University of Camerino

School of Advanced Studies

Doctor of Philosophy in
Computer Science & Mathematics
XXXVII Cycle

Forecasting Demand and Supply Curves in Day-Ahead Electricity Markets

A Machine Learning Approach

Supervisor:
Prof. Carlo Lucheroni

PhD Candidate:
Nabangshu Sinha

Academic Year 2022–2025

Declaration

I herewith declare that I have produced this paper under the supervision of Prof. Carlo Lucheroni at the University of Camerino, without the prohibited assistance of third parties and without making use of aids, other than those specified. Notions taken over directly or indirectly from other sources have been identified as such. This paper has not previously been presented in an identical or similar form to any other Italian or foreign examination board.

Abstract

The electricity day-ahead market (DAM) plays an important role in ensuring efficient electrical energy distribution and price setting across mostly deregulated power systems. A key feature of this market is the set of supply and demand curves that are constructed from aggregated bid and offer data. These curves are challenging to forecast due to the heterogenity in their construction across days and hours, their inherently monotonic structure. Additionally in quite a few markets, there could be a constraint for forecasting on multi-day horizons. This thesis contributes a compilation of forecasting models and techniques designed to address these challenges, with a focus on robust and interpretable forecasting of these curves.

The first contribution is the introduction of a modular framework called the BME-model, which tokenizes market curves into three structurally significant points, B, M, and E, and identifies the significant segment of the curve, for forecasting. We propose two variants, a "pure" variant and a hybrid "combined" variant. These models reconstruct the full curve from individually forecasted tokens and offer improvements over the benchmark naive models.

Building upon this, a non-linear extension of the BME-model, using a monotonic autoencoder, is proposed. This approach replaces linear interpolation with step interpolation and uses the encoding of the significant segment of the curve to capture the non-linear structural patterns. An Enhanced Echo State Network (EESN) is also proposed, tailored specifically to forecast both BME tokens and autoencoder encodings with better performance than the previous linear BME-model.

To assess model performance in this heterogeneous curve setting, a metric, Heterogeneous Curve Mean Absolute Error (HC-MAE), is proposed. This loss function allows consistent and fair comparison of curves with heterogenous structures.

Additionally, an I-Spline-based model is also developed to functionally parameterize the curves while preserving monotonicity. A combination framework is also proposed to combine curve forecasts from different curve forecasting models. The I-Spline-based model and the combination framework were used for a seven day ahead forecasting.

Overall, this thesis shows that explicitly modeling and accounting for curve heterogeneity, enforcing monotonicity, and using combinations various models significantly improves the accuracy and robustness of day-ahead market curve forecasts in one-day ahead and seven-days ahead forecasting. These contributions can contribute to the literature for more reliable electricity DAM forecasting.

Contents

List of Publications

- Sinha, Nabangshu, Carlo Lucheroni, and Carlo Mari. "BME Model: Forecasting Electricity Supply and Demand Curves Using Disentangled Prices and Volumes." Applied Stochastic Models in Business and Industry 41, no. 4 (2025): e70032.
- Sinha, Nabangshu, and Carlo Lucheroni. "Demand and supply curve forecasting using a monotonic autoencoder for short-term day-ahead electricity market bid curves." Applied Energy 397 (2025): 126262.
- Mari, Carlo, Carlo Lucheroni, Nabangshu Sinha, and Emiliano Mari. "Power System Portfolio Selection and CO2 Emission Management Under Uncertainty Driven by a DNN-Based Stochastic Model." Mathematics 13, no. 9 (2025): 1477.

List of Figures

List of Tables

Chapter 1

Introduction

A day-ahead electricity market (DAM) is a special type of market in which a market operator (MO), in collaboration with a system operator (SO), manages a daily auction in which electric energy is traded for the individual 24 hours of the day, in a transparent and market-based way, and electricity prices are set by marginal pricing. In a given day, after the auction is closed, 24 prices for energy unitary volume (in currency/MegaWatthour, like eur/MWh) are set for the 24 hours of the next day. Only a subset of bids in the auction are then marked as accepted, and are dispatched (allocated) for coming delivery. The pool of the participants to this market usually includes electricity generators, distributors, and large consumers. DAMs are organized in this way because the collection of trading information (at least one day in advance of electricity delivery) by part of the MO allows the SO to smoothly handle potentially disruptive electricity production/consumption unbalances. At the moment of clearing the auction, received demand and supply bids are aggregated into 'aggregated demand and supply curves' or, more in short, 'demand and supply curves', in the number of 24 pairs of curves, one pair per hour. In some markets, these curves are disclosed to the public ex-post for free, including information about which bids are accepted. This is a lot of information. The autoregressive forecasting of these data structures is the subject of this thesis.

The reasons why curve forecasting is important come mainly from the two following considerations.

Better modeling and forecasting DAM price dynamics and its volatility comes first. After aggregation, the MO uses the obtained curves for marginal pricing, which is the DAM's main financial aim. Marginal pricing is based

on computing the intersection of the supply curve with the corresponding demand curve to give a market clearing price/volume pair. The MO takes care of this, after keeping into account technical constraints pointed out by the SO, in case. The 24 pairs' intersections become the so-called hourly 'realized price/volume pairs'. These prices get published, and become the official, reference prices for all DAM energy trades of the next day. Noticeably, in Europe, most of the national DAMs are integrated into one continent-wide market, hence foreign entities can participate to each national market, making the fixing procedure even more complicated. Moreover, some of the national markets are zonal, meaning that they can be though as segmented into sub-markets (yet still interacting with the rest of the European system). Hence, gathering information on these curves, better understanding them, and possibly guessing their behavior for the next days, for the local and for the foreign markets, should constitute an all-important activity of market participants, regulators included. Indeed, this activity should be not less important than the activity of forecasting the 24 realized price/volume pairs themselves, a kind of forecast which is for itself nowadays routinely carried forward each day by DAM traders and energy finance researchers [1, 2]. In the end, if the curves themselves can be accurately forecast, even the volatility of price dynamics can be better understood and managed.

Second, besides matters directly related to price dynamics, there are in addition more technical advantages related to full curve modeling and forecasting, especially for the short term of one day, which is the horizon which will be studied in this paper. For example, a good curve forecasting ability allows the SO to better predict the behavior of specific generation agents on one side, and large demand and distribution (wholesale sellers) agents on the other side, and the reciprocal interaction between these market movers. In addition, the SO can exploit good demand and supply curve modeling and forecasting for better pinpointing the relationship between financial and technical aspects of its activity. This includes better handling of transmission congestions by means of anticipating effects of financial supply-demand imbalances from market operations on power flows and grid stability. It includes also better scheduling and dispatching generation resources, especially renewable resources, based on both financial and environmental data [3]. As to the MO itself, such knowledge can also help it in its regulatory capacity of spotting odd bids intended to manipulate the market. Generators and distributors themselves, as market agents, can use this ability in the short term to predict other users' behavior and improve their optimal bidding

strategies. In fact, one of the main prerequisites for finding optimal bidding strategies is knowing the competitors' bids [4, 5]. For example the aggregated supply curve can indeed be considered by generators as surrogate data for competitor generators' behavior [6]. As to longer term horizons forecasting, which can be considered a possible development of short term forecasting, it can also help SOs to manage the integration of renewable energy sources by taking into account the impact of their variable nature on agents, and in planning for an effective and economical deployment of energy storage or demand response systems [7, 8, 9].

Despite all these possible advantages, just a small number of papers in the literature is devoted to forecasting this kind of curves. A possible reason for that is the technical difficulty of modeling and forecasting such curves because of their peculiar data structure. For example, these curves, seen as prices as a function of volumes, are intrinsically monotonic, as it is demanded by microeconomic theory and by market regulations (this aspect will be discussed in detail in the next Section). In addition, they have an irregularly distributed support in volume that changes in distribution each day due to the bidding process. Forecasting such structures is difficult, and in the specific monotonicity itself is difficult to be preserved when forecasting them, whereas a lack of it can result in incoherent forecasts.

1.1 The Curves: Interpolated Sequences

An exemplary plot of a supply and demand curves pair from the NORD zone of the Italian IPEX DAM for hour 9 of January 5, 2018 is shown in Fig.(1.1). The IPEX DAM is managed by the Gestore dei Mercati Energetici (GME) [10], an Italian state-owned company. This pair of curves is taken from the dataset used in this study, which consists of 730 IPEX market days from January 1, 2018, to December 31, 2019, collected from Ref. [10] (more on that in Chapter 6). For clarity of exposition, it is better to divide the discussion of these curves into the three following parts: what supply curves represent (Subs.1.1.1), what demand curves represent (Subs.1.1.2), and what is the role of price/volume curves in DAM pricing (Subs.1.1.3). This discussion is also intended to make even more explicit the reasons why curves' modeling and forecasting are so much important for DAM business intelligence.

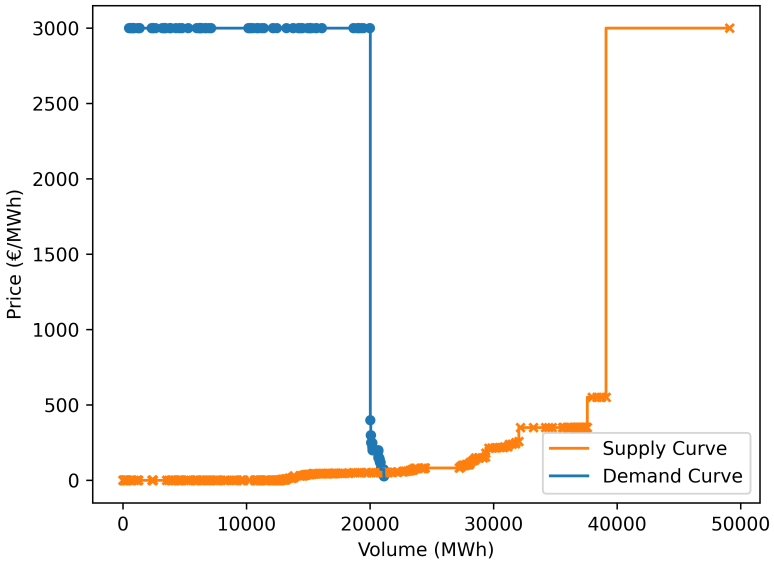

Figure 1.1: Aggregated demand and supply curves for hour 9 of January 5, 2018 from the NORD zone of the Italian IPEX market, seen as prices (on the vertical axis) in function of volumes (on the horizontal axis). The blue dots show the demand bid data (contractual conditions for buying) and the orange crosses show the supply bids data (conditions for selling). The continuous lines are not in the data, but cannot be considered just as interpolations (see text). Noticeably, there are almost no supply bids for aggregated volumes larger than 39,000 MWh, and the line in that segment doesn't represent existing data. Notice also that there are no demand bids at all for volumes larger than about 23,000 MWh.

1.1.1 The Supply Curve as a Sequence of Points

This first Subsection is related to the supply curve $P^S(V^S)$, represented in orange and with crosses in Fig.(1.1). The MO builds up this plot in the following way, based on collected supply bid price/volume data I^S pairs of form (p_j^S, v_j^S), $j = 1, \dots I^S$, where for example a bid pair could be ($p_j^S = 100$ eur/MWh, $v_j^S = 30$ MWh). Starting from the lowest value of p_j^S, usually zero (but not necessarily so), the MO orders the pairs

$$(p_j^S, v_j^S)$$

in an increasing way with respect to p_j^S. If there are more than one pairs corresponding to the same value of p_j^S, the MO sub-orders them in an arbitrary order. In this way the MO obtains a new sequence of pairs, again of length I^S, but now strictly ordered in a new index i. Then it computes the strictly increasing sequence of 'aggregated volumes' $V_i^S = \sum_{k \leq i} v_k^S$ (notice the $k \leq i$ symbol) to which a non-decreasing monotonic sequence $P_i^S = p_i^S$ corresponds (hence not necessarily strictly monotonic). In this way a sequence

$$(P_i^S, V_i^S)$$

of price/'aggregated volume' pairs is obtained, which can be plotted as $P_i^S(V_i^S)$ and can have flat parts, since more than one value p_j^S can correspond to P_i^S. This is the orange sequence of crossed points shown in Fig.(1.1). Overall, this procedure results in a sequence of points monotonously sloping upward in the direction of increasing volumes by construction, with 'aggregated volume' coordinates on the abscissa, irregularly distributed along the horizontal axis, changing daily in their number I^S and in their positions, corresponding to prices P_i^S on the ordinate which change as much.

A lot of information about suppliers' behavior is contained in this sequence. As first, a supplier signals with its bid that it won't sell a specific volume v_j^S at a price less than $p_j^S = P_j^S$. However, there could be in the market other suppliers more than happy to sell a smaller volume for that same price P_j^S, and other suppliers happy to sell a suitably smaller volume for smaller prices. When prices are formed, this way of constructing the curve will take care of that interest, as it will be seen in short. In addition, the position of a data point along the curve is revealing a lot of information about market consistency and structure. For example, the zero-price left flat part of the supply curve usually corresponds the the bids of solar or

other renewables-based generators which in this way declare themselves as available to take the risk of delivering for free the energy which they cannot store (but in the marginal pricing scheme they will actually not produce for free, as it will be seen in short). This means that, daily, the larger is the available quantity of renewables, the more rightward it is placed the first nonzero price. This threshold is then connected to a special point on the curve, which one could call $\mathbf{A}_{supply}$. For serving larger aggregated volumes, the cost of production increases from zero in function of the type of production technology required, and suppliers start on their part requiring minimum remuneration prices that then increase in steps: this stepped up structure is called 'power stack'. The most expensive technology available in the market corresponds to the highest power stack price. This creates a second special point, which one could call $\mathbf{B}_{supply}$. These two special points and the overall power stack pattern (the steps) can be easily detected by eye in the supply curve of Fig.(1.1). In this curve, it is also worth considering the last data point on the r.h.s., the point at the maximum price allowed by the market rules, in this case 3000 euros per MWh. That point is due to typical selling agent strategic behavior. An agent places a bid there at the risk of almost never getting accepted for this bid, but also in order to capture and exploit tight market conditions, in the case when buyers are forced to accept any price.

1.1.2 The Demand Curve as a Sequence of Points

This second Subsection is related to the demand curve $P_i^D(V_i^D)$. A demand curve plot is obtained like the supply curve plot, but in reverse, by first organizing the prices of I^D demand bids (p_j^D, v_j^D) in descending order on the vertical axis, the first price being the highest. In most markets, this first, highest price is set by regulations for defense of the consumers (3000 euros per MWh in the IPEX market). As price values decrease, corresponding aggregated volumes on the horizontal axis are associated to prices $P_i^D = p_i^D$, and are obtained as $V_i^D = \sum_{k \geq i} v_k^D$ (notice the $\geq$ symbol), in reverse analogy to supply volumes. Overall, this procedure results in a sequence (P_i^D, V_i^D) monotonically rightward decreasing down to the minimum price allowed by the market rules, with aggregated volume coordinates irregularly distributed along the horizontal axis, and with a daily varying number I^D of points. This sequence can be plotted as $P_i^S(V_i^S)$ and can have flat parts. This is the blue sequence of dots shown in Fig.(1.1).

A lot of information about demand behavior is contained in this structure. With a bid (p_j^D, v_j^D), a demand-side agent signals the maximum price at which it would buy a specific quantity. However, there could be in the market other demand more than happy to buy a larger volume for that same price $p_j^D = P_j^D$, and there could be other demand happy to buy a suitably larger volume for larger prices. Position along the curve is revealing. Corresponding to the highest price, there could be in the market a demand minimum threshold due to large consumers like production plants, or uninterruptible businesses or public services, that cannot do without electricity, and accept any (then maximum) prices. As working activity decreases, like it happens in weekends (hence this piece of information is all-important), the position of this threshold on the curve moves leftward, and a price appears at a lower value. This is thus another special point, which one could call $\mathbf{A}_{demand}$. A further special point is the threshold point corresponding to all possible demand generated in the market, beyond which demand falls to zero. One can call it $\mathbf{B}_{demand}$. These two special points can be easily detected by eye in the demand curve of Fig.(1.1), where represented uninterruptibles are on the left. In the Figure, the demand curve ends at a volume of about 23,000 MWh (23 GWh). On the r.h.s. of this value there is no further requested aggregated volume in the dataset - which means that the whole electricity system of the NORD zone of IPEX, on this specific day, didn't ask for more electricity than that. Noticeably, the curve segment between the two special points is very steep and the corresponding aggregate volume range very thin, about one GWh in the Figure, and quickly and weekly moving backward and forward as time goes by. This is a very difficult target to track or forecast. However, this segment represents core information for the DAM price fixing procedure, and for explaining price dynamics and price volatility.

1.1.3 DAM Pricing and the Interpolation of the Points

This third Subsection is related to the fact that data come in points, but are represented like curves. Such curves could be thought as some form of interpolation, not being actually present in the original data. Hence the word 'aggregated curve' could be a misnomer, which is not. The need of formally adding such interpolation structure to the sequence structures comes straight from marginal pricing. Based on marginal pricing, the MO finds the clearing volume V_* as the solution of the equation $P_*^S(V_*) = P_*^D(V_*) = P_*(V_*)$. At this condition, any supply below V_* will be paid not less than $P_*^S(V_*)$ (re-

newables included), and demand will be satisfied at not more than $P_*^D(V_*)$ (uninterruptibles included). Supply and demand bids outside this range are rejected, which just means discarding participants that signaled by their bids of not being interested in trading at the set price (for example, generators which are too much expensive). All other participants will be instead paid, or they will pay, P_* per unitary volume (1 MWh, in the case of the Figure), the official price. Many sellers (those which volumes v_j^S end up on the left of V_*, due to the condition $V_i^S = \sum_{k \leq i} v_k^S$) will be paid a price equal to or larger than requested, many buyers (those which volumes v_j^D end up on the left of V_*, due to the condition $V_i^D = \sum_{k \geq i} v_k^D$) will pay a price equal to or less than requested. Yet, finding the crossing point exactly at one of the bid volume points $V_i^{S/D}$ is highly unlikely, and volume values outside those already present in the data must be used. Hence, interpolation is contractually necessary, and each market has its own contractual rules for that. For example, the IPEX interpolates with steps [11, 12]. The PJM (a North American zonal-like market), in contrast, considers some supply curves to be linearly interpolated and others to be step interpolated [6]. Hence, the lines in the Figure are there to make clear and official the procedure by which the MO places the official aggregated volume in that specific position. In addition, these lines suggest at what price on that specific day the market would have priced any other possible volume. For example, price risk management teams are very interested in the positioning of the flat rightmost segment of the supply curve, because if a crossing ends up there a price spike is formed, which is perceived as an extremely favorable (for supply) or adverse (for demand) market condition. Modelers and forecasters take notice, tend to work with curves instead of sequences, and usually simply call the two functions $P^S(V_i^S)$, $i = 1, \ldots, I^S$ and $P^D(V_{i'}^D)$, $i' = 1, \ldots, I^D$, duly endowed with their interpolations, supply and demand curves $P^S(V)$ and $P^D(V)$.

The sequences in the curves change each day in support (positions of aggregated volumes and number of data), shape, positions of special points (this is especially critical in the case of the demand curve segment between its two special points, very steep and quickly moving), and consequently crossing points. As to crossing points, they mostly appear in the non-flat segments between the special points of both curves, hence the dynamics of these special segments plays a major role in price formation, day after day. We will hence use the adjective 'heterogeneous' to characterize the sets of these curves, which are not defined on equi-spaced never-changing grids. The behavior of

curves within their heterogeneous sets depends on weather forecast information, cost of fuel forecasts, scheduled and unscheduled maintenance activity, power grid conditions, conditions of external markets, social events, and weekday [1]. Not even mutual interaction between hours is to forget, since DAM traders can see different hours as statistically coupled, and strategically bid in consequence. Modeling and forecasting realized price/volume pairs (P_*, V_*) for each hour, instead of full curves, is obviously much easier, and there is a very large body of literature on that. However, the quantity of potentially interesting information contained in the full curves, even for the sole aim of improving realized price forecasting, is so large, that it can be just worthwhile to attempt full curve forecasting, and exploring new models for that.

Chapter 2

Literature Review and Problem Statement

2.1 Demand and Supply Curve Forecasting

2.1.1 Traditional forecasting methods

All the few available papers on curve forecasting invariably treat the sequences directly as curves, and can be at large categorized into two categories, depending on which support is used for modeling the curves:

1. papers that consider P as the support, hence focusing on $V(P)$,
2. papers that consider V as the support, hence focusing on $P(V)$.

As seen, Tab.2.1 lists all papers to be discussed in this Section, which in practice amount to most of the available literature devoted to DAM curves forecasting. Its 'Support' column indicates which support uses which model.

2.1.2 P as the Support

The following papers all consider curves such that prices are on the abscissa, and the curves are consequently formulated as $V(P)$. Since the P axis has fixed minimum and maximum values (those defined by related market regulations), all such curves can be seen as varying over this support within fixed limits, which is a useful feature.

Authors	Title	Model Name	Year	Support	Reference no.
Pelagatti	Supply function prediction in electricity auctions	PCA + Linear Autoregression	2012	P	[11]
Ziel and Steinert	Electricity price forecasting using sale and purchase curves: The X-Model	X-model	2016	P	[13]
Shah and Lisi	Forecasting of electricity price through a functional prediction of sale and purchase curves	FAR	2020	V	[14]
Mestre et al.	Forecasting hourly supply curves in the Italian Day-Ahead electricity market with a double-seasonal SARMAHX model	Double seasonal SARHMAX model	2020	P	[12]
Guo et al.	Forecast aggregated supply curves in power markets based on LSTM model	PCA + LSTM	2021	V	[6]
Yildirim et al.	Supply curves in electricity markets: a framework for dynamic modeling and monte carlo forecasting	MCMC modelling and forecasting	2022	P	[9]
Ciarreta et al.	Forecasting electricity prices using bid data	Piecewise functional modelling + forecasting	2023	V	[15]
Tang et al.	Forecasting individual bids in real electricity markets through machine learning framework	Dimensionality reduction + Transformer	2024	V	[16]

Table 2.1: List of the foundational works discussed in the literature review Section. Notice the column 'Support', which indicates the support (P or V) chosen by the modeler.

The first work in market curve forecasting was a paper by Pelagatti [11], published in 2012, which considered P as the support. The Author reduced the number of P points of the curves by collapsing their set into those values that define their quantile boundaries. This allowed to define the curves over a uniform grid of (selected quantile) prices. Principal Component Analysis (PCA) was used to break down the re-defined curves into independent components, of which the weights could be then forecast. By forecasting just these components the Author was able to reconstruct the full forecasted curve. This work forecasted only supply curves. In 2016 Ziel et al. [13] tackled a different problem, that of forecasting hourly prices per MWh by means of predicting the curve crossing points. In their procedure, among others, they took on the problem of modeling irregularly spaced prices and related curves. They defined a grid of price bins common to all curves. The curves

were reorganized using this price grid, and the V values corresponding to the bins were forecasted using independent LASSO (Least Absolute Shrinkage and Selection Operator) autoregressions, that is, linear autoregressions where a large number of uninteresting time lags are automatically discarded. This model was called the X-Model. Three further papers applied this model to hourly price forecasting. Ref.[17] and [18] were a probabilistic extension of the original model. Ref.[19] used some of the mechanisms of the X-model, yet limited to only a segment of entire curve. All this makes apparent that the X-model approach is hence important for curve forecasting too, although not born to be aimed at it. Based on it, curve forecasting can be made by attaching the forecasted V value to each price of the price binning set. Based on the probability empirical distribution of the bid prices, Li et al. [20] developed a new distance metrics which they utilized to cluster curves, and then, for a given day d and hour h, picked up cluster representatives, and forecasted them. This work considered only supply curves. Mestre et al. [12] modeled the supply curve using a deterministic-functional model that took seasonality into account. In this case modeling was based on representing the steplike structure of the curve as a series of sigmoid functions. Notice that this is a limiting assumption about the supply curves' shape, and this approach might not apply to other electricity markets such as the European Power Exchange (EPEX) or the North-American Midcontinent Independent System Operator (MISO) energy market, because these markets consider the curves to be linearly interpolated, either partly (MISO) or fully (EPEX). Linearly interpolated curves may not have the step like structure that is assumed in this work.

As said, all the techniques reported in Refs. [9, 6, 11, 12] were specifically designed for supply curves, their methods being developed using certain transformations that might not apply to demand curves. Except for the model by Mestre (which modeled the supply curve only), no guarantee of monotonicity was taken into consideration.

2.1.3 V as the Support

The following papers consider curves such that volumes are on the abscissa, as P(V).

Shah and Lisi [14] used a stochastic-functional single-lag linear autoregression (FAR) to model and forecast the prices of both demand and supply curves, using V as the support defined on a grid. In this case, curves were

represented as abstract stochastic functionals, of which the market curves were sampled representatives, and the expected value $\pi(V)$ corresponded to a forecast. A functional kernel which could be extracted from data would allow for statistically connecting one day to the next (because of the only available lag), as a forecast. The Authors then extended this approach to arrange the use of an extra lag besides the first one (within the same stochastic one-lag theory) to improve their forecasts. The approach was applied to both supply and demand curves. Ciarreta et al. [15] proposed a gridded deterministic-functional approach called piecewise model which used a piecewise function for the curves. In this case monotonicity was implemented naturally, and not as a constraint. This model was developed with the Spanish market OMIE in mind. This work considered both demand and supply curves. Actually, in the OMIE, prices gradually change from P_{max} to other prices, and there is no abrupt drop or rise in the curves. Hence, the Authors' piecewise model can work without the necessity of any forecast of transition volumes. This, however, makes this approach not applicable to the Italian IPEX, where an abrupt transition from P_{max} to other values (or the way around) does occur. Guo et al. [6] reduced the dimensionality of the curves using a non-linear mapping and PCA. Then, they employed an LSTM neural network to forecast the principal components. This technique is quite akin to the one used in Pelagatti's approach of Ref.[11], although there are some notable differences between the two. This work considered only the supply curves. Tang et al. [16] used a combination of dimensionality reduction methods, namely PCA, non-negative matrix factorization (NMF), dictionary learning, and just downsampling of the curve, to reduce the curve representation, and then used a Transformer-based forecasting model. This work considered only the supply curves. No guarantee of monotonicity was taken into consideration except for the model by Ciarreta at al..

2.1.4 Common Features of the Models in the Literature

The models discussed so far have some elements in common, namely:

1. All of them use some kind of approximation to deal with the curve heterogeneity problem, looking at the data sets as curves and not as sequences. These approximations introduce some amount of error in curve reconstruction.

2. All of these models more or less follow the following methodology: transform the whole curves (large flat parts included) by remapping them on a common support, use a suitable dimensionality reduction method that may be direct (PCA, NMF) or indirect (stochastic functional parameterization like FAR, deterministic functional piecewise linear modeling) which takes into account the whole curve, use a suitable forecasting model to forecast the individual elements of the vectors with reduced dimensionality - linear, or taken from machine learning (ML) like long short-term memory (LSTM) or transformer networks -, and finally reconstruct the curves using an inverse of the dimensionality reduction method. It should be noted that whereas both linear and non-linear forecasting models are used, the dimensionality reduction techniques are always linear.

3. Besides a few models, namely those in Refs.[12] and [15], none of the models enforced the condition of monotonicity in their respective forecast curves.

4. Other than Refs.[13] and [14], none of the other models consider the full span of the support, be it the P axis or the V axis. Instead, they focus on a limited segment of the curve.

5. Only Refs.[13] and [14] work with both demand and supply curves. In addition to these, other works focus mainly on forecasting supply curves, which don't include difficult, back and forth swinging, steep segments unlike demand curves.

6. All of these works except [13], [14], and [15] consider an error function, the loss, that is heavily based on the probability distribution of historical prices. This approach is maybe too much specific to the studied dataset. In fact, it cannot lend itself to situations where the bids happen to deviate from the probability distribution of historical prices, at detriment of generality. For example, after a major energy market shock in 2022, the bidding price patterns of generators changed, and the price definition intervals expanded. If only pre-2022 historical bid prices are considered, then these prices will not be considered as the important parts of the price P axis in the related calibrated forecasting procedure, in this way missing what it is the true bidding nature of the market agents after a market shock. Similar situations could be

due to market regulations changes, or some bids placed for deliberately manipulating the market.

7. All models assume that bidding on a specific hour is independent from bidding on all other hours. This uncoupling is clearly made to make the approach numerically feasible (and stable).

2.1.5 Curve forecasting challenges

1. **Demand and supply curves are monotonic by nature**: Demand curves are monotonically decreasing. This shows that consumers buy less as prices rise. On the other hand, supply curves are monotonically increasing, as producers are willing to supply more at higher prices. Any forecasting model designed to predict such curves should ideally preserve this inherent monotonicity in its outputs, as non-monotonic predictions may be considered unrealistic and invalid.

2. **Wide shifts in the V axis**: The demand and supply curves show huge shifts in the volume V axis. This behavior is particularly pronounced in demand curves. These shifts can occur because of several factors, such as changing patterns in consumption, weather conditions, or market dynamics. The resulting variability in curve positioning makes it difficult to model them, as well as accurately forecast them. Standard models often struggle to catch the shape in addition to the shifting location of the curve simultaneously.

3. **Curves are heterogenous**, meaning they vary significantly in both the number of bids and the distribution of those bids across the price P and volume V axes. This variability occurs from hour to hour and day to day, reflecting adjustments in market player behavior, system conditions, and other external factors. Such heterogeneity poses a serious challenge for forecasting models because it introduces an irregular structure that may be tough to capture consistently.

4. **There are abrupt jumps in the curves.** Both demand and supply curves often contain discrete jumps in price. For supply curves, this is usually a sharp increase to a high price level, often P_{max}. For demand curves, a similar jump occurs downward from a high price level. These abrupt jumps are difficult to model accurately, as they represent

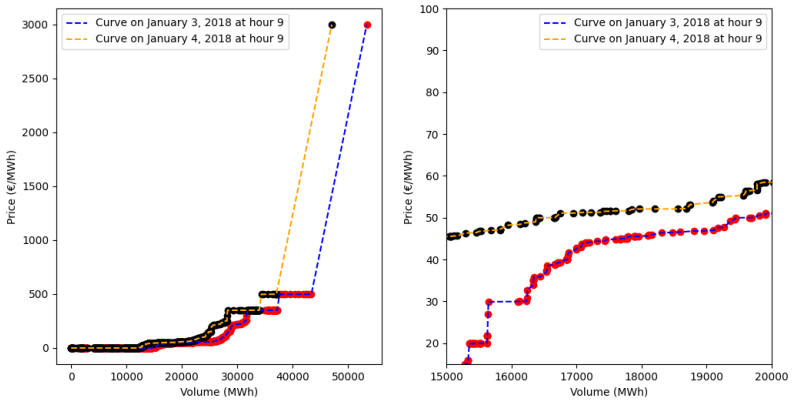

Figure 2.1: (Left) Two supply curves on January 3, 2018 and January 4, 2018 at hour 9; (Right) The supply curves zoomed between P values of 15 € and 100 €, and V values of 15000 MWh and 20000 MWh

near-discontinuities in the curve's shape that standard smooth function approximators may find difficult to capture. As noted in Ref. [6], such jumps significantly increase the complexity of curve forecasting.

2.2 Research Gaps

2.2.1 Existing gaps in the literature

Most existing works in electricity market forecasting do not explicitly enforce monotonicity in their predicted curves. Although demand and supply curves are inherently monotonic due to the market structure, most curve forecasting approaches overlook this property, resulting in forecasts that are structurally invalid. Ignoring monotonicity reduces the practical usability of forecasts and limits their interpretability and accuracy. To the best of my knowledge, monotonicity constraints have been explicitly incorporated into curve forecasting models in only a few studies, notably Refs. [11], [13], [15], and [12].

Furthermore, most existing forecasting studies focus predominantly on supply curves while neglecting demand curves. Both curves are fundamen-

tally different in their structures and underlying dynamics. Thus, the methods developed in these studies are often explicitly tailored for supply-side forecasting and may not easily transfer to demand curves. Demand curves exhibit fundamentally different structural features, such as larger and more frequent shifts along the volume (V) axis and pronounced seasonal variations. To our knowledge, comprehensive studies addressing both the forecasting of supply and demand curves simultaneously are limited to a few references such as [11], [13], and [14].

Additionally, the work in this thesis adds to the literature focusing on the Italian Power Exchange (IPEX) market. Most studies concentrate on larger markets such as EPEX in Germany or the PJM market in the United States. As a result, the unique characteristics of the IPEX market, including distinct bidding behaviors and regional dynamics, are relatively underexplored. Existing research addressing curve forecasting within the IPEX context includes Refs. [11], [14], and [12]; among these, only Ref. [14] considered both demand and supply curves, but mainly for price forecasting rather than explicitly for curve forecasting itself.

Another area that has been insufficiently explored in literature is the use of non-linear dimensionality reduction methods for curve forecasting. While linear dimensionality reduction techniques such as Principal Component Analysis (PCA) have been utilized in several studies, including Refs. [11], [14], and [6], non-linear methods like autoencoders have rarely been examined. Non-linear methods potentially offer superior performance by capturing complex structural characteristics inherent in electricity market curves, leading to richer and more accurate forecasts.

In addition, many current forecasting models take for granted that there's a sequential link between hourly curves. For instance, they often treat the curve for hour 9 as reliant on that of hour 8 before it. That being said, in day-ahead electricity markets, all 24 hourly curves are set at the same time the day before, meaning that assuming a sequential dependency could really distort how the market actually behaves. These models miss the crucial interactions that happen across several hours at the same time. As a result, these models miss out on recognizing important patterns that evolve over time, limiting their forecasting accuracy and missing opportunities to capture richer temporal structures in bidding behaviors.

Furthermore, modeling usually requires some distance measure that measures the distance between raw data and the modeled curve. In this way, modeling becomes an optimization problem. However, due to the curve het-

erogeneity problem as discussed in section 2.1.5, it is difficult to compare two curves based only on raw data, since the supports of the two curves would be different. Some works, such as [6] and [16], use a distance metric that is based on the distribution of bid prices. However, this is a metric learned from data, and may not show curve dissimilarity accurately in the case of sudden market trend changes.

Finally, little attention has been paid in the literature to the long-horizon forecasting of market curves. This situation is especially pertinent due to operational constraints imposed by markets such as IPEX, which disclose bid data with a seven-day lag. Hence, reliable methods for forecasting curves accurately about seven days ahead are required. Moreover, the strategy of combining several forecasting models for the improvement of the overall forecasting performance has also not been considered in the context of curve forecasting. In this regard, only ref. [14] has dealt with the problem of the seven-day forward forecasting of curves.

2.2.2 Contributions in this thesis

1. **Novel Tokenization Framework (BME-Model):** We introduce a modular forecasting framework (BME model) that addresses the curve heterogeneity problem by tokenizing market curves into three structurally significant points (B (standing for 'Beginning'), M (standing for 'Middle') and E (standing for 'End')), without relying on traditional functional approaches. We also extract a segment of the curve that is the most significant and informative. We forecast each of these separately and then combine the forecasted elements to reconstruct the curve.

2. **Effective Combined Model Implementation:** We also propose a combined model variant, integrating linear autoregression, persistence-based naive methods, and a separate module for forecasting a special curve segment, significantly outperforming benchmark models in capturing both linear and nonlinear temporal dynamics across demand and supply curves.

3. **Extension to non-linear models (autoencoder):** We propose a model similar to the BME-model. The key differences are that we consider step interpolation in this model rather than linear interpolation

(BME-model). Also, the significant part of the curve is first parameterized by passing through a monotonic autoencoder, before forecasting.

4. **Forecasting using a proposed Echo State Network variant:** A network called the Enhanced Echo State Network is used to forecast the special points and the autoencoders' encodings.

5. **Integration model:** Finally, an 'integration model' is introduced that puts together the three parts of the curve, optimally reconstructing the final curve over the entire span of the support V.

6. **Step interpolation based metric (HC-MAE):** To evaluate the dissimilarity between ground truth and forecasted heterogeneous curves, a novel metrics called Heterogeneous Curve Mean Absolute Error (HC-MAE) is proposed, which has non-negativity, identity, and symmetry properties. It compares curves for all V values, and not just a segment of the curve. This is to ensure that this metric is not dependent on any factors that are not already explicitly in the data, such as abstract distributions.

7. **Combination schemes for 7-day ahead forecasts:**We also propose a model designed for seven-day-ahead forecasting and propose a straightforward yet effective combination method to merge two distinct curve forecasting models, demonstrating improved forecasting performance over each individual model.

Chapter 3

BME model: Linear Variant

3.1 Definition of the B, M, and E points

In order to be more precise, the demand curve of day d and hour h, should be denoted as $D_{d,h}(V)$, and the supply curve as $S_{d,h}(V)$. Notice however that the dynamics index is d only, and the modeling or forecasting problem is that of dealing with a daily series for which h is fixed. Thus, the curves will be henceforth denoted just as $D_d(V)$ and $S_d(V)$, where it is assumed that the hour index is fixed. In addition, V represents values from a set discrete coordinates $V_i \in [0, \infty]$.

Fig. 3.1(a) shows the structurally special points on $D_d(V)$. The volume coordinates of these points are operatively defined as

$$V_{\mathbf{B},dem} = \max\{V \mid D_d(V) = P_{\max}\}, \tag{3.1a}$$
$$V_{\mathbf{M},dem} = \min\{V \mid D_d(V) < P_{\max}\}, \tag{3.1b}$$
$$V_{\mathbf{E},dem} = \max\{V \mid D_d(V) < P_{\max}\}. \tag{3.1c}$$

An entire demand curve, which is made by volume/price pairs, can be 'tokenized' in terms of these three points and one segment, as follows:

1. For V below and equal to $V_{\mathbf{B},dem}$, bid prices P will all be equal to $P_{\max}$ (which is 3000 euros in the IPEX.

2. $\mathbf{M}_{dem}$ is taken at just one bid after $\mathbf{B}_{dem}$.

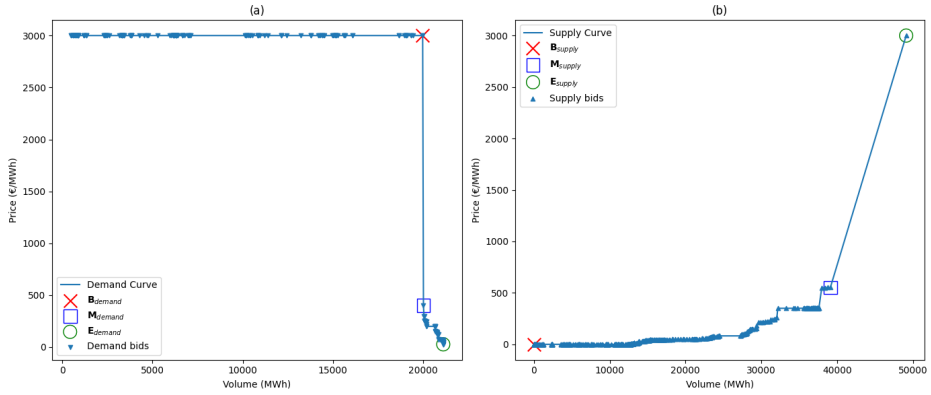

Figure 3.1: (a) A typical demand curve showing actual aggregated bids as small triangles pointing downward, and the locations of the special points $\mathbf{B}_{dem}, \mathbf{M}_{dem}$ and $\mathbf{E}_{dem}$; (b) A typical supply curve showing actual aggregated bids as small triangles pointing upward, and the locations of the special points $\mathbf{B}_{sup}, \mathbf{M}_{sup}$ and $\mathbf{E}_{sup}$. Notice the difference in volume scale. IPEX NORD zone data from GME, for hour 9 of Jan. 2, 2018.

3. As to their coordinate P, by linear interpolation, $\mathbf{B}_{dem}$ and $\mathbf{M}_{dem}$ will be connected by a straight line. This segment is very steep.

4. The portion of the curve between $\mathbf{M}_{dem}$ and $\mathbf{E}_{dem}$ is the most complex part of the demand curve. It is denoted by $\mathbf{L}_{dem}(V)$. This segment is very steep as well.

Eqs. (3.1) and these four rules simplify the curve forecasting problem into the problem of forecasting three points and one section. Once these are forecasted, using the rules a given curve can be reconstructed based on the locations of $V_{\mathbf{B},dem}, V_{\mathbf{M},dem}, V_{\mathbf{E},dem}$, and given the curve between $\mathbf{M}_{dem}$ and $\mathbf{E}_{dem}$, i.e., $\mathbf{L}_{dem}(V)$.

Similarly, Fig. 3.1(b) shows the structurally special points on $S_d(V)$. The volume coordinates of these points are operatively defined as

$$V_{\mathbf{B},sup} = \min\{V\}, \tag{3.2a}$$
$$V_{\mathbf{M},sup} = \max\{V \mid S_d(V) < P_{\max}\}, \tag{3.2b}$$
$$V_{\mathbf{E},dem} = \min\{V \mid S_d(V) = P_{\max}\}. \tag{3.2c}$$

An entire supply curve can be 'tokenized' in terms of these three points and one segment, as follows:

1. The portion of the curve between $V_{\mathbf{B},\text{sup}}$ and $V_{\mathbf{M},\text{sup}}$ is now the most complex part of the supply curve. It is denoted by $\mathbf{L}_{sup}(V)$. It includes the P coordinates $P_{\mathbf{B},\text{sup}}$ and $P_{\mathbf{M},\text{sup}}$. This segment is not steep.

2. $\mathbf{M_{sup}}$ is just one bid before $\mathbf{E_{sup}}$.

3. As to their coordinate P, by linear interpolation, $\mathbf{M_{sup}}$ and $\mathbf{E_{sup}}$ will be connected by a straight line.

4. For V equal or greater than $V_{\mathbf{E},\text{sup}}$, the bid price P is P_{max}. Notice that $\mathbf{E}_{sup}$ is considered the final point of the supply curve, although a flat segment could be added after it to make the curve to appear like a sigmoid (in any case without changing the nature of the problem).

3.2 The 'interesting' part of the curve

It is proposed now to model $\mathbf{L}_{dem}(V)$ by separating the P coordinates of the points of $\mathbf{L}_{dem}(V)$ from their corresponding V coordinates, and treat the two sets, the P set and the V set, as two distinct vectors. The P vector, not a function of V anymore, will be indicated as $\mathbf{L}_{dem}$, by dropping the V dependency. At each time d, $\mathbf{L}_{dem}$ will have N_d entries. The V vector will be discarded, and replaced by a grid with $\bar{N}$ equi-spaced entries. On this grid, a new equi-spaced sequence of prices will be computed by linear interpolation.

This simplification can be justified by looking at the behavior in time of $\mathbf{L}_{dem}(V)$ (an hourly series), depicted in Fig.3.2.

Fig.3.2(a) collects all $\mathbf{L}_{dem}(V)$ segments found in the studied dataset. Due to the weekly periodicity, the segment swings back and forth, clearly a difficult moving target to track. Notice that it was supposed in Chapter 2 that this difficulty could be the reason why the literature tends to concentrate on supply, rather than on demand. Looking carefully at the Figure, it is apparent that some geometry tends to be preserved across these swings. Fig.(3.2)(b) shows the corresponding collection of $\mathbf{L}_{dem}$ vectors (each potentially having a different number of entries). The large variation in P values of Fig.(3.2)(a) has now been reduced a lot. This reduction can be thus be exploited for making forecasting easier and potentially more accurate. More

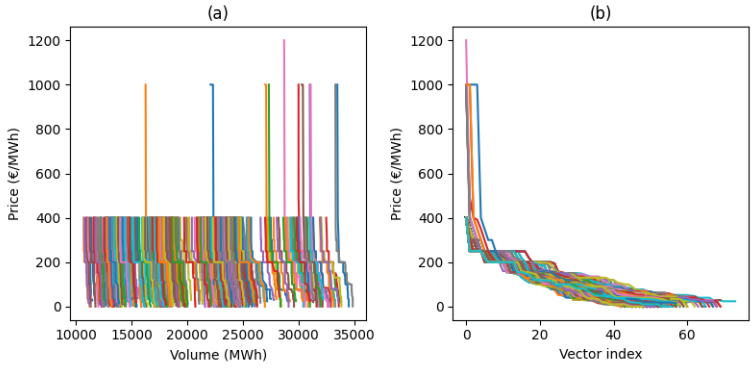

Figure 3.2: (a) shows $\mathbf{L}_{dem}(V)$. (b) shows $\mathbf{L}_{dem}$, where now the abscissa indicates the position of a price value in the vector (a 'vector index').

precisely, once $\mathbf{M}_{dem}$ and $\mathbf{E}_{dem}$ are computed, either directly from data or just forecasted, before separating $\mathbf{L}_{dem}(V)$ from the V axis a uniform grid of $\bar{N} = 100$ points V between $V_{\mathbf{M},dem}$ and $V_{\mathbf{E},dem}$ is created. Then, linear interpolation of the $\mathbf{L}_{dem}$ values is used to get the P values on this grid, so that all new $\mathbf{L}_{dem}(V)$ have the same number of points, and can be plotted together. This makes it possible to use a vector autoregression (VAR) for forecasting $\mathbf{L}_{dem}$ as, indeed, a vector time series.

This procedure can of course be applied to $\mathbf{L}_{sup}(V)$ too, with $\bar{N} = 200$. In this supply case, the curve is less steep, then less singular. This easier situation can be exploited in order to be more accurate in modeling its pattern.

Hence, for forecasting supply it is better to work on the resulting time series of vectors $\mathbf{L}_{sup}$ directly with a feedforward neural network (FFN), defined on a similar grid of volumes.

3.3 Forecasting the individual segments and points

The individual segments and points, as shown in Figures 3.3 and 3.4, are now to be forecast. As shown in the sketches , this is made in two variants.

The so-called 'pure variant' is shown in Fig.3.3.

- The lower part of the sketch shows that $\mathbf{L}$ is forecast by a Vector

Autoregression (VAR) (demand) or by a Feedforward Network (FFN) (supply).

- The upper part shows that the volume coordinates of the points are forecast by a Least Absolute Shrinkage and Selection Operator (LASSO) linear regression [21], designed to include weekend information.

The so-called 'combined variant' is shown in Fig.3.4.

- The lower part of the sketch shows that $\mathbf{L}$ is forecast by a VAR (demand) or by a FFN (supply), like in the pure case.
- However, the upper part shows that the forecast of the points is carried on in a slightly more complicated - but more accurate - way:
  - For supply, two independent forecasts, a standard LASSO and a naive model forecast are first issued, then combined. The combination coefficients can better choose which forecasts is more appropriate for the specific hour.
  - For demand, two independent forecasts, a standard LASSO and a smarter naive model (which can include weekend twists) forecast are first issued, then combined. Here too, the combination coefficients can better choose which forecasts are more appropriate for the specific hour, since night hours are less subject to weekly seasonality.

After that, the pieces have to be assembled into final curves. This reconstruction procedure will need however a measure of the distance between the forecasted curve and the ground truth, as detailed in the following, starting from Subs.3.3.2.

3.3.1 Forecasting L

As to forecasting, a notation will be used where the forecasting target is the current day d, and the regressive data are taken at days before this.

On the one hand, $\mathbf{L}_{dem}$ was forecast using a simple linear vector autoregression regularized by the LASSO technique (because $\hat{N}_{\text{dem}}$ is rather large and can cause instabilities), not including any exogenous variables, and a lag of 1 (thus, a VAR(1)). The VAR model is

$$\mathbf{L}_{dem,d} = \vec{\theta}_{0,dem} + \boldsymbol{\theta}_{1,dem}\mathbf{L}_{dem,d-1} + \vec{\epsilon}_{dem,d}, \tag{3.3}$$

where $\vec{\epsilon}_{dem,d}$ represents i.i.d. vector Gaussian variables. Once the estimated parameters $\hat{\vec{\theta}}_{0,dem}$ (a vector) and $\hat{\boldsymbol{\theta}}_{1,dem}$ (a matrix) are obtained, the forecast at horizon 1 issued at time $d-1$ for time d is the expectation taken in $d-1$

$$\mathbf{L}^f_{dem,d-1} = E_{d-1}[\mathbf{L}_{dem,d}] = \hat{\vec{\theta}}_{0,dem} + \hat{\boldsymbol{\theta}}_{1,dem}\mathbf{L}_{dem,d-1}. \tag{3.4}$$

On the other hand, $\mathbf{L}_{sup}$ was forecast using a FFN, with 1 hidden layer with 164 hidden units, to give $\mathbf{L}^f_{sup,d-1}$. A visual comparison of Figs.3.1(a) and (b) gives further motivation for these two different approaches. Whereas the volume span below $\mathbf{L}_{dem}(V)$ is very short (the curve is steep, there, and the V scale is short), the span below $\mathbf{L}_{sup}(V)$ is large. A neural network is more suitable for capturing the spread nonlinearity of this segment.

3.3.2 Pure Variant

The 'pure variant' is based on a simple linear autoregression for forecasting the V coordinates $V_{\mathbf{B}}$, $V_{\mathbf{M}}$, and $V_{\mathbf{E}}$ of the special points, for both demand and supply curves. It should be remembered that for demand curves $V_{\mathbf{M}}$ corresponds to the left extreme of $\mathbf{L}_{dem}$ and $V_{\mathbf{E}}$ corresponds to the right extreme of $\mathbf{L}_{dem}$, whereas for supply curves $V_{\mathbf{B}}$ corresponds to the left extreme of $\mathbf{L}_{sup}$ and $V_{\mathbf{M}}$ corresponds to the right extreme of $\mathbf{L}_{sup}$. Thus, after having obtained the forecasts of $\mathbf{B}_{dem}$, $\mathbf{M}_{dem}$, $\mathbf{E}_{dem}$, and $\mathbf{L}_{dem}$, these pieces can be glued together in a simple way into a complete $D(V)$ curve. The same for the supply $S(V)$ curve.

On the one hand, three uncoupled autoregression (AR) equations with lags 1 and 7 only, and an exogenous term with a dummy, are used for demand. These are

$$V_{\mathbf{B},dem,d} = \theta^{AR}_{0,dem} + \theta^{AR}_{1,dem}V_{\mathbf{B},dem,d-1} + \theta^{AR}_{2,dem}V_{\mathbf{B},dem,d-7} + \theta^{AR}_{3,dem}\mathbb{1}_{(d\,\in\,\mathbf{H}\,)} + \varepsilon_{\mathbf{B},dem,d}, \tag{3.5}$$

$$V_{\mathbf{M},dem,d} = \theta^{AR}_{4,dem} + \theta^{AR}_{5,dem}V_{\mathbf{M},dem,d-1} + \theta^{AR}_{6,dem}V_{\mathbf{M},dem,d-7} + \theta^{AR}_{7,dem}\mathbb{1}_{(d\,\in\,\mathbf{H}\,)} + \varepsilon_{\mathbf{M},dem,d}, \tag{3.6}$$

$$V_{\mathbf{E},dem,d} = \theta^{AR}_{8,dem} + \theta^{AR}_{9,dem}V_{\mathbf{E},dem,d-1} + \theta^{AR}_{10,dem}V_{\mathbf{E},dem,d-7} + \theta^{AR}_{11,dem}\mathbb{1}_{(d\,\in\,\mathbf{H}\,)} + \varepsilon_{\mathbf{E},dem,d}, \tag{3.7}$$

where $\theta^{AR}_{k,dem}$ ($k \in [0, \ldots, 11]$) are the AR parameters, $\varepsilon_{\cdot,dem,d}$ are i.i.d. Gaussian shocks, and $\mathbf{H}$ in the indicator function in the dummy terms

$$\mathbb{1}_{(d\in\mathbf{H})} = \begin{cases} 1 & \text{if } d \in \mathbf{H}, \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

represents the set of all public holidays. Lag 7 is included because of the weekend weekly periodicity, and the dummy is included because holidays too can affect the demand curve [22][23]. From these equations one gets the forecasts $V^f_{\mathbf{B},dem,d-1}$, $V^f_{\mathbf{M},dem,d-1}$, $V^f_{\mathbf{E},dem,d-1}$ as expectations, like for Eq.(3.4).

On the other hand, three uncoupled AR(1) equations with lag 1 only, and no exogenous terms are used for supply- These are, using the same conventions as above,

$$V_{\mathbf{B},sup,d} = \theta^{AR}_{0,sup} + \theta^{AR}_{1,sup} V_{\mathbf{B},dem,d-1} + \varepsilon_{\mathbf{B},sup,d}, \tag{3.9}$$

$$V_{\mathbf{M},sup,d} = \theta^{AR}_{2,sup} + \theta^{AR}_{3,sup} V_{\mathbf{M},dem,d-1} + \varepsilon_{\mathbf{M},sup,d}, \tag{3.10}$$

$$V_{\mathbf{E},sup,d} = \theta^{AR}_{4,sup} + \theta^{AR}_{5,sup} V_{\mathbf{E},dem,d-1} + \varepsilon_{\mathbf{E},sup,d}, \tag{3.11}$$

Lag 7 and dummies are now not used because weekends and holidays don't affect supply curves much, since such curves are rather more dependent on factors like solar irradiance and cloud cover [9]. From these equations one obtains $V^f_{\mathbf{B},sup,d-1}$, $V^f_{\mathbf{M},sup,d-1}$, $V^f_{\mathbf{E},sup,d-1}$.

The reconstruction of the curves follows in reverse the steps taken when tokenizing them. Reconstructing the forecasted demand curve means first obtaining separately the $V^{f,rec}_{dem}$ and $P^{f,rec}_{dem}$, and $V^{f,rec}_{sup}$ and $P^{f,rec}_{sup}$ vectors from forecast, and then coupling them into $P(V)$ structures. This can be obtained by simple concatenation.

Using a pseudocode-like notation, for demand, this means computing

$$\begin{aligned} V^{f,rec}_{dem,d-1} \leftarrow\ & \texttt{concatenate} \\ & \big(0,\, V^f_{\mathbf{B},dem,d-1},\, \texttt{linearspace}(V^f_{\mathbf{M},dem,d-1}, \\ & \quad V^f_{\mathbf{E},dem,d-1},\, \text{length}(L^f_{dem,d-1}))\big) \end{aligned} \tag{3.12}$$

and

$$P^{f,rec}_{dem,d-1} \leftarrow \texttt{concatenate}(P_{max}, P_{max}, \mathbf{L}^{f}_{dem,d-1}), \tag{3.13}$$

where `linearspace` means constructing an equispaced sequence of values between $V^{f}_{\mathbf{M},dem,d-1}$ and $V^{f}_{\mathbf{E},dem,d-1}$ with the same number of points of $L^{f}_{dem,d-1}$, then stacking these vectors V and P together.

For supply, this means computing the shorter vectors

$$V^{f,rec}_{sup,d-1} \leftarrow \texttt{concatenate} \left(\texttt{linearspace}(V^{f}_{\mathbf{B},sup,d-1},\, V^{f}_{\mathbf{M},sup,d-1},\, \text{length}(L^{f}_{sup,d-1})),\, V^{f}_{\mathbf{E},sup,d-1}\right) \tag{3.14}$$

and

$$P^{f,rec}_{sup,d-1} \leftarrow \texttt{concatenate}(\mathbf{L}^{f}_{sup,d-1}, P_{max}), \tag{3.15}$$

then stacking these vectors P and V together.

3.3.3 Combined Variant

The 'combined variant' is more complex and computationally intensive than the 'pure variant', but it is expected to give better forecasts, because a combination of forecasts has lower variance than any individual forecast, making results more precise, especially if forecasts not well correlated [24]. Hence, a combination of the VAR LASSO model with either a naive (for demand) or a smarter naive (for supply) model is tried. A point to highlight here is that the combination coefficients will not be computed in the usual econometric way, but they will be incorporated in the curve reconstruction procedure itself, by including $\mathbf{L}_{dem}$ and $\mathbf{L}_{sup}$ too. That is, the coefficients will be found as those values that make the forecasted curve as close as possible to the target curve to forecast. As discussed in Subs.3.4, a suitable non-differentiable distance measure is necessary to this end. Hence, the values of the combining coefficients will be obtained by means of a meta-heuristic searching algorithm called differential evolution (DE) [25], appropriate for working with non-differentiable measures.

Before discussing the combination procedure, it is necessary to briefly formally review what the naive and smarter naive models are, and to introduce the distance measure mentioned above.

A naive (or persistence) model [26] for the discrete-time dynamics x_d is given by

$$x_d^f = x_{d-1}. \tag{3.16}$$

No parameters are needed for it. The smarter naive model, also known as the day-of-week model, builds upon the basic naive model by incorporating weekly patterns into the prediction process. It can be seen as a weekly adaptation of the seasonal naive model as defined in Ref. [27], if seasonality is considered to be 1 week long, and only for Saturday, Sunday, and Monday. It is written as

$$x_d^f = \begin{cases} x_{d-7} & \text{if } d \in \{\text{Monday}, \text{Saturday}, \text{Sunday}\} \\ x_{d-1} & \text{otherwise.} \end{cases} \tag{3.17}$$

By incorporating the weekly cycle into the forecasting process, the smarter naive model can capture short-term fluctuations and periodic patterns more effectively than the basic naive model. No parameters are needed for it, either.

Forecasting the full curves within the 'combined variant' scheme

The 'combined variant' consists of expressing the forecasts of the volume coordinates of the important points $V_{\mathbf{B}}$, $V_{\mathbf{M}}$, and $V_{\mathbf{E}}$ as a weighted linear combination of the forecasts of two models, in the specific, LASSO regression and the naive model for supply, and LASSO regression and the smarter naive model for demand. The weighting scheme can be written compactly. For example, for supply (LASSO and naive) it can be written as ($\vec{V} = (.,.,.)$ indicates a vector and its components)

$$\begin{aligned} \vec{V}^{com,f}_{dem,d-1} = \\ (V^{com,f}_{\mathbf{B},sup,d-1}, V^{com,f}_{\mathbf{M},sup,d-1}, V^{com,f}_{\mathbf{E},sup,d-1}) = \\ \theta_1^{com}(V^{LASSO,f}_{\mathbf{B},sup,d-1}, V^{LASSO,f}_{\mathbf{M},sup,d-1}, V^{LASSO,f}_{\mathbf{E},sup,d-1}) + \\ \theta_2^{com}(V^{naive,f}_{\mathbf{B},sup,d-1}, V^{naive,f}_{\mathbf{M},sup,d-1}, V^{naive,f}_{\mathbf{E},sup,d-1}) + \\ \theta_3^{com}\mathbb{1}_{\mathbf{B}} + \theta_4^{com}\mathbb{1}_{\mathbf{M}} + \theta_5^{com}\mathbb{1}_{\mathbf{E}}, \end{aligned} \tag{3.18}$$

where $\mathbb{1}_{\mathbf{B}}$, $\mathbb{1}_{\mathbf{M}}$ and $\mathbb{1}_{\mathbf{E}}$ are scalar indicator variables which add a constant to selected equations, defined for example like

$$\mathbb{1}_{\mathbf{B}} = \begin{cases} 1 & \text{if the component being forecast is } V_{\mathbf{B}} \\ 0 & \text{otherwise,} \end{cases} \tag{3.19}$$

the superscripts 'LASSO' and 'naive' mean that the volume coordinate was obtained, in a preliminary step, by directly using a LASSO AR or a naive model like in Eqs.(3.5), (3.6), (3.7) (exactly like in the 'pure' variant case), and (7.1). Hence, $\vec{\theta}^{com} = (\theta_1^{com}, \theta_2^{com}, \theta_3^{com}, \theta_4^{com}, \theta_5^{com})$ is the weighting parameters vector in Eq.(3.18).Using these parametric linear combinations of volumes to define the 'common' support grid and the corresponding prices vector, in analogy to what obtained for Eqs. (3.12) and (3.13) for the pure case, one can obtain the related combined reconstructed demand curve forecasted on day $d-1$ for day d as the parametric function

$$P_{dem,d-1}^{com,rec,f}(\vec{V}_{dem,d-1}^{com,f}, \mathbf{L}_{dem,d-1}^{f}; \vec{\theta}^{com}), \tag{3.20}$$

which will contain the $\vec{\theta}^{com}$ weights as yet unfitted parameters. This stacked vector can be inserted in the HC-MAE loss of Eq.(6.1), together with the target market curve at d. This loss, in turn, can be minimized by differential evolution, in order to find the estimated weights vector $\hat{\vec{\theta}}^{com}$. Inserting these values $\hat{\vec{\theta}}^{com}$ in Eq.(3.20) allows one to obtain the forecast in the 'combined variant' scheme for the demand curve. The same can be done for the supply curve.

As such, very importantly, with just a little diversification for the two phenomenologically different cases of demand and supply, the proposed model can work for both demand and supply.

3.4 Overall Architecture

Now, results about forecasting at horizon 1 will be presented, with the assumption that the curves from different hours are independent from each other, and hence, can be studied independently from each other. This forecast can be made in different ways.

The model setup tested here is partly based on directly forecasting the three special points of the curve **B**, **M** and **E** (that is, not using a grid as a

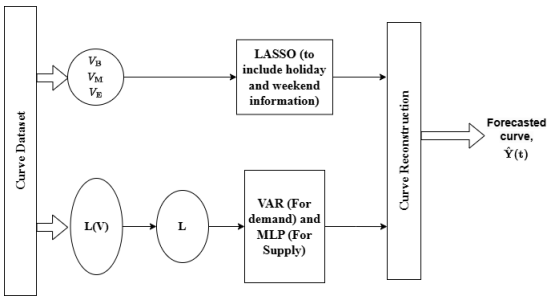

Figure 3.3: Architecture of a pure variant of the framework, using LASSO only.

primary object) shown in Fig.3.1, and forecasting a gridded interpolation of the segment between **M** and **E** in the case of the demand curve, and between **B** and **M** in case of the supply curve. This segment is generically called **L** (that is, without subscripts and superscripts), and ,only in case, a subscript *dem* or *sup* will be appended to it. The forecast of the special points is obtained by a simple linear regression of LASSO (Least Absolute Shrinkage and Selection Operator) [21] type. The forecast of **L** is obtained using a VAR in the case of the demand curve, and using an FFN in the case of the supply curve. Two different simple ways of aggregating these pieces into final curves will be explored, called 'pure variant' and 'combined variant'. The sketches of these two variants are depicted in Figures 3.3 and 3.4. As can be seen from Fig.3.4, the combined variant contains, concerning the 'pure variant' in Fig.3.3, two further small forecasting sub-models. One of them is the so-called 'naive' model, and the other one is the so-called 'smarter naive' model [27]. Both are very popular and very effective in electricity forecasting, but only in the standard autoregression case [28] (that is, not in full curve autoregression). In the standard case, the 'smarter naive' model is obtained as an evolution of the 'naive' model, also called 'persistency' model, in which the forecast of the next value is the current value. The 'smarter naive' model chooses to forecast the next value based on either the current value or some older value, in order to keep into account seasonality. These two parameter-free models can be adapted to the curve case.

The resulting setup, a collection of the two model variants depicted in Figs.3.3 and 3.4, is collectively called the BME model.

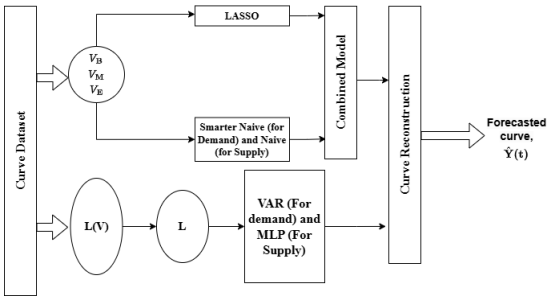

Figure 3.4: Architecture of a combined variant of the framework.

Chapter 4

BME model: Nonlinear extension with monotonic autoencoder

Now extend the BME model into the non-linear domain by using non-linear models for forecasting the special V points, and using non-linear modelling for the special segment of the curve. Furthermore, we use step interpolation in this extension so that we end up with only 2 special points, **A**, and **B** to tokenize the curves.

4.1 Definition of the A, and B points

Fig.(4.1) and Fig.(4.2) show plots of a typical demand $P_t^D(V)$ and supply $P_t^S(V)$ sequence-plus-curve respectively, interpolated in steps, from the NORD zone of the IPEX for hour 9 on January 1, 2018. What we consider special points are marked explicitly as $\mathbf{A}_{t,demand}$, $\mathbf{A}_{t,supply}$, to be called more simply $\mathbf{A}_t$, and $\mathbf{B}_{t,demand}$, $\mathbf{B}_{t,supply}$, to be called more simply $\mathbf{B}_t$. The curve segments between the points $\mathbf{A}_t$ and $\mathbf{B}_t$ (included), to be called respectively the 'interesting segments' $\mathbf{C}_{t,demand}$ and $\mathbf{C}_{t,supply}$ (also to be called more simply $\mathbf{C}_t$), are the parts of the curves with most details, since most of the (aggregate) variation of the bids, both in terms of prices and volumes, are concentrated in these regions. The daily dynamics ('longitudinal', in econometric parlance) of the interesting segment $\mathbf{C}_{t,demand}$ of the demand curve $P_t^D(V)$ over time for the entire dataset, from January 1, 2018 to December

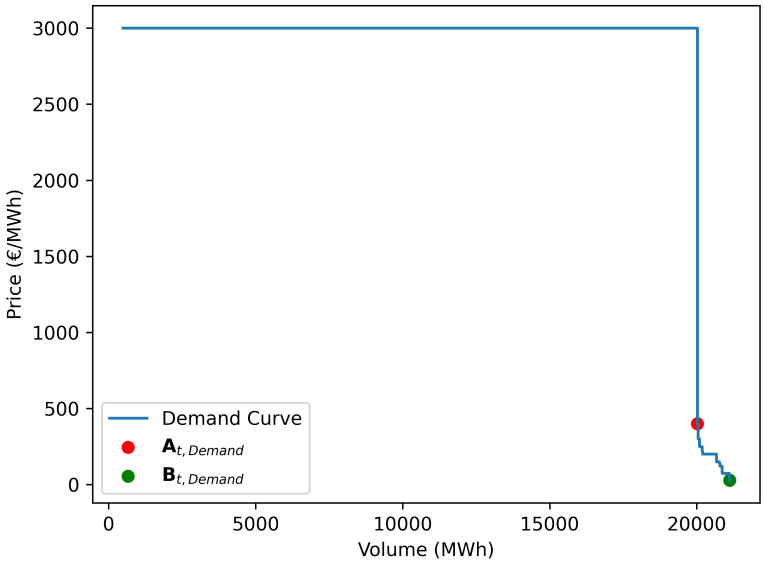

Figure 4.1: $\mathbf{A}_{t,demand}$ and $\mathbf{B}_{t,demand}$ for demand curve, $P_t^D(V)$, for hour 9 of January 1, 2018.

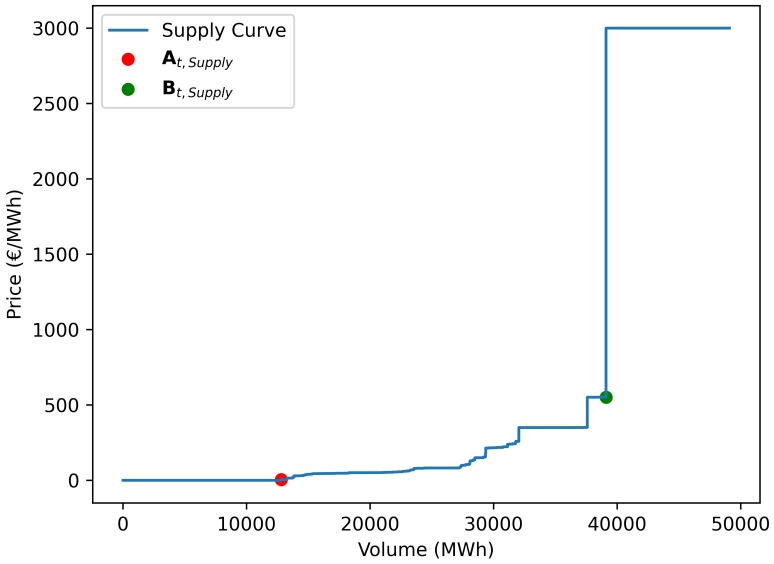

Figure 4.2: $\mathbf{A}_{t,Supply}$ and $\mathbf{B}_{t,Supply}$ for supply curve, $P_t^S(V)$, for hour 9 of January 1, 2018.

31, 2019 is as shown in Fig. (3.2) (a) and (b). It is very relevant to notice that this segment of the sequence, whereas widely and constantly shifting back and forth on the V axis, maintains more or less the same profile.

This regularity is exploited in this model.

4.1.1 Decoupling the Prices from the Volumes: $\hat{C}_t$

One of the major difficulties of modeling and forecasting the curves is that their interesting segment swings back and forth in time (with weekly seasonality, at least), while displaying a changing complicated stepped structure within itself, as discussed in Sec. 3.2. We used the same procedure as described in Sec. 3.2 to model this interesting segment. Instead of using linear interpolation as in Sec. 3.2, for the autoencoder extension, we use step interpolation. After this procedure, a dataset of equisampled vectors $\hat{C}_{t,demand}$ and $\hat{C}_{t,supply}$ is obtained for the entire dataset.

As shown in Fig.(3.2) (b) for the case of demand, these vectors do not shift so much widely on the fixed and equispaced support, especially in comparison with what happens in Fig.(3.2) (a). This can make forecasting much easier. For all t, $\hat{C}_{t,demand}$ and $\hat{C}_{t,supply}$ are defined on the same sequence of indices of the grid, and these indices are not related to volumes anymore.

Hence, the procedure consists of three steps. The first step is to extract from each curve the window where most action is concentrated. The second step is to mark its bounds (the $\mathbf{A_t}$ and $\mathbf{B_t}$ points at the ends of the interesting segment $\mathbf{C_t}(V)$). The third step is to assume that the prices' pattern of each window is somewhat invariant with respect to dilatation and compression of the related volume segment boundaries. Using for all interesting segments in the dataset the same grid means that real data in each segment is dilated or compressed according to the distance between the V components of $\mathbf{A_t}$ and $\mathbf{B_t}$. The full curve forecasting problem will be hence reduced to that of forecasting the box position and that of forecasting the (reduced) internal dynamics of the box.

4.2 Monotonic Autoencoder

Many of the works on curve forecasting use dimensionality reduction [11, 16, 29]. This is because, already at a first sight, part of the curve data look redundant. In the demand curves, for example, price variation is concen-

trated over just a narrow range of V and P values, whereas only one price value like $P_{\max}$ or $P_{\min}$ corresponds to volumes outside the interesting segment. Discarding these 'outside' points would hence have no impact on the shape of the curve. The supply curves as well contain several steps where the price P remains constant over a range of bid volumes. Discarding points 'internal to these internal steps' would have no impact as well on the shape of the curve. That is to say that the the shape of aggregated curves $P(V)$ is actually related to only a small fraction of all available data points $p(v)$. Said in another way, volume aggregation washes out a lot of information contained in the pre-aggregation dataset on which forecast is based. This is in contrast with the expectation that accurate forecasts should make use of all information available. Hence, dimensionality reduction techniques are used in the hope of automatically capturing just relevant pieces of information only, while still using all pre-training information. However, the issue of what is the best balance in information selection has not been resolved in the literature so far, and the presented framework proposes a way in between.

Noticeably, by means of the use of the points $\mathbf{A_t}$ and $\mathbf{B_t}$ and the vector $\hat{C}_t$ discussed above, information present in the curve is already compressed to some extent. However, it is not yet compressed to a satisfactory level. Actually, within $\hat{C}_t$ there seems still to be structure, the internal steps, which can be surely further reduced to some extent. Yet, this reduction must be made with care, because the profile of $\hat{C}_t$ is monotonic both for demand and supply, and such monotonicity must be preserved. It is then hoped to compress a further amount of pre-training information by means of an autoencoder, but this autoencoder must be designed in such a way to preserve monotonicity.

Hence, our setup will start from the general ideas discussed in Ref.[30] for biased generic neural networks, and develop them to achieve guaranteed (hence, not just biased towards) cross-sectional monotonicity, being specifically inspired by Fig.3 therein.

As a first step, the proposed monotonic autoencoder will encode the sequence $\hat{C}_t$, while trying to maintain constant the sign of the sequence changes in the output of the network. This will generate an internal, reduced-dimension representation of monotonic $\hat{C}_t$, while biasing towards monotonicity. The compression chosen will reduce $\hat{C}_t$ from its 200 entries to the 50 entries of its internal encoding vector (this compression ratio was obtained as computationally optimal by trial and error). The autoencoder will be trained on the training plus validation dataset segment before the rest of

training is performed. This is what we call pre-training, aimed to capture the basis. The reduced-dimension representation will be then forecast and back-transformed to the forecast of $\hat{C}_t$. As a second step, a refinement will be operated on the autoencoder output in order to guarantee monotonicity in full.

The autoencoder used has five layers (two in the encoder and two in the decoder), and the number of neuron units used for our architecture are

- Encoder: 200 (input size) $\rightarrow$ 128 $\rightarrow$ 64
- Decoder: 64 $\rightarrow$ 128 $\rightarrow$ 200.

More in detail, a special loss function is applied to the the autoencoder network, in which a regularizing term is added to a standard autoencoder loss. Given N as the number of entries of the equispaced interesting segment vector, each entry can be labeled as $\hat{C}_{t,i}$ where $i = 1, \ldots, N$. $\hat{C}_{t,i,\text{demand}}$ and $\hat{C}_{t,i,\text{supply}}$ can be defined consequently. A monotonicity vector M_t can be defined as

$$M_t = (\hat{C}_{t,i+1} - \hat{C}_{t,i}| \quad \text{for} \quad i = 1, 2, \ldots, N-1). \tag{4.1}$$

$M_{t,demand}$ and $M_{t,supply}$ can be defined consequently for the specific cases of demand and supply. M_t contains the first-differences along the vector $\hat{C}_t$, and, when computed on a monotonic vector, M_t should have values less than or equal to zero for decreasing functions, and greater than or equal to zero for decreasing functions. Using M_t, two different loss functions, L_{demand} and L_{supply} can hence be defined, for demand and supply curves respectively. For demand curves the value of L_{demand} at time t is defined as

$$L_{t,demand} = \frac{1}{N}\sum_{i=1}^{N}(\hat{C}_{t,i,\text{demand}} - C_{t,i,\text{demand}})^2 + \frac{\lambda}{N-1}\sum_{i=1}^{N-1}\max(0, M_{t,demand}), \tag{4.2}$$

where the first term has the usual MSE form, and λ in front of the second (regularizing) term is a hyperparameter that controls the amount of 'pressure' on the loss towards of monotonicity. For supply curves, $L_{t,Supply}$ is defined in analogy as

$$L_{t,supply} = \frac{1}{N}\sum_{i=1}^{N}(\hat{C}_{t,i,\text{supply}} - C_{t,i,\text{supply}})^2 + \lambda\frac{1}{N-1}\sum_{i=1}^{N-1}\min(0, M_{t,\text{supply}}). \tag{4.3}$$

When minimized, $L_{t,demand}$ encourages the 'monotonized' $\hat{C}_{t,\text{demand}}$ to be monotonically decreasing, since $M_{t,\text{demand}}$ is pushed downward towards 0 by the last term of Eq.(4.2). When minimized, $L_{t,supply}$ encourages the 'monotonized' $\hat{C}_{t,\text{supply}}$ to be monotonically increasing. The value of λ is chosen to be 1, after a grid search in $[0.01, 2]$.

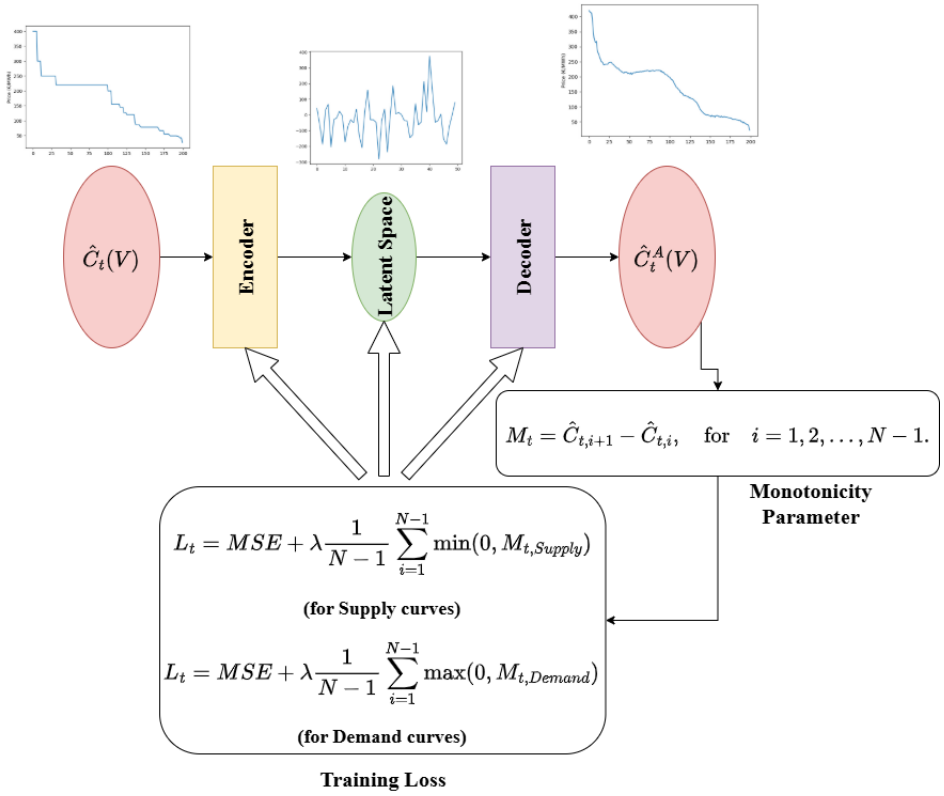

Figure 4.3: Architecture of the monotonic autoencoder as related to its use in the first step. Notice the input and output shapes of $\hat{C}_t$ and $\hat{C}_t^A$

.

The autoencoder has hence an encoder part whose input is $\hat{C}_t$, and which output is an encoding e_t (that is, the embeddings $e_{t,demand}$ or $e_{t,supply}$, respectively). It also has a decoder part whose input is e_t and whose output could also be called $\hat{C}_t^A$, where the superscript A denotes the effect of the monotonic autoencoder. Using this autoencoder on a training set of disentangled interesting segments $\hat{C}_t$, one can hence create a time series of corresponding

training encodings e_t, which are compressed vectors corresponding to the 'almost monotonized' vectors. Subsequently, on this training set we can train and use the EESN defined in Eq.(4.7). Then, on a test set, Eq.(4.7) will allow to forecast independently each entry of e_t obtained inside the autoencoder (fed with the test $\hat{C}_t$). This will deliver the compressed vector e_t^f (that is, $e_{t,demand}^f$ and $e_{t,supply}^f$), as it was shown in Fig.4.4. Thus, the pre-trained decoder is used to construct a 'monotonized' forecast $\hat{C}_t^{Af}$ from the forecast e_t^f.

At this first step monotonicity is yet just encouraged, but not guaranteed. This can be seen numerically and graphically in Fig.4.3, where the entire architecture of the monotonic autoencoder is sketched, and where the outputs of the model at each stage are shown above each related component. Specifically, by looking at the third graph on the right, just above the output, it becomes clear that this output is not necessarily monotonic in full.

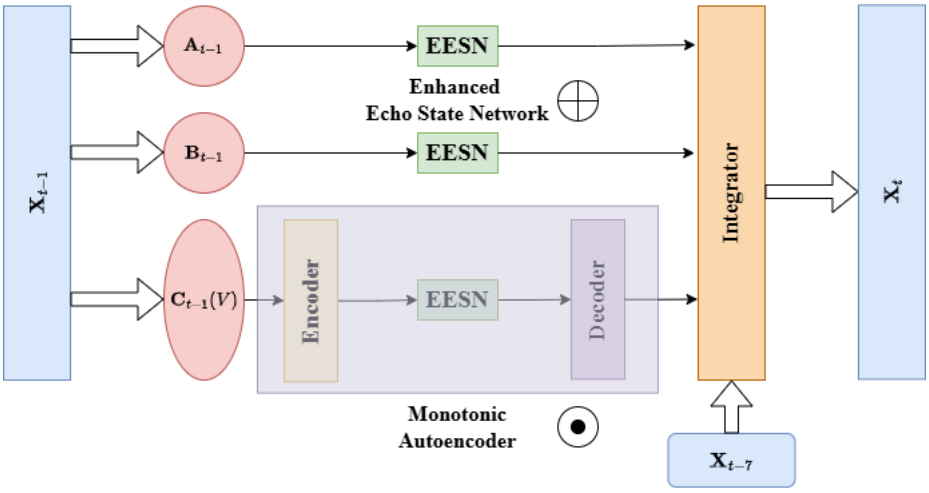

Figure 4.4: Graphical representation of one possible implementation of the entire architecture of the proposed framework. X_t represents the curve sequence, and today's curve at t is forecast with data of yesterday at $t-1$ and one week before at $t-7$. More graphical information about the EESN boxes ($\bigoplus$) in Fig.4.6, and about the monotonic autoencoder box ($\bigodot$) in Figs.4.3 and 4.5.

4.2.1 Nearest Neighbors averaging scheme

At the end of the following second step, monotonicity of the final forecast is on the contrary fully ensured by the following refining procedure, which uses the computed $\hat{C}_t^{Af}$ just as a guiding reference:

1. The $K = 5$ closest neighbors N_k of $\hat{C}_t^{Af}$ are picked up from the original (then, step-like) pre-training set data $\hat{C}_t$ (actually from the training segment). How close these neighbors are is evaluated by using the L_1 distance [31] (also called Manhattan distance) $d(\hat{C}_t^{Af}, N_k)$ between $\hat{C}_t^{Af}$ and N_k.

2. The final, guaranteed monotonic, refined forecast is computed as the distance-weighted average of the neighbors as

$$\hat{C}_t^f = \sum_{k=1}^{K} \frac{N_k}{d(\hat{C}_t^{Af}, N_k)}. \tag{4.4}$$

Now the forecast $\hat{C}_t^f$, built as guided by $\hat{C}_t^{Af}$, is truly monotonic. This is because all the constituent terms of the sum in Eq. (4.4) are monotonic, being a ratio between N_k, monotonic, and a positive factor lower than 1, so that their combination is necessarily monotonic. In passing, note that the five neighbors are chosen from the pre-training data, and hence, no future information is used in the forecasting process when run on the test set. This procedure is graphically sketched in Fig. (4.5).

Recapitulating, our monotonic autoencoder approach to the forecast of the interesting segment is based on: 1. identifying an internal basis by means of the modified autoencoder; 2. using this basis to generate the compressed embeddings corresponding to segments; 3. using the EESN to forecast the embeddings; 4. using the decoding part of the autoencoder to generate quasi-monotonic forecasts of the segments which will act as a 'driver'; 5. refining these forecasts by means of the first-neighbor procedure (guided by the 'driver' $\hat{C}_t^{Af}$), taking them to guaranteed monotonicity. All this procedure doesn't use functional methods, unlike most of the related literature, and mainly relies on sequence representation of the curves.

4.2.2 Reconstruction of the curves

As a final step, given the individual forecasts $\hat{C}_t^f$, $\mathbf{A}_t^f$ and $\mathbf{B}_t^f$, the curves need to be assembled. Demand and supply need a slightly different procedure.

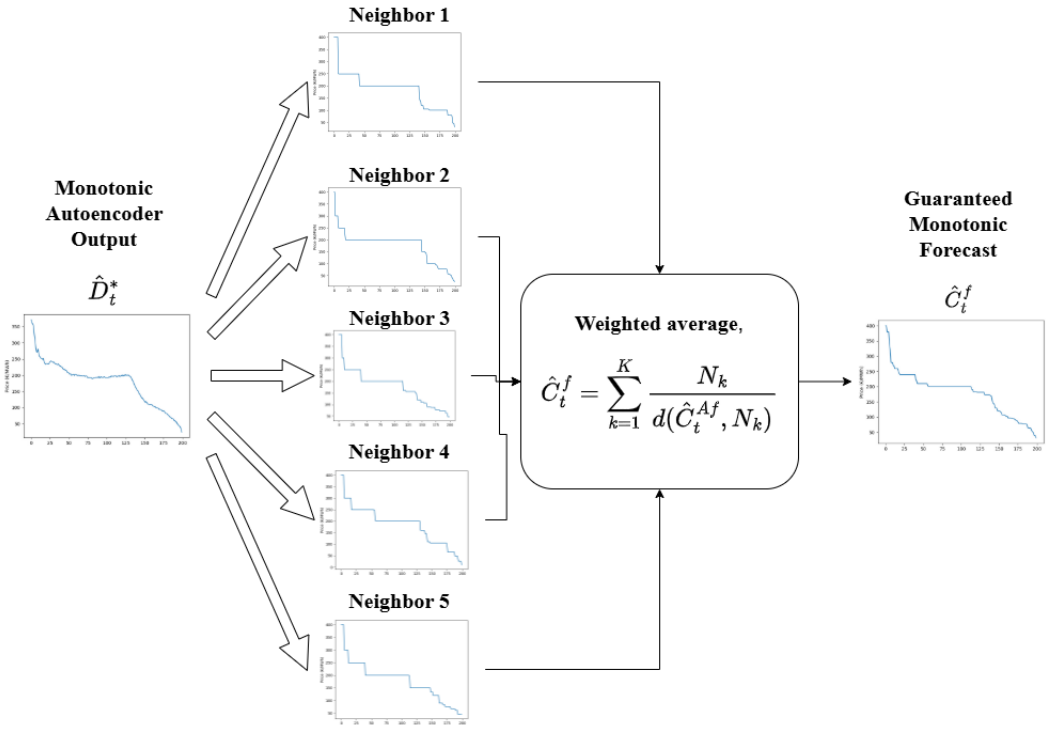

Figure 4.5: Monotonic autoencoder second step. Creating a monotonic approximation from the autoencoder output.

Demand Curves

Given the forecasts $\hat{C}_t^f$, and consequently $\mathbf{A}_{t,demand}^f$ and $\mathbf{B}_{t,demand}^f$, the full demand curve is assembled in the following manner:

1. Define a volume vector, $V_{demand,assem}$, that is the concatenation of 0 and an equispaced grid of 200 points between $\mathbf{A}_{t,demand}^f$ and $\mathbf{B}_{t,demand}^f$.
2. Define a price vector, $P_{demand,recon}$, that is the concatenation of the value of P_{max} (corresponding to $V = 0$), and $\hat{C}_t^f$.
3. Concatenate the two vectors together as $(V_{demand,assem}, P_{demand,assem})$ to obtain the fully assembled curve.

Supply Curves

Similarly, the full supply curve is assembled in the following manner:

1. Define a volume vector, $V_{supply,assem}$, that is the concatenation of 0, and an equispaced grid of 200 points between $\mathbf{A}_{t,supply}^f$ and $\mathbf{B}_{t,supply}^f$.
2. Define a price vector, $P_{supply,assem}$, that is the concatenation of P_{max} (corresponding to the V coordinate of $V = \mathbf{B}_{t,supply}^f$), and $\hat{C}_t^f$.
3. Stack the two vectors together as $(V_{supply,assem}, P_{supply,assem})$ to obtain the fully assembled curve.

4.3 Forecasting the A, and B points and the Interesting Segment $\mathbf{C_t}$

4.3.1 Enhanced Echo State Network (EESN)

One of the machine learning structures used for forecasting is the proposed EESN. It is built in the following way.

As said, the EESN is an enhancement of the vanilla ESN. It not only looks at the history of the time series data as standard ESNs do, but also incorporates external factors (like past, current, and future deterministic exogenous information) that could affect the outcome. In the presented case, the EESN uses holiday timing information to enhance the forecast of $\mathbf{A}_t$, of

$\mathbf{B}_t$, and of each individual element of the autoencoder encodings. For example, when the EESN forecasts a future $\mathbf{A}_t$, it doesn't just rely on patterns from the past, but also it adds that, conditionally on if today is a holiday and tomorrow the same, the volume value of $\mathbf{A}_t$ should be low. In a sense, this helps peep 'legally' into the future. By combining the memory of past patterns with these external influences, the EESN provides a more accurate forecast than an ESN.

The output $\hat{y}_{t,ESN}$ of the ESN is enriched by means of the following regression (notice, not autoregression) approach. The standard regression of a variable both w_t on the regressor x_{t-1} for which future values cannot be known, and on the deterministic sequence z_t for which future values can be known, can be written as

$$w_t = \beta_0 + \beta_1 \, x_{t-1} + \beta_2 \, z_{\text{any } t} + \epsilon_t, \tag{4.5}$$

where ϵ_t is a sequence of i.i.d. Gaussian variables. Applying Ordinary Least Squares, the aim of the regression is to find estimates $\hat{\beta}_0$, $\hat{\beta}_1$ and $\hat{\beta}_2$ of β_0, β_1 and β_2 by finding the argmin of the sum of terms like

$$\left(w_t - \beta_0 - \beta_1 \, x_{t-1} - \beta_2 \, z_{\text{any } t}\right)^2 . \tag{4.6}$$

Be H the set of all public holidays and weekends in Italy, and $\mathbb{1}_{t \in H}$ an indicator function for whether t belongs to H. At time t (that is, day d for a fixed hour h), for a scalar time series of data values $\bar{y}_t$ and for a forecasting horizon $f = 1$:

1. obtain from the series the ESN forecast $\hat{y}^f_{\text{t-1, ESN}}$ issued at $t-1$ for today t using a vanilla ESN. The hyperparameters of the ESN are the same for all 24 hours. They are found during validation to be: reservoir size, 80 (fixed); leaking rate, 1.0 (no past activation required); spectral radius, 0.9995;

2. use Eq.(4.6) to regress the ground truth of today $\bar{y}_t$ (as w_t) on the forecast for today $\hat{y}^f_{\text{t-1, ESN}}$ (as x_{t-1}), and on $\mathbb{1}_{t \in H}$ (as $z_{\text{any } t}$), then obtain $\hat{\beta}_0$, $\hat{\beta}_1$ and $\hat{\beta}_2$;

3. with the estimated coefficients define the EESN forecast $\hat{y}^f_{\text{t-1, EESN}}$ made yesterday at $t-1$ for today at t as

$$\hat{y}^f_{\text{t-1, EESN}} \equiv \hat{\beta}_0 + \hat{\beta}_1 \, \hat{y}^f_{\text{t-1,ESN}} + \hat{\beta}_2 \, \mathbb{1}_{t \in H}. \tag{4.7}$$

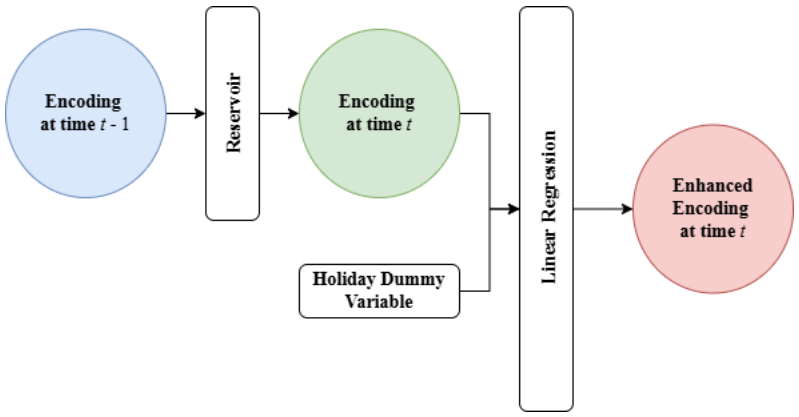

Figure 4.6: Architecture of the enhanced ESN (EESN)

The architecture of the EESN is shown in Fig.(4.6). Notice that a $\beta_2 \mathbb{1}_{t \in H}$ term could also be added inside some of the very many reservoir neuron components. However, because of the highly nonlinear environment of the reservoir, the effect would be not controllable and almost meaningless.

4.3.2 Combination and Integration

The pieces **A**, **C**, and **B** of the stacked structure have been obtained so far using uncoupled autoregressions, thus, these pieces don't share information with each other. One can then try to optimize their assembling: this will be called 'integration'.

This optimized assembling can be obtained by defining a measure for the distance, or dissimilarity, between the assembled forecast curve and the ground truth data, and minimizing with respect to it. This step will also prepare parameters to be used in the out-of-sample test phase. This simple distance will be called Heterogeneous Curve Mean Absolute Error (HC-MAE) metric and is discussed in Sec. 6.3.

Going back to the task of optimally obtaining the assembled curve, one can also exploit in this computation some information about past lags again, doing it in four steps.

For clarity, with a slight abuse of notation in which special point symbols like $\mathbf{A}^f_{t-1}$ stand for their V coordinate:

1. Combine the forecasts made yesterday for today $\mathbf{A}^f_{t-1}$ and $\mathbf{B}^f_{t-1}$ with

$\mathbf{A}_{t-7}$ and $\mathbf{B}_{t-7}$:

$$\mathbf{A}^f_{t-1,combined} = \theta_0 + \theta_1 \mathbf{A}^f_{t-1} + \theta_2 \mathbf{A}_{t-7} \tag{4.8}$$

$$\mathbf{B}^f_{t-1,combined} = \theta_0 + \theta_1 \mathbf{B}^f_{t-1} + \theta_2 \mathbf{B}_{t-7} \tag{4.9}$$

Namely, the previously made forecast $\mathbf{A}^f_{t-1}$ is adjusted by replacing it with a weighted sum of (the volumes of) $\mathbf{A}^f_{t-1}$ and one week prior data $\mathbf{A}_{t-7}$. Similarly, the previously made forecast $\mathbf{B}^f_{t-1}$ is adjusted by replacing it with a weighted sum of $\mathbf{B}^f_{t-1}$ and one week prior data $\mathbf{B}_{t-7}$, using the same coefficients θ_0, θ_1 and θ_2 from Eq.4.8. This distributes information (otherwise obtained individually) between the two forecasts. This combination bolsters the forecast by repeating past information directly. Moreover, in a sense, this direct injection of information from $t-7$ in a higher layer of a forecasting hierarchy can be considered similar to the stabilizing use of skip layers in residual neural networks [32].

2. Do the same with $\mathbf{C}_t$, but with other coefficients.

$$\mathbf{C}^f_{t-1,combined} = \theta_3 + \theta_4 \mathbf{C}^f_{t-1} + \theta_5 \mathbf{C}_{t-7} \tag{4.10}$$

3. Assemble from $\mathbf{A}^f_{t-1,combined}$, $\mathbf{B}^f_{t-t,combined}$ and $\mathbf{C}^f_{t-1,combined}$ the demand and supply curves using the assembling algorithms described in Secs. 4.2.2 and 4.2.2, respectively. The forecasts will contain the parameters θ_j, where $j = 0, \ldots, 7$.

4. Minimize the HC-MAE (between the forecasted and ground truth curves) with respect the parameters θ.

The parameters θ_j of Eqns. (4.8), (4.9) and (4.10) are learned using 'differential evolution' [25], an evolutionary optimization algorithm that can optimize non-differentiable objective function, because the HC-MAE is non-differentiable.

From the optimal integration of $\hat{C}_M$ with $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ in the training plus validation set, the parameters θ are hence obtained. They can be in turn used to assemble the forecasted curve in the test set. The HC-MAE, on its part, can be used to assess quality of these out-of-sample forecasts.

4.3.3 Overall Architecture, recapitulation

In the proposed framework, forecasting a DAM curve as partly a sequence and partly an interpolation curve, is hence divided into four steps, namely:

1. Single out the interesting segment $\mathbf{C}_t$ of the curve, that is, the curve segment between the volume coordinates of the special points $\mathbf{A}_t$ and $\mathbf{B}_t$. Then forecast the volumes related to the points $\mathbf{A}_t$ and $\mathbf{B}_t$.

2. For the segment $\mathbf{C}_t$, the prices P are disentangled from the volumes V, to obtain individual vectors $\hat{C}_t$ instead of curves.

3. These vectors $\hat{C}_t$ are then compressed to lower dimensional encodings via a monotonic autoencoder. The encodings are forecasted to obtain the curve at a future time step, and the decoder is used to reconstruct the curve. To fully ensure the monotonicity, the distance-weighted average of the five nearest neighbors of the forecasted interesting segment is obtained. Averaging is made according to Eq.(4.4). Step interpolation, appropriate for the IPEX data, is used in some of the computations. Since the extrema of $\hat{C}_t$ correspond to the P coordinates of $\mathbf{A}_t^f$ and $\mathbf{B}^f$, this completes the forecast of the special points.

4. In order to assemble the curve from the A, B, C pieces, an optimal integration of $\mathbf{C}_t^f$ with $\mathbf{A}_t^f$ and $\mathbf{B}^f$ is performed by means of the parameters θ. These parameters are estimated in the validation set, and in the test set they are used as they are found in the validation set.

At training, step 1 allows for finding the EESN coefficients of the special points regressions, step 3 allows for estimating the monotonic autoencoder and finding the coefficients of its forecasting regressions, and step 4 allows for finding the parameters θ. At testing, ESSN and autoencoder parameters, and the parameters θ found in the training phase are applied.

For clarity, it is to be recalled here that the architecture of the entire framework is shown in Fig.(4.4), and the relationship between the use of the dataset train, validation and test segments and the four steps described above is shown in Fig.(6.2). Notice that the combination of the autoencoder part and the EESN part forms a spatiotemporal module (the EESN modules are just temporal modules). This is because this part encapsulates both spatial information via the autoencoder, and temporal information via the EESN.

This setup allows the model to produce forecasts that are both monotonic, which is a feature not found in the literature, but still adaptable to external influences like exogenous drivers.

Chapter 5

Forecasting Using Splines and Combination Models

5.1 Motivation

It was shown that curve forecasting is challenging due to the inherent complexities of the curves and the constraints of the markets. It was shown that forecasting demand and supply curves for the Day-Ahead Market (DAM) presents two main challenges: (1) curve bids are unevenly distributed on the Volume-Price (V-P) plane, both the number of points and the spacing between points vary significantly from day to day and hour to hour; and (2) these curves are monotonic, which means that supply curves show a non-decreasing relationship between the V and P values, and demand curves show a non-increasing relationship.

From a practical point of view, one should add that the IPEX market imposes a practical constraint because it releases the detailed bid data only after a mandatory 7-day delay. This lag imposes a practical constraint on the forecasting horizon, requiring models that can reliably predict these curves seven days ahead.

To address this challenge, in this chapter, we propose a further forecasting framework that uses monotonic spline representations (I-splines) to model market curves while preserving monotonicity. Specifically, curves are now represented through spline coefficients. This significantly reduces the representation complexity and ensures monotonicity. Subsequently, these coefficients are forecasted using two approaches: a Vector AutoRegression

(VAR) model and a Multi-Layer Perceptron (MLP) neural network, that is, an FFN.

We assess the performance of this model with respect to several benchamrks. Our methodology not only addresses the 7-day horizon constraint inherent to the IPEX market but also demonstrates broader applicability for forecasting the market demand and supply curves within similarly structured electricity markets.

5.2 I-Spline Regression of Demand and Supply Curves

We model the curves on the Volume-Price (V-P) plane, with volume V as the abscissa. To enforce the monotonicity inherent in these curves, we employ monotonic I-Splines.

I-Splines are monotonic spline functions constructed from M-Splines $M_i(V)$ as follows:

$$I_i(V) = \int_{V_{min}}^{V} M_i(u)\, du \tag{5.1}$$

where:

- $I_i(V)$ is the i-th I-Spline basis function evaluated at volume V.
- $M_i(u)$ is the corresponding M-Spline basis function evaluated at volume u.
- V_{min} is the minimum volume from which integration starts.

M-splines, or monotone splines, are a family of non-negative spline basis functions used particularly when positivity or monotonicity constraints are required. They are constructed from a sequence of ordered points called *knots*, that partition the domain and determine the shape and flexibility of the spline. Specifically, knots are points $t_1 < t_2 < \cdots < t_m$ that define intervals over which spline functions are polynomial segments smoothly joined

together. Mathematically, an M-spline of order k is defined recursively as:

$$M_i^k(x) = \begin{cases} \dfrac{k\,(x - t_i)}{t_{i+k} - t_i} M_i^{k-1}(x) + \dfrac{k\,(t_{i+k+1} - x)}{t_{i+k+1} - t_{i+1}} M_{i+1}^{k-1}(x), & t_i \leq x < t_{i+k+1} \\ 0, & \text{otherwise} \end{cases} \tag{5.2}$$

with initial conditions defined for $k = 1$.

The knots used for M-splines coincide with those for I-splines.

5.2.1 Choice of Basis Functions, Knot Placement, and Spline Order

The effectiveness of spline methods depends significantly on knot placement, spline order, and boundary conditions. Here, the knot grid was designed to balance curve flexibility and model complexity. We also chose an equidistant knot grid to ensure the generalizability of the method. Specifically, we selected 60 internal knots, placed equidistantly along the volume V axis. For demand curves, these knots were placed from 0 to $35,000$ MWh, while for supply curves, they were placed from 0 to $50,000$ MWh. This dense yet uniformly spaced knot placement enabled the spline representation to accurately capture the structural features of the curves frequently observed in DAM curves, without excessive complexity or overfitting.

The spline order and other configuration of the basis functions followed the default values of the implementation from the widely used R package `splines2`. This choice facilitates reproducibility, ensures numerical stability, and provides a robust basis for subsequent forecasting steps in the following way.

To address the issue of non-uniform bid data, we explicitly pad the curves:

- for bid volumes between 0 MWh and the minimum bid volume V_{min} of that particular day/hour, curves are padded with the minimum bid price P_{min} on the left side (for supply curves), and the maximum bid price P_{max} on the left side (for demand curves).

- for volumes between the maximum bid volume V_{max} of that particular day/hour and the upper bound, we pad the curve with the maximum bid price P_{max} on the right side (for supply curves), and the minimum bid price P_{min} on the right side (for demand curves).

This padding strategy explicitly standardizes the domain for each curve, thereby ensuring uniform representation.

5.2.2 Procedure for Transforming Raw DAM Curves into Spline Coefficients

The raw DAM curve data were converted into spline coefficients using a linear regression-based approach. Initially, each observed DAM curve, described by discrete points in the volume (V)-price (P) space, was approximated as a linear combination of the previously defined I-Spline basis functions. This was done by first evaluating spline basis functions at each observed volume point, creating a design matrix that measured the contribution of each spline basis function to the observed pricing data.

An important consideration is the mismatch in monotonicity direction between the spline basis and the natural shape of the demand curves. While standard I-Spline basis functions are monotonically increasing by construction, DAM demand curves are monotonically decreasing. To resolve this, the basis functions evaluated for demand curves were modified by subtracting their values from unity. This effectively reverses their monotonic direction and aligns the coefficients with the decreasing nature of demand curves.

For supply curves, the splines do not have to be reversed. Formally, the adjusted basis functions are shown in Eqs.(5.3) and (5.4).

$$B_{sup}(V) = I_i(V) \tag{5.3}$$

$$B_{dem}(V) = 1 - I_i(V) \tag{5.4}$$

where $I_i(V)$ denotes the I-spline basis function over V.

After constructing the I-Spline basis $B_{sup}(V)$ and $B_{dem}(V)$ (generically termed $B(V)$, without any subscript), each curve $P_i(V)$ is approximated as a linear combination of these spline bases:

$$P_i(V) \approx \sum_{i=1}^{n} \mathbf{c}_i B_i(V) \tag{5.5}$$

where:

- $P_i(V)$ is the approximated supply curve price at volume V.

- c_i are the coefficients representing the spline basis contribution.
- n is the total number of spline bases.

The coefficients c_i are obtained via ordinary least squares regression. Specifically, the regression was performed with the constraint that all coefficients must be non-negative to satisfy monotonicity constraints.

By following this approach, each inherently non-uniform curve, $P_i(V)$ is transformed into a fixed-length vector of coefficients c_i, suitable for subsequent forecasting.

5.3 Models used in forecasting

Due to the structure of day-ahead markets, all 24 hourly curves are determined simultaneously. So in our approach, the forecast for the curve at hour h of day d is assumed to depend solely on the curve at the same hour h exactly seven days prior because of the 7-day forecasting horizon. Hence, the spline coefficient c_i can be rewritten as $c_{d,h}$ for each coefficient vector on day d at hour h. We forecast the spline coefficients $c_{d,h}$ using two distinct approaches.

The first method employs a VAR model, formulated as:

$$\mathbf{c}_{d,h} = A\mathbf{c}_{d-7,h} + B + \epsilon_{d,h} \tag{5.6}$$

where $\mathbf{c}_{d,h}$ represents the vector of spline coefficients at hour h of day d, A is the coefficient matrix, B is the intercept vector, and ϵ denotes the error term.

The second forecasting approach uses a MLP. Before feeding as inputs, the coefficients are normalized by dividing by P_{max}. The MLP then forecasts using a temperature-scaled softmax output layer to keep the output vector elements between 0 and 1. After prediction, the output vector is rescaled by multiplying by P_{max}. The entire process of curve forecasting involves (in order) performing I-Spline regression, forecasting coefficients using VAR or MLP, and subsequently reconstructing the curves using the original spline basis.

5.4 Reconstruction and Overall Architecture

5.4.1 Curve Reconstruction from Forecasted Coefficients

Once the spline coefficients have been forecasted, the demand or supply curve is reconstructed using the known spline basis. Given a set of forecasted spline coefficients $\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_n$ and the selected spline basis functions $B_1(V), B_2(V), \ldots, B_n(V)$, the reconstructed curve $\hat{P}(V)$ is obtained by the linear combination:

$$\hat{P}(V) = \sum_{k=1}^{n} \hat{c}_k B_k(V). \tag{5.7}$$

This reconstruction step converts the forecasted spline coefficients back into the full curve.

5.4.2 Overall Architecture

Overall, the spline-based forecasting architecture consists of three primary stages: curve parameterization using I-splines, spline coefficient forecasting, and curve reconstruction. This approach is modular, which allows for flexibility in the forecasting models employed and the functional model involved.

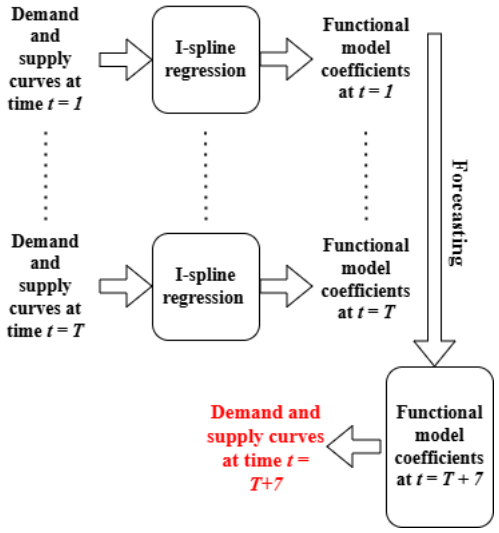

Figure 5.1: Overall architecture of the spline-based forecasting model.

This architectural setup, illustrated in Figure 5.1, shows the overall architecture of the spline-based curve forecasting model.

5.5 Combination of curve forecasting models

5.5.1 Motivation for Combining Forecasting Models

According to our preliminary tests, the naive approach alone outperformed all other models, including the suggested spline-based forecasting models. The significant temporal variability shown along the V axis, especially over a 7-day forecasting horizon, is responsible for the naive model's outstanding performance.

From this experiment it was understood that it is advantageous to use the naive model as a supplementary model rather than as a pure competitor. One well-known method to increase prediction accuracy and robustness, especially in electricity day-ahead markets, is in fact model combination, sometimes referred to as ensemble modeling [33].

For example, combining the naive model with the spline-based models may assist in rectifying forecast variances brought on by functional misspecification while still utilizing the spline approach's structured forecasts. Inspired by this, we suggest a straightforward yet powerful linear combination framework that combines predictions from various models

The structure of the curves, which are made up of two coupled vectors, volume (V) and price (P), is one of the main obstacles to developing a combination model for curve forecasting. The manner in which various forecasting models handle this structure varies greatly. Certain models consider the curve as a function $P = f(V)$ by fixing the volume axis (standardizing or gridding V) and forecasting matching P values. Others treat the curve as $V = f(P)$, setting the price axis and forecasting V values. Since the forecasts from one model may only give a P vector (with V fixed), while the second model provides a V vector (with P fixed), this disparity in output format makes direct combination challenging. The proposed combination framework addresses this issue by formulating separate combination equations for both V and P, allowing it to accommodate and integrate models with differing structural assumptions.

.

5.5.2 Formulation of the Combination Model

The proposed combination model linearly integrates the forecasts from a structured model (such as FAR, X-model, or spline-based models) with the forecasts from the naive model. For a given day d and hour h, the combined forecasts for the volume and price domains are formulated as:

$$V_{d,h}^{f,\text{Combined}} = \theta_0 + \theta_1 V_{d,h}^{f,\text{Model, processed}} + \theta_2 V_{d-7,h}, \tag{5.8}$$

$$P_{d,h}^{f,\text{Combined}} = \theta_3 + \theta_4 P_{d,h}^{f,\text{Model, processed}} + \theta_5 P_{d-7,h}. \tag{5.9}$$

Here:

- $V_{d,h}^{f,\text{Model, processed}}$ and $P_{d,h}^{f,\text{Model, processed}}$ are the processed outputs from the structured forecasting model, after ensuring that the number of points matches that of the naive model via step interpolation.
- $V_{d-7,h}$ and $P_{d-7,h}$ are the naive forecasts, corresponding to the same hour from seven days prior.
- $\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ are combination coefficients, optimized to minimize the error between the combined forecasts and the true market curves.

An important design decision is the dual treatment of both V and P domains. Since some models forecast P as a function of V and others forecast V as a function of P, a combination strategy addressing both domains is crucial for broader applicability.

5.5.3 Practical Considerations and Observations

Several important practical aspects were taken into account while designing and evaluating the combination framework:

- To combine forecasts from various models with the naive model, it is essential to ensure that the number of points in the forecasted curves from both models is the same. To ensure this, the forecasts from the various models were preprocessed using step interpolation to match the number of points of the naive model forecasts.

- The combination framework was designed to handle the coupling of V and P vectors inherent in market curves. This allows it to combine models with different structural assumptions (support on V vs support on P).

Chapter 6

Experimental Setup

6.1 Dataset usage

The dataset used in this thesis is the same for all models, but different time-periods have been used for different cases. In this chapter and the following chapter, we report the data processing, the metrics, and the numerical results related to the models discussed so far. For 1-day ahead forecasting, the dataset used consists of all available individual IPEX DAM bids for the 24 hours of a given day, specific to the NORD zone, recorded by the GME for 730 days from January 1, 2018, to December 31, 2019. These raw data include all bid information necessary to create the aggregated curves.

For the BME model, the dataset usage is shown in Fig. 6.1 for the autoencoder model, the data usage, i.e., the training, validation and testing segments of the data, are shown in Fig. 6.2.

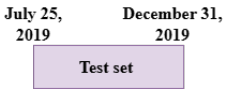

Figure 6.1: Dataset segmentation for the BME model.

For the spline case of 7-day ahead forecasting, curve data from January 1, 2019 to December 31, 2019 was used. The first 265 days from January 1, 2019 to September 23, 2019, were used as the training period. The next 15 days, from September 23, 2019, to October 8, 2019, were used as the validation period, to find the best hyperparameters of the model. The period of 78 days from October 15, 2019, to December 31, 2019, was used as the testing

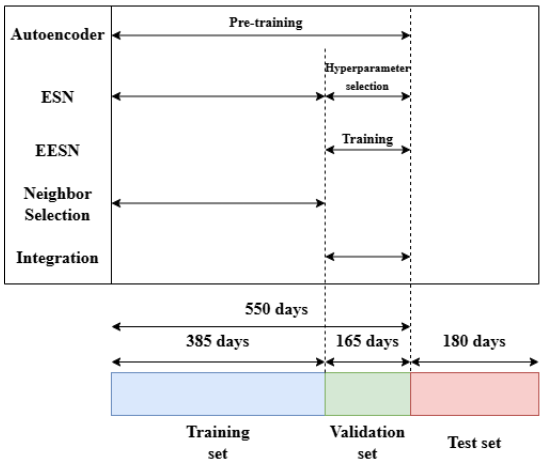

Figure 6.2: Dataset segmentation for the Autoencoder model.

period. Since this is the case of 7-day ahead forecast, we had to skip 7 days between October 8, 2019, and October 15, 2019.

6.2 Obtaining the data

The raw IPEX data used in the paper consist of daily bid information, and can be freely downloaded from the GME web site [10]. In the specific, following the URL which can be found at Ref.[34], one is lead to a page where the options to be selected are "MGP" (Mercato del Giorno Prima), then "Downloads", and then "Public data". A date is then to be entered in order to download the data which are needed for forming the prices/volumes curves of the DAM ('Mercato del Giorno Prima', MGP) for that day. The file is downloaded as "[yyyymmdd]MGPOffertePubbliche.zip" ('MGP Public Offers'). Once unzipped, one gets an XML file that needs to be open with an Excel compatible program. In the XML file, once filtered for the NORD market zone, the bids come in two categories, BID (demand) and OFF (supply). The prices are under the column "ENERGY_PRICE_NO" and the quantities are under "AWARDED_QUANTITY_NO".

Noticeably, on this page one can also find a lot of other interesting information, like data related to the other markets managed by the IPEX besides the MGP, like seven Intra-Day Markets (MI) for intra-day adjustments, the Energy Imports and Exports Management Platform (MPEG) for cross-

border transactions, the Renewable Energy Withdrawal Market (MRR) for renewable energy integration, and the Dispatching Services Market (MSD) for real-time balancing and grid stability.

6.3 Metric used: HC-MAE

The calculation of the dissimilarity between two DAM curves poses a problem. Consider two curves, both of the same type, demand or supply, one the forecast and the other the ground truth. The curves are hence taken at subsequent times, for example $t = 1$ and $t = 2$. In a sloppy notation, one would have forecast (V_1, P_1) and ground truth (V_2, P_2). Consider the task of computing something like the MAE between these two curves. Although they have similar data (V, P), they may not have the same number of V values or even the same V values. Hence, the curves cannot be directly compared, because some points in one curve might not even have a direct counterpart in the other. The proposed procedure, which can also be used to compare the performance of forecasting models of different origin, is thus the following:

1. Input: Begin with the two curves, (V_1, P_1) and (V_2, P_2).
2. Union: Compute and sort the union of V_1 and V_2 to form V_u.
3. Define Grid: Define a grid of U equispaced points that spans from $\min(V_u)$ to $\max(V_u)$. Label this grid as V_{common}.
4. Interpolate: Perform step interpolation for P_1 and P_2 over this same V_{common}, obtaining the corresponding interpolated values. They could be called Π_1 and Π_2, defined on the same grid, and which can be compared.
5. Compute the metrics: Calculate the Mean Absolute Error between Π_1 and Π_2 as follows:

$$\text{HC-MAE} = \frac{1}{U} \sum_{i=1}^{U} |\Pi_{1i} - \Pi_{2i}|. \tag{6.1}$$

   This simple error against ground truth will be considered the error of the forecast. It is just a Mean Absolute Error adapted to the curve forecasting problem, a problem for which a standard MAE cannot be used.

Noticeably, in this procedure the choice of U is crucial as it determines the resolution at which the two curves are compared. A high value of U gives a comparison similar to calculating the area between the curves, though it has a significant computational cost. Conversely, if U is too low, the error metric may fail to capture local differences between the curves, resulting in a less accurate representation of their dissimilarity. In the experiments for this study, U was taken as 10000, a number high enough to capture the local differences and low enough to not incur a high computational cost.

Chapter 7

Results and Discussion

7.1 1-day ahead forecasting

7.1.1 Benchmarks

For comparison, six models which we consider the most representative for this kind of studies were used as benchmarks. Their names are listed in the first row of Tab.7.1, placed on the top of the columns as labels. They are the naive model, the smarter naive model, the PCA plus LASSO approach, the PCA plus LSTM approach, the functional-stochastic FAR model, and the X-model. These models were discussed in the literature Section, Sec.2.1, except for the two naive and smarter naive models to be discussed hereafter. Notice that, as discussed in Sec.2.1, the X-model was not originally designed for curve forecasting, and the results presented here come from an adaptation extrapolated from the original paper.

In scalar electricity daily series autoregression forecasting of price or volume, an effective benchmark model, very difficult to beat and largely used, is the so-called 'naive' (or persistency) model [1]. Given the scalar data daily series $\{x_t\}$, the naive forecast issued at $t-1$ for today at t is obtained from the current value of the series, as $x^f_{t-1} = x_{t-1}$. The 'smarter naive' forecast is intended to improve the 'naive' forecast. It is given by $x^f_{t-1} = x_{t-7}$ if t is Saturday, Sunday or Monday, and $x^f_{t-1} = x_{t-1}$ in the other cases. This is because weekend day prices or volumes have a remarkably different behavior from other days and cannot be well predicted with the values of the other days type. If for example today is Saturday (a weekend day), yesterday's

(that is, Friday's, a working day) demand is not too much suitable to forecast it, whereas Saturday's demand from one week ago can be more suitable. The same happens for Sunday (weekend against working day) and Monday (working day against weekend). In the case of daily curve forecasting a similar approach can be taken. In this case we can define the naive model as

$$X^f_{t-1} = X_{t-1} \tag{7.1}$$

where X^f_{t-1} represents the forecast issued at $t-1$ for the today's time t, obtained as a copy of X_{t-1}, the data curve of yesterday. Notice that the ground truth X_t and its forecast X^f_{t-1} are most surely defined on different volume points, hence they cannot be directly compared point by point. This is, again, why we have to use a special measure like the HC-MAE interpolation procedure for defining a meaningful forecasting error. In the same way the smarter naive model can be defined as

$$X^f_{t-1} = \begin{cases} X_{t-7} & \text{if } t \in \text{Saturday, Sunday, Monday,} \\ X_{t-1} & \text{otherwise,} \end{cases} \tag{7.2}$$

using notation in analogy to the scalar case.

Besides these two standard benchmarks, the third benchmark is a slightly modified version of the 'PCA plus LASSO linear autoregression' model proposed by Pelagatti in Ref. [11]. Since that model was primarily proposed for supply curves, it is modified so that it can forecast demand curves too (because its monotonically decreasing, not increasing).

The fourth model is a modified version of the 'PCA + LSTM model' proposed by Guo et al. in Ref.[6]. Since that model forecasts primarily supply curves, and specifically from the PJM market, the data preprocessing procedure of Ref.[11] was applied before applying PCA and, subsequently, LSTM.

The fifth model is the FAR stochastic functional autoregression proposed by Shah and Lisi in Ref. [14]. In this case, linear regression was used over a uniform grid of 500 points, and then the *far* [35] package of R was used to forecast the curves directly.

The sixth model follows the X-model proposed by Ziel et al. in Ref. [13]. It should be however noted that a key assumption of the X-model is that there must be at least one bid within each price class. This makes the number of price classes significant in the model's operation. As to that, one should notice that the number of price classes which one can get from the

IPEX dataset restricted to the NORD zone is maybe too much small in the specific case of IPEX demand curves. This makes the applicability of the X-model somewhat limited.

7.1.2 Comparison

The results for demand curves are shown in Tab.7.1. According to the proposed HC-MAE metric, these results show that, already at first glance, the proposed models (BME and the autoencoder models) outperform the FAR in capturing the whole shape of the curve, also because FAR often significantly violates curve monotonicity. The proposed models also outperform the X-model, which in turn outperforms the PCA-based models by a significant amount. Moreover, the proposeds model gives better results than the naive and the smarter naive models (which never violate monotonicity). Of our proposed models, the BME model does not enforce monotonicity, but the autoencoder model strictly enforces monotonicity.

Hour	Naïve	Smarter Naïve	PCA + Lasso	PCA + LSTM	FAR	X-Model	BME: Pure Variant	BME: Combined Variant	Autoencoder Model
1	118.87	127.28	243.83	197.22	175.91	125.66	118.88	104.07	**70.79**
2	119.85	128.74	207.82	196.95	180.21	127.95	121.13	107.91	**71.48**
3	119.82	128.59	242.69	193.76	181.01	129.18	119.86	108.97	**72.62**
4	116.66	127.1	204.27	179.44	179.63	128.6	120.99	108.66	**74.93**
5	115.17	124.58	160.3	172.13	176.48	122.58	117.47	109.03	**75.17**
6	134.28	113.06	181.62	185.5	181.54	119.49	98.29	91.99	**76.51**
7	176.5	96.85	282.45	244.39	242.46	129.75	107.91	87.1	**77.79**
8	210.3	93.58	227.19	242.53	253.05	142.55	135.32	85.87	**75.47**
9	227.34	87.15	276.01	274.22	261.74	152.63	142.05	80.36	**75.91**
10	222.1	87.36	336.97	296.94	271.11	151.81	142.43	80.01	**74.94**
11	216.34	89.56	250.27	263.49	271.38	150.71	138.25	81.49	**74.59**
12	211.81	89.55	375.27	281.61	279.66	149.35	135.73	80.57	**79.52**
13	198	88.72	330.8	265.83	274.79	144.4	128.12	78.18	**77.1**
14	211.13	88.47	348.55	290.32	294.94	151.9	132.24	**78.18**	81.13
15	221.18	88.29	361.29	302.29	303.87	157.32	136.43	**79.55**	80.17
16	224.8	89.41	335.45	283.07	306.54	158.57	137.16	**80.16**	83.25
17	219.82	90.02	358.44	309.29	302.77	154.66	135.04	**81.55**	85.82
18	205.38	89.12	329.14	283.47	287.29	147.44	135.33	**83.42**	85.53
19	186.25	83.9	313.14	265.62	250.95	136.21	137.18	79.32	**76.23**
20	166.12	79.17	263.63	230.24	231.56	124.12	128.77	74.32	**73.7**
21	153.15	80.65	217.08	204.77	221.29	126.79	118.97	76.16	**72.9**
22	142.68	78.64	221.19	200.56	223.71	127.66	110.14	73.77	**73.68**
23	130.22	77.56	206.87	184.08	223.64	124.74	103.1	73.91	**69.84**
24	124.18	76.08	231.76	191.38	194.49	124.07	101.58	73.82	**69.76**

Table 7.1: HC-MAE value comparison of the various forecasting models for 24 hours for the demand curve.

For supply curves, the average HC-MAE results are shown in Tab.7.2.

Like in the demand curve case, the proposed models outperform, in order, FAR, the X-model, and both the PCA-based models. They also outperforms both the naive and the smarter naive models.

Hour	Naïve	Smarter Naïve	PCA + Lasso	PCA + LSTM	FAR	X-Model	BME: Pure Variant	BME: Combined Variant	Autoencoder Model
1	79.52	78.36	418.69	334.17	200.36	334.85	86.76	84.66	**73.5**
2	77.64	76.28	485.53	381.47	200.98	339.59	84.31	82.16	**72.15**
3	77.21	76.38	504.24	366.04	201.43	339.44	84.59	81.91	**69.85**
4	74.73	75.49	497.56	384.29	200.88	337.76	82.26	79.91	**68.46**
5	69.91	73.13	404.97	299.07	202.21	340.77	73.39	71.37	**64.96**
6	64.87	70.63	389.35	317.93	203.14	341.1	69.45	67.3	**63.17**
7	58.11	66.09	447.41	323.09	202.58	335.38	61.27	59.5	**57.36**
8	71.82	68.55	450.68	308.55	205.07	313.12	68.75	63.89	**61.67**
9	80.88	69.2	362.35	231.05	203.4	306.98	73.47	67.67	**65.94**
10	80.83	71.7	389.79	232.69	205.18	303.11	73.94	69.65	**66.22**
11	81.09	75.24	403.11	258.98	203.73	303.79	73.61	**69.63**	70.02
12	80.58	79.04	398.49	267.09	199.13	303.4	74.28	69.7	**69.68**
13	76.46	78.89	353.86	245.77	203.16	304.5	75.16	**71.27**	71.91
14	77.13	79.65	396.76	264.41	203.97	307.06	75.07	**71.1**	71.87
15	75.94	77.46	358.18	245.54	206.58	305.8	73.22	68.43	**68.06**
16	73.19	70.61	356.05	242.51	204.96	305.06	71.03	65.43	**62.91**
17	71.99	63.75	370.88	261.06	208.7	310.31	71.46	65.8	**57.88**
18	69.96	62.78	384.23	265.58	211.56	307.41	67.51	61.59	**55.84**
19	66.82	60.4	373.64	222.43	204.82	305.66	64.16	58.52	**55.61**
20	61.28	60.75	390.41	233.1	205.38	312.1	61.97	57.22	**53.75**
21	60.42	62.29	372.78	204.38	204.06	309.63	63.45	58.8	**57.37**
22	54.37	62.38	349.06	166.55	207.37	313.5	60.2	55.29	**53.12**
23	53.55	66.88	423.15	279.78	205.27	326.13	61.69	56.91	**50.9**
24	53.05	65.01	382.99	273.1	204.07	326.18	61.28	57.27	**51.36**

Table 7.2: HC-MAE value comparison of various forecasting models over the 24 hours for the supply curve.

For demand curves, the smarter naive model performs comparable with that of the naive model for the night hours 1 to 6. For daylight hours from 7 to 24, the smarter naive model outperforms the naive model, especially for hours 7 to 18, which are typical working hours. This shows that the seasonal nature of the smarter naive model allows it to account for the sudden drop in demand from Fridays to the weekends and the sudden increase in demand from the weekends to Mondays. In addition, this highlights the general importance of keeping into account at least lag 7 in modeling these data. The PCA + LASSO and PCA + LSTM models do not perform well in this case, for all hours. Our proposed models outperforms all the benchmark models significantly for all the hours.

For supply curves, the naive and smarter naive models have comparable performance for most hours, with each model outperforming the other in some hours. This is because the weekly pattern is much weaker in the supply

curves. After all, most of the stochasticity in supply comes from renewable energy such as solar or wind energy, which is not linked to human weekly cycles. Even in the case of supply curves, the proposed model outperforms all benchmarks.

In order to better perceive where forecasting errors are generated, Fig.(7.1) shows a sample of the forecast demand curves for a choice of representative hours for the specific day of the 15th July, 2019, mainly focusing on the proposed model. In order not to clutter the figures too much, the forecasts of the proposed models were compared with only the naive and smarter naive models (the closest competitors to the proposed models).

The smarter naive model is, as anticipated, very difficult to beat, but the proposed model competes, even in this specific difficult day, head to head with it, whereas all other models lag. From the Figure, it is clear that the most difficult structure to forecast for all models is the beginning and ending points (that is, the special points) of the interesting segment. Missing their positions automatically generates large errors. This visual comparison should hence make more clearly compelling one of the main reasons of the presented approach. With the same spirit, Fig.(7.2) shows a sample of the forecasted supply curves for 15th July, 2019 for the same selected hours.

7.1.3 The Interesting Segment

Figs. (7.3) and (7.4) show comparisons of the forecasted interesting segment of the curve for demand and supply curves respectively, with the ground truth curves for hour 11 of 15th July, 2019. These Figures show that the BME model does not enforce monotonicity and in fact, produces a curve that is locally non-monotonic, but with an overall monotonic trend. On the other hand, the autoencoder model produces forecasted curves that are indeed monotonic in nature.

Finally, the monotonic autoencoder compresses demand or supply curve data more effectively than PCA. This suggests that the curves have characteristics which are complex. Unlike PCA, which relies on linear combinations to capture variance, autoencoders can learn patterns and dependencies that aren't purely additive or proportional. This difference indicates that the demand and supply curves are likely influenced by non-linear interactions.

This improved compression also reveals intricate, non-redundant details in the curve that PCA may overlook. For example, demand and supply curves in electricity markets often exhibit subtle variations — small shifts

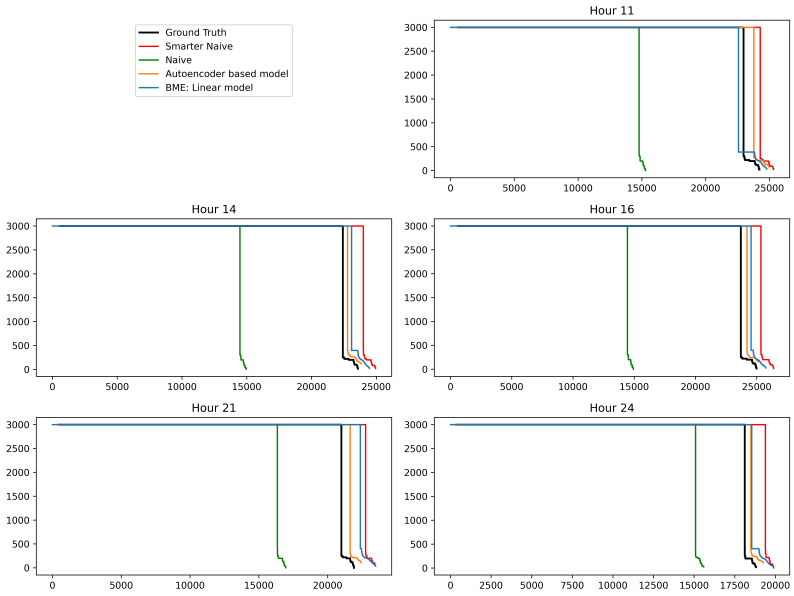

Figure 7.1: Visual comparison of the forecasts for the demand curves on 15th July, 2019, for selected hours.

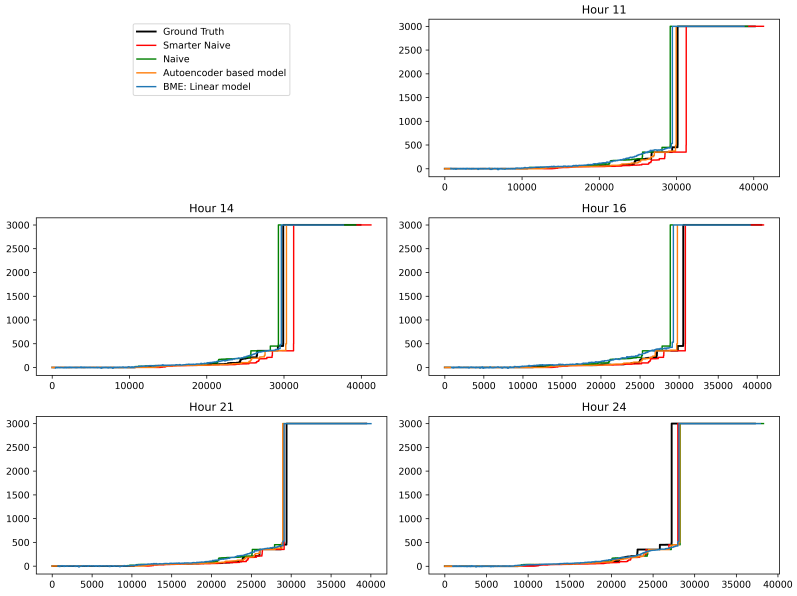

Figure 7.2: Visual comparison of the forecasts for the supply curves on 15th July, 2019, for selected hours.

due to behavioral changes, seasonal effects, or gradual trends — that require a non-linear model to be fully captured. The autoencoder's learned representation isolates these critical features, providing a clear view of the core dynamics. This not only highlights the essential features within the curve but also suggests that the data's structure is more intricate than what linear techniques alone can reveal.

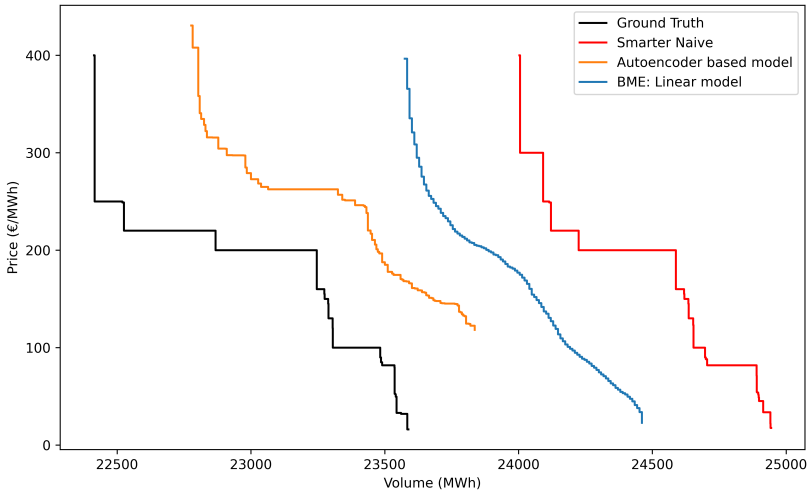

Figure 7.3: Comparison of the forecast with the true curve for demand curves.

7.1.4 Discussion (horizon 1 day)

The performance of the benchmark models reveals some insight into the models themselves. The PCA+Linear model, the PCA+LSTM model, and the X-model, perform significantly worse than the naive and smarter naive models. This might be attributed to the fact that these models rely on approximating the curve using a limited number of points. Specifically, the PCA models use percentiles based on bid prices, and the X-model creates price classes based on the mean bid curve. As a result, the information in the curve is cut down to mostly around the interesting segment of the curve. The information lost from the rest of the curve is never retrieved. While the FAR model does consider the entire curve, it is hard to include exogenous

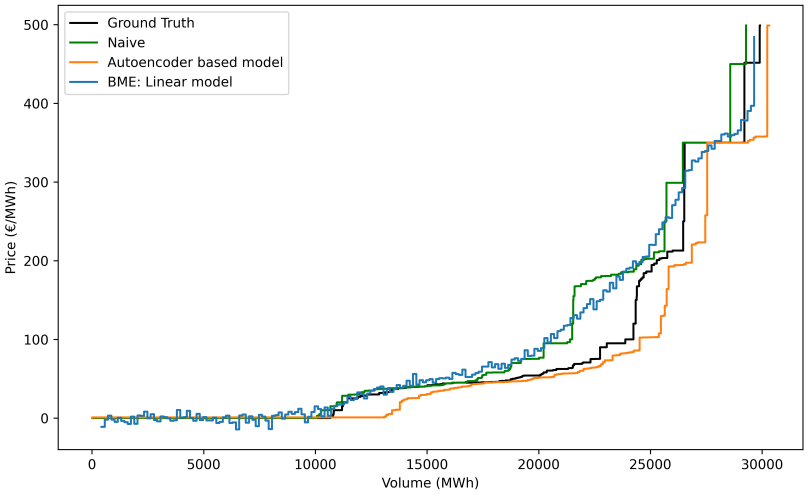

Figure 7.4: Comparison of the forecast with the true curve for supply curves.

information in the FAR scheme. In addition, in this scheme the forecasting model is linear. In comparison to these models, the proposed models keep the curve information outside of the interesting segment, and the forecasting part is independent from the curve modeling part, hence, any forecasting model can be used after the modeling of the curve.

In all, the proposed models outperform all the benchmark models significantly. As said, a major reason for this good performance comes from the fact that the models are specifically designed for tracking the box where the interesting segment lives. The BME model does it through 3 points by assuming linear interpolation between bids, and the autoencoder model does it through 2 points by assuming step interpolation between bids. However, it is also important to highlight a second possible reason. Unlike the models already present in the literature, which either use V as the support and predict P, or use P as the support and predict V, the proposed models have components for predicting both V and P, and that can be another advantage in terms of efficiently using available information. In fact, in the first step of the BME model, the three V values, namely $V_{\mathbf{B}}$, $V_{\mathbf{M}}$, and $V_{\mathbf{E}}$, are forecast. While in the autoencoder model, the two V values, namely $V_{\mathbf{A}}$ and $V_{\mathbf{B}}$, are forecast. The first steps in the two models are equivalent and only differ

in the interpolation scheme assumed. It essentially forecasts the interesting segment of the individual curve. The second step forecasts the P values by forecasting the interesting segment, as described in both models. This feature may be critical in achieving the better performance of the proposed model.

7.2 7-day ahead forecasting

7.2.1 Benchmarks

Most works in literature have not considered the case of a 7-day forecast for curve forecasting. To the best of our knowledge, only Ref. [14] has considered this possibility. The benchmarks that we have considered are the naive model, X-model [13], and FAR [14].

These models were adjusted to provide 7 day ahead forecasts. Since Ref. [14] uses the R package *far*, we used the horizon value as 7 in the *predict* function of this library. In our problem setting, the naive model is defined as:

$$\hat{P}(V)_{d,h} = P(V)_{d-7,h} \tag{7.3}$$

where $P(V)_{d,h}$ is the supply curve on day d at hour h.

7.2.2 Comparison

Table 7.3 shows the HCMAE results for all the models by themselves for supply curves. The sp-VAR and the sp-FFN are the models based on the I-splines. It shows that the spline based models outperform the other benchmark models other than the naive model. The naive model performs much better than all other models. This shows that there is not much temporal variability over a 7-day horizon.

Table 7.4 shows the HCMAE results for all the models by themselves for demand curves. Here, there is a different trend. Among the individual models, the X-model outperforms the other models, but the naive model is still the best among all the models.

Hour	Naïve	FAR	X-Model	sp-VAR	sp-FFN
1	94.84	350.03	386.2	**161.73**	176.08
2	94.05	358.85	391.02	183.19	**154.65**
3	95.08	359.67	391.2	**151.25**	162.35
4	96.05	358.51	394.13	172.6	**169.62**
5	96.45	358.17	398.96	190.15	**165.55**
6	94.76	357.3	398.55	206.48	**179.21**
7	91.72	346.14	382.09	200.13	**156.65**
8	95.84	344.61	367.95	193.41	**165.75**
9	98.18	345.87	374.98	294.47	**197.32**
10	101.15	345.45	372.73	253.45	**210.31**
11	108.14	346.27	368.13	274.51	**206.74**
12	114.83	348.36	368.3	487.23	**230.22**
13	115.15	347.63	377.98	373.25	**220.97**
14	116.08	350.43	383.54	286.72	**218.96**
15	113.3	348.92	388.33	311.61	**218.75**
16	100.24	347.55	389.7	262.48	**189.69**
17	84.68	341.81	377.66	273.18	**184.72**
18	88.11	335.47	363.63	222.74	**177.56**
19	88.09	332.76	354.29	229.11	**165.45**
20	87.26	330.9	357.8	222.49	**166.93**
21	87.77	329.87	358.66	194.62	**141.48**
22	86.94	348.14	371.25	199.72	**156.49**
23	85.46	341.82	381.59	164.48	**162.7**
24	88.38	348.09	389.69	180.9	**151.07**

Table 7.3: HCMAE comparison of individual models for supply

Hour	Naïve	FAR	X-Model	sp-VAR	sp-FFN
1	79.07	150.23	**82.08**	149.04	104.0
2	82.64	160.82	**84.59**	154.68	108.95
3	82.31	164.87	**85.16**	158.87	106.64
4	85.75	167.73	**88.5**	161.76	110.9
5	91.29	168.29	**94.07**	165.81	118.38
6	90.36	175.46	**97.28**	176.2	116.2
7	91.42	191.9	**105.42**	177.1	120.63
8	100.4	195.97	**122.72**	182.96	143.32
9	92.22	192.31	**110.9**	175.89	132.47
10	96.4	196.35	**110.15**	180.41	151.87
11	98.3	201.43	**108.98**	172.05	165.02
12	97.02	215.85	**106.5**	163.74	129.46
13	96.37	220.16	**104.61**	161.87	122.02
14	96.49	236.28	**103.37**	172.16	132.52
15	96.25	244.56	**106.9**	168.45	125.46
16	98.14	247.54	**111.56**	165.22	129.95
17	101.85	253.82	**123.83**	164.42	131.04
18	108.19	264.43	**138.48**	178.33	242.12
19	95.13	237.66	**112.16**	183.89	145.87
20	82.72	200.51	**92.11**	151.1	111.09
21	85.72	191.2	**86.56**	148.83	109.28
22	86.3	184.89	**88.61**	148.17	110.58
23	85.07	161.78	**88.63**	147.19	102.56
24	88.55	156.74	**88.58**	146.3	99.35

Table 7.4: HCMAE comparison of individual models for demand

Table 7.5 shows the HCMAE results for all the models combined with the naive model for supply curves. While all the combined models outperform the naive model for all hours, the combined models of the spline-based models outperform the others over most hours.

A similar trend can be seen for the demand curves in Table 7.6. This shows that combination of two models can improve the performance over the two models individually. This shows the strength of our combining/ensembling approach.

Hour	**Naïve**	**FAR + Naïve**	**X-Model + Naïve**	**sp-VAR + Naïve**	**sp-FFN + Naïve**
1	94.84	88.63	90.02	**86.19**	87.01
2	94.05	87.41	86.49	84.57	**84.02**
3	95.08	87.29	86.92	87.99	**85.85**
4	96.05	**87.37**	88	88.26	90.03
5	96.45	**88.49**	91.56	89.23	90.9
6	94.76	**86.51**	89.3	88.31	89.85
7	91.72	89.62	**84.33**	87.78	88.64
8	95.84	88.61	89.59	87.99	**86.24**
9	98.18	87.67	90.55	**84.14**	90.15
10	101.15	92.79	93.24	**88.93**	96
11	108.14	**96.54**	101.46	103.18	97.09
12	114.83	108.62	107.39	**101.48**	103.16
13	115.15	102.57	108.01	103.58	**101.72**
14	116.08	105.13	109.14	105.19	**103.48**
15	113.3	105.43	106.16	103.4	**102.76**
16	100.24	94.9	93.06	92.36	**89.13**
17	84.68	83.66	79.23	**78.36**	79.85
18	88.11	82.04	83.47	85.43	**81.94**
19	88.09	83.66	82.93	80.35	**79.49**
20	87.26	83.44	84.04	**79.88**	82.15
21	87.77	82.55	83.73	83.62	**78.93**
22	86.94	82.6	82.19	**78.29**	80.18
23	85.46	83.39	81.26	**78.58**	81.02
24	88.38	87.82	**84.83**	87.54	85.75

Table 7.5: HCMAE comparison of the combined models for Supply Curves

Hour	**Naïve**	**FAR + Naïve**	**X-Model + Naïve**	**sp-VAR + Naïve**	**sp-FFN + Naïve**
1	79.07	79.46	79.93	**78.68**	78.85
2	82.64	82.15	81.7	**81.56**	81.82
3	82.31	82.69	81.84	81.61	**81.4**
4	85.75	83.88	84.63	**83.34**	83.82
5	91.29	88.86	89.4	**88.25**	88.32
6	90.36	88.07	88.71	87.82	**87.73**
7	91.42	89.69	90.17	**89.29**	89.89
8	100.4	97.63	97.71	**96.76**	97.15
9	92.22	90.11	90.97	**89.77**	89.95
10	96.4	92.72	94.12	**91.97**	92.58
11	98.3	92.76	94.02	92.21	**92.17**
12	97.02	91.15	92.02	**91.12**	91.35
13	96.37	90.24	90.44	**88.66**	89.03
14	96.49	87.23	87.41	**86.86**	87.37
15	96.25	89.17	89.61	**88.79**	88.88
16	98.14	91.26	91.35	**90.34**	90.53
17	101.85	96.21	96.97	**95.25**	96.14
18	108.19	**103.83**	104.05	104.5	104.58
19	95.13	91.04	91.46	90.79	**90.73**
20	82.72	83.17	83.01	82.03	**81.91**
21	85.72	85.99	**85.64**	85.88	85.84
22	86.3	85.1	85.66	85.28	**84.86**
23	85.07	84.73	85.29	84.9	**84.56**
24	88.55	86.88	86.31	86.24	**85.83**

Table 7.6: HCMAE comparison of the combined models for Demand Curves

The non-linear forecasting model (MLP-based) demonstrates better forecasting capability during dynamic market conditions, likely due to its ability to capture complex interactions over the 7-day horizon. On the other hand, the linear forecasting model (VAR-based) performs well in more predictable market conditions. In fact, when the naive model (highly linear) and the

MLP (highly non-linear) model are combined, it leads to a powerful combination that can tackle both linear and non-linear temporal dependencies.

In order to better perceive where forecasting errors are generated, Fig.(7.5) shows a sample of the forecast demand curves for a choice of representative hours for the specific day of the 27th October, 2019. All the combined models are shown along with the naive forecast. It shows that other than hour 11, over all the other hours shown, the naive model performs worse than the combined models.

With the same spirit, Fig.(7.6) shows a sample of the forecast supply curves for 27th October, 2019 for the same selected hours.

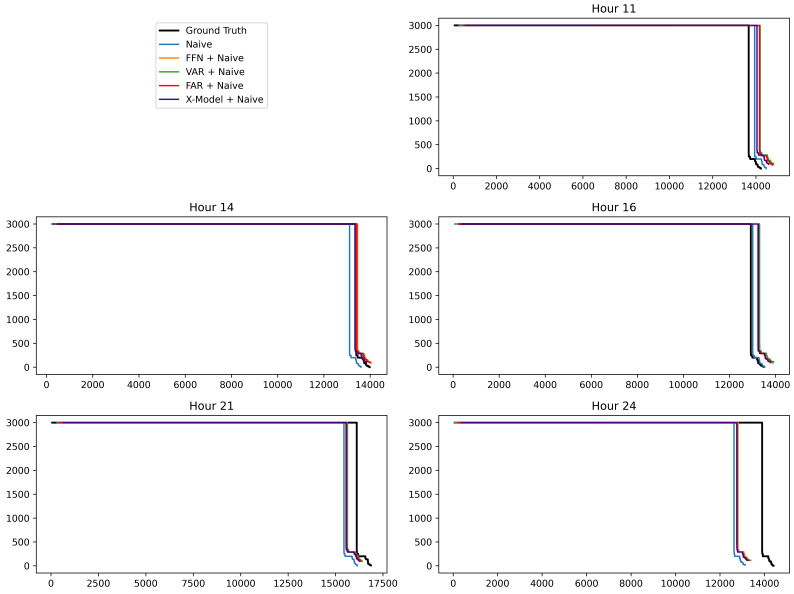

Figure 7.5: Visual comparison of the forecasts for demand curves on 27th October, 2019, for selected hours.

Figs. (7.7) and (7.8) show comparisons of the interesting segment of the forecasted curves for demand and supply respectively. These images show very clearly that the forecasted curves are clearly better than the naive model in the interesting segment of the curves. The better performance is in both the P domain and the V domain.

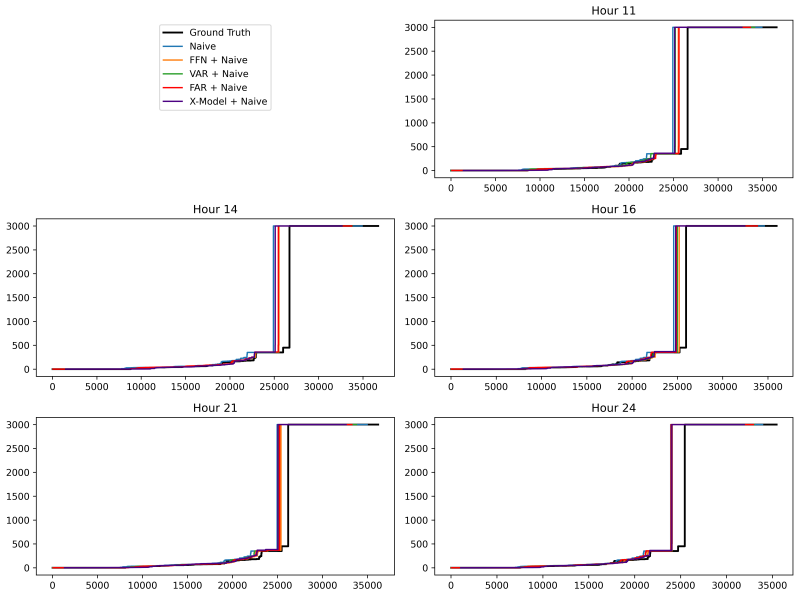

Figure 7.6: Visual comparison of the forecasts for supply curves on 27th October, 2019, for selected hours.

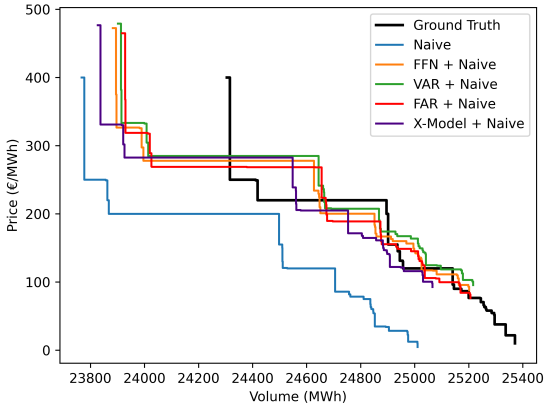

Figure 7.7: Comparison of the forecast with the true curve for demand curves.

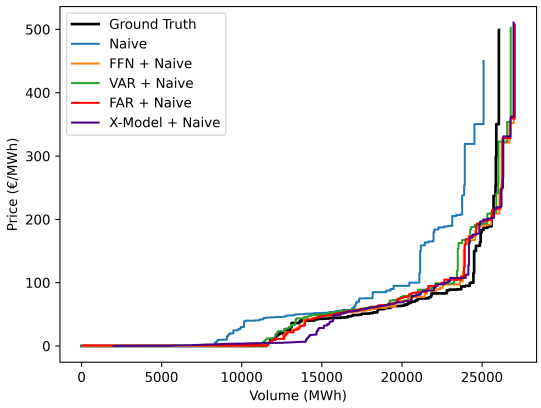

Figure 7.8: Comparison of the forecast with the true curve for supply curves.

7.2.3 Discussion (horizon 7 days)

From Sec. 7.2.2, it can be seen that the naive model outperforms the rest of the models by a significant margin. Figs. 7.7 and 7.8 show that while the shape of the curve is important, the placement of the curve in the V plane is a major factor for curve dissimilarity. For the supply curves, the sp-FFN model performed the best out of all the individual models (except for the naive model). This shows that the spline method has some merit in capturing the shape of the curves. Note that the FAR model does not take monotonicity into consideration.

For demand curves, the X-model seems to outperform the other models. This is because it forecasts V values directly unlike the other models which forecast P values. This is because demand curves typically have a characteristic b̈ig steps̈tructure that mainly shifts along the V axis. This can be well-captured by a model focusing on V values.

The combined model results clearly show that the combined models are superior to the naive model, even if the constituent models are significantly worse by themselves. Owing to the simple linear structure of the combination model, it is scalable and more than 2 models can also be included in the combination. Note that the X-model forecasts P values and the others forecast V values. Hence, the proposed combination model can combine both kinds of models. It is theoretically possible to combine a model that forecasts V values and a model that forecasts P values. This can even allow us to take

advantage of both V forecasting and P forecasting.

Chapter 8

Conclusions

This thesis tried to make several contributions to forecasting day-ahead electricity market demand and supply curves, particularly addressing the unique challenges posed by this kind of curves, namely curve heterogeneity and monotonicity constraints. We also tackled the additional constraint of 7-day ahead forecasting.

First, we introduced a novel modular forecasting framework named as the BME-model. This approach tokenizes market curves into three structurally significant points—**B**, **M**, and **E**—and identifies the most important curve segment. These elements are forecasted separately and subsequently recombined to reconstruct the entire curve. We propose two models, namely the pure variant (linear) and the combined variant. Results show that both BME model effectively manages curve heterogeneity, with the combined variant showing improved forecasting accuracy.

Second, we extended our methodologies to non-linear dimensionality reduction by introducing a monotonic autoencoder-based model. In contrast to the BME model, this approach utilizes step interpolation rather than linear interpolation. The significant curve segment is parameterized through a monotonic autoencoder before forecasting, enhancing the model's ability to capture complex curve structures.

In the non-linear extension, we proposed an Enhanced Echo State Network, a specialized variant designed explicitly for forecasting the identified structural points and autoencoder encodings. This neural network demonstrates improved prediction capabilities, particularly in capturing intricate temporal dynamics.

To integrate forecasts effectively, we proposed an 'integration model', sim-

ilar to the combined variant of the BME model, that systematically reconstructs the complete curve from separately forecasted components.

We also introduced the Heterogeneous Curve Mean Absolute Error (HC-MAE), a novel metric specifically tailored for evaluating the dissimilarity between heterogeneous curves across the full domain, ensuring robust and unbiased performance assessment. All comparisons and optimizations in this thesis have been done on the basis of this metric.

As last, we also introduced an I-Spline based model that does a functional parameterization of these curves while maintaining their monotonic nature. This is done using I-Splines. The curves, parameterized as I-Spline coefficients, are forecasted using VAR and feedforward networks. However, recognizing the practical constraints of data availability in markets such as the Italian Power Exchange (IPEX), we designed a robust seven-day-ahead forecasting combination scheme where the spline-based models are key components. This simple yet effective method successfully merges forecasts from widely different models, consistently yielding better performance compared to individual forecasts.

Results highlight that explicitly addressing curve heterogeneity and monotonicity significantly enhances forecasting accuracy. The combined model approaches consistently outperform single-model forecasts, emphasizing the value of leveraging multiple predictive strategies. Additionally, non-linear approaches, such as monotonic autoencoders, show great potential in accurately capturing complex temporal patterns in such types of market curves.

Bibliography

[1] R. Weron, Electricity price forecasting: A review of the state-of-the-art with a look into the future, International journal of forecasting 30 (4) (2014) 1030–1081.

[2] F. Ziel, R. Weron, Day-ahead electricity price forecasting with high-dimensional structures: Univariate vs. multivariate modeling frameworks, Energy Economics 70 (2018) 396–420.

[3] P. Schwarz, Energy Economics, Routledge, 2022.

[4] M. Kazemi, H. Zareipour, M. Ehsan, W. D. Rosehart, A robust linear approach for offering strategy of a hybrid electric energy company, IEEE Transactions on Power Systems 32 (3) (2016) 1949–1959.

[5] R. Chen, I. C. Paschalidis, M. C. Caramanis, P. Andrianesis, Learning from past bids to participate strategically in day-ahead electricity markets, IEEE Transactions on Smart Grid 10 (5) (2019) 5794–5806.

[6] H. Guo, Q. Chen, K. Zheng, Q. Xia, C. Kang, Forecast aggregated supply curves in power markets based on lstm model, IEEE Transactions on Power Systems 36 (6) (2021) 5767–5779.

[7] H. Kazmi, Z. Tao, How good are tso load and renewable generation forecasts: Learning curves, challenges, and the road ahead, Applied Energy 323 (2022) 119565.

[8] M. Pinhão, M. Fonseca, R. Covas, Electricity spot price forecast by modelling supply and demand curve, Mathematics 10 (12) (2022) 2012.

[9] S. Yıldırım, M. Khalafi, T. Güzel, H. Satık, M. Yılmaz, Supply curves in electricity markets: a framework for dynamic modeling and monte

carlo forecasting, IEEE Transactions on Power Systems 38 (2022) 3056 – 3069.

[10] GME, Italian 'gestore del mercato elettrico' website.
URL https://www.mercatoelettrico.org/En/download/DownloadDati.aspx?val=OfferteFree_Pubbliche

[11] M. Pelagatti, Supply function prediction in electricity auctions, in: Complex models and computational methods in statistics, Springer, 2012, pp. 203–213.

[12] G. Mestre, J. Portela, A. M. San Roque, E. Alonso, Forecasting hourly supply curves in the italian day-ahead electricity market with a double-seasonal sarmahx model, International Journal of Electrical Power & Energy Systems 121 (2020) 106083.

[13] F. Ziel, R. Steinert, Electricity price forecasting using sale and purchase curves: The x-model, Energy Economics 59 (2016) 435–454.

[14] I. Shah, F. Lisi, Forecasting of electricity price through a functional prediction of sale and purchase curves, Journal of Forecasting 39 (2) (2020) 242–259.

[15] A. Ciarreta, B. Martinez, S. Nasirov, Forecasting electricity prices using bid data, International Journal of Forecasting 39 (3) (2023) 1253–1271.

[16] Q. Tang, H. Guo, K. Zheng, Q. Chen, Forecasting individual bids in real electricity markets through machine learning framework, Applied Energy 363 (2024) 123053.

[17] F. Ziel, R. Steinert, Probabilistic mid-and long-term electricity price forecasting, Renewable and Sustainable Energy Reviews 94 (2018) 251–266.

[18] S. Haben, J. Caudron, J. Verma, Probabilistic day-ahead wholesale price forecast: A case study in great britain, Forecasting 3(3) (2021) 596–632.

[19] P. Ghelasi, Z. Florian, Hierarchical forecasting for aggregated curves with an application to day-ahead electricity price auctions, International Journal of Forecasting 40(2) (2024) 581–596.

[20] Z. Li, A. M. Alonso, A. Elías, J. M. Morales, Clustering and forecasting of day-ahead electricity supply curves using a market-based distance, International Journal of Electrical Power & Energy Systems 158 (2024) 109977.

[21] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society Series B: Statistical Methodology 58 (1) (1996) 267–288.

[22] D. Srinivasan, C. Chang, A. Liew, Demand forecasting using fuzzy neural computation, with special emphasis on weekend and public holiday forecasting, IEEE Transactions on Power Systems 10 (4) (1995) 1897–1903.

[23] T. Hodge, Hourly electricity consumption varies throughout the day and across seasons, US Energy Information Administration 21 (2020).

[24] J. Nowotarski, E. Raviv, S. Trueck, W. Rafał, An empirical comparison of alternative schemes for combining electricity spot price forecasts, Energy Economics 46 (1) (2014) 395–412.

[25] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, Journal of global optimization 11 (1997) 341–359.

[26] R. Weron, Modeling and forecasting electricity loads and prices: A statistical approach, John Wiley & Sons, 2006.

[27] R. J. Hyndman, G. Athanasopoulos, Forecasting: principles and practice, OTexts, 2018.

[28] R. Weron, F. Ziel, Electricity price forecasting, in: Routledge handbook of energy economics, Routledge, 2019, pp. 506–521.

[29] H. Hu, L. Wang, S.-X. Lv, Forecasting energy consumption and wind power generation using deep echo state network, Renewable Energy 154 (2020) 598–613.

[30] D. Runje, S. M. Shankaranarayana, Constrained monotonic neural networks, Proceedings of the 40th International Conference on Machine Learning 2023, PMLR 202:29338-29353 (2023).

[31] S. Lang, Introduction to linear algebra, Springer Science & Business Media, 2012.

[32] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Conference on Computer Vision and Pattern Recognition, doi:10.1109/CVPR.2016.90, and also in arXiv:1512.03385 (2016).

[33] J. Nowotarski, R. Weron, To combine or not to combine? recent trends in electricity price forecasting, Tech. rep., Hugo Steinhaus Center, Wroclaw University of Science and Technology (2016).

[34] GME/download, download data from gme. (Accessed on Jan. 2024).
URL https://www.mercatoelettrico.org/en-us/Home/Results/Electricity/MGP/Download/Download?valore=OffertePubbliche

[35] D. J. G. Serge, far: Modelization for Functional AutoRegressive Processes, r package version 0.6-6 (2022).
URL https://CRAN.R-project.org/package=far