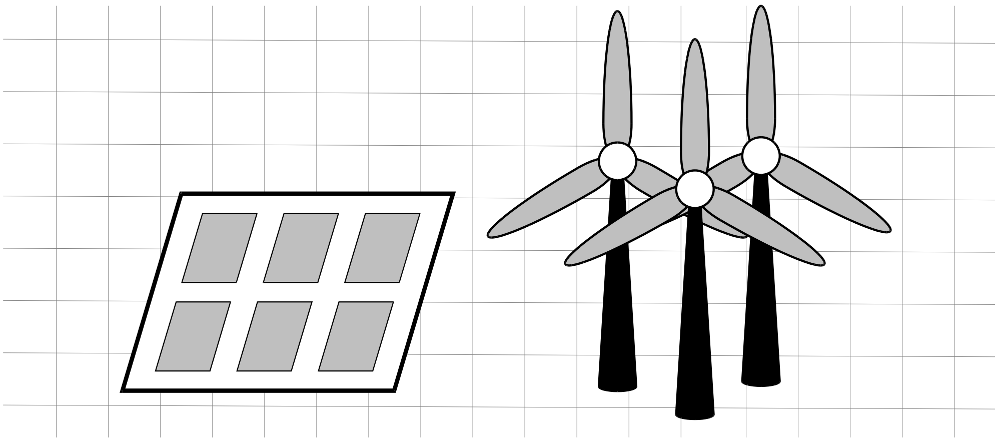


Renewable Energy Forecasting – Extreme Quantiles, Data Privacy and Monetization

Carla Sofia da Silva Gonçalves

Programa Doutoral em Matemática Aplicada
Matemática Aplicada
2021

Orientador
João Manuel Portela da Gama
Professor Catedrático
Faculdade de Economia da Universidade do Porto

Coorientador
Ricardo Jorge Gomes de Sousa Bento Bessa
Coordenador do Centro de Sistemas de Energia
Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência

Faculty of Sciences, University of Porto

U. PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Renewable Energy Forecasting – Extreme Quantiles, Data Privacy and Monetization

Carla Gonçalves

Universidade do Minho

universidade de aveiro

U. PORTO

MAP-PDMA: Doctoral Programme in Applied Mathematics
University of Minho, University of Aveiro, University of Porto

Supervisors: Doctor João Gama
Doctor Ricardo Jorge Bessa

July 14, 2021

Jury President

- Doctor Sofia Balbina Santos Dias de Castor Gothen (University of Porto)

Jury Members

- Doctor Jethro Browell (University of Strathclyde)
- Doctor Yannig Goude (EDF R&D, Université Paris-Sud)
- Doctor Maria Eduarda Silva (University of Porto)
- Doctor Carlos Ferreira (University of Aveiro)
- Doctor Ricardo Jorge Bessa(Supervisor)

Carla Gonçalves: *Renewable Energy Forecasting – Extreme Quantiles, Data Privacy and Monetization*, Doctoral Thesis, Doctoral Program in Applied Mathematics of the Universities of Minho, Aveiro and Porto, July 2021.

Google Scholar: https://scholar.google.com/citations?user=8hwqwn4AAAAJ
ORCID: https://orcid.org/0000-0001-5684-9933
E-mail: carla.s.goncalv@gmail.com

Acknowledgments

Coming to the end of these four years, I would like to express my gratitude to all those who have contributed in some way to conclude this work.

First of all, I would like to thank my supervisors Professor João Gama and Professor Ricardo Bessa for their invaluable guidance and availability. Professor Ricardo Bessa played a key role in the development of this thesis, and his global perspective on academic and industrial research, as well as critical opinions, were essential to identify research gaps and overcome the difficulties that have arisen throughout this work.

I would like to thank all members of the Center for Power and Energy Systems (CPES) at the Institute for Systems and Computer Engineering, Technology and Science (INESC TEC), for creating an excellent environment during these last years.

Additionally, I would also like to thank Professor Pierre Pinson and all members of the Energy Analytics and Markets (ELMA) group, at the Center for Electric Power and Energy of the Technical University of Denmark (DTU), for their hospitality and the excellent research environment they provided during my 6 months external stay. Professor Pierre Pinson's guidance went far beyond this stay.

To Professor Margarida Brito and Laura Cavalcante, thanks for the discussions about extreme value theory, which has contributed immensely to the first chapter.

On a personal level, I would like to thank a group of people who made this work a reality:

To my parents, a special thanks for the effort to give me a better life, despite my early arrival in their lives, always guiding and supporting my decisions. To my sister Francisca, thank you for your support. I could not miss mentioning the family's best friend, *Lucky*.

Thanks also to my friends, Patrícia Dias, Susana Pinto, Ana Margarida, Filipe Oliveira, Luís Baía, Júlio Silva, Renato Fernandes and Marisa Reis – all of whom researchers or students that touched my life one way or another.

Last but not least, a big thanks to my beloved Ricardo Cruz, for all the discussions, patience, support, and affection. I am proud of what we achieved together, and I hope the best is yet to come!

Regarding institutional support, I would like to thank:

FCT for providing me the Ph.D. grant "PD/BD/128189/2016" supported by POCH and the EU.

INESC TEC (through CPES) for providing me with the physical space where I performed the majority of my work – this included a workstation and meeting rooms to discuss ideas with my supervisors and colleagues. INESC TEC also provided me with health insurance and valuable IT services.

DTU for providing me a physical space during my external stay. DTU Elektro also provided me access to its HPC cluster, and summer schools in 2019.

Universities of Minho, Aveiro, and Porto for selecting me for the FCT grant and hosting our yearly PhD symposiums which allowed me to follow the interesting work of my PhD colleagues. During my first year, I had classes at the University of Minho, that provided me also a space to work.

Abstract

The growing integration of Renewable Energy Sources (RES) brings new challenges to system operators and market players due to its dependence on the whims of the weather. Consequently, accurate forecasts are essential to reduce electrical energy imbalances in the electricity market and design advanced decision-aid tools to support the integration of large amounts of RES into the power system.

The first challenge tackled by the thesis is related to the necessity of accurate modeling of extreme quantiles (tails of the probability distribution), since it is paramount to accurately model the risk (i.e., avoid over and underestimation of risk metrics like value-at-risk).

The second and third challenges relate to the fact that geographically distributed wind turbines, photovoltaic panels, and sensors produce large volumes of data that can be used to improve RES forecasting skill – for instance, by benefiting from spatio-temporal dependencies. These dependencies are a consequence of the weather patterns: the weather at a certain site is related to its historical values, and the weather variables at multiple sites within a certain geographic scale are not statistically independent; instead, they have spatial correlations with others.

Due to business competitive factors and personal data protection concerns, data owners (or agents) might be unwilling to share their data, despite the potential benefits. Two important properties (that represent our second and third challenges) are required to motivate the agents to perform collaborative forecasting models: (1) that data privacy is preserved during the collaboration, and (2) that data owners are not allowed to free-ride on others and are compensated for the data they contribute (data monetization).

The main contributions from this thesis are: (i) the development of a conditional extreme quantile forecasting model, that combines extreme value theory estimators for truncated generalized generalized Pareto distribution with non-parametric methods, conditioned by spatio-temporal information; (ii) a numerical and mathematical analysis of the existing privacy-preserving regression models and identification of weaknesses in the current literature; (iii) the development of a privacy-preserving forecasting algorithm for vector autoregressive models, that protects data by combining linear algebra transformations with a decomposition-based algorithm; and (iv) the development of an algorithmic solution for data monetization in RES collaborative forecasting, in which agents buy forecasts from a trusted entity instead of directly buying sensible data.

Along the thesis, the proposed models are empirically evaluated using synthetic data (sampled from autoregressive processes), and real publicly available datasets (from wind and solar energy power plants).

Keywords: Renewable Energy; Forecasting; Conditional Extreme Quantiles; Data Privacy; Data Monetization; Collaborative Forecasting.

Resumo

A integração em larga escala de energia produzida por fontes renováveis representa novos desafios para os operadores e participantes do mercado elétrico devido à sua dependência dos caprichos do clima. Consequentemente, o desenvolvimento de modelos de previsão é essencial para reduzir desvios na produção e projetar ferramentas de auxílio à decisão para apoiar a integração de grandes quantidades de energias renováveis no sistema elétrico.

O primeiro desafio abordado relaciona-se com a necessidade de modelar quantis extremos (caudas da distribuição de probabilidade), uma vez que é fundamental modelar o risco com precisão (ou seja, evitar sobre- e sub-estimação de métricas de risco como o *Value at Risk*).

O segundo e terceiro desafios relacionam-se com o facto de grandes volumes de dados serem produzidos por turbinas eólicas, painéis fotovoltaicos e sensores geograficamente distribuídos, podendo ser usados para melhorar a habilidade de previsão das energias renováveis – por exemplo, beneficiando de dependências espaço-temporais. Essas dependências são uma consequência dos padrões climáticos: o clima num determinado local está relacionado com os seus valores históricos e as variáveis climáticas em vários locais dentro de uma determinada região não são estatisticamente independentes; em vez disso, têm correlações espaciais com outros.

Devido a fatores competitivos e preocupações com a proteção de dados, os proprietários dos dados (ou agentes) podem não estar dispostos a partilhá-los, apesar dos potenciais benefícios. Duas propriedades (que representam o segundo e terceiro desafios) são necessárias para motivar os agentes a participar em modelos de previsão colaborativa: (1) a privacidade dos dados deve ser preservada durante a colaboração, e (2) os agentes dos dados devem ser compensados pelos dados com os quais contribuem (monetização de dados) – evitando situações em que agentes não queiram colaborar por não terem qualquer benefício na sua previsão, apesar dos seus dados serem relevantes para a previsão de produção dos seus competidores.

As principais contribuições desta tese são: (i) o desenvolvimento de um modelo de previsão de quantis extremos condicionados por covariáveis, que combina estimadores da teoria de valores extremos para a distribuição de Pareto generalizada (truncada) com métodos não-paramétricos, condicionados por informação espaço-temporal; (ii) uma análise numérica e matemática dos modelos de preservação de privacidade aplicados a problemas de regressão, com identificação dos pontos fracos na literatura atual; (iii) o desenvolvimento de um algoritmo de previsão que preserva a privacidade considerando colaboração através de modelos autorregressivos vetoriais, os dados são protegidos combinando transformações por álgebra linear com um algoritmo baseado em decomposição; e (iv) o desenvolvimento de uma solução algorítmica para a monetização de dados em previsão colaborativa, assumindo que os agentes compram previsões de uma entidade confiável em vez de comprar diretamente dados sensíveis.

Ao longo da tese, os modelos propostos são empiricamente avaliados usando dados sintéticos (amostrados usando processos autoregressivos vetoriais) e dados reais publicamente disponíveis (de parques eólicos e solares).

Keywords: Energias Renováveis; Previsão; Quantis Extremos Condicionais; Privacidade de Dados; Monetização de Dados; Previsão Colaborativa.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

ADMM Alternating Direction Method of Multipliers

AI Artificial Intelligence

ANFIS Adaptive Neuro-Fuzzy Inference System

ANN Artificial Neural Network

AR AutoRegressive

ARIMA AutoRegressive Integrated Moving Average

ARMA AutoRegressive Moving Average

CDF Cumulative Distribution Function

CRPS Continuous Ranked Probability Score

DM Diebold-Mariano

EU European Union

EVT Extreme Value Theory

Exp_Tails Exponential function

GAN Generative Adversarial Network

GARCH Generalized AutoRegressive Conditional Heteroskedasticity

GBT Gradient Boosting Tree

GBT_EVT GBT combined with Hill estimator

GBT_tGPD Proposed method combining GBT with truncated GPD

GPD generalized Pareto distribution

k-NN k-Nearest Neighbor

KDE Kernel Density Estimation

LASSO Least Absolute Shrinkage and Selection Operator

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

MBE Mean Bias Error

MLP Multi-Layer Perceptron

MSE Mean Square Error

NRMSE Normalized Root Mean Squared Error

NWP Numerical Weather Prediction

PACF Partial AutoCorrelation Function

PCA Principal Component Analysis

PDF Probability Distribution Function

POT Peaks-over-threshold

QR Quantile Regression

QR_EVT QR combined with Hill estimator

QR_EVT_T QR, Hill estimator and transformed power data

QRNN Quantile Regression Neural Network

RBFNN Radial Basis Function Neural Networks

RES Renewable Energy Sources

RMSE Root Mean Squared Error

RNN Recurrent Neural Network

SETAR Self-Exciting Threshold AutoRegressive

STAR Smooth Transition AutoRegressive

SVR Support Vector Regression

TSO Transmission System Operator

VAR Vector AutoRegressive

Symbols

For notation purposes, vectors and matrices are denoted by bold lowercase and bold uppercase letters, e.g., $\mathbf{a}$ and $\mathbf{A}$, respectively. The vector $\mathbf{a} = [a_1, \ldots, a_k]^\top$ represents a column vector with k dimension, where a_i denotes scalars, $\forall i \in \{1, \ldots, k\}$. The column-wise joining of vectors and matrices is indicated by $[\mathbf{a}, \mathbf{b}]$ and $[\mathbf{A}, \mathbf{B}]$, respectively. Random variables are denoted by italic uppercase letters, e.g., Y. Estimators and estimates are denoted by hat operator "^", e.g., $\hat{Q}$ is the quantile estimator.

The main symbols are here summarized:

Notation	**Description**
n	Number of power agents
A_i	Agent i, $i \in \{1, \ldots, n\}$
h	Lead-time
H	Length of the time horizon $h \leq H$
ν	Sample size for tails representation, using GBT_tGPD
k	Sample size for extreme quantiles extrapolation (Chapter 1) or iteration index for the optimization method (Chapters 2 and 3)
p	Number of lags
T	Number of observations
X	Vector of covariates
$\mathbf{x}$	Observed vector of covariates (one observation)
$\mathbf{X}$	Matrix of observed covariates (multiple observations)
$\mathbf{Z}$	Covariate matrix when defining a VAR model
$\mathbf{B}$	LASSO-VAR coefficients
Y	Target variable
y	Observed target variable
$\mathbf{Y}$	Target matrix when predicting multiple random variables
$Y_{1,T}, \ldots, Y_{T,T}$	Ordered sample of Y
C	Installed power capacity
$H(.)$	Heaviside function
τ	Nominal proportion of a quantile, $\tau \in [0, 1]$
$\rho_\tau(.)$	Pinball loss function
$\hat{Q}^{\text{exp}}(\tau\|\mathbf{x})$	Conditional quantile through exponential functions
$\hat{Q}^{\text{GBT}}(\tau\|\mathbf{x})$	Conditional quantile through a GBT model
$\hat{Q}^{\text{QR}}(\tau\|\mathbf{x})$	Conditional quantile through a QR model
$\hat{Q}^{\text{W}}(\tau\|\mathbf{x})$	Conditional extreme quantile through Weissmans estimator
$\hat{Q}_k^{\text{tGPD}}(\tau)$	Extreme quantile through POT estimator for truncated GPD
$\mathbf{1}_{(.)}$	Indicator function
$\boldsymbol{\beta}$	QR model coefficients
$\hat{\gamma}(\mathbf{x})$	Conditional tail index estimator
$\Lambda_\lambda(.)$	Power transformation function

λ	Power parameter (Ch. 1) or ADMM hyperparameter (Ch. 2 and 3)
$s(.)$	Similarity function between two CDF curves
ρ	Hyperparameters when combining ADMM with LASSO-VAR
$\overline{\mathbf{H}}^k, \overline{\mathbf{U}}^k$	Intermediate matrices at iteration k when using ADMM LASSO-VAR
ρ	Electricity profit function
π_t^s	Spot price
$\pi_t^{\uparrow}$, $\pi_t^{\downarrow}$	Imbalance price for upward / downward regulation
$\lambda_t^{\uparrow}$, $\lambda_t^{\downarrow}$	Regulation unit cost for upward / downward directions
$C_t^{\uparrow/\downarrow}$	Imbalance cost
τ_t^*	Nominal level which minimizes $C_t^{\uparrow/\downarrow}$
$\hat{F}_{i,t}^{-1}(\tau_t^*)$	Forecasted conditional quantile for nominal level τ_t^*
$\hat{\psi}_t^{\uparrow}$, $\hat{\psi}_t^{\downarrow}$	Forecasted upward / downward regulation price
$\hat{p}_t^{\uparrow}$, $\hat{p}_t^{\downarrow}$	Probability of up/downward regulation at time t
$\mathcal{A}$	Overall set of power plants, $\mathcal{A}$=$\{1,\ldots,n\}$
$x_{i,t}$	Power measurements for RES agent i at time t
$\hat{q}_{\tau_t^*}^i$	Forecasted quantile τ_t^* for site $i \in \mathcal{A}$ at time t
$\mathbf{x}_i^{\mathrm{S}}$ (or $\mathbf{x}_i^{\mathrm{B}}$)	Data from seller (or buyer) i
$\mathbf{X}^{\mathrm{S}}$	Data from all sellers, $\mathbf{X}^{\mathrm{S}}$=$[\mathbf{x}_1^{\mathrm{S}},\ldots,\mathbf{x}_n^{\mathrm{S}}] \in \mathbb{R}^{T\times n}$
$\mathcal{M}_i$	Forecasting model for power production of agent i
$\mathcal{G}_i$	Gain function for buyer i
μ_i	Private valuation for each unit gain
b_i	Public bid price (buyer i is willing to pay $b_i \leq \mu_i$)
p_i	Data market price for buyer i
$\mathcal{U}_i$	Value (or utility) function for buyer i
$\mathcal{PF}$	Market price update function (price for the buyer)
$\mathcal{RF}$	Revenue function (price to be paid by buyers)
$\mathcal{AF}$	Allocation function (variables allocation given b_i, p_i)
$\mathcal{PD}$	Payment division function (division by sellers)
$\mathcal{N}(0,\sigma^2)$	Normal distribution with σ standard deviation
$\mathcal{SM}$	Similarity function (similarity between two vectors)
$p_{\min}, p_{\max}$	Minimum and maximum possible data market prices
Δ_p	Increments on possible data market prices
$\mathcal{B}_p$	All possible market prices
$\psi_i(m)$	Fraction of money paid by buyer i allocated to agent m
Δ	Length of the period used to estimate the gain
K	Number of repetitions in the Shapley Approximation

I
Introduction

Electrical energy generation comes from many sources that may be divided into conventional generation technologies (Figure I.1), such as nuclear energy and fossil energy (e.g., oil, coal, natural gas), and Renewable Energy Sources (RES) like wind, solar, geothermal, and hydropower (Figure I.2). In 2018, RES accounted for 18.9% of European energy consumption [1], and the goal of the European Union (EU) is to reach at least 32% of its energy consumption from RES by 2030 [2].

Despite the many benefits of renewable energy sources, there are challenges to overcome since their generation depends on non-human factors, i.e., the weather (wind, clouds, solar irradiance, etc.). Consequently, accurate forecasts are essential to reduce electrical energy imbalances in the electricity market and design decision-aid tools to support the integration of large amounts of RES into the power system.

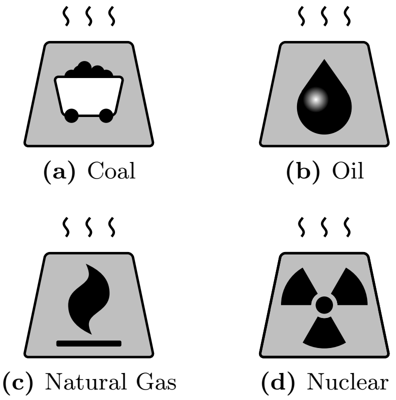

(a) Coal **(b)** Oil **(c)** Natural Gas **(d)** Nuclear

Figure I.1: Conventional generation technologies.

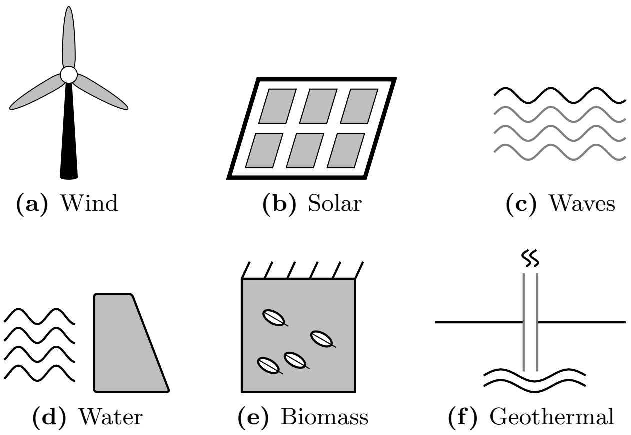

(a) Wind **(b)** Solar **(c)** Waves **(d)** Water **(e)** Biomass **(f)** Geothermal

Figure I.2: Renewable energy sources.

Three main challenges are covered by this PhD thesis for RES:

1. Forecast uncertainty must be minimized so that system operators and electricity market players can make better decisions. Accurate modeling of extreme quantiles (tails of the probability distribution) is paramount to accurately model the risk (i.e., avoid over and underestimation of risk metrics like value-at-risk).

2. Cooperation between multiple RES power plant owners can lead to an improvement in forecast accuracy thanks to the spatio-temporal dependencies in time series data. Such cooperation between agents makes data privacy a necessity since they usually are competitors. However, existing methods of data privacy are unsatisfactory when it comes to time series and can lead to confidentiality breaches – which means the reconstruction of the entire private dataset by another party.

3. Incentives must also exist so that agents are motivated to cooperate by exchanging their data. In fact, agents may be unwilling to share their data, even if privacy is ensured, due to a form of prisoner's dilemma: all could benefit from data sharing, but in practice no one is willing to do so.

The remaining of this chapter contextualizes the thesis by providing its motivation (Section I.1), objectives and contributions (Section I.2), structure (Section I.3), related publications (Section I.4), and a general description of the experimental setup (Section I.5).

I.1 Motivation

A few decades ago, electricity was provided by a single monopoly, usually owned by the state. The EU has pushed (through Directives 2003/54/EC [3], 2009/72/EC [4], and 2019/944 [5]) for energy liberalization, allowing consumers to purchase from different providers. Usually, consumers do not buy directly from the *producer*, they buy from a *retailer*, such as EDP Comercial or Galp Energia – these in turn buy electricity from the producers.

The daily electricity market, so-called **spot market**, is a type of market in which electricity producers offer to sell different quantities of electricity at different prices for each hour of the next day. The line that represents all these proposals constitutes the *supply curve*. In that market, electricity consumers also bid, for each hour of the next day, at what price they are willing to buy electricity. The *demand curve* is the line with all these bids. The intersection of these two lines determines the value of the wholesale market for that hour.

However, unforeseen events arise that vary the energy forecasts and are resolved through **intraday sessions**, in which agents update their offers. For instance, in Portugal and Spain, there are six intraday sessions. Those sessions are necessary because

i) conventional power plants can produce with great reliability, and even then, there is always the possibility of breakdowns and unavailability that cause last-minute changes;

ii) retailers bids are based on demand forecasting, and then deviations can occur;

iii) RES power generation has high variability due to weather dependence, and large deviations may occur between the offered and the real produced value, highly affecting the supply curve, leading to extra costs.

In fact, the increasing introduction of variable RES generation has emphasized the importance of efficient intraday markets – the European Commission is pushing for a new objective of continuous intraday markets with only 60 minutes of margin, enabling electrical energy trading between control areas with surplus and shortage [6]. The main idea is to introduce a single European continuous intraday market based on a common system to which local intraday markets will be linked, as well as the availability of all the capacity of cross-border interconnections that will be facilitated by Transmission System Operators (TSOs) – which are the entities responsible by the transmission grids transporting large quantities of high and very high voltage electricity across vast distances (the Portuguese TSO entity is REN). As long as cross-border transport capacity is available between zones, energy offers and demands introduced by market participants in a country may be matched by orders filed similarly by market participants in any other country that is connected to the central system.

Currently, one of the rules of these TSOs is to ensure the provision of reserves that will allow for sudden imbalances. For the provision of reserves, the TSO tries to determine the optimal reserve production for each market trading period, instructing producers when and how much electricity to generate, and managing any contingent events that cause the balance between supply and demand to be disrupted. The TSO is interested in defining a risk metric for this imbalance, such as *loss of load probability*, and typically wants this probability to be below 1% [7, 8]. These low risk levels motivate the development of probabilistic models that could be improved by using spatio-temporal weather measurements or forecasts, as discussed in what follows.

Probabilistic forecast: Probabilistic RES forecasting models, e.g., to estimate conditional extreme quantiles of power production, are crucial to design advanced decision-aid tools to support not only the definition of reserve levels [7] but also the maximum import net transfer capacity of interconnections [8], simulation of power generation scenarios [9], dynamic line rating [10], etc. However, there is a research gap when it comes to improving probabilistic forecasts using spatio-temporal time series data.

In 2006, the proposal of Gneiting et al. [11] was pioneer. The authors introduced the regime-switching space-time method, which merges meteorological and statistical expertise to obtain probabilistic forecasts of wind resources for two hours-ahead wind speed forecasting. Regarding the expected value forecasting, the monthly Root Mean Squared Error (RMSE) reduced up to 28.6% when compared to the persistence forecasts, and the 90% prediction intervals were, on average, about 18% lower than the intervals obtained by AutoRegressive (AR) models. More recently, in 2017, Andrade and Bessa proposed a forecasting framework to explore information from a Numerical Weather Prediction (NWP) grid applied to both wind and solar energy [12], based on Gradient Boosting Trees (GBTs) models with feature engineering. Relative to a model that only considers one NWP point, it shows an average point forecast improvement, in terms of Mean Absolute Error (MAE), of 16.09% and 12.85% for solar and wind power, respectively. The authors also considered a probabilistic forecast, from the quantile 5% until 95% with 5% increments, which revealed an improvement in the continuous ranking probabilistic score of 13.11% and 12.06% for solar and wind power, respectively. These works show that spatio-temporal information, from measurements (e.g., weather and power) and numerical weather predictions (e.g., a grid of weather forecasts), results in considerable improvements in forecasting accuracy, both in terms of deterministic (expected value) and probabilistic (conditional quantiles) forecasts. However, none of these works consider the forecast of extreme quantiles (e.g., quantiles with a nominal percentage between 0.01% and 1%, or between 99% and 99.99%). Indeed, little research was conducted to predict extreme conditional quantiles, and none that makes use of spatio-temporal information.

This spatio-temporal data might have different owners, which introduces new challenges like data privacy and monetization. Moreover, this requires a collaborative analytics framework, as described in the following paragraphs.

Collaborative models: On the one hand, the weather at a certain site (e.g., a wind farm or a solar farm) is related to its historical values. On the other hand, the weather variables at multiple sites within a certain geographic scale are not statistically independent, instead, they have spatial correlations with others [11]. Given this spatio-temporal reliance on the weather, RES producers located in one region benefit from the data of producers located in another region. In the two aforementioned approaches [11, 12], there is an agent that wants to improve its forecasts using weather measurements and forecasts around its location. However, it would be desirable to combine this with historical power measurements from other producers. For this purpose, Tastu et al. [13] uses a Vector AutoRegressive (VAR) model that makes use of information from multiple power plants. They consider 15 groups of wind power producers. This information can reduce prediction errors by up to 18.46%, in terms of RMSE, when evaluated on the test case of western Denmark. However, sharing data between competing power plants is not always possible, especially if they belong to different companies, due to competitive and privacy concerns.

In this thesis, we evaluate existing approaches for **data privacy** and find them to be unsatisfactory for time series because the methods do not ensure data privacy when lags are used, and existing methods do not work well for data split by features which is necessary for such spatial problems. Therefore, a robust data-privacy protocol is required for VAR models and with the potential to be applied to a broader set of statistical learning models.

Furthermore, agents are unwilling to share data with their competitors unless they are benefited from doing so. Existing literature on how to **monetize information sharing** is either too broad or not generalizable to model forecasting. A mechanism does not exist that can compensate data owners by an amount proportional to the benefit accrued by those who integrate the data into their models. Thus, it is important to develop an algorithmic solution for data markets where the forecasting accuracy, value for a specific use case, and buyer/seller bids define the value of traded data.

I.2 Research Questions and Contributions

From a general point of view, the objective of this thesis is to develop new mathematical and statistical approaches to explore spatio-temporal time series data in forecasting problems, considering data potentially owned by different entities. We recognize three research gaps, related to the three main challenges identified at the beginning of this Prologue, that motivate the following objectives and contributions:

Objective 1. Extreme Conditional Quantiles Forecasting

Research question: *How* to take advantage of covariates when forecasting extreme quantiles of a truncated random variable?

Contributions: Improvement of the probabilistic forecast accuracy on extreme events by combining Extreme Value Theory (EVT) estimators for truncated generalized Pareto distribution (GPD) with non-parametric methods, conditioned by spatio-temporal information.

Objective 2. Privacy-preserving Forecasting Model

Research question: *How* to perform collaborative forecasting without sharing private data and related statistics?

Contributions:
1—Numerical and mathematical analysis of the existing privacy-preserving regression models and identification of weaknesses in the current literature;
2—Development of privacy-preserving forecasting algorithms. Data privacy is ensured by combining linear algebra transformations with a decomposition-based algorithm, allowing to compute the model's coefficients in a parallel fashion.

Objective 3. Data Markets for Collaborative Forecasting

Research question: *Why* should an agent collaborate with others, even when its model accuracy does not improve?

Contributions: Development of an algorithmic solution for data monetization in collaborative forecasting.

While the motivation and empirical evaluation of the thesis is on RES forecasting, the contributions themselves are not specific to RES and could be applied to other areas, such as:

- **Weather and climate:** Forecasting conditional extreme quantiles for wind speed, precipitation, air pollution, etc, can be improved by using data from multiple monitoring stations.
- **Economics and finance:** Extreme events are important for insurance companies [14]. Furthermore, cooperation between entities can help forecasting product prices more accurately. For instance, in [15] the retail prices for a specific product are predicted at every outlet by using historical retail prices of the product at a target outlet and at competing outlets.
- **Logistics:** Cooperation between supply chains can reduce their forecasting errors. For example, the VAR model is used in [16, 17] for inventory control in supply chains; Extreme conditional quantile forecasting is also of interest for production planning and inventory management [18].

I.3 Thesis Structure

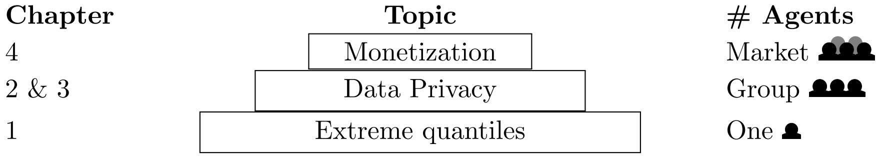


Figure I.3: Relation between objectives and thesis structure.

The thesis is structured into four major chapters related to the contributions described in the previous section, and depicted in Figure I.3. A brief description of each chapter:

Chapter 1 focuses on the first objective: forecasting conditional extreme quantiles given a set of covariates. In this document, extreme quantiles correspond to the quantiles with a nominal proportion below 0.05 and above 0.95. This chapter describes a novel forecasting method combining non-parametric methods with truncated GPD.

Chapter 2 relates to the second objective and analyzes the state-of-the-art and unveils several shortcomings of existing methods in guaranteeing data privacy when employing VAR models.

Chapter 3 relates also to the second objective and proposes a novel forecasting model that allows a model to be estimated in a distributed fashion with privacy protection for the data, coefficients and covariance matrix.

Chapter 4 focuses on the third objective, data monetization. An algorithmic solution for data monetization is proposed, in which agents buy forecasts from a trusted entity instead of directly buying sensible data.

I.4 Publications

Each chapter along the thesis has one companion publication published in a peer-reviewed journal with quartile score Q1 (the impact factor is indicated as IF).

Objective 1. Extreme Conditional Quantiles Forecasting

> **Chapter 1.** C. Gonçalves, L. Cavalcante, M. Brito, R.J. Bessa and J. Gama, "Forecasting conditional extreme quantiles for wind energy," *Electric Power Systems Research*, vol. 190, pp. 106636, Jan. 2021, doi:10.1016/j.epsr.2020.106636. [IF=3.211, Q1]

Objective 2. Privacy-preserving Forecasting Model

> **Chapter 2.** C. Gonçalves, R.J. Bessa, and P. Pinson, "A critical overview of privacy-preserving approaches for collaborative forecasting," *International Journal of Forecasting*, vol. 37, no. 1, pp. 322-342, 2021, doi:10.1016/j.ijforecast.2020.06.003. [IF=2.825, Q1]
>
> **Chapter 3.** C. Gonçalves, R.J. Bessa, and P. Pinson, "Privacy-preserving distributed learning for renewable energy forecasting," *Under review in IEEE Transactions on Sustainable Energy*, 2020. [IF=7.44, Q1]

Objective 3. Data Markets for Collaborative Forecasting

> **Chapter 4.** C. Gonçalves, P. Pinson, and R. J. Bessa. "Towards data markets in renewable energy forecasting". *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 533-542, Jan. 2021, doi: 10.1109/TSTE.2020.3009615. [IF=7.44, Q1]

The work developed in this thesis was also disseminated by conferences:

- The **Chapter 1** proposal was presented at the international XXI "Power Systems Computation Conference" (PSCC 2020).

  C. Gonçalves, L. Cavalcante, M. Brito, R. J. Bessa, and J. Gama, "Forecasting conditional extreme quantiles for wind energy," `https://pscc-central.epfl.ch/repo/papers/2020/225.pdf`, *PSCC 2020*, (accessed November 19, 2020).

- The **Chapter 4** proposal was presented at the 40th International Symposium on Forecasting (ISF 2020).

  C. Gonçalves, R. J. Bessa, and P. Pinson, "Data market for collaborative renewable energy forecasting," `https://whova.com/embedded/speaker/iiofe202006/12531898/`, *ISF 2020*, (accessed November 19, 2020).

In addition, a patent was submitted to the European Patent Office (EPO) in Oct 2020, related to the privacy-preserving model proposed in **Chapter 3**.

> R. J. Bessa and C. Gonçalves, "Method and device for preserving privacy of linear regression distributed learning", *Submitted to EPO.*

Furthermore, during these four years, parallel work related to energy analytics models was conducted as part of collaborations between INESC TEC and energy companies, which we decided not worth including in the body of the thesis, but are here briefly mentioned:

Advantages of uncertainty models. Development of computation methods and tools to assess the advantages of using uncertainty models to predict the security of electrical systems in case of overloading in the branches, in collaboration with RTE France, a transmission system operator.

> M.H. Vasconcelos, C. Gonçalves, J. Meirinhos, N. Omont, A. Pitto, & G. Ceresa. "A methodology to evaluate the uncertainties used to perform security assessment for branch overloads." *International Journal of Electrical Power & Energy Systems* 112 (2019): 169-177. [IF=3.588, Q1]

M.H. Vasconcelos, C. Gonçalves, J. Meirinhos, N. Omont, A. Pitto, & G. Ceresa. ”Evaluation of the uncertainties used to perform flow security assessment: a real case study,” *13th IEEE PES PowerTech Conference* (2019).

Causality analysis. Understanding the main factors that influence the electricity prices and the mobilization of reserves. The applied techniques include Least Absolute Shrinkage and Selection Operator (LASSO) linear and logistic regression, and causal analysis based on algorithms of causality discovery, such as segmentation techniques with classification trees and neural network models. This work was done in collaboration with EDP – Gestão da Produção de Energia, S.A.

C. Gonçalves, M. Ribeiro, J. Viana, R. Fernandes, J. Villar, R. Bessa, et al. “Explanatory and causal analysis of the MIBEL electricity market spot price,” *13th IEEE PES PowerTech Conference* (2019).

C. Gonçalves, M. Ribeiro, J. Viana, R. Fernandes, J. Villar, R. Bessa, et al. “Explanatory and causal analysis of the Portuguese manual balancing reserve,” *Submitted to 14th IEEE PES PowerTech Conference* (2021).

I.5 Datasets used in the Experiments

Table I.1 provides a brief description of the data considered in this thesis, as well as the related chapters. Datasets include power time series data, NWP for a set of locations around power plants, and synthetic data. The use of synthetic data was a common practice to verify the correct implementation of the algorithms proposed in this thesis. Also, some details about practical implementation are provided. The experiments were carried out using R [27] and Python [28] programming languages.

Table I.1: General description of the experimental setup.

Ch.	RES	Variables	Ref.	Period	Horizon	Data Description	Prog. Language and Packages
1	Synthetic	–	–	–	1	Data is generated by assuming Y follows a truncated Burr or GPD distribution, with parameters depending on $X_1, X_2 \sim \mathcal{U}[-2, 2]$.	The R packages include • `quantreg` [19] (for quantile regression) • `Rearrangement` [20] (for quantile crossing problem) • `ReIns` [21] (for truncated GPD estimation). The GBT model was implemented in Python using the `scikit-learn` library [22].
	Wind	Power & NWP	[12]	Jan 2014–Sep 2016 (hourly resolution)	1 to 24h	Power data from the *Sotavento* wind power plant, located in Galicia (Spain), with a total installed capacity of 17.56 MW. Feature engineered variables from NWP data, as proposed by [12].	
	Solar	Power & NWP	[23]	Apr 2013–Jun 2016 (hourly resolution)	1 to 24h	Power data from a solar power plant located in Porto (Portugal), with a total installed capacity of 16.32 kW peak. Feature engineered variables from NWP data, as proposed by [12].	
2	Synthetic	–	–	20000 records	1	Sampled through a VAR model.	These experiments were performed in R, and the code is available in [24].
	Wind	Power	[25]	Jan 2011–Jun 2013 (hourly resolution)	1h	Power data from 15 wind farms, GEFcom 2014 competition.	
	Solar	Power	[24]	Feb 2011–Feb 2013 (hourly resolution)	1h	Power data from 44 micro-generation units located in Évora (Portugal).	
3	Wind	Power	[25]	Jan 2011–Jun 2013 (hourly resolution)	1h	Same as Chapter 2.	These experiments were performed in R, following the pseudo-code presented in Chapter 3.
	Solar	Power	[24]	Feb 2011–Feb 2013 (hourly resolution)	6h	Same as Chapter 2.	
4	Synthetic	–	–	8760×2 records	1	Sampled through a VAR model	These experiments were performed in R, following the pseudo-code presented in Chapter 4.
	Wind	Power & Prices	[26]	Jan 2016–Oct 2017	1h	Wind power data, spot price and imbalance prices for up and downward regulation, available in the NordPool website[1], from 6 regions: 4 in Sweden and 2 in Denmark.	
	Solar	Power	[24]	Feb 2011–Feb 2013 (hourly resolution)	6h	Same as Chapter 2.	

II Background Knowledge

Before proceeding to the proposed forecasting models and applications, an overview of the different aspects of Renewable Energy Sources (RES) forecasting, and the related limitations, is covered in this chapter. First, the main concepts and taxonomy for RES forecasting are introduced in Section II.1, and a brief literature review is provided in Sections II.1.2 to II.1.4 for the forecasting methods. Section II.2 describes the most common evaluation metrics. Then a mathematical description of the relevant models is provided in Section II.3.

II.1 Overview of Renewable Energy Forecasting

In order to improve decision-making under risk in power systems and electricity markets with high RES integration, system operators, market agents, and RES producers require highly accurate point and probabilistic forecasts. It is important to underline that variability in RES time series is inherently and the role of statistical models is to decrease uncertainty.

II.1.1 Taxonomy

When forecasting power generation Y for time $t+h$, Y_{t+h}, it is important to consider the information collected at current time t, Ω_t, such as wind speed, irradiance, past generation level (i.e., lagged power generation), etc. Different types of conditional forecast models are identified in the literature (see Figure II.1):

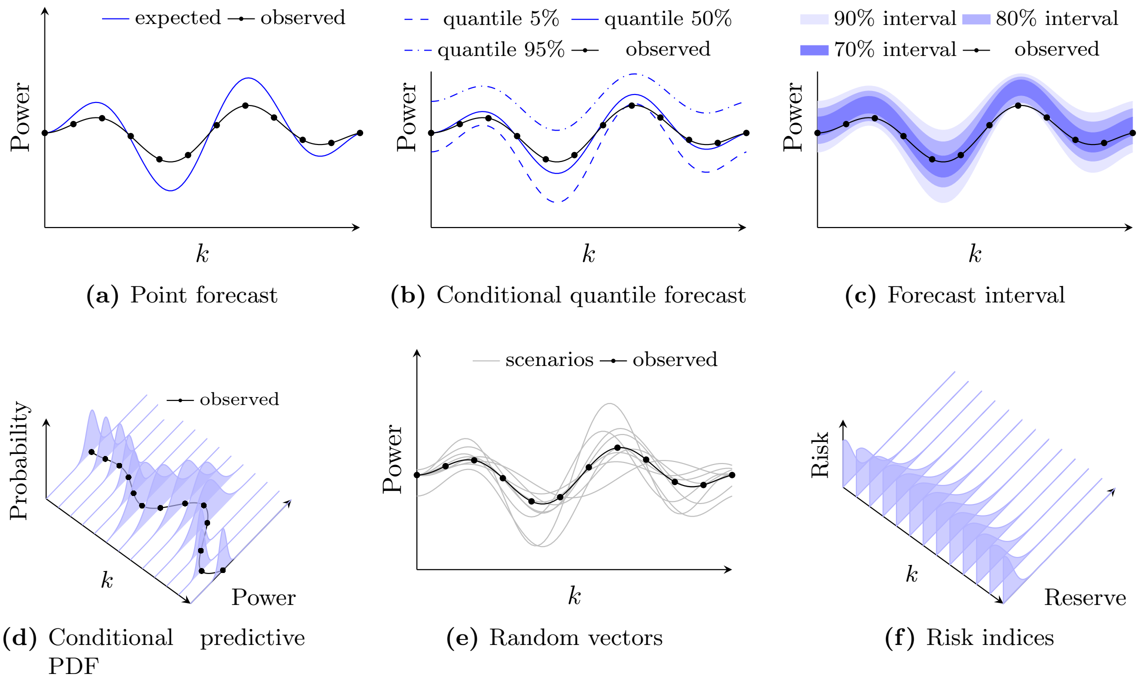
a) ***Point*** **or** ***deterministic forecasts*** are an estimation, issued at time t for time $t+h$, based on the conditional expectation of Y_{t+h} given a model $\mathcal{M}$ with estimated parameters $\hat{\Theta}$, and the information set Ω_t,

$$\hat{y}_{t+h|t} = \mathbb{E}(Y_{t+h}|\Omega_t, \mathcal{M}, \hat{\Theta}). \tag{II.1}$$

b) ***Conditional quantile forecast*** $\hat{q}^{\tau}_{t+h|t}$ with nominal level $\tau \in [0,1]$ is an estimate, issued at time t for time $t+h$, of the quantile q^{τ}_{t+h} for random variable Y_{t+h}, given a model $\mathcal{M}$ with estimated parameters $\hat{\Theta}$, and the information set Ω_t, i.e.,

$$\Pr[Y_{t+h} \leq \hat{q}^{\tau}_{t+h|t}|\Omega_t, \mathcal{M}, \hat{\Theta}] = \tau. \tag{II.2}$$

By issuing a quantile forecast $\hat{q}^{\tau}_{t+h|t}$, the forecaster tells at time t that there is a probability τ that RES generation will be less than $\hat{q}^{\tau}_{t+h|t}$ at time $t+h$. These conditional quantiles need to be carefully interpreted, otherwise misleading information may be relayed to the decision-maker, e.g., misinterpret each one of the quantiles in Figure II.1 (b) as a possible temporal evolution.



(a) Point forecast **(b)** Conditional quantile forecast **(c)** Forecast interval

(d) Conditional predictive PDF **(e)** Random vectors **(f)** Risk indices

Figure II.1: Types of forecasting models output.

c) ***Forecast interval*** is a range $\hat{\mathbf{I}}_{t+h|t}^{(\beta)}$ of potential values for Y_{t+h} for a certain level probability $\beta \in [0,1]$ such that

$$\Pr\left[Y_{t+h} \in \hat{\mathbf{I}}_{t+h|t}^{(\beta)} | \Omega_t, \mathcal{M}, \hat{\Theta}\right] = \beta. \tag{II.3}$$

Such an interval must be defined as

$$\hat{\mathbf{I}}_{t+h|t}^{(\beta)} = \left[\hat{q}_{t+h|t}^{\beta/2}, \hat{q}_{t+h|t}^{1-\beta/2}\right], \tag{II.4}$$

where $\hat{q}_{t+h|t}^{\beta/2}, \hat{q}_{t+h|t}^{1-\beta/2}$ are conditional quantile forecasts. As in the previous case, intervals can only be interpreted individually for each lead time.

d) ***Conditional predictive PDF*** for Y_{t+h} issued at time t is a complete description $\hat{f}_{t+h|t}$ of the Probability Distribution Function (PDF) of Y_{t+h} conditional on a model $\mathcal{M}$ with estimated parameter $\hat{\Theta}$, and information set Ω_t. Similarly, $\hat{F}_{t+h|t}$ is a complete description of the conditional Cumulative Distribution Function (CDF) of Y_{t+h} issued at time t.

e) ***Random vectors***, ***trajectories***, ***scenarios*** **or** ***ensemble forecasting*** are different terms for the same concept: equally-likely samples of multivariate predictive densities for power generation (in time and/or space). Random-vectors are the term used in statistics, while trajectories, scenarios or ensemble forecasting are terms commonly used in economics and finance, meteorology, and energy modeling. Scenarios issued at time t for a set of h successive lead times are samples of the predicted multivariate CDF for $(Y_{t+1}, \ldots, Y_{t+h})$, and consist in a set of J time trajectories $\mathbf{z}^{(j)} = (y_{t+1|t}^{(j)}, y_{t+2|t}^{(j)}, \ldots, y_{t+h|t}^{(j)})$, $j \in \{1, \ldots, J\}$.

f) ***Risk indices*** are not typically considered models, since they are obtained by post-processing the output of some of the previous models. They are a summary of the probability distribution; e.g., (conditional) value-at-risk.

These conditional models may be classified as either *point* or *probabilistic* forecasting models. Point forecasts are easier to interpret since they only provide one value for each lead time $t + h$, but they do not provide any information about the uncertainty around the predicted value. For example, if the power production point forecast for the next hour is 1000 MW, then the Transmission System Operator (TSO) is unaware whether there is a strong possibility of the production coming closer to 500 MW or 1500 MW or some other value, and these values may represent different decisions since the final goal is to balance power system production and consumption. For this reason, probabilistic forecasts through the form of conditional quantiles or prediction intervals or PDFs or trajectories, all of which are essential to help agents to make better decisions.

Recent reviews [29, 30, 31] outline the several methods that have been applied in RES forecasting. These methods are usually classified according to the prediction horizon h (which is the time interval between the actual and effective time of prediction) or the methodology:

- **Prediction horizon:** the classification of RES forecasting approaches is not consensual, but taking [32] as reference, four categories are considered: very short-term (up to 6 hours), short-term ($>$ 6 hours to 3 days), medium-term (4 to 7 days) and long-term ($>$ 7 days). Naturally, the selection of the input variables Ω_t and the most suitable models $\mathcal{M}$ are highly dependent on the time horizon.

- **Methodology:** Four groups of RES forecasting approaches are identified: persistence model, physical models, statistical models, and hybrid models. The following subsections are organized by methodology.

An overview of RES forecasting approaches is provided in the next subsections, detailing more on statistical models because they are the focus of this thesis. Section II.1.2 briefly describes the literature on persistence, physical and statistical models, regarding point forecasting. Then, Section II.1.3 describes the models to perform uncertainty forecasting. Commonly, these point and probabilistic RES forecasting models are focused on individual RES power plants, and only consider data for their geographical location. But, as explained in Prologue I, the weather variables at a certain site are related to its historical values, and their values at multiple sites within a certain geographic scale are not statistically independent. Given this spatio-temporal weather reliance, new approaches are being considered that perform RES forecasting using spatio-temporal time series data, as described in Section II.1.4.

II.1.2 Point Forecast Models

Point forecast models predict the expected value of the series, as illustrated by Figure II.1 (a). The topography of point forecast models, which is summarized in Figure II.2, is now elaborated.

Persistence model

This is a popular model for very short-term forecasting since it has a less computational cost and low time delay. This technique states that the future generation $\hat{Y}_{t+h}$ issued at time t will be the same as the last measured value y_t.

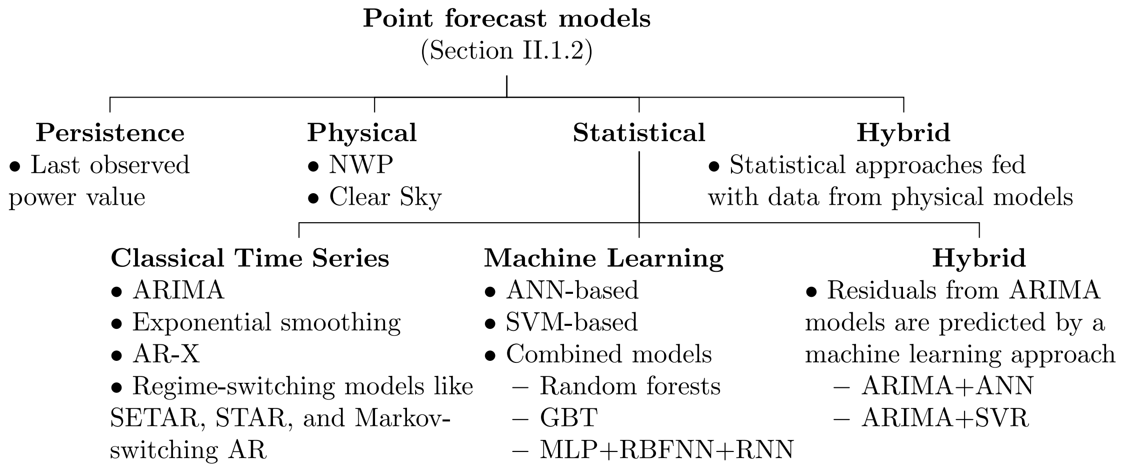


Figure II.2: Topography of point forecast models.

Physical models

The most popular physical model provides a grid of Numerical Weather Predictions (NWPs). This predictor is based on a mathematical set of equations, which describes the physical state and dynamic motion of the atmosphere. Since the NWP are provided to a set of grid points covering an area, a more detailed characterization of the weather variables in the RES power plants location requires an extrapolation of these forecasts. Weather extrapolation is usually performed with mesoscale, computational fluid dynamic (which includes for example Navier-Stokes equations) or linear models [33].

NWP models provide forecasts of variables such as temperature, relative humidity and wind speed and direction at different heights, but also global and direct irradiance, which is relevant for solar power plants. However, extrapolating the irradiance to solar power plants' location can result in misleading predictions since the cloud cover is not taken into account. To address this limitation, clear sky models have been proposed to estimate solar irradiance under clear-sky conditions in specific locations, simplifying the atmospheric dynamics with relatively simple characterizations. An up-to-date review on clear sky models can be found in [34]. The assumption of clear sky conditions is the main disadvantage of these models.

The conversion of weather variables to power may be performed through manufacturers' power curves (e.g., the wind power curve is proportional to the cube of wind speed [35]) or statistically. Physical forecasting methods such as NWP models show higher accuracy when the environment is stable. However, the complex atmospheric information requirements add high computation complexity in solving these models.

Statistical models

This class of models use mathematical equations to extract the patterns and correlation from input data, and are estimated by minimizing the difference between past measured values and predicted values, meaning the prediction accuracy depends on the quality and dimension of the data. Commonly, statistical models are divided into three main subjects: *classical time series models*, *machine learning models* and hybrids between the two.

Classical times series models use historical time series and real-time generated power data to predict the power generation. Usually, these techniques achieve good accuracy

$$\begin{array}{c|ccccc}
\overbrace{\text{target } Y} & \multicolumn{3}{c}{\overbrace{p \text{ lags of wind power}}} & \multicolumn{2}{c}{\overbrace{\text{example of NWP}}} \\
y_{t+1} & y_t & \cdots & y_{t+1-p} & \widehat{wind\,speed}_{t+1} & \widehat{wind\,direction}_{t+1} \\
y_{t+2} & y_{t+1} & \cdots & y_{t+2-p} & \widehat{wind\,speed}_{t+2} & \widehat{wind\,direction}_{t+2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
y_{t+h} & y_{t+h-1} & \cdots & y_{t+h-p} & \widehat{wind\,speed}_{t+h} & \widehat{wind\,direction}_{t+h} \\
 & \multicolumn{3}{c}{\underbrace{\qquad\qquad\qquad}_{\text{classical time series}}} & & \\
 & \multicolumn{5}{c}{\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{\text{machine learning models}}}
\end{array}$$

(a) Wind power forecasting

$$\begin{array}{c|ccccccccc}
\overbrace{\text{target } Y} & \multicolumn{3}{c}{\overbrace{p \text{ lags of solar power}}} & \multicolumn{3}{c}{\overbrace{\text{example of NWP}}} & \multicolumn{2}{c}{\overbrace{\text{calendar variables}}} \\
y_{t+1} & y_t & \cdots & y_{t+1-p} & \widehat{irradiance}_{t+1} & \widehat{temperature}_{t+1} & \widehat{cloud\,cover}_{t+1} & hour_{t+2} & month_{t+1} \\
y_{t+2} & y_{t+1} & \cdots & y_{t+2-p} & \widehat{irradiance}_{t+2} & \widehat{temperature}_{t+2} & \widehat{cloud\,cover}_{t+2} & hour_{t+2} & month_{t+2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
y_{t+h} & y_{t+h-1} & \cdots & y_{t+h-p} & \widehat{irradiance}_{t+h} & \widehat{temperature}_{t+1} & \widehat{cloud\,cover}_{t+h} & hour_{t+h} & month_{t+h} \\
 & \multicolumn{3}{c}{\underbrace{\qquad\qquad\qquad}_{\text{classical time series}}} & & & & & \\
 & \multicolumn{8}{c}{\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{\text{machine learning models}}}
\end{array}$$

(b) Solar power forecasting

Figure II.3: Illustration of data used by statistical models.

for very short-term RES forecasting; however, exogenous variables (e.g. wind speed forecasting or solar irradiance) are fundamental for larger forecast horizons, motivating the extension of these approaches to include those extra input variables. For example, the authors of [36, 37] forecast 24h-ahead wind and solar power, respectively, by combining power measurements with weather forecasts through an AR-X model.

Examples of these models are: AutoRegressive Moving Average (ARMA), (seasonal) AutoRegressive Integrated Moving Average (ARIMA), exponential smoothing, Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) [30], and regime-switching models [38] like Markov-switching AutoRegressive (AR) models, Self-Exciting Threshold AutoRegressive (SETAR), and Smooth Transition AutoRegressive (STAR).

The main limitation of classical times series models is that they are generally limited to linear coefficients.

Machine learning models learn more complex relationships between input variables and power values, and have been used in several papers about RES power forecasting. In addition to historical power data, these models tend to use meteorological information, as illustrated in Figure II.3. For solar forecasting, exogenous variables are usually related to irradiance and temperature, while for wind forecasting the tendency is variables related to wind speed and direction. An up-to-date literature review about RES forecasting using machine learning models can be found at [39]. Successfully applied methods include Support Vector Regression (SVR)-based approaches [40] and Artificial Neural Network (ANN)-based approaches such as Multi-Layer Perceptron (MLP) [41], Radial Basis Function Neural Networkss (RBFNNs) [42], Adaptive Neuro-Fuzzy Inference System (ANFIS) [43], and Recurrent Neural Networks (RNNs) [44] which models temporal dependencies between consecutive input and predicted/observed power. Usually, RNNs perform better than classical time series models [45].

Since these methods capture different relationships, combining machine learning methods can improve the final forecasting accuracy [46]. Two main combination structures are here identified:

i) Weight-based combined approaches assign a weighting coefficient w_i to each method $\mathcal{M}_i$ proportional to their past forecasting performance,

$$\hat{y}_{t+h|t} = \sum_i w_i \hat{y}^{(i)}_{t+h|t}, \tag{II.5}$$

where $\hat{y}^{(i)}_{t+h|t}$ is the final forecast for time $t+h$ performed at time t, using model $\mathcal{M}_i$, $i \geq 2$. For example, random forests perform RES forecasting based on equal weighting multiple regression trees, which are trained from random samples (selected by bootstrapping) of data under analysis [47]. Similarly, the authors of [48] combine MLP, RBFNN and RNN. Thordarson et al. [49] combine multiple forecasts by using conditional weights estimated through a linear regression model, but in this case the individual models are trained with all samples.

ii) Sequential combined approaches first apply a model for RES forecasting and then use its residuals or forecasts as the input of another statistical model. For example, Gradient Boosting Tree (GBT) sequentially trains multiple random trees – first a regression tree models the power values using input variables, then a second regression tree models the residuals from the first tree using the input variables, and so on [50]. Also, in [51], an ANFIS model firstly predicts solar power by using historical power, irradiation and temperature measurements, and then these forecasts are used, in addition to the previous input data, as input in a feed-forward neural network.

Although the choice of the forecasting model is an important factor, imprudent input selection can cause a low accuracy rate on highly accurate forecast models. This has motivated the application of feature engineering techniques that extract relevant information from input data using lags (past values) of the variables, moving averages, differentiation, etc. The relevance of these techniques was demonstrated at the Global Energy Forecasting Competition 2014 (GEFCom2014), in which the models that ranked first and second combined feature engineering techniques with machine learning models [25], and more recently, such techniques won first place in the European Energy Market 2020 (EEM20) competition [52].

Another common practice to improve the conversion of input variables to power forecasts is to decompose the output time series, by using pre-processing models, into stationary and regular subseries which are generally easier to model. Thus, the ensuing subseries are individually predicted by a statistical model, and the final forecast consists of the aggregation of these individual forecasts. In [53], two different signal decomposition methods are introduced for short-term wind power forecasting: wavelet transform, and another is empirical mode decomposition. ANN is then used to model the decomposed time series. Similarly, a combination of empirical mode decomposition with kernel ridge regression and SVR are proposed in [54, 55], respectively.

The main limitation of machine learning techniques is to interpret the models since they map the input variables into power values using highly-complex functions. For this reason, explainable machine learning has become an emerging research field since it is important to understand why a model makes its forecast decision. For example, random forests are applied in Kuzlu et al. [56] to forecast solar power using temperature, humidity, hour of the day, etc. Then, the importance of each variable is determined by: (i) Local Interpretable Model-agnostic Explanations (LIME), i.e., for each observation a "local sensitivity analysis" is performed to understand how sensitive is the prediction with regards to each feature of this particular observation, and (ii) Shapley additive explanations which are a game-theoretic approach that estimates the importance of each feature by training models with multiple combinations of the input variables.

Hybrid models, which combine classical time series with machined learning models, usually model the linear dependencies by using an ARIMA model, and then the residuals are predicted through an ANN or SVR-based model [57], aiming to capture the nonlinear relationships.

Physical and statistical models

Results derived from physical modeling are enriched or statistically corrected by power plant data. The weather conditions predicted by the physical models are used as inputs to the statistical models. Some examples include: combining an ANN with the clear sky solar radiation model [58]; statistical normalization of the solar power data using a clear sky model, followed by an AR model or an AR with exogenous input from NWP [37]; RBFNNs fed with past power measurements and meteorological forecasts of wind speed and direction (from NWP) interpolated at the site of the wind farm [59]; and combination of GBT with feature engineering techniques, such as lagging variables and Principal Component Analysis (PCA), that extract the maximum information from the NWP grid centered at the wind and solar power site [12]. For the solar power forecasting, the considered NWP variables include cloud cover for different levels, irradiance, and temperature; while for wind power forecasting, they are the wind azimuthal and meridional wind speed, wind module, and wind direction at different levels.

II.1.3 Probabilistic Forecast Models

The approaches described in the previous section are commonly used to predict the conditional expected power values. However as discussed before, the information provided by point forecasts is unsatisfactory for some decision-making problems and needs to be complemented with information about the uncertainty around such point forecasts. The topography of probabilistic forecasting models, which is summarized in Figure II.4, is now elaborated:

Weather ensembles. Traditional approaches of physical models have been built on a foundation of deterministic modeling, i.e., the models start with initial conditions, and end up with a prediction about future weather. However, different weather trajectories can be considered with slightly different starting conditions or model assumptions. These trajectories are commonly called weather ensembles [60].

Usually, statistical models are used to post-process weather ensembles, allowing: 1) calibration of quantiles – the employed methods range from simple bias corrections to very sophisticated distribution-adjusting techniques that incorporate correlations among the variables [61]; 2) conversion of meteorological variables to power forecasts.

Statistical models. An up-to-date literature review about RES probabilistic forecasting can be found at [29]. Two main techniques to construct predictive distributions involve: *parametric* and *non-parametric* approaches.

Parametric models assume that data are generated from a known probability distribution (e.g., Gaussian, Beta, generalized logit-Normal distribution), whose parameters are

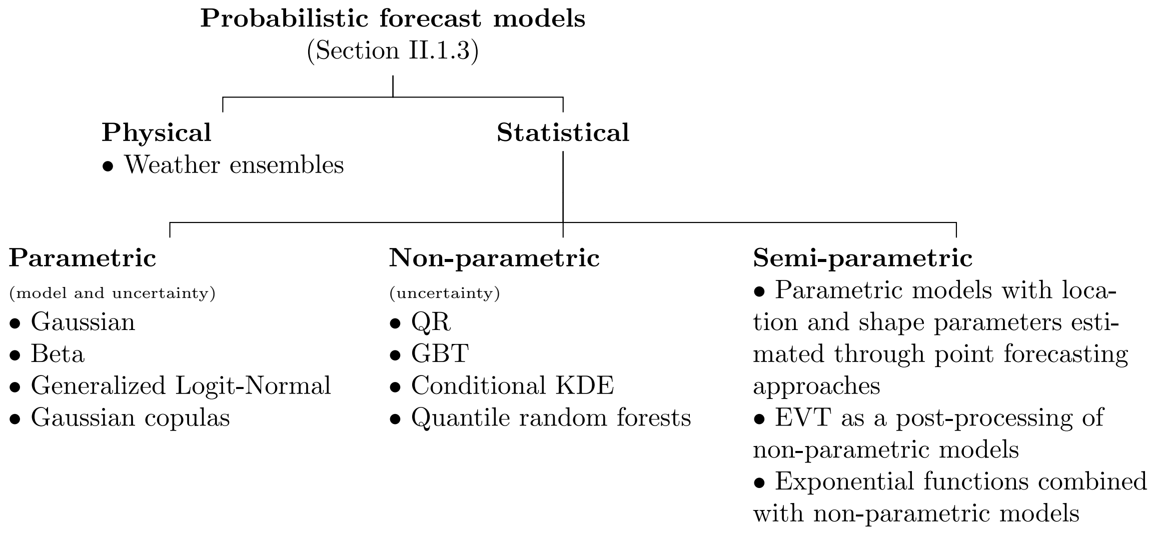

Probabilistic forecast models
(Section II.1.3)
Physical
• Weather ensembles
Statistical
Parametric
(model and uncertainty)
• Gaussian
• Beta
• Generalized Logit-Normal
• Gaussian copulas
Non-parametric
(uncertainty)
• QR
• GBT
• Conditional KDE
• Quantile random forests
Semi-parametric
• Parametric models with location and shape parameters estimated through point forecasting approaches
• EVT as a post-processing of non-parametric models
• Exponential functions combined with non-parametric models

Figure II.4: Topography of probabilistic forecast models.

estimated from the data. *Non-parametric models* do not make any assumptions about the shape of the probability distribution, instead they learn from data by using a parametric formula. Non-parametric models comprise techniques such as linear Quantile Regression (QR) that models uncertainty by linearly combining input variables, QR with radial basis functions [62], local QR [63], additive quantile regression [64], Quantile Regression Neural Network (QRNN) [65], conditional Kernel Density Estimation (KDE) [66], k-Nearest Neighbor (k-NN) based approaches [25], quantile regression forests [52] and GBT [12]. It is also possible to find *semi-parametric approaches*, e.g., a mixture of a censored distribution and probability masses on the upper and lower boundaries that transform wind power data into a Gaussian distribution, whose mean and standard deviation are predicted with a statistical model [67]; a combination of linear regression, inverse (power-to-wind) transformation and censored normal distribution [68]; a combination of QR models (for quantiles with nominal proportions between 0.05 and 0.95) with exponential functions (for the remaining quantiles); extreme quantile forecasting by applying Extreme Value Theory (EVT) as a post-processing step over a set of quantiles first estimated by a non-parametric method [69].

The main advantage of parametric methods is that the distribution's shape only depends on a few parameters, resulting in a simplified estimation and consequently requiring low computational costs. However, the choice of the parametric function is not straightforward. On the other hand, non-parametric models require a large number of observations to achieve good performance. Therefore, when estimating extreme quantiles, non-parametric models tend to have poor performance due to a lack of data representing extreme events. This is critical because a poor forecast of extreme quantiles can have a high impact on different decision-aid problems, in particular when decision-makers are highly risk-averse, as discussed in Prologue I.

Moreover, the generation of temporal and/or spatio-temporal trajectories with a parametric statistical method, such as copulas, requires the estimation of the entire CDF for each time and/or location, and an accurate estimation of the tails avoids trajectories with unrealistic "extreme" values. Let us assume we want to generate temporal trajectories for the random vector $(Y_{t+1}, \ldots, Y_{t+h})$ at time t. Copulas are multivariate cumulative distribution functions for which the marginal probability distribution of each variable is uniform, $\mathcal{U}[0,1]$. By applying the probability integral transform, the random vector $(U_{t+1},\dots,U_{t+h}) = (\hat{F}_{t+1|t}(Y_{t+1}),\dots,\hat{F}_{t+h|t}(Y_{t+h}))$ has uniform marginals. Then, the copula $C(.)$ of $(Y_{t+1},\dots,Y_{t+h})$ is defined as the multivariate cumulative distribution function of $(U_{t+1},U_{t+2},\dots,U_{t+h})$,

$$C(u_{t+1},\dots,u_{t+h}) = \Pr[U_{t+1} \leq u_{t+1},\dots,U_{t+h} \leq u_{t+h}]. \tag{II.6}$$

The temporal trajectories are then sampled by reversing these steps. That is, given a procedure to generate a sample $(U_{t+1},\dots,U_{t+h})$, the random sample for $(Y_{t+1},\dots,Y_{t+h})$ can be computed as

$$\left(\hat{F}^{-1}_{t+1|t}(U_{t+1}),\dots,\hat{F}^{-1}_{t+h|t}(U_{t+h})\right). \tag{II.7}$$

Applied copulas include the Gaussian [70], t-student, vine copulas [71], etc.

There are also non-parametric models to generate such trajectories. Generative Adversarial Networks (GANs) are applied in [72] to generate wind generation trajectories using wind time series historical data. GAN refers to a class of machine learning methods, in which two neural networks are trained: a generator network that samples data, and a discriminator network that distinguishes historical data from the generated data. The main disadvantage of these models is that they have an overly complex structure and require large amounts of wind data for training.

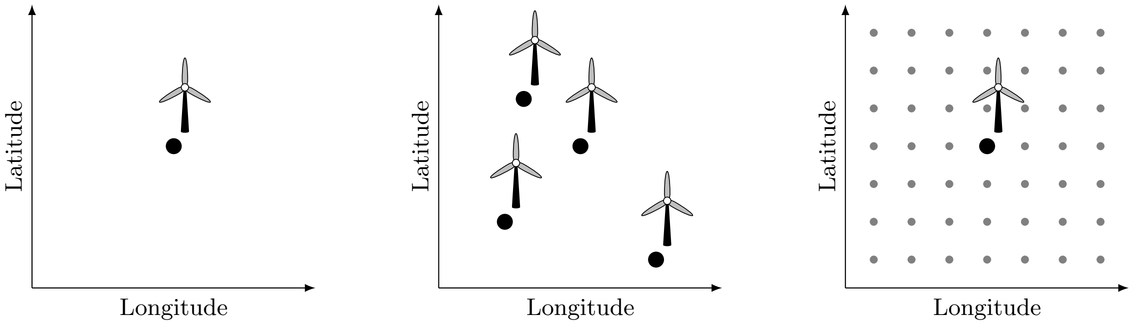
In fact, as argued in [29], when dimension increases using such scenarios may not be practical, owing to the difficulty in solving the resulting optimization problems, and may not be possible at reasonable computational costs. This motivated various developments in stochastic optimization and control that, instead of relying on a large number of trajectories, prefer to solve problems based on multivariate forecast regions, possibly taking the form of ellipsoids [73] or polyhedra [74].

Lastly, regarding forecast intervals, other techniques exist in addition to constructing intervals using the conditional quantile forecasting models described before. For example, an ANN approach with two outputs is used in [75] to predict directly both inferior and superior interval limits. Also, in [76], extreme learning machines (an ANN-based approach) are used to predict forecast intervals for multiple nominal coverage rates at the same time. Both works predict the intervals for each time horizon separately. To capture temporal dependence between consecutive forecast intervals, the work in [77] first generates temporal trajectories by using the Gaussian copula method and the marginal prediction intervals. Then, two methods proposed in the literature are used to construct simultaneous intervals.

II.1.4 Forecasting with Geographically Distributed Data

The development of smart grids and RES dispatch centers provide real-time measurements from sensors distributed geographically, which the forecasting methodologies can take advantage of. In summary, the data used to forecast the power of a production unit can refer only to its specific location (Figure II.5 (a)) or to a more comprehensive set of geographical points. In the latter case, two types of spatio-temporal data are distinguished: power measurements or weather variables collected (or predicted) by other production units (Figure II.5 (b)); and meteorological data for a grid of points (Figure II.5 (c)), usually provided by an external entity that performs NWP.

The proposal of Gneiting et al. [11] was a pioneer in taking advantage of spatio-temporal data, collected by different production units. The authors introduced the regime-switching space-time method, which merges meteorological and statistical expertise to obtain probabilistic forecasts of wind resources, for two hours-ahead wind speed forecasting. This and



(a) Data for the location of the power plant

(b) Spatio-temporal power measurements and/or weather forecasts from a set of power plants

(c) Spatio-temporal weather forecasts from a grid around the power plant location

Figure II.5: Illustration of three major geographical points for data collection.

other similar work motivated the recent research that explores spatio-temporal information.

In 2010, Tastu et al. used Vector AutoRegressive (VAR) models to capture the errors in wind power forecasts which propagate in space and time under the influence of meteorological conditions [13, 78]. Later, in 2015, a sparse autoregressive coefficient matrix constructed by expert knowledge and partial correlation analysis is considered in [79]. The main limitation of these approaches is that automatic feature selection is not performed. A very-short-term sparse-VAR approach is proposed in [80] within a parametric framework based on the logit-normal distribution [67]. In [81], the Least Absolute Shrinkage and Selection Operator (LASSO) is combined with VAR models and Alternating Direction Method of Multipliers (ADMM) has been used to optimize the model.

Concerning machine learning approaches, Kou et al. used an online sparse Bayesian model based on a warped Gaussian process to generate probabilistic wind power forecasts [82]. The explanatory variables of the model come from multiple nearby reference sites and NWP data. In the same vein, but for photovoltaic predictions, a multilayer perceptron neural network is implemented using local meteorological data and measurements of neighboring photovoltaic systems as inputs [83]. In [84], Bessa et al. uses component-wise gradient boosting to explore observations from distributed photovoltaic systems in a smart grid environment. In [85], deep learning models are applied using fully connected multilayer perceptrons and convolutional neural networks that can take advantage of the spatial and feature structure of the NWP patterns. Hierarchical forecasting models to leverage turbine-level data were proposed in [86], which used deterministic power forecasts from the turbine-level as explanatory variables in a wind farm level forecasting model, as well as an alternative based on a spatial multivariate probabilistic forecast of all turbines.

The main limitation of these approaches is that production units might have different owners, which introduces new challenges like data privacy and monetization, which is not addressed in the existing literature.

Regarding the scheme illustrated in Figure II.5 (c), Andrade and Bessa [12] described a forecasting framework to explore information from a NWP applied to both wind and solar energy [12]. Alternative models are also being applied to this problem, most notably deep learning techniques such as convolutional neural networks or long short-term memory networks [87, 88]. Convolution neural networks are widely used in image processing problems aiming to automatically extract relevant information. Since NWP are provided for a grid of points, each of these geographical points can be interpreted as a pixel of an image, and each weather variable is analogous to the color channels. That said, convolution neural networks extract relevant data from NWP which are then used to perform power forecasting.

However, none of these works consider the forecast of extreme quantiles (e.g., quantiles with a nominal percentage between 0.01% and 1%, or between 99% and 99.99%). Indeed, little research was conducted to predict extreme conditional quantiles and none that makes use of spatio-temporal information.

Finally, to generate RES power trajectories for multiple locations, probabilistic spatial models with sparse Gaussian random fields are considered in [89, 90], but the computational time to obtain accurate results is high. In [90], the emphasis is placed on generating space-time trajectories for the wind power generation using a Gaussian copula approach. This study considers the problem of obtaining a joint multivariate predictive density to describe wind power generation, at several distributed locations and for several successive time horizons, from the set of marginal predictive densities, targeting each location and each time horizon individually.

II.2 Forecasting Skill Evaluation Metrics

Evaluation of forecasts is also of great importance to both deterministic and probabilistic forecasting methods. The metrics commonly used are described in this section:

Point Forecasts

The evaluation metrics for point forecasting compares a set of point forecasts $\hat{y}_t$ to corresponding observations y_t, $\forall t \in \{1, \dots, T\}$. Naturally, a good forecast $\hat{y}_t$ should be as close as possible to y_t. Usual metrics include MAE, MSE, RMSE, MBE, and MAPE [91].

Mean Absolute Error (MAE) averages the absolute differences between actual and predicted values, giving all individual deviations equal weight,

$$\mathrm{MAE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\sum_{t=1}^{T} |\hat{y}_t - y_t|}{T}. \tag{II.8}$$

Similarly, **Mean Square Error (MSE)** averages the squared differences between actual and predicted values,

$$\mathrm{MSE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}, \tag{II.9}$$

meaning the units of this metric are squared. Based on MSE, the **Root Mean Squared Error (RMSE)** simply considers the square root of MSE,

$$\mathrm{RMSE}(\mathbf{y}, \hat{\mathbf{y}}) = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}}, \tag{II.10}$$

providing a more interpretable statistic, since it has the same units as the variable being predicted. Notice that RMSE is more robust, when compared to MAE, in dealing with large deviations that are especially undesirable, giving the user the ability to identify outliers. Also, **normalized RMSE** is applied to evaluate overall deviations by taking into account the amplitude of the actual values,

$$\mathrm{NRMSE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\mathrm{RMSE}}{\max(\{y_t\}_{t=1}^{T}) - \min(\{y_t\}_{t=1}^{T})} \times 100. \tag{II.11}$$

Forecasting bias is measured by **Mean Bias Error (MBE)**,

$$\mathrm{MBE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\sum_{t=1}^{T}(\hat{y}_t - y_t)}{T}. \tag{II.12}$$

The MBE is usually not used as a measure of the model error as high individual errors in prediction can also produce a low MBE.

All these metrics are based on equally weighted averages. In contrast, **Mean Absolute Percentage Error (MAPE)** is a standard prediction technique that measures the accuracy of forecasting by weighting the absolute deviations according to the actual values,

$$\mathrm{MAPE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\sum_{t=1}^{T} |\frac{\hat{y}_t - y_t}{y_t}|}{T}, \tag{II.13}$$

but it is not used in RES forecasting due to zeros.

Let us assume the existence of two prediction models A and B. After computing some of the discussed metrics, if the values for model A and B are similar, it is difficult to decide whether the result is due to chance or decisive. To solve this problem, the Diebold-Mariano (DM) test has been proposed.

DM test [92] compares the forecast accuracy of two forecast methods. Let $\hat{y}_t^A$ and $\hat{y}_t^B$ be the forecasting series for model A and B, respectively. Supposing the forecasting errors are $e_t^A = y_t - \hat{y}_t^A$ and $e_t^B = y_t - \hat{y}_t^B$, the accuracy of each forecast is measured by a function $\mathcal{L}$ that can be the MAE, RMSE, etc. To determine whether one forecasting model predicts more accurately than another, the equal accuracy hypothesis is tested. Mathematically, the null hypothesis is

$$\mathrm{H}_0\colon\ \mathbb{E}[d_t] = 0, \tag{II.14}$$

where $d_t = \mathcal{L}(e_t^A) - \mathcal{L}(e_t^B)$, and the alternative hypothesis is

$$\mathrm{H}_1\colon\ \mathbb{E}[d_t] \neq 0. \tag{II.15}$$

The empirical value for $\mathbb{E}[d_t]$ is the sample mean

$$\bar{d} = \frac{1}{T} \sum_{t=1}^{T} \left[\mathcal{L}(e_t^A) - \mathcal{L}(e_t^B)\right]. \tag{II.16}$$

Under the assumption that the loss differential is a covariance stationary series, the sample average, $\bar{d}$, converges asymptotically to a normal distribution and the DM test statistic is

$$\mathrm{DM} = \frac{\bar{d}}{\sqrt{\frac{2\pi \hat{f}_d(0)}{T}}} \xrightarrow{distribution} \mathcal{N}(0, 1) \tag{II.17}$$

where $2\pi \hat{f}_d(0)$ is the consistent estimate of the asymptotic variance of $\sqrt{T}\bar{d}$ based on sample autocovariance [92]. Then, assuming a significance level of 5%, the null hypothesis is rejected if $|\mathrm{DM}| > 1.96$.

The DM test can also be applied to test the null hypothesis against the alternative hypothesis that model B performs better than A, $\mathrm{H}_1\colon\ \mathbb{E}[d_t] > 0$. In this case, the null hypothesis is rejected for a significance level of 5% if $\mathrm{DM} > 1.64$.

Probabilistic Forecasts

The evaluation metrics for probabilistic forecasting through quantiles compare the set of conditional quantiles forecasts $\hat{q}_t^\tau = \hat{Q}(\tau|\mathbf{x}_t)$ to corresponding observations y_t, where $\hat{Q}$ is the estimator and $\mathbf{x}_t$ are the covariates for time t, $\forall t \in \{1, \dots, T\}$. Common metrics to evaluate how well quantiles $\hat{q}_t^\tau$ represent the distribution of Y_t are [93]: calibration, sharpness, Continuous Ranked Probability Score (CRPS), and pinball loss function.

Calibration measures the mismatch between the empirical probabilities (or long-run quantile proportions) and nominal (or subjective) probabilities, e.g. a quantile with nominal proportion 0.25 should contain 25% of the observed values lower or equal to its value. For each quantile τ, the observed proportion $\hat{\alpha}(\tau)$ of observations bellow the estimated quantile is

$$\hat{\alpha}(\tau) = \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}_{y_t \leq \hat{q}_t^\tau}. \tag{II.18}$$

Sharpness measures the "degree of uncertainty" of the probabilistic forecast, which numerically corresponds to compute the average interval size between two symmetric quantiles, e.g., 0.10 and 0.90 centered in the 0.50 quantile (median), as follows

$$\text{sharp}(\tau) = \frac{1}{T}\sum_{t=1}^{T} \hat{q}_t^{1-\tau} - \hat{q}_t^\tau, \tag{II.19}$$

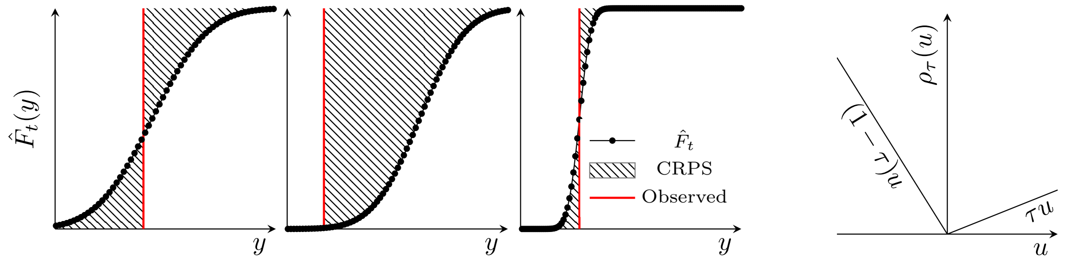
for $\tau \in [0, 0.5]$.

When assessing the quality of probabilistic forecasts, we are guided by the paradigm that probabilistic forecasts strive to maximize the sharpness of the predictive distributions under the constraint of calibration [93]. To assess this trade-off CRPS has been proposed, as described in the next paragraph.

CRPS evaluates the forecasting skill of a probabilistic forecast in terms of the entire predictive CDF, using an omnibus scoring function that simultaneously addresses calibration and sharpness [94]. Let y_t be the observation, and $\hat{F}_t$ the CDF associated with an empirical probabilistic forecast,

$$\text{CRPS}(\hat{F}_t, y_t) = \int_{-\infty}^{\infty} \left(\hat{F}_t(z) - H(z - y_t)\right)^2 dz, \tag{II.20}$$

where H is the Heaviside function. A graphical representation of CRPS is depicted in Figure II.6 (a).



(a) CRPS

(b) Pinball loss function

Figure II.6: Probabilistic forecasting metrics.

Although CRPS is very popular in evaluating the quality of CDF forecast, recent work in [95] concluded that the mean of the CRPS is unable to discriminate forecasts with different tails behavior since it tends to benefit distributions with smaller uncertainty intervals, even if the calibration is poor. A more suitable scoring rule, following the suggestion in [94], is the *pinball function* or quantile loss in (II.21).

Pinball loss function or quantile score (depicted in Figure II.6 (b)) assess the accuracy of each quantile forecast $\hat{q}_t^\tau$ by weighting the differences, between $\hat{q}_t^\tau$ and y_t, according to its sign and τ value [96],

$$\rho_\tau(y_t, \hat{q}_t^\tau) = \begin{cases} \tau \left[y_t - \hat{q}_t^\tau\right], & \text{if } y_t > \hat{q}_t^\tau, \\ (\tau - 1) \left[y_t - \hat{q}_t^\tau\right], & \text{otherwise.} \end{cases} \tag{II.21}$$

Smaller the value of the quantile score, the better the model when forecasting quantile τ.

II.3 Theoretical Background: Statistical Learning Models

The original contributions of this PhD thesis are developed on top of, or compared to, existing statistical learning methods. In what follows, a mathematical description of these relevant models is provided. Section II.3.1 describes the conditional and non-conditional quantiles forecasting models which are useful to understand the contents of Chapters 1 and 4. Section II.3.2 describes the VAR model, a successful collaborative forecasting model, as well as the most common estimators and employed optimization algorithm, essential for Chapters 2 and 3.

II.3.1 Conditional Quantile Forecasting

Again, the two classes of conditional quantile forecast are the non-parametric methods and the parametric methods.

Non-parametric Methods

The following non-parametric methods have been used:

Linear Quantile Regression. The QR model [96] estimates the conditional quantile function of a random variable Y given a set of covariates X_1, X_2,..., X_p,

$$Q^{\mathrm{QR}}(\tau|X) \approx \beta_0(\tau) + \beta_1(\tau)X_1 + \cdots + \beta_p(\tau)X_p, \tag{II.22}$$

for the nominal proportion $\tau \in [0, 1]$, by minimizing

$$\hat{\boldsymbol{\beta}}(\tau) = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{T} \rho_\tau \Big(y_i - \beta_0(\tau) - \sum_{j=1}^{p} \beta_j(\tau) x_{ij} \Big), \tag{II.23}$$

where $\hat{\boldsymbol{\beta}}(\tau) = (\hat{\beta}_0(\tau), \dots, \hat{\beta}_p(\tau))$ are unknown coefficients depending on τ, and $\rho_\tau(u)$ is the *pinball loss function* described in Section II.2.

Gradient Boosting Trees. A GBT model for quantile forecasting is constructed by combining base learners (i.e., regression trees), f_j, recurrently on modified data,

$$Q_j^{\mathrm{GBT}}(\tau|X) \approx Q_{j-1}^{\mathrm{GBT}}(\tau|X) + \eta f_j(\tau|X). \tag{II.24}$$

with each regression tree f_j fitted using the negative gradients as target variable, and as part of an additive training process to minimize the *pinball loss function*

$$\hat{f}_j(\tau|X) = \arg\min_{f_j} \sum_{i=1}^{T} \rho_\tau\big(y_i, \hat{Q}_{j-1}^{\mathrm{GBT}}(\tau|\mathbf{x}_i) + \eta f_j(\tau|\mathbf{x}_i)\big). \tag{II.25}$$

The initial model $\hat{Q}_1^{\mathrm{GBT}}$ is typically the unconditional τ-quantile of $\mathbf{y}$. The challenge of GBT is to tune the different hyperparameters, which are related with the regression trees and the boosting process — in this work they are estimated by using Bayesian Optimization algorithm, see [12] for more details.

Rearrangement of quantiles

Since both QR and GBT independently solve an optimization problem for each quantile τ, quantile crossing may happen, i.e., $Q(\tau_1|\mathbf{x}) < Q(\tau_2|\mathbf{x})$ for $\tau_1 > \tau_2$. Post-processing is applied to the model's output to ensure that the estimated cumulative function is monotonically non-decreasing. We can monotonize the function by considering the proportion of times the quantile $Q(\tau|\mathbf{x})$ is bellow a certain y, mathematically provided by the CDF

$$F(y|\mathbf{x}) = \int_0^1 \mathbf{1}_{Q(\tau|\mathbf{x})\leq y} d\tau \tag{II.26}$$

which is monotone at the level y, and then use its quantile function

$$\tilde{Q}(\tau|\mathbf{x}) = F^{-1}(\tau|\mathbf{x}) \tag{II.27}$$

which is monotone in τ [97].

Parametric Methods for Extreme Quantiles

The following parametric methods have been used, and combined with non-parametric models in order to estimate the entire CDF, as depicted in Figure II.7:

Exponential function. In [98], distribution's tails of wind power are approximated by exponential functions. Given the estimated conditional quantiles for nominal proportion between 0.05 and 0.95, the extreme quantiles are computed as

$$\hat{Q}^{\exp}(\tau|\mathbf{x}) = \begin{cases} \hat{Q}(0.05|\mathbf{x})^{\frac{\log(\frac{0.05}{\rho})}{\log(\frac{\tau}{\rho})}}, & \tau < 0.05, \\ C\left(1 - \left(1 - \frac{\hat{Q}(0.95|\mathbf{x})}{C}\right)^{\frac{\log(\frac{1-0.95}{\rho})}{\log(\frac{1-\tau}{\rho})}}\right), & \tau > 0.95, \end{cases} \tag{II.28}$$

where ρ corresponds to the thickness parameter for the exponential extrapolation and C is the installed capacity. Since the lower and upper tails may have different behaviors, ρ is independently estimated for each tail by maximum likelihood [8].

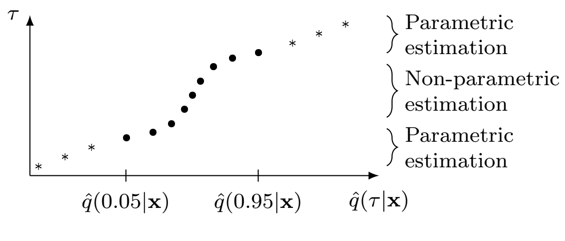

τ
Parametric estimation
Non-parametric estimation
Parametric estimation
$\hat{q}(0.05|\mathbf{x})$
$\hat{q}(0.95|\mathbf{x})$
$\hat{q}(\tau|\mathbf{x})$

Figure II.7: Estimation of the entire CDF.

Hill-based methods. In [69] and [99], a QR model is combined with EVT estimators. First, a local QR model is used to estimate the conditional quantiles $\tau_j = j/(T+1)$, denoted as $\hat{Q}^{\mathrm{QR}}(\tau_j|\mathbf{x})$, $j \in \{1, ..., T - [T^\eta]\}$, for some $0 < \eta < 1$, being [u] the integer part of u, and T the number of observations. Then, using these values, extreme quantiles are computed through an adaptation of Weissman's estimator,

$$\hat{Q}^{\mathrm{W}}(\tau|\mathbf{x}) = \left(\frac{1-\tau_{T-k}}{1-\tau}\right)^{\hat{\gamma}(\mathbf{x})} \hat{Q}^{\mathrm{QR}}(\tau_{T-k}|\mathbf{x}), \tag{II.29}$$

where $\hat{\gamma}(\mathbf{x})$ is based on Hill's estimator,

$$\hat{\gamma}(\mathbf{x}) = \frac{1}{k - [T^\eta]} \sum_{j=[T^\eta]}^{k} \log \frac{\hat{Q}^{\mathrm{QR}}(\tau_{T-j}|\mathbf{x})}{\hat{Q}^{\mathrm{QR}}(\tau_{T-k}|\mathbf{x})}. \tag{II.30}$$

In EVT, the selection of k is an important and challenging problem. The value k represents the effective sample size for tail extrapolation. A smaller k leads to estimators with larger variance, while larger k results in more bias, when estimating $\gamma(\mathbf{x})$. In practice, a commonly used heuristic approach for choosing k is to plot the estimated γ versus k and then choose a suitable k corresponding to the first stable part of the plot [100], see Figure II.8.

In [99], the response variable of the QR model is the power transformation $\Lambda_\lambda(.)$ of Y that aims to improve the linear relation with $\mathbf{x}$. That is,

$$\Lambda_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0, \\ \log(y), & \text{if } \lambda = 0. \end{cases} \tag{II.31}$$

For this approach, k is estimated to minimize

$$\arg\min_{k \geq 1} \sum_{i=1}^{T} \hat{\lambda}\hat{\gamma}(\mathbf{x}_i) - \hat{\gamma}^*(\mathbf{x}_i), \tag{II.32}$$

where

$$\hat{\gamma}^*(\mathbf{x}) = M_{0,T}^{(1)} + 1 - \frac{1}{2}\left(1 - \frac{(M_{0,T}^{(1)})^2}{M_{0,T}^{(2)}}\right)^{-1}, \tag{II.33}$$

$$M_{0,T}^{(i)} = \frac{1}{k - [T^\eta]} \sum_{j=[T^\eta]}^{k} \left(\log \frac{\hat{Q}^{\mathrm{QR}}(\tau_{T-j})}{\hat{Q}^{\mathrm{QR}}(\tau_{T-k})}\right)^i. \tag{II.34}$$

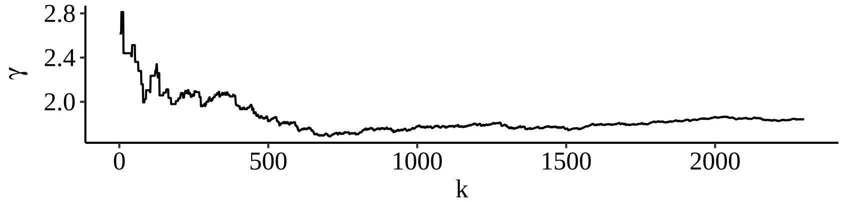


Figure II.8: Illustration of γ value in function of k. The first stable part of the plot happens when $k \approx 700$.

Peaks-over-threshold (POT) method with truncation. Since RES generation is limited between 0 and installed capacity C, we observe the truncated random variable Y, $Y \leq C$. The work in [101] provides an estimator for the extreme quantiles by using a random sample of Y, with independent and identically distributed observations, i.e., does not consider that Y is conditioned by covariates $\mathbf{x}$. The POT method [102] is adapted to estimate extreme quantiles from a generalized Pareto distribution (GPD) distribution affected by truncation at point C. The quantiles for Y are estimated by

$$\hat{Q}^{\text{tGPD}}(1-\tau) = Y_{T-k,T} + \frac{\hat{\sigma}_k}{\hat{\xi}_k}\left(\left[\frac{\hat{D}_{C,k} + \frac{(k+1)}{(T+1)}}{\tau(\hat{D}_{C,k}+1)}\right]^{\hat{\xi}_k} - 1\right), \tag{II.35}$$

where $Y_{1,T} < \cdots < Y_{T,T}$ is the ordered sample, $\hat{\xi}_k$ and $\hat{\sigma}_k$ are the maximum likelihood estimates adapted for truncation, and $\hat{D}_C$ the truncation odds estimator

$$\hat{D}_{C,k} = \max\left\{0, \frac{k}{T}\frac{(1+(\hat{\xi}_k/\hat{\sigma}_k)E_{1,k})^{-1/\hat{\xi}_k} - \frac{1}{k}}{1-(1+(\hat{\xi}_k/\hat{\sigma}_k)E_{1,k})^{-1/\hat{\xi}_k}}\right\}, \tag{II.36}$$

with $E_{j,k} = Y_{T-j+1,T} - Y_{T-k,T}$.

II.3.2 Collaborative Forecasting with VAR

This section presents the VAR model, a model for the analysis of multivariate time series and collaborative forecasting.

Forecasting problem formulation

Let $\{\mathbf{y}_t\}_{t=1}^{T}$ be an n-dimensional multivariate time series, where n is the number of data owners. Then, $\{\mathbf{y}_t\}_{t=1}^{T}$ follows a VAR model with p lags, represented as $\text{VAR}_n(p)$, when the following relationship holds:

$$\mathbf{y}_t = \boldsymbol{\eta} + \sum_{\ell=1}^{p} \mathbf{y}_{t-\ell}\mathbf{B}^{(\ell)} + \boldsymbol{\varepsilon}_t \ , \tag{II.37}$$

for $t = 1, \ldots, T$, where $\boldsymbol{\eta} = [\eta_1, \ldots, \eta_n]$ is the constant intercept (row) vector, $\boldsymbol{\eta} \in \mathbb{R}^n$; $\mathbf{B}^{(\ell)}$ represents the coefficient matrix at lag $\ell = 1, ..., p$, $\mathbf{B}^{(\ell)} \in \mathbb{R}^{n\times n}$, and the coefficient associated with lag ℓ of time series i (to estimate time series j) is positioned at (i, j) of $\mathbf{B}^{(\ell)}$, for $i, j = 1, ..., n$; and $\boldsymbol{\varepsilon}_t = [\varepsilon_{1,t}, \ldots, \varepsilon_{n,t}]$, $\boldsymbol{\varepsilon}_t \in \mathbb{R}^n$, indicates a white noise vector that is independent and identically distributed with mean zero and nonsingular covariance matrix. By simplification, $\mathbf{y}_t$ is assumed to follow a centered process, $\boldsymbol{\eta} = \mathbf{0}$, i.e., as a vector of zeros of appropriate dimensions. A compact representation of a $\text{VAR}_n(p)$ model reads as follows:

$$\mathbf{Y} = \mathbf{Z}\mathbf{B} + \mathbf{E} \ , \tag{II.38}$$

where

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_T \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}^{(1)} \\ \vdots \\ \mathbf{B}^{(p)} \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_T \end{bmatrix}, \text{ and } \mathbf{E} = \begin{bmatrix} \boldsymbol{\varepsilon}_1 \\ \vdots \\ \boldsymbol{\varepsilon}_T \end{bmatrix},$$

are obtained by joining the vectors row-wise, and defining, respectively define the $T \times n$ response matrix, the $np \times n$ coefficient matrix, the $T \times np$ covariate matrix, and the $T \times n$ error matrix, with $\mathbf{z}_t = [\mathbf{y}_{t-1}, \ldots, \mathbf{y}_{t-p}]$.

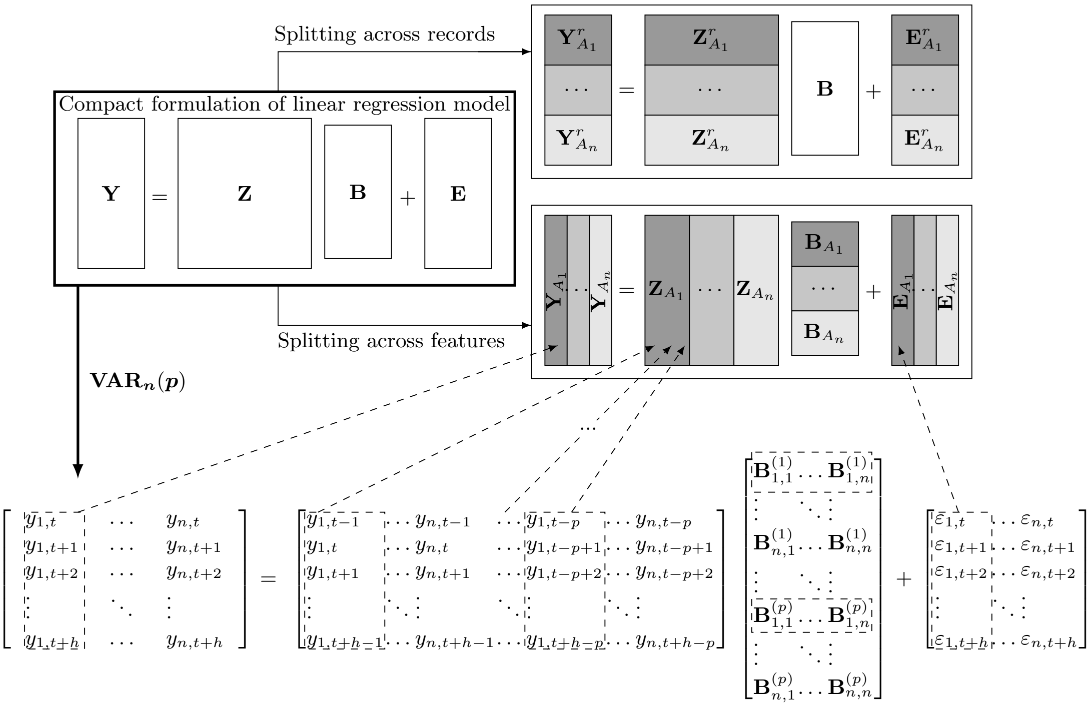

Splitting across records
Compact formulation of linear regression model
Y
=
Z
B
+
E
$\mathbf{Y}^r_{A_1}$
$\mathbf{Z}^r_{A_1}$
$\mathbf{E}^r_{A_1}$
$\mathbf{Y}^r_{A_n}$
$\mathbf{Z}^r_{A_n}$
$\mathbf{E}^r_{A_n}$
$\mathbf{Y}_{A_1}$
$\mathbf{Y}_{A_n}$
$\mathbf{Z}_{A_1}$
$\mathbf{Z}_{A_n}$
$\mathbf{B}_{A_1}$
$\mathbf{B}_{A_n}$
$\mathbf{E}_{A_1}$
$\mathbf{E}_{A_n}$
Splitting across features
$\mathbf{VAR}_n(p)$
$y_{1,t}$ ⋯ $y_{n,t}$
$y_{1,t+h}$ ⋯ $y_{n,t+h}$
$y_{1,t-1}$ ⋯ $y_{n,t-1}$ ⋯ $y_{1,t-p}$ ⋯ $y_{n,t-p}$
$y_{1,t+h-1}$ ⋯ $y_{n,t+h-1}$ ⋯ $y_{1,t+h-p}$ ⋯ $y_{n,t+h-p}$
$\mathbf{B}^{(1)}_{1,1}$ ⋯ $\mathbf{B}^{(1)}_{1,n}$
$\mathbf{B}^{(1)}_{n,1}$ ⋯ $\mathbf{B}^{(1)}_{n,n}$
$\mathbf{B}^{(p)}_{1,1}$ ⋯ $\mathbf{B}^{(p)}_{1,n}$
$\mathbf{B}^{(p)}_{n,1}$ ⋯ $\mathbf{B}^{(p)}_{n,n}$
$\varepsilon_{1,t}$ ⋯ $\varepsilon_{n,t}$
$\varepsilon_{1,t+h}$ ⋯ $\varepsilon_{n,t+h}$

Figure II.9: Common data division structures and VAR model.

Notice that the VAR formulation adopted in this chapter is not the usual $\mathbf{Y}^\top = \mathbf{B}^\top\mathbf{Z}^\top + \mathbf{E}^\top$, because a large proportion of the literature on privacy-preserving techniques derives from the standard linear regression problem, in which each row is a record and each column is a feature.

Notwithstanding the high potential of the VAR model for collaborative forecasting, namely by linearly combining time series from different data owners, data privacy or confidentiality issues might hinder this approach. For instance, renewable energy companies, competing in the same electricity market, will never share their electrical energy production data, even if this leads to a forecast error improvement in all individual forecasts.

For classical linear regression models, there are several techniques for estimating coefficients without sharing private information. However, in the VAR model, the data are divided by features, i.e., the data owners (denoted by $A_i, i \in \{1, \dots, n\}$) observe different features of the same records, as illustrated at the bottom of Figure II.9, and the variables to be forecasted are also covariates. This is challenging for privacy-preserving techniques (especially because it is also necessary to protect the data matrix $\mathbf{Y}$, as illustrated in Figure II.10). In what follows, when defining a VAR model, $\mathbf{Y}_{A_i} \in \mathbb{R}^{T\times 1}$ and $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$ respectively denote the target and covariate matrix for the ith data owner. Therefore, the covariates and target matrices are obtained by joining the individual matrices column-wise, i.e., $\mathbf{Z} = [\mathbf{Z}_{A_1}, \dots, \mathbf{Z}_{A_n}]$ and $\mathbf{Y} = [\mathbf{Y}_{A_1}, \dots, \mathbf{Y}_{A_n}]$. For distributed computation, the coefficient matrix of data owner i is denoted by $\mathbf{B}_{A_i} \in \mathbb{R}^{p\times n}, \forall i \in \{1, \dots, n\}$.

Model sparsity with LASSO

Commonly, when the number of covariates included, np, is substantially smaller than the length of the time series, T, the VAR model can be fitted using multivariate least squares

$$\begin{array}{c|cccccc}
\overbrace{\qquad\qquad}^{\text{Y of } i\text{th data owner}} & \multicolumn{6}{c}{\overbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}^{\text{covariates values of } i\text{th data owner}}} \\
y_{i,t} & y_{i,t-1} & y_{i,t-2} & y_{i,t-3} & \cdots & y_{i,t-p+1} & y_{i,t-p} \\
y_{i,t+1} & y_{i,t} & y_{i,t-1} & y_{i,t-2} & \cdots & y_{i,t-p+2} & y_{i,t-p+1} \\
y_{i,t+2} & y_{i,t+1} & y_{i,t} & y_{i,t-1} & \cdots & y_{i,t-p+3} & y_{i,t-p+2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
y_{i,t+h} & y_{i,t+h-1} & y_{i,t+h-2} & y_{i,t+h-3} & \cdots & y_{i,t+h-p+1} & y_{i,t+h-p}
\end{array}$$

Figure II.10: Illustration of the data used by the ith data owner when fitting a VAR model.

solution, given by

$$\hat{\mathbf{B}}_{\text{LS}} = \arg\min_{\mathbf{B}} \left(\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 \right) \ , \tag{II.39}$$

where $\|.\|_r$ represents both vector and matrix L_r norms. However, in collaborative forecasting, as the number of data owners increases, as well as the number of lags, it becomes crucial to use regularization techniques such as LASSO to introduce sparsity into the coefficient matrix estimated by the model. In the standard LASSO-VAR approach (see [103] for different variants of the LASSO regularization in the VAR model), the coefficients are given by

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda \|\mathbf{B}\|_1 \right), \tag{II.40}$$

where $\lambda > 0$ is a scalar penalty parameter.

Distributed optimization with ADMM

With the addition of the LASSO regularization term, the convex objective function in (II.40) becomes non-differentiable, limiting the variety of optimization techniques that can be employed. In this domain, the ADMM (which is detailed in what follows) is a widespread and computationally efficient technique that enables parallel estimations for data divided by features. In what follows, a description of the ADMM algorithm is provided, as well as its application to estimate LASSO-VAR.

ADMM algorithm. The ADMM is efficient and well suited for distributed convex optimization, in particular for large-scale statistical problems [104]. Let E be a convex forecast error function between the true values $\mathbf{Y}$ and the forecasted values given by the model $\hat{\mathbf{Y}} = f(\mathbf{B}, \mathbf{Z})$ using a set of covariates $\mathbf{Z}$ and coefficients $\mathbf{B}$, and let R be a convex regularization function. The ADMM method [104] solves the optimization problem

$$\min_{\mathbf{B}} E(\mathbf{B}) + R(\mathbf{B}), \tag{II.41}$$

by splitting $\mathbf{B}$ into two variables ($\mathbf{B}$ and $\mathbf{H}$),

$$\min_{\mathbf{B},\mathbf{H}} E(\mathbf{B}) + R(\mathbf{H}) \text{ subject to } \mathbf{A}\mathbf{B} + \mathbf{C}\mathbf{H} = \mathbf{D}, \tag{II.42}$$

and using the related augmented Lagrangian function formulated with dual variable $\mathbf{U}$,

$$L(\mathbf{B}, \mathbf{H}, \mathbf{U}) = E(\mathbf{B}) + R(\mathbf{H}) + \mathbf{U}^\top(\mathbf{A}\mathbf{B}+\mathbf{C}\mathbf{H} - \mathbf{D}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{B} + \mathbf{C}\mathbf{H} - \mathbf{D}\|_2^2. \tag{II.43}$$

The quadratic term $\frac{\rho}{2}\|\mathbf{A}\mathbf{B} + \mathbf{C}\mathbf{H} - \mathbf{D}\|_2^2$ provides theoretical convergence guarantees because it is strongly convex. This implies mild assumptions on the objective function. Even if the original objective function is convex, the augmented Lagrangian is strictly convex (in some cases strongly convex) [104].

The ADMM solution is estimated by the following iterative system:

$$\begin{cases} \mathbf{B}^{k+1} := \arg\min_{\mathbf{B}} L(\mathbf{B}, \mathbf{H}^k, \mathbf{U}^k) \\ \mathbf{H}^{k+1} := \arg\min_{\mathbf{H}} L(\mathbf{B}^{k+1}, \mathbf{H}, \mathbf{U}^k) \\ \mathbf{U}^{k+1} := \mathbf{U}^k + \rho(\mathbf{A}\mathbf{B}^{k+1} + \mathbf{C}\mathbf{H}^{k+1} - \mathbf{D}). \end{cases} \tag{II.44}$$

For data split by records, the data owners observe the same features for different groups of samples as illustrated on top of Figure II.9. In these cases, the consensus problem splits primal variables $\mathbf{B}$ and separately optimizes the decomposable cost function $E(\mathbf{B}) = \sum_{i=1}^{n} E_i(\mathbf{B}_{A_i})$ for all data owners under global consensus constraints. Considering that the sub-matrix $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times np}$ of $\mathbf{Z} \in \mathbb{R}^{T \times np}$ corresponds to the local data of the $i-$th data owner, the coefficients $\mathbf{B}_{A_i} \in \mathbb{R}^{np \times n}$ are given by

$$\begin{aligned} &\arg\min_{\mathbf{\Gamma}} \sum_i E_i(\mathbf{B}_{A_i}) + R(\mathbf{H}) \\ &\text{s.t. } \mathbf{B}_{A_1} - \mathbf{H} = \mathbf{0},\ \mathbf{B}_{A_2} - \mathbf{H} = \mathbf{0}, \dots,\ \mathbf{B}_{A_n} - \mathbf{H} = \mathbf{0}\ , \end{aligned} \tag{II.45}$$

where $\mathbf{\Gamma} = \{\mathbf{B}_{A_1}, \dots, \mathbf{B}_{A_n}, \mathbf{H}\}$. In this case, $E_i(\mathbf{B}_{A_i})$ measures the error between the true values $\mathbf{Y}^r_{A_i}$ and the forecasted values given by the model $\hat{\mathbf{Y}}_{A_i} = f(\mathbf{B}_{A_i}, \mathbf{Z}^r_{A_i})$.

For data split by features, the sharing problem splits $\mathbf{Z}$ into $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}$, and $\mathbf{B}$ into $\mathbf{B}_{A_i} \in \mathbb{R}^{p \times n}$. Auxiliary $\mathbf{H}_{A_i} \in \mathbb{R}^{T \times n}$ are introduced for the ith data owner based on $\mathbf{Z}_{A_i}$ and $\mathbf{B}_{A_i}$. In this case, the sharing problem is formulated based on the decomposable cost function $E(\mathbf{B}) = E(\sum_{i=1}^{n} \mathbf{B}_{A_i})$ and $R(\mathbf{B}) = \sum_{i=1}^{n} R(\mathbf{B}_{A_i})$. Then, $\mathbf{B}_{A_i}$ is given by

$$\begin{aligned} &\arg\min_{\mathbf{\Gamma}'} E(\sum_i \mathbf{H}_{A_i}) + \sum_i R(\mathbf{B}_{A_i}) \\ &\text{s.t. } \mathbf{Z}_{A_1}\mathbf{B}_{A_1} - \mathbf{H}_{A_1} = \mathbf{0},\ \mathbf{Z}_{A_2}\mathbf{B}_{A_2} - \mathbf{H}_{A_2} = \mathbf{0}, \dots,\ \mathbf{Z}_{A_n}\mathbf{B}_{A_n} - \mathbf{H}_{A_n} = \mathbf{0}\ , \end{aligned} \tag{II.46}$$

where $\mathbf{\Gamma}' = \{\mathbf{B}_{A_1}, \dots, \mathbf{B}_{A_n}, \mathbf{H}_{A_1}, \dots, \mathbf{H}_{A_n}\}$. In this case, $E(\sum_{i=1}^{n} \mathbf{H}_{A_i})$ is related to the error between the true values $\mathbf{Y}$ and the forecasted values given by the model $\hat{\mathbf{Y}} = \sum_{i=1}^{n} f(\mathbf{B}_{A_i}, \mathbf{Z}_{A_i})$.

Undeniably, ADMM provides a desirable formulation for parallel computing [105].

LASSO-VAR optimization with ADMM. The ADMM formulation of the non-differentiable cost function associated to LASSO-VAR model in (II.40) solves the following optimization problem:

$$\min_{\mathbf{B},\mathbf{H}} \Big(\frac{1}{2}\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda\|\mathbf{H}\|_1\Big) \text{ subject to } \mathbf{H} = \mathbf{B}\ , \tag{II.47}$$

which differs from (II.40) by splitting $\mathbf{B}$ into two parts ($\mathbf{B}$ and $\mathbf{H}$). Thus, the objective function can be split in two distinct objective functions, $f(\mathbf{B}) = \frac{1}{2}\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2$ and $g(\mathbf{H}) = \lambda\|\mathbf{H}\|_1$. The augmented Lagrangian [104] of this problem is

$$L_\rho(\mathbf{B}, \mathbf{H}, \mathbf{W}) = \frac{1}{2}\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda\|\mathbf{H}\|_1 + \mathbf{W}^\top(\mathbf{B} - \mathbf{H}) + \frac{\rho}{2}\|\mathbf{B} - \mathbf{H}\|_2^2\ , \tag{II.48}$$

where $\mathbf{W}$ is the dual variable and $\rho > 0$ is the penalty parameter. The scaled form of this Lagrangian is

$$L_\rho(\mathbf{B}, \mathbf{H}, \mathbf{U}) = \frac{1}{2}\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda\|\mathbf{H}\|_1 + \frac{\rho}{2}\|\mathbf{B} - \mathbf{H} + \mathbf{U}\|^2 - \frac{\rho}{2}\|\mathbf{U}\|^2\ , \tag{II.49}$$

where $\mathbf{U} = (1/\rho)\mathbf{W}$ is the scaled dual variable associated with the constrain $\mathbf{B} = \mathbf{H}$. Hence, according to (II.44), the ADMM formulation for LASSO-VAR consists in the following iterations [81]:

$$\begin{cases} \mathbf{B}^{k+1} := \underset{\mathbf{B}}{\arg\min} \left(\frac{1}{2}\|\mathbf{Y}-\mathbf{Z}\mathbf{B}\|_2^2+\frac{\rho}{2}\|\mathbf{B}-\mathbf{H}^k+\mathbf{U}^k\|_2^2\right)=(\mathbf{Z}^\top\mathbf{Z}+\rho\mathbf{I})^{-1}(\mathbf{Z}^\top\mathbf{Y}+\rho(\overline{\mathbf{H}}^k-\mathbf{U}^k)) \\ \mathbf{H}^{k+1} := \underset{\mathbf{H}}{\arg\min} \left(\lambda\|\mathbf{H}\|_1+\frac{\rho}{2}\|\mathbf{B}^{k+1}-\mathbf{H}+\mathbf{U}^k\|_2^2\right) = S_{\lambda/\rho}(\mathbf{B}^{k+1}+\mathbf{U}^k) \\ \mathbf{U}^{k+1} := \mathbf{U}^k+\mathbf{B}^{k+1}-\mathbf{H}^{k+1}, \end{cases} \tag{II.50}$$

where $S_{\lambda/\rho}$ is the soft thresholding operator.

Concerning the LASSO-VAR model, and since data are naturally divided by features (i.e., $\mathbf{Y} = [\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$, $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$ and $\mathbf{B} = [\mathbf{B}_{A_1}^\top, \ldots, \mathbf{B}_{A_n}^\top]^\top$) and the functions $\|\mathbf{Y}-\mathbf{Z}\mathbf{B}\|_2^2$ and $\|\mathbf{B}\|_1$ are decomposable (i.e., $\|\mathbf{Y}-\mathbf{Z}\mathbf{B}\|_2^2 = \|\mathbf{Y}-\sum_{i=1}^n \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2$ and $\|\mathbf{B}\|_1 = \sum_{i=1}^n \|\mathbf{B}_{A_i}\|_1$), the model fitting problem (II.40) becomes the following:

$$\underset{\mathbf{\Gamma}}{\arg\min} \left(\frac{1}{2}\|\mathbf{Y}-\sum_{i=1}^n \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2+\lambda\sum_{i=1}^n \|\mathbf{B}_{A_i}\|_1\right), \tag{II.51}$$

$\mathbf{\Gamma} = \{\mathbf{B}_{A_1}, \ldots, \mathbf{B}_{A_n}\}$, which is rewritten as

$$\underset{\mathbf{\Gamma}'}{\arg\min} \left(\frac{1}{2}\|\mathbf{Y}-\sum_{i=1}^n \mathbf{H}_{A_i}\|_2^2+\lambda\sum_{i=1}^n \|\mathbf{B}_{A_i}\|_1\right) \text{ s.t. } \mathbf{B}_{A_1}\mathbf{Z}_{A_1} = \mathbf{H}_{A_1}, \ \ldots, \ \mathbf{B}_{A_n}\mathbf{Z}_{A_n} = \mathbf{H}_{A_n}, \tag{II.52}$$

$\mathbf{\Gamma}' = \{\mathbf{B}_{A_1}, \ldots, \mathbf{B}_{A_n}, \mathbf{H}_{A_1}, \ldots, \mathbf{H}_{A_n}\}$, while the corresponding distributed ADMM formulation [104, 81] is the one presented in the system of equations (II.53),

$$\mathbf{B}_{A_i}^{k+1} = \underset{\mathbf{B}_{A_i}}{\arg\min} \left(\frac{\rho}{2}\|\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k+\overline{\mathbf{H}}^k-\overline{\mathbf{Z}\mathbf{B}}^k-\mathbf{U}^k-\mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2+\lambda\|\mathbf{B}_{A_i}\|_1\right), \tag{II.53a}$$

$$\overline{\mathbf{H}}^{k+1} = \frac{1}{n+\rho}\left(\mathbf{Y}+\rho\overline{\mathbf{Z}\mathbf{B}}^{k+1}+\rho\mathbf{U}^k\right), \tag{II.53b}$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k+\overline{\mathbf{Z}\mathbf{B}}^{k+1}-\overline{\mathbf{H}}^{k+1}, \tag{II.53c}$$

where $\overline{\mathbf{Z}\mathbf{B}}^{k+1} = \frac{1}{n}\sum_{j=1}^n \mathbf{Z}_{A_j}\mathbf{B}_{A_j}^{k+1}$ and $\mathbf{B}_{A_i}^{k+1} \in \mathbb{R}^{p\times n}$, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$,$\mathbf{Y} \in \mathbb{R}^{T\times n}$, $\overline{\mathbf{H}}^k$,$\mathbf{U} \in \mathbb{R}^{T\times n}$, $\forall i \in \{1, \ldots, n\}$, and (II.53a) can be estimated by adapting (II.47) as

$$\underset{\mathbf{B}}{\arg\min} \left(\frac{1}{2}\|\hat{\mathbf{Y}}-\mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2+\hat{\lambda}\|\mathbf{H}_{A_i}\|_1\right) \text{ s.t. } \mathbf{H}_{A_i}=\mathbf{B}_{A_i}, \tag{II.54}$$

where $\hat{\mathbf{Y}}_{A_i} = \mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k+\overline{\mathbf{H}}^k-\overline{\mathbf{Z}\mathbf{B}}^k-\mathbf{U}^k$ and $\hat{\lambda} = \lambda/\rho$.

Although parallel computation is an appealing property for the design of a privacy-preserving approach, the ADMM is an iterative optimization process that requires intermediate calculations. Hence, careful analysis is needed to determine whether a confidentiality breach will occur after enough iterations.

Selection of the number of lags p

When considering autoregressive models, the number of lags to be used is commonly identified by the Partial AutoCorrelation Function (PACF) [106]. PACF gives the correlation between Y_t and $Y_{t-\ell}$ by removing the effect of the lags between Y_t and $Y_{t-\ell}$, $\ell > 1$.

Mathematicaly, when considering the regression of a target variable Y on covariates X_1, X_2, X_3, the partial correlation between Y and X_1 is computed as

$$\text{PACF}(Y, X_1) = \frac{\text{cov}(Y, X_1 | X_2, X_3)}{\sqrt{\text{var}(Y | X_2, X_3)\text{var}(X_1 | X_2, X_3)}}. \tag{II.55}$$

This can be computed as the correlation between the residuals of the regression of Y on X_2 and X_3 with the residuals of X_1 on X_2 and X_3. A small value (i.e., $\left[-1.96/\sqrt{T}, 1.96/\sqrt{T}\right]$) indicates that X_1 is not statistically relevant when predicting Y. Therefore, the partial correlation for multiple lags of multiple data owners can be recursively computed in order to identify the value of p.

A less sophisticated approach is to use cross-validation on a range of values for p, choosing the best p as judged by the metrics from Section II.2.

1 Extreme Conditional Quantile Forecasting

Abstract. Probabilistic forecast of distribution tails (quantiles with nominal proportion below 0.05 and above 0.95) is challenging for non-parametric approaches since data for extreme events are scarce. A poor forecast of extreme quantiles can have a high impact on various power system decision-aid problems. An alternative approach more robust to data sparsity is Extreme Value Theory (EVT), which uses parametric functions for modeling distribution's tails. In this chapter, we apply conditional EVT estimators to historical data by directly combining non-parametric models with a truncated generalized Pareto distribution. The parameters of a parametric function are conditioned by covariates such as wind speed/direction from a numerical weather predictions grid. The results for a synthetic dataset show that the proposed approach better captures the overall tails' behavior, with smaller deviations between real and estimated quantiles. The proposed method also outperforms state-of-the-art methods in terms of quantile score when evaluated using real data from a wind power plant located in Galicia, Spain, and a solar power plant in Porto, Portugal.

1.1 Introduction

The growing integration of Renewable Energy Sources (RES) brings new challenges to system operators and market players and robust forecasting models are crucial for handling variability and uncertainty. This has fomented a growing interest in RES probabilistic forecasting techniques and its integration in decision-aid under risk [107].

Many satisfying methods already exist to forecast RES generation quantiles with nominal proportion between 0.05 and 0.95, which can be parametric or non-parametric, as described in Section II.1.3 of Prologue II. While parametric models assume that data are generated from a known probability distribution (e.g., Gaussian, Beta) whose parameters are estimated from the data, the non-parametric models do not make any assumptions about the shape of the probability distribution.

The main advantage of parametric methods is that the distribution's shape only depends on a few parameters, resulting in a simplified estimation and consequently requiring low computational costs. However, the choice of the parametric function is not straightforward. On the other hand, non-parametric models require a large number of observations to achieve good performance. Therefore, when estimating quantiles below 0.05 and above 0.95, non-parametric models tend to have poor performance due to data sparsity. This suggests the combination of both approaches to forecast the conditional probability function: intermediate quantiles are estimated with a non-parametric model and the extreme quantiles (or tails) with a parametric approach.

A poor forecast of extreme quantiles can have a high impact in different decision-aid problems, in particular when decision-makers are highly risk-averse or the regulatory framework imposes high-security levels. For instance, when setting operating reserve requirements system operators usually define risk (e.g., loss of load probability) levels below 1% [7]; the accuracy forecast of the distribution's tails affects the decision quality of advanced RES bidding strategies that are based on risk metrics such as conditional value-at-risk [108]; dynamic line rating uncertainty forecasting for transmission grids also requires the use of low quantiles (e.g., 1%) [109]. Moreover, the generation of temporal and/or spatial-temporal trajectories (or random vectors) with a statistical method, such as the Gaussian copula [70], requires a full modeling of the distribution function and an accurate estimation of the tails avoids trajectories with "extreme" values. In all these use cases, it is important to underline that poor modeling of distribution' tails might lead to over- and under-estimation of risk and consequently to worst decisions. This impact can be measured by metrics such as the Value of the Right Distribution that measures the difference in the cost of the optimal solution, in stochastic programming, obtained with the forecasted and realized probability distribution [110].

By exploring concepts from EVT, which is dedicated to characterize the stochastic behavior of extreme values [100], the present chapter proposes a novel forecasting methodology, focused on improving the forecasting skill of the distribution's tails, which combines spatio-temporal information (obtained through feature engineering), a non-parametric method for quantiles in the central part of the distribution and the truncated generalized Pareto distribution (GPD) for the tails.

The remaining of this chapter is organized as follows. Section 1.2 presents related work and contributions. Section 1.3 proposes a novel forecasting method that combines a non-parametric model with a truncated GPD, based on the statistical background of non-parametric and parametric methods previously described in Section II.3.1 of Prologue II. Section 1.4 describes experiments and evaluates the proposed method. Concluding remarks are drawn in Section 1.5.

1.2 Related Work and Contributions

In [8] and [98], a Quantile Regression (QR) model is used to forecast the RES power quantiles from 0.05 to 0.95 and the distribution' tails are modeled using an exponential function. The exponential function requires the estimation of a single parameter that controls the tails' decay, the thickness parameter ρ. This parameter can be estimated by computing the mean of the observed power conditioned by the forecasted power, i.e., observed power is divided into equally populated bins according to forecasted power, then ρ is the average power associated to each bin. This procedure is not as flexible as those provided by an EVT estimator like GPD (used in this chapter), which models extreme events through distributions with two parameters (scale and shape), allowing it to estimate lightweight and heavier tails.

A two-stage EVT approach is proposed in [111] to estimate the extreme quantiles of a random variable Y conditioned by covariate X. First, the conditional quantiles are estimated with a local QR. Then, generalized extreme value distribution with a single parameter (i.e., extreme value index estimated using maximum likelihood) is applied to these non-parametrically estimated quantiles in order to construct an estimator for extreme quantiles. Similarly, the authors of [69] apply linear QR to estimate the intermediate conditional quantiles, which are then extrapolated to the upper tails by applying EVT estimators (e.g., Hill estimator) for heavy-tailed distributions (GPD is assumed). However, the conditional quantiles of Y are assumed to have a linear relation with X at the tails, which may be too restrictive in real-world applications. In order to overcome this limitation, the approach proposed in [99] works by first finding an appropriate power

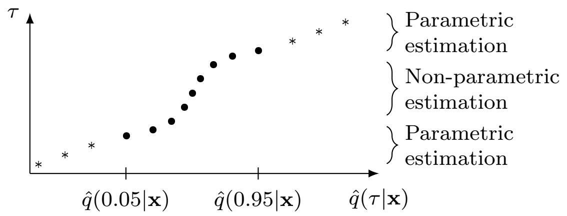

τ
Parametric estimation
Non-parametric estimation
Parametric estimation
$\hat{q}(0.05|\mathbf{x})$
$\hat{q}(0.95|\mathbf{x})$
$\hat{q}(\tau|\mathbf{x})$

Figure 1.1: The proposed method uses different estimators for intermediate and extreme quantiles.

transformation of Y, then estimating the intermediate conditional quantiles of the transformed Y using linear QR and finally extrapolating these estimates to extreme tails with EVT estimators. In the end, these quantiles are transformed back to the original scale.

More importantly, existing works only apply EVT as a post-processing step over a set of quantiles first estimated (or forecasted) by a non-parametric method [69]. However, since non-parametric models can suffer from high variability at the tails, the performance of EVT estimators may be compromised. In order to overcome this problem, we restrict non-parametric estimation to the intermediate quantiles, as depicted in Figure 1.1. This estimation is then used to guide the parametric model by rating historically similar periods conditioned by the covariates.

Finally, two works proposed the use of spatio-temporal data in RES probabilistic forecasting: a combination of Gradient Boosting Tree (GBT) with feature engineering techniques to extract information from a grid of Numerical Weather Prediction (NWP) [12]; hierarchical forecasting models to leverage turbine-level data [86]. Both works do not deal with or propose a specific methodology to forecast conditional distribution's tails.

This chapter proposes combining EVT estimators for truncated GPD with non-parametric methods, conditioned by spatio-temporal information. The GPD estimator is considered because (i) the shape parameter ξ allows modeling everything from extreme events with lightweight distribution (ξ<0) to events with exponential distribution (ξ=0) and events with heavy distribution (ξ>0); (ii) the existence of estimators for truncated GPD that can handle random variables with limited support like RES power.

1.3 Combining Non-parametric Models with a Truncated Generalized Pareto Distribution

As previously discussed in Section 1.2, EVT estimators are, at present, used in post-processing steps for quantiles forecasted with a non-parametric model, i.e., the non-parametric model forecasts all quantiles (including extreme quantiles) and EVT estimators are applied to correct the forecasted distribution's tails. However, since non-parametric approaches do not properly estimate extreme quantiles due to data sparsity, the performance of EVT estimators may be compromised. In this section and to overcome this gap, we propose to apply EVT estimator to historical data directly. The selection of the relevant historical data is guided by the non-parametric model.

Our proposal consists of the following steps, also depicted in Figure 1.2:

Step 1 **Non-parametric estimation:** A non-parametric model $Q(\tau|\mathbf{x})$ is estimated for intermediate quantiles, e.g., $\tau \in \boldsymbol{\tau} = \{0.05, .10, \ldots, 0.95\}$, i.e., 19 models are estimated using available historical data $\{(\mathbf{x}_i^{\text{tr}}, y_i^{\text{tr}})\}_{i=1}^{T}$. A rearrangement is also

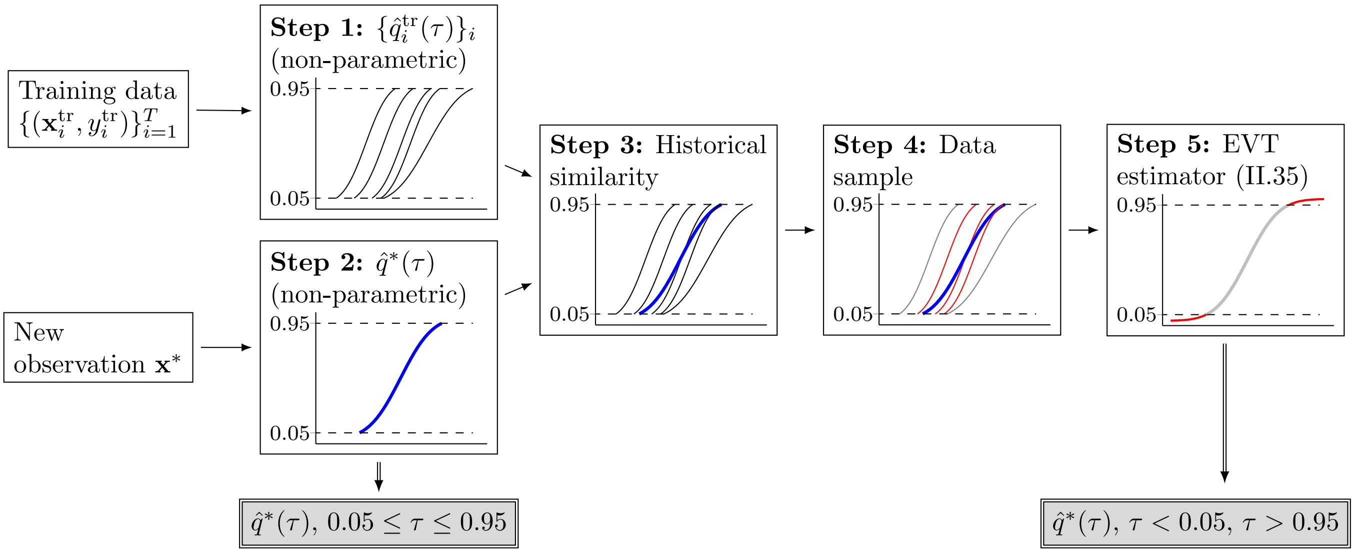

Training data $\{(\mathbf{x}_i^{\text{tr}}, y_i^{\text{tr}})\}_{i=1}^{T}$
New observation $\mathbf{x}^*$
Step 1: $\{\hat{q}_i^{\text{tr}}(\tau)\}_i$ (non-parametric)
Step 2: $\hat{q}^*(\tau)$ (non-parametric)
Step 3: Historical similarity
Step 4: Data sample
Step 5: EVT estimator (II.35)
0.95
0.05
$\hat{q}^*(\tau)$, $0.05 \leq \tau \leq 0.95$
$\hat{q}^*(\tau)$, $\tau < 0.05$, $\tau > 0.95$

Figure 1.2: Overview of the proposed forecasting model.

performed as described in (II.27). For a given training observation i, $(\mathbf{x}_i^{\text{tr}}, y_i^{\text{tr}})$, there is an estimation $\hat{q}_i^{\text{tr}}(\tau) = Q(\tau|\mathbf{x}_i^{\text{tr}})$.

Step 2 **Non-parametric forecast:** Given a new observation $\mathbf{x}^*$, the estimation $\hat{q}^*(\tau)$ is given by the aforementioned non-parametric model $Q(\tau|\mathbf{x})$ for $\tau \in \boldsymbol{\tau}$.

Step 3 **Historical similarity:** A similarity score $s\left(\mathbf{q}_1, \mathbf{q}_2\right)$ is computed between two quantile curves along several values of τ. The quantile curve $\hat{\mathbf{q}}^*$ from the new sample $\hat{\mathbf{q}}^* = [\hat{q}^*(\tau) \,|\, \tau \in \boldsymbol{\tau}]$ is compared with the quantile curve of each historical observation i, $\hat{\mathbf{q}}_i^{\text{tr}} = \left[\hat{q}_i^{\text{tr}}(\tau) \,|\, \tau \in \boldsymbol{\tau}\right]$. This similarity function is the Kolmogorov-Smirnov statistic given by

$$s(\mathbf{q}_1, \mathbf{q}_2) = \sup_{\tau} \left|\hat{\mathbf{q}}_1(\tau) - \hat{\mathbf{q}}_2(\tau)\right| . \tag{1.1}$$

The new observation is scored against each historical observation, $s_i = s(\hat{\mathbf{q}}^*, \hat{\mathbf{q}}_i^{\text{tr}})$. Since both quantile curves $\hat{\mathbf{q}}^*$ and $\hat{\mathbf{q}}_i^{\text{tr}}$ are conditioned by the covariates, the selection of the similar periods through s_i is also conditioned by the covariates.

Step 4 **EVT data sample:** The EVT estimator for the truncated GPD (II.35) is applied twice, for the lower-tail ($\tau < 0.05$) and the upper-tail ($\tau > 0.95$) quantiles. The historical values of y_i, used as the fitting sample of the EVT estimator, are selected as those corresponding to the top-ν (hyperparameter) values of $s_i = s(\hat{\mathbf{q}}^*, \hat{\mathbf{q}}_i^{\text{tr}})$. To avoid quantile crossing, these values are further narrowed down to $y_i \leq \hat{q}^*(0.05)$ and $y_i \geq \hat{q}^*(0.95)$, respectively.
Furthermore, EVT requires that the sample encompasses the entire quantile curve, therefore the remaining 90% quantiles, which correspond to $\frac{0.9\nu}{0.05}$ observations, are sampled from a spline interpolation constructed from the discrete $\hat{\mathbf{q}}^*$ curve. The ensuing sample is called $\mathbf{y}'$.

Step 5 **EVT estimation:** Lower-tail and upper-tail quantiles are estimated through the estimator for the truncated GPD (II.35), considering the sample $\mathbf{y}'$. Since, by convention, EVT distributions are defined for quantiles close to 1, the estimation of the lower-tail is obtained by considering the sample $y_i'' = C - y_i'$. EVT estimation is performed by (II.35) so that forecasted values are non-negative and below the installed capacity, $0 \leq \hat{y} \leq C$.

Note that step **Step 3** chooses i by comparing the probability distribution $\hat{\mathbf{q}}$ of the target variable conditioned on $\mathbf{x}^*$ and $\mathbf{x}_i^{\text{tr}}$. This is different from the usual approach of choosing i by comparing $\mathbf{x}^*$ against $\mathbf{x}_i^{\text{tr}}$ directly, as in [111], which assumes that covariates have equal weight and does not take the target variable into consideration. For instance, covariate j may be uncorrelated to the target, i.e., $\text{corr}((\mathbf{x}^{\text{tr}})_j, y^{\text{tr}}) = 0$, yet it contributes to the similarity through the Euclidean distance as $((\mathbf{x}_i^{\text{tr}})_j - (\mathbf{x}^*)_j)^2$. Our modification avoids that problem.

1.4 Case Studies

To evaluate the added-value of the proposed method, the models described in Table 1.1 are compared using three different datasets. The implementation is performed through R and Python programming languages, as described in Table I.1 of Prologue I. The local_tGPD benchmark is a naive model: the estimator for the truncated GPD (II.35) is applied to a b% of training samples listed in ascending order according to the Euclidean distance between $\mathbf{x}_i^{\text{tr}}$ and $\mathbf{x}^*$. The hyperparameter ν was determined by cross-validation (12 folds) in the training set, testing all values from 5% to 50%, with increments of 5%. This model is used to assess if the mapping between covariates (e.g., weather forecasts) and the target variable is important (as discussed in the last paragraph of the previous section). The hyperparameters of the GBT models were estimated using the Bayesian optimization algorithm from the Python implementation in [112]. A 12-fold cross-validation was employed and, since all real-world training sets contemplate one year of data, 12-folds guarantees 12 different monthly validation scenarios. For the final evaluation, the average of monthly Continuous Ranked Probability Score (CRPS) (II.20) is considered for each training set in the optimization process.

Also, the EVT estimators, in (II.29) and (II.35), require the selection of the number of ordered samples (k) for each time step. We followed the heuristic approach for choosing the first stable part of the plot of γ versus k, as illustrated in Figure II.8 of Prologue II. The stable part is found by computing a moving average on the differences of γ. In our approach, hyperparameter h was selected by cross-validation in the training set (12 folds), testing all values from 50 to 500 with increments of 50.

Three datasets are now described, and results are analyzed. The first experiment consists of using synthetic data that captures the three types of tails (lightweight, exponential,

Table 1.1: Evaluated forecasting models.

Notation	**Description**
GBT	GBT (non-parametric model)
local_tGPD	Hill estimator and truncated GPD in (II.35)*
Exp_Tails	Exponential functions in (II.28), using GBT
QR_EVT	QR combined with Hill estimator in (II.29)**, as in [69]
QR_EVT_T	QR, Hill estimator and transformed power data as in (II.31)**, as in [99]
GBT_EVT	GBT combined with Hill estimator (II.29)**
GBT_tGPD	Proposed method combining GBT with truncated GPD

* applied to b% of training samples ranked by similarity (Euclidean distance) between covariates

** EVT estimator used in post-processing stage and heavy), while the second and third experiments consist of real data from wind and solar production units, respectively. For synthetic data, the results are evaluated in terms of deviations between predicted and real quantiles, but for real data the real quantiles are unknown, motivating the use of literature metrics such as calibration (II.18), sharpness (II.19) and quantile score function (II.21).

1.4.1 Synthetic Data

Data Description

The proposed approach is firstly studied through simulation. The distribution from which we simulated Y is the truncated GPD for which the Cumulative Distribution Function (CDF) is given by

$$F^{\text{tGPD}}_{(C,\mu,\sigma,\xi)}(y) = \frac{F_{(\mu,\sigma,\xi)}(y) - F_{(\mu,\sigma,\xi)}(C)}{1 - F_{(\mu,\sigma,\xi)}(C)} \tag{1.2}$$

with

$$F_{(\mu,\sigma,\xi)}(y) = \begin{cases} 1 - \left(1 + \frac{\xi(y-\mu)}{\sigma}\right)^{-1/\xi} & \text{for } \xi \neq 0, \\ 1 - \exp\left(-\frac{y-\mu}{\sigma}\right) & \text{for } \xi = 0, \end{cases} \tag{1.3}$$

where the support of non-truncated Y is $y \geq \mu$ when $\xi \geq 0$ and $\mu \leq y \leq \mu - \sigma/\xi$ when $\xi < 0$, and C is the truncation value.

In this study, we take $C = 10$, $\mu = 0$, $\sigma = 1$ and $\xi(X_1, X_2) = (X_1 + X_2)\exp(X_1 + X_2)$, where X_1, X_2 are covariates, i.e., the distribution of Y is conditioned by X_1, X_2. We generate 500 datasets of size 4000, and the values for covariates X_1, X_2 are drawn from the $\mathcal{U}[-2, 2]$. Then, the estimation problem at $(x_1^*, x_2^*) \in \{(0,-1), (0,0), (0,1)\}$ is considered to illustrate the proposed approach. The corresponding CDF is depicted in Figure 1.3, for which $\xi < 0$, $\xi = 0$ and $\xi > 0$, respectively.

Results and Discussion

The proposed approach requires choosing two things: (i) the non-parametric model to estimate the quantiles for the central nominal proportions, and (ii) the nominal proportions to apply the selected non-parametric model, i.e., "should we consider $\tau \in \{0.05, \ldots, 0.95\}$ or $\tau \in \{0.01, \ldots, 0.99\}$?" The evaluation of GBT and QR is performed through 400 observations, the remaining 3600 are used to optimize the aforementioned hyperparameters by 12-fold cross-validation. Since the real quantiles values are known, the deviation between estimated and real values for the 500 datasets is depicted in Figure 1.4, considering $\tau = \{0.05, 0.35, 0.5, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.96, 0.99\}$. For nominal proportions below

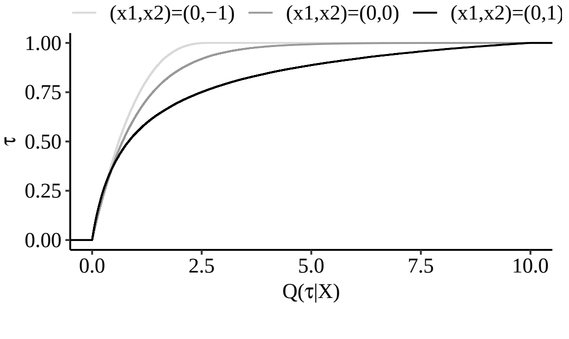


Figure 1.3: CDF for $(x_1^*, x_2^*) \in \{(0,-1), (0,0), (0,1)\}$.

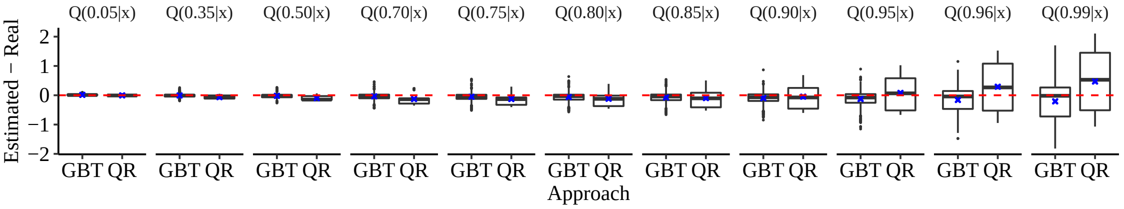


Figure 1.4: Comparison between GBT and QR (× represents the mean values).

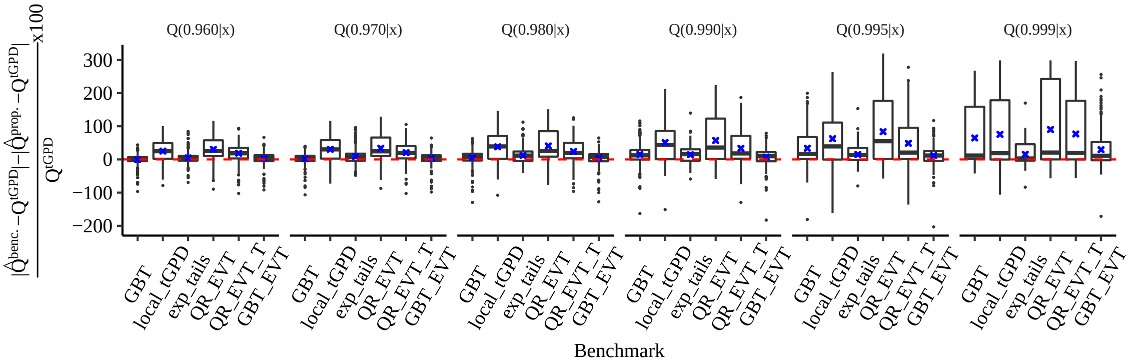


Figure 1.5: Improvement in terms of normalized absolute deviations, considering $(x_1^*, x_2^*) \in \{(0,-1),(0,0),(0,1)\}$ (× represents the mean values).

0.5 the deviations are similar, but for superior levels GBT has smaller deviations, motivating the selection of GBT. In fact, the QR approach tends to result in heavier tails. In addition, due to model degradation when $\tau = \{0.96, 0.99\}$, the benchmark models Exp_Tails, QR_EVT, QR_EVT_T, GBT_EVT and GBT_tGPD consider the non-parametric approach for $\tau \in \{0.05, \ldots, 0.95\}$.

Next, the quantiles with nominal proportion $\boldsymbol{\tau}_e = \{0.96, 0.97, 0.98, 0.99, 0.995, 0.999\}$ are estimated for $(x_1^*, x_2^*) \in \{(0,-1),(0,0),(0,1)\}$. Figure 1.5 summarizes the difference between the normalized absolute deviations,

$$\frac{|\hat{Q}^{\text{benchmark}}(\tau|\mathbf{x}) - Q^{\text{tGPD}}(\tau|\mathbf{x})| - |\hat{Q}^{\text{GBT\_tGPD}}(\tau|\mathbf{x}) - Q^{\text{tGPD}}(\tau|\mathbf{x})|}{Q^{\text{tGPD}}(\tau|\mathbf{x})} \times 100, \tag{1.4}$$

$\tau \in \boldsymbol{\tau}_e$. Positive values indicate the deviations obtained by our proposal are smaller. According to this analysis, for $\tau \in \{0.96, 0.97, 0.98\}$ in almost 75% of the observations our proposal has smaller deviations when compared to QR-based approaches, Exp_Tails, and local_tGPD. But, when compared to GBT, GBT combined with Hill estimator (GBT_EVT), and Exp_Tails, this superiority is not observed, and similar deviations are achieved. However, for the most extreme quantiles, $\tau \in \{0.99, 0.995, 0.999\}$, our proposal has been more effective than all benchmarks.

To complement this analysis, Table 1.2 splits the results by (x_1^*, x_2^*) for $\tau \in \{0.99, 0.995, 0.999\}$. The mean of $\hat{Q}(\tau|\mathbf{x})$ over the 500 datasets is presented and the Diebold-Mariano (DM) test, discussed in Section II.2 of Prologue II, is used to test the hypothesis of equal deviations. When $\xi < 0$ the quantiles estimated by our proposal are closer to the real values. Regarding the exponential tails, $(x_1^*, x_2^*) = (0,0)$, Exp_Tails, and GBT-based methods performed similarly to our proposal. Lastly, since QR-based approaches tend to result in heavier tails, their performance is favored for the point $(x_1^*, x_2^*) = (0,1)$ for which

Table 1.2: Mean quantile forecasts for $\tau \in \{0.99, 0.995, 0.999\}$.

	$\mathbf{x} = (0, -1), \xi < 0$			$\mathbf{x} = (0, 0), \xi = 0$			$\mathbf{x} = (1, 0), \xi > 0$		
τ	0.99	0.995	0.999	0.99	0.995	0.999	0.99	0.995	0.999
$Q^{\mathrm{tGPD}}(\tau)$	2.22	2.33	2.50	4.60	5.29	6.86	9.35	9.67	9.93
GBT	3.85	5.37	8.3	5.17	6.34	8.78	6.97	7.54	8.95
local_tGPD	5.48	6.75	8.98	7.91	8.89	9.78	8.90	**9.46**	9.60
Exp_Tails	3.32	3.73	4.49	5.39	5.85	**6.61**	8.31	8.61	9.00
QR_EVT	6.04	7.89	10.00	7.54	9.57	10.00	**9.01**	9.97	10.00
QR_EVT_T	4.85	6.10	9.05	6.43	7.97	9.83	8.32	9.44	**9.99**
GBT_EVT	3.37	3.96	5.8	**5.09**	**5.37**✓	7.34	6.87	7.28	8.32
GBT_tGPD	**2.89**✓	**3.13**✓	**3.57**✓	5.13	5.68	6.59	8.26	8.90	9.68

✓ statistically significant improvement against all others (DM test)

the quantile 0.9 is 9.34 (almost the limit $C = 10$). QR-based approaches result in larger forecasting intervals $[\hat{Q}(1-\tau), \hat{Q}(\tau)]$ for all considered (x_1^*, x_2^*).

Since QR-based approaches has poor performances when $\xi \in \{-1, 0\}$, we conclude that the proposed approach models better the overall tails' behaviors.

1.4.2 Wind Power Data

Data Description

The proposed method is also tested with a wind power dataset from the *Sotavento* wind power plant, located in Galicia (Spain), as depicted in Figure 1.6, with a total installed capacity of 17.56 MW. The dataset extends from January 1st, 2014 to September 22nd, 2016, with hourly time steps.

The NWP data was retrieved from the MeteoGalicia THREDDS server, which is a publicly available service that provides historical and daily forecasts of several weather variables. The NWP is run at 0h UTC and the time horizon is 96 hours-ahead, meaning that for each day a set of four forecasts are available for each point of the grid (one generated in the current day at 0h UTC plus three generated on the previous days).

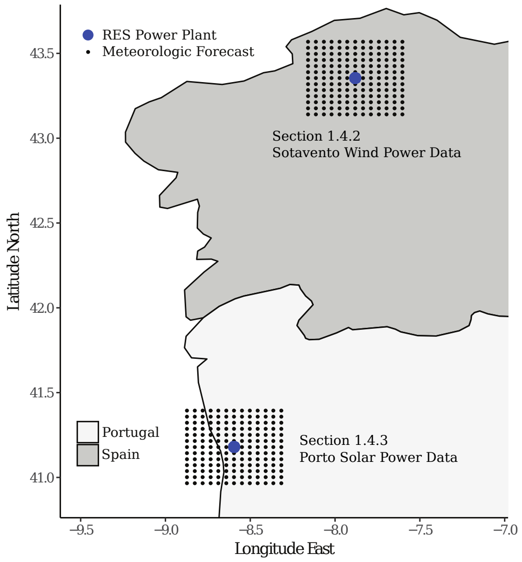


Figure 1.6: Geographical representation of data collection points for real datasets.

The NWP model provides forecasts for: (a) u (m/s), azimuthal wind speed; (b) v (m/s), meridional wind speed; (c) *mod* (m/s), wind speed module; (d) *dir* $[0, 360]$, wind direction. Four model levels (0 to 3) are available, meaning a total of 16 variables in each grid point.

Covariates extracted from the NWP grid. The features created by the authors of [12], from a NWP grid with 13×13 equally distributed points (Figure 1.6), were used in this work and are described below. Our goal is to forecast the wind power for 24h-ahead and the majority of the covariates are constructed with the most recent NWP run.

Temporal information is represented by:

- Temporal variance for the *mod* variable (level 3) at the central point of the grid, computed as

$$\sigma_{\text{time}}(t+h) = \sqrt{\frac{\sum_{i=-7}^{7}(mod_{t+h+i} - \overline{mod})^2}{14}}. \tag{1.5}$$

- *Lags* and *leads*, $x_{t+h\pm z}$, for *mod* and *dir* (level 3) at the central point of the grid, $z = 1, 2, 3$.
- Four predictions generated for *mod* (level 3) at the central point of the grid.

The spatial information is represented through:

- Principal Component Analysis (PCA) applied to *mod* and *dir* (levels 1, 2, 3), and to u and v (level 3) with a 95% variance threshold.
- Spatial standard deviation for mod, u and v at level 3, computed as

$$\sigma_{\text{spatial}}(t+h) = \sqrt{\frac{\sum_{i=1}^{N_p}\left(x_{i,t+h} - \overline{x}_{t+h}\right)^2}{N_p - 1}}, \tag{1.6}$$

  where N_p is the number of geographical points in the NWP grid, $x_{i,t+h}$ is the value of variable x at time $t+h$ and location i, and $\overline{x}_{t+h}$ is the mean of x for all locations.
- Spatial mean computed with the grid values of mod, u and v at model levels 1, 2, 3.

Data division. A sliding-window approach was used for training the models. Table 1.3 presents the four distinct test folds. Each train and test set consists of 12 and 5 months, respectively, allowing an evaluation under different conditions.

Results and Discussion

Since the GBT model performs better for power data, due to the nonlinear relationship between wind and power, GBT is used to estimate quantiles between 0.05 and 0.95 [12].

Table 1.3: Time period for training and testing folds.

Fold	Train set range	Test set range
1	01/01/2014–31/12/2014	01/01/2015–31/05/2015
2	01/06/2014–31/05/2015	01/06/2015–31/10/2015
3	01/11/2015–30/10/2016	01/11/2015–31/03/2016
4	01/04/2015–31/03/2016	01/04/2016–22/09/2016

Table 1.4: Relative quantile loss improvement (%) over the baseline models (wind power dataset), considering the extreme quantiles $\boldsymbol{\tau}_e$.

Folds	Fold 1	Fold 2	Fold 3	Fold 4	W.Avg.
GBT	5.40	1.97	7.03	0.12	3.76
local_tGPD	22.27	29.34	21.71	27.80	26.25
Exp_Tails	12.87	11.03	9.44	14.79	12.55
QR_EVT	10.16	7.10	4.56	8.90	8.21
QR_EVT_T	12.39	7.20	10.78	8.55	10.39
GBT_EVT	12.20	9.06	9.33	5.03	9.75

Table 1.5: Quantile loss for each model (lower is better), with regard to the wind power dataset.

$\boldsymbol{\tau}$	0.001	0.005	0.01	0.99	0.995	0.999
GBT	3.20	15.49	29.60	52.65	30.98	10.60
local_tGPD	3.16	15.74	31.05	84.52	45.21	9.69
Exp_Tails	8.63	20.95	32.47	53.14	32.26	9.43
QR_EVT	3.14	15.64	29.67	54.90	32.17	8.89
QR_EVT_T	3.19	15.55	29.84	59.27	34.48	9.68
GBT_EVT	3.17	15.72	31.97	67.13	35.23	8.45
GBT_tGPD†	**3.13**	**15.28**	**29.30**	**50.35**$^\checkmark$	**28.23**$^\checkmark$	**8.01**$^\checkmark$

† the proposed method

✓ statistically significant improvement against all others (DM test)

The proposed model is then used to estimate the quantiles $\boldsymbol{\tau}_e = \{0.001, 0.005, 0.01, 0.02, 0.03, 0.04, 0.96, 0.97, 0.98, 0.99, 0.995, 0.999\}$.

Table 1.4 summarizes the relative quantile score improvement obtained by GBT_tGPD over the baseline models. Quantile score is computed by considering the extreme quantiles for nominal proportions $\boldsymbol{\tau}_e$. The GBT_tGPD improvement is greater than 3.5% for all testing folds, except over GBT.

The statistics of the wind power generation for the train and test periods are summarized in Figure 1.7. Two factors might justify the different improvements obtained in the four folds: the variability of the wind power values and the differences between train and test data distributions. When high variability is associated with different distributions for train and test sets, as is the case of fold 3, the selection of 200 observations results on more dispersed power measurements and, consequently, the EVT estimator has longer tails.

Table 1.5 shows a finer-grained view of the quantile loss for the most extreme quantiles, averaged over the testing folds. It can be noticed that the improvement of the proposed method is slightly higher for the upper quantiles, but, all in all, the proposed method shows the best results.

Figure 1.8 complements the previous analysis by showing the calibration values for each model. For the upper tail, the GBT_tGPD model exhibits almost perfect calibration for all quantiles. In the lower tail, it produces a lower overestimation of the quantiles. However, when considering all quantiles, QR-based models are the most well-calibrated models. Yet, when analyzing the sharpness of the forecast intervals generated by these methods in Figure 1.9, these methods show that the better calibration comes at the cost of a higher amplitude (i.e., lower sharpness), which is a trade-off well-known in the forecasting literature. The lower sharpness from GBT_EVT, QR_EVT_T and QR_EVT is justified by the fact that the Hill estimator is more suitable for heavy-tailed distributions.

For illustrative purposes, the most extreme forecasted quantiles (i.e., 0.001 and 0.999) obtained with GBT, Exp_Tails and GBT_tGPD are depicted in Figure 1.10. The Exp_Tails

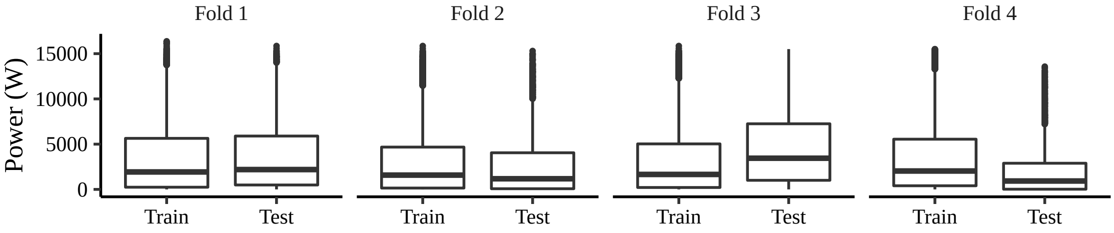


Figure 1.7: Boxplot for the wind power considering the division on Table 1.3.

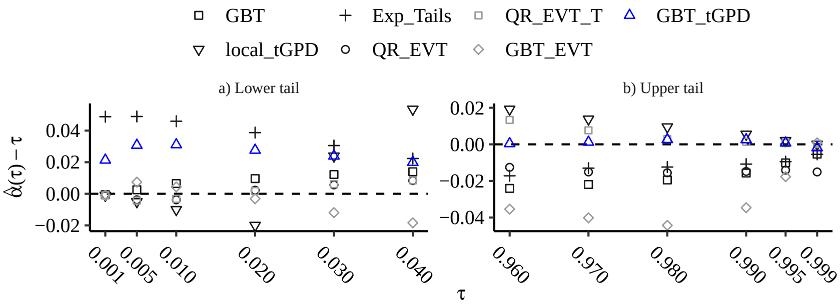


Figure 1.8: Deviation between nominal and empirical quantiles for wind power data, considering all folds. Dashed black line represents perfect calibration.

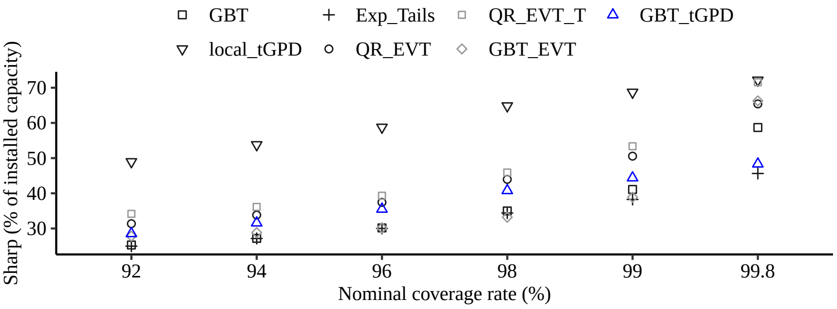


Figure 1.9: Sharpness results for wind power data, considering all folds.

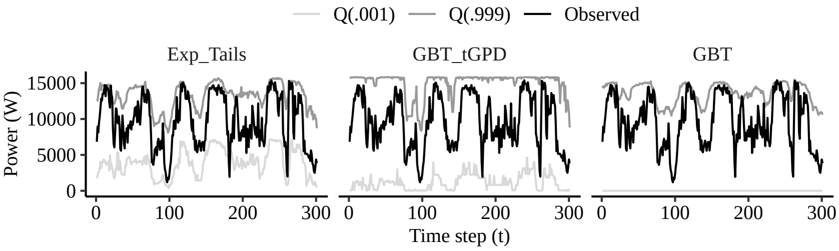


Figure 1.10: Illustrative forecast of extreme quantiles for GBT, Exp_Tails and GBT_tGPD, considering wind power data.

model was chosen since it is the model with the lowest sharpness. This plot clearly shows that GBT_tGPD has a better calibration than Exp_Tails, but wider intervals, and also shows a higher temporal variability of the forecast generated by GBT_tGPD.

The baseline model GBT shows small sharpness for all nominal coverage rates (between 92% and 99%) except the most extreme one (99.8%), as depicted in Figure 1.9. The small sharpness is explained by the fact that GBT fails to capture the variability for the most extreme quantiles. The forecast of the lower quantiles is particularly bad with values very close to zero, as depicted in Figure 1.10.

1.4.3 Solar Power Data

Data Description

The solar power dataset consists of hourly power measurements from a 16320 W peak photovoltaic power plant located in Porto city, Portugal, as illustrated in Figure 1.6. The dataset extends from March 28th, 2013 to June 28th, 2016, with hourly time steps.

As in the previous case study, the NWP data was retrieved from the MeteoGalicia THREDDS server, and the NWP model provides forecasts for: (a) swflx (W/m^2), surface downwelling shortwave flux; (b) temp (K), ambient temperature at 2 meters; (c) cfl $[0, 1]$, cloud cover at low levels; (d) cfm $[0, 1]$, cloud cover at mid levels; (e) cfh $[0, 1]$, cloud cover at high levels; (f) cft $[0, 1]$, cloud cover at low and mid levels.

Covariates extracted from the NWP grid. The features created by the authors of [12], from a NWP grid with 13×13 equally distributed points (Figure 1.6), were used in this work and are described below. Our goal is to forecast solar power for 24h-ahead. Since night hours have zero power production, these hours are removed.

Temporal information is represented by:

- Temporal variance for the swflx variable at the central point of the grid, as in (1.5).
- *Lags* and *leads*, $x_{t+h\pm z}$, for *mod* and *dir* at the central point of the grid, $z = 1, 2, 3$.
- Four predictions generated for mod at the central point of the grid.

The spatial information is represented through:

- PCA applied to swflx, cfl, cfm and cft with a 90% variance threshold.
- Spatial standard deviation for swflx computed as in (1.6).
- Spatial mean computed with the grid values of swflx.

Moreover, calendar variables (month and hour of the day) are also used.

Data division. Five distinct test folds are considered (Table 1.6). Each train and test set consists of 12 and 5 months, respectively, allowing an evaluation under different conditions.

Table 1.6: Time period for training and testing folds.

Fold	Train set range	Test set range
1	01/05/2013–30/04/2014	01/05/2014–30/09/2014
2	01/10/2013–30/09/2014	01/10/2014–28/02/2015
3	01/11/2014–31/10/2015	01/11/2015–31/07/2015
4	01/08/2014–31/07/2015	01/08/2015–31/12/2015
5	01/01/2015–31/12/2015	01/01/2016–28/06/2016

Results and Discussion

Based in [12], GBT is used to estimate quantiles between 0.05 and 0.95. Again, the proposed model is used to estimate the quantiles $\boldsymbol{\tau}_e = \{0.001, 0.005, 0.01, 0.02, 0.03, 0.04, 0.96, 0.97, 0.98, 0.99, 0.995, 0.999\}$.

The relative quantile score improvement obtained by GBT_tGPD over the baseline models is provided in Table 1.7, considering nominal proportions $\boldsymbol{\tau}_e$. The GBT_tGPD improvement over the local_tGPD, QR-based approaches and GBT_EVT is greater than 14% for all folds. Regarding GBT and Exp_Tails, the improvement over all folds is 2.09% and 3.25%, respectively, but in some folds our proposal results in greater quantile scores.

To justify the different improvements obtained in the five folds, the statistics of the solar power generation for the train and test periods are summarized in Figure 1.11. When high variability is associated with different distributions for train and test sets, as is the case of fold 3, the selection of a given number of observations results in more dispersed power measurements and, consequently, the EVT estimator for truncated GPD has longer tails.

Table 1.8 summarizes the quantile loss for the most extreme quantiles, $\tau \in \{0.001, 0.005, 0.01, 0.99, 0.995, 0.999\}$, averaged over the testing folds. The improvement of the proposed method is slightly higher for the lower quantiles, but in general, the proposed method shows the best performance.

Figure 1.12 complements the previous analysis by showing the calibration values for each model. For the lower tail, the GBT_tGPD model exhibits almost perfect calibration for all quantiles. In the upper tail, it produces a lower underestimation of the quantiles for nominal proportions 0.96 and 0.97. However, when considering all quantiles, QR-based models are the most well-calibrated models. Yet, when analyzing the sharpness of the forecast intervals generated by these methods in Figure 1.13, these methods show that the better calibration comes at the cost of higher amplitude (i.e., lower sharpness).

Table 1.7: Relative quantile loss improvement (%) over the baseline models (solar power dataset), considering the extreme quantiles $\boldsymbol{\tau}_e$.

Folds	**Fold 1**	**Fold 2**	**Fold 3**	**Fold 4**	**Fold 5**	**W.Avg.**
GBT	0.65	5.90	-1.42	1.35	3.95	2.09
local_tGPD	56.32	42.73	54.18	46.05	49.40	49.74
Exp_Tails	8.24	10.52	-2.10	0.08	0.65	3.25
QR_EVT	46.66	36.68	41.20	34.17	33.56	38.45
QR_EVT_T	48.55	40.19	44.85	37.15	35.26	41.20
GBT_EVT	25.18	14.84	27.26	19.23	19.72	21.25

Table 1.8: Quantile loss for each model (lower is better), with regard to the solar power dataset.

$\boldsymbol{\tau}$	**0.001**	**0.005**	**0.01**	**0.99**	**0.995**	**0.999**
GBT	4.72	20.90	232.16	31.23	17.37	6.12
local_tGPD	4.79	23.97	479.42	86.15	44.11	8.72
Exp_Tails	5.99	21.39	232.21	34.13	20.07	5.30
QR_EVT	4.72	22.37	360.07	58.99	32.78	8.54
QR_EVT_T	4.95	23.75	360.05	65.28	36.88	9.07
GBT_EVT	4.79	23.97	479.42	29.92	17.57	5.06
GBT_tGPD[†]	**3.76**✓	**17.64**✓	**223.54**✓	**28.88**✓	**16.86**✓	**4.54**✓

[†] the proposed method

✓ statistically significant improvement against all others (DM test)

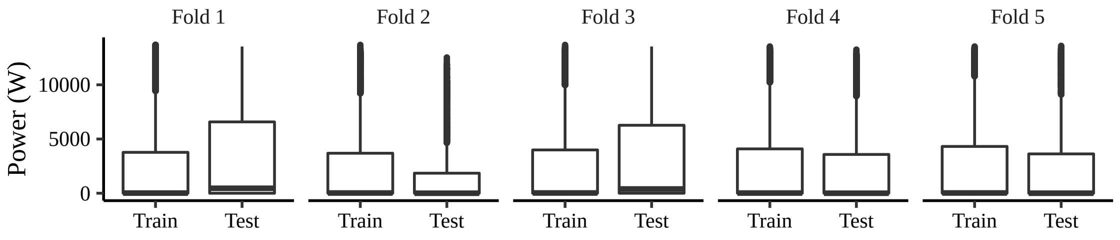


Figure 1.11: Boxplot for the solar power considering the division on Table 1.6.

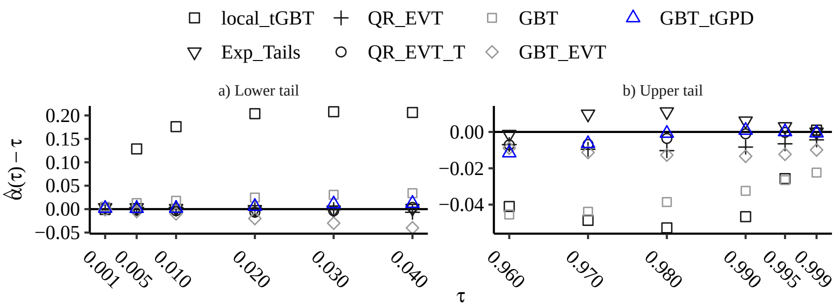


Figure 1.12: Deviation between nominal and empirical quantiles for solar power data, considering all folds. Dashed black line represents perfect calibration.

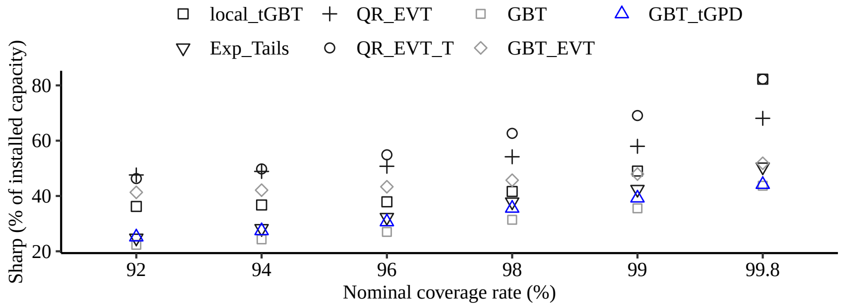


Figure 1.13: Sharpness results for solar power data, considering all folds.

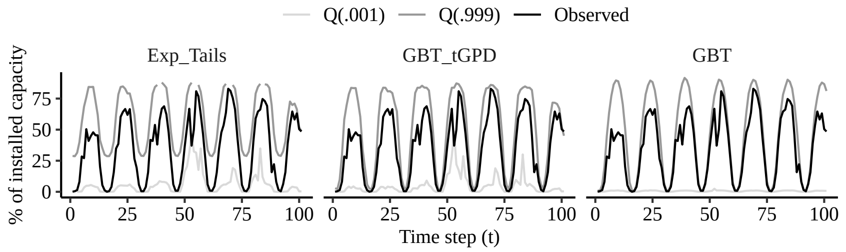


Figure 1.14: Illustrative forecast of extreme quantiles for GBT, Exp_Tails and GBT_tGPD, considering solar power data.

Finally, the most extreme forecasted quantiles (i.e., 0.001 and 0.999) obtained with GBT, Exp_Tails and GBT_tGPD are depicted in Figure 1.14. Considering $\tau = 0.001$, Exp_Tails and GBT_tGPD perform similarly, while GBT tend to provide a value close to zero every time. For $\tau = 0.999$, GBT and GBT_tGPD clearly outperforms Exp_Tails in hours with smaller power production, possibly due to the fact that for this hours the tails are lightweight.

1.5 Concluding Remarks

Accurate forecasting of distribution tails remains a challenge in the RES forecasting literature since are often associated with data sparsity. Furthermore, information from the tails is of major importance in power system operation (e.g., reserve capacity setting, dynamic line rating) and RES market trading. For this reason, concepts were borrowed from EVT for truncated variables and combined with a non-parametric forecasting framework that includes features created from spatial-temporal information.

Two major benefits are provided by this work: (a) covariates are used to produce conditional forecasts of quantiles without any limitation in the number of variables; (b) the parametric EVT-based estimator can be combined with any non-parametric model (artificial neural networks, GBT, random forests, etc.) without any major modification. Moreover, the results for a wind farm located in Galicia, Spain, and a power plant located in Porto, Portugal, show that the proposed method can provide sharp and calibrated forecasts (important to avoid over- and under-estimation of risk) and outperforms state-of-the-art methods in terms of the quantile score. Finally, the proposed method can be transposed to other use cases in the energy sector, such as risk management in portfolio's future returns and study grid resilience to adverse weather events.

2

Finding the Privacy Gaps in Collaborative Forecasting

Abstract. Cooperation between different data owners may lead to an improvement in forecast quality – for instance, by benefiting from spatio-temporal dependencies in geographically distributed time series. Due to business competitive factors and personal data protection concerns, however, said data owners might be unwilling to share their data. Interest in collaborative privacy-preserving forecasting is thus increasing. This chapter analyzes the state-of-the-art and unveils several shortcomings of existing methods in guaranteeing data privacy when employing vector autoregressive models. The methods are divided into three groups: data transformation, secure multi-party computations, and decomposition methods. The analysis shows that state-of-the-art techniques have limitations in preserving data privacy, such as *(i)* the necessary trade-off between privacy and forecasting accuracy, empirically evaluated through simulations and real-world experiments based on wind and solar data; and *(ii)* iterative model fitting processes, which reveal data after a number of iterations.

2.1 Introduction

The progress of the internet-of-things (IoT) and big data technologies is fostering a disruptive evolution in the development of innovative data analytics methods and algorithms. This also yields ideal conditions for data-driven services (from descriptive to prescriptive analysis), in which the accessibility to large volumes of data is a fundamental requirement. In this sense, the combination of data from different owners can provide valuable information for end-users and increase their competitiveness.

In order to combine data coming from different sources, several statistical approaches have emerged. For example, in time series collaborative forecasting, the Vector AutoRegressive (VAR) model has been widely used to forecast variables that may have different data owners. In the energy sector, the VAR model is deemed appropriate to update very short-term forecasts (e.g., from 15 min to 6 h ahead) with recent data, thus taking advantage of geographically distributed data collected from sensors (e.g., anemometers and pyranometers) and/or wind turbines and solar power inverters [78, 84]. The VAR model can also be used in short-term electricity price forecasting [113]. Furthermore, the large number of potential data owners favors the estimation of the VAR model's coefficients by applying distributed optimization algorithms. The Alternating Direction Method of Multipliers (ADMM) is a widely used convex optimization technique; see [104]. The combination of the VAR model and ADMM can be used jointly for collaborative forecasting [81], which consists of collecting and combining information from diverse owners. Collaborative forecasting methods require sharing data or coefficients, depending on the structure of the data, and may or may not be focused on data privacy. This process is also called federated learning [114].

Some other examples of collaborative forecasting include: (a) forecasting and inventory control in supply chains, in which the benefits of various types of information-sharing options are investigated [16, 17]; (b) forecasting traffic flow (i.e., traffic speed) at different locations [115]; (c) forecasting retail prices of a specific product at every outlet by using historical retail prices of the product at a target outlet and at competing outlets [15]. The VAR model is the simplest collaborative model, but conceptually, a collaborative forecasting model for time series does not need to be a VAR. Furthermore, it is possible to extend the VAR model to include exogenous information (see [103] for more details) and to model non-linear relationships with past values (e.g., [116] extend the additive model structure to a multivariate setting).

Setting aside the significant potential of the VAR model for collaborative forecasting, the concerns with the privacy of personal and commercially sensitive data constitute a critical barrier and require privacy-preserving algorithmic solutions for estimating the coefficients of the model.

A confidentiality breach occurs when third parties recover without consent any data provided in confidence. A single record leaked from a dataset is of more or less importance depending on the nature of the data. For example, in medical data, where each record represents a different patient, a single leaked record can disclose all the details about a patient. By contrast, with renewable energy generation time series, the knowledge that 30 MWh was produced in a given hour is not very relevant to a competitor. Hereafter, the term confidentiality breach designates the reconstruction of the entire dataset by another party.

These concerns with data confidentiality motivated research into methods that can handle confidential data, such as linear regression and classification problems [117], ridge linear regression [118], logistic regression [119], survival analysis [120], and aggregated statistics for time series data [121]. Aggregated statistics consist of aggregating a set of time series data through a specific function, such as the average (e.g., the average amount of daily exercise), sum, minimum, and maximum. However, certain approaches are vulnerable to confidentiality breaches, showing that the statistical methods developed to protect data privacy should be analyzed to confirm their robustness, and that additional research may be required to address overlooked limitations [122]. Furthermore, the application of these methods to the VAR model needs to be carefully analyzed, since the target variables are the time series of each data owner, and the covariates are the lags of the same time series, meaning that both target and covariates share a large proportion of values.

The simplest solution would be to have the data owners agree on a commonly trusted entity (or a central node) capable of gathering private data, solving the associated model's fitting problem on behalf of the data owners, and then returning the results [123]. However, in many cases, the data owners are unwilling to share their data even with a trusted central node. This has motivated the development of data markets to monetize data and promote data sharing [124], which can be driven by blockchain and smart contracts technology [125]. Data markets will be the focus of Chapter 4.

Another possibility would be to apply differential privacy mechanisms, which consist of adding properly calibrated noise to an algorithm (e.g., adding noise to the coefficients estimated during each iteration of the fitting procedure) or directly to the data. Differential privacy is not an algorithm, but rather a rigorous definition of privacy that is useful for quantifying and bounding privacy loss (i.e., how much original data a party can recover when receiving data protected with added noise) [126]. It requires computations insensitive to changes in any particular record or intermediate computations, thereby restricting data leaks through the results; see A.1. While computationally efficient and popular, these techniques invariably degrade the predictive performance of the model [114] and are not very effective, as we show in what follows.

This chapter is a review of the state-of-the-art in statistical methods for collaborative forecasting with privacy-preserving approaches. This work is not restricted to a simple overview of the existing methods. It includes a critical evaluation of said methods from a mathematical and numerical point of view – namely, when applied to the VAR model. The major contribution to the literature is to show gaps and downsides to current methods, and to present insights for further improvements towards fully privacy-preserving VAR forecasting methods. Suggestions to improve on these methods are then presented in the chapter after this one.

In this chapter, we analyze existing state-of-the-art privacy-preserving techniques, dividing them into the following groups:

- *Data transformation methods*: each data owner transforms the data before the model's fitting process, by adding randomness to the original data in such a way that high accuracy and privacy can be achieved at the end of the fitting process. The statistical method is independent of the transformation function and it is applied to the transformed data.

- *Secure multi-party computation protocols*: data encryption occurs while fitting the statistical model (i.e., intermediate calculations of an iterative process) and data owners are required to conjointly compute a function over their data with protocols for secure matrix operations. A protocol consists of rules that determine how data owners must operate to determine said function. These rules establish the calculations assigned to each data owner, what information should be shared among them, and the conditions necessary for the adequate implementation of said calculations.

- *Decomposition-based methods*: the optimization problem is decomposed into subproblems, allowing each data owner to fit model coefficients separately.

The remainder of the chapter is organized as follows: Section 2.2 describes the state-of-the-art for collaborative privacy-preserving forecasting. Section 2.3 critically evaluates state-of-the-art methods when applied to the VAR model. Wind and solar energy time series data are used in the numerical analysis. Section 2.4 offers a discussion and comparison of the presented approaches, and concluding remarks are presented in Section 2.5.

2.2 Privacy-preserving Approaches

For notation purposes, in this section, $\mathbf{Z} \in \mathbb{R}^{T \times M}$ is the covariate matrix and $\mathbf{Y} \in \mathbb{R}^{T \times N}$ is the target matrix, considering n data owners. The values T, M and N are the number of records, covariates and target variables, respectively. When considering collaborative forecasting models, different divisions of the data may be considered. Figure 2.1 shows the two most common:

1. *Data split by records:* the data owners, represented as $A_i, i \in \{1, \dots, n\}$, observe the same features for different groups of samples, e.g., different timestamps in the case of time series. $\mathbf{Z}$ is split into $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times M}$ and $\mathbf{Y}$ into $\mathbf{Y}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times N}$, such that $\sum_{i=1}^{n} T_{A_i} = T$;

2. *Data split by features:* the data owners observe different features of the same records. $\mathbf{Z} = [\mathbf{Z}_{A_1}, \dots, \mathbf{Z}_{A_n}]$, $\mathbf{Y} = [\mathbf{Y}_{A_1}, \dots, \mathbf{Y}_{A_n}]$, such that $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times M_{A_i}}$, $\mathbf{Y}_{A_i} \in \mathbb{R}^{T \times N_{A_i}}$, with $\sum_{i=1}^{n} M_{A_i} = M$ and $\sum_{i=1}^{n} N_{A_i} = N$;

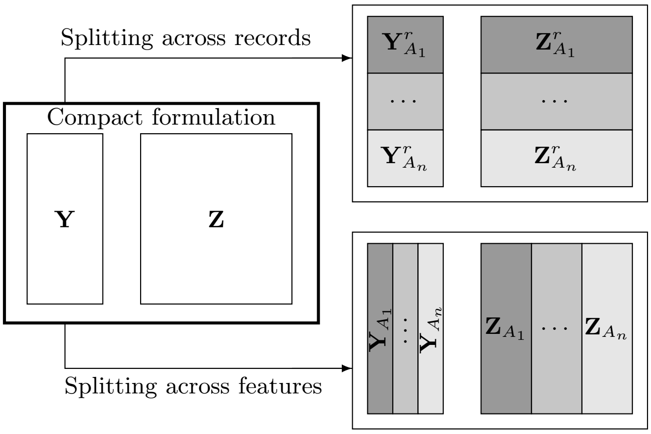


Figure 2.1: Common data division structures.

This section summarizes state-of-the-art approaches to deal with privacy-preserving collaborative forecasting methods. Section 2.2.1 describes the methods that ensure confidentiality by transforming the data. Section 2.2.2 presents and analyzes the secure multi-party protocols. Section 2.2.3 describes the decomposition-based methods.

2.2.1 Data Transformation Methods

Data transformation methods use operator $\mathcal{T}$ to transform the data matrix $\mathbf{X}$ into $\tilde{\mathbf{X}} = \mathcal{T}(\mathbf{X})$. Then, the problem is solved in the transformed domain. A common method of masking sensitive data is adding or multiplying it by perturbation matrices. In additive randomization, random noise is added to the data in order to mask the values of records. Consequently, the more masked the data becomes, the more secure it will be, as long as the differential privacy definition is respected (see A.1). However, the use of randomized data implies the deterioration of the estimated statistical models, and the estimated coefficients of said data should be close to the estimated coefficients after using original data [127].

Multiplicative randomization involves changing the dimensions of the data by multiplying it by random perturbation matrices. If the perturbation matrix $\mathbf{W} \in \mathbb{R}^{k\times m}$ multiplies the original data $\mathbf{X} \in \mathbb{R}^{m\times n}$ on the left (pre-multiplication), i.e., $\mathbf{WX}$, then it is possible to change the number of records; otherwise, if $\mathbf{W} \in \mathbb{R}^{n\times s}$ multiplies $\mathbf{X} \in \mathbb{R}^{m\times n}$ on the right (post-multiplication), i.e., $\mathbf{XW}$, it is possible to modify the number of features. Hence, it is possible to change both dimensions by applying both pre- and post-multiplication by perturbation matrices.

Single Data Owner

The use of linear algebra to mask data is a common practice in recent outsourcing approaches, in which a data owner resorts to the cloud to fit model coefficients without sharing confidential data. For example, in [128] the coefficients that optimize the linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \ , \tag{2.1}$$

with covariate matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$, target variable $\mathbf{y} \in \mathbb{R}^m$, coefficient vector $\boldsymbol{\beta} \in \mathbb{R}^n$ and error vector $\boldsymbol{\varepsilon} \in \mathbb{R}^n$, are estimated through the regularized least squares estimate for the ridge linear regression, with penalization term $\lambda > 0$,

$$\hat{\boldsymbol{\beta}}_{\text{ridge}} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}. \tag{2.2}$$

In order to compute $\hat{\boldsymbol{\beta}}_{\text{ridge}}$ via a cloud server, the authors consider that

$$\hat{\boldsymbol{\beta}}_{\text{ridge}} = \mathbf{A}^{-1}\mathbf{b} \ , \tag{2.3}$$

where $\mathbf{A} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}$ and $\mathbf{b} = \mathbf{X}^\top\mathbf{y}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{b} \in \mathbb{R}^n$. Then, the masked matrices $\mathbf{MAN}$ and $\mathbf{M}(\mathbf{b}+\mathbf{Ar})$ are sent to the server, which computes

$$\hat{\boldsymbol{\beta}}' = (\mathbf{MAN})^{-1}(\mathbf{M}(\mathbf{b}+\mathbf{Ar})) \ , \tag{2.4}$$

where $\mathbf{M}$, $\mathbf{N}$, and $\mathbf{r}$ are randomly generated matrices, $\mathbf{M},\mathbf{N} \in \mathbb{R}^{n\times n}$, $\mathbf{r} \in \mathbb{R}^n$. Finally, the data owner receives $\hat{\boldsymbol{\beta}}'$ and recovers the original coefficients by computing $\hat{\boldsymbol{\beta}}_{\text{ridge}} = \mathbf{N}\hat{\boldsymbol{\beta}}' - \mathbf{r}$.

Data normalization is a data transformation approach that masks data by transforming the original features into a new range through the use of a mathematical function. There are many methods of data normalization, the most important ones being z-score and min-max normalization [129], which are useful when the actual minimum and maximum values of the features are unknown. However, in many applications, these values are either known or publicly available, and normalized values still encompass commercially valuable information.

For time series data, other approaches to data randomization make use of the Fourier and wavelet transforms. A Fourier transform can represent periodic time series as a linear combination of sinusoidal components (sine and cosine). In [130], each data owner generates a noise time series by (i) adding Gaussian noise to relevant coefficients, or (ii) disrupting each sinusoidal component by randomly changing its magnitude and phase. Similarly, a wavelet transform can represent time series as a combination of functions (e.g., the Mexican hat or Poisson wavelets), and randomness can be introduced by adding random noise to the coefficients [130]. However, there are no privacy guarantees, since noise does not respect any formal definition, unlike differential privacy.

Multiple Data Owners

The task of masking data is even more challenging when dealing with different data owners, since it is crucial to ensure that the transformations that data owners make to their data preserve the real relationship between the variables or the time series.

Usually, for generalized linear models (e.g., linear regression models, logistic regression models, etc.), where n data owners observe the same features –i.e., data are split by records, as illustrated in Figure 2.1 – each data owner $A_i, i = 1, ..., n$, can individually multiply their covariate matrix $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i}\times M}$ and target variable $\mathbf{Y}^r_{A_i} \in \mathbb{R}^{T_{A_i}\times N}$ by a random matrix $\mathbf{M}_{A_i} \in \mathbb{R}^{k\times T_{A_i}}$ (with a jointly defined k value), providing $\mathbf{M}_{A_i}\mathbf{Z}^r_{A_i}, \mathbf{M}_{A_i}\mathbf{Y}^r_{A_i}$ to the competitors [131, 132], which allows pre-multiplying the original data,

$$\mathbf{Z}^r = \begin{bmatrix} \mathbf{Z}^r_{A_1} \\ \vdots \\ \mathbf{Z}^r_{A_n} \end{bmatrix} \text{ and } \mathbf{Y}^r = \begin{bmatrix} \mathbf{Y}^r_{A_1} \\ \vdots \\ \mathbf{Y}^r_{A_n} \end{bmatrix},$$

by $\mathbf{M} = [\mathbf{M}_{A_1}, \dots, \mathbf{M}_{A_n}]$, since

$$\mathbf{MZ}^r = \mathbf{M}_{A_1}\mathbf{Z}^r_{A_1} + \cdots + \mathbf{M}_{A_n}\mathbf{Z}^r_{A_n}. \tag{2.5}$$

The same holds for the multiplication $\mathbf{MY}^r$, $\mathbf{M} \in \mathbb{R}^{k\times\sum_{i=1}^n T_{A_i}}, \mathbf{Z}^r \in \mathbb{R}^{\sum_{i=1}^n T_{A_i}\times M}, \mathbf{Y}^r \in \mathbb{R}^{\sum_{i=1}^n T_{A_i}\times N}$. This definition of $\mathbf{M}$ is possible because when multiplying $\mathbf{M}$ and $\mathbf{Z}^r$, the jth column of $\mathbf{M}$ only multiplies the jth row of $\mathbf{Z}^r$. For some statistical learning algorithms, a property of such a matrix is the orthogonality, i.e., $\mathbf{M}^{-1} = \mathbf{M}^\top$. Model fitting is then performed with this new representation of the data, which preserves the solution to the problem. This is true of the linear regression model because the multivariate least squares estimate for the linear regression model with covariate matrix $\mathbf{MZ}^r$ and target variable $\mathbf{MY}^r$ is

$$\hat{\mathbf{B}}_{\text{LS}} = \left((\mathbf{Z}^r)^\top \mathbf{Z}^r\right)^{-1} \left((\mathbf{Z}^r)^\top \mathbf{Y}^r\right), \tag{2.6}$$

which is also the multivariate least squares estimate for the coefficients of a linear regression considering data matrices $\mathbf{Z}^r$ and $\mathbf{Y}^r$, respectively. Despite this property, the application in Least Absolute Shrinkage and Selection Operator (LASSO) regression does not guarantee that the sparsity of the coefficients is preserved, and careful analysis is needed to ensure the correct estimation of the model [127]. Liu et al. [133] discussed attacks based on prior knowledge, in which a data owner estimates $\mathbf{M}$ by knowing a small collection of original data records. Furthermore, when considering the linear regression model for which $\mathbf{Z} = [\mathbf{Z}_{A_1}, \dots, \mathbf{Z}_{A_n}]$ and $\mathbf{Y} = [\mathbf{Y}_{A_1}, \dots, \mathbf{Y}_{A_n}]$, i.e., data is split by features, it is not possible to define a matrix $\mathbf{M}^* = [\mathbf{M}^*_{A_1}, \dots, \mathbf{M}^*_{A_n}] \in \mathbb{R}^{k \times T}$ and then privately compute $\mathbf{M}^*\mathbf{Z}$ and $\mathbf{M}^*\mathbf{Y}$, because as explained, the jth column of $\mathbf{M}^*$ multiplies the jth row of $\mathbf{Z}$, which, in this case, consists of data coming from different owners.

Similarly, if the data owners observe different features, a linear programming problem can be solved in such a way that individual data owners multiply their data $\mathbf{X}_{A_i} \in \mathbb{R}^{T \times M_{A_i}}$ by a private random matrix $\mathbf{N}_{A_i} \in \mathbb{R}^{M_{A_i} \times s}$ (with a jointly defined value s) and, then, shares $\mathbf{X}_{A_i}\mathbf{N}_{A_i}$ [134], $i \in \{1, ..., n\}$, which is equivalent to post-multiplying the original dataset $\mathbf{X} = [\mathbf{X}_{A_1}, ..., \mathbf{X}_{A_n}]$ by $\mathbf{N} = [\mathbf{N}_{A_1}^\top, \dots, \mathbf{N}_{A_n}^\top]^\top$, which represents the joining of $\mathbf{N}_{A_i}, i \in \{1, \dots, n\}$, through a row-wise operation. However, the obtained solution is in a different space, and it needs to be recovered by multiplying it by the corresponding $\mathbf{N}_{A_i}, i \in \{1, ..., n\}$. For linear regression, which models the relationship between the covariates $\mathbf{Z} \in \mathbb{R}^{T \times M}$ and the target $\mathbf{Y} \in \mathbb{R}^{T \times N}$, this algorithm corresponds to solving a linear regression that models the relationship between $\mathbf{ZN_z}$ and $\mathbf{YN_y}$. That is,the solution is given by

$$\hat{\mathbf{B}}'_{\text{LS}} = \arg\min_{\mathbf{B}} \left(\frac{1}{2}\|\mathbf{YN_y} - \mathbf{ZN_zB}\|_2^2\right), \tag{2.7}$$

where $\mathbf{ZN_z}$ and $\mathbf{YN_y}$ are shared matrices. Two private matrices $\mathbf{N_z} \in \mathbb{R}^{M \times s}$, $\mathbf{N_y} \in \mathbb{R}^{N \times w}$ are required to transform the data, since the number of columns for $\mathbf{Z}$ and $\mathbf{Y}$ is different (s and w values are jointly defined). The problem is that the multivariate least squares estimate for (2.7) is given by

$$\hat{\mathbf{B}}'_{\text{LS}} = \left((\mathbf{ZN_z})^\top (\mathbf{ZN_z})\right)^{-1} \left((\mathbf{ZN_z})^\top (\mathbf{YN_y})\right) = (\mathbf{N_z})^{-1} \underbrace{(\mathbf{Z}^\top\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{Y}}_{= \arg\min_{\mathbf{B}} \left(\frac{1}{2}\|\mathbf{Y} - \mathbf{ZB}\|_2^2\right)} \mathbf{N_y}, \tag{2.8}$$

which implies that this transformation does not preserve the coefficients of the linear regression considering data matrices $\mathbf{Z}$ and $\mathbf{Y}$, respectively, and therefore $\mathbf{N}_z$ and $\mathbf{N}_y$ would have to be shared.

Generally, data transformation is performed through the generation of random matrices that pre- or post- multiply the private data. However, there are other techniques through which data are transformed with matrices defined according to that data, as with principal component analysis (PCA). PCA is a widely used statistical procedure for reducing the dimensions of data, by applying an orthogonal transformation that retains as much data variance as possible. Considering the matrix $\mathbf{W} \in \mathbb{R}^{M \times M}$ of the eigenvectors of the covariance matrix $\mathbf{Z}^\top\mathbf{Z}$, $\mathbf{Z} \in \mathbb{R}^{T \times M}$, PCA can be used to represent the data by L variables performing $\mathbf{ZN}_L$, where $\mathbf{N}_L$ denotes the first L columns of $\mathbf{W}$, $L \in \{1, ..., M\}$. For data split by records, Dwork et al. [135] suggested a differentially private PCA, assuming that each data owner takes a random sample of the fitting records to form the covariate matrix. In order to protect the covariance matrix, one can add Gaussian noise to this matrix (determined without sensible data sharing), leading to the computation of the principal directions of the noisy covariance matrix. To finalize the process, the data owners multiply the sensible data by said principal directions before feeding the data into the model fitting. Nevertheless, the application to collaborative linear regression with data split by features would require sharing the data when computing the $\mathbf{Z}^\top\mathbf{Z}$ matrix, since $\mathbf{Z}^\top$ is divided by rows. Furthermore, as explained in (2.7) and (2.8), it is difficult to recover the original linear regression model by performing the estimation of the coefficients using transformed covariates and target matrices, through post-multiplication by random matrices.

Regarding the data normalization techniques mentioned above, Zhu et al. [136] proposed that data owners mask their data by using z-score normalization, followed by the sum of random noise (from uniform or Gaussian distributions), to allow greater control over their data. The data can then be shared with a recommendation system that fits the model. However, the noise does not meet the differential privacy definition (see A.1).

For data collected by different sensors (e.g., smart meters or mobile users) it is common to proceed to the aggregation of data through privacy-preserving techniques – for instance, by adding carefully calibrated Laplacian noise to each time series [137, 138]. The addition of noise to the data is an appealing technique given its easy application. However, even if this noise meets the definition of differential privacy, there is no guarantee that the resulting model will perform well.

2.2.2 Secure Multi-party Computation Protocols

In secure multi-party computations, intermediate calculations required by the fitting algorithms, which require data owners to jointly compute a function over their data, are performed through protocols for secure operations, such as matrix addition or multiplication. In these approaches, the encryption of the data occurs while fitting the model, instead of as a pre-processing step, as with the data transformation methods described in the previous section.

Linear Algebra-based Protocols

The simplest secure multi-party computation protocols are based on linear algebra and address the situation where matrix operations with confidential data are necessary. Du et al. [117] proposed secure protocols for product $\mathbf{A}.\mathbf{C}$ and inverse of the sum $(\mathbf{A}+\mathbf{C})^{-1}$, for any two private matrices $\mathbf{A}$ and $\mathbf{C}$ with appropriate dimensions. The aim is to fit a (ridge) linear regression between two data owners who observe different covariates but share the target variable. Essentially, the $\mathbf{A}.\mathbf{C}$ protocol transforms the product of matrices, $\mathbf{A} \in \mathbb{R}^{m\times s}$, $\mathbf{C} \in \mathbb{R}^{s\times k}$, into a sum of matrices, $\mathbf{V}_a+\mathbf{V}_c$, that are equally secret, $\mathbf{V}_a, \mathbf{V}_c \in \mathbb{R}^{m\times k}$. However, since the estimate of the coefficients for linear regression with covariate matrix $\mathbf{Z}\in \mathbb{R}^{T\times M}$ and target matrix $\mathbf{Y}\in \mathbb{R}^{T\times N}$ is

$$\hat{\mathbf{B}}_{\text{LS}} = (\mathbf{Z}^\top\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{Y}, \tag{2.9}$$

the $\mathbf{A}.\mathbf{C}$ protocol is used to perform the computation of $\mathbf{V}_a, \mathbf{V}_c$ such that

$$\mathbf{V}_a + \mathbf{V}_c = (\mathbf{Z}^\top\mathbf{Z}) \ , \tag{2.10}$$

which requires the definition of an $(\mathbf{A}+\mathbf{C})^{-1}$ protocol to compute

$$(\mathbf{Z}^\top\mathbf{Z})^{-1} = (\mathbf{V}_a + \mathbf{V}_c)^{-1}. \tag{2.11}$$

For the $\mathbf{A}.\mathbf{C}$ protocol, $\mathbf{A} \in \mathbb{R}^{m\times s}$, $\mathbf{C} \in \mathbb{R}^{s\times k}$, there are two different formulations, according to the existence, or not, of a third entity. In cases where only two data owners perform the protocol, a random matrix $\mathbf{M} \in \mathbb{R}^{s\times s}$ is jointly generated and the $\mathbf{A}.\mathbf{C}$ protocol achieves the following results, by dividing the $\mathbf{M}$ and $\mathbf{M}^{-1}$ into two matrices with the same dimensions:

$$\mathbf{AC} = \mathbf{AMM^{-1}C} = \mathbf{A}[\mathbf{M}_{\text{left}}, \mathbf{M}_{\text{right}}] \begin{bmatrix} (\mathbf{M}^{-1})_{\text{top}} \\ (\mathbf{M}^{-1})_{\text{bottom}} \end{bmatrix} \mathbf{C} \tag{2.12}$$

$$= \mathbf{AM}_{\text{left}}(\mathbf{M}^{-1})_{\text{top}}\mathbf{C} + \mathbf{AM}_{\text{right}}(\mathbf{M}^{-1})_{\text{bottom}}\mathbf{C}\ , \tag{2.13}$$

where $\mathbf{M}_{\text{left}}$ and $\mathbf{M}_{\text{right}}$ respectively represent the left and right part of $\mathbf{M}$, and $(\mathbf{M}^{-1})_{\text{top}}$ and $(\mathbf{M}^{-1})_{\text{bottom}}$ respectively denote the top and bottom part of $\mathbf{M}^{-1}$. In this case,

$$\mathbf{V}_a = \mathbf{AM}_{\text{left}}(\mathbf{M}^{-1})_{\text{top}}\mathbf{C}, \tag{2.14}$$

is derived by the first data owner, and

$$\mathbf{V}_c = \mathbf{AM}_{\text{right}}(\mathbf{M}^{-1})_{\text{bottom}}\mathbf{C}, \tag{2.15}$$

by the second data owner. Otherwise, a third entity is assumed to generate random matrices $\mathbf{R}_a, \mathbf{r}_a$ and $\mathbf{R}_c, \mathbf{r}_c$, such that

$$\mathbf{r}_a + \mathbf{r}_c = \mathbf{R}_a\mathbf{R}_c, \tag{2.16}$$

which are sent to the first and second data owners, respectively, $\mathbf{R}_a \in \mathbb{R}^{m\times s}$, $\mathbf{R}_c \in \mathbb{R}^{s\times k}$, $\mathbf{r}_a, \mathbf{r}_c \in \mathbb{R}^{m\times k}$. In this case, the data owners start by trading the matrices $\mathbf{A}+\mathbf{R}_a$ and $\mathbf{C}+\mathbf{R}_c$, and then the second data owner randomly generates a matrix $\mathbf{V}_c$ and sends

$$\mathbf{T} = (\mathbf{A}+\mathbf{R}_a)\mathbf{C} + (\mathbf{r}_c - \mathbf{V}_c)\ , \tag{2.17}$$

to the first data owner in such a way that, at the end of the $\mathbf{A}.\mathbf{C}$ protocol, the first data owner keeps the information

$$\mathbf{V}_a = \mathbf{T} + \mathbf{r}_a - \mathbf{R}_a(\mathbf{C}+\mathbf{R}_c)\ , \tag{2.18}$$

and the second keeps $\mathbf{V}_c$ (since the sum of $\mathbf{V}_a$ with $\mathbf{V}_c$ is $\mathbf{AC}$).

Finally, the $(\mathbf{A}+\mathbf{C})^{\mathbf{-1}}$ protocol considers two steps, where $\mathbf{A}, \mathbf{C} \in \mathbb{R}^{m\times k}$. Initially, the matrix $(\mathbf{A}+\mathbf{C})$ is jointly converted to $\mathbf{P}(\mathbf{A}+\mathbf{C})\mathbf{Q}$ using two random matrices, $\mathbf{P}$ and $\mathbf{Q}$, which are only known to the second data owner preventing the first one from learning matrix $\mathbf{C}$, $\mathbf{P} \in \mathbb{R}^{r\times m}, \mathbf{Q} \in \mathbb{R}^{k\times t}$. The results of $\mathbf{P}(\mathbf{A}+\mathbf{C})\mathbf{Q}$ are known only by the first data owner, who can conduct the inverse computation $\mathbf{Q}^{-1}(\mathbf{A}+\mathbf{C})^{-1}\mathbf{P}^{-1}$. In the following step, the data owners jointly remove $\mathbf{Q}^{-1}$ and $\mathbf{P}^{-1}$ and get $(\mathbf{A}+\mathbf{C})^{\mathbf{-1}}$. Both steps can be achieved by applying the $\mathbf{A}.\mathbf{C}$ protocol. Although these protocols are efficient techniques for solving problems with a shared target variable, one cannot say the same when $\mathbf{Y}$ is private, as further elaborated in Section 2.3.2.

Another example of secure protocols for producing private matrices can be found in [118]. Their protocol applies data from multiple owners who observe different covariates and target features – which are also assumed to be secret. The proposed protocol allows two data owners, with correspondent data matrix $\mathbf{A}$ and $\mathbf{C}$, $\mathbf{A} \in \mathbb{R}^{m\times k}$, $\mathbf{C} \in \mathbb{R}^{m\times s}$, to perform the multiplication $\mathbf{A}^\top\mathbf{C}$ as follows: (i) the first data owner generates $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_g]$, $\mathbf{W} \in \mathbb{R}^{m\times g}$, such that

$$\mathbf{w}_i^\top \mathbf{A}_j = \mathbf{0} \ , \tag{2.19}$$

where $\mathbf{A}_j$ is the jth column of $\mathbf{A}$ matrix, $i \in \{1, ..., g\}$ and $j \in \{1, ..., k\}$, and then sends $\mathbf{W}$ to the second owner; (ii) the second data owner computes $(\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C}$ and shares it; and (iii) the first data owner performs

$$\mathbf{A}^\top(\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C}= \mathbf{A}^\top\mathbf{C} - \underbrace{\mathbf{A}^\top\mathbf{W}\mathbf{W}^\top\mathbf{C}}_{=\mathbf{0},\text{ since }\mathbf{A}^\top\mathbf{W}=\mathbf{0}} = \mathbf{A}^\top\mathbf{C} \ , \tag{2.20}$$

without the possibility of recovering $\mathbf{C}$, since the $rank((\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C}) = m - g$. To generate $\mathbf{W}$, Karr et al. [118] suggested selecting g columns from the $\mathbf{Q}$ matrix, computed by $\mathbf{QR}$ decomposition of the private matrix $\mathbf{C}$, and excluding the first k columns. Furthermore, the authors defined the optimal value for g according to the number of linearly independent equations (represented by NLIE) on the other data owner's data. The second data owner obtains $\mathbf{A}^\top\mathbf{C}$ (providing ks values, since $\mathbf{A}^\top\mathbf{C} \in \mathbb{R}^{k\times s}$) and receives $\mathbf{W}$, knowing that $\mathbf{A}^\top\mathbf{W} = 0$ (which contains kg values). That is,

$$\text{NLIE(Owner\#1)} = ks + kg. \tag{2.21}$$

Similarly, the first data owner receives $\mathbf{A}^\top\mathbf{C}$ (providing ks values) and $(\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C}$ (providing $s(m - g)$ values since $(\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C} \in \mathbb{R}^{m\times s}$ and $rank(\mathbf{W}) = m - g$). That is,

$$\text{NLIE(Owner\#2)} = ks + s(m - g). \tag{2.22}$$

Karr et al. [118] determined the optimal value for g by assuming that both data owners equally share NLIE, so that no agent benefits from the order assumed when running the protocol:

$$|\text{NLIE(Owner\#1)} - \text{NLIE(Owner\#2)}| = 0 \ , \tag{2.23}$$

which allows the optimal value $g^* = \frac{sm}{k+s}$ to be obtained.

An advantage to this approach, when compared to the one proposed by [117], is that $\mathbf{W}$ is simply generated by the first data owner, while the invertible matrix $\mathbf{M}$ proposed by [117] needs to be agreed upon by both parties, which entails substantial communication costs when the number of records is high.

Homomorphic Cryptography-based Protocols

The use of homomorphic encryption was successfully introduced in model fitting and works by encrypting the original values in such a way that the application of arithmetic operations in the public space does not compromise the encryption. Homomorphic encryption ensures that, after the decryption stage (in the private space), the resulting values correspond to the ones obtained by operating on the original data. Consequently, homomorphic encryption is especially responsive and engaging to privacy-preserving applications. As an example, the Paillier homomorphic encryption scheme stipulates that (i) two integer values encrypted with the same public key may be multiplied together to give an encryption of the sum of the values, and (ii) an encrypted value may be taken to some power, yielding encryption of the product of the values. Hall et al. [139] proposed a secure protocol for summing and multiplying real numbers by extending Paillier encryption, aiming to perform the matrix products required to solve linear regression for data divided by features or records.

Equally based in Paillier encryption, the work of Nikolaenko et al. [140] proposed a scheme whereby two parties can correctly perform their tasks without teaming up to discover private data: a crypto-service provider (i.e., a party that provides software- or hardware-based encryption and decryption services) and an evaluator (i.e., a party who runs the learning algorithm). With this scheme, secure linear regression can be performed for data split by records. Similarly, Chen et al. [141] used Paillier and ElGamal encryption to fit the coefficients of ridge linear regression while including these entities. In both works, the use of the crypto-service provider is prompted by assuming that the evaluator does not corrupt its computation by producing an incorrect result. Two conditions are required to prevent confidentiality breaches: the crypto-service provider must publish the system keys correctly, and there can be no collusion between the evaluator and the crypto-service provider. The data can be reconstructed if the crypto-service provider supplies correct keys to a curious evaluator. For data divided by features, [142] extended the approach of [140] by designing a secure multi/two-party inner product.

Jia et al. [143] explored a privacy-preserving data classification scheme with a support vector machine, to ensure that the data owners can successfully conduct data classification without exposing their learned models to a "tester", while the "testers" keep their data private. For example, a hospital (owner) can create a model to learn the relation between a set of features and the existence of a disease, and another hospital (tester) can use this model to obtain forecasting values, without any knowledge about the model. The method is supported by cryptography-based protocols for secure computation of multivariate polynomial functions, but unfortunately, this only works for data split by records.

Li and Cao [144] addresses the privacy-preserving computation of the sum and the minimum of multiple time series collected by different data owners, by combining homomorphic encryption with a novel key management technique to support large data dimensions. These statistics with a privacy-preserving solution for individual user data are quite useful for exploring mobile sensing in different applications such as environmental monitoring (e.g., the average level of air pollution in an area), traffic monitoring (e.g., the highest moving speed during rush hour), healthcare (e.g., the number of users infected by a flu), etc. Liu et al. [145] and Li et al. [146] explored similar approaches based on Paillier or ElGamal encryption concerning their application to smart grids. However, the estimation of models such as the linear regression model also requires protocols for the secure product of matrices. Homomorphic cryptography was further explored to solve secure linear programming problems through intermediate steps of the simplex method, which optimizes the problem by using slack variables, tableaus, and pivot variables [147]. However, the author observed that the proposed protocols are not viable when solving linear programming problems with numerous variables and constraints, which are common in practice.

Aono et al. [148] combined homomorphic cryptography with differential privacy in order to deal with data split by records. In summary, if data are split by records, as illustrated in Figure 2.1, each ith data owner observes the covariates $\mathbf{Z}^r_{A_i}$ and target variable $\mathbf{Y}^r_{A_i}$, $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times M}, \mathbf{Y}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times N}, i \in \{1, ..., n\}$. Then, $(\mathbf{Z}^r_{A_i})^\top \mathbf{Z}^r_{A_i}$ and $(\mathbf{Z}^r_{A_i})^\top \mathbf{Y}^r_{A_i}$ are computed and Laplacian noise is added to them. This information is encrypted and sent to the cloud server, which works on the encrypted domain, summing all the matrices received. Finally, the server provides the result of this sum to a client who decrypts it and obtains relevant information to perform the linear regression, i.e., $\sum_{i=1}^n (\mathbf{Z}^r_{A_i})^\top \mathbf{Z}^r_{A_i}$, $\sum_{i=1}^n (\mathbf{Z}^r_{A_i})^\top \mathbf{Y}^r_{A_i}$, etc. However, the addition of noise can result in a poor estimation of the coefficients, limiting the performance of the model. Furthermore, this approach is not valid when data are divided by features, because $\mathbf{Z}^\top \mathbf{Z} \neq \sum_{i=1}^n \mathbf{Z}^\top_{A_i} \mathbf{Z}_{A_i}$ and $\mathbf{Z}^\top \mathbf{Y} \neq \sum_{i=1}^n \mathbf{Z}^\top_{A_i} \mathbf{Y}_{A_i}$.

In summary, cryptography-based methods are usually robust to confidentiality breaches but may require a third party to generate keys, as well as external entities to per- form the computations in the encrypted domain. Furthermore, the high computational complexity is a challenge when dealing with real applications [147, 149, 150].

2.2.3 Decomposition-based Methods

In decomposition-based methods, problems are solved by breaking them up into smaller sub-problems and solving each separately, either in parallel or in sequence. Consequently, private data are naturally distributed between the data owners. However, this natural division requires sharing intermediate information. For that reason, some approaches combine decomposition-based methods with data transformation or homomorphic cryptography-based methods; here, we focus on these methods separately.

ADMM Method

The ADMM, described in Section II.3.2 in Prologue II, is a powerful algorithm that circumvents problems without a closed-form solution, such as the LASSO regression. The algorithm is efficient and well suited for distributed convex optimization, in particular for large-scale statistical problems. Undeniably, ADMM provides a desirable formulation for parallel computing [105]. However, it is not possible to ensure continuous privacy, since the ADMM requires intermediate calculations, allowing the most curious competitors to recover the data after enough iterations by solving non-linear equation systems [151]. An ADMM-based distributed LASSO algorithm, in which each data owner only communicates with its neighbor to protect data privacy, is described by [152], with applications in signal processing and wireless communications. Unfortunately, this approach is only valid in cases where data are distributed by records.

The concept of differential privacy was also explored in the ADMM by introducing randomization when computing the primal variables. That is, during the iterative process, each data owner estimates the corresponding coefficients and perturbs them by adding random noise [153]. However, these local randomization mechanisms can result in a non-convergent algorithm with poor performance even under moderate privacy guarantees. To address these concerns, Huang et al. [154] used an approximate augmented Lagrangian function and Gaussian mechanisms with time-varying variance. Nevertheless, the addition of noise is insufficient to guarantee privacy, as a competitor can potentially use the results from all iterations to infer information [155].

Zhang et al. [156] recently combined a variant of the ADMM with homomorphic encryption for cases where data are divided by records. As explained by the authors, however, the incorporation of their mechanism in decentralized optimization under data divided by features is quite difficult. Whereas for data split by records, the algorithm only requires sharing the coefficients, the exchange of coefficients in data split by features is insufficient, since each data owner observes different features. Division by features requires a local estimation of $\mathbf{B}_{A_i}^{k+1} \in \mathbb{R}^{M_{A_i} \times N}$ by using information related to $\mathbf{Z}_{A_j}\mathbf{B}_{A_j}^{k}$, and $\mathbf{Y}$, meaning that, for each new iteration, an ith data owner shares TN new values, instead of $M_{A_i}N$ (from $\mathbf{B}_{A_i}^{k}$), $i, j \in \{1, ..., n\}$. Huo and Liu [157] also combined ADMM with homomorphic encryption and they also consider data split by features. However, after each iteration k, agents decrypt the matrix $\sum_j \mathbf{Z}_{A_j}\mathbf{B}_{A_j}^{k}$, which can provide enough information for a curious agent to recover the original data.

For data split by features, Zhang and Wang [158] proposed a probabilistic forecasting method that combines ridge linear quantile regression with the ADMM. The output is a set of quantiles instead of a unique value (usually the expected value). In this case, the ADMM is applied to split the corresponding optimization problem into sub-problems, which are solved by each data owner, assuming that all the data owners communicate with a central node in an iterative process. Consequently, intermediate results are provided, rather than private data. In fact, the authors claimed that their method achieves wind power probabilistic forecasting with off-site information in a privacy-preserving and distributed fashion. However, the authors did not conduct an in-depth analysis of the method, as shown in 2.3. Furthermore, their method assumes that the central node knows the target matrix.

Newton-Raphson Method

The ADMM is now a standard technique used in research on distributed computing in statistical learning, but it is not the only one. For generalized linear models, distributed optimization for model fitting has been efficiently achieved through the Newton–Raphson method, which minimizes a twice differentiable forecast error function E between the true values $\mathbf{Y}$ and the forecasted values given by the model $\hat{\mathbf{Y}} = f(\mathbf{B}, \mathbf{Z})$ using a set of covariates $\mathbf{Z}$, including lags of $\mathbf{Y}$. $\mathbf{B}$ is the coefficient matrix, which is updated iteratively. The estimate for $\mathbf{B}$ at iteration $k+1$, represented by $\mathbf{B}^{k+1}$, is given by

$$\mathbf{B}^{k+1} = \mathbf{B}^{k} - (\nabla^2 E(\mathbf{B}^{k}))^{-1} \nabla E(\mathbf{B}^{k}) \ , \tag{2.24}$$

where ∇E and $\nabla^2 E$ are the gradient and Hessian of E, respectively. With certain properties, convergence to a certain global minima can be guaranteed [159].

In order to enable distributed optimization, ∇E and $\nabla^2 E$ must be decomposable over multiple data owners. That is, these functions can be rewritten as the sum of functions that depend exclusively on local data from each data owner. Slavkovic et al. [160] proposed a secure logistic regression approach for data split by records and features by using secure multi-party computation protocols during iterations of the Newton–Raphson method. Although distributed computing is feasible, there is no sufficient guarantee of data privacy, because it is an iterative process. While a single iteration cannot reveal private information, sufficient iterations can: in a logistic regression with data split by features, for each iteration k the data owners exchange the matrix $\mathbf{Z}_{A_i}\mathbf{B}^{k}_{A_i}$, making it possible to recover the local data $\mathbf{Z}_{A_i}$ after enough iterations [122].

An example of an earlier promising work that combined logistic regression with the Newton-Raphson method for data distributed by records was the Grid binary LOgistic REgression (GLORE) framework [119]. The GLORE model is based on model sharing rather than patient-level data, and it has motivated subsequent improvements. Some of these continue to suffer from confidentiality breaches on intermediate results, and others resort to protocols for matrix addition and multiplication. Later, Li et al. [161] explored the issue concerning the Newton–Raphson method over data distributed by features by considering a server that receives the transformed data and computes the intermediate results, returning them to each data owner. In order to avoid disclosing local data while obtaining an accurate global solution, the authors applied the kernel trick to obtain the global linear matrix, computed using dot products of local records ($\mathbf{Z}_{A_i}\mathbf{Z}^{\top}_{A_i}$), which can be used to solve the dual problem for logistic regression. However, they identified a technical challenge from scaling up the model with a large sample size, since each record requires a parameter.

Gradient-Descent Methods

Different gradient-descent methods have also been explored, aiming to minimize a forecast error function E between the true values $\mathbf{Y}$ and the forecasted values given by the model $\hat{\mathbf{Y}} = f(\mathbf{B}, \mathbf{Z})$ using a set of covariates $\mathbf{Z}$, including lags of $\mathbf{Y}$. The coefficient matrix $\mathbf{B}$ is updated iteratively such that the estimate at iteration $k+1$, $\mathbf{B}^{k+1}$, is given by

$$\mathbf{B}^{k+1} = \mathbf{B}^k + \eta \nabla E(\mathbf{B}^k) \ , \tag{2.25}$$

where η is the learning rate. This allows for parallel computation when the optimization function E is decomposable. A common error function is the multivariate least squared error:

$$E(\mathbf{B}) = \frac{1}{2} \|\mathbf{Y} - f(\mathbf{B}, \mathbf{Z})\|^2. \tag{2.26}$$

With certain properties, convergence to a certain global minima can be guaranteed [162]: (i) E is convex, (ii) ∇E is Lipschitz-continuous with constant L, i.e., for any $\mathbf{F}$, $\mathbf{G}$,

$$\|\nabla E(\mathbf{F}) - \nabla E(\mathbf{G})\|^2 \leq L \|\mathbf{F} - \mathbf{G}\|^2 \ , \tag{2.27}$$

and (iii) $\eta \leq 1/L$.

Han et al. [163] proposed a privacy-preserving linear regression technique for data distributed over features (with shared $\mathbf{Y}$) by combining distributed gradient descent with secure protocols, based on pre- or post-multiplication of the data by random private matrices. Wei et al. [164] consider the case in which an agent wants to improve its logistic regression model with data from a second agent. The authors combine a gradient-based algorithm with homomorphic encryption to protect data from the first agent. However, after each iteration the second agent provides $\mathbf{Z}_{A_2} \mathbf{B}^k_{A_2}$, which can be enough for the first agent to recover data. Song et al. [165] introduced differential privacy by adding random noise $\mathbf{W}$ in the $\mathbf{B}$ updates:

$$\mathbf{B}^{k+1} = \mathbf{B}^k + \eta \Big(\nabla E(\mathbf{B}^k) + \mathbf{W} \Big). \tag{2.28}$$

When this iterative process uses a few randomly selected samples (or even a single sample), rather than the entire data, the process is known as stochastic gradient descent (SGD). The authors argued that the trade-off between performance and privacy is most pronounced when smaller batches are used. A similar framework was also proposed in [166] to perform probabilistic solar irradiation forecasting by using a neural network that combines data split by records.

2.3 Collaborative Forecasting with VAR: Privacy Analysis

This section presents a privacy analysis of collaborative forecasting with the VAR model, a model for the analysis of multivariate time series, described in Section II.3.2 of Prologue II. The VAR model is not only used for forecasting tasks in different domains (and with significant improvements over univariate autoregressive models), but also for structural inference, where the main objective is to explore certain assumptions about the causal structure of the data [167]. A variant with LASSO regularization is also covered. We critically evaluate the methods described in Section 2.2 from a mathematical and numerical point of view.

Using the notation of Section II.3.2 of Prologue II, each of the n data owners is assumed to use the same number of lags p to fit a LASSO-VAR model with a total number of

T records. $\mathbf{Y}_{A_i} \in \mathbb{R}^{T\times 1}$ and $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$ respectively denote the target and covariate matrix for the ith data owner. In LASSO-VAR, the covariates and target matrices are obtained by joining the individual matrices column-wise, i.e., $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$ and $\mathbf{Y} = [\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$. For distributed computation, the coefficient matrix of data owner i is denoted by $\mathbf{B}_{A_i} \in \mathbb{R}^{p\times n}, i \in \{1, \ldots, n\}$.

2.3.1 Data Transformation with Noise Addition

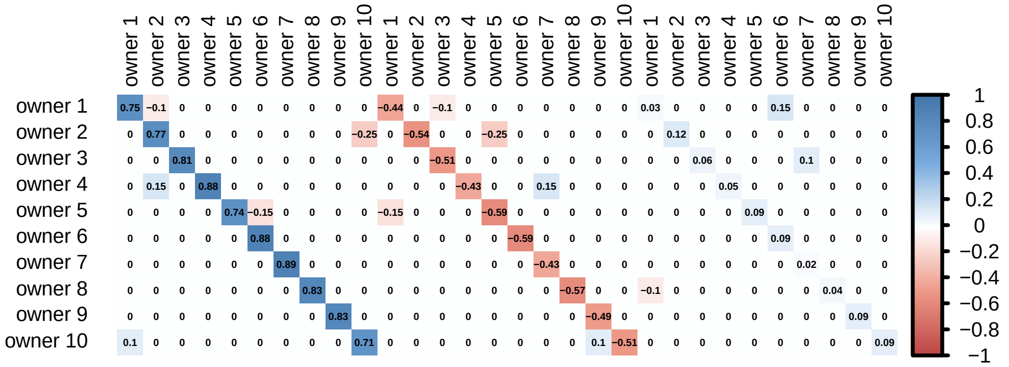
This section presents experiments with simulated data, wind energy data from Global Energy Forecasting Competition 2014 (GEFCom2014), and solar energy data collected from a smart grid pilot in Portugal. The objective was to quantify the impact of data distortion (through noise addition) on the model forecasting skill.

Synthetic Data

An experiment was performed to add random noise from a Gaussian distribution with zero mean and variance b^2, a Laplace distribution with zero mean and scale parameter b and a uniform distribution with support $[-b, b]$ – represented by $\mathcal{N}(0, b^2)$, $\mathcal{L}(0, b)$ and $\mathcal{U}(-b, b)$, respectively. Synthetic data generated by VAR processes were used to measure the differences between the coefficients' values when adding noise to the data. The simplest case considered a VAR with two data owners and two lags, described by

$$\begin{pmatrix} y_{1,t} & y_{2,t} \end{pmatrix} = \begin{pmatrix} y_{1,t-1} & y_{2,t-1} & y_{1,t-2} & y_{2,t-2} \end{pmatrix} \begin{pmatrix} 0.5 & 0.3 \\ 0.3 & 0.75 \\ -0.3 & -0.05 \\ -0.1 & -0.4 \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} & \varepsilon_{2,t} \end{pmatrix} .$$

The second case included ten data owners and three lags and introduced a high percentage of null coefficients ($\approx 86\%$). Figure 2.2 illustrates the considered coefficients. Since a specific configuration can generate various distinct trajectories, 100 simulations were performed for each specified VAR model, with 20,000 timestamps each. For both simulated datasets, the errors $\boldsymbol{\varepsilon}_t$ were assumed to follow a multivariate normal distribution with a zero mean vector and a covariance matrix equal to the identity matrix of appropriate dimensions. A distributed ADMM (detailed in Section II.3.2 of Prologue II) was used to estimate the LASSO-VAR coefficients, considering two different noise characterizations, $b \in \{0.2, 0.6\}$.



	owner 1	owner 2	owner 3	owner 4	owner 5	owner 6	owner 7	owner 8	owner 9	owner 10	owner 1	owner 2	owner 3	owner 4	owner 5	owner 6	owner 7	owner 8	owner 9	owner 10	owner 1	owner 2	owner 3	owner 4	owner 5	owner 6	owner 7	owner 8	owner 9	owner 10
owner 1	0.75	−0.1	0	0	0	0	0	0	0	0	−0.44	0	−0.1	0	0	0	0	0	0	0	0.03	0	0	0	0	0.15	0	0	0	0
owner 2	0	0.77	0	0	0	0	0	0	0	−0.25	0	−0.54	0	0	−0.25	0	0	0	0	0	0	0.12	0	0	0	0	0	0	0	0
owner 3	0	0	0.81	0	0	0	0	0	0	0	0	0	−0.51	0	0	0	0	0	0	0	0	0	0.06	0	0	0	0.1	0	0	0
owner 4	0	0.15	0	0.88	0	0	0	0	0	0	0	0	0	−0.43	0	0	0.15	0	0	0	0	0	0	0.05	0	0	0	0	0	0
owner 5	0	0	0	0	0.74	−0.15	0	0	0	0	−0.15	0	0	0	−0.59	0	0	0	0	0	0	0	0	0	0.09	0	0	0	0	0
owner 6	0	0	0	0	0	0.88	0	0	0	0	0	0	0	0	0	−0.59	0	0	0	0	0	0	0	0	0	0.09	0	0	0	0
owner 7	0	0	0	0	0	0	0.89	0	0	0	0	0	0	0	0	0	−0.43	0	0	0	0	0	0	0	0	0	0.02	0	0	0
owner 8	0	0	0	0	0	0	0	0.83	0	0	0	0	0	0	0	0	0	−0.57	0	0	−0.1	0	0	0	0	0	0	0.04	0	0
owner 9	0	0	0	0	0	0	0	0	0.83	0	0	0	0	0	0	0	0	0	−0.49	0	0	0	0	0	0	0	0	0	0.09	0
owner 10	0.1	0	0	0	0	0	0	0	0	0.71	0	0	0	0	0	0	0	0	0.1	−0.51	0	0	0	0	0	0	0	0	0	0.09

Figure 2.2: Transpose of the coefficient matrix used to generate the $\mathrm{VAR}_{10}(3)$.

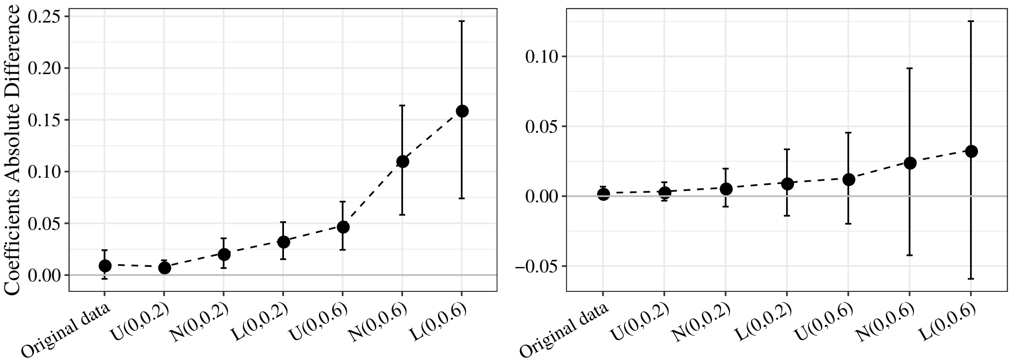


Figure 2.3: Mean ± standard deviation for the absolute difference between the real and estimated coefficients (left: VAR with 2 data owners, right: VAR with 10 data owners).

Figure 2.3 summarizes the mean and the standard deviation of the absolute difference between the real and estimated coefficients for both VAR processes from the 100 simulations. The greater the noise b, the greater the distortion of the estimated coefficients. Moreover, the Laplace distribution, which has desirable properties to make data private according to a differential privacy framework, registered the greater distortion in the estimated model.

Using the original data, the ADMM solution tended to stabilize after 50 iterations, and the value of the coefficients was correctly estimated (the difference was approximately zero). The distorted time series converged faster, but the coefficients deviated from the real ones. In fact, adding noise contributed to decreasing the absolute value of the coefficients. That is, the relationships between the time series weakened.

These experiments allow us to draw conclusions about the use of differential privacy. The Laplace distribution has advantageous properties, since it ensures ε-differential privacy when random noise follows $\mathcal{L}(0, \frac{\Delta f_1}{\varepsilon})$. For the VAR with two data owners, $\Delta f_1 \approx 12$, since the observed values are in the interval $[-6, 6]$. Therefore, $\varepsilon = 20$ when $\mathcal{L}(0, 0.6)$ and $\varepsilon = 15$ when $\mathcal{L}(0, 0.8)$, meaning that the data still encompass much relevant information. Finally, we verified the impact of noise addition on forecasting performance. Figure 2.4 illustrates the improvement of each estimated VAR$_2$(2) model (with and without noise addition) over the autoregressive (AR) model estimated with original time series, in which collaboration is not used. This improvement was measured in terms of the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). In the case of ten data owners and

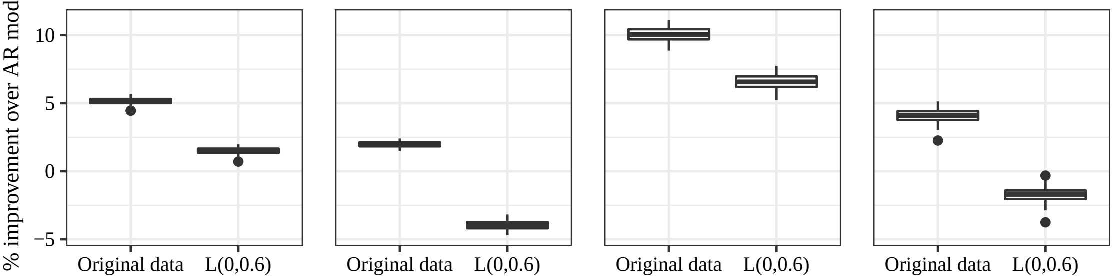
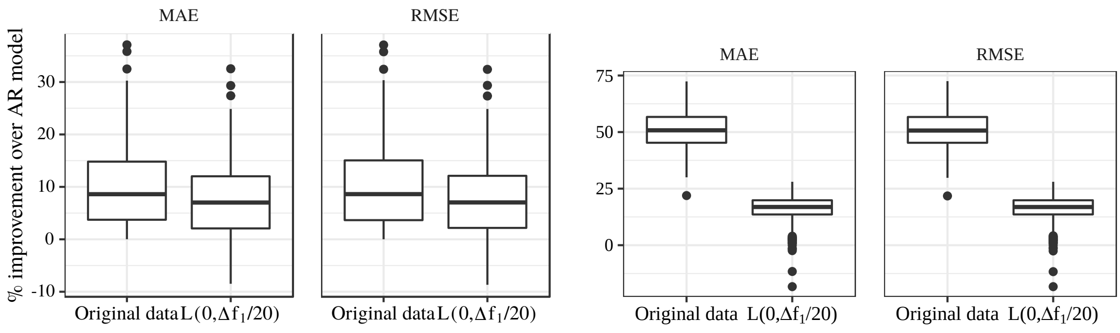


Figure 2.4: Improvement (%) of VAR$_2$(2) model over AR(2) model, in terms of MAE and RMSE for synthetic data.



(a) VAR$_2$(2) model.

(b) VAR$_{10}$(3) model.

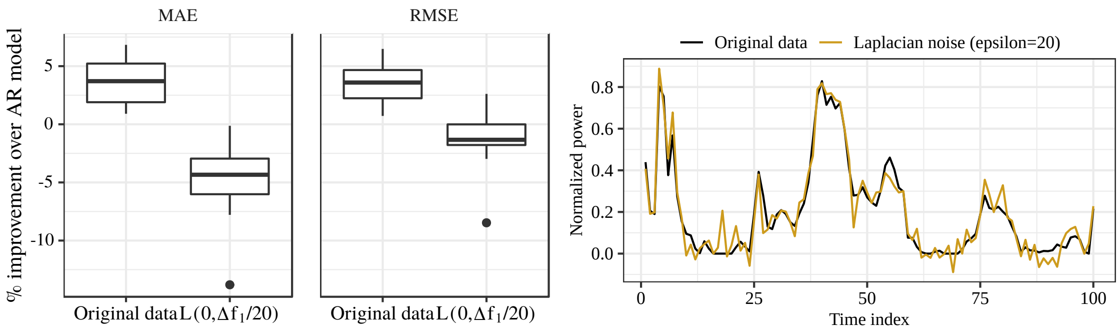
Figure 2.5: Improvement (%) of VAR model over AR model, in terms of MAE and RMSE for synthetic data.

when using data without noise, seven data owners improved their forecasting performance, which was expected from the coefficient matrix in Figure 2.2. When Laplacian noise was applied to the data, only one data owner (the first one) improved its forecasting skill (when compared to the AutoRegressive (AR) model) by using the estimated VAR model. Even though the masked data continued to provide relevant information, the model obtained for the Laplacian noise performed worse than the AR model for the second data owner, making the VAR useless for the majority of the data owners.

However, these results cannot be generalized for all VAR models, especially regarding the illustrated VAR$_{10}$(3), which is very close to the AR(3) model. Given that, we conducted a third experiment, in which 200 random coefficient matrices were generated for a stationary VAR$_2$(2) and VAR$_{10}$(3) following the algorithm proposed by Ansley and Kohn [168]. Usually, the generated coefficient matrix has no null entries and the higher values are not necessarily found on diagonals. Figure 2.5 illustrates the improvement for each data owner when using a VAR model (with and without noise addition) over the AR model. In this case, the percentage of times the AR model performed better than the VAR model with distorted data was smaller, but the degradation of the models was still noticeable, especially in the case with ten data owners.

Wind Power Data

The method was also evaluated in a real dataset, comprising hourly time series of wind power generation in 10 zones, corresponding to 10 wind farms in Australia [25]. These data was used in the Global Energy Forecasting Competition 2014 (GEFCom2014), covering the period from January 1, 2012 to November 30, 2013. The power generation for the next hour was modeled through the VAR model, which combined data from the 10 data owners and considered three consecutive lags (1h, 2h, and 3h), based on the partial correlation discussed in Section II.3.2 of Prologue II. Figure 2.6 (a) summarizes the improvement for the 10 wind power plants over the autoregressive model, in terms of the MAE and RMSE. In both metrics, all data owners improved their forecasting accuracy when using data from other data owners. Although the data obtained after adding Laplacian noise retained its temporal dependency, as illustrated in Figure 2.6 (b), in general the corresponding VAR model was useless for all data owners.



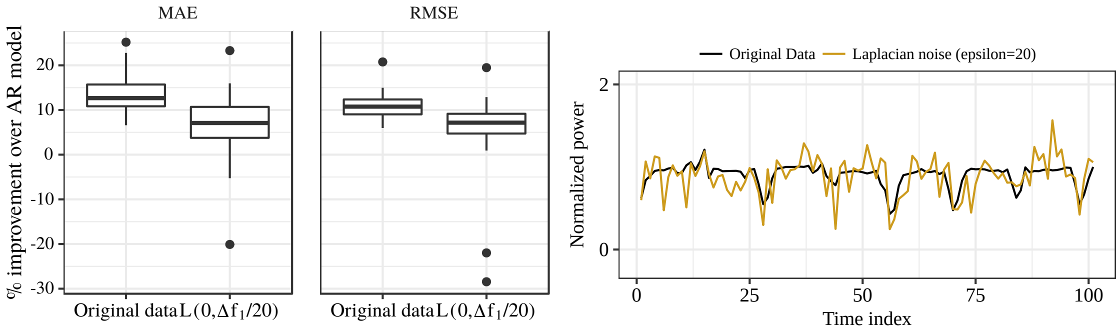
(a) Improvement (%) of VAR_{10} model over AR model, in terms of MAE and RMSE.

(b) Example of the time series.

Figure 2.6: Results for real case-study with wind power time series.

Solar Power Data

Furthermore, the method was evaluated in a real dataset comprising hourly time series of solar power generation from 44 micro-generation units located in Évora city (Portugal), covering the period from February 1, 2011 to March 6, 2013. As in Cavalcante and Bessa [169], records corresponding to a solar zenith angle higher than 90° were removed, in order to take off nighttime hours (i.e., hours without any generation). To make the time series stationary, a normalization of the solar power was applied by using a clear-sky model (see [37]) that gives an estimate of solar power under clear sky conditions at any given time. The power generation for the next hour was modeled through the VAR model, which combined data from the 44 data owners and considered three non-consecutive lags (1h, 2h, and 24h). These lags were selected based on the partial correlation between the multiple lagged time series. Figure 2.7 (a) summarizes the improvement for the 44 solar power plants over the autoregressive model, in terms of the MAE and RMSE. The quartile 25% shows that the MAE improved by at least 10% for 33 of the 44 solar power plants, when the data owners share their observed data. The improvement to the RMSE was not as significant, but is still greater than zero. Although the data obtained after adding Laplacian noise retained its temporal dependency, as illustrated in Figure 2.7 (b), the corresponding VAR model was useless for 4 of the 44 data owners.



(a) Improvement (%) of VAR_{44} model over AR model, in terms of MAE and RMSE.

(b) Example of the normalized time series.

Figure 2.7: Results for real case-study with solar power time series.

When considering the RMSE, 2 of the 44 data owners obtain better results by using an autoregressive model. Once again, the resulting model suffers a significant reduction in terms of forecasting capability.

2.3.2 Linear Algebra-based Protocols

Let us consider a case with two data owners. Since the multivariate least squares estimate for the VAR model with covariates $\mathbf{Z} = [\mathbf{Z}_{A_1}, \mathbf{Z}_{A_2}]$ and target $\mathbf{Y} = [\mathbf{Y}_{A_1}, \mathbf{Y}_{A_2}]$ is

$$\hat{\mathbf{B}}_{\text{LS}} = \left(\begin{bmatrix} \mathbf{Z}_{A_1}^\top \\ \mathbf{Z}_{A_2}^\top \end{bmatrix} [\mathbf{Z}_{A_1}, \mathbf{Z}_{A_2}] \right)^{-1} \left(\begin{bmatrix} \mathbf{Z}_{A_1}^\top \\ \mathbf{Z}_{A_2}^\top \end{bmatrix} [\mathbf{Y}_{A_1}, \mathbf{Y}_{A_2}] \right) \tag{2.29}$$

$$= \begin{pmatrix} \mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_1} & \mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2} \\ \mathbf{Z}_{A_2}^\top \mathbf{Z}_{A_1} & \mathbf{Z}_{A_2}^\top \mathbf{Z}_{A_2} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_1} & \mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2} \\ \mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1} & \mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_2} \end{pmatrix}, \tag{2.30}$$

the data owners need to jointly compute $\mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2}$, $\mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2}$ and $\mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1}$.

As mentioned in the introduction of Section 2.2.2, the work of Du et al. [117] proposed protocols for secure matrix multiplication for situations where two data owners observe the same common target matrix and different confidential covariates. Unfortunately, without assuming a trusted third entity for generating random matrices, the proposed protocol fails when applied to the VAR model. This is because $2(T-1)p$ values of the covariate matrix $\mathbf{Z} \in \mathbb{R}^{T \times 2p}$ are included in the target matrix $\mathbf{Y} \in \mathbb{R}^{T \times 2}$, which is also undisclosed. Additionally, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}$ has $T + p - 1$ unique values instead of Tp – regarding which, see Figure II.10 in Prologue II.

Proposition 1 *Consider a case in which two data owners with private data $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}$ and $\mathbf{Y}_{A_i} \in \mathbb{R}^{T \times 1}$, want to estimate a VAR model without trusting a third entity, $i \in \{1, 2\}$. Assume that the T records are consecutive, as well as the p lags. The multivariate least squares estimate for the VAR model with covariates $\mathbf{Z} = [\mathbf{Z}_{A_1}, \mathbf{Z}_{A_2}]$ and target $\mathbf{Y} = [\mathbf{Y}_{A_1}, \mathbf{Y}_{A_2}]$ requires the computation of $\mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2}$, $\mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2}$ and $\mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1}$.*

If data owners use the protocol proposed by [117] for computing such matrices, then the information exchanged allows to recover data matrices.

Proof As in [117], let us consider a case with two data owners without a third entity generating random matrices.

In order to compute $\mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2}$ both data owners define a matrix $\mathbf{M} \in \mathbb{R}^{T \times T}$ and compute its inverse $\mathbf{M}^{-1}$. Then, the protocol stipulates that

$$\mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2} = \mathbf{Z}_{A_1}^\top \mathbf{M}\mathbf{M}^{\mathbf{-1}} \mathbf{Z}_{A_2} = \mathbf{A}[\mathbf{M}_{\text{left}}, \mathbf{M}_{\text{right}}] \begin{bmatrix} (\mathbf{M}^{-1})_{\text{top}} \\ (\mathbf{M}^{-1})_{\text{bottom}} \end{bmatrix} \mathbf{Z}_{A_2}$$

$$= \underbrace{\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{left}} (\mathbf{M}^{-1})_{\text{top}} \mathbf{Z}_{A_2}}_{\text{derived by Owner \#1}} + \underbrace{\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{right}} (\mathbf{M}^{-1})_{\text{bottom}} \mathbf{Z}_{A_2}}_{\text{derived by Owner \#2}},$$

requiring the data owners to share $\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{right}} \in \mathbb{R}^{p \times T/2}$ and $(\mathbf{M}^{-1})_{\text{top}} \mathbf{Z}_{A_2} \in \mathbb{R}^{T/2 \times p}$, respectively. This implies that each data owner shares $pT/2$ values.

Similarly, the computation of $\mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2}$ implies that the data owners define a matrix $\mathbf{M}^*$, and share $\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{right}}^* \in \mathbb{R}^{p \times T/2}$ and $(\mathbf{M}^{*-1})_{\text{top}} \mathbf{Y}_{A_2} \in \mathbb{R}^{T/2 \times p}$, respectively, providing new $pT/2$ values. This means that Owner #2 receives $\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{right}}$ and $\mathbf{Z}_{A_1}^\top \mathbf{M}_{\text{right}}^*$, i.e., Tp values, and may recover $\mathbf{Z}_{A_1}$, which consists of Tp values and represents a confidentiality breach. Furthermore, when considering a VAR model with p lags, $\mathbf{Z}_{A_1}$ has $T + p - 1$

unique values, meaning there are fewer values to recover. Analogously, Owner #1 may recover $\mathbf{Z}_{A_2}$ through the matrices shared for the computation of $\mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2}$ and $\mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1}$.

Finally, when considering a VAR with p lags, $\mathbf{Y}_{A_i}$ only has p values that are not in $\mathbf{Z}_{A_i}$. While computing $\mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2}$, Owner #1 receives $T/2$ values from $(\mathbf{M}*^{-1})_{\text{top}}\mathbf{Y}_{A_2} \in \mathbb{R}^{T/2\times 1}$, such that a confidentiality breach can occur (in general $T/2 > p$). In the same way, Owner #2 recovers $\mathbf{Y}_{A_1}$ when computing $\mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1}$. □

The main disadvantage of linear algebra-based methods is that they do not take into account that, in the VAR model, both target variables and covariates are private, and that a large proportion of the covariates matrix is determined by knowing the target variables. This means that the data shared between data owners may be enough for competitors to be able to reconstruct the original data. For the method proposed by [118], a consequence of such data is that the assumption $rank\big((\mathbf{I}-\mathbf{W}\mathbf{W}^\top)\mathbf{C}\big) = m{-}g$ may still provide a sufficient number of linearly independent equations on the other data owner's data to recovering the latter's data.

2.3.3 ADMM Method and Central Node

Zhang and Wang [158] offered a promising approach to dealing with the problem of private data during the ADMM iterative process described by (II.53). According to their approach, for each iteration k, each data owner i communicates local results, $\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k$, to the central node, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}, \mathbf{B}_{A_i}^k \in \mathbb{R}^{p\times n}, i \in \{1,\dots,n\}$. Then, the central node computes the intermediate matrices in (II.53b)-(II.53c), i.e.,

$$\overline{\mathbf{H}}^{k+1} = \frac{1}{N+\rho}\left(\mathbf{Y} + \rho\overline{\mathbf{Z}\mathbf{B}}^{k+1} + \rho\mathbf{U}^k\right),$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \overline{\mathbf{Z}\mathbf{B}}^{k+1} - \overline{\mathbf{H}}^{k+1},$$

and returns the matrix $\overline{\mathbf{H}}^k - \overline{\mathbf{Z}\mathbf{B}}^k - \mathbf{U}^k$ to each data owner, in order to update $\mathbf{B}_{A_i}$ in the next iteration, as seen in (II.53a). Figure 2.8 illustrates this method for the LASSO-VAR with three data owners. In this solution, there is no direct exchange of private data. However, as we explain next, not only can the central node recover the original data, but also the individual data owners can obtain a good estimation of the data used by their competitors.

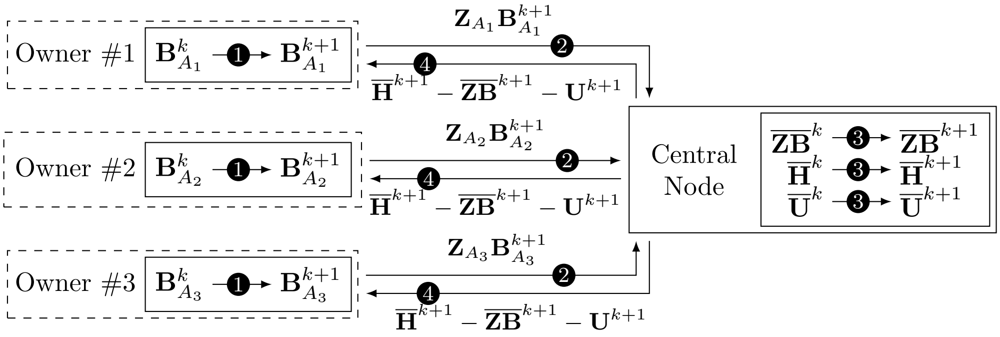

$\mathbf{Z}_{A_1}\mathbf{B}_{A_1}^{k+1}$
Owner #1
$\mathbf{B}_{A_1}^k$ ❶ $\mathbf{B}_{A_1}^{k+1}$
❷
❹
$\overline{\mathbf{H}}^{k+1} - \overline{\mathbf{Z}\mathbf{B}}^{k+1} - \mathbf{U}^{k+1}$
$\mathbf{Z}_{A_2}\mathbf{B}_{A_2}^{k+1}$
Owner #2
$\mathbf{B}_{A_2}^k$ ❶ $\mathbf{B}_{A_2}^{k+1}$
❷
❹
$\overline{\mathbf{H}}^{k+1} - \overline{\mathbf{Z}\mathbf{B}}^{k+1} - \mathbf{U}^{k+1}$
Central Node
$\overline{\mathbf{Z}\mathbf{B}}^k$ ❸ $\overline{\mathbf{Z}\mathbf{B}}^{k+1}$
$\overline{\mathbf{H}}^k$ ❸ $\overline{\mathbf{H}}^{k+1}$
$\overline{\mathbf{U}}^k$ ❸ $\overline{\mathbf{U}}^{k+1}$
$\mathbf{Z}_{A_3}\mathbf{B}_{A_3}^{k+1}$
Owner #3
$\mathbf{B}_{A_3}^k$ ❶ $\mathbf{B}_{A_3}^{k+1}$
❷
❹
$\overline{\mathbf{H}}^{k+1} - \overline{\mathbf{Z}\mathbf{B}}^{k+1} - \mathbf{U}^{k+1}$

Figure 2.8: Distributed ADMM LASSO-VAR with a central node and 3 data owners (related to the algorithm in (II.53)).

Proposition 2 *In the most optimistic scenario, without repeated values in* $\mathbf{Y}_{A_i} \in \mathbb{R}^{T\times 1}$ *and* $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$*, when applying the algorithm from [156] to solve the LASSO-VAR model in* (II.53)*, the central agent can recover the sensible data after*

$$k = \left\lceil \frac{Tp}{Tn - pn} \right\rceil \tag{2.31}$$

iterations, where $\lceil x \rceil$ *denotes the ceiling function.*

Proof Using the notation of Section II.3.2, each of the n data owners is assumed to use the same number of lags p to fit a LASSO-VAR model with a total number of T records. (Importantly, $T > np$; otherwise more coefficients must be determined than system equations.) After k iterations, the central node receives a total of Tnk values from each data owner i, corresponding to $\mathbf{Z}_{A_i}\mathbf{B}^1_{A_i}, \mathbf{Z}_{A_i}\mathbf{B}^2_{A_i}, ..., \mathbf{Z}_{A_i}\mathbf{B}^k_{A_i} \in \mathbb{R}^{T\times n}$, and does not know $pnk + Tp$, corresponding to $\mathbf{B}^1_{A_i}, ..., \mathbf{B}^k_{A_i} \in \mathbb{R}^{p\times n}$ and $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$, respectively, $i \in \{1, ..., n\}$. Given that, the solution of the inequality

$$Tnk \geq pnk + Tp \ , \tag{2.32}$$

in k suggests that a confidentiality breach can occur after

$$k = \left\lceil \frac{Tp}{Tn - pn} \right\rceil \tag{2.33}$$

iterations. Since T tends to be large, k tends to $\lceil p/n \rceil$, which may represent a confidentiality breach if the number of iterations required for the algorithm to converge is greater than $\lceil p/n \rceil$.

□

Proposition 3 *In the most optimistic scenario, without repeated values in* $\mathbf{Y}_{A_i} \in \mathbb{R}^{T\times 1}$ *and* $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$*, when applying the algorithm from [156] to solve the LASSO-VAR model in* (II.53)*, the data owners can recover sensible data from competitors after*

$$k = \left\lceil \frac{Tn + (n-1)(Tp + T)}{Tn - (n-1)pn} \right\rceil \tag{2.34}$$

iterations.

Proof Without loss of generality, Owner #1 is considered a semi-trusted data owner. (A semi-trusted data owner completes and shares his/her computations faithfully, but tries to learn additional information while or after the algorithm runs.) For each iteration k, this data owner receives the intermediate matrix $\overline{\mathbf{H}}^k - \underbrace{\overline{\mathbf{ZB}}^k}_{=\frac{1}{n}\sum_{i=1}^{n}\mathbf{Z}_{A_i}\mathbf{B}^k_{A_i}} - \mathbf{U}^k \in \mathbb{R}^{T\times n}$, which provides Tn values. However, Owner #1 does not know

$$\underbrace{-\mathbf{U}^k + \overline{\mathbf{H}}^k}_{\in\mathbb{R}^{T\times n}}, \ \underbrace{\mathbf{B}^k_{A_2}, \ldots, \mathbf{B}^k_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{p\times n}}, \ \underbrace{\mathbf{Z}_{A_2}, \ldots, \mathbf{Z}_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{T\times p}}, \ \underbrace{\mathbf{Y}_{A_2}, \ldots, \mathbf{Y}_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{T\times 1}}, $$

which corresponds to $Tn+(n-1)pn+(n-1)Tp+(n-1)T$ values. Nevertheless, since all the data owners know that $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$ are defined by the expressions in (II.53b) and (II.53c), it is possible to perform some simplifications in which $\mathbf{U}^k$ and $\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k$ becomes (2.35) and (2.36), respectively:

$$\begin{aligned}\mathbf{U}^k \overset{\text{(II.53c)}}{=} & \mathbf{U}^{k-1} + \overline{\mathbf{ZB}}^k - \overline{\mathbf{H}}^k = \mathbf{U}^{k-1} + \overline{\mathbf{ZB}}^k - \underbrace{\frac{1}{N+\rho}\Big(\mathbf{Y} + \rho\overline{\mathbf{ZB}}^k + \rho\mathbf{U}^{k-1}\Big)}_{=\overline{\mathbf{H}}^k\text{, according to (II.53b)}} \\ = & \Big[1 - \frac{\rho}{N+\rho}\Big]\mathbf{U}^{k-1} + \Big[1 - \frac{\rho}{N+\rho}\Big]\overline{\mathbf{ZB}}^k - \frac{1}{N+\rho}\mathbf{Y}\ ,\end{aligned} \tag{2.35}$$

$$\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k = \underbrace{\frac{1}{N+\rho}\Big(\mathbf{Y} + \rho\overline{\mathbf{ZB}}^k + \rho\mathbf{U}^{k-1}\Big)}_{=\overline{\mathbf{H}}^k\text{, according to (II.53b)}} - \overline{\mathbf{ZB}}^k - \mathbf{U}^k. \tag{2.36}$$

Therefore, the iterative process of finding the competitors' data proceeds as follows:

1. *Initialization:* The central node generates $\mathbf{U}^0 \in \mathbb{R}^{T\times n}$, and the ith data owner generates $\mathbf{B}^1_{A_i} \in \mathbb{R}^{p\times n}$, $i \in \{1, ..., n\}$.

2. *Iteration #1:* The central node receives $\mathbf{Z}_{A_i}\mathbf{B}^1_{A_i}$ and computes $\mathbf{U}^1$, returning $\overline{\mathbf{H}}^1 - \overline{\mathbf{ZB}}^1 - \mathbf{U}^1 \in \mathbb{R}^{T\times n}$ which is returned for all n data owners. At this point, Owner #1 receives Tn values and does not know

$$\underbrace{\mathbf{U}^0}_{\in\mathbb{R}^{T\times n}}\ ,\quad \underbrace{\mathbf{B}^1_{A_2}, ..., \mathbf{B}^1_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{p\times n}}\quad ,\quad \underbrace{\mathbf{Z}_{A_2}, ..., \mathbf{Z}_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{T\times p}}\quad ,$$

and $n-1$ columns of $\mathbf{Y} \in \mathbb{R}^{T\times n}$, corresponding to $Tn + (n-1)[pn + Tp + T]$ values.

3. *Iteration #2:* The central node receives $\mathbf{Z}_{A_i}\mathbf{B}^2_{A_i}$ and computes $\mathbf{U}^2$, returning $\overline{\mathbf{H}}^2 - \overline{\mathbf{ZB}}^2 - \mathbf{U}^2$ for the n data owners. At this point, only new estimations for the matrices $\mathbf{B}_{A_2}, ..., \mathbf{B}_{A_n}$ were introduced in the system, which means more $(n-1)pn$ values must be estimated.

As a result, after k iterations, Owner #1 has received $\mathbf{Z}_{A_i}\mathbf{B}^1_{A_i}, \ldots, \mathbf{Z}_{A_i}\mathbf{B}^k_{A_i} \in \mathbb{R}^{T\times n}$ corresponding to Tnk values and needs to estimate

$$\underbrace{\mathbf{U}^0}_{\in\mathbb{R}^{T\times n}}, \underbrace{\mathbf{B}^1_{A_2}, ..., \mathbf{B}^1_{A_n}, \mathbf{B}^2_{A_2}, ..., \mathbf{B}^2_{A_n}, \ldots, \mathbf{B}^k_{A_2}, ..., \mathbf{B}^k_{A_n}}_{(n-1)k \text{ matrices } \in \mathbb{R}^{p\times n}},\quad \underbrace{\mathbf{Z}_{A_2}, ..., \mathbf{Z}_{A_n}}_{n-1 \text{ matrices } \in \mathbb{R}^{T\times p}}\quad ,$$

and $n-1$ columns of $\mathbf{Y} \in \mathbb{R}^{T\times n}$, corresponding to $Tn+(n-1)[kpn+Tp+T]$. Then, the solution for the inequality

$$Tnk \geq Tn + (n-1)[kpn + Tp + T], \tag{2.37}$$

suggests that a confidentiality breach may occur after

$$k = \left\lceil \frac{Tn + (n-1)(Tp + T)}{Tn - (n-1)pn} \right\rceil \tag{2.38}$$

iterations. □

Figure 2.9 illustrates the k value for different combinations of T , n, and p. In general, the greater the number of records T , the smaller the number of iterations necessary for a confidentiality breach. This is because more information is shared during each iteration of the ADMM algorithm. By contrast, the number of iterations before a possible confidentiality breach increases with the number of data owners (n). The same is true for the number of lags (p).

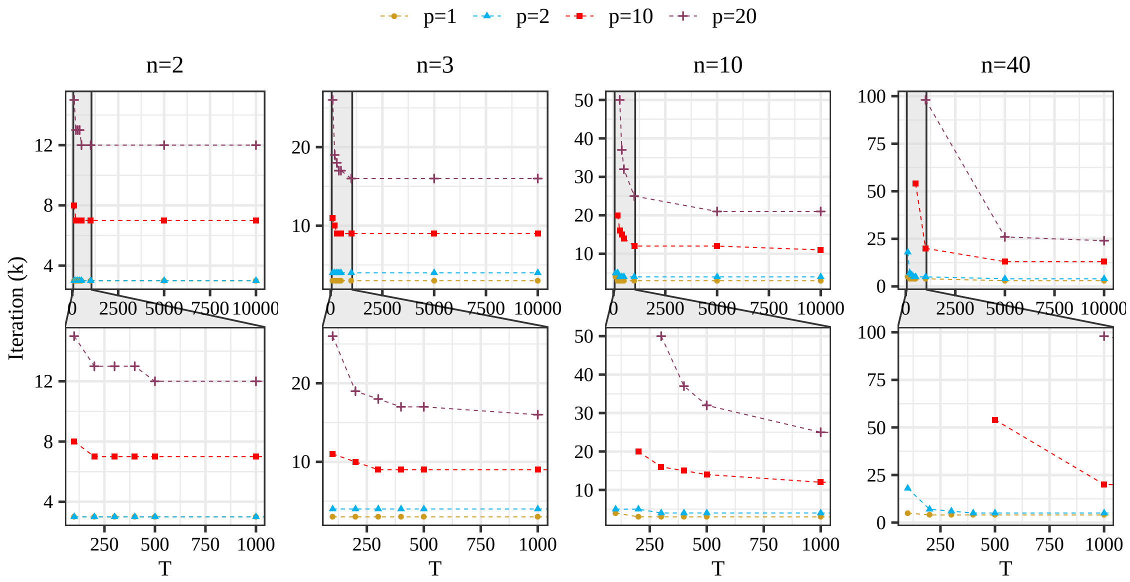


Figure 2.9: Number of iterations until a possible confidentiality breach, considering the centralized ADMM-based algorithm in [156].

2.3.4 ADMM Method and Noise Mechanisms

The target matrix $\mathbf{Y} = [\mathbf{Y}_{A_1}, \dots, \mathbf{Y}_{A_n}]$ corresponds to the sum of private matrices $\mathbf{I}_{\mathbf{Y}_{A_i}} \in \mathbb{R}^{T \times n}$. That is,

$$\underbrace{\begin{bmatrix} y_{1,t} & y_{2,t} & \dots & y_{n,t} \\ y_{1,t+1} & y_{2,t+1} & \dots & y_{n,t+1} \\ y_{1,t+2} & y_{2,t+2} & \dots & y_{n,t+2} \\ \vdots & \ddots & \vdots \\ y_{1,t+h} & y_{2,t+h} & \dots & y_{n,t+h} \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} y_{1,t} & 0 & \dots & 0 \\ y_{1,t+1} & 0 & \dots & 0 \\ y_{1,t+2} & 0 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ y_{1,t+h} & 0 & \dots & 0 \end{bmatrix}}_{\mathbf{I}_{Y_{A_1}}} + \underbrace{\begin{bmatrix} 0 & y_{2,t} & \dots & 0 \\ 0 & y_{1,t+1} & \dots & 0 \\ 0 & y_{1,t+2} & \dots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & y_{1,t+h} & \dots & 0 \end{bmatrix}}_{\mathbf{I}_{Y_{A_2}}} + \dots + \underbrace{\begin{bmatrix} 0 & 0 & \dots & y_{n,t} \\ 0 & 0 & \dots & y_{n,t+1} \\ 0 & 0 & \dots & y_{n,t+2} \\ \vdots & \ddots & \vdots \\ 0 & 0 & \dots & y_{n,t+h} \end{bmatrix}}_{\mathbf{I}_{Y_{A_n}}} , \tag{2.39}$$

where $[\mathbf{I}_{\mathbf{Y}_{A_i}}]_{i,j}$=$[\mathbf{Y}]_{i,j}$ in cases where the entry (i, j) of $\mathbf{Y}$ is from ith data owner and $[\mathbf{I}_{\mathbf{Y}_{A_i}}]_{i,j}$=0 otherwise.

Since the LASSO-VAR ADMM formulation is provided by (II.53), at iteration k, the data owners receive the intermediate matrix $\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k$ and then update their local solution through (II.53a). The combination of (2.35) with (2.39) can be used to rewrite $\mathbf{U}^k$ as

$$\mathbf{U}^k = \Big[1 - \frac{\rho}{N+\rho}\Big]\mathbf{U}^{k-1} + \sum_{i=1}^{n} \underbrace{\Big[1 - \frac{\rho}{N+\rho}\Big]\frac{1}{n}\mathbf{Z}_{A_i}\mathbf{B}^k_{A_i} - \frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}}}_{\text{information from owner } i} , \tag{2.40}$$

and, similarly, $\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k$ can be rewritten as

$$\begin{aligned} \overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k &= \frac{1}{N+\rho}\mathbf{Y} + \Big[\frac{\rho}{N+\rho} - 1\Big]\overline{\mathbf{ZB}}^k + \frac{\rho}{N+\rho}\mathbf{U}^{k-1} - \mathbf{U}^k \\ &= \sum_{i=1}^{n} \underbrace{\Big(\frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}} + \Big[\frac{\rho}{N+\rho} - 1\Big]\frac{1}{n}\mathbf{Z}_{A_i}\mathbf{B}^k_{A_i}\Big)}_{\text{information from owner } i} + \frac{\rho}{N+\rho}\mathbf{U}^{k-1} - \mathbf{U}^k , \end{aligned} \tag{2.41}$$

where

$$\mathbf{Y} = \sum_{i=1}^{n} \mathbf{I}_{\mathbf{Y}_{A_i}}, \tag{2.42}$$

$$\overline{\mathbf{ZB}}^{k+1} = \sum_{i=1}^{n} \frac{\rho}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k+1}. \tag{2.43}$$

By analyzing (2.40) and (2.41), it is possible to verify that data owner i only needs to share

$$\frac{1}{N+\rho} \mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right] \frac{1}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k}, \tag{2.44}$$

for the computation of $\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k$.

Let $\mathbf{W}_{1,A_i} \in \mathbb{R}^{T\times n}$, $\mathbf{W}_{2,A_i} \in \mathbb{R}^{T\times p}$, $\mathbf{W}_{3,A_i} \in \mathbb{R}^{p\times n}$, $\mathbf{W}_{4,A_i} \in \mathbb{R}^{T\times n}$, represent noise matrices generated according to the differential privacy framework. The noise mechanism could be introduced by

(i) adding noise to the data itself, i.e., replacing $\mathbf{I}_{\mathbf{Y}_{A_i}}$ and $\mathbf{Z}_{A_i}$ by

$$\mathbf{I}_{\mathbf{Y}_{A_i}} + \mathbf{W}_{1,A_i} \text{ and } \mathbf{Z}_{A_i} + \mathbf{W}_{2,A_i}\ , \tag{2.45}$$

(ii) adding noise to the estimated coefficients, i.e., replacing $\mathbf{B}_{A_i}^k$ by

$$\mathbf{B}_{A_i}^k + \mathbf{W}_{3,A_i}\ , \tag{2.46}$$

(iii) adding noise to the intermediate matrix (2.44),

$$\frac{1}{N+\rho} \mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right] \frac{1}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k} + \mathbf{W}_{4,A_i}. \tag{2.47}$$

The addition of noise to the data itself (2.45) was empirically analyzed in Section 2.3.1. As we showed, confidentiality comes at the cost of deteriorating model accuracy. The question is whether adding noise to the coefficients or intermediate matrix can ensure that data are not recovered after a number of iterations.

Proposition 4 *Consider noise addition in an ADMM-based framework by*

(i) adding noise to the coefficients, as described in (2.46)*;*

(ii) adding noise to the exchanged intermediate matrix, as described in (2.47)*.*

In both cases, a semi-trusted data owner can recover the data after

$$k = \left\lceil \frac{Tn + (n-1)(Tp+T)}{Tn - (n-1)pn} \right\rceil \tag{2.48}$$

iterations.

Proof These statements are promptly deduced from the Proof presented for Proposition 3. Without loss of generality, Owner #1 is considered the semi-trusted data owner.

(i) Owner #1 can estimate $\mathbf{B}_{A_i}$, without distinguishing between $\mathbf{B}_{A_i}$ and $\mathbf{W}_{3,A_i}$ in (2.46), by recovering $\mathbf{I}_{\mathbf{Y}_{A_i}}$ and $\mathbf{Z}_{A_i}$. Let $\mathbf{B}'_{A_i} = \mathbf{B}_{A_i} + \mathbf{W}_{3,A_i}$ and $\overline{\mathbf{H}'}^k$, $\mathbf{U}'^k$ be the matrices $\overline{\mathbf{H}}^k$, $\mathbf{U}^k$ replacing $\mathbf{B}_{A_i}$ by $\mathbf{B}'_{A_i}$. Then, at iteration k Owner #1 receives $\overline{\mathbf{H}'}^k - \overline{\mathbf{ZB}'}^k - \mathbf{U}'^k \in \mathbb{R}^{T\times n}$ (Tn values) and does not know

$$\underbrace{\overline{\mathbf{H}'}^k - \mathbf{U}'^k}_{\in\mathbb{R}^{T\times n}}, \underbrace{\mathbf{B}'^k_{A_2},\dots,\mathbf{B}'^k_{A_n}}_{n-1 \text{ matrices } \in\mathbb{R}^{p\times n}}, \underbrace{\mathbf{Z}_{A_2},\dots,\mathbf{Z}_{A_n}}_{n-1 \text{ matrices } \in\mathbb{R}^{T\times p}}, \underbrace{\mathbf{Y}_{A_2},\dots,\mathbf{Y}_{A_n}}_{n-1 \text{ matrices } \in\mathbb{R}^{T\times 1}},$$

which corresponds to $Tn+(n-1)pn+(n-1)Tp+(n-1)T$ values. As in Proposition 3, this means that, after k iterations, Owner #1 has received Tnk values and needs to estimate

$$\underbrace{\mathbf{U}'^0}_{\in\mathbb{R}^{T\times n}}, \underbrace{\mathbf{B}'^1_{A_2},...,\mathbf{B}'^1_{A_n},\mathbf{B}'^2_{A_2},...,\mathbf{B}'^2_{A_n},\dots,\mathbf{B}'^k_{A_2},...,\mathbf{B}'^k_{A_n}}_{(n-1)k \text{ matrices } \in\mathbb{R}^{p\times n}}, \underbrace{\mathbf{Z}_{A_2},...,\mathbf{Z}_{A_n}}_{n-1 \text{ matrices } \in\mathbb{R}^{T\times p}},$$

and $n{-}1$ columns of $\mathbf{Y}\in\mathbb{R}^{T\times n}$, corresponding to $Tn{+}(n{-}1)[kpn{+}Tp{+}T]$. Then, the solution for the inequality $Tnk \geq Tn + (n-1)[kpn + Tp + T]$ suggests that a confidentiality breach may occur after

$$k = \left\lceil \frac{Tn + (n-1)(Tp+T)}{Tn - (n-1)pn} \right\rceil$$

iterations.

(ii) Since Owner #1 can estimate $\mathbf{B}_{A_i}$ by recovering data, adding noise to the intermediate matrix reduces to the case of adding noise to the coefficients, in (i), because Owner #1 can rewrite (2.47) as

$$\frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho}-1\right]\frac{1}{n}\mathbf{Z}_{A_i}\underbrace{\left[\mathbf{B}^k_{A_i} + \left[\frac{\rho}{N+\rho}-1\right]^{-1}\mathbf{Z}^{-1}_{A_i}\mathbf{W}_{4,A_i}\right]}_{=\mathbf{B}'_{A_i}}. \tag{2.49}$$

□

2.4 Discussion

Table 2.1 summarizes the methods from the literature. These privacy-preserving algorithms ought to be carefully constructed, and two key components should be considered: (i) how data are distributed between data owners, and (ii) the statistical model used. Decomposition-based methods are very sensitive to data partitioning, while data transformation and cryptography-based methods are very sensitive to the problem structure. Differential privacy methods are notable exceptions, as they simply add random noise, from specific probability distributions, directly to the data. This property makes these methods appealing, but differential privacy usually involves a trade-off between accuracy and privacy.

Cryptography-based methods are usually more robust to confidentiality breaches, but they have some disadvantages: (i) some of them require a third-party to generate keys, as well as external entities to perform the computations in the encrypted domain; and (ii) there are challenges to the scalability and implementation efficiency, mostly due to the high computational complexity and overhead of existing homomorphic encryption

Table 2.1: Summary of state-of-the-art privacy-preserving approaches.

		Split by features	**Split by records**
Data Transformation		[134]	[131], [132], [135]
Secure Multi-party Computation	Linear Algebra	[117], [118], [136], [137]*, [138]	[136], [148]
	Homomorphic-cryptography	[114], [139], [142], [160]	[114], [139], [140], [141], [143], [160]
Decomposition-based Methods	Pure	[123], [158]	[119], [120], [170], [152]
	Linear Algebra	[161], [163]	[153], [154], [155], [166]
	Homomorphic-cryptography	[114], [144]*, [145]*, [146]*, [122], [171], [157], [164]	[114], [156], [122], [171]

* secure data aggregation.

schemes [147, 149, 150]. Regarding some protocols, such as secure multiparty computation through homomorphic cryptography, communication complexity grows exponentially with the number of records [172].

Data transformation methods do not affect the computational time for training the model, since data owners transform their data before the model fitting process. The same is true of decomposition-based methods, in which data are split by data owners. Secure multi-party protocols have the disadvantage of transforming the information while fitting the statistical model, which implies a higher computational cost.

As mentioned above, the main challenge to the application of existing privacy-preserving algorithms in the VAR model is the fact that $\mathbf{Y}$ and $\mathbf{Z}$ share a high percentage of values, not only during the fitting of the statistical model but also when using it to perform forecasts. A confidentiality breach can occur during the forecasting process if, after the model is estimated, the algorithm to maintain privacy provides the coefficient matrix $\mathbf{B}$ for all data owners. When using the estimated model to perform forecasts, we assume that each ith data owner sends its own contribution for time series forecasting to every other j-th data owner:

1. In the LASSO-VAR models with one lag, since ith data owner sends $y_{i,t}[\mathbf{B}^{(1)}]_{i,j}$ for the jth data owner, the value $y_{i,t}$ may be directly recovered when the coefficient $[\mathbf{B}^{(1)}]_{i,j}$ is known by all data owners, being $[\mathbf{B}^{(1)}]_{i,j}$ the coefficient associated with lag 1 of time series i, to estimate j.

2. In the LASSO-VAR models with p consecutive lags, the forecasting a new timestamp only requires the introduction of one new value in the covariate matrix of the ith data owner. In other words, after h timestamps, the jth data owner receives the h values. However, there are $h + p$ values that the data owner does not know about. This may represent a confidentiality breach, since a semi-trusted data owner can assume different possibilities for the initial p values and then generate possible trajectories.

3. In the LASSO-VAR models with p non-consecutive lags, $p_1, \dots, p_p$, after $p_p - p_{p-1}$ timestamps, only one new value is introduced in the covariate matrix, meaning that the model is also subject to a confidentiality breach.

Therefore, and considering the issue of data naturally split by features, it would be more advantageous to apply decomposition-based methods, since the time required for model fitting is unaffected by data transformations and data owners only have access to their own coefficients. However, with state-of-the-art approaches, it is difficult to guarantee that these techniques can indeed offer a robust solution to data privacy when addressing data split by features.

Finally, we offer a remark on specific business applications of VAR, where data owners know some exact past values of competitors. For example, consider a VAR model with lags $\Delta t = 1$, 2 and 24, which predicts the production of solar plants. When forecasting the first sunlight hour of a day, all data owners will know that the previous lags 1 and 2 have zero production (no sunlight). Irrespective of whether the coefficients are shared, a confidentiality breach may occur. In these special cases, the estimated coefficients cannot be used for a long time horizon, and online learning may represent an efficient alternative.

The privacy issues analyzed in this chapter are not restricted to the VAR model, nor to point forecasting tasks. Probabilistic forecasts, using data from different data owners (or geographical locations), can be generated with splines quantile regression [78], component-wise gradient boosting [173], a VAR that estimates the location parameter (mean) of data transformed by a logit-normal distribution [174], linear quantile regression with LASSO regularization [175], and others. These are some examples of collaborative probabilistic forecasting methods. However, none of them considers the confidentiality of data. Moreover, the method proposed by [174] can be influenced by the confidentiality breaches discussed throughout this chapter, since the VAR model is directly used to estimate the mean of transformed data from different data owners. By contrast, when performing non-parametric models such as quantile regression, each quantile is estimated by solving an independent optimization problem, which means that the risk of a confidentiality breach increases with the number of quantiles being estimated. (Note that quantile regression-based models may be solved through the ADMM [156]. However, as discussed in Section 2.2.3, a semi-trusted agent can collect enough information to infer the confidential data. The quantile regression method may also be estimated by applying linear programming algorithms [175], which may be solved through homomorphic encryption, despite being computationally demanding for high-dimensional multivariate time series.

2.5 Concluding Remarks

This chapter presented a critical overview of techniques used to handle privacy issues in collaborative forecasting methods. In addition, we analyzed their application to the VAR model. The techniques were divided into three groups of approaches: data transformation, secure multiparty computation, and decomposition of the optimization problem into subproblems.

For each group, several points can be concluded. Starting with data transformation techniques, two remarks were made. The first concerns the addition of random noise to the data. While the algorithm is simple to apply, this technique demands a trade-off between privacy and the correct estimation of the model's parameters [114]. In our experiments, there was clear model degradation even though the data continued to provide relevant information (Section 2.3.1). The second relates to the multiplication by an undisclosed random matrix. Ideally, and in what concerns data where different data owners observe different variables, this secret matrix would post-multiply data, thus enabling each data owner to generate a few lines of this matrix. However, as demonstrated in (2.8) in Section 2.2.1, this transformation does not preserve the estimated coefficients, and the reconstruction of the original model may require sharing the matrices used to encrypt the data, thus exposing the original data.

The second group of techniques, *secure multi-party computation*, introduce privacy to the intermediate computations by defining the protocols for addition and multiplication of the private datasets. Confidentiality breaches are avoided by using either linear algebra or homomorphic encryption methods. For independent records, data confidentiality is guaranteed for (ridge) linear regression through linear algebra-based protocols; not only do records need to be independent, but some also require that the target variable is known by all data owners. These assumptions might prevent their application when covariates and target matrices share a large proportion of values–in the case of the VAR model, for instance. This means that data shared between agents might be enough for competitors to be able to reconstruct the data. Homomorphic cryptography methods can result in computationally demanding techniques, since each dataset value must be encrypted. The protocols we discussed preserve privacy while using (ridge) linear regression, provided that there are two entities that correctly perform the protocol without agent collusion. These entities are an external server (e.g., a cloud server) and an entity that generates the encryption keys. In some approaches, all data owners know the coefficient matrix $\mathbf{B}$ after model estimation. This is a disadvantage when applying models in which covariates include the lags of the target variable, because confidentiality breaches can occur during the forecasting phase.

Finally, *decomposition of the optimization problem* into sub-problems (which can be solved in parallel) have all the desired properties of a collaborative forecasting problem, since data owners only estimate their own coefficients. A common assumption of such methods is that the objective function is decomposable. However, these approaches consist of iterative processes that require sharing intermediate results for the next update, meaning that each new iteration conveys more information about the secret datasets to the data owners, with the possibility of breaching data confidentiality.

A method will be proposed in the next chapter (Chapter 3) to solve the privacy limitations of the LASSO-VAR model here identified. Furthermore, even if privacy is ensured, a data owner may be unwilling to share their data, therefore an algorithmic solution for data monetization will be proposed in Chapter 4.

3

Privacy-preserving Distributed Learning for RES Forecasting

Data exchange between multiple renewable energy power plant owners can lead to an improvement in forecast skill thanks to the spatio-temporal dependencies in time series data. However, owing to business competitive factors, these different owners might be unwilling to share their data. In order to tackle this privacy issue, this chapter formulates a novel privacy-preserving framework that combines data transformation techniques with the alternating direction method of multipliers. This approach allows not only to estimate the model in a distributed fashion but also to protect data privacy, coefficients and covariance matrix. Besides, asynchronous communication between peers is addressed in the model fitting, and two different collaborative schemes are considered: centralized and peer-to-peer. The results for solar and wind energy datasets show that the proposed method is robust to privacy breaches and communication failures, and delivers a forecast skill comparable to a model without privacy protection.

3.1 Introduction

The forecast skill of Renewable Energy Sources (RES) has improved over the past two decades through R&D activities across the complete model chain, i.e., from Numerical Weather Prediction (NWP) to statistical learning methods that convert weather variables into power forecasts [29]. The need to bring forecast skill to significantly higher levels is widely recognized in the majority of roadmaps that deal with high RES integration scenarios for the next decades. This is expected not only to facilitate RES integration in the system operation and electricity markets but also to reduce the need for flexibility and associated investment costs on remedies that aim to hedge RES variability and uncertainty like storage, demand response, and others.

In this context, intraday and hour-ahead electricity markets are becoming increasingly important to handle RES uncertainty and thus accurate hours-ahead forecasts are essential. Recent findings showed that feature engineering, combined with statistical models, can extract relevant information from spatially distributed weather and RES power time series and improve hours-ahead forecast skill [29]. Indeed, for very short-term lead times (from 15 minutes to 6 hours ahead), the Vector AutoRegressive (VAR) model, when compared to univariate time series models, has shown competitive results for wind [78] and solar [84] power forecasting. Alternative models are also being applied to this problem, most notably deep learning techniques such as convolutional neural networks or long short-term memory networks [88]. While there may always be a debate about the interest and relevance of statistical modeling vs. machine learning approaches, VAR models have the advantages of flexibility, interpretability, acceptability by practitioners, as well as robustness in terms of forecast skill.

Five important challenges for RES forecasting have been identified when using VAR: (a) sparse structure of the coefficients' matrix [176], (b) uncertainty forecasting [174], (c) distributed learning [81], (d) online learning [177], and (e) data privacy.

Data privacy is a critical barrier to the application of collaborative forecasting models. Although multivariate time series models offer forecast skill improvement, the lack of privacy-preserving mechanisms makes data owners unwilling to cooperate. For instance, in the VAR model, the covariates are the lags of the target variable of each RES site, which means that agents (or data owners) cannot provide covariates without also providing their power measurements.

To the best of our knowledge, only three works have proposed privacy-preserving approaches for RES forecasting. Zhang and Wang described a privacy-preserving approach for wind power forecasting with off-site time series, which combined ridge linear quantile regression with Alternating Direction Method of Multipliers (ADMM) [158]. However, privacy with ADMM is not always guaranteed since it requires intermediate calculations, allowing the most curious competitors to recover the data at the end of several iterations, as shown in Section 2.3.3. Moreover, the central node can also recover the original and private data. Sommer et al. [178] considered an encryption layer, which consists of multiplying the data by a random matrix. However, the focus of this work was not data privacy, but rather online learning, and the private data are revealed to the central agent who performs intermediary computations. Berdugo et al. described a method based on local and global analog-search (i.e., template matching) that uses solar power time series from neighboring sites [179]. However, agents only share reference time-stamps and normalized weights of the analogs identified by the neighbors, hence forecast error is only indirectly reduced. In this chapter, we also use ADMM as a central framework for distributed learning and forecasting, in view of its flexibility in terms of communication setup for all agents involved, the possibility to add a privacy-preserving layer, as well as the promising resulting forecast skill documented in the literature.

In the previous chapter, a literature analysis of privacy-preserving techniques for VAR has grouped these techniques as (a) *data transformation*, such as the generation of random matrices that pre- or post-multiply the data [180] or using principal component analysis with differential privacy [181], (b) *secure multi-party computation*, such as linear algebra protocols [182] or homomorphic encryption (encrypting the original data in a way that arithmetic operations in the public space do not compromise the encryption [183]), and (c) *decomposition-based methods* like the ADMM [184] or the distributed Newton-Raphson method [185]. The main conclusions were that *data transformation* requires a trade-off between privacy and accuracy, *secure multi-party computations* either result in computationally demanding techniques or do not fully preserve privacy in VAR models, and that *decomposition-based methods* rely on iterative processes and after a number of iterations, the agents have enough information to recover private data.

With our focus on privacy-preserving protocols for very short-term forecasting with the VAR model, the main research outcome from this chapter is a novel combination of data transformation and decomposition-based methods so that the VAR model is fitted in another feature space without decreasing the forecast skill (which contrasts with [179]). The main advantage of this combination is that the ADMM algorithm is not affected and therefore: (a) asynchronous communication between peers can be addressed while fitting the model; (b) a flexible privacy-preserving collaborative model can be implemented using two different schemes, centralized communication with a neutral node and peer-to-peer communication, and in a way that original data cannot be recovered by central node or peers (this represents a more robust approach compared to the ADMM implementation in [158, 178]).

The remaining of this chapter is organized as follows: Section 3.2 describes the distributed learning framework. Section 3.3 formulates a novel privacy-preserving Least Absolute Shrinkage and Selection Operator (LASSO)-VAR model. Then, two case studies with solar and wind energy data are considered in Section 3.4. Concluding remarks are provided in Section 3.5.

3.2 Distributed Learning Framework

This section discusses the distributed learning framework that enables different agents or data owners (e.g., RES power plant, market players, forecasting service providers) to exploit geographically distributed time series data (power and/or weather measurements, NWP, etc.) and improve forecast skill while keeping data private. In this context, data privacy can either refer to commercially sensitive data from grid-connected RES power plants or personal data (e.g., under European Union General Data Protection Regulation) from households with RES technology. Distributed learning (or collaborative forecasting) means that instead of sharing their data, the model fitting problem is solved in a distributed manner. Two collaborative schemes are possible: centralized communication with a central node (*central hub*) and peer-to-peer communication (*P2P*).

In the *central hub* model, the scope of the calculations performed by the agents is limited by their local data and the only information transmitted to the central node is statistics, e.g., average values or local data multiplied by locally estimated coefficients. The central node is responsible for combining these local estimators and, when considering iterative solvers like ADMM, coordinating the individual optimization processes to solve the main optimization problem. The central node can be either a transmission/distribution system operator (TSO/DSO) or a forecasting service provider. The TSO or DSO could operate a platform that promotes collaboration between competitive RES power plants in order to improve the forecasting accuracy and reduce system balancing costs. On the other hand, the forecasting service provider could host the central node and make available APIs and protocols for information (not data) exchange between different data owners, during model fitting, and receives a payment for this service.

In the P2P, the agents equally conduct a local computation of their estimators, but share their information with peers, meaning that each agent is itself agent and central node. While P2P tends to be more robust (i.e., lower points of failure), it is usually difficult to make it as efficient as the central hub model in terms of communication costs — when considering n agents, each agent communicates with the remaining $n-1$.

The P2P model is suitable for data owners that do not want to rely (or trust) upon a neutral agent. Potential business models could be: P2P forecasting between prosumers or RES power plants [186]; smart cities characterized by an increasing number of sensors and devices installed at houses, buildings, and transportation network [187].

In order to make these collaborative schemes feasible, the following fundamental principles must be respected: (a) ensure improvement in forecast skill, compared to a scenario without collaboration; (b) guarantee data privacy, i.e., agents and the central node cannot have access to (or recover) original data; (c) consider synchronous and asynchronous communication between agents. The formulation that will be described in Section 3.3 fully guarantees these three core principles.

3.3 Privacy-preserving Distributed LASSO-VAR

Using the notation in Section II.3.2, n data owners are assumed to use the same number of lags p to fit a LASSO-VAR model with a total number of T records. $\mathbf{Y}_{A_i} \in \mathbb{R}^{T\times 1}$ and $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$ respectively denote the target and covariate matrix for the ith data owner. In LASSO-VAR, the covariates and target matrices are obtained by joining the individual matrices column-wise, i.e., $\mathbf{Z} = [\mathbf{Z}_{A_1}, \dots, \mathbf{Z}_{A_n}]$ and $\mathbf{Y} = [\mathbf{Y}_{A_1}, \dots, \mathbf{Y}_{A_n}]$. For distributed computation, the coefficient matrix of data owner i is denoted by $\mathbf{B}_{A_i} \in \mathbb{R}^{p\times n}, i \in \{1, \dots, n\}$.

When applying the collaboration schemes discussed in Section 3.2 to the distributed ADMM LASSO-VAR formulation described in (II.53), at each iteration k each agent determines and transmits (II.53a), given by

$$\mathbf{B}_{A_i}^{k+1} = \underset{\mathbf{B}_{A_i}}{\arg\min} \left(\frac{\rho}{2} \| \mathbf{Z}_{A_i}\mathbf{B}_{A_i}^{k} + \overline{\mathbf{H}}^{k} - \overline{\mathbf{ZB}}^{k} - \mathbf{U}^{k} - \mathbf{Z}_{A_i}\mathbf{B}_{A_i} \|_2^2 + \lambda \|\mathbf{B}_{A_i}\|_1 \right)$$

and then it is up to the central agent or peers (depending on the adopted structure) to compute the quantities in (II.53b), i.e.,

$$\overline{\mathbf{H}}^{k+1} = \frac{1}{n+\rho} \left(\mathbf{Y} + \rho \overline{\mathbf{ZB}}^{k+1} + \rho \mathbf{U}^{k} \right)$$

and (II.53c), i.e.,

$$\mathbf{U}^{k+1} = \mathbf{U}^{k} + \overline{\mathbf{ZB}}^{k+1} - \overline{\mathbf{H}}^{k+1}.$$

As shown in the previous chapter, although there is no direct exchange of private data, the computation of (II.53b) and (II.53c) provides indirect information about these data, meaning that confidentiality breaches can occur after a number of iterations.

This section describes the novel privacy-preserving collaborative forecasting method, which combines multiplicative randomization of the data (Section 3.3.1) with the distributed ADMM for the generalized LASSO-VAR model (Section 3.3.2), which had been previously formulated in Section II.3.2. Communication issues (Section 3.3.5) are also addressed since they are common in distributed systems.

3.3.1 Data Transformation with Multiplicative Randomization

Multiplicative randomization of the data [188] consists of multiplying the data matrix $\mathbf{X} \in \mathbb{R}^{T\times ns}$ by full rank perturbation matrices. If the perturbation matrix $\mathbf{M} \in \mathbb{R}^{T\times T}$ pre-multiplies $\mathbf{X}$, i.e., $\mathbf{MX}$, the records are randomized. On the other hand, if perturbation matrix $\mathbf{Q} \in \mathbb{R}^{ns\times ns}$ post-multiplies $\mathbf{X}$, i.e., $\mathbf{XQ}$, then the features are randomized. The challenges related to such transformations are two-fold: (i) $\mathbf{M}$ and $\mathbf{Q}$ are algebraic encryption keys, and consequently should be fully unknown by agents, (ii) data transformations need to preserve the relationship between the original time series.

When $\mathbf{X}$ is divided by features, as is the case with matrices $\mathbf{Z}$ and $\mathbf{Y}$ when defining VAR models, $\mathbf{Q}$ can be constructed as a diagonal matrix – see (3.1), where matrices in diagonal, $\mathbf{Q}_{A_i} \in \mathbb{R}^{s\times s}$, are privately defined by agent $i \in \{1, \dots, n\}$. Then, agents post-multiply their data without sharing $\mathbf{Q}_{A_i}$, since

$$\underbrace{\left[\mathbf{X}_{A_1}, \dots, \mathbf{X}_{A_n}\right]}_{=\mathbf{X}} \underbrace{\begin{bmatrix} \mathbf{Q}_{A_1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{Q}_{A_n} \end{bmatrix}}_{=\mathbf{Q}} = \left[\mathbf{X}_{A_1}\mathbf{Q}_{A_1}, \dots, \mathbf{X}_{A_n}\mathbf{Q}_{A_n}\right]. \tag{3.1}$$

Unfortunately, the same reasoning is not possible when defining $\mathbf{M}$, because all elements of column j of $\mathbf{M}$ multiplies all elements of row j in $\mathbf{X}$ (containing data from every agent). Therefore, the challenge is to define a random matrix $\mathbf{M}$, unknown but at the same time built by all agents.

We propose to define $\mathbf{M}$ as

$$\mathbf{M} = \mathbf{M}_{A_1}\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}, \tag{3.2}$$

where $\mathbf{M}_{A_i} \in \mathbb{R}^{T\times T}$ is privately defined by agent i. This means that

$$\mathbf{M}\mathbf{X} = [\underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_1}}_{=\mathbf{M}\mathbf{X}_{A_1}}, \dots, \underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_n}}_{=\mathbf{M}\mathbf{X}_{A_n}}]. \tag{3.3}$$

Some linear algebra-based protocols exist for secure matricial product, but they were designed for matrices with independent observations and have proven to fail when applied to such matrices as $\mathbf{Z}$ and $\mathbf{Y}$ (see Section 2.3.2 for a proof). The calculation of $\mathbf{M}\mathbf{X}_{A_i}$ is described in Algorithm 1:

Algorithm 1 Data Encryption.

```
Input from ith agent: $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$ and $\mathbf{M}_{A_i} \in \mathbb{R}^{T\times T}$
Input from jth agent ($j \neq i$): $\mathbf{M}_{A_j} \in \mathbb{R}^{T\times T}$
Output: $\mathbf{M}\mathbf{X}_{A_i} = \mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_i}$
Initialization: Agent $i$ generates random invertible matrices $\mathbf{C}_{A_i} \in \mathbb{R}^{T\times(r-s)}$, $\mathbf{D}_{A_i} \in \mathbb{R}^{r\times r}$, and shares $\mathbf{W}_{A_i} \in \mathbb{R}^{T\times r}$ with the $n$-th agent,
    $\mathbf{W}_{A_i} = [\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i}.$    (3.4)
Agent $n$ receives $\mathbf{W}_{A_i}, \forall i$.
Agent $n$ shares $\mathbf{M}_{A_n}\mathbf{W}_{A_i}$ with the $(n-1)$-th agent.
for agent $j = n-1, \dots, 1$ do
    Agent $j$ receives $\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$, and
    if $j > 1$ then
        shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent $j-1$
    else
        shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent $i$
    end if
end for
Agent $i$ receives $\mathbf{M}\mathbf{W}_{A_i}$ from the 1-st agent and recovers $\mathbf{M}\mathbf{X}_{A_i}$,
    $[\mathbf{M}\mathbf{X}_{A_i}, \mathbf{M}\mathbf{C}_{A_i}] = \mathbf{M}\mathbf{W}_{A_i}\mathbf{D}_{A_i}^{-1}.$    (3.5)
```

The privacy of this protocol depends on r, which is chosen according to the number of unique values on $\mathbf{X}_{A_i}$. The optimal value for r is discussed in Proposition 5 of Appendix A.2.

3.3.2 Formulation of the Collaborative Forecasting Model

When applying the ADMM algorithm, the protocol presented in the previous section should be applied to transform matrices $\mathbf{Z}$ and $\mathbf{Y}$ in such a way that: (i) the estimated coefficients do not coincide with the originals, instead they are a secret transformation of them, (ii) agents are unable to recover the private data through the exchanged information, and (iii) cross-correlations cannot be obtained, i.e., agents are unable to recover $\mathbf{Z}^\top\mathbf{Z}$ nor $\mathbf{Y}^\top\mathbf{Y}$.

To fulfill these requirements, both covariate and target matrices are transformed through multiplicative noise. Both $\mathbf{M}$ and $\mathbf{Q}$ must be invertible, which is ensured if $\mathbf{M}_{A_i}$ and $\mathbf{Q}_{A_i}$ are invertible for $i \in \{1, \dots, n\}$.

Formulation

Let $\mathbf{ZQ}$ be the covariate matrix obtained through (3.1) and $\mathbf{Y}$ the target matrix. Covariate matrix $\mathbf{ZQ}$ is divided by features, and the optimization problem which allows recovering the solution in the original space, i.e.,

$$\arg\min_{\mathbf{B}} \Big(\frac{1}{2}\|\mathbf{Y} - \sum_i \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{B}_{A_i}\|_1\Big), \tag{3.6}$$

is

$$\arg\min_{\mathbf{B}^{\text{post}}} \Big(\frac{1}{2}\|\mathbf{Y} - \sum_i \mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}^{\text{post}}_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{Q}_{A_i}\mathbf{B}^{\text{post}}_{A_i}\|_1\Big). \tag{3.7}$$

After a little algebra, the relation between the ADMM solution for (3.6) and (3.7) is

$$\mathbf{B}^{\text{post}\,k+1}_{A_i} = \mathbf{Q}_{A_i}\mathbf{B}^{k+1}_{A_i}, \tag{3.8}$$

suggesting coefficients' privacy since the original $\mathbf{B}$ is no longer used. However, the limitations identified in the previous chapter for (3.6) are valid for (3.7). That is, a curious agent can obtain both $\mathbf{Y}$ and $\mathbf{ZQ}$, and because $\mathbf{Y}$ and $\mathbf{Z}$ share a large proportion of values, $\mathbf{Z}$ can also be recovered.

Taking covariate matrix $\mathbf{MZQ}$ and target $\mathbf{MY}$, the ADMM solution for the optimization problem

$$\arg\min_{\mathbf{B}'} \Big(\frac{1}{2}\|\mathbf{MY} - \sum_i \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_1\Big), \tag{3.9}$$

preserves the relation between the original time series if $\mathbf{M}$ is orthogonal, i.e., $\mathbf{MM}^\top = \mathbf{I}$. In this case, a competitor can only obtain $\mathbf{MY}$ without distinguishing between $\mathbf{M}$ and $\mathbf{Y}$. But the orthogonality of $\mathbf{M}$ ensures that $(\mathbf{MY})^\top\mathbf{MY} = \mathbf{Y}^\top\mathbf{Y}$, meaning that the covariance matrix is not protected.

Note that the orthogonality of $\mathbf{M}$ is necessary to ensure that, while computing $\mathbf{B}'_{A_i}$,

$$\begin{aligned} \mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^\top\Big[\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{MZQB}'}^{k} + \dots\Big] &= \\ \mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\Big[\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{ZQB}'}^{k} + \dots\Big]&. \end{aligned} \tag{3.10}$$

We remove the orthogonality condition on matrix $\mathbf{M}$ by using $\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^{-1}$ instead of $\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^\top$,

$$\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^{-1}\Big[\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{MZQB}'}^{k} + \dots\Big]. \tag{3.11}$$

Our proposal requires agents to compute $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}$, $\mathbf{MY}_{A_i}$ and $\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^{-1}$, where $\mathbf{M}$ is a random invertible matrix. Algorithm 2 summarizes our proposal for estimating a privacy-preserving LASSO-VAR model.

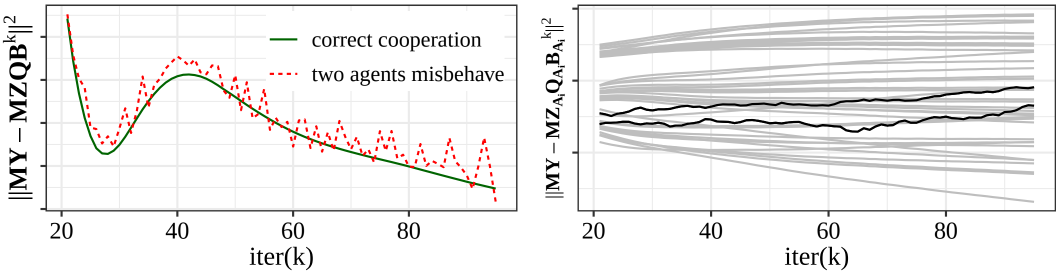


Figure 3.1: Error evolution (left: global error; right: error by agent with black lines representing the two agents who add random noise to $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^k_{A_i}$).

$\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^{-1}$ is obtained by adapting Algorithm 1. In this case, the value of r is more restrictive because we need to ensure that agent i does not obtain both $\mathbf{Y}^\top_{A_i}\mathbf{M}^{-1}$ and $\mathbf{MY}_{A_i}$. Otherwise, the covariance and cross-correlation matrices are again vulnerable. Let us assume that $\mathbf{Z}_{A_i}$ and $\mathbf{Q}_{A_i}$ represent u unique unknown values and $\mathbf{Y}_{A_i}$ has v unique unknown values that are not in $\mathbf{Z}_{A_i}$. Then, privacy is ensured by computing $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}$ and $\mathbf{Q}^\top_{A_i}\mathbf{Z}^\top_{A_i}\mathbf{M}^{-1}$ using the smaller integer r such that $\sqrt{Tp-u}<r<T/2\wedge r>p$, and then $\mathbf{MY}_{A_i}$ with $\sqrt{T-v}<r'<T-2r\wedge r'>1$ (see Proposition 6 in Appendix A.2 for determination of the optimal r). Appendix A.3 presents an analysis of the data privacy for scenarios without and with collusion between agents (data owners) during encrypted data exchange.

Finally, it is important to underline that Algorithm 2 can be applied to both *central hub model* and *P2P model* schemes without any modification – depending on who (central node or peers, respectively) receives $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^{k+1}_{A_i}$ and computes (3.13)–(3.15).

Malicious agents

The proposed approach assumes that agents should only trust themselves, requiring control mechanisms to detect when agents share wrong estimates of their coefficients, compromising the global model. Since $\mathbf{MY}$ and $\mathbf{MZQB'}^k$ can be known by agents without exposing private data, a malicious agent is detected through the analysis of the global error $\|\mathbf{MY}-\mathbf{MZQB'}^k\|_2^2$. That is, during the iterative process, this global error should smoothly converge, as depicted in Figure 3.1 (left plot), and the same is expected for the individual errors $\|\mathbf{MY}-\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^k_{A_i}\|_2^2, \forall i$.

In the example of Figure 3.1, two agents are assumed to add random noise to their coefficients. This results in the erratic curve for the global error shown in Figure 3.1. An analysis of individual errors, in Figure 3.1 (right plot), shows that all agents have smooth curves, except the two who shared distorted information.

3.3.3 Tuning of Hyperparameters

Since the ADMM solutions for (3.6) is related to the solution for (3.9), agents can tune hyperparameters (ρ and λ) by applying common techniques, such as cross-validation grid-search, Nelder-Mead optimization, Bayesian optimization, etc., to minimize the loss function in (3.9). This requires the definition of fitting and validation datasets and corresponding encryption by Algorithm 1, taking into account that, for each fitting and validation pair, the matrix $\mathbf{Q}_{A_i}$ needs to be the same, but all the others should be changed to keep data private.

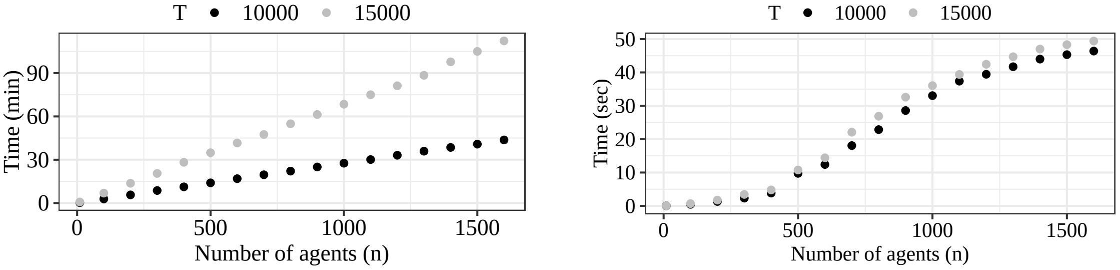
Algorithm 2 Synchronous Privacy-preserving LASSO-VAR.

```
Input: Randomized data $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}$, $\mathbf{MY}_{A_i}$, $\mathbf{Q}_{A_i}^\top \mathbf{Z}_{A_i}^\top \mathbf{M}^{-1}$
Output: Transformed coefficients $\mathbf{B'}_{A_i}{=}\mathbf{Q}_{A_i}\mathbf{B}_{A_i}, i{=}1,\dots,n$
Initialization: $\mathbf{B'}^0_{A_i}$, $\overline{\mathbf{H}}^0$, $\mathbf{U}^0 = \mathbf{0}, \rho \in \mathbb{R}^+, k = 0$
for agent $i = 1,\dots,n$ do
    $\mathbf{P}_{A_i} = \Big((\mathbf{Z}_{A_i}\mathbf{Q}_{A_i})^\top(\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}) + \rho\mathbf{Q}_{A_i}^\top\mathbf{Q}_{A_i}\Big)^{-1}$
end for
while stopping criteria not satisfied do
    for agent $i = 1,\dots,n$ do
        Initialization: $\widetilde{\mathbf{B}}^0_{A_i}$, $\overline{\widetilde{\mathbf{H}}}^0$, $\widetilde{\mathbf{U}}^0 = \mathbf{0}, j = 0$
        $\mathbf{K}_{A_i}{=}\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^k_{A_i}{+}\overline{\mathbf{H}}^k{-}\overline{\mathbf{MZQB'}}^k{-}\mathbf{U}^k$        (3.12)
        while stopping criteria not satisfied do
            $\widetilde{\mathbf{B}}^{j+1}_{A_i} = \mathbf{P}_{A_i}\Big(\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}\mathbf{K}_{A_i}{+}\rho(\overline{\widetilde{\mathbf{H}}}^j - \widetilde{\mathbf{U}}^j)\Big)$
            $\overline{\widetilde{\mathbf{H}}}^{j+1} = S_{\lambda/\rho}\Big(\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}^{j+1}_{A_i} + \widetilde{\mathbf{U}}^j\Big)$
            $\widetilde{\mathbf{U}}^{j+1} = \widetilde{\mathbf{U}}^j + \mathbf{Q}_{A_i}\widetilde{\mathbf{B}}^{j+1}_{A_i} - \overline{\widetilde{\mathbf{H}}}^{j+1}$
            $j = j + 1$
        end while
        $\mathbf{B'}^{k+1}_{A_i} = \widetilde{\mathbf{B}}^j_{A_i}$
    end for
    $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^k_{A_i}$ is shared with peers or central node, who computes (3.13)–(3.15),
    $\overline{\mathbf{MZQB'}}^k = \frac{1}{n}\sum_i \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^k_{A_i}$        (3.13)
    $\overline{\mathbf{H}}^{k+1} = \frac{1}{n+\rho}\Big(\mathbf{MY} + \overline{\mathbf{MZQB'}}^k + \rho\mathbf{U}^k\Big)$        (3.14)
    $\mathbf{U}^{k+1} = \mathbf{U}^k + \overline{\mathbf{MZQB'}}^{k+1} - \overline{\mathbf{H}}^{k+1}$        (3.15)
    $k = k + 1$
end while
```

Table 3.1: Floating-point operations in Algorithm 1.

Encrypted information	Operations
$(\mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i},\, \mathbf{Q}_{A_i}^{\top}\mathbf{Z}_{A_i}^{\top}\mathbf{M}^{-1})$	$\mathcal{O}(2Tr^2 + 2T^2nr + T(p^2 + r^2))$
$\mathbf{M}\mathbf{Y}_{A_i}$	$\mathcal{O}(Tr'^2 + T^2nr' + Tr'^2)$

* $r = \max(\lceil\sqrt{Tp - u}\rceil, p + 1)$ and $\sqrt{T-v}<r'<T-2r \wedge r' > 1$



(a) Data encryption by Algorithm 1.

(b) ADMM (Algorithm 2) iteration.

Figure 3.2: Mean running time as a function of the number of agents.

3.3.4 Computational Complexity

Typically, the computational complexity of an algorithm is estimated by the number of required floating-point operations (defined as one addition, subtraction, multiplication, or division of two floating-point numbers). When compared to the existing distributed ADMM literature applied to the LASSO-VAR model (e.g., [81, 169]), the computational complexity of the ADMM algorithm remains almost the same – only p^2n extra floating-point operations come from considering $\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}_{A_i}^{j+1}$ instead of $\widetilde{\mathbf{B}}_{A_i}^{j+1}$ in line 11 and 12 of Algorithm 2. However, there is also the computational cost related to the data transformation, performed before running the ADMM algorithm. Table 3.1 summarizes the floating-point operations necessary to encrypt the data matrices $\mathbf{Z}_{A_i}$ and $\mathbf{Y}_{A_i}$. The computational time for such data encryption is expected to increase linearly with the number of agents, and quadratically with the number of records.

A numerical analysis was performed by simulating data from VAR models with $n \in \{10, 100, 200, \ldots, 1600\}, T \in \{10000, 15000\}$ and $p = 5$. Figure 3.2 summarizes the mean running times using an i7-8750H @ 2.20GHz with 16 GB of RAM. To properly analyze the mean time per ADMM iteration, the computational times for the cycle between lines 6 to 15 of Algorithm 2 (coefficients' update) are measured assuming that the n agents update it in parallel. That said, considering for example a case with 10000 records and 500 agents, the data encryption takes around 15 minutes, and then the Algorithm 2 takes around 10 seconds per iteration.

3.3.5 Asynchronous Communication

When applying the proposed method, the matrices (3.13)–(3.15) combine the solutions of all data owners, meaning that the "slowest" agent dictates the duration of each iteration. Since communication delays and failures may occur due to computation or communication issues, the proposed algorithm should be robust to this scenario. Otherwise, the convergence to the optimal solution may require too much time. The proposed approach deals with these issues by considering the last information sent by agents, but different strategies are followed according to the adopted collaborative scheme.

Regarding the centralized scheme, let Ω_i^k be the set of iterations for which agent i communicated its information, until current iteration k. After receiving the local contributions, central agent computes $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$, in (3.14)–(3.15), by using $\sum_{i=1}^n \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}_{A_i}^{\max(\Omega_i^k)}$. Then, central agent returns $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$, informing agents about $\max(\Omega_i^k)$. To proceed, $\mathbf{B'}_{A_i}^{k+1}$ is updated by using $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}_{A_i}^{\max(\Omega_i^k)}$ in (3.12).

For the P2P approach, let Λ_i^k be the set of agents sharing information computed at iteration k, with agent i, i.e., $\Lambda_i^k{=}\{j$: agent j sent $\mathbf{MZ}_{A_j}\mathbf{Q}_{A_j}\mathbf{B'}_{A_j}^k$ to agent $i\}$. After computing and sharing $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}_{A_i}^k$, a second round of peer-to-peer communication is proposed, where agents share both Λ_i^k and $\sum_{j\in\Lambda_i^k}\mathbf{MZ}_{A_j}\mathbf{Q}_{A_j}\mathbf{B'}_{A_j}^k$. After this extra communication round, agent i can obtain missing information when $\Lambda_i^k \neq \Lambda_j^k$, $\forall i,j$.

3.4 Case Studies

To simulate the proposed method, communication failures are modeled through Bernoulli random variables F_{it}, with failure probability p_i, $F_{it}{\sim}Bern(p_i)$, for each agent $i{=}1,\dots,n$ at each communication time t. In this experimental setup, equal failure probabilities p_i are assumed for all agents and, since a specific p_i can generate various distinct failure sequences, 20 simulations were performed for each $p_i \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. The ADMM iterative process stops when all agents achieve

$$\frac{\|\mathbf{B}_{A_i}^{k+1}{-}\mathbf{B}_{A_i}^k\|_2}{\max(1,\min(\|\mathbf{B}_{A_i}^{k+1}\|_1,\|\mathbf{B}_{A_i}^k\|_1))}{\leq}\epsilon, \tag{3.16}$$

where ϵ is the tolerance parameter ($\epsilon{=}5{\times}10^{-4}$ is considered).

Regarding the benchmark models, the persistence and LASSO-autoregressive (LASSO-AR) models are implemented to assess the impact of collaboration over a model without collaboration. The analog method described in [179] is also implemented as a benchmark model because: (a) it is the only work in the RES forecasting literature that implements collaborative forecasting without data disclosure; (b) when the forecasting algorithm was designed, a trade-off between accuracy and privacy was necessary and the choice was privacy over accuracy. This method is now briefly described.

Firstly, agent i searches the k situations most similar to the current power production values $\mathbf{y}_{i,t-\ell+1},\dots,\mathbf{y}_{i,t}$. This similarity is measured through the Euclidean distance. Secondly, the k most similar situations (called analogs) are weighted according to the corresponding Euclidean distance. Agent i attributes the weight $w_{A_i}(a)$ to the analog a. The forecast for h steps ahead is obtained by applying the computed weights on the h values registered immediately after the k analogs. The collaboration between agents requires the exchange of the time indexes for the selected analogs and corresponding weights. Two analogs belong to the same global situation if they occur at the same or at close timestamps. Agent i scores the analog a, observed at timestamps t_a, by performing

$$s_{A_i}(a){=}\underbrace{(1{-}\alpha)w_{A_i}(a)}_{\text{own contribution}}+\underbrace{\frac{\alpha}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}w_{A_j}(j)I_\epsilon(t_a,t_j)}_{\text{others' weights for close timestamps}}, \tag{3.17}$$

where α is the weight given to neighbor information, j are the analogs from other agents, registered at timestamps t_j, and $I_\epsilon(t_a,t_j)$ is the indicator function taking value 1 if

$|t_j - t_a| \leq \epsilon$, with ϵ being the maximum time difference for two analogs to be considered part of the same global situation.

In the next subsections, two datasets are described, and results are analyzed. The model's accuracy is measured in terms of Normalized Root Mean Squared Error (NRMSE) calculated for agent i and lead-time h, with h=1, ..., 6, as

$$\text{NRMSE}_{i,h} = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{y}_{i,t+h} - y_{i,t+h})^2}}{\max(\{y_{i,t+h}\}_{t=1}^{T}) - \min(\{y_{i,t+h}\}_{t=1}^{T})}, \tag{3.18}$$

where $\hat{y}_{i,t+h}$ represents the forecast generated at time t.

3.4.1 Solar Power Data

Data Description

The proposed algorithm is also applied to forecast solar power up to 6 hours ahead. The data is publicly available in [24] and consists of hourly time series of solar power from 44 micro-generation units, located in a Portuguese city, and covers the period from February 1, 2011 to March 6, 2013. Since the VAR model requires the data to be stationary, the solar power is normalized through a clear sky model, which gives an estimate of the solar power in clear sky conditions at any given time [37]. This clear-sky model is fully data-driven and does not require any site-specific information (coordinates, rated power, etc.) since it estimates the clear-sky power time series exclusively from historical on-site power observations. Also, night-time hours are excluded by removing data for which the solar zenith angle is larger than 90. Based on previous work [84], a LASSO-VAR model to forecast $y_{i,t+h}$ at time t (using lags $t-1$, $t-2$ and $t+h-23$) is evaluated with a sliding-window of one month and the model's fitting period consists of 12 months, $h \leq 6$.

It is important to note that the LASSO-VAR model can be applied to both solar and wind power time series without any modification. Furthermore, when compared to wind power, solar power forecasting is more challenging because the lags 1 and 2 are zero for the first daylight hours, i.e., there are fewer unknown data, and this makes it easier to recover original data. In our protocol, this means more restrictive values for u and v, which are crucial when defining r and r', as stated in Proposition 6.

Results and Discussion

The hyperparameters ρ and λ were determined by cross-validation (12 folds) in the initial model's fitting dataset, by considering the values of $\rho, \lambda \in \{0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25\}$. Figure 3.3 illustrates the results in terms of NRMSE, for $h = 1$.

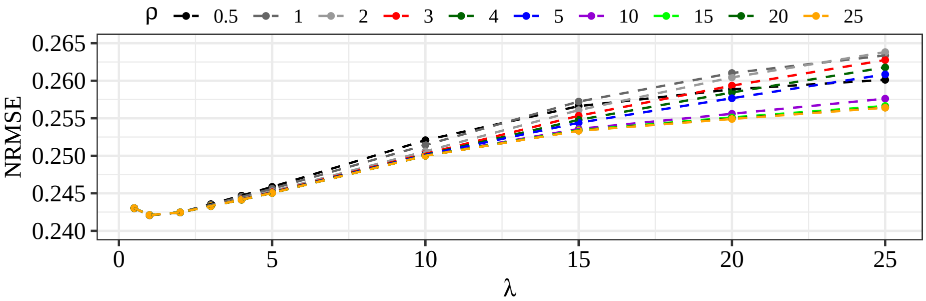


Figure 3.3: Impact of hyperparameters for $h = 1$, considering solar power dataset.

Table 3.2: NRMSE for synchronous models, considering solar power dataset.

	h=1	h=2	h=3	h=4	h=5	h=6
Persistence $(t)^*$	0.1605	0.2792	0.3768	0.4510	0.5020	0.5326
Persistence $(t+h\text{-}23)^*$	0.1728	0.1728	0.1728	0.1728	0.1728	0.1728
Analogs [179]†	0.1044	0.1305	0.1476	0.1578	0.1628	0.1649
LASSO-AR*	0.1010	0.1317	0.1429	0.1475	0.1492	0.1499
LASSO-VAR†	**0.0923**✓	**0.1236**✓	**0.1385**✓	**0.1451**✓	**0.1469**✓	**0.1484**✓

$*$ non-collaborative † collaborative

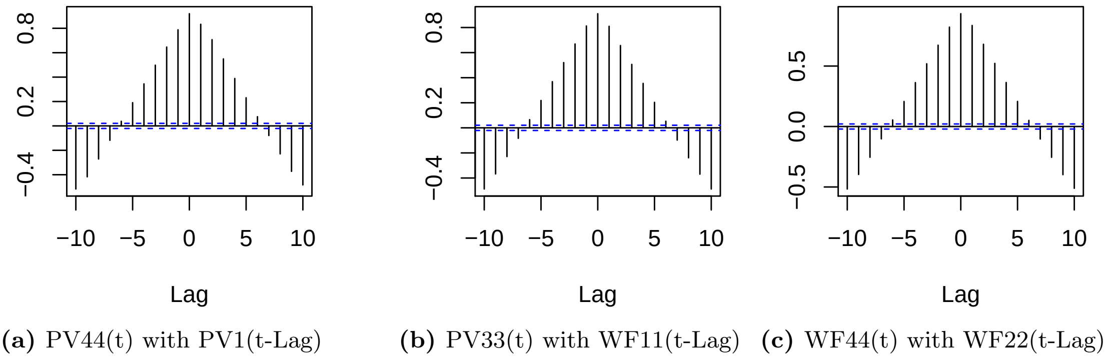
✓ statistically significant improvement against all others (DM test)

To access the quality of the proposed collaborative forecasting model, the synchronous LASSO-VAR is compared with benchmark models. Both *central hub* and *P2P model* have the same accuracy when considering synchronous communication. Table 3.2 presents the NRMSE for all agents, distinguishing between lead-times. In general, the smaller the forecasting horizon, the larger the NRMSE improvement, i.e.,

$$(\text{NRMSE}_{\text{Bench.}} - \text{NRMSE}_{\text{LASSO-VAR}}) \,/\, \text{NRMSE}_{\text{Bench.}} \cdot 100\%.$$

Besides, since the proposed LASSO-VAR and the LASSO-AR models have similar NRMSE for $h > 3$, the Diebold-Mariano test (described in Section II.2) is applied to test the superiority of the proposal, assuming a significance level of 5%. This test showed that the improvement is statistically significant for all horizons. It is important to note that the decrease in the improvement is explained by the cross-correlation between the geographically distributed time series data, as depicted in Figure 3.4. Since the dataset is from a small municipality in Portugal, it is expected that the highest improvement occurs for the first lead times (in particular the first one), where the cross-dependencies between time series have the most effect. However, this depends on the geographical layout and distance between power plants. For instance, in [81], the results for wind power plants show the highest improvement for the second lead time; in the test case of western Denmark [189], the highest cross-dependency between two groups of wind farms was observed for lag two.

Figure 3.5 depicts the relative improvement in terms of NRMSE for the 44 agents. According to the Diebold-Mariano test, the LASSO-VAR model outperforms benchmarks in all lead-times for at least 25 of the 44 agents. Indeed, some agents contribute to improving the competitors' forecast without having a benefit to their own forecasting accuracy. Then, even if privacy is ensured, such agents can be unwilling to collaborate, which motivates data monetization through data markets, as proposed in the next chapter.



(a) PV44(t) with PV1(t-Lag) **(b)** PV33(t) with WF11(t-Lag) **(c)** WF44(t) with WF22(t-Lag)

Figure 3.4: Cross-correlation plot (CCF) between two solar power plants.

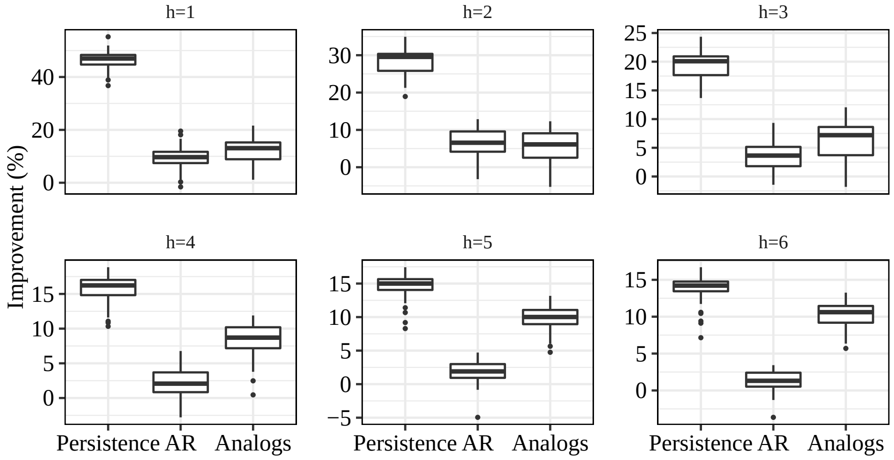


Figure 3.5: Relative NRMSE improvement (%) over the baseline models, considering solar power dataset.

Table 3.3 presents the mean running times and the number of iterations of both non-distributed and distributed approaches. The proposed schemes require larger execution times since they require estimating $\mathbf{B'}^{k}_{A_i}$ through a second ADMM cycle (Algorithm 2). However, the non-distributed LASSO-VAR requires more iterations to converge.

For asynchronous communication, equal failure probabilities p_i are assumed for all agents. Table 3.4 shows the mean NRMSE improvement for different failure probabilities $p_i, i \in \{1, \ldots, n\}$. In general, the greater the p_i the smaller the improvement. Despite the model's accuracy decreases slightly, the LASSO-VAR model continues to outperform the AR model for both collaborative schemes, which demonstrates high robustness to communication failures.

Figure 3.6 complements this analysis by showing the evolution of the loss while fitting the LASSO-VAR model, for $p_i \in \{0.5, 0.9\}$. For the centralized approach, the loss tends to stabilize around larger values. In general, the results are better for the P2P scheme since in the centralized approach if an agent fails the algorithm proceeds with no chance

Table 3.3: Mean running times (in sec) per iteration and number of iterations until convergence, considering solar power dataset.

Non distributed	Central LASSO-VAR		P2P LASSO-VAR	
LASSO-VAR	Enc. data	ADMM	Enc. data	ADMM
0.035 (≈ 410)	65.46	0.052 (≈ 300)	65.46	0.1181 (≈ 300)

Table 3.4: Mean relative NRMSE improvement (%) of the asynchronous ADMM LASSO-VAR over the LASSO-AR model, considering solar power dataset.

	h=1		h=2		h=3		h=4		h=5		h=6	
p_i	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P
0	8.41		6.05		2.95		1.52		1.39		0.93	
0.1	7.93	8.41	5.98	6.05	2.91	2.95	1.49	1.52	1.35	1.39	0.89	0.93
0.3	7.45	”	5.89	”	2.89	”	1.40	”	1.18	”	0.69	”
0.5	6.69	”	5.77	”	2.88	”	1.30	”	1.00	”	0.52	”
0.7	5.71	”	5.54	”	2.84	”	1.24	”	0.89	”	0.33	”
0.9	3.75	8.10	5.19	5.75	2.74	2.78	0.75	1.47	0.62	1.38	-0.82	0.88

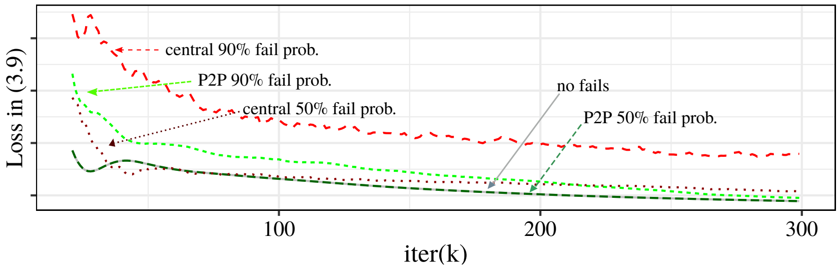


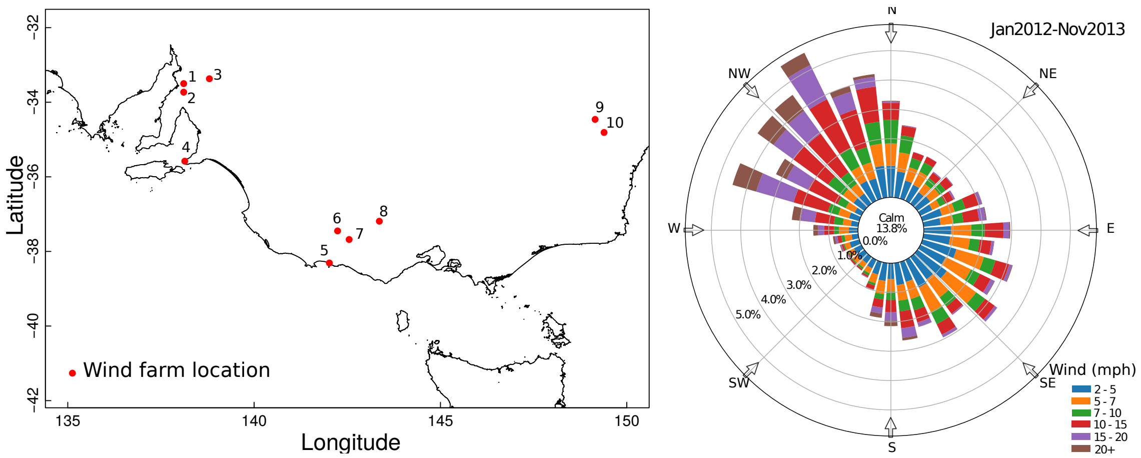
Figure 3.6: Loss while fitting LASSO-VAR model, considering solar power dataset.

of obtaining its information. In P2P, this agent may have communicated his contribution to some peers and the probability of losing information is smaller.

3.4.2 Wind Power data

Data Description

The proposed method is also experimented with a real wind power dataset, comprising hourly time series of wind power generation in 10 zones, corresponding to 10 wind farms in Australia [25], as depicted in Figure 3.7. This dataset was used in the Global Energy Forecasting Competition 2014 (GEFCom2014) and it is publicly available, covering the period from January 1, 2012 to November 30, 2013. The power generation for the next 6 hours is modeled through the LASSO-VAR model, which combines data from the 10 data owners and consider the most recent power measurements (lags 1h to 6h), based on the correlation analysis discussed in Section II.3.2. A sliding-window of one month is considered and the model's fitting period consists of 12 months.



(a) Wind farms' location.

(b) Wind rose (Canberra: close to WF9).

Figure 3.7: GEFCom2014 wind power dataset.

Results and Discussion

The hyperparameters ρ and λ were determined by cross-validation (12 folds) in the initial model's fitting dataset, by considering the values of $\rho, \lambda \in \{1, 2, 3, 4, 5, \ldots, 10\}$. Figure 3.8 illustrates the results in terms of NRMSE, when $h = 1$.

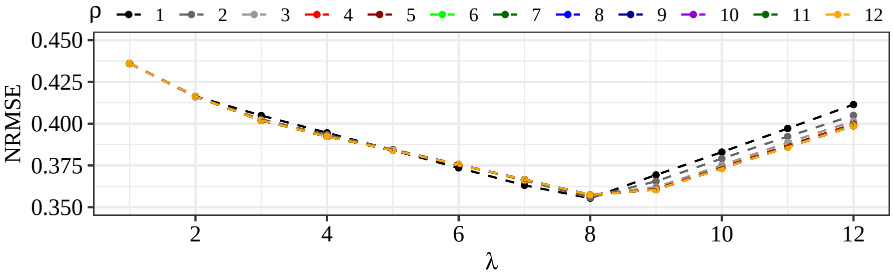


Figure 3.8: Impact of hyperparameters for $h = 1$, considering wind power dataset.

Table 3.5: NRMSE for synchronous models, considering wind power dataset.

	h=1	h=2	h=3	h=4	h=5	h=6
Persistence $(t)^*$	0.1045	0.1578	0.1939	0.2220	0.2452	0.2651
Analogs [179]†	0.1048	0.1552	0.1889	0.2145	0.2346	0.2515
LASSO-AR*	0.1008	0.1513	0.1830	0.2063	0.2242	0.2386
LASSO-VAR†	**0.0985**$^✓$	**0.1446**$^✓$	**0.1729**$^✓$	**0.1938**$^✓$	**0.2104**$^✓$	**0.2239**$^✓$

$*$ non-collaborative † collaborative
✓ statistically significant improvement against all others (DM test)

To access the quality of the proposed collaborative forecasting model, the synchronous LASSO-VAR is compared with benchmark models. Table 3.5 presents the NRMSE for all agents, per lead-time. According to the Diebold-Mariano test with a significance level of 5%, the improvements obtained by our proposal are statistically significant for all horizons.

Figure 3.9 complements this analysis by showing the relative improvement in terms of NRMSE for the 10 agents. Again, according to the Diebold-Mariano test, the LASSO-VAR model outperforms benchmarks in all lead-times for at least 9 out of the 10 agents. In general, the spatio-temporal information is more relevant for the highest lead-times, as corroborated by the cross-correlation plots at Figure 3.10, which shows cross-correlations between a sample of wind power plants. The cross-correlation between these wind power plants keeps increasing until lag 6; this means that, for example, the current power measurement at WF9 is more correlated with the power measurement of WF2 at 6 hours

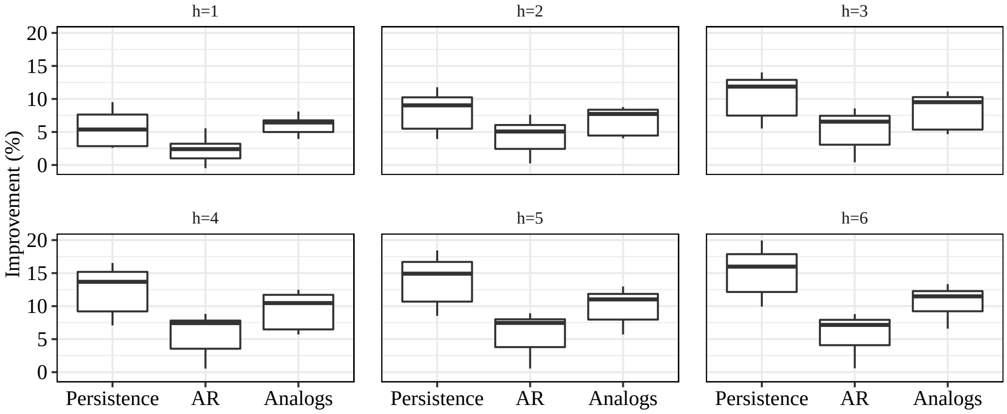
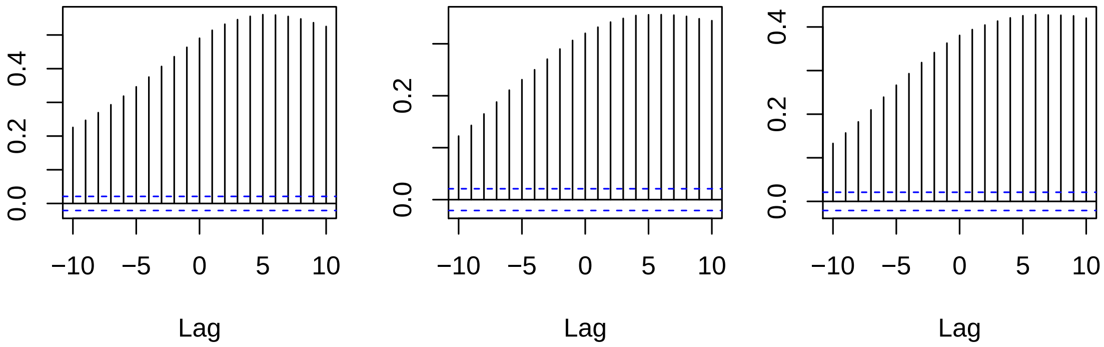


Figure 3.9: Relative NRMSE improvement (%) over the baseline models, considering wind power dataset.



(a) WF3(t) with WF1(t-Lag) **(b)** WF8(t) with WF2(t-Lag) **(c)** WF9(t) with WF2(t-Lag)

Figure 3.10: Cross-correlation plot (CCF) between two wind power plants.

ago. It is intuitively expected that this is due to the geographical layout (Figure 3.7 (a)) of the various wind farms and meteorological particularities of the region, such as wind speed. Figure 3.7 (b) depicts the wind rose for a location close to WF9[1], which shows that the wind direction during these two years was quite varied, but the strongest winds occur mostly from northwest or west, meaning that wind power plants located to the east (WF9, WF10) or southeast (WF5, WF6, WF7, WF8) can strongly benefit from the lags of wind farms WF1 to WF4.

Concerning computational complexity, Table 3.6 presents the mean running times and the number of iterations of both non-distributed and distributed approaches. When compared to a non-distributed LASSO-VAR version, the proposed schemes require larger execution times since they require estimating $\mathbf{B'}^{k}_{A_i}$ through a second ADMM cycle (Algorithm 2). However, the non-distributed LASSO-VAR requires more iterations to converge.

Table 3.6: Mean running times (in sec) per iteration and number of iterations until convergence, considering wind power dataset.

Non distributed	Central LASSO-VAR		P2P LASSO-VAR	
LASSO-VAR	Enc. data	ADMM	Enc. data	ADMM
0.038 (≈ 400)	125.46	0.059 (≈ 300)	125.46	0.1309(≈ 300)

Table 3.7: Mean relative NRMSE improvement (%) of the asynchronous ADMM LASSO-VAR over the LASSO-AR model, considering wind power dataset.

	h=1		h=2		h=3		h=4		h=5		h=6	
p_i	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P	central	P2P
0	2.25		4.26		5.30		5.83		5.94		5.95	
0.1	2.11	2.25	4.18	4.26	5.22	5.30	5.71	5.83	5.76	5.94	5.71	5.95
0.3	1.97	”	4.09	”	4.21	”	4.53	”	5.04	”	5.58	”
0.5	1.85	”	3.48	”	3.65	”	3.84	”	4.27	”	4.72	”
0.7	1.51	”	2.97	”	2.89	”	3.41	”	3.80	”	3.98	”
0.9	0.97	1.04	2.21	4.01	2.32	4.98	2.97	5.52	3.09	5.76	3.12	5.63

Finally, regarding asynchronous LASSO-VAR ($p_i \geq 0.1$), Table 3.7 summarizes the mean NRMSE improvement for all agents over the LASSO-AR model, considering different failure probabilities $p_i, i \in \{1, \dots, n\}$. In general, the greater the p_i the smaller the

[1] https://mesonet.agron.iastate.edu/ (accessed on January 2021)

improvement. Despite the model's accuracy decreases slightly, the LASSO-VAR model continues to outperform the LASSO-AR model for both collaborative schemes, which demonstrates high robustness to communication failures.

3.5 Concluding Remarks

RES forecast skill can be improved by combining data from multiple geographical locations. One of the simplest and most effective collaborative models for very short-term forecasts is the vector autoregressive model. However, different data owners might be unwilling to share their time series data. In order to ensure data privacy, this work combined the advantages of the ADMM decomposition method with data encryption through linear transformations of data. It is important to underline that the coefficients matrix obtained with the privacy-preserving protocol is the same one obtained without any privacy protection.

This novel method also included an asynchronous distributed ADMM algorithm, making it possible to update the forecast model based on information from a subset of agents and improve the computational efficiency of the proposed model. The mathematical formulation is flexible enough to be applied in two different collaboration schemes (central hub model and P2P) and paved the way for learning models distributed by features, instead of observations.

The results obtained for a solar and a wind energy dataset show that the privacy-preserving LASSO-VAR model delivers a forecast skill comparable to a model without privacy protection and outperformed a state-of-the-art method based on analog search. Furthermore, it exhibited high robustness to communication failures, in particular for the P2P scheme.

Lastly, an alternative business model to privacy-preserving models are data markets, where different agents sell and buy data of relevance for RES forecasting. In this case, agents are prone to share their data if being remunerated for it. The next chapter is focused on data monetization, and an auction mechanism is proposed in which both data privacy and monetization are possible by considering that agents buy forecasts from a trusted entity instead of directly buying sensible data.

4
Data Market for RES Forecasting

Abstract. Geographically distributed wind turbines, photovoltaic panels and sensors (e.g., pyranometers) produce large volumes of data that can be used to improve Renewable Energy Sources (RES) forecasting skill. However, data owners may be unwilling to share their data, even if privacy is ensured, due to a form of prisoner's dilemma: all could benefit from data sharing, but in practice no one is willing to do so. Our proposal hence consists of a data marketplace, to incentivize collaboration between different data owners through the monetization of data. We adapt here an existing auction mechanism to the case of RES forecasting data. It accommodates the temporal nature of the data, i.e., lagged time series act as covariates and models are updated continuously using a sliding window. Two test cases, with wind and solar energy data, are presented to illustrate and assess the effectiveness of such data markets. All agents (or data owners) are shown to benefit in terms of higher revenue resulting from the combination of electricity and data markets. The results support the idea that data markets can be a viable solution to promote data exchange between RES agents and contribute to reducing system imbalance costs.

4.1 Introduction

A large amount of data is being collected from geographically distributed RES such as wind turbines and photovoltaic panels. These data include power generation and weather measurements like air temperature, wind speed and direction, irradiation, etc.

Recent literature suggests that time series data from spatially distributed RES agents can improve forecasting skill for different time horizons. For instance, a spatial grid of Numerical Weather Prediction (NWP) can improve days-ahead forecasts [12]; turbine-level data can improve the day-ahead forecasting skill of wind energy through density forecasts generated for all wind turbines with spatial dependency structure modelled via copula theory [86]. Geographically distributed time series data can improve forecasting skill up to 6 hours-ahead for wind [78] and solar energy [84]. In fact, hours-ahead forecasts will become a crucial input for decision-aid as intraday electricity markets (e.g., European cross-border intraday – XBID) become increasingly important for RES technology.

However, since RES agents are most likely competitors in the same electricity market, they are unwilling to share data, particularly power measurements, even if data privacy is ensured. An effective way to encourage agents to share their data is through monetary compensation [190, 191]. A "secondary" market to trade data is necessary to monetize RES forecasting data. Moreover, this data market should operate in a way that, after some iterations, agents realize which data is relevant to improve its gain, so that sellers are paid according to their data. The buyers' gain should be a function of the forecast accuracy and value in a specific use case, e.g., imbalance costs reduction in electricity market bidding. It is important to mention that a RES plant owner can buy, from a vendor, NWP for neighbor power plants, but not their power measurements (or forecasts) that contain relevant information to improve hours-ahead forecasting skill. By joining a data market, the data owner can also sell this additional data (e.g., NWP for nearby sites) and decrease its purchasing cost. Moreover, there are no guarantees that NWP for other locations are cheaper than buying information from a data market where the payment is a function of the forecasting skill improvement. In fact, when buying NWP from vendors, there are no *a priori* guarantees of improvement in the existing forecasting model.

A data auction mechanism is proposed in [192] where sellers compute the privacy cost of selling the data and then send it to a buyer that computes a utility score associated with the data. Several iterations are performed until a Bayesian Nash equilibrium is reached. A market mechanism is introduced in [193] to solve a social welfare maximization problem that defines the data allocation and corresponding price. In this case, data are only shared after payment. However, in order to compute data price, a utility function, which depends solely on quantity (i.e., data quality is not considered), is assumed to exist. This is not directly applicable to time series forecasting with RES spatial data where correlated data from neighbor agents might be less informative than data from more distant agents (or sites). Furthermore, in [194], the impact of a strong correlation between data of different agents is analyzed as a negative externality from data sharing, e.g., buying the data from user A may reveal too much information about user B and the market price tends to zero (i.e., no value for data privacy). Different policies (e.g., “de-correlation”) and regulatory schemes to data markets are proposed and analyzed. In [195], evolutionary game theory is combined with blockchain smart contracts to dynamically adjust incentives and participation costs in data sharing. In the energy domain, a market is proposed in [196] for smart meter data. The proposed game theory mechanism works as follows: (i) the consumer maximizes its reward from sharing consumption data; (ii) data aggregator expects to receive more money from the data analyst, rather than providing incentives to consumers; (iii) data analyst is interested in high quality data at the lowest possible cost. Also for smart meter data, a blockchain smart contract is designed in [197] to define a set of rules for data access control and reward against privacy risk. In both works, the payment is directly related to the privacy loss and not directly linked to the gain obtained from using this data in a specific decision-making problem. The concept of pricing data as a function of privacy loss is further discussed in [198], where the impact of sellers’ risk attitude is analyzed.

Moreover, the temporal nature of RES forecasting also needs to be considered. An auction mechanism for time series data is proposed in [199] where privacy is guaranteed with data distortion by adding random noise, in a way that preserves some time series statistics and avoids the original series to be recreated when sold incrementally. Buyers ask for specific features together with the maximum noise they are willing to tolerate. Based on the level of noise, the market operator determines the privacy loss for selected data owners and sets the market prices to compensate them for the privacy loss. Buyer gain is not considered.

Since RES agents may be unwilling to share their data with competitors and mask of sensible data through noise addition involves a trade-off between privacy and accuracy [114], the framework from [200] offers an appealing alternative based on cooperative game theory. As far as we know, this is the first work to consider a marketplace where data owners purchase forecasts and pay according to resulting forecasting accuracy. This avoids the confidentiality problem of sharing raw data directly. Cooperation between sellers is done through a market operator who receives all agents data and prepares forecasts: (i) sellers with similar information receive similar revenue, (ii) the market price is a function of the buyer’s benefit, and so the buyer does not pay if there is no improvement in the forecasting skill, (iii) buyers pay according to incremental gain, and (iv) buyers purchase forecasts, instead of features, and have no knowledge about which datasets were used to produce these forecasts. Sellers' loss is assumed to be zero.

Nevertheless, adaptions are necessary since time series models require temporal updates of the input variables. Thus, the present chapter presents the following original contributions:

i) The approach from [200] is extended for a sliding window environment and the gain function is adapted for RES forecasting and bidding in the electricity market.

ii) With geographically distributed time series data, buyers want to integrate private and local data into the market operator's forecasts in order to avoid paying for highly-correlated data from close neighbors and this requirement is covered in the proposed approach – the approach in [200] does not consider RES agents with internal forecasting models and for which highly-correlated features might provide no improvement.

iii) Agents trade between themselves, i.e., sellers are buyers and buyers are sellers – sellers and buyers are independent agents in [200], thus adaptions are required to ensure that agents do not pay for their own or redundant data.

To the best of our knowledge, this is the first work to describe an algorithmic solution for data markets that enable different RES agents to sell data (historical power production, NWP, etc.) and buy forecasts of their power production, and where the economic value of this data is fundamentally related to imbalance cost reduction in electricity markets.

The chapter is organized as follows. Section 4.2 formalizes the electricity market and forecasting framework. Section 4.3 proposes a data market for RES forecasting. Then, three test cases are considered in Section 4.4, two with synthetic data and another with Nord Pool wind energy data. The work concludes in Section 4.5.

4.2 Electricity Market

RES market agents aim to minimize imbalance costs (i.e., maximize electricity market profit) by improving forecasting skill. This section presents the market profit function and the formulation of the forecasting problem.

4.2.1 Profit Function

In a typical electricity market with dual price imbalance settlement [201], the profit function of a RES market agent, with power measurement x_t and forecast $\hat{x}_t$, is determined for each time step t as

$$\rho(\hat{x}_t, x_t) = \pi_t^s x_t - C_t^{\uparrow/\downarrow}, \tag{4.1}$$

where

$$C_t^{\uparrow/\downarrow} = \begin{cases} \lambda^{\uparrow}(\hat{x}_t - x_t), & \hat{x}_t > x_t \\ -\lambda^{\downarrow}(\hat{x}_t - x_t), & \hat{x}_t < x_t, \end{cases} \tag{4.2}$$

$$\lambda_t^{\uparrow} = \max(0, \pi_t^{\uparrow} - \pi_t^s), \tag{4.3}$$

$$\lambda_t^{\downarrow} = \max(0, \pi_t^s - \pi_t^{\downarrow}), \tag{4.4}$$

with π_t^s, $\pi_t^{\uparrow}$ and $\pi_t^{\downarrow}$ denoting the spot price, imbalance price for upward and downward regulation, respectively; $\lambda_t^{\uparrow}$ and $\lambda_t^{\downarrow}$ give the regulation unit cost for upward and downward directions.

For simplicity, generation costs are not considered in the profit function ρ. Furthermore, by calculating the derivative of the expected regulation cost with respect to the bid [201], it is possible to conclude that forecasts that maximize the profit in (4.1) do not correspond to the expected value of x_t, instead, they correspond to the quantile of the following nominal level,

$$\alpha_t^* = \frac{\hat{\lambda}_t^{\downarrow}}{\hat{\lambda}_t^{\uparrow} + \hat{\lambda}_t^{\downarrow}}, \tag{4.5}$$

where $\hat{\lambda}_t^{\uparrow}, \hat{\lambda}_t^{\downarrow}$ are deterministic forecasts for $\lambda_t^{\uparrow}, \lambda_t^{\downarrow}$.

This means that the optimal bid (i.e., the one that minimizes the expected imbalance costs in (4.2)) for a RES agent $i \in \{1, \dots, N\}$ is given by $\hat{F}_{i,t}^{-1}(\alpha_t^*)$ [201], where $\hat{F}_{i,t}^{-1}$ is the inverse of the forecasted cumulative distribution function or, in other words, corresponds to the forecasted conditional quantile for nominal level α_t^*. These analytical formulas for optimal bidding can be generalized for other situations, such as a joint offer of energy and reserve capacity [202].

In order to compute the "optimal" quantile from (4.5), a forecast of the regulation unit costs is required. Since we do not aim to propose a new forecasting model for imbalance prices, the Holt-Winters model described in [203] was used in this work. The upward regulation unit cost is estimated as the product between the forecasted upward regulation price ($\hat{\psi}_t^{\uparrow}$) and the probability of the system to be in upward regulation direction ($\hat{p}_t^{\uparrow}$), i.e.

$$\hat{\lambda}_t^{\uparrow} = \hat{\psi}_t^{\uparrow} \hat{p}_t^{\uparrow}. \tag{4.6}$$

Similarly,

$$\hat{\lambda}_t^{\downarrow} = \hat{\psi}_t^{\downarrow} \hat{p}_t^{\downarrow}, \tag{4.7}$$

where $\hat{p}_t^{\downarrow} = 1 - \hat{p}_t^{\uparrow}$ since we only care about relative probabilities for upward and downward regulation. The regulation prices are forecasted by

$$\hat{\psi}_{t|t-1}^i = \begin{cases} \eta \hat{\psi}_{t-1|t-2}^i + (1-\eta)(\lambda_{t-1}^i - \hat{\psi}_{t-1|t-2}^i), & |\lambda_{t-1}^i| > 0 \\ \hat{\lambda}_{t-1|t-2}^i, & |\lambda_{t-1}^i| = 0, \end{cases} \tag{4.8}$$

for $i \in \{\uparrow, \downarrow\}$, and the probability of system regulation direction by

$$\hat{p}_{t|t-1}^{\uparrow} = \begin{cases} \eta \hat{p}_{t-1|t-2}^{\uparrow} + (1-\eta)(p_{t-1}^{\uparrow} - \hat{p}_{t-1|t-2}^{\uparrow}), & p_{t-1}^{\uparrow} \neq 0.5 \\ \hat{p}_{t-1|t-2}^{\uparrow}, & p_{t-1}^{\uparrow} = 0.5, \end{cases} \tag{4.9}$$

where $\eta \in [0, 1[$ is a smoothing factor, and

$$p_{t-1}^{\uparrow} = \begin{cases} 1, & \lambda_{t-1}^{\uparrow} > \lambda_{t-1}^{\downarrow} \\ 0.5, & \lambda_{t-1}^{\uparrow} = \lambda_{t-1}^{\downarrow} \\ 0, & \lambda_{t-1}^{\uparrow} < \lambda_{t-1}^{\downarrow}. \end{cases} \tag{4.10}$$

Initialization of $p_0^{\uparrow}$, $\lambda_0^{\uparrow}$ and $\lambda_0^{\downarrow}$ is required, and η is estimated by minimizing the mean of squared residuals.

Given the forecasted values for regulation unit costs, the last step is to forecast the quantile with nominal level α_t^* using linear quantile regression as described in the next subsection. Note that here we are assuming a price-taker RES agent for the regulation market.

4.2.2 RES Forecasting Problem

In this work, we formulate a very short-term forecasting problem (up to 6h-ahead) involving multiple RES power plants. The forecasting model only uses recent measurements at all sites of interest, but longer time horizons with extra variables, such as grid of NWP [12] and turbine-level data [86], may also be considered using the same framework.

Assume that RES power plants generation data are collected at n sites, and $x_{i,t}$ denotes the power measurement at site i and time t, $i \in \mathcal{A}$, $t \in \{1,\dots,T\}$, where T is the number of time steps in the dataset and $\mathcal{A}=\{1,\dots,n\}$ is the overall set of power plants. We consider that these agents operate a single power plant, but the case where agents operate a portfolio of RES power plants may also be elaborated using the same framework.

The linear Quantile Regression (QR) model, discussed in Section II.3.1, is a standard and straightforward method of conditional quantile estimation [204]. For very short-term forecasts, satisfactory results may be obtained by using the L most recent observations, as shown in [84] and [78] for both solar and wind energy.

In this case, the quantile α^*_{t+h} of power $x_{i,t+h}$ in site $i \in \mathcal{A}$ is expressed as

$$\hat{q}^{i}_{\alpha^*_{t+h}} = \beta^{(\alpha^*_{t+h})}_{0,i} + \sum_{\ell=1}^{L} \Big(\underbrace{\sum_{j\in\mathcal{A}\setminus\{i\}} \hat{\beta}^{(\alpha^*_{t+h})}_{j,i,\ell}\, x_{j,t-\ell}}_{\text{data from the market}} + \underbrace{\hat{\beta}^{(\alpha^*_{t+h})}_{i,i,\ell}\, x_{i,t-\ell}}_{\text{own data}} \Big), \tag{4.11}$$

where $h \leq 6$ is the forecasting horizon, $\beta^{(\alpha^*_t)}_{0,i}$, $\beta^{(\alpha^*_t)}_{j,i,\ell}$ and $\beta^{(\alpha^*_t)}_{i,i,\ell}$ are the unknown coefficients, estimated through the minimization of the pinball loss function [204].

4.3 Market based in Cooperative Game Theory

This section proposes a no-regret auction mechanism for trading RES forecasts, as illustrated in Figure 4.1. The buyers should never buy data because its value is unknown before using it for a forecasting task. Instead, they should purchase forecasts of their power production and pay according to the obtained forecasting accuracy. The data market formulation is inspired by the cooperative game in [200] and described in the following subsection in order to be self-content.

In addition to large RES power plants, this data market is also open to prosumers. Interestingly, data traders can also be interpreted as *data prosumers*, i.e., data owners that consume and supply data.

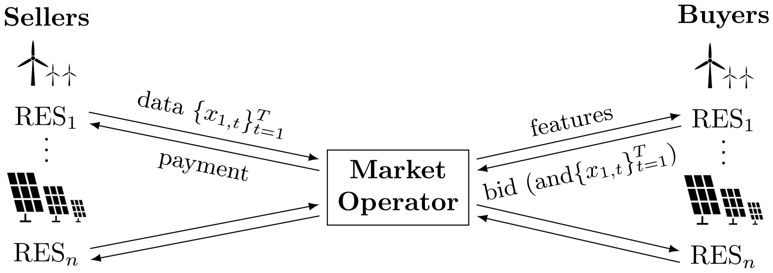


Figure 4.1: Proposed data market framework.

4.3.1 Data Market Agents

Like any standard market, the data market has three types of agents described in this subsection: sellers, buyers and market operator.

Sellers

A seller i observes and sells sample $\mathbf{x}_i^{\mathrm{S}}=\{x_{i,t}\}_{t=1}^{T}$, $\mathbf{x}_i^{\mathrm{S}} \in \mathbb{R}^T$, $i \in \{1, \ldots, n\}$. Additionally, sellers have no idea of the forecasting methods that will use their data and simply aim to maximize their revenue. The set of features provided by all sellers is denoted by $\mathbf{X}^{\mathrm{S}}=[\mathbf{x}_1^{\mathrm{S}}, \ldots, \mathbf{x}_n^{\mathrm{S}}]$, $\mathbf{X}^{\mathrm{S}} \in \mathbb{R}^{T\times n}$.

Buyers

A buyer i observes and seeks to improve sample $\mathbf{x}_i^{\mathrm{B}} = \{x_{i,t}\}_{t=1}^{T}$, $i \in \{1, \ldots, n\}$, and enters the data market to purchase the collection of features that allow a certain gain when forecasting $\{x_{i,t}\}_{t=T+1}^{T+H}$, through a selected method (statistical model) $\mathcal{M}_i$, $H \geq 1$. Buyers naturally have a local forecasting model $\mathcal{M}_i(\mathbf{x}_i^{\mathrm{B}})$, and enter the market to improve it with more features from the other agents, $\mathbf{X}_{\neg i}^{\mathrm{S}}$, where $\mathbf{X}_{\neg i}^{\mathrm{S}}=[\mathbf{x}_1^{\mathrm{S}}, \ldots, \mathbf{x}_{i-1}^{\mathrm{S}}, \mathbf{x}_{i+1}^{\mathrm{S}}, \ldots, \mathbf{x}_n^{\mathrm{S}}]$, $\mathbf{X}_{\neg i}^{\mathrm{S}} \in \mathbb{R}^{T\times(n-1)}$. Therefore, the gain of power agent i at time $t \in \{T{+}1, \ldots, T{+}H\}$ is measured by its marginal profit,

$$\mathcal{G}_i(x_{i,t}; \mathbf{X}^{\mathrm{S}}, \mathcal{M}_i) = \Big(\rho(\hat{x}_{i,t}^{\mathrm{market}}, x_{i,t}) - \rho(\hat{x}_{i,t}^{\mathrm{local}}, x_{i,t})\Big)^+, \tag{4.12}$$

where $(x)^+{=}\max(0,x)$, $\hat{x}_{i,t}^{\mathrm{local}}{=}\mathcal{M}_i(\mathbf{x}_i^{\mathrm{B(ts)}}; \mathbf{x}_i^{\mathrm{B(tr)}})$ is the forecast using only data from buyer i and $\hat{x}_{i,t}^{\mathrm{market}}{=}\mathcal{M}_i(\mathbf{X}^{\mathrm{S(ts)}}; \mathbf{X}^{\mathrm{S(tr)}})$ is the forecast obtained by combining local data and data from other agents — $\mathbf{x}_i^{\mathrm{B(tr)}}, \mathbf{X}^{\mathrm{S(tr)}}$ are the sets used to train the models, while $\mathbf{x}_i^{\mathrm{B(ts)}}, \mathbf{X}^{\mathrm{S(ts)}}$ are the sets used to forecast $\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$. By simplicity, the same model $\mathcal{M}$ and gain function $\mathcal{G}$ are used for all the buyers, but conceptually buyers may provide their own $\mathcal{M}_i$ and $\mathcal{G}_i$ to the market operator.

The last two parameters from buyers are the private valuation of gain $\mu_i \in \mathbb{R}^+$, i.e., a trade-off value that means how much buyer i is willing to pay for a unit increase in gain, and the public bid price $b_i \leq \mu_i, b_i \in \mathbb{R}^+$. Note that buyers enter the market to buy forecasts $\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$, without knowing which data were used to produce the forecasts, $H \geq 1$.

Market Operator

The role of the market operator includes feature allocation (Section 4.3.3), market price definition (Sections 4.3.3 and 4.3.3), revenue extraction from the buyers (Section 4.3.3) and corresponding distribution to the sellers (Section 4.3.3).

It is important to underline that only the market operator has access to input data (power measurements, NWP, etc.) and is responsible for fitting the quantile regression model described in Section 4.2.2. Sellers only have access to their own time series and buyers only have access to power forecasts produced for their power plants. Therefore, data privacy is guaranteed, assuming that the market operator is a trustworthy and neutral agent. Note that the data market framework can be applied to any forecasting methodology and the use of quantile regression is not a fundamental requirement.

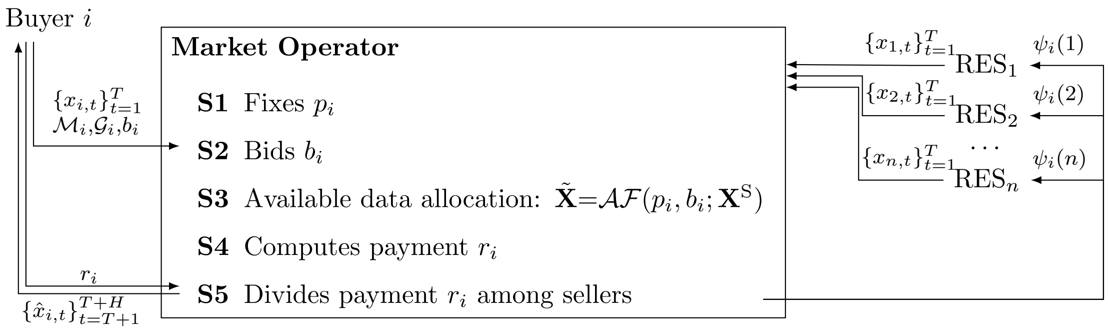

Buyer i
$\{x_{i,t}\}_{t=1}^{T}$
$\mathcal{M}_i$,$\mathcal{G}_i$,b_i
r_i
$\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$
Market Operator
S1 Fixes p_i
S2 Bids b_i
S3 Available data allocation: $\tilde{\mathbf{X}}$=$\mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}})$
S4 Computes payment r_i
S5 Divides payment r_i among sellers
$\{x_{1,t}\}_{t=1}^{T}$
RES$_1$
$\psi_i(1)$
$\{x_{2,t}\}_{t=1}^{T}$
RES$_2$
$\psi_i(2)$
$\{x_{n,t}\}_{t=1}^{T}$
RES$_n$
$\psi_i(n)$

Figure 4.2: Data market mechanism at time $t = T$.

4.3.2 Data Market Mechanism

At time $t = T$, RES agents provide their historical data to the market operator. Then, agent i aims to forecast the power for the next H time steps, $\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$, and the following steps occur in sequence (illustrated in Figure 4.2):

Step 1 The marketplace sets a market price $p_i \in \mathbb{R}^+$ for a unit increase in gain when forecasting $\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$, following the market solution (i.e., bid and market price, forecasting accuracy) for the previous buyer $i-1$,

$$p_i = \mathcal{PF}(b_{i-1}, p_{i-1}; \Theta_{i-1}), \tag{4.13}$$

where $\mathcal{PF}$ is the market price update function, and $\Theta_{i-1} = (\mathcal{M}_{i-1}, \mathcal{G}_{i-1}, \mathbf{X}^{\mathrm{S}}, \mathbf{x}_{i-1}^{\mathrm{B}})$ – market operator decides p_i before buyer i arrives and according to the previous prices, otherwise truthfulness is not ensured.

Step 2 Buyer i bids b_i, which maximizes its value function,

$$b_i{=}\arg\max E_{z\in\mathbb{R}^+} \underbrace{\mu_i \textstyle\sum_t \mathcal{G}_i(x_{i,t}; \Theta_i) {-} \mathcal{RF}(p_i, z; \Theta_i)}_{\mathcal{U}_i(z,\{x_{i,t}\}_{t=T+1}^{T+H}) \text{ = value function}}, \tag{4.14}$$

and is related to the difference between the value derived from the gain in forecasting accuracy and the data market price, $t \in \{T+1, \dots, T+H\}$. $\mathcal{RF}$ is the revenue function.

Step 3 The marketplace allocates available features according to the market price and bid price,

$$\tilde{\mathbf{X}} = \mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}}), \tag{4.15}$$

with $\mathcal{AF}$ representing the allocation function.

Step 4 The marketplace extracts revenue r_i from buyer i,

$$r_i = \mathcal{RF}(p_i, b_i; \Theta_i). \tag{4.16}$$

Step 5 Market divides r_i among the $n-1$ sellers using

$$\psi_i(m) = \mathcal{PD}(\mathbf{x}_i^{\mathrm{B}}, \tilde{\mathbf{X}}, K; \mathcal{M}_i, \mathcal{G}_i), \tag{4.17}$$

where $\mathcal{PD}$ is the payment division function, $m \in \mathcal{A}\backslash\{i\}$.

Step 6 Buyer i receives $\{\hat{x}_{i,t}\}_{t=T+1}^{T+H}$ and leaves the market.

Step 7 If a new time step occurred, sellers update their data and send it to the market operator.

4.3.3 Market Configuration

Certain properties must be met in order to produce a fair auction mechanism when defining $\mathcal{PF}$, $\mathcal{AF}$, $\mathcal{RF}$ and $\mathcal{PD}$, from (4.13) to (4.17). First, the auction mechanism needs to encourage buyers to declare their true valuation for an increase in forecasting skill. This is achieved through the *allocation* and *revenue* functions. From the other side, the market operator needs to incentivize sellers to participate in the market, meaning that the *revenue division* function should ensure three properties:

i) money paid by the buyer is totally divided by the sellers;

ii) sellers with similar information receive the same amount of money;

iii) irrelevant information receives zero payment.

Allocation Function

The allocation function $\mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}})$ defines the information that marketplace should use when forecasting the time series of buyer i. The proposed mechanism assumes that all available features are used to train and evaluate the forecasting model. However, in order to ensure that the allocated features are a function of the difference between the bid price and the market price, the model is fitted (and the gain is estimated) using a perturbed version of competitors' data. More specifically, the allocated features are obtained by

$$\tilde{x}_{j,t} = \begin{cases} x_{j,t} + \max(0, p_i - b_i)\mathcal{N}(0, \sigma^2), & j \neq i \\ x_{j,t}, & j = i. \end{cases} \tag{4.18}$$

where $\mathcal{N}(0, \sigma^2)$ is a univariate Gaussian distribution.

Revenue Function

The revenue function $\mathcal{RF}(p_i, b_i; \Theta_i)$ is computed by the market operator based on its model estimation for each buyer i. The market price is based on the gain to buyer i, which is unknown for the future but can be estimated through holdout cross-validation. While forecasting $\{x_{i,t}\}_{t=T+1}^{T+H}$, the marketplace splits $\mathbf{X}^{\mathrm{S}}$ into training, validation and testing data, $\mathbf{X}^{\mathrm{S(tr)}}$ is used to estimate the model, $\mathbf{X}^{\mathrm{S(val)}}$ is used to estimate the gain and $\mathbf{X}^{\mathrm{S(ts)}}$ to forecast $\{x_{i,t}\}_{t=T+1}^{T+H}$. $\mathbf{X}^{\mathrm{S(val)}}$ corresponds to the set used to forecast the last Δ values $\{x_{i,t}\}_{t=T}^{T-\Delta+1}$, and $\mathbf{X}^{\mathrm{S(tr)}}$ to the sample used to forecast the remaining $T-\Delta$ observations $\{x_{i,t}\}_{t=1}^{T-\Delta}$, as illustrated in Figure 4.3, $\Delta \geq 1$. Moreover, as previously mentioned, the data market should price the forecasts according to the marginal gain accrued to its buyers. Figure 4.4 illustrates the difference between paying by the gain and paying by the marginal gain as defined by the Myerson's payment function rule [205]

$$\begin{aligned} \mathcal{RF}(p_i, b_i; \Theta_i) = {} & b_i \mathcal{G}_i(\mathbf{x}_i^{\mathrm{B(val)}}; \mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}}), \mathcal{M}_i) \\ & - \int_0^{b_i} \mathcal{G}_i(\mathbf{x}_i^{\mathrm{B(val)}}; \mathcal{AF}(z, b_i; \mathbf{X}^{\mathrm{S}}), \mathcal{M}_i) dz, \end{aligned} \tag{4.19}$$

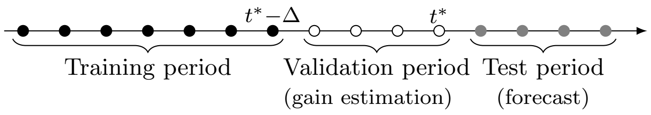
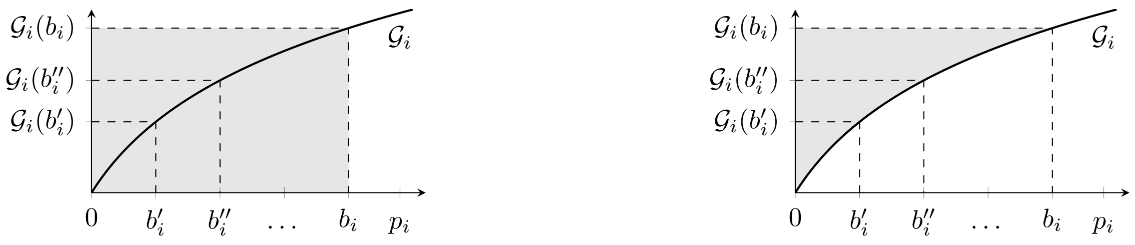


Figure 4.3: Timeline for current time t^*.



(a) Based on gain ($\mathcal{G}_i$)

(b) Based on *marginal* gain ($\mathcal{RF}$)

Figure 4.4: Difference between paying by the gain and paying by the marginal gain (market price = shadow area, x axis = bid price, y axis = gain).

which is adopted in this chapter — if bid prices b_i' and b_i'', with $b_i'' > b_i'$, might produce similar gain, $\mathcal{G}_i(b_i') \approx \mathcal{G}_i(b_i'')$, then a RES agent is incentivized to bid b_i'' anyway since it would only pay b_i' according to the marginal gain rule.

A revenue close to zero means that the buyer is purchasing low-quality forecasts, particularly when bid and market prices are high and an higher revenue from data sharing was expected.

Payment Division Function

The payment division function $\mathcal{PD}(\mathbf{x}_i^{\mathrm{B(val)}}, \mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}}), K; \mathcal{M}_i, \mathcal{G}_i)$ divides the value r_i paid by buyer i among the $n-1$ sellers. Ideally, the relevance of each feature would be estimated by training the statistical model $\mathcal{M}_i$ with all possible feature combinations. This method is known as Shapley Allocation [206] and ensures the three properties listed at the beginning of this section. However, when a large number of sellers is considered, this strategy may be computationally infeasible.

To overcome this challenge, the Shapley Approximation method uses a smaller number of possible feature combinations [207]. Given a random permutation σ of all features' indices $\{1, \dots, n\}$, from an universe $\boldsymbol{\sigma}$, two models are trained using the features given by $\sigma_i < m$ and $\sigma_i \leq m$. The importance of a feature m is given by the difference in gains between these two models. The process is repeated K times and averaged out. Theoretically, the Shapley approximation $\hat{\psi}_i(m)$ achieves $\|\psi_i^{\mathrm{shapley}}(m) - \hat{\psi}_i(m)\| < \varepsilon$, with probability $1-\zeta$ if $K > [n \log(2/\zeta)]/(2\varepsilon)^2$. Since the models are trained multiple times for different agents, the choice of the model $\mathcal{M}_i$ clearly affects the computational efficiency of the payment division function.

Furthermore, a post-processing step is applied to make the algorithm more robust to data replication. Consider a data market with three sellers, S_1, S_2 and S_3, such that S_1 and S_2 have uncorrelated and equally relevant data for buyer i, while S_3 is irrelevant, i.e $\psi_i(1)=\psi_i(2)=0.5$ and $\psi_i(3)=0$. If S_1 replicate its data once and sell again in the marketplace, the proportion of received payment will be $\psi_i(1)=2/3$, $\psi_i(2)=1/3$. Since sellers provide a unique time series, they cannot replicate data; yet, they can collude with other agents and negotiate a portion of the extra revenue. If S_1 and S_3 collude, then $\psi_i(1)=\psi_i(2)=\psi_i(3)=1/3$.

In order to avoid data replication, the weight $\psi_i(m)$ of each seller m is penalized if its data are similar to others in the market. This penalty is related to the cosine similarity, which measures the similarity between two vectors $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^T$ as

$$\mathcal{SM}(\mathbf{x}_1, \mathbf{x}_2) = \frac{|\langle \mathbf{x}_1, \mathbf{x}_2 \rangle|}{\|\mathbf{x}_1\| \|\mathbf{x}_2\|}, \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^T, \tag{4.20}$$

where $|\langle . \rangle|$ and $\|.\|$ denote the absolute value of the dot product and the Euclidean norm, respectively.

Algorithm 3 illustrates the algorithm to determine the proportion that a seller should receive from the buyer's payment. Regarding the example with three sellers, if S_1 replicates data then the Shapley allocation using Algorithm 3 (with λ=1) decreases to $1/(2+e^2) < 1/2$.

Algorithm 3 Payment division algorithm ($\mathcal{PD}$).

```
Input: $\mathbf{x}_i^{\mathrm{S}}$, $\tilde{\mathbf{X}} = \mathcal{AF}(p_i, b_i; \mathbf{X}^{\mathrm{S}})$, $\mathcal{M}_i$, $\mathcal{G}_i$, $K$
Output: $\psi_i = [\psi_i(m) \colon m \in \mathcal{A}\backslash\{i\}]$
for $m \in \mathcal{A}\backslash\{i\}$ do
    for $k \in \{1, \ldots, K\}$ do
        $\sigma_k \leftarrow \mathrm{Uniform}(\boldsymbol{\sigma})$
  # Train models with “tr” data and forecast with “val” data
        $G = \mathcal{G}_i(\mathbf{x}_i^{\mathrm{S(val)}}; \tilde{\mathbf{X}}_{[\sigma_k < m]}, \mathcal{M}_i)$
        $G^{+m} = \mathcal{G}_i(\mathbf{x}_i^{\mathrm{S(val)}}; \tilde{\mathbf{X}}_{[\sigma_k < m] \cup m}, \mathcal{M}_i)$
        $\hat{\psi}_i^k(m) = (G^{+m} - G)^+$
    end for
    $\hat{\psi}_i(m) = \frac{1}{K} \sum_{k=1}^{K} \hat{\psi}_i^k(m)$
end for
$\psi_i'(m) = \hat{\psi}_i(m) \exp(-\lambda \sum_{j \in \mathcal{A}\backslash\{i,m\}} \mathcal{SM}(\mathbf{x}_m^{\mathrm{S}}, \mathbf{x}_j^{\mathrm{S}}))$
$\psi_i(m) = \psi_i'(m) / \sum_{m \in \mathcal{A}\backslash\{i\}} \psi_i'(m)$
```

Market Price Update Function

The function $\mathcal{PF}(b_{i-1}, p_{i-1}; \Theta_{i-1})$ computes the market price of the data for buyer i based on the gain from the other agents. We assume a set of possible market prices $\mathcal{B}_p$, which ranges from a minimum value $p_{\min}$ and a maximum value $p_{\max}$, with increment Δ_p. When the data market initializes, the market price is uniformly sampled from $\mathcal{B}_p$. Then, the market operator uses the forecasting accuracy from the first agent and estimates the revenue for each possible market price. The probabilities are updated and used to generate the market price when a new buyer arrives, iteratively, ensuring the truthfulness of the data market. Algorithm 4 proposes an online balance for the trade-off between large and small market prices. Considering a bid price b_i, if p_i is too large then the positive term in $\mathcal{RF}$ will be small (as the deterioration of $\mathbf{X}^{\mathrm{S}}$ is very high) leading to lower revenue. Similarly, if p_i is too small, the negative term in $\mathcal{RF}$ will be large, which again leads to an undesired loss in revenue.

4.3.4 Available Platforms for Implementation

This marketplace can be implemented in readily available platforms and protocols, reviewed below, which enable data transaction, verification and payment capabilities.

Ocean Protocol is an ecosystem for data trading, built on top of blockchain technology, where Oceans Tokens are used as the unit of exchange for buying or selling data

Algorithm 4 Market price update algorithm ($\mathcal{PF}$).

```
Input: $b_{i-1}$, $p_{i-1}$, $p_{\min}$, $p_{\max}$, $\Delta_p$, $\Theta_i$=($\mathcal{M}_i, \mathcal{G}_i, \mathbf{X}^{\mathrm{S}}, \mathbf{x}_i^{\mathrm{B}}$)
Output: $p_i$
$\mathcal{B}_p \leftarrow [p_{\min}, p_{\min} + \Delta_p, p_{\min} + 2\Delta_p, \dots, p_{\max}]$
# Initialize the weights for each possible market price
$w_1^j \leftarrow 1, \forall j = 1, \dots, |\mathcal{B}_p|$
# When a buyer enters the market, the market price is determined and the weights are updated for the next buyer
for $i = 1, \dots, |\mathcal{A}|$ do
    $p_i \leftarrow \mathcal{B}_p(j)$ with probability $w_i^j / \sum_{j=1}^{|\mathcal{B}_p|} w_i^j$
    for $j = 1, \dots, |\mathcal{B}_p|$ do
        $g_i^j \leftarrow \mathcal{RF}(\mathcal{B}_p(j), b_i; \mathbf{\Theta}_i)$ # revenue for the $j$-th price
        $w_{i+1}^j \leftarrow w_i^j(1 + \delta g_i^j)$ # update weights
    end for
end for
```

services [208]. Enigma provides a protocol for secret contracts, which are similar to smart contracts but bring privacy by offloading the computation over sensitive data to an external network where it may be broken into different nodes and apply cryptographic techniques [209]. SingularityNET is a decentralized platform for trading Artificial Intelligence (AI) services, including data, through the native platform's cryptocurrency [210]. Numerai is an AI platform that aims at bringing together the best experts in data science for making forecasts for a common dataset and those who perform well are reward with some Numeraires (i.e., cryptocurrency token) and those who did not perform well will lose the Numeraires staked [211].

The majority of these platforms lack from an advanced model for data trading and, therefore, a synergy between the market mechanism described in this work and blockchain-powered platforms (e.g, tokens, protocols and smart contracts) can be established for a real-world implementation of this concept.

4.4 Case Studies

In this section, four different case studies are constructed to evaluate the proposed no-regret auction mechanism: (i) synthetic data with 3 agents aiming to verify, with a simple setup, how the data market operates; (ii) synthetic data with 50 agents, aiming to evaluate the effect of different covariance matrices in the data market; (iii) wind power data publicly available from the Nord Pool electricity market; and (iv) solar power data used in Chapters 2 and 3.

4.4.1 Synthetic Data: Simple Setup with 3 Agents

Data Description and Experiments

Three agents are assumed. Let $x_{i,t}$ denote the observations from agent i at time t, and $\mathbf{x}_t$=[$x_{1,t},\ x_{2,t},\ x_{3,t}$], where $i \in \{1, 2, 3\}$ and $t \in \{1, \dots, T\}$. The synthetic data are generated from the Vector AutoRegressive (VAR) model,

$$\mathbf{x}_t = \mathbf{x}_{t-1} \begin{pmatrix} 0.5 & 0.7 & -0.1 \\ 0 & 0.7 & 0.1 \\ 0 & 0 & 0.8 \end{pmatrix} + \boldsymbol{\varepsilon}_t, \tag{4.21}$$

where $\boldsymbol{\varepsilon}_t = \left[\varepsilon_{1,t},\ \varepsilon_{2,t},\ \varepsilon_{3,t}\right]$ are the error terms, $\varepsilon_{i,t} \sim \mathcal{N}(0,1)$.

As experiments, hour-ahead forecasts are validated using an out-of-sample fold with 150 consecutive time steps. The market operator uses a sliding window with the 8760 most recent observations divided in 8592 for model fitting and 168 to estimate the improvement in gain.

For the data market simulation, a linear regression is used as the model $\mathcal{M}_i$, $\forall i \in \{1,2,3\}$, with covariates provided by the 1h-lagged time series. The gain function $\mathcal{G}_i$ is the improvement over the model estimated by using only its own (lagged) time series, in terms of percentage of Normalized Root Mean Squared Error (NRMSE) measured for each agent i as

$$\text{NRMSE} = \frac{\sqrt{\frac{\sum_{t=1}^{T}(\hat{x}_{i,t}-x_{i,t})^2}{T}}}{\max(\{x_{i,t}\}_{t=1}^{T}) - \min(\{x_{i,t}\}_{t=1}^{T})} \times 100. \tag{4.22}$$

The market operator sets a market price between 0.50€ and 10€, with 0.50€ increment, for each 1% improvement in NRMSE when forecasting one time-step ahead. The auction mechanism is simulated through the following experiments, which assume that the buyers have the following bid prices (both market and bid prices are expressed in € per 1% improvement in NRMSE):

E1 A fixed bid price of 5€; i.e., each agent values a marginal improvement of 1% in NRMSE as 5€.

E2 Agents bid fixed values of 3€, 5€ and 7€, respectively.

E3 Agents bid fixed values of 7€, 5€ and 3€, respectively.

E4 Agents bid price according to the NRMSE of their local model. Agents with a poor local model are more prone to improve 1% in NRMSE. The functional relation between the bid price and local model NRMSE is expressed as

$$b(\text{NRMSE}) = \frac{10}{1 + \exp(-0.3 \times \text{NRMSE} + 5)}. \tag{4.23}$$

The NRMSE for the local model is estimated using the Δ most recent observations.

Results and Discussion

Figure 4.5 depicts market dynamics when buyers always bid price 5€. At the end of some iterations, the market price tends to the bid's price values. As expected, when the market price is below or equal to the bid price, the gain corresponds to the gain using the real model. On the other hand, when the market price is higher than the bid price, the gain is reduced as a consequence of the noise addition into the covariates from the other agents. Furthermore, in all experiments, agent 1 has the highest benefit when using data from the market, which was expected by (4.21).

Additionally, since the gain for agents 2 and 3 is small, their payment is also small even when the market is not adding noise to the covariates. The payment from agent 1

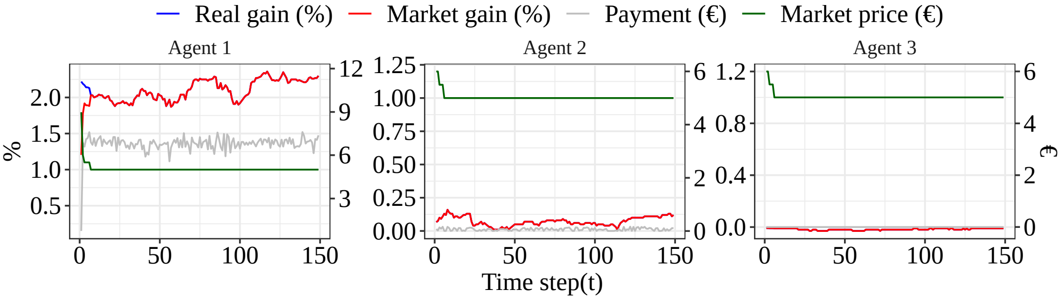


Figure 4.5: Market dynamics for experiment E1 (bid price is constant and equal to 5€).

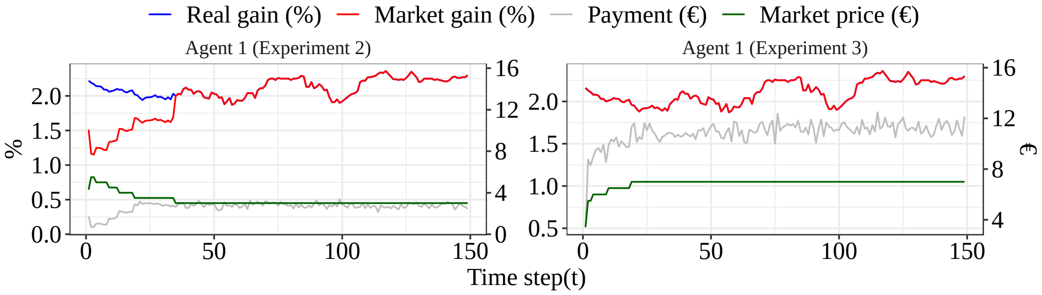


Figure 4.6: Market dynamics for agent 1 in experiments E2 and E3.

is divided by agents 2 and 3 through a mean percentage of 97.6% and 2.4%, respectively, which is coherent with the fair distribution. When some gain is estimated for agent 2, agent 3 receives 100% of the value paid. Even though agent 3 does not benefit in terms of forecast accuracy improvement, it receives money from agents 1 and 2 who are not aware that agent 3 is selling data in the market.

Figure 4.6 depicts data market dynamics for agent 1, at experiments E2 and E3. Since the gain to agents 2 and 3 is small, the market price is influenced by the bid price of agent 1, and the former conclusions stand. Furthermore, when the agent with the highest gain bids closer to the initial market price, the market price converges faster.

The market price and revenue dynamics for E4 (not depicted in Figure 4.6) are similar to the ones from E2, where agent 1 bids at a price higher than agents 2 and 3. Since the NRMSE for the local forecasting model is stationary for all agents (with values around 9.8%, 7.7% and 5.8%, respectively), the agents bid prices around 1.2€, 0.65€ and 0.37€ per 1% improvement in NRMSE, respectively. The market price converges to 1€ per 1% improvement in NRMSE.

4.4.2 Synthetic Data: 50 Agents

Data Description and Experiments

Let $\mathbf{x}_t$=$[x_{1,t},\dots,x_{50,t}]$. The synthetic data for the 50 agents are generated from the VAR model

$$\mathbf{x}_t = \mathbf{x}_{t-1}\mathbf{B}+\boldsymbol{\varepsilon}_t, \tag{4.24}$$

where $\mathbf{B}$ is the coefficient matrix, $\mathbf{B}\in\mathbb{R}^{50\times 50}$, and $\boldsymbol{\varepsilon}_t$=$[\varepsilon_{1,t},\dots,\varepsilon_{50,t}]$ is the error vector, $\varepsilon_{i,t}\sim\mathcal{N}(0,1),\forall i\in\{1,\dots,50\}$. Two datasets ($\mathbf{D}_1$ and $\mathbf{D}_2$) are generated to evaluate the effect of different covariance matrices in the proposed approach.

$\mathbf{D}_1$ assumes a sparse $\mathbf{B}$ matrix where: Agents $1, 2, 12, 16, 21$ and 43 should benefit with forecasts from the data market; agents $2, 3, 11, 12, 36$ and 44 should receive payment from the data market. $\mathbf{D}_2$ assumes a $\mathbf{B}$ matrix such that a large number of time series is highly-correlated.

As in Subsection 4.4.1, hour-ahead forecasts are validated using an out-of-sample fold with 150 consecutive time steps. The market operator uses a sliding window with the 8760 most recent observations divided in 8592 for model fitting and 168 to estimate the improvement in gain. $\mathcal{M}_i$ is a linear regression with covariates given by the 1h-lagged time series, and $\mathcal{G}_i$ is the improvement over the model estimated by using only its own (lagged) time series, in terms of NRMSE. The market operator sets a market price between 0.50€ and 10€, with 0.50€ increment, and each agent values a marginal improvement of 1% in NRMSE as 5€.

Results and Discussion

Table 4.1 summarizes the results for $\mathbf{D}_1$, at the end of 150 time steps. Sellers with data that improve the forecasts of other agents get higher revenue from the data market, when compared to the others. Conversely, agents that buy forecasts with higher accuracy pay higher values, but are compensated by the gain associated with the imbalance costs reduction. For instance, agent 1 pays 589.3€ but the extra gain from using these forecasts, instead of those obtained by its internal (or local) model, is 1107.9€.

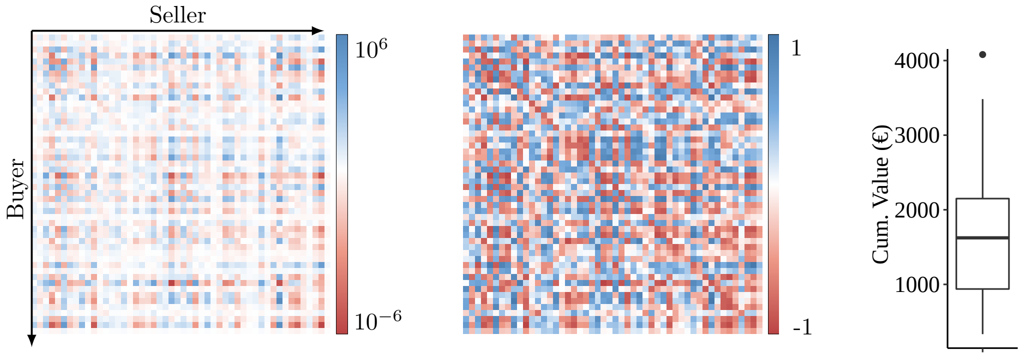
Figure 4.7 summarizes the covariance and correlation matrices for $\mathbf{D}_2$, as well as the total gain (boxplot) for the 50 agents. There is a large number of correlated time series. But once again, agents gain money by improving their forecasting accuracy or by selling their data to others. The lowest total gain is 333.4€ and more than 30 agents receive at least 1000€.

Table 4.1: Cumulative gains with $\mathbf{D}_1$ by agent (€).

	1	2	3	11	12	16	21	36	43	44	Others
Payment	589.3	30.0	0.0	0.1	88.2	58.4	22.7	0.0	101.4	0.0	[0,2[
Revenue*	0.6	570.8	26.6	98.7	48.9	2.3	0.3	19.1	0.5	90.0	[0,2[
Tot. Gain**	519.2	595.1	26.6	98.7	176.3	123.4	17.6	19.1	110.8	90.0	[0,4[

* Revenue = data market revenue (i.e., value received by selling data)

** Tot. Gain = data market revenue + revenue with purchased forecasts - value paid



(a) Covariance matrix.
(b) Pearson correlation.
(c) Total gains.

Figure 4.7: Covariance and correlation for data $\mathbf{D}_2$ and gain after 150 time steps.

4.4.3 Wind Power Data

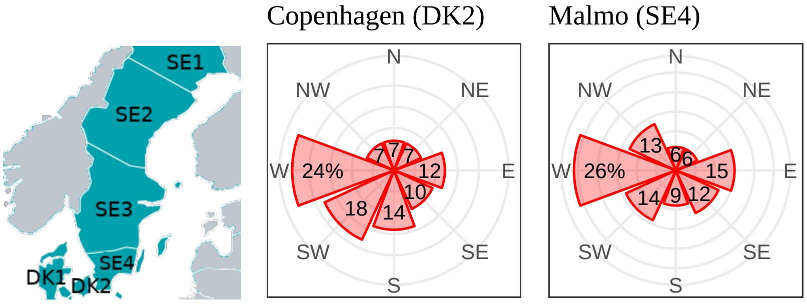


Figure 4.8: Nord Pool regions in Denmark (DK) and Sweden (SE), as well as the wind roses for the wind direction observed in Copenhagen and Malmo.

Data Description and Experiments

Nord Pool runs the largest market for electrical energy in Europe, operating in several northern Europe countries. For illustrative purposes, we use the historical wind power values, spot price and imbalance prices for upward and downward regulation, available in the Nord Pool website[1], from 6 regions: 4 in Sweden (SE1, SE2, SE3, SE4) and 2 in Denmark (DK1 and DK2). In this test case, each region is assumed to represent an electricity market agent. The dataset ranges between 1st January 2016 and 12th October 2017 with hourly resolution. Figure 4.8 provides a geographical representation of these regions as well as the wind roses for the wind direction observed in Copenhagen and Malmo during this period[2].

The agents are assumed to maximize their electricity market's revenue at time t by forecasting the optimal quantile $\tau_t^* = \frac{\hat{\lambda}_t^{\downarrow}}{\hat{\lambda}_t^{\downarrow} + \hat{\lambda}_t^{\uparrow}}$, as in Section 4.2.1. Lags 1, 2 and 3 are used as covariates in the linear quantile regression model provided by (4.11), motivated by preliminary cross-correlation analysis of the time series. The gain is computed by the improvement in the electricity market revenue, as defined in (4.12), which measures how much money an agent earns on the electricity market when using the forecast provided by the data market instead of the forecasts obtained through the use of local data (and model).

As in the previous case-study, hour-ahead power forecasts are generated and validated in the same way. The market operator uses a sliding window with one year divided in 8592 for model fitting and 168 (one week) to estimate the improvement in gain. The parameter η, used for forecasting upward and downward regulation unit costs, is estimated (i.e., select the value with minimum mean square error) by dividing the first one-year data in 9 months for training the Holt-Winters model and the remaining 3 months for computing the corresponding mean squared error, for $\eta \in \{0.9, 0.95, 0.99, 0.999\}$.

In this test case, the market operator is assumed to set a market price between 5% and 70% of the gain, with 5% increments, i.e., for each 1€ increase in electricity market revenue, the market operator may define a market price between 0.05€ and 0.70€. On

[1] `https://www.nordpoolgroup.com/` (accessed on November 2020)
[2] `https://www.weatheronline.co.uk` (accessed on November 2020)

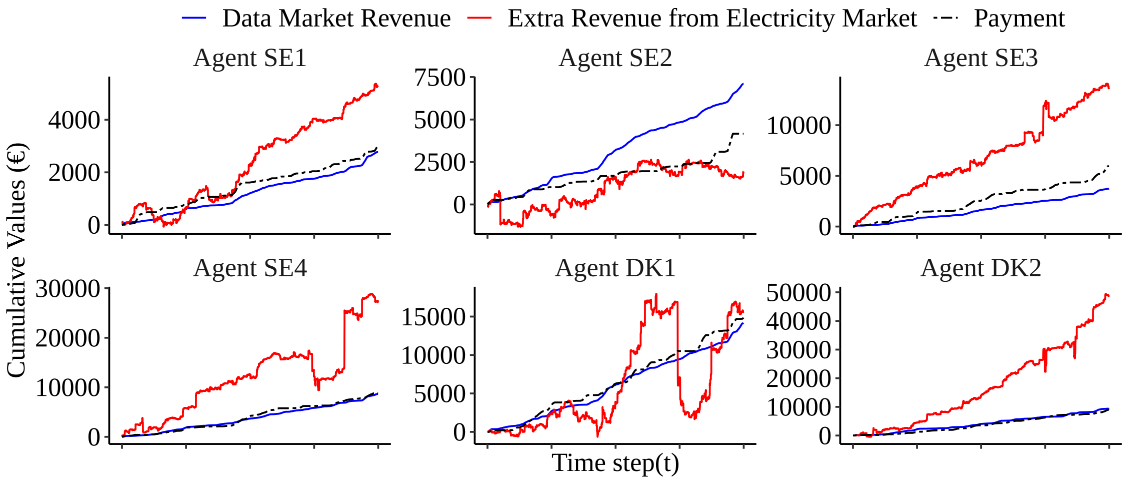


Figure 4.9: Cumulative values for electricity market revenue (over a quantile regression using only local data), data market revenue and payment, considering wind power data.

the other hand, the bid price is 50% for all buyers, i.e., the buyers are willing to pay a maximum of 0.50€ for each 1€ increase in electricity market revenue.

Results and Discussion

For each time step, the gain in electricity market revenue is computed as the difference between the revenue obtained when using forecasts from the data market and the revenue obtained by using a local forecasting model built without neighbor time series. Figure 4.9 depicts the cumulative revenue gain from the electricity market, i.e., the extra revenue obtained by using the forecast provided by the data market. Furthermore, the same plot shows the cumulative revenue from the data market, i.e., how much each agent receives by sharing data with the market operator, and, finally, the cumulative payment that each agent pays to the data market in order to buy forecasts. Table 4.2 supports the graphical analysis by presenting the cumulative gains and total revenue at the end of the testing period (approx. 10 months).

An agent participating in the data market may increase its revenue either by receiving more money from the electricity market (i.e., minimizing imbalance costs) or by receivsing money from the data market (i.e., selling data to competitors). The fundamental goal of the data market is to have a total revenue (i.e., sum of revenues obtained in the data and electricity market minus the payment to the data market) higher than the revenue obtained in the electricity market without third-party data or data monetization.

Agent DK2 benefits the most from the data market, followed by agent SE4. These benefits are mainly due to the increase in the revenue from the electricity market, i.e.,

Table 4.2: Cumulative gains (€) at the end of testing period.

(1st January 2017 to 12th October 2017)

	SE1	SE2	SE3	SE4	DK1	DK2
Electricity market	5303	1907	13668	27393	15609	48883
Paid value	3028	4166	5950	8898	14854	9018
Data market	2770	7105	3751	8688	14151	9449
Total revenue	5045	4846	11469	27184	14907	49315

Table 4.3: Payment division by the competitors (in %).

	SE1	SE2	SE3	SE4	DK1	DK2
SE1	—	29.11	10.70	20.86	35.69	3.64
SE2	19.45	—	12.52	24.51	29.79	13.73
SE3	10.21	25.96	—	15.33	39.80	8.70
SE4	10.62	27.01	9.54	—	42.55	10.28
DK1	1.52	9.60	10.73	28.78	—	49.38
DK2	2.01	9.42	5.13	20.50	62.94	—

from the improvement of the forecasting models. This is explained by the fact that wind comes predominately from the West (as depicted in Fig. 4.8), and their forecast models are improved by the time series from agent DK1 (located to the East).

On the other hand, the agent DK1 receives a higher reward for sharing its data with the market operator, which is also coherent with predominant wind direction. Southwest locations will be more relevant to improve forecasting models. Consequently, northwestern regions tend to benefit most from using forecasts with information from the other agents. The sudden decrease in accumulated gains (e.g., for agent DK1) occur due to extremely high values for regulation unit costs. For agent DK1, the high losses are associated with a upward regulation unit costs higher than 200€/MWh (when the values in 99% of the historical period are smaller than 30€/MWh).

Finally, Table 4.3 summarizes how the value paid by each agent is divided by the other agents (data sellers). By construction, the proportion that a data seller receives is related to the relevance (i.e., explanatory power) of its time series when forecasting the RES generation of a buyer. Agent DK1 receives a higher reward for sharing its data, which is due to its geographical location. Following the same reasoning, it would be expected that SE1 received a smaller proportion of money from all the competitors.

In order to assess the added value of a quantile regression with varying nominal proportions over time (τ_t) instead of a constant value τ, the mean values for $\lambda^{\uparrow}$ and $\lambda^{\downarrow}$ are computed for the testing period and the related nominal proportion is estimated. The value for the nominal proportion is 0.60. The results show that the revenue from the electricity market for agents SE1, SE2, SE3, SE4, DK1 and DK2 increases, respectively, 176,298€, 517,218€, 437,747€, 293,813€, 887,684€ and 344,883€ when using τ_t instead of τ.

4.4.4 Solar Power Data

Data Description and Experiments

The proposed algorithm is now evaluated using a solar power dataset. The power data consist of hourly time series of solar power from 44 micro-generation units, located in a Portuguese city, covering the period from February 1, 2011 to March 6, 2013. To make the data stationary, the solar power is normalized through a clear sky model, which gives an estimate of the solar power in clear sky conditions at any given time [37]. This clear-sky model estimates the clear-sky power time series exclusively from historical on-site power observations.

Since the prices for the electricity market during this period are not available, we illustrate the proposal by using the prices from the region DK1 of the Nord Pool dataset. Furthermore, we change the power units from Wh to kWh to make gains more salient.

Similarly to the previous experiment, we assume that agents select the quantile regres-

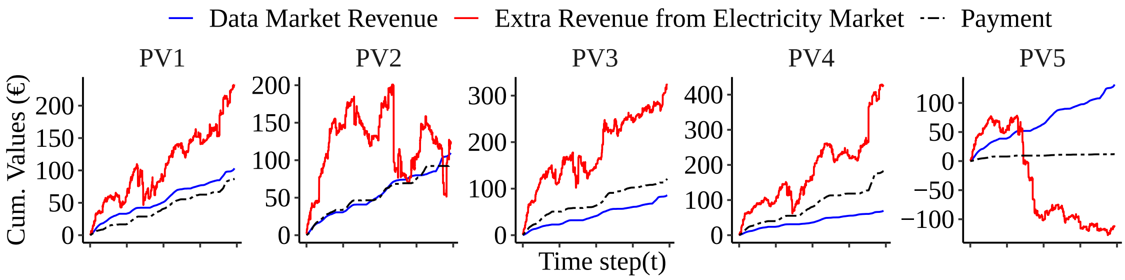


Figure 4.10: Cumulative values for electricity market revenue (over a local quantile regression), data market revenue and payment, considering solar power data.

sion model aiming to maximize their electricity market's revenue for the next hour, as described in Section 4.3.3. The lags 1, 2 and 24 of all agents are used as covariates, motivated by previous work with linear models [84]. The gain function is computed from the improvement in the electricity market revenue, as defined in (4.12). Also, to exclude the night-time hours, data corresponding to a solar zenith angle larger than 90 are removed during the training phase. For these hours, the improvement in electricity market revenue is considered zero. All agents are assumed to bid at a constant value of 50%, i.e., for each increase of €1 in electricity market revenue, the agents are willing to pay up to €0.50.

The market operator uses a sliding window with one year, the most recent week is used to estimate the improvement in gain, and the remaining for model fitting.

Results and Discussion

For each time step, the gain in electricity market revenue is computed as the difference between the revenue obtained when using forecasts from the data market and the revenue obtained by using a local forecasting model built without neighbor time series. A sample of 5 of these 44 agents is considered to illustrate the results. Figure 4.10 depicts the cumulative extra revenue obtained by using the forecast provided by the data market, the cumulative revenue from sharing data with the market operator, and the cumulative payment that each agent pays to the data market to buy forecasts.

In terms of extra revenue from the electricity market, the collaboration between agents seems to benefit agents PV1 to PV4, with PV4 being the most benefited. However, PV5 tends to lose money when considering the forecasts provided by the data market. This result was expected since in the previous chapter this same agent showed that collaboration through a vector autoregressive model does not improve accuracy when compared to an autoregressive model. In fact, PV5 should not consider the forecasts provided by the data market, since the corresponding payments are mostly zero, which means that the data market estimates no value from such forecasts.

Since PV4's data is relevant to some of its competitors, the data market places a positive value on its data, motivating its participation in collaborative forecasting. Therefore, all agents benefit from the higher revenue accrued either from the data market or the better forecast in the electricity market.

4.5 Concluding Remarks

Data sharing between different owners has a high potential to improve RES forecasting skill in different time horizons (e.g., hours-ahead, day-ahead) and consequently the revenue from electricity market players. However, economic incentives, trough data monetization, are fundamental to implement collaborative forecasting schemes since RES agents can be competitors, and therefore unwilling to share their confidential data without benefits. This work was inspired by [200] and adapted for RES forecasting. The gain function of buyers was adapted for RES agents, which have a local model with their own variables and enter the market to improve it with more information. Furthermore, an evaluation was performed using three case studies.

Synthetic data was used in a controlled case study where it was possible to confirm: (i) the correct allocation of revenue across sellers by the market operator, and (ii) the buyers who did not benefit from the forecasts of others did not pay for such forecasts. Data from the Nord Pool market and a small municipality in Portugal were used to evaluate the potential of a data market for RES agents, and it was concluded that: (i) all agents benefit (from the economic point of view) from the data market, (ii) agents that first observe wind-flow (or wind generation) in one location, e.g., at timestep $t-1$, provide relevant information to improve the forecasting model (e.g., for $t+1$) of neighbor agents in other locations, conditioned by wind direction, and then all agents benefit by the higher revenue accrued either from the data market or the better forecast in the electricity market. In summary, data markets can be a solution to foster data exchange between RES agents and contribute to reduce imbalance costs.

In this work, linear quantile regression and the Holt-Winters statistical models were used for the power and imbalance prices forecasts respectively. However, the choice of these models, considering aspects such as time horizon, non-linear relation between power and NWP, etc., must be carefully considered to deliver maximum gains in the electricity and data markets. For instance, the market operator can use a statistical model tailored to each RES agent, as long as the forecasting skill is maximized since it impacts the financial incentives to share data.

III Conclusion

This epilogue summarizes the main contributions and findings from this PhD thesis. The topics for future work are also identified.

III.1 Summary

Despite the many benefits of Renewable Energy Sources (RES), there are challenges to overcome since their generation depends on weather factors (wind speed, clouds, solar irradiance, etc.). Consequently, accurate forecasts are essential to reduce electrical energy imbalances in the electricity market and design advanced decision-aid tools to support the integration of large amounts of RES into the power system.

The following main contributions are provided by this PhD thesis, which had been previously discussed in Section I.2:

1. **Extreme quantile forecasting.** Forecast uncertainty is minimized by combining extreme value theory estimators for truncated generalized Pareto distribution with non-parametric methods, conditioned by spatio-temporal information. In this framework, covariates are used to produce conditional forecasts of quantiles without any limitation in the number of variables, and the parametric extreme value theory-based estimator can be combined with any non-parametric model (artificial neural networks, gradient boosting trees, random forests, etc.) without any major modification.

   The results for a synthetic dataset shows that the proposed approach better captures the overall tails' behavior, with smaller deviations between real and estimated quantiles. The proposed method also outperforms state-of-the-art methods in terms of quantile score when evaluated using real data from wind and solar power plants.

2. **Privacy-preserving collaborative models.** Cooperation between multiple RES power plant owners can lead to an improvement in forecast accuracy thanks to the spatio-temporal dependencies in time series data. Such cooperation between agents makes data privacy a necessity since they usually are competitors. The main contributions to this topic are:

   a) A numerical and mathematical analysis of the existing privacy-preserving regression models and identification of weaknesses in the current literature. Existing methods of data privacy are unsatisfactory when it comes to time series and can lead to confidentiality breaches – which means the reconstruction of the entire private dataset by another party.

      These techniques are grouped as (a) *data transformation*, such as the generation of random matrices that pre- or post-multiply the data or using principal component analysis with differential privacy, (b) *secure multi-party computation*, such as linear algebra protocols or homomorphic encryption (encrypting the original data in a way that arithmetic operations in the public space do not compromise the encryption), and (c) *decomposition-based methods* like the ADMM or the distributed Newton-Raphson method. The main conclusions were that *data transformation* requires a trade-off between privacy and accuracy, *secure multi-party computations* either result in computationally demanding techniques or do not fully preserve privacy in Vector AutoRegressive (VAR) models, and that *decomposition-based methods* rely on iterative processes and after a number of iterations, the agents have enough information to recover private data.

b) Based on the previous state-of-the-art analysis, a privacy-preserving forecasting algorithms is proposed. Data privacy is ensured by combining linear algebra transformations with a decomposition-based algorithm, allowing to compute the model's coefficients in a parallel fashion. This novel method also included an asynchronous distributed algorithm, making it possible to update the forecast model based on information from a subset of agents and improve the computational efficiency of the proposed model. The mathematical formulation is flexible enough to be applied in two different collaboration schemes (central hub model and peer-to-peer) and paved the way for learning models distributed by features, instead of observations.

   The results obtained for wind and solar energy datasets show that the privacy-preserving model delivers a forecast skill comparable to a model without privacy protection and outperformed a state-of-the-art method based on analog search.

3. **Algorithmic solution for data trading.** Incentives must also exist so that agents are motivated to cooperate by exchanging their data. The contribution for this topic is the development of an algorithmic solution for data monetization in RES collaborative forecasting. Cooperation between sellers is done through a market operator who receives all agents data and prepares forecasts: (i) sellers with similar information receive similar revenue, (ii) the market price is a function of the buyer's benefit, and so the buyer does not pay if there is no improvement in the forecasting skill, (iii) buyers pay according to incremental gain, and (iv) buyers purchase forecasts, instead of features, and have no knowledge about which datasets were used to produce these forecasts.

   Experiments have shown that all agents (or data owners) benefit in terms of higher revenue resulting from the combination of electricity and data markets. The results support the idea that data markets can be a viable solution to promote data exchange between RES agents and contribute to reducing system imbalance costs.

All in all, all four main chapters have an associated publication (one under review), in journals ranging in impact factors from 2.8 up to 7.4, as previously mentioned in Section I.4. These works have contributed to the Smart4RES project, a collaboration involving institutions from six countries that aims to improve efficiency from RES, and we are currently collaborating with other colleagues to advance the state-of-the-art, especially with regard to data markets, and the extension of the privacy-preserving analytics to other use cases in the energy sector (e.g., smart grids).

III.2 Future Work

The following topics were identified for future work:

1. **Extreme quantile forecasting.** Forecasting rare events remains a challenge given to the scarcity of data to represent them. Future research should consider:
   a) the inclusion of information from weather ensembles, as additional covariates, in order to exploit its capability to capture extreme events with a physically-based approach;
   b) the generalization of the proposed method to other energy-related time series, e.g., electricity market prices (energy, system services, etc.);
   c) the development of new proper scoring rules are needed to evaluate the forecasting skill of extreme (rare) events (see [212] for instance).
2. **Privacy-preserving collaborative models.** Privacy-preserving techniques are very sensitive to data partitioning and the problem structure. Future research should consider:
   a) Uncertainty forecasting and application to non-linear models (and consequently longer lead times), which we plan to investigate in a forthcoming work. Nevertheless, uncertainty forecast can be readily generated by transforming original data using a logit-normal distribution [174]. The proposed privacy-preserving protocol can be applied to non-linear regression by extending the additive model structure to a multivariate setting [213] or by local linear smoothing [214].
   b) The extension to other non-linear multivariate models recently considered in collaborative learning [215], such as long short-term memory networks and variants which can make use of NWP as input. These models would require changes in the protocol for data transformation. For example, the rectifier (ReLU), which is an activation function commonly used in neural networks and defined as $f(x) = \max(0, x)$, has the problem that $f(\mathbf{MZQB}) \neq \mathbf{M}f(\mathbf{ZQB})$.
3. **Algorithmic solution for data trading.** Topics for future work include:
   a) The loss of RES agents when sharing their data should be considered when defining the data price. Evidently, a seller sharing data with its competitors expects compensation for the potential impact on its business.
   b) Some improvements are required when using a sliding-window approach. The current version of the algorithm works by adding noise to the covariates, which means that, for each new time step, the market operator needs to perform a batch train that can result in a high computational effort as more and more agents enter the market. Ideally, the noise should be introduced in the output of the model, allowing the market operator to update the model weights through online learning whenever the variables in the data market remain the same.
   c) The privacy of the data should be addressed since in our simulations the agents share the data with the market operator, which may represent an obstacle for some agents. The privacy-preserving protocol proposed in this PhD thesis can be combined with the data markets concept in order to increase the privacy of the data market.
   d) The development of peer-to-peer data trading schemes (i.e., without a central node as market operator) for prosumers (producers and consumers of renewable

energy) in local energy communities, in such a way that data sellers can set their own data price.

e) extension of the data markets concept to other data sources, such as a network of weather stations or numerical weather predictions (i.e., monetization of weather forecasts).

Bibliography

[1] eurostat, *Renewable energy statistics*, 2020 (accessed November 15, 2020). [Online]. Available: https://ec.europa.eu/eurostat/statistics-explained/index.php/Renewable_energy_statistics

[2] EU, *2030 climate & energy framework*, Accessed in November 2020. [Online]. Available: https://ec.europa.eu/clima/policies/strategies/2030_en

[3] E. Commission, "Directive 2003/54/EC concerning common rules for the internal market in electricity," *Official Journal of the European Union*, vol. 176, pp. 37–56, 2003.

[4] ——, "Directive 2009/72/EC of the european parliament and of the council of 13 july 2009 concerning common rules for the internal market in electricity and repealing directive 2003/54/ec," *Official Journal of the European Union*, vol. 211, pp. 55–93, 2009.

[5] ——, "Directive (EU) 2019/944 of the european parliament and of the council of 5 june 2019 on common rules for the internal market for electricity and amending directive 2012/27/EU (text with EEA relevance.)," *Official Journal of the European Union*, vol. 158, p. 125–199, 2019.

[6] REN, *European Cross-Border Intraday Market XBID*, 2018 (accessed November 15, 2020). [Online]. Available: https://www.mercado.ren.pt/EN/Electr/InterProj/XBID/Pages/default.aspx

[7] M. A. Matos and R. J. Bessa, "Setting the operating reserve using probabilistic wind power forecasts," *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 594–603, 2010.

[8] M. Matos, R. J. Bessa, C. Gonçalves, L. Cavalcante, V. Miranda, N. Machado, P. Marques, and F. Matos, "Setting the maximum import net transfer capacity under extreme res integration scenarios," in *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*. IEEE, 2016, pp. 1–7.

[9] J. Tastu, P. Pinson, and H. Madsen, "Space-time trajectories of wind power generation: Parametrized precision matrices under a gaussian copula approach," in *Modeling and stochastic learning for forecasting in high dimensions.* Springer, 2015, pp. 267–296.

[10] R. Dupin, L. Cavalcante, R. J. Bessa, G. Kariniotakis, and A. Michiorri, "Extreme quantiles dynamic line rating forecasts and application on network operation," *Energies*, vol. 13, no. 12, p. 3090, 2020.

[11] T. Gneiting, K. Larson, K. Westrick, M. Genton, and E. Aldrich, "Calibrated probabilistic forecasting at the stateline wind energy center: The regime-switching space-time method," *Journal of the American Statistical Association*, vol. 101, pp. 968–979, 2006.

[12] J. R. Andrade and R. J. Bessa, “Improving renewable energy forecasting with a grid of numerical weather predictions,” *IEEE Transactions on Sustainable Energy*, vol. 8, no. 4, pp. 1571–1580, 2017.

[13] J. Tastu, P. Pinson, and H. Madsen, “Multivariate conditional parametric models for a spatio-temporal analysis of short-term wind power forecast errors,” *Proceedings of the European Wind Energy Conference (EWEC 2010)*, 2010.

[14] A. M. Jones, J. Lomas, and N. Rice, “Healthcare cost regressions: going beyond the mean to estimate the full distribution,” *Health economics*, vol. 24, no. 9, pp. 1192–1212, 2015.

[15] H. W. Ahmad, S. Zilles, H. J. Hamilton, and R. Dosselmann, “Prediction of retail prices of products using local competitors,” *International Journal of Business Intelligence and Data Mining*, vol. 11, no. 1, pp. 19–30, 2016.

[16] Y. Aviv, “A time-series framework for supply-chain inventory management,” *Operational Research*, vol. 51, no. 2, pp. 175–342, Mar. 2003.

[17] ——, “On the benefits of collaborative forecasting partnerships between retailers and manufacturers,” *Management Science*, vol. 53, no. 5, pp. 777–794, May 2007.

[18] J. R. Trapero, M. Cardós, and N. Kourentzes, “Quantile forecast optimal combination to enhance safety stock estimation,” *International Journal of Forecasting*, vol. 35, no. 1, pp. 239–250, 2019.

[19] R. Koenker, *quantreg: Quantile Regression*, 2018, R package version 5.38. [Online]. Available: https://CRAN.R-project.org/package=quantreg

[20] W. Graybill, M. Chen, V. Chernozhukov, I. Fernandez-Val, and A. Galichon, *Rearrangement: Monotonize Point and Interval Functional Estimates by Rearrangement*, 2016, R package version 2.1. [Online]. Available: https://CRAN.R-project.org/package=Rearrangement

[21] T. Reynkens and R. Verbelen, *ReIns: Functions from "Reinsurance: Actuarial and Statistical Aspects"*, 2018, R package version 1.0.8. [Online]. Available: https://CRAN.R-project.org/package=ReIns

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, and et al., “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[23] R. Andrade and R. J. Bessa, “Solar power forecasting: measurements and numerical weather predictions,” Apr. 2020. [Online]. Available: https://doi.org/10.25747/edf8-m258

[24] C. Gonçalves and R. J. Bessa, “Geographically distributed solar power time series,” Sep. 2020. [Online]. Available: https://doi.org/10.25747/gywm-9457

[25] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R. J. Hyndman, “Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond,” *International Journal of Forecasting*, vol. 32, pp. 896–913, 2016.

[26] N. Pool, *Nord Pool data*, Accessed in November 2020. [Online]. Available: https://www.nordpoolgroup.com/

[27] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2018. [Online]. Available: https://www.R-project.org/

[28] G. Van Rossum and F. L. Drake Jr, *Python tutorial.* Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995.

[29] C. Sweeney, R. J. Bessa, J. Browell, and P. Pinson, "The future of forecasting for renewable energy," *Wiley Interdisciplinary Reviews: Energy and Environment*, vol. 9, no. 2, p. e365, Mar. 2020.

[30] R. Ahmed, V. Sreeram, Y. Mishra, and M. Arif, "A review and evaluation of the state-of-the-art in pv solar power forecasting: Techniques and optimization," *Renewable and Sustainable Energy Reviews*, vol. 124, p. 109792, 2020.

[31] T. Ahmad, H. Zhang, and B. Yan, "A review on renewable energy and electricity requirement forecasting models for smart grid and buildings," *Sustainable Cities and Society*, vol. 55, p. 102052, 2020.

[32] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, "Current methods and advances in forecasting of wind power generation," *Renewable Energy*, vol. 37, no. 1, pp. 1–8, 2012.

[33] C. Monteiro, R. Bessa, V. Miranda, A. Botterud, J. Wang, G. Conzelmann *et al.*, "Wind power forecasting: State-of-the-art 2009." Argonne National Lab.(ANL), Argonne, IL (United States), Tech. Rep., 2009.

[34] F. Antonanzas-Torres, R. Urraca, J. Polo, O. Perpiñán-Lamigueiro, and R. Escobar, "Clear sky solar irradiance models: A review of seventy models," *Renewable and Sustainable Energy Reviews*, vol. 107, pp. 374–387, 2019.

[35] S. H. Jangamshetti and V. G. Rau, "Normalized power curves as a tool for identification of optimum wind turbine generator parameters," *IEEE Transactions on Energy Conversion*, vol. 16, no. 3, pp. 283–288, 2001.

[36] M. J. Duran, D. Cros, and J. Riquelme, "Short-term wind power forecast based on arx models," *Journal of Energy Engineering*, vol. 133, no. 3, pp. 172–180, 2007.

[37] P. Bacher, H. Madsen, and H. A. Nielsen, "Online short-term solar power forecasting," *Solar Energy*, vol. 83, no. 10, pp. 1772–1783, Oct. 2009.

[38] P. Pinson, L. Christensen, H. Madsen, P. E. Sørensen, M. H. Donovan, and L. E. Jensen, "Regime-switching modelling of the fluctuations of offshore wind generation," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 96, no. 12, pp. 2327–2347, 2008.

[39] A. Mellit, A. Massi Pavan, E. Ogliari, S. Leva, and V. Lughi, "Advanced methods for photovoltaic output power forecasting: A review," *Applied Sciences*, vol. 10, no. 2, p. 487, 2020.

[40] J. Shi, W.-J. Lee, Y. Liu, Y. Yang, and P. Wang, "Forecasting power output of photovoltaic systems based on weather classification and support vector machines," *IEEE Transactions on Industry Applications*, vol. 48, no. 3, pp. 1064–1069, 2012.

[41] R. M. Ehsan, S. P. Simon, and P. Venkateswaran, “Day-ahead forecasting of solar photovoltaic output power using multilayer perceptron,” *Neural Computing and Applications*, vol. 28, no. 12, pp. 3981–3992, 2017.

[42] H. Lu and G. Chang, “Wind power forecast by using improved radial basis function neural network,” in *2018 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2018, pp. 1–5.

[43] Y. Kassa, J. Zhang, D. Zheng, and D. Wei, “Short term wind power prediction using anfis,” in *2016 IEEE international conference on power and renewable energy (ICPRE)*. IEEE, 2016, pp. 388–393.

[44] U. Cali and V. Sharma, “Short-term wind power forecasting using long-short term memory based recurrent neural network model and variable selection,” *International Journal of Smart Grid and Clean Energy*, vol. 8, no. 2, pp. 103–110, 2019.

[45] Shivani, K. S. Sandhu, and A. Ramachandran Nair, “A comparative study of arima and rnn for short term wind speed forecasting,” in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2019, pp. 1–7.

[46] Y. Ren, P. Suganthan, and N. Srikanth, “Ensemble methods for wind and solar power forecasting–a state-of-the-art review,” *Renewable and Sustainable Energy Reviews*, vol. 50, pp. 82–91, 2015.

[47] A. Lahouar and J. B. H. Slama, “Hour-ahead wind power forecast based on random forests,” *Renewable energy*, vol. 109, pp. 529–541, 2017.

[48] A. Chaouachi, R. M. Kamel, R. Ichikawa, H. Hayashi, K. Nagasaka *et al.*, “Neural network ensemble-based solar power generation short-term forecasting,” *World Academy of Science, Engineering and Technology*, vol. 54, pp. 54–59, 2009.

[49] F. Thordarson, H. Madsen, H. A. Nielsen, and P. Pinson, “Conditional weighted combination of wind power forecasts,” *Wind Energy*, vol. 13, no. 8, pp. 751–763, 2010.

[50] J. Browell, C. Gilbert, and D. McMillan, “Use of turbine-level data for improved wind power forecasting,” in *2017 IEEE Manchester PowerTech*. IEEE, 2017, pp. 1–6.

[51] I. P. Panapakidis and G. C. Christoforidis, “A hybrid ann/ga/anfis model for very short-term pv power forecasting,” in *2017 11th IEEE International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)*. IEEE, 2017, pp. 412–417.

[52] K. Bellinguer, V. Mahler, S. Camal, and G. Kariniotakis, “Probabilistic forecasting of regional wind power generation for the eem20 competition: a physics-oriented machine learning approach,” in *2020 17th International Conference on the European Energy Market (EEM)*. IEEE, 2020, pp. 1–6.

[53] W. Lijie, D. Lei, G. Shuang, and L. Xiaozhong, “Short-term wind power prediction with signal decomposition,” in *2011 International Conference on Electric Information and Control Engineering*. IEEE, 2011, pp. 2569–2573.

[54] J. Naik, P. Satapathy, and P. Dash, "Short-term wind speed and wind power prediction using hybrid empirical mode decomposition and kernel ridge regression," *Applied Soft Computing*, vol. 70, pp. 1167–1188, 2018.

[55] W. Zhang, F. Liu, X. Zheng, and Y. Li, "A hybrid emd-svm based short-term wind power forecasting model," in *2015 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*. IEEE, 2015, pp. 1–5.

[56] M. Kuzlu, U. Cali, V. Sharma, and Ö. Güler, "Gaining insight into solar photovoltaic power generation forecasting utilizing explainable artificial intelligence tools," *IEEE Access*, vol. 8, pp. 187 814–187 823, 2020.

[57] J. Shi, J. Guo, and S. Zheng, "Evaluation of hybrid forecasting approaches for wind speed and power generation time series," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 5, pp. 3471–3480, 2012.

[58] A. Dolara, F. Grimaccia, S. Leva, M. Mussetta, and E. Ogliari, "A physical hybrid artificial neural network for short term forecasting of pv plant power output," *Energies*, vol. 8, no. 2, pp. 1138–1153, 2015.

[59] G. Sideratos and N. Hatziargyriou, "Using radial basis neural networks to estimate wind power production," in *2007 IEEE Power Engineering Society General Meeting*. IEEE, 2007, pp. 1–7.

[60] F. Molteni, R. Buizza, T. N. Palmer, and T. Petroliagis, "The ecmwf ensemble prediction system: Methodology and validation," *Quarterly journal of the royal meteorological society*, vol. 122, no. 529, pp. 73–119, 1996.

[61] S. Lerch, S. Baran, A. Möller, J. Groß, R. Schefzik, S. Hemri, and M. Graeter, "Simulation-based comparison of multivariate ensemble post-processing methods," *Nonlinear Processes in Geophysics*, vol. 27, no. 2, pp. 349–371, 2020.

[62] R. Juban, H. Ohlsson, M. Maasoumy, L. Poirier, and J. Z. Kolter, "A multiple quantile regression approach to the wind, solar, and price tracks of GEFCom2014," *International Journal of Forecasting*, vol. 32, no. 3, pp. 1094–1102, 2016.

[63] J. B. Bremnes, "Probabilistic wind power forecasts using local quantile regression," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, vol. 7, no. 1, pp. 47–54, 2004.

[64] S. B. Taieb, R. Huser, R. J. Hyndman, and M. G. Genton, "Forecasting uncertainty in electricity smart meter data by boosting additive quantile regression," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2448–2455, 2016.

[65] Y. Yu, X. Han, M. Yang, and Y. Zhang, "A regional wind power probabilistic forecast method based on deep quantile regression," in *2020 IEEE/IAS 56th Industrial and Commercial Power Systems Technical Conference (I&CPS)*. IEEE, 2020, pp. 1–8.

[66] R. Bessa, V. Miranda, A. Botterud, J. Wang, and E. M. Constantinescu, "Time adaptive conditional kernel density estimation for wind power forecasting," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 4, pp. 660–669, 2012.

[67] P. Pinson, "Very short-term probabilistic forecasting of wind power with generalized logit-normal distributions," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 61, no. 4, pp. 555–576, 2012.

[68] J. W. Messner, A. Zeileis, J. Broecker, and G. J. Mayr, "Probabilistic wind power forecasts with an inverse power curve transformation and censored regression," *Wind Energy*, vol. 17, no. 11, pp. 1753–1766, 2013.

[69] H. J. Wang, D. Li, and X. He, "Estimation of high conditional quantiles for heavy-tailed distributions," *Journal of the American Statistical Association*, vol. 107, no. 500, pp. 1453–1464, 2012.

[70] P. Pinson, H. Madsen, H. A. Nielsen, G. Papaefthymiou, and B. Klöckl, "From probabilistic forecasts to statistical scenarios of short-term wind power production," *Wind Energy*, vol. 12, no. 1, pp. 51–62, 2009.

[71] Z. Wang, W. Wang, C. Liu, Z. Wang, and Y. Hou, "Probabilistic forecast for multiple wind farms based on regular vine copulas," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 578–589, 2017.

[72] Y. Chen, Y. Wang, D. Kirschen, and B. Zhang, "Model-free renewable scenario generation using generative adversarial networks," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 3265–3275, 2018.

[73] F. Golestaneh, P. Pinson, R. Azizipanah-Abarghooee, and H. B. Gooi, "Ellipsoidal prediction regions for multivariate uncertainty characterization," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 4519–4530, 2018.

[74] F. Golestaneh, P. Pinson, and H. B. Gooi, "Polyhedral predictive regions for power system applications," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 693–704, 2018.

[75] H. Quan, D. Srinivasan, and A. Khosravi, "Short-term load and wind power forecasting using neural network-based prediction intervals," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 2, pp. 303–315, 2013.

[76] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Optimal prediction intervals of wind power generation," *IEEE Transactions on Power Systems*, vol. 29, no. 3, pp. 1166–1174, 2013.

[77] R. J. Bessa, "From marginal to simultaneous prediction intervals of wind power," in *2015 18th International Conference on Intelligent System Application to Power Systems (ISAP)*. IEEE, 2015, pp. 1–6.

[78] J. Tastu, P. Pinson, P.-J. Trombe, and H. Madsen, "Probabilistic forecasts of wind power generation accounting for geographically dispersed information," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 480–489, 2014.

[79] M. He, V. Vittal, and J. Zhang, "A sparsified vector autoregressive model for short-term wind farm power forecasting," *Proceedings of the 2015 IEEE Power & Energy Society General Meeting*, 2015.

[80] J. Dowell and P. Pinson, "Very-short-term probabilistic wind power forecasts by sparse vector autoregression," *IEEE Transactions on Smart Grid 2015*, 2015.

[81] L. Cavalcante, R. J. Bessa, M. Reis, and J. Browell, "LASSO vector autoregression structures for very short-term wind power forecasting," *Wind Energy*, vol. 20, no. 4, pp. 657–675, Apr. 2017.

[82] P. Kou, F. Gao, and X. Guan, "Sparse online warped gaussian process for wind power probabilistic forecasting," *Applied Energy*, vol. 108, no. C, pp. 410–428, 2013.

[83] A. Vaz, B. Elsinga, W. Sark, and M. Brito, "An artificial neural network to assess the impact of neighbouring photovoltaic systems in power forecasting in utrecht, the netherlands," *Renewable Energy*, vol. 85, pp. 631–641, 2016.

[84] R. J. Bessa, A. Trindade, and V. Miranda, "Spatial-temporal solar power forecasting for smart grids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 232–241, 2015.

[85] D. Díaz, A. Torres, and J. Dorronsoro, "Deep neural networks for wind energy prediction," in *International Work-Conference on Artificial Neural Networks.* Springer, 2015, pp. 430–443.

[86] C. Gilbert, J. Browell, and D. McMillan, "Leveraging turbine-level data for improved probabilistic wind power forecasting," *IEEE Transactions on Sustainable Energy*, vol. 11, no. 3, pp. 1152–1160, Jul. 2020.

[87] K. Higashiyama, Y. Fujimoto, and Y. Hayashi, "Feature extraction of nwp data for wind power forecasting using 3d-convolutional neural networks," *Energy Procedia*, vol. 155, pp. 350–358, 2018.

[88] Q. Zhu, J. Chen, D. Shi, L. Zhu, X. Bai, X. Duan, and Y. Liu, "Learning temporal and spatial correlations jointly: A unified framework for wind speed prediction," *IEEE Transactions on Sustainable Energy*, vol. 11, no. 1, pp. 509–523, Jan. 2020.

[89] M. Wytock and J. Kolter, "Large-scale probabilistic forecasting in energy systems using sparse gaussian conditional random fields," *International Conference on Machine Learning (ICML 2013). JLMR Workshop and Conference Proceedings*, vol. 28, 2013.

[90] J. Tastu, P. Pinson, and H. Madsen, "Space-time scenarios of wind power generation produced using a gaussian copula with parametrized precision matrix," Technical University of Denmark, Tech. Rep., 2013.

[91] J. W. Messner, P. Pinson, J. Browell, M. B. Bjerregård, and I. Schicker, "Evaluation of wind power forecasts—an up-to-date view," *Wind Energy*, vol. 23, no. 6, pp. 1461–1481, 2020.

[92] F. X. Diebold and R. S. Mariano, "Comparing predictive accuracy," *Journal of Business & economic statistics*, vol. 20, no. 1, pp. 134–144, 2002.

[93] T. Gneiting, F. Balabdaoui, and A. E. Raftery, "Probabilistic forecasts, calibration and sharpness," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 69, no. 2, pp. 243–268, 2007.

[94] P. Friederichs and T. L. Thorarinsdottir, "Forecast verification for extreme value distributions with an application to probabilistic peak wind prediction," *Environmetrics*, vol. 23, no. 7, pp. 579–594, 2012.

[95] M. Taillardat, A.-L. Fougères, P. Naveau, and R. de Fondeville, "Extreme events evaluation using CRPS distributions," *arXiv preprint arXiv:1905.04022*, 2019.

[96] R. Koenker and G. Bassett Jr, “Regression quantiles,” *Econometrica: journal of the Econometric Society*, pp. 33–50, 1978.

[97] V. Chernozhukov, I. Fernández-Val, and A. Galichon, “Quantile and probability curves without crossing,” *Econometrica*, vol. 78, no. 3, pp. 1093–1125, 2010.

[98] P. D. Andersen, “Optimal trading strategies for a wind-storage power system under market conditions,” Master’s thesis, Technical University of Denmark, Lyngby, Denmark, 2009.

[99] H. J. Wang and D. Li, “Estimation of extreme conditional quantiles through power transformation,” *Journal of the American Statistical Association*, vol. 108, no. 503, pp. 1062–1074, 2013.

[100] L. De Haan and A. Ferreira, *Extreme value theory: an introduction.* Springer Science & Business Media, 2007.

[101] J. Beirlant, I. F. Alves, T. Reynkens *et al.*, “Fitting tails affected by truncation,” *Electronic Journal of Statistics*, vol. 11, no. 1, pp. 2026–2065, 2017.

[102] A. J. McNeil and T. Saladin, “The peaks over thresholds method for estimating high quantiles of loss distributions,” in *Proceedings of 28th International ASTIN Colloquium*, 1997, pp. 23–43.

[103] W. B. Nicholson, D. S. Matteson, and J. Bien, “VARX-L: Structured regularization for large vector autoregressions with exogenous variables,” *International Journal of Forecasting*, vol. 33, no. 3, pp. 627–651, 2017.

[104] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[105] W. Dai, S. Wang, H. Xiong, and X. Jiang, “Privacy preserving federated big data analysis,” in *Guide to Big Data Applications.* Springer, 2018, pp. 49–82.

[106] W. A. Fuller, *Introduction to statistical time series.* John Wiley & Sons, 2009, vol. 428.

[107] R. J. Bessa, C. Möhrlen, V. Fundel, M. Siefert, J. Browell, S. H. E. Gaidi, B.-M. Hodge, U. Cali, and G. Kariniotakis, “Towards improved understanding of the applicability of uncertainty forecasts in the electric power industry,” *Energies*, vol. 10, no. 9, p. 1402, Sep. 2017.

[108] A. Botterud, J. Wang, Z. Zhou, R. Bessa, H. Keko, J. Akilimali, and V. Miranda, “Wind power trading under uncertainty in LMP markets,” *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 894–903, 2012.

[109] R. Dupin, “Prévision du dynamic line rating et impact sur la gestion du systéme électrique,” Ph.D. dissertation, MINES ParisTech, PSL Research University, Paris, France, Jul. 2018.

[110] M. Cagnolari, “The value of the right distribution for the newsvendor problem and a bike-sharing problem,” Ph.D. dissertation, University of Bergamo, May 2017.

[111] J. Beirlant, T. D. Wet, and Y. Goegebeur, "Nonparametric estimation of extreme conditional quantiles," *Journal of statistical computation and simulation*, vol. 74, no. 8, pp. 567–580, 2004.

[112] F. Nogueira, "Python bayesian optimization implementation," http://github.com/fmfn/BayesianOptimization, 2020.

[113] F. Ziel and R. Weron, "Day-ahead electricity price forecasting with high-dimensional structures: Univariate vs. multivariate modeling frameworks," *Energy Economics*, vol. 70, pp. 396–420, 2018.

[114] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, p. 12, 2019.

[115] S. Ravi and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models," *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 53–72, 2009.

[116] Y. Li and M. G. Genton, "Single-index additive vector autoregressive time series models," *Scandinavian Journal of Statistic*, vol. 36, no. 3, pp. 369–388, Sep. 2009.

[117] W. Du, Y. S. Han, and S. Chen, "Privacy-preserving multivariate statistical analysis: Linear regression and classification," in *SIAM International Conference on Data Mining (SDM)*. SIAM, 2004, pp. 222–233.

[118] A. F. Karr, X. Lin, A. P. Sanil, and J. P. Reiter, "Privacy-preserving analysis of vertically partitioned data using secure matrix products," *Journal of Official Statistics*, vol. 25, no. 1, p. 125, 2009.

[119] Y. Wu, X. Jiang, J. Kim, and L. Ohno-Machado, "Grid binary LOgistic REgression (GLORE): building shared models without sharing data," *Journal of the American Medical Informatics Association*, vol. 19, no. 5, pp. 758–764, 2012.

[120] C.-L. Lu, S. Wang, Z. Ji, Y. Wu, L. Xiong, X. Jiang, and L. Ohno-Machado, "WebDISCO: a web service for distributed Cox model learning without patient-level data sharing," *Journal of the American Medical Informatics Association*, vol. 22, no. 6, pp. 1212–1219, 2015.

[121] W. Jia, H. Zhu, Z. Cao, X. Dong, and C. Xiao, "Human-factor-aware privacy-preserving aggregation in smart grid," *IEEE Systems Journal*, vol. 8, no. 2, pp. 598–607, Jun. 2014.

[122] S. E. Fienberg, Y. Nardi, and A. B. Slavković, "Valid statistical analysis for logistic regression with multiple sources," in *Protecting persons while protecting the people*. Springer, 2009, pp. 82–94.

[123] P. Pinson, "Introducing distributed learning approaches in wind power forecasting," in *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*. IEEE, 2016, pp. 1–6.

[124] A. Agarwal, M. Dahleh, and T. Sarkar, "A marketplace for data: An algorithmic solution," in *Proceedings of the 2019 ACM Conference on Economics and Computation*, 2019, pp. 701–726.

[125] A. Kurtulmus and K. Daniel, “Trustless machine learning contracts; evaluating and exchanging machine learning models on the ethereum blockchain,” *arXiv:1802.10185*, pp. 1–11, 2018.

[126] C. Dwork and A. Smith, “Differential privacy for statistics: What we know and what we want to learn,” *Journal of Privacy and Confidentiality*, vol. 1, no. 2, 2010.

[127] S. Zhou, J. Lafferty, and L. Wasserman, “Compressed and privacy-sensitive sparse regression,” *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 846–866, 2009.

[128] X. Ma, Y. Zhu, and X. Li, “An efficient and secure ridge regression outsourcing scheme in wearable devices,” *Computers & Electrical Engineering*, vol. 63, pp. 246–256, 2017.

[129] Y. K. Jain and S. K. Bhandare, “Min max normalization based data perturbation method for privacy protection,” *International Journal of Computer & Communication Technology*, vol. 2, no. 8, pp. 45–50, 2011.

[130] S. Papadimitriou, F. Li, G. Kollios, and P. S. Yu, “Time series compressibility and privacy,” in *Proceedings of the 33rd international conference on Very large data bases.* VLDB Endowment, 2007, pp. 459–470.

[131] O. L. Mangasarian, “Privacy-preserving horizontally partitioned linear programs,” *Optimization Letters*, vol. 6, no. 3, pp. 431–436, 2012.

[132] S. Yu, G. Fung, R. Rosales, S. Krishnan, R. B. Rao, C. Dehing-Oberije, and P. Lambin, “Privacy-preserving cox regression for survival analysis,” in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2008, pp. 1034–1042.

[133] K. Liu, C. Giannella, and H. Kargupta, “A survey of attack techniques on privacy-preserving data perturbation methods,” in *Privacy-Preserving Data Mining.* Springer, 2008, pp. 359–381.

[134] O. L. Mangasarian, “Privacy-preserving linear programming,” *Optimization Letters*, vol. 5, no. 1, pp. 165–172, 2011.

[135] C. Dwork, K. Talwar, A. Thakurta, and L. Zhang, “Analyze gauss: optimal bounds for privacy-preserving principal component analysis,” in *Proceedings of the forty-sixth annual ACM symposium on Theory of computing.* ACM, 2014, pp. 11–20.

[136] J. Zhu, P. He, Z. Zheng, and M. R. Lyu, “A privacy-preserving qos prediction framework for web service recommendation,” in *2015 IEEE International Conference on Web Services.* IEEE, 2015, pp. 241–248.

[137] L. Fan and L. Xiong, “An adaptive approach to real-time aggregate monitoring with differential privacy,” *IEEE Transactions on knowledge and data engineering*, vol. 26, no. 9, pp. 2094–2106, 2014.

[138] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and D. Megías, “Individual differential privacy: A utility-preserving formulation of differential privacy guarantees,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 6, pp. 1418–1429, 2017.

[139] R. Hall, S. E. Fienberg, and Y. Nardi, “Secure multiple linear regression based on homomorphic encryption,” *Journal of Official Statistics*, vol. 27, no. 4, p. 669, 2011.

[140] V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft, “Privacy-preserving ridge regression on hundreds of millions of records,” in *2013 IEEE Symposium on Security and Privacy*. IEEE, 2013, pp. 334–348.

[141] Y.-R. Chen, A. Rezapour, and W.-G. Tzeng, “Privacy-preserving ridge regression on distributed data,” *Information Sciences*, vol. 451, pp. 34–49, 2018.

[142] A. Gascón, P. Schoppmann, B. Balle, M. Raykova, J. Doerner, S. Zahur, and D. Evans, “Privacy-preserving distributed linear regression on high-dimensional data,” *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 345–364, 2017.

[143] Q. Jia, L. Guo, Z. Jin, and Y. Fang, “Preserving model privacy for machine learning in distributed systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 8, pp. 1808–1822, 2018.

[144] Q. Li and G. Cao, “Efficient and privacy-preserving data aggregation in mobile sensing,” in *2012 20th IEEE International Conference on Network Protocols (ICNP)*. IEEE, 2012, pp. 1–10.

[145] Y. Liu, W. Guo, C.-I. Fan, L. Chang, and C. Cheng, “A practical privacy-preserving data aggregation (3PDA) scheme for smart grid,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1767–1774, 2018.

[146] S. Li, K. Xue, Q. Yang, and P. Hong, “PPMA: privacy-preserving multisubset data aggregation in smart grid,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 462–471, 2018.

[147] S. Hoogh, “Design of large scale applications of secure multiparty computation: secure linear programming,” Ph.D. dissertation, Technische Universiteit Eindhoven, 2012.

[148] Y. Aono, T. Hayashi, L. T. Phong, and L. Wang, “Input and output privacy-preserving linear regression,” *IEICE TRANSACTIONS on Information and Systems*, vol. 100, no. 10, pp. 2339–2347, 2017.

[149] C. Zhao, S. Zhao, M. Zhao, Z. Chen, C.-Z. Gao, H. Li, and Y.-a. Tan, “Secure multi-party computation: Theory, practice and applications,” *Information Sciences*, vol. 476, pp. 357–372, 2019.

[150] H.-Y. Tran and J. Hu, “Privacy-preserving big data analytics a comprehensive survey,” *Journal of Parallel and Distributed Computing*, vol. 134, pp. 207–218, 2019.

[151] R. J. Bessa, D. Rua, C. Abreu, P. Machado, J. R. Andrade, R. Pinto, C. Gonçalves, and M. Reis, “Data economy for prosumers in a smart grid ecosystem,” in *Proceedings of the Ninth International Conference on Future Energy Systems*. ACM, 2018, pp. 622–630.

[152] G. Mateos, J. A. Bazerque, and G. B. Giannakis, “Distributed sparse linear regression,” *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5262–5276, 2010.

[153] T. Zhang and Q. Zhu, "Dynamic differential privacy for ADMM-based distributed classification learning," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 172–187, 2017.

[154] Z. Huang, R. Hu, Y. Guo, E. Chan-Tin, and Y. Gong, "DP-ADMM: ADMM-based distributed learning with differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1002–1012, 2019.

[155] X. Zhang, M. M. Khalili, and M. Liu, "Recycled ADMM: Improve privacy and accuracy with less computation in distributed algorithms," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2018, pp. 959–965.

[156] C. Zhang, M. Ahmad, and Y. Wang, "ADMM based privacy-preserving decentralized optimization," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 565–580, 2019.

[157] X. Huo and M. Liu, "A novel cryptography-based privacy-preserving decentralized optimization paradigm," 2020.

[158] Y. Zhang and J. Wang, "A distributed approach for wind power probabilistic forecasting considering spatio-temporal correlation without direct access to off-site information," *IEEE Transactions on Power Systems*, vol. 33, no. 5, pp. 5714–5726, 2018.

[159] J. Nocedal and S. Wright, *Numerical optimization.* Springer Science & Business Media, 2006.

[160] A. B. Slavkovic, Y. Nardi, and M. M. Tibbits, "Secure logistic regression of horizontally and vertically partitioned distributed databases," in *icdmw*. IEEE, 2007, pp. 723–728.

[161] Y. Li, X. Jiang, S. Wang, H. Xiong, and L. Ohno-Machado, "Vertical grid logistic regression (vertigo)," *Journal of the American Medical Informatics Association*, vol. 23, no. 3, pp. 570–579, 2015.

[162] Y. Nesterov, "Introductory lectures on convex programming volume i: Basic course," *Lecture notes*, vol. 3, no. 4, p. 5, 1998.

[163] S. Han, W. K. Ng, L. Wan, and V. C. Lee, "Privacy-preserving gradient-descent methods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 6, pp. 884–899, 2010.

[164] Q. Wei, Q. Li, Z. Zhou, Z. Ge, and Y. Zhang, "Privacy-preserving two-parties logistic regression on vertically partitioned data using asynchronous gradient sharing," *Peer-to-Peer Networking and Applications*, pp. 1–9, 2020.

[165] S. Song, K. Chaudhuri, and A. D. Sarwate, "Stochastic gradient descent with differentially private updates," in *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*. IEEE, 2013, pp. 245–248.

[166] X. Zhang, F. Fang, and J. Wang, "Probabilistic solar irradiation forecasting based on variational bayesian inference with secure federated learning," *IEEE Transactions on Industrial Informatics*, 2020.

[167] H. Y. Toda and P. C. Phillips, "Vector autoregressions and causality," *Econometrica: Journal of the Econometric Society*, pp. 1367–1393, 1993.

[168] C. F. Ansley and R. Kohn, "A note on reparameterizing a vector autoregressive moving average model to enforce stationarity," *Journal of Statistical Computation and Simulation*, vol. 24, no. 2, pp. 99–106, 1986.

[169] L. Cavalcante and R. J. Bessa, "Solar power forecasting with sparse vector autoregression structures," in *2017 IEEE Manchester PowerTech.* IEEE, Jun. 2017, pp. 1–6.

[170] H. Ahmadi, N. Pham, R. Ganti, T. Abdelzaher, S. Nath, and J. Han, "Privacy-aware regression modeling of participatory sensing data," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems.* ACM, 2010, pp. 99–112.

[171] P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in *2017 IEEE Symposium on Security and Privacy (SP).* IEEE, 2017, pp. 19–38.

[172] B. S. Rathore, A. Singh, and D. Singh, "A survey of cryptographic and non-cryptographic techniques for privacy preservation," *International Journal of Computer Applications*, vol. 975, p. 8887, 2015.

[173] R. J. Bessa, A. Trindade, C. S. Silva, and V. Miranda, "Probabilistic solar power forecasting in smart grids using distributed information," *International Journal of Electrical Power & Energy Systems*, vol. 72, pp. 16–23, 2015.

[174] J. Dowell and P. Pinson, "Very-short-term probabilistic wind power forecasts by sparse vector autoregression," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 763–770, 2015.

[175] X. G. Agoua, R. Girard, and G. Kariniotakis, "Probabilistic models for spatio-temporal photovoltaic power forecasting," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 2, pp. 780–789, 2018.

[176] Y. Zhao, L. Ye, P. Pinson, Y. Tang, and P. Lu, "Correlation-constrained and sparsity-controlled vector autoregressive model for spatio-temporal wind power forecasting," *IEEE Transactions on Power Systems*, vol. 33, no. 5, pp. 5029–5040, Sep. 2018.

[177] J. W. Messner and P. Pinson, "Online adaptive lasso estimation in vector autoregressive models for high dimensional wind power forecasting," *Int. Journal of Forecasting*, vol. 35, no. 4, pp. 1485–1498, Oct. 2019.

[178] B. Sommer, P. Pinson, J. Messner, and D. Obst, "Online distributed learning in wind power forecasting," *International Journal of Forecasting*, vol. 37, no. 1, pp. 205–223, Jan. 2021.

[179] V. Berdugo, C. Chaussin, L. Dubus, G. Hebrail, and V. Leboucher, "Analog method for collaborative very-short-term forecasting of power generation from photovoltaic systems," in *Proceedings Next Generation Data Mining Summit*, Greece, Sep. 2011, pp. 1–5.

[180] W. Li, H. Li, and C. Deng, "Privacy-preserving horizontally partitioned linear programs with inequality constraints," *Optimization Letters*, pp. 137–144, 2013.

[181] C. Dwork, K. Talwar, K. Talwar, A. Thakurta, and L. Zhang, "Analyze gauss: optimal bounds for privacy-preserving principal component analysis," in *Forty-sixth Annual ACM Symposium on Theory of Computing*, May 2014, pp. 11–20.

[182] W. Du, Y. S. Han, and S. Chen, "Privacy-preserving multivariate statistical analysis: Linear regression and classification," in *2004 SIAM International Conference on Data Mining*, 2004, pp. 222–233.

[183] Y. Liu, W. Guo, C.-I. Fan, L. Chang, and C. Cheng, "A practical privacy-preserving data aggregation (3PDA) scheme for smart grid," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1767–1774, 2018.

[184] G. Mateos, J. A. Bazerque, and G. B. Giannakis, "Distributed sparselinear regression," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5262–5276, 2010.

[185] Y. Li, X. Jiang, S. Wang, H. Xiong, and L. Ohno-Machado, "VERTIcal Grid lOgistic regression (VERTIGO)," *Journal of the American Medical Informatics Association*, vol. 23, no. 3, pp. 570–579, 2015.

[186] B. Elsinga and W. G. van Sark, "Short-term peer-to-peer solar forecasting in a network of photovoltaic systems," *Applied Energy*, vol. 206, pp. 1464–1483, Nov. 2017.

[187] A. Tascikaraoglu, "Evaluation of spatio-temporal forecasting methods in various smart city applications," *Renewable and Sustainable Energy Reviews*, vol. 82, no. 1, pp. 424–435, Feb. 2018.

[188] K. Chen and L. Liu, "A survey of multiplicative perturbation for privacy-preserving data mining," in *Privacy-Preserving Data Mining.* Springer, 2008, pp. 157–181.

[189] J. Tastu, P. Pinson, E. Kotwa, H. Madsen, and H. A. Nielsen, "Spatio-temporal analysis and modeling of short-term wind power forecast errors," *Wind Energy*, vol. 14, no. 1, pp. 43–60, Jan. 2011.

[190] J. Parra-Arnau, "Optimized, direct sale of privacy in personal data marketplaces," *Information Sciences*, vol. 424, pp. 354–384, 2018.

[191] S. Mehta, M. Dawande, G. Janakiraman, and V. Mookerjee, "How to sell a dataset? Pricing policies for data monetization," in *20th ACM Conference on Economics and Computation (EC'19).* ACM, 2019, pp. 679–679.

[192] I. Koutsopoulos, A. Gionis, and M. Halkidi, "Auctioning data for learning," in *2015 IEEE International Conference on Data Mining Workshop (ICDMW).* IEEE, 2015, pp. 706–713.

[193] X. Cao, Y. Chen, and K. R. Liu, "Data trading with multiple owners, collectors, and users: An iterative auction mechanism," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 2, pp. 268–281, 2017.

[194] D. Acemoglu, A. Makhdoumi, A. Malekian, and A. Ozdaglar, "Too much data: Prices and inefficiencies in data markets," National Bureau of Economic Research, Inc., Tech. Rep. NBER Working Papers 26296, 2019.

[195] S. Xuan, L. Zheng, I. Chung, W. Wang, D. Man, X. Du, W. Yang, and M. Guizani, "An incentive mechanism for data sharing based on blockchain with smart contracts," *Computers and Electrical Engineering*, vol. 83, p. 106587, 2020.

[196] A. Yassine, A. A. N. Shirehjini, and S. Shirmohammadi, "Smart meters big data: Game theoretic modelfor fair data sharing in deregulated smart grids," *IEEE Access*, vol. 3, pp. 2743–2754, Dec. 2015.

[197] O. Samuel, N. Javaid, M. Awais, Z. Ahmed, M. Imran, and M. Guizani, "Blockchain model for fair data sharing in deregulated smart grids," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, USA, Dec. 2019.

[198] C. Aperjis and B. A. Huberman, "A market for unbiased private data: Paying individuals according to their privacy attitudes," *First Monday*, vol. 17, no. 5–7, May 2012. [Online]. Available: https://journals.uic.edu/ojs/index.php/fm/article/download/4013/3209

[199] C. Niu, Z. Zheng, S. Tang, X. Gao, and F. Wu, "Making big money from small sensors: Trading time-series data under pufferfish privacy," in *IEEE Conference on Computer Communications (IEEE INFOCOM 2019).* IEEE, 2019, pp. 568–576.

[200] A. Agarwal, M. Dahleh, and T. Sarkar, "A marketplace for data: An algorithmic solution," in *2019 ACM Conference on Economics and Computation.* ACM, 2019, pp. 701–726.

[201] P. Pinson, C. Chevallier, and G. N. Kariniotakis, "Trading wind generation from short-term probabilistic forecasts of wind power," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1148–1156, 2007.

[202] T. Soares, P. Pinson, T. V. Jensen, and H. Morais, "Optimal offering strategies for wind power in energy and primary reserve markets," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 3, pp. 1036–1045, Jul. 2016.

[203] T. Jónsson, P. Pinson, H. Nielsen, and H. Madsen, "Exponential smoothing approaches for prediction in real-time electricity markets," *Energies*, vol. 7, no. 6, pp. 3710–3732, 2014.

[204] R. Koenker and K. F. Hallock, "Quantile regression," *Journal of economic perspectives*, vol. 15, no. 4, pp. 143–156, 2001.

[205] R. B. Myerson, "Optimal auction design," *Mathematics of operations research*, vol. 6, no. 1, pp. 58–73, 1981. [Online]. Available: oceanprotocol.com/tech-whitepaper.pdf

[206] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 307–317, 1953.

[207] S. Maleki, L. Tran-Thanh, G. Hines, T. Rahwan, and A. Rogers, "Bounding the estimation error of sampling-based Shapley value approximation," in *Fourth Workshop on Cooperative Games in Multiagent Systems (CoopMAS-2014)*, May 2014.

[208] "Ocean protocol: A decentralized substrate for AI data & services," 2019. [Online]. Available: oceanprotocol.com/tech-whitepaper.pdf

[209] G. Zyskind, O. Nathan, and A. Pentland, *New Solutions for Cybersecurity.* MIT Press, 2018.

[210] B. Goertzel, S. Giacomelli, D. Hanson, C. Pennachin, and M. Argentieri, “SingularityNET: A decentralized, open market and inter-network for AIs,” *Thoughts, Theories & Studies on Artificial Intelligence (AI) Research*, Dec. 2017. [Online]. Available: airesearch.com/ai-research-papers/singularitynet-a-decentralized-open-market-and-inter-network-for-ais/

[211] R. Craib, G. Bradway, X. Dunn, and J. Krug, “Numeraire: A cryptographic token for coordinating machine intelligence and preventing overfitting,” 2017. [Online]. Available: numer.ai/whitepaper.pdf

[212] S. Lerch, T. L. Thorarinsdottir, F. Ravazzolo, and T. Gneiting, “Forecaster’s dilemma: Extreme events and forecast evaluation,” *Statistical Science*, vol. 32, no. 1, pp. 106–127, 2017.

[213] J. B. de Souza, V. A. Reisen, G. C. Franco, M. Ispany, P. Bondon, and J. M. Santos, “Generalized additive models with principal component analysis: an application to time series of respiratory disease and air pollution data,” *Journal of the Royal Stat. Soc. Series C,*, vol. 67, no. 2, pp. 453–480, Feb. 2018.

[214] J. Jiang, “Multivariate functional-coefficient regression models for nonlinear vector time series data,” *Biometrika*, vol. 101, no. 3, pp. 689–702, Sep. 2014.

[215] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, “Federated learning: Challenges, methods, and future directions,” *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

A

Appendices

A.1 Differential Privacy

Mathematically, a randomized mechanism $\mathcal{A}$ satisfies (ε,δ)-differential privacy [126] if, for every possible output t of $\mathcal{A}$ and for every pair of datasets $\mathbf{D}$ and $\mathbf{D}'$ (differing in at most one record),

$$\Pr(\mathcal{A}(\mathbf{D}) = t) \leq \delta + \exp(\varepsilon)\Pr(\mathcal{A}(\mathbf{D}') = t). \tag{A.1}$$

In practice, differential privacy can be achieved by adding random noise W to some desirable function f of the data $\mathbf{D}$. That is,

$$\mathcal{A}(\mathbf{D}) = f(\mathbf{D}) + W. \tag{A.2}$$

The (ε,0)-differential privacy is achieved by applying noise from Laplace distribution with scale parameter $\frac{\Delta f_1}{\varepsilon}$, with $\Delta f_k = \max\{\|f(\mathbf{D}) - f(\mathbf{D}')\|_k\}$. A common alternative is the Gaussian distribution but, in this case, $\delta > 0$ and the scale parameter which allows (ε,δ)-differential privacy is $\sigma \geq \sqrt{2\log\left(\frac{1.25}{\delta}\right)\frac{\Delta_2 f}{\varepsilon}}$. Dwork and Smith [126] showed that the data can be masked by considering

$$\mathcal{A}(\mathbf{D}) = \mathbf{D} + \mathbf{W}. \tag{A.3}$$

A.2 Optimal value of r

Proposition 5 *Let $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$ be the sensible data from agent i, with u unique values, and $\mathbf{M}_{A_j} \in \mathbb{R}^{T\times T}$ be the private encryption matrix from agent j. If agents compute $\mathbf{M}_{A_j}\mathbf{X}_{A_i}$ applying the protocol in* (3.4)–(3.5)*, then two invertible matrices $\mathbf{D}_{A_i} \in \mathbb{R}^{r\times r}$ and $\mathbf{C}_{A_i} \in \mathbb{R}^{T\times(r-s)}$ are generated by agent i and data privacy is ensured for*

$$\sqrt{Ts - u} < r < T. \tag{A.4}$$

Proof Since agent i only receives $\mathbf{M}_{A_j}[\mathbf{X}_{A_i}\mathbf{C}_{A_i}]\mathbf{D}_{A_i} \in \mathbb{R}^{T\times r}$, the matrix $\mathbf{M}_{A_j} \in \mathbb{R}^{T\times T}$ is protected if $r < T$. Furthermore, agent j receives $[\mathbf{X}_{A_i}\mathbf{C}_{A_i}]\mathbf{D}_{A_i} \in \mathbb{R}^{T\times r}$ and does not know $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}, \mathbf{C}_{A_i} \in \mathbb{R}^{T\times r-s}$ and $\mathbf{D}_{A_i} \in \mathbb{R}^{r\times r}$. Although $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$, we assume this matrix has u unique values whose positions are known by all agents – when defining a VAR model with p consecutive lags $\mathbf{Z}_{A_i}$ has T+p−1 unique values, see Figure II.10 – meaning there are fewer values to recover.

Given that, agent j receives Tr values and wants to determine $u + T(r-s) + r^2$. The solution of the inequality $Tr < u + T(r-s) + r^2$, in r, determines that data from agent i is protected when $r > \sqrt{Ts-u}$.

□

Proposition 6 *Let* $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$ *and* $\mathbf{G}_{A_i} \in \mathbb{R}^{T\times g}$ *be private data matrices, such that* $\mathbf{X}_{A_i}$ *has* u *unique values to recover and* $\mathbf{G}_{A_i}$ *has* v *unique values that are not in* $\mathbf{X}_{A_i}$*. Assume the protocol in* (3.4)*–*(3.5) *is applied to compute* $\mathbf{M}\mathbf{X}_{A_i}$*,* $\mathbf{X}_{A_i}^\top \mathbf{M}^{-1}$ *and* $\mathbf{M}\mathbf{G}_{A_i}$*, with* $\mathbf{M}$ *as defined in* (3.2)*. Then, to ensure privacy while computing* $\mathbf{M}\mathbf{X}_{A_i}$ *and* $\mathbf{X}_{A_i}^\top \mathbf{M}^{-1}$*, the protocol requires*

$$\sqrt{Ts-u} < r < T/2 \wedge r > s. \tag{A.5}$$

In addition, to compute $\mathbf{M}\mathbf{G}_{A_i}$*, the protocol should take*

$$\sqrt{Tg-v} < r' < T - 2r \wedge r' > g. \tag{A.6}$$

Proof *(i)* To compute $\mathbf{M}\mathbf{X}_{A_i}$, the i-th agent shares $\mathbf{W}_{A_i} = [\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i} \in \mathbb{R}^{T\times r}$ with the n-th agent, $\mathbf{C}_{A_i} \in \mathbb{R}^{T\times(r-s)}$, $\mathbf{D}_{A_i} \in \mathbb{R}^{r\times r}$, $r > s$. Then, the process repeat until the 1-st agent receives $\mathbf{M}_{A_2} \dots \mathbf{M}_{A_n}\mathbf{W}_{A_i}$ and computes $\mathbf{M}\mathbf{W}_{A_i} = \mathbf{M}_{A_1}\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}\mathbf{W}_{A_i}$. Consequently, agent $j = 1, \dots, n$ receives Tr values during the protocol.

(ii) $\mathbf{X}_{A_i}^\top \mathbf{M}^{-1}$ is computed using the matrix $\mathbf{W}_{A_i}$ defined before. Since $\mathbf{M}^{-1} = \mathbf{M}_{A_n}^{-1} \dots \mathbf{M}_{A_1}^{-1}$, the n-th agent computes $\mathbf{W}_{A_i}^\top \mathbf{M}_{A_n}^{-1}$. Then, the process repeat until the 1-st agent receives $\mathbf{W}_{A_i}^\top \mathbf{M}_{A_n}^{-1} \dots \mathbf{M}_{A_2}^{-1}$ and computes $\mathbf{W}_{A_i}^\top \mathbf{M}^{-1} = \mathbf{W}_{A_i}^\top \mathbf{M}_{A_n}^{-1} \dots \mathbf{M}_{A_2}^{-1}\mathbf{M}_{A_1}^{-1}$. Again, the j-th agent receives Tr values related to the unknown data from the i-th agent.

In summary, the n-th agent receives Tr values and unknowns $u + T(r-s) + r^2$ (from $\mathbf{X}_{A_i}, \mathbf{C}_{A_i}, \mathbf{D}_{A_i}$). The solution for $Tr{<}u{+}T(r{-}s){+}r^2$ allows to infer that $\mathbf{X}_{A_i}$ is protected if

$$r > \sqrt{Ts-u}.$$

On the other hand, the i-th agent receives $2Tr$ values ($\mathbf{M}\mathbf{W}_{A_i}, \mathbf{W}_{A_i}^\top \mathbf{M}^{-1}$) and unknowns T^2 from $\mathbf{M} \Rightarrow r{<}T/2$.

(iii) Finally, to compute $\mathbf{M}\mathbf{G}_{A_i}$, the i-th agent should define new matrices $\mathbf{C}'_{A_i} \in \mathbb{R}^{T\times(r'-g)}$ and $\mathbf{D}'_{A_i} \in \mathbb{R}^{r'\times r'}$ sharing $\mathbf{W}'_{A_i} = [\mathbf{G}_{A_i}, \mathbf{C}'_{A_i}]\mathbf{D}'_{A_i} \in \mathbb{R}^{T\times r'}$, $r' > g$. The computation of $\mathbf{M}\mathbf{W}'$ provides Tr' new values, meaning that after computing $\mathbf{M}\mathbf{X}_{A_i}$, $\mathbf{X}_{A_i}^\top \mathbf{M}^{-1}$ and $\mathbf{M}\mathbf{G}_{A_i}$, the n-th agent has $Tr + Tr'$ values and does not know $u + T(r-s) + r^2 + v + T(r'-g) + r'^2$ (from $\mathbf{X}_{A_i}$, $\mathbf{C}_{A_i}$, $\mathbf{D}_{A_i}$, $\mathbf{G}_{A_i}$, $\mathbf{C}'_{A_i}$ and $\mathbf{D}'_{A_i}$ respectively). The solution of the inequality $Tr + Tr' < u + T(r-s) + r^2 + v + T(r'-g) + r'^2$ allows to infer that $r' > \sqrt{Ts - u - r^2 - v + Tg} > \sqrt{Tg - v}$.

On the other hand, the i-th agent receives $2Tr + Tr'$ and does not know T^2, meaning that $r' < T - 2r$. □

A.3 Privacy Analysis

The proposed approach requires agents to encrypt their data and then exchange that encrypted data. This appendix section analyzes the global exchange of information. First, we show that the proposed privacy protocol is secure in a scenario without collusion, i.e., no alliances between agents (data owners) to determine the private data. Then, we analyze how many agents have to collude for a privacy breach to occur.

A.3.1 No collusion between agents

While encrypting sensible data $\mathbf{X}_{A_i} \in \mathbb{R}^{T\times s}$ and $\mathbf{G}_{A_i} \in \mathbb{R}^{T\times g}$ such that $\mathbf{X}_{A_i}$ has u unique values to recover and $\mathbf{G}_{A_i}$ has v unique values that are not in $\mathbf{X}_{A_i}$, the 1-st agent obtains

$\mathbf{M}[\mathbf{X}_{A_i},\mathbf{C}_{A_i}]\mathbf{D}_{A_i}\in\mathbb{R}^{T\times r}$, $[[\mathbf{X}_{A_i},\mathbf{C}_{A_i}]\mathbf{D}_{A_i}]^\top\mathbf{M}^{-1}\in\mathbb{R}^{r\times T}$ and $\mathbf{M}[\mathbf{G}_{A_i},\mathbf{C}'_{A_i}]\mathbf{D}'_{A_i}\in\mathbb{R}^{T\times r'}$, $\forall i$, which provides $2nTr+nTr'$ values. At this stage, the agent does not know

$$\underbrace{T^2}_{\mathbf{M}}+\underbrace{(n-1)u}_{\mathbf{X}_{A_i},\forall i\neq 1}+\underbrace{(n-1)v}_{\mathbf{G}_{A_i},\forall i\neq 1}+\underbrace{(n-1)T(r-s)}_{\mathbf{C}_{A_i},\forall i\neq 1}+\underbrace{(n-1)r^2}_{\mathbf{D}_{A_i},\forall i\neq 1}+\underbrace{(n-1)T(r'-g)}_{\mathbf{C}'_{A_i},\forall i\neq 1}+\underbrace{(n-1)r'^2}_{\mathbf{D}'_{A_i},\forall i\neq 1}$$

values. Then, while fitting the LASSO-VAR model, the 1-st agent can recover $\mathbf{MX}\in\mathbb{R}^{T\times ns}$ and $\mathbf{MG}\in\mathbb{R}^{T\times ng}$, as shown in Chapter 2. That said, the 1-st agent receives $2nTr+nTr'+nTs+nTg$, and a confidentiality breach occurs if $T(2nr+nr'+ns+ng)\geq T^2+(n-1)[u+v+T(r-s)+r^2+T(r'-g)+r'^2]$.

After a little algebra, it is possible to verify that taking (A.5), $\exists\ r'$ in (A.6), such as the previous inequality is not satisfied.

A.3.2 Collusion between agents

A set of agents $\mathcal{C}$ can come together to recover the data of the remaining competitors. This collusion assumes that such agents are willing to share their private data. Let c be the number of agents colluding. In this scenario, the objective is to determine $\mathbf{M}\in\mathbb{R}^{T\times T}$, knowing $\mathbf{MW}_{A_i}\in\mathbb{R}^{T\times r}$, $\mathbf{W}_{A_i}^\top\mathbf{M}^{-1}\in\mathbb{R}^{r\times T}$, $\mathbf{MW}'_{A_i}\in\mathbb{R}^{T\times r'}$, $\mathbf{MX}_{A_i}\in\mathbb{R}^{T\times s}$, and $\mathbf{MG}_{A_i}\in\mathbb{R}^{T\times g}$, $i\in\mathcal{C}$.

Mathematically, it means that colluders can recover T^2 values by solving $cT(r+r+r'+s+g)$ equations, which is only possible for $c\geq\lceil\frac{T}{2r+r'+s+g}\rceil$.