UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Data Science Curriculum

Carel Kuusk

Hierarchical Forecasting Methods in Day-Ahead Electricity Consumption Forecasting

Master's Thesis (15 ECTS)

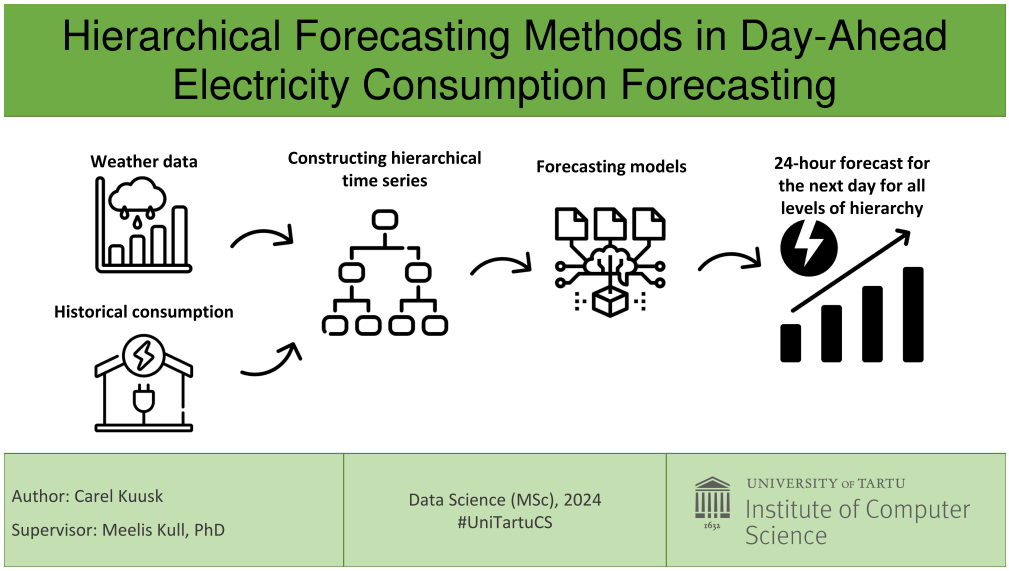
Supervisor: Meelis Kull, PhD

Tartu 2024

Hierarchical Forecasting Methods in Day-Ahead Electricity Consumption Forecasting

Abstract: In various applications, several time series can be organized into one hierarchy, such that lower-level time series can be aggregated into higher-level time series. Forecasting such hierarchical time series requires reconciliation of the final forecasts to ensure that the aggregation constraints present in the original time series are satisfied with the forecasted time series as well. The aim of this thesis is to develop and analyze hierarchical forecasting methods in the context of hourly electricity consumption time series. As a result, hierarchical models based on LightGBM and ridge regression are developed, and their performance is analyzed. Two complex linear reconciliation methods – OLS and Minimal Trace (MinT) reconciliation – are compared against the bottom-up method, and the severe limitations of the OLS and MinT approaches are discovered. Limitations arise due to the electricity consumption forecasting error covariance structure. However, the analyzed reconciliation methods can be used to find forecasts for intermediary levels in the hierarchy.

Visual abstract:



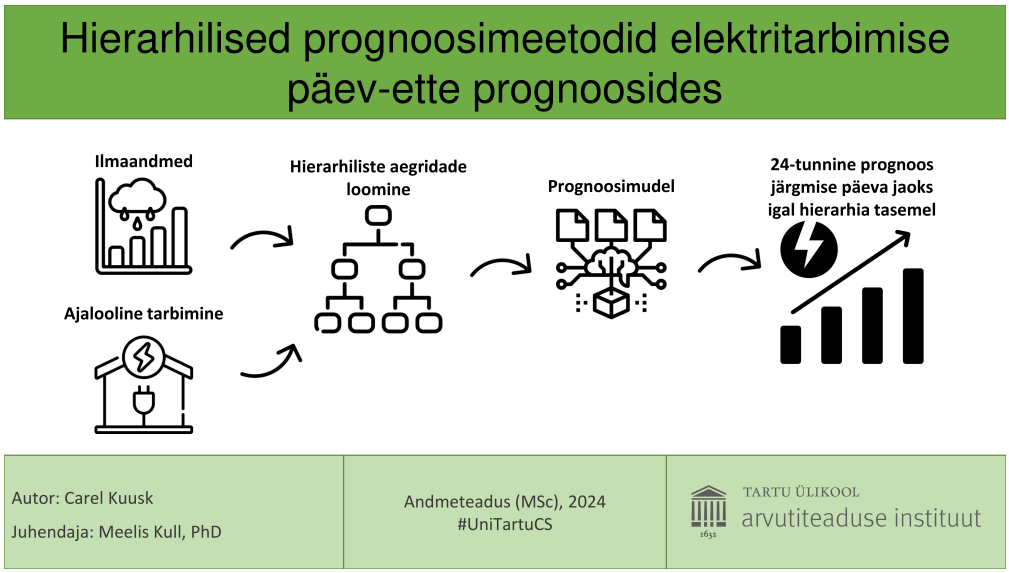
Keywords: time series forecasting, hierarchical forecasting, day-ahead forecasting, LightGBM, ridge regression

CERCS: Artificial intelligence (P176)

Hierarhilised prognoosimeetodid elektritarbimise päev-ette prognoosides

Lühikokkuvõte: Paljudes rakendustes on võimalik mitu aegrida organiseerida ühte hierarhiasse nii, et hierarhia alumiste tasemete aegread saab agregeerida kõrgema taseme aegridadeks. Selliste aegridade prognoosid tuleb omavahel sobitada, et garanteerida prognoositavates aegridades olevate agregatsioonitingimuste täitmine ka aegridade prognoosides. Selle magistritöö eesmärk on arendada ja analüüsida hierarhilisi prognoosimeetodeid elektritarbimise tunnipõhiste aegridade jaoks. Tulemusena on välja töötatud ja analüüsitud LightGBM ja kantregressiooni mudelitele põhinevad hierarhilised mudelid. Kaks keerulisemat lineaarse sobitamise meetodit – OLS ja minimaalse jälje meetod (MinT) – on võrreldud alt-üles sobitamise meetodiga, mille käigus OLS ja MinT lähenemisele on leitud olulised puudujäägid. Puudujäägid tulenevad elektritarbimise prognoosivigade kovariatsioonistruktuurist. Samas, sobitamise meetodeid saab kasutada, et leida prognoose vahepealsetele tasemetele hierarhias.

Visuaalne kokkuvõte:



Võtmesõnad: aegridade prognoos, hierarhiline prognoos, päev-ette prognoos, LightGBM, kantregressioon

CERCS: Tehisintellekt (P176)

Contents

1 Introduction **5**

2 Estonian Electricity Market **7**
2.1 Design of the Electricity Market 7
2.2 Historical, Current and Future Trends 8

3 Methodology **10**
3.1 Hierarchical Time Series 10
3.2 Hierarchical Forecasts 12
3.2.1 Bottom-Up Reconciliation 13
3.2.2 Minimal Trace Reconciliation 14
3.2.3 OLS Reconciliation 15
3.3 Base Models 16
3.3.1 Base Model Selection 16
3.3.2 Ridge Regression 17
3.3.3 LightGBM 18
3.4 Evaluation Methodology 19
3.4.1 Metrics 19
3.4.2 Expanding Window Approach 20

4 Experiments **21**
4.1 Software and Hardware 21
4.2 Data 21
4.3 Results 21
4.3.1 Experiment Design 21
4.3.2 Base Forecast Results – Bottom-Up Approach 22
4.3.3 Reconciliation Results 24
4.3.4 Analysis of Reconciliation Methods 26
4.3.5 Intermediary Aggregations 28
4.4 Discussion and Future Research 30

5 Conclusion **32**

References **36**

Appendix **37**
I. Features 37
II. Licence 38

1 Introduction

Electricity system must be continuously maintained to ensure balance between consumed and produced electricity. In Estonia, Elering is the system operator who ensures that the export and consumption of electricity within Estonia matches the production in Estonia and imports from neighboring countries. If there is an imbalance between production and consumption, the grid frequency will change, which in turn might negatively affect electricity equipment and appliances connected to the grid.

Day-ahead forecasts of consumption and production are an important tool to maintain this balance. The forecasts are collected from all the balance providers, who must generate forecasts for the total consumption and production in their portfolio. Based on these forecasts, Elering plans, monitors, and manages the operations of the electricity system. In addition to system planning, the electricity market is opened, and all the market participants must buy hourly electricity volumes from the market.

The electricity system has already seen massive changes in recent years, with the main impact being a several-fold increase in renewable generation. Although this thesis focuses on forecasting electricity consumption, the increased renewable generation affects market prices, introducing the need for better day-ahead forecasts for consumption portfolios as well. And more changes are on their way, the biggest and most immediate being the desynchronization of the Baltic electricity system from the Russian and Belarusian systems. The final impact of desynchronization is unknown, but it will likely increase the cost of prediction errors. From a consumer behavior perspective, current trends and subsidies point towards increased electric vehicle adoption and storage solutions.

The system operator and market operators are primarily interested in the overall portfolio-level forecasts from the balancing providers. However, balancing providers themselves are interested in the distribution of the cost of forecasting between different client and product segments within their portfolios. Hierarchical time series forecasting methodologies are developed for exactly this kind of problem. The total consumption portfolio can be converted into a hierarchy, where each level in the hierarchy contains information about a particular conceptual feature. Each node in the hierarchy has an hourly consumption time series. Hierarchical time series methodologies are developed to extract forecasts at all levels while maintaining accuracy.

The main objective of this thesis is to develop a hierarchical time series forecasting model for day-ahead electricity consumption on the data of Eesti Energia's consumption portfolio. All data is hourly and can be split into two conceptual types – consumption time series for different disaggregations of the portfolio, and weather parameters. Different model configurations are explored, hierarchical methodologies are investigated, and restrictions arising from the data and reconciliation methodologies are analyzed.

This thesis is separated into three sections. First, a more detailed overview of the Estonian electricity market is given, to motivate the need for improved day-ahead forecasts in Section 2. In Section 3 methodology is introduced. This includes introducing

the concepts and theory of hierarchical time series forecasting used in this thesis, with the most important equations and derivations explained in the text. Base models and their configuration parameters are explained next. Lastly, the validation methodology and metrics used to select the best forecasting models are explained.

In Section 4 the experiments are described. First, the input data and its limitations are described, along with a high-level overview of how the data is distributed. Then, hierarchical time series are constructed, and theoretical descriptions are matched with the experimental reality. Most importantly, the results are demonstrated, with the results analyzed by the different analyzed hierarchical forecast methodologies. The limitations of the data and different methodologies are investigated and explained. Before the work in this thesis is concluded, numerous avenues of future research are presented.

2 Estonian Electricity Market

In this section, a short overview of the current Estonian electricity market is given to give a general background of the functioning of the market and the necessity of day-ahead electricity forecasts.

2.1 Design of the Electricity Market

The electricity market in Estonia was opened for large business customers in 2010 and for smaller business and all the private customers in 2013 [tur12]. The market opening meant that customers were able to choose their electricity providers, who had the option of providing different electricity products. The main change for the end customer was the price formation, which was now determined by market forces, whether via the fixed financial instruments or the daily updated hourly Nord Pool prices, which are often called Nord Pool spot prices, or just spot prices [Ele22a].

The regulation governing the Estonian electricity market is the Electricity Market Law [ele24a]. A more accessible description of the details of the functioning of the electricity market is provided by the Estonian electricity system operator Elering in the form of the Electricity Market Handbook [Ele22a]. The handbook covers all the aspects of how the current electricity market operates.

In general, the market can be divided into four separate markets: financial markets, where the month-long contracts are traded and which determine the price of fixed contracts for customers; day-ahead market, where the day-ahead orders are inserted and where the spot contract prices are determined; intra-day market, where institutional market participants can adjust their day-ahead orders; and finally, regulating market, where the system operator ensures that production and consumption remain in balance [Ele22a].

To reduce the need to understand and operate the complex market trading requirements, the obligation of ordering the electricity volumes from the market is delegated to the electricity providers in standard electricity supply contracts [Ele22a]. The electricity providers aggregate their clients into one or several portfolios and make the orders to day-ahead or intra-day markets for the whole portfolio at once. To make the orders, the providers need to know the expected consumption or production of their portfolios – this is the primary motivation behind the need for consumption forecasting for the providers.

The cost of erroneous forecasts can be quite high and arises primarily from incorrect day-ahead forecasts. For market participants, their imbalance volume is determined as a difference between their fixed supply and the measured supply. Fixed supply is the combination of day-ahead orders and intra-day orders, and measured supply is the actually consumed or produced electricity that reaches the grid. The cost of the imbalance volume – imbalance price – is determined in the regulating market, and the providers are obligated to buy the missing volume, or sell the surplus volume at the imbalance price

[Ele22a]. Since the liquidity at the intra-day market is often limited, with the intra-day trading volume being only roughly 4% of the total trading volume in Nord Pool markets, the accuracy of the day-ahead forecasts is of primary importance [Poo22].

2.2 Historical, Current and Future Trends

The electricity market in Estonia is currently divided between 12 different balance providers, of which the largest is Eesti Energia, whose market share in 2023 fluctuated between 50% and 60%, as can be seen in Figure 1 [Ele24b]. The fluctuations mainly arise from two factors. Firstly, clients can move between different providers, switching due to better prices, contract conditions, or other criteria. Secondly, the portfolio composition might differ between the providers. When the portfolio of one balance provider is more sensitive to temperature changes, the market share of this provider should increase during winter and decrease during summer, in line with temperature fluctuations.

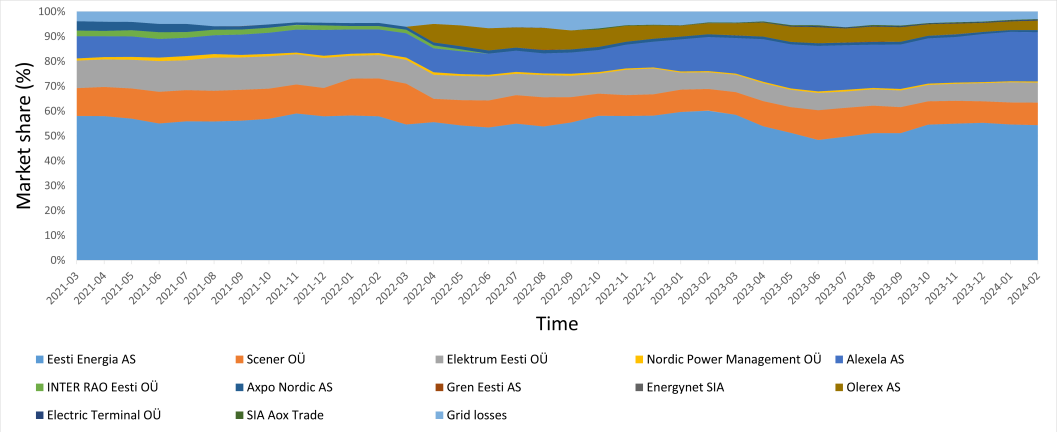


Figure 1. Market shares of balance providers in Estonia from March 2021 to February 2024. Data from [Ele24b].

Since this thesis is done based on consumption data of a significant part of the consumption portfolio of Eesti Energia, the high market share of Eesti Energia is extremely relevant. The system operator target for maintaining whole country imbalance is +/- 30 MWh [Ele23b], which translates to about 2-6% of daily peak consumption. Due to the dominant market share of Eesti Energia's consumption portfolio, the high errors in day-ahead forecasts immediately affect the whole system imbalance. The imbalance settlement methodology is unified across the Baltic states, and since the transmission grids of the Baltic countries are tightly coupled, the system imbalance is strongly influenced by other Baltic states as well [AtA22b].

The boom of renewable generation has brought immense changes to the electricity system of Estonia and the Baltic states. The biggest change has been a massive increase in installed capacities and produced volumes of solar panels. From the security of supply reports by Elering in 2021 [Ele21], 2022 [Ele22b] and 2023 [Ele23a] we can see that in 2020 the solar production volume was not even mentioned, by 2022 the peak generation power was already 383 MW and by 2023 the peak power was 526 MW. Similarly in Lithuania, where the installed capacities of wind power plants have exploded from 540 MW in 2021 to 1288 MW in 2024, and solar power plants from 169 MW to 1165 MW [oTSOfEEE24].

Renewable generation generally increases system imbalances, which means higher balancing volumes and more volatile balancing costs [GPB19]. Although higher imbalances due to renewable generation forecast errors do not directly affect the consumption forecast error, the cost on the consumption portfolio due to the forecast error is still increasing. This necessitates investigations into better consumption-side modeling. Even without a future increase in renewable generation, the increased weather dependence of the generation side is set to pose serious challenges for system balancing and increased balancing cost volatility.

In addition to the realistic further increase of renewable generation, the Baltic electricity system is set to see another massive change. Namely, the Baltic power systems will be synchronized with the Central European Synchronous Area (CESA) and desynchronized from the Russian grid. In the process, the Baltic states are integrated into European balancing markets, which in combination are set to complicate existing dynamics of imbalance pricing further [AtA22a]. These changes cut the Baltic electricity system from the massive Russian synchronization area, forcing Baltic states to be able to fully cover the imbalances with the existing connections to the CESA and thus increase uncertainty, necessitating more accurate day-ahead forecasts.

Given the shifts towards higher renewable energy capacities, particularly in unpredictable wind and solar, the forecast accuracy for both generation and consumption becomes even more crucial. The shorter balancing periods will require consumption forecasts to be significantly more precise to avoid costly imbalances. The "Baltic Balancing Roadmap" outlines several steps taken by the system operators to reduce the impact of desynchronization on the balancing market – 15-minute imbalance settlement period, allowing the possibility for more granular forecasts, especially for wind and large solar parks with real-time data flows; implementation of new frequency control products, balancing capacity markets and provisioning of new balancing capacity reserves [AtA22a]. Overall, the electricity system in Estonia is set to see massive changes, and the best way to manage risk from all the changes for consumption portfolios is to ensure high day-ahead forecast accuracy.

3 Methodology

The final day-ahead forecasts are made for the whole portfolio consumption, but there are multiple ways to achieve these forecasts. The most direct approach is to forecast the whole portfolio consumption directly, which has the least variance. However, this could obscure valuable insights about the accuracy and costs associated with different segments of the portfolio, defined by location, product type, or client type. We can structure the consumption data into a hierarchical time series to preserve this level of detail, allowing for the natural extraction of data at various levels.

Since there is just one forecast sent to the market authorities per portfolio, the cost is the same for the whole portfolio, but different client segments and product types do not have the same contribution to the overall cost. Some segments are more easily forecasted and thus contribute less, while others are more volatile. Hierarchical forecasting techniques provide a natural way to distribute costs fairly between different business segments.

3.1 Hierarchical Time Series

In this section, the online version of the book "Forecasting: Principles and Practice" by Hyndman and Athanasopoulos is used, unless stated otherwise [HA21]. Many time series can be decomposed into smaller, more specific time series, where the decomposition happens by some natural or imposed feature. Such decomposing is called disaggregation. In the case of electricity consumption forecasting, the most natural example would be disaggregation by location, e.g., by county or city level. If, after disaggregation, all the components sum back up to the original aggregated time series, we achieve a simple hierarchically organized time series.

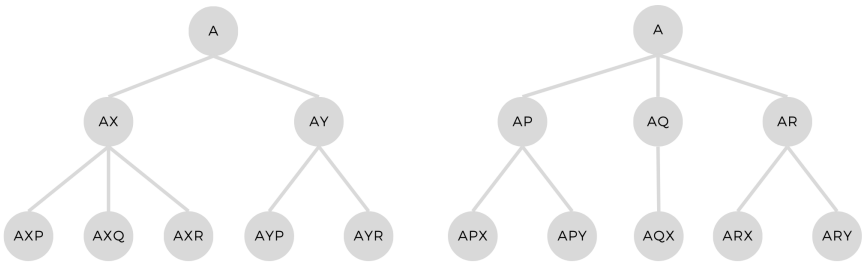


Figure 2. Two simple example hierarchies, where the only difference is ordering the bottom two levels.

Figure 2 shows a simple example of a hierarchy. The hierarchy has three *levels*, and each node represents one time series. In the context of this thesis, each individual time

series is also called a *segment*. The time series are represented by $y_{x,t}$, where the first index identifies the time series and the second index the time period t. For each time period t, the value for each time series is the sum of its child time series values for that time period. In the example shown in Figure 2, we get three sums that must hold for each time period:

$$\begin{aligned} y_{A,t} &= y_{AX,t} + y_{AY,t}, \\ y_{AX,t} &= y_{AXP,t} + y_{AXQ,t} + y_{AXR,t}, \\ y_{AY,t} &= y_{AYP,t} + y_{AYR,t}. \end{aligned} \tag{1}$$

For disaggregations, there is a natural order. For example, each consumption measuring point has a coordinate, but we can aggregate all the measuring points in one city together, then all the measuring points in one county, and finally all the counties to achieve whole country consumption. There is a natural hierarchy in this partition – it does not make sense to order the city level as higher than the county level. In other cases, the levels are switchable. Keeping with the electricity consumption example, contract type, and client type are interchangeable as levels – we can look at business clients regardless of contract type, or we can look at all the customers with fixed contracts, regardless of who they are.

The impact of such *crossed* levels is also shown in Figure 2. The right and left hierarchies contain the same levels, but the bottom two levels are switched. As we can see, the effect of the switch is only visible in the middle layers, where AX and AY in the left hierarchy are not found in the right hierarchy. However, for the bottom layer, the time series corresponding to the AXP in the left hierarchy is the same as the time series APX in the right. The corresponding hierarchical equations are changed to these:

$$\begin{aligned} y_{A,t} &= y_{AP,t} + y_{AQ,t} + y_{AR,t}, \\ y_{AP,t} &= y_{APX,t} + y_{APY,t}, \\ y_{AQ,t} &= y_{AQX,t} + y_{AQY,t}, \\ y_{AR,t} &= y_{ARX,t} + y_{ARY,t}. \end{aligned} \tag{2}$$

The structure of the hierarchy defines a *summing matrix*, denoted by $\mathbf{S}$. The summing matrix is derived from the observation that all the higher levels can be represented as a sum of the bottom-level time series. The summing matrix is an $n \times m$ matrix of zeros and ones, where n is the total number of time series and m is the number of bottom-level time series. In the case of the hierarchies shown in Figure 2, $m = 5$ for both and $n = 8$ for the left and $n = 9$ for the right. The way the summing matrix behaves is shown in equation 3.

$$\begin{bmatrix} y_{A,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AXP,t} \\ y_{AXQ,t} \\ y_{AXR,t} \\ y_{AYP,t} \\ y_{AYR,t} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_{AXP,t} \\ y_{AXQ,t} \\ y_{AXR,t} \\ y_{AYP,t} \\ y_{AYR,t} \end{bmatrix} \tag{3}$$

More compact way to represent this is $\mathbf{y}_t = \mathbf{S}\mathbf{b}_t$, where we have defined the $\mathbf{y}_t$ to be the vector of all the time series at time period t in the hierarchy and $\mathbf{b}_t$ as a vector of all the bottom-level time series at time t.

3.2 Hierarchical Forecasts

The eventual goal of this thesis is to generate accurate forecasts for the time series in the hierarchy. The primary interest is the forecast for the highest level ($y_{A,t}$ in the example) since this forms the basis of the day-ahead orders going to the Nord Pool markets and ultimately determines the imbalance costs for the whole portfolio. But intermediary-level forecasts are interesting as well – being able to determine more accurately how accurate the forecasts would be for different client or product types is an important input for pricing, product development, and marketing efforts.

Forecasts can be made for each individual time series in the hierarchy and such forecasts are denoted by hatted variables, like $\hat{y}_{A,t+h|t} = \hat{y}_{A,h}$, where the index h shows how many time steps into the future the forecast is made. For clarity of notation the time indices t are dropped, unless required by the context. The vector of the forecasts for individual time series in the hierarchy is denoted as $\hat{\mathbf{y}}_h$ and is called the vector of base forecasts, or just *base forecasts*.

However, since the forecast is made independently for each time series, the hierarchical summing conditions defined in equations 1 and 2 might not hold for the respective forecasts and the forecasts are not *coherent*. The process of generating coherent forecasts from the base forecasts is called *reconciliation*.

The coherency of forecasts is better explained with an example from the left graph of Figure 2. For the time series themselves, we know that $y_{AX,t} + y_{AY,t} = y_{A,t}$ for all t. Let us know assume that the forecasts of AX and AY made at time $t = T$ are perfectly accurate, that is $\hat{y}_{AX,h} = y_{AX,T+h}, \hat{y}_{AY,h} = y_{AY,T+h}$. If the forecast for the top level time series A is not perfectly accurate $\hat{y}_{A,h} \neq y_{A,T+h}$ (which is a possibility, since the forecasts for each time series are independent of other forecasts), the first equation of 1 does not hold for the forecasts, and thus the forecasts are not coherent:

$$\hat{y}_{A,h} \neq \hat{y}_{AX,h} + \hat{y}_{AY,h}. \tag{4}$$

The three investigated methodologies all follow similar principles. First, the base forecasts $\hat{\mathbf{y}}_h$ are compressed by a matrix P with dimensions $m \times n$ into bottom-level reconciled forecasts. After that, the summing matrix S is applied to get reconciled forecasts for all the levels in the hierarchy $\tilde{\mathbf{y}}_h$:

$$\tilde{\mathbf{y}}_h = \mathbf{SP}\hat{\mathbf{y}}_h \tag{5}$$

The role of the *mapping matrix* P is to generate new bottom-level forecasts by linearly combining the base forecasts of all levels of the hierarchy. Due to the application of the summing matrix S, the forecasts $\tilde{\mathbf{y}}_h$ are guaranteed to be coherent, and we automatically achieve reconciled forecasts for all the time series in the hierarchy. The hierarchical forecasting problem can now be split into two: generating the base forecasts $\hat{\mathbf{y}}_h$ and finding the best possible matrix P.

In the end, all three methods are linear reconciliation methods, meaning that the reconciled forecasts are calculated as a linear combination of the base forecasts. The difference between the methods is the calculation methodology for the weights. The bottom-up and OLS reconciliation only uses the structure of the hierarchy to calculate the weights, with the structure being encoded in the summing matrix S. Bottom-up reconciliation makes no extra assumptions about the data, but the OLS method sets restrictions on the forecast error distribution. Minimal trace reconciliation uses forecast error covariance information in calculating the weights but is the most general of the three methods.

3.2.1 Bottom-Up Reconciliation

The simplest possible approach to achieve reconciled forecasts is to take the bottom-level forecasts $\hat{\mathbf{b}}_h$ and sum them up to all the other levels using the summing matrix $\mathbf{S}$, in which case we need to set the first $n - m$ rows of the mapping matrix to zero, and the rest is an identity matrix:

$$\tilde{\mathbf{y}}_h = \mathbf{SP}\hat{\mathbf{y}}_h = \mathbf{S}\hat{\mathbf{b}}_h \tag{6}$$

In equation (6) the mapping matrix is $\mathbf{P} = [\mathbf{0}_{m\times(n-m)} \vdots \mathbf{I}_m]$ [SLWH19], with $\mathbf{I}_m$ being the $(m \times m)$ identity matrix and $\mathbf{0}_{m\times(n-m)}$ being a matrix of zeros with dimensions $(m \times (n - m))$. The notation $C = [A_{x\times y} \vdots B_{x\times z}]$ is adopted from [SLWH19] and shows that the first y columns of the matrix C are filled with elements of matrix A and the next z columns are filled with elements of matrix B.

The bottom-up approach automatically guarantees that the forecasts at all levels are coherent, but the trade-off is potentially losing information from higher levels. Another similarly simple approach is to use top-down forecasts, where only the top-level series is forecasted. In such case the mapping matrix $P = [\mathbf{p} \vdots \mathbf{0}_{m\times(n-1)}$, where $\mathbf{p}$ is a set

of proportions $\mathbf{p} = [p_1, p_2, \ldots, p_m]$ that distributed the top-level base forecasts to the bottom level, which are then summed by S to return top-down forecasts [SLWH19].

The bottom-level forecasts are found via a set of proportions that distribute the base forecasts of the top level to the bottom level [HA21]. There are theoretical studies dating back to the 1960s regarding the relative merits of either bottom-up [GG60] or top-down approaches [OWE68], with later empirical literature suggesting the relative advantage of bottom-up forecasts [HAAS11]. In this thesis, only the bottom-up approach is analyzed, since electricity consumption profiles can be completely different from segment to segment and cannot reliably be determined as a simple percentage of the whole country forecasts.

3.2.2 Minimal Trace Reconciliation

Bottom-up and top-down approaches are the simplest possible reconciliation methods, but they can be improved under certain conditions. Wickramasuriya et al. [SLWH19] were able to find a mapping matrix $\mathbf{P}$, such that it minimizes the sum of error variances for the reconciled forecasts. One option is to calculate the covariance matrix of reconciled forecast errors and minimize its trace, earning the method the name Minimal Trace Reconciliation, or MinT.

First, let us fix our current time period to $t = T$, and from now on, all the estimations and forecasts are conditional on the data known only up to time T. Forecasts are called unbiased if the expected values of the forecasts and corresponding true values of the time series are equal, that is

$$E[\hat{\mathbf{y}}_{T+h|T}] = E[\mathbf{y}_{T+h}]. \tag{7}$$

The following derivation for the universal unbiasedness condition follows Hyndman et al. [HAAS11]. Let us assume that equation (7) holds. As the bottom level forecasts $\hat{\mathbf{b}}_h$ are a subset of the base forecasts $\hat{\mathbf{y}}_h$, they are also unbiased. From the definition of unbiased forecasts, the reconciled forecasts are unbiased if $E[\tilde{\mathbf{y}}_h] = E[\mathbf{y}_{T+h}]$. We can expand $\tilde{\mathbf{y}}_h$ by using (5) and use linearity of expectation to get

$$E[\mathbf{y}_{T+h}] = E[\tilde{\mathbf{y}}_h] = E[\mathbf{SP}\hat{\mathbf{y}}_h] = \mathbf{SP}E[\hat{\mathbf{y}}_h]. \tag{8}$$

Let us now define $\boldsymbol{\beta}_h = E[\mathbf{b}_{T+h}]$ to be the unknown true mean of the bottom level time series at time period $T+h$. From the definition of the summing matrix S $\mathbf{y}_{T+h} = \mathbf{S}\mathbf{b}_{T+h}$, so $E[\mathbf{y}_{T+h}] = \mathbf{S}E[\mathbf{b}_{T+h}]$. Substituting this to the left side of equation (8) we get

$$\mathbf{S}E[\mathbf{b}_{T+h}] = \mathbf{SP}E[\hat{\mathbf{y}}_h] \tag{9}$$

. On the other hand $\hat{\mathbf{y}}_h = \mathbf{S}\hat{\mathbf{b}}_h$, again from the definition of $\mathbf{S}$, or $E[\hat{\mathbf{y}}_h] = \mathbf{S}E[\hat{\mathbf{b}}_h]$, so by substituting this to equation (9) we get

$$\mathbf{S}E[\mathbf{b}_{T+h}] = \mathbf{SPS}E[\hat{\mathbf{b}}_h]. \tag{10}$$

The derivation started with the assumption that the base forecasts are unbiased. Since the bottom level forecasts $\hat{\mathbf{b}}_h$ are a subset of the base forecasts $\hat{\mathbf{y}}_h$, they are also unbiased: $E[\mathbf{b}_{T+h}] = E[\hat{\mathbf{b}}_h]$. From equation (10) we then get the unbiasedness condition

$$\mathbf{S} = \mathbf{SPS}. \tag{11}$$

Since no assumptions were made regarding the mapping matrix $\mathbf{P}$, this condition is universal and holds for all $\mathbf{P}$ in linear reconciliation approaches. The unbiasedness of reconciled forecasts is assumed in all the derivations in this section and Section 3.2.3.

Let us define the base forecast errors by

$$\hat{\mathbf{e}}_h = \mathbf{y}_{t+h} - \hat{\mathbf{y}}_{t+h|t}, \tag{12}$$

and reconciled forecast errors by

$$\tilde{\mathbf{e}}_h = \mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t}. \tag{13}$$

Let $\mathbf{W}_h$ be the covariance matrix of the base forecast errors. Wickramasuriya et al. [SLWH19] proved that the reconciled forecast errors can be found from the base forecast errors using the same transformation: $\tilde{\mathbf{e}}_h = \mathbf{SP}\hat{\mathbf{e}}_h$. From this, we get that the covariance matrix of the reconciled forecast errors in (13) is given by

$$\text{Var}(\tilde{\mathbf{e}}_h) = \text{Var}(\mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t}) = \mathbf{SPW}_h\mathbf{P}^T\mathbf{S}^T. \tag{14}$$

Finally we need to find matrix $\mathbf{P}$ such that it minimizes the trace of $\text{Var}(\mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t})$, conditional on the unbiasedness assumption $\mathbf{SPS} = \mathbf{S}$ from equation (7). Wickramasuriya et al. [SLWH19] show that the optimal reconciliation matrix is given by

$$\mathbf{P} = (\mathbf{S}^T\mathbf{W}_h^{-1}\mathbf{S})^{-1}\mathbf{S}^T\mathbf{W}_h^{-1}. \tag{15}$$

Thus the reconciled forecasts are given by

$$\tilde{\mathbf{y}}_h = \mathbf{SP}\hat{\mathbf{y}}_h = \mathbf{S}(\mathbf{S}^T\mathbf{W}_h^{-1}\mathbf{S})^{-1}\mathbf{S}^T\mathbf{W}_h^{-1}\hat{\mathbf{y}}_h. \tag{16}$$

As we can see, the reconciliation is data-dependent; to find the mapping matrix $\mathbf{P}$, we first need to calculate the covariance matrix of the base forecast errors.

3.2.3 OLS Reconciliation

Hyndman et al. [HAAS11] proposed a simpler reconciliation method that considers only the structure of the hierarchy encoded in the summing matrix $\mathbf{S}$. The core idea of the original derivation is to treat base forecasts as the target variable of a linear regression equation, earning the method the name OLS reconciliation method.

The main additional assumption made to reach the final form of mapping matrix $\mathbf{P}$ is that the errors of base forecasts approximately satisfy the same aggregation constraint as the time series themselves. This means that the errors of higher-level base forecasts are derivable from the errors of the bottom-level forecasts [HAAS11]. If we define $\varepsilon_{b,h} = \mathbf{y}_{t+h} - \hat{\mathbf{y}}_{t+h|t}$ as the errors of bottom-level forecasts, then this restriction can be expressed as

$$\varepsilon_h \approx \mathbf{S}\varepsilon_{b,h}. \tag{17}$$

Using this assumption Hyndman et al. [HAAS11] were able to show that $\mathbf{P}$ takes the form that depends only on the structure of the hierarchy, and not the errors of base forecasts, resulting in

$$\mathbf{P} = (\mathbf{S}^T\mathbf{S})^{-1}\mathbf{S}^T, \tag{18}$$

from which

$$\tilde{\mathbf{y}}_h = \mathbf{S}\mathbf{P}\hat{\mathbf{y}}_h = \mathbf{S}(\mathbf{S}^T\mathbf{S})^{-1}\mathbf{S}^T\hat{\mathbf{y}}_h. \tag{19}$$

Comparing equations (15) and (18) we can see that they have the same overall structure. We can get equation (18) from (15) by taking $\mathbf{W}_h = k_h\mathbf{I}_n$, with $k_h > 0$ is a positive constant [SLWH19].

3.3 Base Models

3.3.1 Base Model Selection

While reconciling hierarchical forecasts, the internal workings of the base models are not directly relevant. However, the base models must be accurate and reliable to achieve satisfactory overall results. For this reason, the base models must be carefully selected and analyzed, because the quality of the hierarchical forecasts ultimately depends on the accuracy of these underlying models.

There are numerous methods specifically tailored for time series forecasting, from simple historical averaging to complex neural network algorithms. Classical time series forecasting methods utilize statistical methods and include moving averages, exponential smoothing, and ARIMA models, which, despite their relative simplicity, have been shown to be competitive against the more modern machine learning approaches across a variety of time series [MSA20]. It is also possible to use regression as a time series forecasting method when the future values of the time series are considered as a function of its current and past values. Other variables can also be used as features, whether they are forecasts from some other models (e.g., weather forecasts), known future values (e.g., time values, client counts), or lagged values of non-dependent variables [HA21].

Electricity consumption is highly periodic, with clear patterns repeating daily, weekly, and seasonally. This periodicity allows for applying regression models with lagged values in addition to exogenous or explanatory variables. In Figure 3, the Estonian consumption data is shown for two different time periods. On the left, the consumption for February

2023 is graphed, and the weekly period is clearly visible, with clear peaks during daytime and troughs during nighttime. In addition, there is dependence on weekdays, with clearly higher daily consumption during workdays. From the right graph of Figure 3 the yearly pattern is visible – there is a clear dependence on temperature, with higher consumption occurring during winter months.

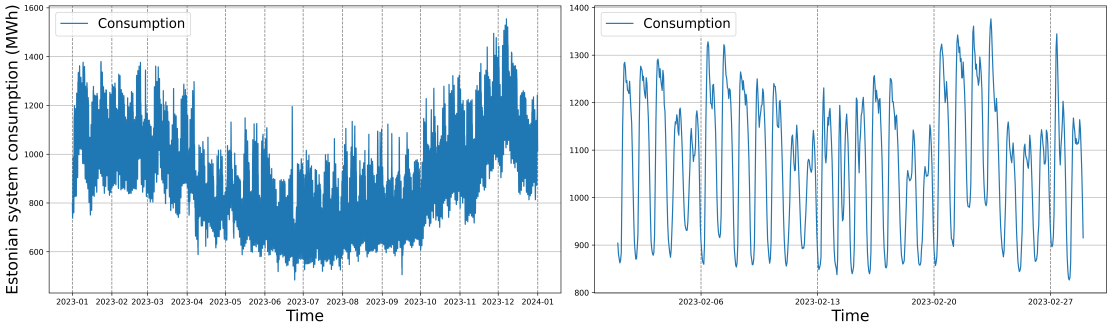


Figure 3. Example profiles of hourly energy consumption in Estonia. The left graph shows the consumption over the year 2023, and the right graph shows the hourly consumption in February 2023. Data from Elering Live Dashboard [Ele24c].

In 2020, the M5 forecasting competition was held on the Kaggle platform [HMS20]. The competition task was a hierarchical forecast on the Walmart retail sales dataset. The competition data covered data from three US States and had several hierarchical levels, including item level, department, and product category levels. The final submission had to be a reconciled forecast for all the levels, and the evaluation was based on the weighted average root mean squared scaled error across all the time series. Several of the top models in the competition, including the winning submission, utilized LightGBM models, which is why this is one of the models considered in this thesis [MSA22]. The second type of selected model is simple ridge regression, chosen for its simplicity, rapid training times, and robust performance with correlated features.

3.3.2 Ridge Regression

Ridge Regression, is an extension of ordinary linear regression that is particularly useful for addressing multicollinearity between predictor variables. It modifies the object function of the least squares method by adding a penalty term proportional to the square of the magnitude of the coefficients.

To see how the ridge regression works we start from the standard multiple regression problem $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}$, where $\mathbf{y}$ is the target variable and $\boldsymbol{\beta}$ the coefficient vector. The simplest method, ordinary least squares, finds the coefficient vector $\boldsymbol{\beta}$ by defining the objective function $(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$, which is then minimized with respect to $\boldsymbol{\beta}$.

This results in the coefficients $\boldsymbol{\beta}$, minimizing the residual sum of squares between the targets and the predicted values.

Ridge regression modifies the objective function, by adding a term proportional to the sum of squares of the coefficients. This sum of squares is equivalent to the L2 norm of the coefficients, which is why the added term is called the L2 regularization term. The minimization task then becomes

$$\min_{\boldsymbol{\beta}}\{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})+\alpha\boldsymbol{\beta}^T\boldsymbol{\beta}\}. \tag{20}$$

Here the α is the regularization parameter that controls the regularization strength. If α approaches zero, the minimization function becomes closer to that of OLS, showing that OLS is a special case of ridge regression. In general, the effect of increasing the regularization α is reducing the absolute magnitude of the coefficients. This effect of the regularization is also the reason why ridge regression works well with collinear features – in the presence of collinear features, OLS solutions can become unstable and overly sensitive to small changes in the model input, which results in large variance of the coefficient estimates. The damping effect of the regularization term forces coefficients smaller and reduces the variance. [HK70]

3.3.3 LightGBM

LightGBM, or Light Gradient Boosting Machine is a specific implementation of the gradient boosting framework. Gradient boosting is an ensemble machine learning technique that builds a model by sequentially adding multiple weak learners, which in the case of LigthGBM are decision trees. The objective is to minimize the predictive errors by iteratively adding new trees in a way that minimizes the objective function [Fri01].

The article describing the XGBoost, another gradient boosting framework, gives a concise overview of the gradient boosting process [CG16]. The objective function quantifies the predictive error, as in ridge regression. The next tree is added at each iteration step to minimize the objective function. At each iteration, the gradient of the loss function is calculated with respect to the prediction of the current model. The new tree is then fitted to these gradients, thus essentially trying to predict the loss gradient at each point of the data.

However, the problem with this approach is that to fit the best decision tree, all the data points must be scanned, which is a massive computational cost. Following the article introducing the LightGBM method [KMF$^+$17], the core ideas to tackle this problem are described in the following paragraphs.

The first of the strategies is a histogram-based algorithm that divides feature values into histograms [RS98]. The best split points are based on the histograms, not the individual data points, and since the number of histogram bins is considerably less than the number of data points, the training speed is substantially increased.

Of the original contributions the first technique employed by LightGBM is Gradient-based One-Side Sampling (GOSS). The gist of the method is to focus on data points that contribute most significantly to the information gain. GOSS selects all the data instances with large gradients and randomly samples a fixed proportion of the instances with small gradients. In addition, to focus more on the under-trained instances, the sampled data with small gradients is amplified by a factor dependent on the sampling proportion.

LightGBM utilizes Exclusive Feature Bundling (EFB) to handle high-dimensional sparse data more efficiently. EFB groups mutually exclusive features, which rarely take nonzero values simultaneously into a single feature. This bundling reduces the feature space's complexity, leading to faster computation and less memory usage without a significant loss in model accuracy.

In addition, the Python LightGBM implementation implements several techniques to reduce the potential for overfitting and enhance generalization. The simplest is the number of boosting iterations which defines the number of trees to be built, with a higher number of iterations resulting in a more precise but potentially overfitting model. There is also a shrinkage, or learning rate parameter, that regularizes the contribution of new models in each boosting iteration [Fri02].

Bagging techniques [Bre96] are also implemented, for both features and samples. Feature fraction determines the fraction of features to be used for each boosting iteration, and is especially useful for datasets with high feature count. Similarly, the bagging fraction determines the percentage of the whole dataset used in specified iterations. The iterations where to apply bagging are defined by the bagging frequency parameter [Lig24].

3.4 Evaluation Methodology

3.4.1 Metrics

The trained base models and reconciled forecasts are evaluated using mean absolute percentage error (MAPE). MAPE shows the average absolute deviation of the forecasts from actual values as a percentage, defined by the formula

$$\text{MAPE} = \frac{100\%}{n} \sum_{i}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|. \tag{21}$$

Here y_i is the actual value for the forecasted period i, $\hat{y}_i$ is the forecast, and n is the number of forecasted periods. Since MAPE has the actual values y_i in the denominator, it cannot be used when the forecasted values are near zero. Intuitively, MAPE works well with relative errors, where the error depends on the scale of the quantity to predict. MAPE is one of the most popular metrics for forecasting tasks in general due to its simplicity and scale invariance. However, the main benefit of MAPE is that it suits other

business processes well, where the margins to cover the costs coming from expected forecast inaccuracies are generally covered as a fixed fee on the true consumption. This makes the allowance for errors higher during higher true consumption periods.

3.4.2 Expanding Window Approach

For validation and test periods, the expanding window strategy was used. This method involves incrementally increasing the training dataset over time and adding newer data points to the training set while performing validation on the next time period. This allows us to use more of the dataset for training while also having a longer time range for validation. From Figure 3, the yearly periodicity of electricity consumption patterns is visible. To catch these yearly variations, the validation should ideally be performed in at least a yearly period.

The availability of the data sets its own limitations, however, so the expanding window approach used in this thesis is illustrated in Figure 4. The training period started in March 2021, and the validation periods from March 2023. For each iteration, the validation period was one calendar month, and the validation period started one month later. Similarly, the training period was expanded with one extra month. The last month in the overall validation period was December 2023, and the first two months of 2024 were used as test periods. The models were similarly trained during the test periods with a monthly expanding training dataset.

The model hyper-parameters were chosen and evaluated during the validation period from March 2023 to December 2023. Due to long training times, a preliminary search of reasonable range for the hyper-parameters was made with about 30 segments before the full hierarchy hyper-parameter search presented in Section 4.3 was performed. The segments chosen for the preliminary search had either high overall consumption or high variance of consumption profile.

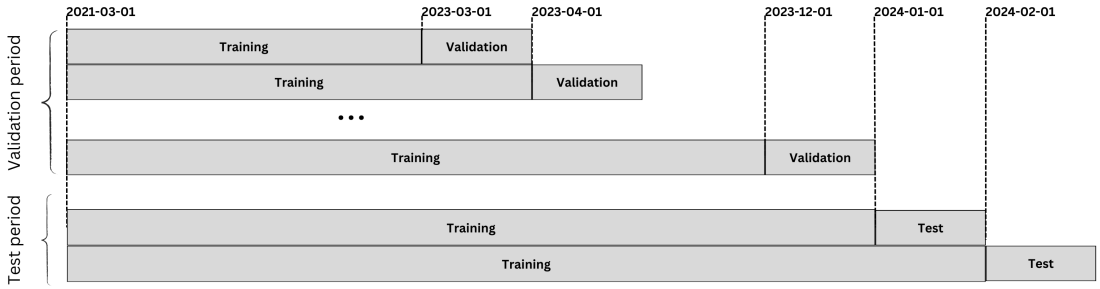


Figure 4. The expanding window validation approach used in this thesis.

4 Experiments

4.1 Software and Hardware

Data collection, wrangling, model training, and analysis were done on the same Windows 11 computer with 32 GB of RAM and an Intel i5-1135G7 processor. All the work was done in the same Python 3.11.4 environment, Jupyter Notebook version 6.0 was used as the main tool for analysis, with *pandas* and *numpy* as the main data wrangling libraries. Two main Python libraries were used for configuring, training, and evaluating models. For the metrics and simpler regression models, the *Scikit-Learn* 1.4.1 library, and as the LightGBM implementation, the *lightgbm* 4.3.0 Python library was used. OpenAI's ChatGPT version 4 was used as a tool to aid in the preparation of some of the data analysis and plotting scripts, in addition as an aid for formatting citations and some paragraphs of the text in the thesis [Ope24].

4.2 Data

This section is redacted.

4.3 Results

4.3.1 Experiment Design

On all the segments, two types of models were tested using different hyper-parameters: ridge regression models and LightGBM models. For both types of models, the same evaluation strategy was used, and a hyper-parameter search was conducted. For ridge regression 12 different L2 regularization terms were tested from 0.001 to 128. For LightGBM parameters, the hyper-parameter search was conducted manually due to the longer training time of the models and larger parameter space.

The models were evaluated with the expanding window strategy as described in Section 3.4.2, with the illustrated strategy shown in Figure 4. Each of the 874 new base models was retrained with the expanded training data and again evaluated in the next month. The last evaluated month was February 2024.

There is one notable exception to the trained models. The models were not trained on 163 bottom-level segments where the size type corresponded to 'tiny'. Instead, week-ago values were used for these segments as forecasts. The reasoning was stability – since many of the individual time series in those segments are very volatile, a trained model would be prone to very unstable forecasts. This also sped up the training process, and since the total volume of those segments is very small by nature, the effect on the overall performance is negligible.

After the base models on all the 874 segments were trained and evaluated hierarchical reconciliation was applied. Hierarchical reconciliation was applied with three different methodologies as described in Section 3.1. The first option is the bottom-up approach, and since the primary goal is the total forecast, we can look at the aggregate to the top level from all the bottom levels. This simplest approach also helps us discover the best hyper-parameters for each level. After that, the reconciliation approaches described in Section 3.2.3 and Section 3.2.2 were applied and analyzed.

The models and their configurations are shown in Table 1. In total, 15 different LightGBM models were tested with the shown hyper-parameter configuration. The rest of the hyper-parameters were not changed from their default values. For ridge regression, 12 different regularization parameter values alpha were tested. The ID column in Table 1 is used to later reference the models in the text.

4.3.2 Base Forecast Results – Bottom-Up Approach

The first step is forecasting all 874 segments at different levels, resulting in the base forecasts, which are later reconciled. It is instructive to look at the results by levels by

Table 1. Model configuration and hyper-parameters

ID	Model	Alpha	Bagging fraction	Bagging frequency	Feature fraction	Iteration number	Learning rate
Model 0	LightGBM	–	0.6	10	0.7	100	0.1
Model 1	LightGBM	–	0.6	10	0.7	50	0.1
Model 2	LightGBM	–	0.6	10	0.7	100	0.05
Model 3	LightGBM	–	0.7	5	0.9	100	0.1
Model 4	LightGBM	–	0.5	5	0.7	100	0.1
Model 5	LightGBM	–	0.7	5	0.8	100	0.1
Model 6	LightGBM	–	0.6	10	0.9	100	0.1
Model 7	LightGBM	–	0.4	5	0.4	100	0.1
Model 8	LightGBM	–	0.5	5	0.7	100	0.2
Model 9	LightGBM	–	0.5	5	0.7	100	0.3
Model 10	LightGBM	–	0.5	10	0.7	100	0.2
Model 11	LightGBM	–	0.6	10	0.9	100	0.2
Model 12	LightGBM	–	0.5	10	0.7	100	0.5
Model 13	LightGBM	–	0.5	20	0.7	100	0.2
Model 14	LightGBM	–	0.6	20	0.7	100	0.1
Model 15	LightGBM	–	0.7	5	0.7	100	0.1
Model 16	LightGBM	–	0.4	5	0.7	100	0.1
Model 17	Ridge	0.001	–	–	–	–	–
Model 18	Ridge	0.01	–	–	–	–	–
Model 19	Ridge	0.1	–	–	–	–	–
Model 20	Ridge	1	–	–	–	–	–
Model 21	Ridge	2	–	–	–	–	–
Model 22	Ridge	4	–	–	–	–	–
Model 23	Ridge	8	–	–	–	–	–
Model 24	Ridge	16	–	–	–	–	–
Model 25	Ridge	32	–	–	–	–	–
Model 26	Ridge	64	–	–	–	–	–
Model 27	Ridge	96	–	–	–	–	–
Model 28	Ridge	128	–	–	–	–	–

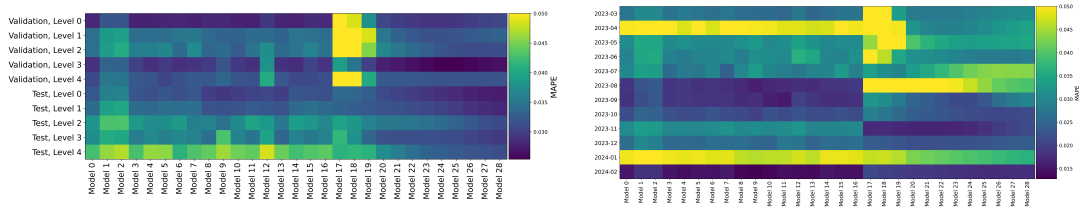


Figure 5. Left graph: heatmap of MAPE values for each model and by taking each level as the bottom-most level. Right graph: heatmap of monthly MAPE values by model when aggregated to the top level from Level 0. In both graphs, the values are clipped at 5% to ensure readability.

summing the forecast to the top-level forecasts, i.e., the whole country forecasts. This results in five different forecasts for the top-level forecast – the top-level base forecast itself and aggregated from different lower levels. The five forecasts are not reconciled but are comparable against each other. For such a procedure, the MAPE metric is calculated for each model, and the result is visualized as a heatmap in Figure 5.

From Figure 5 in the left graph, we can notice a couple of trends. First, with ridge regression models (Models 17-28) it is clearly visible that the regularization parameter is important. When the parameter is set near zero, the validation MAPE goes even over 5%. For ridge regression models, we can clearly see the trend that the higher the regularization parameter, the better the models performed during the test period, while for the validation period, the best regularization parameter is around 16. Similarly, the validation MAPEs for the LightGBM models are slightly better than for the test period, which is analyzed in more detail later in this section.

Another noticeable trend is the difference in performance at different levels. Both models are better at Level 0 and Level 3, with more noticeable improvement at Level 0 for LightGBM and at Level 3 for ridge regression models. This suggests that metering point size (introduced at Level 0) and client type (business or private) are more important for forecasting than product type or location.

A more granular way to look at the data would be to look at the monthly performance of each model, instead of just validation and test periods. In Figure 5 in the right graph this data is shown as a heatmap, where only the Level 0 (bottom level) forecasts are aggregated to the top level. It is clearly visible that different model types have issues with certain months: LightGBM with April 2023, ridge regression with July and August 2023 and both have troubles with January 2024.

Importantly, the worse performance during the test period is not due to simple overfitting, since January is extremely bad for both models while February is extremely good. This result is easily explainable with temperature patterns. As can be seen in Figure 6, the beginning of the year 2024 had extremely cold temperatures, which resulted in high consumption values and high forecasting errors for that period. A complicating

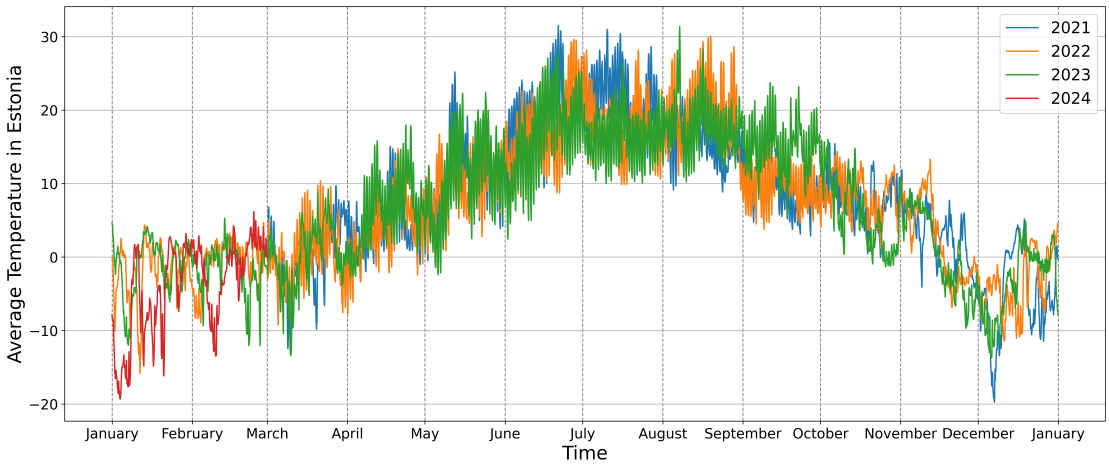


Figure 6. Average hourly temperatures in Estonia during the period covered by the data.

factor was that this period with high temperatures fell to the first week of the year when the consumption pattern, in general, was different from the rest of the year. The fact that the ridge regression model with a high regularization parameter was relatively accurate suggests that very regularized models might be suitable for use during unusual weather periods.

The performance drops in April 2023 for LightGBM, and summer 2023 for ridge regression models are less obvious but suggest that different model architectures might complement each other.

4.3.3 Reconciliation Results

The three methods described in Section 3.2.1, Section 3.2.3, and Section 3.2.2 were applied to the data. In the case of bottom-up reconciliation, the lowest level is Level 0. Since the bottom-up and OLS reconciliation calculations do not depend on the forecasted values, the base-level forecasts are made, and from those, the reconciled forecasts are calculated.

Minimal Trace reconciliation depends on the covariance matrix of the errors of the base forecasts. The estimate of the covariance matrix was found with the expanding window strategy. The base models were trained and evaluated using the expanding window strategy. For each evaluated month in the expanding window strategy, the forecasting errors from March 2023 up to the beginning of the month were used to calculate the covariance matrix. This covariance matrix was then used in the formula (16) to reconcile the forecasts for the month under evaluation. The need to estimate the covariance matrix means that the validation period was one month shorter: from April 2023 up to the end of 2023. To ensure comparability, the validation period for bottom-up

and OLS reconciliation methods was also shortened, the resulting MAPE values are shown in Table 2.

Table 2. MAPE comparison of reconciliation methods.

Model	Validation Bottom-Up	Validation OLS	Validation MinT	Test Bottom-Up	Test OLS	Test MinT
Model 0	2.83%	2.83%	3.24%	3.27%	3.90%	3.62%
Model 1	3.21%	3.30%	3.76%	3.67%	4.26%	3.87%
Model 2	3.21%	3.30%	4.48%	3.70%	4.31%	3.73%
Model 3	2.77%	2.94%	3.25%	3.26%	3.89%	3.62%
Model 4	2.78%	2.89%	3.33%	3.20%	4.09%	3.80%
Model 5	2.80%	2.83%	3.18%	3.26%	4.05%	3.52%
Model 6	2.82%	2.87%	3.29%	3.20%	3.81%	3.36%
Model 7	2.79%	2.90%	3.72%	3.26%	3.92%	3.42%
Model 8	2.84%	2.88%	3.62%	3.05%	3.90%	3.21%
Model 9	2.85%	2.94%	3.48%	2.99%	4.19%	3.46%
Model 10	2.78%	2.85%	3.74%	3.04%	3.97%	3.46%
Model 11	2.79%	2.83%	3.14%	3.08%	3.84%	3.71%
Model 12	2.95%	3.43%	3.36%	3.12%	4.05%	3.46%
Model 13	2.83%	2.84%	3.23%	3.05%	3.88%	3.21%
Model 14	2.78%	2.82%	3.06%	3.26%	3.91%	3.64%
Model 15	2.86%	2.87%	3.24%	3.22%	3.91%	3.87%
Model 16	2.84%	2.85%	3.40%	3.19%	3.96%	3.49%
Model 17	4.85%	6.00%	12.67%	3.42%	3.99%	4.89%
Model 18	4.61%	5.30%	12.81%	3.41%	3.88%	6.41%
Model 19	3.83%	3.52%	15.38%	3.32%	3.69%	5.27%
Model 20	3.09%	3.07%	10.69%	3.13%	3.47%	8.07%
Model 21	3.02%	3.03%	11.87%	3.07%	3.40%	8.48%
Model 22	2.97%	2.98%	11.68%	3.02%	3.32%	5.88%
Model 23	2.94%	2.94%	9.88%	2.97%	3.27%	4.88%
Model 24	2.93%	2.90%	7.73%	2.92%	3.22%	5.22%
Model 25	2.94%	2.89%	7.56%	2.86%	3.19%	5.15%
Model 26	2.99%	2.91%	8.04%	2.80%	3.13%	4.70%
Model 27	3.03%	2.93%	8.35%	2.77%	3.09%	4.42%
Model 28	3.07%	2.95%	8.30%	2.75%	3.05%	4.56%

Overall the observations from Section 4.3.2 hold – for the bottom-up approach from Level 0 the LightGBM models are slightly better during the validation period, with very similar MAPE values, and for the validation period, the best ridge model is Model 24, corresponding to the regularization parameter 16 as shown in Table 1. The test period bottom-up results are the same as in Figure 5 in the left graph in row 'Test, Level 0' since the test period is not changed.

It is clear that the more complicated reconciliation methods did not really improve the performance, the reasons for this are analyzed in Section 4.3.4 with the best overall model as an example. The OLS reconciliation retained the performance, with a minor increase in MAPE values for most LightGBM models and a minor increase for most ridge regression models. For most models, the difference in accuracies was less than 0.2%. However, the MinT reconciliation produced considerably worse results, both for validation and test periods and especially for ridge models.

The best overall model was Model 14, which had the best average MAPE of 2.89% across the reconciliation methods. It also had the best MAPE for OLS and MinT

reconciliations, while for the bottom-up approach, it was the third-best, having only 0.01% worse MAPE than Model 3, which had the best MAPE for this method. For the test period, the selected model had the average MAPE of 3.6%, with the bottom-up method having the best accuracy of 3.05%. The main reason for the worse performance on the test set remains the same as for all the models – bad accuracy in January, as can be seen from Figure 5, not overfitting.

4.3.4 Analysis of Reconciliation Methods

The analysis in this section is done using the best model during the validation period, Model 14. Table 2 shows that reconciliation significantly worsened the top-level forecast, especially the MinT method. This shows that the assumptions of reconciliation methods are not true, and thus, the reconciliation cannot be performed. The MinT reconciliation uses the covariance matrix of the forecasting errors, estimated from the previous months' forecasting errors. Already from Figure 5, we can see that the forecasting MAPEs vary from month to month quite significantly.

This is more visible in Figures 7 and 8. In Figure 7 the kernel density estimations for the errors of Model 14 for the single Level 4 segment are shown for each month on the left and for all the errors from April 2023 to the corresponding month on the right. From the left graph, we can see that the monthly error distributions look quite different already for the most aggregated time series, which suggests that just a few months is not a long enough time to estimate the covariance matrix of errors. From the right graph, we can see that the distribution for longer periods shows signs of converging but is still relatively different.

A similar conclusion can be drawn from Figure 8, where the top-level MAPEs by month for each model after the MinT reconciliation are shown on a heatmap. We can see that both LightGBM and ridge regression models have very high MAPE values for the first months in the validation set. However, especially for LightGBM models, the post-reconciliation MAPEs are within reasonable bounds, suggesting that for the MinT approach to work, the whole year of forecast errors is necessary.

The difference between bottom-up and OLS reconciliation is less extreme and nearly unnoticeable. For Model 14, the accuracy was reduced by just 0.04%. The main assumption that went into the derivation of the OLS method was that the base level errors are approximately additive the same way the base forecasts are, as shown in equation (17). That is, when we sum the lowest level base forecast errors using the summing matrix S, as defined in Section 3.1 they should match the higher level base forecast errors. This assumption is generally upheld, though not precisely. This explains why the OLS reconciliation results roughly match the bottom-up method accuracies but do not improve significantly on them.

The extent of violation of the error aggregation constraint can be seen by looking at the forecast error reconciliation errors (EREs). First, the bottom-level base forecasts are

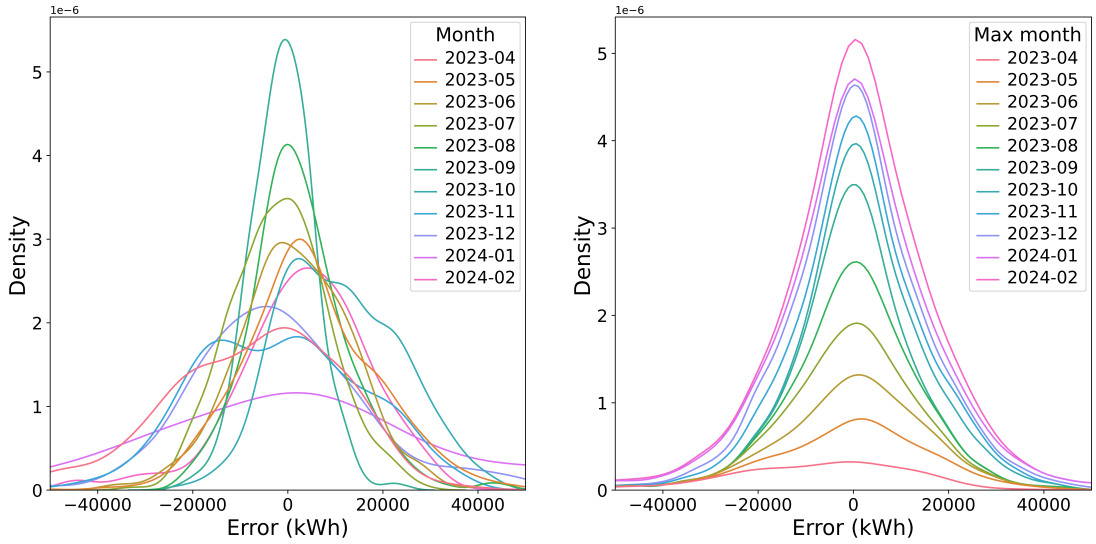


Figure 7. Kernel density estimations for the errors of LightGBM Model 3 at whole country level base forecasts. The left plot shows the distribution estimations by monthly values, and the right plot shows the cumulative error values from April 2024 up to and including the 'Max month'.

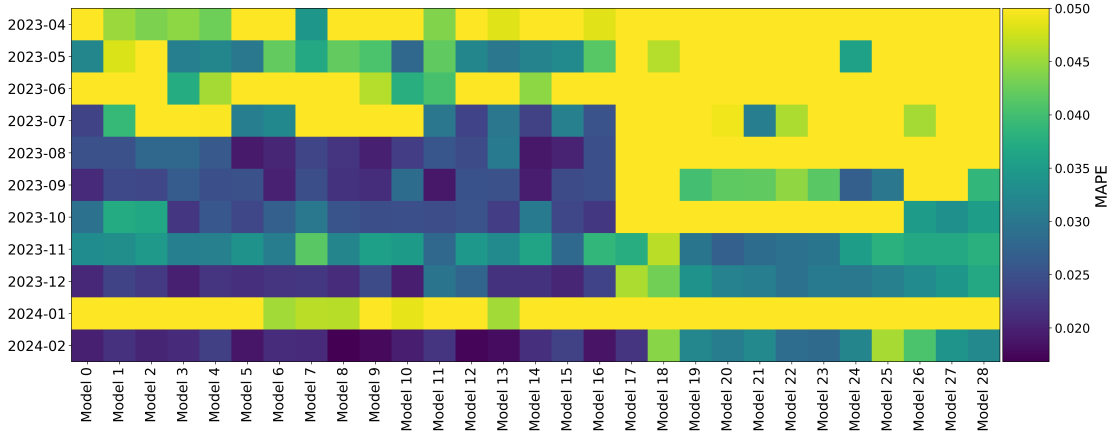


Figure 8. Top-level MAPEs by month for each model after the MinT reconciliation is applied. The MAPE values are clipped at 5% for readability.

aggregated, producing the calculated expected errors for each segment Levels 1-4. The calculated errors are subtracted from the base forecasts of the corresponding segment at Levels 1-4. This subtraction results in the ERE for each segment, which shows how much the assumption of the OLS reconciliation method is violated. This calculation is

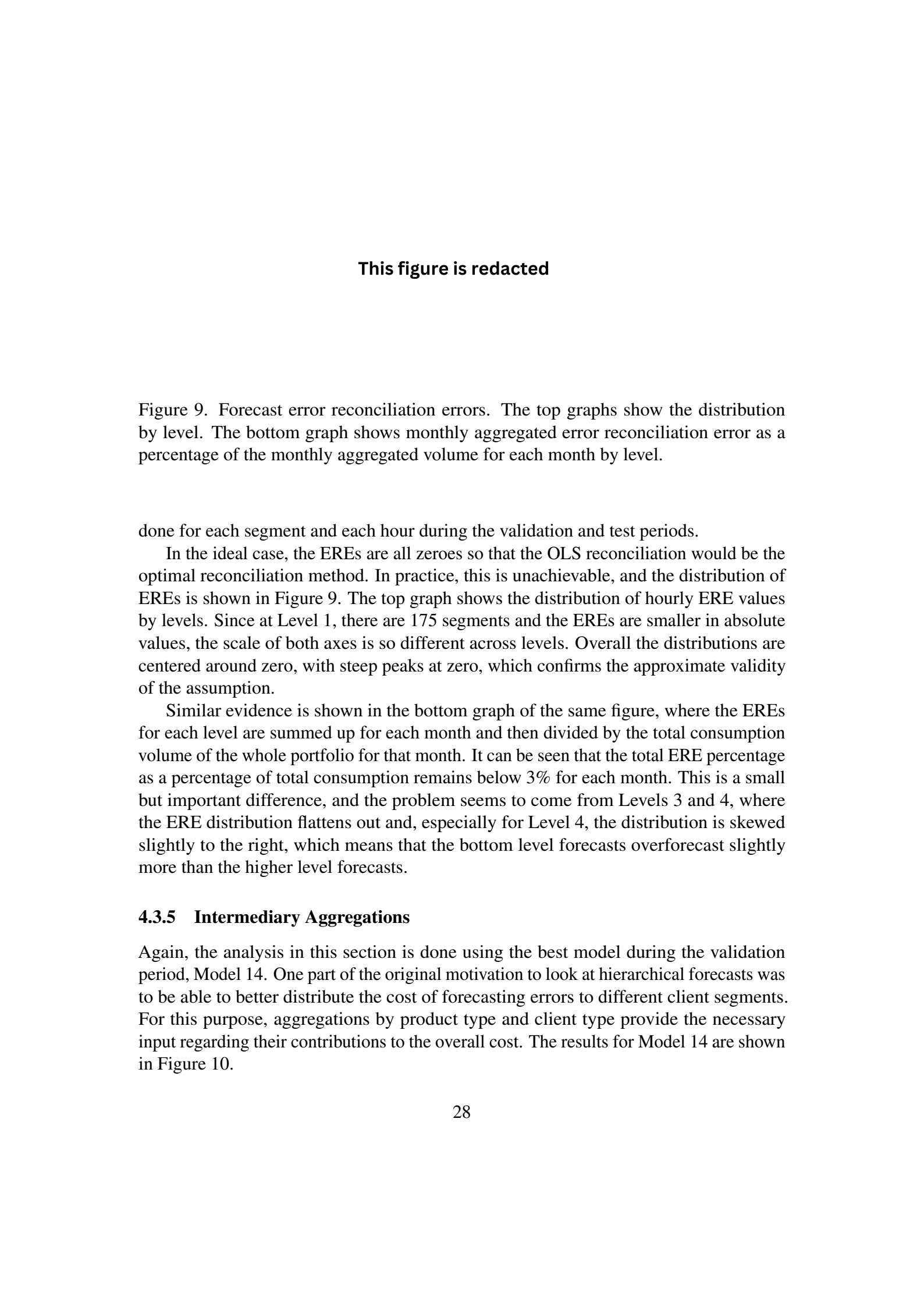


Figure 9. Forecast error reconciliation errors. The top graphs show the distribution by level. The bottom graph shows monthly aggregated error reconciliation error as a percentage of the monthly aggregated volume for each month by level.

done for each segment and each hour during the validation and test periods.

In the ideal case, the EREs are all zeroes so that the OLS reconciliation would be the optimal reconciliation method. In practice, this is unachievable, and the distribution of EREs is shown in Figure 9. The top graph shows the distribution of hourly ERE values by levels. Since at Level 1, there are 175 segments and the EREs are smaller in absolute values, the scale of both axes is so different across levels. Overall the distributions are centered around zero, with steep peaks at zero, which confirms the approximate validity of the assumption.

Similar evidence is shown in the bottom graph of the same figure, where the EREs for each level are summed up for each month and then divided by the total consumption volume of the whole portfolio for that month. It can be seen that the total ERE percentage as a percentage of total consumption remains below 3% for each month. This is a small but important difference, and the problem seems to come from Levels 3 and 4, where the ERE distribution flattens out and, especially for Level 4, the distribution is skewed slightly to the right, which means that the bottom level forecasts overforecast slightly more than the higher level forecasts.

4.3.5 Intermediary Aggregations

Again, the analysis in this section is done using the best model during the validation period, Model 14. One part of the original motivation to look at hierarchical forecasts was to be able to better distribute the cost of forecasting errors to different client segments. For this purpose, aggregations by product type and client type provide the necessary input regarding their contributions to the overall cost. The results for Model 14 are shown in Figure 10.

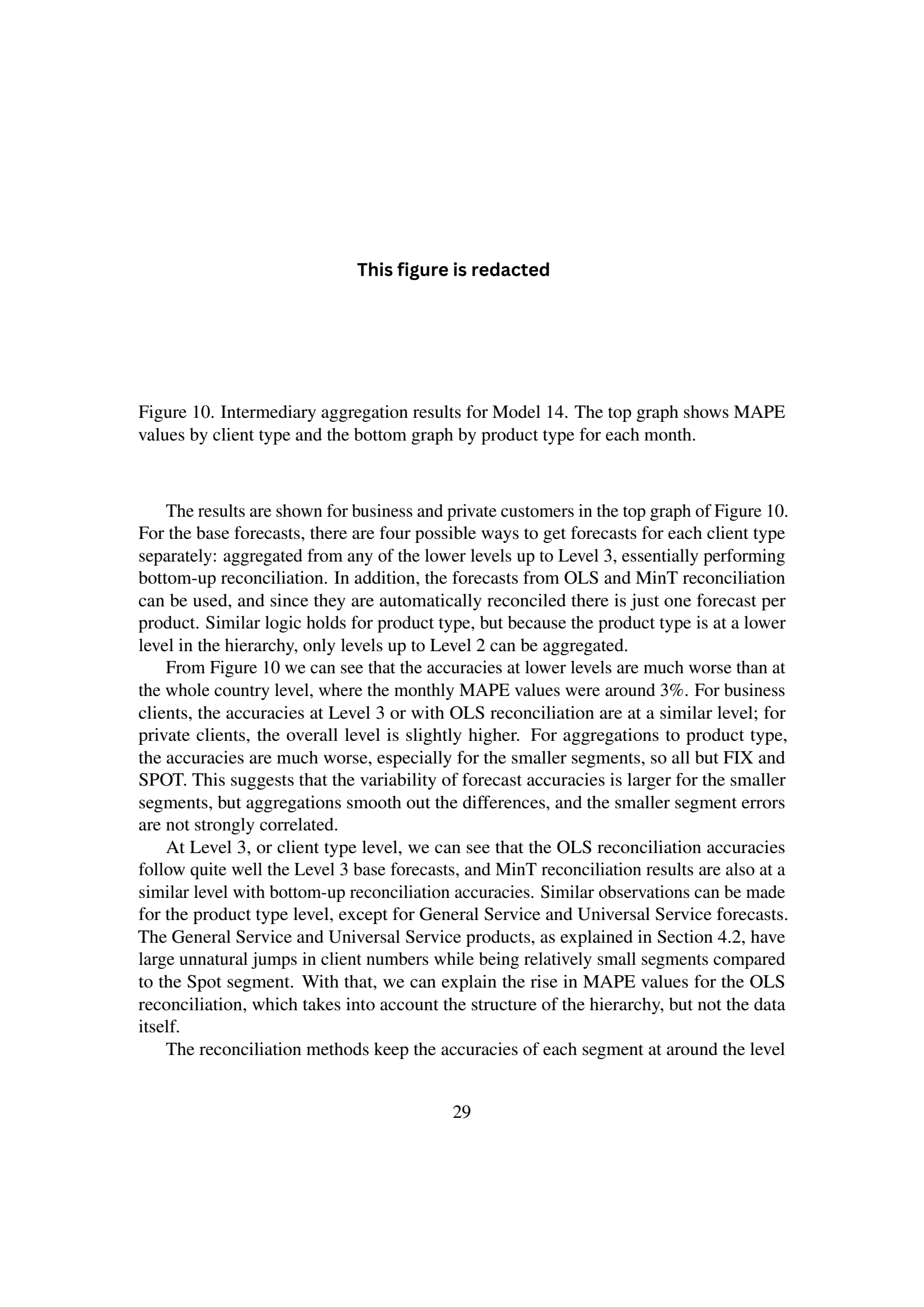


Figure 10. Intermediary aggregation results for Model 14. The top graph shows MAPE values by client type and the bottom graph by product type for each month.

The results are shown for business and private customers in the top graph of Figure 10. For the base forecasts, there are four possible ways to get forecasts for each client type separately: aggregated from any of the lower levels up to Level 3, essentially performing bottom-up reconciliation. In addition, the forecasts from OLS and MinT reconciliation can be used, and since they are automatically reconciled there is just one forecast per product. Similar logic holds for product type, but because the product type is at a lower level in the hierarchy, only levels up to Level 2 can be aggregated.

From Figure 10 we can see that the accuracies at lower levels are much worse than at the whole country level, where the monthly MAPE values were around 3%. For business clients, the accuracies at Level 3 or with OLS reconciliation are at a similar level; for private clients, the overall level is slightly higher. For aggregations to product type, the accuracies are much worse, especially for the smaller segments, so all but FIX and SPOT. This suggests that the variability of forecast accuracies is larger for the smaller segments, but aggregations smooth out the differences, and the smaller segment errors are not strongly correlated.

At Level 3, or client type level, we can see that the OLS reconciliation accuracies follow quite well the Level 3 base forecasts, and MinT reconciliation results are also at a similar level with bottom-up reconciliation accuracies. Similar observations can be made for the product type level, except for General Service and Universal Service forecasts. The General Service and Universal Service products, as explained in Section 4.2, have large unnatural jumps in client numbers while being relatively small segments compared to the Spot segment. With that, we can explain the rise in MAPE values for the OLS reconciliation, which takes into account the structure of the hierarchy, but not the data itself.

The reconciliation methods keep the accuracies of each segment at around the level

we would forecast them directly. This means there are no weird distribution effects, where overforecast in one segment compensates for underforecast in another segment after the reconciliation is performed. This gives us the confidence to use the reconciliated forecasts to distribute the cost originating from the whole country level forecasts between different client or product types.

4.4 Discussion and Future Research

The primary aim of this thesis was to investigate hierarchical time series forecasting methods to see if they can be applied to the day-ahead forecasts of electricity consumption. The investigation was performed with somewhat mixed results – the simplest, bottom-up approach was surprisingly accurate, considering the extremely different size and profile of forecasted time series, as seen from Section 4.2. At the same time, the use of theoretically optimal forecasting methods was severely limited by the restricting assumptions the optimal methods imposed on the forecasting models and their performance.

There is ample opportunity for further research on this topic. The first opportunity is to look at different datasets. Currently, the MinT reconciliation is hindered by the requirements imposed on the covariance matrix of base forecasts. If a longer period of data is available for training, a longer validation period can also be used, thus providing year-round forecasting errors for the empirical covariance matrix. The hope that a longer validation period might help with the reconciliation methods is supported by the fact that in-sample MinT reconciliation resulted in MAPE values of less than 2%.

Secondly, only actual weather measurements are currently available, but in production systems, the actual weather is not known during prediction time. Internal analyses have shown limited improvement for previous models, but this is an unexplored nuance in this thesis.

In this thesis, only two types of models, LightGBM and ridge regression, were analyzed. From the test period, we saw that higher regression regularization parameter values were extremely good at predicting the highly volatile and expensive period at the beginning of the year. This is a promising discovery since the sudden increase in consumption is always hard to predict and has caused prices to skyrocket in both spot and imbalance markets. Exploring where the strengths of different models lie, and other model types would be useful in handling future special situations.

The current hierarchy was built out of convenience, and there are no physical limitations that prevent us from switching the order of the lower four levels. Whether this would improve performance is unknown currently, and it might be that some levels are even unnecessary. As seen in Figure 5, it might be that, for example, county or location level does not really provide any benefit. This would be surprising, considering there are days when the difference in temperature in different parts of Estonia might reach 20 degrees Celsius, but alternatively, the aggregated information might smooth out the noise.

In addition, the reconciliation methods themselves can be looked at theoretically. Current methods make very general assumptions and do not restrict the reconciled forecasts very strictly. This results in a somewhat weird reconciliation, where some segments have negative consumption forecasts. This, of course, is not physically accurate and is compensated by higher positive forecasts for some other segments, but it may be feasible to put restrictions on the forecasts as shown in [VEC15] if the requirement for linear reconciliation is dropped.

The current analysis was done on only Estonian consumers. There are other big segments in Estonia and other countries, for which similar methodology could be applied. The other portfolios have different customer composition and behavioral patterns (e.g., solar panels on the roof), which might mean that the current methodology and model hyper-parameters cannot be directly transferred to those segments. Whether this is true or whether the models are universally usable would be another direction for future research.

Furthermore, the general behavior of customers has changed significantly in the past few years. There has been a massive boom in solar panel installations, customers are more price-aware and thus price-sensitive, there are more electric cars on the road, and the number of storage units is increasing, to name a few changes. This implies that older training data might not be as relevant and weighting more recent or otherwise similar training data with higher weights might improve the performance of the models.

This thesis only looked at day-ahead forecasts, but for system balancing intra-day forecasts are just as important. Thus, looking at whether the same models can be applied to forecast intra-day consumption with better accuracy than day-ahead portfolio consumption is another possible direction for future analysis.

To dive deeper into the nuances of models' behavior, the distribution is also a potential avenue of investigation. As seen in Figure 9, the error distributions differ from month to month, with some months containing extremely fat tails. In terms of electricity system functioning and the cost to the electricity retailers, avoiding extremely high forecasting errors is of great financial interest. One way to trade error variance with error mean would be to investigate alternative target metrics and objective functions.

Finally, the electricity system itself will be seeing important changes, as described in Section 2, of which the most relevant will be the 15-minute balancing period for day-ahead forecasting. If and how this impacts the performance of the models is currently unknown but will be a very relevant question in the years to come.

5 Conclusion

The primary goal of this thesis was to investigate hierarchical time series forecasting methods in the context of day-ahead electricity consumption forecasting. Three hierarchical forecast reconciliation approaches were analyzed more thoroughly: the bottom-up method, OLS reconciliation, and Minimal Trace (MinT) reconciliation.

The source of the consumption dataset is Eesti Energia, which includes consumption values at various levels of aggregation. The weather dataset was sourced from OpenWeather and included the measured values of several of the most commonly used weather parameters with hourly resolution. All data had hourly granularity from March 2021 to March 2024. Based on the consumption data, the hierarchy was built, and feature engineering was performed.

Base models were then trained on all the consumption time series in the hierarchy. Two types of models were used with different hyper-parameters: ridge regression and LightGBM, with 28 models trained in total. Monthly expanding window was used for hyper-parameter validation and testing. The validation period was from March 2023 to December 2023, and the test period was the first two months of 2024.

The bottom-up reconciliation method was used as a baseline, against which the more complex methods were compared. With the bottom-up reconciliation method, most LightGBM models performed better than ridge regression models, although there was some variation based on the selected set of hyper-parameters. However, the performance during the test period was more varied, with the best models being ridge regression models.

The main result of this thesis is that the more complex reconciliation methods did not consistently improve forecasts for day-ahead electricity consumption forecasting. For LightGBM-based models, the OLS reconciliation was only slightly worse than the bottom-up method, with MAPE being only 0.04% worse for the selected model in the validation period. The MinT reconciliation was already considerably worse in performance, with the accuracy decreasing by 0.28% during the validation period. For ridge regression models, the relative performance of different reconciliation approaches varied more. For higher values of the regularization parameter, the OLS method was better than the bottom-up method, but the MinT approach was considerably worse for all the ridge regression models.

A more detailed analysis was done to explain the drop in performance of supposedly optimal reconciliation methods. The MinT approach relies on the covariance matrix of base forecast errors. However, it appears that the model error distribution varies over different months, so the covariance matrix built on the forecasts of one month might not hold for the next month. The possible approach to solve this issue would be to use a longer period for calculating the covariance matrix, but due to the limited amount of available training data, this was not performed.

The OLS reconciliation approach does not directly use the covariance information of

the base forecast errors. However, this simplification depends on an assumption made regarding the structure of the base forecast errors. The base forecast errors are assumed to approximately satisfy the same aggregation constraint as the time series themselves. It appears that there are significant enough deviations from this constraint, which explains the underperformance of this method.

One of the primary motivations for investigating the hierarchical time series forecasting methods was the possibility of constructing coherent forecasts for different levels of the hierarchy. This question was analyzed, and it appears the reconciliation methods work as intended and it is possible to use the hierarchical forecasting approaches for simultaneous forecasting for different client or product segments.

This thesis provided an overview and analysis of the hierarchical time series forecasting for Estonian consumption forecasting. There are several possible avenues for future research, from using different datasets and other model types, changing the structure of the hierarchy, increasing validation length, or using alternative reconciliation methodologies.

References

[AtA22a] Elering AS, AS "Augstsprieguma tīkls", and Litgrid AB. Baltic balancing roadmap. Technical report, Baltic Transmission System Operators, Baltics, October 2022. Accessed: 2024-05-05.

[AtA22b] Elering AS, AS "Augstsprieguma tīkls", and LITGRID AB. Baltic coba imbalance settlement rules. https://elering.ee/sites/default/files/2022-04/Baltic_CoBA_Imbalance_Settlement_Rules_confirmed.pdf, 2022. Accessed: 2024-05-05.

[Bre96] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.

[CG16] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[Ele21] Elering. Eesti tarbijate elektrivarustuskindluse aruanne aastani 2030. https://elering.ee/sites/default/files/2021-12/Varustuskindlus%202021%20lk.pdf, 2021. Accessed: 2024-05-05.

[Ele22a] Elering. *Elektrituru Käsiraamat*, 2022. Accessed: 2024-05-05.

[Ele22b] Elering. Security of supply report of the estonian electricity system. https://elering.ee/sites/default/files/2023-05/elering_vka_2022-ENG.pdf, 2022. Accessed: 2024-05-05.

[Ele23a] Elering. Eesti elektrivarustuskindluse aruanne. https://elering.ee/sites/default/files/2023-12/Elering_VKA_2023_WEB.pdf, 2023. Accessed: 2024-05-05.

[Ele23b] Elering. Xb system imbalance. https://www.elering.ee/en/xb-system-imbalance, 2023. Accessed: 2024-05-05.

[ele24a] Elektrituruseadus. https://www.riigiteataja.ee/akt/130062023006, 01.01.2024. Accessed: 2024-05-05.

[Ele24b] Elering. Bilansiportfellide osakaalud 2023. https://www.elering.ee/bilansiportfellide-osakaalud, 2024. Accessed: 2024-05-05.

[Ele24c] Elering. Production and consumption dashboard. https://dashboard.elering.ee/et/system/with-plan/production-consumption, 2024. Accessed: 2024-05-15.

[Fri01] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[Fri02] Jerome H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

[GG60] Yehuda Grunfeld and Zvi Griliches. Is aggregation necessarily bad? *The Review of Economics and Statistics*, 42(1):1–13, 1960.

[GPB19] Shadi Goodarzi, H. Niles Perera, and Derek Bunn. The impact of renewable energy forecast errors on imbalance volumes and electricity spot prices. *Energy Policy*, 134:110827, 2019. Accessed: 2024-05-05.

[HA21] Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, Melbourne, Australia, 3 edition, 2021. Accessed: 2024-05-05.

[HAAS11] Rob J Hyndman, Roman A Ahmed, George Athanasopoulos, and Han Lin Shang. Optimal combination forecasts for hierarchical time series. *Computational statistics & data analysis*, 55(9):2579–2589, 2011.

[HK70] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[HMS20] Addison Howard, Spyros Makridakis, and Evangelos Spiliotis. M5 forecasting - accuracy. `https://kaggle.com/competitions/m5-forecasting-accuracy`, 2020.

[KMF+17] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[Lig24] LightGBM Development Team. Parameters — lightgbm. `https://lightgbm.readthedocs.io/en/latest/Parameters.html`, 2024. Accessed: 2024-05-14.

[MSA20] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74, 2020.

[MSA22] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4):1346–1364, 2022.

[Ope24] OpenAI. Chatgpt. version 4. https://www.openai.com/chatgpt, 2024. Accessed: 2024-05-05.

[oTSOfEEE24] European Network of Transmission System Operators for Electricity (ENTSO-E). Entso-e transparency platform. https://transparency.entsoe.eu/generation/r2/installedGenerationCapacityAggregation/show, 2024. Accessed: 2024-05-05.

[OWE68] Guy H. Orcutt, Harold W. Watts, and John B. Edwards. Data aggregation and information loss. *The American Economic Review*, 58(4):773–787, 1968.

[Poo22] Nord Pool. Nord pool annual review 2022. https://www.nordpoolgroup.com/4ac1a6/globalassets/download-center/annual-report/nord-pool-annual-review-2022.pdf, 2022. Accessed: 2024-05-05.

[RS98] Sanjay Ranka and Vineet Singh. Clouds: A decision tree classifier for large datasets. In *Proceedings of the 4th knowledge discovery and data mining conference*, volume 2, pages 2–8, 1998.

[SLWH19] George Athanasopoulos Shanika L. Wickramasuriya and Rob J. Hyndman. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association*, 114(526):804–819, 2019.

[tur12] Eesti elektrituru täielik avanemine. https://energiatalgud.ee/sites/default/files/images_sala/c/c8/Eesti_elektrituru_t%C3%A4ielik_avanemine.pdf, 2012. Accessed: 2024-05-05.

[VEC15] Tim Van Erven and Jairo Cugliari. Game-theoretically optimal reconciliation of contemporaneous hierarchical time series forecasts. In *Modeling and stochastic learning for forecasting in high dimensions*, pages 297–317. Springer, 2015.

Appendix

I. Features

Feature name	Description
Weather features	
air_temperature	Measured air temperature
feels_like_temperature	Accounts for the human perception of temperature
pressure	Atmospheric pressure at sea level, hPa
relative_humidity	Humidity percentage
rain_1h	Precipitation of rain in mm
snow_1h	Precipitation of snow in mm
wind_x, wind_y	Wind vector components, found from wind speed and direction components using simple trigonometry
clouds_all	Cloudiness percentage
clear_sky_ghi, clear_sky_dni, clear_sky_dhi	Global horizontal, direct normal and diffuse horizontal irradiation for clear sky scenarios
cloudy_sky_ghi, cloudy_sky_dni, cloudy_sky_dhi	Global horizontal, direct normal and diffuse horizontal irradiation for cloudy sky scenarios
Interval features	
county	Location of the measuring point
product_type	Type of product in the contract for the corresponding hour
size_type	The size indication of the measuring point
P_OR_B	Client type of the measuring point
SP_YEARCON	The expected yearly consumption of the measuring point
interval_sum	The consumption at the measuring point for the corresponding hour

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Carel Kuusk**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Hierarchical Forecasting Methods in Day-Ahead Electricity Consumption Forecasting,

   supervised by Meelis Kull.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright with the following restrictions:

   (a) Section 4.2 is redacted due to a request from Eesti Energia until 31.05.2027.

   (b) Figure 13 an Figure 14 are redacted due to a request from Eesti Energia until 31.05.2027.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Carel Kuusk
15/05/2024