DEPARTMENT OF MATHEMATICS

Master's Degree in Mathematics

State-Space Models for Multivariate Electricity Load Forecasting

Supervisor:
Vinciotti Veronica

Co-Supervisor:
Mazuelas Santiago

Candidate:
Cartier van Dissel Mauritz

SIGNATURE

Academic Year 2021 - 2022
16th December 2022

Contents

Chapter 1

Introduction

The management of energetic resources has been one of the most important and controversial topics of 2022. The start of the Russian-Ukrainian conflict brought widespread consequences to the global energy economy that will last for a long period of time. Citing the International Energy Agency Executive Director, *"Energy markets and policies have changed as a result of Russia's invasion of Ukraine, not just for the time being, but for decades to come"*[1]. Furthermore, the economic effects are felt on all levels of society. Countries, especially in Europe, need to ensure the supply of the necessary energy resources from new providers, and the large increase in oil and electricity prices is posing a big threat to energy-greedy businesses and private households.

Nevertheless, the current energy crisis is not the only reason for concern. The main sources of electricity production are fossil fuels, for example, coal and natural gas (see Figure 1.1). These resources are limited and produce countless environmental issues, and therefore, a shift to a more sustainable energy supply is needed, especially to tackle the urgent issue of climate change. One of the obvious solutions would be to rely on renewable energy. However, the availability of electricity produced by renewables is not as reliable as for fossil fuels or other types of energy sources. Managing these resources requires a thorough analysis of the weather and environmental conditions and also of the energetic requirements of certain areas.

To solve these problems, intelligent management of electricity production is necessary. To help with operations and decision-making, electricity providers and policymakers rely on accurate demand and supply load forecasts, and these forecasts are especially important since electricity is a good that is difficult to store. In particular, for long-term decisions, which may include building a new hydroelectric power plant or relying on the import of electricity for the next 10 years, a long-horizon forecast is required. This kind of prediction, called Long-Term Load Forecast, is usually used for large areas and economic and socio-economic information is adopted.

Conversely, in this thesis, the focus is set on Short-Term Load Forecasting (STLF) algorithms. In this type of forecast, the prediction horizon is much shorter, and can usually go from one hour to two weeks ahead. These forecasts are pivotal to organising the electrical grid and making sure that the electricity generation matches the demand in the very near future, thus preventing useless waste of energy. In particular, Short-

[1]Fatih Birol, IEA Executive Director, during the press release on October 27th 2022.

Term Load Forecasting determines the amount of electricity that is generated in a given moment, and from a retailer's perspective, it regulates the purchase of electricity, while from a consumer perspective, individual load forecasting can reduce the cost of the electricity bill.

STLF has been studied thoroughly in the last decades and numerous papers have been published proposing innovative methods and revising the state-of-the-art, and statistical models or machine learning algorithms are commonly used for the prediction. Typically, these predictions consist of single-value forecasts, although it is becoming more common to find publications that focus on creating probabilistic forecasts, so that not only the load value is predicted, but the whole distribution as well. This enables the user to obtain confidence intervals and perform a probabilistic analysis of the forecast. Also, most algorithms rely on univariate load forecasts, even though predicting the load for more buildings or regions simultaneously may enhance the prediction accuracy.

All these considerations have led to an analysis of electricity load forecasting based on State-Space Models. These models are robust statistical tools that enable to forecast the load and enhance the predictions with the help of additional information that is available to the forecaster. In particular, this information can be in the form of meteorological data, electricity prices or even socio-economic factors. The first model that has been developed is the MAPLF model, which consists of a generalization of the model present in Álvarez et al. (2021) using multivariate Gaussian distributions, which enables the creation of multivariate probabilistic forecasts that are easy to use and analyse. Given the similarities that emerged with Kalman Filtering, a comparison between the two methods was conducted. Ultimately, a third novel forecasting method, named the Inverted State-Space Model, was developed and added to the comparison.

In the remaining sections of Chapter 1, an introductory analysis of electricity forecasting is proposed. In Section 1.1, a brief analysis of the electric power industry and the different types of energy forecasting is presented. Section 1.2 discusses the applications and the driving factors behind Short-Term Load Forecasting. Finally, in Section 1.3, several STLF techniques from the literature are described.

In Chapter 2, the State-Space Model framework and the forecasting algorithms are introduced. In Section 2.1, State-Space Models are defined, and the significance of the variables in load forecasting is shown. Section 2.2 shows the first forecasting technique, namely Kalman Filter, which is a widely used algorithm for prediction. Section 2.3 describes the second forecasting technique, MAPLF. In Section 2.4 the third forecasting technique is presented, which is another generalization of the Kalman Filter algorithm. In Section 2.5, the recursive Vector Autoregressive model of order 1 is introduced and will serve as a benchmark method. Then, Section 2.6 describes how the Gaussian distributions that are present in the models are learnt, with a focus on recursive online learning, and in Section 2.7, a brief analysis of the computational cost of the four forecasting algorithms is presented.

Chapter 3 applies the techniques that have been introduced in the previous chapter to a multiregional load forecasting scenario. In Sections 3.1 and 3.2, the ISO New England dataset is described and analysed, and an exploratory analysis of the relationships between variables is carried out. In Section 3.3, the evaluation metrics that will be used to analyse the forecasting algorithms are illustrated. Section 3.4 shows the results of

the forecasting techniques in a univariate scenario, in which the aggregated load has to be forecasted. Here, the impact of the type of day and of the meteorological variables is shown. In Section 3.5 the same procedures are applied to the multivariate dataset. First, the influence of global and regional temperatures is discussed. Then, an analysis of the accuracy of the algorithms is proposed. Section 3.6 shows how the accuracy of the multivariate forecasts changes when the forecast horizon is modified. Finally, in Section 3.7, the conclusions are drawn.

1.1 The electric power system

1.1.1 The electric power industry

Nowadays, the electric power industry has become a huge highly interconnected business. It comprises all the different units that compose the energy market, from the giant generators to the low-power electric devices, and all the different actors, from the big energy providers to the local consumers.

The electric power industry consists of four processes that enable the end user to have access to electricity.

Electricity generation

The production of electricity is the foundation of the whole electric power industry. It takes sources of primary energy and turns them into electric energy. For example, it can take raw resources like coal and turn them into energy thanks to combustion, or renewable resources like wind and create electricity via kinetic energy.

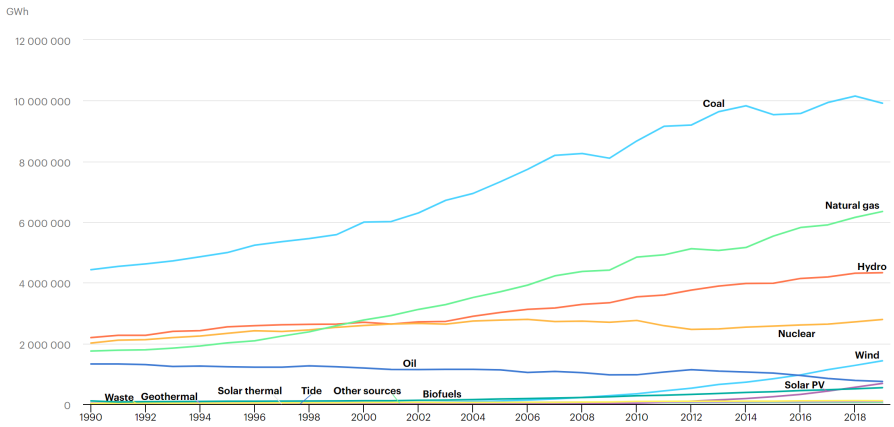

Figure 1.1: Global electricity generation by source, 1990-2019.
The figure has been obtained from the IEA official website.

Figure 1.1 shows the total electricity generation in the whole world from 1990 until 2019, and in particular, the figure shows the quantity (in GWh) that each source supplies to the global system. It is evident how fossil fuels like coal (especially in China and India) and natural gas make up the majority of today's global supply and this quantity is looking to increase in the near future. Renewable energy resources like hydro, wind, and solar are increasing their importance in the system, but they still take only a small percentage of the total supply.

Electric power transmission

Electric power transmission is the first step in the delivery of electricity from the power plants to the end users. The transmission line covers the segment from the generation facilities to the local substations. The transmission line is characterized by the use of

a high-voltage current, since it reduces electric losses significantly, especially for long distances. Not every power plant is connected to the transmission line. Usually, power stations that are located near cities are linked directly to the distribution line.

Electric power distribution

The distribution line is the second step of the delivery process, and it carries electricity from the electric substations to the consumers. This line is characterized by medium to low voltage and the current coming from the transmission line is converted in the substations using transformers. It consists of primary and secondary transmission lines. The first delivers medium voltage current from the transmission line to distribution transformers located near the customers or directly to high energy-demanding customers such as industries. The second carries the electricity with utilization voltage from the distribution transformers to the individual users.

Electricity retailing

The final step of the electric power industry is the sale of electricity to individual customers. This phase may be carried out by public or private organizations and the market can be regulated or deregulated. Historically this process was subject to monopoly regulations, but in the last few decades in many countries, there has been deregulation of the electricity market.

The retailing phase involves a big variety of consumers, from individual households to big industries. As shown by Figure 1.2, in the global electricity market, the biggest

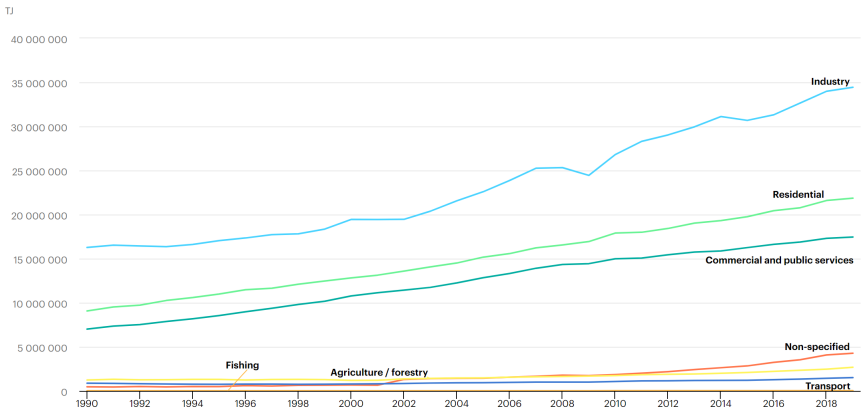

Figure 1.2: Global electricity consumption by sector, 1990-2019.
The figure has been obtained from the IEA official website.

share of electricity is sold to the industrial sector. The rest of the electricity is then mainly sold to the residential user and to commercial and public services. From this figure, we can also observe that the global electricity market is continuing to grow.

1.1.2 Types of energy forecasting

Managing such a complex system is a gigantic effort and it is a fundamental component of our everyday life. To supervise such a system, there is the need to provide various forecasts, in order to take actions and decisions regarding the provision of electricity more accurately.

Electric load forecasting

The most important type of prediction that is required in the electric power industry is the forecast of the electric power load. Since electricity is a good that is difficult to store, the quantity of electricity that is generated for a given moment in time should match the demand as accurately as possible. For this purpose, having solid forecasts for the future load demand allows to better organise energy production.

Load forecasting is typically classified into three classes depending on the time horizon of the prediction.

- **Long-term load forecasting** focuses on a one-year to ten years horizon. This type of forecast is typically used in electric power system planning for decisions about new power plants. For example, if the forecast shows a significant increase in electricity demand in the next years that cannot be covered with the current power generation facilities, a new power generation station needs to be constructed. In this case, forecasting also helps in the decision about the location of the plant, its size, and the source of energy that will be used. This type of forecast is usually used for decisions that involve large geographical areas and the external variables that are considered are for example the GDP, population rate increase, etc. (Seifi and Sepasian, 2011, Ch. 4).

- **Medium-term load forecasting** focuses on a monthly or yearly horizon. This type of forecast usually handles the monthly scheduling of maintenance operations and fuel provision. The maintenance of electricity generating facilities needs to be planned months ahead, in order to organise how to account for the power that will be missing. Fuel provision needs to be planned months ahead as well, and in a competitive market, it usually combines load forecasting together with price forecasting so that the revenue received from sales exceeds the cost of production (Wood et al., 2013, Ch. 12).

- **Short-term load forecasting** can be used on very different scales, and it ranges from forecasts that predict the load for the next minutes, or even seconds, to forecasts that have a daily or weekly horizon. These predictions are usually used to take actions regarding the current energy demand and supply.

Electricity price forecasting

Since the deregulation of the energy market, electric power providers had the necessity to acquire accurate price forecasts to manage their bidding strategy. Electricity is usually more costly when demand is high, i.e. during demand peaks, and is usually more affordable when the demand is low. Knowing where and especially when to buy

electricity from the producers can put a company at a significant advantage over its competitors. Moreover, price forecasting is useful also for entities that generate electricity for production scheduling.

The price of electricity is extremely unstable and this volatility is the reason why forecasts that increase the accuracy even by a small amount can generate important savings for the companies that operate in the electric market. There are different driving factors that influence the cost. The main one is probably the weather since cooling and heating are the most important uses of energy that will vary from day to day. Another important variable is the availability of renewable resources. Other causes include planned or unplanned outages, government regulations, etc.

Renewable power forecasting

As we mentioned above, the availability of renewable resources is one of the important variables in the prediction of electricity prices. Hence, predicting the amount of energy that will be produced by renewables such as wind and solar plays a very important role in the energy system. In contrast to load forecasting, in these models, the aim is to predict power generation instead of electricity demand.

Wind power forecasting is really important especially when wind farms generate a significant percentage of local power production. For example, if for a given day there is little wind, it is necessary to find alternative energy sources that substitute the amount of energy that is missing. As we may expect, wind power forecasting depends mainly on the weather conditions, and in particular, on wind speed and direction. Therefore, these forecasts depend heavily on weather forecasts, usually obtained thanks to Numerical Weather Prediction models, which are mathematical models that simulate the physics of the atmosphere and the oceans. Other models are usually based on statistical or machine-learning techniques.

Likewise, also solar power forecasting is very important for the management of electricity sources. Moreover, solar power is generated only during the daytime and since electricity demand often peaks during the evening, this source of energy cannot be used and other supply sources need to be found. Also, solar power predictions depend necessarily on the weather conditions, and in particular on the sun's path, the atmospheric conditions, and the light scattering. Furthermore, the amount of energy that is produced depends also on the conditions of the plant, and often a statistical model is used to predict the power output of a given photovoltaic power station using as input the raw solar resources.

1.2 Short-Term Load Forecasting

From now on, we will primarily focus on energy load forecasting. In particular, we will be looking at Short-Term Load Forecasting (STLF).

1.2.1 Applications

As we have seen in the previous section, load forecasting is fundamental for the management of the electric industry. Short-term load forecasting plays an especially important

role since the current managing operations of the electricity system depend highly on it. Short-term predictions can be used in numerous applications and, depending on their purpose, they can have different time horizons.

Automatic Generation Control serves as a tool to balance the generation of electricity from different plants and the total load demand in a very short time. The possible discrepancy between demand and generation is observed by measuring the power line frequency of the current. In order to re-balance the system, AGC takes into consideration many different aspects, from the type of power station to the cost of producing energy from a certain source. Among this information, STLF helps in deciding the generated output from each plant by predicting the load demand in the next minutes. In particular, a forecast horizon of 15 minutes, with time intervals of approximately 5 seconds is used.

When managing an electric power system, operations need to be planned ahead if possible, and Unit Commitment is part of this planning. Decision makers need to determine which plants should be active in the following days, and this will be done by taking into consideration the plants that are not available for scheduled maintenance or a sudden outage, the cost to maintain a certain plant active and the generation capacities of the plants. This planning is of crucial importance, mainly economically, since having more units that are committed than the ones that are needed has a significant cost. To decide which units to commit to at a certain hour of the day, it is important to know the load demand on the system for that hour, especially for peak hours. Since power plants take time to be activated, short-term load forecasts are used to predict the load demand for the next few days. The load forecast can be for a few days or even a few weeks, and the time interval is usually one hour.

There are other applications that may consider different time periods. Economic dispatch determines the optimal generated electrical output to match the demand, while at the same time minimizing the operational costs of the power plants. The horizon is usually of the next hour, while the interval is of 30 seconds. Optimal Power Flow serves the same purpose of balancing power generation and demand, taking into consideration the thermal limits of the transmission lines, together with voltage and electrical stability constraints. Also in this case STLF is useful and the forecast horizon is a maximum of two days, while the forecast interval is of about 5 minutes (Wood et al., 2013, Ch. 12).

1.2.2 Driving factors

Many different elements influence the electrical load demand. In Fahad and Arbab (2014) the authors summarise the most important factors that determine the quantity of electricity that is needed in the system.

Weather

Weather plays an important role in determining the need for electricity. This is because a large part of the electricity produced is used for either heating or cooling.

The temperature is probably the most meaningful meteorological variable. The needs for heating and cooling directly depend on the temperature. In winter, a decrease in temperature will correspond to an increase in load demand, while the opposite happens in summer. In fact, usually, a piecewise linear relationship is used to describe the

correlations between the load demand and the temperature (Fan et al., 2009). For hot days, a positive linear relationship is present, while for cold days, the linear relationship is negative. As stated by Fahad and Arbab (2014), the division between positive and negative correlation is around 20°C-25°C, while for Fan et al. (2009) it is around 15°C.

Humidity is another factor affecting load demand. The main consequence of humidity is making hot days be perceived as hotter, increasing the need for Air Conditioning, and equivalently making cold days be perceived as colder, increasing the quantity of power used for heating.

Precipitations also influence energy consumption. First of all, during a rainy day, people are keener to stay inside and therefore the electricity consumption for home entertainment will increase the overall demand. Secondly, the rain will usually decrease the temperature, leading to the consequences we described before. Also the wind plays a role. The presence of wind, and in particular its speed or the so-called wind chill index, may decrease the apparent temperature, thus increasing or decreasing the consumption depending on the season. Finally, also cloud cover may influence the load demand, reducing the temperature during the day, but trapping the heat during the night.

Time

Time is without a doubt the main factor that determines the need for electricity at a certain moment. As stated above, the temperature has a relevant effect on electric consumption. Therefore, seasons play an important role in the amount of electricity used for heating and cooling. Even events like the start of the school year may significantly influence the demand.

Another important time variable is the type of day: during work days, the total consumption is usually at its peak, while during weekends and holidays the demand is significantly lower. This is particularly true for the influence that industries and other companies have on total consumption since they usually operate only on weekdays. The opposite behaviour may be observed when looking at residential buildings, because during weekends and holidays people may stay at home for longer periods of time, leading to an increase in energy consumption.

Nevertheless, when doing short-term load forecasts, the hour of the day is the most important variable to consider. Load demand usually shows a daily cyclic pattern, in which the demand is low during the night when most people are asleep, whereas during the day consumption becomes much higher leading to one or more peaks. Forecasting the time and size of the peaks is one important aspect to focus on, because it is the moment of maximum demand, and the power generation facilities must be able to satisfy the need. The peaks depend on many conditions: for example, during winter, heating is needed the most during the morning and the evening, and the peaks are usually present in the evening, while during summer, the highest temperatures are present in the middle of the day, leading to peaks in electricity demand for cooling purposes. Also, the type of day and type of building affect the peaks: in a workday, residential consumption during the day is often close to zero, and the same happens in most industries during the evening and the night.

Economic factors and occasional events

When looking at the load demand, it becomes clear that also economic variables influence the system. First of all, the development and industrialization of a certain area affect consumption, since often industries are responsible for the biggest share of energy needs. Following the same reasoning, when looking at the demand for a single building also the type of construction plays an important role since industrial or agricultural buildings have different consumption patterns than residential ones.

Also the price of electricity is an essential part of the demand dynamic, and when the price is high, demand may be reduced, especially by domestic consumers. Furthermore, in cases in which there is time-of-use pricing, with more affordable costs during the night, users may adapt and decide to increase their consumption during these hours. This will flatten the curve, decreasing the usually big difference between day and night consumption.

Furthermore, occasional or unpredictable events can affect the load demand as well. For example, with the current electricity crisis of 2022 that followed the start of the Russian-Ukrainian conflict, electricity prices spiked to record high values, and consequently, many European countries are trying to overcome the related issues by cutting unnecessary electricity consumption. Other unpredictable events that could lead to sudden changes in electricity consumption are for example unforeseen outages, religious and cultural events like Christmas or special TV programs like the Super Bowl or the Champions League final.

In this thesis, we will use data on the electricity consumption taken from the New England (US) region, which we obtain from the ISO dataset that was used in Hong et al. (2019). Chapter 3 will present an exploratory analysis of the data, and in particular, we will analyse the load in different seasons and in different hours, and we will study the relationship between the load and the temperature.

1.3 STLF techniques

1.3.1 Single-value forecast

The literature about short-term load forecasting algorithms is huge and growing. There are many different techniques, each one having its strong and weak points. Historically, the techniques were developed and used to produce single-valued forecasts, while nowadays many algorithms are designed to output a probabilistic distribution. Let us start by focusing on single-value forecasts. No method has proven to be the best, and often the choice of the algorithm depends on its application. For example, in Jacob et al. (2020) the focus of the book is the development of forecasting algorithms that work well for individual buildings, and the authors display several state-of-the-art methods. Nevertheless, STLF is usually used to forecast the load of an area that comprises many different buildings since it is easier than doing it for a single building. This is because the aggregated curve is significantly smoother and more predictable. In Wang et al. (2019) and in Mirowski et al. (2014) the authors create a review about different data analytics that are performed using data collected from Smart Meters, and in particular, different

Load Forecasting methods are described. In Upadhaya et al. (2019) a brief systematic review of different STLF techniques is performed, analyzing more than a hundred related articles published on the IEEE portal. In Salleh et al. (2020) a systematic review of machine learning techniques is proposed, in which they analyze articles from IEEE, Science Direct and ResearchGate.

Single-valued forecast techniques are usually divided into statistical techniques and machine learning techniques. The first ones are simpler and more interpretable, while the second ones are more flexible and enable the modelling of non-linear relationships.

Linear models

Multiple linear regression models have been among the most used models for forecasting purposes, especially in the past. Their simplicity and their interpretability are the reasons they are still used today, and sometimes they manage to achieve relatively accurate results. The linear model is usually constructed using the current load as the response variable and several different variables as regressors. For example, temperature and other meteorological data may be used, together with information about past loads, the hour of the day, the type of day, etc.

AR, ARMA, SARIMA and similar models

One way to treat the problem of load forecasting is to look at the demand in the form of a time series. In contrast to linear models, this enables to account for the dependency between an observation and other observations in the past. Autoregressive (AR) models enable us to treat demand at a certain time point as a linear model of the p (where p is called the order of the AR model) previous observations. This allows us to better represent the daily cyclic pattern of load demand. Also, modifications of AR models are often used. For example, in Li et al. (2015) they use an AR model with a high order to model the time dependencies and afterwards they apply the LASSO regression algorithm to select only the most relevant times.

ARMA models are also used, and more generally Seasonal AutoRegressive Integrated Moving Average (SARIMA) models are usually adopted in this case, since these models are able to capture the daily pattern of the time series. Often, SARIMA models with exogenous variables (SARIMAX) are used, in order to add the dependency on other variables such as temperature, wind chill, etc. For example, in Kim and Kim (2021) the authors use ARIMA and VAR models to predict the loads, using also information from an exogenous variable which is the amount of internet trafficking.

Other modifications of time series models can be used. We just mentioned that in Kim and Kim (2021) they use a VAR model, that is a Vector Autoregressive model. These models are used when instead of a single time series, we want to model multiple time series at the same time, considering also the correlations between them. Other alterations can be found: for example in Jeong et al. (2021) a 24-hour prediction is performed in one step, modelling the dependencies between the different hours using a logistic mixture vector autoregressive model.

Exponential smoothing models

As described in Jacob et al. (2020), exponential smoothing models may also be used. In this case, differently from a moving average in which the past observations are weighted equally, we assign exponentially decreasing weights over time. This makes the model more dynamic to seasonal changes. Other types of exponential smoothing models can be used, like the double or triple exponential smoothing (the latter is also referred to as the Holt-Winters model) to model also intraday and intraweek seasonality.

Support Vector Regression

Let us now look at so-called machine learning techniques. Support Vector Regression is a tool that originates from Support Vector Machines, which are classification methods based on the maximum-margin separating hyperplane. In a regression context, SVMs search for the best function that approximates the observations with a certain error tolerance. Usually, input data is mapped into a higher dimensional feature space, so that non-linear solutions can be found. In Vrablecová et al. (2018), an online version of Support Vector Regression is used for the prediction of Smart Grid consumption.

Multi-layer Perceptron

Multi-layer perceptron (MLP) is the simplest example of an Artificial Neural Network. It is an acyclic neural network that links a layer of input nodes to a layer of output nodes. In between these two layers, some hidden layers are used. The more layers there are, the more deep the architecture and therefore the complexity of the model becomes. This may lead to more accurate results, but also to more parameters and thus a higher computational time. The weights of the layers are learnt via an algorithm that is called backpropagation, and non-linear relationships between the different layers are learnt using the so-called activation functions, for example, the Sigmoid or the ReLu functions. Jacob et al. (2020) describes briefly two load forecasting algorithms that use MLPs with 24 output layers to forecast the hourly load.

Recurrent Neural Networks

A more appropriate type of neural network for studying time series is the Recurrent Neural Network. These networks can contain cycles and are designed specifically to retrieve interesting information from past observations that can be useful to predict future values. In particular, Long Short-Term Memory (LSTM) RNNs are often used for this purpose. These models enable the learning of trends associated with human behaviour and this will often lead to good accuracy results. For example, in Guo et al. (2022) the authors construct different bidirectional LSTM models to predict different sources of energy, that are subsequently fed to a multi-task learning pipeline. The accuracy of this model is significantly higher compared to other state-of-the-art algorithms.

Other RNNs may be used. For example, instead of an LSTM, in Niu et al. (2022) the authors use a bidirectional Gated Recurrent Unit (GRU) with an attention mechanism, paired with a CNN, to produce accurate forecasts for Integrated Energy Systems.

1.3.2 Probabilistic forecast

As mentioned above, significant effort is now put into the development of techniques that output a probabilistic forecast instead of a single-valued one. Load demand is unpredictable in its nature, and the single value that is forecasted by many techniques is often unreliable. In fact, while load depends greatly on the time patterns such as time and season, short meteorological variations can strongly influence the accuracy of the prediction. This is why probabilistic forecasts are now preferred to single-value ones, especially for real-life applications. Nowadays, many industrial operations require probabilistic forecasts as input for their algorithms. For example, this type of prediction is used for probabilistic load flow analysis, unit commitment and for reliability planning (Hong and Fan, 2016). The output of a probabilistic forecast can be in the form of quantiles, intervals or density functions. A complete review of probabilistic electricity load forecasting has been published in Hong and Fan (2016), and this review also contains an analysis of single-value load forecasting. Another briefer review of probabilistic STLF techniques has been developed in Zhu et al. (2021), where the authors describe probabilistic forecasting methods for wind and solar power generation forecasts and for load forecasts.

Now we will describe two non-parametric approaches and one parametric approach that are widely used methods for probabilistic load forecasting.

Kernel density estimation

Kernel density estimation is a non-parametrical statistical technique that estimates the entire probability density function using so-called kernels as weighting functions. These methods can be unconditional, only using data from historical observations of the variable, or conditional, in which external variables such as day or temperature are used. An important analysis needs to be put into the choice of the kernel function, like for example Gaussian or biweight kernels, and also on its bandwidth (Jacob et al., 2020).

Quantile regression

Another non-parametric technique is given by quantile regression. This method is often used for regression when the assumptions of linear regression such as linearity, homoscedasticity, etc. are not satisfied. In these cases, the median is usually forecasted. Nonetheless, this method can be used to forecast any conditional quantile, and therefore using different quantiles, we can create different probabilistic forecasts. In Bracale et al. (2019) the authors propose a two-step online multivariate quantile regression algorithm.

State-space model

In paper Álvarez et al. (2021) called *Probabilistic Load Forecasting Based on Adaptive Online Learning*, which will be the reference paper for one of the methods developed in this thesis, the authors use a state-space model. This model is made of one observable sequence that contains variables such as weather conditions for a given time, and one hidden chain that represents the electricity load for that time. The two chains are then

modelled using two Gaussian probability distributions for each hour of the day, and the forecast is given by another Gaussian distribution with a certain mean and variance.

A similar approach is presented in Sharma et al. (2020), in which a so-called Blind Kalman-Filter algorithm is used to produce the Gaussian forecasts. This method is very similar to the Kalman Filter approach that will be defined in Section 2.2, with the slight difference that in this paper, the authors additionally use a Bayesian Smoothing step to update the predictions.

1.3.3 Qualitative forecast

As described in (Wood et al., 2013, Ch. 12), qualitative methods consist of forecasts that are based on the personal judgement or opinion of one or more experts. These methods are only used when little to no historical data is available and include subjective curve fitting, Delphi method, and technological comparisons. It may be used during unprecedented occurrences, like special events on TV.

1.3.4 Adaptive forecast

When developing a technique for load forecasting, training the model over the historic data and then predicting the future demand using always the same model can lead to inaccurate results. This is due to the fact that dynamic changes in consumption patterns cannot be detected using a static model. Adaptive forecasts overcome this issue by adjusting the model using the most recent available data. In this case, the training occurs online, which means that the parameters are updated continuously, in contrast to the previous static offline training procedure. As described in Alfares and Nazeeruddin (2002), Kalman Filtering applied to regression analysis is usually employed. When using a linear model, the Recursive Least Squares (RLS) algorithm has been adopted for online learning. Also Weighted RLS is often used, since it allows to put a higher weight on the most recent observations, enabling the model to adapt to the dynamic patterns.

In the probabilistic state-space models that we mentioned before Álvarez et al. (2021); Sharma et al. (2020), the parameters of the several Gaussian distributions are learnt online using a set of recursive equations. Also in Bracale et al. (2019) the parameters are learnt adaptively. In Obst et al. (2021) the authors use two adaptive algorithms based on Generalized Additive Models to forecast consumption during the COVID-19 lockdown period in France. At first, they construct the adaptive forecast using Kalman filters. Afterwards, they use consumption data during the lockdown period in Italy combined with transfer learning, to improve the forecast of the French demand. Thanks to the use of these adaptive models, they manage to adapt the predictions to the sudden change in consumption given by the pandemic.

1.3.5 Hierarchical forecast

With the huge amount of granular data that smart meters provide to today's electric companies, a deeper study of the influence of the different levels of aggregations can be pursued. In particular, smart meters collect data about the consumption of single households, and this information can then be aggregated into larger areas, such

as neighbourhoods, cities and regions. For these reasons, hierarchical load forecasting has emerged, in which the goal is to predict the load in larger areas using information contained in the data at a lower level of aggregation.

Single smart meter time series is usually highly volatile and skewed. To predict such a series, further effort should be put into establishing the right load forecasting method, especially to forecast the peak demand (Jacob et al., 2020). Aggregated series are much smoother, showing seasonality and weather-dependency patterns, and are therefore easier to predict. Nonetheless, when the goal is to predict the aggregated load, it is often best to use a lower level of aggregation. In fact, as described in Fan et al. (2009), when forecasting the load for a large geographical area, weather variations across different locations require the use of different models for different regions. Then the total aggregated forecast is given by the aggregation of the single forecasts.

In the Global Energy Forecasting Competition, hierarchical load forecasting has been one of the prevalent problems that have been proposed. In particular, both in the first edition in 2012 (Hong et al., 2014) and in the last one in 2017 (Hong et al., 2019), the participants were asked to create a hierarchical probabilistic forecast.

1.3.6 Multivariate forecast

Even though the data, especially the one collected from smart grids, contain historic loads for multiple buildings or regions, most techniques only consider the prediction models using one single time series at a time. Nevertheless, using information from other buildings/regions may be a good way to improve prediction accuracy. Predicting multiple loads simultaneously, by considering the information between similar users' behaviour, could lead to a significant improvement. For example, if we consider a house that has very noisy historical load data, the knowledge from other buildings in the same neighbourhood or from other people with a similar lifestyle could help in predicting the future load.

In this case, we are talking about multiregional load forecasting. Fan et al. (2009) explains very well the framework of having to forecast the load for multiple regions. Nonetheless, this paper does not consider the dependencies between the different locations. In Li et al. (2015) they improve the accuracy given by an AR + LASSO model by using the information given by one additional user's load, which is chosen among the pool of users using a significance test. In Fiot and Dinuzzo (2018) the authors solve MTLF using a multi-task learning procedure and by using kernel-based regression. In particular, the multi-task learning procedure is applied considering all the other houses at the same time, enabling the kernel-based method to learn also the correlations between the different buildings. Finally, in Bracale et al. (2019), a Multivariate Quantile Regression (MQR) algorithm is proposed. This method obtains a multivariate probabilistic forecast. Furthermore, the forecast is also adaptive, since they developed online refinement of the parameters.

Another interesting area in which multivariate forecasting is used is in Multi Energy Systems (MES), often referred to as Integrated Energy Systems (IES). In this case, the output of the forecast consists of different energy types instead of different regions. For example, in a typical situation, the output is given by the energetic load, the cooling load and the heating load. In Guo et al. (2022), at first, the authors look at the seasonal

MIC coefficient upon which they divide the three energy classes into 1 2 or 3 classes. Then the bidirectional LSTM is used to predict the outcome of these classes. In Chen and Wang (2021) a synergetic electric load forecasting formula was constructed and in Niu et al. (2022) a CNN-BiGRU model with an attention mechanism is used.

There are also other cases in which a multivariate forecast is useful. For example, in Lemos-Vinasco et al. (2021) the authors use multiple probabilistic models to analyze the temporal dependencies in a multiple hours forecast.

Chapter 2

Multivariate Load Forecasting using State-Space Models

In this chapter, three methods for load forecasting are presented. They are all based on Multivariate Linear Gaussian State-Space Models or modifications of such and the output of the forecast consists of the mean and the covariance matrix for the future load normal distribution. A fourth method, namely a Vector Autoregressive model, is added and will serve as a benchmark method. Furthermore, the learning of the parameters that are used to create the predictions is conducted online via a recursive method, to adapt to dynamic changes in the consumption patterns.

2.1 State-Space Models and general setting

The goal of Short Term Load Forecasting is to predict the load for the following days. In particular, in Probabilistic STLF, a complete or partially complete probability distribution is predicted. In this thesis, the forecasts will be produced using multiple Gaussian distributions that originate from State-Space Models or similar models. Therefore, we start with an exposition of State-Space Models (SSMs).

2.1.1 Discrete-time State-Space Models

Since the measurements that are used for load forecasting are sampled with a regular time interval between every two consecutive observations, for example, every 15 minutes or every hour, the focus is put only on discrete-time processes.

Definition 2.1.1 (Discrete-time State-Space Model)**.** *Let* $\{\mathbf{X}_t\}$ *and* $\{\mathbf{Y}_t\}$ *be two discrete-time stochastic processes with the variables* $\mathbf{X}_t$ *in a continuous state space.*
The pair $(\mathbf{X}_t, \mathbf{Y}_t)$ *is a* ***state-space model*** *if*

- $\{\mathbf{X}_t\}$ *is a Markov process;*
- $P(\mathbf{Y}_n \in B | \mathbf{X}_1 = \mathbf{x}_1, \ldots, \mathbf{X}_n = \mathbf{x}_n) = P(\mathbf{Y}_n \in B | \mathbf{x}_n = \mathbf{x}_n)$ *for every* $n \geq 1$*, and every Borel set* B*.*

In the literature, SSMs are often referred to as Hidden Markov Models (HMMs). Even though the structure is equivalent, according to Fahrmeir and Tutz (2001) the difference is that in an HMM, the process $\{\mathbf{X}_t\}$ is constructed over a finite state space instead of a continuous one.

To visualize the structure of a state-space model, we can use a Probabilistic Graphical Model. For $t = 1, \ldots, n$ the associated graph is depicted in Figure 2.1.

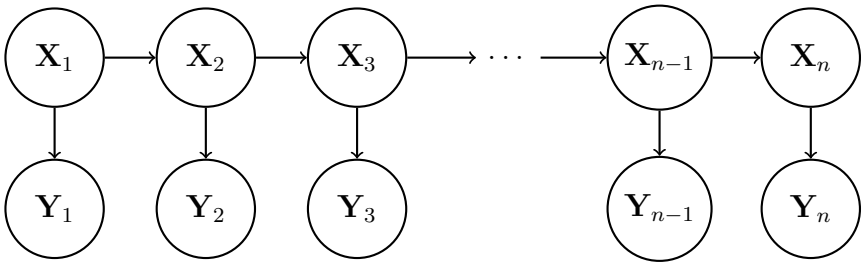

Figure 2.1: Conditional Independence graph of State-Space Model.

From this graph, the following conditional independence statements can be derived:

- $\mathbf{Y}_i \perp\!\!\!\perp \mathbf{Y}_j | \mathbf{X}_k$ for $\min(i, j) \leq k \leq \max(i, j)$;
- $\mathbf{X}_i \perp\!\!\!\perp \mathbf{X}_j | \mathbf{X}_k$ for $\min(i, j) < k < \max(i, j)$ and $|i - j| > 1$;
- $\mathbf{Y}_i \perp\!\!\!\perp \mathbf{X}_j | \mathbf{X}_k$ for $j < k \leq i$ if $j < i$, or $i \leq k < j$ if $j > i$.

Another property that originates from Graphical Models is the factorization of the joint distribution of the entire model.

$$\begin{aligned} p(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n, \mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n) &= p(\mathbf{X}_1)p(\mathbf{Y}_1|\mathbf{X}_1)p(\mathbf{X}_2|\mathbf{X}_1)p(\mathbf{Y}_2|\mathbf{X}_2)\ldots \\ &\quad \ldots p(\mathbf{Y}_{n-1}|\mathbf{X}_{n-1})p(\mathbf{X}_n|\mathbf{X}_{n-1})p(\mathbf{Y}_n|\mathbf{X}_n) \\ &= p(\mathbf{X}_1)p(\mathbf{Y}_1|\mathbf{X}_1)\prod_{i=2}^{n} p(\mathbf{X}_i|\mathbf{X}_{i-1})p(\mathbf{Y}_i|\mathbf{X}_i) \end{aligned}$$

It has been highlighted that when using State-Space Models, continuous random variables are employed. This means that in order to identify the model, different probability distributions are used.

Definition 2.1.2 (Transition distribution)**.** *The transition distribution* $\mathbf{X}_{n+1}|\mathbf{X}_n$ *models the relationship between the latent variable at time* $n + 1$ *and the one at time* n*. The probability density function* $p(\mathbf{x}_{n+1}|\mathbf{x}_n)$ *is called the transition density.*

Definition 2.1.3 (Emission distribution)**.** *The emission distribution models the relationship between the observable variable at time* n *and the latent variable at time* n*. The probability density function is called the emission density.*

2.1.2 SSMs for load forecasting

When using State-Space Models for electricity load forecasting, the latent variable at time t, $\mathbf{X}_t$, will consist of the multivariate load at time t. The variable space is $\mathbb{R}^d$, where $d \geq 1$ indicates the dimension of the load vector. The vector of loads may represent different scenarios:

- $\mathbf{X}_t$ may represent the vector of loads at time t in different household or regions (Fan et al., 2009);
- $\mathbf{X}_t$ may represent the vector of different load types at time t. For example, we could consider the triplet consisting of electric power load, cooling load and heating load (Guo et al., 2022);
- $\mathbf{X}_t$ may represent the vector of active and reactive power loads at time t (Bracale et al., 2020);
- $\mathbf{X}_t$ may represent the vector of minimum, maximum, and average load at time t (Hu et al., 2018).

The observed variable at time t, $\mathbf{Y}_t$, on the other hand, will indicate the vector of exogenous variables at time t. The variable has dimension $p \geq 1$, where p is the number of exogenous variables considered. The exogenous variables may consist of weather conditions or other information, and in practice, the choice of the variables usually depends on the data available. Furthermore, there may not be a linear relationship between the load and the exogenous variables. In this case, to have better results, a further study needs to be conducted. Depending on the model, a function of the exogenous variable or of the loads may be used instead, so that non-linear relationships may appear. In case the number of exogenous variables that are considered is too high, dimensionality reduction algorithms, such as PCA, may be applied to the vector of observed variables.

2.1.3 Calendar types

Both transition and emission distribution presented above will be modelled using time-dependent Multivariate Gaussian distributions, and the methods to learn the parameters will be explained in section 2.6. The objective is to learn time-specific dependencies, and for this purpose, a time non-homogeneous model is used. Therefore, the set of parameters of the two distributions depends on the time, and will therefore be different from one time step to another. This will allow the model to learn time-specific relations and adapt to dynamic changes. Nonetheless, since in a fully saturated non-homogeneous model there exists only a single observation for each time t, in order to learn the parameters from the available data, the model is simplified with the use of the set Δ, called the calendar types set.

A typical way to define the calendar types is to use the partition of a day given by the daily measurements that are present in the data. For example, in a case in which the dataset is made of hourly observations, the set $\Delta = \{1, 2, \ldots, 24\}$ is used. In this case, 24 sets of parameters are introduced, one for each hour of the day.

Additionally, one could also add information about the type of day to the calendar types set. For example, since consumption patterns change significantly between workdays and holidays, this information may be included. If the data is made of hourly measurements, 48 calendar types can be used and the set Δ will be equal to $\{(0,1),(0,2),\ldots,(0,24),(1,1),(1,2),\ldots,(1,24)\}$, where $(0,i)$ represents hour i for a workday and $(1,i)$ for a holiday.

2.1.4 Recursive predictions

When forecasting the future value of the load, we assume that the historic values of the load until the present state $\mathbf{X}_T$ are known. On the other hand, we assume that the observed variables are also known for R steps into the future, or can otherwise be predicted with high accuracy. For example, if the observed variables consist of meteorological data, the future values are obtained using accurate weather predictions. This means that the observed variables are assumed known until the future state $\mathbf{Y}_{T+R}$.

The precise goal of Load Forecasting is to predict the load distribution at L steps into the future, for $L \leq R$. In particular, we are interested in predicting the load by knowing all the historic data of the load up to time T and all the observed variables up to time $T+L$. To simplify the notation, let $\mathbf{X}_{1:k}$ denote the union of the variables $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_k$ and analogously, $\mathbf{Y}_{1:k} = \mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_k$. Thus, the objective is to estimate the target distribution

$$p(\mathbf{X}_{T+L}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+L})$$

Since Multivariate Normal Distributions are employed, the precise goal is to estimate the mean and the covariance matrix of this distribution. To do this, a recursive approach will be used. That is, we will assume that for each $k = T+1, \ldots, T+L$, the distribution of $p(\mathbf{X}_{k-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:k-1})$ is known and we will derive and use time-specific normal distributions to estimate the parameters for the target distribution. In this thesis, we will consider four methods, namely:

1. Kalman Filter;
2. A generalization in a multivariate setting of the APLF method proposed in Álvarez et al. (2021);
3. A new model in which the arrows that are present in the State-Space Models between the latent and observed variables, as depicted in 2.1, are reversed;
4. A simple Vector Autoregressive model of order 1.

Finally, we assume that to start the recursion the distribution of the 0-th step ahead is $p(\mathbf{X}_T|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T})$ with known mean $\mathbf{X}_T$ and zero covariance matrix.

2.1.5 Lemmas

In this subsection, three Lemmas are introduced that will be used in the forecasting procedures. The proofs of the Lemmas are presented in Appendix A.1.

Lemma 2.1.1 (Conditional distribution of a Multivariate Gaussian distribution)**.** *Let us consider a multivariate Gaussian distribution of the form*

$$\begin{bmatrix}\mathbf{X}\\ \mathbf{Y}\end{bmatrix} \sim \mathcal{N}_{k_1+k_2}\left(\begin{bmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right)$$

where $\mathbf{X}$ *has dimension* k_1 *and* $\mathbf{Y}$ *has dimension* k_2*.*

Then, the conditional distribution of $\mathbf{X}$ *given* $\mathbf{Y}$ *is:*

$$\mathbf{X}|\mathbf{Y} = \mathbf{y} \sim \mathcal{N}_{k_1}\left(\boldsymbol{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{y} - \boldsymbol{\mu}_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right).$$

Lemma 2.1.2 (Product of Gaussian pdfs of related variables)**.** *Let us consider two multivariate normal probability distribution functions of the form:*

- $\phi_{k_1}(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1)$
- $\phi_{k_2}(\mathbf{y}; M\mathbf{x} + \boldsymbol{\mu}_2, \Sigma_2)$.

Then, we have that the product between these two pdfs is equal to:

$$\phi_{(k_1+k_2)}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}; \begin{bmatrix}\boldsymbol{\mu}_1\\ M\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2\end{bmatrix}, \begin{bmatrix}\Sigma_1 & \Sigma_1 M^t\\ M\Sigma_1 & \Sigma_2 + M\Sigma_1 M^t\end{bmatrix}\right).$$

Lemma 2.1.3 (Product of Gaussian pdfs of the same variable)**.** *Let us consider two multivariate normal probability density functions of the form:*

- $\phi_k(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1)$
- $\phi_k(\mathbf{x}; \boldsymbol{\mu}_2, \Sigma_2)$.

Then, we have that the product between the two pdfs is equal to:

$$\phi_k\left(\mathbf{x}; \boldsymbol{\mu}^*, \Sigma^*\right) \cdot \phi_k(\boldsymbol{\mu}_1; \boldsymbol{\mu}_2, \Sigma_1 + \Sigma_2)$$

where $\Sigma^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$ *and* $\boldsymbol{\mu}^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2)$.

2.2 Kalman Filter

The first prediction method to be analysed is the so-called Kalman Filter (KF). KF have been first introduced by Rudolf E. Kalman and is nowadays a consolidated method for prediction. In the context of a Linear Gaussian State-Space Model, Kalman Filter is a recursive algorithm that for each time-step works in two phases: a prediction phase and a correction phase. In the prediction phase, an estimate for the next value of the latent variable is computed, together with its uncertainties that are represented by the covariance matrix. Afterwards, this estimate is updated using information from the observed variables, to produce a corrected estimate for the latent variable. The two steps are discussed in detail in Subsection 2.2.2.

2.2.1 Linear Gaussian State-Space Model

For the case of Linear Gaussian State-Space Models, both the transition and the emission distributions are modelled using a Gaussian distribution. When presenting the two normal distributions, to simplify the notation we will not use the calendar type-specific parameters that have been introduced in Subsection 2.1.3. Nevertheless, we may assume that parameters with index n are in practice parameters with index $c(n)$, where c maps each time step to the corresponding calendar type.

Transition distribution

To model the transition distribution, the following multivariate normal distribution is used:

$$\mathbf{X}_n|\mathbf{X}_{n-1} = \mathbf{x}_{n-1} \sim \mathcal{N}_d(A_n\mathbf{x}_{n-1} + \boldsymbol{\alpha}_n, Q_n)$$

with A_n the $d \times d$ transition matrix, $\boldsymbol{\alpha}_n$ the transition intercept and Q_n the $d \times d$ transition covariance matrix.

Let $\phi_d(\mathbf{x}; \dots, \dots)$ indicate the probability density function for a d - dimensional multivariate normal distribution. The transition density is given by:

$$\begin{aligned} &p(\mathbf{x}_n|\mathbf{x}_{n-1}) = \phi_d(\mathbf{x}_n; A_n\mathbf{x}_{n-1} + \boldsymbol{\alpha}_n, Q_n) \\ &= \left((2\pi)^d \cdot |Q_n|\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x}_n - A_n\mathbf{x}_{n-1} - \boldsymbol{\alpha}_n)^t Q_n^{-1}(\mathbf{x}_n - A_n\mathbf{x}_{n-1} - \boldsymbol{\alpha}_n)\right). \end{aligned}$$

Emission distribution

The emission distribution will use a multivariate normal distribution as well. In particular, $\{\mathbf{Y}_t\}$ will be considered as a continuous random variable, and a Gaussian distribution is used to model $\mathbf{Y}_n|\mathbf{X}_n$, namely:

$$\mathbf{Y}_n|\mathbf{X}_n = \mathbf{x}_n \sim \mathcal{N}_p(B_n\mathbf{x}_n + \boldsymbol{\beta}_n, R_n)$$

with B_n the $p \times d$ emission matrix, $\boldsymbol{\beta}_n$ the emission intercept and R_n the $p \times p$ emission covariance matrix.

The emission density is therefore given by:

$$\begin{aligned} &p(\mathbf{y}_n|\mathbf{x}_n) = \phi_p(\mathbf{y}_n; B_n\mathbf{x}_n + \boldsymbol{\beta}_n, R_n) \\ &= \left((2\pi)^p \cdot |R_n|\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{y}_n - B_n\mathbf{x}_n - \boldsymbol{\beta}_n)^t R_n^{-1}(\mathbf{y}_n - B_n\mathbf{x}_n - \boldsymbol{\beta}_n)\right). \end{aligned}$$

2.2.2 Forecasting

As we mentioned previously, we assume that the values of the load are known up to time T and the values of the observed variables are known up to time $T + R$. Then the goal is to predict the target distribution

$$p(\mathbf{X}_{T+L}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+L})$$

for a certain $1 \leq L \leq R$, and since the distribution is Gaussian, the goal is to determine its mean $\hat{\boldsymbol{\mu}}_{T+L}$ and its covariance $\hat{\Sigma}_{T+L}$.

Since we will use a recursive procedure, we now focus on the recursive prediction steps. For $k = T+1, \ldots, T+L$, the target distribution of the previous time step, $p(\mathbf{X}_{k-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:k-1})$, is assumed to be known, with mean $\hat{\boldsymbol{\mu}}_{k-1}$ and covariance $\hat{\Sigma}_{k-1}$. The goal is to deduce the distribution of $p(\mathbf{X}_k|\mathbf{X}_{1:T}, \mathbf{Y}_{1:k})$, and since $\mathbf{X}_T$ is completely known, the initial distribution is degenerate with mean $\hat{\boldsymbol{\mu}}_T = \mathbf{X}_T$ and a zero $d \times d$ covariance matrix.

With these assumptions, we have that for prediction step k,

$$\begin{aligned}
p(\mathbf{x}_{k-1}|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= \phi_d\big(\mathbf{x}_{k-1}; \hat{\boldsymbol{\mu}}_{k-1}, \hat{\Sigma}_{k-1}\big) \\
p(\mathbf{x}_k|\mathbf{x}_{k-1}) &= \phi_d\big(\mathbf{x}_k; A_k\mathbf{x}_{k-1} + \boldsymbol{\alpha}_k, Q_k\big) \\
p(\mathbf{y}_k|\mathbf{x}_k) &= \phi_p\big(\mathbf{y}_k; B_k\mathbf{x}_k + \boldsymbol{\beta}_k, R_k\big).
\end{aligned}$$

We will now derive the target distribution at k, and we divide the derivation into four steps. To simplify the notation, from now on the indices of the parameter k and $k-1$ will be dropped.

First step

The joint distribution of $\mathbf{x}_{k-1}, \mathbf{x}_k$ given all the past loads is equal to:

$$\begin{aligned}
p(\mathbf{x}_{k-1}, \mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= p(\mathbf{x}_{k-1}|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{x}_{k-1}, \mathbf{y}_{1:k-1}) \\
&= p(\mathbf{x}_{k-1}|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})p(\mathbf{x}_k|\mathbf{x}_{k-1}) \\
&= \phi_d\left(\mathbf{x}_{k-1}; \hat{\boldsymbol{\mu}}, \hat{\Sigma}\right)\phi_d\left(\mathbf{x}_k; A\mathbf{x}_{k-1} + \boldsymbol{\alpha}, Q\right).
\end{aligned}$$

Then, from Lemma 2.1.2,

$$p(\mathbf{x}_{k-1}, \mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) = \phi_{2d}\left(\begin{bmatrix}\mathbf{x}_{k-1}\\ \mathbf{x}_k\end{bmatrix}; \begin{bmatrix}\hat{\boldsymbol{\mu}}\\ A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}\end{bmatrix}, \begin{bmatrix}\hat{\Sigma} & \hat{\Sigma}A^t\\ A\hat{\Sigma} & Q + A\hat{\Sigma}A^t\end{bmatrix}\right).$$

Second step

The second step consists in deleting the dependency on $\mathbf{X}_{k-1}$ by marginalizing the distribution.

$$\begin{aligned}
p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= \int_{\mathbb{R}^d} p(\mathbf{x}_{k-1}, \mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})d\mathbf{x}_{k-1} \\
&= \int_{\mathbb{R}^d} \phi_{2d}\left(\begin{bmatrix}\mathbf{x}_{k-1}\\ \mathbf{x}_k\end{bmatrix}; \begin{bmatrix}\hat{\boldsymbol{\mu}}\\ A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}\end{bmatrix}, \begin{bmatrix}\hat{\Sigma} & \hat{\Sigma}A^t\\ A\hat{\Sigma} & Q + A\hat{\Sigma}A^t\end{bmatrix}\right) d\mathbf{x}_{k-1} \\
&= \phi_d\left(\mathbf{x}_k; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}, Q + A\hat{\Sigma}A^t\right).
\end{aligned}$$

The distribution we have obtained can be used to obtain an estimate at the so-called prediction phase.

Third step

Now we introduce observation $\mathbf{Y}_k$.

$$\begin{aligned} p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= p(\mathbf{x}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) p(\mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{x}_k, \mathbf{y}_{1:k-1}) \\ &= p(\mathbf{x}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) p(\mathbf{y}_k | \mathbf{x}_k) \\ &= \phi_d \left(\mathbf{x}_k; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}, Q + A\hat{\Sigma}A^t \right) \phi_p \left(\mathbf{y}_k; B\mathbf{x}_k + \boldsymbol{\beta}, R \right) \end{aligned}$$

and again using Lemma 2.1.2, $p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})$ is equal to:

$$\phi_{d+p} \left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{y}_k \end{bmatrix}; \begin{bmatrix} A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} \\ B(A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}) + \boldsymbol{\beta} \end{bmatrix}, \begin{bmatrix} Q + A\hat{\Sigma}A^t & (Q + A\hat{\Sigma}A^t)B^t \\ B(Q + A\hat{\Sigma}A^t) & R + B(Q + A\hat{\Sigma}A^t)B^t \end{bmatrix} \right).$$

Fourth step

Finally, we obtain the target distribution by conditioning the distribution we just found on $\mathbf{Y}_k$ using Lemma 2.1.1. We find that distribution $p(\mathbf{x}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k})$ has mean $\hat{\boldsymbol{\mu}}_k$ equal to

$$A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} + (Q + A\hat{\Sigma}A^t)B^t \left(R + B(Q + A\hat{\Sigma}A^t)B^t \right)^{-1} (\mathbf{y}_k - B(A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}) - \boldsymbol{\beta})$$

and covariance $\hat{\Sigma}_k$ equal to

$$(Q + A\hat{\Sigma}A^t) - (Q + A\hat{\Sigma}A^t)B^t \left(R + B(Q + A\hat{\Sigma}A^t)B^t \right)^{-1} B(Q + A\hat{\Sigma}A^t).$$

This is the final distribution, and the last two steps make up the so-called correction phase.

Literature notation

To have a coherent notation with the literature about Kalman Filtering, we now rewrite the results using a similar notation to the one used in (Fahrmeir and Tutz, 2001, Section 8.1.2). The mean and covariance of the target distribution at time step k are $\hat{\boldsymbol{\mu}}_{k|k}$ and$\hat{\Sigma}_{k|k}$, and the prediction step is given by

- $\hat{\boldsymbol{\mu}}_{k|k-1} = A\hat{\boldsymbol{\mu}}_{k-1|k-1} + \boldsymbol{\alpha}$
- $\hat{\Sigma}_{k|k-1} = Q + A\hat{\Sigma}_{k-1|k-1}A^t$.

Now, the so-called Kalman gain is introduced to determine the correction step.

$$K = \hat{\Sigma}_{k|k-1}B^t \left(R + B\hat{\Sigma}_{k|k-1}B^t \right)^{-1}.$$

From here, the correction step becomes:

- $\hat{\boldsymbol{\mu}}_{k|k} = \hat{\boldsymbol{\mu}}_{k|k-1} + K\left(\mathbf{y}_k - B\hat{\boldsymbol{\mu}}_{k|k-1} - \boldsymbol{\beta}\right)$
- $\hat{\Sigma}_{k|k} = \hat{\Sigma}_{k|k-1} - KB\hat{\Sigma}_{k|k-1}$.

Simplification

To simplify notation, we may apply the Woodbury Matrix identity

$$(D + V^t C V)^{-1} = D^{-1} - D^{-1}V^t(C^{-1} + VD^{-1}V^t)^{-1}VD^{-1}$$

to the covariance matrix. By choosing $D^{-1} = (Q + A\hat{\Sigma}A^t), C^{-1} = R$ and $V = B$, we have that the covariance can be rewritten more compactly as

$$\hat{\Sigma}_k = ((Q + A\hat{\Sigma}A^t)^{-1} + B^t R^{-1} B)^{-1}.$$

To rewrite the mean, we now consider formula (158) from Petersen et al. (2012). Since both $\hat{\Sigma}_{k|k-1}$ and R are positive definite, we can rewrite K as

$$K = \left(\hat{\Sigma}_{k|k-1}^{-1} + B^t R^{-1} B\right)^{-1} B^t R^{-1} = \hat{\Sigma}_k B^t R^{-1}.$$

From this, we can rewrite

$$\hat{\boldsymbol{\mu}}_k = A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} + \hat{\Sigma}_k B^t R^{-1}\big(\mathbf{y}_k - B(A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}) - \boldsymbol{\beta}\big).$$

Finally, we introduce Theorem 2.2.1, which summarizes the Kalman Filter forecasting procedure.

Theorem 2.2.1 (Kalman Filters L-step ahead forecast)**.** *Let us consider the SSM* $(\mathbf{X}_t, \mathbf{Y}_t)$ *and let us suppose that the values of* $\mathbf{X}_{1:T}$ *and* $\mathbf{Y}_{1:T+L}$ *are known. Then, the distribution of* $\mathbf{X}_{T+L}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+L}$ *can be found recursively.*

Let $j = 1, \ldots, L$ *and suppose that the distribution* $\mathbf{X}_{T+j-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j-1}$ *is known and normal with mean* $\hat{\boldsymbol{\mu}}_{T+j-1}$ *and covariance matrix* $\hat{\Sigma}_{T+j-1}$*. In particular for* $j = 1$*,* $\hat{\Sigma}_T = \mathbf{0}$ *and* $\hat{\boldsymbol{\mu}}_T = \mathbf{X}_T$*.*

Then, letting $r = c(T+j)$*, the distribution of* $\mathbf{X}_{T+j}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j}$ *can be obtained with the following formulas recursively. The covariance is*

$$\hat{\Sigma}_{T+j} = \left(\left(Q_r + A_r\hat{\Sigma}_{T+j-1}A_r^t\right)^{-1} + B_r^t R_r^{-1} B_r\right)^{-1}$$

and the mean is

$$\hat{\boldsymbol{\mu}}_{T+j} = A_r\hat{\boldsymbol{\mu}}_{T+j-1} + \boldsymbol{\alpha}_r + \hat{\Sigma}_{T+j}B_r^t R_r^{-1}\big(\mathbf{y}_{T+j} - B_r(A_r\hat{\boldsymbol{\mu}}_{T+j-1} + \boldsymbol{\alpha}_r) - \boldsymbol{\beta}_r\big).$$

2.3 MAPLF

As a second approach for forecasting, we propose a generalization to a multivariate setting of the techniques used in paper Álvarez et al. (2021). The original framework is called Adaptive Probabilistic Load Forecasting (APLF). The generalization that is proposed will be called Multivariate Adaptive Load Forecasting, in short MAPLF.

2.3.1 Model

We will consider the same State-Space Model of Section 2.1, depicted in Figure 2.1. The only difference with the previous Gaussian Linear State-Space Model lies in the emission distribution. In fact, under some prior assumptions, we model the distribution of $\mathbf{Y}_t|\mathbf{X}_t$ indirectly, by modelling the distribution of $\mathbf{X}_t|\mathbf{Y}_t$ using a Gaussian distribution. This will allow the model to use the load as the response variable, with other external factors as regressors. This may be more intuitive, since the load depends on external variables such as the weather, while the opposite is not true. For example, it is reasonable to assume that the weather is causally independent of the load demand. Furthermore, this change in dependencies also allows looking for particular non-linear relationships between the load and the other variables. For example, it is possible to model the temperature quadratically by adding the square of the temperature to the observed variables. Finally, it enables the use of non-continuous random variables as observed variables.

Emission distribution

The transition distribution of the model, $\mathbf{X}_n|\mathbf{X}_{n-1}$, remains the same as in 2.2.1. The emission distribution on the other hand will model $\mathbf{X}_n|\mathbf{Y}_n$ using a multivariate Gaussian distribution.

$$\mathbf{X}_n|\mathbf{Y}_n = \mathbf{y}_n \sim \mathcal{N}_d(D_n\mathbf{y}_n + \boldsymbol{\delta}_n, P_n)$$

with D_n the $d \times p$ emission matrix, $\boldsymbol{\delta}_n$ the emission intercept and P_n the $d \times d$ emission covariance matrix.

The emission density will be:

$$\begin{aligned} &p(\mathbf{x}_n|\mathbf{y}_n) = \phi_p(\mathbf{x}_n; D_n\mathbf{y}_n + \boldsymbol{\delta}_n, P_n) \\ &= \left((2\pi)^d \cdot |P_n|\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x}_n - D_n\mathbf{y}_n - \boldsymbol{\delta}_n)^t P_n^{-1}(\mathbf{x}_n - D_n\mathbf{y}_n - \boldsymbol{\delta}_n)\right). \end{aligned}$$

2.3.2 Forecasting

Let us start by taking the same assumptions as for Kalman Filters. This means that

- the values of the load are known up to time T and the values of the observed variables are known up to time $T + R$;
- for $k = T + 1$, $p(\mathbf{X}_{k-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:k-1}) = \phi_d\left(\mathbf{X}_{k-1}; \mathbf{X}_T, 0\right)$;
- for $k = T + 1, \ldots, T + L$, with $L \leq R$, we have:

$$\begin{aligned} p(\mathbf{x}_{k-1}|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= \phi_d\left(\mathbf{x}_{k-1}; \hat{\boldsymbol{\mu}}_{k-1}, \hat{\Sigma}_{k-1}\right) \\ p(\mathbf{x}_k|\mathbf{x}_{k-1}) &= \phi_d\left(\mathbf{x}_k; A_k\mathbf{x}_{k-1} + \boldsymbol{\alpha}_k, Q_k\right) \\ p(\mathbf{x}_k|\mathbf{y}_k) &= \phi_d\left(\mathbf{x}_k; D_k\mathbf{y}_k + \boldsymbol{\delta}_k, P_k\right). \end{aligned}$$

  To simplify the notation, from now on the indices of the parameter k and $k-1$ will be dropped.

Now, the goal is to find the distribution of $p(\mathbf{X}_k|\mathbf{X}_{1:T}, \mathbf{Y}_{1:k})$. The first two steps we take are the same that have been taken in the prediction phase of the Kalman Filter. Therefore, by applying the same rationale, we obtain the following distribution:

$$p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) = \phi_d\left(\mathbf{x}_k; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}, Q + A\hat{\Sigma}A^t\right).$$

Now, to allow the model to use $p(\mathbf{X}_k|\mathbf{Y}_k)$, we first look at the Bayes theorem.

$$p(\mathbf{Y}_k|\mathbf{X}_k) = p(\mathbf{X}_k|\mathbf{Y}_k)\frac{p(\mathbf{Y}_k)}{p(\mathbf{X}_k)}.$$

Since $\mathbf{Y}_k$ is known, we can consider that $p(\mathbf{Y}_k)$ is a constant. Furthermore, we assume that $p(\mathbf{X}_k)$ can be modelled using an improper prior. This will permit to interchange $p(\mathbf{Y}_k|\mathbf{X}_k)$ and $p(\mathbf{X}_k|\mathbf{Y}_k)$ since

$$p(\mathbf{Y}_k|\mathbf{X}_k) \propto p(\mathbf{X}_k|\mathbf{Y}_k).$$

Let us turn back to the target distribution. We observe that

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k}) &= \frac{p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})} \\ &= \frac{p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_{1:T}, \mathbf{x}_k, \mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})} \\ &= \frac{p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_k)}{p(\mathbf{y}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})}. \end{aligned}$$

Now, considering the distribution at the bottom as a constant, since all the variables involved are known, and using the fact that $p(\mathbf{y}_k|\mathbf{x}_k) \propto p(\mathbf{x}_k|\mathbf{y}_k)$:

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k}) &\propto p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k-1})p(\mathbf{x}_k|\mathbf{y}_k) \\ &= \phi_d\left(\mathbf{x}_k; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}, Q + A\hat{\Sigma}A^t\right)\phi_d\left(\mathbf{x}_k; D\mathbf{y}_k + \boldsymbol{\delta}, P\right). \end{aligned}$$

Using Lemma 2.1.3, we can rewrite this product as

$$p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k}) = \phi_d\left(\mathbf{x}_k; \hat{\boldsymbol{\mu}}_k, \hat{\Sigma}_k\right) \cdot \phi_d(D\mathbf{y}_k + \boldsymbol{\delta}; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha}, Q + A\hat{\Sigma}A^t + P)$$

where

$$\hat{\Sigma}_k = (P^{-1} + (Q + A\hat{\Sigma}A^t)^{-1})^{-1}$$

and

$$\hat{\boldsymbol{\mu}}_k = \hat{\Sigma}_k\left[P^{-1}(D\mathbf{y}_k + \boldsymbol{\delta}) + (Q + A\hat{\Sigma}A^t)^{-1}(A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha})\right].$$

Then, since $\mathbf{y}_k$ is known, the distribution on the right is a constant. Finally, we are left with

$$p(\mathbf{x}_k|\mathbf{x}_{1:T}, \mathbf{y}_{1:k}) \propto \phi_d\left(\mathbf{x}_k; \hat{\boldsymbol{\mu}}_k, \hat{\Sigma}_k\right)$$

and the proportionality operation becomes an equality.

To summarize all the operations, we introduce Theorem 2.3.1.

Theorem 2.3.1 (MAPLF L-step ahead forecast)**.** *Let us consider the SSM* $(\mathbf{X}_t, \mathbf{Y}_t)$ *and let us suppose that the values of* $\mathbf{X}_{1:T}$ *and* $\mathbf{Y}_{1:T+L}$ *are known. Then, the distribution of* $\mathbf{X}_{T+L}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+L}$ *can be found recursively.*

Let $j = 1, \dots, L$ *and suppose that the distribution* $\mathbf{X}_{T+j-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j-1}$ *is known and normal with mean* $\hat{\boldsymbol{\mu}}_{T+j-1}$ *and covariance matrix* $\hat{\Sigma}_{T+j-1}$*. In particular for* $j = 1$*,* $\hat{\Sigma}_T = \mathbf{0}$ *and* $\hat{\boldsymbol{\mu}}_T = \mathbf{X}_T$*.*

Then, letting $r = c(T+j)$*, the distribution of* $\mathbf{X}_{T+j}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j}$ *can be obtained with the following formulas recursively. The covariance is*

$$\hat{\Sigma}_{T+j} = \left(\left(Q_r + A_r \hat{\Sigma}_{T+j-1} A_r^t\right)^{-1} + P_r^{-1}\right)^{-1}$$

and the mean is

$$\hat{\boldsymbol{\mu}}_{T+j} = \hat{\Sigma}_{T+j}\Big[P_r^{-1}(D_r \mathbf{y}_{T+j} + \boldsymbol{\delta}_r) + \left(Q_r + A_r \hat{\Sigma}_{T+j-1} A_r^t\right)^{-1}(A_r \hat{\boldsymbol{\mu}}_{T+j-1} + \boldsymbol{\alpha}_r)\Big].$$

Similarities with Kalman Filter

The recursive equations from the MAPLF model are different from the Kalman Filter, but there are some similarities. In fact, the prediction phase in MAPLF is equal to the one used for Kalman Filter, and the differences lie in the correction phase. The first thing that we analyse, is the structure of the covariance matrix. In particular, by looking at Theorem 2.2.1, we have that the KF covariance matrix is equal to

$$\hat{\Sigma}_{KF} = \left((Q + A\hat{\Sigma}A^t)^{-1} + B^t R^{-1} B\right)^{-1}$$

and by recalling the formula for the one from MAPLF, we have

$$\hat{\Sigma}_{MAPLF} = \left((Q + A\hat{\Sigma}A^t)^{-1} + P^{-1}\right)^{-1}.$$

We can see that the only difference is that inside the inverse, for MAPLF the term $B_k^t R_k^{-1} B_k$ is substituted by P_k^{-1}.

To compare the means, we first assume for simplicity that the intercept terms $\boldsymbol{\alpha}, \boldsymbol{\beta}$ and $\boldsymbol{\delta}$ are zero. Then, we rewrite the mean from Kalman Filter, by observing that the term that multiplies $A\hat{\boldsymbol{\mu}}_{KF}$ is equal to

$$\begin{aligned} I - \hat{\Sigma}_{KF} B^t R^{-1} B &= \hat{\Sigma}_{KF}\left((\hat{\Sigma}_{KF})^{-1} - B^t R^{-1} B\right) \\ &= \hat{\Sigma}_{KF}\left((Q + A\hat{\Sigma}A^t)^{-1} + B^t R^{-1} B - B^t R^{-1} B\right) \\ &= \hat{\Sigma}_{KF}(Q + A\hat{\Sigma}A^t)^{-1}. \end{aligned}$$

From here, we have that the KF mean is equal to

$$\hat{\boldsymbol{\mu}}_{KF} = \hat{\Sigma}_{KF}\big[(Q + A\hat{\Sigma}A^t)^{-1} A\hat{\boldsymbol{\mu}} + B^t R^{-1} \mathbf{y}\big]$$

and the MAPLF mean is equal to

$$\hat{\boldsymbol{\mu}}_{MAPLF} = \hat{\Sigma}_{MAPLF}\big[(Q + A\hat{\Sigma}A^t)^{-1} A\hat{\boldsymbol{\mu}} + P^{-1} D\mathbf{y}\big].$$

Thanks to this, we can see that the structure of the estimate is very similar. In particular, in both cases, the term that multiplies $\hat{\boldsymbol{\mu}}$ is given by the product between the method-specific covariance and $(Q + A\hat{\Sigma}A^t)^{-1}A$. The main difference lies in the term that multiplies $\mathbf{y}$. This term is given by the product between the method-specific covariance, and then, for Kalman Filter we have $B^t R^{-1}$, while for MAPLF the second term is given by $P^{-1}D$.

2.4 Inverted State-Space Model

SSMs are usually used when the variable of interest, in our case the load, is entirely or partially hidden. Hence, in order to gather information about this variable, another variable, namely the observed variable, is used. When studying Kalman Filtering, we have modelled the observed variable $\mathbf{Y}_t$ as dependent on the latent variable $\mathbf{X}_t$. This is because generally, the observed variables are completely dependent on the hidden ones. In fact, the observed variables may represent the only reliable information that is available when predicting the future states of the latent variable and it is usually supposed to be determined by the unknown hidden variable. This introduces a direct causality between $\mathbf{X}_t \to \mathbf{Y}_t$.

Nonetheless, when introducing MAPLF, we have seen that it may be more convenient, at least intuitively, to model $\mathbf{X}_t$ given $\mathbf{Y}_t$ instead. When using SSMs for electricity load forecasting, the most common variables used consist of meteorological information. In this case, there is no direct causality between the load and the observed variable. On the contrary, the causality is present in the opposite directions, since the weather conditions partially determine the load.

This change in dependencies between the variables does not necessarily correspond to a change in the State-Space Model. Nonetheless, if we would like to interpret the Graphical Model corresponding to the State-Space Model as a causal graph, the arrow between the latent variable and the observed variable should be inverted. This is the rationale behind the following new model: the Inverted SSM.

2.4.1 Inverted State-Space Model

The following model consists of a modification of the standard State-Space Model. In particular, the arrows between $\mathbf{X}_t$ and $\mathbf{Y}_t$ for all t in the Probabilistic Graphical Model are inverted. This results in the graph in Figure 2.2.

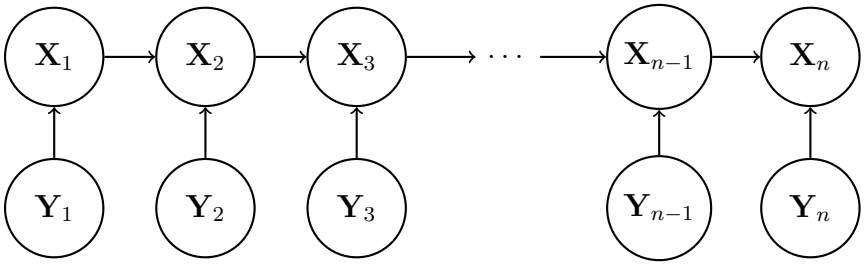

Figure 2.2: Conditional Independence graph of Inverted State-Space Model.

The first thing that one can notice, is that the complete joint distribution changes and the factorization becomes

$$\begin{aligned} &p(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n, \mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n) \\ &= p(\mathbf{X}_1)p(\mathbf{X}_1|\mathbf{Y}_1)p(\mathbf{Y}_2)p(\mathbf{X}_2|\mathbf{X}_1, \mathbf{Y}_2)\ldots p(\mathbf{Y}_n)p(\mathbf{X}_n|\mathbf{X}_{n-1}, \mathbf{Y}_n) \\ &= p(\mathbf{X}_1)p(\mathbf{X}_1|\mathbf{Y}_1)\prod_{i=2}^{n} p(\mathbf{X}_i|\mathbf{X}_{i-1}, \mathbf{Y}_i)p(\mathbf{Y}_i) \end{aligned}$$

which is a significant change.

The consequence of this inversion is that now $\mathbf{X}_{t-1}$ and $\mathbf{Y}_t$ given $\mathbf{X}_t$ are no longer independent. This is particularly evident in the Moral Graph associated with the graphical model, which is depicted in Figure 2.3.

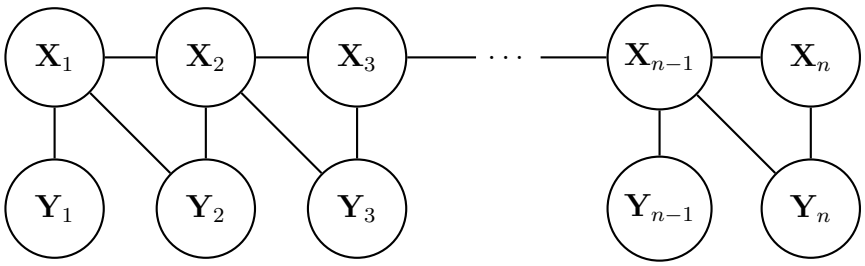

Figure 2.3: Moral graph of the Inverted State-Space Model.

From the graph, it is possible to deduce the following conditional independence statements:

- $\mathbf{Y}_i \perp\!\!\!\perp \mathbf{Y}_j|\mathbf{X}_k$ for $\min(i,j) \leq k < \max(i,j)$;
- $\mathbf{X}_i \perp\!\!\!\perp \mathbf{X}_j|\mathbf{X}_k$ for $\min(i,j) < k < \max(i,j)$ and $|i-j| > 1$;
- $\mathbf{Y}_i \perp\!\!\!\perp \mathbf{X}_j|\mathbf{X}_k$ for $j < k < i$ if $j < i$, or $i \leq k < j$ if $j > i$.

The changes with respect to the original State-Space Model are highlighted in red.

Furthermore, from this graph it is clear that modelling $\mathbf{X}_t|\mathbf{X}_{t-1}$ and $\mathbf{Y}_t|\mathbf{X}_t$ or $\mathbf{X}_t|\mathbf{Y}_t$ is not sufficient. It is necessary to jointly model the triplet $\mathbf{X}_t, \mathbf{X}_{t-1}, \mathbf{Y}_t$.

Joint distribution

We start by modelling the distribution

$$p(\mathbf{X}_n, \mathbf{Y}_n|\mathbf{X}_{n-1})$$

for its convenience when applying the recursive forecasting procedures. Since in order to derive the joint distribution, $\mathbf{X}_n|\mathbf{X}_{n-1}$ will also be modelled, we keep the same notation that was used to model the transition distribution in Kalman Filter. Moreover, the distribution $\mathbf{Y}_n|\mathbf{X}_{n-1}$ will be called the past-emission distribution. Then,

$$\begin{bmatrix} \mathbf{X}_n \\ \mathbf{Y}_n \end{bmatrix} |\mathbf{X}_{n-1} = \mathbf{x}_{n-1} \sim \mathcal{N}_{d+p}\left(\begin{bmatrix} A_n \\ E_n \end{bmatrix} \mathbf{x}_{n-1} + \begin{bmatrix} \boldsymbol{\alpha}_n \\ \boldsymbol{\epsilon}_n \end{bmatrix}, \begin{bmatrix} Q_n & U_n \\ U_n^t & S_n \end{bmatrix} \right)$$

with A_n the $d \times d$ transition matrix, E_n the past-emission $p \times d$ matrix, $\boldsymbol{\alpha}_n$ the transition intercept, $\boldsymbol{\epsilon}_n$ the past-emission intercept, Q_n the $d \times d$ transition covariance matrix, S_n the $p \times p$ past-emission covariance matrix and $U_n = Cov(\mathbf{X}_n, \mathbf{Y}_n | \mathbf{X}_{n-1})$.

We may also observe that the distribution can be written as

$$p(\mathbf{X}_n, \mathbf{Y}_n | \mathbf{X}_{n-1}) = p(\mathbf{X}_n | \mathbf{X}_{n-1}) p(\mathbf{Y}_n | \mathbf{X}_{n-1}, \mathbf{X}_n)$$

and therefore, another way to proceed could have been to first model only $p(\mathbf{Y}_n | \mathbf{X}_{n-1}, \mathbf{X}_n)$, and then to use the transition distribution that has been found in the Kalman Filter to find the joint distribution.

2.4.2 Forecasting

In the recursive forecast, for each recursive step $k = T+1, \dots, T+L$ with $L \leq R$, we are interested in finding the distribution for $p(\mathbf{x}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k})$, when

- the values of the load are known up to time T and the values of the observed variables are known up to time $T + R$;
- for $k = T+1$, $p(\mathbf{X}_{k-1} | \mathbf{X}_{1:T}, \mathbf{Y}_{1:k-1}) = \phi_d(\mathbf{X}_{k-1}; \mathbf{X}_T, 0)$;
- the known distributions are:

$$p(\mathbf{x}_{k-1} | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) = \phi_d\left(\mathbf{x}_{k-1}; \hat{\boldsymbol{\mu}}_{k-1}, \hat{\Sigma}_{k-1}\right)$$
$$p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{k-1}) = \phi_{d+p}\left(\begin{bmatrix}\mathbf{x}_k \\ \mathbf{y}_k\end{bmatrix}; \begin{bmatrix}A_k \mathbf{x}_{k-1} + \boldsymbol{\alpha}_k \\ E_k \mathbf{x}_{k-1} + \boldsymbol{\epsilon}_k\end{bmatrix}, \begin{bmatrix}Q_k & U_k \\ U_k^t & S_k\end{bmatrix}\right).$$

  To simplify the notation, from now on the indices of the parameter k and $k-1$ will be dropped.

This time the prediction consists of three steps.

First step

At first, we combine the two distributions that we have:

$$\begin{aligned} p(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) &= p(\mathbf{x}_{k-1} | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{x}_{k-1}, \mathbf{y}_{1:k-1}) \\ &= p(\mathbf{x}_{k-1} | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{k-1}) \\ &= \phi_d\left(\mathbf{x}_{k-1}; \hat{\boldsymbol{\mu}}, \hat{\Sigma}\right) \phi_{d+p}\left(\begin{bmatrix}\mathbf{x}_k \\ \mathbf{y}_k\end{bmatrix}; \begin{bmatrix}A \\ E\end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix}\boldsymbol{\alpha} \\ \boldsymbol{\epsilon}\end{bmatrix}, \begin{bmatrix}Q & U \\ U^t & S\end{bmatrix}\right). \end{aligned}$$

Applying Lemma 2.1.2, we obtain

$$\begin{aligned} &p(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) \\ &= \phi_{2d+p}\left(\begin{bmatrix}\mathbf{x}_{k-1} \\ \mathbf{x}_k \\ \mathbf{y}_k\end{bmatrix}; \begin{bmatrix}\hat{\boldsymbol{\mu}} \\ A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} \\ E\hat{\boldsymbol{\mu}} + \boldsymbol{\epsilon}\end{bmatrix}, \begin{bmatrix}\hat{\Sigma} & \hat{\Sigma}A^t & \hat{\Sigma}E^t \\ A\hat{\Sigma} & Q + A\hat{\Sigma}A^t & U + A\hat{\Sigma}E^t \\ E\hat{\Sigma} & U^t + E\hat{\Sigma}A^t & S + E\hat{\Sigma}E^t\end{bmatrix}\right). \end{aligned}$$

Second step

In the second step, we marginalize over $\mathbf{X}_{k-1}$.

$$\begin{aligned}
&p(\mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) \\
&= \int_{\mathbb{R}_d} p(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k-1}) d\mathbf{x}_{k-1} \\
&= \int_{\mathbb{R}_d} \phi_{2d+p}\left(\begin{bmatrix} \mathbf{x}_{k-1} \\ \mathbf{x}_k \\ \mathbf{y}_k \end{bmatrix} ; \begin{bmatrix} \hat{\boldsymbol{\mu}} \\ A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} \\ E\hat{\boldsymbol{\mu}} + \boldsymbol{\epsilon} \end{bmatrix} , \begin{bmatrix} \hat{\Sigma} & \hat{\Sigma}A^t & \hat{\Sigma}E^t \\ A\hat{\Sigma} & Q + A\hat{\Sigma}A^t & U + A\hat{\Sigma}E^t \\ E\hat{\Sigma} & U^t + E\hat{\Sigma}A^t & S + E\hat{\Sigma}E^t \end{bmatrix} \right) d\mathbf{x}_{k-1} \\
&= \phi_{d+p}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{y}_k \end{bmatrix} ; \begin{bmatrix} A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} \\ E\hat{\boldsymbol{\mu}} + \boldsymbol{\epsilon} \end{bmatrix} , \begin{bmatrix} Q + A\hat{\Sigma}A^t & U + A\hat{\Sigma}E^t \\ U^t + E\hat{\Sigma}A^t & S + E\hat{\Sigma}E^t \end{bmatrix} \right).
\end{aligned}$$

Third step

In the final step, we condition over $\mathbf{Y}_k$ using Lemma 2.1.1.

$$\begin{aligned}
p(\mathbf{x}_k | \mathbf{x}_{1:T}, \mathbf{y}_{1:k}) = \phi_d\Big(&\mathbf{x}_k; A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha} + (U + A\hat{\Sigma}E^t)(S + E\hat{\Sigma}E^t)^{-1}(\mathbf{y}_k - E\hat{\boldsymbol{\mu}} - \boldsymbol{\epsilon}), \\
&(Q + A\hat{\Sigma}A^t) - (U + A\hat{\Sigma}E^t)(S + E\hat{\Sigma}E^t)^{-1}(U^t + E\hat{\Sigma}A^t)\Big).
\end{aligned}$$

Now we have found the required target distribution. To summarize all the procedures, we introduce the following Theorem.

Theorem 2.4.1 (Inverted SSM L-step ahead forecast)**.** *Let us consider the inverted SSM $(\mathbf{X}_t, \mathbf{Y}_t)$ and let us suppose that the values of $\mathbf{X}_{1:T}$ and $\mathbf{Y}_{1:T+L}$ are known. Then, the distribution of $\mathbf{X}_{T+L}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+L}$ can be found recursively.*

Let $j = 1, \ldots, L$ and suppose that the distribution $\mathbf{X}_{T+j-1}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j-1}$ is known and normal with mean $\hat{\boldsymbol{\mu}}_{T+j-1}$ and covariance matrix $\hat{\Sigma}_{T+j-1}$. In particular for $j = 1$, $\hat{\Sigma}_T = \mathbf{0}$ and $\hat{\boldsymbol{\mu}}_T = \mathbf{X}_T$.

Then, letting $r = c(T+j)$, the distribution of $\mathbf{X}_{T+j}|\mathbf{X}_{1:T}, \mathbf{Y}_{1:T+j}$ can be obtained with the following formulas recursively. The covariance is

$$\begin{aligned}
\hat{\Sigma}_{T+j} =&(Q_r + A_r\hat{\Sigma}_{T+j-1}A_r^t) - (U_r + A_r\hat{\Sigma}_{T+j-1}E_r^t)(S_r + E_r\hat{\Sigma}_{T+j-1}E_r^t)^{-1} \\
&(U_r^t + E_r\hat{\Sigma}_{T+j-1}A_r^t)
\end{aligned}$$

and the mean is

$$\begin{aligned}
\hat{\boldsymbol{\mu}}_{T+j} =&A_r\hat{\boldsymbol{\mu}}_{T+j-1} + \boldsymbol{\alpha}_r + (U_r + A_r\hat{\Sigma}_{T+j-1}E_r^t)(S_r + E_r\hat{\Sigma}_{T+j-1}E_r^t)^{-1} \\
&(\mathbf{y}_{T+j} - E_r\hat{\boldsymbol{\mu}}_{T+j-1} - \boldsymbol{\epsilon}_r).
\end{aligned}$$

2.5 Vector Autoregressive Model

The three methods that have been defined in the previous sections all rely on the assumption that to improve the predictions, information from a so-called observed variable

is added to the model. When the accuracy of the predictions will be analysed in Chapter 3, it may interesting to investigate how this additional data influence the electricity forecasts. To do so, we now introduce a fourth method, namely a Vector Autoregressive model of order 1, that will act as a benchmark method for future discussion.

The Vector Autoregressive model is constructed over the Markov chain $\{\mathbf{X}_t\}$ and will not be influenced by the exogenous variable $\mathbf{Y}_t$. This independence from $\mathbf{Y}_t$ will allow us to see if the presence of the observed variable will enhance the predictions for the other three methods or not.

2.5.1 Forecasting

The Vector Autoregressive model of order 1 relies only on the transition distribution that was introduced in Section 2.2, and therefore on the set of parameters $\{A_k, \boldsymbol{\alpha}_k\}_{k\in\Delta}$. Furthermore, the estimates of the forecast will be produced using the recursive prediction phase of the Kalman Filter algorithm. Theorem 2.5.1 shows the details of the prediction process for this technique.

Theorem 2.5.1 (VAR L-step ahead forecast)**.** *Let us consider the Markov chain $\mathbf{X}_t$ and let us suppose that the values of $\mathbf{X}_{1:T}$ are known. Then, the distribution of $\mathbf{X}_{T+L}|\mathbf{X}_{1:T}$ can be found recursively.*

Let $j = 1, \dots, L$ and suppose that the distribution $\mathbf{X}_{T+j-1}|\mathbf{X}_{1:T}$ is known and normal with mean $\hat{\boldsymbol{\mu}}_{T+j-1}$ and covariance matrix $\hat{\Sigma}_{T+j-1}$. In particular for $j = 1$, $\hat{\Sigma}_T = \mathbf{0}$ and $\hat{\boldsymbol{\mu}}_T = \mathbf{X}_T$.

Then, letting $r = c(T + j)$, the distribution of $\mathbf{X}_{T+j}|\mathbf{X}_{1:T}$ can be obtained with the following formulas recursively. The covariance is

$$\hat{\Sigma}_{T+j} = Q_r + A_r \hat{\Sigma}_{T+j-1} A_r^t$$

and the mean is

$$\hat{\boldsymbol{\mu}}_{T+j} = A_r \hat{\boldsymbol{\mu}}_{T+j-1} + \boldsymbol{\alpha}_r.$$

2.6 Exponentially weighted parameter estimation

In this section, we will focus on learning the parameters of the Gaussian distributions that have been introduced in the previous sections. In particular, the method to infer the parameters is explained. A more precise description of the way in which the parameters are calculated for the distributions that have been defined previously, including considerations about the calendar types, can be found in Appendix A.2. The parameters will be inferred using the exponentially weighted least squares method. Let us first define the problem at hand.

Let $\mathbf{Y}_t$ be the q dimensional response variable, and let $\mathbf{X}_t$ be the vector of k explanatory variables. We have

$$\mathbf{Y}_t|\mathbf{X}_t = \mathbf{x}_t \sim \mathcal{N}_q(\Theta^t \cdot \mathbf{x}_t, \Gamma)$$

with Θ a $k \times q$ matrix and Γ a $q \times q$ covariance matrix.

Our objective is now to estimate the set of parameters $\{\Theta, \Gamma\}$. To address this problem, we consider the two time series of past observations, $\{\mathbf{y}_t\}$ and $\{\mathbf{x}_t\}$ with $t = 1, \ldots, n$.

2.6.1 The likelihood

To solve the problem of parameter estimation, an exponentially weighted log-likelihood function is employed. The weights are obtained using a forgetting factor $\lambda > 0$, and this hyperparameter will be tuned separately for each distribution. This factor will determine the importance of the most recent observations in the model. In particular, the smaller the value of λ, the more weight is given to the most recent observations and vice versa.

The weighted log-likelihood is therefore given by

$$L_n(\Theta, \Gamma; \mathbf{y}_1, \ldots, \mathbf{y}_n, \mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i=1}^{n} \lambda^{n-i} \log p(\mathbf{y}_i | \mathbf{x}_i)$$

and by making the density explicit, we have that it is equal to

$$\sum_{i=1}^{n} \lambda^{n-i} \left[-\frac{d}{2} \log(2\pi) - \frac{1}{2} \log |\Gamma| - \frac{1}{2} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right)^t \Gamma^{-1} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right) \right].$$

2.6.2 Maximum Likelihood Estimates

To find the parameters, we resort to maximising the weighted log-likelihood. To find the weighted Maximum Likelihood Estimation, we fix λ and derive the result with respect to Θ and Γ. Let us start with Θ.

Let us first retrieve formula (88) from the Matrix Cookbook Petersen et al. (2012). For W symmetric,

$$\frac{d}{dA}(\boldsymbol{x} - A\boldsymbol{s})^t W (\boldsymbol{x} - A\boldsymbol{s}) = -2W(\boldsymbol{x} - A\boldsymbol{s})\boldsymbol{s}^t.$$

Since Γ^{-1} is the precision matrix, this matrix is symmetric. Therefore:

$$\begin{aligned}
\frac{d}{d\Theta^t} L_n &= -\frac{1}{2} \sum_{i=1}^{n} \lambda^{n-i} \frac{d}{d\Theta^t} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right)^t \Gamma^{-1} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right) \\
&= -\frac{1}{2} \sum_{i=1}^{n} \lambda^{n-i} \left[-2\Gamma^{-1} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right) \mathbf{x}_i^t \right] \\
&= \Gamma^{-1} \sum_{i=1}^{n} \lambda^{n-i} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i \right) \mathbf{x}_i^t \\
&= \Gamma^{-1} \left[\sum_{i=1}^{n} \lambda^{n-i} \mathbf{y}_i \mathbf{x}_i^t - \Theta^t \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^t \right].
\end{aligned}$$

Furthermore, we have that $\frac{dL}{dA^t} = (\frac{dL}{dA})^t$, and so:

$$\frac{d}{d\Theta} L_n = \left[\sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}_i \mathbf{y}_i^t - \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^t \Theta \right] \Gamma^{-1}.$$

Hence, setting it to zero we find that the MLE estimate for Θ is:

$$\hat{\Theta}_n = \left[\sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}_i \mathbf{x}_i^t\right]^{-1} \left[\sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}_i \mathbf{y}_i^t\right].$$

Let us now take the derivative of the log-likelihood with respect to Γ. In order to do so, we use formulas number (57) and (61) from Petersen et al. (2012):

$$\frac{d}{dW} \log |W| = W^{-t}$$

$$\frac{d}{dW} \mathbf{a}^t W^{-1} \mathbf{a} = -W^{-t} \mathbf{a}\mathbf{a}^t W^{-t}.$$

Let us also observe that we are deriving with respect to a symmetric matrix, and this has an influence on the derivation result in general. In this case, let us denote with R the result that we would obtain without considering the symmetry. Then the correct result, in which the symmetry of Γ is considered, would be $2R - diag(R)$. Nonetheless, since our goal is to look at when the derivative is zero, when $R = 0$, also $2R - diag(R)$ will be 0. Hence, we will simply take derivatives with respect to Γ without accounting for its symmetry. This leads to

$$\begin{aligned}
\frac{d}{d\Gamma} L_n &= -\frac{1}{2} \sum_{i=1}^{n} \lambda^{n-i} \frac{d}{d\Gamma} \left(\log |\Gamma| + \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right)^t \Gamma^{-1} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right) \right) \\
&= -\frac{1}{2} \sum_{i=1}^{n} \lambda^{n-i} \left(\Gamma^{-1} - \Gamma^{-1} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right) \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right)^t \Gamma^{-1} \right) \\
&= -\frac{1}{2} \left[\sum_{i=1}^{n} \lambda^{n-i} - \Gamma^{-1} \sum_{i=1}^{n} \lambda^{n-i} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right) \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right)^t \right] \Gamma^{-1}.
\end{aligned}$$

Setting this to zero, we get:

$$\sum_{i=1}^{n} \lambda^{n-i} = \Gamma^{-1} \sum_{i=1}^{n} \lambda^{n-i} \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right) \left(\mathbf{y}_i - \Theta^t \mathbf{x}_i\right)^t.$$

Hence, if we substitute in the estimate for Θ, the weighted MLE estimate for Γ is given by:

$$\hat{\Gamma}_n = \frac{1}{\sum_{i=1}^{n} \lambda^{n-i}} \sum_{i=1}^{n} \lambda^{n-i} \left(\mathbf{y}_i - \hat{\Theta}_n^t \mathbf{x}_i\right) \left(\mathbf{y}_i - \hat{\Theta}_n^t \mathbf{x}_i\right)^t.$$

2.6.3 Matrix form

It is possible to rewrite the estimates we have just found more compactly in matrix form. We show this in the following theorem:

Theorem 2.6.1 (Exponentially weighted estimators)**.** *Let us consider the model*

$$Y_t|X_t = \mathbf{x}_t \sim \mathcal{N}_q(\Theta^t \cdot \mathbf{x}_t, \Gamma).$$

Our goal is to maximize the exponentially weighted log-likelihood function

$$L_n = \sum_{i=1}^{n} \lambda^{n-i} \log p(\mathbf{y}_i|\mathbf{x}_i).$$

We have that the estimates for Θ *and* Γ *are given by:*

$$\hat{\Theta}_n = \left(X_n^t \Lambda_n X_n\right)^{-1} X_n^t \Lambda_n Y_n$$

$$\hat{\Gamma}_n = \frac{(Y_n - X_n\hat{\Theta}_n)^t \Lambda_n (Y_n - X_n\hat{\Theta}_n)}{tr(\Lambda_n)} = \frac{Y_n^t \Lambda_n (Y_n - X_n\hat{\Theta}_n)}{tr(\Lambda_n)}$$

where $\Lambda_n = \begin{bmatrix} \lambda^{n-1} & & & \\ & \lambda^{n-2} & & \\ & & \ddots & \\ & & & \lambda \\ & & & & 1 \end{bmatrix}$, $X_n = \begin{bmatrix} (\mathbf{x}_1^c)^t \\ (\mathbf{x}_2^c)^t \\ \vdots \\ (\mathbf{x}_n^c)^t \end{bmatrix}$ *and* $Y_n = \begin{bmatrix} (\mathbf{y}_1^c)^t \\ (\mathbf{y}_2^c)^t \\ \vdots \\ (\mathbf{y}_n^c)^t \end{bmatrix}$.

Proof. The only thing that is left to prove is the second equality for $\hat{\Gamma}_n$. This follows from expanding the first term by

$$\begin{aligned}
&\frac{Y_n^t \Lambda_n Y_n - Y_n^t \Lambda_n X_n \hat{\Theta}_n - \hat{\Theta}_n^t X_n^t \Lambda_n Y_n + \hat{\Theta}_n^t X_n^t \Lambda_n X_n \hat{\Theta}_n}{tr(\Lambda_n)} \\
&= \frac{Y_n^t \Lambda_n Y_n - Y_n^t \Lambda_n X_n \hat{\Theta}_n - \hat{\Theta}_n^t X_n^t \Lambda_n Y_n + \hat{\Theta}_n^t X_n^t \Lambda_n X_n \left(X_n^t \Lambda_n X_n\right)^{-1} X_n^t \Lambda_n Y_n}{tr(\Lambda_n)} \\
&= \frac{Y_n^t \Lambda_n Y_n - Y_n^t \Lambda_n X_n \hat{\Theta}_n - \hat{\Theta}_n^t X_n^t \Lambda_n Y_n + \hat{\Theta}_n^t X_n^t \Lambda_n Y_n}{tr(\Lambda_n)} \\
&= \frac{Y_n^t \Lambda_n Y_n - Y_n^t \Lambda_n X_n \hat{\Theta}_n}{tr(\Lambda_n)} = \frac{Y_n^t \Lambda_n (Y_n - X_n \hat{\Theta}_n)}{tr(\Lambda_n)}.
\end{aligned}$$

□

2.6.4 Recursive form

Now that we found the parameters, it is time to find an expression to use online learning. This will allow us to update the parameters recursively.

Let us start by looking at a general sum with exponentially decaying weights, $k_n = \sum_{i=1}^{n} \lambda^{n-i} a_i$. We can notice that we can rewrite it as

$$k_n = \sum_{i=1}^{n} \lambda^{n-i} a_i = a_n + \lambda \sum_{i=1}^{n-1} \lambda^{n-1-i} a_i = a_n + \lambda k_{n-1}$$

and so we can obtain a recursive way of calculating the sum. Let us follow the same reasoning to rewrite $\hat{\Theta}_n$. Looking at the structure of the estimate

$$\hat{\Theta}_n = \left[\sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_i\mathbf{x}_i^t\right]^{-1}\left[\sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_i\mathbf{y}_i^t\right]$$

we can see that we can divide it into two parts. Let us define

$$H_n = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_i\mathbf{y}_i^t = \mathbf{x}_n\mathbf{y}_n^t + \lambda H_{n-1}$$

$$J_n = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_i\mathbf{x}_i^t = \mathbf{x}_n\mathbf{x}_n^t + \lambda J_{n-1}.$$

From this, we have that:

$$\hat{\Theta}_n = (J_n)^{-1}H_n.$$

We can do the same for $\hat{\Gamma}_n$. The estimate is

$$\hat{\Gamma}_n = \frac{1}{\sum_{i=1}^{n} \lambda^{n-i}} \sum_{i=1}^{n} \lambda^{n-i}\left(\mathbf{y}_i - \hat{\Theta}_n^t\mathbf{x}_i\right)\left(\mathbf{y}_i - \hat{\Theta}_n^t\mathbf{x}_i\right)^t$$

that can be rewritten, thanks to the previous theorem, as

$$\begin{aligned}\hat{\Gamma}_n &= \frac{1}{\sum_{i=1}^{n} \lambda^{n-i}} \sum_{i=1}^{n} \lambda^{n-i}\mathbf{y}_i\left(\mathbf{y}_i - \hat{\Theta}_n^t\mathbf{x}_i\right)^t \\ &= \frac{1}{\sum_{i=1}^{n} \lambda^{n-i}}\left(\sum_{i=1}^{n} \lambda^{n-i}\mathbf{y}_i\mathbf{y}_i^t - \sum_{i=1}^{n} \lambda^{n-i}\mathbf{y}_i\mathbf{x}_i^t\hat{\Theta}_n\right).\end{aligned}$$

Defining now

$$\gamma_n = \sum_{i=1}^{n} \lambda^{n-i} = 1 + \lambda\gamma_{n-1}$$

$$K_n = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{y}_i\mathbf{y}_i^t = \mathbf{y}_n\mathbf{y}_n^t + \lambda K_{n-1}.$$

$\hat{\Gamma}_n$ can be rewritten as follows:

$$\hat{\Gamma}_n = \frac{1}{\gamma_n}\left(K_n - H_n^t\hat{\Theta}_n\right).$$

Remark. It is possible to rewrite γ_n

$$\sum_{i=1}^{n} \lambda^{n-i} = \lambda^n \sum_{i=1}^{n}\left(\frac{1}{\lambda}\right)^i = \lambda^n \frac{\lambda^{-n-1} - \lambda^{-1}}{\lambda^{-1} - 1} = \frac{1-\lambda^n}{1-\lambda}$$

so that $\frac{1}{\gamma_n} = \frac{1-\lambda}{1-\lambda^n}$.

To summarize, if we want to estimate the parameters Θ and Γ recursively, the n-th step of the recursion consists of first updating the support structures

- $H_n = \mathbf{x}_n \mathbf{y}_n^t + \lambda H_{n-1}$
- $J_n = \mathbf{x}_n \mathbf{x}_n^t + \lambda J_{n-1}$
- $K_n = \mathbf{y}_n \mathbf{y}_n^t + \lambda K_{n-1}$
- $\gamma_n = 1 + \lambda \gamma_{n-1}$

and then calculating the desired values

- $\hat{\Theta}_n = (J_n)^{-1} H_n$
- $\hat{\Gamma}_n = \frac{1}{\gamma_n} \left(K_n - H_n^t \hat{\Theta}_n \right)$.

All four support structures are initialized as zero-matrices when $n = 0$.

Remark. Let us observe that for small n, we cannot compute J_n^{-1}, since J_n is a singular matrix. In particular, $J_0 = 0$ obviously does not have an inverse. Also $J_1 = \mathbf{x}_1 \mathbf{x}_1^t$ is singular. For this reason, when we start to train the model recursively, for the first steps we only learn the support structures, until the moment in which J_n admits an inverse. Only after, we can compute the parameters adaptively.

2.6.5 Recursive form for J_n^{-1}

One of the most computationally expensive steps of the recursion is the inversion of J_n. There is actually a way to update J_n^{-1} recursively starting from J_{n-1}^{-1}. In particular, using the Woodbury matrix inversion formula (formula (156) in Petersen et al. (2012)), we have that:

$$\begin{aligned} J_n^{-1} &= \left(\mathbf{x}_n \mathbf{x}_n^t + \lambda J_{n-1} \right)^{-1} = \frac{1}{\lambda} \left(\frac{1}{\lambda} \mathbf{x}_n \mathbf{x}_n^t + J_{n-1} \right)^{-1} \\ &= \frac{1}{\lambda} \left(J_{n-1}^{-1} - J_{n-1}^{-1} \mathbf{x}_n (\lambda + \mathbf{x}_n^t J_{n-1}^{-1} \mathbf{x}_n)^{-1} \mathbf{x}_n^t J_{n-1}^{-1} \right) \\ &= \frac{1}{\lambda} \left(J_{n-1}^{-1} - \frac{J_{n-1}^{-1} \mathbf{x}_n \mathbf{x}_n^t J_{n-1}^{-1}}{\lambda + \mathbf{x}_n^t J_{n-1}^{-1} \mathbf{x}_n} \right). \end{aligned} \tag{2.1}$$

This way is computationally cheaper than calculating the inverse. The problem with this method is its initialization. Indeed, in the remark above, we explained that for small n, J_n does not admit an inverse. To overcome this issue, we compute J_n^{-1} when the matrix is finally invertible, and for the next steps we update the inverse recursively.

2.7 Computational cost

In this section, we analyze the computational cost of the training and prediction steps for the algorithms. Let us start by looking at the training of parameters Θ and Γ for a generic model

$$Y_t|X_t = \mathbf{x}_t \sim \mathcal{N}_q(\Theta^t \cdot \mathbf{x}_t, \Gamma)$$

with $\mathbf{x}_t \in \mathbb{R}^k$. Looking at the recursive equations derived in Subsection 2.6.4, we have

- $\gamma = \lambda\gamma + 1$
- $H_{k\times q} = \lambda H + \mathbf{x}\mathbf{y}^t$
- $J_{k\times k} = \lambda J + \mathbf{x}\mathbf{x}^t$
- $K_{q\times q} = \lambda K + \mathbf{y}\mathbf{y}^t$
- $\hat{\Theta}_{k\times q} = J^{-1}H$
- $\hat{\Gamma}_{q\times q} = \frac{1}{\gamma}\left[K - H^t\hat{\Theta}\right]$

where $\mathbf{y}, \mathbf{x}$ are the new observations of the response and regressor variables respectively.

We will consider the asymptotic cost of the multiplication of two matrices $A_{k\times h} \cdot B_{h\times q}$ to be equal to $O(khq)$. There are multiple algorithms to perform matrix multiplication that have a smaller cost, especially for square matrices, but, for simplicity, we will use the standard iterative multiplication algorithm that has the cost we have just defined. Analogously, we consider the asymptotic cost of performing the inverse of a matrix $A_{k\times k}$ to be equal to $O(k^3)$ using Gauss elimination.

We can see that the computational cost of the recursive equation for H, J and K is given by the cost of the product of the two vectors. In particular, the cost to obtain H is $O(qk)$, the cost to obtain J is $O(k^2)$ and the cost to obtain K is $O(q^2)$. At the end of the previous section, we discussed what we will need to train the model without the parameters $\hat{\Theta}$ and $\hat{\Gamma}$ for a few days before doing predictions because otherwise the matrix J is still singular and we cannot compute its inverse. Therefore, when training the model without performing any forecast, the asymptotic cost is simply $O(k^2 + q^2)$.

Let us now look at the computational cost of computing $\hat{\Theta}$. At first, we will compute the inverse J^{-1}. The cost of the inversion would be $O(k^3)$, but if we compute the inverse once and then use the formula (2.1) to update the inverse recursively, the cost becomes $O(k^2)$. Then we will multiply the inverse with H, and this operation has cost $O(qk^2)$. Therefore the asymptotic cost of computing $\hat{\Theta}$ will be $O(qk^2)$. When computing $\hat{\Gamma}$ the cost is given by the product between H^t and $\hat{\Theta}$ that is $O(q^2k)$. Summing all the costs together, we have that to perform one update step for all the parameters, the total computational cost is $O(qk^2 + q^2k)$.

2.7.1 Cost for Kalman Filter

Training

We have just analyzed the cost of a general model. Let us now start to analyse the cost of one recursive training step for Kalman Filter. In particular, we need to learn

both transition and emission distribution parameters. Since the asymptotic cost will not change, we will consider only the case in which there is no intercept in the models.

For the transition distribution, the cost is

- $O(dd^2 + d^2d) = O(d^3)$ to learn the parameters;
- $O(d^2 + d^2) = O(d^2)$ to update only the support structures.

For the emission distribution, the cost is

- $O(dp^2 + d^2p)$ to learn the parameters;
- $O(d^2 + p^2)$ to update only the support structures.

Therefore, the total computational cost of one recursive step in which we update the parameters is $O(d^3 + dp^2)$, while a recursive step in which we only update the support structures costs $O(d^2 + p^2)$.

Prediction

For the asymptotic prediction cost, let us first look at the recursive equation for the covariance:

$$\Big(\big(Q_{d\times d} + A_{d\times d}\hat{\Sigma}_{d\times d}(A^t)_{d\times d}\big)^{-1} + (B^t)_{d\times p}R^{-1}_{p\times p}B_{p\times d}\Big)^{-1}.$$

The product $A\hat{\Sigma}A^t$ and the inverse of $Q + A\hat{\Sigma}A^t$ both have cost $O(d^3)$. R^{-1} has cost $O(p^3)$, and the product $B^tR^{-1}B$ has cost $O(dp^2 + d^2p)$. Finally, the total inverse has cost $O(d^3)$.

Let us now look at the operations for the mean:

$$A_{d\times d}\hat{\boldsymbol{\mu}}_d + \boldsymbol{\alpha}_d + \hat{\Sigma}_{d\times d}(B^t)_{d\times p}R^{-1}_{p\times p}\big(\mathbf{y}_p - B_{p\times d}(A_{d\times d}\hat{\boldsymbol{\mu}}_d + \boldsymbol{\alpha}_d) - \boldsymbol{\beta}_p\big).$$

$A\hat{\boldsymbol{\mu}}$ has cost $O(d^2)$ and $B(A\hat{\boldsymbol{\mu}} + \boldsymbol{\alpha})$ has cost $O(pd + d^2)$. Then, if we take this term and multiply it (on the left) with R^{-1} the cost is $O(p^2)$. Then multiplying the result by B^t has cost $O(dp)$ and finally multiplying this with $\hat{\Sigma}$ has cost $O(d^2)$.

Putting all together, the total asymptotic cost is $O(d^3 + p^3)$.

2.7.2 Cost for MAPLF

Training

The asymptotic cost to learn the parameters for the transition distribution is the same as before. Instead, the emission distribution has a cost of

- $O(dp^2 + d^2p)$ to learn the parameters;
- $O(d^2 + p^2)$ to update only the support structures.

Therefore, the total computational cost of one recursive step in which we update the parameters is $O(d^3 + dp^2)$, while a recursive step in which we only update the support structures costs $O(d^2 + p^2)$.

Prediction

For the prediction cost, let us first look at the recursive equation for the covariance:

$$\left(\left(Q_{d\times d} + A_{d\times d}\hat{\Sigma}_{d\times d}(A^t)_{d\times d}\right)^{-1} + P^{-1}_{d\times d}\right)^{-1}.$$

Since P^{-1} has an associated $O(d^3)$ computational cost, the total cost for computing the covariance is $O(d^3)$.

Let us now look at the mean:

$$A_{d\times d}\hat{\boldsymbol{\mu}}_d + \boldsymbol{\alpha}_d + \hat{\Sigma}_{d\times d}P^{-1}_{d\times d}(D_{d\times p}\mathbf{y}_p + \boldsymbol{\delta}_d - A_{d\times d}\hat{\boldsymbol{\mu}}_d - \boldsymbol{\alpha}_d).$$

$A\hat{\boldsymbol{\mu}}$ has cost $O(d^2)$ and $D\mathbf{y}$ has cost $O(dp)$. Then, the product between P^{-1} and the vector $D\mathbf{y} + \boldsymbol{\delta} - A\hat{\boldsymbol{\mu}} - \boldsymbol{\alpha}$ costs $O(d^2)$. When we then multiply $\hat{\Sigma}$ by it, we find the same computational cost.

Putting it all together, the total asymptotic prediction cost is $O(d^3 + dp)$.

2.7.3 Cost for Inverted SSM

Training

When training the parameters, the response variable has dimension $d + p$ and the regressor has dimension d. Therefore, the asymptotic cost for a recursive step is

- $O((d+p)d^2 + (d+p)^2 d) = O(d^3 + pd^2 + (d^2 + p^2 + dp)d) = O(d^3 + dp^2)$ to learn the parameters;
- $O(d^2 + (d+p)^2) = O(d^2 + p^2)$ to update only the support structures.

Prediction

The covariance recursive equation is

$$\begin{aligned}&(Q_{d\times d} + A_{d\times d}\hat{\Sigma}_{d\times d}(A^t)_{d\times d}) - (U_{d\times p} + A_{d\times d}\hat{\Sigma}_{d\times d}(E^t)_{d\times p})\cdot\\ &\cdot (S_{p\times p} + E_{p\times d}\hat{\Sigma}_{d\times d}(E^t)_{d\times p})^{-1}((U^t)_{p\times d} + E_{p\times d}\hat{\Sigma}_{d\times d}(A^t)_{d\times d}).\end{aligned}$$

The computation of the first term has cost $O(d^3)$. Afterwards, we have a product of three terms. The computation of the first and the third has cost $O(d^2 p)$. The term in the middle has an associated computational cost of $O(d^2 p)$, and the calculation of the inverse of $O(p^3)$. Then, the product between the three terms has cost $O(dp^2)$. We have that the cost to compute the covariance is $O(d^3 + p^3)$. Since the cost for the mean cannot be higher than that, it is also the total computational cost for one prediction step.

2.7.4 Cost for VAR

The cost for training is given by the cost of the recursive step to obtain the transition distribution, which is:

- $O(d^3)$ to learn the parameters;

- $O(d^2)$ to update only the support structures.

The computational cost for the prediction step is determined by the cost of the product $A\hat{\Sigma}A^t$, which is $O(d^3)$.

2.7.5 Comparison

Let us now do a quick comparison between the different prediction algorithms.

	Kalman Filter	MAPLF	Inv. SSM	VAR
Training (only support)	$d^2 + p^2$	$d^2 + p^2$	$d^2 + p^2$	d^2
Training (parameters)	$d^3 + dp^2$	$d^3 + dp^2$	$d^3 + dp^2$	d^3
Prediction	$d^3 + p^3$	$d^3 + dp$	$d^3 + p^3$	d^3

Table 2.1: Computational costs for one recursive step of the four proposed methods.

As shown in Table 2.1, Kalman Filter and Inverted SSM have exactly the same computational cost for both training and prediction. Moreover, the training steps for the three State-Space Model methods have the same computational cost for all methods: $O(d^2 + p^2)$ when only updating the support structures, $O(d^3 + dp^2)$ when requiring also the computation of the parameters. The VAR method has a computational cost of only $O(d^3)$.

Let us turn our attention to the prediction cost. In this case, there is a shared term of $O(d^3)$ in all methods, but KF and Inverted SSM have an additional cost of $O(p^3)$, while MAPLF has an additional cost of $O(dp)$. This difference is given by the fact that the MAPLF method does not need to invert a $p \times p$ matrix, while the VAR model doesn't contain any observed variables. This also implies that when $p > d$, MAPLF is more efficient than Kalman Filter and Inverted SSM.

	Kalman Filter	MAPLF	Inverted SSM
Tr: Independent case	dp^2		
Pr: Independent case	dp^3	dp	dp^3
Tr: $p = \bar{p}d$	$d^3\bar{p}^2$		
Pr: $p = \bar{p}d$	$d^3\bar{p}^3$	$d^2(d + \bar{p})$	$d^3\bar{p}^3$

Table 2.2: Computational costs for one recursive step: special cases.

Table 2.2 shows the total computational cost in two particular cases. In this analysis, we don't consider the cost of the VAR model. The first is the case in which we have a d-dimensional load vector, but we train and predict each load component independently, and each component uses a p-dimensional vector as observed variable. To compute the effect of this procedure, we first consider the cost in which $d = 1$, i.e., the computational cost for each component. Then, we multiply the result by d, since we apply this procedure for each component. The first thing we can observe is that the training of the parameter becomes computationally more efficient since the term d^3 is missing. The same applies

to the prediction step in MAPLF, which is of order $O(dp)$. On the other hand, for Kalman Filter and Inverted SSM, the computational cost of a prediction step is $O(dp^3)$, and this means that the cost could increase, especially if p is much bigger than d. In particular, the computational cost for the independent case is bigger if $O(d^2) \subset O(p^3)$.

The second case to be analysed is the case in which the observed variables are $\bar{p}d$. For example, if we consider the case of multiregional load forecasting, we may have d regions and then $\bar{p}$ observed variables for each region, reaching a total of $\bar{p}d$ observed variables. This cost will be compared to the case of independent models, in which for each region we only use the $\bar{p}$ regional variables. The computational training cost is $O(d^3\bar{p}^2)$, which is d^2 times the cost for the independent models. The same happens for the cost of the prediction step in Kalman Filter and Inverted SSM, which is $O(d^3\bar{p}^3)$, and this cost may become really large in the presence of many variables. Finally, the prediction asymptotic cost for MAPLF is $O(d^2(d+\bar{p}))$, which is always smaller than the cost of the other two methods and may therefore be preferable.

Chapter 3

Multiregional Load Forecasting

In Chapter 2, four Short-Term Load Forecasting algorithms have been developed. In this chapter, the accuracy of these algorithms will be analysed. Multivariate electricity load forecasting can be applied to different scenarios, and for the purpose of this thesis, we will focus on Multiregional Load Forecasting.

3.1 The ISO New England dataset

The dataset that has been chosen for the analysis is based on the measurements from the ISO New England electricity transmission organization. This dataset consists of historic load data for the region of New England (US), and the specific dataset that will be used is the one that was adopted for the Global Energy Forecasting Competition (GEFCOM) of 2017 (Hong et al., 2019)[1].

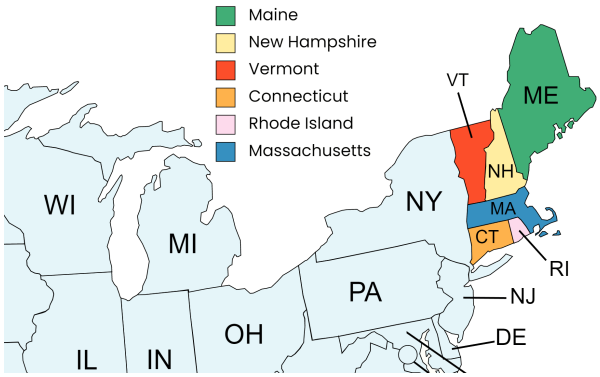

Figure 3.1: Map of New England (US).

The measurements are taken for each hour of the day, from March the 1st 2003 to April the 30th 2017. Additional information about the day of the week and about public holidays is present. The measurements are taken for all six states of the United States region of New England: Maine (ME), New Hampshire (NH), Vermont (VT), Connecticut (CT), Rhode Island (RI) and Massachusetts (MA), and in addition, the state of Massachusetts is divided into three sub-regions: Southeast Massachusetts (SEMA),

[1]The data has been retrieved from: https://github.com/camroach87/gefcom2017data

Western-Central Massachusetts (WCMA) and Northeast Massachusetts (NEMA), with the last region containing the city of Boston.

For each area, also meteorological measurements are collected. In particular, the measurements consist of

- Dry Bulb Temperature (°F): it refers to the temperature of the air;
- Dew Point Temperature (°F): it refers to the temperature at which water vapour starts to condense out of the air. If the dew-point temperature is close to the air temperature, the relative humidity is high, and if the dew point is well below the air temperature, the relative humidity is low.[2]

In the remaining of this section and in the next section, we perform an exploratory analysis of the dataset. From Section 3.3 onwards, we will first present the evaluation metrics, and then the results of the Load Forecasting techniques that have been presented in Chapter 2.

3.1.1 The demand

Yearly behaviour

The first thing that we are gonna analyse is the demand curve. Let us take a look at the aggregated load, i.e., the sum of all the regional loads. Figure 3.2 shows the electricity consumption in the whole region of New England from 2004 to 2010, with a one-month moving average. Looking at the plot, notice that for each year there are two important peaks in demand. One in winter, between January and February, which is due to the high demand for electricity for heating purposes, and another one during summer, between July and August when electricity is used for cooling. This second peak is usually higher.

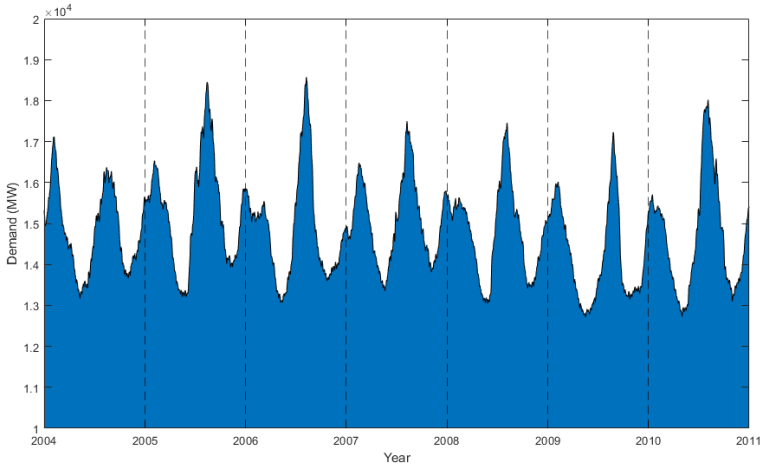

Figure 3.2: New England's electricity demand, with a one-month moving average, from 2004 to 2010.

[2]The meanings of the different temperatures are taken from: https://www.weather.gov/source/zhu/ZHU_Training_Page/definitions/dry_wet_bulb_definition/dry_wet_bulb.html.

It is also possible to look at how much the different subregions contribute to the aggregated curve. Figure 3.3 shows all the different subregional curves together. With this plot, we understand the size of each region's electrical industry. It is clear that Massachusetts by itself would be the most significant electrical consumer, but because it is divided into three subregions, Connecticut becomes the region having the highest demand. On the contrary, the least consuming region is Vermont, followed by Rhode Island. Furthermore, the seasonal peak's time of occurrence is generally the same for each area. Even the size of the peaks is related: in years in which the summer peak is relatively small, for example in 2004, all the regions experience this downfall equally. This leads to the conclusion that the size of the peaks depends on a common factor for all the states of New England.

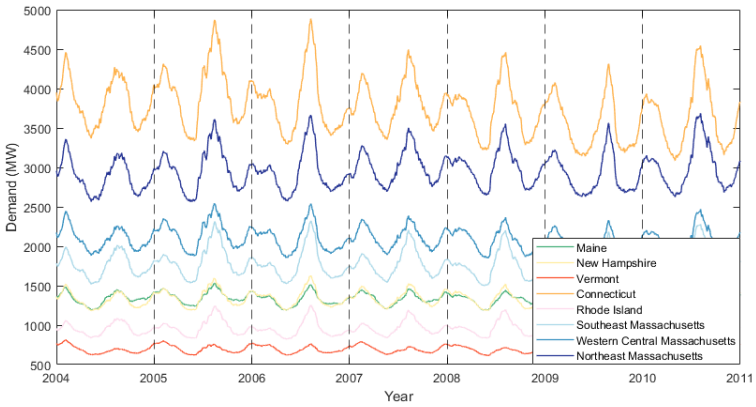

Figure 3.3: New England's subregional electricity demand, with a one-month moving average, from 2004 to 2010.

Daily behaviour

We now turn our attention to the daily behaviour of the series. In this case, no moving average is applied. Figure 3.4 shows the aggregated demand in a 9 days period, starting from Monday November 6th 2006.

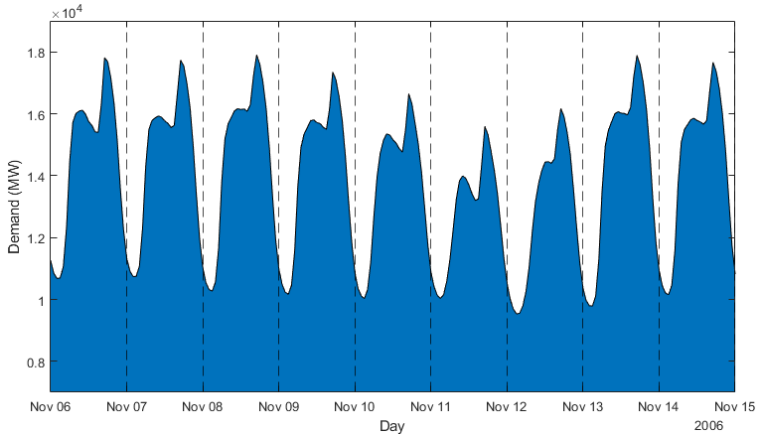

Figure 3.4: New England's electricity demand, 6 Nov. 2006 - 14 Nov. 2006.

November 11th and 12th are respectively a Saturday and a Sunday, and we may notice that during the weekend the consumption is slightly smaller. This is usually the case since during the weekend the offices and the industrial sector are closed. Nonetheless, this change is not as big as someone may expect by looking at Figure 1.2 from the first chapter, where industries were shown to take a huge share of energy consumption. To understand this, it may be useful to look at Figure 3.5, in which it is clear how the deindustrialization of the US during the end of the '90s heavily changed the proportion between the amount of electricity used for industrial purposes and the one used for residential and commercial usage.

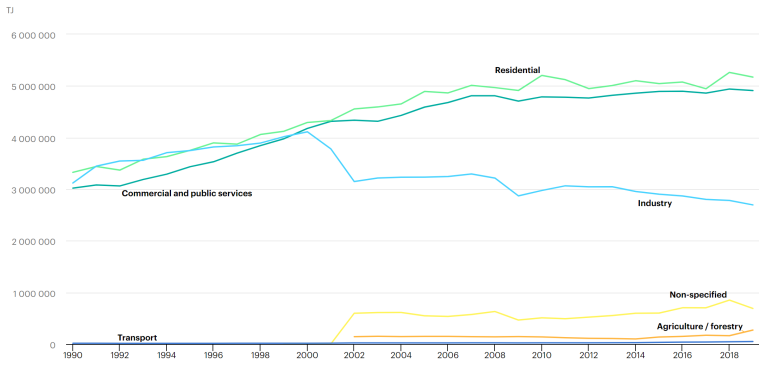

Figure 3.5: Electricity consumption by sector, United States 1990-2019.
The figure has been obtained from the IEA official website.

Looking further at the aggregated consumption of New England, each day presents two peaks, one in the morning, and a bigger one in the evening. This is probably happening because we are in November and electricity is starting to be used for heating purposes. Therefore, electricity demand spikes during the coldest hours of the day. Furthermore, the fact that the spike in the evening is higher is probably given by users returning home and using electricity for cooking, home entertainment, etc.

It may be interesting to also take a look at the consumption during the summer to see if the pattern changes. Figure 3.6 shows the aggregated demand starting from Monday August 7th 2006. Indeed, looking at a usual weekday, this time the electricity demand has only one big spike in the middle of the day. This is because in August electricity is mainly used for cooling purposes, causing a big spike in demand during the warmest hours of the day. Furthermore, a small increase in demand is also present during the evening, when people return to their homes.

Moreover, we can notice that in August, the change in consumption between a workday and a non-workday is more pronounced than in November. During the weekend the spike that appears in the middle of the day has a flat peak, and the evening increase is more evident than during the rest of the week. The reason behind this may be that during the week, in the warmest hours, electricity is mainly used in factories, offices, shops, etc. for cooling purposes, while during the weekend, these places are almost all closed. Residential cooling takes less energy than the amount needed for big factories or offices and people are generally more responsible about electricity spending at home. Additionally, during summer weekends, more people are spending their time outside, and this could justify the difference between the winter season demand in a weekend, in

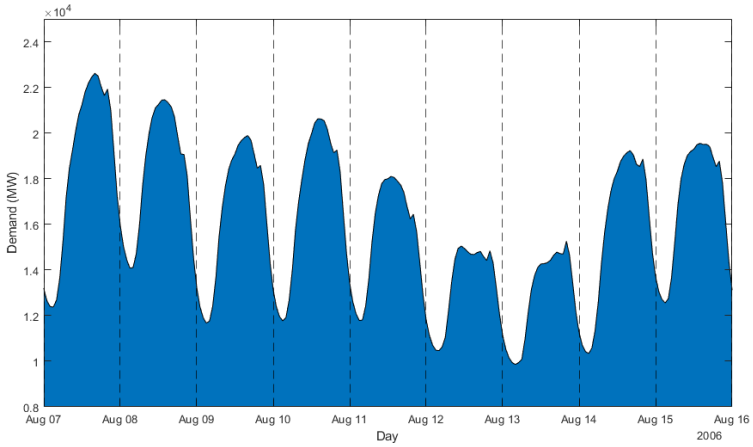

Figure 3.6: New England's electricity demand, 7 Aug. 2006 - 16 Aug. 2006.

which people tend to stay more inside.

3.1.2 The temperatures

Now we turn our attention to the temperature variables. Figure 3.7 shows the mean dry bulb and dew point temperatures between all the regions for a period of three years, from 2006 to the end of 2008, and a one-week moving average has been applied to the temperature curves.

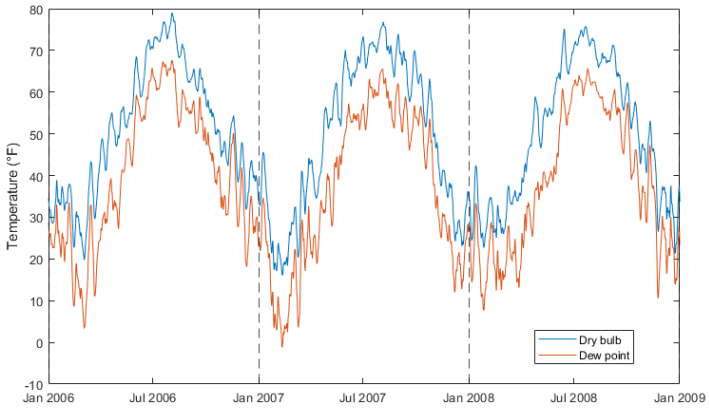

Figure 3.7: New England's average dry bulb and dew point temperatures, with one-week moving average, from 2006 to 2008.

From the plot, the seasonality of the temperatures is evident. In summer both temperatures are high and in winter both temperatures are low. Furthermore, the temperatures show a certain regularity each year. There has been a colder winter in 2007 than in 2008 and the summer of 2006 was the hottest one of the three, but in general, the temperatures were similar.

Let us now examine the differences in temperatures between the different states, considering Massachusetts' subregions together. For simplicity, in Figure 3.8, we only look at the dry bulb temperature for a single year, 2006, and we will use a one-week

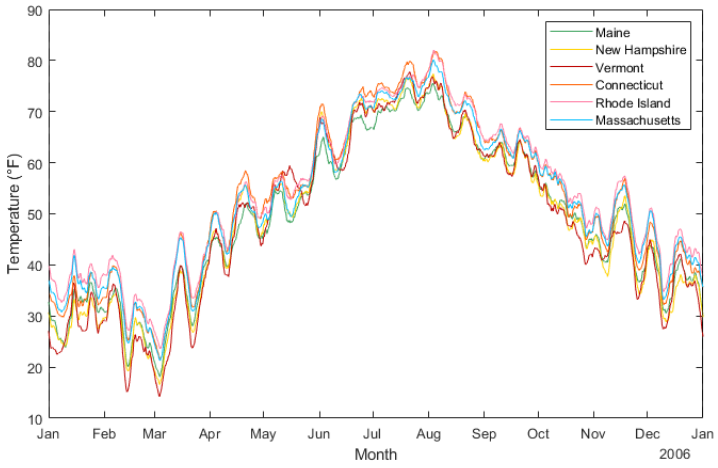

Figure 3.8: New England's subregional mean dry bulb temperature, with three-days moving average, from 2006 to 2008.

lag moving average. It is clear that generally, the regions share a similar climate, and that sudden drops or spikes in temperature happen in all the regions at the same time. Vermont appears to be the coldest region, while Rhode Island seems to be the warmest one.

Nonetheless, to notice the subtle difference in temperature, we should concentrate on certain time intervals. To do this, in Figure 3.9 the winter and summer mean dry bulb temperatures are shown. The temperature is smoothed using a 3-days moving average.

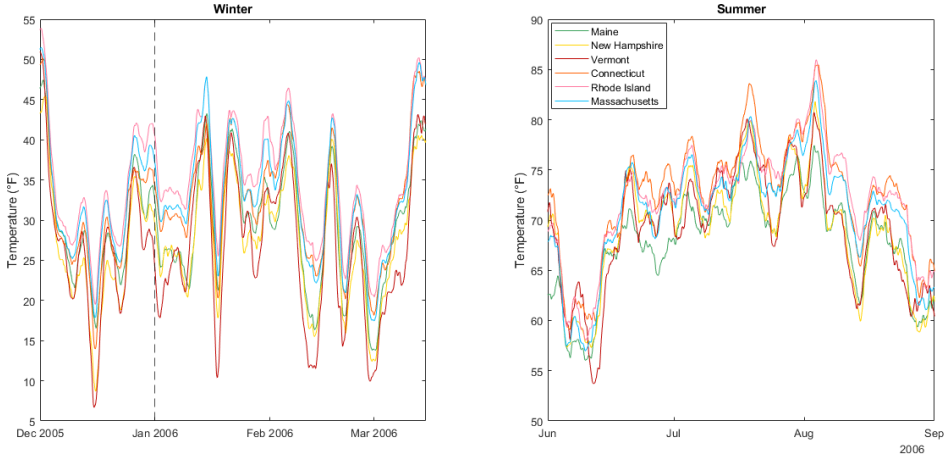

Figure 3.9: New England's subregional mean dry bulb temperature, with three-days moving average,
Winter: 1 Dec. 2005 - 15 March 2006; Summer: 1 June - 31 Aug. 2006.

When looking at the winter plot, we can immediately see that the temperatures are very low also in December and January, which was not clear from the previous plot. This is because the one-week moving average simplified the temperature series too heavily. In winter, the coldest region is usually Vermont, followed by New Hampshire and Maine. Rhode Island and Massachusetts, on the other hand, are the warmest regions. In summer

the warmest states are Connecticut and Rhode Island, while the coldest ones are Maine and Vermont. Furthermore, we can notice that the winter period lasts much longer than the summer period.

These observations are consistent with the description present on the Wikipedia page of New England[3]:

Maine, New Hampshire, Vermont, and western Massachusetts have a humid continental climate. In this region the winters are long and cold [...]. The summer's months are moderately warm, though summer is rather short [...]. In central and eastern Massachusetts, northern Rhode Island, and northern Connecticut, the same humid continental prevails, though summers are warm to hot, winters are shorter [...]. Southern and coastal Connecticut is the broad transition zone from the cold continental climates of the north to the milder subtropical climates to the south. [...] Winters also tend to be much sunnier in southern Connecticut and southern Rhode Island compared to the rest of New England.

3.2 Linear relationships

In the forecasting techniques that we introduced in Chapter 2, several linear models have been created, e.g., the ones for the transition and emission distributions. In this section, we analyse the linear relationships between variables.

3.2.1 Transition model: $\mathbf{X}_t|\mathbf{X}_{t-1}$

We start by looking at the Markov Chain given by the variable $\mathbf{X}_t$. How are the variables between two consecutive time-steps related to one another?

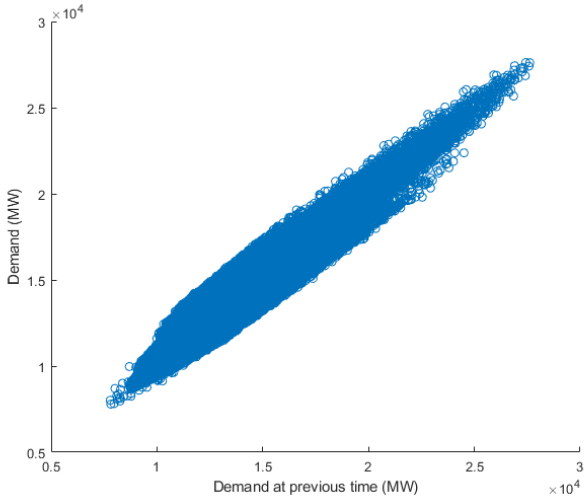

Figure 3.10: New England's electricity demand, time $t-1 \times t$.

For simplicity, we will only study the aggregated loads and assume that all the conclusions apply to the particular subregions as well. In the first plot, shown in Figure

[3]The Wikipedia page has been accessed on October 13th 2022.

3.10, we analyse a scatter plot in which each point is given by the aggregated load at time t on the y-axis and the load at the previous time-step $t-1$ on the x-axis. There is a clear linear relationship between the two consecutive loads, and it looks like a fitted linear model would be extremely close to the identity function. This leads to the conclusion that the load at time t is almost equal to the load at time $t-1$. This strong linearity is also given by the fact that hourly observations are considered. Indeed if the observations had been less frequent, for example, every 3 hours, then the linearity would have been less clear. On the other hand, if the observations had been sampled every minute, the linearity would have been even more pronounced.

Even though the linearity is indisputable, the cloud of points still presents a non-neglectable error. This fact is relieved by using hour-specific models, as we do via calendar type-specific parameters. Therefore, it is interesting to look at the relationship between load at time t and at time $t-1$ for each hour of the day. The results are shown in Figure 3.11. A black line indicates the identity function and a red line indicates the curve of the hour-specific fitted linear model.

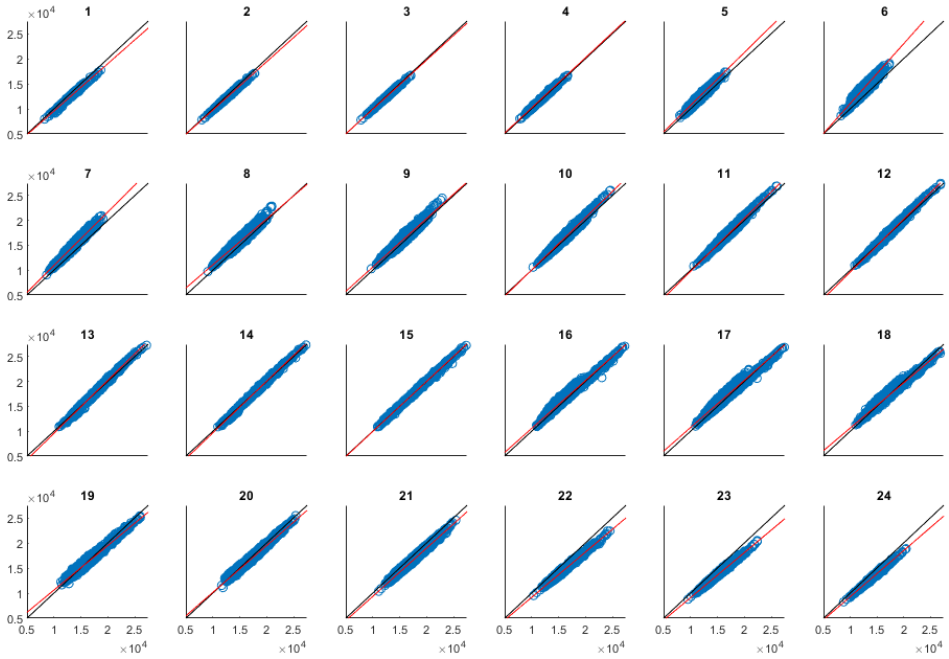

Figure 3.11: New England's electricity demand, time $t-1 \times t$, by hour, with identity function (in black) and best linear model (in red).

At a first glance, it is clear that the lines are smoother than when considering all the hours together. Furthermore, the difference in magnitude between the hours is evident. The hours of the day in which demand is lower, from 1 am to 5 am, have a cloud of points that is on the bottom left, while the hours of the day in which demand is higher, usually in the middle of the day, have the cloud of points on the top right.

The slope of the linear relationship may slightly change from hour to hour. To analyse this behaviour we now look at the red line. We start from the first hours of the day, and in particular from 3 am to 4 am, in which the demand is very low and stable. In fact, during these hours the identity function models the relationship perfectly. From

5 to 7 am, there is an increase in demand, and we can notice that the slope of the curves increases as well. This happens until 8 am. Afterwards, from 8 am until 9 pm, the slope of each hour is more or less close to 1. What changes during these hours is the "fatness" of the cloud of points. In fact, in the morning around 8-9 am and in the afternoon from 4 to 7 pm, we have that the cloud of points is "less linear". This is probably due to the fact that we are considering the relationship between consecutive loads for all the days of the dataset, and during these hours there is a high variability between winter and summer demand. Therefore, the season-specific demand curves are averaged out to obtain something similar to the identity function. After 9 pm, the slopes start to decrease until 1 am. This is because the demand decreases during the night, reaching the lowest point around 3 am.

After this brief study, we can confirm that linearity holds well for the transition model. Indeed, if we create a linear model for each hour in MATLAB, we obtain a mean R squared of 0.974, making the model very robust. In fact, the worse value is 0.893 for 6 am, and the best value is 0.997 for 2 pm. Furthermore, if we consider the subregion-specific loads and predict them using not only their load at the previous time step but also the demand of the other areas, we can increase the total mean R squared to 0.981.

3.2.2 MAPLF emission model: $\mathbf{X}_t|\mathbf{Y}_t$

Now we start to analyse the relationship between the loads and the temperatures by examining the emission model for the MAPLF algorithm.

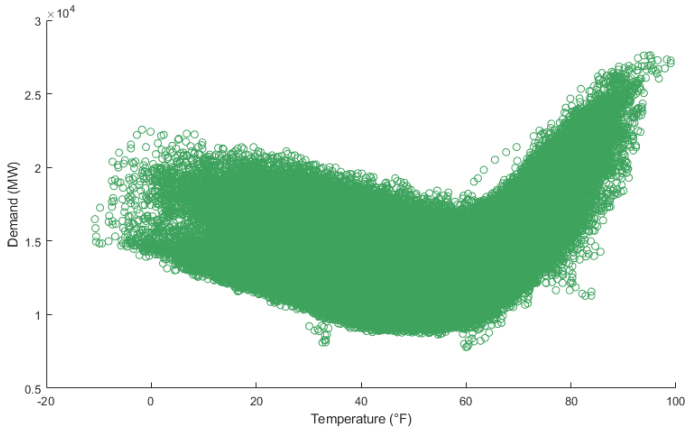

Figure 3.12: New England's dry bulb temperature × demand.

To begin the analysis, we look at the scatter plot with dry bulb temperature on one axis and the aggregated load on the other. This plot is shown in Figure 3.12. We can see that there exists a noisy relationship between the two variables. As we discussed previously, the load is usually higher when very high or very low temperatures are present. This can be observed also from the graph, in which the load is increased when the temperatures are either very high or very low. In particular, it seems that there is a breakpoint around 60°F (15.6°C). Starting from this point, if the temperature decreases, the demand increases linearly. If on the other hand the temperature increases, the demand increases as well linearly, but this time with a steeper slope.

Let us now analyse this relationship for a specific hour of the day, since we will be using time-specific linear models. In particular, we consider the case for 1 am. When plotting the temperature against the load, and fitting a simple linear model, we obtain Figure 3.13. We can now see that the pattern at 1 am is much smoother than when considering all the hours together. Furthermore, a simple linear model is not sufficient to model the relationship between load and temperature. What we could do now is either fit the relationship using a piecewise linear function or look for quadratic or higher-degree polynomial relationships.

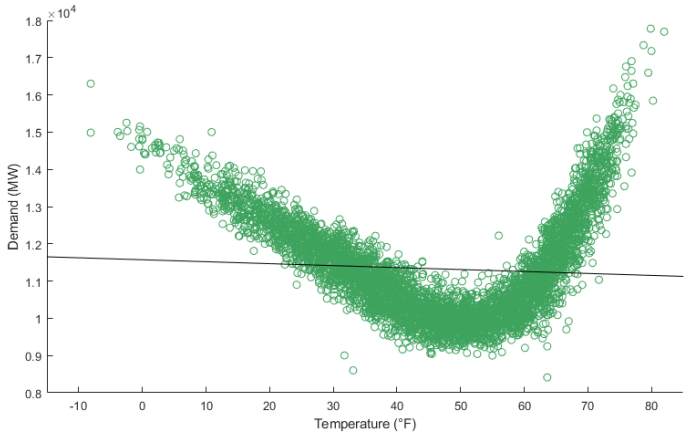

Figure 3.13: New England's dry bulb temperature × demand, at 1 am, with simple linear model.

Let us start using a piecewise linear function. We may expect that for this hour the breakpoint of the function will be around 50-55°F. We start by initializing the breakpoint at 60°F. Then, we create two linear models, one for high temperatures and one for low temperatures. In the beginning, these two models will not collide at the breakpoint. Therefore, to look for the correct breakpoint, we search for the point at which the two estimated lines collide. To do this, we start by looking at the difference between the two lines in the point of interest and then use a while cycle that modifies the breakthrough point until a certain tolerance for the difference is obtained. Running this algorithm, we get that the point is at 54.9°F, as shown in Figure 3.14. Furthermore, we have that the mean R squared between the two models, weighted on the number of observations, is 0.837. If instead, we are not interested in making the two lines collide, but only in the breakpoint that provides the best score, we have that the new breakpoint is at 53.3°F and the mean R squared is 0.841.

Let us now instead investigate other polynomial relationships. Let us start by constructing a quadratic linear model and a cubic linear model. The graph of the fitted curves is shown in Figure 3.15. It is clear that the quadratic model is a big improvement with respect to the simple linear model from Figure 3.13. Nonetheless, it is not able to capture the exact relation. Instead, the cubic linear model manages to link temperature to the demand more accurately. This can also be seen by looking at the adjusted R squared: for the quadratic model, the score is 0.703, while for the cubic one, it is 0.858. We can extend the research also on higher-degree polynomials. In this case, we investigate the polynomial models by looking at the Bayesian Information Criterion (BIC).

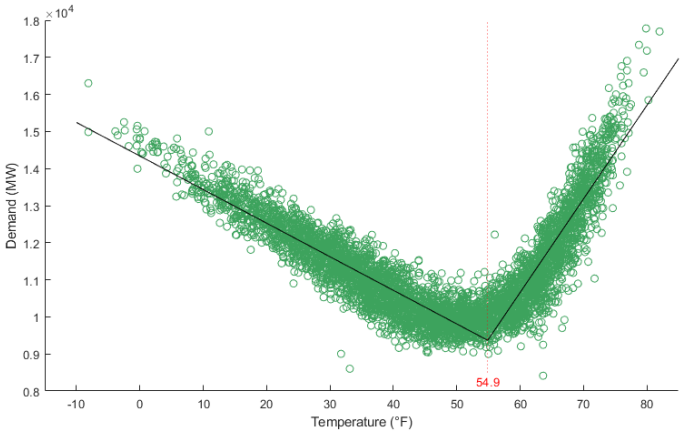

Figure 3.14: New England's dry bulb temperature × demand, at 1 am, with piecewise linear model.

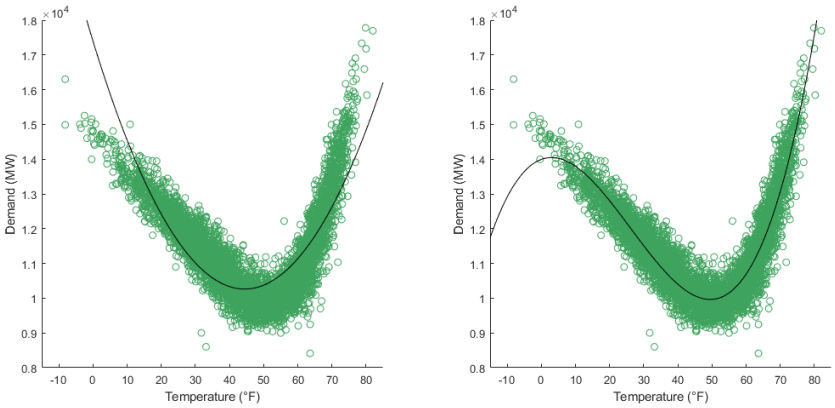

Figure 3.15: New England's dry bulb temperature × demand, at 1 am, with fitted quadratic and cubic polynomials.

We have investigated the scores until the fourth-degree polynomial, and we applied this procedure to the following cases: aggregated load at 1 am; aggregated load for all 24 hours, in which the average between the hourly BICs is displayed; subregional load for all 24 hours, in which the average between the hourly and regional BICs is displayed, and each subregional model considers all the temperatures from all regions as regressors; subregional load for all 24 hours, in which the average between the hourly and regional BICs is displayed, and each subregional model only considers the temperatures in that specific area as a regressor. The results are available in Table 3.1.

By looking at the table, we can see that the results for a higher polynomial are generally better, even though in the case when the subregional loads use all the other temperatures as regressors, the BIC is higher for the degree 4 polynomial. For the other cases, we can notice that after the third-degree polynomial, the BIC only decreases by a slight amount, indicating that the cubic polynomial may be sufficient. Furthermore, we can see that the aggregated load at 1 am has better results than when considering all the hours together. This happens because the relationship is less noisy at 1 am than for other hours of the day. Moreover, an interesting conclusion can be drawn from

Degree	1	2	3	4
Aggregated load, 1 am	8.882	8.261	7.883	7.869
Aggregated load	9.232	8.802	8.627	8.622
Subregional loads	6.895	6.530	6.465	6.706
Independent subregional loads	6.909	6.546	6.431	6.430

Table 3.1: BIC$\cdot 10^{-4}$, on the columns the degree of the polynomial relationship between demand and dry bulb temperature, on the rows the case of study.

the two subregional loads score. From the table, we can observe that after degree 3, the subregional models perform better when only the subregion-specific temperature is used. From this, we can assume that the relationship between load and temperature is subregion specific. Following this observation, when creating the models for Electrical Load Forecasting, it may be wise to consider subregion-specific emission distributions.

If we apply the same procedure for the dew point temperature, we obtain similar results. In Figure 3.16, we show the fitted fourth-degree polynomial over the dew point temperature, and we can see that the model's accuracy looks similar to the dry bulb case. In particular, when analysing again the BIC criterion, we find that again, higher degree polynomials are indeed performing better, but the scores are slightly higher than in the case for dry bulb temperature. This means that dry bulb temperature is a better predictor of the load.

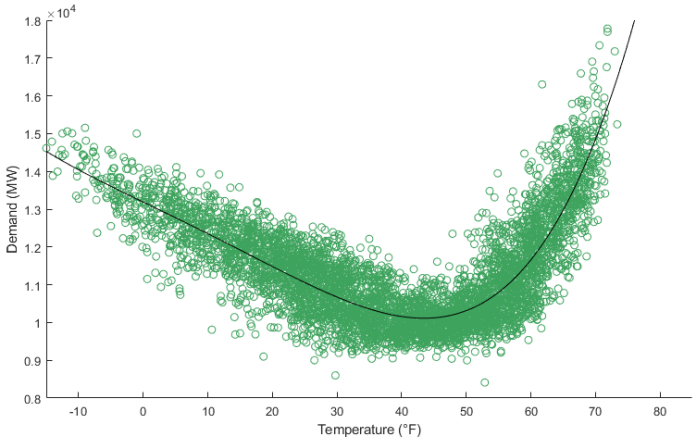

Figure 3.16: New England's dew point temperature × demand, at 1 am, with fitted degree 4 polynomial.

Finally, by fitting a four-degree polynomial using both dry bulb and dew point temperature, we find the best results. This implies that, even though dew point temperature performs worse than dry bulb temperature, it still contains valuable information that manages to improve the results obtained using only dry bulb temperature.

3.2.3 KF emission model: $\mathbf{Y}_t|\mathbf{X}_t$

Now we analyse the relationship between load and temperature that is present in the Kalman Filter emission distribution, which means that this time we model the load to predict the dry bulb and dew point temperature. Let us start by looking at Figure 3.17, which shows the scatter plot between the aggregated load and the dry bulb temperature at 1 am. From the plot we can clearly see that, there is a crucial issue in creating a linear model to predict the temperature starting from the load. In fact, there will not be any polynomial linear model that manages to predict the temperature accurately. This is because the curve formed by the cloud of points is not a function. For example, let us suppose that we observe a value of 14 GW in demand (red dotted line). In this situation, we do not know if this high demand is given by very high or very low temperatures. The temperature could in fact be around 10°F or around 70°F, and there is no way to tell the difference.

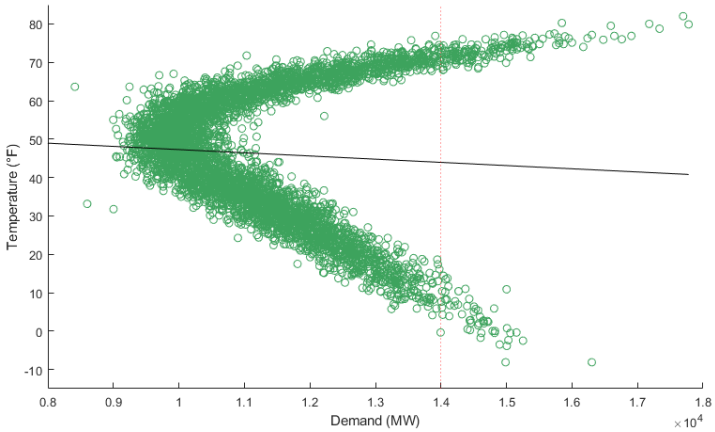

Figure 3.17: New England's demand × dry bulb temperature, at 1 am, with linear model.

One solution to tackle this problem would be to expand the space of the response variable, adding polynomial terms for the different temperatures. This way would prove effective since in the previous subsection it was shown that a linear relationship between the load and the polynomial temperatures does indeed exist. Another solution to deal with this problem is to consider also seasonal information that is present in the data. In fact, we know that if we find ourselves in winter, a high demand probably corresponds to a low temperature, and vice versa if we are in summer, high demand is the consequence of warm weather. This leads to the idea of adding a dummy variable that corresponds to the two winter and summer categories. For this analysis, and by looking at the best split in the data, we define winter as the union of the following months: October, November, December, January, February, March and April, while summer is the union of the remaining months.

We can now fit the linear model with the new categorical variable, using also an interaction term in order to change also the slope of the two lines. The results of this division is shown in Figure 3.18, in which the cloud of points is divided into two categories. In red, there are the summer observations and in green the ones in winter.

Furthermore, we added the lines corresponding to the new fitted linear models: in black the summer linear model and in blue the winter one.

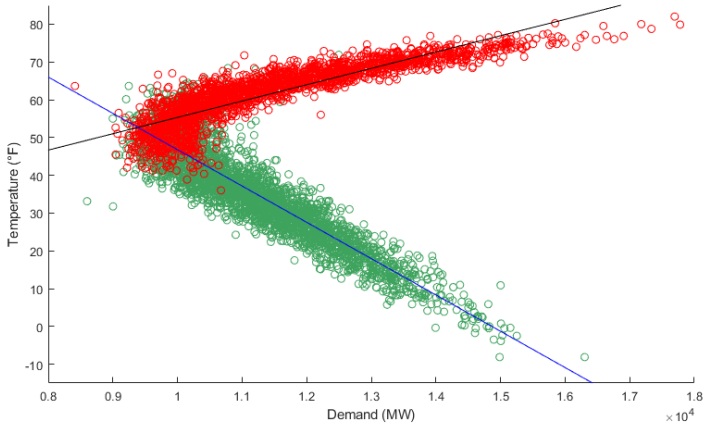

Figure 3.18: New England's demand × dry bulb temperature, at 1 am, with two seasonal linear models. In red, we have summer observations, in green winter observations.

It is already clear by looking at the plot that these two categories create a sufficiently good split between summer and winter observations, and that both clouds of points may be described using a polynomial function. When creating the two linear models, we have already good results, with an R squared score of 0.911. Remember that the scores are not comparable with the ones from the MAPLF emission distribution. When instead of two simple linear models, we use two linear quadratic models, the score slightly increases to 0.915.

In this section, we analysed three different relationships. At first, we showed that there exists a linear relationship between the demand at two consecutive hours of the day, and also that the parameters of this model are hour-specific. Then we showed that the load can be successfully modeled starting from the temperatures, but we need to rely on either piecewise linear models or on polynomial regression, since a simple linear model is not sufficient. This remark will be useful when we will analyse the accuracy of the forecasting models in Subsection 3.4.2. Moreover, when using polynomial regression, we have noticed that when we need to model the load for a specific region, the accuracy do not benefit from the knowledge of other regions' temperatures. At last, we found that the accuracy of the load forecasting algorithms will probably be improved by using a seasonal dummy variable.

3.3 Evaluation metrics

In order to make an informative comparison between the different forecasting methods, a set of metrics that evaluate the accuracy of the predictions need to be defined. All the forecasts consist of multivariate probabilistic forecasts, and therefore, different types of evaluation metrics can be used. We first start presenting three metrics that are used for univariate forecasts, and at the end, we present a method used for multivariate forecasts.

3.3.1 Single-value forecast metrics

At first, single-value forecast metrics are introduced. In particular, since the forecasts consist of a normal distribution, the mean will be adopted as a point forecast. There are many different metrics that are found in the literature and a complete study can be found in Hyndman and Koehler (2006). For this study, the focus is set on the two most commonly used ones. Furthermore, we first concentrate on the univariate case and apply these measures separately for each forecasted component.

Let x_i be the actual real value of the load, and let f_i be the forecasted value.

Root mean squared error (RMSE)

Root mean squared error is one of the most common measures for evaluating the accuracy of several point forecasts.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - f_i)^2}$$

Like MSE, it is a scale-dependent metric, but it is usually preferred since it has the same scale as the original time series and is therefore more interpretable. In addition, for its scale-dependency, we should pay attention when comparing the values for the different time series in the multivariate loads. One way to mitigate this issue is to normalize the various time series. Furthermore, this metric is sensible to outliers, since a large deviation between the forecast and the real value leads to a large value for $(x_i - f_i)^2$.

Mean absolute percentage error (MAPE)

Another common metric that is used is the mean absolute percentage error.

$$\text{MAPE} = \frac{100}{n} \cdot \sum_{i=1}^{n} \left| \frac{x_i - f_i}{x_i} \right|$$

This metric is a modification of the Mean Absolute Error (MAE) metric that overcomes the issue of scale dependency by dividing the error by the observed value x_i. However, one problem with MAPE is that it cannot be used when the time series points are or are close to zero since the term in the denominator will lead to values that are or are near to infinity. Moreover, MAPE puts a heavier penalty on positive errors, so modifications such as the symmetric mean absolute error have been proposed.

3.3.2 Probabilistic forecast metrics

Probabilistic forecast metrics are in general less utilised than point forecast measures. An analysis of probabilistic forecast evaluation and a brief summary of the main methods is conducted in Hong and Fan (2016).

Pinball loss

Pinball loss is probably the most common metric for evaluating the quantiles of the probabilistic forecast. Let x_i be the real value of the load, and let $f_i^{(q)}$ be the q-th predicted quantile.

$$\text{Pinball}^{(q)} = \frac{1}{n}\sum_{i=1}^{n} p_i^{(q)}, \qquad \text{where } p_i^{(q)} = \begin{cases} (1-q)(f_i^{(q)} - x_i) & \text{if } x_i < f_i^{(q)} \\ q(x_i - f_i^{(q)}) & \text{if } x_i \geq f_i^{(q)} \end{cases}$$

Pinball loss corresponds to the function that has to be minimized in quantile regression. The lower the value, the better the quantiles are estimated. When using it to measure the accuracy of the whole multivariate distribution, the problem of retrieving the quantiles arises. Therefore, when adopting this metric to evaluate a multivariate time series, the score is applied to each component independently and then the mean establishes the total loss.

3.3.3 Multivariate probabilistic forecast metrics

Multivariate probabilistic forecasts require further investigation. Univariate probabilistic forecast evaluation metrics are not sufficient and often there is not a direct translation into a multivariate scenario. For example, while the q-th quantile for a univariate distribution is easily determined, when working on multiple dimensions, an infinite number of quantiles can be found. New metrics are required, and to tackle this problem, in Bjerregård et al. (2021) the authors propose three methods for multivariate forecasts evaluation. We will focus only on one of these.

Let $\mathbf{x}_i$ be the real value of the load, and let $f_i(\cdot)$ be the multivariate density function of the forecasted distribution.

Logarithmic score

The Logarithmic score is a simple metric that originates from information theory. This is defined by

$$\text{LogScore} = \frac{1}{n}\sum_{i=1}^{n} -\log(f_i(\mathbf{x}_i))$$

This score is equivalent to the negative log-likelihood of the forecast model, and when working on Gaussian distributions, the Logarithmic score is proportional to the Dawid-Sebastiani score. The score is easy to compute when the density is available, and when there is no density, a kernel density estimate is adopted. One downside of this score is that it hardly penalizes unlikely observations.

3.4 Electricity Load Forecasting: aggregated load

Having analysed the data and the relationship between the variables, it is time to apply the electricity load forecasting methods introduced in Chapter 2 to forecast New England's demand. The four methods that will be analyzed are Kalman Filter, MAPLF, the

Inverted State-Space Model and the Vector Autoregressive model. The fourth method is used to understand the importance of the temperature variable.

The procedure for testing the accuracy of the methods is the following. For each model:

1. the parameters of the model are trained for T_{tr} days;

2. afterwards, for a period of T_{pr} days, for each day the following process is conducted: the model creates a 24 hours ahead forecast with the data collected until that day, the accuracy is calculated using the four metrics that were discussed, subsequently, the real observations are included in the training data and used to update the parameters.

This procedure is used to simulate a real-world scenario, in which an electric power company needs to compute a one-day ahead forecast and during the following day it can evaluate the forecast, by comparing it with the real consumption. We should point out that the temperatures that are used to produce these 24-hour forecasts consist of real observations. In a real-world application, this information would instead be formed by accurate temperature forecasts, and therefore the prediction may be less accurate.

Furthermore, all observations, i.e., both the loads and the exogenous time series, are normalized so that all the data is between 0 and 1 using the following formula:

$$\frac{x_t - \min_t x_t}{\max_t x_t - \min_t x_t}.$$

This normalization was used in Guo et al. (2022), and for the scope of this thesis, it was preferred to the standardization, since for the latter, errors in calculating the MAPE score may occur.

In this section, we present an analysis that only considers New England's aggregated load, and in Section 3.5 we will study the predictions for all of New England's subregions. The parameters are trained starting from January 1st 2004 for a period of 2 years, and afterwards, the daily prediction+training procedure is applied for a period of 3 years, until the end of December 2008. For all four models, we update the parameters with a value $\lambda = 1$ as exponential weight, which means that the parameters are equivalent to the ones found with the OLS estimator. To evaluate the predictions, MAPE, RMSE, Pinball score (with quantiles ranging from 0.1 to 0.9) and Logarithmic score are used.

For the first round of forecasting, only the hour of the day is used as calendar type. The latent variable is given by the univariate aggregated load, while the observed variable is given by the two-dimensional vector that comprises New England's average for both dry bulb and dew point temperatures.

The forecasting results can be seen in Table 3.2. The columns represent the technique that has been used to create the forecast, and the rows the specific evaluation score that has been used to calculate the score. The table shows that the best model is the Inverted SSM according to MAPE and Logarithmic score, while the simple VAR model performs better according to the RMSE and Pinball loss. Furthermore, the forecast given by Kalman Filter is in general more accurate than the one from the MAPLF algorithm.

	MAPLF	Kalman Filter	Inverted SSM	VAR
MAPE	15.80	14.24	12.84	13.01
RMSE $\cdot 10^2$	7.77	6.54	6.05	5.95
Pinball Loss $\cdot 10^2$	2.74	2.35	2.12	2.10
Log Score	-0.37	-1.40	-1.57	-1.48

Table 3.2: Forecasting results using New England's aggregated load and the temperatures, on the columns the forecasting technique, on the rows the evaluation metric that was used.

Now we look at the plot of the predictions for a random winter week. Figure 3.19 shows the forecasts from January 30th to February 5th 2006. When looking at the green line that corresponds to the real load curve, we can clearly see that weekend's demand cannot be forecasted successfully. This was expected, since no calendar type corresponding to the type of day was included in the model, and moreover, the choice of $\lambda = 1$ does not allow for dynamic changes in the models. Furthermore, the forecasts from Kalman Filter and Inverted SSM are very close to each other, while the MAPLF forecast appears more robust to changes between the days. In general for a given algorithm, the forecasts for each day are similar, and this may be determined by the choice of $\lambda = 1$, which does not consider the effects of recent changes.

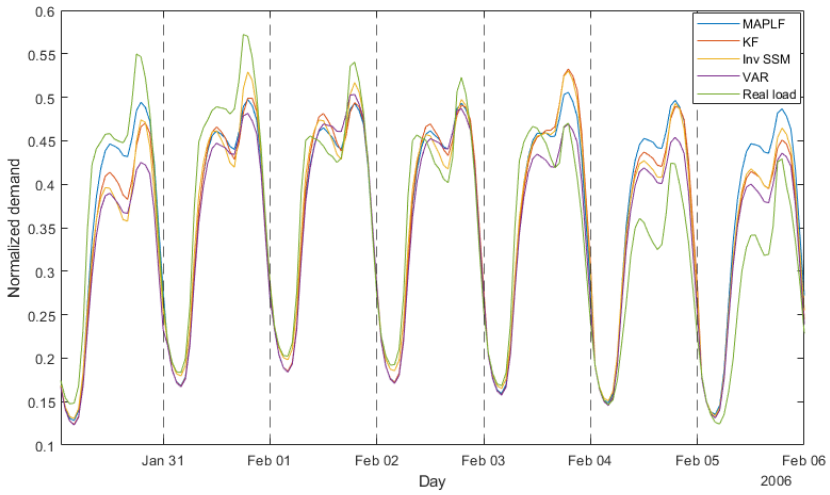

Figure 3.19: New England's total demand together with the predictions given by the four forecasting methods, 30 Jan. 2006 - 5 Febr. 2006.

3.4.1 Type of day influence

When introducing calendar types in section 2.1.3, it was mentioned that in order to improve prediction, one way would be to distinguish between workdays and weekends. Indeed, consumption patterns change significantly between these two types of days. In particular, now we will modify the calendar types set so that we can indicate whether a

specific day is a workday or a non-workday, which includes weekends and public holidays.

	MAPLF	KF	Inv. SSM	VAR
MAPE	$12.25_{-22\%}$	$10.00_{-30\%}$	$7.39_{-44\%}$	$8.70_{-33\%}$
RMSE $\cdot 10^2$	$6.50_{-16\%}$	$5.21_{-20\%}$	$4.06_{-33\%}$	$4.43_{-26\%}$
Pinball Loss $\cdot 10^2$	$2.27_{-17\%}$	$1.76_{-25\%}$	$1.32_{-38\%}$	$1.47_{-30\%}$
Log Score	$-0.73_{-0.36}$	$-1.65_{-0.25}$	$-2.00_{-0.43}$	$-1.71_{-0.23}$

Table 3.3: Forecasting results using New England's aggregated load, the temperatures and information regarding the type of day. In red the differences with Table 3.2 are shown.

The results for the 24-hour forecasts are shown in Table 3.3. Additionally to the scores, in red we show also the improvement with respect to the metrics from Table 3.2. The improvements are shown in percentage for the first 3 scores, while for the Log score, we show the absolute improvement since this score can be both positive and negative.

From the table, it is clear that adding the type of day significantly improves all results. Moreover, we can clearly see that inverted SSM is now the method with the best results for all four scores, and it is also the one with the most improved results. The second best method is the simple VAR, which shows a big improvement as well. The third best method is KF and at the last position, we have MAPLF, which is the worst method, and also the one with the least improvements.

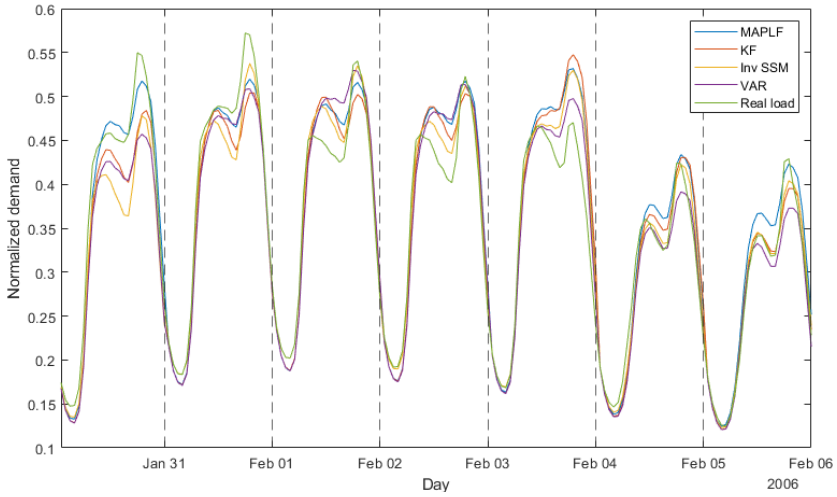

Figure 3.20: New England's total demand together with the predictions given by the 4 forecasting methods, with day type, 30 Jan. - 5 Febr. 2006.

To have a further look at the effects of the type of day, we can see how the prediction changes graphically. Taking as reference the predictions from Figure 3.19, we want to see how the predictions change. Figure 3.20 shows the same plot with the new predictions. It is clear that now the weekends are modelled much better. Consequently, also workdays are predicted slightly better since the specific parameters that are learnt do not contain

observations of holidays or weekends. From now on, all predictions will distinguish between workdays and weekdays.

3.4.2 Polynomial temperatures

In the two previous scenarios, we have seen that only the Inverted SSM is able to score generally better than the VAR model. This may lead to the conclusion that, for Kalman Filer and MAPLF, the temperature is not able to add important information to enhance the forecast.

Nonetheless, it is important to notice that until this moment, we have simply used as observed variable the vector that contains the temperature. In Subsections 3.2.2 and 3.2.3, we have studied the relationships between the load and the temperatures, and we found that a simple linear relationship is not sufficient to model this relationship. Therefore, now we will study how the forecasts are affected when we change the vector of observed variables. In particular, we will add a quadratic and a cubic term for both dry bulb and dew point temperatures to the vector of observed variables. The results of this modification are shown in Table 3.4, in which in red we have the improvement with respect to the results in Table 3.3.

	MAPLF	KF	Inv. SSM	VAR
MAPE	$6.40_{-48\%}$	$6.46_{-35\%}$	$5.10_{-31\%}$	8.70
RMSE $\cdot 10^2$	$3.01_{-54\%}$	$3.10_{-40\%}$	$2.69_{-34\%}$	4.43
Pinball Loss $\cdot 10^2$	$1.10_{-52\%}$	$1.14_{-37\%}$	$0.93_{-32\%}$	1.55
Log Score	$-1.87_{-1.14}$	$-1.86_{-0.21}$	$-2.04_{-0.04}$	-1.71

Table 3.4: Forecasting results using New England's aggregated load, polynomial temperatures up to the third-degree and information regarding the type of day. In red the differences with Table 3.3 are shown.

By looking at the scores, MAPLF is definitely the method that has enhanced its prediction the most, with an improvement of around 50% in all the scores. Furthermore, we can see that in this case, MAPLF is performing better than KF and that the Logarithmic score has decreased significantly. This means that now the multivariate probabilistic forecast obtained using MAPLF is more reliable. The second method that has significantly increased its performance is the one based on Kalman Filter. Finally, we have the Inverse SSM, which achieves gains that are slightly above 30%. In the table, we can also observe the VAR model, which is left untouched by the changes in the observed variables and has now the worse scores among all methods.

This significant improvement in the prediction's accuracy implies that the observed variable, which in our case is the temperature, plays an important role in enhancing the forecast. In particular, the Inverted State-Space Model reaches an improvement with respect to the VAR model of 41% for the MAPE, 39% for the RMSE and 40% for the Pinball loss. The Logarithmic score is also smaller, by an amount of 0.33.

These gains in accuracy highlight that the simple temperature vector is not able to use the temperature information successfully. On the other hand, if we add quadratic

and cubic terms, we can predict the load much more accurately. Since the idea of adding polynomial terms comes from the observations in Subsection 3.2.2, it follows that it is important to conduct a preliminary analysis of the relationship between the latent variable and the observed variable.

In Subsection 3.2.3 we have seen that another way to better capture the relationship between load and temperature is to distinguish between winter and summer observations. We will consider that the winter season starts in October and ends in April. To implement the seasonality into the emission distribution, we use a winter dummy variable, which is 1 if the observations occur in winter, and 0 otherwise. In particular, for each component that is currently present in the observed variable vector (which contains also quadratic and cubic terms), we add a copy of it which multiplies the dummy variable. In this way, we let the model learn season-specific polynomial relationships.

The results are shown in Table 3.5, where the improvements with respect to the previous table without seasonality are shown in red. The improvements that occur are in general around 6% for each method and for each score, reaching overall a very good forecasting performance for these load prediction techniques. Furthermore, we expect that the seasonal dummy variable is mainly useful when $\lambda = 1$, since when $\lambda < 1$, the dynamical seasonal patterns are modelled by the choice of the hyperparameter. We will see the effects of this in the next subsection.

	MAPLF	KF	Inv. SSM
MAPE	$6.03_{(-6\%)}$	$6.06_{(-6\%)}$	$4.76_{(-7\%)}$
RMSE $\cdot 10^2$	$2.85_{(-5\%)}$	$2.91_{(-6\%)}$	$2.52_{(-6\%)}$
Pinball Loss $\cdot 10^2$	$1.03_{(-6\%)}$	$1.06_{(-7\%)}$	$0.86_{(-8\%)}$
Log Score	$-1.93_{(-0.06)}$	$-1.94_{(-0.08)}$	$-2.10_{(-0.06)}$

Table 3.5: Forecasting results using New England's aggregated load, polynomial temperatures up to the third-degree, a winter dummy variable and information regarding the type of day. In red the differences with Table 3.4 are shown.

3.4.3 Exponentially weighted likelihood: the effect of λ

In Section 2.6, the hyperparameter λ has been introduced for estimating the parameters of the different distributions. This parameter determines the importance of each observation in the definition of the support structures. In particular, this weight is given by λ^{n-i}, where n is the number of observations and i is the index of each observation, with $i = n$ being the most recent one. To be valid, λ needs to be a positive number, and usually, it is set between 0 and 1. If the value of λ is 1, then all observations are considered equally in their contribution to the likelihood function, and therefore, the estimated parameters equal the ones obtained by the multivariate Ordinary Least Squares method. If on the other hand, $\lambda < 1$, then the most recent observations will have a higher weight than the older ones. This will lead to the possibility of dynamic changes in the models because bigger importance is given to the newly observed loads and temperatures.

As the likelihood functions are clearly model-dependent, we may find different optimal values for λ for each forecasting technique. In the remaining of the section, we look at each of the algorithms in turn.

Inverted SSM

We start by looking at how the choice of λ impacts the scores of the inverted SSM, by looking at different versions of the model and comparing the scores across a range of values of lambda.

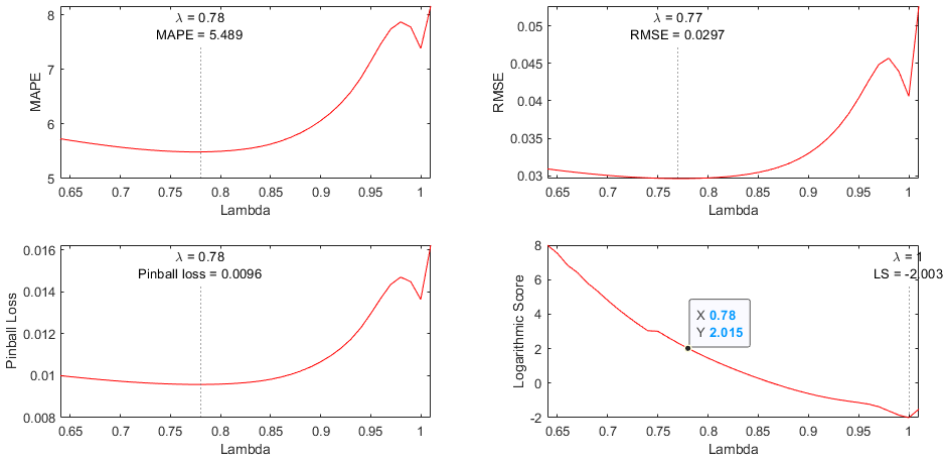

Figure 3.21: MAPE, RMSE, Pinball loss and Logarithmic score plots, in which on the x-axis there is λ, and on the y- axis, the evaluation metric for the Inverted SSM algorithm is computed. The model uses the temperature linearly, and the point of minimum score is highlighted.

Since we do not know how the meteorological variables impact the model when $\lambda < 1$, we start to look at the scores in the case in which we model the relationship between load and temperature linearly, i.e., we only use dry bulb and dew point temperatures in the observed variables vector. To see the impact of different values for λ, we plot the 4 scores with respect to the value of λ. Figure 3.21 shows the results for $0.64 < \lambda < 1.01$. Additionally, in the plot, the point of minimum is highlighted for each score. We can see that the minimum for MAPE, RMSE and Pinball Loss is attained around $\lambda = 0.78$, and, at that point, the model reaches sufficiently good results in terms of performance. The results are not as good as the ones we found in table 3.5, but considering that we use the temperature linearly, there is a big improvement using values for λ that are smaller than 1. Nonetheless, the logarithmic score for $\lambda = 0.78$ is equal to 2.015, which is a rather poor score. This is in contrast with the other scores' results, and in particular, this leads to different conclusions about the importance of the hyperparameter λ: MAPE, RMSE and Pinball loss affirm that λ should be equal to 0.78, while the Logarithmic score requires λ to be equal to 1.

Let us now introduce quadratic and cubic temperature terms in the observed variables. Figure 3.22 shows the results for $0.9 < \lambda < 1.01$. In this case, we reach the best results for MAPE, RMSE and Pinball Loss when λ is around 0.95. In particular, the values for these scores at the optimal point are very similar to the ones in the Inverted

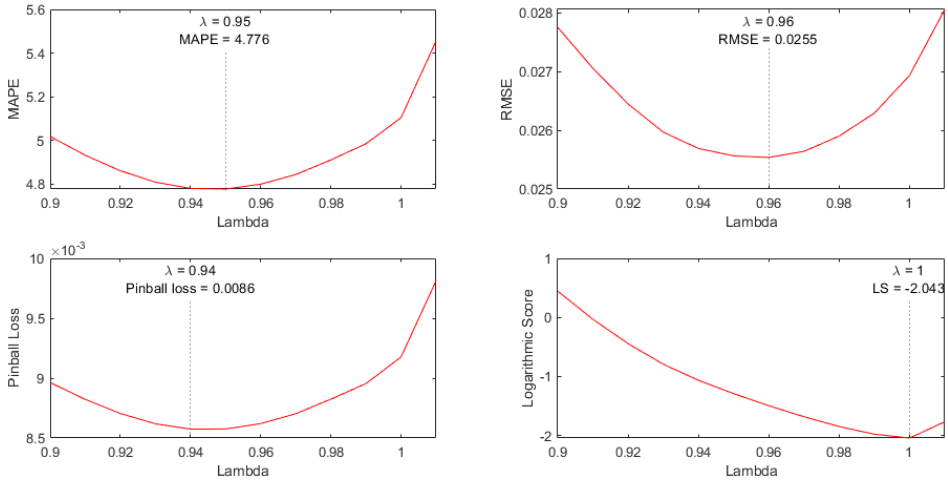

Figure 3.22: MAPE, RMSE, Pinball loss and Logarithmic score plots for the Inverted SSM algorithm. The model uses the temperature with additional quadratic and cubic terms, and the point of minimum score is highlighted.

SSM column of Table 3.5. Nevertheless, as before, the logarithmic score reaches the best results only when $\lambda = 1$. This again poses a problem, since we do not know which λ is preferred in a probabilistic forecast, as it depends on the measure being used. For this reason, the model from the previous subsection with a cubic relationship between load and temperature and with the seasonal dummy variables may be preferred.

Finally, we point out that when using values for λ that are smaller than 1 together with the winter dummy variable, the results are all worse than when using $\lambda = 1$. The only value that is barely better is the RMSE that reaches the minimum when $\lambda = 0.99$ with a value of 0.0251.

MAPLF

Now we turn our attention to the MAPLF model. We directly choose to use the model that contains also quadratic and cubic temperatures, without the seasonal dummy variable. In the case of MAPLF, the values for λ are two: one for the transition distribution and one for the emission one. This means that the space in which to look for the minimum is much bigger than before. In particular, this means that instead of a line, we will be looking at a surface.

Figure 3.23 shows the surfaces for each score, in which for the transition model, $0.6 < \lambda < 1.02$ and for the emission model, $0.86 < \lambda < 1.02$. First of all, we can see that in general, the optimal λ for the emission distribution is 1 or close to 1. This is also probably due to the fact that we are using a higher degree polynomial to model the relationship, which is flexible enough to account for recent changes. Instead, regarding the λ for the transition distribution, the optimal λ value appears to be much lower. Indeed, for MAPE, RMSE and Pinball loss, the best λ is around 0.8. Furthermore, at the minimum, the scores have decreased significantly. When looking at the differences with the results in Table 3.4, which corresponds to the case in which both transition and emission λs are 1, we have that MAPE decreases of 25%, RMSE of 17% and Pinball loss

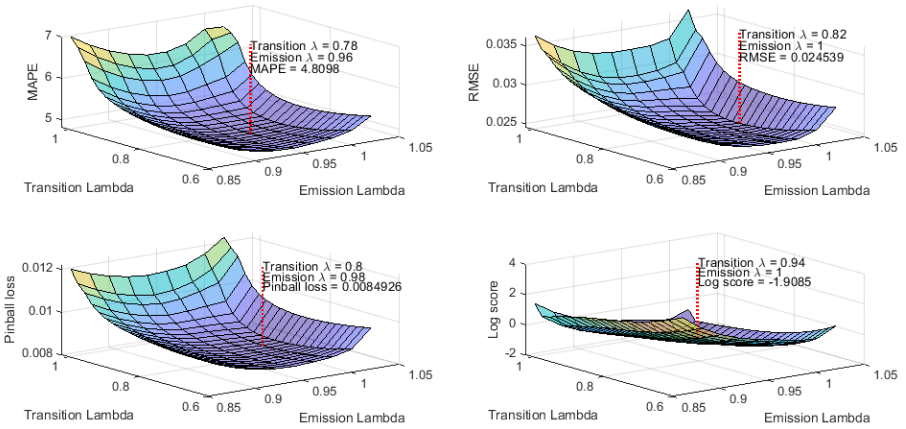

Figure 3.23: MAPE, RMSE, Pinball loss and Logarithmic score plots, in which on the x and y-axis there are the transition and the emission λs, and on the z- axis, the evaluation metric for the MAPLF algorithm is computed. The model uses the temperature with additional quadratic and cubic terms, and the point of minimum score is highlighted.

of 23%. This means that for MAPLF, changing the values for λ will definitely help.

On the other hand, as observed for Inverted SSM, the Logarithmic score points to different conclusions. According to this metric, the value for λ relative to the transition probability should be much closer to 1, around 0.96.

Kalman Filter

When we apply the same procedure also for Kalman Filter, we obtain Figure 3.24, in which we have used the same intervals for λ.

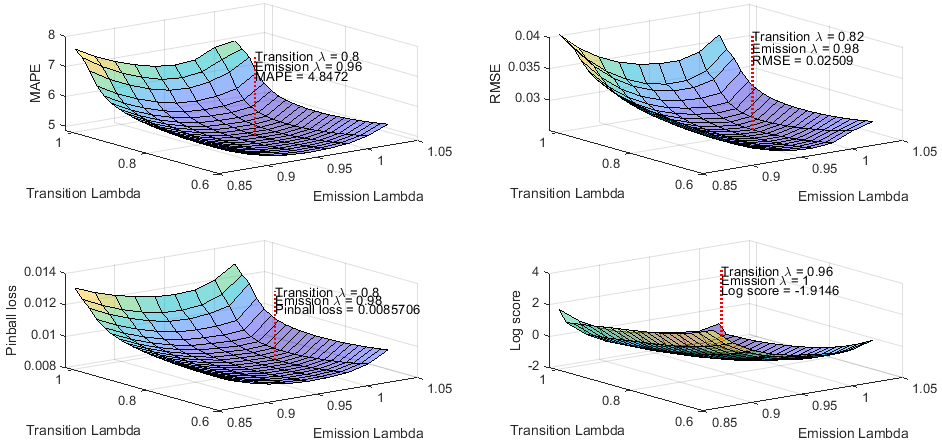

Figure 3.24: MAPE, RMSE, Pinball loss and Logarithmic score plots, in which on the x and y-axis there are the transition and the emission λs, and on the z- axis, the evaluation metric for the Kalman Filter algorithm is computed. The model uses the temperature with additional quadratic and cubic terms, and the point of minimum score is highlighted.

As for MAPLF, by looking at MAPE, RMSE and Pinball loss, the optimal λ for

the transition distribution is around 0.8, while for the emission distribution it is around 0.98. The scores that are obtained at the minimum are very similar to the ones obtained by MAPLF: MAPE of 4.85 (-25%), RMSE of 0.025 (-19%) and Pinball loss of 0.0086 (-25%). Again, when looking at the Logarithmic score the conclusions change, with the best λ for the transition distribution which is 0.96, and the best value for the emission distribution which is equal to 1.

VAR

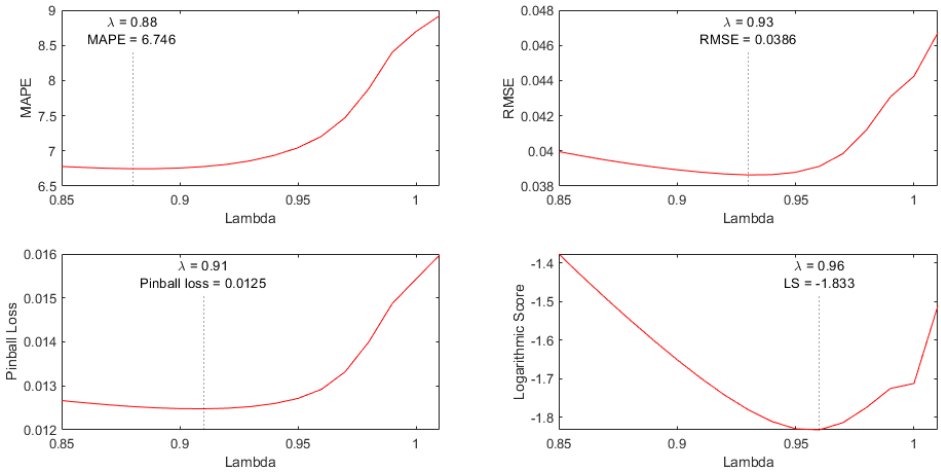

Figure 3.25: MAPE, RMSE, Pinball loss and Logarithmic score plots for the VAR model. The point of minimum score is highlighted.

Finally, we look at the effects of changing λ for the VAR model. In this case, we will have only one hyperparameter and we investigate the results over the interval $0.85 < \lambda < 1.01$. Figure 3.25 shows that according to MAPE, the best score is obtained considering $\lambda = 0.88$. In this case, the MAPE is equal to 6.75, which is a 22% decrease with respect to Table 3.4. Furthermore, RMSE and Pinball loss have a minimum around $\lambda = 0.92$ and in this case, there is a decrease of respectively 13% and 19%. This time the minimum according to the Logarithmic score is not when $\lambda = 1$, but when λ is equal to 0.96.

3.5 Electricity Load Forecasting: subregional load

One peculiarity of the forecasting techniques that were proposed is the possibility to predict multivariate load distributions. Therefore, we now turn our attention to the forecast of all 8 subregions of the New England dataset. In particular, the subregions are Maine (ME), New Hampshire (NH), Vermont (VT), Connecticut (CT), Rhode Island (RI), Southeastern Massachusetts (SEM), Western Central Massachusetts (WCM) and Northeastern Massachusetts (NEM).

We use the same procedure as for the aggregated load, in which we first train the parameters for a period of 2 years, and afterwards, for a period of 3 years, for each day we provide a 24-hour forecast and then update the parameters. We will start again using

$\lambda = 1$ for all models, and we will consider the type of day in the calendar types in all the results.

Furthermore, all the time series are again normalized so that all the observations lie between 0 and 1. This permits better comparisons between the regions since the effect caused by the magnitude of the regional loads are accounted for. Furthermore, this enables the comparison also when using scale-dependent scores, like RMSE.

3.5.1 Global temperatures

To start the analysis, we first use the global temperatures that have been used when predicting the aggregated load. These temperatures are a mean of the subregional temperatures. In the following analysis, the latent variable is made of the vector which contains the load of all 8 subregions. The observed variable is made of the two-dimensional vector which contains the values for the aggregated dry bulb and dew point temperatures.

Table 3.6 shows the scores that are obtained using the multivariate forecasting algorithms that have been presented in Chapter 2. There are four subtables that are shown, one for each evaluation metric. For MAPE, RMSE and Pinball loss, the columns represent the region, and the rows indicate the forecasting method. In this case, the regions that obtain the best and worse results are highlighted using respectively a green and a red asterisk. Instead, the subtable referring to the Logarithmic score shows one single score for each forecasting method.

First of all, we can see that there is a bit of heterogeneity between different regions. When looking at MAPE, the region with the most accurate forecasts is Western Central Massachusetts, and there are three regions with the worst accuracy, which vary depending on the forecasting method that was used: Southeastern Massachusetts, Maine and Northeastern Massachusetts. According to RMSE and Pinball loss, on the other hand, the regions with the best predictions are Rhode Island and Western Central Massachusetts, while the one with the worst predictions is Maine. Furthermore, MAPLF is the method that has the most difficulties in creating accurate forecasts, followed by Kalman Filtering, while Inverted SSM and VAR have very similar scores.

In the column relative to the mean score between the regions, a percentage is added in red. This percentage indicates the improvement of the model with respect to the score that would be obtained if each regional component would be trained and predicted separately. The complete results from the independent models are shown in Appendix A.3. We can see that each method benefitted from the dependency between the variables. This means that the interactions between the different loads are important for enhancing the forecast. When looking at the percentages, we see that Kalman Filter had an important improvement of around 25%, MAPLF of around 19% and Inverted SSM of around 11%. The method that had the smallest gain is the VAR model.

3.5.2 Subregional temperatures

Now we want to find out whether the inclusion of region-specific temperature for each model can increase the accuracy or not, compared to using the aggregated temperature, as in the results up to now.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
	MAPE								
MAPLF	10.6	7.7	8.1	9.0	8.4	11.5*	6.9*	10.9	9.1 (−19%)
KF	9.9*	6.5	7.2	6.9	6.6	8.7	5.6*	8.9	7.5 (−26%)
Inv. SSM	9.4*	6.1	7.0	6.4	6.1	8.1	5.3*	8.4	7.1 (−11%)
VAR	8.4	6.2	6.4	6.4	6.1	8.6	5.4*	9.0*	7.1 (−12%)
	RMSE $\cdot 10^2$								
MAPLF	5.9*	5.0	5.5	5.2	4.9	5.4	4.7*	5.4	5.2 (−18%)
KF	6.0*	4.2	5.0	4.1	3.8*	4.1	3.9	4.3	4.5 (−24%)
Inv. SSM	5.7*	4.1	4.9	4.0	3.6*	3.9	3.8	4.2	4.3 (−10%)
VAR	4.8*	4.1	4.3	4.0	3.7*	4.1	3.8	4.4	4.2 (−6%)
	Pinball loss $\cdot 10^2$								
MAPLF	2.1*	1.7	2.0	1.9	1.7	1.8	1.6*	1.8	1.8 (−19%)
KF	2.0*	1.4	1.7	1.4	1.3*	1.4	1.3	1.4	1.5 (−26%)
Inv. SSM	2.0*	1.3	1.7	1.3	1.2*	1.3	1.3	1.3	1.4 (−13%)
VAR	1.7*	1.3	1.5	1.3	1.2*	1.4	1.3	1.4	1.4 (−9%)

Logarithmic score			
MAPLF	KF	Inv. SSM	VAR
−11.60	−20.41	−20.69	−21.68

Table 3.6: Forecasting results using New England's subregional load and the temperatures. For MAPE, RMSE and Pinball loss, on the columns the region, on the rows the forecasting technique that was used. For the Logarithmic score, on the columns, the technique is represented. In red the differences with Table A.1 are shown.

When trying to run the algorithms for the multivariate load forecast, we come across a problem: the emission covariance matrices for MAPLF and KF and the covariance matrix for the distribution of the Inverted SSM are singular or close to singular. This makes it impossible to forecast the demand. One way to solve this issue may be to consider fewer regions so that the structure of the covariance matrices remains simpler and therefore invertible. This would then involve an additional study to choose which regions to consider together.

Instead, another way to look at the influence of subregional loads is to run the forecasting algorithms using the subregional temperatures from one specific region. Intuitively, the region from which the temperature is taken will perform slightly better, while the other regions will predict the load slightly worse. Then, we extract the subregion-specific prediction from the model and apply these procedures to all eight regions. From here we can have an idea of how much the subregional temperatures improve the predictions. Using these procedures and looking at the mean of the MAPE score, by comparing it to the results from Table 3.6, we obtain the following:

- MAPLF will actually perform generally worse, with a worsening of 0.33%.
- KF will improve an average of 4.46%.

- Inverted SSM will also improve, showing a decrease in the mean MAPE of 2.36%.

We can see that there hasn't been a significant improvement regarding the forecast accuracy, and the MAPLF method even obtained worse results. Another way to investigate the influence of subregional temperatures would be to look at how the independent models are enhanced by the use of these variables. Table A.2, in the Appendix, shows the evaluation metrics score that we obtain when running this scenario. In particular, all the metrics apart from RMSE are showing slightly worse results when using subregion-specific temperatures with respect to the global ones.

The observations collected in this subsection lead to the conclusion that, especially for computational reasons, it is preferred to use models that consider global temperatures. A scenario in which the use of subregion-specific temperatures may be beneficial is if we would consider regions that have a more diverse climate between each other.

3.5.3 Polynomial temperatures

The most accurate forecasts that we have obtained thus far were the ones produced by the VAR model in Table 3.6. Using this result as a baseline, we now take a look at how to improve the performance of the other methods by looking at the meteorological variables. The weather variables we will be using are the global ones, for the reasons that were explained in the previous subsection. Moreover, we will be using both dry bulb and dew point temperatures, including quadratic and cubic terms, respectively. Also, we will add the winter dummy variable.

At first, we analyse the MAPE. Table 3.7 shows the MAPE of the different models with the new variables. Furthermore, the improvements (in percentage) with respect to the VAR model results are shown.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
VAR	8.4	6.2	6.4	6.4	6.1	8.6	5.4	9.0	7.1
MAPLF	8.2	4.6	5.5	5.1	4.7	6.2	3.8	6.3	5.6
(%)	-2%	-26%	-14%	-20%	-23%	-28%	-30%	-30%	-21%
KF	8.6	4.9	6.3	5.3	5.2	6.3	4.0	6.5	5.9
(%)	+2%	-21%	-2%	-17%	-15%	-27%	-26%	-28%	-17%
In. SSM	8.2	4.6	6.2	4.8	4.8	6.0	3.8	6.2	5.6
(%)	-2%	-26%	-3%	-25%	-21%	-30%	-30%	-31%	-21%

Table 3.7: MAPE results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable. On the columns the region, on the rows the forecasting technique that was used. In red, the improvements with respect to the VAR model.

It is clear that the inclusion of quadratic and cubic terms for the temperatures increases the accuracy significantly. By looking at the MAPE, in general, Inverted SSM and MAPLF both obtain the best scores, followed by Kalman Filter. Nonetheless, we need to point out that all three models obtain remarkably similar scores. The general

gain with respect to the VAR model is around 20%, which is a significant improvement. The region that has the highest errors, Maine, is also the region which has the smallest gains, and in this region, Kalman Filter is not able to exceed VAR's forecast accuracy. On the contrary, Western Central and Northeastern Massachusetts show the best scores and also the biggest decrease in error with respect to the VAR model. Moreover, MAPLF, Kalman Filter and Inverted SSM all achieve a significant improvement with respect to Table 3.6, with MAPLF holding an important 38% average decrease in MAPE.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
VAR	4.8	4.1	4.3	4.0	3.7	4.1	3.8	4.4	4.1
MAPLF	4.5	2.9	3.8	3.0	2.7	2.8	2.5	2.7	3.1
(%)	-6%	-29%	-12%	-25%	-27%	-32%	-34%	-39%	-24%
KF	5.2	3.2	4.4	3.3	3.0	3.0	2.7	2.7	3.4
(%)	+8%	-22%	+2%	-18%	-19%	-27%	-29%	-39%	-17%
In. SSM	5.0	3.1	4.3	3.1	2.7	2.8	2.6	2.6	3.3
(%)	+4%	-24%	-	-23%	-27%	-32%	-32%	-41%	-20%

Table 3.8: RMSE $\cdot 10^2$ results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable. In red, the improvements with respect to the VAR model.

Table 3.8 shows the results in terms of RMSE. According to this score, the best method is MAPLF, followed by Inverted SSM and Kalman Filter. Also in this case, the improvements are around 20%. The regions with the worst and best scores are again Maine and Western Central Massachusetts, respectively. Moreover, for both Maine and Vermont, Kalman Filter and Inverted SSM are not able to achieve better accuracies than the VAR model.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
VAR	1.68	1.32	1.50	1.32	1.19	1.35	1.25	1.40	1.38
MAPLF	1.60	0.94	1.33	1.04	0.90	0.95	0.85	0.95	1.07
(%)	-5%	-29%	-11%	-21%	-24%	-30%	-32%	-32%	-22%
KF	1.74	1.02	1.53	1.10	1.00	0.98	0.91	0.95	1.15
(%)	+4%	-23%	+2%	-17%	-16%	-27%	-27%	-32%	-17%
In. SSM	1.69	0.97	1.50	1.01	0.92	0.92	0.86	0.91	1.10
(%)	+1%	-27%	-	-23%	-23%	-32%	-31%	-35%	-20%

Table 3.9: Pinball loss $\cdot 10^2$ results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable. In red, the improvements with respect to the VAR model.

Now we would like to evaluate the forecast in terms of the distribution. Therefore, let us start by looking at the Pinball loss score. Table 3.9 shows the results and the improvements in percentage with respect to the loss that was incurred by the VAR model.

Notice that the Pinball loss evaluates only the distribution of each component separately. According to this score, the multiple univariate probabilistic forecasts are improving as well. Once again, MAPLF, KF and Inverted SSM are reaching very similar results, with MAPLF scoring slightly better in general. The second best method is Inverted SSM followed by Kalman Filter. Similarly to the other scores, Maine and Vermont are again the regions for which the methods have more difficulties in forecasting the correct load, while Western Central Massachusetts is the one that produces the most accurate predictions. From these results, we may assume that the univariate probabilistic forecasts are enhanced by the presence of quadratic and cubic terms for the temperature variables.

Finally, let us now take a look at the evaluation for the multivariate probabilistic forecast given by the Logarithmic score. Table 3.10 shows the scores that we obtain using the polynomial terms for the temperatures and compares them to the scores that were obtained in Table 3.6.

	MAPLF	KF	Inv. SSM	VAR
Logscore in Tab. 3.6	-11.60	-20.41	-20.69	-21.68
Logscore	-12.59	-18.07	-18.57	-21.68
Difference	-0.99	$+2.34$	$+2.12$	-

Table 3.10: Logarithmic score results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable. A comparison with the Logarithmic score in Table 3.6 is shown.

Surprisingly, by looking at the Logarithmic score it is clear that the VAR model is still the best model for multivariate prediction. It obtains a score of -21.68, which is lower than the one from the other methods. Furthermore, only the MAPLF model slightly improved, while the Kalman Filter and the Inverted SSM even decreased the accuracy of the probabilistic forecast. Among the methods that use the temperature variable, Inverted SSM is the best-performing technique, followed by Kalman Filter and the MAPLF. The big difference in score between MAPLF and the other methods hints at the fact that this technique is not able to predict the covariance matrix successfully.

These observations indicate that if we are interested in a point forecast or in the univariate distribution of the regions, MAPLF, Kalman Filter and Inverted SSM all prove to be stable and efficient algorithms for multivariate forecasts. On the other hand, when evaluating the structure of the covariance matrix by looking at the Logarithmic score, the simpler Vector Autoregressive model proves to be the most accurate and also the most computationally efficient algorithm.

In the Appendix, Table A.3 shows the evaluation metrics scores that are calculated for the same scenario using independent models. The scores are significantly worse.

3.5.4 The effect of λ

Let us now investigate the influence of the hyperparameter λ. For simplicity, this time we will only study the effects on the Inverted SSM model.

Let us start by considering the model that contains the quadratic and cubic temperature terms, but without the addition of the winter dummy variable. Taking $0.9 < \lambda < 1.01$, the results are shown in Figure 3.26, in which the best value for λ for the average between the regional load metrics is highlighted.

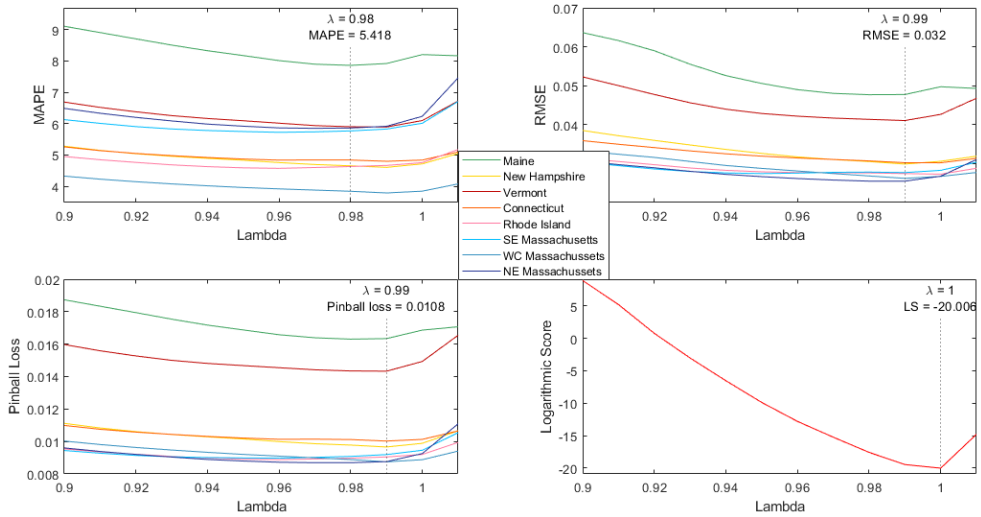

Figure 3.26: MAPE, RMSE, Pinball loss and Logarithmic score plots, in which on the x-axis there is λ, and on the y- axis, the evaluation metric for the Inverted SSM algorithm is computed. For MAPE, RMSE and Pinball loss, the subregional score is shown. The model uses the temperature with additional quadratic and cubic terms, and the point of minimum average score is highlighted.

It is evident how the different regions have similar behaviours when λ changes. Nonetheless, there are exceptions. For example for Rhode Island, the best λ according to RMSE is equal to 1. For the regional mean, according to MAPE, RMSE and Pinball loss, the best λ is between 0.98 and 0.99. Looking at the subregional curves, this seems to be a reasonable value also for the different regions. At the minimum, there is a slight improvement for MAPE, RMSE and Pinball loss, and the results are slightly better than the ones obtained using $\lambda = 1$ together with the seasonal dummy variable, which are shown in Tables 3.7, 3.8 and 3.9.

Finally, according to the Logarithmic score, the best λ is equal to 1. This is coherent with the behaviour that was observed in Subsection 3.4.3. The score at the minimum is equal to -20.01, which is lower than the one in Table 3.10. Since the minimum is reached for $\lambda = 1$, this indicates that regarding the multivariate probabilistic forecast, the winter dummy variable worsens the forecast accuracy.

3.6 Different forecast horizons

In Section 3.4 and 3.5, we displayed the forecasting results for 24-hour ahead forecasts. To complete the analysis about the accuracy of the forecasting algorithms that have been proposed, let us now look at what happens if we modify the time horizon of the forecasts. We will investigate two scenarios: a one-hour ahead forecast, in which the

load is only predicted for the next hour, and a two-weeks ahead forecast, in which a longer-term prediction is analysed. Regarding the latter case, we should recall that the temperature that is used to create the forecasts consists of real observations. When using these models to create real-world forecasts, these observations will be replaced by accurate weather forecasts. This will definitely influence the models' accuracy, especially for long-term forecasts.

In the following subsections, the latent variable consists of the vector containing the eight subregional loads, and the observed variable consists of the global temperatures, including quadratic and cubic terms, and the winter dummy variable.

3.6.1 One-hour ahead forecast

The first case consists of one-hour ahead forecasts. In this scenario, the forecasts will be more accurate, since the prediction errors do not accumulate.

	MAPLF	Kalman Filter	Inverted SSM	VAR
MAPE	2.06	1.99	1.73	2.12
RMSE $\cdot 10^2$	1.14	1.11	0.97	1.17
Pinball Loss $\cdot 10^3$	3.90	3.78	3.27	4.02
Log Score	-30.06	-30.53	-30.78	-30.48

Table 3.11: One-hour ahead average forecasting results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable. On the columns the forecasting technique, on the rows the evaluation metric that was used.

Table 3.11 shows the results for the evaluation metrics for each forecasting method. For MAPE, RMSE and Pinball loss, the average score is shown. As we may expect, the scores are all significantly smaller than the ones we found in Tables 3.7, 3.8, 3.9 and 3.10, since the demand is simpler to predict. Looking at the outcomes, Inverted SSM is the method that reaches the best accuracy according to all the metrics. In this case, also the Logarithmic score shows a lower value than for the other methods, which is in contrast with what we observed in previous cases. The second-best method is Kalman Filter, and also this technique reaches a better Logarithmic score than the VAR model. Moreover, MAPLF is the third-best method according to MAPE, RMSE and Pinball loss, while instead, the Logarithmic score shows poor results and the VAR model is performing better according to this metric.

Let us now investigate the subregional performances. Figure 3.27 shows four plots, one for each evaluation metric, in which the score for the four forecasting techniques is shown. For MAPE, RMSE and Pinball loss, also the regional score is shown and is represented on the x-axis. From the plots, it is clear that for all four metrics and for all subregions, the Inverted SSM is the method that creates the most accurate forecasts. Moreover, MAPLF, Kalman Filter and VAR model all perform similarly for all subregions, and the ranking between these three methods, which was described previously, remains stable for all regions. Furthermore, it is noticeable that both Southeastern and

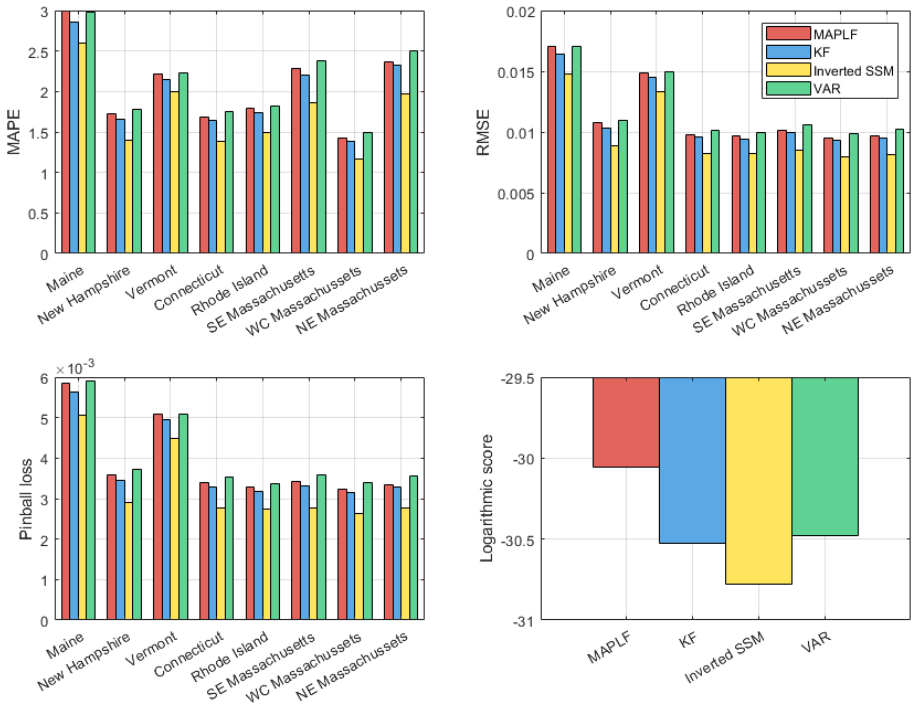

Figure 3.27: MAPE, RMSE, Pinball loss and Logarithmic score plots evaluated on one-hour ahead forecasts. For MAPE, RMSE and Pinball loss, on the x axis the region is indicated, and on the y-axis the score for each method is shown. For the Logarithmic score, on the x-axis, the method is shown, and on the y-axis there is the corresponding score.

Northeastern Massachusetts have relatively imprecise forecasts according to MAPLF, while instead, they obtain good performance when looking at RMSE and Pinball loss.

3.6.2 Two-weeks ahead forecast

Let us now turn our attention to the two-weeks long load forecasts. In this scenario, we expect the forecasting algorithms to produce less accurate forecasts since the prediction errors will accumulate along the time horizon.

	MAPLF	Kalman Filter	Inverted SSM	VAR
MAPE	6.87	8.78	8.63	16.65
RMSE $\cdot 10^2$	3.36	4.31	4.25	7.55
Pinball Loss $\cdot 10^2$	1.22	1.54	1.52	2.77
Log Score	-6.80	-9.60	-9.89	-17.74

Table 3.12: Two-weeks ahead average forecasting results using New England's subregional load, polynomial temperature up to the third degree and a winter dummy variable.

Table 3.12 shows the results for the evaluation metrics for each forecasting method.

For MAPE, RMSE and Pinball loss, the average score is shown. As expected, this time the scores are higher than the ones we found in Tables 3.7, 3.8, 3.9 and 3.10. According to MAPE, RMSE and Pinball loss, a big difference in scores between the VAR model and the other three forecasting techniques can be observed, with the former showing a significantly poorer forecasting performance. This happens because the only information that the VAR model is using is given by the inferred parameters and by the demand at the start of the forecast horizon. On the contrary, the other three models are able to enhance the forecast using the temperature information and have a bigger advantage in predicting the load far into the future. MAPLF is, among the three, the technique that is showing the best forecasting accuracy, and even though the forecasting horizon is 14 times longer, manages to obtain an RMSE that is only 8% bigger, with MAPE which increased by 18% and Pinball loss by 12%. The second best forecasting method is Inverted SSM, followed by Kalman Filter.

When looking at the Logarithmic score, the considerations change significantly. The most accurate method becomes the VAR model, followed, in this order, by Inverted SSM, Kalman Filter and finally MAPLF. This observation, together with the results from Table 3.11, lead to the conclusion that the bigger the forecasting horizon is, the more inaccurate the multivariate probabilistic predictions given by the temperature-using models become.

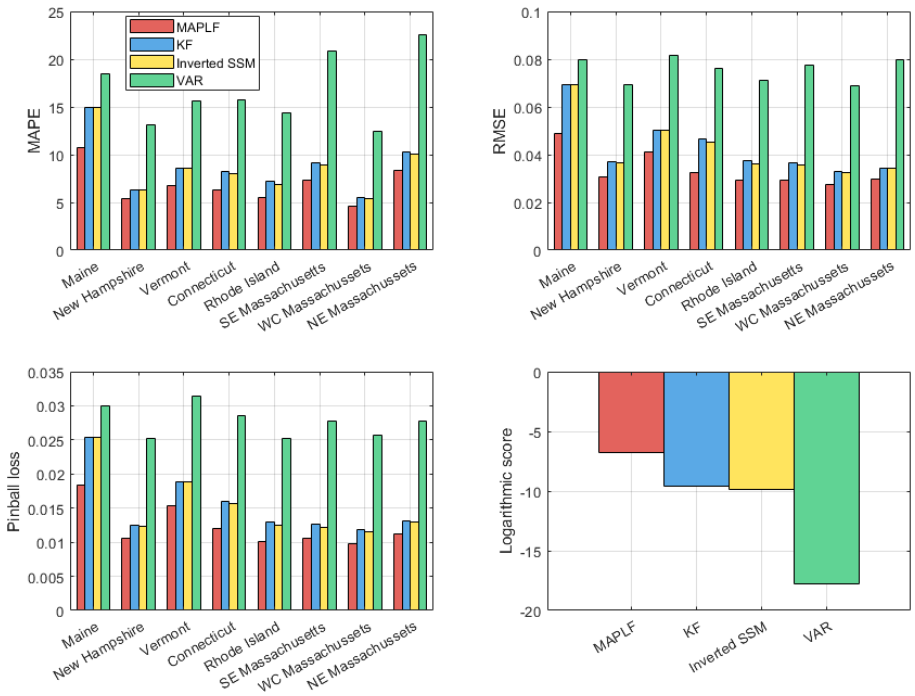

Figure 3.28: MAPE, RMSE, Pinball loss and Logarithmic score plots evaluated on two-weeks ahead forecasts. For MAPE, RMSE and Pinball loss, on the x axis the region is indicated, and on the y-axis the score for each method is shown. For the Logarithmic score, on the x-axis, the method is shown, and on the y-axis there is the corresponding score.

Let us investigate how the different subregions are performing. Figure 3.28 shows four plots, one for each evaluation metric, in which the score for the four forecasting

techniques is shown. For MAPE, RMSE and Pinball loss, also the regional score is shown and is represented on the x-axis. According to MAPE, RMSE and Pinball loss, the VAR model is clearly the technique that has the most difficulties in creating a reliable forecast for all regions. Kalman Filter and Inverted SSM have really similar scores through all the subregions, and MAPLF is the method that creates the most accurate predictions.

Finally, to analyse the accuracy of the single-value forecasts for two-week ahead predictions, we show Figure 3.29. In this plot, similar to Figures 3.19 and 3.20, we show the real and the predicted electricity demands for the state of Connecticut, during a standard winter week, from Monday, January 30th, to Sunday, February 5th 2006. In particular, all four algorithms are starting to forecast the load from Monday, January 23rd, which means that the figure shows the forecasts during the second week of predictions. From the plot, it appears evident that the VAR model is not able to capture the intra-day variability, and appears to be over-predicting the demand. On the contrary, MAPLF, Kalman Filter and Inverted SSM are able to change the daily pattern, because of the presence of the temperature variable. This is particularly noticeable by looking at the size of the evening demand peaks, which are very high on Monday and Tuesday but are significantly lower during the rest of the week. From this, it also follows that the electricity demand is highly temperature-dependent.

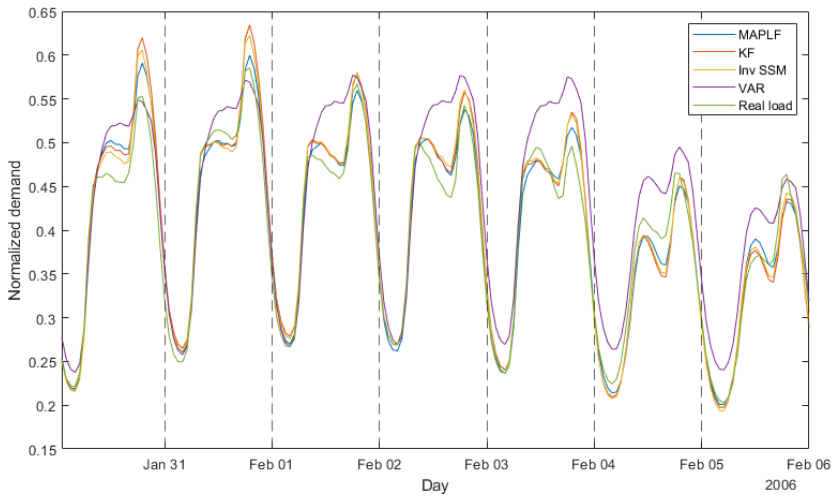

Figure 3.29: Connecticut's demand together with the predictions given by the four forecasting methods during the second week of forecast, 30 Jan. 2006 - 5 Febr. 2006.

3.7 Conclusions

Load forecasting plays an important role in the provision of electricity, and having efficient and accurate ways to predict demand, especially in a probabilistic manner, is crucial for our everyday lives. The objective of this thesis was to propose different statistical methods for multivariate electricity load forecasting, and to this end, State-Space Models were chosen since they allowed to enhance the load forecast by using also the information provided by external variables, for example the temperature. Additionally, the forecasts that are produced consist of the mean and covariance of a normal distribution, and can therefore be considered a probabilistic forecast, which is more reliable than a point forecast.

The first method that was described is Kalman Filtering. This method, based on Linear Gaussian State-Space Models, is a well-known forecasting method, that relies on a two-phase prediction process. In particular, the predictions are obtained in a recursive way, in which each prediction step relies on the forecasts estimated in the previous step.

The second method is called MAPLF and consists of a novel modification of Kalman Filtering based on the paper Álvarez et al. (2021), in which the emission distribution has been modified. In particular, this technique uses the emission distribution to model the load by using the external variables as regressors, enabling a more flexible and intuitive way to employ the additional observations. However, this method relies on the assumption that the underlying distribution of the latent variable can be modelled using an improper prior. Since this method is constructed upon Kalman Filter, it relies on recursive predictions. Another advantage of this algorithm is that it is computationally more efficient than Kalman Filter, especially when the number of observed variables is large.

The third method, called the Inverted State-Space Model, is another novel technique in which in the construction of the State-Space Model, the arrows between the latent and observed variables are inverted. In particular, it is a generalization of the Kalman Filter algorithm, in which the assumption of independence between the observed variable at time t and the latent variable at time $t-1$, given the latent variable at time t, is removed. Also this method relies on recursive prediction steps.

To analyse the importance of the external variables in the creation of the forecasts, an order one Vector Autoregressive model is introduced, which will serve as a benchmark. This model is constructed solely over the series of latent variables, and does not take into consideration the additional observed variables.

All the models that were described rely on estimated Gaussian distributions, and the parameters of these distributions will be inferred using the exponentially weighted least squares method. In particular, the parameters are learnt recursively, and this permits to update the estimates for each new observation that becomes available. Furthermore, thanks to the hyperparameter present in the weights, it can be used to dynamically adapt to changes in consumption patterns. To handle the intrinsic differences between each hour of the day and between work days and non-work days, specific calendar-type sets of parameters have been created. This means that the parameters from the different distributions are now calendar-type specific.

To study the accuracy of the different forecasts, the ISO New England dataset has

been chosen. An exploratory analysis of the variables of the dataset and the relationship between the load and the temperature has been conducted. From this examination, it is clear that the load between two consecutive hours can be modelled using a simple linear model, and that there exists a polynomial relationship between the load and the temperature. Furthermore, the knowledge of the season for which the forecast is created can additionally enhance the relationship between load and temperature.

The first case scenario in which the forecasting techniques have been applied is for the prediction of the aggregated load. It has been observed that the distinction between workdays and weekends improved the forecast significantly. Furthermore, the polynomial relationship between load and temperature was found beneficial also for the load forecasting algorithm, by decreasing the error scores by more than 35%, with an additional 6% decrease than can be reached by considering also a winter dummy variable.

To enable the parameters' adaptive learning, different values for the weight hyperparameter have been studied. It was found that in general, if the temperature is considered linearly, the tuning of λ can improve the forecasts greatly. If instead the temperature is included in the model in a polynomial form, the results are generally good and the tuning only leads to small gains in accuracy. In this case, the biggest improvements have been observed for MAPLF and Kalman Filter, because the fact that there are two hyperparameters to tune gives more space for refinement. It should be noted that when tuning the hyperparameter, MAPE, RMSE and Pinball loss pointed to values for λ that were smaller than 1, while the Logarithmic score performed better when λ was close to 1.

In a second analysis, the forecasting techniques were applied to the multivariate load time series. At first, it has been found that the usage of subregional temperatures did not improve the forecasts by a significant amount. Therefore, a global temperature variable turned out to be a much simpler and more efficient choice. This indicates that the global temperature contains the necessary information for subregional load prediction and that in this dataset, there is no big difference between the subregional climates. Moreover, the difference between independent and correlated models has been briefly studied, showing that the relationships between the different subregional loads positively influence the predictions, allowing the models to perform more accurate predictions. Also for multivariate load forecasting, using the temperature as a cubic polynomial increases significantly the precision of the predictions. When the temperature is considered linearly, the simpler VAR model is generally performing better according to all four scores. On the contrary, the use of polynomial temperatures enables Kalman Filter, MAPLF and Inverted SSM to reach error scores that are around 20% lower than the ones for the VAR model according to MAPE, RMSE and Pinball loss. It follows that for multivariate point forecasts and for the probabilistic predictions for each region, these three methods are performing better. One interesting observation is that by looking at the Logarithmic score, even with the cubic temperatures, the VAR model is the most accurate method for prediction. This may be determined by a more correct prediction of the correlation structure given by the Vector Autoregressive model forecasts.

Finally, a study on how different forecast horizons may influence the accuracy of the predictions was conducted. First, one-hour ahead forecasts were analysed, and all techniques produced very accurate predictions, with Inverted SSM that proved to be the best algorithm according to all four evaluation metrics. Afterwards, two weeks ahead

forecasts were examined. According to MAPE, RMSE and Pinball loss, the best forecasting algorithm was MAPLF, and the VAR model showed poor accuracy scores. This indicates that the temperature variable is a highly informative variable for electricity load prediction. Nevertheless, the VAR model achieved the best accuracy according to the Logarithmic score.

Looking at these results, we may conclude that Kalman Filter, MAPLF and Inverted SSM are reliable forecasting methods, but require an additional study regarding the relationship between the load and the additional observed variables. The VAR model, on the other hand, is a stable algorithm that permits obtaining good results even without this exploratory analysis. Furthermore, for multivariate probabilistic forecasts, this model may be preferred, since according to the Logarithmic score, it is able to forecast the correlation structure more accurately.

A further study using other multivariate probabilistic metrics may be conducted, in order to conclude if the VAR model is always better at predicting the multivariate distribution. Furthermore, a more in-depth analysis of the structure of the predicted covariance matrix could show how the different methods estimate the correlations, and an investigation into the structure of the corresponding precision matrix may highlight the partial correlation structure between the subregional loads. This could lead to the conclusion that it may be more efficient to estimate clusters of subregions independently, instead of all subregions together.

It should be taken into consideration that many conclusions are data-specific, which means that the results may change using different datasets. Furthermore, the type of dataset used is also important. For example, if multiple energy types should be forecasted instead of multiple regions, the impact of the correlations may change significantly. To this scope, it would be interesting to analyse the effects of these algorithms when forecasting renewable energy availability instead of demand, since the high dependence that renewable energy sources have on weather conditions may be modelled through the state-space framework defined in this thesis.

The study about the hyperparameter λ did not highlight the necessity of dynamic parameter learning when the cubic relationship between load and temperatures is considered. Nonetheless, since each calendar type has its own set of parameters, a further study may be conducted into hour-specific or day-type-specific hyperparameters. This will definitely increase the effort for parameter tuning, but it may capture certain hour-specific characteristics.

Finally, another interesting field of study may be the use of these models for hierarchical clustering. In fact, the enhanced predictions determined by the correlated multivariate models may improve the prediction of the aggregated forecast.

Appendix A

Appendix

A.1 Proof of lemmas

In this appendix, the proofs for the three lemmas concerning multivariate normal distributions are presented.

A.1.1 Lemma 2.2.1

Lemma 2.2.1 (Conditional distribution of a Multivariate Gaussian distribution). *Let's consider a multivariate Gaussian distribution of the form*

$$\begin{bmatrix}\mathbf{X}\\ \mathbf{Y}\end{bmatrix} \sim \mathcal{N}_{k_1+k_2}\left(\begin{bmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right)$$

where $\mathbf{X}$ *has dimension* k_1 *and* $\mathbf{Y}$ *has dimension* k_2.
Then, the conditional distribution of $\mathbf{X}$ *given* $\mathbf{Y}$ *is:*

$$\mathbf{X}|\mathbf{Y} = \mathbf{y} \sim \mathcal{N}_{k_1}\left(\boldsymbol{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right)$$

Proof. Let's start the proof by observing that

$$\begin{aligned} p(\mathbf{X}|\mathbf{Y}) &= \frac{p(\mathbf{X},\mathbf{Y})}{p(\mathbf{Y})} \\ &= \frac{(2\pi)^{-\frac{k_1+k_2}{2}}\begin{vmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{vmatrix}^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-\boldsymbol{\mu}_2\end{bmatrix}^t \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}^{-1} \begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-\boldsymbol{\mu}_2\end{bmatrix}\right)}{(2\pi)^{-\frac{k_2}{2}}\Sigma_{22}^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\right)} \end{aligned}$$

Let's call the target mean $\eta = \boldsymbol{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)$ and the covariance $\Gamma = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$. We have that Σ_{22} and Γ are invertible. From this, for the blockwise matrix inversion formula, we have

$$\begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}^{-1} = \begin{bmatrix}\Gamma^{-1} & -\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1}\\ -\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1} & \Sigma_{22}^{-1}+\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1}\end{bmatrix}$$

Let's now look at what we have inside the exponential on top.

$$\begin{aligned}
&\begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-\boldsymbol{\mu}_2\end{bmatrix}^t \begin{bmatrix}\Gamma^{-1} & -\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1}\\ -\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1} & \Sigma_{22}^{-1}+\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1}\end{bmatrix}\begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-\boldsymbol{\mu}_2\end{bmatrix}\\
&= (\mathbf{x}-\boldsymbol{\mu}_1)^t\Gamma^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) - 2(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)\\
&\quad + (\mathbf{y}-\boldsymbol{\mu}_2)^t(\Sigma_{22}^{-1}+\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1})(\mathbf{y}-\boldsymbol{\mu}_2)\\
&= \mathbf{x}^t\Gamma^{-1}\mathbf{x} - 2\boldsymbol{\mu}_1^t\Gamma^{-1}\mathbf{x} + \boldsymbol{\mu}_1^t\Gamma^{-1}\boldsymbol{\mu}_1 - 2(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\mathbf{x}\\
&\quad + 2(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\boldsymbol{\mu}_1\\
&\quad + (\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\Sigma_{12}\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2) + (\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\\
&= \mathbf{x}^t\Gamma^{-1}\mathbf{x} - 2\eta^t\Gamma^{-1}\mathbf{x} + \eta^t\Gamma^{-1}\boldsymbol{\mu}_1 + (\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}\Sigma_{21}\Gamma^{-1}\eta\\
&\quad + (\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\\
&= (\mathbf{x}-\eta)^t\Gamma^{-1}(\mathbf{x}-\eta) + (\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)
\end{aligned}$$

Furthermore, we have that

$$\begin{vmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{vmatrix} = |\Gamma|\cdot|\Sigma_{22}|$$

Now, putting everything together, we have that $p(\mathbf{X}|\mathbf{Y})$ is equal to

$$\begin{aligned}
&(2\pi)^{-\frac{k_1}{2}}\left(\frac{|\Gamma|\cdot|\Sigma_{22}|}{|\Sigma_{22}|}\right)^{-\frac{1}{2}}\frac{\exp\left(-\frac{1}{2}((\mathbf{x}-\eta)^t\Gamma^{-1}(\mathbf{x}-\eta)+(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2))\right)}{\exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_2)^t\Sigma_{22}^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\right)}\\
&= (2\pi)^{-\frac{k_1}{2}}|\Gamma|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{x}-\eta)^t\Gamma^{-1}(\mathbf{x}-\eta)\right)
\end{aligned}$$

□

A.1.2 Lemma 2.2.2

Lemma 2.2.2 (Product of Gaussian pdfs of related variables). *Let's consider two multivariate normal probability distribution functions of the form:*

- $\phi_{k_1}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)$
- $\phi_{k_2}(\mathbf{y};M\mathbf{x}+\boldsymbol{\mu}_2,\Sigma_2)$

Then, we have that the product between these two pdfs is equal to:

$$\phi_{(k_1+k_2)}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix};\begin{bmatrix}\boldsymbol{\mu}_1\\ M\boldsymbol{\mu}_1+\boldsymbol{\mu}_2\end{bmatrix},\begin{bmatrix}\Sigma_1 & \Sigma_1M^t\\ M\Sigma_1 & \Sigma_2+M\Sigma_1M^t\end{bmatrix}\right)$$

Proof. Let's start by looking at the target pdf. The part inside the exponential is:

$$-\frac{1}{2}\begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-M\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\end{bmatrix}^t\begin{bmatrix}\Sigma_1 & \Sigma_1M^t\\ M\Sigma_1 & \Sigma_2+M\Sigma_1M^t\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{x}-\boldsymbol{\mu}_1\\ \mathbf{y}-M\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\end{bmatrix}$$

Let's first compute the inverse of the covariance matrix. To do this, we will use the properties of the inversion of block matrices. In particular, we have

- Σ_1 is invertible;
- $(\Sigma_2 + M\Sigma_1 M^t) - M\Sigma_1(\Sigma_1)^{-1}\Sigma_1 M^t = \Sigma_2 + M\Sigma_1 M^t - M\Sigma_1 M^t = \Sigma_2$ is invertible.

It follows that

$$\begin{aligned}\begin{bmatrix} \Sigma_1 & \Sigma_1 M^t \\ M\Sigma_1 & \Sigma_2 + M\Sigma_1 M^t \end{bmatrix}^{-1} &= \begin{bmatrix} \Sigma_1^{-1} + \Sigma_1^{-1}\Sigma_1 M^t \Sigma_2^{-1} M \Sigma_1 \Sigma_1^{-1} & -\Sigma_1^{-1}\Sigma_1 M^t \Sigma_2^{-1} \\ -\Sigma_2^{-1} M \Sigma_1 \Sigma_1^{-1} & \Sigma_2^{-1} \end{bmatrix} \\ &= \begin{bmatrix} \Sigma_1^{-1} + M^t \Sigma_2^{-1} M & -M^t \Sigma_2^{-1} \\ -\Sigma_2^{-1} M & \Sigma_2^{-1} \end{bmatrix}\end{aligned}$$

We can return to the exponential:

$$\begin{aligned}&-\frac{1}{2}\begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_1 \\ \mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 \end{bmatrix}^t \begin{bmatrix} \Sigma_1^{-1} + M^t\Sigma_2^{-1}M & -M^t\Sigma_2^{-1} \\ -\Sigma_2^{-1}M & \Sigma_2^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_1 \\ \mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 \end{bmatrix} \\ &= -\frac{1}{2}\Big[(\mathbf{x} - \boldsymbol{\mu}_1)^t \left(\Sigma_1^{-1} + M^t\Sigma_2^{-1}M\right)(\mathbf{x} - \boldsymbol{\mu}_1) - 2(\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1}(\mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \\ &\quad + (\mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t \Sigma_2^{-1}(\mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\Big]\end{aligned}$$

The first term can be rewritten like

$$(\mathbf{x} - \boldsymbol{\mu}_1)^t \Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1) + (\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1} M(\mathbf{x} - \boldsymbol{\mu}_1)$$

and the last term like

$$\begin{aligned}&(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2 + M(\mathbf{x} - \boldsymbol{\mu}_1))^t \Sigma_2^{-1}(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2 + M(\mathbf{x} - \boldsymbol{\mu}_1)) \\ &= (\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2)^t \Sigma_2^{-1}(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2) + 2(\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1}(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2) \\ &\quad + (\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1} M(\mathbf{x} - \boldsymbol{\mu}_1) \\ &= (\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2)^t \Sigma_2^{-1}(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2) + 2(\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1}(\mathbf{y} - M\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \\ &\quad - (\mathbf{x} - \boldsymbol{\mu}_1)^t M^t \Sigma_2^{-1} M(\mathbf{x} - \boldsymbol{\mu}_1)\end{aligned}$$

Substituting the terms above, we have that some parts simplify out and in the exponential we are left with

$$-\frac{1}{2}\left((\mathbf{x} - \boldsymbol{\mu}_1)^t \Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1) + (\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2)^t \Sigma_2^{-1}(\mathbf{y} - M\mathbf{x} - \boldsymbol{\mu}_2)\right)$$

Let's now look at the determinant of the covariance matrix. For the properties of block matrices and the previous observations, we have

$$\begin{vmatrix} \Sigma_1 & \Sigma_1 M^t \\ M\Sigma_1 & \Sigma_2 + M\Sigma_1 M^t \end{vmatrix} = |\Sigma_1| \cdot |\Sigma_2|$$

With these properties, the lemma follows naturally.

□

A.1.3 Lemma 2.2.3

Lemma 2.2.3 (Product of Gaussian pdfs of the same variable). *Let's consider two multivariate normal probability density functions of the form:*

- $\phi_k(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1)$
- $\phi_k(\mathbf{x}; \boldsymbol{\mu}_2, \Sigma_2)$

Then, we have that the product between the two pdfs is equal to:

$$\phi_k\left(\mathbf{x}; \boldsymbol{\mu}^*, \Sigma^*\right) \cdot \phi_k(\boldsymbol{\mu}_1; \boldsymbol{\mu}_2, \Sigma_1 + \Sigma_2)$$

where $\Sigma^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$ *and* $\boldsymbol{\mu}^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2)$.

Proof. First, we can notice that if we multiply the two pdfs directly, we obtain

$$(2\pi)^k(|\Sigma_1| \cdot |\Sigma_2|)^{-1/2} \exp\left(-\frac{1}{2}[(\mathbf{x} - \boldsymbol{\mu}_1)^t\Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1) + (\mathbf{x} - \boldsymbol{\mu}_2)^t\Sigma_2^{-1}(\mathbf{x} - \boldsymbol{\mu}_2)]\right)$$

Let's now focus on the part inside the exponential. By expanding the products and isolating the part that contains $\mathbf{x}$:

$$-\frac{1}{2}\left[\mathbf{x}^t(\Sigma_1^{-1} + \Sigma_2^{-1})\mathbf{x} - 2(\boldsymbol{\mu}_1^t\Sigma_1^{-1} + \boldsymbol{\mu}_2^t\Sigma_2^{-1})\mathbf{x} + \boldsymbol{\mu}_1^t\Sigma_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}\boldsymbol{\mu}_2\right]$$

From this we can notice that the covariance matrix of the distribution regarding $\mathbf{x}$ will be $\Sigma^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$. Let $\boldsymbol{\mu}^*$ be the mean of this distribution. We have that the distribution of $\mathbf{x}$ will have

$$(\mathbf{x} - \boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}(\mathbf{x} - \boldsymbol{\mu}^*) = \mathbf{x}^t(\Sigma^*)^{-1}\mathbf{x} - 2(\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}\mathbf{x} + (\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}\boldsymbol{\mu}^*$$

inside the exponential.

From this we have that, equalizing the parts that depend on $\mathbf{x}$ linearly

$$\boldsymbol{\mu}_1^t\Sigma_1^{-1} + \boldsymbol{\mu}_2^t\Sigma_2^{-1} = (\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}$$

and consequently

$$\boldsymbol{\mu}^* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2)$$

We know that the distribution of $\mathbf{x}$ is

$$\phi_k\left(\mathbf{x}; \boldsymbol{\mu}^*, \Sigma^*\right) = (2\pi)^{k/2}(|\Sigma^*|)^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}(\mathbf{x} - \boldsymbol{\mu}^*)\right)$$

and therefore we can rewrite the product of the two distributions as

$$\phi_k\left(\mathbf{x}; \boldsymbol{\mu}^*, \Sigma^*\right) \cdot (2\pi)^{k/2}\left(\frac{|\Sigma_1| \cdot |\Sigma_2|}{|\Sigma^*|}\right)^{-1/2} \exp\left(-\frac{1}{2}\left[\boldsymbol{\mu}_1^t\Sigma_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}\boldsymbol{\mu}_2 - (\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}\boldsymbol{\mu}^*\right]\right)$$

Let's now focus on the quantity in the exponential and see if something simplifies out. First, we can expand $(\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}\boldsymbol{\mu}^*$:

$$\begin{aligned}
&(\boldsymbol{\mu}_1^t\Sigma_1^{-1} + \boldsymbol{\mu}_2^t\Sigma_2^{-1})(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2) \\
&\quad = (\boldsymbol{\mu}_1^t\Sigma_1^{-1} + \boldsymbol{\mu}_2^t\Sigma_2^{-1})(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2) \\
&\quad = \boldsymbol{\mu}_1^t\Sigma_1^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_2^{-1}\boldsymbol{\mu}_2 \\
&\qquad + \boldsymbol{\mu}_1^t\Sigma_1^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_2^{-1}\boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}\Sigma_1^{-1}\boldsymbol{\mu}_1
\end{aligned}$$

Looking at formula (163) from Petersen et al. (2012), we have that

$$(\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} = \Sigma_1(\Sigma_1 + \Sigma_2)^{-1}\Sigma_2 = \Sigma_2(\Sigma_1 + \Sigma_2)^{-1}\Sigma_1$$

hence the previous element becomes

$$\begin{aligned}
&\boldsymbol{\mu}_1^t\Sigma_1^{-1}\Sigma_2(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t(\Sigma_1 + \Sigma_2)^{-1}\Sigma_1\Sigma_2^{-1}\boldsymbol{\mu}_2 \\
&\quad + \boldsymbol{\mu}_1^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_1
\end{aligned}$$

Now we can notice that

$$\Sigma_1^{-1}\Sigma_2(\Sigma_1 + \Sigma_2)^{-1} = \Sigma_1^{-1}(\Sigma_1 + \Sigma_2 - \Sigma_1)(\Sigma_1 + \Sigma_2)^{-1} = \Sigma_1^{-1} - (\Sigma_1 + \Sigma_2)^{-1}$$

and analogously

$$(\Sigma_1 + \Sigma_2)^{-1}\Sigma_1\Sigma_2^{-1} = \Sigma_2^{-1} - (\Sigma_1 + \Sigma_2)^{-1}$$

Summarizing the operations, we have

$$\begin{aligned}
(\boldsymbol{\mu}^*)^t(\Sigma^*)^{-1}\boldsymbol{\mu}^* &= \boldsymbol{\mu}_1^t\Sigma_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}\boldsymbol{\mu}_2 + \boldsymbol{\mu}_1^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_1 \\
&\quad - \boldsymbol{\mu}_1^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^t(\Sigma_1 + \Sigma_2)^{-1}\boldsymbol{\mu}_2 \\
&= \boldsymbol{\mu}_1^t\Sigma_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^t\Sigma_2^{-1}\boldsymbol{\mu}_2 - (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t(\Sigma_1 + \Sigma_2)^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)
\end{aligned}$$

Looking back at the quantities inside the exponential, we have that many quantities simplified out, and what is left is:

$$\exp\left(-\frac{1}{2}\left[(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t(\Sigma_1 + \Sigma_2)^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\right]\right)$$

Finally, if we look at the fraction between determinants:

$$\begin{aligned}
\frac{|\Sigma_1| \cdot |\Sigma_2|}{|\Sigma^*|} &= \frac{|\Sigma_1| \cdot |\Sigma_2|}{|\Sigma_1(\Sigma_1 + \Sigma_2)^{-1}\Sigma_2|} = \frac{|\Sigma_1| \cdot |\Sigma_2|}{|\Sigma_1| \cdot |\Sigma_1 + \Sigma_2|^{-1} \cdot |\Sigma_2|} \\
&= |\Sigma_1 + \Sigma_2|
\end{aligned}$$

From this, the lemma follows naturally.

□

A.2 Parameter learning

When doing multivariate load forecasting, the data that is available consists of two multivariate time series. In particular, time series

- $\{\mathbf{s}_t\}$ that consists of load historical data at time $t = 1, 2, \ldots, T$;
- $\{\mathbf{y}_t\}$ that consists of the observational data. At time $t = 1, 2, \ldots, T$ it consists of historical data, but additional observations at time $t = T + 1, \ldots, T + R$ are also present.

Furthermore, using the calendar types that have been defined in section 2.1.3, a unique map $c : \{1, \ldots, T + R\} \to \Delta$ can be created, that maps each time point to the corresponding calendar type.

With this map, calendar type-specific time series can be defined. For a certain $k \in \Delta$ the

- load data is $\{\mathbf{s}_t^k\} = \{\mathbf{s}_i : c(i) = k \text{ and } 1 \leq i \leq T\} = \mathbf{s}_1^k, \ldots, \mathbf{s}_{n_k}^k$, where n_k is the total number of load samples belonging to calendar type k and the loads are ordered in chronological order;
- observation data is $\{\mathbf{y}_t^k\} = \{\mathbf{y}_i : c(i) = k \text{ and } 1 \leq i \leq T\} = \mathbf{y}_1^k, \ldots, \mathbf{y}_{n_k}^k$. In this case, only the historic observations until time T are considered.

This new partition of the data can now be used for parameter learning.

A.2.1 Kalman Filter

The goal of parameter learning for Kalman Filter is to infer the set of parameters $\{A^k, \boldsymbol{\alpha}^k, Q^k, B^k, \boldsymbol{\beta}^k, R^k\}$. This will be done for each $k \in \Delta$ independently.

Transition distribution parameters

For calendar type k, the transition distribution is

$$\mathbf{X}_t^k | \mathbf{X}_{t-1} = \mathbf{x}_{t-1} \sim \mathcal{N}_d(A^k \mathbf{x}_{t-1} + \boldsymbol{\alpha}^k, Q^k)$$

with parameters $A^k, \boldsymbol{\alpha}^k$ and Q^k.

The parameters will be learnt recursively. Since the regressor variable consists of one-step past information, the first observation will be skipped. This means that the response variable data for calendar type $c(1)$ will be $\{\mathbf{x}_t^{c(1)}\} = \mathbf{x}_2^{c(1)}, \ldots, \mathbf{x}_{n_{c(1)}}^{c(1)}$.

Returning to a general calendar type k, the response variable data will be $\{\mathbf{x}_t^k\} = \mathbf{x}_1^k, \ldots, \mathbf{x}_{n_k}^k$. The regressor data will instead consist of observations of a one-time step before in the original time series. For example, when considering hourly observations, for all the hours apart from 1 a.m., the regressors consist of $\{\mathbf{x}_t^{k-1}\} = \mathbf{x}_1^{k-1}, \ldots, \mathbf{x}_{n_k}^{k-1}$. For the situation during 1 a.m., the situation is slightly different. When the day type is not considered in the calendar type, the series to be used is $\{\mathbf{x}_t^{24}\} = \mathbf{x}_1^{24}, \ldots, \mathbf{x}_{n_1}^{24}$. If instead the day type is considered, the regressor is given by the 24-th hour of the day

type of the previous day. In general, the regressor series of the previous time step will be denoted by $\{\mathbf{p}_t^k\} = \mathbf{p}_1^k, \dots, \mathbf{p}_{n_k}^k$.

To start the recursion, the support structures H_0^k, J_0^k, K_0^k and γ_0^k are initialized to zero-matrices of respectively dimensions $(d+1) \times d, (d+1) \times (d+1), d \times d$ and 1. Furthermore, the value of $\lambda > 0$ is chosen, in general such that $\lambda \leq 1$.

Then, looking at the recursion from chapter 2.6.4, the parameters are found recursively for $j = 1, \dots, n_k$. First, the support structures are updated

- $H_j^k = \begin{bmatrix} 1 \\ \mathbf{p}_j^k \end{bmatrix} (\mathbf{x}_j^k)^t + \lambda H_{j-1}^k$
- $J_j^k = \begin{bmatrix} 1 \\ \mathbf{p}_j^k \end{bmatrix} \begin{bmatrix} 1 & (\mathbf{p}_j^k)^t \end{bmatrix} + \lambda J_{j-1}^k$
- $K_j^k = \mathbf{x}_j^k (\mathbf{x}_j^k)^t + \lambda K_{j-1}^k$
- $\gamma_j^k = 1 + \lambda \gamma_{j-1}^k$

and then the parameters are retrieved from the support structures via

- $\begin{bmatrix} (\hat{\boldsymbol{\alpha}}_j^k)^t \\ (\hat{A}_j^k)^t \end{bmatrix} = (J_j^k)^{-1} H_j^k$
- $\hat{Q}_j^k = \frac{1}{\gamma_j^k} \left(K_j^k - (H_j^k)^t \begin{bmatrix} (\hat{\boldsymbol{\alpha}}_j^k)^t \\ (\hat{A}_j^k)^t \end{bmatrix} \right)$

And it can be rewritten as

- $\begin{bmatrix} \hat{\boldsymbol{\alpha}}_j^k & \hat{A}_j^k \end{bmatrix} = (H_j^k)^t (J_j^k)^{-1}$
- $\hat{Q}_j^k = \frac{1}{\gamma_j^k} \left(K_j^k - (H_j^k)^t \begin{bmatrix} \hat{\boldsymbol{\alpha}}_j^k & \hat{A}_j^k \end{bmatrix}^t \right)$

As analyzed in the corresponding chapter, the parameters can be computed only after a few iterations, since at the beginning the matrix J_j^k is singular.

Emission distribution parameters

The emission distribution is

$$\mathbf{Y}_t^k | \mathbf{X}_t = \mathbf{x}_t \sim \mathcal{N}_d(B^k \mathbf{x}_t + \boldsymbol{\beta}^k, R^k)$$

with parameters $B^k, \boldsymbol{\beta}^k$ and R^k.

In this case, the response variable data consists of the series $\{\mathbf{y}_t^k\} = \mathbf{y}_1^k, \dots, \mathbf{y}_{n_k}^k$ and the regressor data of $\{\mathbf{x}_t^k\} = \mathbf{x}_1^k, \dots, \mathbf{x}_{n_k}^k$.

Like before, to start the recursion, the support structures H_0^k, J_0^k, K_0^k and γ_0^k are initialized to zero-matrices of respectively dimensions $d \times p, d \times d, p \times p$ and 1. Furthermore, the value of $\lambda > 0$ is chosen. For simplicity, the same notation as in the transition distribution is kept.

Then, for $j = 1, \dots, n_k$, the support structures are updated

- $H_j^k = \begin{bmatrix} 1 \\ \mathbf{x}_j^k \end{bmatrix} (\mathbf{y}_j^k)^t + \lambda H_{j-1}^k$
- $J_j^k = \begin{bmatrix} 1 \\ \mathbf{x}_j^k \end{bmatrix} \begin{bmatrix} 1 & (\mathbf{x}_j^k)^t \end{bmatrix} + \lambda J_{j-1}^k$
- $K_j^k = \mathbf{y}_j^k (\mathbf{y}_j^k)^t + \lambda K_{j-1}^k$
- $\gamma_j^k = 1 + \lambda \gamma_{j-1}^k$

and the parameters become

- $\begin{bmatrix} \hat{\boldsymbol{\beta}}_j^k & \hat{B}_j^k \end{bmatrix} = (H_j^k)^t (J_j^k)^{-1}$
- $\hat{R}_j^k = \frac{1}{\gamma_j^k} \left(K_j^k - (H_j^k)^t \begin{bmatrix} \hat{\boldsymbol{\beta}}_j^k & \hat{B}_j^k \end{bmatrix}^t \right)$

A.2.2 MAPLF

Let $k \in \Delta$. The parameters for the transition distribution are learnt using the same equations as for the Kalman filters.

Emission distribution parameters

The emission distribution is

$$\mathbf{X}_t^k | \mathbf{Y}_t = \mathbf{y}_t \sim \mathcal{N}_d(D^k \mathbf{y}_t + \boldsymbol{\delta}^k, P^k)$$

with parameters $D^k, \boldsymbol{\delta}^k$ and P^k.

In this case, the response variable data consists of the series $\{\mathbf{x}_t^k\} = \mathbf{x}_1^k, \ldots, \mathbf{x}_{n_k}^k$ and the regressor data of $\{\mathbf{y}_t^k\} = \mathbf{y}_1^k, \ldots, \mathbf{y}_{n_k}^k$.

Like before, to start the recursion, the support structures H_0^k, J_0^k, K_0^k and γ_0^k are initialized to zero-matrices of respectively dimensions $p \times d, p \times p, d \times d$ and 1. Furthermore, the value of $\lambda > 0$ is chosen.

Then, for $j = 1, \ldots, n_k$, the support structures are updated

- $H_j^k = \begin{bmatrix} 1 \\ \mathbf{y}_j^k \end{bmatrix} (\mathbf{x}_j^k)^t + \lambda H_{j-1}^k$
- $J_j^k = \begin{bmatrix} 1 \\ \mathbf{y}_j^k \end{bmatrix} \begin{bmatrix} 1 & (\mathbf{y}_j^k)^t \end{bmatrix} + \lambda J_{j-1}^k$
- $K_j^k = \mathbf{x}_j^k (\mathbf{x}_j^k)^t + \lambda K_{j-1}^k$
- $\gamma_j^k = 1 + \lambda \gamma_{j-1}^k$

and the parameters become

- $\begin{bmatrix} \hat{\boldsymbol{\tau}}_j^k & \hat{T}_j^k \end{bmatrix} = (H_j^k)^t (J_j^k)^{-1}$
- $\hat{P}_j^k = \frac{1}{\gamma_j^k} \left(K_j^k - (H_j^k)^t \begin{bmatrix} \hat{\boldsymbol{\tau}}_j^k & \hat{T}_j^k \end{bmatrix}^t \right)$

A.2.3 Inverted SSM

Let $k \in \Delta$.

The distribution is

$$\begin{bmatrix} \mathbf{X}_t^k \\ \mathbf{Y}_t^k \end{bmatrix} | \mathbf{X}_{t-1} = \mathbf{x}_{t-1} \sim \mathcal{N}_{d+p} \left(\begin{bmatrix} A^k \\ E^k \end{bmatrix} \mathbf{x}_{n-1} + \begin{bmatrix} \boldsymbol{\alpha}^k \\ \boldsymbol{\epsilon}^k \end{bmatrix}, \begin{bmatrix} Q^k & U^k \\ (U^k)^t & S^k \end{bmatrix} \right)$$

with parameters $A^k, E^k, \boldsymbol{\alpha}^k, \boldsymbol{\epsilon}^k Q^k, S^k$ and U^k.

To learn the parameters, let's rewrite the distribution as:

$$\begin{bmatrix} \mathbf{X}_t^k \\ \mathbf{Y}_t^k \end{bmatrix} | \mathbf{X}_{t-1} = \mathbf{x}_{t-1} \sim \mathcal{N}_{d+p} \left(\begin{bmatrix} \boldsymbol{\alpha}^k & A^k \\ \boldsymbol{\epsilon}^k & E^k \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{x}_{n-1} \end{bmatrix}, \begin{bmatrix} Q^k & U^k \\ (U^k)^t & S^k \end{bmatrix} \right)$$

In this case, the response variable data consists of the series $\left\{ \begin{bmatrix} \mathbf{x}_t^k \\ \mathbf{y}_t^k \end{bmatrix} \right\} = \begin{bmatrix} \mathbf{x}_1^k \\ \mathbf{y}_1^k \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{x}_{n_k}^k \\ \mathbf{y}_{n_k}^k \end{bmatrix}$ and the regressor data of the previous load observation $\{\mathbf{p}_t^k\} = \mathbf{p}_1^k, \ldots, \mathbf{p}_{n_k}^k$. We must also notice that, like for the transition distribution of the Kalman Filters, also here we must skip the first observation.

To start the recursion, the support structures H_0^k, J_0^k, K_0^k and γ_0^k are initialized to zero-matrices of respectively dimensions $d \times (d+p), d \times d, (d+p) \times (d+p)$ and 1. Furthermore, the value of $\lambda > 0$ is chosen.

Then, for $j = 1, \ldots, n_k$, the support structures are updated

- $H_j^k = \begin{bmatrix} 1 \\ \mathbf{p}_j^k \end{bmatrix} \begin{bmatrix} (\mathbf{x}_j^k)^t & (\mathbf{y}_j^k)^t \end{bmatrix} + \lambda H_{j-1}^k$
- $J_j^k = \begin{bmatrix} 1 \\ \mathbf{p}_j^k \end{bmatrix} \begin{bmatrix} 1 & (\mathbf{p}_j^k)^t \end{bmatrix} + \lambda J_{j-1}^k$
- $K_j^k = \begin{bmatrix} \mathbf{x}_j^k \\ \mathbf{y}_j^k \end{bmatrix} \begin{bmatrix} (\mathbf{x}_j^k)^t & (\mathbf{y}_j^k)^t \end{bmatrix} + \lambda K_{j-1}^k$
- $\gamma_j^k = 1 + \lambda \gamma_{j-1}^k$

and the parameters become

- $\begin{bmatrix} \hat{\boldsymbol{\alpha}}_j^k & \hat{A}_j^k \\ \hat{\boldsymbol{\epsilon}}_j^k & \hat{E}_j^k \end{bmatrix} = (H_j^k)^t (J_j^k)^{-1}$
- $\begin{bmatrix} \hat{Q}_j^k & \hat{U}_j^k \\ (\hat{U}_j^k)^t & \hat{S}_j^k \end{bmatrix} = \frac{1}{\gamma_j^k} \left(K_j^k - (H_j^k)^t \begin{bmatrix} \hat{\boldsymbol{\alpha}}_j^k & \hat{A}_j^k \\ \hat{\boldsymbol{\epsilon}}_j^k & \hat{E}_j^k \end{bmatrix}^t \right)$

A.3 Evaluation metrics for subregional independent models

In this subsection, we show the tables corresponding to the evaluation metrics for independent models. This means that the training and forecasting steps are applied to each component separately.

Global temperatures

The first table shows the scores when the observed variable contains the couple formed by the dry bulb and dew point global temperatures.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
	MAPE								
MAPLF	12.5	9.6	9.4	11.2	10.4	14.4	8.6	13.3	11.2
KF	11.7	8.5	9.9	9.4	9.9	13.4	7.3	11.9	10.2
Inv. SSM	9.9	6.6	8.5	6.8	7.5	9.6	5.6	9.2	8.0
VAR	9.7	7.2	7.1	7.5	7.0	9.6	6.4	10.1	8.1
	RMSE $\cdot 10^2$								
MAPLF	7.05	6.15	6.44	6.44	5.95	6.64	5.77	6.36	6.35
KF	6.97	5.41	7.26	5.65	5.60	6.27	4.91	5.47	5.94
Inv. SSM	6.05	4.36	6.22	4.24	4.31	4.64	3.97	4.43	4.78
VAR	5.31	4.35	4.62	4.32	3.86	4.29	4.20	4.61	4.45
	Pinball loss $\cdot 10^2$								
MAPLF	2.56	2.12	2.32	2.32	2.05	2.31	2.03	2.14	2.23
KF	2.43	1.83	2.48	1.93	1.88	2.10	1.70	1.82	2.02
Inv. SSM	2.07	1.44	2.10	1.42	1.45	1.54	1.33	1.45	1.60
VAR	1.90	1.51	1.64	1.51	1.32	1.46	1.47	1.55	1.54
	LogScore								
MAPLF	-0.73	-0.66	-0.87	-0.63	-0.82	-0.60	-0.84	-0.68	-0.73
KF	-1.38	-1.59	-1.20	-1.52	-1.56	-1.40	-1.72	-1.56	-1.49
Inv. SSM	-1.58	-1.87	-1.48	-1.96	-1.97	-1.86	-2.02	-1.81	-1.82
VAR	-1.56	-1.71	-1.76	-1.71	-1.95	-1.77	-1.79	-1.71	-1.75

Table A.1: Scores, Subregional independent models with global temperatures

Subregional temperatures

The second table shows the scores when the observed variable for each regional model contains the couple formed by the dry bulb and dew point regional temperatures. Furthermore, the mean column also shows the difference from the previous table.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
	MAPE								
MAPLF	12.6	9.6	9.4	11.1	10.4	14.5	8.6	13.5	$11.2_{(+0.3\%)}$
KF	11.2	8.6	9.9	9.5	10.2	13.8	7.4	12.2	$10.4_{(+1.1\%)}$
Inv. SSM	9.5	6.7	8.7	6.8	7.7	10.0	5.8	9.4	$8.1_{(+1.3\%)}$
VAR	9.7	7.2	7.1	7.5	7.0	9.6	6.4	10.1	8.1
	RMSE $\cdot 10^2$								
MAPLF	7.11	6.13	6.37	6.41	5.94	6.63	5.74	6.34	$6.33_{(-2.6\%)}$
KF	6.59	5.42	7.23	5.69	5.70	6.36	4.99	5.54	$5.94_{(-0.2\%)}$
Inv. SSM	5.73	4.37	6.32	4.22	4.41	4.75	4.04	4.47	$4.79_{(+0.2\%)}$
VAR	5.31	4.35	4.62	4.32	3.86	4.29	4.20	4.61	4.45
	Pinball loss $\cdot 10^2$								
MAPLF	2.58	2.12	2.30	2.31	2.06	2.31	2.02	2.15	$2.23_{(0\%)}$
KF	2.31	1.85	2.47	1.94	1.94	2.15	1.73	1.87	$2.03_{(+0.5\%)}$
Inv. SSM	1.98	1.45	2.12	1.42	1.49	1.59	1.36	1.47	$1.61_{(+0.6\%)}$
VAR	1.90	1.51	1.64	1.51	1.32	1.46	1.47	1.55	1.54
	LogScore								
MAPLF	−0.72	−0.66	−0.88	−0.63	−0.82	−0.60	−0.86	−0.69	$-0.73_{(0)}$
KF	−1.46	−1.58	−1.17	−1.50	−1.52	−1.37	−1.70	−1.53	$-1.48_{(+0.01)}$
Inv. SSM	−1.65	−1.86	−1.44	−1.95	−1.93	−1.83	−1.99	−1.82	$-1.81_{(+0.01)}$
VAR	−1.56	−1.71	−1.76	−1.71	−1.95	−1.77	−1.79	−1.71	−1.75

Table A.2: Scores, Subregional independent models with subregional temperatures

Cubic temperatures

The third table shows the scores when the observed variable for each regional model contains the dry bulb and dew point global temperatures, and the respective quadratic and cubic terms. In addition, also the winter dummy variable is added to the model.

	ME	NH	VT	CT	RI	SEM	WCM	NEM	Mean
	MAPE								
MAPLF	9.1	5.2	6.0	6.1	5.3	7.2	4.5	7.3	6.3
KF	9.3	5.4	6.5	6.0	5.4	7.0	4.6	7.7	6.5
Inv. SSM	8.1	4.5	6.0	4.8	4.5	5.7	3.7	6.5	5.5
VAR	9.7	7.2	7.1	7.5	7.0	9.6	6.4	10.1	8.1
	RMSE $\cdot 10^2$								
MAPLF	4.72	3.10	3.95	3.32	2.95	3.04	2.78	2.98	3.35
KF	5.19	3.18	4.35	3.42	2.96	3.04	2.89	3.01	3.50
Inv. SSM	4.81	2.87	4.10	3.01	2.62	2.67	2.54	2.64	3.16
VAR	5.31	4.35	4.62	4.32	3.86	4.29	4.20	4.61	4.45
	Pinball loss $\cdot 10^2$								
MAPLF	1.75	1.06	1.42	1.21	1.01	1.08	0.99	1.08	1.20
KF	1.86	1.09	1.57	1.21	1.03	1.07	1.02	1.11	1.25
Inv. SSM	1.67	0.93	1.45	1.01	0.87	0.90	0.84	0.95	1.08
VAR	1.90	1.51	1.64	1.51	1.32	1.46	1.47	1.55	1.54
	LogScore								
MAPLF	−1.21	−1.60	−1.51	−1.57	−1.88	−1.83	−1.87	−1.76	−1.65
KF	−1.17	−1.64	−1.37	−1.58	−1.95	−1.90	−1.86	−1.81	−1.66
Inv. SSM	−1.32	−1.67	−1.52	−1.72	−2.07	−2.03	−1.97	−1.88	−1.77
VAR	−1.56	−1.71	−1.76	−1.71	−1.95	−1.77	−1.79	−1.71	−1.75

Table A.3: Scores, Subregional independent models with cubic global temperatures

Bibliography

Alfares, H. K. and Nazeeruddin, M. (2002). Electric load forecasting: literature survey and classification of methods. *International journal of systems science*, 33(1):23–34.

Álvarez, V., Mazuelas, S., and Lozano, J. A. (2021). Probabilistic load forecasting based on adaptive online learning. *IEEE Transactions on Power Systems*, 36(4):3668–3680.

Bjerregård, M. B., Møller, J. K., and Madsen, H. (2021). An introduction to multivariate probabilistic forecast evaluation. *Energy and AI*, 4:100058.

Bracale, A., Caramia, P., De Falco, P., and Hong, T. (2019). Multivariate quantile regression for short-term probabilistic load forecasting. *IEEE Transactions on Power Systems*, 35(1):628–638.

Bracale, A., Caramia, P., De Falco, P., and Hong, T. (2020). A multivariate approach to probabilistic industrial load forecasting. *Electric Power Systems Research*, 187:106430.

Chen, B. and Wang, Y. (2021). Short-term electric load forecasting of integrated energy system considering nonlinear synergy between different loads. *IEEE Access*, 9:43562–43573.

Fahad, M. U. and Arbab, N. (2014). Factor affecting short term load forecasting. *Journal of Clean Energy Technologies*, 2(4):305–309.

Fahrmeir, L. and Tutz, G. (2001). State space and hidden markov models. *Multivariate Statistical Modelling Based on Generalized Linear Models*, pages 331–383.

Fan, S., Methaprayoon, K., and Lee, W.-J. (2009). Multiregion load forecasting for system with large geographical area. *IEEE Transactions on Industry Applications*, 45(4):1452–1459.

Fiot, J.-B. and Dinuzzo, F. (2018). Electricity demand forecasting by multi-task learning. *IEEE Transactions on Smart Grid*, 9(2):544–551.

Guo, Y., Li, Y., Qiao, X., Zhang, Z., Zhou, W., Mei, Y., Lin, J., Zhou, Y., and Nakanishi, Y. (2022). Bilstm multi-task learning based combined load forecasting considering the loads coupling relationship for multi-energy system. *IEEE Transactions on Smart Grid*, pages 1–1.

Hong, T. and Fan, S. (2016). Probabilistic electric load forecasting: A tutorial review. *International Journal of Forecasting*, 32(3):914–938.

Hong, T., Pinson, P., and Fan, S. (2014). Global energy forecasting competition 2012.

Hong, T., Xie, J., and Black, J. (2019). Global energy forecasting competition 2017: Hierarchical probabilistic load forecasting. *International Journal of Forecasting*, 35(4):1389–1399.

Hu, Y., Xia, X., Fang, J., Ding, Y., Jiang, W., and Zhang, N. (2018). A multivariate regression load forecasting algorithm based on variable accuracy feedback. *Energy Procedia*, 152:1152–1157.

Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688.

Jacob, M., Neves, C., and Vukadinović Greetham, D. (2020). *Forecasting and assessing risk of individual electricity peaks*. Springer Nature.

Jeong, D., Park, C., and Ko, Y. M. (2021). Short-term electric load forecasting for buildings using logistic mixture vector autoregressive model with curve registration. *Applied Energy*, 282:116249.

Kim, Y. and Kim, S. (2021). Electricity load and internet traffic forecasting using vector autoregressive models. *Mathematics*, 9(18).

Lemos-Vinasco, J., Bacher, P., and Møller, J. K. (2021). Probabilistic load forecasting considering temporal correlation: Online models for the prediction of households' electrical load. *Applied Energy*, 303:117594.

Li, P., Zhang, B., Weng, Y., and Rajagopal, R. (2015). A sparse linear model and significance test for individual consumption prediction.

Mirowski, P., Chen, S., Ho, T. K., and Yu, C.-N. (2014). Demand forecasting in smart grids. *Bell Labs technical journal*, 18(4):135–158.

Niu, D., Yu, M., Sun, L., Gao, T., and Wang, K. (2022). Short-term multi-energy load forecasting for integrated energy systems based on cnn-bigru optimized by attention mechanism. *Applied Energy*, 313:118801.

Obst, D., De Vilmarest, J., and Goude, Y. (2021). Adaptive methods for short-term electricity load forecasting during covid-19 lockdown in france. *IEEE transactions on power systems*, 36(5):4754–4763.

Petersen, K. B., Pedersen, M. S., et al. (2012). The matrix cookbook. *Technical University of Denmark*, 7(15):510.

Salleh, N. S. M., Suliman, A., and Jørgensen, B. N. (2020). A systematic literature review of machine learning methods for short-term electricity forecasting. In *2020 8th International conference on information technology and multimedia (ICIMU)*, pages 409–414. IEEE.

Seifi, H. and Sepasian, M. (2011). *Electric Power System Planning: Issues, Algorithms and Solutions*. Power Systems. Springer Berlin Heidelberg.

Sharma, S., Majumdar, A., Elvira, V., and Chouzenoux, E. (2020). Blind kalman filtering for short-term load forecasting. *IEEE Transactions on Power Systems*, 35(6):4916–4919.

Upadhaya, D., Thakur, R., and Singh, N. K. (2019). A systematic review on the methods of short term load forecasting. In *2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC)*, pages 6–11. IEEE.

Vrablecová, P., Ezzeddine, A. B., Rozinajová, V., Šárik, S., and Sangaiah, A. K. (2018). Smart grid load forecasting using online support vector regression. *Computers & Electrical Engineering*, 65:102–117.

Wang, Y., Chen, Q., Hong, T., and Kang, C. (2019). Review of smart meter data analytics: Applications, methodologies, and challenges. *IEEE Transactions on Smart Grid*, 10(3):3125–3148.

Wood, A., Wollenberg, B., and Sheblé, G. (2013). *Power Generation, Operation, and Control*. Wiley.

Zhu, W., Yu, Y., Yang, M., and Zhao, Y. (2021). Review on probabilistic short-term power forecast. In *2021 IEEE/IAS Industrial and Commercial Power System Asia (ICPS Asia)*, pages 880–884.