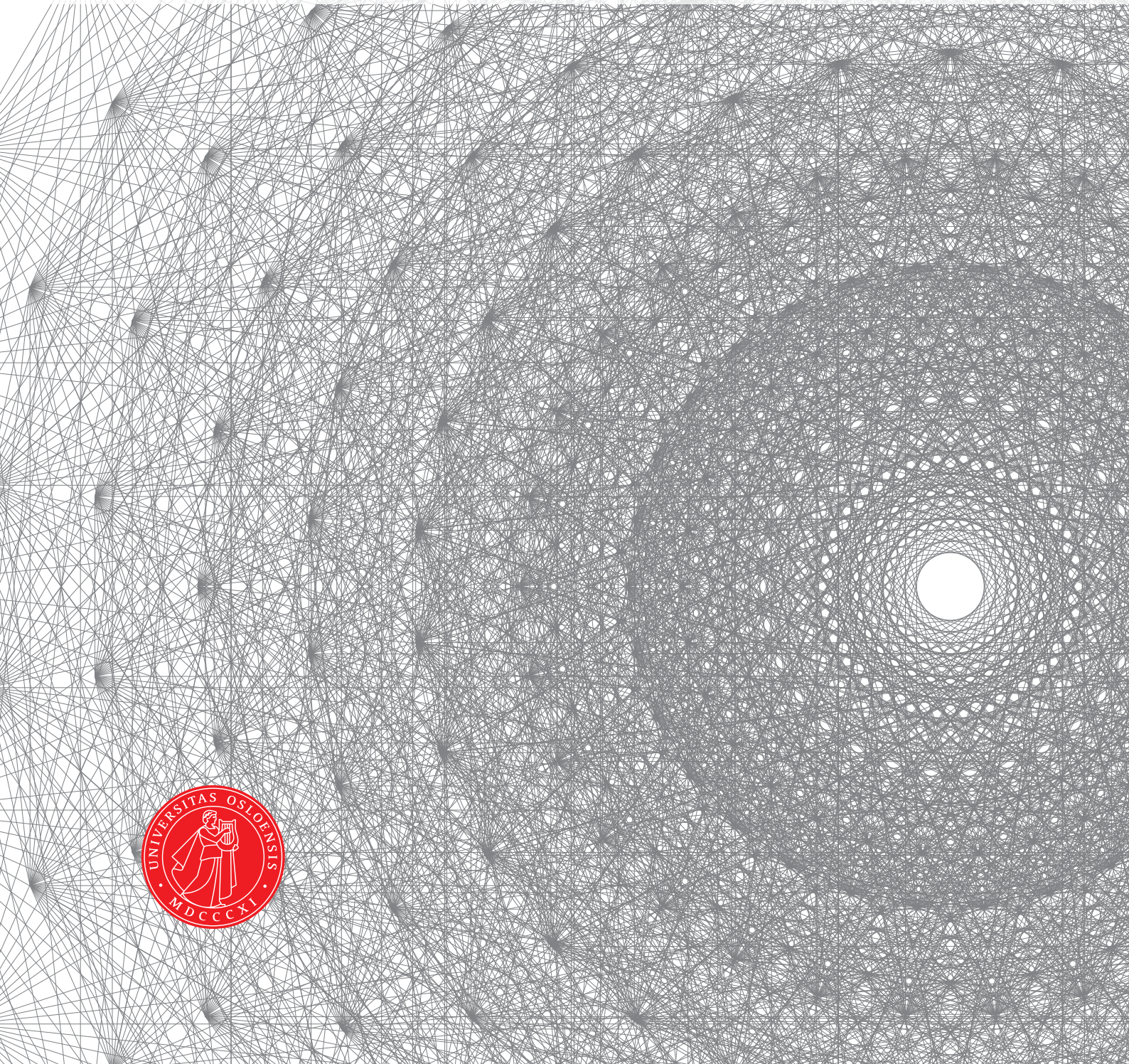
UiO : Department of Mathematics
University of Oslo

Electricity Demand Forecasting

Probabilistic Demand Forecasting Using Principal Components of Seasonal Temperature Forecasts

Eirik Sjåvik

Master's Thesis, Spring 2023



This master's thesis is submitted under the master's programme *Data Science*, with programme option *Data Science*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Abstract

In this thesis we introduce a medium-term forecast model for electricity demand in the Nordic region utilizing seasonal Numerical Weather Prediction (NWP) temperature forecasts.

Our demand model is composed of two integral parts. The first part is a structural demand model, which seeks to model electricity demand, at a specific target time, by utilizing temperature at the same target time. The temperature data consist of observations across a grid over the Nordic countries. By employing a principal component transformation of the temperature grid we seek to describe the relation between demand and the temperature field as a whole by a small subset of principal components. We model this relation through a Generalized Additive Model. The second part is a probabilistic temperature forecast model utilizing NWP forecasts in principal component space. By combing the two parts we can form a probabilistic forecast of demand for the Nordic region. We show that the models employed show great performance when compared to relevant baseline models.

We also introduce a re-weighting scheme for NWP forecasts in principal component space. By re-weighing temperature forecasts after how well they recently have performed, we can 'update' the forecast and obtain short-term improvements in skill at any time point.

Acknowledgements

Writing this thesis has been an inspiring challenge. I have been extremely fortunate to have had a lot of people supporting me on my continued quest to acquire more knowledge.

First of all, I would like to thank my supervisor, Alex Lenkoski, who has made this master project undertaking a thoroughly enjoyable affair. Throughout the year and a half I have been working on this thesis we have met regularly to discuss progress and new ideas. I cannot remember a single meeting after which I did not think to myself: "Yes, it would be really cool to check this [insert ML model, meteorological phenomenon, statistical framework] out." If anything you made the writing of this thesis too interesting. Thank you, Alex, for your encouragement and sound advice.

I would also like to thank co-supervisor Geir Storvik who provided valuable help and insights throughout the writing of this thesis. I'm also thankful to the statistics department staff for keeping what I have perceived to be an open door policy, and for answering all my naive questions clearly and concisely. Thank you also to my fellow students at the study hall for offering inspiring theoretical discussions and ample distractions. A big thank you also goes to Eirik Ørevik Aadland for reading through the final drafts.

I would like to thank my family and friends who have been incredibly supportive. Thank you for all your help and support during the last couple of years. Mostly, however, I would like to thank Tove. I literally could not have done this without you, thank you for all your help and love through this whole process. I love you.

To Freyr and Bergljot, you are the best!

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Energy use is a hot-button and multifaceted topic with potential relevance for a wide array of issues including climate change, geopolitics, global supply chain security, economic inequality, environmental protection and major infrastructure projects. How much energy people use, or rather how much they want to use, expressed as energy demand, will have potentially large impacts on energy price, production and infrastructure (Coninck et al. 2022, Austvik 2019; IEA 2022; Oswald et al. 2020; Achuo et al. 2022).

This thesis concerns the topic of medium-term electricity demand forecasting. Specifically, it focuses on how to employ temperature forecasts as features in an electricity demand forecasting model. Medium-term electricity demand forecasting has a broad applicability for stakeholders in industry, energy markets, government, and the broader public.

For energy trading companies, having access to forecasts of future energy demand is crucial, since demand has a substantial effect on the energy spot price. Medium-term demand forecasts are also required for companies in electricity generation and electricity providers, because the planning horizon for both is typically on the order of weeks or months (Kristiansen 2014).

Improved demand forecasts also have an important role to play with regard to climate change mitigation, as they may help improve energy efficiency and reduce energy waste (Coninck et al. 2022; Bala et al. 2022). Transitioning to zero-emission energy systems further involves the electrification of the car-fleet and an increasing reliance on renewables, which in turn requires updated knowledge about when electric energy will be consumed (Olatomiwa et al. 2016; Orlov et al. 2020). In addition, demand forecasts at this horizon might prove useful for households to adjust their consumption. Despite these reasons, the research on medium-term demand forecasting has been given comparatively little attention relative to both short- and long-term forecasts (Kuster et al. 2017).

In general one can talk about two non-exclusionary approaches to forecast improvement: 1) finding better data; and 2) finding a better model. The motivating idea behind this thesis is encapsulated in the question: 'What is the next best data?' In the case of demand forecasting we take the answer to be temperature, or more specifically seasonal temperature forecasts. Our strategy for improving demand forecasts, then, centers on the incorporation of seasonal temperature forecast data into an electricity demand model.

In this thesis we present:

1. An electricity demand forecast model based on dimensionality reduced temperature grids which aims to forecast energy demand at the medium-term (up to 60 days) horizon.

2. A probabilistic temperature forecast model based on seasonal Numerical Weather Prediction (NWP) ensemble models whose output can be folded into the energy demand model.

The incorporation of temperature in the field of electricity demand forecasting is not new. Previous research has shown both the importance of temperature as a predictor in demand forecasting and incorporated temperature forecasts into demand models. The main innovations of our approach are the following: First, we introduce a medium-term forecast model for electricity demand in the Nordic region, utilizing seasonal NWP temperature forecast data. Second, other researchers (e.g. De Felice et al. (2015)) have focused on season specific temperature effects on electricity demand. We develop a year-round demand forecast model, which takes account of the varying effects of temperature on demand throughout the year. Third, we also introduce a re-weighting scheme for NWP forecasts in principal component space. By reweighing temperature forecasts after how well they recently have performed, we can 'update' the forecast and obtain short term improvements in skill at any time point. Fourth, we build a Gaussian Copula (GC) aggregation method for making forecast model skill apparent at longer lead times. Our approach also stands out in the large amount of data employed. We are working with 88392 hourly observations of electricity demand, over 172 million individual temperature grid observations, and over 9 million forecasts of temperature at individual grid points.

Energy demand is heavily dependent on local context. Cultural and climatic factors together with the energy supply mix and infrastructure all influence how energy is consumed and how much (Wilhite et al. 1996). Our specific case is the Nordic region, but the models presented in this thesis will have a broader applicability. The Nord Pool electricity demand data encompasses 7 countries: Norway, Sweden, Denmark, Finland, Estonia, Latvia and Lithuania, hereafter for simplicity referred to as the Nordic region. These countries exhibit a high level of energy consumption per capita, driven by its affluence, cold winters, and good access to cheap energy sources (SSB 2014). In many continental European countries, natural gas is the dominant household energy source. In Scandinavia, by contrast, homes are often powered by electric energy produced by hydropower, which up until the recent European energy crisis has been very affordable (Energy Facts Norway 2023). Demand in the Nordics increases in the winter months due to colder weather which leads to an increased use of heating appliances (Foldvik Eikeland et al. 2021). Electricity demand thus exhibits clear seasonal variation, and a clear dependence on temperature. Of particular concern is peak electricity demand as it determines the generation capacity needed to satisfy demand at all times (Lindberg et al. 2019). Since peak electricity demand occurs at low temperatures, forecasting temperature at extreme quantiles becomes important.

The overarching question this thesis aims to answer is: How can we forecast future energy demand at the medium-term range based on probabilistic temperature data? This overarching question can be divided into two research

problems: 1) How can we build a good structural model for predicting energy demand by using temperature data? 2) How can we use seasonal NWP forecasts to give us distributional estimates of future temperature which we can fit into our demand model? We give a brief outline of our strategies for solving these problems below.

Problem 1: Structural Demand Model

The first problem concerns how we can build a good structural model for forecasting electricity demand by using observed temperature data. Our interest is to forecast demand at the regional level covering the Nordic region. The Nord Pool electricity demand data we will utilize, summarizes the demand volume (MWh) for the Nordic market. The main predictive source we will employ is the ERA5 temperature observations covering a 21×22 sized grid over the Nordic region for latitudes $55°$ to $75°$ and longitudes $4°$ to $25°$. This data is dense and highly correlated both temporally and spatially (see Section 4.2). By utilizing a PCA decomposition of the temperature grid and extracting the first principle components (PCs) we obtain features that effectively summarize the temperature variation of the grid across time points. Our working assumption, then, is that the principal components of the temperature grid provide an effective summary of the state of the temperature grid, which we can relate directly to electricity demand. To model this relation, we use a Generalized Additive Model (GAM). We will refer to variations of these models by the term GAM-PC.

The GAM-PC models are structural models in the sense that they seek to model demand at a specific target time, t, by utilizing temperature at the same target time. Since the target time in a forecast setting always is at some point in the future, the structural models work under the assumption that we have access to future temperature observations, or equivalently, perfect forecasts. By describing the relation between demand and temperature, the structural model prepares the ground for the incorporation of NWP forecast inputs.

A central part of modelling the relation between electricity demand and temperature is to uncover how much improvement can be gained by including temperature information in the demand model. For this purpose, we will explore and contrast different combinations and parametrizations of temperature data and time information (which we will model as fixed effects), to ascertain the source of increased predictive performance.

Problem 2: Probabilistic Temperature Forecasting

The second problem relates to how we can use seasonal NWP temperature forecasts to give distribution estimates of future temperature which we can utilize in our demand model. NWP forecasts are ensemble forecasts, which means that they are composed of a set of forecast members which together form a predictive distribution. The NWP ensemble forecasts are meant to give a probabilistic description of future temperature at individual grid points.

Our concern is how we can transfer the predictive distribution over a set grid into predictive distributions for principal components (PCs) of the forecasted temperature grid. We solve this by first performing a PCA decomposition on each individual forecast member. We then obtain the predictive distribution of

each principal component of interest by employing a sample quantile estimator over all ensemble members. We go on to evaluate how well the NWP forecasts in PC space perform in estimating observed PC temperature. By employing quantile regression we also look at whether we can gain predictive performance by adding weights, lagged forecasts and time covariates to the sample quantile estimates.

A constraint with regard to the utility of the NWP forecasts manifests itself when assessing performance at the hourly level. Compared to the baseline reference model, performance is only better for the first 15 days after a forecast is issued. The term 'forecast issuance' is used throughout to refer to the release of a forecast. In our thesis we tackle this problem in two ways. We first introduce a re-weighing scheme in principal component space. It can be performed at any time after a forecast has been issued for the purpose of obtaining short term improvements in skill. Each ensemble forecast member will be weighted according to its recent performance in forecasting temperature PCs. Based on these weights we form new quantile estimates of the predictive temperature PC distribution. We refer to the release of the forecast with updated quantile estimates as a forecast re-issuance.

The second way in which we deal with the disappearance of skill at longer lead times, revolves around forecast aggregation. Aggregation is not a method for improving forecast skill, but for making the skill of the NWP forecasts apparent at longer lead times. If a cold front appeared on a Saturday, the forecast would be wrong if it predicted it would happen on Sunday, but correct if it predicted it would appear during the weekend. By aggregating forecast outputs, we allow a skillful model to be rewarded for being slightly correct. For the purpose of forecast aggregation, we build a Gaussian Copula (GC) aggregation method which takes account of the correlation structure between the time points we aggregate over.

1.1 Thesis Outline

The rest of the thesis is organised as follows:

In **Chapter 2** we first present an overview of previous research on electricity demand forecasting, focusing on medium-term forecasts. Even though this field is somewhat underdeveloped, we find that the influence of temperature on demand is well established. We then provide an overview of literature connected to probabilistic temperature forecasting, focusing on NWP ensemble forecasts. We highlight issues connected to i) forecast updating through the incorporation of information from new observations, and ii) forecast aggregation.

Chapter 3 consists of an outline of the main theoretical framework. We first present the backbone of our modelling approach, namely Principal Component Analysis (PCA) and Generalized Additive Models (GAM). We go on to describe the structural demand model. This model combines time information, in the form of fixed effects, with temperature information, in the form of principal components within a GAM framework. We then outline different approaches connected to probabilistic forecasting, namely

Weighted Quantile Estimation (WQE), Quantile Regression (QR), Re-weighted Quantile Estimation (RQE) and aggregated quantile estimation by Gaussian Copulas (GC). We also give a description of our methodological approach, specifically describing the Prequential Cross-Validation (PCV) procedure we will employ.

In **Chapter 4** we move on to present the main data sources used in this thesis: the Nord Pool electricity demand data; the ERA5 temperature data; and the NWP temperature forecast data. We also give a brief description of computing resources, and programming software.

Chapter 5 presents the results of our investigation of Problem 1 concerning the structural demand model. We especially focus on three issues. Assuming we have access to near-future temperature, we first seek to find what model parametrization of the GAM-PC model gives the most accurate prediction of near-future energy demand. We also seek to find how much the inclusion of temperature contributes to increase predictive performance. In addition, we look at how well the best GAM-PC model performs compared to alternative implementations, specifically Lasso and XGBoost.

In **Chapter 6** we present results pertaining to Problem 2 on probabilistic temperature forecasting. We first look at the performance of the WQE sample quantile model. It utilizes principle components of NWP ensemble member forecasts as inputs for the purpose of forecasting the predictive distribution of PC temperature. Then, by employing Quantile Regression we also investigate to what extent adding weights, time fixed effects, and lagged forecasts to the WQE-estimates contributes to increased forecast performance. We further demonstrate the effect of re-weighing NWP PCs based on recent temperature observation, utilizing the RQE model. And, lastly, we look at how we can make forecast skill apparent at longer lead times by employing aggregated forecasts which we form by utilizing Gaussian Copulas.

In **Chapter 7** we report results related to the final probabilistic demand model, which utilizes the NWP forecast data in principal component form in the demand prediction task.

Chapter 8 provides a summary of the main findings, as well as suggestions for future research.

Though our specific case is the Nordic region, the methods described in this thesis will have a utility beyond the application on energy demand prediction. A plethora of phenomena are influenced by weather generally or temperature specifically, e.g. shipping, agriculture, wildfires, tourism, and energy production. If we are interested in a temperature-dependent phenomenon the utilization of NWP forecasts should be easily transferable to other settings than the one described here, energy demand prediction. If the phenomena we are interested in can be modeled with the help of observed temperature, we can use our model extension to obtain future temperature distributions which we in turn can use to give predictions for the temperature dependent phenomenon of interest.

Abbreviations

- ANN: Articifial Neural Networks
- ARIMA: Autoregressive Integrated Moving Average
- BATS: Box-Cox transformation, ARMA errors, Trend, and Seasonal component
- CDF: Cumulative Distribution Function
- CMCC: Centro Euro-Mediterraneo sui Cambiamenti Climatici (Euro-Mediterranean Center on Climate Change)
- C3S: Copernicus Climate Change Service
- DWD: Deutscher Wetterdienst (German Meteorological Service)
- ECMWF: European Centre for Medium-Range Weather Forecasts
- EMOS: Ensemble Model Output
- ERA5: ECMWF Reanalysis v5
- GAM: General Additive Model
- GC: Gaussian Copulas
- GCV: Generalized Cross-Validation (GCV)
- GDP: Gross Domestic Product
- MWh: Mega Watt Hour
- MOS: Model Output Statistics
- Météo-France: (French Meteorological Service)
- NWP: Numerical Weather Prediction
- NorCPM: Norwegian Climate Prediction Model
- OOS: Out-of-Sample Validation
- PCA: Principal Component Analysis
- PCV: Prequential Cross-Validation
- PC: Principal Component
- QR: Quantile Regression
- RAFT: Rapid Adjustment of Forecast Trajectories
- RMSE: Root Mean Squared Error
- SARIMA: Seasonal Autoregressive Integrated Moving Average
- SSW: Sudden Stratospheric Warming
- SS: Skill Score
- SVM: Support Vector Machine
- SVD: Singular Value Decomposition
- TBATS: Trigonometric seasonality, Box-Cox transformation, ARMA errors, Trend, and Seasonal component
- UKMET: United Kingdom Meteorological Office
- WQE: Weighted Quantile Estimation (WQE)

PART I

Part 1: Literature Review, Theory, Data

CHAPTER 2

Literature Review

In this chapter we will give an overview of the background literature for both the electricity demand forecasting problem and the temperature forecasting problem. Our aim is not to provide an exhaustive review, but to showcase a targeted selection of relevant research contributions. In Section 2.1 we will outline the literature on electricity demand. Our focus will be on approaches sharing our aim of forecasting at a medium-term horizon at a country/region scale. Compared to short- and long-term forecasts, little attention has been given to medium-term forecasts in previous research. In our review we find that no modeling approach dominates. Nevertheless, the influence of temperature on demand is well established.

In (Section 2.2) we present relevant research on probabilistic temperature forecasting centering on Numerical Weather Prediction (NWP) ensemble forecasts. We will first give a brief outline of NWPs, before we move on to focus on two specific sub-problems: the incorporation of new information in ensemble forecasts and how to aggregate model predictions over lead times.

2.1 Electricity Demand Forecasting

In this section we will provide a brief overview of the electricity demand forecasting literature. This research field is broad, and shows great variation in terms of modeling framework, predictors, methodology, scale, time frame, and overall aim (Kuster et al. 2017; Tamba et al. 2018). It covers the full range from hour-ahead predictions of single building electricity demand to decadal forecasts at the country or regional level scale. Our focus, however, will lie with the part of the literature that shares our general aim: medium-term forecasting of electricity demand at the country or regional level scale. Accordingly, we will devote most attention to discussing medium-term forecasts, and especially those that utilize weather information. Further, we will only look at regional or country-level forecasts, while excluding approaches aimed at local or sector level analyses (e.g. of single buildings, residential areas or industry sectors).

The literature tends to be country-, or region-specific, as opposed to unified or general. This ties down to country specific peculiarities in data collection and availability; energy system infrastructure; and climatic conditions effecting energy consumption. These elements also make results and methodologies difficult to compare directly between countries and across cases. An additional challenge is that electricity demand forecasting exhibits a considerable overlap with adjoining fields concerned with forecasting electricity price and electricity

load, as well as related topics such as forecasting demand for natural gas. A characteristic of the Nordic energy system (which is our case) is that cooling and heating installations often are electricity-based. On the European continent, in contrast, similar temperature-dependent installations are often gas-based. The overlap with natural gas demand forecasting (e.g. similar predictors, scales, horizons and forecasting models) lead us to treat the literature on gas demand as part of the same discourse as electricity demand.

A convenient way of categorizing the demand forecast literature is by forecast horizon. While lead time denotes the time span from the forecast is issued (issuance) to the target time, the forecast horizon refers to the longest such span of interest (the time frame). Depending on the forecast horizon, approaches might differ considerably in terms of the forecasting models employed, model accuracy, predictors, as well as overall aim, and stakeholders (Kuster et al. 2017; Tamba et al. 2018). Before we look closer at medium-term forecasts, we will briefly summarize the differences between forecast horizons. Even though distinguishing between short-, medium- and long-term horizons is common, there is no set definition demarcating the dividing line between them. For our purposes, we take the medium-term to mean forecasting with lead times between one week and one year.

Short-term forecast approaches look to forecast at lead times ranging from minutes ahead up to one week ahead. The forecast outputs are highly relevant for energy market trading and energy grid use optimization (Foldvik Eikeland et al. 2021). At this horizon the most common modeling approaches are artificial neural networks (ANN) and time-series models (ARIMA, TBATS), but Support Vector Machines (SVM) and regression analysis are also used (Kuster et al. 2017). These models often have a high resolution (i.e. short lead time intervals) and largely rely upon time information, but they also incorporate meteorological data.

Long-term forecasts operate with 1—100 year horizons. They are especially relevant for infrastructure planning (Lindberg et al. 2019), climate change mitigation (Malka et al. 2023), and industrial development (Huang et al. 2018). Long-term forecasts tend to employ regression frameworks and econometric tools which focus on integrating socio-economic indicators such as population change, GDP growth, inflation, and urbanization. Meteorological information in long-term models is employed at a relatively crude level, e.g. utilizing average yearly or seasonal temperature (Günay 2016), or incorporated through overarching trends such as climate change (Hor et al. 2006). At this time-frame, model frameworks rely on either extrapolating trends for their predictor variables or building secondary forecast models for them.

Medium-term forecasts exist in a middle space between the two others where both long-term trends and short-term fluctuations might complicate the forecasting task (Mirasgedis et al. 2006). A long-term trend might be clearly discernible over a time span of several years, but difficult to observe or forecast at the scale of months. Reliance on meteorological information becomes harder than in the short-term case since forecast skill (see Section 3.7) decreases by lead time (Bauer et al. 2015). A systematic overview from 2017 of English language papers on electrical load forecasting shows that comparatively few papers focus on medium-term prediction (which they define as between 1 week and up to several seasons), and that most country level forecasts were long-term (Kuster et al. 2017). Neural nets, SVM, time series models and regression analysis are

all utilized to forecast at this horizon, but no single modeling approach seems to dominate (Kuster et al. 2017; Bala et al. 2022). In the following we will give a presentation of a selection of relevant articles on medium-term forecasts.

A study on forecasting Greek electricity demand found temperature, and to a lesser degree humidity, to be important predictors (Mirasgedis et al. 2006). Using two regression models (daily and monthly) they incorporated lagged variables to account for auto-correlation structures. Training on nine years of observations ($n = 3652$) and testing on lead times of up to 1 year, they report a maximum prediction error of 2.7% and 4.6% for lead times of 1 month and 1 year respectively. The study is, however, subject to a set of potential limitations. First, it relies on a single year of validation testing, which makes their results susceptible to the peculiarities of the validation set. Second, while observing a non-linear relationship between demand and temperature, they model this through dichotomizing temperature into new variables for heating and cooling days. Thirdly, the model assumes that accurate weather forecasts are available at relevant lead times, but they do not provide test results with actual forecast data.

An important contribution to the medium-term forecasting literature is the introduction of the TBATS model applied to the task of forecasting Turkish energy demand (Livera et al. 2011). Training on $n = 2191$ daily observations over 6 years, the approach focuses on capturing three seasonal components including two calendar systems affecting demand patterns of holidays and religious festivals. The TBATS approach is an innovations state space model and outperforms the similar BATS model across all lead times (up to 1 year) in the out-of-sample set covering 3 years of observations.

Bala et al. (2022) forecast demand of both electricity and natural gas in the UK for lead times up to 36 months. They explore the performance of a set of models (SARIMA, ETS, NNAR, STL and TBATS) both individually and combined using simple model averaging (SMA). Training on monthly observations ($n = 327$) they employ average temperature and energy price as their main predictors. For the out-of-sample validation they report that the SARIMA model obtains the best results on predicting electricity demands, while the TBATS model performs the best on the natural gas forecasting task. Like the case for Mirasgedis et al. (2006) the models involved implicitly rely on forecasts of temperature and energy price. Consequently, the paper does not distinguish results by lead time, but reports performance across all lead times. The authors themselves also remark that the low resolution of the data (monthly observations) makes it difficult to obtain a more granular understanding of the structure between temperature and energy consumption.

A 2019 paper on electricity demand, temperature and price elasticity in the Oslo metropolitan region is foremost relevant because of its context (Hofmann et al. 2019). The aim of the study is not to forecast demand, but to explain the relation between demand, price and temperature. Employing two similar regression models, they use demand averaged either daily or over peak 6-hour consumption as the response variable ($n = 1548$). In addition to using average daily temperature as an explanatory variable, they also include time covariates and other meteorological predictors. They found that temperature is "... by far the most important explanatory variables [sic] when estimating the short-term price elasticity of electricity demand in metropolitan areas with electricity-based

heating" (Hofmann et al. 2019).[1] They also report that variables representing wind speed, humidity and sun exposure are not needed to adequately model electricity demand in Oslo.

Another important contribution is De Felice et al. (2015), which introduces seasonal forecasts into the energy forecasting literature for the purpose of both deterministic and probabilistic forecasting. Their case is Italy, where they focus exclusively on summertime electricity demand. This is driven by the use of cooling devices, especially during warm weather periods. They utilize an expansive grid of temperature observations and a regional grid for electricity consumption as their main data sources. To deal with the high-dimensionality of the data they employ a coupled manifold approach which involves subjecting both grids to a Principal Component Analysis (PCA) transformation. This reduces their temperature field from 8892 to 214 dimensions, and their demand field from 7 to 6. For model evaluation they use a leave-one-out cross-validation (applied year by year), utilizing correlation coefficients and Brier Skill Score as performance metrics for the deterministic and probabilistic cases respectively. Results are highly dependant on lead time, the 30+ days ahead forecast far outperforms the 60+ days ahead, especially for probabilistic forecasts. The performance of the probabilistic forecast is similar for their SVM model and their linear regression model. For the deterministic forecasts they find their SVM model to outperform the linear model (with the same inputs) both on average and for 5 of the 7 regions. We were made aware of the work by De Felice et al. at a relatively late stage of our project, thus it had no effect on the planning or testing of our models. We will, however, contrast this approach with our own when applicable.

To summarize, we note, first, that there is a dearth of approaches focused on forecasting demand at the medium-term horizon relative to short- or long-term approaches. Kuster et al. (2017) claims that this is largely because short- and long-term horizons are the most relevant for industry stakeholders. We find this line of reasoning unconvincing, and refer back to the introduction for relevant applications of medium-term forecasting. Kristiansen (2014), which is concerned with forecasting electricity spot price for the Nord Pool market, claims that algorithms in this area often are proprietary, precisely because of stakeholder interest, and therefore represent a lack in the public discussion. Another explanation offered is that forecasting at this horizon is hard because it involves balancing short- and long-term patterns, as well as managing complex high-dimensional data sources (Mirasgedis et al. 2006; De Felice et al. 2015).

We further observe that several different forecast models are employed, including regression, time series and neural nets. No modeling approach is clearly dominant, and no consensus on model choice has been reached. The important link between temperature and energy use has long been known (Valor et al. 2001), and most approaches use temperature information in their forecasts. An overview of natural gas demand forecast comes with the concluding recommendation that the effect on model performance of including weather forecasts instead of just weather measurements should be examined (Tamba et al. 2018). Time information is the other commonly used data source, while other meteorological information (e.g. humidity, and wind) is observed to have

[1] Since this is not a forecasting paper, the use of short-term here is potentially misleading. In this case it refers to the lag in the measured price elasticity, not to forecast horizon.

little or no effect on demand.

Lastly, we note that the field lacks explicit performance benchmarks for set forecast horizons. Moreover, we do not find a common set of baseline models that forecasts are compared against. Further, even if temperature is noted as an important predictor, little emphasis has been put on how much the addition of temperature increases model performance. Overall, the field is relatively new, and valuable contributions could still be given both in terms of specific country-level analysis, but also with regard to comparing methods and results across contexts.

2.2 Probabilistic Temperature Forecasting

As we have seen, the utilization of temperature information in the literature on electricity demand forecasting ranges from taking crude averages to incorporating complex temperature forecasting grids. Our approach resembles De Felice et al. (2015) and involves folding in probabilistic temperature forecasts from Numerical Weather Prediction (NWP) models into our demand forecast. In this section, instead of providing a full overview of temperature forecasting as such, we will restrict our attention to outlining central aspects of NWPs. In addition, we will highlight the discussion of two issues related to NWP ensembles, namely the incorporation of new information in ensemble forecasts; and how to aggregate model predictions over lead times.

Numerical Weather Predictions (NWPs) use mathematical models based on physical principles to forecast future atmospheric conditions. NWPs rely on first characterizing the present state of the atmosphere through observations and data assimilation, before projecting that state forward through solving partial differential equations describing laws of atmospheric motion (Pu et al. 2019). Data assimilation refers to the process by which the model simulations are corrected by observed meteorological states in order to find the best initial conditions before the forecast is issued. The equations are solved numerically and often require both substantial time and computational resources (Brajard et al. 2023).

Ensemble forecasts were introduced in the 1990s and are now the most prevalent form of NWPs. They are composed of a set of ensemble members each being a deterministic simulation of future weather. Acting together, the ensemble forms a distribution, the spread of which reflects the uncertainty in the forecast (Bauer et al. 2015). Since the ensembles rely on simplifications of the physical system they model, they might be biased. They might also produce underdispersed forecasts, meaning that the ensemble spread might not reflect the uncertainty in the forecast by being too narrow. The ensemble therefore requires statistical post-processing where it is re-calibrated according to past performance (Heinrich et al. 2021, Schuhen et al. 2020). Ensemble Model Output Statistics (EMOS) is a common post-processing technique which involves using the spread of the ensemble to improve the uncertainty in predictive distribution of the forecast. The utilization of NWP forecast output as input in other forecasting tasks has been attempted in several fields, including forecasting of demand, and wind-power production (De Felice et al. 2015; Al-Yahyai et al. 2010). The seasonal NWP forecasts we will utilize are further described in Section 4.3.

A problem with ensemble forecasts, such as NWP, is that developing them is often computationally heavy and time-consuming, resulting in long (often monthly) intervals between forecasts. New observations might be available already before the forecast is shipped, rendering them not up-to-date even upon release (Brajard et al. 2023). Since forecast performance drops over time, the end period between forecast issuances might (depending on the model) also see a substantial drop in forecast skill. To update the forecast ensemble and increase model performance it is therefore desirable to incorporate information from the newest available observations into the ensemble forecast.

Lean et al. (2021) approach this by integrating new observations continuously during the process of assimilation (where the model members are corrected by observed weather information) before the NWP forecast issuance. Using this continuous data assimilation framework, they report a reduction in RMSE for medium-range forecasts by $2-3\%$. Schuhen et al. (2020) address the issue of incorporating new observations for short term NWP forecasts by introducing the Rapid Adjustment of Forecast Trajectories (RAFT) method. In addition to standard post-processing the method updates the forecast every time new information becomes available. Based on the error correlation structure within a forecast trajectory, adjustments are estimated by least squares and are unique to each lead time. The adjusted RAFT forecast is obtained by adding the adjustments directly to the EMOS mean forecast. They report an improvement in RMSE forecast skill of 40% on average when updating 32-hours old forecasts based on new data.

Also relevant for our purposes is Brajard et al. (2023), which introduce a weighing method for adjusting the contribution of ensemble forecast members. Their case centers on the Norwegian Climate Prediction Model (NorCPM) model, which assimilates sea surface temperature and hydrographic information through a Kalman Filter. Using a 1-week weighting period they improve the accuracy of the ensemble forecast up to a lead time of two months. Their weighting scheme relies on estimating localised weights at each grid point for each member based on local accuracy. Employing a Bayesian framework, these weights form part of the posterior density for the model state given the recent observations. The weights themselves are proportional to a Gaussian likelihood over the distance between the observed and predicted sea surface temperature. We will revisit this problem in Section 3.5 where we present a method for adjusting ensemble forecasts in principle component space, and in Section 6.4 where we will test this method using NWP input data.

The second problem we will discuss, revolves around how to aggregate model predictions over lead times. Temporal aggregation of time series forecasts is the process of combining observations for the purpose of improved predictive skill and has been utilized since the 1970s. When non-overlapping temporal aggregation is applied to a times series, it filters out high-frequency components, leaving the trend and cyclical patterns (the lower frequency components) to dominate. In this manner the underlying skill of the model might become apparent (Nystrup et al. 2021). A paper relevant for its application to short-term electricity load forecasting in Scandinavia is Nystrup et al. (2021). The method that they develop, takes account of a hierarchy of temporal aggregations at different aggregate levels reconciled through doing an eigendecomposition of the cross-correlation matrix. Even though their case is relevant, their interest lies foremost in the relation between different temporal hierarchies, not in the

specific aggregate levels themselves.

More relevant for our purposes are works that utilize copula estimation methods to deal with the correlation structure across different variables or time points. In a probabilistic setting Henze et al. (2020) utilize copulas to take account of the correlation structure between wind-farms and time points to form probabilistic forecasts of wind energy production. Also, for the application of wind-power production estimation, Pinson et al. (2009) utilizes a copula method for the purpose of estimating the correlation between forecast horizons. Especially relevant for our purposes is Möller et al. (2013). They describe a copula approach meant to take account of the correlation structure across weather variables of different types over time intervals. This involves first estimating the marginal distributions for each variable, and transforming these to latent Gaussian factors. From these, one can estimate a correlation matrix, which one can utilize as a basis for sampling new observations where the correlation between variables is taken account of. They highlight the flexibility of their copula-based approach, as the marginal distributions can be estimated by any method, without the joint distribution being affected.

The problem of forecast aggregation is addressed in Section 3.6, where we describe a copula aggregation method inspired by Möller et al. (2013), and in Section 6.5, where we will show the results of the aggregation method.

CHAPTER 3

Theory

In this chapter we will describe the main theoretical frameworks employed in this thesis. As stated in the introduction the main model we explore is a GAM model which incorporates principal components (PCs) of the temperature grid as feature inputs, for the purpose of electricity demand forecasting. Accordingly, in Section 3.1 we present the main properties of Principal Component Analysis (PCA) as well as a method for obtaining PCs through a Singular Value Decomposition (SVD). Then, in Section 3.2, we describe the main characteristics of Generalized Additive Models (GAM). In Section 3.3 we specify the form of the GAM-PC models which we will explore. We will focus on describing them as structural demand models.

The second problem of this thesis concerns how NWP temperature forecasts can be utilized, within principal component space, to obtain temperature predictions. In Section 3.4 we contrast probabilistic forecasting with point forecasts and describe the methods of Weighted Quantile Estimation (WQE) and Quantile Regression (QR). We then (Section 3.5) introduce a re-weighting scheme specifically for the principal component space. In Section 3.6, we describe methods for performing aggregated forecasts focusing on Gaussian Copulas (GC). We conclude this chapter with Section 3.7 concerning evaluation procedures, and Section 3.8, which presents the evaluation metrics we will employ.

3.1 Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction technique employed in a wide variety of statistical applications (Hastie et al. 2009; Jolliffe 2002; Lay 2021). With regard to forecasting it is often used within the domain of retail, weather and macro-economics (Petropoulos et al. 2022; Esmaeili et al. 2011). PCA works by transforming a (usually large) set of p correlated variables in such a way that most of the variation within the original set is concentrated within a small subset of the new variables. The new variables are called principal components (PCs). They are uncorrelated and ordered after which variable explains most of the variance; the first PC explaining the most. The dimensionality reduction is performed by keeping only the first $m < p$ principal components which account for most of the variation. The choice of m is usually based on the variance explained by each PC. For our forecast application we will use a combination of heuristics and testing to decide upon

m (see Sections 4.2 and 5.4).

Let $\mathbf{X}$ be a centred $n \times p$ matrix, with column vectors $\mathbf{x}_j$, whose column means are $\bar{x}_j = 0$, with $\mathbf{S}^* = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$ being its diagonalizable, positive semi-definite $p \times p$ sample covariance matrix. For convenience we will work with $\mathbf{S} = \mathbf{X}^T\mathbf{X}$ which will yield the same results up to a constant. The eigendecomposition of $\mathbf{S}$ is then given by:

$$\mathbf{S} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}, \tag{3.1}$$

where $\mathbf{V}$ is a $p \times p$ matrix containing the eigenvectors of $\mathbf{S}$, and $\mathbf{D}$ is a diagonal matrix whose diagonal elements λ_{jj} are the eigenvalues of $\mathbf{S}$. Since $\mathbf{S}$ is positive semi-definite all eigenvalues are positive. Consider, then, the linear combination across all p columns of $\mathbf{X}$:

$$\mathbf{x}_1 a_1 + \mathbf{x}_2 a_2 + ... + \mathbf{x}_p a_p = \mathbf{X}\mathbf{a}, \tag{3.2}$$

where $\mathbf{a} \in \mathbb{R}^p$ is a vector of constants. We then define the first principal component as the vector $\mathbf{c}^1 = \mathbf{X}\mathbf{a}_1 \in \mathbb{R}^n$ to be the version of (3.2) with the highest sample variance. We therefore want to find the vector $\mathbf{a}_1$ that provides the maximum of $\mathrm{Var}(\mathbf{X}\mathbf{a}) = \mathbf{a}^T\mathbf{S}\mathbf{a}$. This can be found by introducing the normalizing constraint $\mathbf{a}^T\mathbf{a} = 1$, in a Lagrange multiplier optimization problem (Jolliffe 2002):

$$\mathbf{a}_1 = \underset{\mathbf{a}\in\mathbb{R}^p}{\operatorname{argmax}}\ \mathbf{a}^T\mathbf{S}\mathbf{a} - \lambda(\mathbf{a}^T\mathbf{a} - 1), \tag{3.3}$$

where λ is the Lagrange multiplier and the constraint is specified in the parenthesis. By differentiation with respect to $\mathbf{a}$ and setting the expression equal to zero to find the maximum value, we get:

$$\begin{aligned} \mathbf{S}\mathbf{a} - \lambda\mathbf{S} &= \mathbf{0} \\ (\mathbf{S} - \mathbf{I_p}\lambda)\mathbf{a} &= \mathbf{0} \end{aligned} \tag{3.4}$$

The first solution, obtained by setting $\mathbf{a} = \mathbf{0}$, is not viable as it yields the minimum at $\mathrm{Var}(\mathbf{X}\mathbf{a}) = \mathbf{0}$. The second solution is obtained when the covariance matrix equals the multiplier vector or equivalently when $\mathbf{S}\mathbf{a} = \lambda\mathbf{a}$, which is the standard eigenvalue equation. This allows us to re-frame the question, as it means that the Lagrangian multiplier λ can be found as an eigenvalue λ_{jj} along the diagonal of $\mathbf{D}$, and that we can find $\mathbf{a}$ as the corresponding eigenvector $\mathbf{v_j} \in \mathbf{V}$. Utilizing the identity implicit in (3.4), and the normalizing constraint, we can reformulate the variance of (3.2):

$$\begin{aligned} \mathrm{Var}(\mathbf{X}\mathbf{a}) &= \mathbf{a}^T\mathbf{S}\mathbf{a} \\ &= \mathbf{a}^T\lambda\mathbf{a} \\ &= \lambda\mathbf{a}^T\mathbf{a} \\ &= \lambda. \end{aligned} \tag{3.5}$$

Finding the largest variance, then, is tantamount to finding the largest eigenvalue (and this must also be the maximum). If we denote by λ_1 the largest eigenvalue, and $\mathbf{v}_1$ the corresponding eigenvector, then the vector of constants that maximizes the variance of (3.2) is $\mathbf{a}_1 = \mathbf{v}_1$. The first eigenvector, $\mathbf{v}_1$, points in the direction of the most variance, and the first principal component is the

linear combination of the columns of $\mathbf{X}$ with the highest sample variance. That is to say, it describes the most predominant structures of the original data. Each subsequent jth principal component is created by finding the jth largest eigenvalue and its corresponding eigenvector. Each $\mathbf{v}_j$ points in a direction orthogonal to all preceding eigenvectors; each one pointing in a direction of less variance than the previous. On a terminological note, the literature on PCA across disciplines employ many terms for both eigenvectors and principal components (including loading, coefficient and empirical orthogonal function) (Jolliffe 2002). Most confusingly different parts of the literature refer either to the eigenvectors of $\mathbf{S}$ (Lay 2021) or to the linear combination (3.2) (Jolliffe 2002) as principal components. We will stick to the latter usage.

PCA by Way of SVD

To obtain the principal components we only need to find the eigenvector-eigenvalue pairs. A computationally efficient method for this purpose is the Singular Value Decomposition (SVD) (Jolliffe 2002). The SVD can be attained for any (finite-dimensional) matrix $\mathbf{X}$, and is given by:

$$\mathbf{X} = \mathbf{U}\mathbf{L}\mathbf{A}^T, \tag{3.6}$$

where $\mathbf{U}$ is a $n \times p$ orthogonal matrix containing the left singular vectors, $\mathbf{L}$ is a $p \times p$ diagonal matrix, containing the singular values, and $\mathbf{A}$ is a $p \times p$ orthogonal matrix with columns called the right singular vectors. We can then show that $\mathbf{S}$ can be expressed through the components of the SVD of $\mathbf{X}$ (Hastie et al. 2009):

$$\begin{aligned}
\mathbf{S} &= \mathbf{X}^{\mathrm{T}}\mathbf{X} \\
&= (\mathbf{U}\mathbf{L}\mathbf{A}^{\mathrm{T}})^{\mathrm{T}}(\mathbf{U}\mathbf{L}\mathbf{A}^{\mathrm{T}}) \\
&= \mathbf{A}\mathbf{L}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathbf{L}\mathbf{A}^{\mathrm{T}} \\
&= \mathbf{A}\mathbf{L}^{\mathrm{T}}\mathbf{L}\mathbf{A}^{\mathrm{T}} \\
&= \mathbf{A}\mathbf{L}^2\mathbf{A}^{-1}.
\end{aligned} \tag{3.7}$$

Since this is on the same form as (3.1) this is also an eigendecomposition of $\mathbf{S}$. One can therefore obtain the eigenvectors and eigenvalues of $\mathbf{S}$ from the elements of the SVD decomposition of $\mathbf{X}$. The eigenvalues are found in $\mathbf{A}$ and the eigenvalues are found as $\mathbf{L}^2$. From these we can form the principal component matrix as $\mathbf{C} = \mathbf{X}\mathbf{A}$ or equivalently as $\mathbf{C} = \mathbf{U}\mathbf{L}$.

A computational advantage can be obtained by applying the SVD directly to $\mathbf{S}$. By utilizing the identity obtained in (3.7) one can form:

$$\begin{aligned}
\mathbf{S} &= \tilde{\mathbf{U}}\tilde{\mathbf{L}}\tilde{\mathbf{A}}^T \\
&= \mathbf{A}\mathbf{L}^2\mathbf{A}^{-1} \\
&= \mathbf{A}\mathbf{L}^2\mathbf{A}^T.
\end{aligned} \tag{3.8}$$

Because $\mathbf{S}$ is a $p \times p$ matrix, so is $\tilde{\mathbf{U}}$. Which means we can obtain the eigenvectors by $\tilde{\mathbf{U}} = \mathbf{A} = \tilde{\mathbf{A}}$, while $\tilde{\mathbf{L}} = \mathbf{L}^2$ yields the eigenvalues. A property of this decomposition is that it is antipodaly symmetric. This means both $\tilde{\mathbf{U}}$ and $-\tilde{\mathbf{U}}$ will provide valid decompositions, i.e. the identity $\tilde{\mathbf{U}} = \mathbf{A}$ is only accurate

up to the sign. The practical consequences of this are minor. The sign itself ought not affect the predictive performance of models including the PCs, but it will affect the sign of the model coefficients. For our purposes we have chosen the sign direction which aligns the first PC with the electricity demand. This will be further explored in Section 4.2. Applying the SVD directly to $\mathbf{S}$ speeds up computations by a factor of 3.5 at 80.000 time points (this includes forming $\mathbf{S}$). Compared to applying SVD on $\mathbf{X}$ the results are equivalent up to a tolerance level close to machine precision, with a mean relative difference of matrix elements of 3.12e-15.

In our applications we utilize PCs in two manners. The first manner involves obtaining pre-trained eigenvectors, once only, from a specified time interval that provide a 'historic' representation of the structure of the temperature grid. These will then be applied to later occurring training and test observations of the temperature observations. The other approach involves iteratively obtaining new eigenvectors for each monthly forecast issuance. This provides an 'up to date' depiction of the temperature grid structure.

Either way we will use the same general framework, specified in Algorithm 1. We first find the appropriate eigenvectors based on a training set, $\mathbf{X}^{train}$, which is either 'historic' or 'up to date'. We subsequently apply these eigenvectors to an 'up to date' training set, $\mathbf{X}^{up}$, to form the training PCs, $\mathbf{C}^{train}$. Note that depending on the application we might have $\mathbf{X}^{train} = \mathbf{X}^{up}$. We then apply the eigenvectors from $\mathbf{X}^{train}$ to the test set, $\mathbf{X}^{test}$, to obtain $\mathbf{C}^{test}$. The final output is then acquired by applying $\psi_j(\cdot)$, a function selecting the principal component corresponding to the jth largest eigenvalue. Once the PC matrix is formed $\psi_j(\cdot)$ is usually applied for $\forall\, j \leq m$, but we will also test different combinations of PCs (see Section 5.3).

From this structure it is clear that we do not always apply the PC transformation back on the original training data, but also on new train and test observations. In this manner we avoid contaminating the training data with values from the test set. The forecasts we form will be based on: 1) a model structure learned from the relation between a set of PC vectors, $\mathbf{c}_j^{train}$, and demand; and 2) input values, $\mathbf{c}_j^{test}$, which are PCs formed from eigenvectors derived from the training set. The selection of principal components to include in our model is then subject to forecast performance on the test set.

3.2 Generalized Additive Models (GAM)

Following notation by Agresti (2015), we let y_i be a random response variable for observations $i = \{1, ..., n\}$, with mean $\mu_i = E[y_i]$. Further, let $\mathbf{X}$ be an $n \times p$ model matrix, with explanatory variables x_j as columns and with row inputs $x_i = (x_{i1}, ..., x_{ip})$. A Generalized Linear Model (GLM) is a regression model on the form:

$$g(\mu_i) = \eta_i, \tag{3.9}$$

where the link function, $g(\cdot)$, connects the mean of the response to a linear predictor $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$. This is a generalization from the ordinary linear model which we obtain by using the identity link $g(\mu_i) = \mu_i$ and a normal response, y_i, which we can write as:

$$y_i = \eta_i + \epsilon_i, \tag{3.10}$$

Algorithm 1 Finding Principal Components

Input: $\mathbf{X}^{train}$, $\mathbf{X}^{up}$, $\mathbf{X}^{test}$, j.
for $i = 1$ **to** p **do**
  $\mathbf{x}_i \leftarrow \mathbf{x}_i^{train} - \bar{x}_i^{train}$ ▷ Center columns of $\mathbf{X}^{train}$.
  $\ddot{\mathbf{x}}_i \leftarrow \mathbf{x}_i^{up} - \bar{x}_i^{up}$ ▷ Center columns of $\mathbf{X}^{up}$.
  $\tilde{\mathbf{x}}_i \leftarrow \mathbf{x}_i^{test} - \bar{x}_i^{train}$ ▷ Center $\mathbf{X}^{test}$ using column means from $\mathbf{X}^{train}$.
$\mathbf{X} \leftarrow (\mathbf{x}_1, ..., \mathbf{x}_p)$ ▷ Form centered training matrix $\mathbf{X}$.
$\ddot{\mathbf{X}} \leftarrow (\ddot{\mathbf{x}}_1, ..., \ddot{\mathbf{x}}_p)$ ▷ Form centered 'up to date' matrix $\mathbf{X}^*$.
$\tilde{\mathbf{X}} \leftarrow (\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_p)$ ▷ Form centered test matrix $\tilde{\mathbf{X}}$.
$\mathbf{S} \leftarrow \mathbf{X}^t\mathbf{X}$ ▷ Find covariance matrix up to a factor.
$\tilde{\mathbf{U}}\tilde{\mathbf{L}}\tilde{\mathbf{A}}^T \leftarrow \mathbf{S}$ ▷ Perform SVD decomposition directly on $\mathbf{S}$.
if $\mathbf{X}^{train} \neq \mathbf{X}^{up}$ **then**
  $\mathbf{C}^{train} \leftarrow \ddot{\mathbf{X}}\tilde{\mathbf{U}}$ ▷ Transform 'up to date' training data to PC space.
else
  $\mathbf{C}^{train} \leftarrow \mathbf{X}\tilde{\mathbf{U}}$
$\mathbf{C}^{test} \leftarrow \tilde{\mathbf{X}}\tilde{\mathbf{U}}$ ▷ Transform test data to PC space.
Output: $\mathbf{c}_j^{train}, \mathbf{c}_j^{test} \leftarrow \psi_j(\mathbf{C}^{train}), \psi_j(\mathbf{C}^{test})$ ▷ Obtain jth PCs.

where the random component, $\epsilon_i \sim N(0, \sigma^2)$, specifies the distribution of the residuals (Agresti 2015). A Generalized Additive Model (GAM) is an extension of the GLM that allows the specification of the linear predictor with flexible smoothing terms while keeping the additive structure as well as the customizability of both $g(\cdot)$ and the choice of distribution (Wood 2017; Hastie et al. 2009). It has the form:

$$g(\mu_i) = f_1(x_{i1}) + f_2(x_{i2}) + ... + f_p(x_{ip}), \tag{3.11}$$

where each $f_j(\cdot)$ can be customized to be a regular linear term, e.g. $\beta_j x_{ij}$, or to be smooth functions $s(\cdot)$, also called splines. In the special case of all $f_j(\cdot)$-functions being linear, then (3.11) defaults to a GLM. The smooth function is a sum of weighted basis functions $b_k(\cdot)$ applied to the same covariate input x_{ij}:

$$s(x_{ij}) = \sum_{k=1}^{K} \beta_k b_k(x_{ij}). \tag{3.12}$$

In contrast to polynomials, which are defined on the whole range of the covariate x_j, each basis function might only affect a small interval of x_j-values. This constitutes the main advantage of utilizing the GAM framework: it allows for more flexible modelling options. The coefficients for the basis functions of the spline terms are found by minimizing the sum of squares subject to a penalty term:

$$PRSS(f_1, ..., f_p) = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} f_j(x_{ij}) \right)^2 + \sum_{j=1}^{p} \lambda_j \int \left(\frac{\partial^2 f_j(x_j)}{\partial x_j^2} \right)^2 dx_j. \tag{3.13}$$

The first part of this objective function penalizes the model fit. The second is a penalty term controlled by the smoothing parameter λ. It penalizes the amount

of wigglyness, expressed as the second derivative of each spline. The coefficients are found by the back-fitting algorithm. To update the function estimate $\hat{f}_j(\cdot)$ it applies a smoothing term to residuals obtained by fitting all functions in the model except $\hat{f}_j(\cdot)$. This is iteratively performed for each function until the change in each $\hat{f}_j(\cdot)$ sinks below a threshold (Agresti 2015; Hastie et al. 2009).

In our application we have used the `gam`-function from the **mgcv** (mixed GAM computational vehicle) package. When modelling PCs we have used the default thin plate spline setting for finding the coefficients. This is a low rank smoother which automates the selection of smoothing parameters in the basis functions. It utilizes an eigendecomposition, not of the input, but of the basis spline functions to reduce the size of the basis spline expansion, where the term k sets the dimensionality of the basis expansion. With regard to the specification of k we found that the default setting was adequate without further adjustment. For the cyclical time covariate terms we have used cyclical splines to satisfy the requirement that the ends meet up at the end of each cycle (see Section 5.2). The smoothing parameter λ is found by Generalized Cross-Validation (GCV), it is the default option and recommended for minimizing out-of-sample error.

3.3 Structural Demand Model

Having outlined the central facets of Principal Component Analysis and Generalized Additive Models we are now in position to present the key model form that we want to explore in Chapter 5 concerning electricity demand forecasting. The aim of exploring this model is to establish a structural relationship between demand and temperature focused on predictive accuracy. We will first present the model form, before we describe the specific characteristics of our modelling approach.

Let $y_t \in \mathbb{R}_+$ be a random variable representing electricity demand at forecast target time t. Then let $x_t = (x_{t1}, ..., x_{tp+1})$ be an input vector consisting of an intercept and p time covariates (e.g. 'hour', 'month', etc), while we let $z_t = (z_{t1}, ..., z_{td})$ be a d-length input vector indicating temperature information. A further presentation of the data is given in Chapter 4. Let further $f(x_t)$ be some function specifying any combination of the time covariates which might include an intercept, linear terms, interaction terms or splines. And let $h(z_t)$, similarly, be some function of the temperature information. The model form of the structural electricity demand model we want to explore is a GAM which we write as:

$$y_t = f(x_t) + h(z_t) + \epsilon_t, \tag{3.14}$$

where $\epsilon_i \sim N(0, \sigma^2)$. For the sake of model simplicity we assume normal errors and we employ the identity link $g(\mu_t) = \mu_t$, which is the canonical link function for a normal response variable. Since the observed demand is $y_t > 0 \, \forall \, t$ we could have opted for a log-link to constrain the output to be positive. In practice, our model estimates far exceed 0, so the log-link is not needed.

Of special interest (and thus anticipating the results) is the GAM-PC1+2-model. This is a version of (3.14) which incorporates the first two principal components of the temperature field. Let the jth PC be a function of the temperature information, i.e. $C_t^j = \psi_j(z_t)$, and we can then specify $h(z_t)$ by two smoothing terms $s(\cdot)$ applied to C_t^1 and C_t^2 at target time t:

$$y_t = f(x_t) + s(C_t^1) + s(C_t^2) + \epsilon_t. \tag{3.15}$$

The performance of this model is investigated in Section 5.4.

We have approached the demand forecasting problem as a structural modelling task. Expanding on an outline from Shumway et al. (2017) we take this to involve three aspects:

- First, we model trend and seasonality by time covariates, through $f(x_t)$, which act as fixed effect terms.
- Second, we make the simplifying assumption that the time dependency is captured by the time covariates so we do not attempt to model the correlation structure between timed observations explicitly.
- Third, we assume we have access to future temperature observations, or equivalently, perfect forecasts.

The structural model builds a relation between variables at target time t. By working on a good specification of this model we can establish a straight-forward interpretable model framework, which is easy to expanded upon. Its main utility, then, is as a cog in a larger model framework. Specifically, it facilitates the incorporation of probabilistic inputs of temperature.

As forecast engines issued by trusted weather services, we assume the NWP forecasts provide a good representation of the observed temperature field, including the correlation structure between time points. Under this assumption we will rely on the NWP forecasts to handle the correlated aspects of the demand time series related to temperature. Instead of working with the correlation structure explicitly through more traditional time series modeling, our strategy is to work with the correlation structure implicitly through the incorporation of NWP forecast information. Now, there are some correlation structures specific to the demand observations that is not captured by the inclusion of temperature in the model, that we have not worked with. We consider this work to be a possible but less central expansion of our model.

When the GAM-PC models are fed forecast inputs we no longer refer to them as structural models, but as probabilistic models that incorporate forecast information. To make this distinction clear we utilize a specific notation scheme to distinguish when we use 'perfect forecast information' in the demand model and when we utilize actual NWP forecasts. Let us first note that the structural demand model is a point forecast. The point forecast is an estimate of the conditional expectation of a random variable, y_t, whose forecast is issued at time $t-k$ for a given lead time k, thus being realized at target time t. Slightly altering the notation from Pinson et al. (2009) we write this as $\hat{y}_{t|t-k}$. Now, when we use perfect forecast information, the conditioning on the issuance time becomes irrelevant. If we have perfect information we will form the same forecast whether we are one month or one year away from the target time. We indicate this by omitting the reference to the issuance and simply write $\hat{y}_t$. In Chapter 5 we only concern ourselves with models where we assume we have perfect knowledge about future temperature. When dealing with NWP forecasts, however, the conditioning on issuance time becomes vital since performance is directly related to the lead time.

To summarize: The demand forecasting problem involves specifying the best performing combination of different versions of $f(x_t)$ and $h(z_t)$ with regard to prediction accuracy for a point forecast. Problem 2, on the other hand,

concerns finding probabilistic forecasts of temperature PCs, C_t^j, which we can fold into the electricity demand model. In the following sections we turn to the theoretical framework for the temperature forecasting problem.

3.4 Probabilistic Forecasting

A probabilistic forecast, in contrast to the aforementioned point forecast, takes uncertainty into account and gives a fuller picture of possible realizations of the predicted variable (Henze et al. 2020). To describe this uncertainty we use different forms of quantile estimators, which we denote by $B^{\alpha}(\cdot)$.[1]

Consider the random variable $y_t \in \mathbb{R}$, at time t, with its strictly increasing cumulative distribution function $F_t(\cdot)$. The quantile $q_t^{\alpha} \in \mathbb{R}$, for a specified proportion $\alpha \in [0, 1]$, is then uniquely defined by the inverse CDF (Pinson et al. 2009):

$$q_t^{\alpha} = F_t^{-1}(\alpha). \tag{3.16}$$

The quantile value itself forms the dividing line of the value range where the probability of observing a realization that is lower than some q_t^{α} equals α, i.e. $P(y_t < q_t^{\alpha}) = \alpha$. The aim of a quantile estimator, $B_{Y_t}^{\alpha}(\cdot)$, is to estimate the inverse CDF of y_t at a specified α, through the utilization of some input, thus obtaining an estimated quantile:

$$\hat{q}_t^{\alpha} = B_{y_t}^{\alpha}(\cdot). \tag{3.17}$$

Thus, in contrast to Bayesian approaches, where any posterior quantile or interval can be obtained from the same posterior predictive distribution (Gelman 2013), the quantile estimators, $B_{y_t}^{\alpha}(\cdot)$, must be refitted for each α (e.g. by obtaining α-specific coefficients, β^{α}). In some applications we will also utilize a range of m quantile forecasts (typically $m = 9$), which together form the predictive distribution set $\hat{\mathcal{B}}_{y_t}$(Pinson et al. 2009):

$$\hat{\mathcal{B}}_{y_t} = \{\hat{q}_t^{\alpha_i} | 0 < \alpha_1 < \ldots < \alpha_m < 1\}. \tag{3.18}$$

Our application of probabilistic forecasting is detailed in Chapter 6. It concerns the task of forecasting principal components of ERA5 temperature, C_t^j, from principal components of monthly issued NWP forecasts. Of special interest for this purpose is estimating the 0.9-quantile of the first temperature PC. We refer to this as a quantile forecast of C_t^1 at $\alpha = 0.9$ and it involves finding a good expression for $B_{C_t^1}^{0.9}(\cdot)$. The 0.9-quantile is of importance as it marks the PC value associated with especially cold deviations from the temperature mean. In Section 4.2 we will establish that this corresponds with periods of high energy demand.

Before we move on to describe these methods it is necessary to introduce the structure of the NWP forecasts which will form the backbone of our probabilistic modelling. Let $\mathcal{N}_{i+k|i}^{1:M}$ be a collection of M matrices each containing forecasts for an NWP ensemble member. At the beginning of each month, at forecast

[1] Alternative approaches include making interval or density forecasts, e.g. through Bayesian modelling.

issuance time i, each member, m, issues a $k \times p$ forecast matrix, $\mathbf{N}^m_{i+k|i}$ over lead times $k = \{1, \ldots, 500\}$, where each step is a 6-hour interval over 125 days, covering $p = 462$ grid points.

The subscript notation makes explicit the relation between the target time, $t = i + k$, and the forecast issuance time i. For matrices we use this notation over a set of lead times k. For individual forecasts we will use the subscript $t|t-k$, indicating the forecast which is targeting time t, and issued at time $t-k$, for a specific lead time k. We use this notation consistently in order to be able to refer to models that include data from different forecasts. To form a forecast we depend both on a model and predictor values for the target time. These predictor values must either be forecasted, in which case they are dependent on lead time, or we can assume perfect information, which means the predictors are not dependent on lead time. We prefer this notation as it also enables us to contrast elements in a model which has lead time critical information (where we use the conditional subscript) from elements that do not (where we only refer to the target time).

Even though the NWPs are issued as point forecasts, together they effectively constitute an estimate of the temperature distribution for each grid point at each target time. The same perspective can be applied in principal component space: Let $\tilde{\mathbf{U}}$ be a $p \times p$ matrix of eigenvectors obtained from the SVD of the covariance matrix of historic ERA5-temperature data (as described in Section 3.1). We can then obtain the jth principal component for NWP member m by applying the function $\psi_j(\cdot)$ (which selects the PC vector corresponding to the jth largest eigenvalue) on the transformed forecasts:

$$\hat{C}^{j,m}_{i+k|i} = \psi_j(\mathbf{N}^m_{i+k|i}\tilde{\mathbf{U}}). \tag{3.19}$$

Since this is a principal component transformation of the original forecast, we view this as a point forecast in PC space, hence the specification of issuance time. Obtaining this for all NWP members we form $\mathcal{C}^{j,1:M}_{i+k|i}$, a set as of M PC vectors. This set can be used as an estimate of the distribution of the jth ERA5 temperature PC at every target time $t = i + k$. For a specific target time t and a specific lead time k we write this set as $\mathcal{C}^{j,1:M}_{t|t-k}$. Our strategy for estimating a quantile of C^j_t, then, for a specific target time t with lead time k is to apply a quantile estimator, of some form, with the above mentioned set as input:

$$\hat{q}^{\alpha}_{t|t-k} = B^{\alpha}_{C^j_t}(\mathcal{C}^{j,1:M}_{t|t-k}). \tag{3.20}$$

In the following we describe the main methods we use for estimating PC quantiles, namely quantile regression, weighted quantile estimation, re-weighted quantile estimation, and copula quantile estimation. These methods should not necessarily be viewed as in competition with each other, but as tools for solving slightly different problems.

Weighted Quantile Estimation

The most basic quantile estimation modelling framework we consider is called Weighted Quantile Estimation (WQE). It will also be utilized as a building block by other models described below. The WQE model employs a standard

sample estimator on the set of NWP principal components and applies this straight-forwardly as a forecast of the temperature PC quantile.

For this purpose we utilize the `quantile`-function implemented in the **stats** package in R. This is not a consecutive order statistic where the quantile of interest is found directly from the order, but a weighted average of said order. We use the default method (Type 7) which employs a linear interpolation between points (α_k, x_k) where $\alpha_k = \frac{k-1}{n-1}$ is the modal position of the k'th ordered observation of a vector $\mathbf{x}$. The quantiles are then given by a weighted average over two ordered observations:

$$W_X^\alpha(\mathbf{x}) = (1-\gamma)x_j + \gamma x_{j+1}, \tag{3.21}$$

where the weight is given by $\gamma = (n-1)\alpha - j + 1$, and the index is $j = \lfloor (n-1)\alpha + 1 \rfloor$, with n being the sample size (Hyndman et al. 1996;R Core Team 2022).

At each forecast issuance time, the WQE-model forms, for a specific lead time k and target time t, a sample quantile estimate based on the members of the set of NWP PCs $\mathcal{C}_{t|t-k}^{j,1:M}$:

$$\hat{q}_{t|t-k}^\alpha = W_{C_t^j}^\alpha(\mathcal{C}_{t|t-k}^{j,1:M}). \tag{3.22}$$

We then make the assumption that the members of the set of NWP PCs, $\mathcal{C}_{t|t-k}^{j,1:M}$, can be seen as sample realizations of the same underlying variable C_t^j that we want to model. The WQE model for the PC temperature quantile forecast task then utilizes the estimate $\hat{q}_{t|t-k}^\alpha$ directly as an input, $q_{t|t-k}^\alpha$, in a forecast of the jth principal component at quantile α:

$$C_t^{j,\alpha} = q_{t|t-k}^\alpha. \tag{3.23}$$

The error term here is left unspecified, and we make no assumption with regard to its distribution in the modelling itself, like in the case of quantile regression described below. The performance of this model is detailed in Section 6.2.

Quantile Regression

Quantile Regression (QR) is a standard modelling approach where instead of modelling the conditional expected value based on a set of covariates, we can instead utilize these covariates to model a range of conditional quantiles (Koenker 2005). This approach provides a more complete picture of the relation between variables, especially when said relation differs at different levels of the response (Cade et al. 2003). At each α-quantile QR models describe the conditional quantile realizations as a linear combination of covariates. For a response variable, $\mathbf{y} \in \mathbb{R}^n$, and a design matrix $\mathbf{X}$, the quantile regression model is given as:

$$Q_{y_t}^\alpha(\mathbf{X}) = \mathbf{X}\beta^\alpha. \tag{3.24}$$

The β-coefficients, which are specific to each quantile α, are then found by minimizing the pinball loss function (see also Section 3.8) given by:

$$\begin{aligned}\rho_\alpha(x) &= x(\alpha - I_{x<0}) \\ &= \begin{cases} x\alpha, & \text{if } x \geq 0; \\ |x|(1-\alpha), & \text{if } x < 0. \end{cases}\end{aligned} \tag{3.25}$$

Specifically for the regression task we look at minimizing the difference between the observed y_t and the linear predictor $\eta_t = x_t\beta$:

$$\hat{\beta}^\alpha = \underset{\beta \in \mathbb{R}^k}{\operatorname{argmin}} \sum_{t=1}^{n} \rho_\alpha(y_t - x_t\beta). \tag{3.26}$$

In this manner quantile regression minimizes a sum of asymmetric penalties (Koenker 2005). The QR-model offers a semi-parametric structure of the estimates. While the deterministic part can be described in terms of model parameters (quantile specific β-coefficients), the error term ϵ_t is not assumed to take on any specific distributional form. This is advantageous for forming prediction intervals, e.g. compared to OLS, in cases where the error departs from the assumed distribution (Cade et al. 2003).

In Section 6.3 we will, again for the purpose of modelling PC temperature quantiles, explore a set of quantile regression models at target time t, for a specific lead time k, of the form:

$$Q^\alpha_{C^j_t}(x_t, q^\alpha_{t|t-k}) = \beta_0 + f(x_t) + h(q^\alpha_{t|t-k}), \tag{3.27}$$

where $f(x_t)$ is a function of time covariates (e.g. hour, month, etc.) which may include interaction terms, and the inputs, $q^\alpha_{t|t-k}$, are the WQE-output described in the previous section. In the linear case we have $h(q^\alpha_{t|t-k}) = \beta_1 q^\alpha_{t|t-k}$. This parametrization will allow us to explore to what degree the WQE-estimates should be adjusted for improving forecasting performance. We will also look at a spline version of $h(\cdot)$. In this case the splines were implemented through the `bs`-function from the **splines**-package, which flexibly fits each quantile by a piecewise cubic polynomial procedure (Koenker 2005).

3.5 Re-weighted Quantile Estimation

NWP forecasts are issued once monthly, and (as we will demonstrate in Chapter 6) at the hourly level performance with regard to forecasting temperature PCs degrades fairly rapidly. Substantial short term improvements can be achieved, however, by re-issuing a modified version of the forecast, at time point R, based on a quantile re-weighting procedure. In this manner we can 'regain' forecast skill before a new, proper, forecast is issued. This re-weighting is available at any time point as it only relies on the r most recently observed temperature PCs at hand, and can be repeated as many times as desired.

The idea is to adjust the set of NWP principal components according to their performance during the last r time points for which we have observed realizations. For this purpose we utilize a standard weighting scheme. The weight for NWP ensemble member m is based on the squared distance between the observed

and forecasted PC. The adjustments, then, are based on information from the level of expected value of the output, but is performed at the level of a specific quantile. The best predictors of the expected value are weighted upwards, and the worst are weighted down. We are in a way shifting the distribution of the NWP forecast members closer to the observed values by filtering out the contribution of the worst performing members.

The re-weighting procedure consists of two parts and is described in algorithm 2, where the second part follows the general outline of Akinshin (2023). The first part concerns finding what we refer to as the importance weights.

Let R be the last time point for which we have observed temperature data, and let it also mark the re-issuance of the forecast. The importance weights, w_R^m, for an individual member m, is then found by summing over a re-weighting interval with length r:

$$w_R^m = \sum_{t=R-r+1}^{R} -\frac{1}{2}\gamma(C_t - \hat{C}^m_{t|t-k})^2. \tag{3.28}$$

The tuning parameter γ controls the overall size of the weight adjustment and the process of finding a good value for γ is detailed in Section 6.4. Notice that the acquired weight only references the re-issuance, at time point R, not any specific future time point. This reflects that we apply the same weights to all relevant target time points $t > R$. This is done under the assumption that the trajectories of the best performing NWP members will continue to reflect the temperature distribution more accurately than the unweighted ensemble as a whole. In Section 6.4 we test for how long this assumption holds.

The importance weights for each member is collected in a vector, $\mathbf{w_R} = (w_R^1, ..., w_R^M)^T$, before being normalized through a softmax function. To ensure numerical stability we also include a maximum value subtraction. For each member we find the normalized importance weight:

$$\bar{w}_R^m = \frac{\exp\{w_R^m - \max(\mathbf{w_R})\}}{\sum_{m=1}^{M} \exp\{w_R^m - \max(\mathbf{w_R})\}}. \tag{3.29}$$

These importance weights quantify how well each NWP member performs estimating the observed PC relative to each other.

From here one could easily obtain a re-weighted quantile estimate through a single order statistic. After sorting the member values in ascending order, one lets the corresponding weights follow the same order. The weight index is then found as $m^* = \inf\{s : \sum_{i=1}^{s} w^i \geq \alpha\}$, and the quantile value estimate is given by $q^\alpha = C^{m^*}$. Because the quantile is selected directly from the ensemble the obtained estimates might differ considerably from the true underlying distribution, especially for small sample sizes (Akinshin 2022). Though our sample size of ensemble members is not particularly small, ranging between 50-200 depending on forecast month, the effective sample size can be much smaller as a results of the weighting process. If a substantial number of weights are close to zero the single ordered statistic will decrease in accuracy. In order to ensure good quantile estimates we will instead utilize a weighted sum of all order statistics.

Our approach therefore centers around finding the contribution weights, $\widetilde{w}_R^m$, which specifies the contribution of each NWP member to the estimation of a

specific quantile. The contribution weights will sum up to 1, so we build the re-weighted quantile, $\tilde{q}^{\alpha}_{t|t-k}$, up as a weighted sum from the values of the input and the member specific weight:

$$\tilde{q}^{\alpha}_{t|t-k} = \sum_{m=1}^{M} \widetilde{w}^{m}_{R} C^{m}_{t|t-k}. \tag{3.30}$$

The contribution weights are found by applying a weighted quantile estimation function. We utilize the Harrell-Davis quantile estimator (Akinshin 2023), which has a higher statistical efficiency than single order statistics. It is also especially suited for light-tailed distributions (Akinshin 2022), such as the normal distribution, which is the assumed distribution of the ensemble after post-processing.

The estimator is based on the beta distribution, whose PDF is defined over the support $x \in [0, 1]$ as:

$$\text{Beta}(a, b) = \frac{1}{B(a, b)} x^{a-1}(1 - x)^{b-1}, \tag{3.31}$$

where $B(a, b)$ is the beta function. Based on the desired α-quantile we can specify the parameters as $a = \alpha(n^* + 1)$ and $b = (1 - \alpha)(n^* + 1)$. This will yield a smooth uni-modal density curve with the mode close to α,[2] but slightly adjusted by Kish's effective sample size $n^* = \frac{(\sum_{i=m}^{M} \bar{w}^{m}_{R})^2}{\sum_{i=m}^{M} (\bar{w}^{m}_{R})^2}$. The adjustments are larger for smaller sample sizes and will drag the mode from the quantile in the direction away from the median. The size of the performance weights for each member are then used to divide the support of the distribution into M intervals. For ordered member m the lower interval border along the support is formed as $l_m = \sum_{j=1}^{m-1} \bar{w}^{j}_{R}$, which is the sum of the normalized performance weights up to and including the previous member. This applies to all members expect the first, where we have $l_1 = 0$. For the upper border we have $u_m = l_{m+1}$. The contribution weight of a member is then found as the area under the density curve, over the support designated by its interval. It can be obtained through employing the regularized incomplete beta function: $I_x(a, b) = \frac{1}{B(a,b)} \int_0^x t^{a-1}(1 - t)^{b-1}\, dt$, which also is the CDF of the beta distribution. In this manner the contribution weight of a member is a function of the performance weight, the sample size, and 'roughly' its proximity to the desired quantile, and is given by (Akinshin 2023):

$$\widetilde{w}^{m}_{R} = I_{u_m}(a, b) - I_{l_m}(a, b). \tag{3.32}$$

We describe the application of the re-weighted quantiles in Section 6.4. Note that the re-weighted models will be identical to the previously described WQE and QR-models expect for using the re-weighted quantile, $\tilde{q}^{\alpha}_{t|t-k}$, as input instead of the $q^{\alpha}_{t|t-k}$.

Through the fact that we are adjusting forecast outputs based on new observations the re-weighting method we employ here bears some resemblance to both Brajard et al. (2023) and Schuhen et al. (2020). There are three major

[2] The mode of the Beta distribution for our specification of a and b is given by $\frac{a-1}{a+b-2} = \frac{\alpha(n^*+1)-1}{\alpha(n^*+1)+(1-\alpha)(n^*+1)-2} = \frac{\alpha(n^*+1)-1}{n^*-1}$.

Algorithm 2 Find Re-weighted Quantiles

Input: C_t , $\hat{C}^m_{t|t-k}$, R, r, γ, α.
for $m = 1$ **to** M **do**
    $w^m_R \leftarrow \sum_{t=R-r}^{R-1} -\frac{1}{2}\gamma(C_t - \hat{C}^m_{t|t-k})^2)$ ▷ Find performance weights.
$\mathbf{w_R} \leftarrow (w^1_R, ..., w^M_R)$ ▷ Form vector of performance weights.
for $m = 1$ **to** M **do**
    $\bar{w}^m_R \leftarrow \frac{\exp\{w^m_R - \max(\mathbf{w_R})\}}{\sum_{m=1}^{M} \exp\{w^m_R - \max(\mathbf{w_R})\}}$ ▷ Normalize performance weights.
$a, b \leftarrow \alpha(n^* + 1), (1 - \alpha)(n^* + 1)$ ▷ Specify quantile through Beta -parameters
for $m = 1$ **to** M **do**
    $l_m \leftarrow \sum_{j=1}^{m-1} \bar{w}^j_R$ ▷ Find lower interval endpoint.
    $u_m \leftarrow l_{m+1}$ ▷ Find upper interval endpoint.
    $\widetilde{w}^m_R \leftarrow I_{u_m}(a, b) - I_{l_m}(a, b)$ ▷ Find contribution weight.
$\tilde{q}^{\alpha}_{t|t-k} \leftarrow \sum_{m=1}^{M} \widetilde{w}^m_R C^m_{t|t-k}$ ▷ Form re-weighted quantile.
Output: $\tilde{q}^{\alpha}_{t|t-k}$

differences, however. First, we perform the weighting based on observations in principal component space as opposed to at the level of grid point observations. Second, our approach is lead time agnostic, while the other approaches are lead time specific. And third, our approach involves applying the weights directly to the ensemble members to form quantiles. In contrast, Brajard et al. (2023) is concerned with estimating a posterior density function for the model state given new observation, while Schuhen et al. (2020) performs adjustments to the underlying distribution parameters.

3.6 Aggregated Quantile Estimation by Gaussian Copulas

As we have mentioned, when forecasting PC temperature utilizing NWP data, forecast skill at the 6-hourly level disappears after around 15 days (see also Section 6.3). It is, however, possible to make model skill manifest at longer lead times by employing temporal aggregation methods (Nystrup et al. 2021). An intuition for the application of forecast aggregation can be given from Jensen's inequality applied to a linear combination of random variables. For a convex function $f(\cdot)$, Jensen's inequality states that the function of a linear combination is less than or equal to the linear combination of that function applied to each of the elements in the linear combination. If we then take the expected value on both sides of the inequality we obtain:

$$\mathbb{E}[f(x_1 + x_2)] \leq \mathbb{E}[f(x_1)] + \mathbb{E}[f(x_2)]. \tag{3.33}$$

In our case $f(\cdot)$ is a convex loss function, and from (3.33) we have that the expected loss of an aggregate is less than the sum of the expected loss applied to each observation. Obviously, this holds for the output of baseline models as well as for any models of interest. If, however, the model is skillful then the amount of loss reduction is greater for the model of interest than for the baseline reference model.

Our aggregation method of choice is the Gaussian Copula (GC) method. To motivate its use we first consider a naive aggregation approach. The quantile estimators described in the previous sections have at time t estimated quantile α of the random variable y_t by a partial estimation (i.e. at quantile α) of the inverse cumulative function F_t^{-1}, on this pattern:

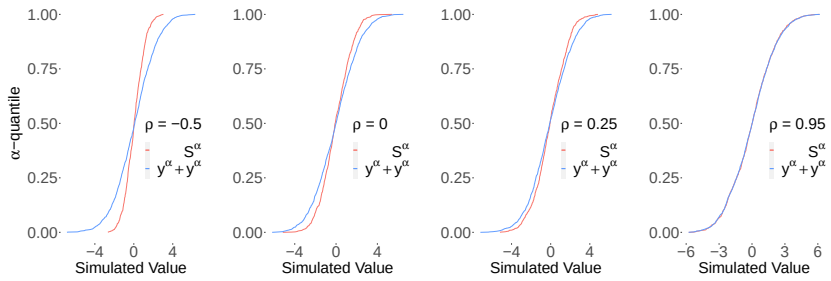
$$\hat{y}_t^\alpha = \hat{F}_t^{-1}(y_t, \alpha). \tag{3.34}$$

However, the elements of the sequence $\{\hat{y}_t^\alpha\}_{t=1}^T$ are not independent of each other, but form a correlated time series. After having obtained these predictions we are interested in estimating the quantile of a summary statistic, e.g. the aggregate $S_{1:T}^\alpha = (\sum_{t=1}^T y_t)^\alpha$, over an interval of time points, $t = \{1, ..., T\}$. Consider, then, the naive post-hoc quantile estimator of a simple linear combination of α-quantiles over the interval of interest utilizing the previously obtained predictions:

$$\hat{S}_{1:T}^\alpha = \hat{y}_1^\alpha + \hat{y}_2^\alpha + ... + \hat{y}_T^\alpha. \tag{3.35}$$

Even if the individual $\hat{y}_t^\alpha$ is a good estimate of the marginal quantiles at each target time, the sum of the quantiles cannot be used directly to estimate the quantile of the summary statistic. This is because it ignores the underlying correlation structure between y_t at different time points. Whereas the predicted quantiles will lie close to α, the realizations of y_t will be spread around according to their distributions. This means that the degree to which they line up is dependent on the correlation structure between the variables. This can readily be shown through simulation. Consider four scenarios where in each we simulate ($m = 1000$) from a bi-variate standard normal distribution with mean 10, with different correlation structures (ρ = -0.5, 0, 0.25 and 0.95). In all scenarios (Figure 3.1), except for the nearly perfectly correlated one, the post-hoc estimator over-estimates the values of high quantiles and under-estimates the values of low quantiles. Notice also that the behaviour at the 0.5-quantile is unproblematic. This is to be expected in symmetric distributions like the standard normal where the mean and median are close.

This means that if we want to estimate quantiles over aggregated intervals we ought to take account of the correlation structure within the predictive distribution between time points. For this purpose we will follow an approach



(a) $\rho = -0.5$. (b) $\rho = 0$. (c) $\rho = 0.25$. (d) $\rho = 0.95$.

Figure 3.1: Plots comparing the sum of the quantiles ($y_1^\alpha + y_2^\alpha$) with the quantiles of the sum $S^\alpha = (y_1 + y_2)^\alpha$ for four values of correlation (ρ) between y_1 and y_2.

centered on utilizing Gaussian Copulas to describe the dependence structure between the variables within the interval of interest (Pinson et al. 2009; Möller et al. 2013).

For this section, for expositional convenience, we break with the above established notation and use capitalized letters to denote random variables. Consider T random variables $X_1, ..., X_T$ each with continuous marginal cumulative distributions $F_t(X_t)$. The joint cumulative distribution of the realisations, $F(x_1, ..., x_T)$, can be decomposed by applying the probability integral transform $U_t = F_t(X_t)$:

$$\begin{aligned} F(x_1, ..., x_T) &= \mathbb{P}(X_1 \leq x_1, ..., X_T \leq x_T) \\ &= \mathbb{P}(F_1(X_1) \leq F_1(x_1), ..., F_T(X_T) \leq F_T(x_T)) \\ &= \mathbb{P}(U_1 \leq F_1(x_1), ..., U_T \leq F_T(x_T)) \\ &= \mathbb{P}(F_1(x_1), ..., F_T(x_T)) \\ &= \xi(F_1(x_1), ..., F_T(x_T)). \end{aligned} \tag{3.36}$$

The decomposition of the joint cumulative distribution yields two components: the marginal cumulative distributions, $F_t(x_t)$, and the copula, $\xi(\cdot)$, which is the joint cumulative distribution over the marginals. It is a multivariate distribution function over marginal distributions that are uniformly distributed: $F_t(x_t) = u_t \in [0, 1]$. By Sklar's theorem the representation (3.36) is available for any joint cumulative distribution $F(\cdot)$ (Salinas et al. 2019). While the copula captures everything about the dependence structure between the component variables $X_1, ..., X_T$, the marginal describes the distribution information particular to each X_t.

In our case we will induce the correlation structure through the use of Gaussian Copulas (GC). Employing GC is convenient as it only requires the marginals, $F_t(\cdot)$, and the correlation matrix of the joint CDF to be fully defined (Möller et al. 2013). Now, assume that we are in possession of the marginal $F_t(X_t)$. We can then obtain the latent factor $Z_t = \Phi^{-1}(F_t(X_t))$ through applying $\Phi^{-1}(\cdot)$, the inverse CDF of the standard normal distribution. Let us then consider $F(\cdot|\Sigma)$, the same joint cumulative distribution as above. But, we now assume that the correlation structure between each X_t is captured through some Σ, a $T \times T$ correlation matrix. A Gaussian copula for the joint distribution can then be written as:

$$\begin{aligned} F(x_1, ..., x_T|\Sigma) &= \xi(F_1(x_1), ..., F_T(x_T)|\Sigma) \\ &= \Phi_T(\Phi^{-1}(F_1(x_1)), ..., \Phi^{-1}(F_T(x_T))|\Sigma) \\ &= \Phi_T(z_1, ..., z_T|\Sigma), \end{aligned} \tag{3.37}$$

where $\Phi_T(\cdot|\Sigma)$ is the CDF of a multivariate normal distribution with mean 0, and correlation matrix Σ. The Gaussian copula is a function from $\mathbb{R}^T$ to the unit hypercube $[0, 1]^T$, that preserves the uniform marginals (Tedesco et al. 2023). By applying a Gaussian copula structure we are re-framing the assumption made above on the correlation structure between each X_t. We are now assuming that the correlation can be captured as a parameter in the distribution of the latent factor variable $Z = (Z_1, ..., Z_T) \sim MVN(0, \Sigma)$. This is the simplest assumption on the dependence structure for latent variables (Pinson et al. 2009), and is chosen for convenience.

Algorithm 3 Estimate Correlation Matrix

Input: $x_{l\in\tau}$, $c_{l\in\tau}$, τ
$\hat{F}^{-1}_{l\in\tau} \leftarrow Q(x_{t\in\tau})$ ▷ Make 6-hour level prediction CDF.
for $l = 1$ **to** L **do**
  $u_l \leftarrow \hat{F}_l(c_l)$ ▷ Find in-sample observed quantile.
  $z_l \leftarrow \Phi^{-1}(u_l)$ ▷ Transform to standard normal.
$\boldsymbol{Z}_{m\times|\tau|} \leftarrow z_{l\in\tau}$ ▷ Form matrix of latent Gauss. factors.
$\hat{\Sigma} \leftarrow \frac{1}{n-1}\sum_{t=1}^{n} \frac{(\boldsymbol{z}_t-\bar{\boldsymbol{z}})(\boldsymbol{z}_t-\bar{\boldsymbol{z}})^T}{\boldsymbol{s}\boldsymbol{s}^T}$ ▷ Estimate correlation matrix.
Output: $\hat{\Sigma}$

The advantage of this structure is that it provides a manner of estimating the correlation matrix in a way that does not rely directly on the properties of the individual marginal distributions. Once the correlation matrix is estimated we can sample from it and obtain (through a back-transformation) any desired aggregate which we in turn can find a quantile estimation of.

Now, our concern is to estimate the α-quantile of an aggregate over an interval of correlated variables. For the random variable C_t of PC temperature at target time t we write this as $S^{\alpha}_{1:T} = (\sum_{t=1}^{T} C_t)^{\alpha}$. Required for this is finding the correlation structure across some interval of the predictive distribution which we want to aggregate over. The estimation of the correlation must be based on more than one interval, however. It could be based on any collection of d intervals of length T. Our choice is to look at the correlation over lead time intervals across forecast issuance months. This is done mainly because it simplifies the estimation structure in a way that aligns with the forecasting set-up (which is based around monthly forecast issuance). To make clear that each marginal (in our case) is lead time specific, we use the notation $F_l(\cdot)$ to denote the marginal for any target time t with lead time l relative to some forecast month m, and similarly $F_{l\in\tau}(\cdot)$ to indicate a set of marginals covering lead time interval τ.

Additionally, each marginal must be estimated from data. For this purpose we use the QR-model (Section 3.4), which can provide us with a predictive distributions set on the form of (3.18) over a range of probabilities: $\hat{F}_l(\cdot) = \{\hat{q}_l^{\alpha_i} | 0 < \alpha_1 < \ldots < \alpha_m \leq 1\}$. The effect of the granularity of this set will be investigated in Section 6.5. We can then formulate the Gaussian copula for our case as:

$$F(c_1, ..., c_L|\Sigma) = \Phi_T(\Phi^{-1}(\hat{F}_1(c_1)), ..., \Phi^{-1}(\hat{F}_T(c_L))|\Sigma). \tag{3.38}$$

The processes of estimating the correlation matrix for a given forecast month, m, is given in algorithm 3. The first step consists of estimating the inverse marginal distributions $\hat{F}^{-1}_{l\in\tau}(\cdot)$ based on some training data $x_{l\in\tau}$. Next, for each lead time l, we obtain the observed quantile location, $u_l = \hat{F}_l(c_l)$, by checking where within the predictive set the observed PCs are located. From this we form a matrix of latent Gaussian factors which we can easily estimate the correlation matrix from.

After having estimated the correlation matrix we go on to obtain estimates of the quantile aggregates of a predictive distribution (Algorithm 4). In contrast

Algorithm 4 Find Aggregated Quantile Estimate

Input: $\hat{\Sigma}$.
$z^{(1)}, ..., z^{(M)} \leftarrow \text{MVN}(0, \hat{\Sigma})$ ▷ Sample M obs. from MVN distribution.
for $i = 1$ **to** M **do**
  $u^{(i)} \leftarrow \Phi(z^{(i)})$ ▷ Back-transform to find quantile locations.
  $y^{(i)} \leftarrow \hat{F}_{l^*}^{-1}(u^{(i)})$ ▷ Find simulated realization
  $S_{1:L}^{(i)} \leftarrow y_1^{(i)} + ... + y_L^{(i)}$ ▷ Form aggregate over lead time interval.
$\hat{S}_{1:L}^{\alpha} \leftarrow W^{\alpha}(S_{1:L}^{(1)}, ..., S_{1:L}^{(M)})$ ▷ Find α-quantile of aggregate.
Output: $\hat{s}^{\alpha}$

to Algorithm 3 which applies the predictive distribution set backward on the training set, we are now only interested in applying it forwards to new forecasted values.

We first simulate M samples $z^{(i)} = (z_1^{(i)}, ...z_L^{(i)})$ from the multivariate normal distribution utilizing $\hat{\Sigma}$, the estimated correlation matrix found in algorithm 3. Then, for each sample, we back-transform the simulated values using $\Phi(\cdot)$ to obtain $u^{(i)} \in [0, 1]^L$, a vector of simulated quantile locations. These in turn inform us where in the predictive inverse CDF the simulated realizations, $y^{(i)}$, can be found. Now, the simulated $u^{(i)}$-vectors are only specified with regard to a lead time interval, but not to any target time. The predictive distribution, however, is target time specific. We denote this as $\hat{F}_{l^*}^{-1}(\cdot)$, letting the star indicate that we are forecasting a new observation at lead time l. Each $y^{(i)}$ is a simulated realization of the predictive distribution over a lead time interval based on: i) the marginal distributions estimated by our model; and ii) the estimated correlation structure between lead times. From these we can form a set of M aggregations. And at last, we can estimate the quantile of interest based on the set of aggregations using the WQE quantile estimator (6.2):

$$\hat{S}_{1:L}^{\alpha} = W^{\alpha}(S_{1:L}^{(1)}, ..., S_{1:L}^{(M)}). \tag{3.39}$$

3.7 Evaluation Procedures

In this section we will describe our evaluation procedures. We will first describe the Prequential Cross-Validation (PCV) procedure that we will employ in all our model evaluation scenarios. We will then discuss the inclusion of a hold-out set and the model selection process.

Cross-Validation (CV) is the most common method employed for model assessment and selection within the field of machine learning (Hastie et al. 2009). The standard K-fold CV involves first splitting the data into K folds. Then for each k one sets aside the kth fold for testing while one fits the model on all other folds. The performance of the model can then be found by averaging results across all K test sets.

In the case of time series forecasting it has been noted that because observations are not independent, standard CV is theoretically inadequate (Cerqueira, Torgo and Soares 2023). This notwithstanding, it has been showed

to perform well in specific settings, i.e. for purely auto-regressive models without correlated errors (Bergmeir et al. 2018). In time-series forecasting, several types of CV have been employed in addition to standard CV. These include variants of the following procedures: 'hold-out' (one split); 'blocked CV'; LOOCV and PCV (Cerqueira, Torgo and Soares 2023; De Felice et al. 2015).

The approach we have followed is the Prequential Cross-Validation (PCV) method. PCV centers around the idea of forecast intervals, in our case forecast months. This method also goes by the name 'sequential' CV, or, highlighting its prominence, simply by 'time-series CV'. Suppose all available data can be ordered by month. For a given forecast month, m, in a sequence of forecast months, we form the training data based on the preceding forecast months up to but not including m. Employing the training data, we make a forecast which we test on month m. We then incorporate the test data from month m into the training data we use to fit a model for forecast month $m + 1$. In this way previous test sets are continuously incorporated into the training data. The results are reported as an average over all test sets. There are several versions of prequential CV, distinguished from each other by the length of the training window and the gap between the last training observations and the first test observation (Cerqueira, Torgo and Soares 2023). In Section 5.5 we will look at the sensitivity of the results based on using different training window sizes.

In contrast to the standard CV, the splits in PCV are not random, but follow a monthly order. In addition, a subset of the earlier months will never be tested on, while the last months might only be part of the training set for a limited set of months. The earlier months may also have an out-sized effect on the fitting process since they (depending on the application) might be part of all or many of the training sets.

The PCV method has several advantages. First, like other CV methods, it keeps training and test sets separate. More importantly for our case is that the sequential CV mimics the way a model would be deployed in a real life setting, where we at each month train on all relevant data up to forecast issuance. The results would then serve as a counterfactual estimation of how well a model would have performed if we had deployed it throughout a sequence of forecast months. This is of high practical utility for stakeholders as it provides an intuitive way of understanding results. An additional practical advantage is that the NWP temperature forecasts also follow a monthly sequence, which makes them fit in seamlessly in our evaluation structure.

In addition to employing cross-validation for the purpose of model assessment, results are sometimes reported for a hold-out set. Within the literature there is a noticeable ambiguity with regard to this. On the one hand, you have the standard machine learning approach where the data is divided in three parts: training, validation, and test sets. The standard CV approach would then mix the first two sets, and then evaluate final modal performance on the last separate test, or hold-out, set (Hastie et al. 2009). In the energy demand literature, the paper Bala et al. (2022) is an example of this. They report results for a hold-out period after having built and tested their model using regular CV.

The literature more specific to time-series cross-validation, on the other hand, in general sees the hold-out set as an alternative to CV, indeed as a specific form (CV with one split). It is unclear whether they intend for another hold-out set to be used for final model assessment. Within demand forecasting this approach is followed by De Felice et al. (2015), who only report results for

the leave-year-out-CV and not for any hold-out.

The attractiveness of the hold-out-set as a final evaluation in the standard machine learning case lies in it being an independent assessment of model performance based on randomly drawn observations. In the case of sequential models, however, the observations in the hold-out set would not be random, but possess a specific time signature. In the case of both demand and temperature forecasting, performance by month is subject to substantial fluctuations. Suppose we had a hold-out set, e.g. composed of observations over a year (which seams appropriate in the demand forecasting task which has 10 years of data). Such a set would not give information about the performance of, e.g., randomly drawn January observations, it would only give information about a very specific January that might exhibit anomalous tendencies that could skew the results. For an example of month-by-month variation in performance (see Section 5.5). A more reliable assessment of model performance would be an average over all forecasting periods we have tested on, i.e. the PCV results.

Furthermore, the time signature might be what we are the most interested in learning about in relation to model performance. In our case we want good estimates for model performance for specific months and specific lead times. And averaging over lead times across a substantial number of test sets provide more stable results than relying on a single hold-out set. In short, the hold-out set does not accurately reflect the properties we want for the model evaluation, and we will therefore, like (De Felice et al. 2015), only report results obtained over the PCV test periods.

An effect of this is that the model selection and the final model evaluation is performed on the same test data. These are separate problems (Hastie et al. 2009). Within the time-series domain, attention has been given to the studying how well different CV methods estimate predictive performance (Bergmeir et al. 2018; Cerqueira, Torgo and Mozetič 2020). But until recently little work has been done on studying how well different CV-procedures does in selecting model predictors (Cerqueira, Torgo and Soares 2023). Our approach to model selection has centered around a two-step selection process using the PCV procedure for evaluating results. We have first tested different parametrizations of a relatively small number of time covariates, which we have had good reason to include in our model. We have then combined these with temperature information either in the form of mean grid temperature or in the form of principal components. This approach was motivated by two aims. First, to obtain good predictive ability, and second to gain insight into how the addition of specific predictors changed forecasting performance. We have especially been interested in obtaining a good parametrization of time covariates in order to properly gauge the effect of including temperature information on predictive performance. The approach could be made more robust with a more involved model selection procedure.

Predictive performance is, however, not the only model selection criteria we will employ. In building a model we also consider it important for the model to be reliable, interpretable, parsimonious, and stable. Reliability is especially important for stakeholders which depend on model output. If the model gets too complex to be fitted every month, then it is of no practical use. An important indicator of a good model is also that it exhibits stable results. Specifically that it will perform relatively robustly with regard to different data inputs and small tweaks of the data. Interpretability is important for getting stakeholders to trust the model. If transparent reasons can be offered for why

a specific forecast is given, then it might increase the reliance on the model in practical settings. An interpretable model often also simplifies the model structure, making it more parsimonious. In our case we have large amounts of data, and many moving parts. In such a setting, any simplification of the model structure that does not come at a huge cost in accuracy might be beneficial for both computational and data-handling issues.

3.8 Evaluation Metrics

The main evaluation metrics that we will employ in this thesis are the following: root mean squared error (RMSE), skill score, pinball loss, and an approximation of the continuous ranked probability score (CRPS). Their formulas are given below.

Root Mean Squared Error (RMSE)

With regard to the structural demand task we employ the root mean squared error as our main evaluation metric. For forecasted value $\hat{y}_t$, and observation y_t, the RMSE is given as:

$$RMSE(\hat{y}_t, y_t) = \sqrt{\frac{1}{n}\sum_{t}^{n}(\hat{y}_t - y_t)^2}. \tag{3.40}$$

Pinball Loss

In the probabilistic forecast setting we are estimating quantiles. This requires a different accuracy measure than estimating expected values. A standard error metric for this task is the pinball loss, a proper scoring rule, which is a measure of how good the quantile estimate is. For a quantile of interest $\alpha \in [0, 1]$, predicted value, $\hat{y}_t$, and observed value y_t, the pinball loss is defined as:

$$\rho_\alpha(\hat{y}_t, y_t) = \begin{cases} (y_t - \hat{y}_t)\alpha, & \text{if } \hat{y}_t < y_t, \\ 0, & \text{if } \hat{y}_t = y_t, \\ (\hat{y}_t - y_t)(1-\alpha), & \text{if } \hat{y} > y. \end{cases} \tag{3.41}$$

This loss function derives its name from the way the loss curve points in different directions, which resembles the arms of a pinball machine. What makes pinball loss suitable for quantile prediction is that, except for the case $q = 0.5$, where it reduces to the absolute error, it will penalize under- and over-prediction differently. Like the absolute error, the worse the prediction is, the bigger the penalty. Additionally for values of q strictly over 0.5 the penalty for predicting over the observed value will be higher, and vice versa for $q < 0.5$.

A well-behaved forecast at high quantiles ought to have a sizeable proportion of its predictions above the observed value thus receiving low penalties. Correspondingly for low quantiles a higher proportion will be lower than the observed value, again receiving low penalties. Closer to the median a well-behaved forecast should be more evenly spread around the observed value, thus receiving a penalty multiplier closer to 0.5 across the board.

Skill Score

For model comparison we use the skill score, a relative measure, frequently used to compare model performance with respect to a competing baseline or reference model. The skill score, SS, for models evaluated by RMSE is given by:

$$\mathrm{SS} = 1 - \left(\frac{\mathrm{RMSE}_{\mathrm{forecast}}}{\mathrm{RMSE}_{\mathrm{reference}}} \right)^2, \tag{3.42}$$

The skill score for models evaluated by pinball loss is given by:

$$\mathrm{SS} = 1 - \frac{\rho^{\alpha}_{\mathrm{forecast}}}{\rho^{\alpha}_{\mathrm{reference}}}. \tag{3.43}$$

Any improvement over the baseline or reference model results in a skill score above 0. The limit case where MSE of the model of interest approaches 0 and the reference model is constant, the skill score is 1. The baseline models we will use are detailed for each specific problem in Sections 5.1 and6.1.

CRPS

To evaluate the performance of a predictive distribution set across quantiles we utilize an approximation of the continuous ranked probability score (CRPS), $\int_{0,1} \rho_{\alpha} d\alpha$, in the following manner:

$$\mathrm{CRPS} \approx \frac{1}{9} \sum_{\alpha=0.1}^{0.9} \rho_{\alpha}, \tag{3.44}$$

where $\alpha \in \{0.1, 0.2, ..., 0.9\}$. See also Gneiting et al. (2007).

CHAPTER 4

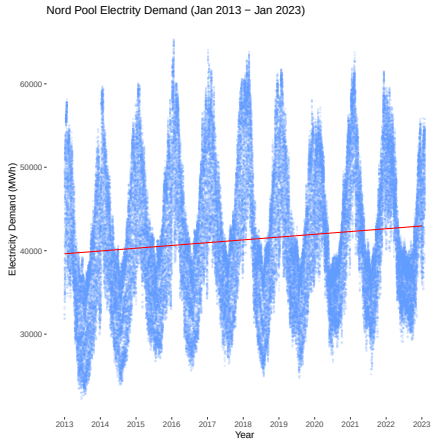
Data and Programming

The main data utilized in this thesis comes from three sources: 1) the Nord Pool electricity demand volume data for the Nordic region;[1] 2) the ERA5 temperature data provided in grid form over Northern Europe; and 3) the NWP weather forecast data over the same grid. In this chapter we provide a description of the data, presenting its key characteristics as well as detailing pre-processing steps. In Section 4.1 we describe the response variable for our main model: electricity demand. This includes a presentation of its cyclical trends. Then, in Section 4.2, we give an overview of the ERA5 temperature data. We will first describe properties of its original grid form, before presenting key characteristics of its principal component form. We discuss the selection of PCs, and put special emphasis on the two first, which will form a key aspect of our modeling. In Section 4.3 we describe the structure and pre-processing steps involved in gathering the NWP forecasts. Finally, we provide a brief overview of the data structure as a whole.

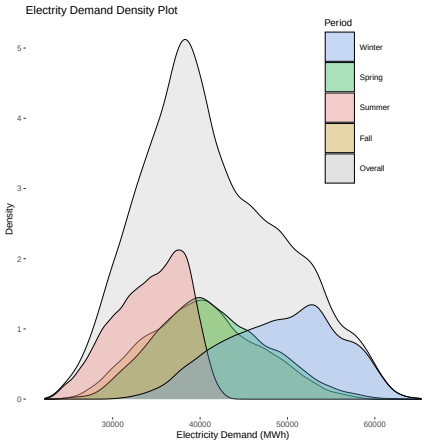
4.1 Nord Pool Electricity Demand Data

Like we saw in Section 2.1, other electricity demand forecasting studies have utilized 'electricity load' (De Felice et al. 2015; Mirasgedis et al. 2006) , or 'electricity consumption' (Bala et al. 2022) as their response variable working as a proxy for demand. In our study we have used the slightly different 'electricity demand volume' (measured in MWh) as our response variable. The data is gathered from Nord Pool, a private joint-stock market exchange operating in Northern Europe. Their day-ahead market is a closed auction with over 300 buyers and sellers placing bids on energy delivery for the next day (Nord Pool 2023). The buyers are typically representatives of electricity providers purchasing on behalf of customer need. The demand volume is the sum total of successful bids for a set region, in our case the Nordic region. Nord Pool also refers to this as 'turnover at system price'. As such, it is an indirect estimate of customer energy demand. To what extent the actual consumed energy differs from demand has not been investigated, but we do not expect large discrepancies between these.

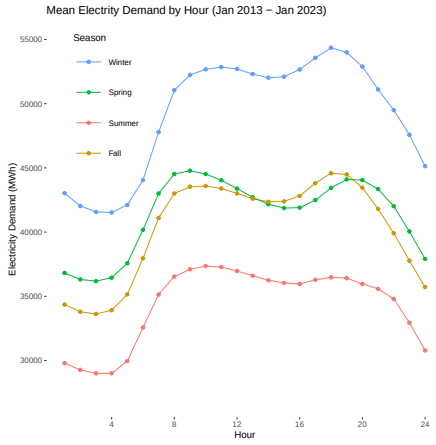
[1] The Nord Pool electricity demand data encompasses 7 countries: Norway, Sweden, Denmark, Finland, Estonia, Latvia and Lithuania, for simplicity referred to as the Nordic region.



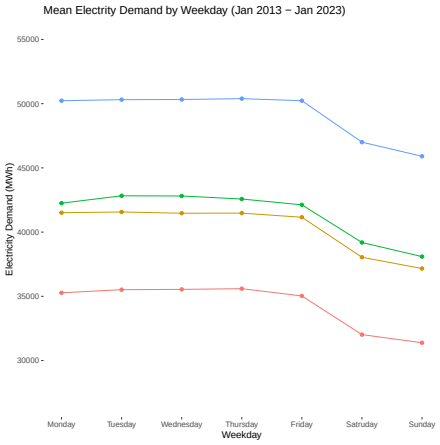
(a) Electricity Demand with Trend.



(b) Demand Density by Season.



(c) Mean Demand by Hour.



(d) Mean Demand by Weekday.

Figure 4.1: Plots of Nord Pool energy demand (Jan 2013 - Jan 2023) showcasing the annual, weekly and daily demand cycles. The density plot as well as the short cycles are broken down by season. The red line in 4.1a indicates the hourly trend with $\beta_1 = 0.03$.

The demand data have only been subjected to very modest pre-processing. The raw inputs contained missing or doubled observations connected to daylight savings time changes. This was fixed by shifting observations during summer back one time step, in effect cancelling summer time adjustment. This creates a complete time series without missing data, which as a bonus lines up with the temperature and forecast information.

The main descriptive statistics of the demand data are provided in Table 4.1. Over 121 months (Jan-2013–Jan-2023) we have 88392 hourly spaced observations ranging between 22211.6 and 65311.1 MWh. In Figure 4.1b we see the distribution of all demand observations contrasted with the same distribution decomposed by season. The whole distribution is uni-modal, light tailed, and right-skewed. There are considerable seasonal differences. The summer and winter distributions only have a small overlap, while the spring

Data	Min	1st Q	Median	Mean	3rd Q	Max	Unit
Demand	22211	35586	40167	41353	46982	65311	MWh
ERA5	225	274	278	278	282	308	K

Table 4.1: Summary statistics for electricity demand (Jan 2013 – Jan 2023) for 88392 hourly spaced observation in MWh, and ERA5 temperature (Jan 1979 - Jan 2023) in K for 178 million grid points over 386448 time points.

and fall distributions are almost identical.

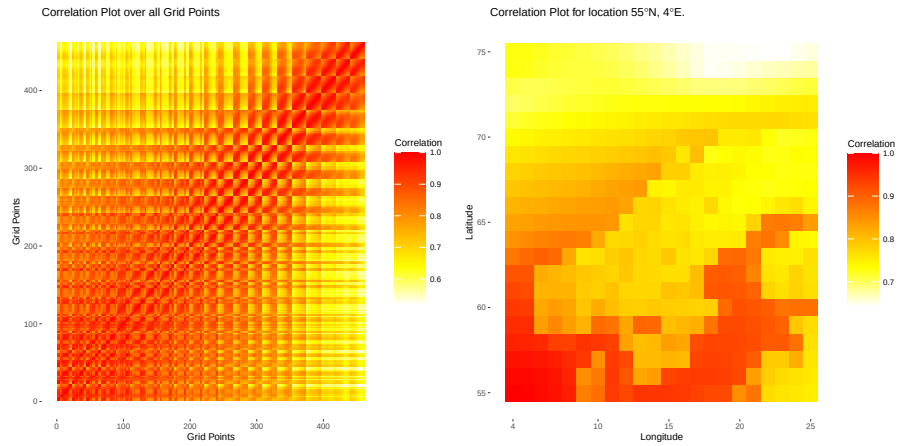
The demand observations exhibit a slight upward trend. A linear regression ($y_t = \beta_0 + \beta_1 t$) over all demand observations, y_t, by time t yields $\hat{\beta}_1 = 0.03$, indicating a slight increase in electricity demand per time step within the time period (see red line in Figure 4.1a). Using yearly mean values we also run the model: $\bar{y}_{yr} = \beta_0 + \beta_1 yr$. The coefficient estimate indicates a yearly increase in the mean of 400.4 MWh. In both estimates we exclude the January 2023 observations as these only reflect a part of the annual cycle which would exaggerate the upward trend.

In addition to the trend, the demand observations display three clear cyclic structures: annual, weekly and daily. The annual cycle (Figure 4.1a) shows demand rising each winter and sinking each summer. As has been pointed out, this variation, in the Norwegian case, is largely driven by heating installations reflecting outside temperature levels (Hofmann et al. 2019). Figure 4.1c demonstrates the hourly variation in demand. Demand is lower at night, rises in the morning, and keeps roughly at the same high level between 8:00-20:00, before again falling in the evening. In contrast to the annual cycle, the daily cycle seems to follow a general societal activity pattern. If demand only followed temperature (through heat installations) then demand would increase at night when it is colder. We also observe a weekly cycle (Figure 4.1d) where demand is fairly constant during the work week while falling considerably during the weekend. It's plausible this pattern reflects the weekly cycle of economic activity. This suggests that we ought to include not only temperature information in our structural demand model, but also a trend term, as well as terms capturing the short cycles not directly related to temperature.

Both the daily and weekly cycles seem fairly similar across seasons, but there are nuances. Notice for example the evening bump in demand during winter which seems absent during summer. These differences become more pronounced for more granular subdivisions of the annual cycle, e.g. using weeks or months instead of seasons. In Section 5.2 we look closer at interaction combinations between short cycle (daily, weekly) terms and an annual cycle terms (week, month, season) within the context of the demand forecast model.

4.2 ERA5 Temperature Data

The ERA5 temperature data we utilize are produced by the Copernicus Climate Change Service (C3S) which is part of the European Centre for Medium-Range Weather Forecasts (ECMWF). It has also been through post-processing to remove bias.



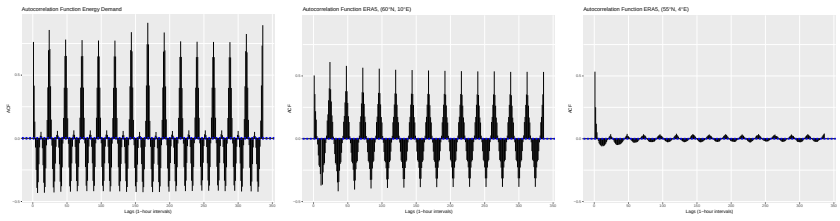
(a) Temperature correlation. (b) Correlation relative to 55°N, 4°E.

Figure 4.2: Plots of correlation between temperature grid points. Figure 4.2a shows the correlation between all grid points, while 4.2b shows the correlation relative to one location (55°N, 4°E). Red indicates higher correlation.

The temperature observations, measured in Kelvin (K), span from 1.1.1979 to 31.1.2023, for a total of 386448 hourly spaced time points. At each time point they cover 462 grid locations across a 21×22 sized grid over Scandinavia for latitudes 55° to 75° and longitudes 4° to 25°. Together the observations constitute 462 complete time series (after interpolating values for 2 missing time points). Summary statistics are provided in Table 4.1.

The ERA5 temperature field exhibits a high degree of correlation, both spatially and temporally. Across all time points we observe a high positive correlation ($\rho > 0.5$) between any two grid points as can be seen in figure 4.2a. Correlation increases with proximity and is lowest when grid points are located further apart. This can be seen clearly in Figure 4.2a where the correlation between location 55°N, 4°E (bottom-left) over the North Sea is plotted relative to all other grid points. This plot is also a clear example of locations at sea often having a higher correlation with each other than with grid points at land. This correlation structure is so evident, in this case, that it clearly demarcates the Scandinavian coastline.

The temporal correlation is very location dependent. This can be exemplified



(a) ACF Demand. (b) ACF 60°N, 10°E. (c) ACF 54°N, 4°E.

Figure 4.3: ACF plots (after differencing) of time series over a 2-week time span.

through plotting the auto-correlation structure (after differencing) for two grid points over a time span of two weeks. We choose grid locations 55°N, 4°E (over the North Sea), and 60°N, 10°E (roughly corresponding to Oslo) as they exemplify fairly typical sea and land locations, respectively. In both plots 4.3a and 4.3b we see a clear daily cycle. Temperature at any given time is correlated with observations at the same time of day even several days apart, and negatively correlated with observations at the opposite time of day. The correlation on land, however, is much higher than at sea. Compared to the electricity demand data 4.3a, the correlation structures both at sea and on land are weaker.

Principal Components of ERA5 Temperature Data

As we have seen, demand is expressed as a single continuous variable representing electricity demand in the Nordic region. The highly correlated temperature grid makes it difficult to ascertain what grid points should have the most effect on demand. By employing PCA on the temperature grid, we do not have to rely on any individual grid point. Instead, we create a small set of new variables that describe most of the variation across the temperature field, which is used in the demand models. Two widely used ways of selecting principal components are the 'cumulative percentage of variance' method and the 'scree graph' method. Both are ad-hoc rules of thumb ultimately relying on subjective judgement (Jolliffe 2002). The 'cumulative' method sets a threshold (usually in the 70-90% range) for the cumulative percent of the variance one wants the selected PCs to account for. The last component, m, to be selected is the one that makes the cumulative percent of the m first PCs exceed that threshold. Jolliffe (2002) advises that if the first two components are especially dominant, a cut-off higher than 90% may be necessary. The scree graph method relies on finding the elbow of the plot of the variance of each PC. The last PC to be selected is the

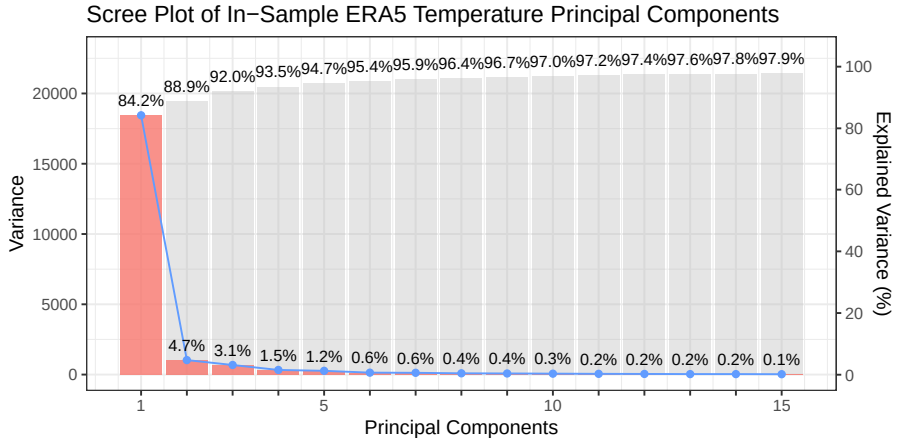


Figure 4.4: Scree plot of in-sample ERA5 temperature principal components, with variance explained superimposed for the 15 first (out of 462) PCs. The first PC is the most prominent explaining 84.2 % of the variance in the in-sample temperature grid data.

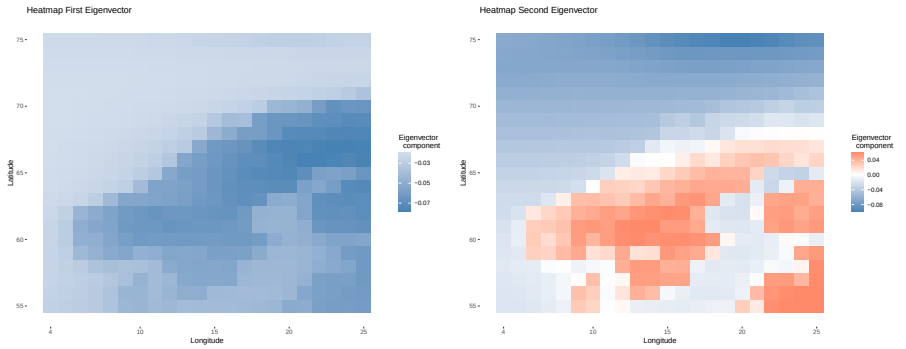
one with a 'steep' slope to the left and a 'not steep' slope to the right. An alternative heuristic is to let the above-mentioned PC be the first not to be selected (Jolliffe 2002).

Forming PCs based on all in-sample observations we see (in Figure 4.4) that the first principal component is particularly dominant, representing 84.2% of the variance of the temperature grid data over the whole in-sample period. The elbow of the scree plot is clearly located at the second PC. Based on this we should only select the first two PCs (alternatively just the first). But the scree line does not completely flatten out. With regard to the 'cumulative' method we have no reference threshold available, but given the dominant position of the first component it seems advisable to set it higher than 90%. This would mean to include at least the three first PCs. Since our primary objective is predictive performance it seems apt to forego any conclusion for the moment and instead subject the issue to cross-validation testing (Section 5.3). For this purpose, we therefore set out to test the 10 first PCs. The choice of 10 allows us to explore principal components that together account for almost all the variation (97.1%) in the temperature data. At the same time, it represents an extremely substantive dimensionality reduction, from 462 to 10.

The most important variance structures marked out by the first two PCs are the overall temperature difference and the difference between land and sea temperature. To exemplify these structures, we plot both the eigenvectors of the first two PCs back on the grid coordinates of the original temperature data. In Figure 4.5a we observe the first eigenvector. Even without the superimposition of a map it is possible to make out the Scandinavian peninsula, as we observe a gradual transition in the temperature structure between land and sea. From the heatmap it is clear that all grid points have the same sign, indicating that the eigenvector primarily points out the overall temperature in the Nordic region. When projecting the first eigenvector onto the temperature observations to form PCs, we project the average temperature per grid point. Large PCs indicate colder temperatures, while lower PCs indicate higher temperatures. The eigenvector corresponding to the second PC (Figure 4.5b) has a slightly larger range of values. The land-sea demarcation is prominent, and is marked by different signs. When projecting the second eigenvector, we are projecting the land-sea temperature differential back on the observed temperature data. The land-sea temperature differential might be an indicator of sea breeze (Steele et al. 2014).

In Figure 4.6 we see the monthly distribution of the two first principal components of ERA5 temperature. The first principal component has a monthly variation that follows the negative of temperature (see Section 3.1). In winter, when temperatures are low, the principal component value is high. It then steadily decreases during spring and summer (as temperature increases) before it continually rises from August to December. The monthly pattern for PC2 is not as clear, and the distributions have considerably less variation between them. They can roughly be grouped in two. In the first half of the year the distributions tilt slightly positive, whereas in the latter half the distributions are centered around 0.

To illustrate the connection between the first PC and electricity demand, we plot a spline fit over all in-sample observations (Figure 4.7). The relation is clearly substantial and non-linear. The general trend is that when the principal component value increases (corresponding to temperatures getting colder), the



(a) Heatmap of first eigenvector. (b) Heatmap of second eigenvector.

Figure 4.5: Heatmap of first and second eigenvector projected back on the dimensions of the original temperature grid.

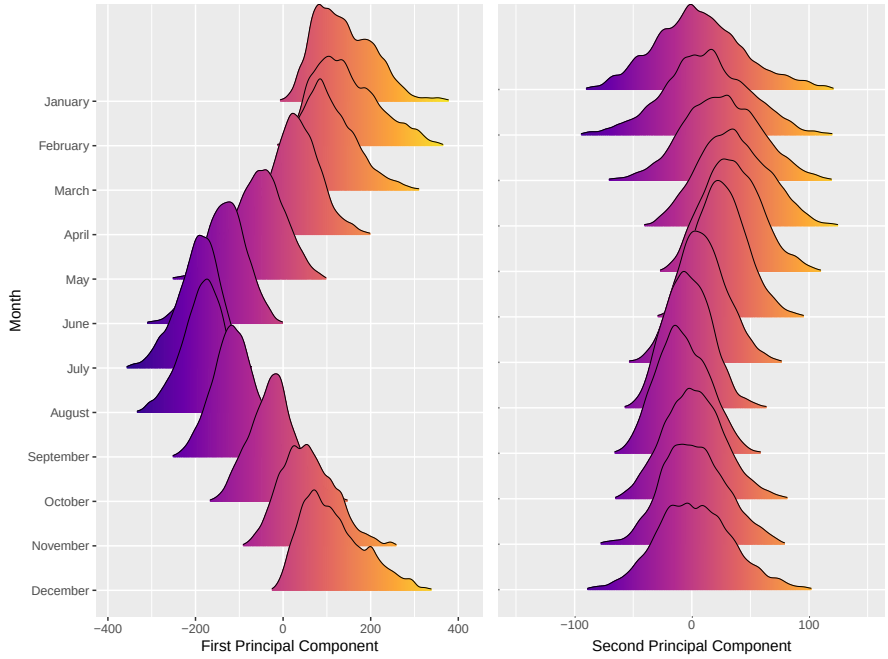


Figure 4.6: Monthly distributions of first and second principal component over all years.

demand increases almost linearly. But this trend does not hold for the lowest values of the principal component (corresponding to high temperatures). After a certain point, an increase in temperature does not translate into a decrease in demand. The spline fit seems particularly suitable as it encapsulates the shift in the relation occurring at around -100. But notice also that at any given value there is a great deal of spread around the fitted line.

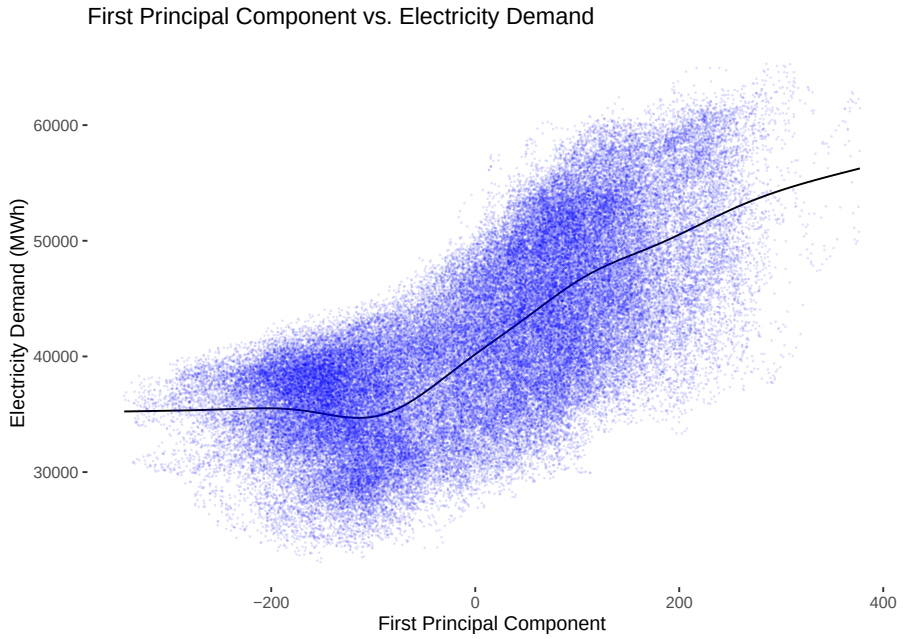


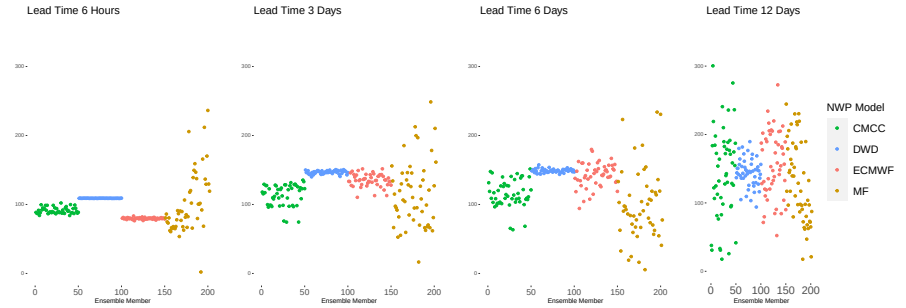
Figure 4.7: Plot of first principal component vs electricity demand. The black line shows the spline fit.

4.3 NWP Temperature Forecasts

In addition to the ERA5 temperature data (1979-2023), we also employ temperature forecasts from four different seasonal NWP (Numerical Weather Prediction) ensemble models: Météo-France, CMCC, DWD, and ECMWF. They cover the period from January 1993 to January 2023. As seasonal forecasts they are medium range coupled models integrating information both from the ocean and from the atmosphere. NWPs increase long-term predictability by latching on to information from large scale weather system phenomena that are predictable on longer time scales, such as El Niño. Seasonal forecasts incorporate larger and more complex interactions than sub-seasonal forecasts, which utilize only atmospheric data (Vitart et al. 2019). After forecast issuance the NWP outputs need to be post-processed to remove bias and variance issues. A detailed overview of different post-processing methods is available in Hemri et al.(2020).

At each forecast issuance the NWP forecasts number between 50 and 202 ensemble members for lead times $k = \{1, \ldots, 500\}$, where each step is a 6-hour interval. The notation specific to the NWP forecasts is detailed in Section 3.4. We are primarily interested in the forecasts in the principal component form. To obtain the principal components we employ Algorithm 1, detailed in Section 3.1. We use eigenvectors obtained from a decomposition of historical ERA5 from the period 1979-1992. This is done for convenience so we don't have to obtain the eigenvector at every forecast month.

We restrict ourselves to highlighting two central aspects of the NWP ensemble members at the principal component level. The first is that the average PC



(a) 6 hours. (b) 3 days. (c) 6 days. (d) 12 days.

Figure 4.8: Plots showing the development of the first PC (shown along the y-axis) for each NWP ensemble member over 4 lead times (6 hours, 3, 6, and 12 days) for an example month (Jan 1993).

value of the NWP ensemble seem to revert to a very general pattern after a while. At the beginning of a given forecast period the average shows no specific structure, but at lead times over 2 months there emerges a daily cycle in the average. This should indicate that the forecasts at PC level seem to rely more on the historic temperature cycles and less on other elements as lead time increases.

The second aspect concerns that for each NWP model there is at the beginning of every forecast month very little variation between members. In Figure 4.8 we see the PC values for 202 members from all four NWP models. At short lead times some models hardly have any variation between members. As time passes, however, all four ensemble models show a large spread in the PC values of the temperature forecasts.

In Figure 4.9 we provide an overview of the relationship between the data structures. The outline shows the process involved in forming the demand forecast which incorporates NWP PC temperature forecasts. It relates, first, how we use the ERA5 data to form different sets of principal components (see Section 3.1). Based on these we also form PCs for each NWP ensemble member. Over the set of NWP PCs we find specific quantiles. These are used to form forecasts of PC temperature, which we in turn utilise as inputs in the demand forecast. The chart also shows the case where we are re-weighing the NWP ensemble member contributions (Section 3.5). This is done based on the NWP PCs of each ensemble member combined with recent observations of PC temperature (indicated by the dotted line). Lastly, the figure shows how the incorporation of additional data structures could be done, here exemplified by stratospheric wind data (see also Chapter 8.

4.4 Programming Language, File System and Libraries

R

The programming for this thesis was done using the R version 4.2.1 and RStudio version 2022.07.1 as the main coding interface locally. The main packages utilised have been **data.table** for swift tabular data operations, **mgcv** for GAM-models,

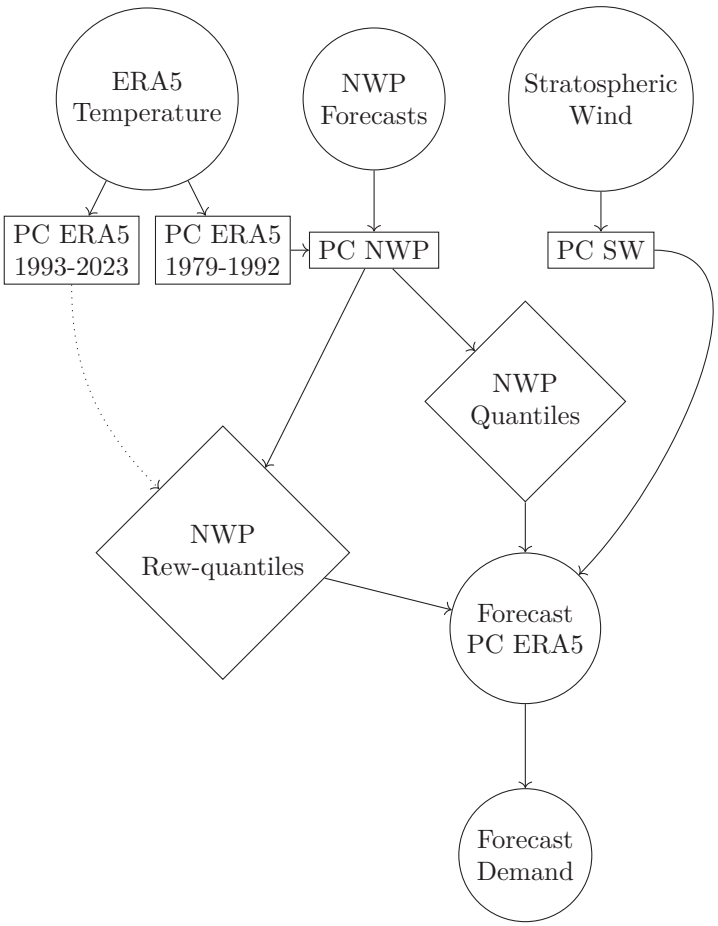


Figure 4.9: Flowchart of data pipeline.

qreg for quantile regression, **parallel** for multi-core parallelisation, and **ncdf4** for reading data. The plotting has been done using base R, and **ggplot2**. The density ridgeline plots were made using the **ggridges** package.

Computation

Through the math faculty at UiO I was granted access to high-performance computing terminals 'abacus-as' and 'nam-shub'. This enabled fast computation through utilization of 'embarrassingly parallelized' code through the function 'mclapply' in R. As the name suggests, this is a basic form of optimization which replaces iteratively performing tasks one by one in a for-loop by distributing them to different cores. A requirement is that these tasks can be done independently (Peng 2022). The R version used in this environment was R 4.2.0-foss-2021b.

Code Documentation / GIT

The code base is available on Github through the R package DemandForecast at https://github.com/eirikhsj/DemandForecast. The magnitude of data makes

it inconvenient to store on Github, but can be made accessible on the UiO data science server upon request. For the models to run, several rounds of data pre-processing have to be done. The package is made with the aim of establishing core functionality, it is not optimized for speed.

PART II

Part 2: Results

CHAPTER 5

Problem 1: Structural Demand Model

In Section 3.3 we introduced the structural GAM-PC electricity demand model and described the motivation behind utilizing it. In this chapter we direct our attention toward three questions: 1) Assuming we have access to near-future temperature observations, what model parametrization of the GAM-PC model gives the most accurate forecast of near-future energy demand? 2) How much does the inclusion of temperature contribute to increase predictive performance? 3) How well does this model compare to alternative implementations? The overarching goal in this chapter, then, is to find and present the best GAM-PC structural seasonal energy demand model and assess its merits.

For model evaluation purposes we use the PCV evaluation procedure described in Section 3.7. Each model is trained on up to 5 years of training data before each forecast issuance date and tested on 85 forecast periods issued on the first of every month, from January 2016 to January 2023. The first issuance months will have 3 years of data, then building to 5 years as we keep rolling on. The effect of training period length is discussed in Section 5.5. All models have the same outcome variable: hourly observed energy demand. The evaluation metric is RMSE over each predicted hour between 15 and 45 days from forecast issuance, and all models were fitted using the `gam`-function from the **mgcv** package to allow for spline fits.

The models under consideration are based on two data sources: 1) time covariates and 2) ERA5 temperature data either in the form of principal components or as mean grid temperature. We have approached this as a structural modeling task, where we model trend and seasonality as fixed effect terms alongside temperature in a GAM model framework (Section 3.3). For conceptual simplicity we have made an assumption of normal uncorrelated errors.

The main challenge is finding the best combination of time and temperature covariates. The full model space is large. Each of the six time covariates have four different plausible parametrizations, not counting interactions, and we have 462 temperature measurements (stemming from the 22×21 temperature grid) and the same number of principal components. Combine this with an extensive cross-validation procedure and it becomes apparent that a full exploration of the model space is not feasible. However, the purpose of the principal component decomposition is to reduce the dimensionality of the relevant data. We limit our testing to the first 10 PCs, which together, over all data points, have an

explained variance of 97.1% (Section 4.2).

This still leaves an enormous number of possible models. One option would be to reduce either the size of the training data or the number of forecasting test periods. The effect of training data size is explored in Section 5.5, but the number of forecast test periods is desired to be kept high. More test observations are especially important for month to month accuracy estimation. There is considerable monthly variation in skill, and each month has one twelfth of the test data, making month-evaluation susceptible to outlier months.

Instead of an exhaustive search, therefore, we have opted for a two step procedure. We first look at the predictive performance of different parametrizations of each covariate. Then, having found good parametrizations, we combine them into more complicated multi-variable models (see Section 3.7). Thus, consistency in prediction results over many forecast periods is prioritized over a full exploration of the model space.[1] This approach is obviously faster, but another benefit is that we focus on a subset of the model space consisting of plausible models it makes sense to compare with each other. And from these comparisons we can ascertain where our prediction skill is coming from. The downside is that we are not guaranteed to find the best model in the set.

The rest of the chapter is as follows. In 5.1 we first describe the two baseline models, the intercept model and the climatology model, which are used as benchmarks for modeling predictive performance for uni- and multi-variable models respectively. Then, in Section 5.2, we describe the process through which we find the best performing parametrization of the time covariates. The best time covariate model is found to have a skill score of 0.42 compared to the climatology model. In Section 5.3 we investigate models which only utilizes temperature data. We find that the first principal component of temperature is a more accurate predictor than the grid mean temperature.

We then, in Section 5.4, look at models combining time and temperature information. The best model, GAM-PC1+2, includes both time covariates and crucially, the two first temperature principal components, and gains a skill score of 0.59 relative to climatology. In Section 5.5 we look at the factors which contribute to the high score of GAM-PC1+2, and assess the impact of changing the parameters of the training data. Finally, in Section 5.6 we compare our model with Lasso and XGBoost implementations. We find that the GAM-PC1+2 model performs slightly worse, but that it has key advantages in terms of interpretability and parsimony.

5.1 Baseline Models (Intercept and Climatology)

A natural baseline is the intercept model, or the empirical average for all observations before forecast issuance. Here, it is used to compare univariable models to see if a single predictor offers any advantage compared to the simple average. For observation at time t in an hourly sequence j with lead time l we define the intercept model estimate for demand observation y_t as:

$$\hat{y}_t = \frac{1}{t-l} \sum_{j=1}^{t-l} y_j. \tag{5.1}$$

[1] Alternative procedures e.g. Bayesian Model Averaging or lasso could also have been entertained, see also Section 5.6.

In meteorology a more common baseline model is referred to as the climatology or the climatological normal. It is an average, commonly set over a 30-year range, of monthly, daily or hourly observations (Arguez et al. 2012; Wilks 2011). For our purposes we refer to the climatology model as the empirical average over all observations for a specific hour on a specific day of the year for all preceding years. Formally, for observation y_{hdn} on hour h, day of year d and year n, the climatology baseline model estimate is given as:

$$\hat{y}_{hdn} = \frac{1}{n-1} \sum_{k=1}^{n-1} y_{hdk}. \tag{5.2}$$

It is a sensible choice of baseline for full models as it is relatively simple, and at the same time it encapsulates both the daily and annual cycles of the demand data.

5.2 Time Covariate Models

In this section we investigate the improvement in terms of predictive skill we can obtain from a good parametrization of the time covariates.[2] We will look at four model categories: 1) univariable models, 2) multivariable models without interaction terms, 3) bi-variable interaction models, and 4) multivariable models with interaction terms.

From the hourly time series of energy demand observations, we have access to date and hour information from which we can extrapolate weekdays, weeks, months, seasons and years, all of which plausibly warrant inclusion.[3] The main challenge in utilizing the time covariates is to find the best parametrizations of these features. Each of these covariates could be expressed in several different ways – specifically as factors, as continuous variables, through splines, or by sinusoidal terms.[4]

For each of the time covariates and for each parametrization of that covariate, we fit a model using that covariate as a single predictor with energy demand volume as the outcome. These models have the general form:

$$y_t = \beta_0 + f(x_t) + \epsilon_t. \tag{5.3}$$

Here $f(x_t)$ is a function of a single time covariate and differs between parametrizations in the following manner:

$$f(x_t) = \begin{cases} \beta_1 x_t, & \text{if } x_t \text{ param. as cont.,} \\ \sum_j^m \beta_j \, I_j(x_t), & \text{if } x_t \text{ param. as a factor,} \\ \beta_1 \cos(\frac{2\pi x_t}{\omega}) + \beta_2 \sin(\frac{2\pi x_t}{\omega}), & \text{if } x_t \text{ param. as sinusoidal,} \\ s(x_t), & \text{if } x_t \text{ param. as a spline,} \end{cases} \tag{5.4}$$

[2] Here the term 'covariate' is used loosely, as under some parametrizations a variable (e.g. 'hour') might be continuous or composed of several dummy variables.

[3] The 'season' variable is a function of month. The months December, January, February are combined into 'winter', March, April, May are combined into 'spring', etc.

[4] The sinusoidal term here employed is the decomposition of the sinusoidal waveform: $A\cos(2\pi\omega t + \phi) = \beta_1 \cos(2\pi\omega t) + \beta_2 \sin(2\pi\omega t)$, where A is the amplitude, ϕ is the phase shift, ω is the frequency, $\beta_1 = A\cos(\phi)$ and $\beta_2 = -A\sin(\phi)$ (Shumway et al. 2017).

where m is the number of levels in the factor variable x_t; the indicator function $I_j(x_t) = 1$ if $x_t = j$ or 0 if $x_t \neq j$; ω is a fixed frequency oscillation (e.g. set to 24 for hours, 7 for weekdays etc.); and $s(\cdot)$ is a smoothing term $s(x_t) = \sum_j^{d+1} \beta_j \, bs_j(x_t)$, which is a basis spline function with d degrees of freedom.

	Parametrization:			
Covariate	**Cont.**	**Factor**	**Sinusoidal**	**Spline**
Intercept	8004.57	–	–	–
Year	**8467.63**	9450.03	–	–
Season	6989.82	5901.40	**5896.06**	6726.04
Month	7830.97	5173.46	**5157.30**	5171.53
Week	7832.89	5067.29	5086.29	**5066.63**
Weekday	7894.09	**7831.86**	7881.07	7899.76
Hour	7822.06	**7289.62**	7490.59	7291.37

Table 5.1: RMSE for 22 univariable time models on the form of eq.(5.3) and the intercept model (5.1). The best predictive performance of a single covariate is the spline fit using 'week'. Bold indicates best parametrization.

The results (Table 5.1) show that for all variables, except for the trend term 'year', the continuous parametrization is the worst performing, which is not surprising given the cyclical nature of the other variables. Notice also that models including just the trend term 'year' are notably worse than the intercept model. Models including the terms 'week', 'month' and 'season', which describe the annual cycle, all showed considerable predictive performance compared to the intercept, and especially for the two former terms there was little sensitivity to the parametrization (excluding the continuous option). The two short cycles 'hour' and 'weekday' were best parameterized as factors, but their improvement over the intercept model is relatively modest, especially for the latter. The better performance of the annual cycle terms is mainly due to the greater variation in the annual demand cycle than in hourly or weekly demand cycles. Whereas the longer cycle terms give adequate predictions throughout the year, the short cycle models only perform well in select periods in spring and autumn when the observations are close to the hourly or weekly average. We also see that the more granular the annual cycle is described, the better the performance.

The next batch of models combine these covariates into non-interaction multi-variable models on the form:

$$y_t = \beta_0 + f_1(x_t^{(1)}) + ... + f_p(x_t^{(p)}) + \epsilon_t, \tag{5.5}$$

where $f_p(x_t^p)$ for variable $p \in 1, \cdots, 6$ is some parametrization of $x_t^{(p)}$. For 6 variables with up to 5 parametrizations (including also $f_p(x_t^{(p)}) = 0$ for non-inclusion) we have $5^5 \cdot 3 = 9375$ possible models on this form. As mentioned above, because of computational time we restrict ourselves to looking at only a small subset based on the results from the univariate models. The models we consider either consist of the best parametrization of each covariate found above or are neighbours of this model. In total, 6 such models were tested (Table 5.2). The model consisting of the best parametrization of each covariate

	Metric:	
Model	**RMSE**	**Skill Score**
Climatology	4114.54	–
Best param.	3264.24	0.371
No year term	3460.04	0.293
Season as factor	3263.51	0.371
Month as spline	**3255.39**	**0.374**
Week as factor	3262.58	0.371
Hour as spline	3268.49	0.369

Table 5.2: Model performance (RMSE and skill score) for multivariable non-interaction time models on the form of eq. (5.5) and the climatology model (5.2). The model utilizing the best parametrizations is a considerable improvement on the climatology model. Modifications of the former only improved predictive performance slightly, or not at all, having skill scores at roughly the same level. Bold indicates best performance.

found above had an RMSE score of 3264.24.[5] This is a substantial improvement not only on the univariable models, but also on the climatology model, with a skill score of 0.371 (Section 3.8). Since the trend variable 'year' performed worse than the intercept model, we also tried a modification of the above model, removing only the 'year' term. This model is noticeably worse, so even if 'year' on its own was not a good predictor, when other covariates are added, it is advantageous to keep it (at least in this setting). In the case of 'season', 'month', 'week' and 'hour', the RMSE scores were fairly similar for several parametrizations. This led us to also consider four other modifications where the best performing parametrization was swapped out in favor of the second best. Only very slight improvements in predictive performance were found. The best result was achieved by swapping the sinusoidal month parametrization with a spline (RMSE = 3264.24 vs 3255.39).

Having found a seemingly good parametrization, we also looked at how the model could be improved using interaction terms. This was motivated by data inspection where it is clear that the hourly and weekly cycles are quite different in summer compared to winter (Section 4.1). Since both the 'week' and 'hour' terms were parametrized fairly well both by splines and factors, we looked at both spline and factor interactions between these two shorter cycles, and the annual cycles of 'season', 'month' and 'week' respectively. We tested 12 such bi-variable models with one interaction term, using the same outcome as before, on the form:

$$y_t = \beta_0 + g(x_t^{(a)}, x_t^{(b)}) + \epsilon_t, \tag{5.6}$$

where $x_t^{(a)}$ is a short-term cycle term and $x_t^{(b)}$ an annual cycle term and where

[5] The model formula for this model is for observation at target time t, indicator function $I_k(\cdot)$, basis spline function $bs_k(\cdot)$, year yr, month m, season sn, week w, weekday wd, and hour h: $y_t = \beta_0 + \beta_1 yr_t + \beta_2 \cos(\frac{2\pi m_t}{12}) + \beta_3 \sin(\frac{2\pi m_t}{12}) + \beta_4 \cos(\frac{2\pi sn_t}{4}) + \beta_5 \sin(\frac{2\pi sn_t}{4}) + \sum_{k=1}^{9} \beta_{5+k}\, bs_k(w_t) + \sum_{k=1}^{6} \beta_{14+k}\, I_k(wd_t) + \sum_{k=1}^{23} \beta_{20+k}\, I_k(h_t) + \epsilon_t$.

$g(\cdot)$ depends on the type of interaction:

$$g(x_t^{(a)}, x_t^{(b)}) = \begin{cases} \sum_k \sum_j \beta_{(j-1)\cdot|k|+k} \, I_k(x_t^{(a)}) : I_j(x_t^{(b)}), & \text{if factor interaction,} \\ s(x_t^{(a)}, x_t^{(b)}), & \text{if spline interaction,} \end{cases} \tag{5.7}$$

where $s(\cdot)$ is a spline function. On the whole, the spline interaction models performed considerably worse than the factor interaction ones (Table 5.3). Both 'short cycle' terms achieved the best results when paired in a factor interaction with 'week'. These are also the computationally heaviest models to run; the interaction between 'hour' and 'week' containing $24 \cdot 53 + 1 = 1273$ parameters.

		Annual Cycle Term:		
Short Cycle Term	**Interaction**	**Season**	**Month**	**Week**
Weekday	Spline	**6894.59**	7752.75	7758.70
Weekday	Factor	5662.67	4903.64	**4838.51**
Hour	Spline	**6340.32**	7245.58	7263.93
Hour	Factor	4818.49	3869.38	**3719.31**

Table 5.3: RMSE for time covariate models with 1 interaction on the form of eq. (5.6). Each row shows a short-term cycle, the type of interaction, and the RMSE for the interaction between the short-term cycle and 3 different annual cycles. The best performing model on this form is the factor interaction between 'hour' and 'week'. Bold indicates best interaction pair for short cycle term.

These interaction models do not show any overfitting tendencies, as the model with the most parameters for each short-term cycle has also been the best performing model. The factor interactions can be seen as a set of intercept-relative averages for specific time-based intersections of the demand observations. They are performing well not only because the intersections capture relevant distinct sections of the output, but also because of sufficient amounts of data enabling a good description of each section. Notice that the climatology model can be viewed similarly, but with more parameters ($24 \cdot 365 = 8760$) and without the intercept term. The worse performance of the climatology model compared to the interaction model with 'hour' and 'week' shows that there is a limit to the utility of carving up smaller and smaller intersections.

The last set of models in this section adds interaction terms to the best multivariable non-interaction model. This naturally increased the model size; the biggest such model contained over 1500 parameters. Concerns about possible overfitting and computational time therefore led us to consider not only the best interaction term for each short cycle, but also the second best interaction term. Four models were tested using different combinations of the interaction terms in addition to the other covariates from the previous full non-interaction model. They were on the following form:

$$y_t = \beta_0 + f_3(x_t^{(3)}) + ... + f_p(x_t^{(p)}) + g_1(h_t, x_t^{(a)}) + g_2(\mathrm{wd}_t, x_t^{(b)}) + \epsilon_t, \tag{5.8}$$

where $g_1(\cdot)$ and $g_2(\cdot)$ are factor interaction functions, h_t is hour and wd_t is weekday and $x_t^{(a)}$ and $x_t^{(b)}$ is either 'month' or 'week'. Notice that when we

include 'hour' and 'weekday' in interaction terms, we remove them as free-standing covariates.[6] Four different full interaction models were tested where the model with the fewest parameters was the best performing (Table 5.4). In this case we see some evidence of overfitting. The model with the most parameters which includes both hour:week and weekday:week has the lowest training error, but it is outperformed on the test evaluation by two more parsimonious versions.

	Hour Interaction (g_1):	
Weekday Interaction (g_2):	**Hour:Week**	**Hour:Month**
Weekday:Week	3199.39	3211.30
Weekday:Month	3136.35	**3126.09**

Table 5.4: RMSE for time covariate models with two interactions on the form of eq.(5.8). The cross-table shows the performance of the full model with four different combinations of two interaction functions $g_1(\cdot)$ and $g_2(\cdot)$. The best combination is the model utilizing the interactions between 'hour' and 'month' as well as 'weekday' and 'month'.

Overall, the best time covariate model is the full interaction model utilizing 'hour' and 'month' as well as 'weekday' and 'month'.[7] Comparing the time covariate models with the climatology model (Table 5.5) we can see that the best interaction model improves upon climatology with a skill of 0.423 over the 30 day test period. As was evident from the data inspection (Section 4.1), energy demand is strongly time dependent, and by utilizing an effective parametrization we have built a model that is a considerable improvement on the climatology model when it comes to forecasting energy demand.

Metric	**Climatology**	**No Interaction**	**Interaction**
RMSE	4114.54	3255.39	**3126.09**
Skill	–	0.374	**0.423**

Table 5.5: Model performance of time covariate models comparing climatology with the best models with and without interaction terms.

An additional benefit of these time covariate models is that they add another point of reference with which we can compare the models that incorporate temperature. In this manner we obtain an assessment of how much the addition of temperature data (explored in the next section) in our model is worth to the prediction task.

[6] This was done because of singularity issues in some models, but also to keep the number of parameters down. Whether predictive accuracy could be improved by keeping these terms is a question left for another time. The standard choice is to include them.

[7] We write for hour h, week w, weekday wd, month m, season sn and year yr, where each $bs_k(\cdot)$ is a spline term: $y_t = \beta_0 + \beta_1 yr_t + \beta_2 \cos(\frac{2\pi sn_t}{4}) + \beta_3 \sin(\frac{2\pi sn_t}{4}) + \sum_{k=1}^{8} \beta_{3+k}\, bs_k(m_t) + \sum_{k=1}^{8} \beta_{11+k}\, bs_k(w_t) + \sum_{k=1}^{24} \sum_{j=1}^{12} \beta_{19+(k-1)\cdot 12+j}\, I_k(h_t) : I_j(m_t) + \sum_{k=1}^{7} \sum_{j=1}^{12} \beta_{19+24\cdot 12+(k-1)\cdot 12+j}\, I_k(wd_t) : I_j(m_t) + \epsilon_t$

5.3 Temperature Models

In this section we look at different ways of utilizing temperature data for predicting energy demand. The first models considered utilize the mean grid temperature, $\bar{z}_t$, at each target time t as a single predictor. We write this as:

$$y_t = \beta_0 + f(\bar{z}_t) + \epsilon_t, \tag{5.9}$$

where $f(\bar{z}_t) = \beta_1 \bar{z}_t$ in the linear case and $f(\bar{z}_t) = \sum_{d=1}^{D} \beta_d b_d(z_t)$ in the spline case, where $b_d(\cdot)$ is a basis spline. The spline fit performs slightly better (Table 5.6), but notice that the mean grid temperature is a worse single predictor than both 'month' and 'week' (Table 5.1).

The rest of the models in this section are based on temperature data in the form of principal components. We first look at models with only one principal component as a predictor, of the form:

$$y_t = \beta_0 + f(C_t^j) + \epsilon_t, \tag{5.10}$$

where C^j is principal component j (described in Chapter 3) limiting ourselves to the first 10 principal components so we have $j \in \{1, \cdots, 10\}$, and where $f(\cdot)$ is either a linear or a spline function.

		Principal Component:				
Model	**M-G**	$\mathbf{C^1}$	$\mathbf{C^2}$	$\mathbf{C^3}$	$\mathbf{C^4}$	$\mathbf{C^5}$
RMSE - Spline	5401.71	**5323.50**	7971.52	8485.08	8343.64	8263.97
RMSE - Linear	5584.67	5549.05	8000.77	8191.44	8129.25	8239.27
Model		$\mathbf{C^6}$	$\mathbf{C^7}$	$\mathbf{C^8}$	$\mathbf{C^9}$	$\mathbf{C^{10}}$
RMSE - Spline	–	7907.42	7837.66	8032.13	8599.72	7820.39
RMSE - Linear	–	7946.14	7974.37	8005.99	8018.85	8026.34

Table 5.6: RMSE for univariable principal component models, on the form of eq. (5.10) and mean grid (M-G) models (eq 5.9). The C^1 and Mean Grid models are the only two models that substantially outperform the intercept model, the former having a slight edge.

The results (Table 5.6) show that among the principal component models there is only one standout model which offers a large improvement over the intercept model (RMSE of 8004.57). This is the C_t^1 model, which is best parameterized through a spline. Compared to the single time covariate models we see that C_t^1 is better than all other covariate models except for the 'month' and 'week' predictors when under their best parametrization. The improvement of the C_t^1 model with regard to the mean of the temperature grid is slight; the skill score using the latter as baseline is just 0.029. However, an advantage of the principal component dimensionality reduction is that we can combine more than one principal component in the model, as we do below:

Two different strategies for combining the principal components were attempted. Both were centered around the first principal component, C_t^1, which as we saw in Section 4.2 has an in-sample variance explained of 84.2%. Since the C_t^1 spline model was an improvement on the linear version, we

	Principal Component Combination:				
Model	$\mathbf{C^1}$	$\mathbf{C^1+C^2}$	$\mathbf{C^1+\cdots+C^3}$	$\mathbf{C^1+\cdots+C^4}$	$\mathbf{C^1+\cdots+C^5}$
RMSE	**5323.50**	5334.81	5777.55	6270.58	6948.27
Model	$\mathbf{C^1+\cdots+C^6}$	$\mathbf{C^1+\cdots+C^7}$	$\mathbf{C^1+\cdots+C^8}$	$\mathbf{C^1+\cdots+C^9}$	$\mathbf{C^1+\cdots+C^{10}}$
RMSE	6906.23	6942.78	6869.10	6862.39	6905.28

Table 5.7: RMSE for models with multiple principal components on the form of eq. (5.11). None of the combination models represent an improvement upon the C_t^1 model.

henceforth continue modeling all principal components by splines. We fitted 9 models where the first of these combined C_t^1 and C_t^2 and each subsequent model added another spline, the last one containing all PCs including C_t^{10}. These models are of the form:

$$y_t = \beta_0 + s(C_t^1) + ... + s(C_t^j) + \epsilon_t, \tag{5.11}$$

for $j \in \{2, \cdots, 10\}$ and $s(\cdot)$ is a spline function. It is clear there is no gain in overloading the model with principal components (Table 5.7). Adding C_t^2 gives a model performance that is very slightly worse than the model with just C_t^1. Then there is a marked drop off at each step when adding C_t^3, C_t^4 and C_t^5. Performance thereafter stabilized at a low level around 6900. These models show an overfitting pattern from the addition of C_t^2, as when the number of parameters increase the training RMSE decreases while the test RMSE increases.[8]

Even if model performance did not improve when adding several principal components to C_t^1, we also checked whether a specific combination of C_t^1 and one other principal component could improve performance. We therefore ran another 8 models, each a combination of C_t^1 and another principal component, on the form:

$$y_t = \beta_0 + s(C_t^1) + s(C_t^j) + \epsilon_t, \tag{5.12}$$

for $j \in \{2, \cdots, 10\}$, with $s(\cdot)$ being a spline function. These two-predictor models (Table 5.8) fared better than the last batch, but overall the results were lackluster. Adding C_t^2 (the same model as in 5.7), C_t^7, and C_t^{10} to C_t^1 had a very slight negative effect on predictive performance, while adding C_t^6 actually improved the model slightly.

To summarize, utilizing just temperature information for predicting energy demand did not yield as good results as when utilizing just time covariates. Of importance is the observation that none of the models containing just temperature information performs better than the climatology model. The best temperature model employed the first and sixth principal components in a spline fit. The first principal component is clearly a good predictor as was surmised in Section 4.2, but adding other principal components generally did not improve model performance. To see whether the model containing C_t^6 represents an actual improvement over the C_t^1-model we employed a permutation test using skill score as a test statistic over 10000 permutations. We obtain an observed

[8] We see the same pattern in Figure 5.1 for a similar model type.

	Principal Component Combination:				
Model	$\mathbf{C^1}$	$\mathbf{C^1 + C^2}$	$\mathbf{C^1 + C^3}$	$\mathbf{C^1 + C^4}$	$\mathbf{C^1 + C^5}$
RMSE	5323.50	5334.81	5759.80	5474.16	5545.83
Model	$\mathbf{C^1 + C^6}$	$\mathbf{C^1 + C^7}$	$\mathbf{C^1 + C^8}$	$\mathbf{C^1 + C^9}$	$\mathbf{C^1 + C^{10}}$
RMSE	**5298.79**	5328.91	5372.79	5389.64	5334.22

Table 5.8: RMSE for two-predictor PC models on the form of eq. (5.12). Only the addition of C_t^6 to the first principal component resulted in a slight improvement in predictive performance.

test static skill score of 0.009 with a p-value of 0.0898, which is not significant at $\alpha = 0.05$. For a note on the use of permutation test see Section 5.5.

5.4 Final Structural Demand Model

Finally, we consider models that combine the time covariates with temperature information. The models in this section will build upon the 4 multivariate interaction models presented in Table 5.4 and add to each of these temperature information of the form:

$$y_t = f(x_t) + h(z_t) + \epsilon_t, \tag{5.13}$$

where $f(x_t)$ is a multivariable interaction model utilizing time covariates, x_t, specified in eq. (5.8). $f(x_t)$ includes the interaction functions $g_1(\cdot)$ and $g_2(\cdot)$, and $h(z_t)$ is a function of temperature information z_t.

	Hour Interaction (g_1):	
Weekday Interaction (g_2):	**Hour:Week**	**Hour:Month**
Weekday:Week	2881.55	2901.25
Weekday:Month	**2818.73**	2832.62

Table 5.9: RMSE for mean grid models with 4 different time covariate interaction combinations, using $h(z_t) = s(\bar{z}_t)$.

The first set of such models include temperature in the form of mean grid, i.e. using $h(z_t) = s(\bar{z}_t)$, where $s(\cdot)$ is a spline function. The mean grid models all represented a substantial improvement on the time covariate models. The best model employs the hour:week and weekday:month interactions (Table 5.9). It has a skill of 0.187 compared to the best non-temperature model, which makes it clear that the addition of temperature information greatly improves predictive ability.

We then combined the four time covariate interaction models with both of the principal component combination set-ups from the last section. For the set-up with an increasing number of principal components, we tested a total of 36 models, using $h(z_t) = s(C_t^1) + ... + s(C_t^j)$. Figure 5.1 summarizes the results. The interaction setup including hour:week and weekday:month was consistently the best performing and is shown in red for different numbers of principal

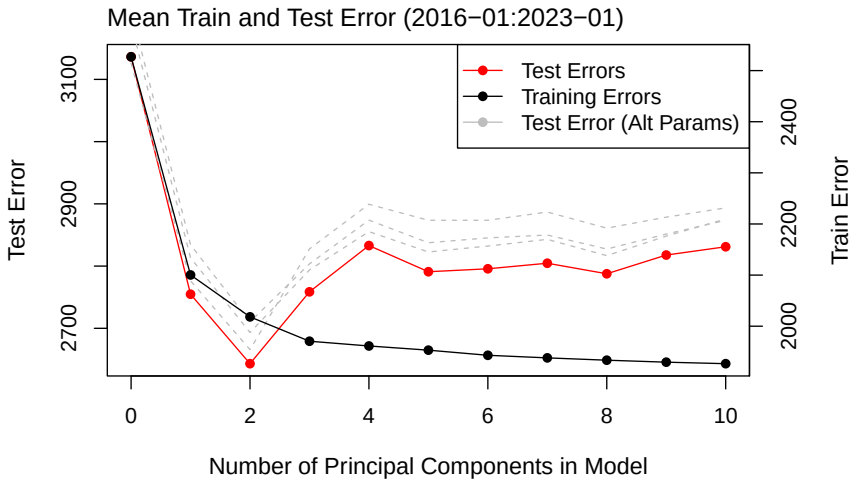


Figure 5.1: Training and test errors for 36 GAM-PC models. The red line shows the test RMSE for the best performing interaction model. It achieves the best results by using the two first PCs. The black line shows the training error for the best interaction model. It is steadily decreasing, while the improvement in the test error stops after the addition of the second PC. Test results for the other interaction models are shown in the grey dashed lines.

components in the model. For models including up to the three first principal components we see a clear improvement over the mean grid models. This is largely independent of time covariate interaction parametrization. Earlier, we saw no improvement by adding C_t^2 to C_t^1 in a model without time covariates. Now, we see a substantial improvement by having C_t^2 in the model, when the time covariates are present as well. For models including more than the two first PCs there is clear evidence of overfitting. Even though the training error is reduced for each added principal component this improvement stops in the test sets after adding C_t^2.[9] These points also hold for the performance of the other 3 interaction setups whose results follow the same general behaviour (shown in grey). Table 5.10 shows the exact results for the 4 interaction models containing both C_t^1 and C_t^2.

	Hour Interaction (g_1):	
Weekday Interaction (g_2):	**Hour:Week**	**Hour:Month**
Weekday:Week	2693.27	2710.35
Weekday:Month	**2643.16**	2665.76

Table 5.10: RMSE for GAM-PC1+2 models with 4 different time covariate interaction combinations, using $h(z_t) = s(C_t^1) + s(C_t^2)$.

Finally, an additional 32 models were tested. For each interaction setup,

[9] This pattern was also present in the similar model set-up without time covariates from the last section.

we added a pairing of C_t^1 with one other principal component. The results (Figure 5.2) show that no other principal component beyond C_t^2 improves the model substantially when added to C_t^1, and again we see the same pattern for all interaction frameworks. In the previous section we saw that adding C_t^6 to C_t^1 increased the model performance slightly, but no such effect is seen with the presence of the time covariates.

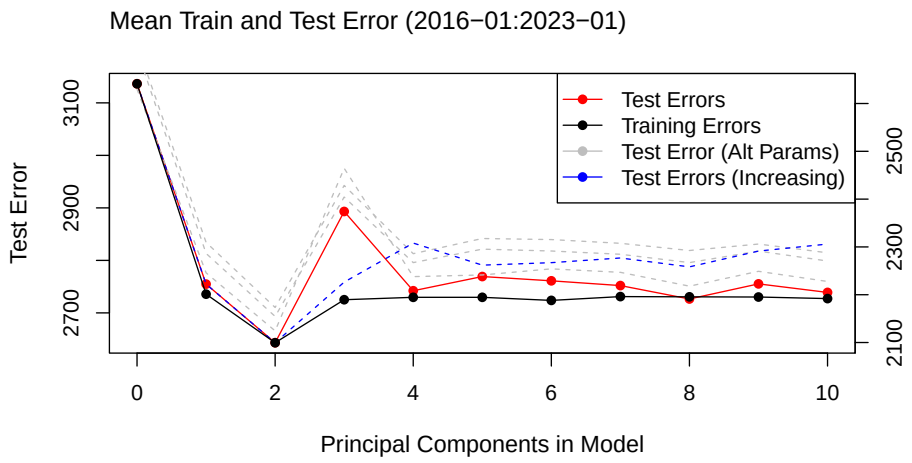


Figure 5.2: Training and test errors for 32 GAM-PC models with $h(z_t) = s(C_t^1) + s(C_t^j)$. The x-axis shows the additional principal component C_t^j.

Summing up our findings, we see that the best model overall is GAM-PC1+2, which we write as:

$$y_t = f(x_t) + s(C_t^1) + s(C_t^2) + \epsilon_t, \tag{5.14}$$

where $f(x_t)$ is specified by the interaction model including the interaction terms for hour:week and weekday:month, and $s(\cdot)$ is a spline function. From Table 5.11 we see that GAM-PC1+2 offers a skill improvement on all other models. We also see that for each step in the model selection procedure, we have been able to find better performing models. The best performing model by RMSE achieves a skill score of 0.59 when compared to climatology. It leverages both the effective parametrization of time covariates as well as using a spline on PC transformed temperature data. 67.2% of the reduction in RMSE between the climatology model and GAM-PC1+2 comes from utilizing time covariates; a further 20.9% comes from adding temperature information (as mean-grid); and the last 11.9% comes from utilizing PCs instead of mean grid temperature.

5.5 Assessing Model Performance

So far, we have observed, first, that adding time covariates to the model improves model performance substantially, and second, that adding temperature data in the form of PCs gave a further increase in skill. In this section we will look at the performance, virtues and limitations of the GAM-PC1+2 model in terms of model sensitivity to training and test modifications.

		Skill Score with Reference to:		
Model	**RMSE**	**Climatology**	**Interaction**	**Mean-Grid**
Climatology	4114.54	–	–	–
Interaction	3126.09	0.42	–	–
Mean-Grid	2818.73	0.53	0.19	–
GAM-PC1	2754.81	0.55	0.22	0.04
GAM-PC1+2	**2643.16**	**0.59**	**0.29**	**0.12**

Table 5.11: Model performance (RMSE and Skill) of key models. The best performing model is GAM-PC1+2, which had a substantial positive skill compared to all other models.

We start by looking at how the GAM-PC1-2 model obtains better results than the other models we have looked at. Electricity demand has clear temporal cycles and adding temperature information enables us not only to predict, e.g., that demand is high because it is winter, but also that demand is very high because it is a colder winter than usual. In Figure 5.3 this is illustrated by comparing model predictions for two weeks (10 and 49) in 2021. The top-right plot shows that week 49 in 2021 had an unusually low temperature (red) and a corresponding spike in demand (green). The GAM-PC1+2 model is able to translate this temperature information into adjusting its prediction of demand upwards to accurately track the demand spike. This can be seen directly (bottom-right plot) where the red line (GAM-PC1+2) does a much better job of tracking the observed demand (black dots) than the other two models. We also see (top-left) that for week 49, GAM-PC1+2 had a particularly high skill compared both to climatology and the best time interaction model.

Week 10 offers a contrasting case. There was a high positive temperature anomaly and a corresponding slight drop in demand. The performance of GAM-PC1+2 is very good compared to climatology, but the interaction model without temperature is better overall for this week. The climatology does not have weekday information, which accounts for the bad performance on the 6th and 7th of March.[10] And towards the end of the week it appears that the GAM-PC1+2 model over-adjusts upward.

If we look at performance by month there are large variations in RMSE scores, not only for GAM-PC1+2, but also for the other main models (from Table 5.11) we have considered. These models generally perform worse in winter and better in summer. For GAM-PC1+2, the worst month is December with an average RMSE of 3353.91 compared to August's average of 2169.563, which is the best performing. This might be tied to larger variations in demand during the winter months connected to rapid temperature shifts.

Crucially, however, even if model performance is worse in the winter months, the model improvement of GAM-PC1+2 with respect to the other models is higher during winter. In the top left plot of Figure 5.4, we see that the skill of the GAM-PC1+2 model compared to the other models is generally positive

[10]The **data.table** `week`-function does not count week numbers in the same way as a standard calendar. This was discovered too late to fix the plot, but has no bearing on the results of the analysis. The plot reports results for 7-day periods (artificial weeks) starting on Thursdays.

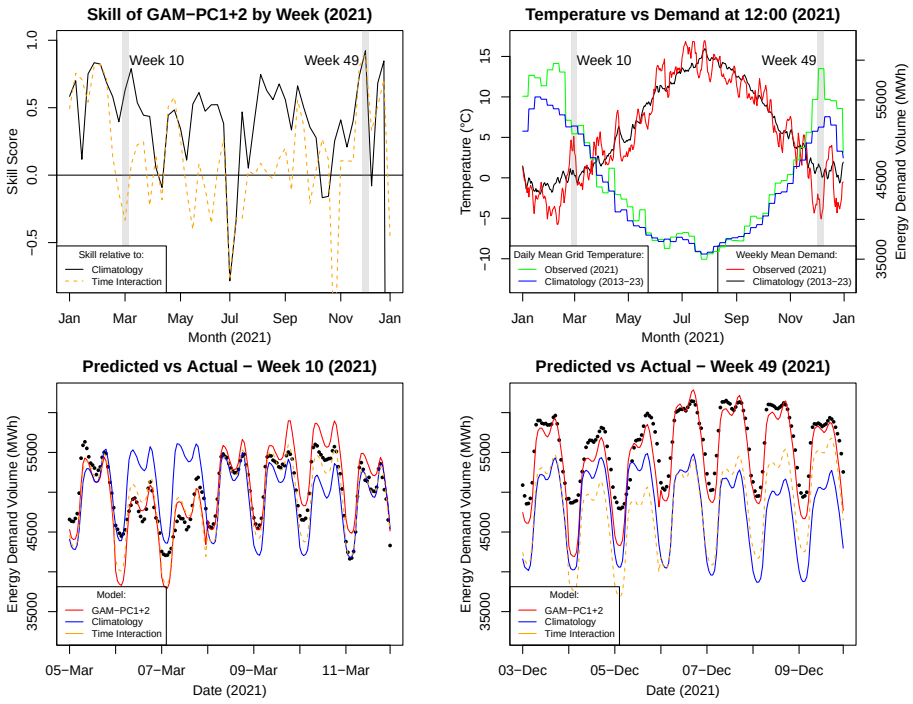


Figure 5.3: Prediction performance using 2021 observations comparing the GAM-PC1+2 model with climatology and the best time interaction model in terms of skill by week (top-left) and actual vs predicted values for weeks 10 and 49 (bottom). Top-right shows the relation between temperature and demand throughout the year.

across the board, but especially high during the winter months. We also see that GAM-PC1+2 outperforms GAM-PC1 in all months but October and November. As we saw in Section 4.2, the second principal component, C_t^2, is related to the land-sea temperature differential. It appears that projecting this differential back on the temperature data is most beneficial in winter, and especially for the months of February and March. This corresponds with a time of year where C_t^2 tilts mostly positive and exhibits more variation. It is plausible to think that this variation reflects variation in the sea breeze (Steele et al. 2014), which in turn might affect demand patterns.

Shifting our focus to the bottom plot (Figure 5.4) we see the skill relative to climatology for each individual test month for three models: The interaction model, GAM-PC1 and GAM-PC1+2. Their performance is broadly similar, but the latter seems to outperform the other two rather consistently. There are, however, two periods where drops in performance is particularly jarring. In spring and summer of 2018 we find a prolonged period with negative skill for all three models. The poor skill here is not because these models have very elevated RMSE scores in this period, but largely because the climatology model is doing unusually well. The second drop occurs in autumn 2022 and the models which include temperature information actually do worse here, while the interaction model stays close to the performance of the climatology. The GAM-

PC models predicted a higher energy demand than both the non-temperature models and compared to what actually occurred. It is tempting to suggest that the GAM-PC models simply did not account for the effect of the new European energy situation, including higher energy prices and encouragements of using less energy even during cold weather periods (IEA 2023).

In the top right plot we see that the skill of the models depend on the hourly cycle. The models are not doing as well at night-time relative to climatology as they do during the day, when demand is higher. The difference between GAM-PC1+2 (red) and GAM-PC1 (black) is fairly constant through the daily cycle, but adding temperature information is most beneficial at night (between 21:00 and 4:00) if we compare the temperature models with the time interaction model without temperature (yellow).

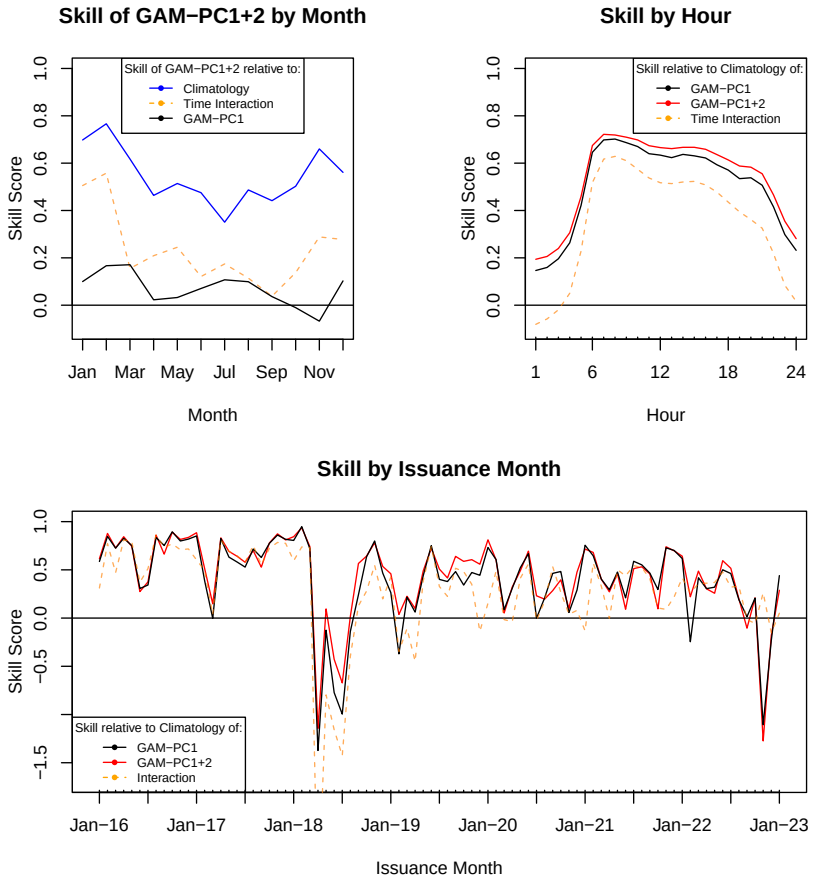


Figure 5.4: Skill by month, hour and forecast issuance month. The top-left plot shows the skill of GAM-PC1+2 relative to three other models, while the other two plots each show the skill of three models relative to climatology.

To test the robustness of the results, we perform a non-parametric Monte Carlo residual permutation test. We want to detect whether results of different models are significantly different. These tests assume exchangeability. But since our model outputs are highly correlated time series, we do not strictly satisfy

test assumptions. In Section 3.6 we described a copula method that takes account of the correlation structure between model outputs to form aggregates. Unfortunately, we have not been able to develop a testing framework that takes account of the correlation, but we will nonetheless present test results as a guiding tool. We tested all models in Table 5.11, except climatology. For each model we ran a permutation test (10000 permutations, $n = 60864$) to check if said model was significantly different from the model performing slightly worse. In all cases we obtained a p-value of 0.0000, strongly indicating that the results are significantly different.

To check for sensitivity to changes in the training period we ran the GAM-PC1 and GAM-PC1+2 models over 6 different training period intervals. The effect of changing the training period is considerable in terms of RSME scores, but not with regard to which model performs the best. In the initial set-up we used a 5-year training period, meaning all models were trained on the five years of data directly preceding each forecast issuance. The initial training period choice was made as a compromise between the wish for using as much data as possible and computational time.

Using the 5-year training interval the best GAM-PC model had an RMSE of 2643.16. By reducing the training period down to 3 years (Table 5.12) we saw a considerable improvement with an RMSE of 2537.97 which represents a skill of 0.078 compared to the same model with 5 years of training. As expected, the performances are worse with limited training data. It is more surprising that for each additional training year added, after 3 years, the performance of the models drops, and that the worst performing training interval included all years up to forecast issuance. This suggests that the relation between energy demand and the predictors might have undergone a shift during the period for which we have data. A line of investigation we will not pursue further here, could re-run the models from Sections 5.2 and 5.3 over different training intervals and check whether this change was more apparent for the time or temperature predictors.

	Training Period (years):					
Model	**1**	**2**	**3**	**4**	**5**	**All**
GAM-PC1	2861.92	2646.71	2637.48	2707.67	2754.81	2958.89
GAM-PC1+2	2827.23	2588.27	**2537.97**	2608.55	2643.16	2834.80

Table 5.12: Effect of training period on RMSE of the two best GAM-PC models.

Lastly, we look at the issue of sensitivity to testing periods or lead time. A key assumption made during the tests of the structural demand model (Section 3.3) has been that we have access to perfect information about future weather 15-45 days out from forecast issuance. The effect of this assumption should be seen directly from the model output. If we look at model performance for GAM-PC1+2 by lead time (Figure 5.5) we first observe that it exhibits strong fluctuations. Performance does, unexpectedly, slightly worsen further away from forecast issuance. This is indicated by the linear trend (red) with an increase per lead time of $\beta_1 = 0.18$. We consider this spurious, and an artefact of the both the fluctuations and the test window. A slightly smaller or larger test window would diminish the trend. This can be seen by the blue line which shows a diminished ($\beta_1 = 0.08$) trend for days 15:40. A much clearer lead time

dependence will become apparent when considering models with temperature forecast inputs (see Chapter 7).

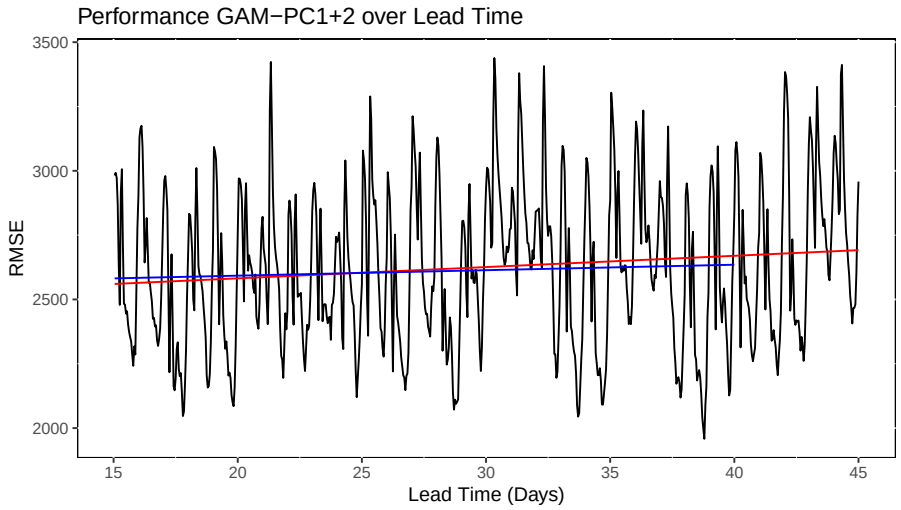


Figure 5.5: RMSE GAM-PC1+2 over lead time. The red line indicates the linear trend over days 15:45, while the blue line shows the trend for days 15:40.

5.6 Alternative Model Implementations

In this section we will compare the results for the best performing GAM-PC1+2 model with two different models: Lasso and XGBoost. The same cross-validation set-up as before was utilized, with a caveat: For these models, good performance relies on finding the best performing tuning parameter. Because of this, we first ran broader grid searches over a range of tuning parameter values, before we based on the results of the broad search ran finer searches over smaller areas.

Lasso

The first alternative model implementation we ran was the Lasso. The Lasso model is a regularizer which constrains the size of the β-coefficients through including a penalty term regulated by the tuning parameter λ. It has feature selection properties in the sense that it can set coefficient values to 0, effectively removing the corresponding covariate from the model. The formula for the estimated β-coefficient is given by:

$$\hat{\beta}^{lasso} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{t=1}^{n} (y_t - f(x_t, \beta_j))^2 + \lambda \sum_{j=1}^{p} |\beta_j|_1 \right\}, \tag{5.15}$$

where p is the number of parameters, λ is the regularizing tuning parameter and $|\cdot|_1$ is the L^1 (Manhattan) norm (Hastie et al. 2009). We ran three model set-ups with different specifications of the model function $f(x_t, \beta_j)$ utilizing the Lasso framework. Tuning these models was done in two steps where we did a broader and then a narrower search for finding the best tuning parameter, λ.

The performance of the `glmnet`-function is seed dependent because of the warm start, so marginally different results are obtained if the λ-sequence is specified differently. Since the differences are so marginal, we have opted to report only one results for each model.

In the first set-up we utilized only temperature information from the 462 individual grid points, each grid point being a predictor. We have $f(z_t, \beta_j) = \sum_{j=1}^{p} z_t^{(j)} \beta_j$, where z_t is a vector of grid point temperatures. The first broader grid search for this model was done over 1000 λ-values in the range of 0-20. The lowest RMSE was 4763.75 and was found for $\lambda = 4.14$. We then ran a second run over 1000 λ-values in the range 3-5. This resulted in a very slight improvement; the best RMSE-value was 4758.90. This performance is better than all other models with just temperature (compare Table 5.6).

In the second set-up we included time covariates in addition to the temperature grid points. In this set-up we use a straight-forward factorization of all time covariates. The best results found in the broader search achieved an RMSE of 2606.10 at $\lambda = 7.91$, a more granular search over the range 7-9 yielded the same result 2606.10 now at $\lambda = 7.79$.

The third set-up utilized a similar time covariate set-up to the best one. Here we expand the parameter space by utilizing the same two factor interactions and instead of using a spline and sinusoidal term, we factorize these as well, in addition to all the temperature grid points. At $\lambda = 7.31$ we found the best RMSE of 2537.05, a marked improvement on the previous best model found. A narrower search over values 7-8 again yielded a slight improvement with 2535.971 at $\lambda = 7.28$.

XGBoost

For the XGBoost model, we used the same cross-validation approach as previously. Similarly to the Lasso framework, we ran three model set-ups with different covariate inputs. The XGBoost algorithm, which utilizes components from regression trees, gradient boosting and penalized regression works iteratively to minimize the following equation:

$$[\sum_{t=1}^{n} L(y_t; f(x_t; \beta))] + \gamma\, T + \frac{1}{2}\, \lambda\, O(\beta)^2. \tag{5.16}$$

where γ is the pruning term controlling the number of terminal nodes T, and $\lambda\, O(\beta)^2$ is the regularization term (Chen and Guestrin 2016; Chen 2014). In our case we use the squared error loss: $L(y_t; f(x_t)) = \frac{1}{2}(y_t - f(x_t))^2$.

For regression problems XGBoost can be operationalized either as an ensemble of decision trees ('gbtree') or as an ensemble of linear models ('gblinear'). Preliminary trial runs heavily favoured the latter as both faster and more accurate for our problem. We will therefore only present results for the linear version.

The XGBoost is an iterative algorithm which at each step re-fits the residuals of the last iteration's prediction using weak learners. The most important tuning parameter for XGBoost is the stopping parameter M, controlling the number of refitting steps. The step size parameter η was set to 0.5 throughout the testing phase. This is higher than the standard of 0.3. Higher step size is faster as the algorithm takes bigger steps toward the solution at each iteration, but

comes with the risk of over-shooting good solutions. The choice of $\eta = 0.5$ was made as a compromise between computation time and minimizing the risk of over-shooting. [11]

XGBoost also has a penalty term which can be specified either as L^1 or L^2-regularization. We use the L^1 penalty because of its variable selection properties. To find well-performing combinations of M and λ, we used a two-step procedure. In step 1, with λ set to 0, we found the best stopping point M. We then searched for the best λ at step M previously found. In step 2, this combination was then used as a starting point for a small grid search over parameter values close to those already found. The ranges used in both steps varied according to the covariate input, but the ranges were much less granular than in the case of the lasso algorithm. This is not only because we are tuning two hyper-parameters, but also because the XGBoost model had to be refitted for each choice of M, in contrast to the Lasso where predictions for each λ are directly available. The best results (Table 5.13) using only grid temperatures

Model	**Grid Temp**	**Factor**	**Interaction**
Lasso	4758.90	2606.10	2535.97
XGBoost	4761.17	2604.95	2544.03

Table 5.13: Model performance (RMSE and skill score) comparing Lasso and XGBoost models with different parametrizations.

were found at $M = 820$ and $\lambda = 4.25$. When using time covariates as factors in addition to the temperature grid, we settled on $M = 120$ and $\lambda = 4.00$. And when including time interactions the best hyper-parameters was $M = 33$ and $\lambda = 7.00$. The results for the XGBoost models are roughly at the same level as for the Lasso models, and slightly better than the GAM-PC1+2 models. It must be said that only a fraction of the possible tuning settings have been explored for combinations of M and λ. And it might also be possible to improve the model with a smaller choice of step size, η. It is notable that all models did better when time interactions were introduced.

Model	**Climatology**	**XGBoost**	**Lasso**	**GAM-PC1+2**
RMSE	4114.54	2544.03	2535.97	2643.16
Skill	–	0.618	0.620	0.587

Table 5.14: Model performance (RMSE and skill score) comparing alternative model implementations with GAM-PC1+2 for models using 5 years of training data. The skill score is shown with reference to climatology.

The performance of the Lasso and XGBoost models is clearly better than that of the GAM-PC1+2 model (Table 5.14). But the latter model has two clear advantages: Interpretability and model parsimony. With regard to interpretability, the intuition behind GAM-PC1+2 is described both in Section 4.2 and in Section 5.5. The GAM-PC1+2 model attempts a slightly

[11] A similar outline of the XGBoost model was presented in a written exam assignment in the course STK-MAT2011 titled "Shapley Values in Explainable Machine Learning", by Eirik Sjåvik.

different task than the two other models. The latter might be (somewhat simplistically) described as looking for the temperature grid points which have the most influence on energy demand. The GAM-PC1+2 is modeling the relation between demand and the temperature field as a whole over the Nordic region. The advantage of this is that we can, through PCA, drastically reduce the number of predictors in the model. This parsimony greatly facilitates the incorporation of NWP forecasts into the model, as we now only have two temperature targets to predict.

The Lasso models, in contrast, are despite the regularization, not able to reduce the size of the temperature variable set by a great amount. Depending on the model run, the Lasso models consistently keep well over 100 temperature grid points in the model. This means that it is also harder to conceptualize how the Lasso models make use of the temperature grid point predictors. Because the temperature field is highly correlated, the selected grid points are often located in areas far from where the electricity is consumed. The high correlation might also be the reason why the Lasso models are not able to select the same grid points on a consistent basis. If the consistency is not there, we cannot take advantage of the dimensionality reduction achieved by regularization when porting the model to another setting. The same considerations apply for the XGBoost models, except that these exhibit more stability in the covariates they select.

We conclude this section with a short summary of the findings of this chapter. Most importantly, we have found that the GAM-PC1+2 model is the best model we have explored that incorporates both time covariates and principal components of temperature. This model represents a clear improvement in terms of predictive performance both with regard to the baseline climatology model and compared to models employing only time covariates. We have also seen that the inclusion of temperature information leads to substantial improvements in predictive performance. But we also observed that a good parametrization of time covariates yielded better predictive performance than temperature information on its own. Compared to alternative model implementations (Lasso, XGBoost), the GAM-PC1+2 performs slightly worse, but has key advantages when it comes to interpretability and parsimony.

CHAPTER 6

Problem 2: Probabilistic Temperature Forecasting Utilizing Seasonal NWP Model Output

The performance of the GAM-PC1+2 structural demand model (5.14) described in the previous chapter is conditioned upon knowing future temperature, or more specifically, future principal components of temperature fields. Unfortunately, we do not have access to these at the time of forecast issuance. Thus, in this chapter we will focus on several methods that enable us to forecast future temperature principal components (the alternative would be to rely on estimates from e.g. climatology). These forecasts will in turn be used as inputs in our final demand forecast model (Chapter 7). Our attention will be centered on probabilistic temperature forecasts utilizing seasonal NWP model output (described in Chapter 4) as a way of estimating temperature principal component quantiles.

The first three sections will concern the use of NWP forecast data to estimate quantiles of the predictive distribution of temperature PCs at the 6-hour interval level according to models described in Section 3.4. We will first (Section 6.1) describe the evaluation set-up and introduce the climatology baseline. We will then (Section 6.2) look at the performance of the direct Weighted Quantile Estimation (WQE) model. It exhibits considerable skill up until 15 days from forecast issuance, after which model performance is only marginally better than the climatology baseline. In Section 6.3 we investigate the performance of models which, in addition to the NWP input, include weights, lagged forecasts, and time covariates in a Quantile Regression (QR) framework. We find that the additional information offer only minor predictive improvement.

Since NWPs are issued once monthly there is a period of ca. 10-15 days virtually without skill at the end of each month before a new forecast is issued. In Section 6.4 we will attempt to remedy this weakness by employing a re-weighting scheme (described in Section 3.5). This is based on observed temperature principal components over a set interval (1-3 days), and re-estimates the forecast covariates to emphasize the forecast members with the best performance. We show that this re-weighting procedure enables a short-term skill increase where the performance at days 1-5 after re-weighting reverts to the same level as it was at days 1-5 after forecast issuance. Performance, e.g. in the middle of the month, can be nearly recovered to the level of the start of the month. This

re-weighting can regain skill any number of times and at any arbitrary time point after forecast issuance.

As noted in Section 3.6, even for models with considerable skill it might not be possible to observe model skill at a high time resolution (in our case 6-hourly) at longer time-horizons. But if we aggregate forecast predictions over larger periods we can observe skill even at longer horizons. In Section 6.5 this is done both naively and by using Gaussian Copula (GS) estimation. We find that by increasing the length of the aggregation interval we can substantially prolong the period for which we observe model skill.

In the following, attention will be kept on probabilistic temperature forecasts. The final step of utilizing these forecasts as input in a demand model will be considered in Chapter 7.

6.1 Temperature PC Quantile Estimation

As described in Chapter 4 each NWP ensemble member is on the form of the ERA5 temperature data. At each time point we have between 50-200 ensemble members, each yielding predictions for every point of the temperature grid at 6-hour intervals. This 1) enables a direct comparison between the forecasted and observed weather, and 2) allows for the use of the same principal component dimensionality reduction as employed in Problem 1. As discussed in Section 5.6, in addition to showing promising results, the main advantage of the GAM-PC1+2 model over the other models considered is that it simplifies the temperature prediction task. Instead of estimating all (or in the case of Lasso over 100) temperature grid points, utilizing principal components reduces the problem down to two prediction tasks: Finding C_t^1 and C_t^2. Because of this advantage with respect to parsimony, we will only consider models in principal component space. For ease of presentation we focus solely on the first PC; other PCs may be forecasted in a similar manner.

Following a similar evaluation set-up as for the demand problem we use a rolling prequential cross-validation (PCV) testing set-up also for the temperature PC forecast. For this problem, training is performed on data going back to 1993. We test on 12 monthly forecasts over 16 years (and an additional month) yielding 193 forecast periods from January 2007 to January 2023. To assess model performance, we will primarily look at lead times up to 60 days, but will also consider shorter segments of this interval as well as lead times up to 125 days for some applications. The central metrics employed for model comparison are the pinball loss and the skill score based on pinball loss as described in Section 3.8. When assessing performance across a quantile range we will utilize the continuous ranked probability score CRPS. Mostly, however, we will focus on the 0.9-quantile. Specifically this is the quantile that marks the point for which we expect 90% of principal component realizations will fall below. This in turn is associated with the coldest part of the temperature distribution at any given time (Section 4.2). We expect analyses for other quantiles to yield roughly similar results.

Each month, each NWP ensemble member, m, issues a forecast matrix over 500 lead times k (i.e. 125 days), covering $p = 462$ grid points. At target time t, for a specific lead time k, we denote this as $N^m_{t|t-k}$ (see Section 3.4). Since the NWP forecasts are overlapping, each temperature grid observation, z_t, is

targeted by 4 or 5 forecasts issued at different times, depending on the day of the month of the target. This means we have $N^m_{t|t-k}$ for $k \in \{k_1, ..., k_5\}$, where the interval between k's are roughly 30 days (depending on month length). To conceptually separate between these we call the forecasts issued closest in time to the actual realization by 'NWP1', the second closest 'NWP2', etc.. When we do not distinguish between issuance months in this manner, we use just 'NWP'.

Climatology Temperature Baseline

As in the case of energy demand, temperature exhibits regular temporal cycles (specifically daily and annual). The natural comparison point for our models is again the climatology, this time applied over quantiles. Utilizing the observed first principal component, C^1_{hdj}, at hour h, day of year d, and year j, the Climatology Quantile Temperature Baseline estimate at year n and quantile α is given as:

$$\hat{C}^{1,\alpha}_{hdn} = W^\alpha(C^1_{hd1}, ..., C^1_{hd(n-1)}), \tag{6.1}$$

where $W^\alpha(\cdot)$ is the weighted quantile estimation function described in Section 3.4, taken over all previous years $j = \{1, ..., (n-1)\}$.

6.2 Weighted Quantile Estimation of NWP Forecasts

The NWP Weighted Quantile Estimation (NWP-WQE) model utilizes a quantile estimate of the NWP first PC as the forecast of the first temperature PC. For target time t, at a specific lead time k, we employ $W^\alpha(\cdot)$, the WQE-function, over the set of NWP principal components, $\mathcal{C}^{1,1:M}_{t|t-k}$, to obtain the quantile estimate:

$$\hat{q}^\alpha_{t|t-k} = W^\alpha(\mathcal{C}^{1,1:M}_{t|t-k}). \tag{6.2}$$

The NWP-WQE model then utilizes the estimate, $\hat{q}^\alpha_{t|t-k}$, directly as an input, $q^\alpha_{t|t-k}$, in a forecast of the first principal component at quantile α:

$$C^{1,\alpha}_t = q^\alpha_{t|t-k}. \tag{6.3}$$

Comparing the results with climatology (Table 6.1), we see that across all quantiles the pinball loss averaged over the first 60 days after forecast issuance is considerably lower for the NWP-WQE model. The variation in the pinball scores between the quantiles (relatively symmetric around the median) stems from the nature of the pinball loss function, as described in Section 3.8.

Overall for the 60-day period, the model skill is consistent across quantiles. It lies between 0.110 for the 0.1- and 0.137 for the 0.8-quantile, where the top half of quantiles have a slightly higher skill level (Table 6.1). Segmenting the skill score of the NWP-WQE model relative to climatology into 20-day periods we see that the model performance across quantiles are similar within each interval: We observe high skill for the first 20-day period, some skill for the next 20 days and for days 41 to 60 the skill drops slightly except for the highest quantiles where the skill counter-intuitively increases slightly compared to the 21:40 period. Notice that we do not observe negative skill (indicating a forecast performance worse than climatology) for any period or quantile. During the

	Pinball Loss (Days 1:60)		Skill Score Days:			
Quantile	**Clim.**	**NWP-WQE**	**1:60**	**1:20**	**21:40**	**41:60**
0.1	8.452	**7.527**	0.110	0.261	<u>0.052</u>	0.023
0.2	13.761	**12.117**	0.119	0.301	0.045	0.017
0.3	17.448	**15.181**	0.130	0.325	0.045	0.026
0.4	19.629	**17.047**	0.132	0.336	0.041	0.024
0.5	20.548	**17.775**	0.135	<u>0.344</u>	0.039	0.028
0.6	20.092	**17.386**	0.135	0.341	0.036	0.034
0.7	18.295	**15.803**	0.136	0.337	0.035	0.041
0.8	14.899	**12.853**	<u>0.137</u>	0.334	0.035	0.041
0.9	9.391	**8.172**	0.130	0.323	0.025	<u>0.044</u>

Table 6.1: Pinball Loss and Skill Score for quantiles 0.1-0.9 averaged over different lead time days for Climatology (6.1) and NWP-WQE-model (6.3). Bold indicates best pinball loss at each quantile, underscore indicates quantile with best skill score for each period.

first 20-day period the best performing quantiles are found at the center of the distribution mass with a skill level around 0.33-0.34, but the 0.9-quantile is very nearly just as good. The 0.1-quantile has a skill score a step under the other quantiles (at 0.261) indicating that the model shows least improvement at the level where the principal component is the lowest, i.e. where temperature is the highest.

In Figure 6.1 we look in more detail at the skill level by lead time for the 0.9-quantile, now for lead times up to 125 days. We see that the NWP-WQE model shows very high skill for the first days (at around 0.7). It then swiftly drops and reaches 0.1 for the first time after 15 days, and hitting 0 for the first time after 23 days. After this, apart from a couple of exceptions, the skill fluctuates within a band of 0 ± 0.1 for the rest of the lead time duration. The other quantiles (not shown) show very similar patterns in both pinball loss and skill score differing mostly at the level at which the pinball loss settles. The pinball loss of the climatology model over lead time seems settled at an equilibrium mean of 9.36. In a linear model fit with climatology pinball loss as outcome and lead time as predictor ($y_i = \beta_0 + \beta_1 x_i$) we obtained the coefficient $\beta_1 = -0.0003$, i.e. we observe no linear trend. The fluctuations exhibit a prominent monthly pattern. The auto-correlation function (ACF) of the pinball loss for climatology (Figure 6.2) corroborates this. At lags 116-128 corresponding to 27-32 lead time days the ACF-values are all above 0.15 topping out at 0.368 for lag 124. This is likely due to it estimating (with the same model) the same observation 4-5 times with monthly intervals (which because of uneven month lengths do not line up perfectly between estimated dates). After the initial 'skilled' 23-day period the trajectories of the pinball loss for both models follow each other fairly closely (Figure 6.1). In fact, it seems that the performance of the NWP-WQE model to a large extent reverts to climatology. Apart from short bursts of skill at seemingly intermittent points there are also a couple of periods where the NWP model improves upon climatology even if the latter is also performing well (notice the dips at around lead times 45, 75, and 105 where NWP-WQE shows improvement). These dip periods look similar to the period between

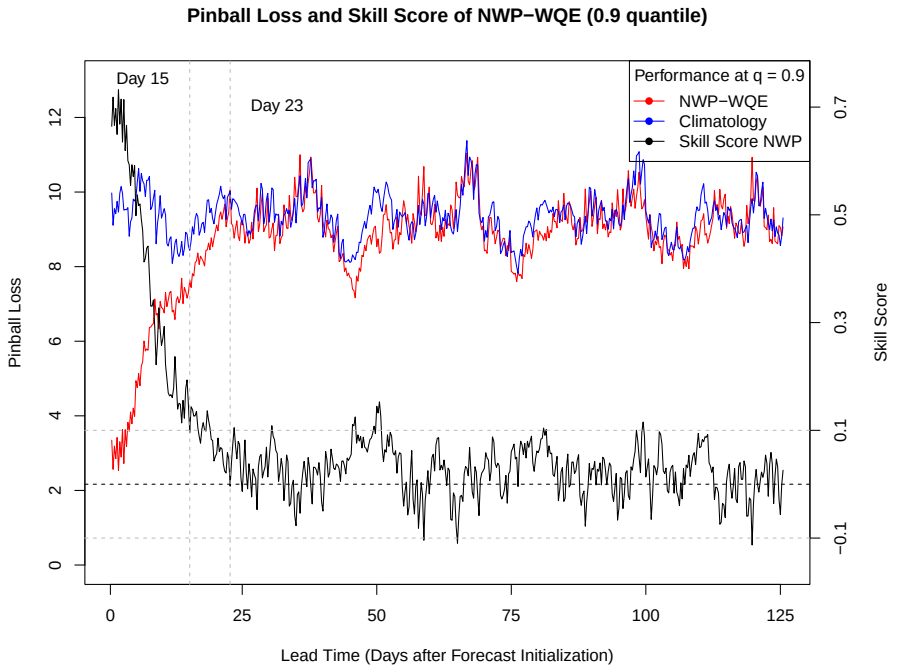


Figure 6.1: Pinball loss and skill score of NWP-WQE model (6.2) with reference to Climatology (6.1) at the 0.9-quantile. The skill of the NWP-WQE model (black line, with axis-values to the right), is considerable up until day 15 when it dips under 0.1. This reflects the good performance of the model (red) in terms of pinball loss during the same period.

days 15 and 23 which suggest that one could view the point at which the NWP model reverts to climatology as starting at 15 days out, not 23.

For the purpose of comparison we can also draw the estimated quantiles over an example period (winter 2013/14) for both models (Figure 6.3), where we only look at the 6:00 temperatures for a clearer overview.[1] The observed principal component temperatures (black) show a characteristic pattern for the winter months with abrupt jumps or drops which at short intervals cover the entire range. We see that the NWP-WQE model (red lines) draws a tighter distribution around the center of the probability distribution than the climatology (blue shading). Over the test period for all lead times the average distance between the 0.1- and the 0.9-quantiles is 126.28 for the climatology model, and 116.32 for the NWP-WQE model. Between quantiles 0.3 and 0.7 the distances are 53.84 and 49.96, respectively. Restricting our attention to the period after 25 days we observe the spans 126.18 and 121.76 for the 0.1-0.9 and 53.90 and 52.52 for 0.3-0.7. Thus the NWP-WQE distribution is tighter, not only for the first days when performance is the best, but also after model skill has subsided.

Additionally, we can readily observe that the 0.1-quantile of the NWP-WQE model is set at a substantially lower level than for the climatology. This is a

[1] The same plot at hours 12:00, 18:00, and 24:00 is slightly shifted up or down. The overall trends are made clearer when focusing on just 6:00.

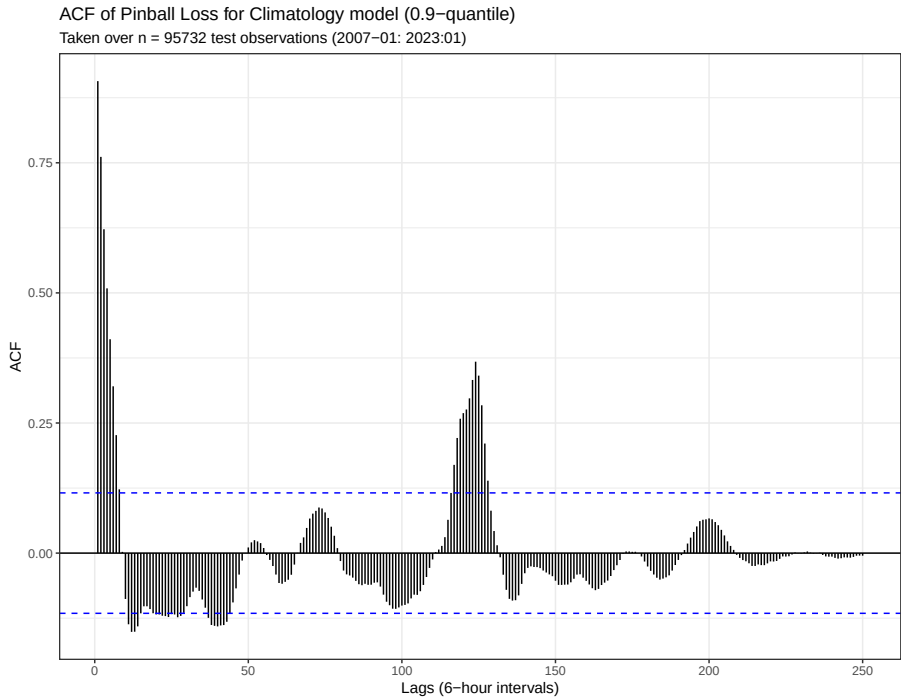


Figure 6.2: ACF of pinball loss for Climatology (6.1) by lead time at 0.9-quantile. The pinball loss exhibits clear patterns in the correlation structure. Especially notable is the monthly correlation around lags 116-128 (days 27-32).

fairly consistent feature across forecast months. In fact all NWP quantiles are on average shifted down compared to climatology. For each quantile we subtract the predicted quantile value of the NWP-WQE model from the climatology. Averaging these differences between models we obtain for quantiles 0.1-0.9: 3.49, 4.00, 4.65, 5.10, 5.84, 7.36, 8.72, 10.70, 13.87. The downward shift holds across all quantiles and is more pronounced the higher the quantile. The forecast based model consistently presents a warmer distribution than the climatology indicates.

Overall it is clear that utilizing the NWP data for the purpose of probabilistic forecasting vastly outperforms climatology. We have successfully improved predictive accuracy at all quantile levels and we have tightened the distribution bands around the expected value most pertinently for the lead times closest to forecast issuance, but also for later periods.

Some questions remain, however: Can we find a better performing model utilizing NWP data? And is it possible to extend skill for an extended period of time after the initial 15-23 days? In the following the focus, unless otherwise stated, will lie on the 0.9-quantile. This is not only done for ease of presentation, but also because it is of application interest.

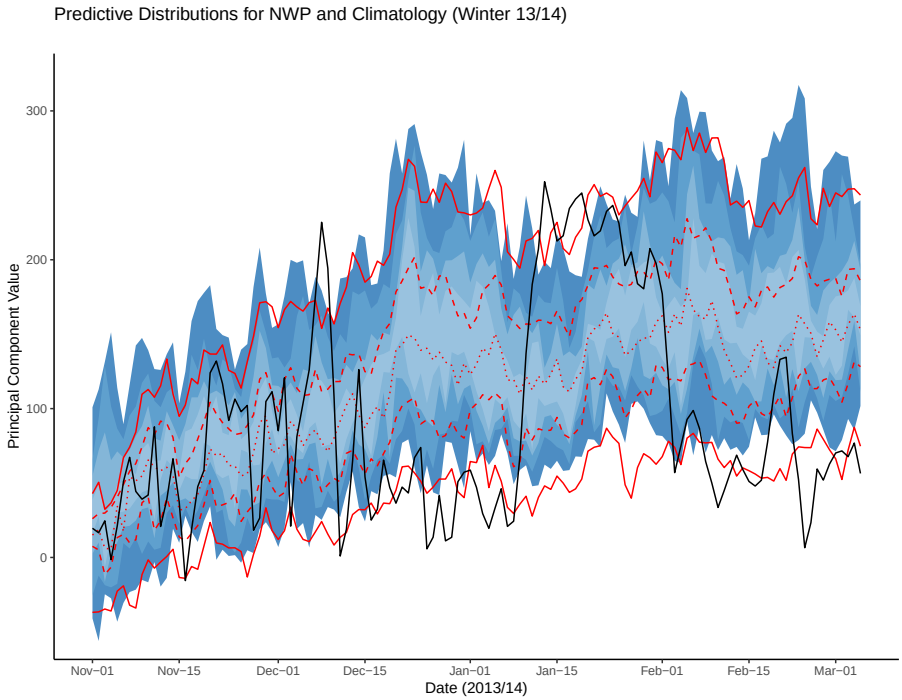


Figure 6.3: Probabilistic PC forecast (at 6:00) for Winter 2013/14, comparing Climatology (blue shading) with NWP-WQE (red lines) over the quantile range of the first principal component. The black line is the observed first PC over the period. The NWP-WQE model draws a tighter distribution that is slightly downward-shifted compared to the Climatology.

6.3 Quantile Regression Models

Quantile Regression (QR) models offer more flexible modelling options than the WQE-model through coefficients that adjust the impact of the NWP input (Section 3.4). By introducing coefficients we want to 1) find out whether we can obtain more accurate forecasts, and 2) investigate properties of the NWP input. This section is split into three parts. We will first look at QR-models with just NWP input. We will then investigate whether time covariates improve model performance. Lastly, we will assess the performance of the best QR-models compared to the basic WQE-model. We find that the QR-models offer very little improvement compared to the WQE-model.

QR-Models with NWP Input

We start this section by looking at the simplest NWP-QR model. This uses the WQE output, $q^{\alpha}_{t|t-k}$, as a single predictor and introduces the quantile specific weight, β^{α}_1, but has no intercept:

$$C^{1,\alpha}_t = \beta^{\alpha}_1 q^{\alpha}_{t|t-k} \tag{6.4}$$

In the last section we saw that the performance of the NWP-WQE model is dependent on lead time. Our first batch of models will look at several

	Pinball Loss ($\alpha = 0.9$) for Days:				
Model	**1:20**	**21:40**	**41:60**	**1:60**	**61:125**
1 day	6.367	9.422	8.870	8.217	9.166
2 days	6.358	9.413	8.867	8.211	9.159
4 days	6.354	9.417	8.862	8.209	9.147
8 days	6.337	9.404	8.841	8.192	9.134
16 days	6.329	9.397	8.838	8.186	9.127
32 days	**6.327**	**9.383**	8.831	**8.178**	9.127
64 days	6.345	9.389	**8.809**	8.179	9.125
125 days	6.356	9.398	8.818	8.189	**9.115**
NWP-WQE	6.329	9.382	8.803	8.172	9.140
Climatology	9.346	9.624	9.204	9.361	9.330

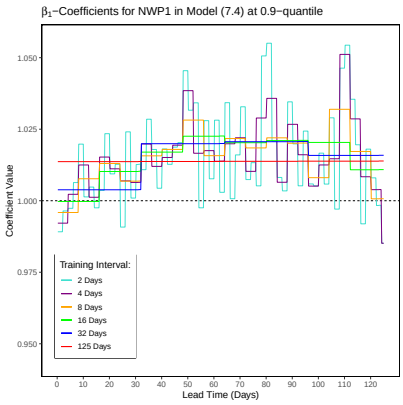
Table 6.2: Mean test pinball loss over lead time intervals at the 0.9-quantile for QR-models of the form (6.4) for different training intervals. The best training period among QR-models performs at the same level as the NWP-WQE model. Bold indicates best score among training interval models for each lead time segment.

implementations of model (6.4) where we vary the length of the training periods to cover different lead time intervals. In this manner we can investigate whether different lead times might exhibit differing relations between the forecast ensemble quantile and the temperature. Thus, the estimated coefficient $\hat{\beta}_{1,\tau}$, which is specific for lead time interval τ, is found for different models by restricting the training period to time points $t \in \tau$:

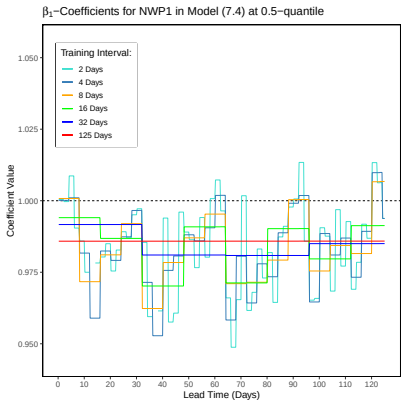
$$\hat{\beta}^{\alpha}_{1,\tau} = \underset{\beta^{\alpha}_1 \in \mathbb{R}}{\operatorname{argmin}} \sum_{t \in \tau} \rho_{\alpha}(C^1_t - \beta^{\alpha}_1 q^{\alpha}_t), \tag{6.5}$$

where $\rho_{\alpha}(\cdot)$ is the pinball loss function for quantile α, and C^1_t is the observed first temperature principal component at time t. We look exclusively at the 0.9-quantile for segmented lead time training intervals where τ has the length of 1, 2, 4, 8, 16, 32, 64 or 125 lead time days. An alternative to this segmented implementation would be a set-up with rolling intervals.

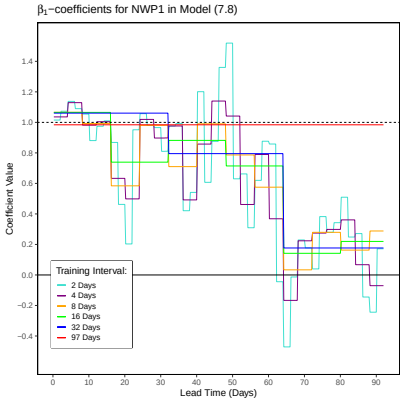
Overall, the results for the best training intervals (Table 6.2) only show an incremental improvement over the NWP-WQE model at some lead time intervals (days 1:20 and 61:125). Whether we train on smaller or larger lead time intervals is also of low importance, but the longest intervals perform slightly better. The QR-models and the NWP-WQE show the same trends across lead time intervals: The best predictive performance comes from the first period after forecast issuance. The best model for the first two periods uses a 32-day training interval, the best for days 41-60 used 64 and the best for the remaining lead time was the model trained on 125 days. Since we generally care more about performance before 60 days out, the 32-day model is preferable, noticing that its performance was fairly good over the last two lead time intervals as well. Model performance for this model is virtually identical to the NWP-WQE model over the 60-day period. Based on these models, little is gained predictively by adding an adjustment in the form of the β-coefficient to the NWP input.



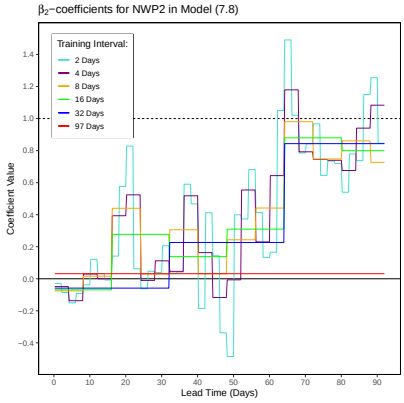
(a) β_1-coefficients (6.4), $\alpha = 0.9$.



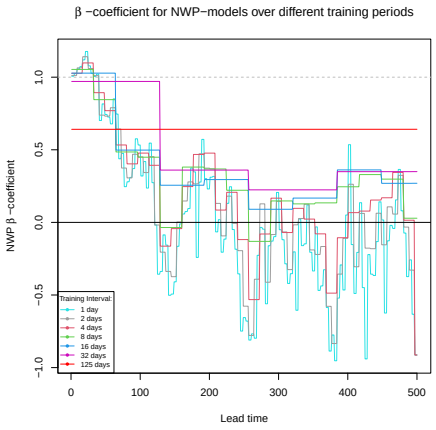
(b) β_1-coefficients (6.4), $\alpha = 0.5$.



(c) β_1-coefficients (6.7).



(d) β_2-coefficients (6.7).



(e) β_1-coefficients for best QR-model (6.10).

Figure 6.4: Mean β-coefficients for models (6.4), (6.7) and (6.10). While model (6.4) adjusts the WQE estimate ($q^{\alpha}_{t|t-k}$) slightly upward, the introduction of NWP2 (Figures 6.4c and 6.4d), and other predictors (Figure 6.4e) decreases the influence of NWP1. Note that the y-axis range varies between plots.

To see the impact of the NWP-QR quantile predictor at different lead times we can look at how the model coefficients (averaged over all forecast test periods) for different training periods change over lead time (Figure 6.4a).[2] Overall the coefficients concentrate around a level slightly above 1, indicating that only a slight upward adjustment of the input variable is performed. The predictive distribution of the NWP-WQE model was set substantially lower than climatology, and the QR-model slightly adjusts this back. The models using the smallest interval are, as can be expected, the most variable. The adjustment is smaller at the beginning reflecting the higher predictive quality of the input at this stage. This adjustment might be an artifact of the post-processing of the NWP ensemble members forcing the mean and variance to be equal to those for the climatology (Section 4.3). At the extreme quantiles this process might not be spread out enough. We also check the median to see if the same overdispersion effect is present there (Figure 6.4b). We find that the coefficients for the median ($\alpha = 0.5$) is on the same order as at the 0.9-quantile, but now the adjustment is made slightly downwards. Because the magnitude of these adjustments are so small and the difference in predictive performance is negligible, we consider these adjustments as being of little importance.

Before moving on to include time-covariates in the modelling we also fitted models that included NWP2. This predictor consisting of forecasts that are issued at least one month from the target date. We therefore introduce the restriction $k > c$, where the lead time k has to be above $c \in \{28, 29, 30, 31\} \cdot 4$, a month dependent cutoff value. The model including just NWP2 has the form:

$$C_t^{1,\alpha} = \beta_1^\alpha q_{t|t-k2}^\alpha, \tag{6.6}$$

where k_2 satisfies $k_2 > c$ at each month. As we have seen at this level of lead time, the NWP is scarcely better than climatology. The rationale for including NWP2 as a predictor is twofold. Firstly, to see if it might improve predictive performance through creating a model averaging effect. And second, to look at the impact this addition has on the weights of NWP1. To assess this we introduce the model:

$$C_t^{1,\alpha} = \beta_1 q_{t|t-k_1}^\alpha + \beta_2 q_{t|t-k_2}^\alpha, \tag{6.7}$$

where k_1 has no restrictions. Including NWP2 naturally means we restrict the lead times by roughly a month. This means we cannot use the 125-day set-up, the longest will be a 97-day one (the maximum lead time for NWP2). For training intervals of 32, 64 and 97 lead time days we tested linear and spline ($df = 3$) versions of models (6.4), (6.6), and (6.7) with and without intercepts. The results (Table 6.3)[3] show that the spline models with an intercept do consistently better, but they offer only small improvements over the linear model versions. Including an intercept term also improves prediction for all models. For linear models the effect of the intercept is small, but for spline

[2]Whether we choose an intercept in these models or not has very little impact on the model coefficients, as since the relation between predictor and outcome is so close to 1-to-1, when the predictor is 0 the expected value (the intercept mean) is also going to be close to zero. Here we show the model without intercept. By omitting the intercept we allow for the interpretation of the coefficient as a direct weight adjustment.

[3]Notice that the coefficients diverge slightly from previously reported here, because the inclusion of NWP2 lead to very slight differences in the formation of the training sets.

	Pinball Loss (0.9) for Training Periods:					
	No Intercept			Intercept		
Model	**32**	**64**	**97**	**32**	**64**	**97**
NWP1	8.180	8.180	8.187	**8.166**	8.170	8.176
NWP2	9.126	9.126	9.124	9.104	9.103	**9.103**
NWP1+NWP2	8.156	8.182	8.192	**8.150**	8.171	8.180
s(NWP1)	9.156	8.689	8.678	**8.155**	8.166	8.175
s(NWP2)	10.42	10.390	10.385	9.083	9.080	**9.079**
s(NWP1)+s(NWP2)	8.880	8.352	8.342	**8.147**	8.163	8.172

Table 6.3: Mean test pinball loss for linear and spline versions of QR-models (6.4), (6.6) and (6.7) with and without intercepts for three different training intervals at lead time days 1:60. Bold indicates best parametrization.

models there is a considerable difference, especially for shorter training intervals. With regard to the NWP2 predictor we see that the spline model does slightly better than the linear one. And the results are roughly similar to the NWP1 model if we push lead times back a month and look at the interval between 30 and 90 days (this yields a pinball loss of 9.054). The addition of NWP2 to NWP1 represented a slight improvement for the shortest training intervals (32 days), but not for the longer ones. The best model among these is the spline model with both NWP predictors, which has a very modest skill (0.006) with reference to the NWP-WQE model.

The linear model (without intercept) which combines the NWP1 and NWP2 predictors (6.7) can be interpreted as a weighted mean of forecast outputs issued one month apart. By examining it we can uncover the degree to which the introduction of NWP2 acts as an adjusting factor on the β_1-coefficient of NWP1. Figure 6.4c shows the average β_1- coefficients over the test periods from model (6.7) for six training intervals. For the model (shown in red) trained on all available lead times (97 days) the β_1-coefficient is 0.986. This is slightly down-shifted compared to the model (6.4) with only NWP1 ($\beta_1 = 1.014$) shown in Figure 6.4a. When using this training period interval the impact of the NWP2 predictor (shown in red in Figure 6.4d) is fairly small ($\beta_2 = 0.032$), and did not improve performance (Table 6.3). For the other training intervals the general trend is that the higher the lead time used to train on, the lower the weight of the β_1-coefficient, and thus the lower the reliance on NWP1. For lead times under 16 days, model (6.7) weights the NWP1-predictor close to 1 while the NWP2-coefficients lie slightly below 0. This reflects the good performance of the NWP forecasts immediately after they are issued. Between 16 and 64 days, model (6.7) increases the size of the weights for NWP2, while the NWP1 weights are correspondingly decreased. After 64 days the predictive quality of NWP1 has degraded to a point where the model (6.7) consistently weighs the NWP2-predictor higher. We should not interpret this to mean that older forecasts are better at these lead times. Rather it corroborates our previous finding that as the lead time increases the performance of the NWP forecasts reverts to climatology.

These results illustrate two aspects of the NWP forecast data. Firstly, they

	Parametrization:			
Covariate	**Cont.**	**Factor**	**Sinusoidal**	**Spline**
Intercept	**22.22**	–	–	–
Year	**22.44**	23.88	–	23.23
Season	18.37	**13.48**	14.04	–
Week	21.36	10.16	**9.96**	9.99
Month	21.33	**10.48**	10.60	10.48
Hour	22.22	**22.12**	22.22	22.12

Table 6.4: Mean test pinball loss for univariate QR-models of the form (6.8), over lead time days 1:60, at the 0.9-quantile. Bold indicates best parametrization, underlined indicates best overall.

demonstrate how little the model averaging effect of adding older forecasts has on prediction accuracy. And secondly, they show how fast the NWP forecast degrade in forecast quality in principal component space at the hourly level. In Section 6.5 we will look at methods that enable us to observe skill in the NWP-based models at longer lead times by focusing on higher aggregate levels.

QR-models with Time Covariates

In this section we will investigate how much improvement can be gained by adding time covariates to the QR-model set-up. We will report results only from the longest training interval (97 lead time days), both for convenience, but also because models including interactions encountered difficulties when using shorter lead time training intervals. We start by looking at parametrizations of single time covariates, now with the aim of forecasting temperature principal components. These models are on the form:

$$C_t^{1,\alpha} = \beta_0^\alpha + f(x_t), \tag{6.8}$$

where x_t is the time covariate and $f(x_t)$ is a parametrization described in eq. (5.4). The factor parametrization is the best for 'hour', 'month' and 'season' (Table 6.4). The continuous version is the best for 'year', while the sinusoidal was the best for 'week'. 'Week', 'month' and 'season' all seem to be good predictors, as they represent clear improvements on the intercept model.

Having found good parametrizations for each of the time covariates we combined them and ran four models on the form:

$$C_t^{1,\alpha} = \beta_0^\alpha + f(x_t) + g(h_t, x_t^{(a)}), \tag{6.9}$$

where $f(x_t)$ is a combination model which includes the best parametrization of each time covariate found above. The four models differ only in $g(\cdot)$, which is a factor interaction between hour, h_t, and an annual cycle term, $x_t^{(a)}$. As with the demand problem in Chapter 5, we are only exploring a limited part of the model space. The results for models of the form (6.9) (left-most column of Table 6.5) show that all versions of (6.9) improve upon climatology. The best interaction term was between 'hour' and 'season'. This model has a skill relative to climatology of 0.098. Thus by utilizing only time information we can

improve upon the climatology model quite substantially without any additional temperature data.

The last batch of models in this section includes both time and temperature data, we refer to these as QR-combined models. We tested 16 such models of the form:

$$C_t^{1,\alpha} = \beta_0^\alpha + f(x_t) + g(h_t, x_t^{(a)}) + h(q^\alpha_{t|t-k_1}, q^\alpha_{t|t-k_2}) \tag{6.10}$$

Here, building upon the set-up of (6.9), we add NWP1 and NWP2 through the function $h(\cdot)$. For each time covariate set-up we looked at 3 versions of $h(\cdot)$:

- A linear model including just NWP1, i.e. with $h(\cdot) = q^\alpha_{t|t-k_1}$.
- A linear model including both NWP terms with $h(\cdot) = q^\alpha_{t|t-k_1} + q^\alpha_{t|t-k_2}$.
- A spline version of the latter where $h(\cdot) = s(q^\alpha_{t|t-k_1}) + s(q^\alpha_{t|t-k_2})$.

The predictive performance of these models are shown in Table 6.5. For all versions of $h(\cdot)$, the best interaction term was again 'hour' and 'season'. Adding NWP2 leads to slightly improved predictive performance over all interaction versions. The linear and spline implementations have very similar performance. The best among these models makes use of both NWP predictors, time covariates, as well as an interaction 'hour' and 'season', and had a pinball loss of 8.075. It does, however, not represent a big improvement, as it only has a modest skill of 0.012 with reference to the NWP-WQE model. Compared to climatology it has a skill score of 0.137, versus the NWP-WQE which had 0.130. In contrast with the demand problem, very little is gained by using time covariates. The temporal information these variables represent seem already to be incorporated in the NWP ensemble members. When time covariates are included, adding NWP2 to the model gives a slight performance increase.

		Linear		Spline
Set-up	**Time-vars**	**NWP1**	**NWP1+2**	**NWP1+2**
Combination	9.050	8.168	8.100	**8.098**
Comb+h:w	9.203	8.651	8.561	**8.584**
Comb+h:m	8.907	8.265	**8.163**	8.170
Comb+h:s	8.920	8.164	**<u>8.075</u>**	8.082
Intercept	–	**8.176**	8.195	8.180

Table 6.5: Mean pinball loss, over lead time days 1:60, for QR-models with just time covariates on the form (6.9), and QR-models with both time covariates and NWP input on the form (6.10). Bold indicates best parametrization, underlined indicates best overall.

Even though very little improvement in predictive performance was gained by adding time covariates these models change the impact of the NWP predictor. We can illustrate this (Figure 6.4e) by looking at an average of β_1-coefficients for the QR-combined model (6.10) with $g(\cdot) = 0$ (without interaction) and $h(\cdot) = q^\alpha_{t|t-k_1}$ (only NWP1). We choose this model for illustration purposes because models with interactions had problems running for shorter training

intervals. We can see that for the longest training interval the coefficient for NWP1 is around 0.6. This means that considerably less information from NWP1 is used for this model in forming the predictions. For shorter training intervals we see that the further from forecast issuance the less weight is given to NWP1. The model is more and more relying on the time covariates. This is, as we have seen, because at larger lead times, at the hourly level, the NWP predictor reverts to climatology.

Model Assessment

That the models (both WQE and QR) based on NWP ensemble forecasts perform better than climatology is perhaps not surprising given that these forecasts are targeting temperature. We have now shown that the NWP forecasts are directed at temperature also at the quantile level in principal component space for the first principal component. We have also shown that neither the introduction of coefficient weights, of lagged forecasts (NWP2), or of time covariates have improved predictive performance more than slightly.

To check if these QR-model results represent an actual improvement upon the NWP-WQE-model we ran three permutation tests (see note in Section 5.5). Using 10000 permutations the test results show clearly that both the WQE and the best QR-model outperforms climatology. When compared against each other we observe a p-value of 0.0564. This is close to, but above, the standard significance threshold at 0.05. The results of the QR-model are not significantly different from the WQE-model. As mentioned in the previous chapter, we do not have exchangeability, so the tests are meant more as a heuristic device than a final say.

	Permutation Tests			
	Climatology		WQE	
Model	**Obs. Skill**	**p-value**	**Obs. Skill**	**p-value**
WQE	0.131	0.0000	–	–
QR-comb	0.140	0.0000	0.011	0.0564

Table 6.6: Permutation test results comparing WQE (6.3) and QR-comb (6.10) with Climatology (6.1) and each other over lead time days 1:60.

Even if the tests indicate that the QR-model tests are not statistically significant compared to the WQE model, it is clear that the NWP-based models do represent an improvement with reference to climatology. This improvement is subject to considerable variability in predictive accuracy by target month. Winter months are the hardest to predict; summer months being the easiest. Figure 6.5 illustrates this monthly variation by showing the distribution of the pinball loss across months for both NWP-WQE and climatology (the performance of NWP-QR is almost identical). Note that the x-axis is cut for presentation as the upper tails of these empirical distributions are very long and thin, with outlier values at a maximum of 160.49. In all months we can see a red climatology shadow to the right of the NWP-distributions indicating the upward shift (and worse performances) of the climatology model.

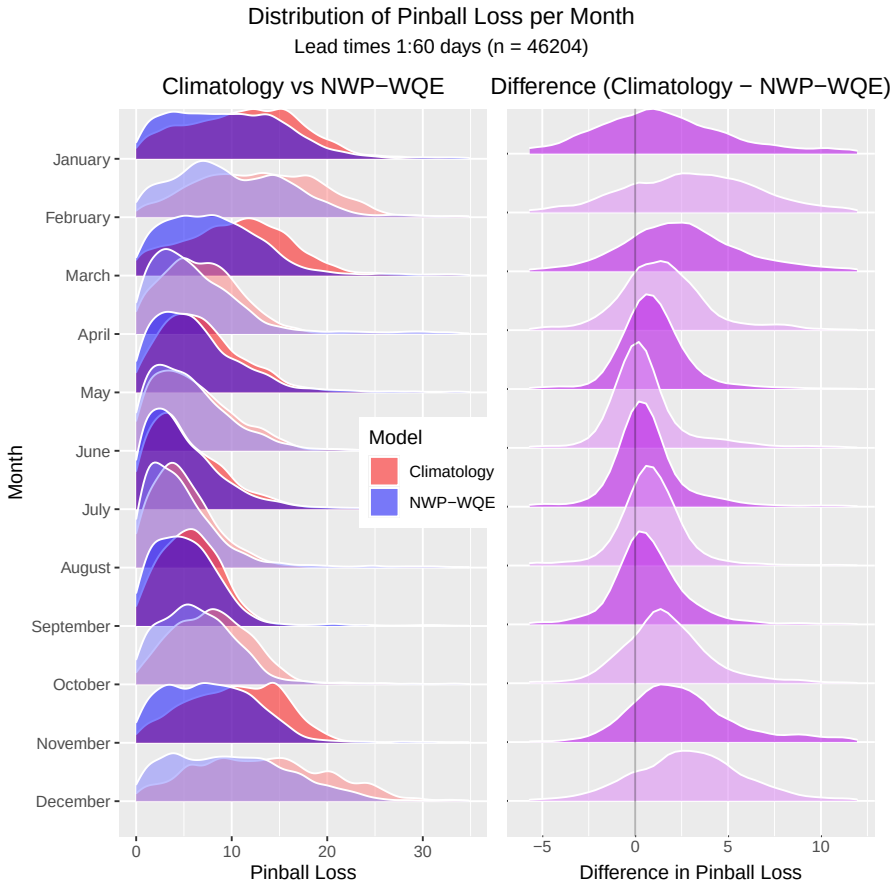


Figure 6.5: Pinball loss distribution by month.

The left-most plot shows the distribution of the difference in performance between the two models for each predictive observation. Positive values mean the climatology had a bigger loss. Even though the shape and especially the spread of the difference distributions vary by month, the share of predictions that are better is roughly the same (ca. between $25-35\%$). Notice especially the rarity with which the NWP model is more than 5 points worse than climatology compared to the other way around.

Another way of quantifying the improvement is to look at the fraction of predictions with a lower loss than climatology (Table 6.7). We see that overall for the days 1:60, 70.2% of predictions from the NWP-WQE model had a lower loss than for later periods. The best QR-combination model does slightly better at 71.7%, where improvement compared to NWP-WQE comes after the 10 first days.

As a final assessment of the performance of the WQE and the best QR-model we also compute the continuous ranked probability score (CRPS), which we approximate by averaging over the pinball loss at each quantile (Section 3.8). For lead time days 1:60, over quantiles 0.1-0.9, the climatology CRPS is 15.835 while the NWP-WQE obtains 13.762, which yields a CRPS skill score of 0.131.

	Improvement Ratio Days:				
Model	**1:10**	**11:20**	**21:30**	**31:60**	**1:60**
NWP-WQE	**0.833**	0.669	0.665	0.681	0.702
QR-comb	**0.833**	**0.691**	**0.680**	**0.700**	**0.717**

Table 6.7: Pinball loss improvement fraction for the NWP-WQE model (6.2), and the best QR-model (6.10) at 0.9-quantile. For each lead time interval it shows the ratio of predictions targets for which each model improves upon climatology.

For the best QR-model we observe a slight improvement across all quantiles for a CRPS score of 13.663, and a CRPS skill score of 0.137. With respect to NWP-WQE, the CRPS skill score for the best QR-model is 0.007.

6.4 Re-weighted Quantile Estimation

The NWP-based models for temperature forecasting investigated in the previous sections have shown a marked improvement on climatology. But at the 6-hourly level the period of substantial model skill after forecast issuance only lasts for 15 days. Since seasonal NWP forecasts are issued on the first of every month, there is a period of 10-15 days at the end of each month where these models show little or no skill. We will now look at a method for extending model skill past this initial period. This will be done by re-weighting the NWP principal component inputs from each ensemble member based on the most recent temperature observations made available after forecast issuance. By weighting the NWP ensemble members according to their recent performance, the re-issued forecasts will to a larger extent be based on the members that recently performed the best (Section 3.4).

The re-weighting procedure is applied at each target time t for a specific lead time k, to the first principal component of the temperature grid for each NWP ensemble member, m. Producing the re-weighted quantile estimate, $\tilde{q}^{\alpha}_{t|t-k}$ involves finding the multiplicative weight, $\widetilde{w}^{m}_{R}$, which adjusts the impact of the principal component of each NWP member:

$$\tilde{q}^{\alpha}_{t|t-k} = \sum_{m=1}^{M} \widetilde{w}^{m}_{R} C^{m}_{t|t-k}. \tag{6.11}$$

As before, k marks the lead time interval from forecast issuance at the 1st of the month, while R marks the time of forecast re-issuance. The procedure for obtaining the weights are described in Section 3.4.

To test the predictive impact of the re-weighting procedure we ran a similar rolling-cross validation set-up as described above (Section 6.1). We utilize a 3-month (97 days) lead time training interval over the same training periods as for the previous NWP models. Like previously, the focus is on forecasting the temperature principal component at the 0.9-quantile. The re-weighting is based on observations from one of three different re-weighting intervals. We either use: i) 4 observations of PC temperature from the 15th of every month; ii) 8 observations from the 14-15th of every month; or iii) 12 observations from the

		Pinball Loss Days after Re-issuance:		
Model	**Re-weight period**	**1:5**	**1:10**	**1:15**
NWP-RQE	1 day	**6.397**	**7.813**	8.260
NWP-RQE	2 days	6.542	7.925	8.326
NWP-RQE	3 days	6.647	7.975	8.358
QR-NWP1	1 day	6.482	7.862	8.304
QR-NWP1	2 days	6.629	7.966	8.367
QR-NWP1	3 days	6.711	8.003	8.390
QR-combined	1 day	6.674	7.817	**8.206**
QR-combined	2 days	6.858	7.900	8.254
QR-combined	3 days	6.843	7.928	8.270
NWP-WQE	–	8.555	8.821	8.962

Table 6.8: Mean pinball loss for versions of QR-models (6.3), (6.4), and (6.10), with re-weighted NWP input for three re-weighting periods and three loss intervals after completion of re-weighting period. Results for the un-weighted NWP-WQE model shown for reference. Bold indicates best performance for loss interval.

13-15th of every month. After the re-weighting period is finished, we re-issue the forecast with updated predictions from the 16th and onward. Compared to previous sections, the test periods differ as we in this method in effect discard the test observations for the 15 first days the test periods. The cut-off-date of the 15th is in principle arbitrary. This procedure can be initialized at any desirable time point after having obtained new observations. It is chosen to illustrate that we can re-gain skill from a period where the skill of the original model is dropping.

Having obtained the re-weighted principal component values for each member we ran three model set-ups:

1. NWP-RQE: It uses the raw re-weighted values as a direct estimate of the temperature principal component quantile, analogous to the NWP-WQE model (6.3).
2. QR-NWP1: A basic QR model (6.4) with NWP1 as sole predictor (including an intercept).
3. QR-combined: Using the best performing combined QR model of the form (6.10).

Thus the tests are performed using the models from Sections 6.2 and 6.3. The only change is the input data, which now consist of $\tilde{q}^{\alpha}_{t|t-k}$, which is based on re-weighted NWP temperature principal components. For each training set-up we tested to find the best performing tuning parameter γ, which adjusts the size of the weights. For each of the three setups we tested 25 γ-values in the range between 0.000001 and 0.01 in an exponentially increasing sequence.

The results for the best performing γ-value for each model and re-weighting period is shown in Table 6.8. It is clear that the shorter the re-weighing interval

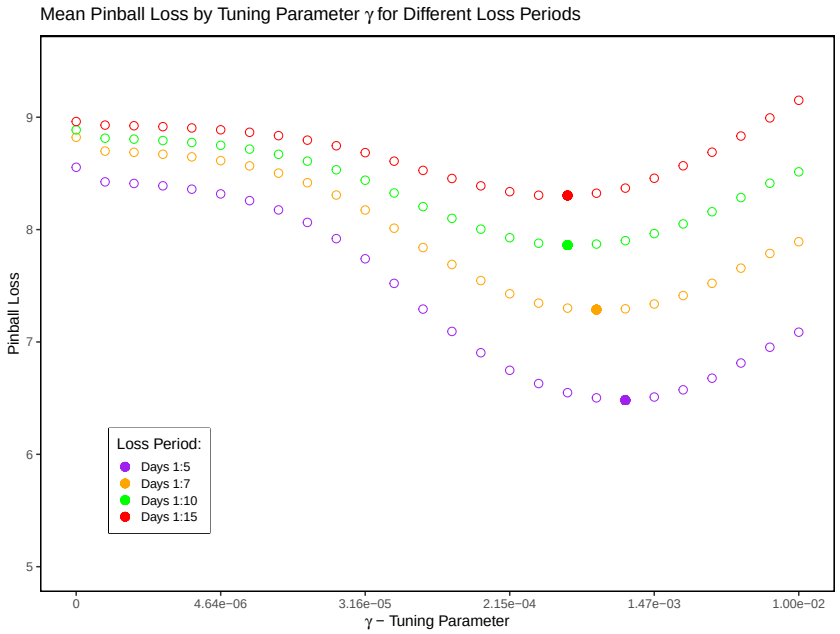


Figure 6.6: Pinball loss for NWP-RQE model (6.3) by values for tuning parameter γ for 4 loss periods after 1 day of re-weighting. Filled in dots mark the best performing γ-value for each period.

the better the model performance. We want to weight on values as close to the ones we are predicting as possible. Based on these results we see no need to test re-weighting intervals that go further back in time. For the first two test periods the NWP-RQE models show the best performance, while the QR-combined models are markedly a step behind the others. The reason for this lies in the composition of the QR-combined model, which in addition to the NWP input is also composed of time covariates. As we saw in Figure 6.4e the β-coefficient for the NWP1-predictor plays a smaller role in these models than for models of the form of (6.3) and (6.4). Therefore, any adjustments made to this input in the QR-combination model will have less impact than in the other models. The QR-combined model does, however, perform better for the overall 15-day period. This is largely because of performance after the skill gain has dropped. This might be more indicative of an over-adjustment on the part of the simpler models at the end of the 15-day period. It is also notable that the QR-NWP1 model has less effect of the re-weighting than the NWP-RQE across all time periods. Above we have reported the results for the best performing model at each loss interval using the best tuning parameter γ for that interval. The size of γ controls the size of the weight adjustments; larger γ's corresponding to larger adjustments. In Figure 6.6 we see the performance of different γ-values for the NWP-RQE model re-weighted over 1 day (4 observations). The same general pattern is repeated for the other models. The curves for the different loss intervals all have a similar shape. Performance changes little when weights

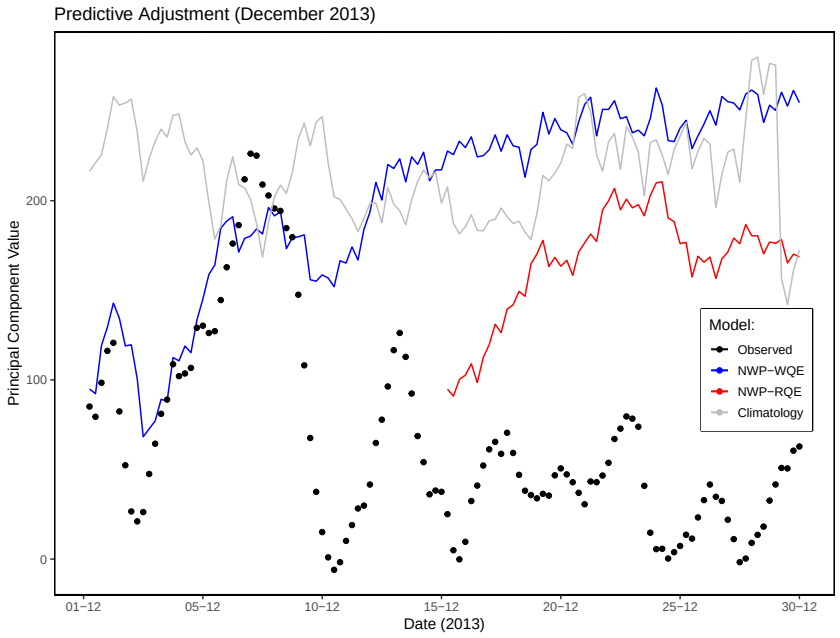


Figure 6.7: Predictive adjustment for NWP-RQE during example month (December 2013).

and therefore also weight-adjustments are low. At the medium range of γ-values the curve dips and we find our chosen γ-values properly located close to the minimum of these curves. Which γ to choose depends on the interval of interest. Larger weights do better at shorter intervals. Over most runs the best γ-values were to be found in the range between 0.000215 and 0.00147. Setting $\gamma = 0.001$ might not be a bad agnostic choice, it is also the value for which we obtain the best performance for the first 5 days after re-issuance.

For an example month (December 2013) we can see the effect of the adjustment at the level of predicted principal component at the 0.9-quantile (Figure 6.7). The performance of the un-adjusted WQE model (blue) has two general phases that we can recognize from this plot. At the beginning of the month this model is informed by high accuracy forecasts. Thus it draws a predictive distribution very close to the observed values. After the 10th, the forecast information looses accuracy and the predictive distribution is drawn much looser and follows climatology to a greater extent. Even if the WQE predicted values keep above the observed values, the increased distance indicate a higher loss than earlier in the month. Its performance during the latter half of the month is slightly worse to that of climatology (grey). The first forecast of the RQE-model is made on the 16th. The improved performance obtained by re-weighting can be clearly seen during the first five days after re-issuance. The RQE-model up-weighs information from those ensemble members that most accurately followed the dip in the observed principal component values around the 15th. However, the trajectories of the up-weighted ensemble members are

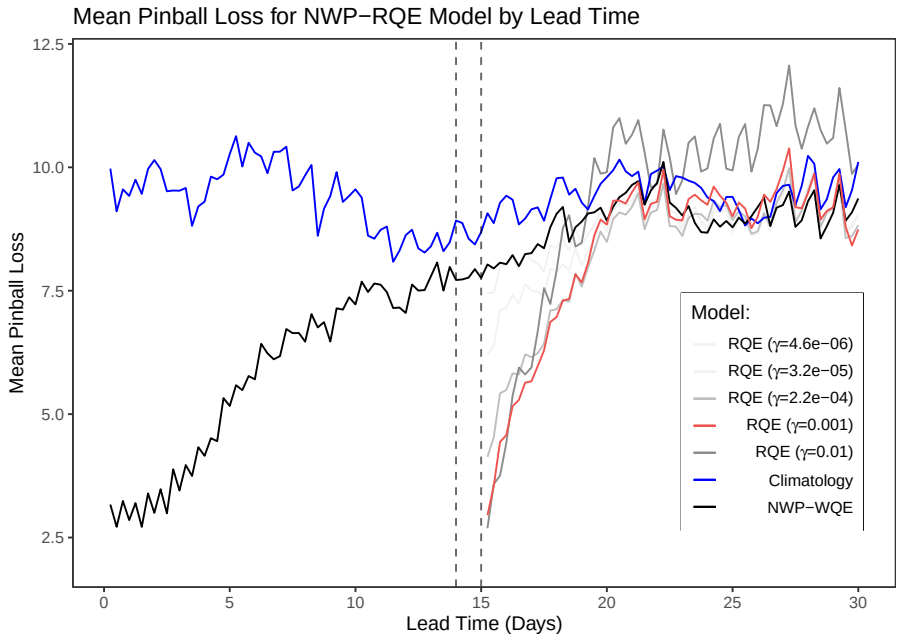


Figure 6.8: Comparing pinball loss over lead time for NWP-RQE for different γ values (best γ is marked red) against climatology and NWP-WQE (black).

in general not the best for an extended period of time. At the end of the month both the WQE and the RQE fail to track the low principal component values (corresponding to a higher temperature), though the latter is performing considerably better during this example month.

To illustrate the gain in performance achieved by re-weighting we can look at Figure 6.8, which compares the performance of the re-weighted NWP-RQE model (red) with the NWP-WQE and climatology by lead time. The re-weighting period is finished on the 15th at 24:00 (dotted line). The performance of the re-weighted models is best immediately thereafter. And it is comparable to the performance at the beginning of a forecast month (roughly days 1:3). The loss then increases fairly rapidly and by 5 days after re-weighting it settles into the same pattern as the other un-weighted models, reverting to climatology by day 20. Performance for other γ-values (grey) show similar behaviour, but with less gain in performance.

Figure 6.9 shows the skill score relative to climatology for all three set-ups, using 1 day of weighting and $\gamma = 0.001$, the best scoring setting for the first 5 days. The skill of NWP-RQE and QR-NWP1 are similar throughout with the former having a slight advantage. The QR-combined model is slightly worse the first 5 days before thereafter consistently showing higher skill. In Table 6.9 we show the skill score for the same three models both with reference to climatology and to the same model without re-weighted input. In contrast to the previous table we now show results for three segmented lead time periods. For the first five days after re-weighting the NWP-RQE model has the highest skill both with reference to climatology and compared to its own previous performance without re-weighted inputs. The performance of QR-NWP1 is

		Skill Score (Days after R)		
Model	**Reference**	**1:5**	**6:10**	**11:15**
NWP-RQE	Climatology	**0.313**	**0.039**	**0.017**
QR-NWP1	Climatology	0.298	0.027	−0.024
QR-combined	Climatology	0.286	0.005	−0.033
NWP-RQE	NWP-WQE	**0.238**	**-0.011**	−0.013
QR-NWP1	QR-NWP1	0.226	−0.022	−0.053
QR-combined	QR-combined	0.181	−0.079	−0.090

Table 6.9: Average skill score for models with re-weighted inputs with reference to climatology (top) and with reference to the same models without re-weighted input (bottom). Bold indicates best parameterization.

slightly worse, while the QR-combined model is a step behind the other two. In the following to segmented periods 6-10 and 11-15 days after re-issuance there is a sharp drop-off in skill for all re-weighted models. We are seeing some slight improvements compared to climatology in this period, but compared to the unweighted models, the re-weighted models do worse.

In this section we have shown that the re-weighting procedure can obtain a short term skill increase on average over the test periods lasting for up to 5 days. The skill level obtained is (for a short period) comparable to the skill at the beginning of a forecast month. We have here shown the performance of the re-weighting using only one re-weighing period, but it can be initiated at any given time after NWP forecast issuance.

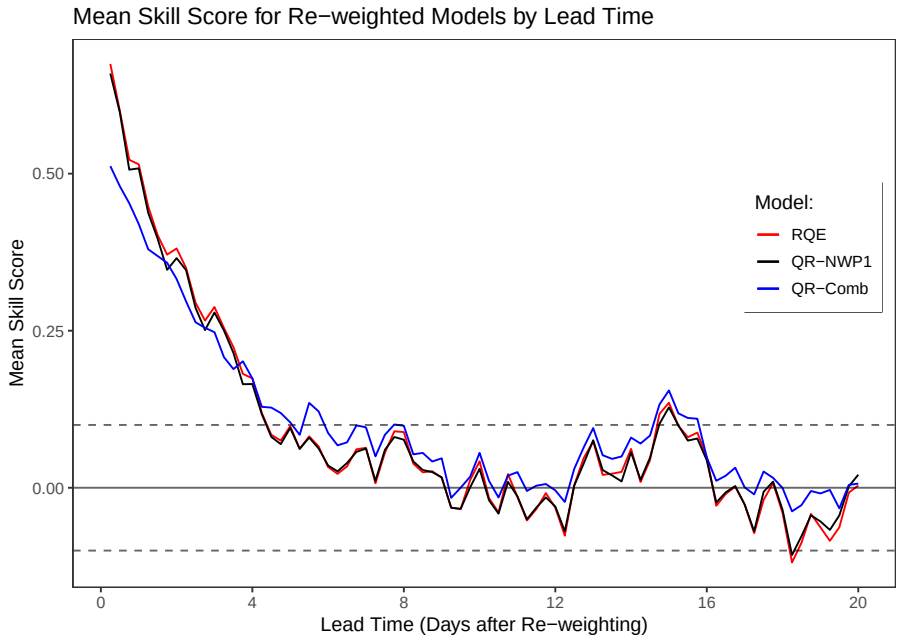


Figure 6.9: Mean skill score by lead time for models with reweighted input, with reference to climatology.

6.5 Forecast Aggregation

So far, we have been looking at the performance of the probabilistic PC temperature forecasts at the hourly level (with 6-hour intervals). We have seen that at this level of granularity model skill is negligible after 15 days. By re-weighing the principal components of the forecast ensemble members, based on the most recently observed temperature, we can regain skill for short periods after forecast re-issuance. The seasonal NWP forecasts are, however, meant to be skillful at lead times far exceeding 15 days (Section 4.3). In this section we will show that we can make model skill apparent at longer lead time horizons, in principal component space, by aggregating forecast predictions over longer time intervals.

By aggregating predictions over longer time intervals the low-frequency aspect of the series, the trend and cyclical patterns, might become apparent (Nystrup et al. 2021). For example, if an event happened on Tuesday, we are wrong if we predicted it would happen on Wednesday, but correct if we predicted it would happen this week. A forecast might be correct in tracking the path of e.g. a high pressure system, but at a low hierarchical forecast level (e.g. hourly predictions) a model might miss its movement to a specific location by a couple of days. Thus the model's ability to track the path goes unrewarded at a low aggregation level. By aggregating over a sufficiently large period, however, the skill of the model might be made apparent (Section 3.6). In our presentation we will focus on the 0.5- and 0.9-quantiles.

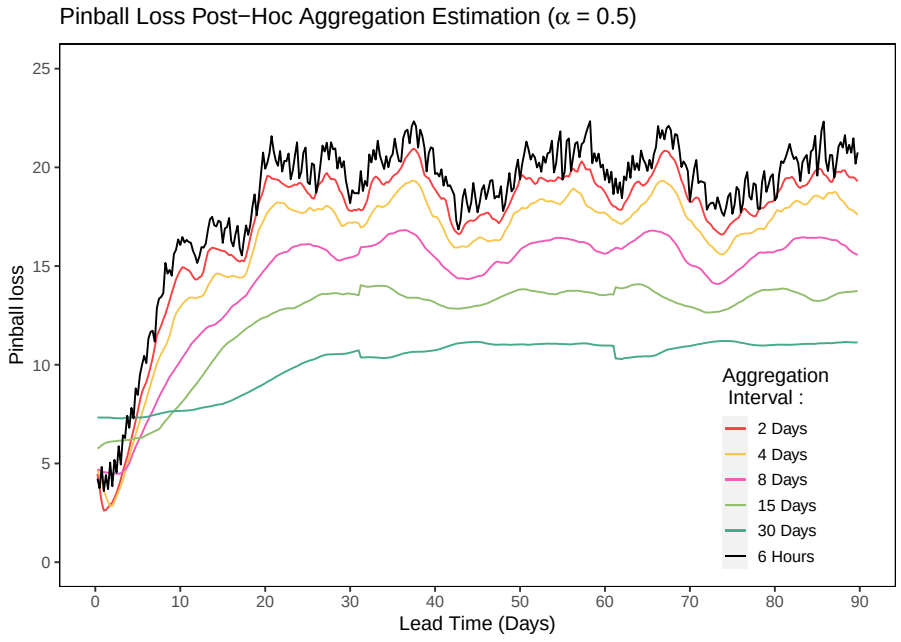


Figure 6.10: Pinball loss by lead time using post-hoc aggregation for model QR-NWP (6.4) with reference to climatology at 0.5-quantile for different aggregation intervals.

Post-hoc Aggregation

A relatively easy way of aggregating predictions might be termed the post-hoc aggregation method (Section 3.6). Instead of re-estimating the model for larger intervals we are working with quantile predictions already made at the base 6-hourly level. We are interested in the quantile, α, of an aggregate, namely $S^{\alpha}_{1:T} = (\frac{1}{T}\sum_{t=1} C_t)^{\alpha}$. To estimate it using post-hoc aggregation we take a simple average of the forecasted quantile values: $\hat{S}^{\alpha}_{1:T} = \frac{1}{T}\sum_{t=1}^{T} \hat{C}^{\alpha}_{t}$, over an interval of interest, $t \in 1, ..., T$. We compare this to the average of the observed temperature principal component $\bar{C}^{1}_{1:T}$, using the pinball loss function, $\rho_{\alpha}(\cdot)$, at quantile α:

$$J_{1:T} = \rho_{\alpha}(\bar{C}^{1}_{1:T}, \hat{S}^{\alpha}_{1:T}), \tag{6.12}$$

where $J_{1:T}$ is the loss over the aggregated interval. We apply this method to forecast outputs from the QR-model (6.4) with intercept: $\hat{C}^{\alpha}_{t} = \hat{\beta}_0 + \hat{\beta}_1 q^{\alpha}_{t|t-k}$. In this application we have used a rolling average where the mean of the observations at the extremes of the lead-time interval are filled in. In Figure 6.5 we see the results for 6 aggregation levels with the median, $\alpha = 0.5$, as the target. The top black line is the pinball loss for the base un-aggregated 6-hour level. The colored loss curves below clearly follow the behaviour indicated by Jenssen's inequality. The larger the interval under consideration, the lower the loss.

The pinball loss and skill score for lead time days 1:60 is shown in Table 6.10 for the 0.5- and 0.9-quantile. The general behaviour for both climatology and the QR-model is the same at both quantiles. For all models we observe the pinball loss decreasing when the length of the aggregation interval increases. We also observe that the skill score at the 0.5-quantile increases by aggregation length. A similar picture is painted at the 0.9-level with a large caveat: the method is not accurate at this quantile. As discussed in Section 3.6 the mean of the quantile does not accurately reflect the quantile one is interested in estimating. This is especially problematic for quantiles further away from the median, for which we are especially interested. Its use here is only as a reference to see the difference in estimation compared to a proper estimation technique.

	Pinball Loss 1:60				Skill Score 1:60	
	0.5-quantile		0.9-quantile		0.5	0.9
Aggreg.	**Climat.**	**QR-NWP**	**Climat.**	**QR-NWP**	**-**	**-**
6-hour	20.548	**17.740**	9.391	8.178	**0.137**	0.129
2 day	19.473	**16.569**	9.001	7.755	**0.149**	0.138
4 day	18.311	**15.433**	8.623	7.402	**0.157**	0.142
8 day	16.369	**13.700**	8.150	6.901	**0.163**	0.152
15 day	14.173	**11.821**	7.785	6.510	**0.166**	0.164
30 day	11.776	**9.705**	7.550	6.307	**0.176**	0.164

Table 6.10: Pinball loss and skill score using post-hoc aggregation for model QR-NWP (6.4) with reference to climatology at 0.5- and 0.9-quantiles. Bold indicates best parameterization.

Copula Aggregation

A proper way of performing quantile aggregation has to take account of the joint distribution of the time points one wants to aggregate over. The other aggregation method we consider is the Gaussian Copula (GC) method which does exactly this (Section 3.4). It involves simulating from the correlation matrix of the joint distribution of the predicted values in the interval of interest. One then forms the prediction by taking the weighted sample quantile over a set of simulated aggregates, $S^{(m)}$, using the WQE function:

$$\hat{S}^{\alpha} = W_S^{\alpha}(S^{(1)}, ..., S^{(M)}). \tag{6.13}$$

Just as for the post-hoc method we are interested in estimating the quantile of the mean PC temperature: $S^{\alpha}_{1:T} = (\frac{1}{T}\sum_{t=1} C_t)^{\alpha}$.

Like above, we look at the performance of the QR-model (6.4), which means we use this model to obtain the marginal distributions at each target time t. The level of precision of the estimation of the predictive CDF can be set at any level. We opted to use two different settings, estimating the quantile range using either 100 or 1000 quantile values. By increasing the number of quantile values we in effect increase the accuracy of the estimate of the correlation matrix we simulate from. For the GC method we also chose a computationally less demanding, segmented, interval structure instead of the rolling average used in the post-hoc method in the previous section.

Figure 6.11 showcases the pinball loss test results for a set of aggregation intervals (between 2 and 60 days) for QR-model (6.4) using 1000 quantile values. We observe the same general pattern implied by Jensen's inequality as for the post-hoc method. The longer the interval, the lower the pinball loss. The slight bumps in the loss at points around lead times 30, 60, and 90 are artefacts of an unequal number of lead times for each prediction.

The results for the first 60 lead days are summarized in Table 6.11. Most importantly we see that the skill score of the QR-model increases with the length of the aggregation interval. We also observe that the number of quantile values used to estimate the predictive CDF does not matter much for the climatology model. For the QR-model the number of quantile values is not important for the shortest aggregation intervals, but it does seem to matter for the larger ones. At aggregation intervals 30 and 60 days the increase in precision when estimating the correlation matrix using 1000 quantile values seem to lead to better results.

Looking exclusively at the 0.9-quantile there are noticeable differences in results between the GC and the post-hoc method. The GC pinball loss for both climatology and the QR-model decrease much faster than for the post-hoc method. This difference does not translate into a difference in skill score, at least not for the shorter aggregation intervals. However, we do see a difference for the 30-day aggregation interval. While the post-hoc method has a skill score of 0.164, the best GC-method obtains 0.223. Bear in mind that this result does not mean that the GC-method has achieved a better predictive score. Rather, it means that by taking account of the correlation structure between predictions we should trust the GC-estimate more in accurately portraying the skill of the model over the aggregation interval.

In Figure 6.12 we see the skill score for QR-model (6.4) using 1000 quantile values. We first observe that the 60-day aggregation interval has a positive skill

score, relative to climatology, throughout the 120-day lead time period. The 30-day aggregation has a positive skill lasting for 60 days. While the 15- and 8-day aggregations are skillful up until around 30 days, the shortest intervals show positive skill lasting to around 25 days. It is clear that by increasing the length of the aggregation interval we are seeing the period of model skill prolonged substantially.

In Section 6.2 we observed that model skill at the 6-hour level disappeared after 15-23 days. By employing the GC aggregation method we take account of the correlation structure between PC temperature observations. Utilizing this to form aggregations we have been able to make model skill for principal component temperature forecast models apparent at longer lead times. We expect to see similar results for other quantiles than the 0.9-quantile. The results presented here are, however, based on single model runs, each of which being computationally heavy. Since the GC method involves simulating predictive values it is seed-dependent. To improve the robustness of the results, more model runs would be necessary.

	Pinball Loss 1:60				Skill Score 1:60	
	Climatology		QR-NWP			
Aggregation	**100**	**1000**	**100**	**1000**	**100**	**1000**
1 day	9.057	9.054	**7.935**	7.943	0.124	0.123
2 days	8.864	8.860	**7.712**	7.712	0.130	0.130
4 days	8.416	8.379	**7.240**	7.251	0.140	0.135
8 days	7.630	7.661	**6.494**	6.471	0.149	0.155
15 days	6.605	6.590	**5.529**	5.585	0.163	0.152
30 days	5.323	5.307	**4.368**	4.121	0.179	0.223
60 days	4.970	4.994	**3.517**	3.469	0.292	0.305
6-hour		9.391		8.178		0.129

Table 6.11: Pinball loss and skill score using copula estimation for QR-NWP with reference to climatology at the 0.9-quantile. Bold indicates best parameterization. Results are shown for estimation methods using either 100 or 1000 quantile values. Non-aggregated (6-hour) results shown for reference.

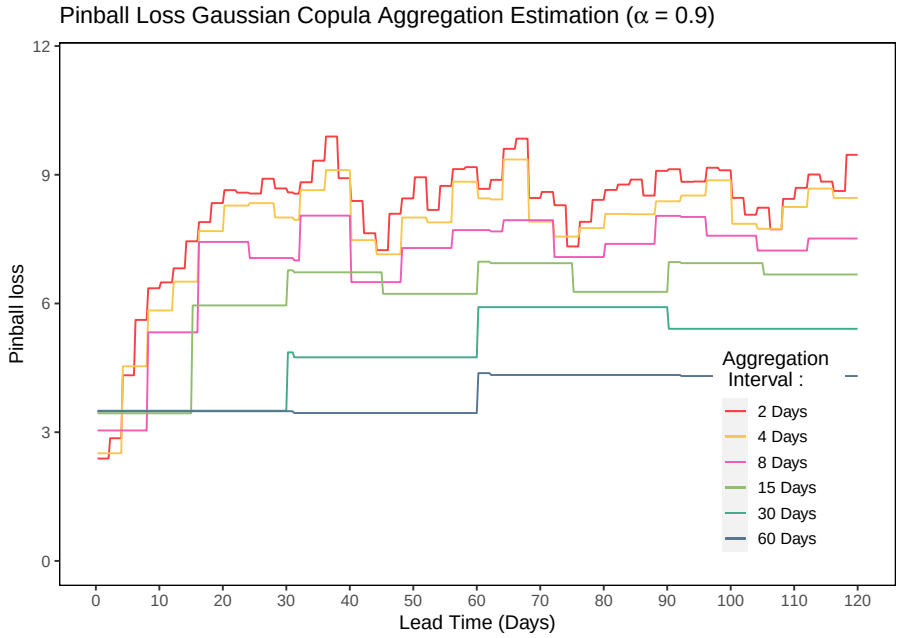


Figure 6.11: Pinball loss for QR-model (6.4) for six aggregation intervals using GC aggregation estimation at the 0.9-quantile.

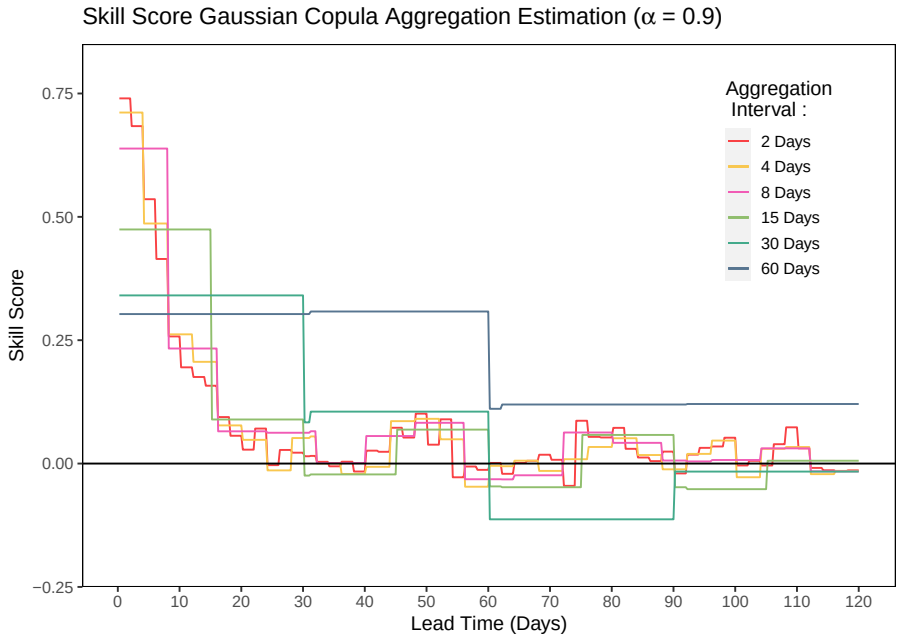


Figure 6.12: Skill score for QR-model (6.4) for six aggregation intervals using GC aggregation estimation at the 0.9-quantile.

CHAPTER 7

Demand Forecasting with NWP Temperature PCs

The results from our investigation of the electricity demand problem in Chapter 5 showed that the best GAM-PC model represented a substantial improvement on both the climatology demand model (5.2) and all time covariate models. This was under the assumption that we, at the time of forecast issuance, had access to near-future temperature. In Chapter 6 we saw that we could utilize NWP temperature forecasts as a foundation for forming probabilistic forecasts of temperature PCs. At the hourly level our models showed positive skill compared to temperature climatology (6.1) up to 15 days from forecast issuance.

In this chapter we will merge these two approaches. We will fold the probabilistic forecasts of PC temperature as feature inputs into the demand model. In Section 7.1 we utilize NWP forecasts of PC temperature as inputs into the GAM-PC1+2 model to form a point forecast of electricity demand. We contrast the performance of this point forecast with feeding the same model with two other forms of temperature inputs: 1) climatology estimates of PC temperature (the real-life baseline alternative); 2) observed temperature (the 'optimal' case which we made use of in Chapter 5. The aim of Section 7.2 is then to extend the probabilistic temperature model explored in Chapter 6 onto the demand domain. Again we will be using the temperature PC climatology model as the baseline comparison.

The general set-up for model evaluation is similar to the one used in Chapter 5. We will use three year training periods, while keeping the hourly interval data as before. The difference lies first in the feature inputs used for testing, and second, in the interval length between test target observations. Because we are using NWP forecasts with 6-hour intervals, the test observations will also have 6-hour intervals. The RMSE and skill score results presented will be roughly on the same order as previous chapters. The pinball loss, however, is now applied in a different setting than in Chapter 6. It will lie on a different scale for the demand forecasting problem than for the temperature forecasting problem.

7.1 NWP-Based Point Forecast of Demand

In this section we build upon the structural demand models described in Section 3.3, and investigated in Chapter 5. We will now, instead of assuming we have

access to future PC temperature, use NWP forecasts of PC temperature as inputs in a point forecast of electricity demand. The model we will be utilizing for this purpose is the best performing GAM-PC model from Chapter 5, namely GAM-PC1+2:

$$y_t = f(x_t) + s(C_t^1) + s(C_t^2) + \epsilon_t, \tag{7.1}$$

where $f(x_t)$ is specified by the interaction model including the interaction terms for hour:week and weekday:month, and $s(\cdot)$ is a spline function (see Section 5.4). We will look at three versions of this model. The first, 'optimal' version, uses the observed principal components, C_t^j, as inputs. The second version uses the climatology estimate of PC temperature (6.1). The third version uses as input the mean over the set $\mathcal{C}_{t|t-k}^{j,1:M}$, which is a set of the jth temperature PC for each NWP ensemble member, at target time t, issued at a specific lead time k (see Section 3.4). We write this as:

$$\bar{C}_{t|t-k}^{j} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{C}_{t|t-k}^{j,m}. \tag{7.2}$$

Besides the PC temperature inputs, the different model versions are the same.

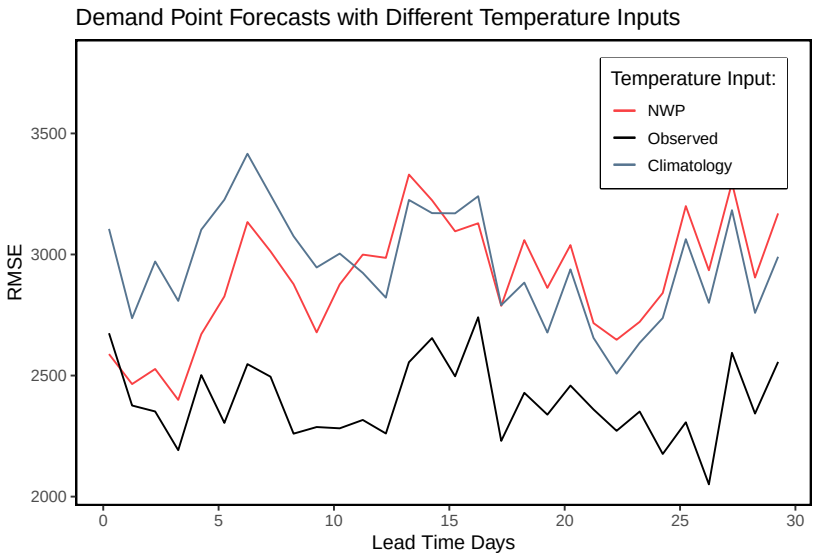


Figure 7.1: Mean RMSE over lead time for demand model point forecast with three types of temperature inputs: NWP, observed temperature, and climatology.

In Figure 7.1 we see the comparative performance of the GAM-PC1+2 model utilizing three different PC temperature inputs: NWP, observed, and climatology. For ease of presentation we only show results at 6:00 for each lead time day. The other hours exhibit only slight differences. The black line indicating observed temperature shows the optimal performance of our model.

		Skill Score with Reference to:	
Temp. Input	**RMSE**	**Interaction**	**Climatology Input**
Interaction	3312.55	–	–
Climatology	2930.50	0.217	–
NWP	2712.80	0.329	0.143
Observed	**2429.42**	**0.462**	**0.313**

Table 7.1: Model performance (RMSE and skill score) for GAM-PC1+2 with three different PC temperature inputs. Skill score with reference to model version with inputs from the temperature PC climatology model (6.1), and with reference to the best time covariate interaction model.

If we had perfect knowledge of future temperature this is the results we would obtain. To be expected, the performance of the NWP-folded demand model is worse than the optimal case. Importantly, we observe that for the first 5 days performance of the NWP model is roughly at the same level as the optimal version. Compared to using inputs from the climatology temperature PC model (6.1), we see a clear improvement during the first 10 days. Thereafter, the performance for NWP and climatology inputs are roughly the same, with climatology performing slightly better during lead time days 15:30. Given the performance of the temperature forecasting model, which showed positive skill up to 15 days, this is close to what we would expect.

In table 7.1 we summarize the performance of the different versions of the GAM-PC1+2 model for the first 15 lead time days. The NWP model has a skill score of 0.143 compared to using climatology inputs. We see that by employing NWP-folded inputs we can substantially improve upon using inputs from climatology. We also observe that utilizing the NWP inputs represent a substantial improvement upon the best model utilizing only time covariates with a skill score of 0.329.

7.2 Probabilistic Electricity Demand Forecasting

In this section we will extend the probabilistic temperature forecast explored in Chapter 6 onto the demand forecasting problem. We will first describe the process we follow for forecasting the 0.9-quantile of demand, before we provide an example of a predictive distribution that covers the quantile range. For data-handling reasons we have opted to form the quantile forecast of demand by first finding the α-quantile of the NWP members through using the WQE function, and then feeding this estimate once into the structural model. Alternatively, we could have chosen to input the PC forecast from each NWP member individually, and then forming the quantile estimate based on the model output. Because the estimating function is monotonic the two methods are considered to be equivalent. Our approach is a pragmatic way of forming a predictive distribution.

In order to forecast the 0.9-quantile of demand we will at each forecast month first train the same GAM-PC1+2 model as described above. Then, for each target time, t, we utilize the NWP-WQE quantile estimate, $q^{0.9}_{t|t-k}$, of the first principal component as test input into the GAM-PC1+2 model. As input

	Pinball Loss		Skill Score	
Temp. Input	**1:15**	**1:30**	**1:15**	**1:30**
Climatology	506.46	515.79	–	–
NWP-WQE	**462.06**	**488.95**	**0.087**	**0.052**

Table 7.2: Model performance (mean pinball loss and skill score) for GAM-PC1+2 at 0.9-quantile using climatology (6.1) and NWP-WQE (6.3) temperature PC input. Skill score relative to climatology temperature PC input.

for the second PC we will, for conceptual simplicity, utilize the mean over NWP ensemble members, $\bar{C}^2_{t|t-k}$. The next step consists in sampling M errors from a normal distribution utilizing the estimated variance of the residuals from the model output. We then add the error uncertainty to the model function for each $m \in \{1, ..., M\}$:

$$y_t^{(m)} = f(x_t) + s(q^{0.9}_{t|t-k}) + s(\bar{C}^2_{t|t-k}) + \epsilon_m, \tag{7.3}$$

where $\epsilon_m \sim N(0, \sigma^2)$. By taking the sample quantile over this set we can finally obtain a forecast of the 0.9-quantile of demand:

$$y_t^{0.9} = W^{0.9}(y_t^{(1)}, ..., y_t^{(M)}). \tag{7.4}$$

The mean pinball loss at the 0.9-quantile of the GAM-PC-1+2 model with temperature inputs from climatology and from NWP is shown in Figure 7.2. For the NWP input we again witness a lead time dependent pattern. The performance in terms of pinball loss is best in the period immediately following forecast issuance. It outperforms the temperature PC climatology input for the first 12 days. Again, this is close to what we would expect given the performance of the temperature PC forecasting model. In Table 7.2 we compare the performance of the climatology and NWP inputs in terms of pinball loss and skill score. We observe that utilizing the NWP-WQE input represents an improvement on using the climatology input, obtaining a skill score of 0.087 for the period covering the first 15 days.

To estimate the predictive distribution of demand we follow the same procedure as described above for 9 quantiles in the range 0.1-0.9. Figure 7.3 displays the predictive distribution of demand for an example month (January 2022). The black line indicates the observed demand, while the red line is the point forecast based on NWP input. The blue shading shows the predictive distribution around the point forecast. From the top, the darkest blue shading marks the area between the 0.8- and the 0.9-quantiles. We observe that the point forecast for this months tracks the observed demand fairly well. We also see that the observed values keeps within the predictive range between the 0.1- and the 0.9-quantile for the whole month. By utilizing forecasted PCs as input in the structural demand model we have transformed a temperature forecast into a demand forecast. By taking account of the uncertainty in the temperature (here in the form of the first principal component) we have shown that we can form a fairly good predictive distribution of electricity demand in the Nordic region.

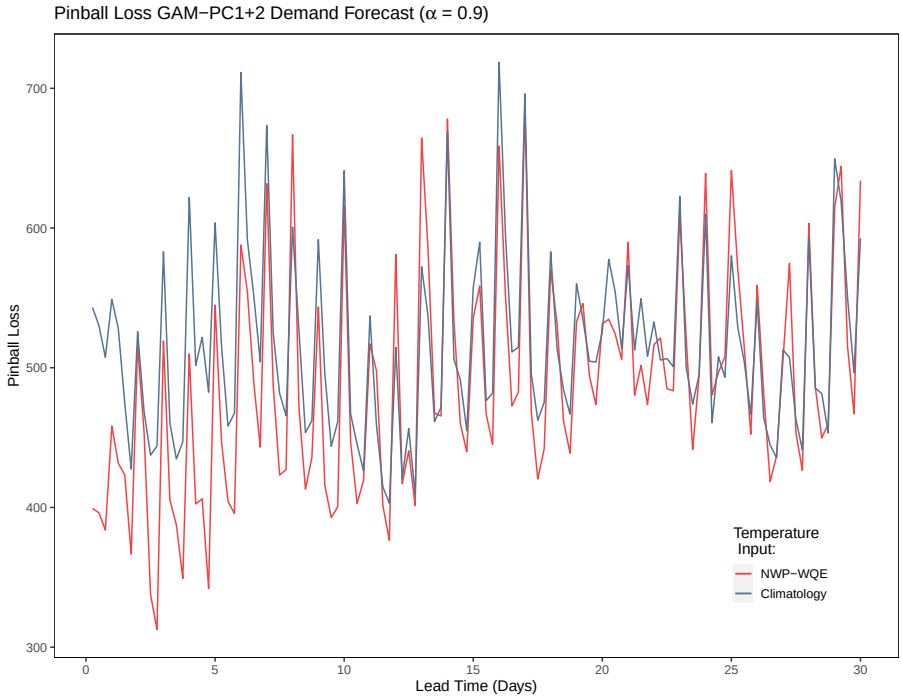


Figure 7.2: Mean pinball loss by lead time ($\alpha = 0.9$) for demand forecasts with temperature input in the form of NWP-WQE (6.3) or climatology (6.1).

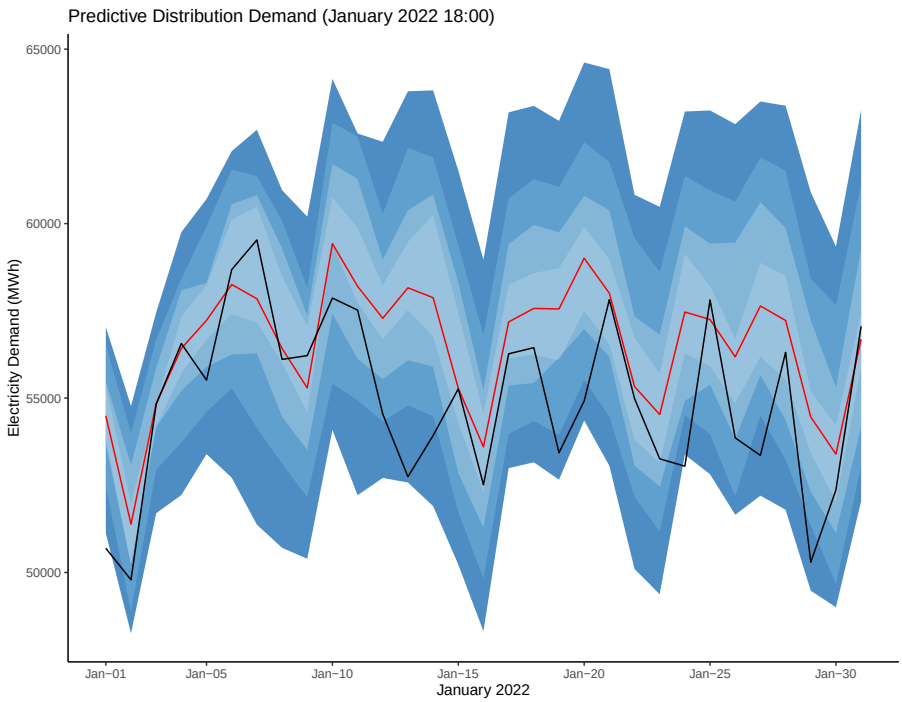


Figure 7.3: Predictive distribution of demand (Jan 2022 at 18:00). The point forecast of demand is show by the red line, observed demand is indicated by the black line, while the ranges of the predictive distribution is shown in shades of blue.

CHAPTER 8

Conclusion

In the beginning of this thesis we asked the overarching research question: How can we forecast future energy demand at the medium-term based on probabilistic temperature data? We re-framed this question into two problems: 1) How can we build a good structural model for predicting energy demand by using temperature data? 2) How can we use seasonal NWP forecasts to give us distribution estimates of future temperature which we can fit into our demand model? In these concluding remarks we will first summarize the main findings from each of the result chapters. We will at the very end address how the results could be strengthened, as well as striking a path for future research.

The main contribution of this thesis has been the introduction a medium-term forecast model for electricity demand in the Nordic region utilizing NWP temperature forecast data. This is a year-round model which takes account of the varying effects of temperature on demand throughout the year. We developed this model by first (in Chapter 5) building a structural electricity demand model which relates temperature to demand at any given target time t. In Chapter 6 we developed a probabilistic temperature model in principal component space. And lastly, in Chapter 7 we established that by folding the NWP temperature forecasts into the demand model we can extend the probabilistic temperature PC forecast into a probabilistic electricity demand forecast. We will now, in turn, briefly go over each of the results chapters.

In Chapter 5, our overarching goal was to find and present the best GAM-PC structural demand model and asses its merits. For model evaluation purposes we used the PCV procedure applied over lead times between 15 and 45 days from forecast issuance. Most importantly, we found that the best model which incorporates information from both time covariates and principal components of the temperature grid was the GAM-PC1+2 model. In addition to including an involved parametrization of the time covariates, it also included the two first temperature PCs. Its skill score relative to climatology was 0.59, which constitutes a large improvement in forecasting ability.

With regard to the isolated contribution of temperature on predictive performance we assessed the performance of models containing either just temperature information or just time covariates. We found that a good parametrization of the time covariates led to a substantial improvement on climatology, with a skill score of 0.42. In contrast, none of the models containing just temperature information performed better than the climatology model. Among the temperature predictors we found that the first PC is a more accurate predictor than the mean grid temperature. Importantly, we observed that by

adding the two first PCs to the best parametrization of time covariates we could substantially improve predictive performance: The skill score of the GAM-PC1+2 relative to the best model with only time covariates was 0.29. How these findings compare with results from similar forecast studies is difficult to ascertain. This is because the contribution of temperature to predictive performance is often not explicitly stated, or it is made at different resolutions and time scales (Section 2.1).

Furthermore, we examined the impact of changing the length of the training data. We noted that model performance was improved by restricting the training period to 3 years of preceding data. This might suggest that the relation between demand and temperature have been undergoing a slight shift. We also demonstrated that the performance of the GAM-PC1+2 model is strongly month dependent. The improvement in performance is especially high during the winter months when cold temperatures drive the usage of heating appliances. We also provided a very straight-forward interpretation of the good performance of the GAM-PC1+2 model: The model makes us able to say not only that demand is high because it is January, but that it is exceptionally high because it is a very cold January. Finally, when comparing the GAM-PC1+2 model with alternative implementations, we found that it performs slightly worse, but has key advantages when it comes to interpretability and parsimony.

In Chapter 6 our goal was to explore methods that enable us to forecast future temperature principal components. We focused on probabilistic temperature forecasts utilizing seasonal NWP model output as a way of estimating temperature principal component quantiles. We first looked at the performance of the direct WQE model. We found that this model exhibited considerable skill up until 15 days from forecast issuance. After 15 days, model performance was only marginally better than the climatology baseline. Furthermore, we investigated the performance of models which included weights, lagged forecasts and time covariates in a QR framework. We found that this additional information offered only minor predictive improvement. The main takeaways from this discussion is, first, that we have successfully shown that NWP forecasts perform well at forecasting temperature also within principal component space. And second, that the NWP forecasts in PC space require little or no modification to achieve these results. Like the case was for the demand model we also see considerable monthly variation in terms of performance for the temperature PC forecasting task. Forecasting PC temperature during winter months is markedly harder than for the summer period.

An important contribution of this thesis was the introduction of the re-weighing scheme for NWP principal components, described in Section 3.5. By re-weighing temperature forecasts based on recent performance, we can 'update' the forecast and obtain short-term improvements in skill at any time point. We found that the best weighing interval was the 1-day period immediately preceding forecast re-issuance. The improvement in skill score was found to last for a period of 5 days. We also built a Gaussian Copula (GC) aggregation method for making forecast model skill apparent at longer lead times. We showed that for each increase in the length of the aggregation interval we observed not only lower pinball loss, but also higher skill scores for our QR-model.

Finally, in Chapter 7, we merged the two approaches from Chapter 5 and Chapter 6. By folding the probabilistic forecasts of PC temperature as feature inputs into the demand model, we first contrasted the point forecast utilizing

perfect temperature information with one based on NWP inputs. We found that by utilising the NWP forecast inputs we obtained a skill score with respect to climatology of 0.143 for the first 15 days after forecast issuance. In addition, the improvement with respect to the best model without temperature information was found to be substantial, with a skill score of 0.329. We then extended the probabilistic temperature PC model onto the demand domain. Focusing on the 0.9-quantile we found that using NWP-WQE inputs led to a skill score of 0.087 during the first 15 days.

Even if this thesis has covered a lot of ground there are several things we would have liked to improve upon if we had more time. Foremost, we would have liked to have presented a fuller scope of results relating to the final probabilistic demand model. What we have demonstrated with regard to the predictive distribution of demand is more a conceptual outline, than a full evaluation of performance. A complete treatment would also have contrasted these results with alternative methods of conceptualizing the uncertainty in the demand forecast. In addition, we would have liked to observe the performance of both the re-weighting scheme and the forecast aggregation method applied in a demand forecast setting. A fuller analysis would also have included a presentation of the monthly variation in the probabilistic demand forecasting performance.

Concerning the re-weighting scheme, an obvious path of further development is to implement a rolling version where the re-weighting is performed as soon as new temperature information is recorded and available. The upside to such an implementation would be that we at any given time would have forecast estimates weighted to prioritize the ensemble members tracking the most recently observed temperature the best. For this purpose we might consider a Bayesian approach which keeps track of the weights as new information is recorded. We would also have liked to look at how much the choice of quantile estimator effects the predictive performance of the re-weighting.

With regard to the Gaussian Copula forecast aggregation method, we would have liked to present results accounting for the seed dependence in the sampling. The GC application could also have benefited from a rolling average implementation instead of the segmented implementation we utilized. In general, we also would have liked to establish a firmer testing framework for utilizing permutation tests.

There is also ample room for further exploration of alternative modelling strategies. In this thesis we have relied on one main strategy of dimensionality reduction, through PCA. It might also be worthwhile to investigate the performance of alternative dimensionality reduction methods such as utilizing auto-encoders. With regard to the quantile estimation task we have only scratched the surface of possible modeling frameworks. In order to incorporate predictors in our quantile estimates we have relied on the Quantile Regression framework. Options to this approach include Quantile Random Forest and Quantile Gradient Boosting. As we mentioned in Chapter 3.3 we have chosen to work with the correlation structure through the incorporation of NWP forecasts. Model improvements could potentially be gained from investigating GAM models with ARMA errors to account also for the correlation structure not related to temperature.

Our modeling approach has been centered around the incorporation of new data. A natural path of further investigation would be to either look at

alternative versions of the data we already have, or look for entirely new data to incorporate into the structural demand model.

A core aspect of our approach has been the temperature grid over the Nordic region. In this thesis we have taken the dimension and resolution of this grid to be fixed. An interesting line of investigation would be to look at the effect of utilizing temperature grids with other dimensions. It is possible that either expanding or contracting the grid might better reflect the areas where electric energy is consumed in the Nordic region. It might also be worthwhile to look into the effect of increasing the resolution of the grid to obtain more granular temperature estimates. In this thesis we have been working with only one response variable for demand. Following De Felice et al. (2015) it would also be interesting to divide the demand data by country and apply the coupled manifold approach which involves subjecting both the temperature grid and the demand regions to a coupled PCA transformation.

One of the biggest benefits of the structural demand model is that it allows for the incorporation of several data sources directly into the model. This framework could for example incorporate other forecasting data, or information from weather-related phenomena. One interesting model expansion is to incorporate information from stratospheric wind-data, especially relating to Sudden Stratospheric Warming (SSW) events, into our forecasting model. In winter the stratosphere in the Arctic is characterized by cold rotating air flowing westwards at high speeds (50-80 km/h). This is the northern hemisphere stratospheric circumpolar vortex or for short the polar vortex. It is a seasonal phenomenon as it spins up in late autumn and typically lasts until April when the winds abide and turn easterly. Roughly every other year the polar vortex breaks down mid-winter (before its usual downturn). This is associated with a sudden warming of the stratosphere, which in turn has an effect on tropospheric temperature (Butler et al. 2017). Mining information relating to the time span between vortex breakdown and warming might yield predictive gains in forecasting temperature at longer horizons. This could in turn improve demand forecast performance especially during the winter months.

Bibliography

Achuo, E. D., Miamo, C. W. and Nchofoung, T. N. (2022). 'Energy consumption and environmental sustainability: What lessons for posterity?' In: *Energy Reports* vol. 8, pp. 12491–12502. URL: https://www.sciencedirect.com/science/article/pii/S2352484722017577.

Agresti, A. (2015). *Foundations of linear and generalized linear models*. Hoboken, N.J.

Akinshin, A. (2022). 'Trimmed Harrell-Davis quantile estimator based on the highest density interval of the given width'. In: *Communications in Statistics - Simulation and Computation* vol. 0, no. 0, pp. 1–11. URL: https://doi.org/10.1080/03610918.2022.2050396.

— (2023). 'Weighted quantile estimators'. In: URL: https://arxiv.org/abs/2304.07265.

Arguez, A. et al. (2012). 'NOAA's 1981–2010 U.S. Climate Normals: An Overview'. In: *Bulletin of the American Meteorological Society* vol. 93, no. 11, pp. 1687–1697. URL: https://journals.ametsoc.org/view/journals/bams/93/11/bams-d-11-00197.1.xml.

Austvik, O. G. (2019). 'Norway: Small State in the Great European Energy Game'. In: *New Political Economy of Energy in Europe: Power to Project, Power to Adapt*. Ed. by Godzimirski, J. M. Cham: Springer International Publishing, pp. 139–164. URL: https://doi.org/10.1007/978-3-319-93360-3_6.

Bala, D. A. and Shuaibu, M. (2022). 'Forecasting United Kingdom's energy consumption using machine learning and hybrid approaches'. In: *Energy & Environment* vol. 0, no. 0, p. 0958305X221140569. URL: https://doi.org/10.1177/0958305X221140569.

Bauer, P., Thorpe, A. and Brunet, G. (2015). 'The quiet revolution of numerical weather prediction'. In: *Nature (London)* vol. 525, no. 7567, pp. 47–55.

Bergmeir, C., Hyndman, R. J. and Koo, B. (2018). 'A note on the validity of cross-validation for evaluating autoregressive time series prediction'. In: *Computational Statistics & Data Analysis* vol. 120, pp. 70–83. URL: https://www.sciencedirect.com/science/article/pii/S0167947317302384.

Brajard, J. et al. (2023). 'Enhancing seasonal forecast skills by optimally weighting the ensemble from fresh data'. In: *Weather and Forecasting*. URL: https://journals.ametsoc.org/view/journals/wefo/aop/WAF-D-22-0166.1/WAF-D-22-0166.1.xml.

Butler, A. H. et al. (2017). 'A sudden stratospheric warming compendium'. In: *Earth System Science Data* vol. 9, no. 1, pp. 63–76.

Cade, B. S. and Noon, B. R. (2003). 'A Gentle Introduction to Quantile Regression for Ecologists'. In: *Frontiers in ecology and the environment* vol. 1, no. 8, pp. 412–420.

Cerqueira, V., Torgo, L. and Mozetič, I. (2020). 'Evaluating time series forecasting models: an empirical study on performance estimation methods'. In: *Machine learning* vol. 109, no. 11, pp. 1997–2028.

Cerqueira, V., Torgo, L. and Soares, C. (2023). 'Model Selection for Time Series Forecasting An Empirical Analysis of Multiple Estimators'. In: *Neural processing letters*.

Chen, T. (2014). *Introduction to Boosted Trees*. Accessed: June 1, 2023. URL: https://web.njit.edu/~usman/courses/cs675_summer20/BoostedTree.pdf.

Chen, T. and Guestrin, C. (2016). 'XGBoost: A Scalable Tree Boosting System'. In: *Proceedings of the 22nd ACM SIGKDD International Conference on knowledge discovery and data mining*. Vol. 13-17-. KDD '16. ACM, pp. 785–794.

Coninck, H. d., Vuuren, D. P. v. and al, et (2022). *Climate Change 2022: Impacts, Adaptation and Vulnerability: Summary for Policymakers*. IPCC. URL: https://www.ipcc.ch/report/ar6/wg3/.

De Felice, M., Alessandri, A. and Catalano, F. (2015). 'Seasonal climate forecasts for medium-term electricity demand forecasting'. In: *Applied Energy* vol. 137, pp. 435–444. URL: https://www.sciencedirect.com/science/article/pii/S030626191401071X.

Energy Facts Norway (2023). *Day-ahead market*. URL: https://www.nordpoolgroup.com/en/the-power-market/Day-ahead-market/ (visited on 03/03/2023).

Esmaeili, A. and Shokoohi, Z. (2011). 'Assessing the effect of oil price on world food prices: Application of principal component analysis'. In: *Energy Policy* vol. 39, no. 2. Special Section on Offshore wind power planning, economics and environment, pp. 1022–1025. URL: https://www.sciencedirect.com/science/article/pii/S0301421510008104.

Foldvik Eikeland, O. et al. (2021). 'Predicting Energy Demand in Semi-Remote Arctic Locations'. In: *Energies* vol. 14, no. 4. URL: https://www.mdpi.com/1996-1073/14/4/798.

Gelman, A. (2013). *Bayesian Data Analysis, Third Edition*. Boca Raton: CRC Press.

Gneiting, T. and Raftery, A. E. (2007). 'Strictly Proper Scoring Rules, Prediction, and Estimation'. In: *Journal of the American Statistical Association* vol. 102, no. 477, pp. 359–378.

Günay, M. E. (2016). 'Forecasting annual gross electricity demand by artificial neural networks using predicted values of socio-economic indicators and climatic conditions: Case of Turkey'. In: *Energy Policy* vol. 90, pp. 92–101. URL: https://www.sciencedirect.com/science/article/pii/S0301421515302329.

Hastie, T. J., Tibshirani, R. and Jerome, F. (2009). *The elements of statistical learning : data mining, inference, and prediction*. New York: Springer.

Heinrich, C. et al. (2021). 'Multivariate Postprocessing Methods for High-Dimensional Seasonal Weather Forecasts'. In: *Journal of the American*

Statistical Association vol. 116, no. 535, pp. 1048–1059. URL: https://doi.org/10.1080/01621459.2020.1769634.

Hemri, S. et al. (2020). 'How to create an operational multi-model of seasonal forecasts?' In: *Climate dynamics* vol. 55, no. 5-6, pp. 1141–1157.

Henze, J. et al. (2020). 'Probabilistic upscaling and aggregation of wind power forecasts'. In: *Energy, Sustainability and Society* vol. 10, no. 1, p. 15. URL: https://doi.org/10.1186/s13705-020-00247-4.

Hofmann, M. and Lindberg, K. B. (2019). 'Price elasticity of electricity demand in metropolitan areas – Case of Oslo'. In: *2019 16th International Conference on the European Energy Market (EEM)*, pp. 1–6.

Hor, C.-L., Watson, S. J. and Majithia, S. (2006). 'Daily Load Forecasting and Maximum Demand Estimation using ARIMA and GARCH'. In: *2006 International Conference on Probabilistic Methods Applied to Power Systems*, pp. 1–6.

Huang, J., Tang, Y. and Chen, S. (2018). 'Energy Demand Forecasting: Combining Cointegration Analysis and Artificial Intelligence Algorithm'. In: *Mathematical problems in engineering* vol. 2018, pp. 1–13.

Hyndman, R. J. and Fan, Y. (1996). 'Sample Quantiles in Statistical Packages'. In: *The American statistician* vol. 50, no. 4, pp. 361–365.

IEA (2022). *Securing Clean Energy Technology Supply Chains*. URL: https://www.iea.org/reports/securing-clean-energy-technology-supply-chains.

— (2023). *Europe's energy crisis: What factors drove the record fall in natural gas demand in 2022?* Accessed: 10/06/2023. URL: https://www.iea.org/commentaries/europe-s-energy-crisis-what-factors-drove-the-record-fall-in-natural-gas-demand-in-2022.

Jolliffe, I. T. (2002). *Principal component analysis*. New York: Springer.

Koenker, R. (2005). *Quantile regression*. Cambridge: Cambridge University Press.

Kristiansen, T. (2014). 'A time series spot price forecast model for the Nord Pool market'. In: *International Journal of Electrical Power & Energy Systems* vol. 61, pp. 20–26. URL: https://www.sciencedirect.com/science/article/pii/S0142061514001094.

Kuster, C., Rezgui, Y. and Mourshed, M. (2017). 'Electrical load forecasting models: A critical systematic review'. In: *Sustainable Cities and Society* vol. 35, pp. 257–270. URL: https://www.sciencedirect.com/science/article/pii/S2210670717305899.

Lay, D. C. (2021). *Linear algebra and its applications*. Harlow, Essex: Pearson Education.

Lean, P. et al. (2021). 'Continuous data assimilation for global numerical weather prediction'. In: *Quarterly Journal of the Royal Meteorological Society* vol. 147, no. 734, pp. 273–288. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3917.

Lindberg, K. et al. (2019). 'Long-term electricity load forecasting: Current and future trends'. In: *Utilities Policy* vol. 58, pp. 102–119. URL: https://www.sciencedirect.com/science/article/pii/S0957178719300116.

Livera, A. M. D., Hyndman, R. J. and Snyder, R. D. (2011). 'Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing'. In: *Journal of the American Statistical Association* vol. 106, no. 496, pp. 1513–1527. URL: https://doi.org/10.1198/jasa.2011.tm09771.

Malka, L. et al. (2023). 'Energy system analysis with a focus on future energy demand projections: The case of Norway'. In: *Energy* vol. 272, p. 127107. URL: https://www.sciencedirect.com/science/article/pii/S0360544223005017.

Mirasgedis, S. et al. (2006). 'Models for mid-term electricity demand forecasting incorporating weather influences'. In: *Energy* vol. 31, no. 2, pp. 208–227. URL: https://www.sciencedirect.com/science/article/pii/S0360544205000393.

Möller, A., Lenkoski, A. and Thorarinsdottir, T. L. (2013). 'Multivariate probabilistic forecasting using ensemble Bayesian model averaging and copulas'. In: *Quarterly Journal of the Royal Meteorological Society* vol. 139, no. 673, pp. 982–991. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2009.

Nord Pool (2023). *Day-ahead market*. URL: https://energifaktanorge.no/en/norsk-energiforsyning/kraftproduksjon/#hydropower (visited on 05/04/2023).

Nystrup, P. et al. (2021). 'Dimensionality reduction in forecasting with temporal hierarchies'. In: *International Journal of Forecasting* vol. 37, no. 3, pp. 1127–1146. URL: https://www.sciencedirect.com/science/article/pii/S0169207020301898.

Olatomiwa, L. et al. (2016). 'Energy management strategies in hybrid renewable energy systems: A review'. In: *Renewable & sustainable energy reviews* vol. 62, pp. 821–835.

Orlov, A., Sillmann, J. and Vigo, I. (2020). 'Better seasonal forecasts for the renewable energy industry'. In: *NATURE ENERGY* vol. 5, no. 2, pp. 108–110.

Oswald, Y., Owen, A. and Steinberger, J. K. (2020). 'Large inequality in international and intranational energy footprints between income groups and across consumption categories'. In: *NATURE ENERGY* vol. 5, no. 3, pp. 231–239.

Peng, R. D. (2022). *R Programming for Data Science*. URL: https://bookdown.org/rdpeng/rprogdatascience/.

Petropoulos, F. et al. (2022). 'Forecasting: theory and practice'. In: *International Journal of Forecasting* vol. 38, no. 3, pp. 705–871. URL: https://www.sciencedirect.com/science/article/pii/S0169207021001758.

Pinson, P. et al. (2009). 'From probabilistic forecasts to statistical scenarios of short-term wind power production'. In: *Wind energy (Chichester, England)* vol. 12, no. 1, pp. 51–62.

Pu, Z. and Kalnay, E. (2019). 'Numerical Weather Prediction Basics: Models, Numerical Methods, and Data Assimilation'. In: *Handbook of Hydrometeorological Ensemble Forecasting*. Ed. by Duan, Q. et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 67–97. URL: https://doi.org/10.1007/978-3-642-39925-1_11.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: https://www.R-project.org/.

Salinas, D. et al. (2019). *High-Dimensional Multivariate Forecasting with Low-Rank Gaussian Copula Processes*. URL: https://arxiv.org/abs/1910.03002.

Schuhen, N., Thorarinsdottir, T. L. and Lenkoski, A. (2020). 'Rapid adjustment and post-processing of temperature forecast trajectories'. In: *Quarterly Journal of the Royal Meteorological Society* vol. 146, no. 727, pp. 963–978. URL: https://doi.org/10.1002$%5C%$2Fqj.3718.

Shumway, R. H. and Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples*. 4th. New York: Springer. URL: https://www.springer.com/gp/book/9783319524511.

SSB (2014). *På verdenstoppen i bruk av strøm*. Accessed on 5-5-2023. URL: https://www.ssb.no/energi-og-industri/artikler-og-publikasjoner/pa-verdenstoppen-i-bruk-av-strom.

Steele, C. J. et al. (2014). 'Modelling sea-breeze climatologies and interactions on coasts in the southern North Sea: implications for offshore wind energy'. In: *Quarterly Journal of the Royal Meteorological Society* vol. 141, no. 690, pp. 1821–1835. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2484.

Tamba, J. G. et al. (2018). 'Forecasting natural gas: A literature survey'. In: *International Journal of Energy Economics and Policy* vol. 8, no. 3, p. 216.

Tedesco, P. et al. (2023). 'Gaussian copula modeling of extreme cold and weak-wind events over Europe conditioned on winter weather regimes'. In: *Environmental Research Letters* vol. 18, no. 3, p. 034008. URL: https://arxiv.org/abs/2209.12556.

Valor, E., Meneu, V. and Caselles, V. (2001). 'Daily Air Temperature and Electricity Load in Spain'. In: *Journal of applied meteorology (1988)* vol. 40, no. 8, pp. 1413–1421.

Vitart, F. and Robertson, A. W. (2019). 'Chapter 1 - Introduction: Why Sub-seasonal to Seasonal Prediction (S2S)?' In: *Sub-Seasonal to Seasonal Prediction*. Ed. by Robertson, A. W. and Vitart, F. Amsterdam: Elsevier, pp. 3–15. URL: https://www.sciencedirect.com/science/article/pii/B9780128117149000012.

Wilhite, H. et al. (1996). 'A cross-cultural analysis of household energy use behaviour in Japan and Norway'. In: *Energy Policy* vol. 24, no. 9, pp. 795–803. URL: https://www.sciencedirect.com/science/article/pii/0301421596000614.

Wilks, D. S. (2011). *Statistical methods in the atmospheric sciences*. Amsterdam: Elsevier.

Wood, S. N. (2017). *Generalized additive models : an introduction with R*. Boca Raton: Taylor & Francis.

Al-Yahyai, S., Charabi, Y. and Gastli, A. (2010). 'Review of the use of Numerical Weather Prediction (NWP) Models for wind energy assessment'. In: *Renewable and Sustainable Energy Reviews* vol. 14, no. 9, pp. 3192–3198. URL: https://www.sciencedirect.com/science/article/pii/S1364032110001814.