 inputs. Ludwig et al. [76] then explicitly used LASSO to select relevant weather station features for German price forecasting, achieving up to 17% in Mean Absolute Error (MAE) improvement by automating what had traditionally been manual selection.

Another major class of embedded FS comes from decision tree ensembles. Tree-based

3. LITERATURE REVIEW

models, such as Random Forests (RF) and Gradient Boosted Trees, inherently perform FS by selecting the most informative split variables during training. Although they typically use all features (unless explicitly limited), one can derive feature importance scores from the trained model to determine which inputs had predictive value. In the energy domain, RF have frequently been used, not only as forecasting models but also as a tool to rank features. For example, Ludwig et al. [76] used random forest variable importance alongside LASSO in their price study and found largely overlapping sets of top weather predictors. Gonzalez et al. [46] also applied random forests in a ML approach for intraday price forecasting. A specific example in system imbalance forecasting is the use of Quantile Regression Forests (QRF) by Salem et al. [94] for the Norwegian's TSO. Similarly, Huang et al. [59] and Niu et al. [84] applied permutation importance in RF to reduce dimensionality in load and PV forecasting, respectively. Tree-based embedded selection is convenient because it can handle non-linear relationships and interactions implicitly. However, one limitation is that tree models might sometimes consider many features of marginal importance rather than performing a strict subset selection, thus requiring a cutoff to define a selected subset.

Another embedded method involves Variable Selection Networks (VSNs), in which neural networks learn attention weights for each input and can suppress uninformative features during training [45].

For an overview of the feature selection categories, including their defining features and representative methods, refer to the taxonomy in Table 3.1.

TABLE 3.1: Strengths, weaknesses and examples of the three main feature selection categories. Adapted from [89]

Feature Selection Method	Strengths	Weaknesses	Examples
Filter- Univariate	Fast and scalable	Feature dependencies not modeled	Pearson correlation [112]
	Independent of classifier	Interaction with classifier not modeled	Mutual information [3]
	Reduce risk of overfitting		χ^2 /chi-squared test [98]
Filter- Multivariate	Can model feature dependencies	Slower and not as scalable as univariate filters	Cross-correlation [72]
	Independent of the classifier	Interaction with classifier not modeled	minimal-Redundancy-Maximal-Relevance [75]
	Less risk of overfitting		Relieff algorithms [59]
Wrapper	Model feature dependencies	Slower than filter and embedded methods	Sequential forward and backward selection [89]
	Better performance than filter method	More prone to overfitting	Particle swarm [114]
	Model interaction with classifier	Selected features are classifier dependent	Genetic algorithm [72] Recursive feature elimination
Embedded	Model feature dependencies	Slower than filter methods	Random forest [12]
	Faster than wrapper	Selected features are classifier dependent	Lasso (L1) regression [106]

3.3.4 Dimensionality reduction

Unlike feature selection, Dimensionality Reduction (DR) transforms the original feature space into a smaller set of derived features, rather than selecting a subset of existing ones. This can significantly reduce dimensionality with minimal information loss. A key drawback, however, is reduced interpretability, as the transformed features are typically linear or nonlinear combinations of the original inputs [61]. Techniques such as Principal Component Analysis (PCA) and autoencoders are widely used in energy forecasting.

PCA has been applied in load and price forecasting for decades. It identifies orthogonal linear combinations (principal components, PCs) that capture the greatest variance in the data. By retaining only the first few PCs, dimensionality is reduced while preserving most of the informational content. For example, Veeramsetty et al. [108] applied PCA to reduce nine input features to six PCs, which explained approximately 90% of the variance, before feeding them into an RNN model. PCA has also been used to compress meteorological data prior to LSTM-based forecasting [49]. Hong and Wu [56] proposed a hybrid model combining a PCA neural network (trained with Sanger's rule [96]) and a multilayer feedforward network for day-ahead price forecasting.

Autoencoders, a nonlinear counterpart to PCA, are neural networks trained to compress and reconstruct input data. Once trained, the encoder provides a reduced representation for downstream forecasting models. Stacked autoencoders (SAEs) have been used to capture nonlinear patterns in energy data. He and Wang [51] combined Recursive Feature Elimination (RFE) with an SAE to compress features before forecasting stock indices, an approach mirrored by Zhai et al. [117] in short-term load forecasting, where selected inputs were compressed using an SAE before being fed to an LS-SVM model.

3.3.5 Application to system imbalance forecasting

System imbalance forecasting is a relatively recent application in the energy domain and dedicated literature on feature selection remains limited compared to load or price forecasting. Nonetheless, several studies offer insights into effective approaches.

Early work by Garcia and Kirschen [44] relied on expert knowledge to select explanatory variables for forecasting imbalance volumes. Kratochvil [67] later identified key imbalance predictors in the Czech system, implicitly applying a filter-based selection by ranking demand- and supply-side features. The Czech market, however, uses a 60-minute settlement period, unlike the 15-minute resolution in most European countries. Contreras [18] also highlighted relevant features via a linear forecasting and bidding model, though without formal FS methodology. Urdiales [107] constructed an input feature set for system imbalance forecasting by evaluating correlations, effectively applying a filter method.

3. LITERATURE REVIEW

More recent studies employ embedded and wrapper techniques. For example, random forest models have been used in Spain and Norway to forecast imbalances and derive variable importance scores. Gompel et al. [45] introduced a neural network ensemble with a variable selection network (VSN), dynamically suppressing less relevant inputs. Their use of ablation studies, a wrapper method, quantified the marginal value of each input, illustrating that including recent imbalances of flexible assets improves predictive accuracy.

Each feature selection (FS) approach offers distinct advantages for energy time-series forecasting. Filter methods (e.g. correlation, MI, mRMR) are computationally efficient and model-agnostic, making them suitable for initial variable screening. However, they often overlook feature interactions and model-specific relevance. Wrapper methods tailor the feature set to a specific model by evaluating predictive performance directly, but are computationally intensive and prone to overfitting without proper validation.

Embedded methods, such as LASSO or tree-based models, balance efficiency and model-specific selection. They have been widely adopted in recent literature for their ability to produce parsimonious and accurate models. Their main limitation is model dependency, selected features may not generalize across model types. Dimensionality reduction techniques, such as PCA and autoencoders, compress the input space and can enhance model stability, though at the cost of interpretability.

In this thesis, ten representative feature selection methods from different categories are implemented: Pearson correlation (filter); Random Forest and LASSO (embedded); and several wrapper methods including Recursive Feature Elimination (RFE), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and sequential selection strategies (SBS, SFS, SBFS, SFFS). These methods are applied in combination with linear regression and a shallow neural network to evaluate and compare their performance.

Chapter 4

Methodology

This chapter outlines the methodology used to forecast system imbalance using feature selection and machine learning. Section 4.1 defines the forecasting target. Section 4.2 follows with a description of the dataset and the preprocessing steps applied. Section 4.3 presents an exploratory analysis to support the feature selection process. Building on this, Section 4.4 details the feature selection procedures. Section 4.5 then introduces the forecasting models used in this study. To ensure reliable model evaluation, Section 4.6 details the time-series cross-validation strategy employed. Finally, Section 4.7 outlines the performance metrics used to assess and compare the forecasting results.

4.1 Forecasting target

The aim of this work is to forecast the system imbalance value for the next quarter-hour, SI_{qh+1} , from the viewpoint of the current quarter-hour, SI_{qh+0} . Specifically, the forecast is issued at the 3th minute of $qh + 0$, based on all information available up to that point, aligning with the operational requirements for bid evaluation, coordination and BSP activation. See Section 2.2. While the average system imbalance of the current quarter-hour, SI_{qh+0} , is not known at the time of prediction, partial minute-wise telemetry data for $qh + 0$ is available and incorporated into the input features. Care is taken to ensure no time leakage occurs.

The primary focus of this thesis is feature selection: identifying which features contribute most to forecast accuracy when using machine learning models and defining their underlying dynamics. Two baseline models are used, a Linear Regression (LR) model and a Multi-Layer Perceptron (MLP) model, which are evaluated alongside different feature selection techniques.

4.2 Data collection

All data used in this thesis is sourced from Elia, the Belgian TSO, via its open data platform. To maintain consistency and manageability, the Belgian data after covid-19

4. METHODOLOGY

and pre-Elia’s accession to MARI, covering the period from 2022 to 2023, is used. The original data spans multiple temporal resolutions. Most features are available on a quarter-hourly basis, while others are recorded at hourly or minute-wise granularity.

Preprocessing addressed several quality-related issues. Missing values were rare and handled using linear interpolation or forward-filling, depending on the feature and gap length. All timestamps were normalized to Central European Time (CET) and all features were normalized to a [0, 1] range using the MinMax scaler, ensuring that each feature contributed proportionally during model training. To address differences in data granularity, lower-resolution data were upsampled within their respective time intervals. Although fast Fourier transform-based resampling has been proposed as an alternative [87], direct temporal upsampling was deemed more suitable for this application, as its preserves the signal’s original meaning.

Spike handling, such as placing an upper limit [110], was not applied, as large imbalance events are considered informative during volatile periods [45]. Special days, such as public holidays, were not treated separately, although prior work has recommended smoothing anomalies based on surrounding periods [103].

4.3 Exploratory data analysis

Prior to formal feature selection, an exploratory time-series analysis was conducted to examine temporal dynamics and lead-lag relationships between system imbalance and other relevant variables. This step is necessary to reduce the extensive Elia database to a manageable set of inputs for the forecasting models.

Filter-based metrics were then applied as an initial step to remove irrelevant or highly redundant features, thereby reducing the search space for more targeted selection methods used in subsequent stages. This sequential approach aligns with a hybrid selection strategy.

4.3.1 Filter methods

As introduced in Section 3.3, filter methods assess the relevance of features based on intrinsic statistical properties of the data, independently of the forecasting model. Filter methods are fast and model-agnostic, making them suitable for a initial screening to eliminate irrelevant inputs. Two well-established ranking methods are applied: the Pearson correlation coefficient and Mutual Information (MI).

Due to their inability to account for multivariate interactions, filter methods are typically used as a first pass selection step in forecasting pipelines, often followed by more targeted methods, such as wrapper and embedded techniques to account for feature interactions. This preliminary analysis is performed in Chapter 5.

Correlation criteria

One of the simplest criteria to assess the linear dependence between a candidate feature $X^{(j)}$ and the target variable y is the **Pearson correlation coefficient**. It is defined as the covariance between two processes divided by the product of their standard deviations:

$$R(j) = \frac{\text{cov}(x_j, y)}{\sqrt{\text{var}(x_j) \cdot \text{var}(y)}}, \quad (4.1)$$

where $\text{cov}(\cdot, \cdot)$ denotes covariance and $\text{var}(\cdot)$ denotes variance [16].

When this criterion is applied to lagged versions of the target variable itself (i.e. $x_j = y_{t-k}$), it becomes equivalent to the **autocorrelation function** (ACF), which measures the linear relationship between current and past values of the same signal. High autocorrelation at specific lags reveals persistent temporal structure and justifies inclusion of autoregressive terms.

In contrast, the **partial autocorrelation function** (PACF) isolates the effect of a specific lag k by removing the influence of intermediate lags $1, \dots, k-1$. PACF directly informs the selection of truly explanatory lags for autoregressive modeling by identifying which past values add unique predictive power.

High absolute values of $R(j)$ (typically above a predefined threshold) indicate strong linear association with the target and justify the inclusion of the corresponding feature in initial model iterations. However, Pearson correlation is limited in scope. It only captures linear relationships and cannot detect nonlinear dependencies or redundancy among features.

While autocorrelation and PACF assess internal memory within the target signal, the **Cross-Correlation Function** (CCF) quantifies the temporal dependence between an exogenous variable X and the target Y . Specifically, CCF measures the linear correlation between X_{t-k} and Y_t across a range of time lags $k \in \mathbb{Z}$.

It is defined as:

$$\rho_{XY}(k) = \frac{\text{cov}(X_{t-k}, Y_t)}{\sqrt{\text{var}(X_{t-k}) \cdot \text{var}(Y_t)}} \quad (4.2)$$

By computing $\rho_{XY}(k)$ for both positive and negative values of k , the cross-correlation plot reveals whether the exogenous signal *leads* or *lags* behind the target.

For example, a strong peak at positive lag k suggests that past values of X may contain predictive information for Y .

CCF helps identify explicit lag structures between two time series. The results

4. METHODOLOGY

of CCF analysis informed the construction of lagged exogenous features. For each input variable X , a set of lags $\{k_1, k_2, \dots\}$ with significant cross-correlation was retained and used as candidate predictors.

Mutual Information (MI)

To address the limitations of linear correlation, *Mutual Information* (MI) is used as a complementary relevance criterion. MI measures the total dependence between a feature X_j and the target Y , including both linear and nonlinear associations. It is defined as:

$$I(X; Y) = H(Y) - H(Y|X), \quad (4.3)$$

where $H(Y)$ denotes the Shannon entropy of Y , and $H(Y|X)$ the conditional entropy of Y given X . In other words, mutual information measures how much knowing X reduces the uncertainty about Y [102].

Shannon entropy is a fundamental concept from information theory and describes the average unpredictability or information content in a variable [97]. A variable with high entropy has more variability and is harder to predict. When two variables are independent, observing one provides no information about the other, and mutual information is zero. Conversely, high MI values indicate strong dependency, regardless of whether the relationship is linear or nonlinear.

The filter methods provided an initial screening by producing a ranked list of features, and typically the top N features are retained. The union of these candidates then entered the next stage of evaluation. It is worth noting that filter methods do not account for inter-feature correlations. Two highly correlated features might both rank high even though one would suffice. Therefore, these results are treated as suggestive rather than definitive, proceeding to more informed selection techniques next.

The reduced subset of features resulting from this preliminary analysis served as input to the subsequent feature selection procedures and model training. Further details on this process are provided in Chapter 5. An overview of the key features analysed is presented in Table 4.1. The availability column refers to when each feature becomes known relative to the prediction time, $qh+0$. Lagged values (e.g., $qh-1$, $min-2$) are known at $qh+0$. Forecasts labeled ($t+24$) are fixed Day-Ahead (DA) values published in advance, while ($qh+24$) refers to Intraday (ID) forecasts typically made around 6 hours before delivery. The full list of 45 individual features is included in Section A.1.

Abbreviation [unit]	Description	Availability at qh+0
	<i>Quarter-hour parameters</i>	
SI [MW]	System Imbalance	qh-1
NRV [MW]	Net regulation volume, GUV-GDV	qh-1
IP [€/MW]	Imbalance price	qh-1
MIP [€/MW]	Marginal incremental price	qh-1
MDP [€/MW]	Marginal decremental price	qh-1
LOAD_RT [MW]	Total load in Belgium	qh-1
LOAD_ID [MW]	Intraday (ID) load forecast	qh+24
LOAD_DA [MW]	Day ahead (DA) load forecast	qh, t+24
NETPOS_BE_ID [MW]	ID implicit net position Belgium	qh+24
XB_DA [MW]	Cross-border DA commercial schedule (nominated capacity)	qh, t+24
XB_RT [MW]	Cross border physical flow	qh-1
WIND_ID [MW]	Wind Intraday Forecast	qh+24
SOLAR_ID [MW]	Solar Intraday forecast	qh+24
GEN_DA [MW]	Day ahead generation schedule	qh, t+24
GUV	Total activated upward balancing energy	qh-1
IGCC+	Upward IGCC exchanges	qh-1
aFRR+	Upward activated aFRR	qh-1
mFRR+	Upward activated mFRR	qh-1
	<i>Minute-wise parameters</i>	
SI_min [MW]	Cumulative system imbalance	min-2
NRV_min [MW]	Cumulative net regulation volume	min-2
IP_min [€/MW]	Positive imbalance price	min-2
	<i>Hourly parameters</i>	
NETPOS_GB_DA [MW]	Day ahead net position United Kingdom	h, t+24
NETPOS_GB_ID [MW]	Intraday net position United Kingdom	h, t+24

TABLE 4.1: Overview of key features and their availability timing at qh+0.

4.4 Feature selection methods

Following the exploratory analysis in Chapter 5, an initial feature set is constructed based on correlation. Subsequently, filter, wrapper and embedded methods are applied, as introduced in Section 3.3. The filter method is also used to construct a candidate feature set based on correlation. Wrapper and embedded methods refined the initial feature set based on model-specific performance. The complete workflow is summarized in Algorithm 1, which provides a step-by-step overview of how the final feature subset is determined.

4. METHODOLOGY

Algorithm 1 Feature Selection Procedure

Require: All available features F , target variable y (SI_{qh+1})

- 1: Perform exploratory time-series analysis (see Chapter 5)
- 2: **for** each feature $f \in F$ **do**
- 3: Compute filter-based relevance score (e.g. Pearson, MI)
- 4: **end for**
 ▷ Filter method
- 5: Select top- k features to form F_{filter}
- 6: Create candidate subsets from F_{filter}
 ▷ Wrapper method
- 7: **for** each subset $S \subseteq F_{\text{filter}}$ **do**
- 8: Train model using S
- 9: Evaluate performance (MAE, RMSE, SMAPE with cross-validation)
- 10: **end for**
11: Extract selected features as F_{wrapper}
 ▷ Embedded method
- 12: Apply embedded method on F_{filter}
- 13: Extract selected features as F_{embedded}
 ▷ Final feature set
- 14: Determine F_{final} based on applied FS methods (see Chapter 6)
- 15: **return** Final feature set F_{final}

4.4.1 Wrapper methods

Wrapper methods involve selecting features by repeatedly training and evaluating a predictive model on different feature subsets. Wrappers directly optimize for the end forecasting objective, often leading to better in-sample performance than filters. They can detect interactions. For instance, a wrapper might keep two features that are only useful together. However, this comes at the cost of heavy computation: training numerous models on different feature sets. There is also a danger that a wrapper will overfit the feature selection to historical data, especially if the evaluation criterion is not properly cross-validated [89].

Thus a time-aware cross-validation method, described in Section 4.6, is used to evaluate performance at each number of features, and the subset size with the best performance was chosen.

Recursive feature elimination

Recursive Feature Elimination (RFE) is a greedy wrapper method that iteratively removes the least important feature under a given model. The linear regression (LR) model and Multi-Layer Perceptron (MLP) are used as the base estimators for RFE. The procedure starts with all candidate features, then at each step trains

the model and computes an importance metric (the absolute weight magnitude for linear regression and a proxy like permutation importance for MLP). The feature with the smallest importance is removed and the model is refit with the remaining features. This process "eliminates" features one by one, until a desired number of features remains (or until further removal causes a significant drop in performance).

RFE is computationally heavier, because it requires training many models. It often yields very good subsets considering feature dependencies. For the linear model, RFE tends to mimic stepwise regression, removing features that add least incremental explanatory power. For the MLP, a variant of RFE is employed using the network's learned weights to decide feature importance. This wrapper strategy directly optimizes selection for each model, albeit with high computational cost.

Sequential selection methods

As an alternative wrapper strategy, sequential selection methods are used. These approaches are similar to RFE. They start with either all features or none, and they iteratively remove or add features one at a time. However, the key difference is that the inclusion or removal is explicitly guided by model performance at each step.

Sequential Backward Selection (SBS). SBS starts with the full feature set and removes one feature at a time. At each step, it evaluates the validation error for all possible single-feature removals and eliminates the feature whose absence leads to the smallest increase (or largest decrease) in error. Unlike RFE, which ranks features based on importance metrics from a single model fit, SBS directly uses performance-based evaluation at each step. This allows it to account for interaction effects between features. As with other wrappers, the method is computationally expensive but benefits from parallelization.

Sequential Forward Selection (SFS). SFS is the forward counterpart to SBS. It begins with an empty set and iteratively adds the feature that provides the greatest improvement in performance. SFS is less robust to early suboptimal selections since no features are removed once added.

Sequential Floating Selection. To mitigate the limitations of greedy methods, such as SFS and SBS, floating variants introduce a flexible backtracking mechanism. In Sequential Forwarded Floating Selection (SFFS), after each feature inclusion (forward step), the algorithm performs one or more conditional backward steps, removing already-selected features if their exclusion leads to improved model performance. This dynamic approach allows recovery from suboptimal decisions and enables a more thorough exploration of the feature space. The flow of the SFFS algorithm is illustrated in Figure 4.1, where d denotes the desired number of selected features.

A similar mechanism underlies Sequential Backward Floating Selection (SBFS), which starts from the full feature set. At each iteration, the algorithm removes the

4. METHODOLOGY

feature whose exclusion leads to the smallest performance degradation (or largest improvement), as in SBS. However, unlike SBS, SBFS introduces conditional inclusions: after each removal, it evaluates whether reintroducing any previously eliminated feature improves the model. If so, the feature is added back. This flexible backtracking allows the algorithm to correct earlier missteps, avoiding the rigidity of greedy backward elimination and improving overall feature subset quality.

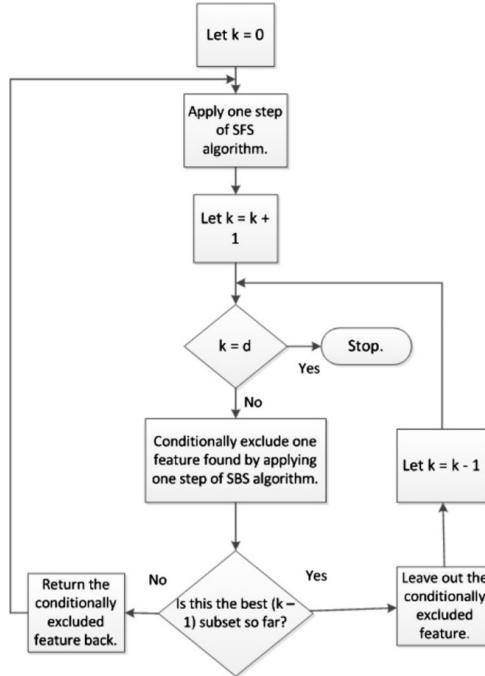


FIGURE 4.1: SFFS flow chart, from [15].

Heuristic Search Methods In addition to sequential methods, heuristic search algorithms such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) offer a more global perspective on feature selection. These methods explore the search space more broadly and are less prone to being trapped in local optima, making them attractive for high-dimensional or non-convex feature selection problems.

Particle Swarm Optimization (PSO) [14] models the optimization process as a swarm of particles moving through the search space. Each particle adjusts its position based on its own best-known solution and the globally best-known solution. The velocity update rule is given by:

$$\mathbf{v}(t+1) = u\mathbf{v}(t) + c_1r_1(\mathbf{p}(t) - \mathbf{x}(t)) + c_2r_2(\mathbf{g}(t) - \mathbf{x}(t)), \quad (4.4)$$

where $\mathbf{v}(t)$ and $\mathbf{x}(t)$ denote the current velocity and position, $\mathbf{p}(t)$ and $\mathbf{g}(t)$ are the personal and global best positions found so far, u is the inertia weight, c_1, c_2 are

cognitive and social coefficients, and r_1, r_2 are random scalars in $[0, 1]$. The particle's new position is:

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \mathbf{v}(t+1). \quad (4.5)$$

PSO iteratively updates the swarm until convergence criteria are met, selecting feature subsets that optimize the forecast performance metric.

Genetic Algorithms (GA) [72]. encode feature subsets as binary chromosomes, where each gene represents the inclusion (1) or exclusion (0) of a feature. The algorithm begins with a random population of such chromosomes and evolves them using selection, crossover and mutation. Fitness is typically evaluated by the forecasting performance of a model trained on the selected features. Poor performers are replaced, fitter individuals are propagated. Over successive generations, the GA converges towards high-quality feature subsets.

4.4.2 Embedded methods

Embedded methods perform feature selection during model training by incorporating regularization or built-in importance metrics. They are model-aware and efficient, often achieving a strong balance between predictive performance and computational cost. In this thesis, LASSO and Random Forests are employed, two commonly used techniques in energy forecasting due to their scalability, interpretability, and empirical success [76]. However, the performance of embedded methods is inherently tied to the assumptions of the underlying model and requires careful hyperparameter tuning.

LASSO Regression

Lasso regression extends LR and MLP by adding an L1 regularization term, that penalizes the values of the coefficients. The objective is on the one hand to minimize the least squared error between the predicted and actual values, and on the other hand to enforce sparsity in the model by shrinking certain coefficients to zero [50]. Formally, the LASSO problem in Lagrangian form is:

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta}} \left\{ \frac{1}{2} \sum_{i=1}^N \left(y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j \right)^2 + \lambda \sum_{j=1}^p |\beta_j| \right\} \quad (4.6)$$

where:

- y_i is the target value for observation i ,
- x_{ij} is the value of feature j for observation i ,
- β_0 is the intercept term,
- $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)$ are the feature coefficients,

4. METHODOLOGY

- $\lambda \geq 0$ is the regularization parameter controlling the strength of the penalty.

The optimal value of λ is tuned using cross-validation, as λ increases, more coefficients are shrunk to zero, thereby doing a kind of continuous subset selection. Features corresponding to non-zero coefficients at the optimal λ were retained. LASSO is computationally efficient and jointly considers all features, accounting for their linear dependencies. However, in the presence of highly correlated predictors, LASSO tends to select one arbitrarily while discarding the others. While this behaviour is not problematic from a predictive standpoint, it can reduce interpretability by obscuring equally informative correlated features.

Random Forest Importance

Random Forest (RF) [12] is an ensemble learning method that constructs multiple decision trees, each trained on a bootstrap sample of the data (hence "random" forest) and a random subset of features. At each node, the best split is selected based on impurity reduction, typically the mean squared error for regression tasks. This dual source of randomness decorrelates the individual trees, reduces variance and increases generalization performance.

In this work, RF is used in two distinct roles:

- a stand-alone predictive model, in order to provide a nonlinear benchmark for forecasting performance
- a feature selection tool, where its internal importance measures are used to identify relevant features before the final forecasting model (LR or MLP).

Although RF is traditionally considered an embedded method (since it derives feature importance internally), it is more filter-like, since the importance scores are derived from a model trained separately from the target forecaster. Forecasting results using Random Forest as the predictive model are also reported.

Feature importance is quantified in two ways. First, the impurity-based importance quantifies the total reduction in the splitting criterion (e.g. root mean squared error) attributed to each feature across all trees. Second, permutation importance offers a more robust, model-agnostic alternative by estimating the increase in prediction error when a feature's values are randomly shuffled in the out-of-bag (OOB) data. For feature F_j , the permutation importance is computed as:

$$PI(F_j) = \frac{1}{B} \sum_{b=1}^B \left[\text{OOBError}_b(F_j^{\text{shuffled}}) - \text{OOBError}_b \right], \quad (4.7)$$

where B is the number of trees and OOBError is the error on the out-of-bag samples.

This dual approach captures both main and interaction effects, making RF particularly suitable for high-dimensional datasets with nonlinear relationships. Based on

these importance scores, RF retains the most influential features for use in downstream forecasting models. This not only improves interpretability and reduces overfitting risk, but also enhances computational efficiency.

The full RF training configuration is provided in Appendix B.1.

4.4.3 Feature reduction

Principal Component Analysis (PCA) is a widely used dimensionality reduction technique, and finds orthogonal linear combinations of the original variables. The complete PCA procedure is detailed in Appendix B.2. Conventionally, PCA is performed by centering the data and solving the eigenvalue problem of the sample covariance matrix. However, on larger datasets, it is often implemented via Singular Value Decomposition (SVD) of the centered data matrix [62]. In this formulation, the principal components correspond to the right singular vectors and the proportion of explained variance is determined by the squared singular values.

In this study, PCA is applied using the scikit-learn implementation [86], which adopts the SVD-based approach. To balance dimensionality reduction with interpretability, a hybrid strategy is used: after model-based feature selection (e.g. SBS or LASSO), the most relevant features are retained in their original form, while PCA is applied to the remaining, less informative or highly correlated features. The resulting principal components are then appended to the retained feature subset and used as input to the forecasting models described in Section 4.5. This strategy helps mitigate multicollinearity, reduces noise and improves computational efficiency, while preserving the interpretability of key features for downstream analysis.

4.5 Modelling

This section presents the theoretical background regarding the machine learning models, employed to evaluate the effects of feature selection on system imbalance forecasting. Two model types are considered: a linear regression model and a feed-forward neural network (MLP) capable of capturing temporal dependencies. Each model is trained separately using different feature subsets generated by the selection techniques introduced in Section 4.4.

The underlying principle of these models is to transform the input data through successive layers, linear or nonlinear, thereby enabling the learning of complex relationships between features and the forecasting target.

4.5.1 Linear Regression

Linear regression serves as a competitive benchmark model, as seen in Section 3.2. The forecast \hat{y}_{t+h} is modeled as a weighted sum of M selected features $x_t^{(j)}$ at time t :

$$\hat{y}_{t+h} = \beta_0 + \sum_{j=1}^M \beta_j x_t^{(j)}, \quad (4.8)$$

where β_0 is the intercept and β_j are the coefficients learned by minimizing the mean squared error on the training data. This model is retrained for each feature subset and forecast horizon ($h = 1$), offering a simple baseline to assess the incremental benefit of more complex architectures.

Despite its simplicity, linear regression serves as a useful benchmark and offers insight into feature effects through its coefficients.

Libraries, such as scikit-learn, have made regression modeling accessible to a wider audience. The linear regression algorithm used is based on Ordinary Least Squares (OLS), a widely adopted method that estimates the optimal linear relationship by minimizing the sum of squared residuals. OLS is valued for its computationally efficiency and interpretability. The content is mainly adopted from [48], that implements the scikit-learn library for different models.

4.5.2 Multi-Layer Perceptron (MLP)

In supervised learning, the goal is to approximate a functional relationship between input features and a target output. Multi-Layer Perceptrons (MLPs), a type of feedforward neural network, are well-suited for capturing nonlinear dependencies in such settings. Their flexibility makes them particularly effective for forecasting tasks where the relationship between inputs and the target variable is not strictly linear.

This section introduces the MLP architecture used in this study, followed by an explanation of the training procedure, optimization strategies, regularization methods and hyperparameter tuning.

The MLP architecture maps the input vector $\mathbf{x}_t \in \mathbb{R}^M$, composed of M selected features at time t , to a forecasted output \hat{y}_{t+h} via one hidden layer:

$$\mathbf{z}^{(1)} = \sigma(W^{(1)}\mathbf{x}_t + \mathbf{b}^{(1)}), \quad (4.9)$$

$$\hat{y}_{t+h} = W^{(2)}\mathbf{z}^{(1)} + b^{(2)}, \quad (4.10)$$

where $W^{(1)}, W^{(2)}$ are weight matrices, $\mathbf{b}^{(1)}, b^{(2)}$ are biases, and $\sigma(\cdot)$ denotes the activation function. In this study, the Rectified Linear Unit (ReLU), defined as $h(z) = \max(0, z)$, is used in all hidden layers. ReLU is widely adopted in deep learning due to its computational simplicity and ability to accelerate convergence [116]. It is also employed by Elia in their forecasting documentation [28]. An example of such an MLP architecture is illustrated in Figure 4.2, which depicts a network with six input features, one hidden layer of three neurons, and a single output unit.

There are several choices for the nonlinearity transformation h . The transformations

are also called activation functions. In deep learning, the default recommendation is to use the rectified linear unit, or ReLu, defined by the activation function $h(z) = \max(0, z)$ for it tends to show better convergence performance and typically learn much faster [116], it is also used by Elia in their documentation [28].

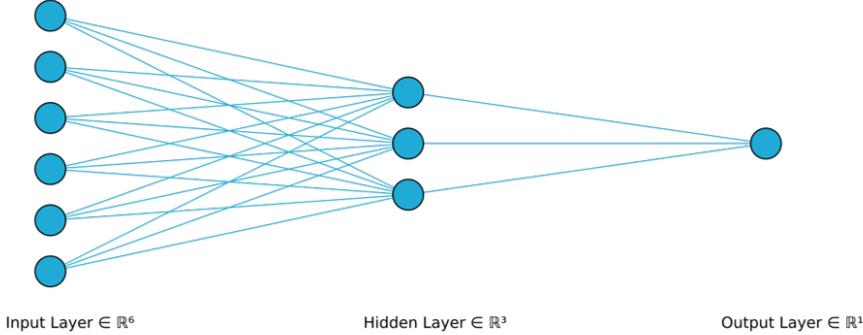


FIGURE 4.2: Feedforward neural network consisting of six input units in the input layer, one hidden layer with 3 neurons and one output unit in the output layer.

The model is trained to minimize the mean squared error (MSE) between predictions and true values, using the Adam optimizer, a widely adopted gradient-based optimization method combining momentum and adaptive learning rates [66]. Hyperparameters such as learning rate, weight decay and hidden layer size are tuned through grid search, which is favored over more complex alternatives, like random search algorithm, and due to the relatively small model and limited training time.

For a comprehensive discussion of the MLP's forward propagation mechanisms, backpropagation algorithm, loss function derivations and optimization variants (SGD, mini-batch SGD, momentum, RMSprop, Adam), the reader is referred to Appendix C.1, which consolidates key theoretical elements adapted from [52, 83].

4.5.3 Model-Agnostic Feature Diagnostics

To complement traditional feature selection methods, model-agnostic interpretability techniques are employed to assess the influence of input features after model training. These diagnostic tools are particularly useful for complex models such as neural networks, where direct interpretation of learned parameters is limited. Two post-hoc methods are considered.

Permutation Feature Importance (PFI) estimates the contribution of each input variable by measuring the drop in predictive performance when its values are randomly permuted. This process breaks the relationship between the feature and the target variable. The resulting increase in prediction error reflects the model's dependence on that feature. PFI is simple to compute and aligns directly with model

4. METHODOLOGY

performance metrics. However, it can underestimate importance when features are strongly correlated, as related variables may retain the predictive information lost through permutation.

SHapley Additive exPlanations (SHAP) [77], by contrast, is based on cooperative game theory and attributes importance by computing the marginal contribution of each feature across all possible subsets of features. It evaluates how much each feature contributes to the difference between the actual prediction and a baseline (expected) prediction. One of SHAP’s key properties is its additive consistency: the sum of all SHAP values for a given prediction equals the model output minus the expected output. This allows SHAP to explain individual predictions as well as global patterns. Unlike PFI, SHAP can fairly distribute importance among correlated features and capture interactions.

These methods are applied post-training, when appropriate, to both the linear regression and MLP models. They do not influence the training process itself. Instead, PFI and SHAP serve to strengthen understanding of model specific feature dependencies and to support feature interpretability.

4.6 Validation Strategy

Accurate model evaluation in time series forecasting requires validation techniques that respect the temporal order of the data. A commonly used approach in machine learning is k -fold cross-validation (CV), where the dataset is randomly divided into k equally-sized subsets or "folds". The model is trained on $k - 1$ folds and tested on the remaining one, rotating through all folds so each is used for validation once. The overall performance is then averaged across all folds, offering a robust estimate of model generalization.

However, this technique assumes that data points are independent, which is not the case for time series. Applying random k -fold cross-validation in a time-dependent context can result in training the model on future data and evaluating it on the past, violating the temporal structure. This leads to information leakage and overly optimistic performance estimates [8, 107].

To address these limitations, this study employs a time-series-aware cross-validation approach based on a rolling-window backtesting procedure. A schematic representation of the training and test splits is shown in Figure 4.3. This technique ensures that models are trained on past data and evaluated on future, unseen observations, closely mimicking real-world forecasting conditions. In each iteration, the model is trained on a moving window of historical data ending at time t , and predictions are generated for the subsequent period. The forecast errors from all iterations are concatenated to construct a continuous out-of-sample prediction series. This setup enables a reliable estimation of overall model performance.

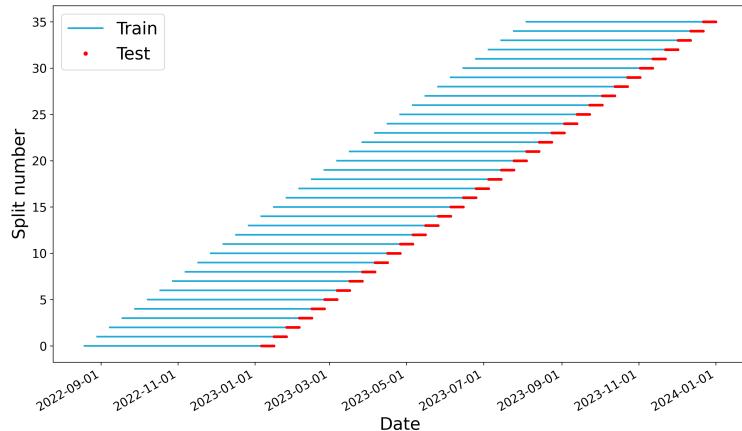


FIGURE 4.3: Rolling-window cross-validation: each line represents a model with training (blue) and test (red) periods.

Training data is restricted to 2022 and 2023, thereby avoiding disruptions related to COVID-19 and preceding the accession to the MARI platform. All models are trained and evaluated under this same temporal window to ensure fair comparability. Forecasts are generated for the full year 2023, allowing a year-long assessment of predictive performance.

The number of rolling splits depends on model complexity and training cost. 36 splits are used for LR, while the more computationally intensive MLP uses 13 splits with longer training windows to ensure convergence.

To mitigate overfitting, feature selection and hyperparameter optimization are nested within each rolling split [8]. Feature selection is performed exclusively on the training portion of each fold, ensuring no information from the test set influences variable selection.

This cross-validation methodology adheres to best practices for time series forecasting [8]. It ensures robust generalization estimates while simulating operational forecasting scenarios.

All modeling steps, including feature selection, hyperparameter tuning and training are fully embedded within each fold of the rolling cross-validation. This setup ensures that the test data remains completely unseen during every stage of the modeling process. By adopting this nested approach, information leakage is prevented, and the evaluation results provide a reliable estimate of true out-of-sample performance.

4.7 Evaluation Metrics

Accurate evaluation of forecasting performance is essential to assess and compare models meaningfully. In this study, both point-based error metrics and statistical tests are used to evaluate predictive accuracy, in line with common practices in the forecasting literature [23, 79]. Although probabilistic metrics are not used, their relevance is briefly discussed in Appendix D.1.

Point Forecast metrics

Forecast performance is assessed using standard point-based metrics, computed over the complete out-of-sample prediction sequence. This approach ensures that evaluation respects the temporal structure of the forecasting task and avoids misleading per-fold averages.

- **Mean Absolute Error (MAE):**

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4.11)$$

MAE is widely used for its intuitive interpretation and equal treatment of all errors [59].

- **Root Mean Square Error (RMSE):**

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4.12)$$

RMSE penalizes larger errors more strongly, making it particularly useful for capturing large deviations [45].

- **Symmetric Mean Absolute Percentage Error (SMAPE):**

$$\text{SMAPE} = \frac{100\%}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} \quad (4.13)$$

SMAPE and related percentage-based metrics are commonly used in load and price forecasting due to their scale independence [79], though care is needed when values approach zero [110].

To address limitations such as sensitivity to scale or outliers, alternatives like the *Mean Absolute Scaled Error (MASE)* have been proposed. MASE is scale-independent and robust to outliers. It avoids undefined values in most practical scenarios.

Transparent and consistent computation of evaluation metrics is crucial, as differences in definitions across studies can lead to misleading comparisons [4].

Chapter 5

Exploratory data analysis

Before applying formal feature selection methods, a preliminary filtering step is essential to manage the large number of potential input features. In time series forecasting, the space of candidate features can grow rapidly due to the use of multiple time lags and resolutions (e.g. minute, quarter-hour, hourly). Without an initial screening, this feature space becomes too large to handle efficiently and may introduce noise or redundancy, which can degrade model performance.

To address this, a model-independent exploratory analysis is conducted to identify variables that are plausibly relevant to the prediction of system imbalance. This phase, serves two purposes: (i) to examine the intrinsic properties of the system imbalance time series, and (ii) to investigate lead-lag relationships between system imbalance and other variables.

The resulting insights provide a basis for the subsequent selection and evaluation of features. In Section 5.1, the analysis begins with a univariate assessment of system imbalance, followed by an exploration of its temporal associations with candidate features. The following sections describe the process of feature categorization, cross-correlation analysis, composite feature construction and preprocesing choices used to construct the final input matrix.

5.1 One-dimensional analysis of system imbalance

The system imbalance (SI) signal forms the core target variable of this study. A better understanding of its statistical properties is essential prior to building any forecasting model. This section presents a descriptive and exploratory analysis of the SI time series, aimed at characterizing its distribution, dynamics and temporal evolution.

This one-dimensional analysis uses validated quarter-hourly data from Elia's transparency platform over the extended period 2020-2023, instead of the 2022-2023 range used elsewhere in the study. Unless otherwise specified, all results are derived from the full dataset.

5. EXPLORATORY DATA ANALYSIS

5.1.1 Properties of system imbalance

The SI signal fluctuates around a mean of -21.35 MW, with a median of -15.41 MW. The 25th and 75th percentiles are -112.3 and $+73.36$ MW, respectively, yielding an interquartile range of 185.66 MW. The standard deviation is 170.07 MW. These statistics confirm the high volatile nature of the signal, which frequently oscillates (96.81%) between -400 and $+400$ MW confirms the highly volatile nature of the signal, which frequently oscillates between -300 and $+300$ MW, but occasionally reaches extreme values. The maximum and minimum observed values are 1135.15 MW and -1419.71 MW, respectively.

These values indicate that the imbalance is more often negative, suggesting demand exceeds supply. The TSO primarily relies on upward regulation. This observation is consistent with frequency data showing a historical mean below 50 Hz.

TABLE 5.1: Key statistics on the Belgian system imbalance (MW).

	2020	2021	2022	2023	2020-23
Mean	-0.14	-29.56	-26.33	-29.41	-21.35
Standard Deviation	155.68	174.07	185.84	161.34	170.07
25th Quantile	-83.65	-126.18	-128.36	-111.56	-112.3
50th Quantile	3.76	-23.01	-21.79	-21.74	-15.42
75th Quantile	86.81	66.37	79.60	58.65	73.36

Figure 5.1 presents the histogram and quantile-quantile plot of the SI signal. The distribution is centered around zero, with slightly more mass in the negative region (skewness: -0.23), reflecting the persistent imbalance in favor of demand. The signal is leptokurtic, exhibiting a sharper peak and fatter tails than a Gaussian distribution. This is confirmed by the quantile-quantile plot in Figure 5.1b, where the empirical quantiles are compared to those of a Gaussian distribution. Deviations from the reference line in both tails illustrate the presence of heavier than normal tails, indicating an overrepresentation of extreme imbalance values, relative to a standard normal distribution.

5.1.2 Stationarity and trend components

Stationarity of the SI signal was assessed using the Augmented Dickey-Fuller (ADF) test [6]. The test strongly rejects the null hypothesis, with an ADF statistic of -33.69 and a p-value of 0, indicating that the SI signal is stationary. This implies that there is no trend or seasonality, as it has a constant mean and variance.

However, this result does not rule out the presence of low-frequency structures in the data. Both seasonal-trend decomposition using LOESS (STL), as well as rolling statistics and long-window smoothing, suggest that certain trend and sea-

5.1. One-dimensional analysis of system imbalance

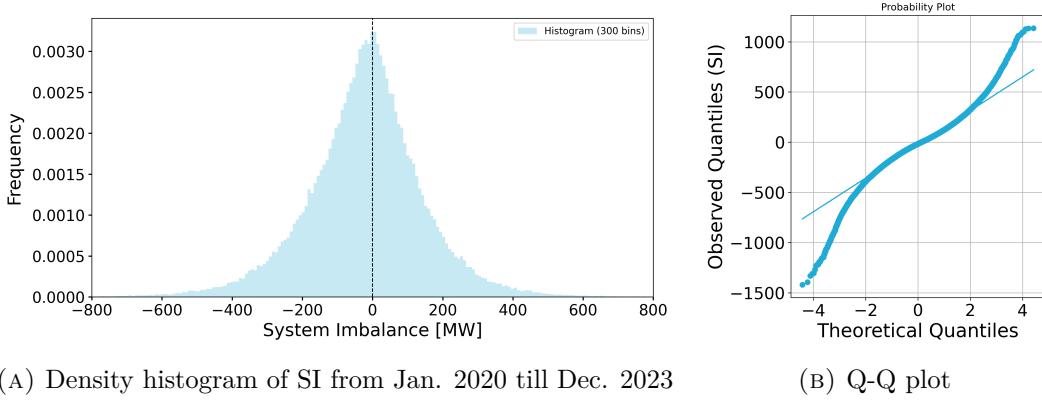


FIGURE 5.1: Statistical evaluation of the system imbalance signal.

sonal components are present. These findings suggest that structural changes occur over time, highlighting the potential value of time-dependent features in forecasting models.

In Appendix D.2, weekly and monthly STL decompositions reveal recurring components and a long-term trend. These evolving characteristics likely reflect underlying shifts in the Belgian energy system, particularly the increasing share of renewable energy sources. Historical datasets from Elia confirm a marked expansion in installed solar and wind capacity, with solar generation showing especially strong growth in peak output levels [107]. Figure 5.2 illustrates a steady increase in the monthly net system imbalance volume (defined as the sum of absolute positive and negative SI values) from January 2020 to July 2023. A clear deviation from this upward trend in July 2022 is potentially linked to the energy crisis: electricity demand declined in the third quarter of 2022 and remained stagnant through the second quarter of 2023 [2]. While this temporal correlation aligns with the observed drop in imbalance volumes, causality can not be definitively established.

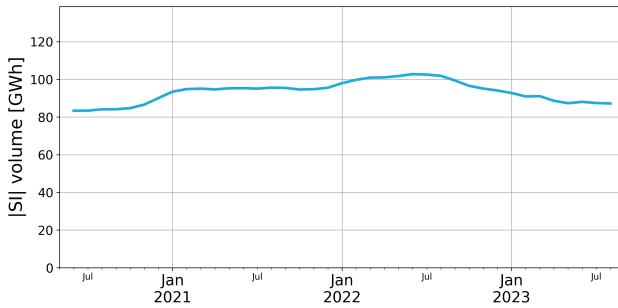


FIGURE 5.2: Evolution of 10-month centered moving average of SI volumes.

In addition to these structural trends, a weak but consistent daily pattern is observed. As shown in Figure 5.3, the distribution of system imbalance varies significantly over

5. EXPLORATORY DATA ANALYSIS

the course of the day. Volatility is lowest between 1:00 and 5:00, while a pronounced negative median imbalance emerges around the evening demand peak (approximately 17:00). Imbalance volumes during high-activity hours (10:00-12:00 and 17:00-18:00) are up to 50% higher than those during the quietest nighttime periods.

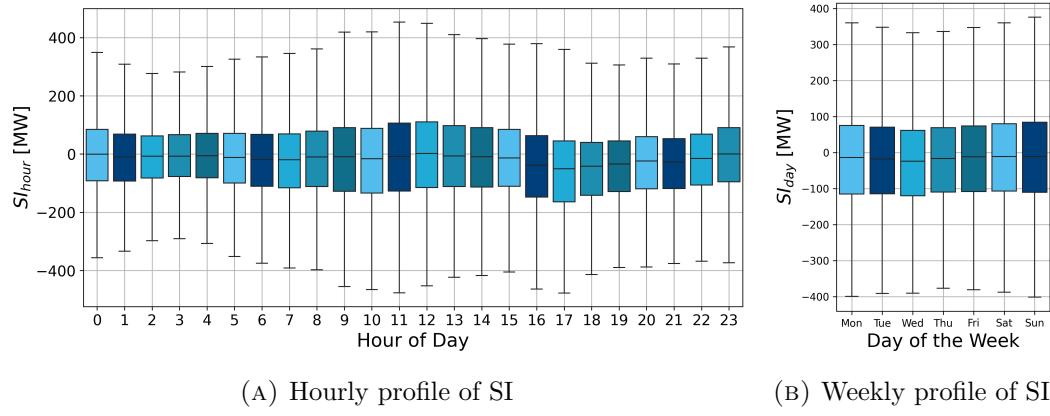


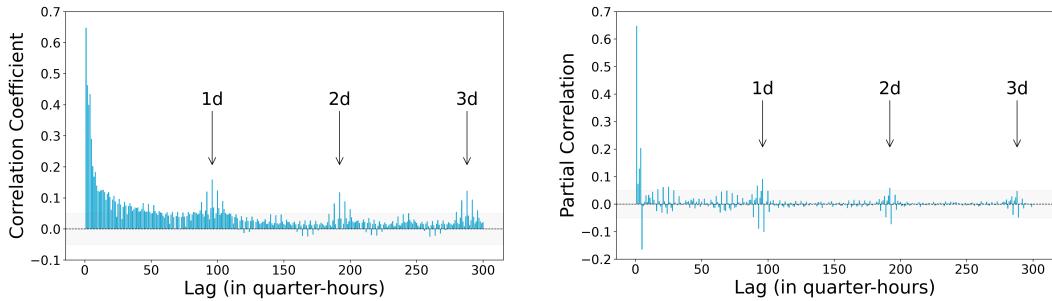
FIGURE 5.3: Distributional profiles of the SI. The box represents the interquartile range (IQR). The black line inside the box indicates the median value.

The presence of weak intraday seasonality suggests potential for improved predictive performance through time-aware features. To capture this periodicity, cyclical time variables, such as hour-of-day, day-of-week and month-of-year were encoded using sine and cosine transformations [78]. These representations facilitate the modeling of temporal patterns by preserving continuity and aiding in the detection of recurring behaviours. Unlike linear encodings, cyclical features ensure that temporal proximity is maintained (e.g. the similarity between hour 23 and hour 0), which is essential for learning time-dependent relationships.

5.1.3 Autocorrelation

Figure 5.4a displays the autocorrelation function (ACF) of the quarter-hourly system imbalance (SI) signal. As expected from a highly stochastic time series, the strongest autocorrelation occurs at the first few lags, particularly within the first eight quarter-hours, before decaying rapidly. However, non-negligible correlations remain at longer lags, most notably at lag-96, lag-192, and lag-288, corresponding to one, two, and three days respectively. This aligns with the weak daily seasonality observed in Figure 5.3a. The additional small peaks around higher multiples of lag-96, thus suggests the potential presence of daily structures in the signal.

Partial autocorrelation function (PACF) identifies which lagged values carry unique predictive power after accounting for intermediate lags, as shown in Figure 5.4b. In this plot, the zero-lag component is excluded, as it is equal to one. The first five quarter-hour lags exhibit the highest partial autocorrelations, indicating strong



(A) Autocorrelation of SI (quarter-hourly), the grey band indicates the 95% confidence interval

(B) Partial autocorrelation of SI (quarter-hourly, Lag 0 excluded), the grey band indicates the 95% confidence interval

FIGURE 5.4: Autocorrelation and partial autocorrelation of the system imbalance signal.

short-term memory in the system imbalance. Further PACF spikes are observed near lag-96 (one day), reinforcing the presence of a daily autoregressive component. These values suggest that both short-term and daily lagged features should be considered when constructing forecasting models.

From the raw SI signal, a range of lagged and transformed features were engineered to improve predictive performance. Autoregressive lag features (e.g. SI from 15 minutes to 24 hours ago) were included to capture the observed persistence and diurnal cycles. The selection of these lags was guided by both ACF and PACF insights, ensuring that each added lag contributes distinct information rather than redundant noise.

5.1.4 Mutual Information

To evaluate whether nonlinear dependencies contribute additional insight beyond linear relationships, Mutual Information (MI) was computed alongside the PACF. Figure 5.5 presents a side-by-side comparison of both measures over the first 100 quarter-hourly lags. Both metrics highlight the strongest dependencies within the first few lags and around lag 96, corresponding to the daily cycle.

While MI captures total dependence, including nonlinear effects, it did not reveal any additional informative lags beyond those identified by PACF. As such, MI did not influence feature selection in this case. Nonetheless, MI may prove useful in settings where nonlinear effects are expected or when interactions between features may not be linearly separable.

5.2 Cross-correlation

To move beyond autoregressive components and identify additional relevant features, a cross-correlation analysis was conducted between system imbalance (SI) and se-

5. EXPLORATORY DATA ANALYSIS

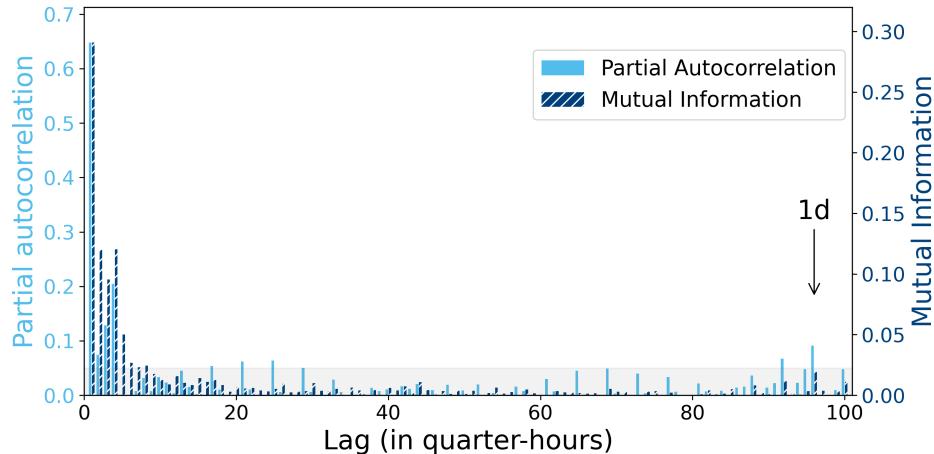


FIGURE 5.5: Comparison of Partial Autocorrelation and Mutual Information for lagged SI features. PACF values (full bars) are shown on the left axis, MI values (hatched bars) on the right.

lected exogenous time series.

Thus, a broad set of exogenous variables was engineered to capture potential drivers of imbalance. These include grid-level measurements (e.g. actual and forecasted demand and generation), temporal indicators (e.g. hour-of-day, day-of-week, month-of-year), and market-related signals such as interconnector flows and reserve activation volumes. Temporal features were encoded using cyclic transformations (e.g. sine and cosine of hour-of-day) to preserve the periodic structure of time. In addition, new features were derived through lagging, differencing and combining base variables to enrich the dataset.

This analysis served two primary objectives: (i) to uncover lead-lag dependencies that consistently precede variations in SI, and (ii) to support data-driven feature selection by quantifying the predictive value of external signals. For each exogenous variable X , the cross-correlation function (CCF) with SI was computed over a lag window $k \in [-n, +n]$, where positive values of k indicate that X leads SI by k time steps. Lagged versions of variables exhibiting significant correlation at positive lags were retained as candidate features.

As shown in Figure 5.6, several features exhibit distinct correlation patterns. The net position (Figure 5.6a) and wind generation (Figure 5.6d) show a clear downward trend in correlation as the lag increases, indicating that recent values carry more predictive information. In contrast, solar generation (Figure 5.6c) and the positive imbalance price (Figure 5.6b) display more cyclical correlation patterns, reflecting the diurnal and market-driven nature of these variables. For solar generation in particular, the correlation oscillates between positive and negative values, consistent

5.2. Cross-correlation

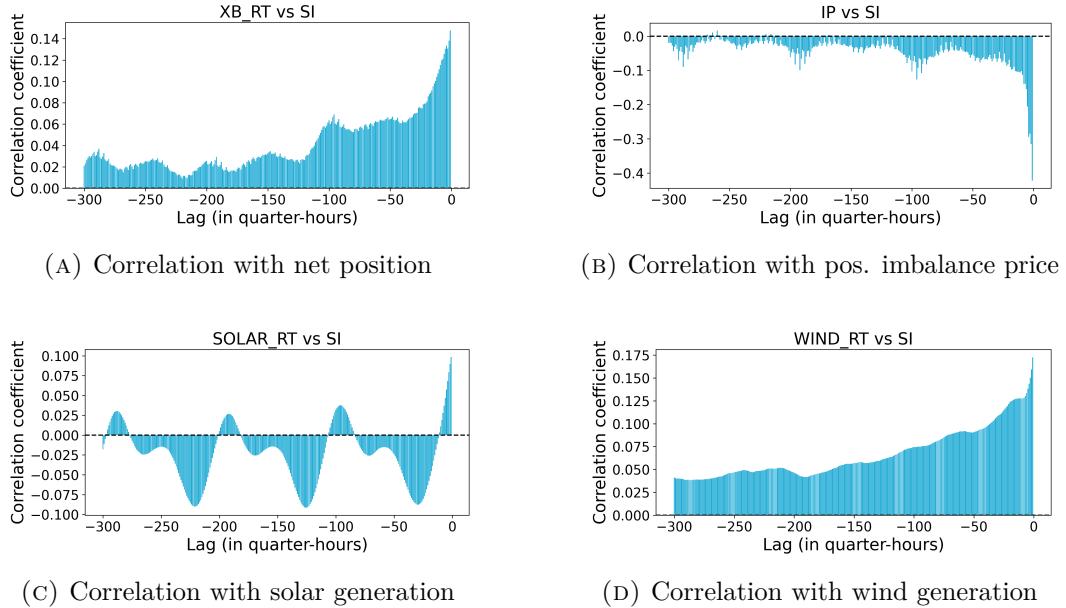


FIGURE 5.6: Cross-correlations of 300 to 1 lagged input features with system imbalance (SI). Each subplot shows the correlation structure for a specific feature category.

with its strong dependency on time-of-day and weather conditions. Around qh-36 (approx. 9 hours earlier), solar generation exhibits a strong negative correlation with SI. This likely reflects inverse daily patterns, SI typically peaks around midday, as seen in Figure 5.3a, while solar generation is absent at night. Such diurnal phase-shifted behaviour explains the observed lagged anti-correlation. Seasonal effects could be subject of further research, as this is limited in this paper.

Since the first lagged instance of a feature often yields the strongest correlation with system imbalance (SI), comparing these peak values provides a preliminary indication of feature relevance. This is shown in Appendix D.3 for both quarter-hourly and minute-resolution data. Appendix D.4 complements this by ranking the maximum cross-correlation of each exogenous variable across multiple lags. While this offer useful initial insights, examining a broader lag range provides a more complete view of temporal relationships with SI.

To complement the cross-correlation analysis, a correlation heatmap (Figure 5.7) was used to identify highly collinear features. While strongly correlated variables were selectively removed to reduce redundancy, as excessive correlation can degrade model performance [113]. Some multicollinearity was retained to allow feature selection methods to perform their intended role in assessing feature relevance.

Another example involves solar and wind features: day-ahead forecasts and actual generation data are highly correlated. When real-time generation data is unavailable,

5. EXPLORATORY DATA ANALYSIS

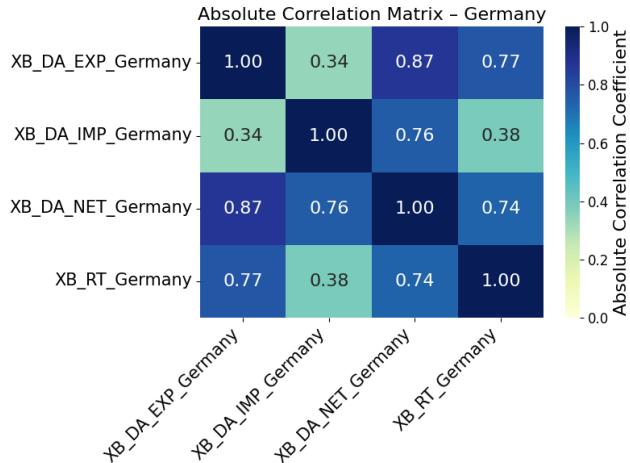


FIGURE 5.7: Correlation matrix of Germany in absolute values.

only lead values (i.e. forecasts) should be used to avoid redundancy and ensure the input reflects information available at prediction time.

In conclusion, cross-correlation analysis provided insight into dynamic relationships between SI and external drivers, by building a temporally coherent and information-rich feature matrix, to improve both model performance and interpretability.

5.2.1 Resulting input features

The initial dataset included over hundreds time-lagged and lead features derived from autoregressive and exogenous variables. In the exploratory phase, lags up to -1000 were considered, while cross-correlations were evaluated within a range of -300 to +300, depending on data availability. To reduce dimensionality and multicollinearity, features were filtered based on their correlation with the system imbalance (SI) and with one another. A correlation threshold of 0.1 was applied, with the top three features below this threshold retained based on their absolute correlation with the target. This procedure resulted in an input matrix of approximately 350 features, which was used for the linear regression and random forest models.

For the MLP model, a smaller subset was selected based on cross-correlation analysis and the most relevant features identified in the linear model.

The complete list of the 350 selected features is provided in Appendix [A.2](#).

Chapter 6

Results and discussion

This chapter presents the forecasting results and evaluates the effectiveness of various feature selection (FS) methods, as outlined in Chapter 4.

The chapter is structured as follows. Section 6.1 introduces the forecasting target and outlines the model configurations. Section 6.2 describes the computational setup and parallelization strategy used to manage the complexity of FS experiments. Section 6.3 details which FS methods were applied to which models, along with the rationale for those decisions. Section 6.4 presents the forecasting performance of all configurations, including comparisons with two baseline models. Section 6.5 analyzes the similarity between selected feature subsets across methods. Section 6.6 explores the interpretability and stability of selected features. Finally, Section 6.7 presents several extensions and concludes the chapter with key findings and directions for future work.

The analysis addresses three central research questions. First, the evaluation seeks to determine which feature selection method offers the best trade-off between predictive accuracy and computational efficiency. Second, it investigates the consistency of selected feature subsets. Third, it examines whether certain lagged features are repeatedly selected.

6.1 Forecasting target and model setup

The forecasting target is the quarter-hourly system imbalance value SI_{qh+1} , predicted from minute 3 of the current quarter-hour. This setup aligns with operational requirements for real-time balancing in the Belgian electricity market.

Two models are employed: a Linear Regression (LR) model implemented in `scikit-learn`, and a Multi-Layer Perceptron (MLP) implemented using PyTorch and `scikit-learn`. Both models are evaluated using three standard metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Symmetric Mean Absolute Percentage Error (SMAPE), with lower values indicating better performance.

The LR model uses 36 rolling cross-validation splits with a 20-week training and

6. RESULTS AND DISCUSSION

10-day test window. The MLP model, requiring more data to generalize, is trained using 13 splits with 50-week training and 28-day test windows, allowing for year-round coverage. The MLP consists of a single hidden layer with 32 neurons, trained over 500 epochs with a learning rate of 0.01. This configuration was selected based on grid search results, balancing training time and performance. Deeper architectures showed only marginal gains (1%) at significantly higher computational cost.

TABLE 6.1: Model training setup and hyperparameters

Parameter	LR	MLP
CV Splits	36	13
Training Window	20 weeks	50 weeks
Testing Window	10 days	28 days
Total Predicted Days	360	364
Hidden Layers	–	1 layer / 32 neurons
Evaluation metrics		MAE, RMSE, SMAPE
Framework	scikit-learn	PyTorch & scikit-learn

6.2 Computational Considerations

Due to the computational cost of wrapper-based FS methods, experiments were parallelized using the resources of the Flemish Supercomputer Center (VSC). Feature subset evaluations and cross-validation folds were distributed across multiple cores using `n_jobs=-1` in `scikit-learn` and `mlxtend` [91]. Performance is measured in terms of wall-clock time and total number of model evaluations on a 64-core machine. All methods share the same CV setup and fixed hyperparameters to ensure comparability.

Acknowledgment - VSC *The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government.*

6.3 Implementation of FS methods

To evaluate the effectiveness of feature selection (FS) in a forecasting method, nine methods were considered:

- One filter-based method (correlation ranking),
- Seven wrapper-based methods: Recursive Feature Elimination (RFE), Sequential Forward Selection (SFS), Sequential Backward Selection (SBS), Sequential Forward Floating Selection (SFFS), Sequential Backward Floating Selection (SBFS), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO),
- Two embedded methods: LASSO regression and Random Forest.

All ten FS methods were first applied to the Linear Regression (LR) model. Due to its lower computational complexity, LR allowed for systematic testing across a large number of feature subsets and methods. The large number of wrapper methods was motivated by their demonstrated effectiveness in prior literature and their ability to explore different search strategies, producing diverse subsets that enhance interpretability and robustness in the analysis.

Of these, four methods, RFE, SFS, SBS and LASSO, were also applied to the Multi-Layer Perceptron (MLP) model. The remaining six (Filter, SFFS, SBFS, GA, PSO and RF) were excluded from MLP due to their higher computational demands and the need for extensive hyperparameter tuning, which made them impractical to scale to more complex neural architectures.

The initial feature sets used for Linear Regression (LR) and the Multi-Layer Perceptron (MLP) differ significantly. Due to LR's low computational cost, feature selection (FS) could be performed on a large set of 355 features, allowing for extensive experimentation. In contrast, MLP required a more compact input set, informed by preliminary data analysis and insights gained from FS results on the LR model.

6.4 Results

To benchmark performance, two baselines were included. *Elia Linear Model*, reconstructed using publicly available information. It is an LR model based on 68 features derived from 11 base variables, using multiple lags and leads (the last four and/or next four values). However, this restricted scope likely omits relevant information [27]. *"All Features" Model*, is trained on the full initial feature set of 355 features identified during exploratory analysis.

Performance metrics include MAE, RMSE (both in MW) and SMAPE (percentage). Lower values indicate better performance. Results are summarized in Table 6.2.

6.4.1 Interpretation of results

Baseline comparisons show that using the Full Feature Set, derived from cross-correlation metrics, already improves performance over the reconstructed Elia model. The LR model with all 355 features achieves a 3.4% reduction in MAE, which is expected given that the Elia model relies on a limited, manually selected set of features that omits relevant temporal information.

Further performance improvements were obtained by applying feature selection (FS), except with the Filter method. While both the Full feature set and Filter subset are based on cross-correlation, the former uses a broader inclusion threshold,

6. RESULTS AND DISCUSSION

TABLE 6.2: Forecast performance with different feature selection methods (Linear Regression vs. MLP). Lower MAE/RMSE/SMAPE indicate better accuracy.

FS	Model	No. of Features	MAE [MW] ↓	RMSE [MW] ↓	SMAPE [%] ↓
ELIA [27]	LR	44	97.33	130.05	117.46
All features		355	94.02	125.81	111.75
Filter		99	99.89	134.08	120.42
RFE		86	91.80	122.46	110.46
SFS		113	92.03	125.48	110.78
SBS		113	91.80	122.49	110.46
SBFS		78	91.73	122.49	110.71
SFFS		120	91.77	122.39	110.88
GA		67	92.98	124.83	111.97
PSO		59	93.95	125.3	113.4
RF	MLP	96	92.58	123.22	111.48
LASSO		156	93.25	123.48	111.44
RFE		23	97.06	130.46	115.28
SFS		44	95.77	129.45	115.36
SBS		38	95.43	129.60	115.52
LASSO		26	95.08	128.05	114.73
RF	RF	96	96.46	129.87	118.23

while the latter, restricts inputs to top-ranked features. This stricter filtering reduced performance, likely by excluding moderately correlated yet informative features and increasing noise sensitivity due to multicollinearity.

Among all FS techniques, Sequential Backward Floating Selection (SBFS) yielded the lowest MAE (91.73 MW), with only 78 features, indicating its effectiveness in identifying an optimized feature subset through a greedy search strategy with the flexibility to revisit prior selections. Meanwhile, Sequential Backward Selection (SBS) yielded the lowest SMAPE (110.46%), also reflecting robustness in capturing relative error. Across the board, sequential methods such as RFE, SBS, SFS, SFFS, and SBFS demonstrated consistently strong performance. This trend highlights the advantage of systematic search strategies that iteratively evaluate feature contributions.

In contrast, the metaheuristic approaches, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), selected significantly smaller feature subsets and yielded higher SMAPE values. While scalable and efficient, their performance was less stable, likely due to sensitivity to initialization and hyperparameters. LASSO and Random Forest (RF) achieved moderate results but required further tuning. LASSO in particular may have over-penalized correlated yet valuable predictors.

Regarding MLP models, performance was generally lower than that of LR, even when using feature subsets derived from earlier FS methods. The best MLP, based on LASSO-selected features, achieved a SMAPE of 114.73% and an RMSE of 128.05

6.4. Results

MW. Although comparable to LR, the consistently higher errors suggest that the reduced feature sets may have constrained the MLP's learning capacity. Sequential selection methods are likely more effective when selecting from a broad feature pool rather than from already pre-filtered subsets. Unlike LR, the MLP was only evaluated on pre-selected inputs, potentially omitting valuable information. More complex architectures or further tuning may improve performance but lie beyond the scope of this study.

Lastly, differences between FS methods were relatively small, typically under 2 MW in MAE and 1% in SMAPE, suggesting that model interpretability and computational cost may outweigh minor accuracy gains. The consistently high SMAPE values (over 100%) reflect the volatility of system imbalance, characterized by large positive and negative deviations. These results also reinforce the importance of reducing feature redundancy, as the "no FS" model consistently underperformed those using selected subsets, despite having access to the full information set.

6.4.2 Computational cost and efficiency

The impact of feature selection on model performance must be balanced against its computational cost. As shown in Figure 6.1, increasing the number of features generally improves forecasting performance, particularly in early stages. However, this improvement plateaus after a certain point, where the marginal benefit of adding more features diminishes. This trend is particularly important for MLP models, where each additional feature can lead to substantially increased training time due to their higher computational complexity. The figure presents the negative mean absolute error (MAE), so higher values indicate better performance. Note that the x-axis represents the number of selected features, not time. In SFS, the process begins with an empty set and adds features, whereas in SBS, the process starts with the full feature set and removes features sequentially.

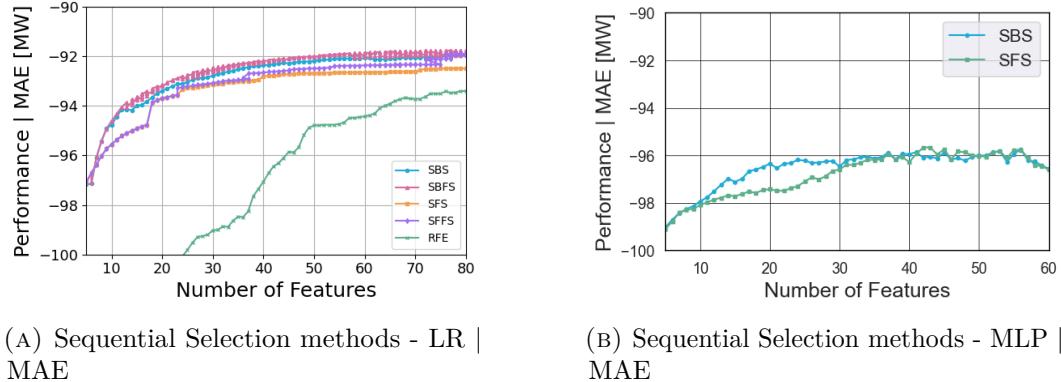


FIGURE 6.1: MAE score for sequential selection methods in function of number of features.

6. RESULTS AND DISCUSSION

While wrapper-based methods often yield high-performing feature subsets, they come at a high computational cost. Figure 6.2 visualizes the trade-off between performance gains and computation time for SBS and SBFS, applied to LR. For LR (Figure 6.2a), performance improvements diminish as more features are removed. Beyond a certain point, further pruning results in negligible gains, while computation time increases significantly. In the MLP case (Figure 6.2c), where SBS was applied to a pre-filtered feature set derived from LR, the improvements are less stable. This can be attributed to the already high quality of the retained features, reducing the effectiveness of further selection. This observation is supported by a comparison with SBS on MLP applied to a mixed set of high- and low-potential features, where the performance trend appears more stable

Figure 6.2 shows the marginal gain in cross-validation score relative to computation time for SBS and SBFS applied to both LR and MLP models.

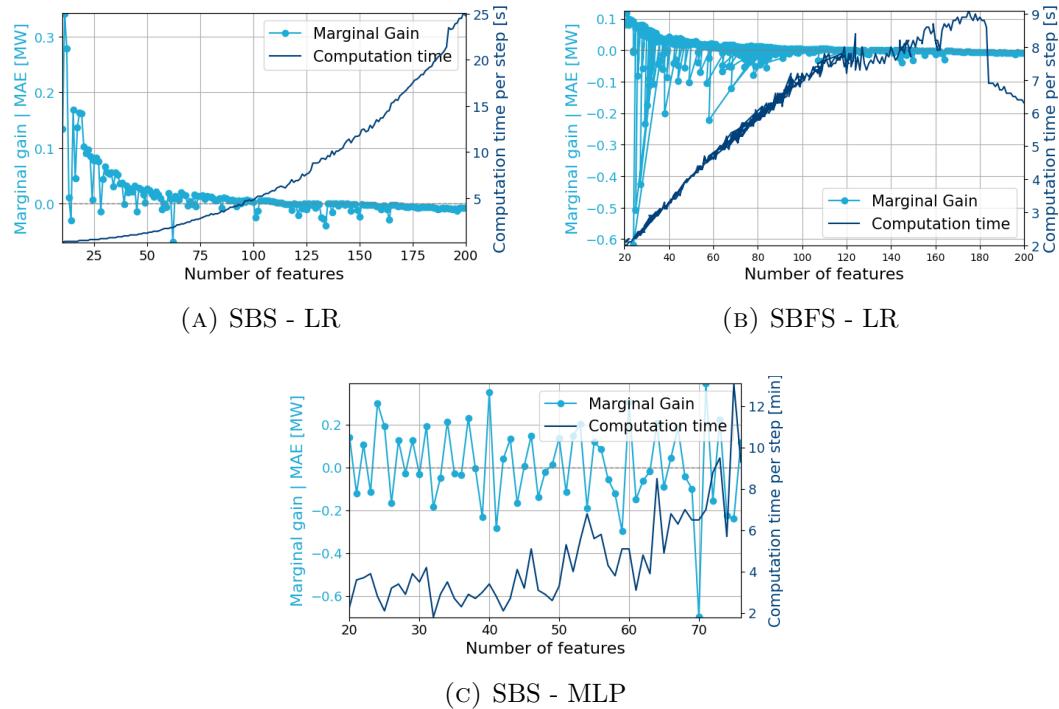


FIGURE 6.2: Marginal gain and computation time per step as a function of the number of selected features. The top row shows Sequential Backward Selection (SBS) and Sequential Backward Floating Selection, both for LR. The bottom row shows SBS for MLP.

This type of analysis aligns with recent efforts in cost-aware machine learning, where accuracy improvements are weighed against computational resources [43, 73, 90].

Table 6.3 summarizes the computational characteristics of each feature selection method, including the number of analyzed features, estimated model fits across

cross-validation folds, and wall-clock execution time. All experiments were executed in parallel on a 64-core high-performance computing (HPC) node, although actual CPU utilization was not monitored. It should be noted that the mlxtend library used for sequential selection methods does not support GPU acceleration. Consequently, MLP models were trained on CPU, increasing computation time.

The number of features analyzed varies due to both practical constraints and methodological choices. Some methods (e.g. GA, PSO) were applied to reduced feature sets to avoid the added complexity and cost of hyperparameter tuning on larger inputs. In contrast, methods such as RFE, SBS, and SFFS were applied to larger sets (up to 250 or 355 features), made feasible by the low cost of training LR models. For MLP, all methods were limited to 62 features to keep cross-validated training times tractable.

TABLE 6.3: Computational cost of feature selection methods. Model fits include training across cross-validation folds. HPC: 64-core parallel execution.

FS	Nr. features analyzed	Model	Model fits	Wall time [h:m]
RFE	250		9000	3m
SFS / SBS	150 / 250		793,872 / 1,111,536	23m / 51m
SFFS /SBFS	150 / 250		1,984,680 / 2,778,840	3h03 /3h06
GA	210	LR	10,332	20m
PSO	114		46,820	21m
RF	250		36	4m
LASSO	250		36	11m
RFE	62		806	40m
SFS / SBS	62	MLP	47,580	3h31 / 4h13
LASSO	62		13	35m

The number of model fits reflects the total training operations across cross-validation folds. For LR, 91 folds were used; for MLP, 36. For SBS, the number of model fits is given by:

$$\text{Model fits} = \text{folds} \times \sum_{i=1}^{n-1} i = \text{folds} \times \frac{n(n-1)}{2} \quad (6.1)$$

This quadratic growth in Table 6.3 illustrates the scalability challenge of wrapper-based methods. For instance, SBFS applied to 250 features required nearly 2.8 million model fits and over three hours of wall-clock time, even with full parallelization. In contrast, simpler methods such as filter selection or LASSO completed in under 15 minutes with minimal model evaluations. The metaheuristic methods (GA, PSO) also offered a reasonable compromise, producing usable results within 20-21 minutes with moderate evaluation counts.

Overall, sequential wrapper methods such as SFS, SBS, and SFFS offer superior performance in terms of feature quality and forecast accuracy but impose substantial computational overhead. Their application is only practical on large feature sets

6. RESULTS AND DISCUSSION

when high-performance computing (HPC) resources are available. In contrast, RFE remains lightweight due to fast model fits and modest parallelization demands. For routine or resource-constrained settings, faster alternatives such as filter methods or hybrid approaches may offer a more practical accuracy-efficiency balance.

6.4.3 Performance of cross-validation

This section evaluates the temporal generalization of the forecasting models using a custom time-aware cross-validation framework, introduced in Section 4.6. Based on `TimeSeriesSplit`, this approach ensures each training fold uses only past data relative to the test set, preventing data leakage and simulating real-world forecasting conditions.

The evaluation, performed on the best-performing feature subsets, highlights key differences between training and test performance. As shown in Appendix D.5, test scores are consistently more variable, reflecting the greater uncertainty associated with unseen data. Among the metrics, SMAPE exhibits the narrowest spread, consistent with its robustness to outliers in relative terms.

More specifically, the right panel in Appendix D.5 shows SMAPE scores across every sixth fold, illustrating that model performance fluctuates over time. The SMAPE varies by approximately 15% between the best and worst folds, indicating that generalization is affected by temporal shifts in the data. This variation suggests the presence of nonstationarities or structural changes, such as evolving seasonal patterns, are not fully captured by the models.

While cyclical features (e.g. sine and cosine of `MonthOfYear`) try to capture seasonality, the model does not adapt to changes in the data over time. These results suggest that time-dependent strategies, such as periodic retraining, seasonal models or larger datasets, could improve performance.

6.5 Similarity between feature selection methods

While error metrics provide insight into the predictive performance of feature selection (FS) methods, they do not reveal how similar the selected feature subsets are to one another. In practice, different FS techniques may yield comparable results while selecting distinct subsets of features. To assess the degree of agreement among methods, the pairwise similarity between selected feature subsets is examined.

A natural way to quantify this is through the **Jaccard similarity index**, which measures the proportion of shared features between two sets. For subsets A and B , the index is defined as [14]:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (6.2)$$

6.5. Similarity between feature selection methods

The Jaccard index ranges from 0 (no shared features) to 1 (identical sets), making it particularly well-suited to comparing methods with different subset sizes, unlike raw overlap counts.

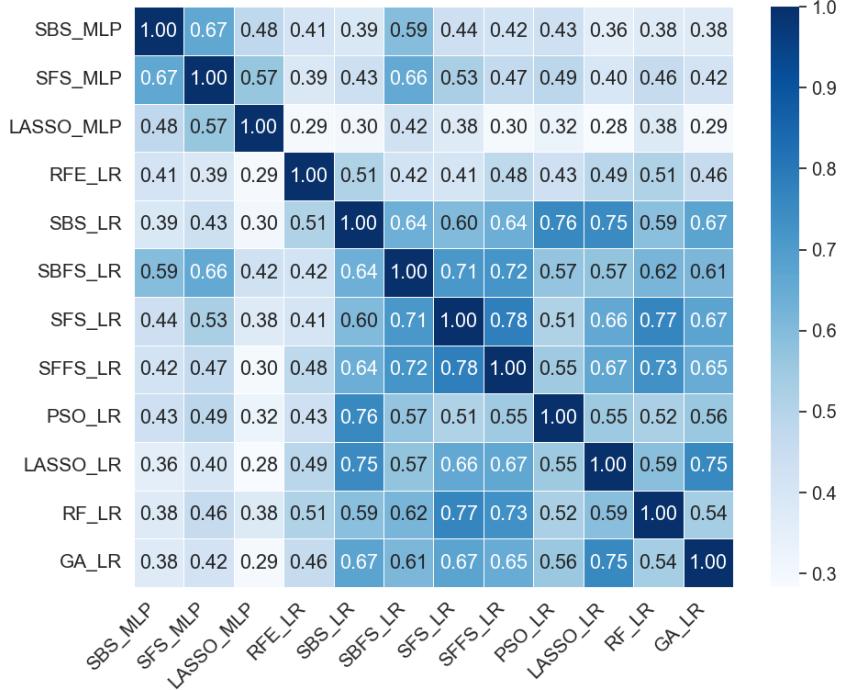


FIGURE 6.3: Jaccard similarity matrix comparing feature subsets selected by different FS methods. Higher values indicate greater agreement between selected subsets, normalized for subset size.

Figure 6.3 shows that FS methods used with linear regression (suffix `_LP`) exhibit greater mutual similarity than those used with MLP, indicating more stable feature selection for simpler models. Greedy wrapper methods like SFS and SBS display strong internal agreement, while metaheuristic approaches (GA, PSO) yield more varied subsets. This complements the absolute overlap matrix in Appendix D.4, which does not normalize for subset size and can overstate similarity for methods selecting many features.

Ranked subset comparisons, where features are ordered by importance and truncated (e.g., top 50) [14], were also explored but offered limited insight, likely due to inconsistent importance weights driven by multicollinearity.

Among LR-based methods, SBFS showed the strongest accuracy and was most similar to SFS (0.71) and RF (0.62), both of which require less computation, making them promising alternatives. This suggests that in practice, simpler or faster methods that align with more complex approaches can offer an effective trade-off between accuracy and efficiency in FS.

6.6 Feature interpretability

A central aim of this thesis is to determine which features contribute most effectively to forecasting accuracy across models and selection methods. Beyond measuring raw performance, it is equally important to examine which features are selected, how consistently they appear, and whether their selection aligns with domain expectations.

To address this, a heatmap in Figure 6.4 was constructed to visualize the selection frequency of each feature-lag pair across all feature selection (FS) methods and models. Each cell reflects a weighted frequency score, where selection counts are adjusted by the model’s performance. This representation captures not only how often a feature-lag was selected, but also how much it contributed to the final accuracy. The x-axis shows the lag index, ordered by total representation (e.g. lag -1 appears most frequently), while the y-axis lists the features.

The resulting pattern reveals a number of robust and interpretable insights. First and foremost, lagged system imbalance (SI) emerged as the most influential predictor. Specifically, lag $-1, -3$ on quarter-hour granularity, and $-2, -3, -4$ on minutewise granularity was selected across virtually all models and methods, confirming the system’s strong autoregressive behaviour. This aligns with the expectations that previous imbalance values are strong signals of short-term dynamics.

Beyond autoregressive terms, the most consistently selected features were related to renewable energy generation, particularly the actual and forecasted values for wind and solar power. This highlights a strong link between rising system imbalance and the growing share of renewable generation in the grid.

Load-related features, including real-time values, day-ahead forecasts, and forecast errors also proved to be informative, as did the ID net position of Belgium. These features capture demand-side variability and are linked to the likelihood of reserve activation. Additionally, cross-border day-ahead export nominations (from France, the Netherlands, and Germany) were selected at short lags. This is further investigated in Section 6.6.1. Their importance highlights the interdependence of the Belgian grid with neighbouring systems and indicates that export intentions can serve as leading indicators for imbalance, particularly at lead time 1, which coincides with SI_{qh+1} . This observation suggests that feature relevance is forecast-horizon dependent and that selections may shift with longer horizons.

As this study focused exclusively on one-step-ahead forecasting (SI_{qh+1}), a natural extension would be to investigate longer horizons (SI_{qh+2}, SI_{qh+3} , or SI_{qh+4}). Such forecasts may be especially valuable for flexibility providers like battery storage operators, who require lead time to schedule resources. These longer-horizon tasks may benefit from different, potentially less autoregressive, feature subsets.

Interestingly, time-related features are largely absent from the heatmap, despite being

6.6. Feature interpretability

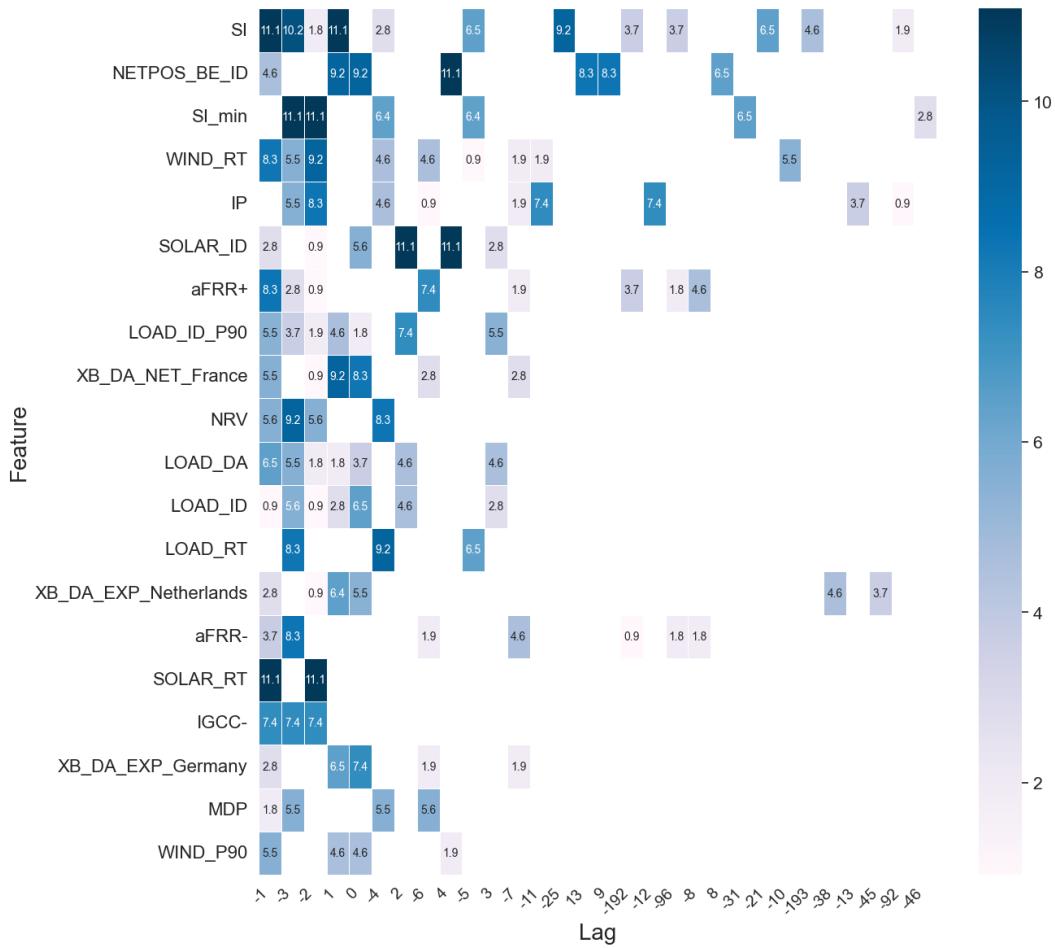


FIGURE 6.4: Weighted selection frequency of the top 20 feature-lag pairs across all models and feature selection methods.

present in the original feature space (as discussed in Chapter 5). Only `DayOfWeek_sin` appears in the full selection table (Section A.3). This may suggest either limited predictive value or underutilization by the models.

The analysis also reveals model-specific feature selection behavior. Linear models like LR tend to prioritize features with strong individual correlations to the target, such as NRV or `SI_-1`. In contrast, MLP models often select features that may not be highly predictive on their own but contribute value in combination with others. This suggests that neural networks can uncover more complex patterns in the data when given the opportunity.

It is worth noting that the difference in initial feature sets between LR and MLP complicates direct comparisons. Due to the MLP’s higher computational demands, only a prefiltered subset of features could be used as a starting point. As a result,

6. RESULTS AND DISCUSSION

potentially valuable weak predictors were excluded.

In conclusion, feature interpretability offers not only transparency into model behaviour but also practical guidance for future forecasting applications. High-performing models rely on a compact, interpretable, and temporally coherent subset of features. Among all methods, wrapper-based techniques like SFS and SFSS proved especially effective in identifying these subsets. When both performance and explainability are priorities, as they are in operational grid forecasting, these methods offer a compelling balance between accuracy and insight.

6.6.1 Cross-border nominations

System imbalance forecasting studies [28, 67], have shown that incorporating cross-border variables can improve prediction accuracy. In Belgium, the relationship between import and export volumes and the corresponding system imbalance signals show a clear pattern of cross-correlation at the country level. This subsection investigates whether such real-life dynamics are reflected in machine learning models.

To examine whether such dynamics are captured by machine learning models, both SHAP values and permutation importance scores were analyzed. These results, presented in Appendix D.6, indicate that while other features exert greater influence, cross-border nominations, especially those involving France and the Netherlands, contribute substantially to model performance.

Notably, some features display asymmetric SHAP distributions. For instance, the day-ahead import nomination from France at $qh + 1$ displays a long right-hand tail, suggesting a causal relationship between the import nomination and the system imbalance at $qh + 1$. Conversely, a long leftward tail associated with the current import nomination is seen, possibly indicating an inverse relationship with the future SI. However, it is crucial to distinguish between correlation and causation in these interpretations.

The permutation importances largely align with the SHAP results, albeit with slight reordering of feature ranks and a greater emphasis on export-related values. But SHAP accounts more for feature correlations, whereas permutation importance does not.

Figure 6.5 complements these findings by comparing the feature importance of cross-border nominations with actual electricity transmission volumes. France and the Netherlands emerge as Belgium’s primary trading partners in both 2022 and 2023 in terms of physical exchange. Interestingly, although Dutch volumes exceed those of France, both SHAP and permutation analyses indicate that French nominations carry greater predictive value for system imbalance forecasting.

This suggests that the informativeness of cross-border features is not solely de-

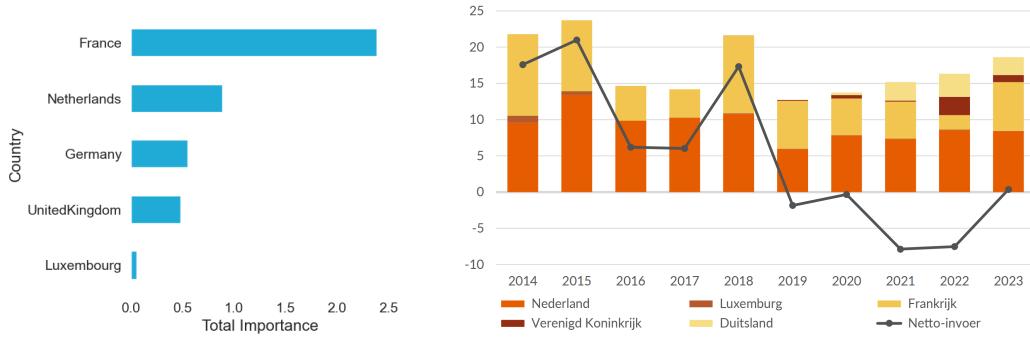


FIGURE 6.5: Permutation importance of cross-border features (left) and observed electricity transmission volumes over time [TWh] (right) [42].

terminated by physical volume, but also by their tendency to coincide with periods of system stress or imbalance. Notably, Belgium has four interconnections with France compared to two with the Netherlands, which may partly explain the stronger predictive signal. By contrast, nominations involving Germany, the UK, and Luxembourg play a more limited role, both in volume and model relevance.

Future work could investigate whether participation in cross-border balancing platforms such as PICASSO (aFRR) and MARI (mFRR) increases the predictive value of nominations from participating countries.

6.7 Expansion

This section explores two extensions to FS. The application of Principal Component Analysis (PCA) for dimensionality reduction and the use of imbalance price as alternative forecasting target.

6.7.1 Principal Component Analysis (PCA)

To further address multicollinearity and reduce dimensionality beyond direct feature selection, Principal Component Analysis (PCA) was applied in a hybrid fashion. After selecting a core subset of important features using feature selection (SBFS, 78 features), the remaining less-influential or highly correlated features were ranked based on their linear regression coefficients and transformed into a smaller number of principal components. These components were then combined with the retained original features and used as input to the forecasting models.

This approach preserved the interpretability of key predictors while mitigating redundancy and noise in the lower-ranked feature subset. The figure in Appendix D.7 illustrates how the validation error (MAE) and runtime evolved as a function of (i) the number of features included in the PCA transformation and (ii) the number of

6. RESULTS AND DISCUSSION

principal components retained.

The results indicate that applying PCA to a moderate portion of the least important features (e.g. 30%, or 24 features) and retaining a limited number of components (corresponding to a final total of 70%, or 54 features) strikes a favorable trade-off: model performance decreased marginally (-1% MAE), while runtime was significantly reduced (-40%). Beyond this configuration, further runtime benefits diminished and more aggressive dimensionality reduction began to degrade accuracy in linear regression.

While PCA does obscure the physical meaning of inputs, it can be a valuable supporting tool, particularly when the feature set is large and noisy. Further investigation could explore PCA-based dimensionality reduction in combination with other neural architectures.

6.7.2 System imbalance price comparison

An interesting extension of this work is the direct forecasting of the imbalance price for the next quarter-hour ($qh+1$). This is supported by Elia's recent trial publication of imbalance prices, which, although not publicly available, signals the potential importance of such forecasts.

FS using SBS and SBFS for imbalance price revealed some overlap with SI, but also highlighted features more tied to market stress, such as `mFRR+` and minute-level imbalance price (`IP_min`), which were not selected for SI. This indicates that certain features are uniquely relevant to economic rather than physical forecasting tasks. The resulting similarities and differences are shown in the Jaccard similarity matrix (Appendix D.8).

A further direction for future work is to vary the forecast issuance time within the quarter-hour; issuing predictions later (e.g. minute 7 instead of 3) could improve accuracy by leveraging more recent data and allow exploration of how feature relevance evolves in real time.

6.8 Final Feature Set

The final feature sets, determined by evaluating various combinations, are presented in Appendix A.4. They consist of 78 features for the Linear Regression (LR) model and 26 features for the MLP model.

Chapter 7

Conclusion

This thesis set out to explore how feature selection techniques can improve short-term forecasting of system imbalance in the Belgian electricity grid. To this end, lagged versions of input variables were constructed across multiple time resolutions and ten FS methods, ranging from simple filters to computationally intensive wrappers, were applied. Their impact on the forecasting performance was evaluated on two model types: linear regression (LR) and a multi-layer perceptron (MLP).

The results show that effective feature selection improves forecast accuracy by up to 6% compared to the TSO benchmark, while also reducing model complexity and increasing interpretability. LR achieved the highest absolute accuracy, but well-tuned MLP models with selected features performed comparably, suggesting that linear models only marginally outperform more complex alternatives in this context.

Among the FS methods, wrapper-based techniques, such as SBFS, delivered the most accurate results, though at a high computational cost. In contrast, filter and embedded methods, such as correlation filtering, LASSO, or Random Forest, offered a more practical balance between accuracy and efficiency.

The analysis also identified key predictors: autoregressive terms, renewable energy output, and cross-border exchange nominations were consistently selected across models and methods. These findings carry direct implications for grid operators and stakeholders seeking to enhance imbalance forecasting and balancing operations. Focused use of these features can support the development of more efficient and operationally viable forecasting models. Nevertheless, this study should be viewed as a starting point rather than a definitive solution.

7. CONCLUSION

There are several limitations to this study. Only two model types were considered, and the focus was limited to deterministic point forecasts. More advanced architectures, probabilistic models, and alternative forecast targets were beyond the scope of this work due to time and resource constraints. In addition, the models were trained and tested on historical Belgian data and may not generalize to other grid systems or stress conditions such as extreme weather or outages.

Future research should explore ensemble and hybrid models, probabilistic forecasting, and extended forecast horizons. Advanced techniques like LSTM or attention-based models could further capture temporal dynamics. Integrating economic simulations to assess the operational value of improved forecasts would also strengthen the practical impact.

To ensure reproducibility, all code and data used in this thesis are available in a public GitHub repository. The repository includes the scripts used for model training and figure generation, along with relevant log files from the high-performance computing (HPC) cluster.

GitHub repository: github.com/jannesds/Feature-selection-for-system-imbalance-forecasting

Appendices

Appendix A

Selected features

A.1 Full Feature list

Abbreviation [unit]	Description	Availability at qh+0
SI_min	<i>Minute-wise parameters</i> Cumulative system imbalance	min-2
NRV_min	Cumulative net regulation volume	min-2
IP_min	Positive imbalance price	min-2
MIP_min	Marginal incremental price	min-2
MDP_min	Marginal decremental price	min-2
GUV_min	Total activated upward balancing energy	min-2
IGCC+_min	Upward IGCC exchanges	min-2
aFRR+_min	Upward aFRR	min-2
GDV_min	Total activated downward balancing energy	min-2
IGCC-_min	Downward IGCC exchanges	min-2
aFRR-_min	Downward activated aFRR	min-2
NETPOS_GB_DA	<i>Hourly parameters</i> Day ahead net position United Kingdom	h, t+24
NETPOS_GB_ID	Intraday net position United Kingdom	h, qh+24
Hour_sin / cos	<i>Temporal parameters</i> Cyclical encoding of the hour of the day	current
DayOfWeek_sin / cos	Cyclical encoding of the day of the week	current
Month_sin / cos	Cyclical encoding of the month of the year	current

TABLE A.1: Overview of minute-wise and hourly parameters and their availability timing at qh+0.

A. SELECTED FEATURES

Abbreviation [unit]	Description	Availability at qh+0
SI [MW]	<i>Quarter-hour parameters</i>	
NRV	System Imbalance	qh-1
IP	Net regulation volume, GUV-GDV	qh-1
MIP	Imbalance price	qh-1
MDP	Marginal incremental price	qh-1
	Marginal decremental price	qh-1
LOAD_RT [MW]	Total load in Belgium	qh-1
LOAD_ID [MW]	Load intraday (ID) forecast	qh+24
LOAD_ID_P90	P90 of ID Load forecast	qh+24
LOAD_DA	Load day ahead (DA) forecast	qh, t+24
NETPOS_BE_ID	ID implicit net position Belgium	qh+24
XB_ID [MW]	Cross-border ID commercial schedule (nominated capacity)	qh+4
XB_DA [MW]	Cross-border DA commercial schedule (nominated capacity)	qh, t+24
XB_DA_NET_(ctry)	DA commercial schedule for FR, DE, NL, UK, LU	qh, t+24
XB_DA_IMP_(ctry)	DA import from FR, DE, NL, UK, LU	qh, t+24
XB_DA_EXP_(ctry)	DA export to FR, DE, NL, UK, LU	qh, t+24
XB_RT [MW]	Cross border physical flow	qh-1
XB_RT_IMP_(ctry)	RT import from FR, DE, NL, UK, LU	qh-1
XB_RT_EXP_(ctry)	RT export to FR, DE, NL, UK, LU	qh-1
WIND_RT [MW]	Wind Real-time production	qh-1
WIND_ID [MW]	Wind Intraday Forecast	qh+24
WIND_P90 [MW]	P90 of Wind Production forecast	qh+24
SOLAR_RT [MW]	Solar Real-time production	qh-1
SOLAR_ID [MW]	Solar Intraday forecast	qh+24
SOLAR_P90 [MW]	P90 of Wind Production forecast	qh+24
GEN_DA [MW]	Day ahead generation schedule	qh, t+24
GUV	Total activated upward balancing energy	qh-1
IGCC+	Upward IGCC exchanges	qh-1
aFRR+	Upward activated aFRR	qh-1
mFRR+	Upward activated mFRR	qh-1
GDV	Total activated downward balancing energy	qh-1
IGCC-	Downward IGCC exchanges	qh-1
aFRR-	Downward activated aFRR	qh-1
mFRR-	Downward activated mFRR	qh-1

TABLE A.2: Overview of quarter-hour parameters and their availability timing at qh+0.

A.2 List of selected input features

Feature	Lags (-) & Leads (+)
SI	<i>Quarter-hour parameters</i> [-1, -2, -3, -4, -5, -8, -10, -13, -17, -21, -24, -25, -29, -33, -52, -61, -64, -65, -68, -69, -73, -77, -88, -89, -92, -93, -94, -95, -96, -97, -100, -101, -105, -185, -188, -189, -191, -192, -193, -196, -197, -284, -285, -287, -288, -289, -381, -383, -384, -385, -477, -480, -481, -573, -576, -577, -669, -672, -673, -769, -961]
NRV	[-1, -2, -3, -4, -5, -6, -7, -8, -13, -14, -17, -18, -96]
IP	[-1, -2, -3, -4, -5, -6, -7, -8, -9, -11, -12, -13, -92, -96]
MIP	[-1, -2, -3, -4, -5, -6, -96]
MDP	[-1, -2, -3, -4, -5, -6, -96]
LOAD_RT	[-3, -4, -5]
LOAD_ID	[3, 2, 1, 0, -1, -2, -3]
LOAD_ID_P90	[3, 2, 1, 0, -1, -2, -3]
LOAD_DA	[3, 2, 1, 0, -1, -2, -3]
NETPOS_BE_ID	[22, 14, 13, 9, 8, 4, 1, 0, -1]
XB_DA	[66, 0, -6, -25, -26]
XB_DA_EXP_CNTRY	\approx [1, 0, -1, -2, -6, -7, -38]
XB_DA_IMP_CNTRY	\approx [1, 0, -1, -2, -6, -7, -14]
XB_DA_NET_CNTRY	\approx [1, 0, -1, -2, -6, -7]
XB_RT	[-3, -4, -5, -6, -7, -8, -9, -10]
XB_RT_CNTRY	[-3, -4, -5]
WIND_ID	[16, 11, 4, 3, 2, 1, 0, -1, -2, -3, -10, -14]
WIND_P90	[4, 3, 2, 1, 0, -1, -2, -3, -10]
WIND_RT	[-1, -2, -3, -4, -5, -6, -7, -8, -9, -10, -11, -12]
SOLAR_ID	[4, 3, 2, 0, -1, -2, -30]
SOLAR_P90	[3, 2, -30]
SOLAR_RT	[-1, -2, -29, -125, -126,]
GEN_DA	[-4, -26, -261, -262]
GDV	[-1, -2, -3, -4, -5, -6, -7, -8, -96, -192]
GUV	[-1, -2, -3]
IGGC \pm	[-1, -2, -3, -4, -5]
aFRR \pm	[-1, -2, -3, -4, -5, -6, -7, -8, -96, -192]
mFRR \pm	[-1, -2, -3, -4, -5]
SI_min	<i>Minute-wise parameters</i> [-2, -3, -4, -5, -6, -15, -16, -17, -18, -41, -42, -43, -44, -45, -46, -51, -52, -53, -54, -91, -166, -226, -286]
NRV_min	[-2, -3, -4, -5]
NETPOS_GB_DA	<i>Hourly parameters</i> [2, 1, 0, -1, -2]
NETPOS_GB_ID	[2, 1, 0, -1, -2]

TABLE A.3: Overview of input the 355 features and their lag/lead structure after preliminary analysis

A. SELECTED FEATURES

A.3 Feature-Lag Selection Heatmap

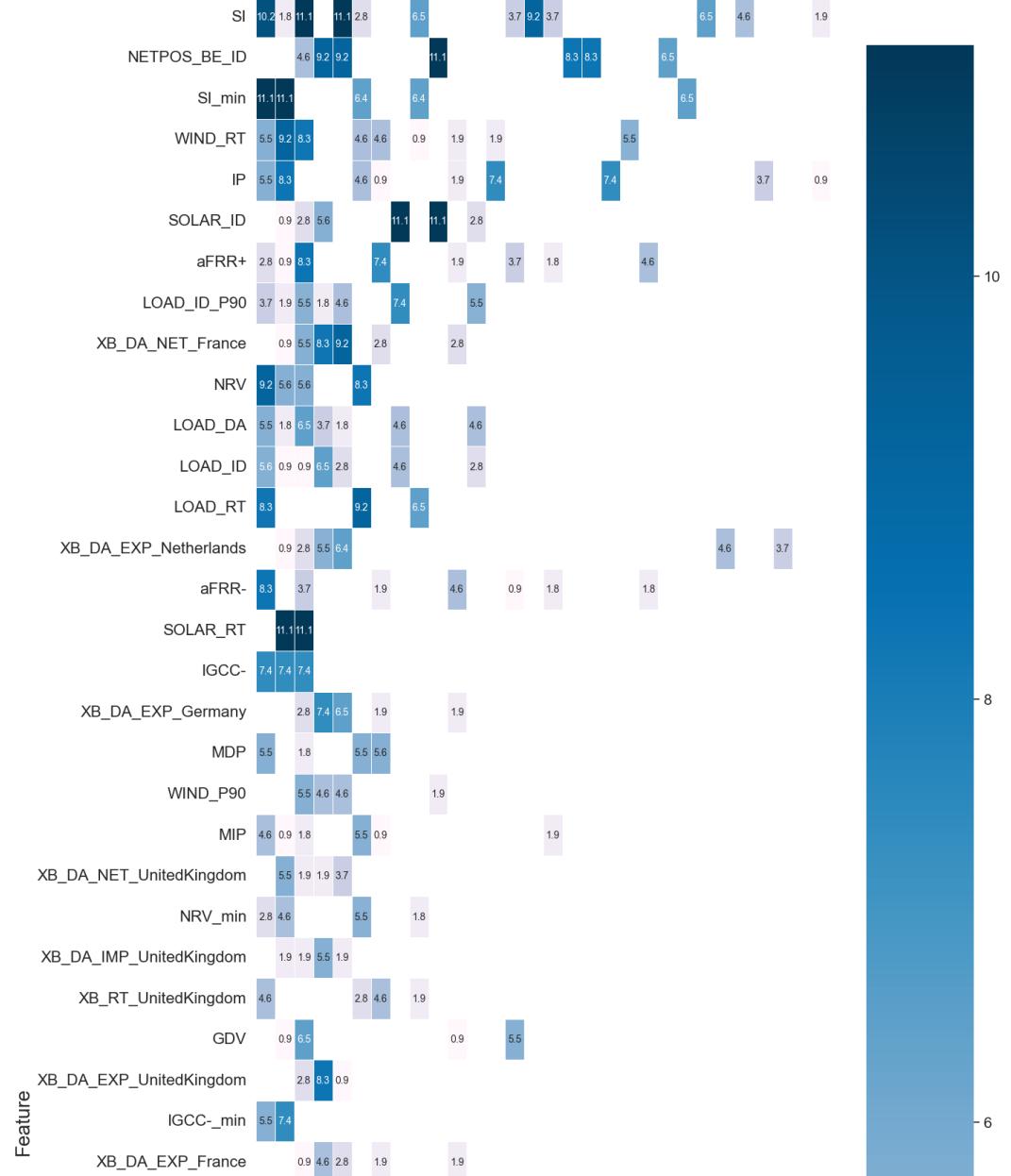


FIGURE A.1: Comprehensive heatmap showing the weighted frequency of feature-lag selections across all models and feature selection methods. The intensity and label values reflect how often each combination was chosen, weighted by model performance. (top half)

A.3. Feature-Lag Selection Heatmap

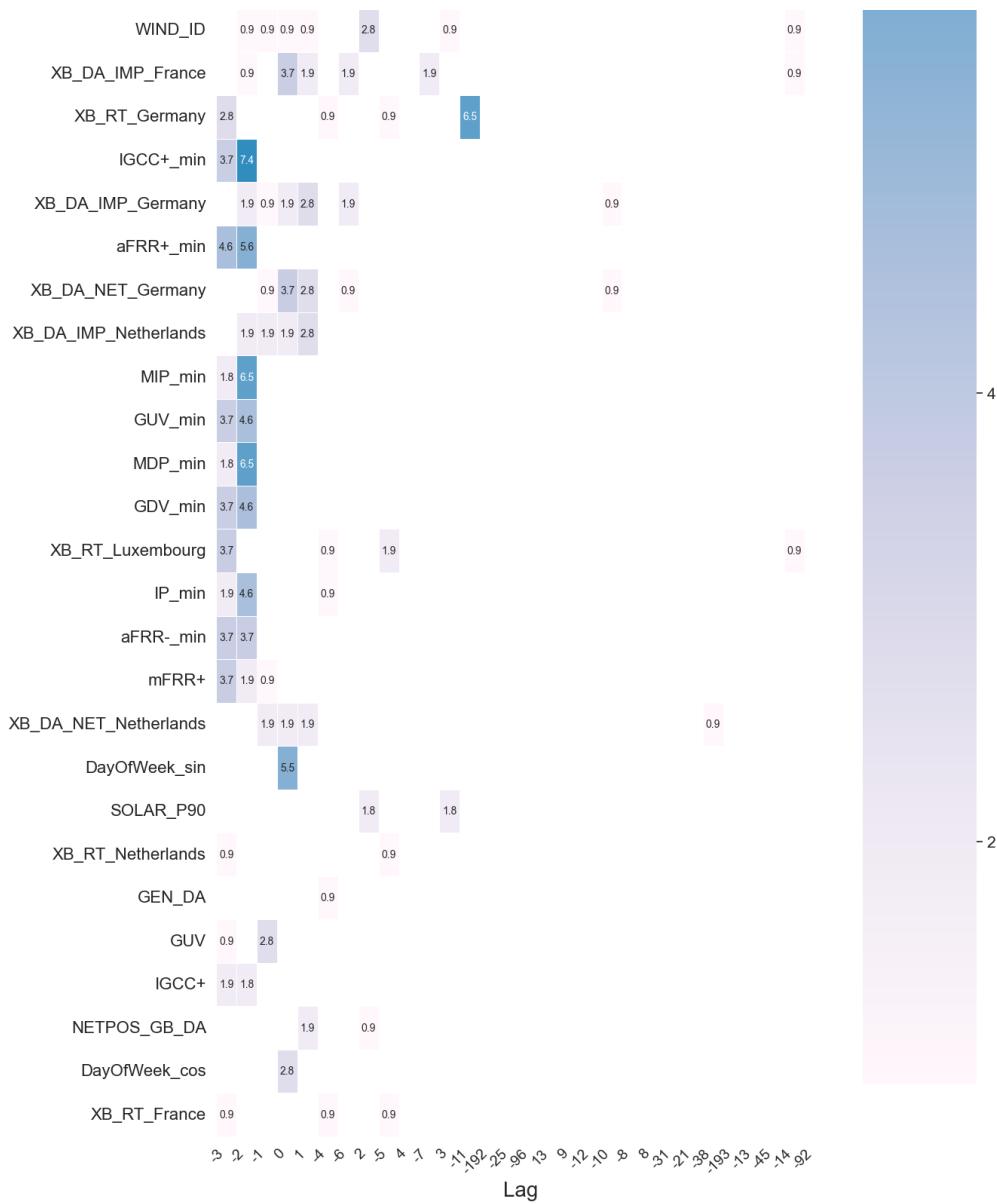


FIGURE A.2: Heatmap (bottom half) of weighted feature-lag selection frequencies.

A. SELECTED FEATURES

A.4 Final feature set

The tables below summarize the final selected features for each model after applying the best-performing feature selection techniques. Features are categorized by their temporal resolution: quarter-hourly (QH), minute-level and hourly. Each feature is accompanied by the corresponding selected lags.

TABLE A.4: Final selected features with corresponding lags and leads for the Multi-Layer Perceptron (MLP) model | MAE = 95.07

Feature	Lags (-) & Leads (+)
<i>Quarter-hour parameters</i>	
SI	[-3, -1]
NRV	[-3]
IP	[-12]
LOAD_RT	[-4, -3]
LOAD_ID	[-3, 3]
SOLAR_RT	[-2, -1]
SOLAR_ID	[2, 4]
WIND_RT	[-2]
NETPOS_BE_ID	[0, 1, 4, 8, 13]
GDV	[-1]
aFRR+	[-1]
aFRR-	[-3]
XB_DA_NET_France	[0]
<i>Minute-wise parameters</i>	
SI_min	[-3, -2]
IP_min	[-2]
<i>Hourly parameters</i>	

A.4. Final feature set

TABLE A.5: Final selected features with corresponding lags and leads for the Linear Regression (LR) model | MAE = 91.72

Feature	Lags (-) & Leads (+)	Feature	Lags (-) & Leads (+)
<i>Quarter-hour parameters</i>			
SI	[-1, -3, -21, -25, -96, -97, -192, -193]	SI_min	[-2, -3, -5, -31]
NRV	[-1, -2, -3, -4]	NRV_min	[-2, -3, -4, -5]
IP	[-2, -3, -4, -11, -12]	GDV_min	[-2, -3]
MIP	[-3, -4]	GUV_min	[-2, -3]
MDP	[-3, -5, -6]	IGCC+_min	[-2, -3]
LOAD_RT	[-3, -4, -5]	IGCC-_min	[-2, -3]
LOAD_ID	[3, 2, 1, 0, -3]	MDP_min	[-2, -3]
LOAD_ID_P90	[2, 1, 0, -1, -3]	MIP_min	[-2]
LOAD_DA	[2, 0, -1, -3]	aFRR+_min	[-2, -3]
NETPOS_BE_ID	[14, 13, 9, 8, 4, 1, 0]	aFRR-_min	[-2]
XB_DA_EXP		NETPOS_GB_DA	<i>Hourly parameters</i>
_France	[0]		[2, 1]
_Germany	[1, 0]		
_Netherlands	[1, 0]		
_UnitedKingdom	[0, -1]		
XB_DA_IMP			
France	[0]		
Germany	[1]		
UnitedKingdom	[0, -1]		
XB_DA_NET			
_France	[1, -1]		
_Germany	[0]		
_Netherlands	[-38]		
_UnitedKingdom	[1]		
XB_RT			
_Germany	[-3, -11]		
_Luxembourg	[-3]		
WIND_P90	[1, 0]		
WIND_RT	[-1, -2, -3, -4, -6, -10]		
SOLAR_ID	[4, 3, 2, 0, -1, -2, -30]		
SOLAR_RT	[-1, -2]		
GEN_DA	[-262]		
GDV	[-192]		
GUV	[-1]		
IGGC-	[-1, -3]		
aFRR+	[-1, -6, -96, -192]		
aFRR-	[-3, -7]		
DayOfWeek_sin	[0]		
Month_cos	[0]		

Appendix B

FS methods in depth

B.1 Random forest

Although the Random Forest (RF) model is not used directly for forecasting in this study, it serves an important auxiliary role as a robust feature importance estimator. Its ability to rank features based on their contribution to prediction accuracy makes it a valuable tool in the feature selection pipeline.

The training strategy for RF regression is outlined in Algorithm 2. The model builds an ensemble of decision trees, each trained on a bootstrap sample of the training data. At each split within a tree, a random subset of features is considered to reduce correlation among trees and improve generalization. The final prediction is computed as the average of the outputs from all individual trees.

Algorithm 2 Random Forest for Regression (adapted from [50])

```
1: for  $b = 1$  to  $B$  do
2:   Draw a bootstrap sample  $\mathbf{Z}^*$  of size  $N$  from the training data.
3:   Grow a tree  $T_b$  to the bootstrapped data by recursively repeating:
4:     while node size  $\geq n_{\min}$  do
5:       Select  $m$  variables at random from the  $p$  features.
6:       Choose the best variable and split-point among the  $m$ .
7:       Split the node into two daughter nodes.
8:     end while
9:   end for
10:  Output the ensemble of trees  $\{T_b\}_{b=1}^B$ .
11:  Prediction for a new input  $x$ :  $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B} \sum_{b=1}^B T_b(x)$ 
```

B.2 Principle component analysis

Given a dataset $X \in \mathbb{R}^{n \times p}$ of n observations and p variables, the principal components are computed as follows:

1. Center the data:

$$X^* = X - \mathbf{1}_n \mu^T$$

where μ is the vector of column means of X and $\mathbf{1}_n$ is a column vector of ones.

2. Compute the covariance matrix:

$$S = \frac{1}{n-1} (X^*)^T X^*$$

3. Solve the eigenvalue problem:

$$Sv_k = \lambda_k v_k$$

where v_k are the eigenvectors (principal directions) and λ_k the eigenvalues (variances explained).

4. Alternatively, perform Singular Value Decomposition (SVD):

$$X^* = U \Sigma V^T$$

where columns of V correspond to principal directions and diagonal entries of Σ relate to explained variances.

Appendix C

Model architecture

C.1 Theoretical background of feedforward neural networks

This appendix briefly summarizes the theoretical foundations of feedforward neural networks (MLP), including their structure, activation functions, loss functions, and optimization strategies [52, 83].

C.1.1 Architecture and forward propagation

A feedforward neural network models a mapping $\mathbf{x} \mapsto \hat{y}$ by passing the input through multiple layers of transformations. These layers alternate between linear combinations of the input and nonlinear activation functions, allowing the network to learn complex, nonlinear relationships.

Consider an input vector $\mathbf{x} = [1, x_1, x_2, \dots, x_M]^\top \in \mathbb{R}^{M+1}$, where the first entry is a constant bias term. Each neuron j in the first hidden layer computes a linear combination of the input as:

$$a_j^{(1)} = \mathbf{x}^\top \mathbf{w}_j^{(1)}, \quad (\text{C.1})$$

where $\mathbf{w}_j^{(1)}$ is the weight vector associated with neuron j in the first hidden layer. The resulting activation is passed through a nonlinear function σ , typically a ReLU or tanh, to obtain the neuron's output:

$$z_j^{(1)} = \sigma(a_j^{(1)}). \quad (\text{C.2})$$

The outputs of all H_1 neurons in the first hidden layer can be stacked into a vector:

$$\mathbf{z}^{(1)} = [1, z_1^{(1)}, z_2^{(1)}, \dots, z_{H_1}^{(1)}]^\top, \quad (\text{C.3})$$

where the leading 1 accounts for the bias term in the next layer.

C. MODEL ARCHITECTURE

In deeper architectures, this process is repeated. For a second hidden layer, the input becomes $\mathbf{z}^{(1)}$, and each neuron k computes its activation as:

$$a_k^{(2)} = (\mathbf{z}^{(1)})^\top \mathbf{w}_k^{(2)}, \quad (\text{C.4})$$

followed by the nonlinear transformation:

$$z_k^{(2)} = \sigma^{(2)}(a_k^{(2)}). \quad (\text{C.5})$$

This layer-wise propagation continues until the final output layer is reached. For regression tasks, the output neuron often applies a linear transformation without activation:

$$\hat{y} = (\mathbf{z}^{(L-1)})^\top \mathbf{w}^{(L)} + b^{(L)}, \quad (\text{C.6})$$

where L is the total number of layers and $\mathbf{z}^{(L-1)}$ is the output of the last hidden layer.

This chain of operations is referred to as the forward pass. The network parameters $\{\mathbf{w}^{(l)}, b^{(l)}\}$ are optimized during training to minimize a loss function that quantifies the prediction error. A feedforward neural network models a mapping $\mathbf{x} \mapsto \hat{y}$ using a sequence of linear transformations followed by nonlinear activation functions.

Thus for an MLP with one hidden layer, the forward pass is given by:

$$\mathbf{z}^{(1)} = \sigma(W^{(1)}\mathbf{x} + \mathbf{b}^{(1)}), \quad (\text{C.7})$$

$$\hat{y} = W^{(2)}\mathbf{z}^{(1)} + b^{(2)}, \quad (\text{C.8})$$

where $W^{(1)}, W^{(2)}$ are weight matrices, $\mathbf{b}^{(1)}, b^{(2)}$ are biases, and $\sigma(\cdot)$ is a nonlinear activation function applied element-wise.

C.1.2 Activation functions

nonlinear activation functions are used to enable the network to model complex relationships. Common choices include:

- **ReLU:** $\sigma(z) = \max(0, z)$, widely used due to simplicity and strong empirical performance.
- **Sigmoid:** $\sigma(z) = \frac{1}{1+e^{-z}}$, maps input to $[0, 1]$, historically popular but less used due to vanishing gradients.
- **Tanh:** $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, outputs values in $[-1, 1]$ and zero-centered.

In this thesis, ReLU is chosen for all hidden layers, as it is the default recommendation in deep learning, due to its efficiency and effectiveness in practice, typically learning much faster.

C.1.3 Cost functions

For regression tasks, the network is trained to minimize the Mean Squared Error (MSE) loss:

$$J(w) = \frac{1}{N} \sum_{i=1}^N (f(\mathbf{x}_i; w) - y_i)^2, \quad (\text{C.9})$$

where $f(\cdot; w)$ denotes the network output and y_i is the true target.

C.1.4 Optimization methods

Training the network involves minimizing the loss function using gradient-based optimization. Gradients are computed efficiently via backpropagation.

- **Gradient Descent:** Updates weights based on the gradient over the full training set.
- **Stochastic Gradient Descent (SGD):** Updates weights using gradients from individual samples.
- **Mini-batch Gradient Descent:** A compromise using small batches for stability and efficiency.

Adam optimizer

In this thesis, we use the Adam optimizer [83], which combines momentum and adaptive learning rates. It is widely used due to its fast convergence and minimal tuning requirements. The default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and $\alpha = 0.001$) generally work well across a variety of tasks.

Regularization techniques

Overfitting is a common issue in neural network training, where the model fits the training data well but performs poorly on unseen data. To mitigate this, regularization techniques are used to constrain model complexity and improve generalization.

This work applies L1 regularization, which penalizes the absolute values of the model weights and encourages sparsity by driving some weights to zero. This makes it particularly useful for feature selection.

$$J_{L1}(\mathbf{w}) = J(\mathbf{w}) + \lambda \|\mathbf{w}\|_1, \quad (\text{C.10})$$

where λ is a regularization parameter. The full Lagrangian form of the LASSO objective is also presented in Equation (4.6).

In contrast, L2 regularization penalizes the squared magnitude of the weights:

$$J_{L2}(\mathbf{w}) = J(\mathbf{w}) + \lambda \mathbf{w}^\top \mathbf{w}. \quad (\text{C.11})$$

C. MODEL ARCHITECTURE

While both methods mitigate overfitting, L1 regularization is particularly effective for feature selection due to its ability to produce sparse solutions [104].

Further technical details (e.g. full derivation of backpropagation and convergence proofs) are beyond the scope of this thesis. For a rigorous treatment, the reader is referred to [83, 52, 116].

Appendix D

Supplementary analyses

D.1 Probabilistic Forecast Metrics

Although not applied in this study, probabilistic forecasting metrics are frequently used in recent literature. The *Continuous Ranked Probability Score (CRPS)* generalizes MAE to full predictive distributions by assessing both calibration and sharpness, while the *Pinball Loss Function (PLF)* is designed for quantile forecasts, particularly useful under asymmetric risk [25]. To enable comparison between point and probabilistic forecasts, normalized metrics such as *NMAE* and *NRMSE* are often employed.

D.2 STL: Seasonal and Trend decomposition using Loess

Additive Model

An additive decomposition model assumes that the components of a time series combine linearly as follows:

$$y(t) = \text{Level} + \text{Trend} + \text{Seasonality} + \text{Noise}$$

This implies that changes in the time series over time are consistently additive. By contrast, a multiplicative model assumes that components interact nonlinearly (i.e., $y(t) = \text{Level} \times \text{Trend} \times \text{Seasonality} \times \text{Noise}$), which is only suitable for strictly positive series. Since the system imbalance signal includes negative values, the additive model is more appropriate for this analysis.

Application to the System Imbalance Signal Figures D.1 and D.2 display the additive seasonal decomposition of the Belgian system imbalance signal. Figure D.1 shows the decomposition applied to the weekly averaged series in 2022, while Figure D.2 presents the decomposition of monthly averages from January 2020 to December 2023.

In both cases, the trend component is relatively limited and does not exhibit a strong

D. SUPPLEMENTARY ANALYSES

directional change. The seasonal component appears irregular and weak, confirming the absence of a consistent cyclical pattern. The residual component dominates, particularly during the early and later parts of the time series, where larger fluctuations are evident. These results support the interpretation of the imbalance signal as primarily stochastic, with limited deterministic structure.

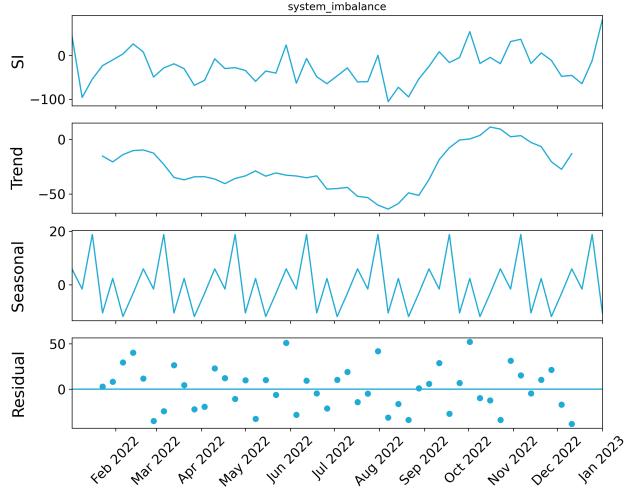


FIGURE D.1: Additive seasonal decomposition of the weekly average system imbalance for 2022.

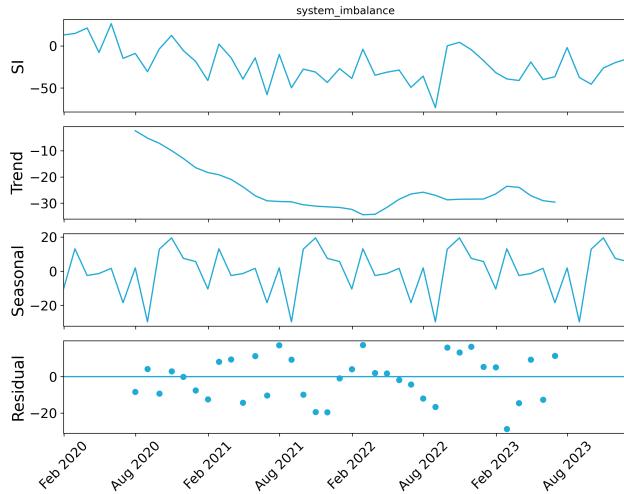


FIGURE D.2: Additive seasonal decomposition of the monthly average system imbalance from January 2020 to December 2023.

D.3 Extended exploratory results

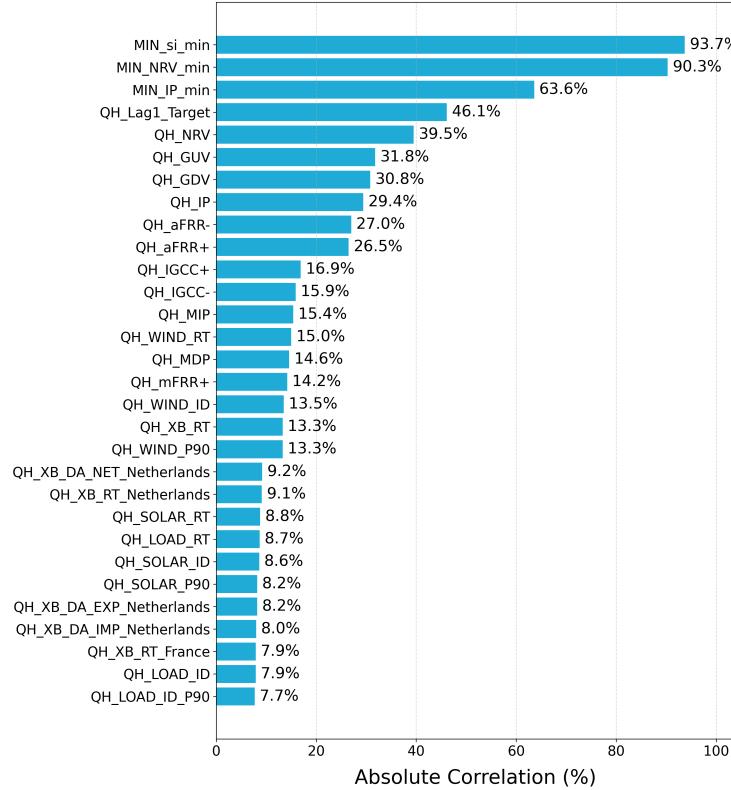


FIGURE D.3: Absolute Pearson correlation of candidate features with system imbalance (SI).

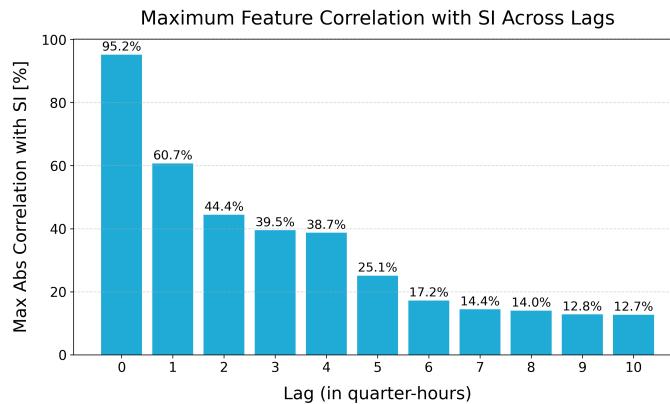


FIGURE D.4: Maximum cross-correlation of exogenous variables with SI across multiple lags

D. SUPPLEMENTARY ANALYSES

D.4 Feature selection overlap matrix

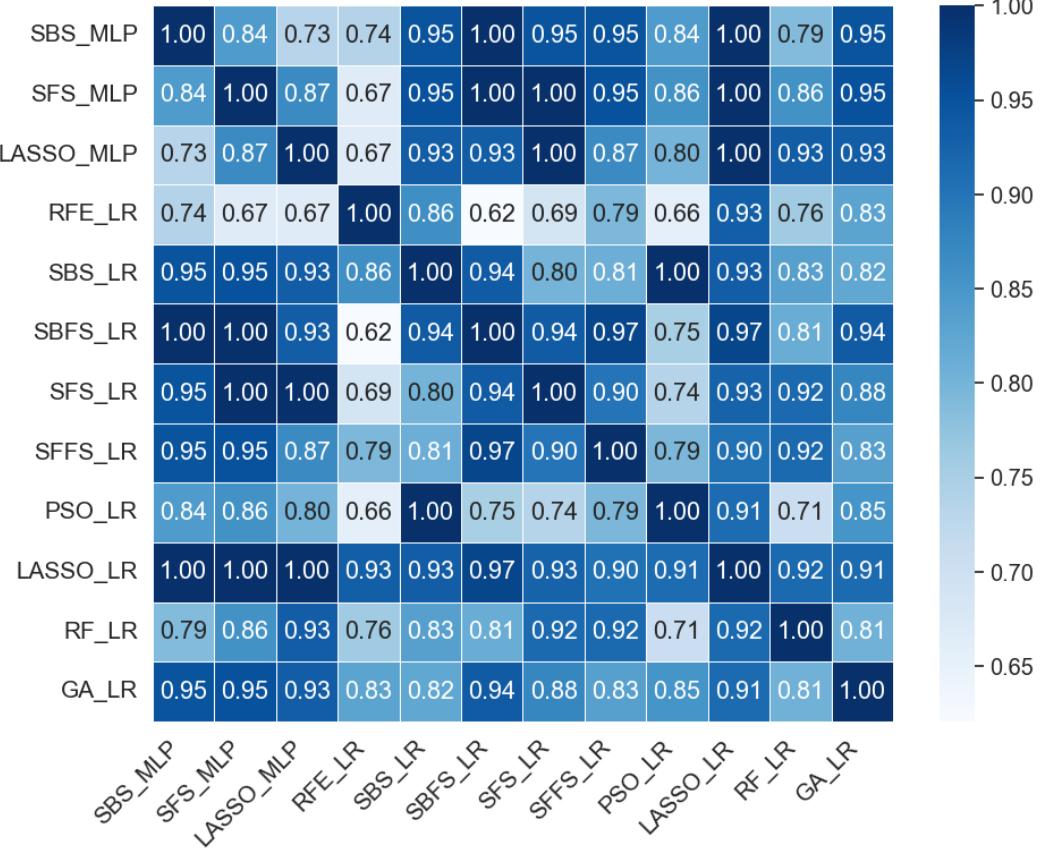


FIGURE D.5: Overlap matrix showing the number of shared features between different selection methods or models. Higher values indicate greater consistency in the selected feature subsets.

D.5. Cross-validation performance details

D.5 Cross-validation performance details

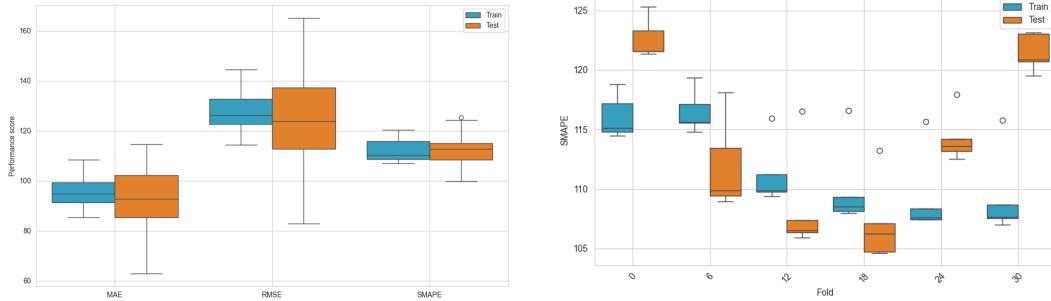


FIGURE D.6: Cross-validation performance across time-aware folds.

D.6 Cross-border feature importance analysis

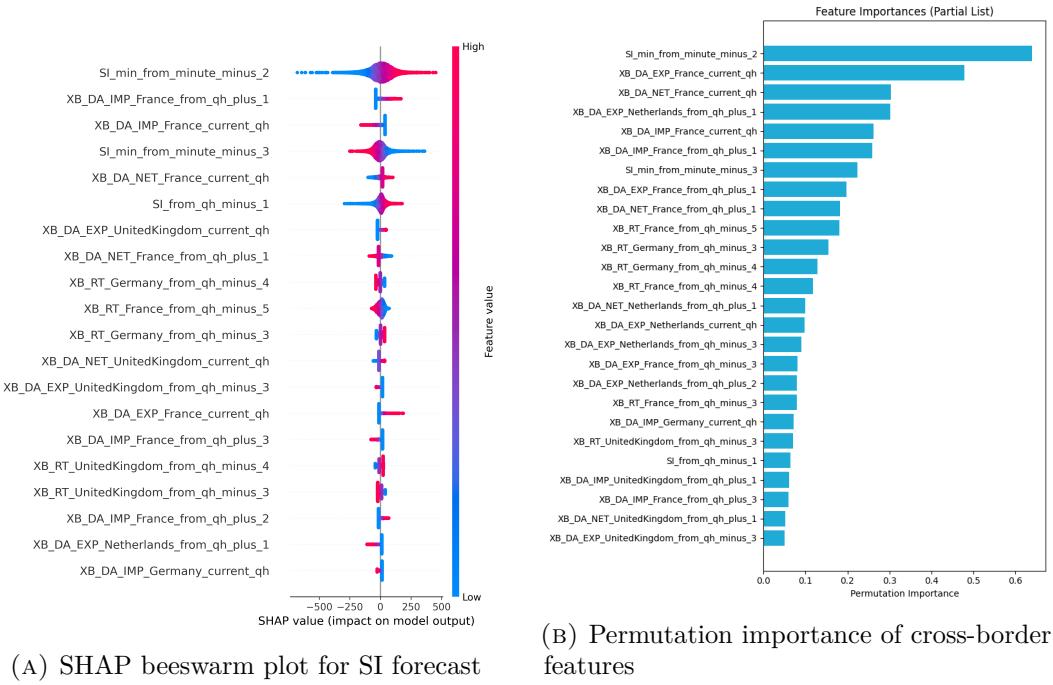


FIGURE D.7: Feature importance of cross-border nominations using SHAP and permutation metrics.

D. SUPPLEMENTARY ANALYSES

D.7 PCA

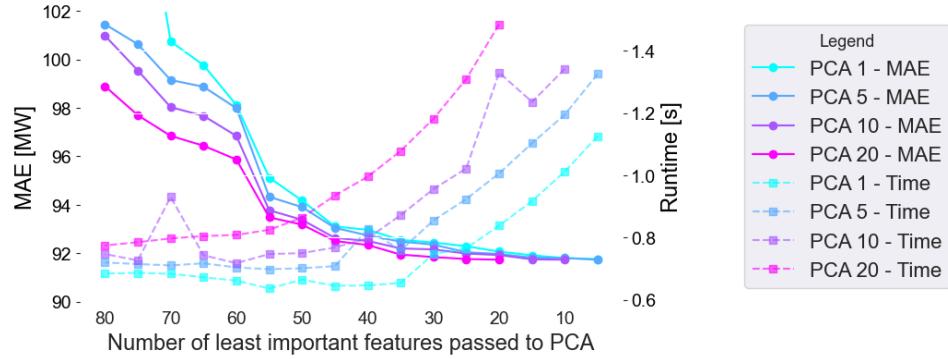


FIGURE D.8: Effect of applying PCA to the least important features: impact on MAE and runtime across different component settings.

D.8 Imbalance price

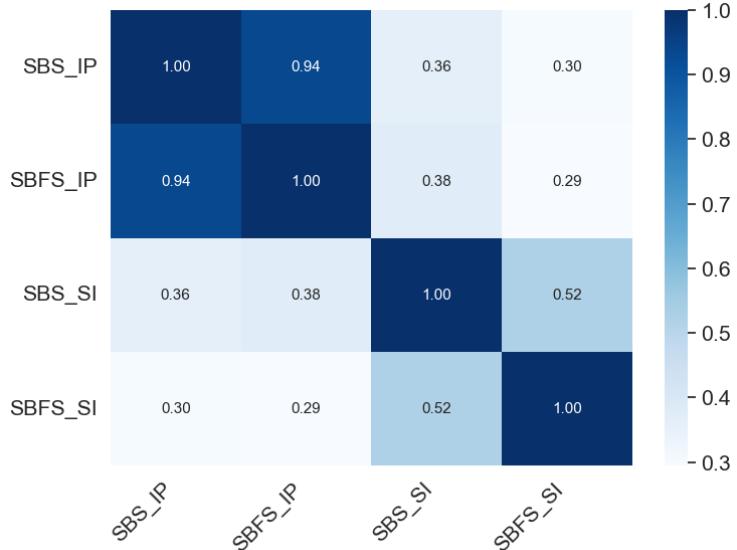


FIGURE D.9: Jaccard similarity matrix for feature subsets selected by SBS and SBFS for imbalance price and system imbalance.

Bibliography

- [1] Agency for the Cooperation of Energy Regulators (ACER). Market coupling development. <https://www.acer.europa.eu/electricity/market-rules/capacity-allocation-and-congestion-management/market-coupling-development>, 2024. Accessed: 2025-04-29.
- [2] Agency for the Cooperation of Energy Regulators (ACER). Market monitoring report 2024: Key developments in the electricity sector, April 2024. Accessed: 2025-05-16.
- [3] N. Amjadi and F. Keynia. Day-ahead price forecasting of electricity markets by mutual information technique and cascaded neuro-evolutionary algorithm. *IEEE Transactions on Power Systems*, 24(1):306–318, 2009.
- [4] A. F. Atiya. Why does forecast combination work so well? *International Journal of Forecasting*, 36(1):197–200, 2020. M4 Competition.
- [5] J. Baetens, J. Laveyne, G. Van Eetvelde, and L. Vandeveld. Imbalance pricing methodology in belgium: Implications for industrial consumers. In *2020 17th International Conference on the European Energy Market (EEM)*, pages 1–6, 2020.
- [6] I. Balazs, A. Fodor, and A. Magyar. Short-term system imbalance forecast using linear and nonlinear methods. *Energy Systems*, Mar. 2024.
- [7] A. Barnes and J. Balda. Sizing and economic assessment of energy storage with real-time pricing and ancillary services. In *2013 4th IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*, pages 1–7. IEEE, 2013.
- [8] C. Bergmeir, R. J. Hyndman, and B. Koo. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120:70–83, 2018.
- [9] J. Bottieau, L. Hubert, Z. De Greve, F. Vallee, and J.-F. Toubeau. Very-short-term probabilistic forecasting for a risk-aware participation in the single price imbalance settlement. *IEEE Transactions on Power Systems*, 35(2):1218–1230, 2020.

BIBLIOGRAPHY

- [10] J. Bottieau, F. Vallée, Z. De Grève, and J.-F. Toubeau. Leveraging provision of frequency regulation services from wind generation by improving day-ahead predictions using lstm neural networks. In *2018 IEEE International Energy Conference (ENERGYCON)*, pages 1–6. IEEE, 2018.
- [11] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
- [12] L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001.
- [13] D. Bunn. Forecasting loads and prices in competitive power markets. *Proceedings of the IEEE*, 88(2):163–169, 2000.
- [14] L. M. Cannas, N. Dessì, and B. Pes. Assessing similarity of feature selection techniques in high-dimensional domains. *Pattern Recognition Letters*, 34(12):1446–1453, 2013.
- [15] G. Chandrashekhar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, Jan. 2014.
- [16] I. Cohen, Y. Huang, J. Chen, J. Benesty, J. Benesty, J. Chen, Y. Huang, and I. Cohen. Pearson correlation coefficient. *Noise reduction in speech processing*, pages 1–4, 2009.
- [17] E. Commision. Commission regulation (EU) 2017/2195 of 23 november 2017 establishing a guideline on electricity balancing. *Official Journal of the European Union*, 60:6–53, 2017.
- [18] C. Contreras. System Imbalance Forecasting and Short-Term Bidding Strategy To Minimize Imbalance Costs of Transacting in the Spanish Electricity Market. 2016.
- [19] J. Contreras, R. Espinola, F. Nogales, and A. Conejo. Arima models to predict next-day electricity prices. *IEEE Transactions on Power Systems*, 18(3):1014–1020, 2003.
- [20] J. E. Contreras-Ocana, M. A. Ortega-Vazquez, and B. Zhang. Participation of an energy storage aggregator in electricity markets. *IEEE Transactions on Smart Grid*, 10(2):1171–1183, 2019.
- [21] J. M. Cortina. Interaction, nonlinearity, and multicollinearity: Implications for multiple regression. *Journal of management*, 19(4):915–922, 1993.
- [22] J. Cui, N. Gu, T. Zhao, C. Wu, and M. Chen. Forecast competition in energy imbalance market. *IEEE Transactions on Power Systems*, 37(3):2397–2413, 2022.
- [23] A. Davydenko and R. Fildes. Measuring forecasting accuracy: The case of judgmental adjustments to sku-level demand forecasts. *International Journal of Forecasting*, 29(3):510–522, 2013.

- [24] F. X. Diebold and R. S. Mariano. Comparing predictive accuracy. *Journal of Business & economic statistics*, 20(1):134–144, 2002.
- [25] J. Dumas, I. Boukas, M. M. de Villena, S. Mathieu, and B. Cornelusse. Probabilistic Forecasting of Imbalance Prices in the Belgian Context. In *2019 16th International Conference on the European Energy Market (EEM)*, pages 1–7, Sept. 2019. ISSN: 2165-4093.
- [26] A. Eicke, O. Ruhnau, and L. Hirth. Electricity balancing as a market equilibrium: An instrument-based estimation of supply and demand for imbalance energy. *Energy Economics*, 102:105455, 2021.
- [27] Elia. System Imbalance forecast and its publication to stakeholders, 2021.
- [28] Elia. Documentation of the methodology used for the system imbalance forecast publications, Oct. 2022.
- [29] Elia. Implementation guide: Mari mfrr activation. <https://www.elia.be/en/electricity-market-and-system/icaros-and-mari-projects>, 2023. Accessed: 2025-04-29.
- [30] Elia. Rules for the compensation of the quarter-hourly imbalances (into force until 26/11/2024). [Online]. Available: https://www.elia.be/-/media/project/elia/elia-site/electricity-market-and-system---document-library/balancing---balancing-services-and-bsp/2024/20240328_balancing_rules_en.pdf, 2023.
- [31] Elia. Elia transmission belgium successfully accesses picasso, the european platform for exchanging secondary balancing energy. https://www.elia.be/en/press/2024/11/20241128_picasso-press-release, 2024. Accessed: 2025-04-29.
- [32] Elia. Rules for the compensation of the quarter-hourly imbalances (into force as of 26/11/2024). [Online]. Available: https://www.elia.be/-/media/project/elia/elia-site/electricity-market-and-system---document-library/balancing---balancing-services-and-bsp/2024/20241126_balancing_rules_en_clean.pdf, 2024.
- [33] Elia. System services: Balancing products - fcr, afrr and mfrr. <https://www.elia.be/en/electricity-market-and-system/system-services/keeping-the-balance>, 2024. Accessed: 2025-04-29.
- [34] Elia. Terms and conditions for balance responsible parties (BRPs) ("T&C BRP"). [Online]. Available: <https://www.elia.be/-/media/project/elia/elia-site/electricity-market-and-system---document-library/balance-responsible-party-and-system-imbalance/2024/>

BIBLIOGRAPHY

- [tc-brp-into-force-as-of-the-local-go-live-of-the-mfrr-serviceenv12024.pdf](#), 2024. Accessed: 2025-04-25.
- [35] Elia. How to become a balance responsible party, 2025. Accessed: 2025-05-07.
 - [36] Elia Group. Legal framework in belgium, 2024. Accessed: 2025-04-28.
 - [37] ENTSO-E. Electricity balancing in europe. https://eepublicdownloads.enties.eu/clean-documents/Network%20codes%20documents/NC%20EB/entso-e_balancing_in%20_europe_report_Nov2018_web.pdf, 2018. Accessed: 2025-04-29.
 - [38] ENTSO-E. Entso-e balancing report 2024. https://eepublicdownloads.blob.core.windows.net/public-cdn-container/clean-documents/news/2024/240628_ENTSO-E_Balancing_Report_2024.pdf, 2024. Accessed: 2025-04-29.
 - [39] ENTSO-E. Manually Activated Reserves Initiative (MARI). https://www.enties.eu/network_codes/eb/mari/, 2025. Accessed: 2025-05-06.
 - [40] H. Eskandari, H. Saadatmand, M. Ramzan, and M. Mousapour. Innovative framework for accurate and transparent forecasting of energy consumption: A fusion of feature selection and interpretable machine learning. *Applied Energy*, 366:123314, 2024.
 - [41] Federale Overheidsdienst Justitie. 29 april 1999 - act concerning the organisation of the electricity market. [Online]. Available: https://www.ejustice.just.fgov.be/cgi_loi/change_lg.pl?language=nl&la=N&table_name=wet&cn=1999042942, 1999. Original title in Dutch: *Wet betreffende de organisatie van de elektriciteitsmarkt*.
 - [42] FOD Economie, K.M.O., Middenstand en Energie. Belgian energy data overview - zomer 2023, July 2023. Uitgever: Severine Waterbley.
 - [43] M. A. Ganjei and R. Boostani. A hybrid feature selection scheme for high-dimensional data. *Engineering Applications of Artificial Intelligence*, 113:104894, 2022.
 - [44] M. Garcia and D. Kirschen. Forecasting system imbalance volumes in competitive electricity markets. *IEEE Transactions on Power Systems*, 21(1):240–248, Feb. 2006. Conference Name: IEEE Transactions on Power Systems.
 - [45] J. V. Gompel, B. Claessens, and C. Develder. Probabilistic forecasting of power system imbalance using neural network-based ensembles.
 - [46] C. González, J. Mira-McWilliams, and I. Juárez. Important variable assessment and electricity price forecasting based on regression tree models: Classification and regression trees, bagging and random forests. *IET Generation, Transmission & Distribution*, 9(11):1120–1128, 2015.

- [47] G. Gross and F. Galiana. Short-term load forecasting. *Proceedings of the IEEE*, 75(12):1558–1573, 1987.
- [48] G. Hackeling. *Mastering Machine Learning with scikit-learn*. Packt Publishing Ltd, 2017.
- [49] S. Han, H. Ru, G. Wang, X. Fu, G. Zhou, and C. Yang. Research on power load forecasting of pca-cnn-lstm based on sliding window. In *2023 3rd International Conference on New Energy and Power Engineering (ICNEPE)*, pages 466–471, 2023.
- [50] T. Hastie. The elements of statistical learning: data mining, inference, and prediction, 2009.
- [51] X. He and J. Wang. A hybrid forecasting system based on comprehensive feature selection and intelligent optimization for stock price index forecasting. *Mathematics (2227-7390)*, 12(23), 2024.
- [52] J. Heaton. Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618. *Genetic programming and evolvable machines*, 19(1):305–307, 2018.
- [53] L. Hirth and I. Ziegenhagen. Balancing power and variable renewables: Three links. *Renewable and Sustainable Energy Reviews*, 50:1035–1051, 2015.
- [54] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 11 1997.
- [55] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, and H. Zareipour. Energy forecasting: A review and outlook. *IEEE Open Access Journal of Power and Energy*, 7:376–388, 2020.
- [56] Y.-Y. Hong and C.-P. Wu. Day-ahead electricity price forecasting using a hybrid principal component analysis network. *Energies*, 5(11):4711–4725, 2012.
- [57] S. Hu, T. Zhang, F. Yang, Z. Gao, Y. Ge, Q. Zhang, H. Sun, and K. Xu. Short-term load forecasting based on mutual information and bi-lstm considering fluctuation in importance values of features. *IEEE Access*, 12:23653–23665, 2024.
- [58] Y. Hu, M. Armada, and M. Jesus Sanchez. Potential utilization of battery energy storage systems (bess) in the major european electricity markets. *Applied Energy*, 322:119512, 2022.
- [59] N. Huang, G. Lu, and D. Xu. A permutation importance-based feature selection method for short-term electricity load forecasting using random forest. *Energies*, 9(10):767, Sept. 2016.

BIBLIOGRAPHY

- [60] International Energy Agency. Electricity 2025. <https://www.iea.org/reports/electricity-2025>, 2025. IEA, Paris. Licence: CC BY 4.0. Accessed: 2025-05-07.
- [61] A. Janecek, W. Gansterer, M. Demel, and G. Ecker. On the relationship between feature selection and classification accuracy. In Y. Saeys, H. Liu, I. Inza, L. Wehenkel, and Y. V. d. Pee, editors, *Proceedings of the Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery at ECML/PKDD 2008*, volume 4 of *Proceedings of Machine Learning Research*, pages 90–105, Antwerp, Belgium, 15 Sep 2008. PMLR.
- [62] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002.
- [63] N. V. Karakatsani and D. W. Bunn. Forecasting electricity prices: The impact of fundamentals and time-varying coefficients. *International Journal of Forecasting*, 24(4):764–785, 2008.
- [64] D. Keles, J. Scelle, F. Paraschiv, and W. Fichtner. Extended forecast methods for day-ahead electricity spot prices applying artificial neural networks. *Applied Energy*, 162:218–230, 2016.
- [65] J. H. Kim. Multicollinearity and misleading statistical results. *Korean journal of anesthesiology*, 72(6):558–569, 2019.
- [66] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [67] S. Kratochvil. System imbalance forecast. PhD thesis, Czech Technical University, 2016.
- [68] S. Kulkarni, S. P. Simon, and K. Sundareswaran. A spiking neural network (snn) forecast engine for short-term electrical load forecasting. *Applied Soft Computing*, 13(8):3628–3635, 2013.
- [69] J. Lago, F. De Ridder, and B. De Schutter. Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms. *Applied Energy*, 221:386–405, July 2018.
- [70] J. Lago, G. Marcjasz, B. De Schutter, and R. Weron. Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. *Applied Energy*, 293:116983, July 2021.
- [71] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM Computing Surveys*, 50(6):1–45, Nov. 2018.
- [72] W. Li and D. M. Becker. Day-ahead electricity price prediction applying hybrid models of LSTM-based deep learning methods and feature selection algorithms under consideration of market coupling. *Energy*, 237:121543, Dec. 2021.

- [73] Y. Li, J. Chang, and Y. Tian. Improved cost-sensitive multikernel learning support vector machine algorithm based on particle swarm optimization in pulmonary nodule recognition. *Soft Computing*, 26(7):3369–3383, 2022.
- [74] Z. Li, J. Han, W. E, and Q. Li. On the curse of memory in recurrent neural networks: Approximation and optimization analysis. (arXiv:2009.07799), Aug. 2024. arXiv:2009.07799 [cs].
- [75] Y. Liang, D. Niu, and W.-C. Hong. Short term load forecasting based on feature extraction and improved general regression neural network model. *Energy*, 166:653–663, 2019.
- [76] N. Ludwig, S. Feuerriegel, and D. Neumann. Putting Big Data analytics to work: Feature selection for forecasting electricity prices using the LASSO and random forests. *Journal of Decision Systems*, 24(1):19–36, Jan. 2015.
- [77] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.
- [78] T. Mahajan, G. Singh, and G. Bruns. An Experimental Assessment of Treatments for Cyclical Data.
- [79] S. Makridakis. Accuracy measures: theoretical and practical concerns. *International journal of forecasting*, 9(4):527–529, 1993.
- [80] A. Misiorek. Short-term forecasting of electricity prices: Do we need a different model for each hour? 2008.
- [81] National Energy System Operator (NESO). Platform for energy forecasting (pef) programme overview, 2020. Accessed: 2025-05-07.
- [82] National Energy System Operator (NESO). Balancing programme: Pef and asdp release updates, 2023. Accessed: 2025-05-07.
- [83] M. A. Nielsen. *Neural networks and deep learning*, volume 25. Determination press San Francisco, CA, USA, 2015.
- [84] D. Niu, K. Wang, L. Sun, J. Wu, and X. Xu. Short-term photovoltaic power generation forecasting based on random forest feature selection and ceemd: A case study. *Applied Soft Computing*, 93:106389, 2020.
- [85] I. Pavic, T. Capuder, and H. Pandzic. Analysis of afrr and mfrr balancing capacity & energy demands and bid curves. In *2022 IEEE 7th International Energy Conference (ENERGYCON)*, pages 1–6, 2022.
- [86] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. <http://scikit-learn.org/>, 2011.

BIBLIOGRAPHY

- [87] K. Plakas, N. Andriopoulos, A. Birbas, I. Moraitis, and A. Papalexopoulos. A forecasting model for the prediction of system imbalance in the greek power system. *Engineering Proceedings*, 39(1), 2023.
- [88] L. Pozzetti and J. P. Cartlidge. Trading electricity markets using neural networks. In *32nd European Modelling and Simulation Symposium*, pages 311–318, 2020.
- [89] N. Pudjihartono, T. Fadason, A. Kempa-Liehr, and J. O’Sullivan. A review of feature selection methods for machine learning-based disease risk prediction. *Frontiers in Bioinformatics*, 2:927312, 06 2022.
- [90] S. Ramjee and A. E. Gamal. Efficient wrapper feature selection using autoencoder and model based elimination. *arXiv preprint arXiv:1905.11592*, 2019.
- [91] S. Raschka. Sequentialfeatureselector. https://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector/, n.d. Accessed: 2025-05-21.
- [92] F. J. Ribeiro, J. A. P. Lopes, F. J. Soares, and A. G. Madureira. A novel tso settlement scheme for the frequency containment reserve cooperation in europe’s integrated electricity market. *Utilities Policy*, 91:101821, 2024.
- [93] RTE. Margins and imbalance: a new data service at your disposal, 2021. Accessed: 2025-05-07.
- [94] T. S. Salem, K. Kathuria, H. Ramampiaro, and H. Langseth. Forecasting intra-hour imbalances in electric power systems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:9595–9600, 07 2019.
- [95] Y. Saeys, T. Abeel, and Y. Van de Peer. Robust feature selection using ensemble feature selection techniques. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2008, Antwerp, Belgium, September 15-19, 2008, Proceedings, Part II* 19, pages 313–325. Springer, 2008.
- [96] T. D. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2(6):459–473, 1989.
- [97] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- [98] N. Sharma and K. Saroha. Study of dimension reduction methodologies in data mining. In *Communication & Automation International Conference on Computing*, pages 133–137, May 2015.
- [99] P. Shinde. Brainstorming on some key ongoing projects in the european electricity markets. <https://www.linkedin.com/pulse/brainstorming-some-key-ongoing-projects-european-markets-shinde/>, june 2022. Accessed: 2025-05-05.

- [100] R. H. Shumway, D. S. Stoffer, and D. S. Stoffer. *Time series analysis and its applications*, volume 3. Springer, 2000.
- [101] R. Smets, K. Bruninx, J. Bottieau, J.-F. Toubeau, and E. Delarue. Strategic implicit balancing with energy storage systems via stochastic model predictive control. *IEEE Transactions on Energy Markets, Policy and Regulation*, 1(4):373–385, Dec. 2023.
- [102] R. Steuer, J. Kurths, C. O. Daub, J. Weise, and J. Selbig. The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, 18(suppl_2):S231–S240, 2002.
- [103] J. W. Taylor and P. E. McSharry. Short-term load forecasting methods: An evaluation based on european data. *IEEE Transactions on Power Systems*, 22(4):2213–2219, 2007.
- [104] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.
- [105] J.-F. Toubeau, J.-r. Bottieau, Z. De Greve, F. Vallée, and K. Bruninx. Data-driven scheduling of energy storage in day-ahead energy and reserve markets with probabilistic guarantees on real-time delivery. *IEEE Transactions on Power Systems*, 36(4):2815–2828, 2021.
- [106] B. Uniejewski and R. Weron. Efficient forecasting of electricity spot prices with expert and lasso models. *Energies*, 11(8):2039, 2018.
- [107] T. Urdiales. Forecasting Grid System Imbalance. *Research portal Eindhoven University of Technology*.
- [108] V. Veeramsetty, D. R. Chandra, F. Grimaccia, and M. Mussetta. Short term electric power load forecasting using principal component analysis and recurrent neural networks. *Forecasting*, 4(1):149–164, 2022.
- [109] Q. Wen and Y. Liu. Feature engineering and selection for prosumer electricity consumption and production forecasting: A comprehensive framework. *Applied Energy*, 381:125176, Mar. 2025.
- [110] R. Weron. *Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach*. John Wiley & Sons, Dec. 2006. Google-Books-ID: TEXKC-QAAQBAJ.
- [111] E. Wessel, R. Smets, and E. Delarue. Risk-aware participation in day-ahead and real-time balancing markets for energy storage systems. *Electric Power Systems Research*, 235:110741, Oct. 2024.
- [112] T. Xie, Y. Zhang, G. Zhang, K. Zhang, H. Li, and X. He. Research on electric vehicle load forecasting considering regional special event characteristics. *Frontiers in Energy Research*, Volume 12 - 2024, 2024.

BIBLIOGRAPHY

- [113] L. Yu and H. Liu. Efficient feature selection via analysis of relevance and redundancy. *Journal of machine learning research*, 5(Oct):1205–1224, 2004.
- [114] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed. A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction. *Journal of Applied Science and Technology Trends*, 1(1):56–70, May 2020.
- [115] A. Zeng, M. Chen, L. Zhang, and Q. Xu. Are Transformers Effective for Time Series Forecasting?, Aug. 2022. arXiv:2205.13504 [cs].
- [116] Y. Zhang, J. Gao, and H. Zhou. Breeds classification with deep convolutional neural network. In *Proceedings of the 2020 12th international conference on machine learning and computing*, pages 145–151, 2020.
- [117] Y. Zhang, C. Zhao, M. Chen, and M. Yuan. Integrating stacked sparse auto-encoder into matrix factorization for rating prediction. *IEEE Access*, 9:17641–17648, 2021.
- [118] T. Zhao, H. Yi, M. Chen, C. Wu, and Y. Xu. Efficient and robust equilibrium strategies of utilities in day-ahead market with load uncertainty. *IEEE Systems Journal*, 16(4):5246–5257, 2022.
- [119] H. zhi Wang, G. qiang Li, G. bin Wang, J. chun Peng, H. Jiang, and Y. tao Liu. Deep learning based ensemble approach for probabilistic wind power forecasting. *Applied Energy*, 188:56–70, 2017.
- [120] P. Zou, Q. Chen, Q. Xia, G. He, and C. Kang. Evaluating the contribution of energy storages to support large-scale renewable generation in joint energy and ancillary service markets. *IEEE Transactions on Sustainable Energy*, 7(2):808–818, 2016.