Smart4RES

Distributed and Collaborative Forecasting

D4.1 Distributed and Collaborative Forecasting

WP4, T4.1

Version V1.0

Authors:
Carla Gonçalves, INESC TEC
Ricardo Jorge Bessa, INESC TEC
José Ricardo Andrade, INESC TEC
Hodajt Mariji, INESC TEC
Pierre Pinson, DTU
Liyang Han, DTU
Amandine Pierrot, DTU
Jalal Kazempour, DTU

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Disclaimer

The present document reflects only the author's view. The European Climate, Infrastructure and Environment Executive Agency (CINEA) is not responsible for any use that may be made of the information it contains.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Technical references

Project Acronym	Smart4RES
Project Title	Next Generation Modelling and Forecasting of Variable Renewable Generation for Large-scale Integration in Energy Systems and Markets
Project Coordinator	ARMINES – MINES ParisTech
Project Duration	November 2019 – April 2023

Deliverable	D4.1 Distributed and Collaborative Forecasting
Dissemination level[1]	PU
Nature[2]	R
Work Package	WP 4 – Collaborative Framework to RES Forecasting and Resulting Business Models
Task	T 4.1 – Distributed and Collaborative Forecasting
Lead beneficiary	INESC TEC (05)
Contributing beneficiary(ies)	DTU (03)
Reviewers	–
Due date of deliverable	March 2022 (M)

1 PU = Public

PP = Restricted to other program participants (including the Commission Services)

RE = Restricted to a group specified by the consortium (including the Commission Services)

CO = Confidential, only for members of the consortium (including the Commission Services)

2 R = Report, P = Prototype, D = Demonstrator, O = Other

Document history		
V	Date	Description
0.1	08/03/2022	First version containing contributions from all partners
1.0	28/03/2022	Final version published by the Coordinator

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Executive summary

This Deliverable Report presents the work developed by INESC TEC and DTU in the framework of Task 4.1 ("Distributed and collaborative forecasting") from Smart4RES project. The aim of this Task was to rethink forecasting problems with geographically distributed renewable energy sources (RES) data (power measurements, grid of weather predictions, etc.) by reformulating them as distributed learning problems, considering aspects such as data privacy/confidentiality, online learning, and probabilistic forecasts (including conditional distribution's tails). The following paragraphs present a summary of the develop work and main outcomes.

Extreme Conditional Quantile Forecasting. 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 work, 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.

Privacy-preserving algorithms for RES forecasting. Cooperation between different data owners may lead to an improvement in RES forecasting quality – for instance, by benefiting from spatio-temporal dependencies in geographically distributed time series. Due to business competitive factors and personal data protection concerns, the data owners might be unwilling to share their data. Hence, interest in collaborative privacy-preserving forecasting (or vertical federated learning) is thus increasing. Firstly, this work starts by analyzing the state-of-the-art and unveils several shortcomings of existing methods in guaranteeing data privacy when employing vector autoregressive models for multivariate RES time series forecasting. The state-of-the-art methods were divided into three groups: data transformation, secure multi-party computations, and decomposition methods. The analysis showed 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 solar data; and *(ii)* iterative model fitting processes, which reveal data after a number of iterations. Secondly, in order to tackle this privacy issue, Smart4RES formulated 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.

Online distributed learning and reconciliation in RES forecasting. Forecasting RES generation up to a few hours ahead is of utmost importance for the efficient operation of power systems and for participation in electricity markets. Recent statistical learning approaches exploit spatio-temporal dependence patterns among neighboring sites but their requirement of sharing confidential data with third parties may limit their use in practice. This explains the recent interest in distributed, privacy preserving algorithms to high-dimensional statistical learning, e.g., for autoregressive models. The few approaches that have been proposed are based on batch learning.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

These approaches are potentially computationally expensive while not allowing the accommodation of nonstationary characteristics of stochastic processes like wind power generation. Additionally, since many agents in power systems and electricity markets generate their own forecasts, at various aggregation levels and independently of each other, these forecasts may end up not being coherent. Smart4RES first closes the gap between online and distributed optimisation by presenting two novel approaches that recursively update model parameters while limiting information exchange between wind farm operators and other potential data providers, and then proposes an approach to the forecast reconciliation problem using a recursive and adaptive multivariate least squares estimator, with equality constraints on the coefficients, which guarantees the coherency property not only in-sample but also out-of-sample. A simulation study allows the comparison of the convergence and tracking ability of both approaches. In addition, a case study using a large dataset from 311 wind farms in Denmark confirms that online distributed approaches generally outperform existing batch approaches, while agents do not have to actively share their private data. Finally, the effectiveness of the reconciliation approach is then verified in a separate case study using a Danish wind energy dataset with 100 wind farms.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table of contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

LIST OF TABLES

LIST OF FIGURES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Acronyms

ADMM Alternating Direction Method of Multipliers.

AR AutoRegressive.

CAE Cumulative Absolute Error.

CDF Cumulative Distribution Function.

CRPS Continuous Ranked Probability Score.

D-MIDAS Distributed MIrror Descent Algorithm made Sparse.

DM Diebold-Mariano.

DSO Distribution System Operator.

EVT Extreme Value Theory.

Exp_Tails Exponential function.

GBT Gradient Boosting Tree.

GBT_EVT GBT combined with Hill estimator.

GBT_tGPD Proposed method combining GBT with truncated GPD.

GPD generalized Pareto distribution.

LASSO Least Absolute Shrinkage and Selection Operator.

MAE Mean Absolute Error.

NRMSE Normalized Root Mean Squared Error.

NWP Numerical Weather Prediction.

OADMM Online Alternating Direction Method of Multipliers.

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.

RES Renewable Energy Sources.

RMSE Root Mean Squared Error.

SMIDAS Stochastic MIrror Descent Algorithm made Sparse.

TSO Transmission System Operator.

VAR Vector AutoRegressive.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

I. Introduction

In renewable energy sources (RES) forecasting, past results showed that geographically distributed weather and RES power time series can improve wind (Tastu et al., 2010) and solar power forecasting (Bessa et al., 2015b) skill in hours-ahead forecasting, and features extracted from a grid of NWP (Andrade and Bessa, 2017) or turbine-level data improve days-ahead forecasting (Gilbert et al., 2020a). Most of the works that have shown the interest of using spatially distributed data have assumed that data could be gathered centrally and used, either at the wind farm level, or at the level of a system operator. This is not in line with current practice, where data is distributed in terms of ownership, limitation is data transfer capabilities, and with agents being reluctant to share their data anyway. This motivates the construction of new business models for RES forecasting driven to exploit data from different owners and create economic signals for data sharing and collaborative analytics. Hence, the following requirements need to be addressed: (1) data privacy and confidentiality when combining data from different owners; (2) robust data exchange schemes (centralized, peer-to-peer, asynchronous, etc.) for collaborative analytics; (3) algorithmic solutions for data markets in RES forecasting (covered in Deliverable D4.2 of Smart4RES project).

In this context, this deliverable details the work developed by INESC TEC and DTU for Task 4.1 ("Distributed and collaborative forecasting") of Smart4RES Task 4.1, producing the following main outcomes:

1. Conditional extreme quantile (i.e., distribution's tails) forecasting model, that combines extreme value theory estimators for truncated generalized generalized Pareto distribution with non-parametric methods, conditioned by spatio-temporal information. **(Section II)**
2. Numerical and mathematical analysis of the existing privacy-preserving regression models for time series forecasting and identification of weaknesses in the current literature. **(Section III)**
3. Privacy-preserving forecasting algorithm (or vertical federated learning according to the federated learning nomenclature, Chen et al. (2020)) for vector autoregressive forecasting models (i.e., multivariate time series forecasting), that protects data by combining linear algebra transformations with a decomposition-based algorithm. This method was extended to capture nonlinear relations (e.g., between wind speed and wind power) through splines (additive model framework). **(Section IV)**
4. Two novel approaches that recursively update model parameters while limiting information exchange between wind farm operators and other potential data providers – tackles the gap between online and distributed optimisation. **(Section V)**
5. An online forecast reconciliation approach in a constrained regression framework, which relies on a multivariate least squares estimator, with equality constraints on the coefficients. A recursive and adaptive version of that estimator is derived, hence allowing to track the optimal reconciliation in a fully data-driven manner. **(Section VI)**

The novel RES forecasting models were evaluated using synthetic data (when justified to check the validity of the mathematical modeling) and real publicly available datasets (from wind and solar energy power plants).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

II. Extreme conditional quantile forecasting

II.1 Introduction

The growing integration of renewable energy sources (RES) brings new challenges to system operators and market players and robust forecasting models are fundamental for handling their variability and uncertainty. This fomented a growing interest in RES probabilistic forecasting techniques and its integration in decision-aid under risk (Bessa et al., 2017).

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. An up-to-date literature review about RES probabilistic forecasting can be found in (Sweeney et al., 2020b). Parametric models assume that data are generated from a known probability distribution (e.g. Gaussian, Beta), whose parameters are estimated from the data. Non-parametric models do not make any assumptions about the shape of the probability distribution and comprise techniques such as quantile regression (QR) with radial basis functions (Juban et al., 2016), local QR (Bremnes, 2004), conditional kernel density estimation (Bessa et al., 2012b) and gradient boosting trees (GBT) (Andrade and Bessa, 2017). It is also possible to find semi-parametric approaches, e.g., 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 forecasted with a statistical model (Pinson, 2012b); combination of linear regression, inverse (power-to-wind) transformation and censored normal distribution (Messner et al., 2013).

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 with nominal proportion 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% (Matos and Bessa, 2010); the distribution's tails forecasting accuracy affects the decision quality of advanced RES bidding strategies that are based on risk metrics such as conditional value-at-risk (Botterud et al., 2012); dynamic line rating uncertainty forecasting for transmission grids also requires the use of low quantiles (e.g., 1%) (Dupin, 2018). Moreover, the generation of temporal and/or spatial-temporal trajectories (or random vectors) with a statistical method, such as the Gaussian copula (Pinson et al., 2009), requires a full modelling 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 modelling 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 optimal solution, in stochastic programming, obtained with the forecasted and realized probability distribution (Cagnolari, 2017).

By exploring concepts from extreme value theory (EVT), which is dedicated to characterise the stochastic behaviour of extreme values De Haan and Ferreira (2007), the present section proposes a novel wind power forecasting methodology, focused in improving the forecasting skill

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

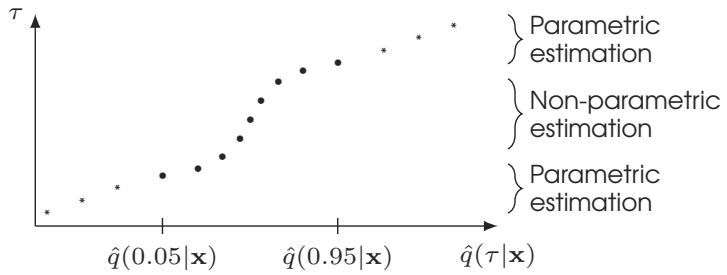


Figure 1 The proposed method uses different estimators for intermediate and extreme quantiles.

of the distribution's tails, which combines spatio-temporal information (obtained trough feature engineering), gradient boosting trees (GBT) as a non-parametric method for quantiles between 0.05 and 0.95 and the truncated generalized Pareto distribution (GPD) for the tails.

The remaining of this section is organized as follows. Subsection II.2 presents related work and identify contributions. Subsection II.3 introduces the relevant statistical background of non-parametric and parametric methods. Subsection II.4 describes a novel forecasting method combining GBT with truncated GPD. Subsection II.5 describes the experiments to evaluate the proposed method and conclusions are drawn in Subsection II.6.

II.2 Related Work and Contributions

Andersen (2009) and Matos et al. (2016) used a QR model to forecast the wind 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 wind power, i.e., observed power is divided into equally populated bins according to forecasted wind 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 work), 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 by Beirlant et al. (2004) 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, Wang et al. (2012) 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 (Wang and Li, 2013) works by first finding an appropriate power transformation of Y, then estimating the intermediate conditional quantiles of transformed Y using linear QR and finally extrapolating these estimates to extreme tails with EVT estimators. At 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 (Wang et al., 2012). 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. This estimation is then used to guide the parametric model by rating historically similar periods conditioned by the covariates.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Finally, two works proposed the use of spatio-temporal data in RES probabilistic forecasting: combination of GBT with feature engineering techniques to extract information from a grid of Numerical Weather Predictions (NWP) (Andrade and Bessa, 2017); hierarchical forecasting models to leverage turbine-level data (Gilbert et al., 2020b). Both works do not deal or propose a specific methodology to forecast conditional distribution's tails.

II.3 Background: Non-parametric and Parametric Methods

This section presents the main statistical methods to construct the proposed method and baseline approaches. In what follows, $\mathbf{x}_i$ is the observed p-dimensional vector of covariates and y_i is the target variable, with $i \in \{1, \dots, n\}$.

II.3.1 Non-parametric Methods

II.3.1.1 Quantile Regression The QR model (Koenker and Bassett Jr, 1978) estimates the conditional quantile function of Y given X,

$$Q^{\text{QR}}(\tau|X) = \beta_0(\tau) + \beta_1(\tau)X_1 + \cdots + \beta_p(\tau)X_p, \tag{1}$$

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

$$\hat{\boldsymbol{\beta}}(\tau) = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \rho_\tau \left(y_i - \beta_0(\tau) - \sum_{j=1}^{p} \beta_j(\tau) x_{ij} \right), \tag{2}$$

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* (Koenker and Bassett Jr, 1978).

II.3.1.2 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,

$$Q_j^{\text{GBT}}(\tau|X) = Q_{j-1}^{\text{GBT}}(\tau|X) + \eta f_j(\tau|X). \tag{3}$$

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*

$$\hat{f}_j(\tau|X) = \arg\min_{f_j} \sum_{i=1}^{n} \rho_\tau \left(y_i, \hat{Q}_{j-1}^{\text{GBT}}(\tau|\mathbf{x}_i) + \eta f_j(\tau|\mathbf{x}_i) \right). \tag{4}$$

The initial model Q_1^{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 — see (Andrade and Bessa, 2017) for more details.

II.3.1.3 Rearrangement of quantiles Since both QR and GBT solve an optimization problem for each quantile τ independently, 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 cumulative distribution function (CDF)

$$F(y|\mathbf{x}) = \int_0^1 \mathbf{1}_{Q(\tau|\mathbf{x}) \leq y} d\tau \tag{5}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

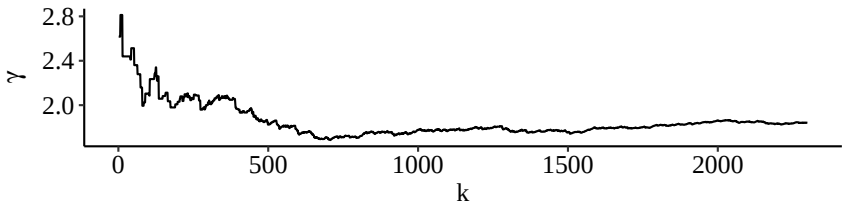


Figure 2 Illustration of γ value in function of k. The first stable part of the plot happens for $k \approx 700$.

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

$$\tilde{Q}(\tau|\mathbf{x}) = F^{-1}(\tau|\mathbf{x}) \tag{6}$$

which is monotone in τ (Chernozhukov et al., 2010).

II.3.2 Parametric Methods for Extreme Quantiles

II.3.2.1 Exponential function In (Andersen, 2009), distribution' 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

$$\hat{Q}^{\text{exp}}(\tau|\mathbf{x}) = \begin{cases} \hat{Q}(0.05|\mathbf{x}) \frac{\ln(\frac{0.05}{\rho})}{\ln(\frac{\tau}{\rho})}, & \tau < 0.05, \\ C\left(1 - \left(1 - \frac{\hat{Q}(0.95|\mathbf{x})}{C}\right) \frac{\ln(\frac{1-0.95}{\rho})}{\ln(\frac{1-\tau}{\rho})}\right), & \tau > 0.95, \end{cases} \tag{7}$$

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 (Matos et al., 2016).

II.3.2.2 Hill-based methods In (Wang et al., 2012) and (Wang and Li, 2013), a QR model is combined with EVT estimators. First, a local QR model is used to estimate the conditional quantiles $\tau_j = j/(n+1)$, denoted as $\hat{Q}^{\text{QR}}(\tau_j|\mathbf{x})$, $j \in \{1, ..., n - [n^\eta]\}$, for some $0 < \eta < 1$, being $[u]$ the integer part of u, and n the number of observations. Then, using these values, extreme quantiles are computed through an adaptation of Weissman's estimator,

$$\hat{Q}^{\text{W}}(\tau|\mathbf{x}) = \left(\frac{1-\tau_{n-k}}{1-\tau_n}\right)^{\hat{\gamma}(\mathbf{x})} \hat{Q}^{\text{QR}}(\tau_{n-k}|\mathbf{x}), \tag{8}$$

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

$$\hat{\gamma}(\mathbf{x}) = \frac{1}{k - [n^\eta]} \sum_{j=[n^\eta]}^{k} \log \frac{\hat{Q}^{\text{QR}}(\tau_{n-j}|\mathbf{x})}{\hat{Q}^{\text{QR}}(\tau_{n-k}|\mathbf{x})}. \tag{9}$$

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 (De Haan and Ferreira, 2007), see Figure 2.

In (Wang and Li, 2013), 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,

$$\Lambda_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0, \\ \log(y), & \text{if } \lambda = 0. \end{cases} \tag{10}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

For this approach, k is estimated to minimize

$$\arg\min_{k\geq 1} \sum_{i=1}^{n} \hat{\lambda}\hat{\gamma}(\mathbf{x}_i) - \hat{\gamma}^*(\mathbf{x}_i), \tag{11}$$

where

$$\hat{\gamma}^*(\mathbf{x}) = M_{0,n}^{(1)} + 1 - \frac{1}{2}\left(1 - \frac{(M_{0,n}^{(1)})^2}{M_{0,n}^{(2)}}\right)^{-1} \tag{12}$$

$$M_{0,n}^{(i)} = \frac{1}{k - [n^\eta]} \sum_{j=[n^\eta]}^{k} \left(\log \frac{\hat{Q}^{QR}(\tau_{n-j})}{\hat{Q}^{QR}(\tau_{n-k})}\right)^i. \tag{13}$$

II.3.2.3 Peaks-over-threshold (POT) method with truncation Since wind power generation is limited between 0 and installed capacity C, we observe the truncated random variable $Y, Y <= C$. (Beirlant et al., 2017) provide 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 (McNeil and Saladin, 1997) is adapted to estimate extreme quantiles from a GPD distribution affected by truncation at point C. The quantiles for Y are estimated by

$$\hat{Q}_k^{\text{tGPD}}(1-p) = Y_{n-k,n} + \frac{\hat{\sigma}_k}{\hat{\xi}_k}\left(\left[\frac{\hat{D}_{C,k} + \frac{(k+1)}{(n+1)}}{p(\hat{D}_{C,k}+1)}\right]^{\hat{\xi}_k} - 1\right), \tag{14}$$

where $Y_{1,n} < \cdots < Y_{n,n}$ 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

$$\hat{D}_{C,k} = \max\left\{0, \frac{k}{n}\frac{(1+(\hat{\xi}_k/\hat{\sigma}_k)E_{1,k})^{-1/\hat{\xi}_k} - \frac{1}{k}}{1-(1+(\hat{\xi}_k/\hat{\sigma}_k)E_{1,k})^{-1/\hat{\xi}_k}}\right\}, \tag{15}$$

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

The GDP estimator will be used in our proposed method because (i) the shape parameter ξ allows modeling everything from extreme events with lightweight distribution ($\xi<0$) to events with exponential distribution ($\xi=0$) and events with heavy distribution ($\xi>0$); (ii) the existence of estimators for truncated GPD that can handle random variables with limited support like wind power.

II.3.3 Evaluation Metrics

This subsection describes the set of metrics adopted to evaluate probabilistic forecasting skill of extreme quantiles.

II.3.3.1 Calibration Measures the mismatch between the empirical probabilities (or long-run quantile proportions) and nominal (or subjective) probabilities, e.g. a .25 quantile 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

$$\hat{\alpha}(\tau) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{y_i \leq \hat{Q}_y(\tau|\mathbf{x}_i)}. \tag{16}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

II.3.3.2 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

$$\text{sharp}_Y(\tau) = \frac{1}{n}\sum_{i=1}^{n} \hat{Q}_Y(1-\tau|\mathbf{x}_i) - \hat{Q}_Y(\tau|\mathbf{x}_i), \tag{17}$$

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

II.3.3.3 Continuous Ranked Probability Score (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 (Friederichs and Thorarinsdottir, 2012). Let y be the observation, and F_Y the CDF associated with an empirical probabilistic forecast,

$$\text{CRPS}(F_Y, y) = \int_{-\infty}^{\infty} \Big(F_Y(z) - H(z-y)\Big)^2 dz, \tag{18}$$

where H is the Heaviside function.

Although CRPS is very popular in evaluating the quality of CDF forecast, recent work in (Taillardat et al., 2019) 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 (Friederichs and Thorarinsdottir, 2012), is the *pinball function* or quantile loss. Smaller the value of the quantile score, better the model when forecasting quantile τ.

II.3.3.4 Pinball loss function or quantile score Assess the accuracy of each quantile forecast $\hat{Q}_Y(\tau|\mathbf{x}_i)$ by weighting the differences, between $\hat{Q}_Y(\tau|\mathbf{x}_i)$ and y_i, according to its sign and τ value (Koenker and Bassett Jr, 1978),

$$\rho_\tau(y_i, \hat{Q}_Y(\tau|\mathbf{x}_i)) = \begin{cases} \tau\left[y_i - \hat{Q}_Y(\tau|\mathbf{x}_i)\right], & \text{if } y_i > \hat{Q}_Y(\tau|\mathbf{x}_i), \\ (\tau-1)\left[y_i - \hat{Q}_Y(\tau|\mathbf{x}_i)\right], & \text{otherwise.} \end{cases} \tag{19}$$

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

II.4 Gradient Boosting Trees with a Truncated Generalized Pareto Model

As previously discussed in Section II.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 3:

Step 1 Non-parametric estimation: A non-parametric model $Q(\tau|\mathbf{x})$ is estimated for intermediate quantiles, e.g., $\tau \in \boldsymbol{\tau} = \{0.05, 0.10, \ldots, 0.95\}$, i.e., 19 models are estimated using

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

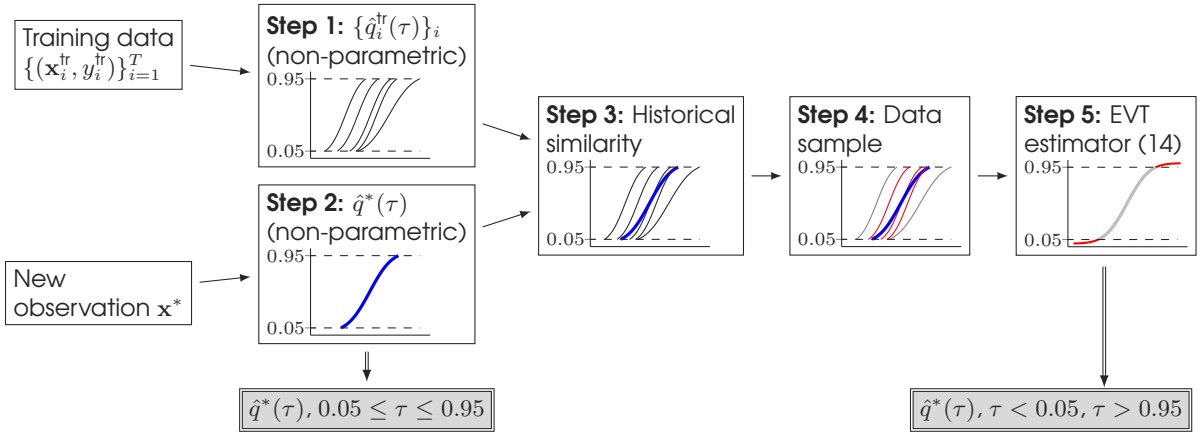


Figure 3 Overview of the proposed forecasting model.

available historical data $\{(\mathbf{x}_i^{\text{tr}}, y_i^{\text{tr}})\}_{i=1}^T$. A rearrangement is also performed as described in (6). 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 $s(\mathbf{q}_1, \mathbf{q}_2)$ 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, $\hat{\mathbf{q}}_i^{\text{tr}} = [\hat{q}_i^{\text{tr}}(\tau) \,|\, \tau \in \boldsymbol{\tau}]$. This similarity function is the Kolmogorov-Smirnov statistic given by

$$s(\mathbf{q}_1, \mathbf{q}_2) = \sup_{\tau} |\hat{\mathbf{q}}_1(\tau) - \hat{\mathbf{q}}_2(\tau)| \,. \tag{20}$$

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 (14) 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 (14), 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 (14) 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 Beirlant et al. (2004), 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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 1 Evaluated forecasting models.

Notation	Description
GBT	Gradient Boosting Trees (non-parametric model)
local_tGPD	Hill estimator and truncated GPD in (14)*
Exp_Tails	Exponential functions in (7), using GBT
QR_EVT	QR combined with Hill estimator in (8)**, as in Wang et al. (2012)
QR_EVT_T	QR, Hill estimator and transformed power data as in (10)**, as in Wang and Li (2013)
GBT_EVT	GBT combined with Hill estimator (8)**
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

II.5 Case Studies

To evaluate the added-value of the proposed method, the models described in Table 1 are compared using three different datasets. The implementation is performed through R and Python programming languages. The local_tGPD benchmark is a naive model: the estimator for the truncated GPD (14) 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 Nogueira (2020). 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 CRPS (18) is considered for each training set in the optimization process.

Also, the EVT estimators, in (8) and (14), 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 2. 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, 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 (16), sharpness (17) and quantile score function (19).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

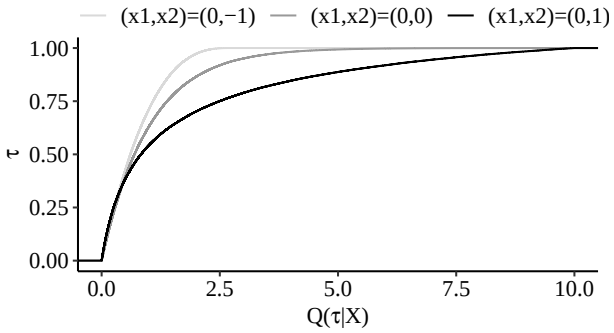


Figure 4 CDF for $(x_1^*, x_2^*) \in \{(0,-1),(0,0),(0,1)\}$.

II.5.1 Synthetic Data

II.5.1.1 Data Description The proposed approach is firstly studied through simulation. The distribution from which we simulated Y is the truncated GPD for which the CDF is given by

$$F^{\text{tGPD}}_{(C,\mu,\sigma,\xi)}(y) = \frac{F_{(\mu,\sigma,\xi)}(y) - F_{(\mu,\sigma,\xi)}(C)}{1 - F_{(\mu,\sigma,\xi)}(C)} \tag{21}$$

with

$$F_{(\mu,\sigma,\xi)}(y) = \begin{cases} 1 - \left(1 + \frac{\xi(y-\mu)}{\sigma}\right)^{-1/\xi} & \text{for } \xi \neq 0, \\ 1 - \exp\left(-\frac{y-\mu}{\sigma}\right) & \text{for } \xi = 0, \end{cases} \tag{22}$$

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 4, for which $\xi < 0$, $\xi = 0$ and $\xi > 0$, respectively.

II.5.1.2 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 5, 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 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 6 summarizes the difference between the normalized absolute deviations,

$$\frac{|\hat{Q}^{\text{benchmark}}(\tau|\mathbf{x}) - Q^{\text{tGPD}}(\tau|\mathbf{x})| - |\hat{Q}^{\text{GBT\_tGPD}}(\tau|\mathbf{x}) - Q^{\text{tGPD}}(\tau|\mathbf{x})|}{Q^{\text{tGPD}}(\tau|\mathbf{x})} \times 100, \tag{23}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

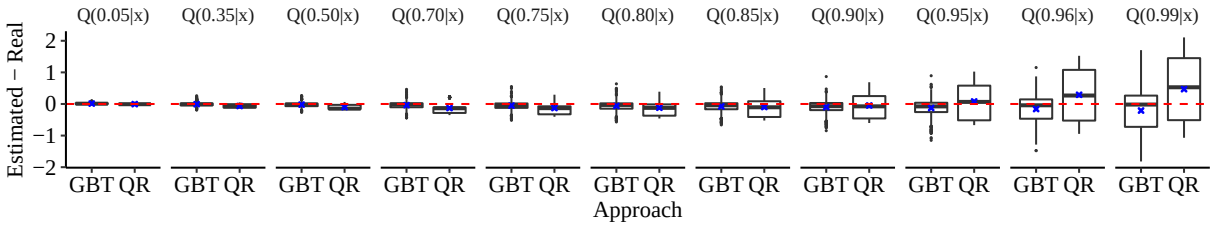


Figure 5 Comparison between GBT and QR (× represents the mean values).

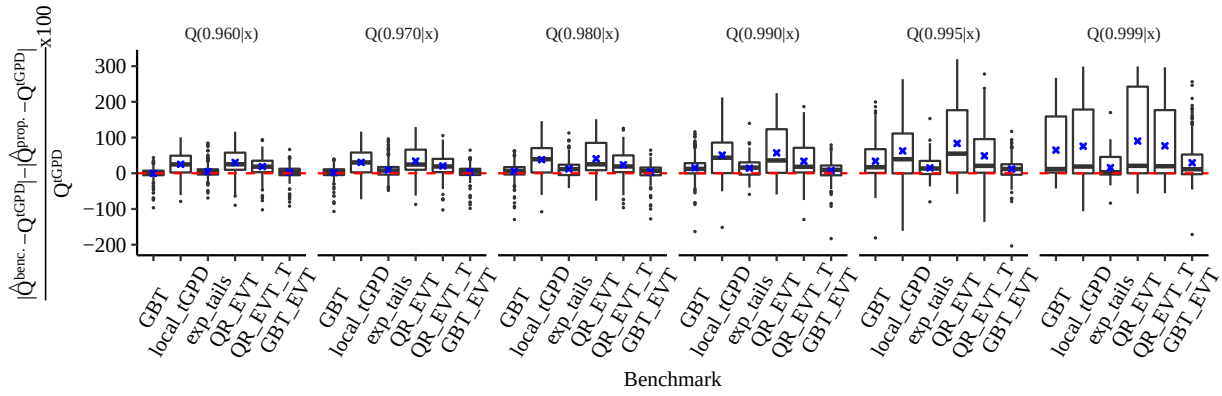


Figure 6 Improvement in terms of normalized absolute deviations, considering $(x_1^*, x_2^*) \in \{(0,-1),(0,0),(0,1)\}$ *(× represents the mean values).*

$\tau \in \boldsymbol{\tau}_e$. 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, GBTEVT, 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 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 DM test (Diebold and Mariano, 2002) 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 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.

II.5.2 Wind Power Data

II.5.2.1 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 7, 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 2 Mean quantile forecasts for $\tau \in \{0.99, 0.995, 0.999\}$.

	$\mathbf{x} = (0, -1), \boldsymbol{\xi} < 0$			$\mathbf{x} = (0, 0), \boldsymbol{\xi} = 0$			$\mathbf{x} = (1, 0), \boldsymbol{\xi} > 0$		
τ	0.99	0.995	0.999	0.99	0.995	0.999	0.99	0.995	0.999
$Q^{\text{tGPD}}(\tau)$	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)

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).

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 Andrade and Bessa (2017), from a NWP grid with 13×13 equally distributed points (Figure 7), 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

$$\sigma_{\text{time}}(t+h) = \sqrt{\frac{\sum_{i=-7}^{7}(mod_{t+h+i} - \overline{mod})^2}{14}}. \tag{24}$$

- *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:

- 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

$$\sigma_{\text{spatial}}(t+h) = \sqrt{\frac{\sum_{i=1}^{N_p}\left(x_{i,t+h} - \overline{x}_{t+h}\right)^2}{N_p - 1}}, \tag{25}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

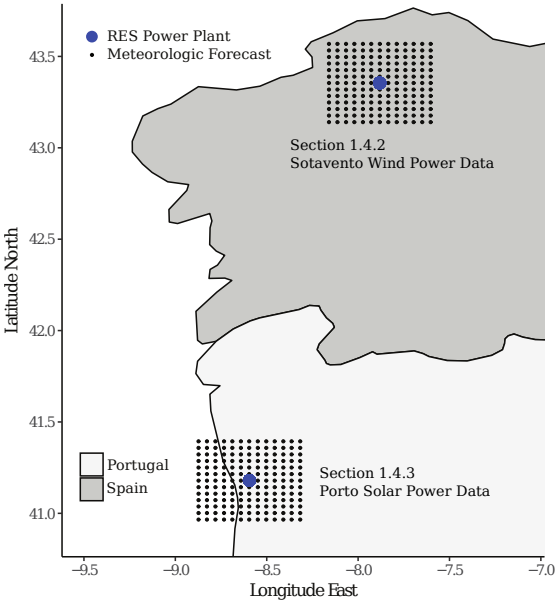


Figure 7 Geographical representation of data collection points for real datasets.

Table 3 Time period for training and testing folds (wind power dataset).

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

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 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.

II.5.2.2 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 Andrade and Bessa (2017). 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 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 8. 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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 4 Relative quantile loss improvement (%) over the baseline models (wind power dataset), considering the extreme quantiles τ_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
GBTEVT	12.20	9.06	9.33	5.03	9.75

Table 5 Quantile loss for each model (lower is better), with regard to the wind power dataset.

τ	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**	✓	**28.23**	✓	**8.01**	✓

† the proposed method

✓ statistically significant improvement against all others (DM test)

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 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 9 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 10, 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 GBTEVT, 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 11. The Exp_Tails model was chosen since it is the model with the lowest sharpness. This plot clearly shows that GBT_tGPD has a better

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

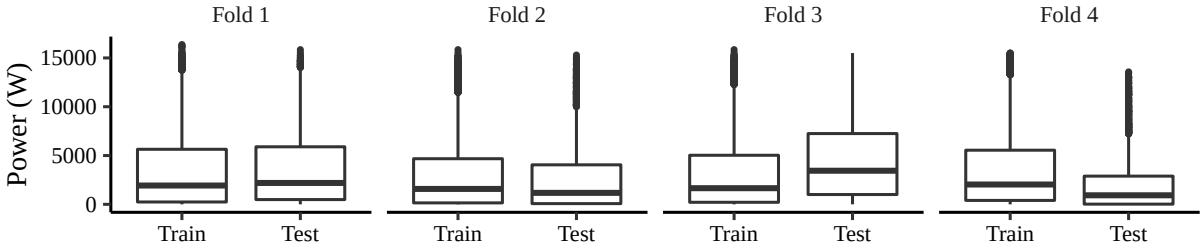


Figure 8 Boxplot for the wind power considering the division on Table 3.

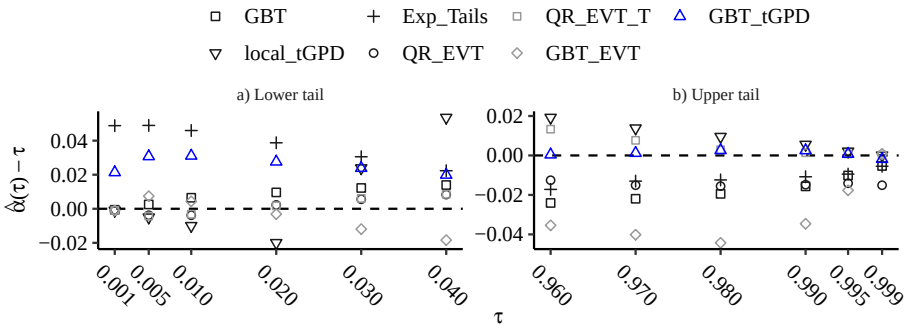


Figure 9 Deviation between nominal and empirical quantiles for wind power data, considering all folds. Dashed black line represents perfect calibration.

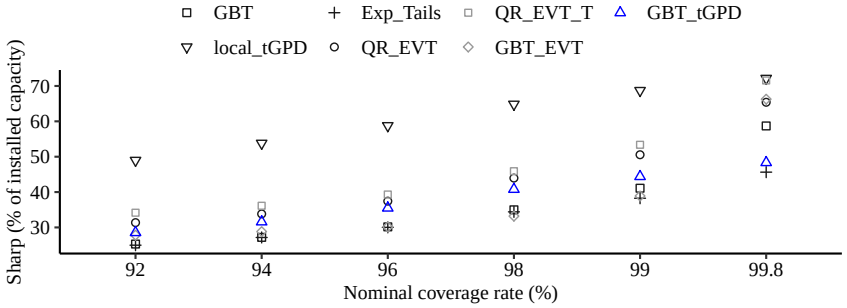


Figure 10 Sharpness results for wind power data, considering all folds.

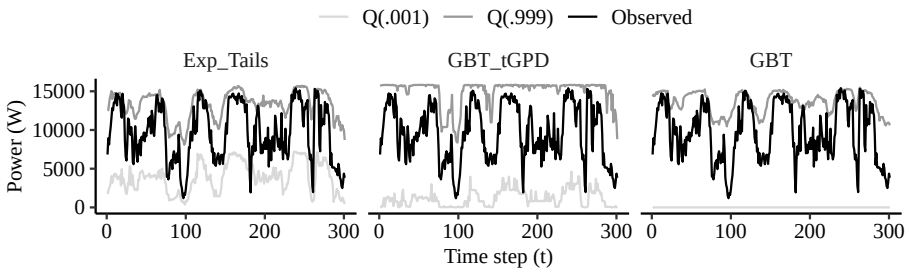


Figure 11 Illustrative forecast of extreme quantiles for GBT, Exp_Tails and GBT_tGPD, considering wind power data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 10. 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 11.

II.5.3 Solar Power Data

II.5.3.1 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 7. 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 Andrade and Bessa (2017), from a NWP grid with 13×13 equally distributed points (Figure 7), 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 (24).
- *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 (25).
- 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 6). Each train and test set consists of 12 and 5 months, respectively, allowing an evaluation under different conditions.

II.5.3.2 Results and Discussion Based in Andrade and Bessa (2017), GBT is used to estimate quantiles between 0.05 and 0.95. Again, the proposed model is used to estimate the quantiles $\tau_e = \{0.001, 0.005, 0.01, 0.02, 0.03, 0.04, 0.96, 0.97, 0.98, 0.99, 0.995, 0.999\}$.

The relative quantile score improvement obtained by GBTtGPD over the baseline models is provided in Table 7, considering nominal proportions τ_e. The GBTtGPD improvement over the local_tGPD, QR-based approaches and GBTEVT is greater than 14% for all folds. Regarding GBT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 6 Time period for training and testing folds (solar power dataset).

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

Table 7 Relative quantile loss improvement (%) over the baseline models (solar power dataset), considering the extreme quantiles τ_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

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 12. 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 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 13 complements the previous analysis by showing the calibration values for each model. For the lower tail, the GBTtGPD 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 14, these methods show that the better calibration comes at the cost of higher amplitude (i.e., lower sharpness).

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 15. 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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

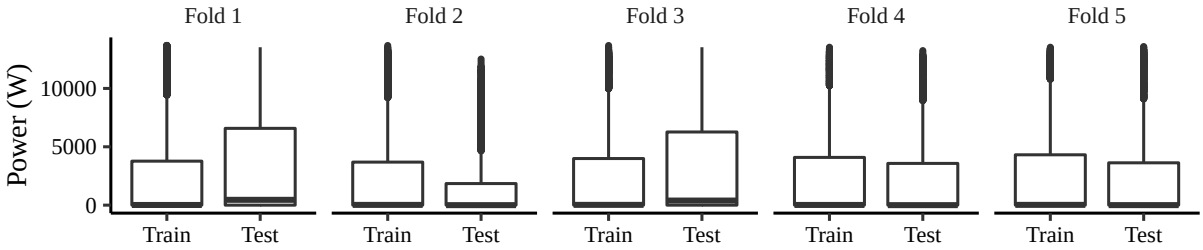


Figure 12 Boxplot for the solar power considering the division on Table 6.

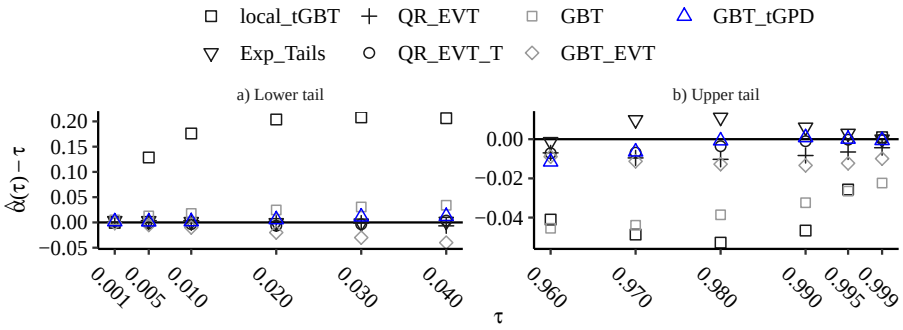


Figure 13 Deviation between nominal and empirical quantiles for solar power data, considering all folds. Dashed black line represents perfect calibration.

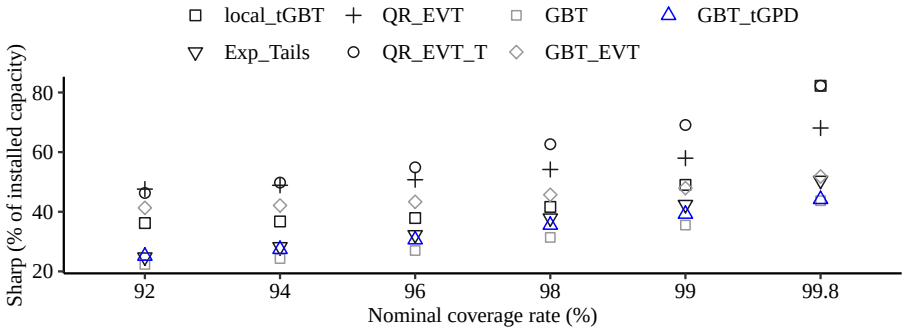


Figure 14 Sharpness results for solar power data, considering all folds.

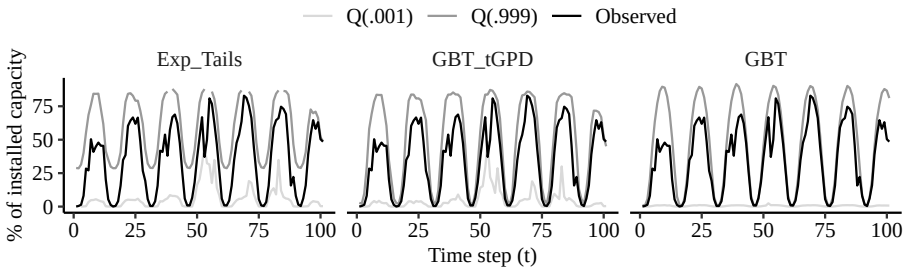


Figure 15 Illustrative forecast of extreme quantiles for GBT, Exp_Tails and GBT_tGPD, considering solar power data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 8 Quantile loss for each model (lower is better), with regard to the solar power dataset.

τ	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)

II.6 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.

III. Analysis of the privacy-preserving algorithms

III.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)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 (Tastu et al., 2012; Bessa et al., 2015b). The VAR model can also be used in short-term electricity price forecasting (Ziel and Weron, 2018). 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 Boyd et al. (2011). The combination of the VAR model and ADMM can be used jointly for collaborative forecasting (Cavalcante et al., 2017a), 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 (Yang et al., 2019).

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 (Aviv, 2003, 2007); (b) forecasting traffic flow (i.e., traffic speed) at different locations (Ravi and Al-Deek, 2009); (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 (Ahmad et al., 2016). 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 Nicholson et al. (2017) for more details) and to model non-linear relationships with past values (e.g. Li and Genton (2009) 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 (Du et al., 2004b), ridge linear regression (Karr et al., 2009), logistic regression (Wu et al., 2012), survival analysis (Lu et al., 2015), and aggregated statistics for time series data (Jia et al., 2014). 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 (Fienberg et al., 2009). 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 (Pinson, 2016b). However, in many cases, the data owners are unwilling to share their data even with a trusted central

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

node. This has motivated the development of data markets to monetize data and promote data sharing (Agarwal et al., 2019), which can be driven by blockchain and smart contracts technology (Kurtulmus and Daniel, 2018).

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) (Dwork and Smith, 2010). It requires computations insensitive to changes in any particular record or intermediate computations, thereby restricting data leaks through the results; see A. While computationally efficient and popular, these techniques invariably degrade the predictive performance of the model (Yang et al., 2019) and are not very effective, as we show in what follows.

This section 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.

In this work, 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 sub-problems, allowing each data owner to fit model coefficients separately.

The remainder of the paper is organized as follows: Section III.2 describes the state-of-the-art for collaborative privacy-preserving forecasting. Section III.3 describes the VAR model, as well as coefficients estimators, and critically evaluates state-of-the-art methods when applied to the VAR model. Solar energy time series data are used in the numerical analysis. Section III.4 offers a discussion and comparison of the presented approaches, and conclusions are presented in Section III.5.

III.2 Privacy-preserving Approaches

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, $i = 1, \ldots, k$. The column-wise joining of vectors and matrices is indicated by $[\mathbf{a}, \mathbf{b}]$ and $[\mathbf{A}, \mathbf{B}]$, respectively.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

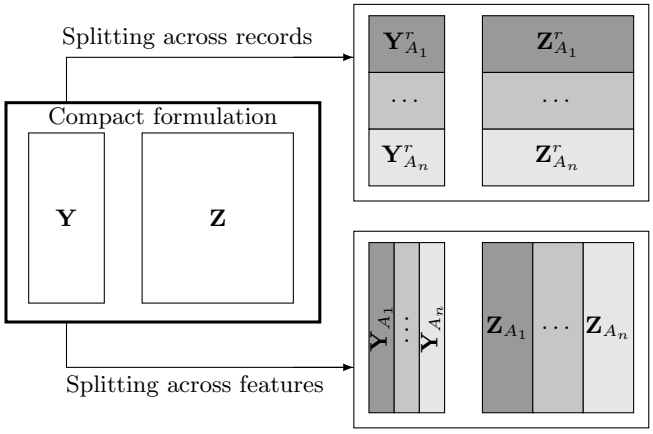


Figure 16 Common data division structures.

Furthermore, $\mathbf{Z} \in \mathbb{R}^{T \times M}$ 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 16 shows the two most common:

1. *Data split by records:* the data owners, represented as $A_i, i = 1, \ldots, n$, 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. $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$, $\mathbf{Y} = [\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$, 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 $\sum_{i=1}^{n} N_{A_i} = N$;

This section summarizes state-of-the-art approaches to deal with privacy-preserving collaborative forecasting methods. Section III.2.1 describes the methods that ensure confidentiality by transforming the data. Section III.2.2 presents and analyzes the secure multi-party protocols. Section III.2.3 describes the decomposition-based methods.

III.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). 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 (Zhou et al., 2009).

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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

III.2.1.1 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 Ma et al. (2017) the coefficients that optimize the linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}\ , \tag{26}$$

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 $\lambda > 0$,

$$\hat{\boldsymbol{\beta}}_{\text{ridge}} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}. \tag{27}$$

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

$$\hat{\boldsymbol{\beta}}_{\text{ridge}} = \mathbf{A}^{-1}\mathbf{b}\ , \tag{28}$$

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

$$\hat{\boldsymbol{\beta}}' = (\mathbf{MAN})^{-1}(\mathbf{M}(\mathbf{b} + \mathbf{Ar}))\ , \tag{29}$$

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 (Jain and Bhandare, 2011), 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 Papadimitriou et al. (2007), 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 (Papadimitriou et al., 2007). However, there are no privacy guarantees, since noise does not respect any formal definition, unlike differential privacy.

III.2.1.2 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 16– 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 (Mangasarian, 2012; Yu et al., 2008), which allows pre-multiplying the original data,

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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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

$$\mathbf{M}\mathbf{Z}^r = \mathbf{M}_{A_1}\mathbf{Z}^r_{A_1} + \cdots + \mathbf{M}_{A_n}\mathbf{Z}^r_{A_n}. \tag{30}$$

The same holds for the multiplication $\mathbf{M}\mathbf{Y}^r$, $\mathbf{M} \in \mathbb{R}^{k \times \sum_{i=1}^n T_{A_i}}$, $\mathbf{Z}^r \in \mathbb{R}^{\sum_{i=1}^n T_{A_i} \times M}$, $\mathbf{Y}^r \in \mathbb{R}^{\sum_{i=1}^n T_{A_i} \times N}$. This definition of $\mathbf{M}$ is possible because when multiplying $\mathbf{M}$ and $\mathbf{Z}^r$, the j-th column of $\mathbf{M}$ only multiplies the j-th 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 $\mathbf{M}\mathbf{Z}^r$ and target variable $\mathbf{M}\mathbf{Y}^r$ is

$$\hat{\mathbf{B}}_{\text{LS}} = \left((\mathbf{Z}^r)^\top \mathbf{Z}^r\right)^{-1} \left((\mathbf{Z}^r)^\top \mathbf{Y}^r\right), \tag{31}$$

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 (Zhou et al., 2009). Liu et al. (2008) 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 $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$ and $\mathbf{Y} = [\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$, i.e., data is split by features, it is not possible to define a matrix $\mathbf{M}^* = [\mathbf{M}^*_{A_1}, \ldots, \mathbf{M}^*_{A_n}] \in \mathbb{R}^{k \times T}$ and then privately compute $\mathbf{M}^*\mathbf{Z}$ and $\mathbf{M}^*\mathbf{Y}$, because as explained, the j-th column of $\mathbf{M}^*$ multiplies the j-th 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 $\mathbf{X}_{A_i}\mathbf{N}_{A_i}$ (Mangasarian, 2011), $i = 1, ..., n$, which is equivalent to post-multiplying the original dataset $\mathbf{X} = [\mathbf{X}_{A_1}, ..., \mathbf{X}_{A_n}]$ by $\mathbf{N} = [\mathbf{N}^\top_{A_1}, \ldots, \mathbf{N}^\top_{A_n}]^\top$, which represents the joining of $\mathbf{N}_{A_i}, i = 1, \ldots, n$, 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 $\mathbf{N}_{A_i}, i = 1, ..., n$. 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 $\mathbf{Z}\mathbf{N}_\mathbf{z}$ and $\mathbf{Y}\mathbf{N}_\mathbf{y}$. That is,the solution is given by

$$\hat{\mathbf{B}}'_{\text{LS}} = \arg\min_{\mathbf{B}} \left(\frac{1}{2}\|\mathbf{Y}\mathbf{N}_\mathbf{y} - \mathbf{Z}\mathbf{N}_\mathbf{z}\mathbf{B}\|_2^2\right), \tag{32}$$

where $\mathbf{Z}\mathbf{N}_\mathbf{z}$ and $\mathbf{Y}\mathbf{N}_\mathbf{y}$ are shared matrices. Two private matrices $\mathbf{N}_\mathbf{z} \in \mathbb{R}^{M \times s}$, $\mathbf{N}_\mathbf{y} \in \mathbb{R}^{N \times w}$ 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 (32) is given by

$$\hat{\mathbf{B}}'_{\text{LS}} = \left((\mathbf{Z}\mathbf{N}_\mathbf{z})^\top(\mathbf{Z}\mathbf{N}_\mathbf{z})\right)^{-1}\left((\mathbf{Z}\mathbf{N}_\mathbf{z})^\top(\mathbf{Y}\mathbf{N}_\mathbf{y})\right) = (\mathbf{N}_\mathbf{z})^{-1} \underbrace{(\mathbf{Z}^\top\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{Y}}_{= \arg\min_\mathbf{B}\left(\frac{1}{2}\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2\right)} \mathbf{N}_\mathbf{y}, \tag{33}$$

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 $\mathbf{Z}^\top\mathbf{Z}$, $\mathbf{Z} \in \mathbb{R}^{T \times M}$,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

PCA can be used to represent the data by L variables performing $\mathbf{Z}\mathbf{N}_L$, where $\mathbf{N}_L$ denotes the first L columns of $\mathbf{W}$, $L = 1, ..., M$. For data split by records, Dwork et al. (2014b) 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 (32) and (33), 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. (2015) 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).

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 (Fan and Xiong, 2014; Soria-Comas et al., 2017). 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.

III.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 (as discussed in Section III.2.2.1). In these approaches, the encryption of the data occurs while fitting the model (as discussed in Section III.2.2.2), instead of as a pre-processing step, as with the data transformation methods described in the previous section.

III.2.2.1 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. (2004b) 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

$$\hat{\mathbf{B}}_{\text{LS}} = (\mathbf{Z}^\top\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{Y}, \tag{34}$$

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

$$\mathbf{V}_a + \mathbf{V}_c = (\mathbf{Z}^\top\mathbf{Z}) \, , \tag{35}$$

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

$$(\mathbf{Z}^\top\mathbf{Z})^{-1} = (\mathbf{V}_a + \mathbf{V}_c)^{-1}. \tag{36}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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:

$$\mathbf{AC} = \mathbf{AMM^{-1}C} = \mathbf{A}[\mathbf{M}_{\text{left}}, \mathbf{M}_{\text{right}}] \begin{bmatrix} (\mathbf{M}^{-1})_{\text{top}} \\ (\mathbf{M}^{-1})_{\text{bottom}} \end{bmatrix} \mathbf{C} \tag{37}$$

$$= \mathbf{AM}_{\text{left}}(\mathbf{M}^{-1})_{\text{top}}\mathbf{C} + \mathbf{AM}_{\text{right}}(\mathbf{M}^{-1})_{\text{bottom}}\mathbf{C}\ , \tag{38}$$

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,

$$\mathbf{V}_a = \mathbf{AM}_{\text{left}}(\mathbf{M}^{-1})_{\text{top}}\mathbf{C}, \tag{39}$$

is derived by the first data owner, and

$$\mathbf{V}_c = \mathbf{AM}_{\text{right}}(\mathbf{M}^{-1})_{\text{bottom}}\mathbf{C}, \tag{40}$$

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

$$\mathbf{r}_a + \mathbf{r}_c = \mathbf{R}_a\mathbf{R}_c, \tag{41}$$

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

$$\mathbf{T} = (\mathbf{A} + \mathbf{R}_a)\mathbf{C} + (\mathbf{r}_c - \mathbf{V}_c)\ , \tag{42}$$

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

$$\mathbf{V}_a = \mathbf{T} + \mathbf{r}_a - \mathbf{R}_a(\mathbf{C} + \mathbf{R}_c)\ , \tag{43}$$

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

Finally, the $(\mathbf{A} + \mathbf{C})^{-1}$ 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 $(\mathbf{A} + \mathbf{C})^{-1}$. 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 III.3.3.2.

Another example of secure protocols for producing private matrices can be found in Karr et al. (2009). 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

$$\mathbf{w}_i^\top \mathbf{A}_j = \mathbf{0}\ , \tag{44}$$

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

$$\mathbf{A}^\top(\mathbf{I} - \mathbf{WW}^\top)\mathbf{C} = \mathbf{A}^\top\mathbf{C} - \underbrace{\mathbf{A}^\top\mathbf{WW}^\top\mathbf{C}}_{=\mathbf{0},\ \text{since}\ \mathbf{A}^\top\mathbf{W}=\mathbf{0}} = \mathbf{A}^\top\mathbf{C}\ , \tag{45}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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. (2009) 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,

$$\text{NLIE(Owner\#1)} = ks + kg. \tag{46}$$

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,

$$\text{NLIE(Owner\#2)} = ks + s(m - g). \tag{47}$$

Karr et al. (2009) 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:

$$|\text{NLIE(Owner\#1)} - \text{NLIE(Owner\#2)}| = 0\ , \tag{48}$$

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 Du et al. (2004b), is that $\mathbf{W}$ is simply generated by the first data owner, while the invertible matrix $\mathbf{M}$ proposed by Du et al. (2004b) needs to be agreed upon by both parties, which entails substantial communication costs when the number of records is high.

III.2.2.2 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. (2011) 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. (2013) 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. (2018) 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, Gascón et al. (2017) extended the approach of Nikolaenko et al. (2013) by designing a secure multi/two-party inner product.

Jia et al. (2018) explored a privacy-preserving data classification scheme with a support vector machine, to ensure that the data owners can successfully conduct data classification without

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 (2012) 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. (2018b) and Li et al. (2018) 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 (Hoogh, 2012). 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. (2017) 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 16, each i-th data owner observes the covariates $\mathbf{Z}^r_{A_i}$ and target variable $\mathbf{Y}^r_{A_i}$, $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times M}$, $\mathbf{Y}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times N}$, $i = 1, ..., n$. 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 (Hoogh, 2012; Zhao et al., 2019; Tran and Hu, 2019).

III.2.3 Decomposition-based Methods

In decomposition-based methods, problems are solved by breaking them up into smaller subproblems 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.

III.2.3.1 ADMM Method The ADMM 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 (Boyd et al., 2011). Let E be a convex forecast error function between the true values $\mathbf{Y}$ and the forecasted

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 (Boyd et al., 2011) solves the optimization problem

$$\min_{\mathbf{B}} E(\mathbf{B}) + R(\mathbf{B}), \tag{49}$$

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

$$\min_{\mathbf{B},\mathbf{H}} E(\mathbf{B}) + R(\mathbf{H}) \text{ subject to } \mathbf{AB} + \mathbf{CH} = \mathbf{D}\ , \tag{50}$$

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

$$L(\mathbf{B}, \mathbf{H}, \mathbf{U}) = E(\mathbf{B}) + R(\mathbf{H}) + \mathbf{U}^\top(\mathbf{AB}+\mathbf{CH} - \mathbf{D}) + \frac{\rho}{2}\|\mathbf{AB} + \mathbf{CH} - \mathbf{D}\|_2^2. \tag{51}$$

The quadratic term $\frac{\rho}{2}\|\mathbf{AB} + \mathbf{CH} - \mathbf{D}\|_2^2$ 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) (Boyd et al., 2011).

The ADMM solution is estimated by the following iterative system:

$$\begin{cases} \mathbf{B}^{k+1} := \arg\min_{\mathbf{B}} L(\mathbf{B}, \mathbf{H}^k, \mathbf{U}^k) \\ \mathbf{H}^{k+1} := \arg\min_{\mathbf{H}} L(\mathbf{B}^{k+1}, \mathbf{H}, \mathbf{U}^k) \\ \mathbf{U}^{k+1} := \mathbf{U}^k + \rho(\mathbf{AB}^{k+1} + \mathbf{CH}^{k+1} - \mathbf{D}). \end{cases} \tag{52}$$

For data split by records, 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 $\mathbf{Z}^r_{A_i} \in \mathbb{R}^{T_{A_i} \times M}$ of $\mathbf{Z} \in \mathbb{R}^{T \times M}$ corresponds to the local data of the $i-$th data owner, the coefficients $\mathbf{B}_{A_i} \in \mathbb{R}^{M \times N}$ are given by

$$\begin{aligned} &\arg\min_{\mathbf{\Gamma}} \sum_i E_i(\mathbf{B}_{A_i}) + R(\mathbf{H}) \\ &\text{s.t. } \mathbf{B}_{A_1} - \mathbf{H} = \mathbf{0},\ \mathbf{B}_{A_2} - \mathbf{H} = \mathbf{0}, \ldots,\ \mathbf{B}_{A_n} - \mathbf{H} = \mathbf{0}\ , \end{aligned} \tag{53}$$

where $\mathbf{\Gamma} = \{\mathbf{B}_{A_1}, \ldots, \mathbf{B}_{A_n}, \mathbf{H}\}$. 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 $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times M_{A_i}}$, and $\mathbf{B}$ into $\mathbf{B}_{A_i} \in \mathbb{R}^{M_{A_i} \times N}$. Auxiliary $\mathbf{H}_{A_i} \in \mathbb{R}^{T \times N}$ are introduced for the i-th 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

$$\begin{aligned} &\arg\min_{\mathbf{\Gamma}'} E(\sum_i \mathbf{H}_{A_i}) + \sum_i R(\mathbf{B}_{A_i}) \\ &\text{s.t. } \mathbf{Z}_{A_1}\mathbf{B}_{A_1} - \mathbf{H}_{A_1} = \mathbf{0},\ \mathbf{Z}_{A_2}\mathbf{B}_{A_2} - \mathbf{H}_{A_2} = \mathbf{0}, \ldots,\ \mathbf{Z}_{A_n}\mathbf{B}_{A_n} - \mathbf{H}_{A_n} = \mathbf{0}\ , \end{aligned} \tag{54}$$

where $\mathbf{\Gamma}' = \{\mathbf{B}_{A_1}, \ldots, \mathbf{B}_{A_n}, \mathbf{H}_{A_1}, \ldots, \mathbf{H}_{A_n}\}$. 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 (Dai et al., 2018). 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 (Bessa et al., 2018). An ADMM-based distributed LASSO algorithm, in which each data owner only commu nicates with its neighbor to protect data

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

privacy, is described by Mateos et al. (2010a), 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 (Zhang and Zhu, 2017). 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. (2019) 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 (Zhang et al., 2018).

Zhang et al. (2019) 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 $i-$th data owner shares TN new values, instead of $M_{A_i}N$ (from $\mathbf{B}_{A_i}^{k}$), $i, j = 1, ..., n$.

For data split by features, Zhang and Wang (2018b) 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 III.3. Furthermore, their method assumes that the central node knows the target matrix.

III.2.3.2 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, represented by $\mathbf{B}^k$, is given by

$$\mathbf{B}^{k+1} = \mathbf{B}^k - (\nabla^2 E(\mathbf{B}^k))^{-1} \nabla E(\mathbf{B}^k) \,, \tag{55}$$

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 (Nocedal and Wright, 2006).

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. (2007) 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

matrix $\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k$, making it possible to recover the local data $\mathbf{Z}_{A_i}$ after enough iterations (Fienberg et al., 2009).

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 (Wu et al., 2012). 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. (2015b) 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 ($\mathbf{Z}_{A_i}\mathbf{Z}_{A_i}^\top$), 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.

III.2.3.3 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, $\mathbf{B}^k$, is given by

$$\mathbf{B}^k = \mathbf{B}^{k-1} + \eta \nabla E(\mathbf{B}^{k-1}) \,, \tag{56}$$

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:

$$E(\mathbf{B}) = \frac{1}{2}\|\mathbf{Y} - f(\mathbf{B}, \mathbf{Z})\|^2. \tag{57}$$

With certain properties, convergence to a certain global minima can be guaranteed (Nesterov, 1998): (i) E is convex, (ii) ∇E is Lipschitz-continuous with constant L, i.e., for any $\mathbf{F}$, $\mathbf{G}$,

$$\|\nabla E(\mathbf{F}) - \nabla E(\mathbf{G})\|^2 \leq L\|\mathbf{F} - \mathbf{G}\|^2 \,, \tag{58}$$

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

Han et al. (2010) 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. Song et al. (2013) introduced differential privacy by adding random noise $\mathbf{W}$ in the $\mathbf{B}$ updates:

$$\mathbf{B}^k = \mathbf{B}^{k-1} + \eta\Big(\nabla E(\mathbf{B}^{k-1}) + \mathbf{W}\Big). \tag{59}$$

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.

III.3 Collaborative Forecasting with VAR

This section presents a privacy analysis of collaborative forecasting with the VAR model, a model for the analysis of multivariate time series. The VAR model is not only used for forecasting tasks in different domains (and with significant improvements over univariate autoregressive models), but

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

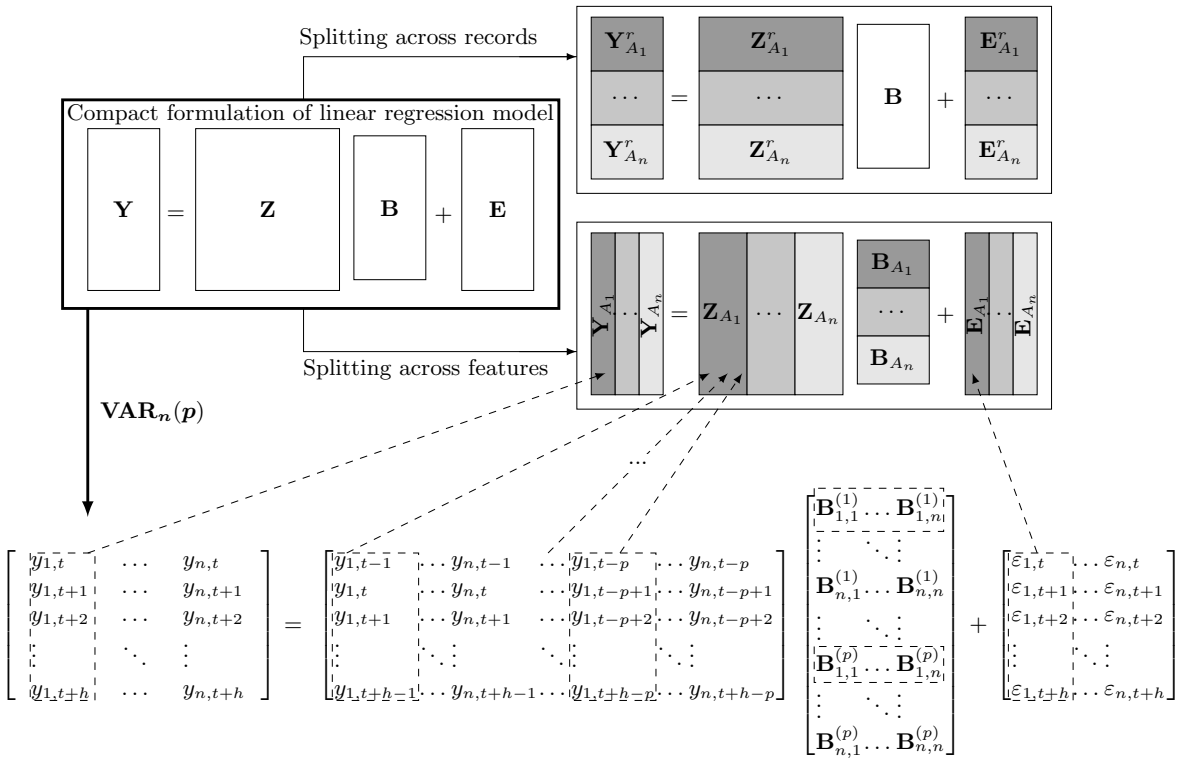


Figure 17 Common data division structures and VAR model.

also for structural inference, where the main objective is to explore certain assumptions about the causal structure of the data (Toda and Phillips, 1993). A variant with LASSO regularization is also covered. We critically evaluate the methods described in Section III.2 from a mathematical and numerical point of view in Section III.3.3. The solar energy time series dataset and R scripts are published in an online supplement (Gonçalves and Bessa, 2020).

III.3.1 VAR Model 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 VAR$_n(p)$, when the following relationship holds:

$$\mathbf{y}_t = \boldsymbol{\eta} + \sum_{\ell=1}^{p} \mathbf{y}_{t-\ell}\mathbf{B}^{(\ell)} + \boldsymbol{\varepsilon}_t \,, \tag{60}$$

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 VAR$_n(p)$ model reads as follows:

$$\mathbf{Y} = \mathbf{ZB} + \mathbf{E} \,, \tag{61}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

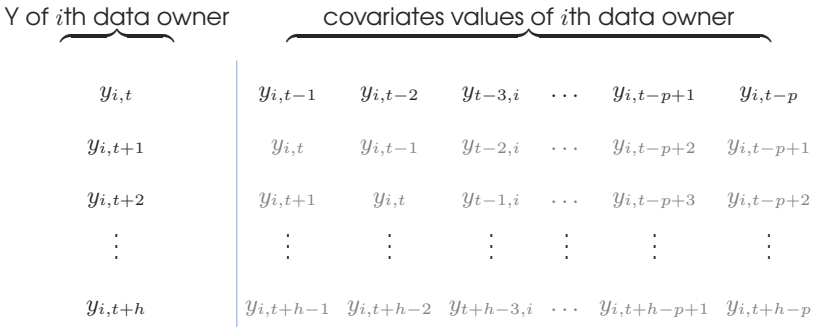


Figure 18 Illustration of the data used by the i-th data owner when fitting a VAR model.

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 $\mathbf{z}_t = [\mathbf{y}_{t-1}, \ldots, \mathbf{y}_{t-p}]$.

Notice that the VAR formulation adopted in this paper 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 (Figure 17) 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 18). 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 i-th data owner. Therefore, the covariates and target matrices are obtained by joining the individual matrices column-wise, i.e., $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$ 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 $\mathbf{B}_{A_i} \in \mathbb{R}^{p \times n}, i = 1, \ldots, n$.

III.3.2 Estimation in VAR Models

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 solution, given by

$$\hat{\mathbf{B}}_{\text{LS}} = \arg\min_{\mathbf{B}} \left(\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 \right) , \tag{62}$$

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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

estimated by the model. In the standard LASSO-VAR approach (see Nicholson et al. (2017) for different variants of the LASSO regularization in the VAR model), the coefficients are given by

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda \|\mathbf{B}\|_1 \right), \tag{63}$$

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

With the addition of the LASSO regularization term, the convex objective function in (63) becomes non-differentiable, limiting the variety of optimization techniques that can be employed. In this domain, the ADMM (which was described in Section III.2.3.1) is a widespread and computationally efficient technique that enables parallel estimations for data divided by features. The ADMM formulation of the non-differentiable cost function associated to LASSO-VAR model in (63) solves the following optimization problem:

$$\min_{\mathbf{B},\mathbf{H}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda \|\mathbf{H}\|_1 \right) \text{ subject to } \mathbf{H} = \mathbf{B} \,, \tag{64}$$

which differs from (63) 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 (Boyd et al., 2011) of this problem is

$$L_\rho(\mathbf{B}, \mathbf{H}, \mathbf{W}) = \frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda \|\mathbf{H}\|_1 + \mathbf{W}^\top (\mathbf{B} - \mathbf{H}) + \frac{\rho}{2} \|\mathbf{B} - \mathbf{H}\|_2^2 \,, \tag{65}$$

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

$$L_\rho(\mathbf{B}, \mathbf{H}, \mathbf{U}) = \frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \lambda \|\mathbf{H}\|_1 + \frac{\rho}{2} \|\mathbf{B} - \mathbf{H} + \mathbf{U}\|^2 - \frac{\rho}{2} \|\mathbf{U}\|^2 \,, \tag{66}$$

where $\mathbf{U} = (1/\rho)\mathbf{W}$ is the scaled dual variable associated with the constrain $\mathbf{B} = \mathbf{H}$. Hence, according to (52), the ADMM formulation for LASSO-VAR consists in the following iterations (Cavalcante et al., 2017a):

$$\begin{cases} \mathbf{B}^{k+1} := \arg\min_{\mathbf{B}} \left(\frac{1}{2} \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|_2^2 + \frac{\rho}{2} \|\mathbf{B} - \mathbf{H}^k + \mathbf{U}^k\|_2^2 \right) \\ \mathbf{H}^{k+1} := \arg\min_{\mathbf{H}} \left(\lambda \|\mathbf{H}\|_1 + \frac{\rho}{2} \|\mathbf{B}^{k+1} - \mathbf{H} + \mathbf{U}^k\|_2^2 \right) \\ \mathbf{U}^{k+1} := \mathbf{U}^k + \mathbf{B}^{k+1} - \mathbf{H}^{k+1}. \end{cases} \tag{67}$$

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 (63) becomes the following:

$$\arg\min_{\mathbf{\Gamma}} \left(\frac{1}{2} \|\mathbf{Y} - \sum_{i=1}^n \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2 + \lambda \sum_{i=1}^n \|\mathbf{B}_{A_i}\|_1 \right), \tag{68}$$

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

$$\arg\min_{\mathbf{\Gamma}'} \left(\frac{1}{2} \|\mathbf{Y} - \sum_{i=1}^n \mathbf{H}_{A_i}\|_2^2 + \lambda \sum_{i=1}^n \|\mathbf{B}_{A_i}\|_1 \right) \text{ s.t. } \mathbf{B}_{A_1}\mathbf{Z}_{A_1} = \mathbf{H}_{A_1}, \ \ldots, \ \mathbf{B}_{A_n}\mathbf{Z}_{A_n} = \mathbf{H}_{A_n} \,, \tag{69}$$

$\mathbf{\Gamma}' = \{\mathbf{B}_{A_1}, \ldots, \mathbf{B}_{A_n}, \mathbf{H}_{A_1}, \ldots, \mathbf{H}_{A_n}\}$, while the corresponding distributed ADMM formulation (Boyd et al., 2011; Cavalcante et al., 2017a) is the one presented in the system of equations (70),

$$\mathbf{B}_{A_i}^{k+1} = \arg\min_{\mathbf{B}_{A_i}} \left(\frac{\rho}{2} \|\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k + \overline{\mathbf{H}}^k - \overline{\mathbf{Z}\mathbf{B}}^k - \mathbf{U}^k - \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2 + \lambda \|\mathbf{B}_{A_i}\|_1 \right), \tag{70a}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

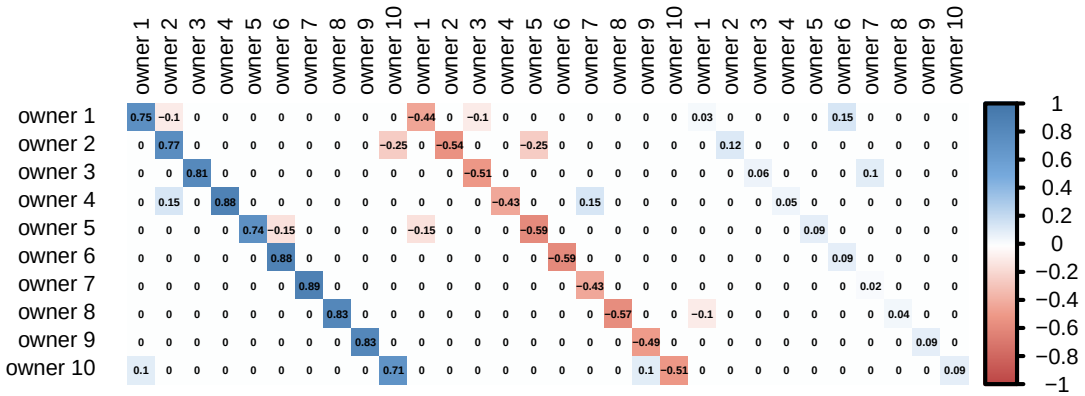


Figure 19 Transpose of the coefficient matrix used to generate the VAR with 10 data owners and 3 lags.

$$\overline{\mathbf{H}}^{k+1} = \frac{1}{N+\rho}\left(\mathbf{Y} + \rho\overline{\mathbf{ZB}}^{k+1} + \rho\mathbf{U}^{k}\right), \tag{70b}$$

$$\mathbf{U}^{k+1} = \mathbf{U}^{k} + \overline{\mathbf{ZB}}^{k+1} - \overline{\mathbf{H}}^{k+1}, \tag{70c}$$

where $\overline{\mathbf{ZB}}^{k+1} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{Z}_{A_j}\mathbf{B}_{A_j}^{k+1}$ and $\mathbf{B}_{A_i}^{k+1} \in \mathbb{R}^{p\times n}$, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T\times p}$,$\mathbf{Y} \in \mathbb{R}^{T\times n}$, $\overline{\mathbf{H}}^{k}$,$\mathbf{U} \in \mathbb{R}^{T\times n}$, $i = 1, ..., n$.

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. Thus, careful analysis is needed to determine whether a confidentiality breach will occur after enough iterations.

III.3.3 Privacy Analysis

III.3.3.1 Data Transformation with Noise Addition This section presents experiments with simulated data 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.

a) 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 19 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

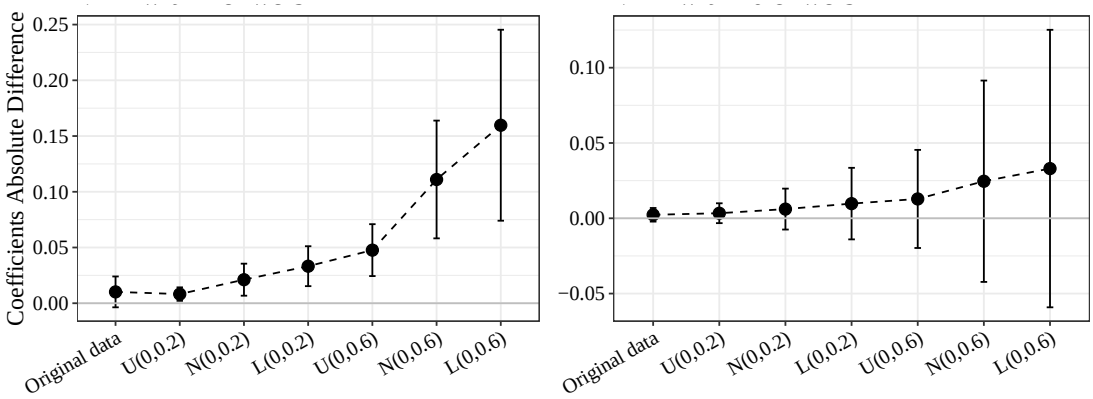


Figure 20 Mean ± standard deviation for the absolute difference between the real and estimated coefficients (left: VAR with 2 data owners, right: VAR with 10 data owners).

ε_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 III.2.3.1) was used to estimate the LASSO-VAR coefficients, considering two different noise characterizations, $b \in \{0.2, 0.6\}$.

Figure 20 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 fore- casting performance. Figure 21 illustrates the improvement of each estimated $\text{VAR}_2(2)$ 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 when using data without noise, seven data owners improved their forecasting performance, which was expected from the coefficient matrix in Figure 19. When Laplacian noise was applied to the data, only one data owner (the first one) improved its forecasting skill (when compared to the 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 $\text{VAR}_{10}(3)$, 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 $\text{VAR}_2(2)$ and $\text{VAR}_{10}(3)$ following the algorithm proposed by Ansley and Kohn (1986). Usually, the generated coefficient matrix has no null entries and the higher values are not necessarily found on

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

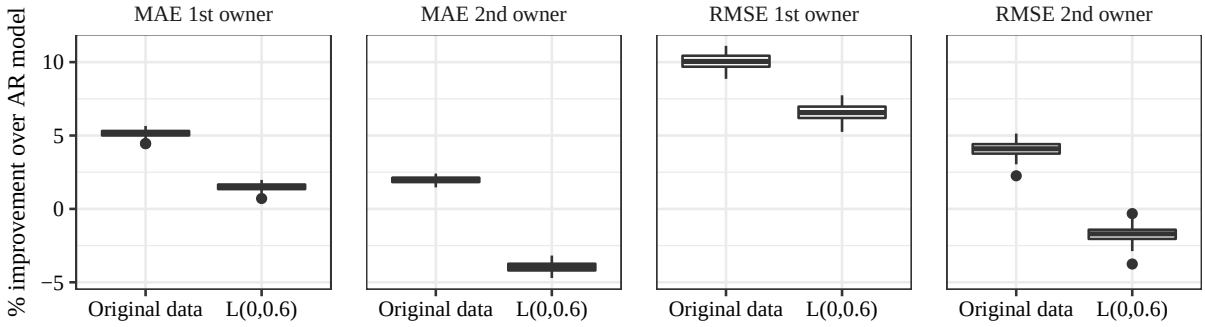
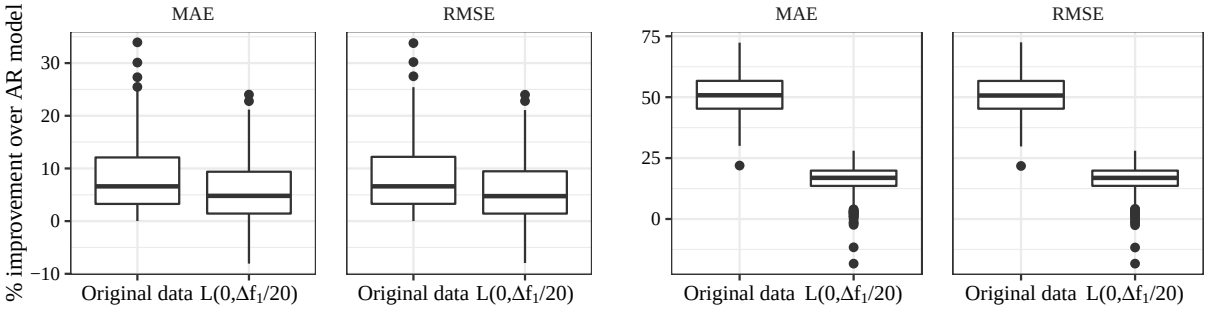


Figure 21 Improvement (%) of $VAR_2(2)$ model over AR(2) model, in terms of MAE and RMSE for synthetic data.



(a) $VAR_2(2)$ model. *(b) $VAR_{10}(3)$ model.*

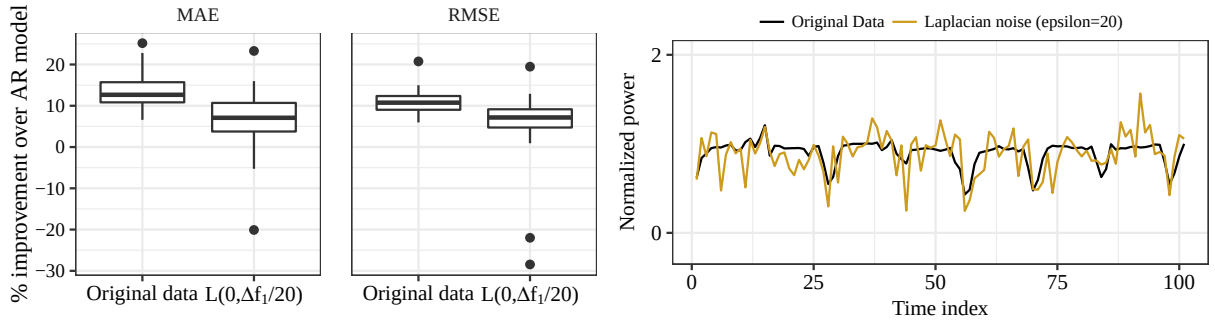
Figure 22 Improvement (%) of VAR model over AR model, in terms of MAE and RMSE for synthetic data.

diagonals. Figure 22 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.

b) Real Data: We also used 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 (2017), 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 station- ary, a normalization of the solar power was applied by using a clear-sky model (see Bacher et al. (2009)) 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 (1 h, 2 h, and 24 h). Figure 23 (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 23 (b), the corresponding VAR model was useless for 4 of the 44 data owners. 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.

III.3.3.2 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



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

(b) Example of the normalized time series.

Figure 23 Results for real case-study with solar power time series.

target $\mathbf{Y} = [\mathbf{Y}_{A_1}, \mathbf{Y}_{A_2}]$ is

$$\hat{\mathbf{B}}_{\text{LS}} = \left(\begin{bmatrix} \mathbf{Z}_{A_1}^\top \\ \mathbf{Z}_{A_2}^\top \end{bmatrix} [\mathbf{Z}_{A_1}, \mathbf{Z}_{A_2}] \right)^{-1} \left(\begin{bmatrix} \mathbf{Z}_{A_1}^\top \\ \mathbf{Z}_{A_2}^\top \end{bmatrix} [\mathbf{Y}_{A_1}, \mathbf{Y}_{A_2}] \right) \tag{71}$$

$$= \begin{pmatrix} \mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_1} & \mathbf{Z}_{A_1}^\top \mathbf{Z}_{A_2} \\ \mathbf{Z}_{A_2}^\top \mathbf{Z}_{A_1} & \mathbf{Z}_{A_2}^\top \mathbf{Z}_{A_2} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_1} & \mathbf{Z}_{A_1}^\top \mathbf{Y}_{A_2} \\ \mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_1} & \mathbf{Z}_{A_2}^\top \mathbf{Y}_{A_2} \end{pmatrix}, \tag{72}$$

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 III.2.2.1, the work of Du et al. (2004b) 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 18.

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, $i = 1, 2$. 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 Du et al. (2004b) for computing such matrices, then the information exchanged allows to recover data matrices.

Proof As in Du et al. (2004b), 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

inverse $\mathbf{M}^{-1}$. Then, the protocol stipulates that

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

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 Karr et al. (2009), a consequence of such data is that the assumption $rank\left((\mathbf{I} - \mathbf{W}\mathbf{W}^\top)\mathbf{C}\right) = m - g$ may still provide a sufficient number of linearly independent equations on the other data owner's data to recovering the latter's data.

III.3.3.3 ADMM Method and Central Node Zhang and Wang (2018b) offered a promising approach to dealing with the problem of private data during the ADMM iterative process described by (70). According to their approach, for each iteration k, each data owner i communicates local results, $\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^{k+1}$, to the central node, $\mathbf{Z}_{A_i} \in \mathbb{R}^{T \times p}, \mathbf{B}_{A_i}^{k+1} \in \mathbb{R}^{p \times n}, i = 1, \ldots, n$. Then, the central node computes the intermediate matrices in (70b)-(70c) 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 (70a). Figure 24 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.

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 Zhang et al. (2019) to solve the LASSO-VAR model in* (70)*, the central agent can recover the sensible data after*

$$k = \left\lceil \frac{Tp}{Tn - pn} \right\rceil \tag{73}$$

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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

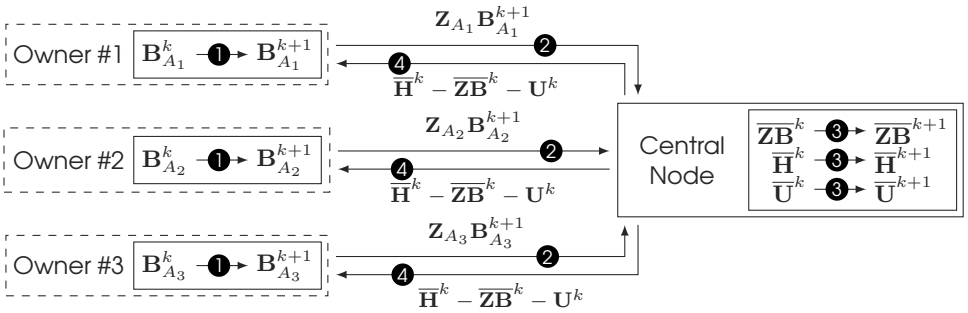


Figure 24 Distributed ADMM LASSO-VAR with a central node and 3 data owners (related to the algorithm in (70)).

Proof Using the notation of Section III.3.1, 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, $i = 1, ..., n$. Given that, the solution of the inequality

$$Tnk \geq pnk + Tp\,, \tag{74}$$

in k suggests that a confidentiality breach can occur after

$$k = \left\lceil \frac{Tp}{Tn - pn} \right\rceil \tag{75}$$

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 Zhang et al. (2019) to solve the LASSO-VAR model in (70), the data owners can recover sensible data from competitors after*

$$k = \left\lceil \frac{Tn + (n-1)(Tp+T)}{Tn - (n-1)pn} \right\rceil \tag{76}$$

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 (70b) and (70c), it is possible to

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

perform some simplifications in which $\mathbf{U}^k$ and $\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k$ becomes (77) and (78), respectively:

$$\begin{aligned}\mathbf{U}^k &\overset{(70c)}{=} \mathbf{U}^{k-1} + \overline{\mathbf{ZB}}^k - \overline{\mathbf{H}}^k = \mathbf{U}^{k-1} + \overline{\mathbf{ZB}}^k - \underbrace{\frac{1}{N+\rho}\left(\mathbf{Y} + \rho\overline{\mathbf{ZB}}^k + \rho\mathbf{U}^{k-1}\right)}_{=\overline{\mathbf{H}}^k, \text{ according to (70b)}} \\ &= \left[1 - \frac{\rho}{N+\rho}\right]\mathbf{U}^{k-1} + \left[1 - \frac{\rho}{N+\rho}\right]\overline{\mathbf{ZB}}^k - \frac{1}{N+\rho}\mathbf{Y}\,,\end{aligned} \tag{77}$$

$$\overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k - \mathbf{U}^k = \underbrace{\frac{1}{N+\rho}\left(\mathbf{Y} + \rho\overline{\mathbf{ZB}}^k + \rho\mathbf{U}^{k-1}\right)}_{=\overline{\mathbf{H}}^k, \text{ according to (70b)}} - \overline{\mathbf{ZB}}^k - \mathbf{U}^k. \tag{78}$$

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 i-th data owner generates $\mathbf{B}_{A_i}^1 \in \mathbb{R}^{p\times n}$, $i \in \{1, ..., n\}$.
2. *Iteration #1:* The central node receives $\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^1$ 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
$$\underbrace{\mathbf{U}^0}_{\in\mathbb{R}^{T\times n}}, \quad \underbrace{\mathbf{B}_{A_2}^1, ..., \mathbf{B}_{A_n}^1}_{n-1 \text{ matrices} \in \mathbb{R}^{p\times n}}, \quad \underbrace{\mathbf{Z}_{A_2}, ..., \mathbf{Z}_{A_n}}_{n-1 \text{ matrices} \in \mathbb{R}^{T\times p}},$$
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 $\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^2$ 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 estimate.

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

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

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

$$Tnk \geq Tn + (n-1)[kpn + Tp + T]\,, \tag{79}$$

suggests that a confidentiality breach may occur after

$$k = \left\lceil \frac{Tn + (n-1)(Tp+T)}{Tn - (n-1)pn} \right\rceil \tag{80}$$

iterations. □

Figure 25 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).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

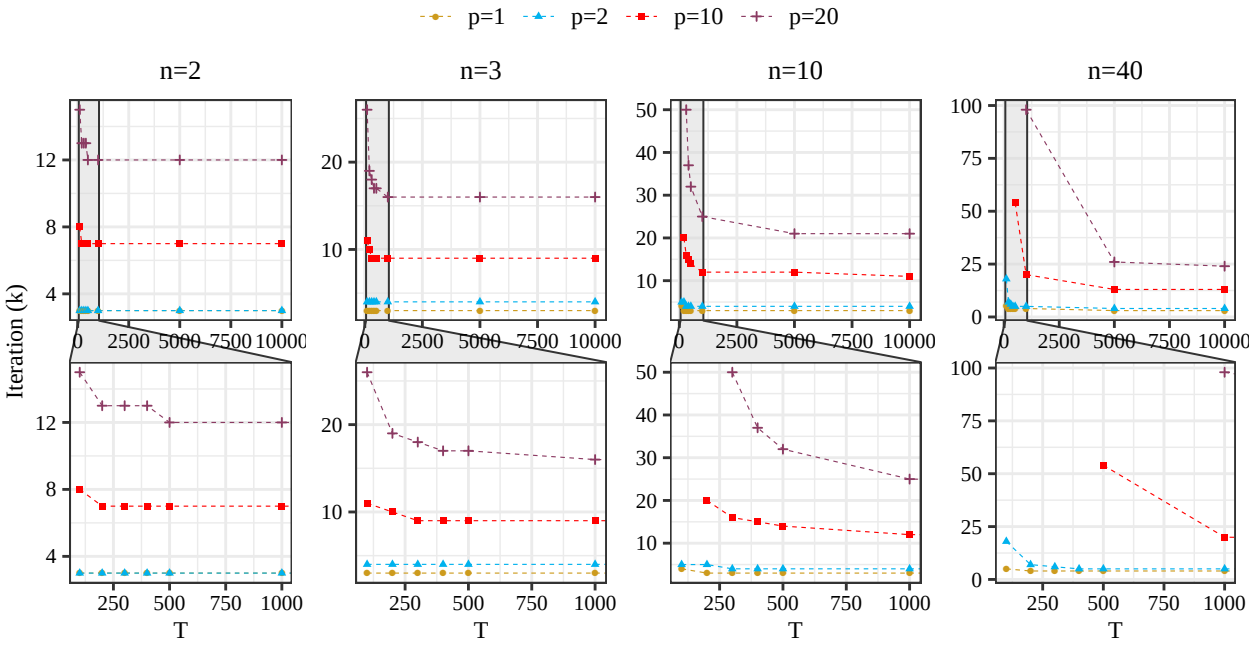


Figure 25 Number of iterations until a possible confidentiality breach, considering the centralized ADMM-based algorithm in (Zhang et al., 2019).

III.3.3.4 ADMM Method and Noise Mechanisms The target matrix $\mathbf{Y} = [\mathbf{Y}_{A_1}, \ldots, \mathbf{Y}_{A_n}]$ corresponds to the sum of private matrices $\mathbf{I}_{\mathbf{Y}_{A_i}} \in \mathbb{R}^{T \times n}$. That is,

$$\underbrace{\begin{bmatrix} y_{1,t} & y_{2,t} & \cdots & y_{n,t} \\ y_{1,t+1} & y_{2,t+1} & \cdots & y_{n,t+1} \\ y_{1,t+2} & y_{2,t+2} & \cdots & y_{n,t+2} \\ \vdots & \ddots & \vdots & \\ y_{1,t+h} & y_{2,t+h} & \cdots & y_{n,t+h} \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} y_{1,t} & 0 & \cdots & 0 \\ y_{1,t+1} & 0 & \cdots & 0 \\ y_{1,t+2} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \\ y_{1,t+h} & 0 & \cdots & 0 \end{bmatrix}}_{\mathbf{I}_{Y_{A_1}}} + \underbrace{\begin{bmatrix} 0 & y_{2,t} & \cdots & 0 \\ 0 & y_{1,t+1} & \cdots & 0 \\ 0 & y_{1,t+2} & \cdots & 0 \\ \vdots & \ddots & \vdots & \\ 0 & y_{1,t+h} & \cdots & 0 \end{bmatrix}}_{\mathbf{I}_{Y_{A_2}}} + \cdots + \underbrace{\begin{bmatrix} 0 & 0 & \cdots & y_{n,t} \\ 0 & 0 & \cdots & y_{n,t+1} \\ 0 & 0 & \cdots & y_{n,t+2} \\ \vdots & \ddots & \vdots & \\ 0 & 0 & \cdots & y_{n,t+h} \end{bmatrix}}_{\mathbf{I}_{Y_{A_n}}}, \quad (81)$$

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 i-th data owner and $[\mathbf{I}_{\mathbf{Y}_{A_i}}]_{i,j}=0$ otherwise.

Since the LASSO-VAR ADMM formulation is provided by (70), 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 (70a). The combination of (77) with (81) can be used to rewrite $\mathbf{U}^k$ as

$$\mathbf{U}^k = \left[1 - \frac{\rho}{N+\rho}\right]\mathbf{U}^{k-1} + \sum_{i=1}^{n} \underbrace{\left[1 - \frac{\rho}{N+\rho}\right]\frac{1}{n}\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k - \frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}}}_{\text{information from owner } i}, \quad (82)$$

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

$$\begin{aligned} \overline{\mathbf{H}}^k - \overline{\mathbf{ZB}}^k &= \frac{1}{N+\rho}\mathbf{Y} + \left[\frac{\rho}{N+\rho} - 1\right]\overline{\mathbf{ZB}}^k + \frac{\rho}{N+\rho}\mathbf{U}^{k-1} - \mathbf{U}^k \\ &= \sum_{i=1}^{n} \underbrace{\left(\frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right]\frac{1}{n}\mathbf{Z}_{A_i}\mathbf{B}_{A_i}^k\right)}_{\text{information from owner } i} + \frac{\rho}{N+\rho}\mathbf{U}^{k-1} - \mathbf{U}^k, \end{aligned} \quad (83)$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

where

$$\mathbf{Y} = \sum_{i=1}^{n} \mathbf{I}_{\mathbf{Y}_{A_i}}, \tag{84}$$

$$\overline{\mathbf{ZB}}^{k+1} = \sum_{i=1}^{n} \frac{\rho}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k+1}. \tag{85}$$

By analyzing (82) and (83), it is possible to verify that data owner i only needs to share

$$\frac{1}{N+\rho} \mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right] \frac{1}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k}, \tag{86}$$

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

$$\mathbf{I}_{\mathbf{Y}_{A_i}} + \mathbf{W}_{1,A_i} \text{ and } \mathbf{Z}_{A_i} + \mathbf{W}_{2,A_i}\ , \tag{87}$$

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

$$\mathbf{B}_{A_i}^{k} + \mathbf{W}_{3,A_i}\ , \tag{88}$$

(iii) adding noise to the intermediate matrix (86),

$$\frac{1}{N+\rho} \mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right] \frac{1}{n} \mathbf{Z}_{A_i} \mathbf{B}_{A_i}^{k} + \mathbf{W}_{4,A_i}. \tag{89}$$

The addition of noise to the data itself (87) was empirically analyzed in Section III.3.3.1. 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 (88)*;*

(ii) adding noise to the exchanged intermediate matrix, as described in (89)*.*

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

$$k = \left\lceil \frac{Tn + (n-1)(Tp+T)}{Tn - (n-1)pn} \right\rceil \tag{90}$$

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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

(i) Owner #1 can estimate $\mathbf{B}_{A_i}$, without distinguishing between $\mathbf{B}_{A_i}$ and $\mathbf{W}_{3,A_i}$ in (88), 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 $\overline{\mathbf{H}'}^k - \overline{\mathbf{Z}\mathbf{B}'}^k - \mathbf{U}'^k \in \mathbb{R}^{T\times n}$ (Tn values) and does not know

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

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

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

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 (89) as

$$\frac{1}{N+\rho}\mathbf{I}_{\mathbf{Y}_{A_i}} + \left[\frac{\rho}{N+\rho} - 1\right]\frac{1}{n}\mathbf{Z}_{A_i}\Big[\underbrace{\mathbf{B}^k_{A_i} + \left[\frac{\rho}{N+\rho} - 1\right]^{-1}\mathbf{Z}^{-1}_{A_i}\mathbf{W}_{4,A_i}}_{=\mathbf{B}'_{A_i}}\Big]. \tag{91}$$

□

III.4 Discussion

Table 9 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 schemes (Hoogh, 2012; Zhao et al., 2019; Tran and Hu, 2019). Regarding some protocols, such as secure multiparty computation through homomorphic cryptography, communication complexity grows exponentially with the number of records (Rathore et al., 2015).

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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 9 Summary of state-of-the-art privacy-preserving approaches.

		Split by features	Split by records
Data Transformation		Mangasarian (2011)	Mangasarian (2012), Yu et al. (2008), Dwork et al. (2014b)
Secure Multi-party Computation	Linear Algebra	Du et al. (2004b), Karr et al. (2009), Zhu et al. (2015), Fan and Xiong (2014)*, Soria-Comas et al. (2017)	Zhu et al. (2015), Aono et al. (2017)
	Homomorphic cryptography	Yang et al. (2019), Hall et al. (2011), Gascón et al. (2017), Slavkovic et al. (2007)	Yang et al. (2019), Hall et al. (2011), Nikolaenko et al. (2013), Chen et al. (2018), Jia et al. (2018), Slavkovic et al. (2007)
Decomposition-based Methods	Pure	Pinson (2016b), Zhang and Wang (2018b)	Wu et al. (2012), Lu et al. (2015), Ahmadi et al. (2010), Mateos et al. (2010a)
	Linear Algebra	Li et al. (2015b), Han et al. (2010)	Zhang and Zhu (2017), Huang et al. (2019), Zhang et al. (2018)
	Homomorphic cryptography	Yang et al. (2019), Li and Cao (2012)*, Liu et al. (2018b)*, Li et al. (2018)*, Fienberg et al. (2009), Mohassel and Zhang (2017)	Yang et al. (2019), Zhang et al. (2019), Fienberg et al. (2009), Mohassel and Zhang (2017)

* secure data aggregation.

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 i-th 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 i-th data owner sends $y_{i,t}[\mathbf{B}^{(1)}]_{i,j}$ for the j-th 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 i-th data owner. In other words, after h timestamps, the j-th 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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

3. In the LASSO-VAR models with p non-consecutive lags, $p_1, \ldots, p_p$, 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 section 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 (Tastu et al., 2012), component-wise gradient boosting (Bessa et al., 2015c), a VAR that estimates the location parameter (mean) of data transformed by a logit-normal distribution (Dowell and Pinson, 2015), linear quantile regression with LASSO regularization (Agoua et al., 2018), 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 Dowell and Pinson (2015) can be influenced by the confidentiality breaches discussed throughout this section, 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 (Zhang et al., 2019). However, as discussed in Section III.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 (Agoua et al., 2018), which may be solved through homomorphic encryption, despite being computationally demanding for high-dimensional multivariate time series.

III.5 Concluding Remarks

This section 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 multi-party computation, and decomposition of the optimization problem into sub-problems.

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 (Yang et al., 2019). In our experiments, there was clear model degradation even though the data continued to provide relevant information (Section III.3.3.1). 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 (33) in Section III.2.1.2, this transformation does not

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 possibly of breaching data confidentiality.

IV. Federated learning for renewable energy forecasting

IV.1 Introduction

The forecast skill of RES has improved over the past two decades through R&D activities across the complete model chain, i.e., from NWP to statistical learning methods that convert weather variables into power forecasts (Sweeney et al., 2020a). 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 (Sweeney et al., 2020a). Indeed, for very short-term lead times (from 15 minutes to 6 hours ahead), the VAR model, when compared to univariate time series models, has shown competitive results for wind (Tastu et al., 2014) and solar (Bessa et al., 2015b) 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 (Zhu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

et al., 2020). 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 (Zhao et al., 2018), (b) uncertainty forecasting (Dowell and Pinson, 2015), (c) distributed learning (Cavalcante et al., 2017b), (d) online learning (Messner and Pinson, 2019), 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 ADMM (Zhang and Wang, 2018b). 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 III.3.3.3. Moreover, the central node can also recover the original and private data. Sommer et al. (2021) 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. (2011) described a method based on local and global analog-search (i.e., template matching) that uses solar power time series from neighboring sites. 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 section, 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 section, 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 (Li et al., 2013) or using principal component analysis with differential privacy (Dwork et al., 2014a), (b) *secure multi-party computation*, such as linear algebra protocols (Du et al., 2004a) or homomorphic encryption (encrypting the original data in a way that arithmetic operations in the public space do not compromise the encryption (Liu et al., 2018a)), and (c) *decomposition-based methods* like the ADMM (Mateos et al., 2010b) or the distributed Newton-Raphson method (Li et al., 2015a). 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 section 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 (Berdugo et al., 2011)). 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 when compared to the ADMM implementation by Zhang and Wang (2018b) and Sommer et al.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

(2021)).

The remaining of this section is organized as follows: Subsection IV.2 describes the distributed learning framework. Subsection IV.3 formulates a novel privacy-preserving LASSO-VAR model. Then, two case studies with solar and wind energy data are considered in Subsection IV.4. Concluding remarks are provided in Subsection IV.5.

IV.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 (Elsinga and van Sark, 2017); smart cities characterized by an increasing number of sensors and devices installed at houses, buildings, and transportation network Tascikaraoglu (2018).

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 IV.3 fully guarantees these three core principles.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

IV.3 Privacy-preserving Distributed LASSO-VAR

Using the notation in Section III.3.1, 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., $\mathbf{Z} = [\mathbf{Z}_{A_1}, \ldots, \mathbf{Z}_{A_n}]$ 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 $\mathbf{B}_{A_i} \in \mathbb{R}^{p\times n}, i \in \{1, \ldots, n\}$.

When applying the collaboration schemes discussed in Section IV.2 to the distributed ADMM LASSO-VAR formulation described in (70), at each iteration k each agent determines and transmits (70a), given by

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

and then it is up to the central agent or peers (depending on the adopted structure) to compute the quantities in (70b), i.e.,

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

and (70c), i.e.,

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

As shown in the previous section, although there is no direct exchange of private data, the computation of (70b) and (70c) 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 IV.3.1) with the distributed ADMM for the generalized LASSO-VAR model (Section IV.3.2), which had been previously formulated in Section III.3.2. Communication issues (Section IV.3.5) are also addressed since they are common in distributed systems.

IV.3.1 Data Transformation with Multiplicative Randomization

Multiplicative randomization of the data Chen and Liu (2008) 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., $\mathbf{M}\mathbf{X}$, 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., $\mathbf{X}\mathbf{Q}$, 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 (92), where matrices in diagonal, $\mathbf{Q}_{A_i} \in \mathbb{R}^{s\times s}$, are privately defined by agent $i \in \{1, \ldots, n\}$. Then, agents post-multiply their data without sharing $\mathbf{Q}_{A_i}$, since

$$\underbrace{\left[\mathbf{X}_{A_1}, \ldots, \mathbf{X}_{A_n}\right]}_{=\mathbf{X}} \underbrace{\begin{bmatrix} \mathbf{Q}_{A_1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{Q}_{A_n} \end{bmatrix}}_{=\mathbf{Q}} = \left[\mathbf{X}_{A_1}\mathbf{Q}_{A_1}, \ldots, \mathbf{X}_{A_n}\mathbf{Q}_{A_n}\right]. \tag{92}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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

$$\mathbf{M} = \mathbf{M}_{A_1}\mathbf{M}_{A_2}\dots\mathbf{M}_{A_n}, \tag{93}$$

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

$$\mathbf{MX} = [\underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_1}}_{=\mathbf{MX}_{A_1}},\dots,\underbrace{\mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_n}}_{=\mathbf{MX}_{A_n}}]. \tag{94}$$

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 III.3.3.2 for a proof). The calculation of $\mathbf{MX}_{A_i}$ 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: $\mathbf{MX}_{A_i} = \mathbf{M}_{A_1}\dots\mathbf{M}_{A_n}\mathbf{X}_{A_i}$

1: **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,

$$\mathbf{W}_{A_i} = [\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i}. \tag{95}$$

2: Agent n receives $\mathbf{W}_{A_i}, \forall i$.
3: Agent n shares $\mathbf{M}_{A_n}\mathbf{W}_{A_i}$ with the $(n-1)$-th agent.
4: **for** agent $j = n-1,\dots,1$ **do**
5: Agent j receives $\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$, and
6: **if** $j > 1$ **then**
7: shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent $j-1$
8: **else**
9: shares $\mathbf{M}_{A_j}\left(\prod_{k=j+1}^{n}\mathbf{M}_{A_k}\right)\mathbf{W}_{A_i}$ with agent i
10: **end if**
11: **end for**
12: Agent i receives $\mathbf{MW}_{A_i}$ from the 1-st agent and recovers $\mathbf{MX}_{A_i}$,

$$[\mathbf{MX}_{A_i}, \mathbf{MC}_{A_i}] = \mathbf{MW}_{A_i}\mathbf{D}_{A_i}^{-1}. \tag{96}$$

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 B.

IV.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 $\mathbf{Y}^\top\mathbf{Y}$.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 $i \in \{1, \ldots, n\}$.

IV.3.2.1 Formulation Let $\mathbf{ZQ}$ be the covariate matrix obtained through (92) 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.,

$$\arg\min_{\mathbf{B}} \Big(\frac{1}{2}\|\mathbf{Y} - \sum_i \mathbf{Z}_{A_i}\mathbf{B}_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{B}_{A_i}\|_1\Big), \tag{97}$$

is

$$\arg\min_{\mathbf{B}^{\text{post}}} \Big(\frac{1}{2}\|\mathbf{Y} - \sum_i \mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}^{\text{post}}_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{Q}_{A_i}\mathbf{B}^{\text{post}}_{A_i}\|_1\Big). \tag{98}$$

After a little algebra, the relation between the ADMM solution for (97) and (98) is

$$\mathbf{B}^{\text{post}\,k+1}_{A_i} = \mathbf{Q}_{A_i}\mathbf{B}^{k+1}_{A_i}, \tag{99}$$

suggesting coefficients' privacy since the original $\mathbf{B}$ is no longer used. However, the limitations identified in the previous section for (97) are valid for (98). 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

$$\arg\min_{\mathbf{B}'} \Big(\frac{1}{2}\|\mathbf{MY} - \sum_i \mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_2^2 + \lambda \sum_i \|\mathbf{Q}_{A_i}\mathbf{B}'_{A_i}\|_1\Big), \tag{100}$$

preserves the relation between the original time series if $\mathbf{M}$ is orthogonal, i.e., $\mathbf{M}\mathbf{M}^\top = \mathbf{I}$. 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 $\mathbf{B}'_{A_i}$,

$$\begin{aligned}\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^\top\Big[\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{MZQB}'}^{k} + \ldots\Big] = \\ \mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\Big[\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{ZQB}'}^{k} + \ldots\Big].\end{aligned} \tag{101}$$

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

$$\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}\Big[\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{k}_{A_i} - \overline{\mathbf{MZQB}'}^{k} + \ldots\Big]. \tag{102}$$

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

$\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$ 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 $\mathbf{Y}_{A_i}^\top\mathbf{M}^{-1}$ 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 $\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}$ 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 B for determination of the optimal r). Appendix C presents an analysis

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}{\mathbf{B}'}_{A_i}^{k+1}$ and computes (104)–(106).

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: ${\mathbf{B}'}_{A_i}^0$, $\overline{\mathbf{H}}^0$, $\mathbf{U}^0=\mathbf{0}, \rho\in\mathbb{R}^+, k=0$
for agent $i=1,\dots,n$ **do**
  $\mathbf{P}_{A_i}=\left((\mathbf{Z}_{A_i}\mathbf{Q}_{A_i})^\top(\mathbf{Z}_{A_i}\mathbf{Q}_{A_i})+\rho\mathbf{Q}_{A_i}^\top\mathbf{Q}_{A_i}\right)^{-1}$
end for
while stopping criteria not satisfied **do**
  for agent $i=1,\dots,n$ **do**
    Initialization: $\widetilde{\mathbf{B}}_{A_i}^0$, $\widetilde{\overline{\mathbf{H}}}^0$, $\widetilde{\mathbf{U}}^0=\mathbf{0}, j=0$

$$\mathbf{K}_{A_i}=\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}{\mathbf{B}'}_{A_i}^k+\overline{\mathbf{H}}^k-\overline{\mathbf{MZQB}'}^k-\mathbf{U}^k \tag{103}$$

    while stopping criteria not satisfied **do**
      $\widetilde{\mathbf{B}}_{A_i}^{j+1}=\mathbf{P}_{A_i}\left(\mathbf{Q}_{A_i}^\top\mathbf{Z}_{A_i}^\top\mathbf{M}^{-1}\mathbf{K}_{A_i}+\rho(\widetilde{\overline{\mathbf{H}}}^j-\widetilde{\mathbf{U}}^j)\right)$
      $\widetilde{\overline{\mathbf{H}}}^{j+1}=S_{\lambda/\rho}\left(\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}_{A_i}^{j+1}+\widetilde{\mathbf{U}}^j\right)$
      $\widetilde{\mathbf{U}}^{j+1}=\widetilde{\mathbf{U}}^j+\mathbf{Q}_{A_i}\widetilde{\mathbf{B}}_{A_i}^{j+1}-\widetilde{\overline{\mathbf{H}}}^{j+1}$
      $j=j+1$
    end while
    ${\mathbf{B}'}_{A_i}^{k+1}=\widetilde{\mathbf{B}}_{A_i}^j$
  end for
  $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}{\mathbf{B}'}_{A_i}^k$ is shared with peers or central node, who computes (104)–(106),

$$\overline{\mathbf{MZQB}'}^k=\frac{1}{n}\sum_i\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}{\mathbf{B}'}_{A_i}^k \tag{104}$$

$$\overline{\mathbf{H}}^{k+1}=\frac{1}{n+\rho}\left(\mathbf{MY}+\overline{\mathbf{MZQB}'}^k+\rho\mathbf{U}^k\right) \tag{105}$$

$$\mathbf{U}^{k+1}=\mathbf{U}^k+\overline{\mathbf{MZQB}'}^{k+1}-\overline{\mathbf{H}}^{k+1} \tag{106}$$

  $k=k+1$
end while

IV.3.2.2 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 26 (left plot), and the same is expected for the individual errors $\|\mathbf{MY}-\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}{\mathbf{B}'}_{A_i}^k\|_2^2, \forall i$.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

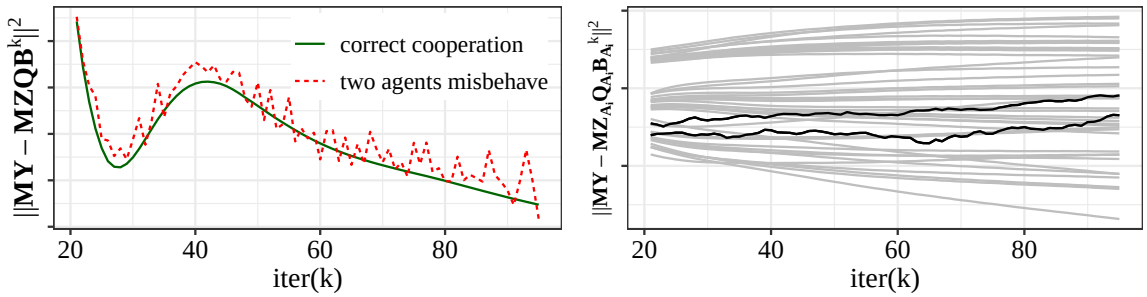


Figure 26 Error evolution (left: global error; right: error by agent with black lines representing the two agents who add random noise to $\mathbf{MZ}_{A_i}\mathbf{Q}_{A_i}\mathbf{B'}^{k}_{A_i}$).

Table 10 Floating-point operations in Algorithm 1.

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

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

In the example of Figure 26, two agents are assumed to add random noise to their coefficients. This results in the erratic curve for the global error shown in Figure 26. An analysis of individual errors, in Figure 26 (right plot), shows that all agents have smooth curves, except the two who shared distorted information.

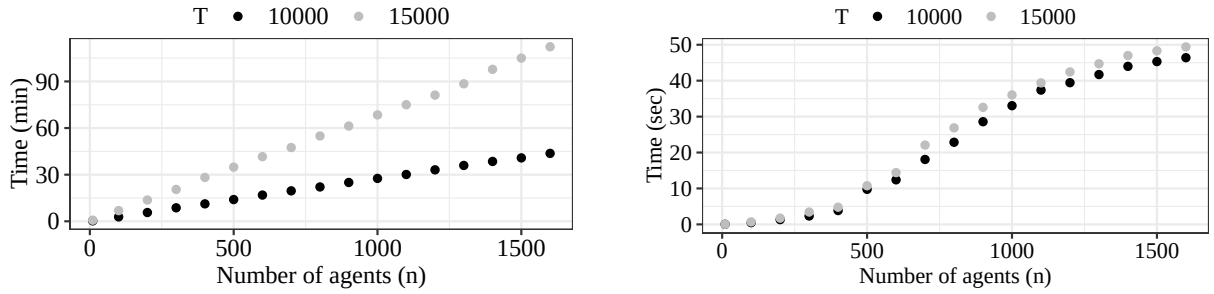
IV.3.3 Tuning of Hyperparameters

Since the ADMM solutions for (97) is related to the solution for (100), 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 (100). 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.

IV.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., Cavalcante et al. (2017b); Cavalcante and Bessa (2017)), 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 10 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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



(a) Data encryption by Algorithm 1.

(b) ADMM (Algorithm 2) iteration.

Figure 27 Mean running time as a function of the number of agents.

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 27 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.

IV.3.5 Asynchronous Communication

When applying the proposed method, the matrices (104)–(106) 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 (105)–(106), by using $\sum_{i=1}^{n} \mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{\max(\Omega_i^k)}_{A_i}$. Then, central agent returns $\overline{\mathbf{H}}^k$ and $\mathbf{U}^k$, informing agents about $\max(\Omega_i^k)$. To proceed, $\mathbf{B}'^{k+1}_{A_i}$ is updated by using $\mathbf{M}\mathbf{Z}_{A_i}\mathbf{Q}_{A_i}\mathbf{B}'^{\max(\Omega_i^k)}_{A_i}$ in (103).

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

IV.3.6 Extension to Short-time Forecasting

The simplicity and competitive performance of VAR models for renewable energy predictions up to 6h ahead motivated us to explore their extension for longer prediction horizons. Since cross-correlation within power measurements is limited to a few hours, such extension requires the use of weather forecasts. The vector autoregressive model with exogenous variables (VAR-

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

X) allows predicting power generation by linearly combining power measurements with weather forecasts.

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

$$\mathbf{y}_t = \boldsymbol{\eta} + \sum_{\ell=1}^{p} \mathbf{y}_{t-\ell}\mathbf{B}^{(\ell)} + \mathbf{x}_t\mathbf{B}^{(exog)} + \boldsymbol{\varepsilon}_t \,, \tag{107}$$

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 $i, j = 1, ..., n$; $\mathbf{B}^{(exog)}$ is the coefficient matrix associated to the exogenous variables; 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. Similar to VAR, a compact representation of a VAR$_n(p)$-X model reads as follows:

$$\mathbf{Y} = \mathbf{Z}\mathbf{B} + \mathbf{E} \,, \tag{108}$$

where

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

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

Although VAR-X allows the integration of exogenous variables, such as wind speed forecast, it is not reasonable to assume a linear relationship between exogenous variables and power generation. Therefore, we explore additive VAR-X models (VAR-AX), that capture the nonlinear relations between power and weather variables through smooth functions, while maintaining some of the positive aspects of the linear approaches,

$$\mathbf{y}_t = \boldsymbol{\eta} + \sum_{\ell=1}^{p} \mathbf{y}_{t-\ell}\mathbf{B}^{(\ell)} + \mathbf{x}_t^{(s)}\mathbf{B}^{(s)} + \boldsymbol{\varepsilon}_t \,, \tag{109}$$

where $\mathbf{x}_t^{(s)} = [f_1(x_{t,1}), f_2(x_{t,2}), \ldots, f_m(x_{t,m})]$ and the functions $f_j(x_{ij})$ are smooth functions fit from the data, e.g., splines (Hastie and Tibshirani, 2017). Like VAR and VAR-X, this model can also be represented in a matrix form $\mathbf{Y} = \mathbf{Z}\mathbf{B} + \mathbf{E}$. Once again, each time series is observed by a single data owner and, consequently, the privacy-preserving protocol proposed above can be extended to the VAR-X and VAR-AX models without major effort.

IV.4 Case Studies

IV.4.1 Very-short Term Forecasting

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 $i=1, \ldots, n$ at each

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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

$$\frac{\|\mathbf{B}_{A_i}^{k+1} - \mathbf{B}_{A_i}^{k}\|_2}{\max(1, \min(\|\mathbf{B}_{A_i}^{k+1}\|_1, \|\mathbf{B}_{A_i}^{k}\|_1))} \leq \epsilon, \tag{110}$$

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 Berdugo et al. (2011) 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 $\mathbf{y}_{i,t-\ell+1}, \ldots, \mathbf{y}_{i,t}$. 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

$$s_{A_i}(a) = \underbrace{(1-\alpha) w_{A_i}(a)}_{\text{own contribution}} + \underbrace{\frac{\alpha}{n} \sum_{i=1}^{n} \sum_{j=1}^{k} w_{A_j}(j) I_\epsilon(t_a, t_j)}_{\text{others' weights for close timestamps}}, \tag{111}$$

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 NRMSE calculated for agent i and lead-time h, with $h = 1, \ldots, 6$, as

$$\text{NRMSE}_{i,h} = \frac{\sqrt{\frac{1}{T} \sum_{t=1}^{T} (\hat{y}_{i,t+h} - y_{i,t+h})^2}}{\max(\{y_{i,t+h}\}_{t=1}^{T}) - \min(\{y_{i,t+h}\}_{t=1}^{T})}, \tag{112}$$

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

IV.4.1.1 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 (Gonçalves and Bessa, 2020) 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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

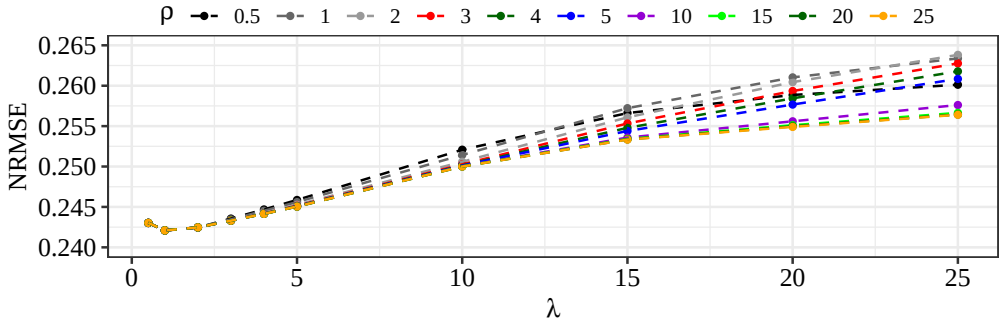


Figure 28 Impact of hyperparameters for $h = 1$, considering solar power dataset.

Table 11 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 ($t+h$-23)*	0.1728	0.1728	0.1728	0.1728	0.1728	0.1728
Analogs Berdugo et al. (2011)†	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)

in clear sky conditions at any given time Bacher et al. (2009). 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 Bessa et al. (2015b), 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, $h \leq 6$.

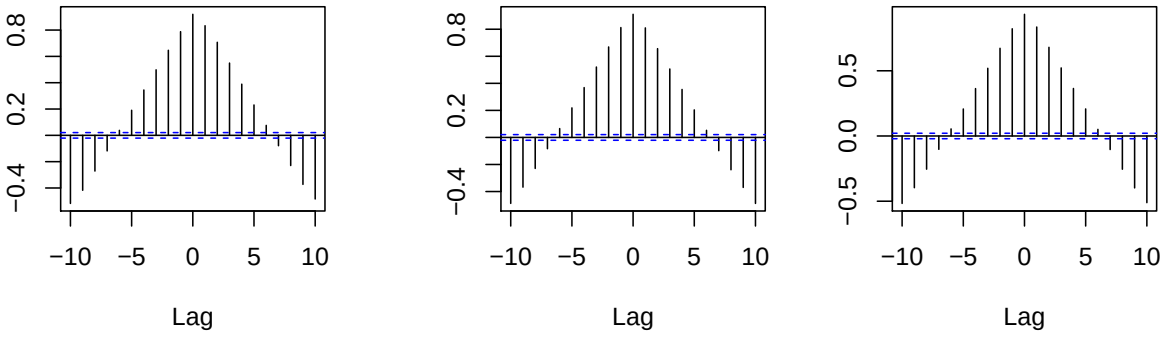
It is important to note that the LASSO-VAR model can be applied to both solar and wind power time series without any modification. Furthermore, when compared to wind power, solar power forecasting is more challenging because the lags 1 and 2 are zero for the first daylight hours, i.e., there are fewer 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 28 illustrates the results in terms of NRMSE, for $h = 1$.

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 11 presents the NRMSE for all agents, distinguishing between lead-times. In general, the smaller the forecasting horizon, the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



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

Figure 29 Cross-correlation plot (CCF) between two solar power plants.

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 (Diebold and Mariano, 2002)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 29. 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 (Cavalcante et al., 2017b), the results for wind power plants show the highest improvement for the second lead time; in the test case of western Denmark (Tastu et al., 2011), the highest cross-dependency between two groups of wind farms was observed for lag two.

Figure 30 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 section.

Table 12 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 $\mathbf{B'}_{A_i}^{k}$ 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 13 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 de-

Table 12 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 ($\approx$ 410)	65.46	0.052 ($\approx$ 300)	65.46	0.1181 ($\approx$ 300)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 13 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

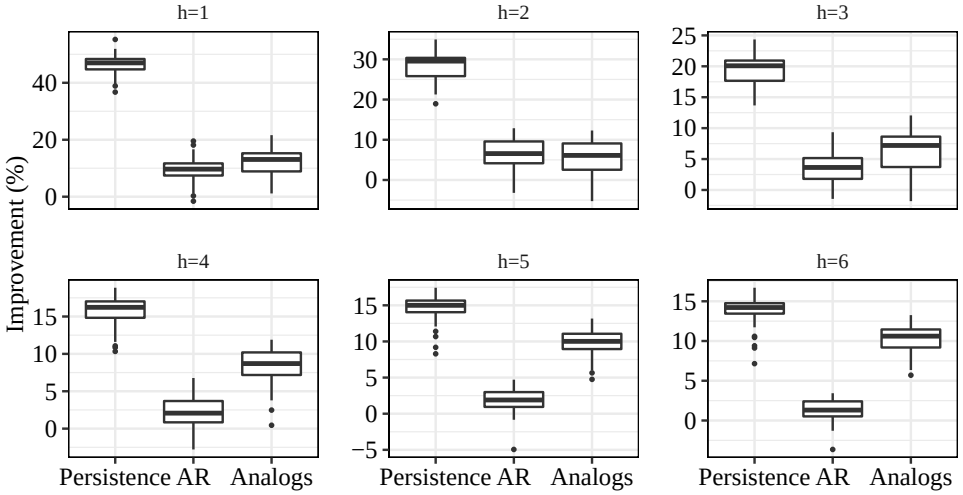


Figure 30 Relative NRMSE improvement (%) over the baseline models, considering solar power dataset.

creases slightly, the LASSO-VAR model continues to outperform the AR model for both collaborative schemes, which demonstrates high robustness to communication failures.

Figure 31 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 of obtaining its information. In P2P, this agent may have communicated his contribution to some peers and the probability of losing information is smaller.

IV.4.1.2 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 (Hong et al., 2016), as depicted in Figure 32. This dataset was used in the Global Energy Forecasting

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

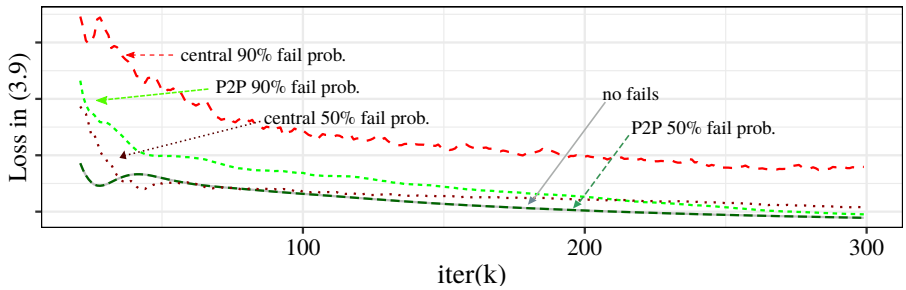
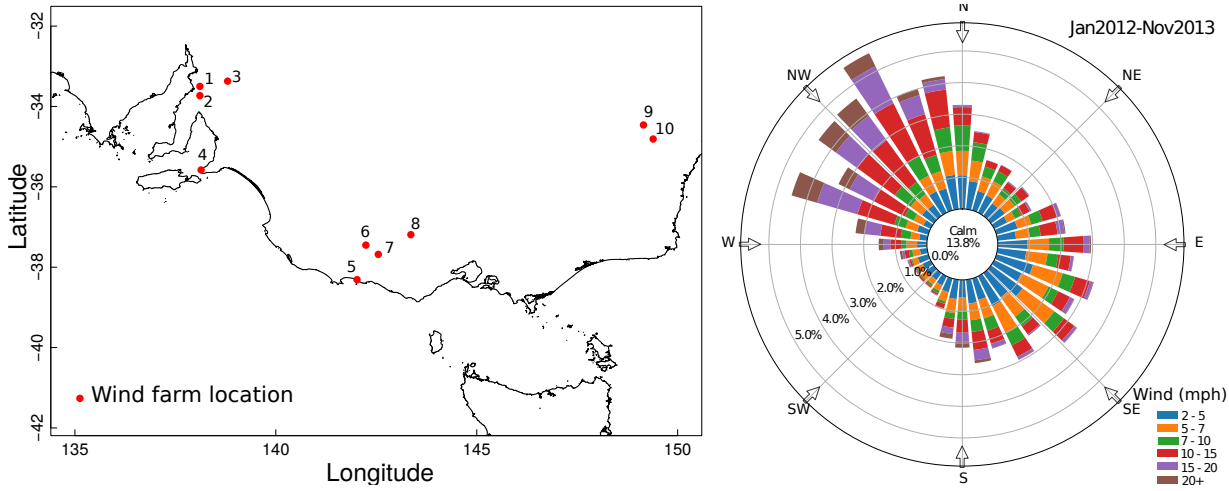


Figure 31 Loss while fitting LASSO-VAR model, considering solar power dataset.



(a) Wind farms' location. *(b) Wind rose (Canberra: close to WF9).*

Figure 32 GEFCom2014 wind power dataset.

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. A sliding-window of one month is considered and the model's fitting period consists of 12 months.

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 33 illustrates the results in terms of NRMSE, when $h = 1$.

To access the quality of the proposed collaborative forecasting model, the synchronous LASSO-VAR is compared with benchmark models. Table 14 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 34 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 35, 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 ago. It is intuitively expected that this is due to the geographical

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

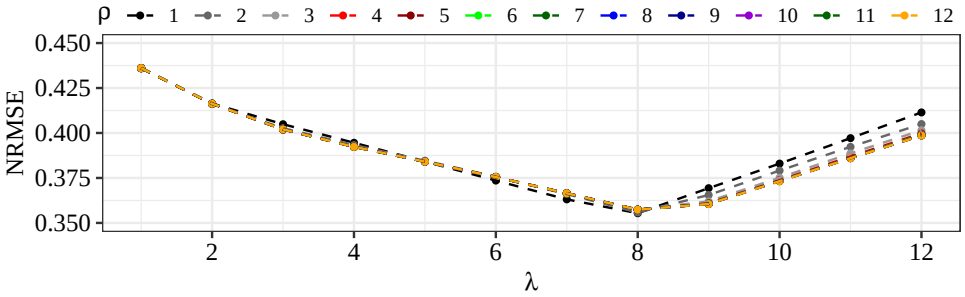


Figure 33 Impact of hyperparameters for $h = 1$, considering wind power dataset.

Table 14 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 Berdugo et al. (2011)†	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)

layout (Figure 32 (a)) of the various wind farms and meteorological particularities of the region, such as wind speed. Figure 32 (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 15 presents the mean running times and the number of iterations of both non-distributed and distributed approaches. When compared to a

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

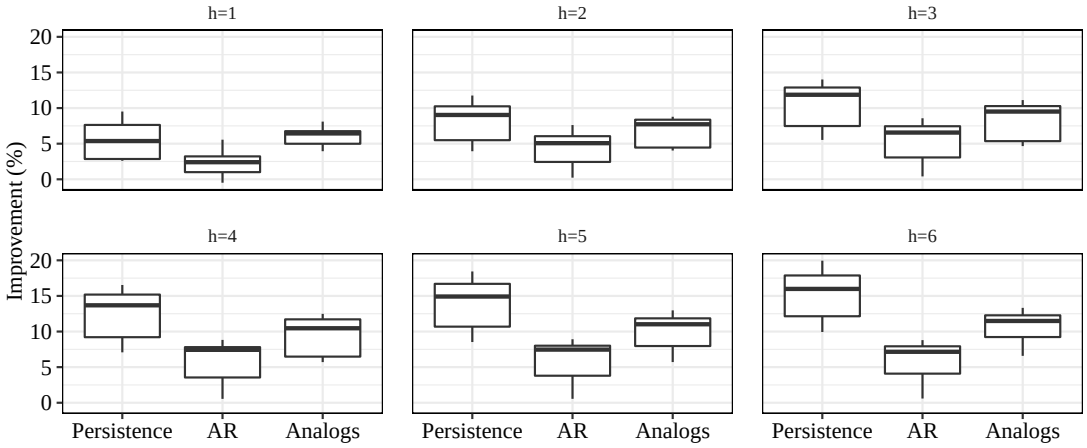


Figure 34 Relative NRMSE improvement (%) over the baseline models, considering wind power dataset.

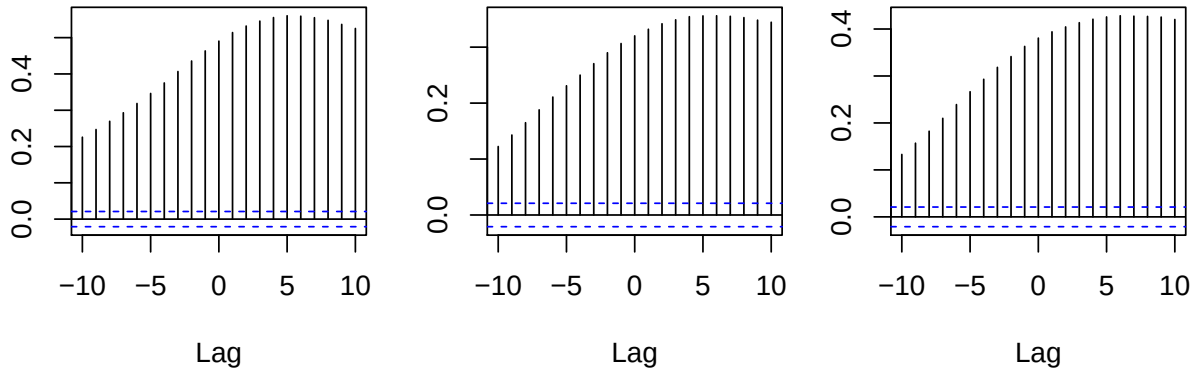
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 15 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 ($\approx$ 400)	125.46	0.059 ($\approx$ 300)	125.46	0.1309($\approx$ 300)

Table 16 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



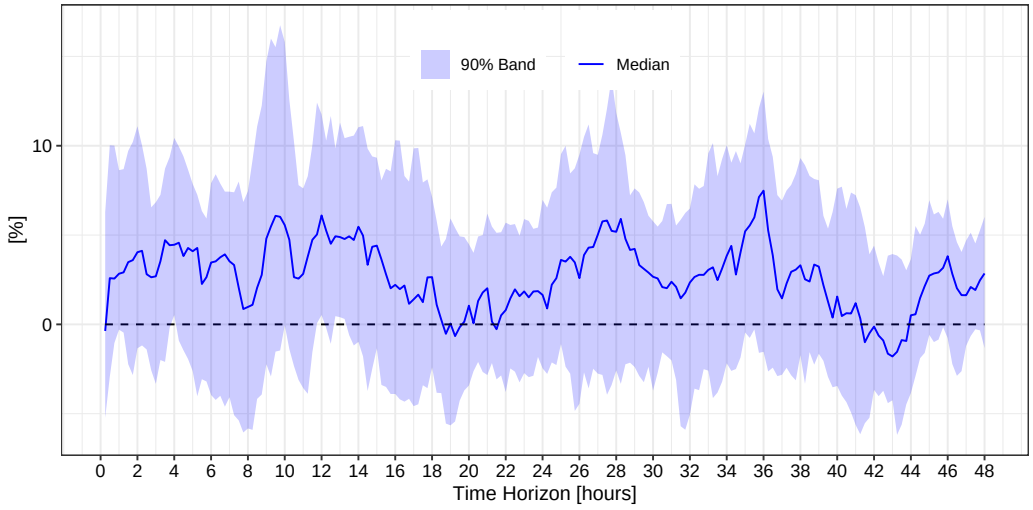
(a) WF3(t) with WF1(t-Lag) *(b) WF8(t) with WF2(t-Lag)* *(c) WF9(t) with WF2(t-Lag)*

Figure 35 Cross-correlation plot (CCF) between two wind power plants.

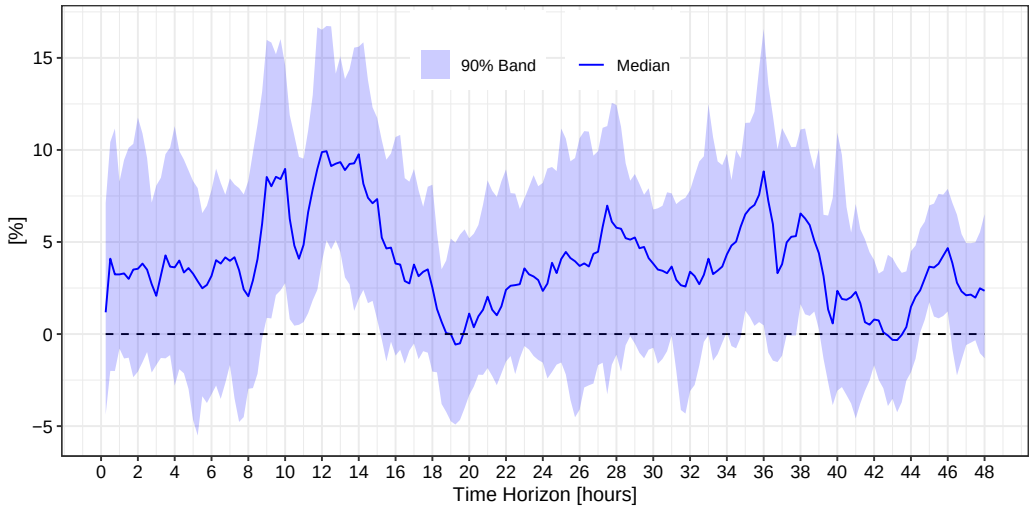
non-distributed LASSO-VAR version, the proposed schemes require larger execution times since they require estimating $\mathbf{B}'^{k}_{A_i}$ through a second ADMM cycle (Algorithm 2). However, the non-distributed LASSO-VAR requires more iterations to converge.

Finally, regarding asynchronous LASSO-VAR ($p_i \geq 0.1$), Table 16 summarizes the mean NRMSE improvement for all agents over the LASSO-AR model, considering 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 LASSO-AR model for both collaborative schemes, which demonstrates high robustness to communication failures.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



(a) MAE



(b) NRMSE

Figure 36 Relative improvement (%) when comparing LASSO-VAR-AX (collaborative model) with LASSO-AR-AX (non-collaborative model).

IV.4.2 Short Term Forecasting

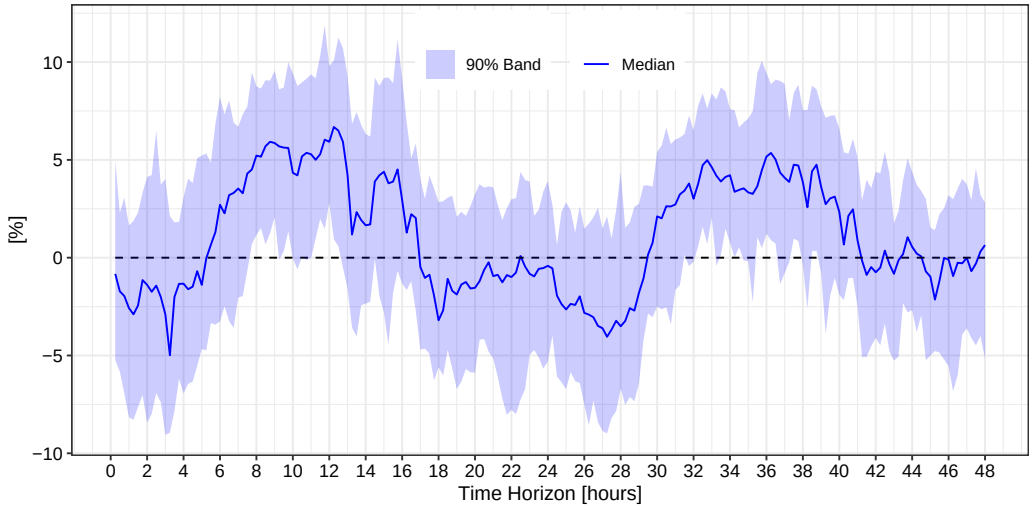
Data Description

The proposed additive method is tested with a confidential real wind power dataset , comprising 15 min resolution time series of wind power generation for 60 wind power turbines (from 13 different wind farms). In addition, the NWP from ECMWF-HRES on a grid surrounding the production sites are available, with forecasting horizons from 15 min to 48h-ahead. The weather variables consist of predictions for u and v wind components at 100 m height. Data covers the period from October 2018 to September 2020.

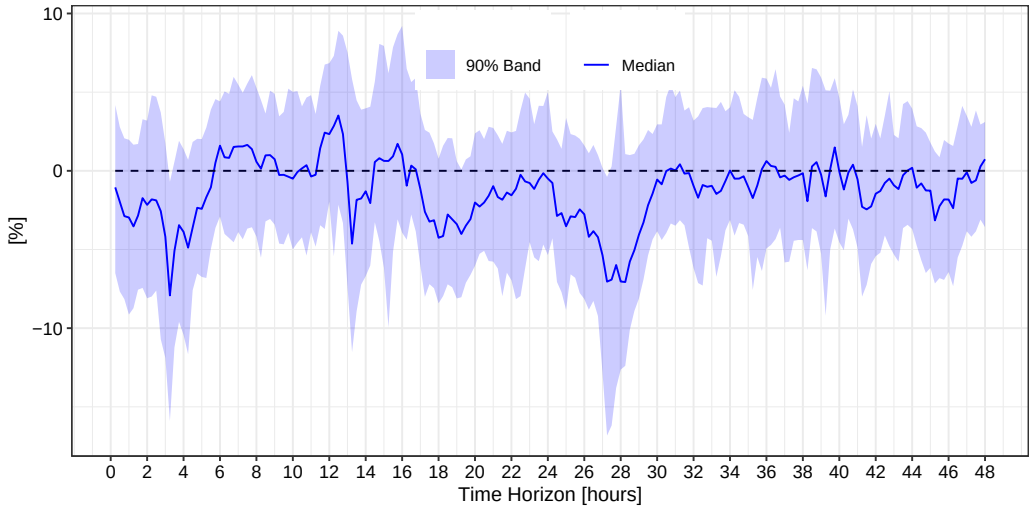
Benchmarks

LASSO-VAR-AX is compared with LASSO-AR-AX (model using only local data) and GBT models, considering forecasts up to 48h-ahead with 15 min resolution. Two scenarios are considered: (i) for the prediction horizons between 0 and 24h, a model is trained considering the weather

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



(a) MAE



(b) NRMSE

Figure 37 Relative improvement (%) when comparing LASSO-VAR-AX with GBT.

forecasts generated at midnight of that day, and the ones generated at midnight of the previous day; (ii) for horizons >24h, a model is trained with a unique set of weather forecasts. In terms of lags, only the 4 most recent power measurements are considered, based on the cross-correlation analysis.

The performance is measured through MAE and RMSE. LASSO-based models are estimated by using ADMM, meaning two hyper-parameters to tune (λ and ρ). Both ADMM and GBT hyper-parameters are determined by Bayesian optimization, through 12-fold cross-validation within the one year training set. The hyper-parameters are updated every 6 months.

Results and Discussion

The results consider a synchronous communication scenario, and the goal is to show the potential of extending linear models to predict longer horizons.

Figure 36 shows the relative improvement in terms of MAE and NRMSE, for the 60 wind turbines, when comparing LASSO-VAR-AX (collaborative model with polynomial splines (Hastie and Tib-

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

shirani, 2017)) with LASSO-AR-AX (local model with polynomial splines). In general, at least 30 of the 60 wind turbines improve their accuracy when using collaborative forecasting.

Since the additive methods aim to capture nonlinear relationships, the performance of the LASSO-VAR-AX is also compared to the performance of GBT models. Figure 37 shows the corresponding relative improvement. While the figure shows each model have different errors for different time periods, the average trend is similar on both MAE and NRMSE metrics. We would like to emphasize that the focus of this LASSO-VAR extension was not to outperform GBT models, but rather to ensure data privacy by using the protocol we proposed in this section.

IV.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 section 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.

V. Online distributed learning in wind power forecasting

V.1 Introduction

Following the sustained deployment of renewable energy generation capacities, especially in the case of wind energy, forecasting has received increasing interest. Accurate wind power forecasts enhance the profitability of wind farms when participating in electricity markets (Mazzi and Pinson, 2017). Power system operators rely on generation and load forecasts for the optimal scheduling of conventional generation units and operating reserves (Matos et al., 2017). An overview of state-of-the-art methods for wind power forecasting is presented in (Giebel and Kariniotakis, 2017). While lead times ranging from hours to days have been of central focus

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

owing to wind power participation in electricity markets, other lead times ranging from a few minutes to a week ahead (or possibly more) are of relevance to a broad range of operational and decision-making problems. The demand for accurate wind power forecasts with very short lead times ranging from minutes to a few hours has supported the development of novel forecasting methods (Pinson, 2012a; Pinson and Madsen, 2012). For very short lead times, statistical and machine-learning methods clearly dominate over methods based on numerical weather forecasts. The majority of forecasting methods only utilise local data that is recorded at the wind farm of interest. Numerous publications have shown that using high-dimensional learning methods in combination with data from surrounding sites, such as meteorological stations or wind farms, can improve forecast accuracy substantially (He et al., 2014; Tastu et al., 2011, 2014). Generally, this modelling approach explores and exploits spatial-temporal patterns in wind power generation. It is fairly intuitive since a propagating wind field causes lagged changes in the power production between two or more geographically dispersed wind farms. Naturally, an upwind location experiences changes in wind speed first before a downwind location that is in the trajectory of the same wind field does. Hence, using explanatory variables that are related to wind speed or power production for an upwind location helps to forecast future changes in the power production for a downwind location. These dependencies may be very complex and conditional on prevailing weather conditions (Girard and Allard, 2013). Exploiting off-site information and space-time dynamics is something that is broadly considered in environmetrics e.g. for ozone forecasting (Paci et al., 2013), for the prediction of weather variables e.g. precipitation (Fabio Sigrist et al., 2012), and in traffic forecasting (Min and Wynter, 2011), etc. It also shares similarities with the problem of forecasting panel data with cross-sectional dependencies in econometrics (Baltagi et al., 2014).

In practice, many algorithms that exploit spatial-temporal dependencies in wind power forecasting are based on *batch learning*, i.e., based on the assumption that model coefficients are time-invariant. They hence are estimated once and for all on a training dataset (the so-called "batch" of data). The estimated coefficients are then used to issue predictions even though new data arrives sequentially. In applications where the true model coefficients are time-varying, such as in wind power forecasting due to seasonal variations in wind dynamics, as well as the environment of wind farms, using batch-learning algorithms impacts forecast accuracy negatively. An approach that is then often considered is to re-estimate the coefficients using a sliding or expanding training window whenever new data samples are available (Dowell and Pinson, 2016; Zhang and Wang, 2018a). The training samples are usually weighted to control how fast the estimated coefficients adapt to changes in the dataset. However, this approach is unattractive in cases where the learning algorithm cannot efficiently re-estimate the model coefficients. Especially for high-dimensional models the time required to estimate model coefficients can be prohibitive for many applications. Computationally efficient algorithms estimate time-varying model coefficients on the fly by using recursivity, whenever a new data sample is available. We use the term *online learning* when referring to such learning algorithms. Analogously, the term offline (batch) learning refers to algorithms where the time-invariant model coefficients are estimated on a batch of training samples.

Most of the proposed online learning algorithms in the wind power forecasting literature are used in combination with models that rely on explanatory variables which are measured exclusively at the wind farm of interest. Relevant methods are presented in, e.g., (Møller et al., 2008) and (Bessa et al., 2012a). Notable exceptions are the sparse online warped Gaussian model of (Kou et al., 2013) and the proposal of (Messner and Pinson, 2018). In the latter case, the authors described an online algorithm for high-dimensional vector autoregressive (VAR) models. A limitation is that all explanatory variables must be collected by a single agent to eventually employ that algorithm. Throughout the paper we use the term *centralised learning* when referring to situations where it is necessary to have direct access to all explanatory variables centrally in order to estimate model coefficients. Considering that wind farms are operated by competing agents and that power production data and related measurements are often deemed confidential, the requirement to collect all explanatory variables centrally brings some limitations. The unwillingness of wind farm operators to share data with third parties motivates the recent

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

interest in *distributed learning* (and possibly *privacy-preserving*) algorithms in the field of wind power forecasting.

Distributed learning algorithms conceptually aim at relaxing this necessity of collecting all explanatory variables centrally, by decomposing a learning problem into many subproblems and one master problem. When estimating the coefficients of a forecasting model for a given wind farm of interest, for which some explanatory variables are provided by other wind farms, the distributed algorithm assigns a subproblem to each wind farm where explanatory variables are available. The model coefficients are then estimated by alternating between solving the master problem and subproblems, taking advantage of algorithm-specific variables that link the subproblems to the master problem and vise versa. With such algorithms it is no longer necessary to collect all explanatory variables centrally since the explanatory variables that are provided by other wind farms are only used in their respective subproblem. The appropriate design of distributed learning algorithms protects the explanatory variables of wind farm operators by not exposing them to others. We refer to this condition when stating that the data privacy of a wind farm operator is protected.

Today, to the best of our best knowledge, only a handful of papers have investigated distributed learning algorithms for wind power forecasting (and renewable energy forecasting, more generally). The most prominent papers all build upon the Alternating Direction Method of Multipliers (ADMM). In (Pinson, 2016a) and (Cavalcante et al., 2017a) algorithms are developed to estimate the coefficients of an AR-X model while regularising with the LASSO. (Zhang and Wang, 2018a) extends prior work to probabilistic forecasts but replace the L_1-penalization of the LASSO with an L_2-penalization to obtain a computationally cheaper algorithm. Unfortunately, prior distributed algorithms do not allow online learning to be performed. Therefore, to estimate time-varying coefficients it is necessary to apply the algorithms on a sliding or expanding training window while weighting the data samples. As a consequence, we aim here to close the gap between online and distributed learning methods. Our contribution consequently includes: *(i)* the development of an online ADMM version, and *(ii)* additionally proposing a mirror-descent-inspired algorithm for online distributed learning. Both have advantages and caveats to be explored through simulation studies and the application to a case study with a large real-world dataset consisting of hundreds of wind farms in Denmark.

The remainder of this paper is organised as follows. The general model and forecasting framework is introduced in Section V.2. Section V.3 describes an online ADMM version which we refer to as Online ADMM (OADMM). Anticipating the non-negligible computational complexity of the OADMM, the computationally lighter Adaptive Distributed MIrror Descent Algorithm made Sparse (Adaptive D-MIDAS) is presented thereafter in Section V.4. The inherent properties of these two approaches are analysed through a simulation study in Section V.5. Thereafter, the algorithms are benchmarked on a large real-world dataset consisting of 311 wind farms in Section V.6. Conclusions and perspectives for future work close the paper in Section V.7.

V.2 Modelling and forecasting framework

V.2.1 From agents and their data to relevant models

Wind power generation is observed at regular time intervals at S sites. Let us write $y_{s,t}$ for the power measurement of site $s \in \Omega_S = \{s_1, s_2, ..., s_S\}$ and time stamp $t \in \{1, 2, \ldots, T\}$. Power measurements are commonly normalised by the nominal capacity of the site, such that eventually, $y_{s,t} \in [0, 1]$. We restrict ourselves to AR-X models using recent power measurements as explanatory variables, in a fashion similar to the models used by (Messner and Pinson, 2018), (Cavalcante et al., 2017a), (Pinson, 2016a) and (Zhang and Wang, 2018a). However, extending such AR-X models to accommodate additional explanatory variables like wind speed for

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

instance is straightforward. Generalisation to nonlinear modelling approaches would be more complicated. Depending on the type of data collected, it may be sensible to centre the data. Other types of transformations may additionally be considered. For instance for nonlinear and bounded processes like wind power generation, the generalized logit-Normal transformation of (Pinson, 2012a) may render more Gaussian innovations and yield a stochastic process that is more homoskedastic. Without any loss of generality, we assume that $y_{s,t}$ are transformed power measurements.

The operator of site s_j, referred to as *central agent*, contracts a set $\Omega_S^{(j)} \subset \Omega_S \setminus s_j$ of other sites to enter a *learning agreement*. Consequently, all sites $s_i \in \Omega_S^{(j)}$ are referred to as *contracted agents*. In practice, this means that the contracted agents will support s_j in improving wind power forecasts through a distributed learning framework without exposing their explanatory variables to the central agent. The cardinality of $\Omega_S^{(j)}$ will certainly be small in practice, since it may not be of relevance for a central agent to contract a large number of sites e.g. due to the limited scale of dependence structures in space and time, and possible transaction costs. Here, for simplicity, we assume that the cardinality of $\Omega_S^{(j)}$ is $S-1$, so as to overlook the selection problem. We further assume that all contracted agents are rational and act truthfully. Therefore, we overlook the potential of malicious behaviour by assuming that the learning network design incentivises all agents to be fully collaborative (e.g. through contracts).

An AR-X model is used to link the power measured at site s_j and time t with past measurements of site s_j and the sites of the contracted agents. This gives

$$y_{s_j,t} = \beta_{s_j,0,t} + \sum_{l=1}^{L} \left(\underbrace{\beta_{s_j,l,t}\, y_{s_j,t-l}}_{\text{on-site}} + \sum_{s \in \Omega_S^{(j)}} \underbrace{\beta_{s,l,t}\, y_{s,t-l}}_{\text{off-site}} \right) + \epsilon_{s_j,t} \tag{113}$$

i.e., as a linear combination of past power measurements for all sites plus an intercept term $\beta_{s_j,0,t}$ and an innovation term $\epsilon_{s_j,t}$ with zero mean and finite variance. The scalars $\beta_{s,l,t}$ are the model coefficients for lag $l = 1, \ldots, L$ and site $s \in \Omega_S$. L denotes the order of the auto-regressive process. For simplicity, we consider that the maximum lag L is the same for the central and contracted agents, though it does not need to be. In addition a time index t is used, as it is assumed that the model coefficients are time-varying. While it may be common in the econometrics literature to assume that those coefficients follow some process e.g. autoregressive (Bekierman and Manner, 2018), we consider that these coefficients follow a random walk with varying means. They can hence be tracked with some simple form of Kalman filtering where parameters are updated recursively. This approach is common in wind power forecasting, as in the examples of (Pinson, 2012a) and (Pinson and Madsen, 2012) among others.

The model in (113) has many coefficients, since a different coefficient is used for each combination of location and lagged value. This potentially leads to the need to estimate $L \times S + 1$ coefficients with $L \times S + 1$ being large. As an alternative one may parameterise the spatio-temporal dynamics of wind power generation, as commonly done in environmetrics and statistical modelling of meteorogical variables (Fabio Sigrist et al., 2012). However, here, these dynamics are very complex and conditional on prevailing weather conditions (Girard and Allard, 2013). Consequently, when having access to large datasets as is common with wind power forecasting, it is possible to increase the number of coefficients to be estimated. In parallel, note that in practice many of the $\beta_{s,l,t}$ coefficients are expected to be 0, depending on the de-correlation range and prevailing wind direction. This is why we employ a fully data-driven approach to variable selection and coefficient estimation through L_1 regularisation. In addition, since we are working within an online learning framework, the resulting model coefficients are time-varying and are thus expected to capture the slow variations in wind power dynamics e.g. induced by seasons and changes in the environment of the wind farms.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

For convenience we rewrite (113) in the compact form

$$y_{s_j,t} = \sum_{s \in \Omega_S} \mathbf{a}_{s,t-1} \boldsymbol{\beta}_{s,t} + \epsilon_{s_j,t} \tag{114}$$

where $\mathbf{a}_{s,t-1}$ is an horizontal vector gathering the values of explanatory variables, at time t and location s, and $\boldsymbol{\beta}_{s,t}$ is the corresponding vector of model coefficients, i.e,

$$\mathbf{a}_{s,t-1} = \begin{cases} [1,\, y_{s,t-1},\, \ldots,\, y_{s,t-L}], & s = s_j \\ [y_{s,t-1},\, \ldots,\, y_{s,t-L}], & \text{otherwise} \end{cases} \tag{115}$$

and

$$\boldsymbol{\beta}_{s,t} = \begin{cases} [\beta_{s,0,t},\, \beta_{s,1,t},\, \ldots, \beta_{s,L,t}]^\top, & s = s_j \\ [\beta_{s,1,t},\, \ldots,\, \beta_{s,L,t}]^\top, & \text{otherwise} \end{cases} \tag{116}$$

Within this modelling framework, the largest contribution to explaining the dynamics of $y_{s_j,t}$ comes from local information given by lagged values of this process. In comparison, offsite information provides a lower contribution, though still allowing a significant improvement of forecast accuracy for short lead times (Messner and Pinson, 2018). Since most of the $\beta_{s,l,t}$ coefficients are expected to be 0, this also implies that a central agent eventually does not need to make a learning agreement with many other wind farms, hence limiting communication needs and potential contracts if distributed learning was to be remunerated.

V.2.2 Framework for distributed and online learning

When estimating the model coefficients of such an AR-X model in a centralised setup, an agent is required (most likely the operator of site s_j, i.e., the central agent, or the contracted forecast vendor) to gather all explanatory variables. In a distributed learning network, however, the coefficient estimation problem is decomposed into many subproblems that are solved by the agents who entered the learning agreement. In our case the problem is conveniently decomposed across all S wind farm operators. The architecture of our distributed learning network is visualised in Figure 38, where the arrows indicate information exchange.

Regularly applied in distributed networks, a fusion centre (supervisory node) oversees the communication among all agents. In practice, the central agent does not directly communicate with its contracted agents, i.e., information is not directly exchanged via a peer-to-peer connection. The reason for designing the network like this is twofold. On the one hand, the communication becomes more structured for large-scale applications where each member of the learning agreement receives a forecast for its site. This requires estimating the coefficients of at least S models in parallel. On the other hand, it may support some of the privacy concerns of wind farm operators who do not wish their private information to be exposed to other agents.

In centralised learning the flow of information is unidirectional from the contracted agents to the central agent. Distributed learning algorithm require instead a bidirectional exchange of information, as the arrows show in Figure 38. Our distributed learning algorithms require each agent to solve their assigned subproblem. This is fundamentally different from centralised learning, where only the central agent performs computations when estimating the model coefficients.

Numerous distributed and online algorithms have been proposed in the literature, though not for application in renewable energy forecasting. While first focusing on the available online versions of the ADMM it was observed that all algorithms address consensus problems, i.e., require the design matrix to be horizontally partitioned across all agents (Suzuki, 2013; Wang and Banerjee, 2012; Matamoros, 2017). Figure 39 illustrates the difference between a horizontal and vertical partitioning of the set of explanatory variables in a model like the one we use here as a basis for forecasting.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

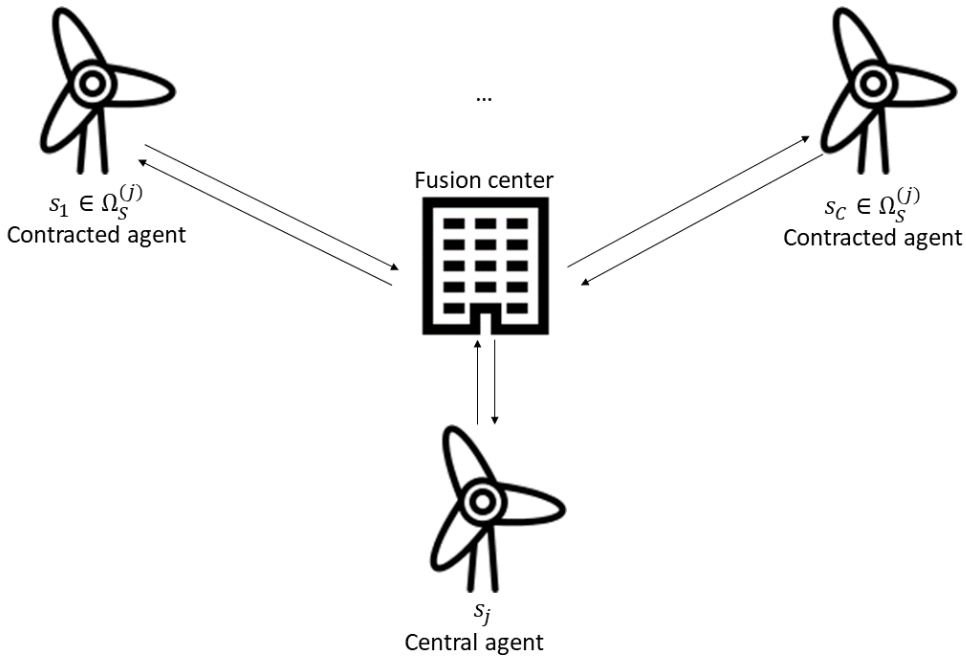


Figure 38 Architecture of the distributed learning network

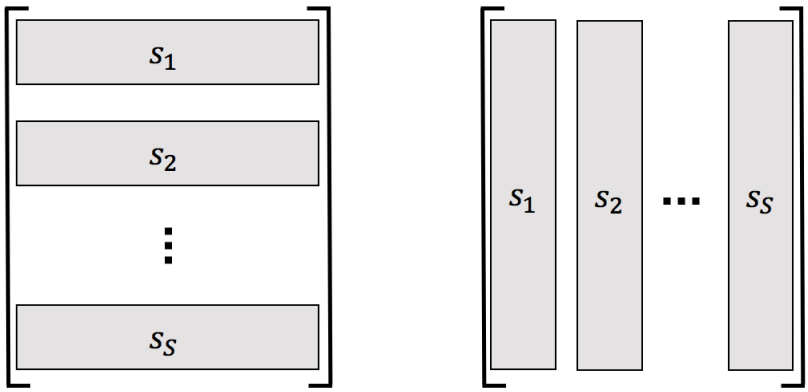


Figure 39 Horizontal (left) and vertical (right) partitioning of a matrix across S agents. Both matrices have equal dimensions. Each column represents a unique feature whereas a row is related to a time instance.

From (114), it can clearly be seen that in our forecasting problem the design matrices are naturally vertically partitionable across all S agents, i.e. each agent observes a unique subset of the whole set of explanatory variables. Horizontally partitionable datasets are found in applications where instances of the design matrix are recorded at different locations but with identical features (e.g., in clinical trials carried out across multiple hospitals). While online versions of ADMM have already been proposed for horizontally partitionable design matrices, this is not the case for vertically partitionable ones. This motivates our proposal asdescribed in the following section.

V.3 Online Alternating Direction Method of Multipliers (OADMM)

(Pinson, 2016a) originally proposed using the ADMM (Boyd et al., 2010) to estimate the ARX model coefficients in (113) in a distributed fashion while applying L_1-regularisation with the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

LASSO. The applied ADMM estimates the model coefficients on a batch of training samples and does not allow for efficient coefficient re-estimates in applications where the true coefficients are expected to be time-varying. We thus extend this algorithm to an online version that minimises the cumulative loss over all observed data samples. Our online version efficiently re-estimates all model coefficients through recursions whenever a new data sample is available. A flowchart for the Online ADMM approach (abbreviated OADMM) is presented in Figure 40, and a detailed algorithm is available in Algorithm 3.

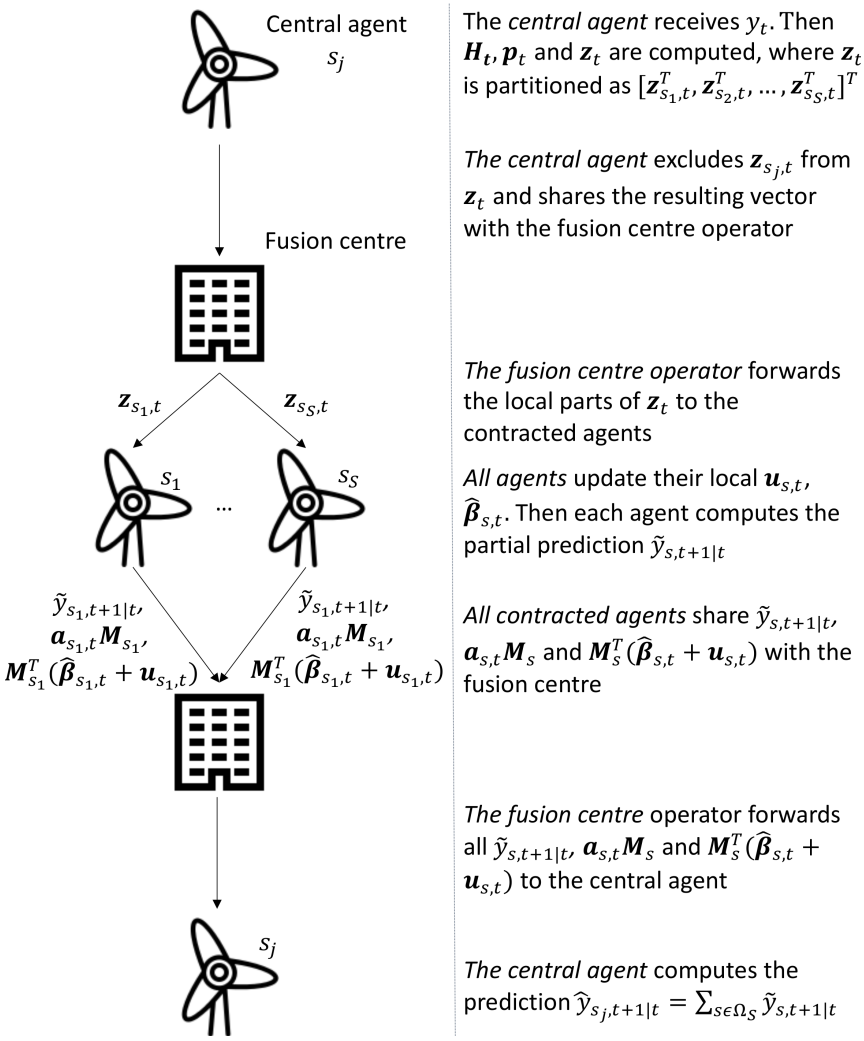


Figure 40 Flowchart for the Online ADMM (OADMM) approach for online distributed learning applied to wind power forecasting.

V.3.1 Coefficient estimation through a time-varying optimisation problem

Considering 1-step ahead forecasting the OADMM approach solves an unconstrained minimisation problem. For every time stamp t, it can be formulated as

$$\min_{\{\boldsymbol{\beta}_{s,t}\}_s} \frac{1}{2}\sum_{\tau=1+L}^{t}\left(\sum_{s\in\Omega_S}\boldsymbol{a}_{s,\tau-1}\boldsymbol{\beta}_{s,t}-y_{s_j,\tau}\right)^2+\lambda\sum_{s\in\Omega_S}\|\boldsymbol{\beta}_{s,t}\|_1 \quad (117)$$

where $\boldsymbol{a}_{s,\tau-1}$ and $\beta_{s,t}$ are as defined in (115) and (116). In parallel, $\lambda \geq 0$ is the L_1 regularisation parameter that controls sparsity. L_1 regularisation penalizes the model coefficient absolute

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

values and thereby shrinks coefficients deemed to be insignificant towards 0.

In order to solve the minimisation problem in (117) every time a new data sample is made available, it should be made computationally efficient. Additionally, we want to control the level of adaptivity via a forgetting factor as also done by (Messner and Pinson, 2018), (Møller et al., 2008) and (Pinson and Madsen, 2012). By giving less weight to older data, the model coefficient estimates better reflects the recent dynamics in the time-series data. Introducing an exponential forgetting factor ν into (117) results in

$$\min_{\{\boldsymbol{\beta}_{s,t}\}_s} \frac{1}{2} \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\sum_{s \in \Omega_S} \boldsymbol{a}_{s,\tau-1} \boldsymbol{\beta}_{s,t} - y_{s_j,\tau} \right)^2 + \lambda \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_{s,t}\|_1 \tag{118}$$

where $\nu \in]0, 1]$. A value of 1 results in no forgetting while decreasing values increase the amount of forgetting. Values slightly less than 1 are generally preferred. ν may be optimised in practice through, e.g., cross-validation.

The standard ADMM builds on the dual-ascent method, which is used to solve optimisation problems where the objective function is separable, by splitting the complete model coefficient vector into sub-vectors. Our problem is naturally separable since each wind farm operator has unique explanatory variables $\boldsymbol{a}_{s,\tau-1}$ and related model coefficients $\boldsymbol{\beta}_{s,t}$ in (118). The optimisation problem is transformed into an appropriate ADMM sharing form by adding the auxiliary vector $\boldsymbol{z}_{s,t}$ to (118). The constrained optimisation problem then reads

$$\begin{aligned} \min_{\{\boldsymbol{\beta}_{s,t}\}_s} \quad & \frac{1}{2} \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\sum_{s \in \Omega_S} \boldsymbol{a}_{s,\tau-1} \boldsymbol{z}_{s,t} - y_{s_j,\tau} \right)^2 + \lambda \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_{s,t}\|_1 \\ \text{subject to} \quad & \boldsymbol{\beta}_{s,t} - \boldsymbol{z}_{s,t} = 0, \quad \forall s \in \Omega_S \end{aligned} \tag{119}$$

The ADMM uses the augmented Lagrangian to solve the constrained optimisation problem with respect to $\boldsymbol{\beta}_{s,t}$ and $\boldsymbol{z}_{s,t}$, by updating the variables in an alternating fashion. For a detailed description of the ADMM and its application in distributed networks, the reader is referred to (Boyd et al., 2010). When following the standard ADMM to solve (119), the central agent may be able to retrieve the explanatory variables of all contracted agents. Thus, the data privacy of the contracted agents is violated. In the offline ADMM for distributed learning of (Pinson, 2016a), the explanatory variables $\boldsymbol{a}_{s,\tau-1}$ of each agent are protected in each step of the algorithm naturally by being multiplied by the respective $\boldsymbol{\beta}_{s,t}$. Taking this as inspiration, our idea is to introduce the encryption matrix $\boldsymbol{M}_s \in \mathbf{R}^{L,L}$ and multiply it by $\boldsymbol{a}_{s,\tau-1}$ whenever it appears. We achieve this by changing the affine constraint in (119) into

$$\boldsymbol{\beta}_{s,t} - \boldsymbol{M}_s \boldsymbol{z}_{s,t} = 0, \quad \forall s \in \Omega_S \tag{120}$$

and additionally adjusting the objective function by replacing the term $\boldsymbol{a}_{s,\tau-1}\boldsymbol{z}_{s,t}$ with $\boldsymbol{a}_{s,\tau-1}\boldsymbol{M}_s\boldsymbol{z}_{s,t}$. As a requirement, each encryption matrix must be non-singular and chosen by each agent privately. This eventually yields the encrypted version of the constrained optimisation problem (119), i.e.,

$$\begin{aligned} \min_{\{\boldsymbol{\beta}_{s,t}\}_s} \quad & \frac{1}{2} \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\sum_{s \in \Omega_S} \boldsymbol{a}_{s,\tau-1} \boldsymbol{M}_s \boldsymbol{z}_{s,t} - y_{s_j,\tau} \right)^2 + \lambda \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_{s,t}\|_1 \\ \text{subject to} \quad & \boldsymbol{\beta}_{s,t} - \boldsymbol{M}_s \boldsymbol{z}_{s,t} = 0, \quad \forall s \in \Omega_S \end{aligned} \tag{121}$$

The augmented Lagrangian of (121) in its scaled form is then written as

$$\begin{aligned} \mathcal{L}_\rho(\boldsymbol{\beta}_t, \boldsymbol{z}_t, \boldsymbol{u}_t) = \frac{1}{2} \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\sum_{s \in \Omega_S} \boldsymbol{a}_{s,\tau-1} \boldsymbol{M}_s \boldsymbol{z}_{s,t} - y_{s_j,\tau} \right)^2 + \lambda \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_{s,t}\|_1 \\ + \frac{\rho}{2} \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_{s,t} - \boldsymbol{M}_s \boldsymbol{z}_{s,t} + \boldsymbol{u}_{s,t}\|^2 \end{aligned} \tag{122}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

where $\rho > 0$ is a penalty parameter and $\boldsymbol{u}_{s,t}$ are the dual variables for the constraints in (121). The OADMM then performs the minimisation of the augmented Lagrangian by sequentially optimising for $\boldsymbol{\beta}_{s,t}$ and $\boldsymbol{z}_{s,t}$ while updating the dual variables $\boldsymbol{u}_{s,t}$ as part of the dual ascent algorithm. By optimising for $\boldsymbol{\beta}_{s,t}$ and $\boldsymbol{z}_{s,t}$ individually it is possible to take advantage of the separability of (122) with respect to all $\boldsymbol{\beta}_{s,t} \in \Omega_S$.

V.3.2 Recursive updates of parameters

i. Central agent updates

To perform an update at time t, let us first focus on the master problem of the central agent. The central agent first observes the true power production $y_{s_j,t}$ and subsequently derives the prediction error $y_{s_j,t} - \hat{y}_{s_j,t|t-1}$. The augmented Lagrangian is then minimised with respect to the auxiliary variable z. The minimisation is written as a parameter update which is carried out exclusively by the central agent. To obtain the update equations we first define

$$\boldsymbol{\beta}_{t-1} = \left[\boldsymbol{\beta}_{s_1,t-1}^\top, \ldots, \boldsymbol{\beta}_{s_S,t-1}^\top\right]^\top, \tag{123a}$$

$$\boldsymbol{z}_{t-1} = \left[\boldsymbol{z}_{s_1,t-1}^\top, \ldots, \boldsymbol{z}_{s_S,t-1}^\top\right]^\top, \tag{123b}$$

$$\boldsymbol{u}_{t-1} = \left[\boldsymbol{u}_{s_1,t-1}^\top, \ldots, \boldsymbol{u}_{s_S,t-1}^\top\right]^\top, \tag{123c}$$

$$\boldsymbol{a}_{t-1} = \left[\boldsymbol{a}_{s_1,t-1}, \ldots, \boldsymbol{a}_{s_S,t-1}\right], \tag{123d}$$

the model coefficient estimates $\hat{\boldsymbol{\beta}}_t$ at time t, and the block-wise diagonal matrix

$$\boldsymbol{M} = \operatorname{diag}\left(\boldsymbol{M}_{s_1}, \ldots, \boldsymbol{M}_{s_S}\right) \tag{124}$$

Differentiating the augmented Lagrangian with respect to $\boldsymbol{z}_t$ yields

$$\frac{\partial \mathcal{L}_\rho(\hat{\boldsymbol{\beta}}_t, \boldsymbol{z}_t, \boldsymbol{u}_t)}{\partial \boldsymbol{z}_t} = \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right)^\top \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\boldsymbol{z}_t - y_{s_j,t}\right) - \rho \boldsymbol{M}^\top \left(\hat{\boldsymbol{\beta}}_{t-1} - \boldsymbol{M}\boldsymbol{z}_t + \boldsymbol{u}_{t-1}\right) \tag{125}$$

By writing

$$\boldsymbol{H}_t = \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right)^\top \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right) \tag{126}$$

and

$$\boldsymbol{p}_t = \sum_{\tau=1+L}^{t} \nu^{t-\tau} \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right)^\top y_{s,t} \tag{127}$$

the auxiliary vectors are updated by equating (125) to 0 and then solving for $\boldsymbol{z}_t$. Hence, the OADMM requires

$$\left(\boldsymbol{H}_t + \rho \boldsymbol{M}^\top \boldsymbol{M}\right) \boldsymbol{z}_t = \boldsymbol{p}_t + \rho \boldsymbol{M}^\top \left(\hat{\boldsymbol{\beta}}_{t-1} + \boldsymbol{u}_{t-1}\right) \tag{128}$$

to be solved for $\boldsymbol{z}_t$. Before solving the equation system, the covariance structures $\boldsymbol{H}_t$ and $\boldsymbol{p}_t$ are efficiently updated via the recursions

$$\boldsymbol{H}_t = \nu \boldsymbol{H}_{t-1} + \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right)^\top \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right) \tag{129a}$$

$$\boldsymbol{p}_t = \nu \boldsymbol{p}_{t-1} + \left(\boldsymbol{a}_{\tau-1}\boldsymbol{M}\right)^\top y_{s,t} \tag{129b}$$

Both covariance structures comprise the memory of the recursive updating process, controlled by the forgetting factor ν.

After the central agent has updated the auxiliary vectors, it shares them via the fusion centre with its contracted agents. This is considered to be a broadcasting operation where the central agent distributes local variables within the network.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

ii. Contracted agent updates

After each contracted agent receives its respective auxiliary vector $\boldsymbol{z}_{s,t}$, all S agents update their dual variables in parallel with the recursion

$$\boldsymbol{u}_{s,t} = \boldsymbol{u}_{s,t-1} + \hat{\boldsymbol{\beta}}_{s,t-1} - \boldsymbol{M}_s \boldsymbol{z}_{s,t} \tag{130}$$

where the update is part of the dual ascent method.

Next follows the update of $\hat{\boldsymbol{\beta}}_t$ where the augmented Lagrangian is separable across all $\hat{\boldsymbol{\beta}}_{s,t} \in \Omega_S$. Hence, the update is also carried out in parallel. Due to the L_1-norm of the LASSO the Lagrangian is not differentiable with respect to $\hat{\boldsymbol{\beta}}_t$, though sub-differentiable. The final update then reads

$$\hat{\boldsymbol{\beta}}_{s,t} = \mathbb{S}_{\lambda/\rho} \left(\boldsymbol{M}_s \boldsymbol{z}_{s,t} - \boldsymbol{u}_{s,t} \right) \tag{131}$$

where $\mathbb{S}_\kappa (c)$ is a soft-thresholding operator

$$\mathbb{S}_\kappa (c) = \begin{cases} c - \kappa & \text{if c} > 0 \text{ and } \kappa < |c| \\ c + \kappa & \text{if c} < 0 \text{ and } \kappa < |c| \\ 0 & \text{if } \kappa > |c| \end{cases} \tag{132}$$

which is applied element-wise to the input $\boldsymbol{M}_s \boldsymbol{z}_{s,t} - \boldsymbol{u}_{s,t}$.

iii. Back to the central agent

After each agent updates their model coefficient estimates $\hat{\boldsymbol{\beta}}_{s,t}$, they compute the partial prediction $\boldsymbol{a}_{s,t} \hat{\boldsymbol{\beta}}_{s,t}$ with the latest explanatory variables. Besides sharing the partial prediction with the central agent, due to the z-update the algorithm also requires each contracted agent to share $\boldsymbol{a}_{s,t} \boldsymbol{M}_s$ and $\boldsymbol{M}_s^\top \left(\hat{\boldsymbol{\beta}}_{s,t} + \boldsymbol{u}_{s,t} \right)$ with the central agent. $\boldsymbol{M}_s^\top \boldsymbol{M}_s$ are also required by the central agent but only have to be shared once.

The last step before obtaining the next prediction requires the central agent to sum all partial predictions, i.e.

$$\hat{y}_{s_j, t+1|t} = \sum_{s \in \Omega_S} \boldsymbol{a}_{s,t} \hat{\boldsymbol{\beta}}_{s,t} \tag{133}$$

Because the OADMM requires each contracted agent to share the prior stated vectors and scalar with the central agent, the central agent has access to $L^2 + L + 1$ equations for each contracted agent and time stamp. The central agent cannot retrieve the elements of $\boldsymbol{a}_{s,t}$ because the obtained equations contain $2(L^2 + L)$ unknowns. Therefore, the data privacy of the contracted agents is protected. Besides protecting the data of each wind farm operator, the OADMM requires only a single bidirectional data exchange between the central agent and its contracted agents. Taking into consideration that only low-dimensional vectors and matrices are exchanged, the algorithm is efficient communication-wise. The pseudocode of the OADMM's final version is presented in Algorithm 3.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Algorithm 3 Online ADMM

Central agent decides on λ, ρ, ν and L. Initialize $\hat{\boldsymbol{\beta}}_{s,0}$, $\boldsymbol{z}_{s,0}$ and $\boldsymbol{u}_{s,0}$ for $s \in \Omega_s$, $\boldsymbol{H}_0$, $\boldsymbol{P}_0$ and t to be 0. To build $\boldsymbol{M}^\top \boldsymbol{M}$, contracted agents share $\boldsymbol{M}_s^\top \boldsymbol{M}_s$ with the central agent j.
while agents want to perform distributed online learning **do**
  $t := t + 1$
  $y_{s_j,t}$ is revealed to the central agent
  $\boldsymbol{H}_t := \nu \boldsymbol{H}_{t-1} + (\boldsymbol{a}_{t-1}\boldsymbol{M})^\top (\boldsymbol{a}_{t-1}\boldsymbol{M})$
  $\boldsymbol{p}_t := \nu \boldsymbol{p}_{t-1} + y_{s_j,t} (\boldsymbol{a}_{t-1}\boldsymbol{M})$
  Central agent updates $\boldsymbol{z}_{t-1}$ and distributes local values to contracted agents
  $\left(\boldsymbol{H}_t + \rho \boldsymbol{M}^T \boldsymbol{M}\right) \boldsymbol{z}_t = \boldsymbol{p}_t + \rho \boldsymbol{M}^\top (\hat{\boldsymbol{\beta}}_{t-1} + \boldsymbol{u}_{t-1})$
  $[\boldsymbol{z}_{s_1,t}, ..., \boldsymbol{z}_{s_S,t}] := \boldsymbol{z}_t$
  for $s \in \Omega_s$ **do**
    $\boldsymbol{u}_{s,t} := \boldsymbol{u}_{s,t-1} + \hat{\boldsymbol{\beta}}_{s,t-1} - \boldsymbol{M}_s \boldsymbol{z}_{s,t}$
    $\hat{\boldsymbol{\beta}}_{s,t} := \mathbb{S}_{\lambda/\rho} (\boldsymbol{M}_s \boldsymbol{z}_{s,t} - \boldsymbol{u}_{s,t})$
    Agent s uses latest observation $y_{s,t}$ to form $\boldsymbol{a}_{s,t}$
    $\widetilde{y}_{s,t+1|t} := \boldsymbol{a}_{s,t} \hat{\boldsymbol{\beta}}_{s,t}$
    share $\widetilde{y}_{s,t+1|t}$, $\boldsymbol{a}_{s,t}\boldsymbol{M}_s$ and $\boldsymbol{M}_s^\top \left(\hat{\boldsymbol{\beta}}_{s,t} + \boldsymbol{u}_{s,t}\right)$ with central agent
  end for
  $\hat{y}_{s_j,t+1|t} := \sum_{s \in \Omega_s} \widetilde{y}_{s,t+1|t}$
  Central agent j stacks local $\boldsymbol{a}_{s,t}\boldsymbol{M}_s$ and $\boldsymbol{M}_s^\top \left(\hat{\boldsymbol{\beta}}_{s,t} + \boldsymbol{u}_{s,t}\right)$
  $\boldsymbol{a}_t \boldsymbol{M} := [\boldsymbol{a}_{s_1,t}\boldsymbol{M}_{s_1}, ..., \boldsymbol{a}_{s_S,t}\boldsymbol{M}_{s_S}]$
  $\boldsymbol{M}^\top \left(\hat{\boldsymbol{\beta}}_t + \boldsymbol{u}_t\right) := \left[\boldsymbol{M}_{s_1}^\top \left(\hat{\boldsymbol{\beta}}_{s_1,t} + \boldsymbol{u}_{s_1,t}\right), ..., \boldsymbol{M}_{s_S}^\top \left(\hat{\boldsymbol{\beta}}_{s_S,t} + \boldsymbol{u}_{s_S,t}\right)\right]$
end while

Considering all 5 required algorithm parameter updates, due to their low complexity it is expected that the β-, u- and covariance structure updates can be performed efficiently and quickly. However, the z-update is more expensive because a linear system is solved which grows linearly with the number of agents S and the order of the AR process L. Therefore, for large-scale applications with hundreds or thousands of contracted agents the z-update becomes time-intensive. This motivated us to develop a computational-wise lighter algorithm which can perform all parameter updates quickly in very large learning networks as well.

V.4 Adaptive Distributed MIrror Descent Algorithm made Sparse (Adaptive D-MIDAS)

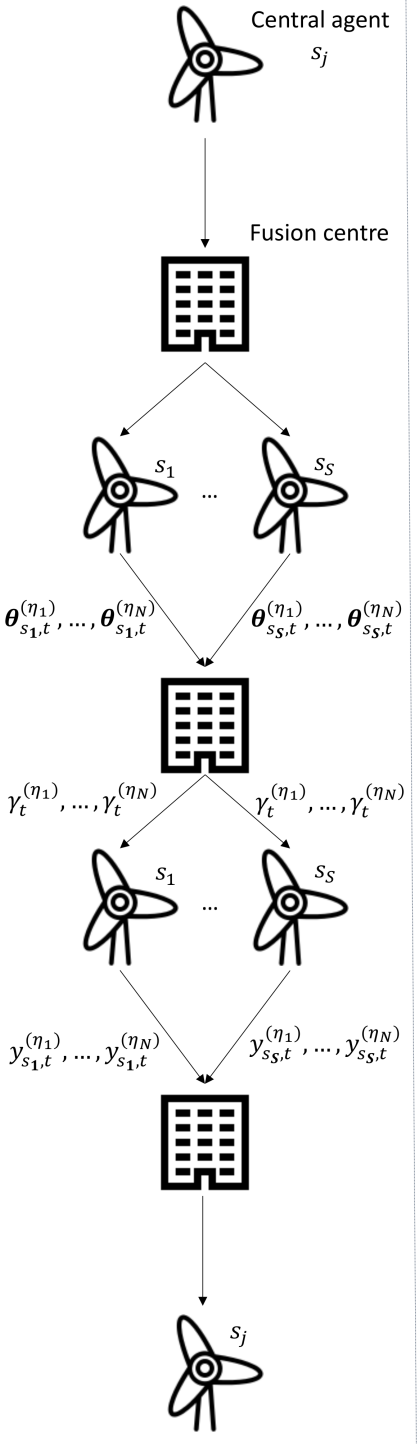
In the following, we first present basic concepts about stochastic gradient descent algorithms, which are of relevance to the proposal of an online distributed learning algorithm. A flowchart for the resulting Adaptive Distributed MIrror Descent Algorithm made Sparse (abbreviated to Adaptive D-MIDAS) is presented in Figure 41, and a detailed algorithm is available in Algorithm 4.

V.4.1 Basics of the SMIDAS

Stochastic gradient descent algorithms provide a great platform for designing computationally inexpensive online distributed learning methods. We derive in the following an algorithm that is greatly influenced by the work of (Shalev-Shwartz and Tewari, 2011). The authors proposed the Stochastic MIrror Descent Algorithm made Sparse (SMIDAS) for solving problems of the form

$$\min_{\boldsymbol{\beta}_{s_1},...,\boldsymbol{\beta}_{s_S}} C\left(\boldsymbol{\beta}_{s_1}, \ldots, \boldsymbol{\beta}_{s_S}\right) + \lambda \sum_{s \in \Omega_S} \|\boldsymbol{\beta}_s\|_1 \tag{134}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337



The *central agent* receives y_t. Then the forecast error $r_{t|t-1}^{(\eta)}$ and the cumulative absolute error $CAE_t^{(\eta)}$ are computed for all $\eta \in \Omega_\eta$

The central agent uses the key $\mathbb{K}$ to encrypt all $r_{t|t-1}^{(\eta)}$ and shares them then with the fusion centre operator

The *fusion centre operator* forwards the encrypted forecast errors to all contracted agents

After decrypting the forecast errors with $\mathbb{K}^*$, *all agents* compute the dual variables $\boldsymbol{\theta}_{s,t}^{(\eta)}$ for all $\eta \in \Omega_\eta$

All agents share all dual variables $\boldsymbol{\theta}_{s,t}^{(\eta)}$ with the fusion centre operator

The *fusion centre operator* computes the denominator of the link function, $\gamma_t^{(\eta)}$, for all $\eta \in \Omega_\eta$

The *fusion centre operator* shares the denominators with all agents

All agents compute $\widehat{\boldsymbol{\beta}}_{s,t}^{(\eta)}$ *and* $\tilde{y}_{s,t+1|t}^{(\eta)}$ for all $\eta \in \Omega_\eta$

All contracted agents share all $\tilde{y}_{s,t+1|t}^{(\eta)}$ with the fusion centre operator

The *fusion centre operator* forwards all partial predictions to the central agent

The central agent computes for all $\eta \in \Omega_\eta$ $\hat{y}_{s,t+1|t}^{(\eta)} = \sum_{s\epsilon\Omega_S} \tilde{y}_{s,t+1|t}^{(\eta)}$. The prediction $\hat{y}_{s_j,t+1|t}$ is obtained as $\min_\eta CAE_t^{(\eta)}$

Figure 41 Flowchart for the Adaptive Distributed MIrror Descent Algorithm made Sparse (Adaptive D-MIDAS) approach for online distributed learning applied to wind power forecasting.

where in regression problems C is commonly the squared loss

$$C\left(\boldsymbol{\beta}_{s_1}, \ldots, \boldsymbol{\beta}_{s_S}\right) = \sum_{\tau=1+L}^{T} \left(\sum_{s\in\Omega_S} \boldsymbol{a}_{s,\tau-1}\boldsymbol{\beta}_s - y_{s_j,\tau} \right)^2 \tag{135}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

The proposal of (Shalev-Shwartz and Tewari, 2011) is motivated by previous work on stochastic optimisation for L_1-regularized problems. First, (Duchi et al., 2008) described an algorithm which replaces the L_1 regularisation term in (134) with the constraint $\|\sum_{s\in\Omega_S}\boldsymbol{\beta}_s\|_1 \leq B$ and then uses a stochastic gradient projection procedure to estimate the model coefficients. Second, (Langford et al., 2009) introduced a stochastic gradient descent algorithm where sparse solutions are obtained by truncating the model coefficients, i.e., elements in the model coefficient vector that cross 0 during a gradient step are truncated to 0. The runtime of both algorithms might grow in some situations in a quadratic way with the dimension of the feature space even though the optimal coefficient vector is very sparse (Shalev-Shwartz and Tewari, 2011). Mirror descent algorithms instead achieve a runtime which is linear in the dimension of the feature space of the problem (Beck and Teboulle, 2003). This makes them particularly suitable for high-dimensional learning. However, they do not necessarily yield sparse solutions. In a nutshell, the SMIDAS uses mirror descent updates in combination with the truncation method of (Langford et al., 2009). Hence, the SMIDAS achieves a superior runtime compared to the algorithms of (Duchi et al., 2008) and (Langford et al., 2009) while still yielding sparse solutions. Based on these properties we use the SMIDAS as a starting point for the proposal of an online distributed learning approach.

In the following, we first apply the SMIDAS to learn the time-invariant model coefficients of (134). This will facilitate the understanding of the subsequent derivation of our algorithm for learning time-varying model coefficients in a distributed setting.

V.4.2 Batch estimation with SMIDAS

Like most gradient-based optimisation methods, the algorithm in (Langford et al., 2009) updates only one weight vector $\boldsymbol{\beta}$ every iteration. Mirror descent algorithms are conceptually different because they maintain two weight vectors, the primal vector $\boldsymbol{\beta}$ and the dual vector $\boldsymbol{\theta}$. The mirror descent algorithm was first derived in (Nemirovski and Yudin, 1983), while a new derivation is presented in (Beck and Teboulle, 2003). We recommend both works for a more detailed description of the algorithm.

The two weight vectors are linked via the transformation $\boldsymbol{\theta} = f(\boldsymbol{\beta})$, where f is a link function. From the derivation in (Nemirovski and Yudin, 1983) and under the right conditions, f is invertible. Hence, the inverse transformation $\boldsymbol{\beta} = f^{-1}(\boldsymbol{\theta})$ exists. In (Shalev-Shwartz and Tewari, 2011) a p-norm link function is used, which writes

$$\beta_n = f_n^{-1}(\boldsymbol{\theta}) = \frac{\operatorname{sign}(\theta_n)|\theta_n|^{p-1}}{\|\boldsymbol{\theta}\|_p^{p-2}} \tag{136}$$

with

$$\|\boldsymbol{\theta}\|_p = \left(\sum |\theta_n|^p\right)^{\frac{1}{p}} \tag{137}$$

and θ_n being the n$^{\text{th}}$ element in $\boldsymbol{\theta}$.

After this initial description of the mirror descent algorithm, let us apply the SMIDAS to learn the time-invariant model coefficients of (134), in a centralised setup where the central agent receive the explanatory variables from all its contracted agents.

At each iteration k, the algorithm uniformly samples a training example $i \in \{1+L,\ldots,T\}$. Then, the gradient of the squared loss function C is estimated with

$$\nabla C\left(\hat{\boldsymbol{\beta}}_1^{(k-1)},\ldots,\hat{\boldsymbol{\beta}}_S^{(k-1)}\right) = 2\left(\sum_{s\in\Omega_S}\boldsymbol{a}_{s,i-1}\right)^{\top}\left(\sum_{s\in\Omega_S}\boldsymbol{a}_{s,i-1}\hat{\boldsymbol{\beta}}_s^{(k-1)} - y_{s_j,i}\right) \tag{138}$$

where $\hat{\boldsymbol{\beta}}_s^{(k-1)}$ are the estimated model coefficients of the previous iteration and agent s. Next,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

the estimated gradient is used in

$$\widetilde{\boldsymbol{\theta}}_s^{(k)} = \boldsymbol{\theta}_s^{(k-1)} - \eta \nabla C\left(\boldsymbol{\beta}_s^{(k-1)}\right) , \quad \forall s \in \Omega_S \tag{139}$$

to update the dual variables, where $\eta > 0$ is a fixed learning rate. The SMIDAS then applies the truncation step

$$\theta_{s,j}^{(k)} = \text{sign}\left(\widetilde{\theta}_{s,j}^{(k-1)}\right) \max\left(0, |\widetilde{\theta}_{s,j}^{(k-1)}| - \eta\lambda\right) , \quad \forall j \in \{1, \ldots, L\}, \forall s \in \Omega_S \tag{140}$$

where the regularisation strength λ pulls the dual variables towards 0. As soon as a coefficient crosses 0, it is truncated to 0. This procedure is conceptually the same as in (Langford et al., 2009), though applied to the dual variables of the mirror descent instead of to the primal variables of the stochastic gradient descent. The last step of the SMIDAS applies the p-norm link function

$$\hat{\boldsymbol{\beta}}_1^{(k)}, \ldots, \hat{\boldsymbol{\beta}}_S^{(k)} = f^{-1}\left(\boldsymbol{\theta}_1^{(k-1)}, \ldots, \boldsymbol{\theta}_S^{(k-1)}\right) \tag{141}$$

to update the model coefficient estimates.

Equations (138) to (141) are applied until a defined convergence criterion is reached. Depending on the dataset size and convergence criterion, the algorithm can sample a single training example multiple times.

V.4.3 Online distributed MIDAS

The motivation to use the SMIDAS as the basis for our distributed online algorithm comes from the separability of (138) in the explanatory variables and the possibility of obtaining sparse model coefficient vectors through the truncation step (140). By choosing the loss function C as the quadratic criterion, the term $\sum_{s\in\Omega_S} \boldsymbol{a}_{s,i-1}\hat{\boldsymbol{\beta}}_s^{(k-1)} - y_{s_j,i}$ in (138) is the 1-step ahead forecast error of iteration k. Hence, if the central agent shares the forecast error for its site with the contracted agents, each agent is able to estimate their share of the gradient locally. Given the remaining steps of the SMIDAS, this allows, with a few modifications only, the obtainment of a distributed online algorithm where the privacy of each agent is protected. However, one might argue that the forecast error is a private information for the central agent, who hence is not willing to share it. In situations where the central agent is not willing to share forecast errors with competitors, we propose the following distributed learning network design. When sharing the forecast error through a fusion centre with contracted agents, the forecast error is anonymised such that the contracted agents cannot infer the identity of the central agent, and hence cannot identify the location of the related site. The anonymisation of the central agent would further require that potential compensations for the participation in a learning agreement are handled by the fusion centre operator.

To obtain a learning algorithm which is able to track time-varying model coefficients, our algorithm does not randomly sample training examples. Instead, the algorithm estimates the gradient for a sample only once as observations arrive sequentially. Therefore, the index t replaces i in all previous SMIDAS formulations. We further change the name of the algorithm to MIDAS because we remove the stochasticity by not using random samples. Last, we remove the iteration counter k (t is the equivalent in online learning) from all formulations. Estimating the gradient of a sample only once is fundamentally different from the OADMM where the cumulative loss is minimised over all past observations. This means that, when updating the model coefficients, all past information is implicitly considered. Consequently, when using the distributed MIDAS version, we expect a greater variance in the estimated model coefficients.

Starting from the willingness of the central agent to share its forecast error $r_{t|t-1} = \hat{y}_{s_j,t|t-1} - y_{s_j,t}$ with the contracted agents at time t, we propose the following distributed MIDAS. Instead of

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

sharing the forecast error directly with the contracted agents, it is shared through a fusion centre. This is considered to be the first broadcasting step. Each agent then updates their local dual variables by taking a step into the direction of the negative estimated gradient while controlling the step size with the learning rate η. The update is performed by all agents in parallel and is written as

$$\widetilde{\boldsymbol{\theta}}_{s,t} = \boldsymbol{\theta}_{s,t-1} - \eta \boldsymbol{a}_{s,t-1} r_{t|t-1} \tag{142}$$

The SMIDAS subsequently uses the element-wise truncation (140). A simulation study revealed that a single sample evaluation does not yield significant benefits in terms of forecast accuracy. Furthermore, we realised that the p-norm link function is sufficient to shrink unimportant model coefficients to 0, though the estimated model coefficients never became exactly 0. Therefore, we dismissed the option to obtain sparse coefficient vectors by neglecting the truncation step in our distributed version (i.e., $\widetilde{\boldsymbol{\theta}}_{s,t}$ is hereafter replaced by $\boldsymbol{\theta}_{s,t}$). We subsequently obtain an algorithm that has one less hyperparameter. However, the inability to shrink unimportant model coefficients to 0 is a setback if compared to the case of OADMM.

The next step of the algorithm utilises the fusion centre, with which all agents share their dual variables. This allows the computation of the denominator of the link function with

$$\gamma_t = \|\boldsymbol{\theta}_t\|_p^{p-2} \tag{143}$$

where $\boldsymbol{\theta}_t$ is the assembly of all local dual vectors $\boldsymbol{\theta}_{s,t} \in \Omega_S$ and p is a hyperparameter. This step marks the first gathering step, even though the local variables are not gathered by the central agent. The norm γ_t is consequently shared with all agents such that they can apply the link function to their respective dual variables. This is the second broadcasting step of the algorithm. The final update is the element-wise application of the link function

$$\hat{\boldsymbol{\beta}}_{s,t} = \frac{\operatorname{sign}(\boldsymbol{\theta}_{s,t})|\boldsymbol{\theta}_{s,t}|^{p-1}}{\gamma_t} \tag{144}$$

The aforementioned shrinkage behaviour of the link function is controlled via the hyperparameter p, where a greater value in p applies a greater shrinkage to all $\hat{\boldsymbol{\beta}}_{s,t}$'s.

After obtaining re-estimated model coefficients, each agent calculates a new partial prediction and shares it through the fusion centre with the central agent, who eventually calculates the next prediction for its site. The final exchange of information accounts for the second gathering step. In total the algorithm requires 2 broadcasting and 2 gathering steps for each t.

V.4.4 Extending the distributed MIDAS

With the distributed MIDAS version, the fusion centre operator could have the possibility to retrieve the information about local explanatory variables. This possibility exists since the access to all dual variables and the forecast errors results in an equal amount of equations and unknowns. Therefore, additional measures are required to protect the data of the wind farm operators. Due to the different structure of the algorithms, introducing an encryption matrix as in the OADMM was unsuccessful. Our proposal is therefore to encrypt the forecast errors using an encryption technique such as AES (Li et al., 2009) and then share it through the fusion centre with the contracted agents. This requires the direct exchange of the decryption key with the contracted agents before all agents start performing online learning. Because the fusion centre operator does not have access to the forecast error anymore, it has more unknowns than equations to solve. Hence, it is not possible to retrieve the explanatory variables with sufficient accuracy. In a setting with anonymized forecast errors, the central agent still shares the decryption key with its contracted agents. However, the central agent does not reveal its identity and therefore the contracted agents cannot obtain information about the location of the central agent's site.

In the presented algorithm, named Distributed MIDAS (D-MIDAS), the learning rate η controls the general speed with which the algorithm approaches the global optimum of the minimisation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

problem within a given period. When η is large, the global optimum is approached faster but at the same time the estimated model coefficients experience a greater variance between consecutive time stamps. This statement is derived from (142), where a large forecast error translates directly to a significant change in the dual variables, and subsequently to a notable change in the model coefficients. Ideally, in stationary periods the algorithm requires smaller η values compared to non-stationary periods. Taking into account that all algorithm parameter updates are computationally cheap, our proposal is to learn multiple AR-X models in parallel while varying the learning rate η between the models.

Based on the past performance of each model, the algorithm adaptively chooses which model to use for the next prediction. This can considered as adaptive learning, where in stationary periods a small η is used, and in non-stationary periods a larger η is applied instead. We use the cumulative absolute error (CAE) with decaying weights

$$CAE_t^{(\eta)} = \mu CAE_{t-1}^{(\eta)} + |\hat{y}_{s_j,t|t-1}^{(\eta)} - y_{s_j,t}| \tag{145}$$

to evaluate the performance of each model, where μ allows the control of the level of decay. The superscript η indicates from which model the prediction is coming from. We name this extension Adaptive D-MIDAS and the pseudocode is shown in Algorithm 4.

The computational complexity and the amount of exchanged data increases linearly with the number of models that the Adaptive D-MIDAS learns in parallel. Concerning the amount of exchanged data, the Adaptive D-MIDAS exchanges an almost equal amount of data as the OADMM when learning two models in parallel. However, the Adaptive D-MIDAS requires two bidirectional data exchange steps whereas the OADMM requires only one. The additional data exchange step comes from the requirement to compute the denominator of the link function at the fusion centre. Based on this insight we obtain a communication-reduced version of the algorithm by using the denominator and dual variables of the previous time stamp to update the model coefficients via the link function. Consequently, it is no longer necessary to send the dual variables to the fusion centre before updating the weight vector. The denominator of the link function can instead be calculated after the central agent has calculated the next prediction for its site. With this strategy that reduces the overall time between obtaining the newest observation and calculating a new prediction with re-estimated model coefficients, it is expected that a negative impact on forecast accuracy will be observed. However, as the later following case study using real-world data shows, the reduction in forecast accuracy is small.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Algorithm 4 Adaptive D-MIDAS

Central agent creates decryption key $\mathbb{K}^*$, selects a set of learning rates (collected in Ω_η) and shares both quantities with its contracted agents.
Initialize t and $\hat{\boldsymbol{\beta}}_{s,0}^{(\eta)}$, $\boldsymbol{\theta}_{s,0}^{(\eta)}$ for $s \in \Omega_s$ and $\eta \in \Omega_\eta$ to be 0. Additionally, initialize $CAE_{s_j}^{(\eta)}$ for $\eta \in \Omega_\eta$ to be 0. Central agent selects μ and p but only shares p with the fusion centre and contracted agents.
while agents want to perform distributed online learning **do**
  $t := t + 1$
  $y_{s_j,t}$ is revealed to the central agent
  for $\eta \in \Omega_\eta$ **do**
    $r_{t|t-1}^{(\eta)} := \hat{y}_{s_j,t|t-1}^{(\eta)} - y_{t,s_j}$
    $CAE_t^{(\eta)} = \mu CAE_{t-1}^{(\eta)} + |r_{t|t-1}^{(\eta)}|$
  end for
  Central agent encrypts forecast errors with $\mathbb{K}(\cdot)$ and shares them through the fusion centre with its contract agents
  for $s \in \Omega_s$ **do**
    Agent s decrypts forecast errors with $\mathbb{K}^*(\cdot)$
    for $\eta \in \Omega_\eta$ **do**
      $\boldsymbol{\theta}_{s,t}^{\eta} := \boldsymbol{\theta}_{s,t-1}^{(\eta)} - \eta \cdot \boldsymbol{a}_{s,t-1} r_{t|t-1}^{(\eta)}$
    end for
    Transmit list of dual vectors to fusion centre
  end for
  Fusion centre operator computes denominators of link function, $\gamma_t^{(\eta)}$
  for $\eta \in \Omega_\eta$ **do**
    $\boldsymbol{\theta}_t^{(\eta)} := \left[\boldsymbol{\theta}_{s_1,t}^{(\eta)}, ..., \boldsymbol{\theta}_{s_S,t}^{(\eta)}\right]$
    $\gamma_t^{(\eta)} := \|\boldsymbol{\theta}_t^{(\eta)}\|_p^{p-2}$
  end for
  Fusion centre operator shares $\gamma_t^{(\eta)}$ with all agents
  for $s \in \Omega_s$ **do**
    Agent s uses latest observation $y_{s,t}$ to form $\boldsymbol{a}_{s,t}$
    for $\eta \in \Omega_\eta$ **do**
      $\forall\ k,\ \hat{\beta}_{s,t,k}^{(\eta)} := \frac{\text{sign}\,(\theta_{s,t,k}^{(\eta)})|\theta_{s,t,k}^{(\eta)}|^{p-1}}{\gamma_t^{(\eta)}}$
      $\widetilde{y}_{s,t|t-1}^{(\eta)} := \boldsymbol{a}_{s,t}\hat{\boldsymbol{\beta}}_{s,t}^{(\eta)}$
      Each contracted agent shares partial predictions through fusion centre with central agent
    end for
  end for
  for $\eta \in \Omega_\eta$ **do**
    $\hat{y}_{s_j,t|t-1}^{(\eta)} := \sum_{s\in\Omega_s} \widetilde{y}_{s,t|t-1}^{(\eta)}$
  end for
  Central agent selects final prediction according to $\min_\eta MAE_{s_j}^{(\eta)}$
end while

V.5 Simulation study

A study on simulated data investigates the ability of both algorithms to estimate time-varying model coefficients and the related computational costs. We only consider the standard Adaptive D-MIDAS and not its communication-reduced version since the lagged calculation of the denominator γ_t was verified to have only a small impact on the estimated model coefficients.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

V.5.1 Tracking of time-varying coefficients

We first generate a multivariate time series with time-varying coefficients of the form

$$\boldsymbol{y}_t = \boldsymbol{A}\boldsymbol{y}_{t-1} + \boldsymbol{\epsilon}_t \tag{146}$$

where $\boldsymbol{\epsilon}_t$ is a vector of independent standard Gaussian noise with 0 mean and finite variance (set to 0.1 in our experiments). Each simulated time series has 25 000 times steps. The coefficient matrix $\boldsymbol{A}$ is further defined as

$$\boldsymbol{A} = \begin{bmatrix} 0.9 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.8 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.85 & 0 & 0 & 0 & 0 & -0.15 & 0 & 0 \\ 0 & 0 & 0 & 0.75 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_1 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & a_2 & 0 & 0 & 0 & 0.9 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.85 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.1 & 0 & 0 & 0 & 0.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.9 & 0 \\ 0 & 0 & 0 & 0 & 0.15 & 0 & 0 & 0 & 0 & 0.8 \end{bmatrix} \tag{147}$$

where a_1 and a_2 are time-varying coefficients (as illustrated in Figure 42). We performed a Monte-Carlo simulation with 1 000 replicates to estimate the variance of $\hat{a}_1$ and $\hat{a}_2$. The hyperparameters p and μ of the Adaptive D-MIDAS were set to 2.5 and 0.996, respectively. In this experiment we used 6 evenly spaced (from 0.025 to 0.15) learning rates.

Figure 42 shows the temporal evolution of the mean, as well as the 5*th* and 95*th* quantiles of the estimated coefficient distributions for a_1 and a_2. Both algorithms are able to track the time-varying coefficients in expectation (the mean of the estimated coefficient distributions follows the true values). However, some noticeable differences are observed between both algorithms. First of all, the Adaptive D-MIDAS has a greater "burn-in" period, i.e., the number of samples required to learn from before a fair approximation of the true coefficient value is reached. Secondly, the spread in the estimated model coefficients for the 1 000 replicates (represented by the difference between both quantiles) increases for the Adaptive D-MIDAS when the true coefficients change. Contrary to this, the spread in coefficient estimation for the OADMM is either constant or even decreases for changing true coefficient values. The increased spread for the Adaptive D-MIDAS coefficient estimates is a consequence of the faster learning rate which is required to keep track of the decreasing true coefficient value.

This can also be observed from Figure 43, which shows the average (over all 1 000 Monte-Carlo replicates) learning rate of the Adaptive D-MIDAS. It reveals a clear relationship between the spread in the coefficient estimates and the best-performing learning rate (based on (145)).

Thus, there is a clear trade-off for the Adaptive D-MIDAS between the variation in the estimated model coefficients and the ability to track time-varying coefficients: when trying to achieve a high degree of adaptivity, one has to pay the price of higher variation in the estimated coefficients.

A similar trade-off is observed for the OADMM, where the adaptivity is controlled by the forgetting factor ν. When applying a smaller ν value, the algorithm becomes more adaptive but since the effective training data length decreases the variance in the estimated coefficient increases. However, the OADMM provides a better trade-off between adaptivity and estimated coefficient variance due to the fact that it minimises the cumulative loss over all past observations. Thus,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

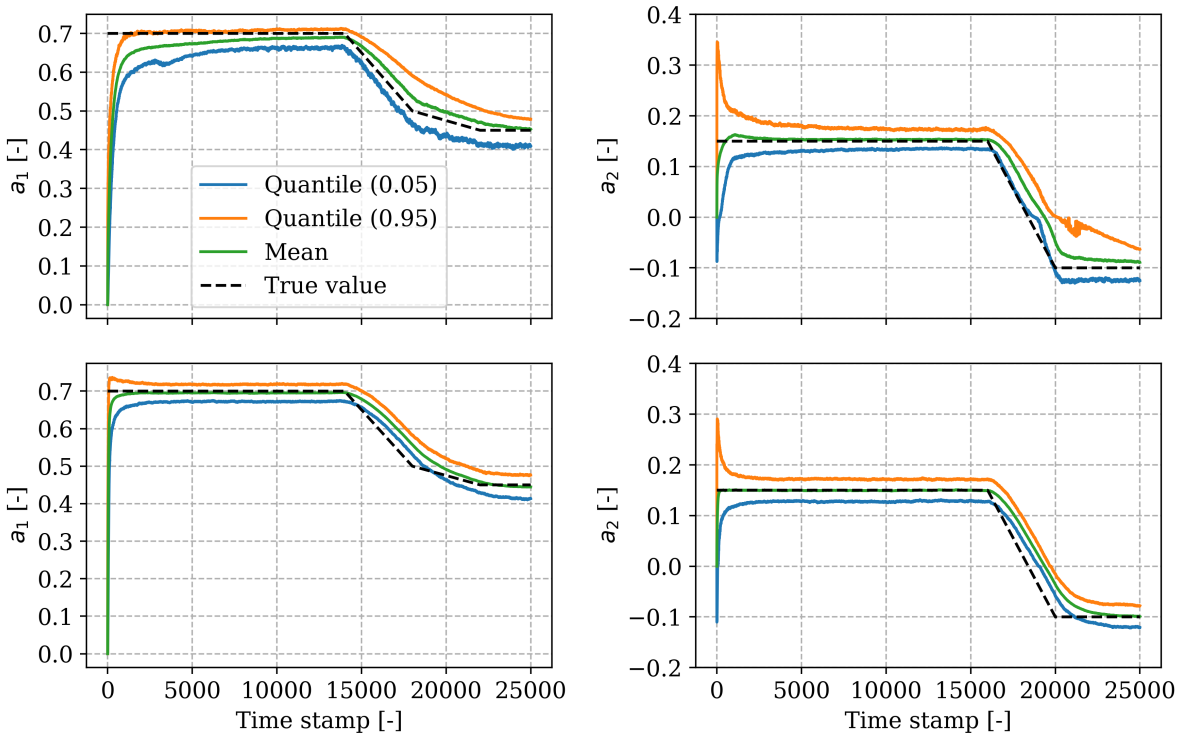


Figure 42 Coefficient estimates obtained through the Monte-Carlo simulation. Top row: Adaptive D-MIDAS, bottom row: OADMM

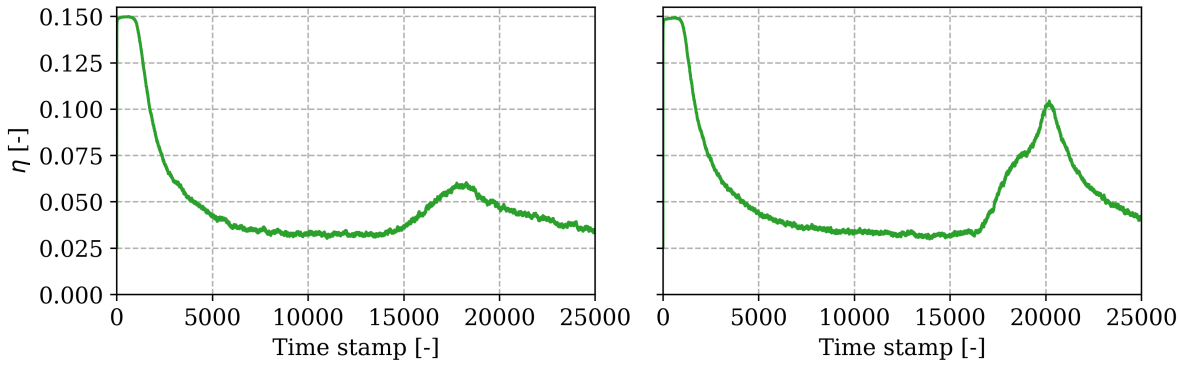


Figure 43 Average learning rate of the Adaptive D-MIDAS across all 1 000 replicates. Left a_1 right a_2

occasional outliers in the form of large forecast errors have a smaller impact on the estimated coefficients.

V.5.2 Computational costs

Besides assessing the ability of both algorithms to track time-varying model coefficients, a simulation study is performed to compare the computational costs. The study estimates the time required by each agent and algorithm to calculate a new prediction after the central agent obtain a new data sample. To show the expected better scaling properties of the D-MIDAS, we estimated the computational time for an increasing learning network size of contracted agents.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

We again used simulated time series data that are generated with the previously introduced approach. However, instead of creating an AR(1)-, we used an AR(4)-process. In this study we performed online learning for 1 000 simulated time steps while recording the time it took to complete each operation. To achieve comparability between the Adaptive D-MIDAS and OADMM, the Adaptive D-MIDAS learned two models in parallel. For both algorithms, this resulted in an almost equal amount of data that was exchanged within the learning network. The simulation study was performed on a system with a i5-5200U CPU, 8 GB DDR3 RAM and a Windows 10 OS. Because both algorithms were used locally, we neglected the encryption and decryption steps of the Adaptive D-MIDAS. Hence, the observed performance gap in computational speed would decrease in case encryption was required to ensure data privacy.

Computational times are summarised in Figure 44. They were obtained by averaging the computational time for each and every one of the 1 000 time steps. The Adaptive D-MIDAS is faster than the OADMM overall. Furthermore, the algorithm shows a better scaling behaviour with respect to the learning network size. This is expected since, at each and every time t, the OADMM needs to solve a linear system of equations. Depending on the solving technique, complexity grows at least quadratically with the number of equations. The Adaptive D-MIDAS scales better because its updates are simple linear operations.

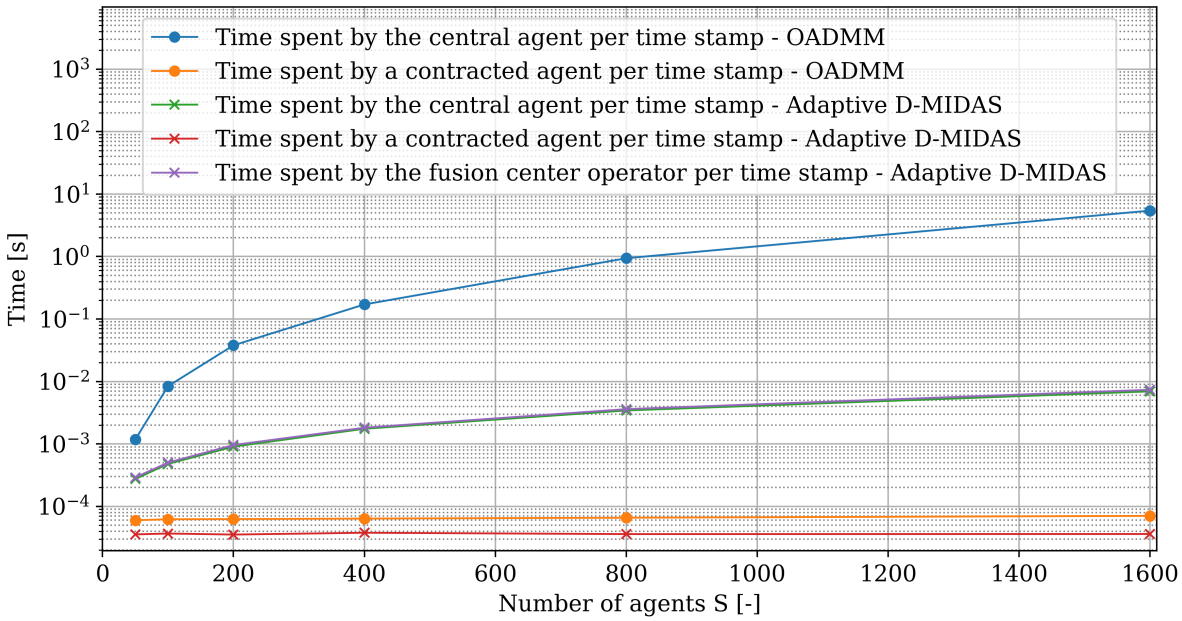


Figure 44 Average time (over 1 000 time steps) required by each agent to complete its tasks at a given time step, for both OADMM and Adaptive D-MIDAS approaches, as a function of total number of agents S.

V.6 Case study

Our distributed and online learning algorithms were benchmarked on a real-world dataset of wind power generation for 311 sites. The dataset is a subset of the one used in (Girard and Allard, 2013). The temporal resolution is of 15 minutes and our subset covers 40 000 time steps, corresponding to 416 days. Figure 45 shows the location of the sites in Western Denmark. Many sites are located in close proximity to each other. This allows accounting for relevant spatial-temporal patterns when forecasting wind power generation for short lead times. To highlight the benefits of online learning, we benchmarked the forecasts from our online algorithms against those that would be otinaed from L_1-regularized AR-X models with time-invariant coefficients. The model coefficients were then estimated on the training part of the dataset only.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

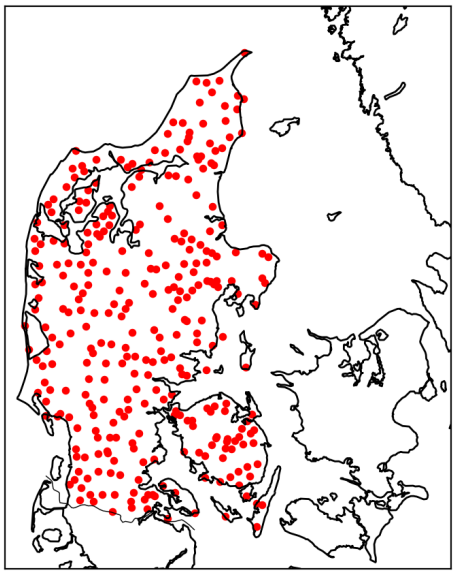


Figure 45 Location of sites in Western Denmark.

The benefits of exploring spatial-temporal patterns in wind power generation data have been shown for a different subset in (Messner and Pinson, 2018). The authors showed that high-dimensional regularised AR-X models outperform univariate AR models which only use on-site power measurements. All model coefficients were estimated in a time-varying fashion. Based on these results, we allowed ourselves to disregard univariate AR models with time-varying coefficients in our case study.

V.6.1 Data preprocessing

First, the raw data was normalised by dividing the time series of each site by the respective nominal capacity. Write $x_{s,t}$ the normalised wind power generation observed at time t and for site s. In addition, a logit-Normal transformation of the original time-series was considered, as proposed by (Lau and McSharry, 2010), i.e., at each and every time t and site s,

$$y_{s,t} = \ln\left(\frac{x_{s,t}}{1-x_{s,t}}\right), \quad \forall s,t\,. \tag{148}$$

To account for the bound effects, a coarsening approach was used (Pinson, 2012a), for which values of 0 and 1 are set to 0.01 and 0.99, respectively.

V.6.2 Case study setup

The data is split into two equal sub-periods of 20 000 time steps. The first part is used for training and hyperparameter optimisation, and the second for genuine out-of-sample forecast verification. Over the first period, the hyperparameters of all algorithms were optimized with a grid search scheme. After identifying suitable hyperparameters for the Adaptive D-MIDAS, the same hyperparameters were then applied to its communication-reduced version. The LASSO's L_1-regularisation parameter λ for the batch AR-X models was determined through 1-fold cross-validation. For both approaches, the forgetting factors are to be seen as variables that control how much of the past data is used for estimation. Hence, optimising these forgetting factors

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

through cross-validation is to be seen as equivalent to determining an optimal training set size in the case of batch learning.

For a given site s, the Mean Absolute Error (MAE),

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T}|y_{s,t} - \hat{y}_{s,t}| \tag{149}$$

and the Root Mean Squared Error (RMSE),

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_{s,t} - \hat{y}_{s,t})^2} \tag{150}$$

were used as performance metrics. To further quantify the performance of each model, the skill score I

$$I_S = 1 - \frac{S_{\text{Model}}}{S_{\text{Pers}}} \tag{151}$$

was used to assess the improvement over persistence forecasts (the latest observed power measurement is used as the next prediction), where S could be the metrics MAE or RMSE.

Owing to the large number of wind farms in the dataset, hyperparameter optimisation was performed for all wind farms at once, instead of for each site individually. A set of hyper-parameters was evaluated by considering the skill score distribution that contained the scores of all 311 sites. The median, the lower quartile and the inter-quartile range were used here as decision criteria. It was further decided to perform multi-step-ahead forecasting with up to 4 steps ahead. In general, different strategies exist for multi-step-ahead forecasting, where a good overview is presented in (Ben Taieb et al., 2012). The most common approaches are either the iterative calculations of 1-step-ahead predictions or training separate models for each lead time. The iterative calculation of 1-step-ahead predictions results in the accumulation of forecast errors. Therefore, we used the direct approach and trained models for each lead time.

The hyperparameters of the Adaptive D-MIDAS and the batch AR-X model were optimized for each of the 4 lead times. Based on the significantly longer simulation times, the hyperparameters of the OADMM were optimized for 1-step-ahead predictions only. The selected hyperparameters were then applied for all other lead times. Due to the observed longer burn-in period of the Adaptive D-MIDAS, the performance metrics were calculated for both online algorithms only for the time steps between t =10 000 and t =20 000.

To obtain predictions for all sites of the dataset, each site took the role of the central agent once, while acting as a contracted agent in the other 310 simulations (i.e., for all other sites). Therefore, to obtain predictions for all sites and a single lead time, in total 311 AR-X models were estimated.

V.6.3 Results

Focusing first on hyperparameter optimisation, Figure 46 gives an example of the results obtained by optimizing the forgetting factor μ for the Adaptive D-MIDAS approach, when performing 1-step ahead forecasting. The boxplot shows the improvement over persistence forecasts for all 311 sites, as a function of the forgetting factor μ. Each box extends from the lower to the upper quartile (denoted Q_1 and Q_3, respectively), where the horizontal line indicates the median of the obtained RMSE skill score distributions. The maximum length of the whiskers is set to 1.5 times the interquartile range (Q_3-Q_1). The upper whisker then indicates the last sample which is found to be below or equal to the threshold of $Q_3 + 1.5(Q_3 - Q_1)$. If a data point of the distribution is found outside this range, it is classified as an outlier and marked with a circle. The same concept

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

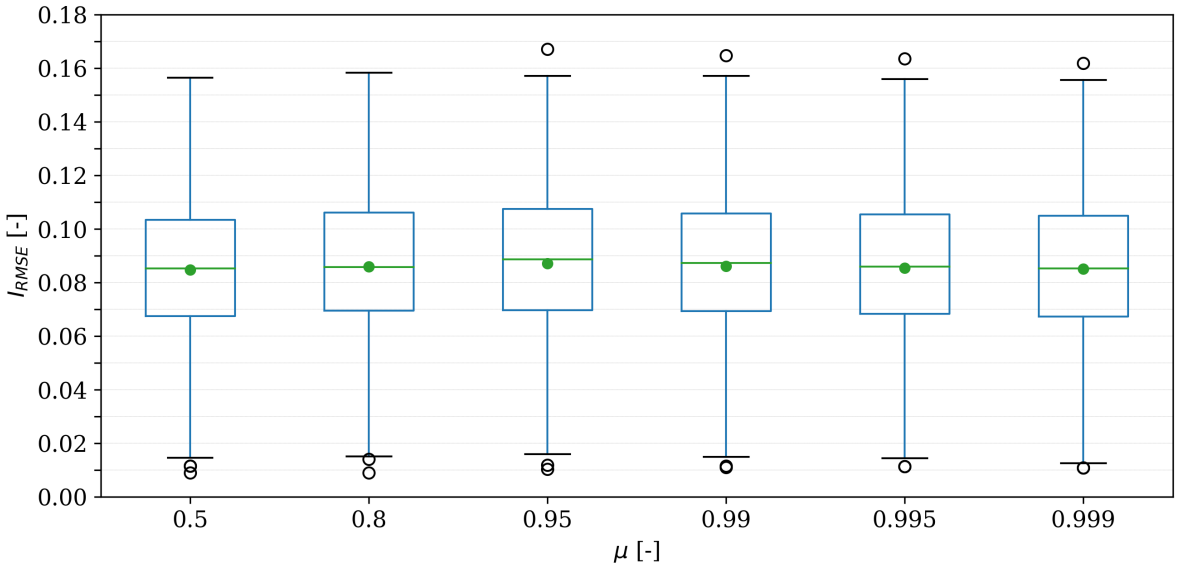


Figure 46 RMSE skill score of the Adaptive D-MIDAS with reference to the persistence forecast for 1-step ahead forecasts, as a function of the forgetting factor μ. The skill score values are computed for the time stamps $t = 10000$ to $t = 20000$.

applies to the lower whisker, which marks the first sample that is found to be within the range of $Q_1 - 1.5(Q_3 - Q_1)$.

A μ value of 0.95 performed slightly better than the remaining selected values when considering the aforementioned decision criteria. Therefore, this value was subsequently selected when performing online learning to estimate the performance on unseen data. The same approach was followed when tuning the other hyperparameters.

After finding suitable hyperparameters for both online algorithms, online learning was performed on the complete dataset. In contrast, the batch AR-X model coefficients were estimated over the first 20 000 time steps and then used to generate predictions over the remaining 20 000 time steps without re-estimating the model coefficients. Results are collated in Figure 47, for all approaches considered and for all sites, again with boxplots for skill score values (both in terms of RMSE and MAE).

Overall, all online distributed learning algorithms outperform the batch learning one (LASSO estimation in AR-X models), with the advantage that no data from contracted agents is actually shared with the central agents. A paired t-test supported the statistical significance by rejecting the null hypothesis of equal means at the 0.05 significance level. The number of outliers for the batch LASSO additionally emphasises the strength of online learning because the poor performance of batch estimation can be explained by the non-stationarity of the wind power generation time series. Hence, there are significant differences between the time-varying coefficients throughout the value period, and the coefficients estimated over and fixed at the end of the training period. Furthermore, when computing the bias it was observed that all forecasting models exhibit negligible bias values (not shown here).

The results further show that OADMM, despite being computationally more expensive, outperforms the Adaptive D-MIDAS for all lead times and skill scores. A paired t-test also supported the statistical significance of the results here. In addition, the performance gap increases for further lead times. This may be due to the structure and workings of both online algorithms. Indeed, the OADMM minimises the cumulative loss over all past observations where the covariance structures $\boldsymbol{H}_t$ and $\boldsymbol{p}_t$ carry the information of all previous samples. By varying the applied forgetting

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

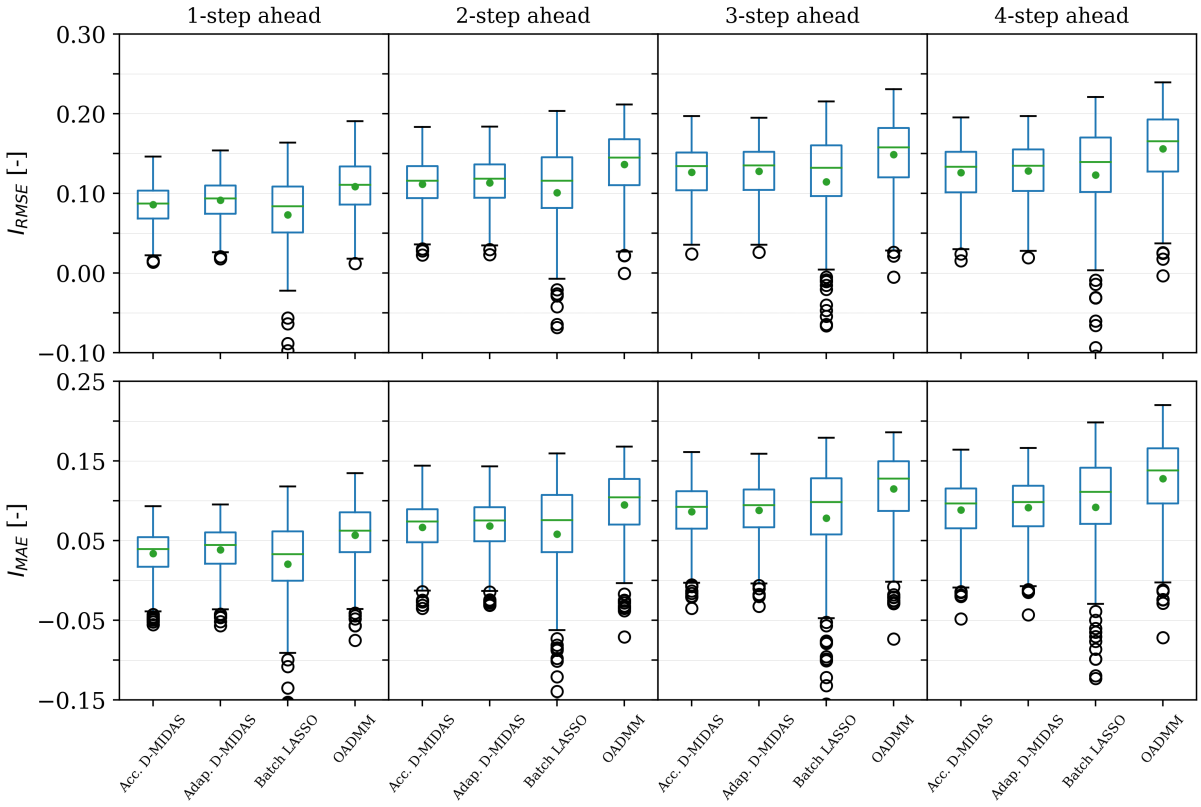


Figure 47 RMSE (top row) and MAE (bottom row) skill scores for the online distributed and batch learning approaches for different lead times. The skill score values are computed over the evaluation period, from $t = 20000$ to $t = 40000$. The dots indicate the mean of the skill score distributions.

factor the number of past samples that are effectively used to update all algorithm parameters is controlled. As a result, even when a set of successive large forecast errors are observed, the estimated model coefficients are less subject to variations. The Adaptive D-MIDAS on the other hand does not utilize covariance structures to estimate model coefficients. Instead it uses the estimated gradient of the current squared loss between the observation and prediction to re-estimate the model coefficients. Therefore, large forecast errors directly translate to noticeable variations in the estimated model coefficients. A resulting shortcoming may be that the algorithm could be highly sensitive to outliers and structural breaks in the time series. While for 1-step-ahead predictions the model coefficient update is performed right after, i.e., naturally 1 time step after the prediction is made, for greater lead times there is a time lag because one must wait k steps to obtain the forecast error of a k-step-ahead forecast. This, paired with the sensitivity with respect to large forecast errors, may explain the lower forecast accuracy of the Adaptive D-MIDAS for further lead times. As mentioned earlier within the study on simulated data, the OADMM deals essentially better with this condition since the covariance structures carry an inertia whereby single or multiple large forecast errors do not affect the model coefficient estimates as much. Here it should be noted that this statement only holds for a sufficiently large forgetting factor.

Finally, one can verify that the communication-reduced Adaptive D-MIDAS version (Acc. D-MIDAS) does not perform significantly worse than the standard version. Thus, this version is an alternative in applications where re-estimating the model coefficients as quickly as possible has a high priority.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

V.7 Concluding Remarks

Two novel online distributed learning algorithms, the OADMM and Adaptive D-MIDAS, were proposed for high-dimensional AR-X model coefficient estimation to be used in wind power forecasting. The distributed component of both algorithms enables the estimation of AR-X models without the necessity of sharing sensitive data, such as power measurements, directly with other agents or entities. This enables competing wind farm operators to cooperate and collectively improve the forecasts for their sites. On the other hand, the online component allows the estimated model coefficients to follow the time-varying conditions of wind power generation time-series. Our main focus has been on distributed learning and forecasting for a class of linear models. Obviously then, the quality of the forecasts obtained is linked to the relevance of such linear models in practice. In view of the literature on short-term wind power forecasting, AR-X models are highly relevant for the lead times and forecasting setups considered in the paper. Some relevant generalisation could readily be considered e.g. to some types of regime-switching models (Self-Exciting Threshold Auto-Regressive – SETAR, and Smooth Transition Auto-Regressive – STAR). As long as the models involved are linear and separable, the methods discussed in the paper could be used in a similar manner. More broadly though, generalisation to nonlinear and more complex models may be more involved.

The OADMM relies on a LASSO-type objective function to estimate the coefficients of regularised AR-X models, in combination with an exponential forgetting factor to control the level of adaptivity. The algorithm minimises at any time t the cumulative loss over all observed samples (up to t), which requires the solving of a linear system of equations to update the algorithm parameters. Due to the non-negligible time for solving large equation systems, the Adaptive D-MIDAS is subsequently introduced. Owing to its design, all parameter updates are computationally cheaper to obtain. The algorithm is based on a mirror descent method where the gradient of the current squared forecast error is used to update dual variables. These are then mapped via a link function to the AR-X model coefficients. In addition, an accelerated version of the Adaptive D-MIDAS was proposed, i.e., a communication-reduced version. The algorithm achieves faster model coefficient re-estimates by using the dual variables from the previous time stamp. We verified that the impact on forecast accuracy is small.

A study on simulated data verified the ability of both algorithms to track time-varying model coefficients. However, the OADMM approach brings a better trade-off between adaptivity and limited variability of the estimated model coefficients, than the Adaptive D-MIDAS approach. Owing to its design, by minimising the cumulative loss over all past samples, large forecast errors do not directly cause large variations in the model coefficient estimates. The better controllability between adaptivity and the estimated model coefficient variance is the reason why the OADMM achieves a better forecast accuracy than the Adaptive D-MIDAS in the case study with a real-world dataset of 311 wind farms. The case study additionally confirmed that online learning is superior to offline learning, as already supported by previous work, although based on centralised learning algorithms.

Future works should address strategies to reduce the greater variability in the estimated model coefficients of the Adaptive D-MIDAS. Since we only considered deterministic forecasting models, future works should investigate extensions of the online distributed learning algorithms for the case of probabilistic forecasting. Then, besides other proposals for distributed online learning, and to relax the assumption such that agents are willing to collaborate, truthfully and rationally, it may be crucial to investigate federated learning and data markets. These new concepts may incentivise and support improvements in forecast quality when relevant data and features are distributed, both geographically and in terms of ownership.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

VI. Online forecast reconciliation in wind power prediction

VI.1 Introduction

Large-scale deployment of renewable energy generation sources brings a wealth of opportunities and challenges. For forecasting especially, the fact that production sites are geographically distributed, in a fairly dense manner, yields an observation network that can be exploited. This eventually allows improving the accuracy of wind power forecasts by accounting for spatio-temporal dependencies in the underlying processes, e.g. (Tastu et al., 2014). This effect was also observed for the case of solar power forecasts (Bessa et al., 2015a), hence making the methods proposed for wind power equally relevant for solar power generation. However, other challenges that were unforeseen (or possibly considered as futile) are being identified. In fact, since many agents in power systems and electricity markets generate their own forecasts, at various aggregation levels and independently of each other, these forecasts may end up not being coherent. For example, for a portfolio composed of two wind farms, the sum of the forecasts made for these wind farms, individually, will not necessarily be equal to the forecasts readily made for the portfolio. This lack of additive coherency is a challenge when forecasts are used as input to decision-making problems in power system operation and electricity markets.

The issue of forecast reconciliation has already been identified in the statistical modelling and forecasting literature for quite some time now, with the first work related to energy applications described in (van Erven and Cugliari, 2015). Since then, a wealth of relevant works appeared, including methodological contributions and applications, e.g. (Wickramasuriya et al., 2018). Some were readily focused on the wind power forecasting application, as for the case of (Zhang and Dong, 2018) for instance. In fact, reconciliation approaches for probabilistic forecasts were also proposed, for both electric load (Ben Taieb et al., 2021) and wind power generation (Jeon et al., 2019). Others have looked at novel approaches to temporal reconciliation for large-scale electricity consumption (Nystrup et al., 2020). Distributed approaches to forecast reconciliation (Bai and Pinson, 2019), based on the Alternating Direction Method of Multipliers (ADMM), allowed to prevent potentially sensitive information exchange between wind farm operators. However, most of these approaches make unrealistic unbiasedness assumptions and overlook the fact that the underlying stochastic processes and optimal reconciliation may be nonstationary.

As a result, our objective is to propose a new online forecast reconciliation approach which relaxes these assumptions and allows to adapt to changes in the underlying characteristics of the stochastic processes. Specifically, we make the following contributions. First, we formulate a new objective function for forecast reconciliation based on a multivariate regression problem with equality constraints on the regression parameters. This leads to a batch multivariate least squares estimator with equality constraints (MLSE). Then, we extend the MLSE estimator to the online setting, and derive a recursive and adaptive estimator inspired by recursive least squares (RLS) estimation with exponential forgetting, which we denote MRLSE. Finally, we prove that our estimators guarantee the coherency property not only in-sample but also out-of-sample. In other words, the out-of-sample forecasts will be coherent by design even though the objective function only constrains the in-sample forecasts to be coherent.

The remainder of the paper is structured as follows. The forecast reconciliation problem is described in Section VI.2. Our proposal for forecast reconciliation is described in Section VI.3, in both their batch and online versions. Section VI.4 presents some experiments with Danish wind data, while conclusions and perspectives for future work are gathered in Section VI.5.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

VI.2 Forecast Reconciliation

Let $\{Y^*_{s,t}\}$ ($s = 1,\dots,m$, $t = 1,\dots,T$) be the stochastic process for wind power generation, with indices s and t for location and time, respectively, as well as corresponding realizations $y^*_{s,t}$. We denote the power observations for all m individual sites at a given time t as $\mathbf{y}^*_t = [y^*_{1,t},\dots,\, y^*_{s,t},\, \dots, y^*_{m,t}]^\top$.

VI.2.1 Defining a Hierarchy

Individual sites are organized in a hierarchy, where quantities at upper levels are obtained by aggregating the quantities of the individual sites. The hierarchy has L levels and N total number of nodes. $\mathcal{S}$ is the set of all nodes. N_l is the number of nodes at level l, as a subset $\mathcal{S}_l \subset \mathcal{S}$, such that $N = \sum_{l=1}^{L} N_l$ and $\mathcal{S} = \bigcup_{l=1}^{L} \mathcal{S}_l$. The tuple (l, j) then uniquely identifies node j at level l. Nodes at a lower level of the hierarchy are referred to as child nodes, and those at the lowest level (the individual sites) are the bottom nodes. The number N_L of bottom nodes is equal to the number of individual sites m. An example of a 3-level hierarchy, based on 5 individual sites, is depicted in Fig. 48.

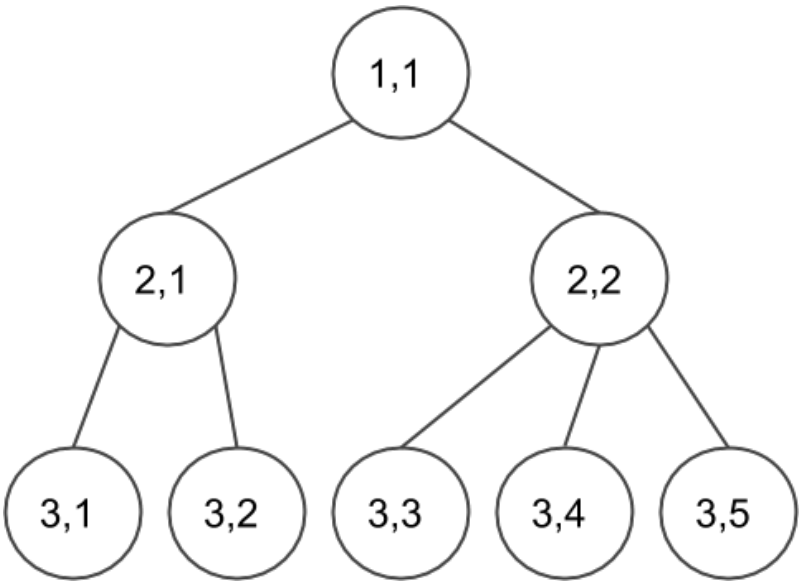


Figure 48 Example of a 3-level hierarchy based on 5 individual sites, with $S_1 = \{(1,1)\}$, $S_2 = \{(2,1),(2,2)\}$ and $S_3 = \{(3,1),(3,2),(3,3),(3,4),(3,5)\}$.

If $\mathbf{y}^*_t$ are the observations at time t in the bottom nodes, the observations at all levels of the hierarchy $\mathbf{y}_t$ are given by

$$\mathbf{y}_t = \mathbf{S}\,\mathbf{y}^*_t, \quad \forall t\,, \tag{152}$$

where $\mathbf{S} \in \{0,1\}^{N \times N_L}$ is a summing matrix defined as

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 \in \{0,1\}^{N_1 \times N_L} \\ \mathbf{S}_2 \in \{0,1\}^{N_2 \times N_L} \\ \vdots \\ \mathbf{S}_{L-1} \in \{0,1\}^{N_{L-1} \times N_L} \\ \mathbf{I}_{N_L} \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{I}_{N_L} \end{bmatrix}, \tag{153}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

and $\mathbf{S}_l \in \mathbb{R}^{N_l \times N_L}$ is a matrix whose elements $s_{l_{ij}}$ are 1 if the j^{th} node of the bottom-level is a child (or grand-child) of the i^{th} node of level l, 0 otherwise. $\mathbf{I}_{N_L}$ is an identity matrix of dimension N_L. Thus, it has a block structure with a first block $\mathbf{A} \in \{0,1\}^{(N-N_L)\times N_L}$ for the summing operations to go up in the hierarchy and a second block being an identity matrix of size N_L to copy the elements of the bottom nodes.

For the example of Fig. 48, the summing matrix reads

$$\mathbf{S} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ \hline & & \mathbf{I}_5 & & \end{bmatrix}. \tag{154}$$

In parallel, consider that given a lead time k, forecasts are issued at time t for time $t+k$. The forecasts for all individual sites are denoted by $\hat{y}^*_{s,t+k|t}$, with $\hat{\mathbf{y}}^*_{t+k|t} = [\hat{y}^*_{1,t+k|t}, \dots, \hat{y}_{s,t+k|t}, \dots, \hat{y}^*_{m,t+k|t}]^\top$. Forecasts are also issued for all nodes of the hierarchy, individually and independently of each other, and collated in the vector of forecasts $\hat{\mathbf{y}}_{t+k|t}$.

VI.2.2 Additive Coherency and Reconciliation

Many agents in power systems and electricity markets generate their own forecasts at various aggregation levels independently of each other. As a result, it is highly likely that one has

$$\hat{\mathbf{y}}_{t+k|t} \neq \mathbf{S}\,\hat{\mathbf{y}}^*_{t+k|t}, \quad \forall t, k\,, \tag{155}$$

meaning that the forecasts do not satisfy the hierarchical aggregation constraints, also called *additive coherency*.

Definition 1 *(additive coherency) The forecasts $\hat{\mathbf{y}}_{t+k|t}$ for a hierarchy defined by a summing matrix $\mathbf{S}$ are said to be additively coherent if*

$$\hat{\mathbf{y}}_{t+k|t} = \mathbf{S}\,\hat{\mathbf{y}}^*_{t+k|t} \iff \mathbf{H}^\top \hat{\mathbf{y}}_{t+k|t} = \mathbf{0}\,, \tag{156}$$

where

$$\mathbf{H}^\top = \begin{bmatrix} \mathbf{I}_{(N-N_L)} & -\mathbf{A} \end{bmatrix}. \tag{157}$$

Note that the matrix $\mathbf{H}$ naturally depends on the structure of the hierarchy through the matrix $\mathbf{A}$. As we need one equality constraint per non-bottom node, this yields $N - N_L$ equality constraints. The matrix $\mathbf{H}^\top$ therefore is a $(N - N_L) \times N$ matrix. For the specific case of the 3-level hierarchy depicted in Fig. 48, we have

$$\mathbf{H}^\top = \left[\begin{array}{ccc|ccccc} 1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\ 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & -1 & -1 \end{array}\right]. \tag{158}$$

Given some probably incoherent forecasts $\hat{\mathbf{y}}_{t+k|t}$, the process of forecast reconciliation is defined as the transformation of the forecast vector $\hat{\mathbf{y}}_{t+k|t}$ such that it is made additively coherent (i.e., the equality is restored). For a review of the alternative approaches to forecast reconciliation, the reader is referred to (Athanasopoulos et al., 2016).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Remark 1 *Contrarily to the case of forecasts, power measurements are naturally additively coherent, since measurements for upper level of the hierarchy are obtained by directly using the summing matrix* $\mathbf{S}$ *as in* (152).

VI.3 Forecast Reconciliation with Multivariate Least Squares Estimation

We propose a new forecast reconciliation method which involves solving a multivariate least squares regression problem. A set of constraints on the coefficients are added to the objective function to ensure coherent forecasts. By doing so, we relax the unbiasedness assumption of existing reconciliation methods (Wickramasuriya et al., 2018), and we allow to use the wealth of modern approaches for estimation in regression models including the online learning setting. We first introduce a batch version of our method, then we derive an online version based on recursive and adaptive estimation with exponential forgetting.

VI.3.1 Multivariate Least Squares Estimation

We model the observations at all nodes in the hierarchy as a linear combination of the corresponding forecasts. Specifically, given lead time k, we consider the following regression model:

$$\mathbf{y}_{t+k} = \mathbf{\Theta}_k^\top \tilde{\mathbf{y}}_{t+k|t} + \boldsymbol{\varepsilon}_{t+k}, \quad \forall t\,, \tag{159}$$

where $\mathbf{\Theta}_k \in \mathbb{R}^{(N+1)\times N}$ is a matrix of regression coefficients, $\tilde{\mathbf{y}}_{t+k|t}^\top = \begin{bmatrix} 1 \, \hat{\mathbf{y}}_{t+k|t}^\top \end{bmatrix} \in \mathbb{R}^{1\times(N+1)}$, and $\boldsymbol{\varepsilon}_{t+k}$ a noise term with zero mean and finite variance.

In the batch setting, we are given a dataset composed of T pairs of forecasts and observations, for a given lead time k. With our method, this dataset is used to estimate the regression coefficients in (159). More precisely, we solve the following multivariate least squares problem with equality constraints (MLSE):

$$\hat{\mathbf{\Theta}}_k^{\text{MLSE}} = \underset{\mathbf{\Theta}}{\arg\min} \quad ||\mathbf{Y}_k - \hat{\mathbf{Y}}_k\mathbf{\Theta}||_2^2 \tag{160a}$$

$$\text{s.t.} \quad \hat{\mathbf{Y}}_k\mathbf{\Theta}\mathbf{H} = \mathbf{0}\,, \tag{160b}$$

where $\mathbf{Y}_k \in [0,1]^{T\times N}$ and $\hat{\mathbf{Y}}_k \in [0,1]^{T\times(N+1)}$ are given by

$$\mathbf{Y}_k = \begin{bmatrix} \mathbf{y}_{1+k}^\top \\ \vdots \\ \mathbf{y}_{T+k}^\top \end{bmatrix} \text{ and } \hat{\mathbf{Y}}_k = \begin{bmatrix} \tilde{\mathbf{y}}_{1+k|1}^\top \\ \vdots \\ \tilde{\mathbf{y}}_{T+k|T}^\top \end{bmatrix}. \tag{161}$$

The constraint $\hat{\mathbf{Y}}_k\mathbf{\Theta}\mathbf{H} = \mathbf{0}$ ensures that the reconciled forecasts $\hat{\mathbf{Y}}_k\mathbf{\Theta}$ are coherent as presented in Definition 1. After estimating $\hat{\mathbf{\Theta}}_k^{\text{MLSE}}$, when a new forecast $\hat{\mathbf{y}}_{t+k|t}$ for all nodes of the hierarchy is available, the vector of reconciled forecasts is obtained as $\left(\hat{\mathbf{\Theta}}_k^{\text{MLSE}}\right)^\top \tilde{\mathbf{y}}_{t+k|t}$.

For the MLSE problem in (160), assuming $\hat{\mathbf{Y}}_k^\top\hat{\mathbf{Y}}_k$ is invertible, a closed-form solution can be readily obtained following the developments in (Kubáček, 2007), as

$$\hat{\mathbf{\Theta}}_k^{\text{MLSE}} = \left(\hat{\mathbf{Y}}_k^\top\hat{\mathbf{Y}}_k\right)^{-1}\hat{\mathbf{Y}}_k^\top\mathbf{Y}_k\left(\mathbf{I}_{N_L} - \mathbf{C}_k\right)\,, \tag{162}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

where $\mathbf{I}_{N_L}$ is an identity matrix of size N_L and $\mathbf{C}_k$ is a matrix whose elements depend on the structure of the hierarchy and on the variance of the forecast error, i.e.

$$\mathbf{C}_k = \mathbf{H}\left(\mathbf{H}^\top \boldsymbol{\Sigma}_k \mathbf{H}\right)^{-1} \mathbf{H}^\top \boldsymbol{\Sigma}_k \,. \tag{163}$$

The covariance matrix $\boldsymbol{\Sigma}_k$ needs to be estimated, possibly making some assumptions about its structure, as for some other reconciliation approaches (Athanasopoulos et al., 2016). Looking at (162), one observes that the MLSE estimator is a variant of the unconstrained multivariate Least Squares one, with a projection given by $(\mathbf{I}_{N_L} - \mathbf{C}_k)$,

$$\hat{\boldsymbol{\Theta}}_k^{\text{MLSE}} = \hat{\boldsymbol{\Theta}}_k^{\text{MLS}} \left(\mathbf{I}_{N_L} - \mathbf{C}_k\right) \,, \tag{164}$$

with

$$\hat{\boldsymbol{\Theta}}_k^{\text{MLS}} = (\hat{\mathbf{Y}}_k^\top \hat{\mathbf{Y}}_k)^{-1} \hat{\mathbf{Y}}_k^\top \mathbf{Y}_k. \tag{165}$$

Based on the equality constraints in (160b), coherency is imposed for all T pairs of forecasts and corresponding observations in the training dataset used to estimate the model parameters. This does not ensure that those parameters will guarantee coherency of forecasts reconciled for new data not seen in the training set (i.e., out-of-sample). The following Theorem shows that our method has the nice property of implicitly reconciling out-of-sample forecasts.

Theorem 1 (reconciliation by design) *By computing $\hat{\boldsymbol{\Theta}}_k^{MLSE}$ using (162), for any new forecast (out-of-sample) $\hat{\mathbf{y}}_{t+k|t}$, the reconciled forecasts given by $\left(\hat{\boldsymbol{\Theta}}_k^{MLSE}\right)^\top \tilde{\mathbf{y}}_{t+k|t}$ are additively coherent.*

Proof Consider any set of forecasts $\hat{\mathbf{y}}_{t+k|t}$ for a hierarchy defined by the summation matrix $\mathbf{S}$, and corresponding matrix $\mathbf{H}$. Based on the augmented vector of forecasts $\tilde{\mathbf{y}}_{t+k|t}$, one has

$$\tilde{\mathbf{y}}_{t+k|t}^\top \hat{\boldsymbol{\Theta}}_k^{\text{MLSE}} \mathbf{H} = \tilde{\mathbf{y}}_{t+k|t}^\top \hat{\boldsymbol{\Theta}}_k^{\text{MLS}} \left(\mathbf{I}_{N_L} - \mathbf{C}_k\right) \mathbf{H} \,. \tag{166}$$

It then means that

$$\tilde{\mathbf{y}}_{t+k|t}^\top \hat{\boldsymbol{\Theta}}_k^{\text{MLSE}} \mathbf{H} = \tilde{\mathbf{y}}_{t+k|t}^\top \hat{\boldsymbol{\Theta}}_k^{\text{MLS}} \left(\mathbf{H} - \mathbf{C}_k \mathbf{H}\right) \,. \tag{167}$$

Considering the definition of $\mathbf{C}_k$ in (163), one has

$$\begin{aligned} \mathbf{H} - \mathbf{C}_k \mathbf{H} &= \mathbf{H} - \mathbf{H}\left(\mathbf{H}^\top \boldsymbol{\Sigma}_k \mathbf{H}\right)^{-1} \left(\mathbf{H}^\top \boldsymbol{\Sigma}_k \mathbf{H}\right) && \text{(168a)} \\ &= \mathbf{H} - \mathbf{H}\,\mathbf{I}_{(N-N_L)} = 0 \,. && \text{(168b)} \end{aligned}$$

This therefore yields

$$\tilde{\mathbf{y}}_{t+k|t}^\top \hat{\boldsymbol{\Theta}}_k^{\text{MLSE}} \mathbf{H} = 0 \,, \tag{169}$$

for any forecast $\hat{\mathbf{y}}_{t+k|t}$ and whatever the chosen covariance matrix $\boldsymbol{\Sigma}$. □

VI.3.2 Online Version of the Estimator

For most practical applications, it is beneficial to consider online estimation, i.e., involving recursive estimation based on update equations and some form of history forgetting. This has the benefit of accommodating nonstationarity of the underlying stochastic processes, while lightening the computational burden. The online version of our estimator is therefore abbreviated as MRLSE (with 'R' for recursive).

At a given time t, the MRLSE estimator is defined as

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_{t,k}^{\text{MRLSE}} &= \arg\min_{\boldsymbol{\Theta}} \; S_t(\boldsymbol{\Theta}) && \text{(170a)} \\ &\qquad \text{s.t. } \tilde{\mathbf{y}}_{i+k|i}^\top \boldsymbol{\Theta} \mathbf{H} = \mathbf{0}, \;\; \forall i \leq t \,, && \text{(170b)} \end{aligned}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

where

$$S_t(\boldsymbol{\Theta}) = \frac{1}{2}\sum_{i\leq t} \lambda^{t-i}\left(\mathbf{y}_{t+k} - \boldsymbol{\Theta}^\top \tilde{\mathbf{y}}_{t+k|t}\right)^2 , \tag{171}$$

and where $0 < \lambda < 1$ is a forgetting factor, generally in the range $[0.95, 1]$. It is often more convenient to work with the equivalent number n_λ of observations instead, defined as $n_\lambda = (1-\lambda)^{-1}$.

In practice, as common for RLS estimators, the update equations for $\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t,k}$ given the previous value of the estimator, $\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t-1,k}$, and the new information available at time t, is obtained through a Newton-Raphson step. An additional projection $\pi_{\mathbf{H}}$ on the feasible space defined by (170b) ought to be used, similarly to (Pinson and Madsen, 2012). This yields

$$\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t,k} = \pi_{\mathbf{H}}\left\{\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t-1,k} - \frac{\nabla S_t(\boldsymbol{\Theta}_{t-1,k})}{\nabla^2 S_t(\boldsymbol{\Theta}_{t-1,k})}\right\} . \tag{172}$$

After a little algebra, one obtains the update equations at time t as

$$\mathbf{R}_{t,k} = \lambda \mathbf{R}_{t-1,k} + \tilde{\mathbf{y}}_{t+k|t}\tilde{\mathbf{y}}^\top_{t+k|t}, \tag{173a}$$

$$\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t,k} = \hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t-1,k} + \mathbf{R}_t^{-1}\tilde{\mathbf{y}}_{t+k|t}\left(\mathbf{y}_{t+k}(\mathbf{I}-\mathbf{C}) - \tilde{\mathbf{y}}^\top_{t+k|t}\hat{\boldsymbol{\Theta}}^{\mathrm{MRLSE}}_{t-1,k}\right) . \tag{173b}$$

The MRLSE estimator naturally inherits its fundamental reconciliation property from the MLSE estimator, i.e., reconciliation by design for any new (out-of-sample) forecasts.

VI.4 Application and Results

We compare our new forecast reconciliation method with the state-of-the-art approaches using a real-world dataset from Denmark. After introducing our case-study, we present our forecast verification framework and some relevant benchmarks. Finally, we provide a number of results and discuss the advantages and limitations of the different forecast reconciliation methods described previously.

VI.4.1 Case Study Based on a Danish Dataset

The dataset provided by the Danish Transmission System Operator, Energinet.dk, includes wind power measurements for 349 wind farms in western Denmark, for the period between January 2006 and March 2012. The measurements have a 15-minute temporal resolution. An extensive analysis of this dataset has been performed by (Girard and Allard, 2013; Lenzi et al., 2018; Messner and Pinson, 2019). These studies identified the conditional space-time dependencies of power generation at the various sites, including the nonstationarity of the underlying stochastic processes.

Only a subset of the available dataset, both in terms of number of wind farms and time period, was selected. Firstly, sites with non-negligible episodes with missing data were discarded. Out of the 250 sites left, only 100 sites were randomly selected, for simplicity. They are shown in Fig. 49.

Out of the complete dataset, a period with 70 080 time steps (2 years) was extracted for this analysis, from 2010 and 2011. The power measurement time-series for the 100 sites were then further cleaned, considering both erroneous and suspicious data points. For each site, observations exceeding 1.5 times the quantile with nominal level 0.99 of the distri- bution of observations

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

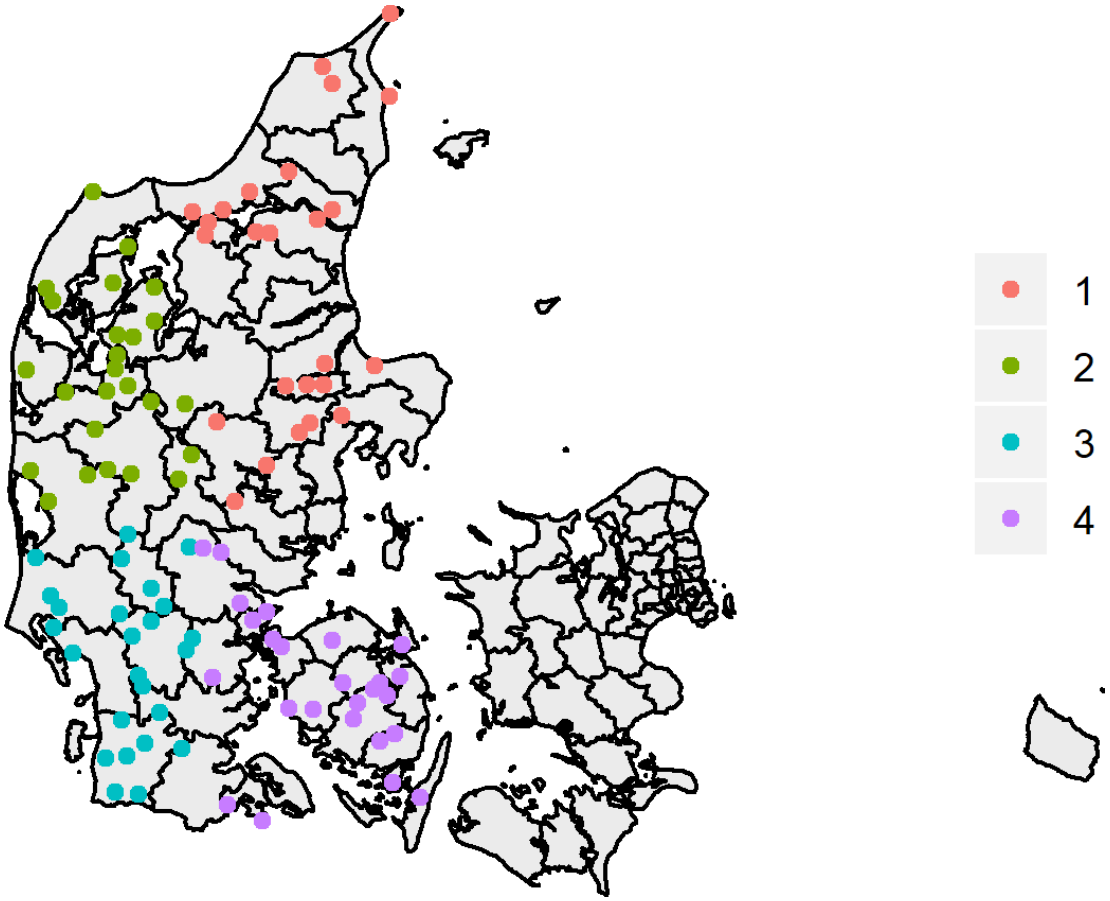


Figure 49 The 100 Danish sites selected from the complete Danish wind power dataset, then divided into 4 regions.

were removed. Power measurements were then normalized by the nominal capacity of that site. However, as this nominal capacity may change with time, a function computing rolling maxima was used for its estimation (package *zoo*[2]) and adaptive normalization. Rolling windows of 5 000 time steps were used. Consequently for the bottom nodes, all resulting observations take value in the unit interval (0,1).

The aggregate time-series for the various regions and the whole portfolio were obtained by using a summing matrix $\mathbf{S}$ (see Section VI.2.1). As in the example of Fig. 48, our hierarchy has 3 levels, with bottom nodes, regions, and the overall sum (referred to as total). The 100 wind farms were grouped in 4 regions, as shown in Figure 49. Each region is composed of 25 wind farms, by dividing the Western Denmark area into 4 quadrants. Owing to this summation, power values for the region level and the whole portfolio are within (0,25) and (0,100), respectively.

Forecasts are to be generated for each and every node of the hierarchy, i.e., for the 100 bottom nodes, the 4 regions and the overall portfolio (total). These are referred to as base forecasts. For simplicity, only 1-step ahead forecasts were considered, though the methodology could be readily used to reconcile forecasts for further lead times. Following the analysis and results in (Girard and Allard, 2013; Lenzi et al., 2018; Messner and Pinson, 2019), Auto-Regressive models with 2 lags - AR(2), were found sufficient to model the temporal dynamics of the time-series as input to forecasting. Thus, using the first 6 months of data as training dataset, AR(2) models were fitted through LS minimization for each node in the hierarchy. It was verified that those forecasts were competitive and their quality at the level of the state of the art for such short

[2]Available on CRAN at: `https://cran.r-project.org/web/packages/zoo`

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

lead times. These could be improved by considering more advanced models and possibly online learning, though only seen as different and possibly more accurate forecasts as input to forecast reconciliation. The following 6 months were then used as training for the batch reconciliation approaches. Specifically the online approach was initialized on the first time step of that period and then recursively updated trough the remainder of the dataset. For simplicity, the equivalent number of observations was set to $n_\lambda = 10,000$ though it could have been optimized through cross-validation. This eventually leaves the last year (2011) of data for genuine forecast verification.

VI.4.2 Forecast Verification Framework and Benchmarking

Our evaluation procedure is based on current practices for the verification of wind power forecasts, as recently described in (Messner et al., 2020). To be consistent with the least squares objective used to fit the models, i.e. the quadratic loss function, we use the Normalized Root Mean Square Error (NRMSE) as forecast verification criterion. For a set of T forecast-observation pairs for the node i of the hierarchy, the Scaled Root Mean Square Error (SRMSE) is given by

$$\mathrm{SRMSE}_i = \left(\frac{1}{T} \sum_{t=1}^{T} \left(\frac{\varepsilon_{i,t+1|t}}{s_i} \right)^2 \right)^{\frac{1}{2}} \tag{174}$$

$$\text{with } s_i = \begin{cases} 100, & \text{if } i = 1 \quad \text{(total)} \\ 25, & \text{if } i = 2, \dots, 5 \quad \text{(region level)} \\ 1, & \text{if } i = 6, \dots, 105 \quad \text{(bottom level)}, \end{cases} .$$

where $\varepsilon_{i,t+1|t}$ is the one-step ahead forecast error for the forecast issued at time t for the i^{th} node of the hierarchy. Score values are commonly multiplied by 100 as if expressed as percentages (of capacity). We additionally introduce a score that combines results for all nodes of the hierarchy, accounting for the number of nodes at each level. This Weighted Root Mean Square Error (WRMSE) is defined as

$$\mathrm{WRMSE} = \frac{1}{N_L\, L} \sum_{i=1}^{N} \mathrm{NRMSE}_i, \tag{175}$$

where NRMSE_i naturally reflects the importance of each node since relying on different scales (directly related to the number of bottom nodes it aggregates).

Since we aim to show how forecast reconciliation contributes to both restoring coherency and improving forecast accuracy, we report improvements with respect to the base forecasts. These improvements can be interpreted as percentage decrease in SRMSE compared to the base forecasts. For a given node i and reconciliation method, this writes

$$\mathrm{ISRMSE}_{i,\mathrm{method}} = \frac{\mathrm{SRMSE}_{i,\mathrm{base}} - \mathrm{SRMSE}_{i,\mathrm{method}}}{\mathrm{SRMSE}_{i,\mathrm{base}}}, \tag{176}$$

where $\mathrm{SRMSE}_{i,\mathrm{base}}$ and $\mathrm{SRMSE}_{i,\mathrm{method}}$ are the SRMSE values for the base forecasts and reconciliation method considered, respectively. A similar criterion can defined using the WRMSE criterion.

In the following, we will consider forecast reconciliation based on our two estimators, i.e., MLSE and MRLSE, as well as the state-of-the-art MinT approach. A complete description of the MinT approach to forecast reconciliation is available in (Athanasopoulos et al., 2016), while applications to wind power forecasting are described in (Zhang and Dong, 2018; Bai and Pinson, 2019). In this work, we consider the covariance matrix of the one-step-ahead forecast errors is estimated using the in-sample model residuals. More advanced shrinkage covariance estimators can also be used in high-dimensional setting. Finally, to measure the statistical significance of the differences in scores for the various reconciliation methods, we use the Diebold-Mariano (DM) test (see (Messner et al., 2020)). The differences are always found significant.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

VI.4.3 Results and Discussion

A. Observing the need for forecast reconciliation

To first illustrate the need for forecast reconciliation based on our case study, we look at the lack of coherence between forecasts at various levels of the hierarchy. Forecasts for the upper levels of the hierarchy (regions and total) are obtained based on the summing matrix $\mathbf{S}$ and then compared to the forecasts readily produced at these levels. These differences are therefore in the range of (-25,25) at the region level and (-100,100) at the total level. Results are depicted in Fig. 50 (in a fashion similar to the results in (Zhang and Dong, 2018)) and support the statement made with (155). These inconsistency errors are up to 4% here, at both region and total levels. Since the various approaches we consider hereafter allow for reconciliation by design, all those inconsistencies are then removed.

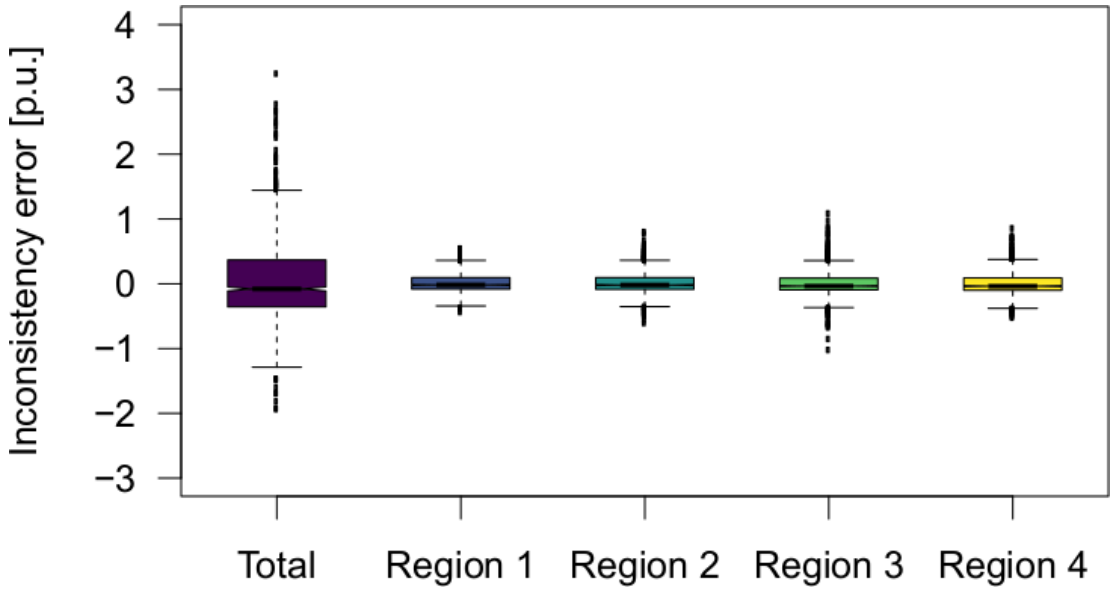


Figure 50 Incoherency, as expressed by (155)*, observed in the upper levels of the hierarchy over a randomly chosen period of 2 weeks.*

B. Impact on forecast quality

The literature on forecast reconciliation has regularly covered the fact that reconciliation eventually yields improvements in forecast quality. For instance already in (van Erven and Cugliari, 2015), the authors made a point that their game-theoretical optimal projection approach could reconcile forecasts by design while providing a geometry-inspired proof of forecast quality improvement (under a quadratic criterion). We consequently investigate here whether forecast improvements are obtained based on the approaches we proposed, and how it compares with the existing e.g. MinT.

We first look at the score values obtained over the one-year evaluation period covering 2011. These score values are collated in Table 17, using the SRMSE criterion expressed in percentage of nominal capacity (as an average for all nodes at a given level) and related improvements with the ISRMSE criterion. Scores values are lower as we go to more aggregate levels thanks to smoothing effects. All approaches yield forecast improvements, also at all levels of the hierarchy. The online forecast reconciliation approach based on MRLSE consistently gives the largest forecast improvements, those being larger as one gets towards lower levels of the hierarchy.

More than those average values, the distribution of improvements among bottom nodes and regions are of utmost importance. Results are qualitatively similar at these two levels of the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Table 17 Impact of forecast reconciliation on the quality of the forecasts, based on the SRMSE criterion (in % of nominal capacity) with related ISRMSE values (in %).

		bottom (av.)	regions (av.)	total
SRMSE	base	4.90	1.31	0.703
	MinT	4.81	1.28	0.699
	MLSE	4.65	1.26	0.690
	MRLSE	**4.53**	**1.22**	**0.676**
ISRMSE	base	–	–	–
	MinT	1.84	2.29	0.57
	MLSE	5.1	3.82	1.84
	MRLSE	**7.55**	**6.87**	**3.84**

hierarchy, hence we place emphasis on bottom nodes since relying on larger populations (100 nodes). Corresponding boxplots are depicted in Fig. 51. While forecast quality improvements are highest on average for our online forecast reconciliation approach based on the MRLSE estimator, there is also a high variability in those improvement. Those are always positive and up to more than 15% for a given site.

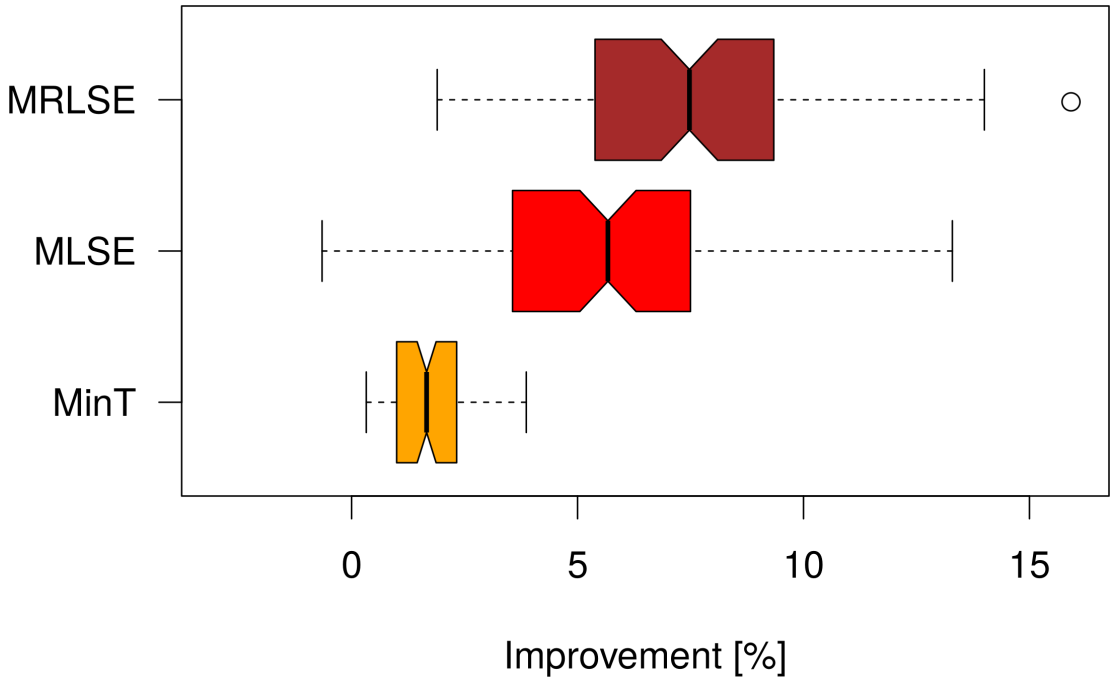


Figure 51 Distribution of improvements (ISRMSE) for bottom nodes and for the 3 forecast reconciliation approaches.

C. Time-varying aspect of forecast reconciliation

As a motivation for the proposal of an online forecast reconciliation approach, we mentioned the fact that the underlying stochastic processes are nonstationary. As a consequence, we expect that the parameters Θ evolve with time throughout the dataset. This is illustrated by Fig. 52 which show the temporal evolution of the coefficients associated to sites 25, 31, and 96 to

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

obtain the reconciled forecast values for the total level. Their evolution combine smoother and higher-frequency fluctuations. Remember that the forgetting factor used is very large (n_λ = 10 000) hence yielding an MRLSE estimator with fairly long memory.

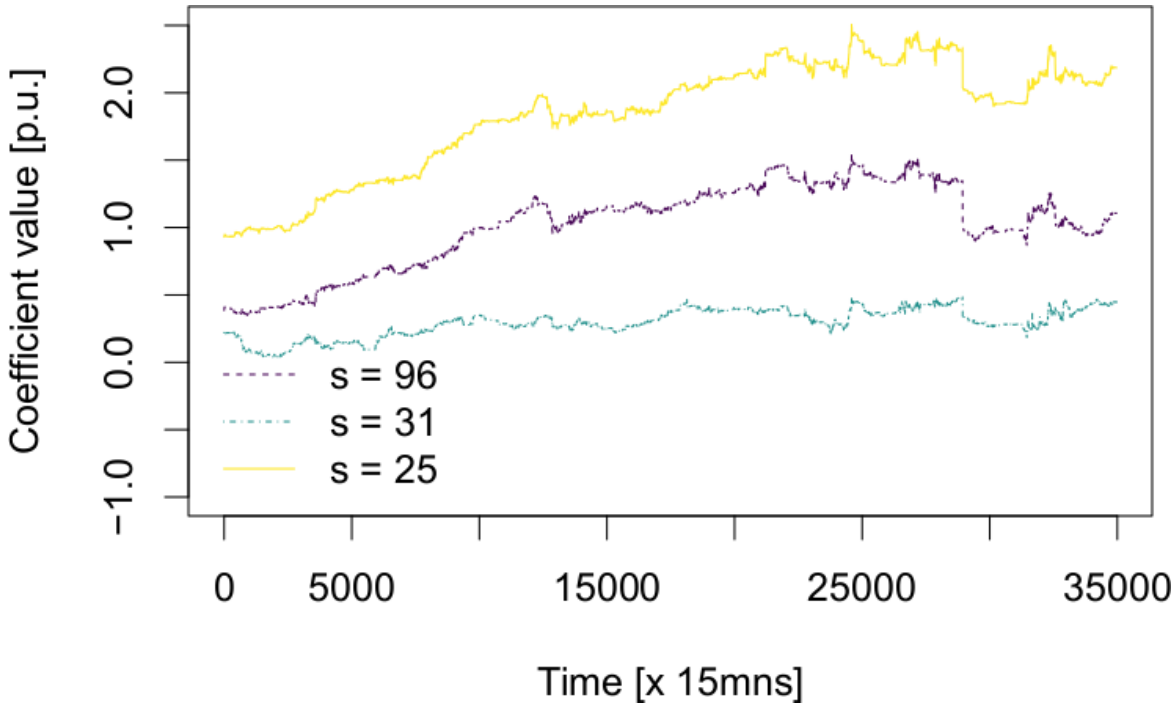


Figure 52 Evolution of randomly chosen coefficient (for sites 25,31 and 96) contributing to obtaining the reconciled forecasts at total level.

Subsequently we look at the impact of nonstationarity on the quality of the forecasts obtained after forecast reconciliation. Figure 53 gathers monthly IWRMSE values for the 12 months of the verification period, and for the 3 reconciliation approaches considered. The MRLSE estimator, which accom- modate nonstationarity, systematically performs better than the MLSE one, for which parameters are static throughout that year. There is also a trend that the improvement from MRLSE increases with time, which is consistent with the fact it is the only approach that aims to accommodate nonstationarity.

D. Consistency among potential hierarchies

A fairly specific hierarchy was considered. Indeed, Western Denmark is specifically split into 4 quadrants, i.e., contiguous areas with the same number of wind power production sites. However, it is of interest to see how the forecast reconciliation approaches would perform if we were to consider different types of hierarchies. For simplicity, we stick to a 3-level hierarchy and the idea of having the same number (25) of wind power generation sites in each of the mid-level nodes. Consequently we perform a Monte-Carlo simulation experiment, for which instead of considering geographical information, sites are randomly assigned to the 4 regions. Strictly speaking these are not regions anymore, but geographically dispersed port- folios instead. 100 replicates of this Monte-Carlo simulation experiment are used to obtain a distribution of scores values (SRMSE, WRMSE, and related improvements) at the bottom, region and total levels. The results for the ISRMSE criterion are depicted in the form of boxplots, for the various levels and reconciliation approaches, in Fig. 54.

At the region and total levels level, there is variability in the forecast improvements obtained, though the online forecast reconciliation through the MRSLE estimator consistently performs best. The variability is highest at the region level, since the structure of the hierarchy highly influences potential forecast improvement. Actually by comparing the results with Table 17, one observes

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

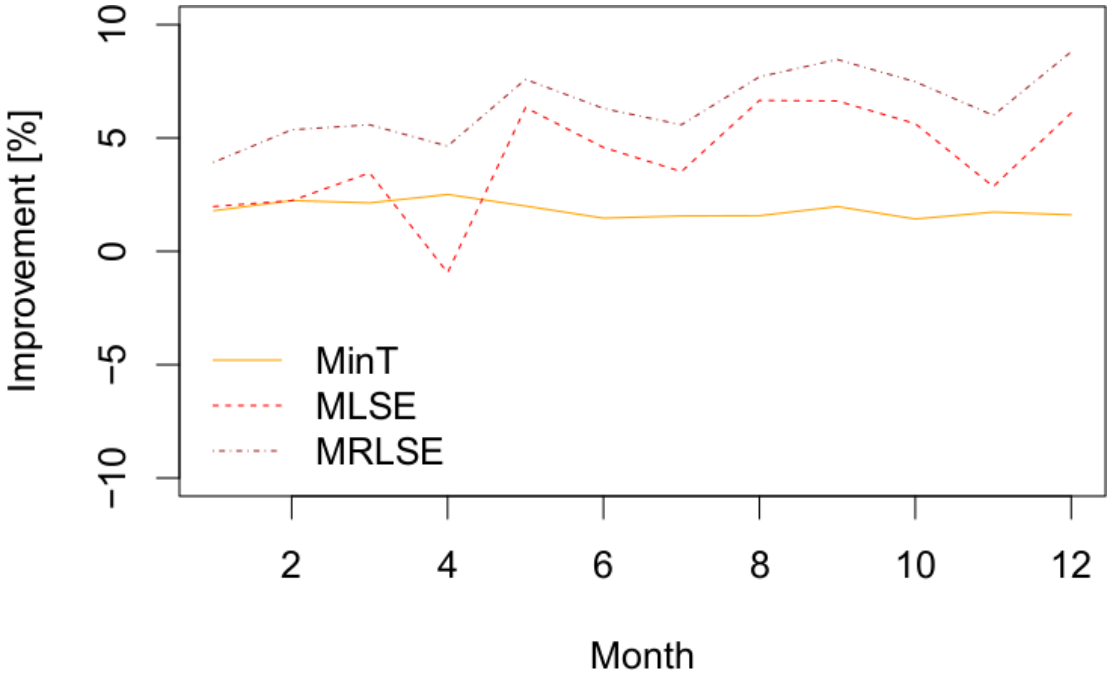


Figure 53 IWRMSE calculated on a monthly basis through the one-year verification period.

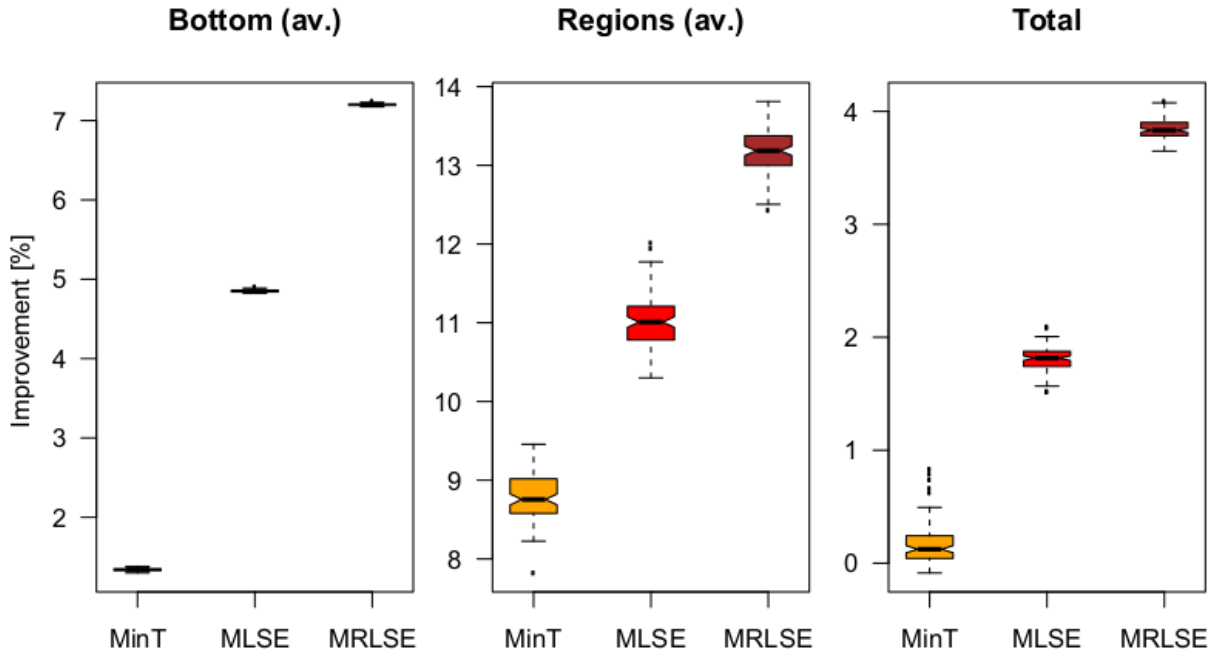


Figure 54 Boxplots for the distribution of ISRMSE values over a Monte-Carlo experiment with 100 replicates.

that the quadrant based hierarchy is the worst (with much lower ISRMSE values) as it is the worst hierarchy to pick, i.e., with the smallest possible smoothing effect. Such hierarchy randomization study could be extended to the case of having different number of sites per region.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

VI.5 Concluding Remarks

A data-driven approach to forecast reconciliation was introduced, in a multivariate regression framework. The main interest of that approach is that it eventually allows for online forecast reconciliation, hence allowing to adapt to nonstationarity in the underlying stochastic processes. A proof of reconciliation by design was also provided, making that, even trained on specific past data, our approach allows to reconcile any new forecasts out of sample.

The case study application concentrated on a fairly simple setup, with 1-step ahead and short-term forecasts only, as the main focus was the reconciliation process, which is independent from the lead time, rather than the forecasting one. The approach may be readily used for multi-step ahead forecasts and day-ahead forecasting, though we expect the results to be qualitatively equivalent.

In addition, the forecast reconciliation problem is seen as centralized, but it could be readily distributed using e.g. ADMM and the likes, since consisting of a convex optimization problem. Similarly, sparsification was not considered here, while it may be clearly of interest to minimize the number of alterations to forecasts in the reconciliation process. This may be considered in the future, the same way MinT has been generalized by allowing for shrinkage. However, this will bring some complexity in the derivation of the online estimator due to the L_1-regularization which is not continuously differentiable. Finally, other types of models may be thought of in a multivariate regression framework. Although we are restricting our model to the linear setting, as done in all reconciliation literature, one could generalize it to the non linear setting, e.g. by using Support Vector Regression, Gradient Boosting or Random Forests. While clearly reconciliation properties would need to be verified in those cases, the non-linear setting would make it possible to account for conditional effects (e.g. from weather conditions and prevailing wind direction) as well as regime-switching, either explicitly or by the use of an adaptive forgetting factor scheme.

VII. Conclusions

This section summarizes the main contributions and findings from Task 4.1. The topics for future work are also identified.

VII.1 Summary

Despite the many benefits of 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 Task 4.1:

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.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 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 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. **Online learning and reconciliation.** Due to the high variability of RES generation, forecasting RES generation close to real-time is of utmost importance for the efficient operation of power systems and electricity markets. Using distributed learning approaches described above that help preserve the privacy of RES agents, two novel approaches are proposed to recursively update model parameters while limiting information exchange between RES agents and other potential data providers. Specifically, the OADMM and Adaptive D-MIDAS, were proposed for high-dimensional AR-X model coefficient estimation, closing the gap between online and distributed optimisation in RES forecasting.

   The ability of both algorithms to track time-varying model coefficients is verified in a study on simulated data. Then, a case study with a real-world dataset of 311 wind farms compares the two algorithms and demonstrates a better forecast accuracy of the OADMM than the Adaptive D-MIDAS. This is largely due to the OADMM's better controllability between adaptivity and the estimated model coefficient variance. The case study additionally confirms that online learning is superior to offline learning, as already supported by previous work, although based on centralised learning algorithms.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Additionally, a data-driven approach for online forecast reconciliation is formulated in a multivariate regression framework, which ensures the coherency of forecasts among various agents at various aggregation levels. The main interest of that approach is that it eventually allows for online forecast reconciliation, hence allowing to adapt to nonstationarity in the underlying stochastic processes. It relies on a multivariate least squares estimator, with equality constraints on the coefficients. A recursive and adaptive version of that estimator is derived, hence allowing to track the optimal reconciliation in a fully data-driven manner. A proof of reconciliation by design is also provided, making that, even trained on specific past data, the proposed approach allows to reconcile any new forecasts out of sample.

All in all, all sections have an associated publication, in journals ranging in impact factors from 3.414 up to 7.917, as described in what follows.

VII.2 Dissemination

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

Extreme Conditional Quantiles Forecasting

Section II. 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=3.414, Q1)

Privacy-preserving Forecasting Model

Section III. 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=3.779, Q1)

Section IV. C. Gonçalves, R.J. Bessa, and P. Pinson, "Privacy-preserving distributed learning for renewable energy forecasting," *IEEE Transactions on Sustainable Energy*, vol.12, no. 3, pp. 1777-1787, 2021, doi: 10.1109/TSTE.2021.3065117. (IF=7.917, Q1)

Online distributed learning and reconciliation in RES forecasting

Section VI. C. di Modica, P. Pinson and S. Ben Taieb, "Online forecast reconciliation in wind power prediction," *Electric Power Systems Research*, vol. 190, pp. 106637, Jan. 2021, doi: 10.1016/j.epsr.2020.106637. (IF=3.779, Q1)

The work developed in Task 4.1 was also disseminated by conferences:

- The **Section II** 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", *PSCC 2020*

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

  C. di Modica, P. Pinson and S. Ben Taieb, "Online forecast reconciliation in wind power prediction", *PSCC 2020*.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

VII.3 Future Work

The following topics were identified for future work:

1. **Extreme quantile forecasting.** Forecasting rare events remains a challenge given 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 Lerch et al. (2017) 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 (Dowell and Pinson, 2015). The proposed privacy-preserving protocol can be applied to non-linear regression by extending the additive model structure to a multivariate setting (de Souza et al., 2018) or by local linear smoothing (Jiang, 2014).
   (b) The extension to other non-linear multivariate models recently considered in collaborative learning Li et al. (2020), 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. **Online Forecasting and Reconciliation Models.** The proposed online forecasting and reconciliation models assume deterministic linear regression models and the agents' willingness to participate. Future research should consider:
   (a) extensions of the online distributed learning algorithms for the case of probabilistic forecasting.
   (b) the relaxation of the assumption such that agents are willing to collaborate, truthfully and rationally, it may be crucial to investigate federated learning and data markets.
   (c) nonlinear models, accounting for conditional effects (e.g. from weather conditions and prevailing wind direction) as well as regime-switching, either explicitly or by the use of an adaptive forgetting factor scheme.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Appendices

A. Differential Privacy

Mathematically, a randomized mechanism $\mathcal{A}$ satisfies (ε,δ)-differential privacy Dwork and Smith (2010) 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),

$$\Pr(\mathcal{A}(\mathbf{D}) = t) \leq \delta + \exp(\varepsilon)\Pr(\mathcal{A}(\mathbf{D}') = t). \tag{177}$$

In practice, differential privacy can be achieved by adding random noise W to some desirable function f of the data $\mathbf{D}$. That is,

$$\mathcal{A}(\mathbf{D}) = f(\mathbf{D}) + W. \tag{178}$$

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 (2010) showed that the data can be masked by considering

$$\mathcal{A}(\mathbf{D}) = \mathbf{D} + \mathbf{W}. \tag{179}$$

B. 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* (95)–(96)*, 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*

$$\sqrt{Ts - u} < r < T. \tag{180}$$

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 18 – 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* (95)–(96) *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* (93)*. Then, to ensure privacy while computing $\mathbf{M}\mathbf{X}_{A_i}$ and $\mathbf{X}_{A_i}^\top\mathbf{M}^{-1}$, the protocol requires*

$$\sqrt{Ts - u} < r < T/2 \wedge r > s. \tag{181}$$

In addition, to compute $\mathbf{M}\mathbf{G}_{A_i}$, the protocol should take

$$\sqrt{Tg - v} < r' < T - 2r \wedge r' > g. \tag{182}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

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 $\mathbf{X}_{A_i}$, $\mathbf{C}$, $\mathbf{D}$). 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 $r' < T - 2r$. □

C. 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.

C.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 $\mathbf{M}[\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i} \in \mathbb{R}^{T\times r}$, $[[\mathbf{X}_{A_i}, \mathbf{C}_{A_i}]\mathbf{D}_{A_i}]^\top\mathbf{M}^{-1} \in \mathbb{R}^{T\times r}$ 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 $\mathbf{M}\mathbf{X} \in \mathbb{R}^{T\times ns}$ and $\mathbf{M}\mathbf{G} \in \mathbb{R}^{T\times ng}$, as shown in Section III. 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 (181) and (182), the previous inequality has no solution in $\mathbb{R}_0^+$.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

C.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 $c \geq \lceil \frac{T}{2r+r'+s+g} \rceil$.

D. Online Reconciliation: additional corollary

The proposal in Section VI is based on the equality constraint in (160b), leading to Theorem 1 that ensures reconciliation by design (for the MLSE estimator, as well as its online version MRLSE). Actually, one can get an even more general version of that result, which does not require the equality constraint, as long as the measurements are themselves additively coherent. This leads to the following corollary to Theorem 1 (which is also valid for the online version MRLS of the MLS estimator). A proof is also given.

Corollary 1 (reconciliation by design of the MLS estimator) *By computing $\hat{\mathbf{\Theta}}_k^{MLS}$ using* (165) *and given that $\mathbf{Y}_k$ are additively coherent, for any new forecast (out-of-sample) $\hat{\mathbf{y}}_{t+k|t}$, the reconciled forecasts given by $\left(\hat{\mathbf{\Theta}}_k^{MLS}\right)^\top \tilde{\mathbf{y}}_{t+k|t}$ are additively coherent.*

Proof Given the training dataset of measurements and base forecasts, respectively

$$\mathbf{Y}_k = \begin{bmatrix} \mathbf{y}_{1+k}^\top \\ \vdots \\ \mathbf{y}_{T+k}^\top \end{bmatrix} \text{ and } \hat{\mathbf{Y}}_k = \begin{bmatrix} \tilde{\mathbf{y}}_{1+k|1}^\top \\ \vdots \\ \tilde{\mathbf{y}}_{T+k|T}^\top \end{bmatrix}, \tag{183}$$

the MLS estimator is obtained by

$$\hat{\mathbf{\Theta}}_k^{\text{MLS}} = \left(\hat{\mathbf{Y}}_k^\top \hat{\mathbf{Y}}_k\right)^{-1} \hat{\mathbf{Y}}_k^\top \mathbf{Y}_k = \mathbf{\Omega}_k \mathbf{Y}_k\,, \tag{184}$$

where

$$\mathbf{\Omega}_k = \left(\hat{\mathbf{Y}}_k^\top \hat{\mathbf{Y}}_k\right)^{-1} \hat{\mathbf{Y}}_k^\top \in \mathbb{R}^{(N+1)\times T}. \tag{185}$$

Breaking down matrices $\mathbf{\Omega}_k$ and $\mathbf{Y}_k$ element-wise, and dropping index k from the element indexing to avoid clutter.

$$\hat{\mathbf{\Theta}}_k^{\text{MLS}} = \begin{bmatrix} \Omega_{1,1} & \dots & \Omega_{1,T} \\ \vdots & & \vdots \\ \Omega_{N,1} & \dots & \Omega_{T,N} \end{bmatrix} \begin{bmatrix} y_{1,1} & \dots & y_{1,N} \\ \vdots & & \vdots \\ y_{T,1} & \dots & y_{T,N} \end{bmatrix} = \tag{186}$$

$$\begin{bmatrix} \sum_{j=1}^T \Omega_{1,j} y_{j,1} & \dots & \sum_{j=1}^T \Omega_{1,j} y_{j,N} \\ \vdots & & \vdots \\ \sum_{j=1}^T \Omega_{N+1,j} y_{j,1} & \dots & \sum_{j=1}^T \Omega_{N+1,j} y_{j,N} \end{bmatrix}. \tag{187}$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

For any new forecast (out-of-sample) $\hat{\mathbf{y}}_{t+k|t}$, the product $\left(\hat{\mathbf{\Theta}}_k^{\text{MLS}}\right)^\top \tilde{\mathbf{y}}_{t+k|t}$ yields

$$\begin{bmatrix} \sum_{j=1}^{T} \Omega_{1,j} y_{j,1} & \dots & \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,1} \\ \vdots & & \vdots \\ \sum_{j=1}^{T} \Omega_{1,j} y_{j,N} & \dots & \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,N} \end{bmatrix} \begin{bmatrix} 1 \\ \hat{y}_{t,1} \\ \vdots \\ \hat{y}_{t,N} \end{bmatrix} = \tag{188}$$

$$= \begin{bmatrix} \sum_{j=1}^{T} \Omega_{1,j} y_{j,1} + \dots + \hat{y}_{t,N} \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,1} \\ \sum_{j=1}^{T} \Omega_{1,j} y_{j,2} + \dots + \hat{y}_{t,N} \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,2} \\ \vdots \\ \sum_{j=1}^{T} \Omega_{1,j} y_{j,N} + \dots + \hat{y}_{t,N} \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,N} \end{bmatrix}. \tag{189}$$

The first element is equal to the sum of the others (i.e. the forecasts are reconciled) if and only if

$$\begin{cases} \sum_{j=1}^{T} \Omega_{1,j} y_{j,1} = \sum_{j=1}^{T} \Omega_{1,j} y_{j,2} + \dots \\ \qquad\qquad \dots + \sum_{j=1}^{T} \Omega_{1,j} y_{j,N} \\ \qquad\qquad \vdots \\ \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,1} = \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,2} + \dots \\ \qquad\qquad \dots + \sum_{j=1}^{T} \Omega_{N+1,j} y_{j,N}\,, \end{cases} \tag{190}$$

or equivalently,

$$y_{j,1} = y_{j,2} + \dots + y_{j,N}\,, \qquad \forall j = 1, \dots, T, \tag{191}$$

i.e. when the measurements are consistent. □

It is worth noting that although this is the case in theory and in some practical applications, there might be cases where this consistency is not achieved in practice. Measuring errors, missing data and subsequent imputation techniques could for example affect data coherency and justify reconciliation through $\hat{\mathbf{\Theta}}_k^{\text{MLSE}}$.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

References

Agarwal, A., Dahleh, M., and Sarkar, T. A marketplace for data: An algorithmic solution. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 701–726, 2019.

Agoua, X. G., Girard, R., and Kariniotakis, G. Probabilistic models for spatio-temporal photovoltaic power forecasting. *IEEE Transactions on Sustainable Energy*, 10(2):780–789, 2018.

Ahmad, H. W., Zilles, S., Hamilton, H. J., and Dosselmann, R. Prediction of retail prices of products using local competitors. *International Journal of Business Intelligence and Data Mining*, 11(1): 19–30, 2016.

Ahmadi, H., Pham, N., Ganti, R., Abdelzaher, T., Nath, S., and Han, J. Privacy-aware regression modeling of participatory sensing data. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 99–112. ACM, 2010.

Andersen, P. D. Optimal trading strategies for a wind-storage power system under market conditions. Master's thesis, Technical University of Denmark, Lyngby, Denmark, 2009.

Andrade, J. R. and Bessa, R. J. Improving renewable energy forecasting with a grid of numerical weather predictions. *IEEE Transactions on Sustainable Energy*, 8(4):1571–1580, 2017.

Ansley, C. F. and Kohn, R. A note on reparameterizing a vector autoregressive moving average model to enforce stationarity. *Journal of Statistical Computation and Simulation*, 24(2):99–106, 1986.

Aono, Y., Hayashi, T., Phong, L. T., and Wang, L. Input and output privacy-preserving linear regression. *IEICE TRANSACTIONS on Information and Systems*, 100(10):2339–2347, 2017.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N., and Petropoulos, F. Forecasting with temporal hierarchies. *European Journal of Operational Research*, 262(1):60–74, 2016.

Aviv, Y. A time-series framework for supply-chain inventory management. *Operational Research*, 51(2):175–342, March 2003.

Aviv, Y. On the benefits of collaborative forecasting partnerships between retailers and manufacturers. *Management Science*, 53(5):777–794, May 2007.

Bacher, P., Madsen, H., and Nielsen, H. A. Online short-term solar power forecasting. *Solar Energy*, 83(10):1772–1783, October 2009.

Bai, L. and Pinson, P. Temporal hierarchies with autocorrelation for load forecasting. *Energies*, 12 (6):1112, 2019.

Baltagi, B., Fingleton, B., and Pirotte, A. Estimating and forecasting with a dynamic spatial panel data model. *Oxford Bulletin of Economics and Statistics*, 76(1):112–138, 2014. doi: 10.1111/obes.12011.

Beck, A. and Teboulle, M. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31:167–175, 2003.

Beirlant, J., Wet, T. D., and Goegebeur, Y. Nonparametric estimation of extreme conditional quantiles. *Journal of statistical computation and simulation*, 74(8):567–580, 2004.

Beirlant, J., Alves, I. F., Reynkens, T., et al. Fitting tails affected by truncation. *Electronic Journal of Statistics*, 11(1):2026–2065, 2017.

Bekierman, J. and Manner, H. Forecasting realized variance measures using time-varying coefficient models. *International Journal of Forecasting*, 34(2):276–287, 2018. doi: 10.1016/j.ijforecast.2017.12.005.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Ben Taieb, S., Bontempi, G., Atiya, A., and Sorjamaa, A. A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition. *Expert Systems with Applications*, 39(8):7067–7083, 2012. doi: 10.1016/j.eswa.2012.01.039.

Ben Taieb, S., Taylor, J. W., and Hyndman, R. J. Hierarchical probabilistic forecasting of electricity demand with smart meter data. *Journal of the American Statistical Society*, 116(533):27–43, 2021.

Berdugo, V., Chaussin, C., Dubus, L., Hebrail, G., and Leboucher, V. Analog method for collaborative very-short-term forecasting of power generation from photovoltaic systems. In *Proceedings Next Generation Data Mining Summit*, pages 1–5, Greece, September 2011.

Bessa, R. J., Trindade, A., Silva, C. S. P., and Miranda, V. Probabilistic solar power forecasting in smart grids using distributed information. *International Journal of Electrical Power & Energy Systems*, 72:16–23, 2015a.

Bessa, R. J., Miranda, V., Botterud, A., Zhou, Z., and Wang, J. Time-adaptive quantile-copula for wind power probabilistic forecasting. *Renewable Energy*, 40(1):29–39, 2012a.

Bessa, R. J., Trindade, A., and Miranda, V. Spatial-temporal solar power forecasting for smart grids. *IEEE Transactions on Industrial Informatics*, 11(1):232–241, 2015b.

Bessa, R. J., Trindade, A., Silva, C. S., and Miranda, V. Probabilistic solar power forecasting in smart grids using distributed information. *International Journal of Electrical Power & Energy Systems*, 72:16–23, 2015c.

Bessa, R. J., Möhrlen, C., Fundel, V., Siefert, M., Browell, J., Gaidi, S. H. E., Hodge, B.-M., Cali, U., and Kariniotakis, G. Towards improved understanding of the applicability of uncertainty forecasts in the electric power industry. *Energies*, 10(9):1402, September 2017.

Bessa, R. J., Rua, D., Abreu, C., Machado, P., Andrade, J. R., Pinto, R., Gonçalves, C., and Reis, M. Data economy for prosumers in a smart grid ecosystem. In *Proceedings of the Ninth International Conference on Future Energy Systems*, pages 622–630. ACM, 2018.

Bessa, R., Miranda, V., Botterud, A., Wang, J., and Constantinescu, E. M. Time adaptive conditional kernel density estimation for wind power forecasting. *IEEE Transactions on Sustainable Energy*, 3(4):660–669, 2012b.

Botterud, A., Wang, J., Zhou, Z., Bessa, R., Keko, H., Akilimali, J., and Miranda, V. Wind power trading under uncertainty in LMP markets. *IEEE Transactions on Power Systems*, 27(2):894–903, 2012.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. Distributed optimization and statistical learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine learning*, 3(1):1–122, 2011.

Bremnes, J. B. Probabilistic wind power forecasts using local quantile regression. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, 7(1):47–54, 2004.

Cagnolari, M. *The Value of the Right Distribution for the Newsvendor Problem and a bike-sharing problem*. PhD thesis, University of Bergamo, May 2017.

Cavalcante, L., Bessa, R. J., Reis, M., and Dowell, J. LASSO vector autoregression structures for very short-term wind power forecasting. *Wind Energy*, 20(4):657–675, April 2017a.

Cavalcante, L. and Bessa, R. J. Solar power forecasting with sparse vector autoregression structures. In *2017 IEEE Manchester PowerTech*, pages 1–6. IEEE, June 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Cavalcante, L., Bessa, R. J., Reis, M., and Browell, J. LASSO vector autoregression structures for very short-term wind power forecasting. *Wind Energy*, 20(4):657–675, April 2017b.

Chen, K. and Liu, L. A survey of multiplicative perturbation for privacy-preserving data mining. In *Privacy-Preserving Data Mining*, pages 157–181. Springer, 2008.

Chen, S., Xue, D., Chuai, G., Yang, Q., and Liu, Q. FL-QSAR: a federated learning-based QSAR prototype for collaborative drug discovery. *Bioinformatics*, 36(22–23):5492–5498, 2020.

Chen, Y.-R., Rezapour, A., and Tzeng, W.-G. Privacy-preserving ridge regression on distributed data. *Information Sciences*, 451:34–49, 2018.

Chernozhukov, V., Fernández-Val, I., and Galichon, A. Quantile and probability curves without crossing. *Econometrica*, 78(3):1093–1125, 2010.

Dai, W., Wang, S., Xiong, H., and Jiang, X. Privacy preserving federated big data analysis. In *Guide to Big Data Applications*, pages 49–82. Springer, 2018.

De Haan, L. and Ferreira, A. *Extreme value theory: an introduction*. Springer Science & Business Media, 2007.

de Souza, J. B., Reisen, V. A., Franco, G. C., Ispany, M., Bondon, P., and Santos, J. M. 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,*, 67(2):453–480, February 2018.

Diebold, F. X. and Mariano, R. S. Comparing predictive accuracy. *Journal of Business & economic statistics*, 20(1):134–144, 2002.

Dowell, J. and Pinson, P. Very-short-term probabilistic wind power forecasts by sparse vector autoregression. *IEEE Transactions on Smart Grid*, 7(2):63–770, March 2016.

Dowell, J. and Pinson, P. Very-short-term probabilistic wind power forecasts by sparse vector autoregression. *IEEE Transactions on Smart Grid*, 7(2):763–770, 2015.

Du, W., Han, Y. S., and Chen, S. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *2004 SIAM International Conference on Data Mining*, pages 222–233, 2004a.

Du, W., Han, Y. S., and Chen, S. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *SIAM International Conference on Data Mining (SDM)*, pages 222–233. SIAM, 2004b.

Duchi, J., Shalev-Shwartz, S., Singer, Y., and Chandra, T. Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 272–279, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390191.

Dupin, R. *Prévision du Dynamic Line Rating et impact sur la gestion du systéme électrique*. PhD thesis, MINES ParisTech, PSL Research University, Paris, France, July 2018.

Dwork, C. and Smith, A. Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1(2), 2010.

Dwork, C., Talwar, K., Talwar, K., Thakurta, A., and Zhang, L. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Forty-sixth Annual ACM Symposium on Theory of Computing*, pages 11–20, May 2014a.

Dwork, C., Talwar, K., Thakurta, A., and Zhang, L. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 11–20. ACM, 2014b.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Elsinga, B. and van Sark, W. G. Short-term peer-to-peer solar forecasting in a network of photovoltaic systems. *Applied Energy*, 206:1464–1483, November 2017.

Fabio Sigrist, F., Künsch, H. R., and Stahel, W. A. A dynamic nonstationary spatio-temporal model for short term prediction of precipitation. *Annals of Applied Statistics*, 6(4):1452–1477, 2012.

Fan, L. and Xiong, L. An adaptive approach to real-time aggregate monitoring with differential privacy. *IEEE Transactions on knowledge and data engineering*, 26(9):2094–2106, 2014.

Fienberg, S. E., Nardi, Y., and Slavković, A. B. Valid statistical analysis for logistic regression with multiple sources. In *Protecting persons while protecting the people*, pages 82–94. Springer, 2009.

Friederichs, P. and Thorarinsdottir, T. L. Forecast verification for extreme value distributions with an application to probabilistic peak wind prediction. *Environmetrics*, 23(7):579–594, 2012.

Gascón, A., Schoppmann, P., Balle, B., Raykova, M., Doerner, J., Zahur, S., and Evans, D. Privacy-preserving distributed linear regression on high-dimensional data. *Proceedings on Privacy Enhancing Technologies*, 2017(4):345–364, 2017.

Giebel, G. and Kariniotakis, G. *Wind power forecasting-a review of the state of the art*, pages 59–109. Renewable Energy Forecasting: From Models to Applications. Woodhead Publishing, 2017. doi: 10.1016/B978-0-08-100504-0.00003-2.

Gilbert, C., Browell, J., and McMillan, D. Leveraging turbine-level data for improved probabilistic wind power forecasting. *IEEE Transactions on Sustainable Energy*, 11(3):1152–1160, 2020a.

Gilbert, C., Browell, J., and McMillan, D. Leveraging turbine-level data for improved probabilistic wind power forecasting. *IEEE Transactions on Sustainable Energy*, 11(3):1152–1160, July 2020b.

Girard, R. and Allard, D. Spatio-temporal propagation of wind power prediction errors. *Wind Energy*, 16(7):999–1012, 2013.

Gonçalves, C. and Bessa, R. J. Geographically distributed solar power time series, September 2020. URL `https://doi.org/10.25747/gywm-9457`.

Hall, R., Fienberg, S. E., and Nardi, Y. Secure multiple linear regression based on homomorphic encryption. *Journal of Official Statistics*, 27(4):669, 2011.

Han, S., Ng, W. K., Wan, L., and Lee, V. C. Privacy-preserving gradient-descent methods. *IEEE Transactions on Knowledge and Data Engineering*, 22(6):884–899, 2010.

Hastie, T. J. and Tibshirani, R. J. *Generalized additive models*. Routledge, 2017.

He, M., Yang, L., Zhang, J., and Vittal, V. A spatio-temporal analysis approach for short-term forecast of wind farm generation. *IEEE Transactions on Power Systems*, 29(4):1611–1622, July 2014.

Hong, T., Pinson, P., Fan, S., Zareipour, H., Troccoli, A., and Hyndman, R. J. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond. *International Journal of Forecasting*, 32:896–913, 2016.

Hoogh, S. *Design of large scale applications of secure multiparty computation: secure linear programming*. PhD thesis, Technische Universiteit Eindhoven, 2012.

Huang, Z., Hu, R., Guo, Y., Chan-Tin, E., and Gong, Y. DP-ADMM: ADMM-based distributed learning with differential privacy. *IEEE Transactions on Information Forensics and Security*, 15:1002–1012, 2019.

Jain, Y. K. and Bhandare, S. K. Min max normalization based data perturbation method for privacy protection. *International Journal of Computer & Communication Technology*, 2(8):45–50, 2011.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Jeon, J., Panagiotelis, A., and F., P. Probabilistic forecast reconciliation with applications to wind power and electric load. *European Journal of Operational Research*, 279(2):364–379, 2019.

Jia, Q., Guo, L., Jin, Z., and Fang, Y. Preserving model privacy for machine learning in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, 29(8):1808–1822, 2018.

Jia, W., Zhu, H., Cao, Z., Dong, X., and Xiao, C. Human-factor-aware privacy-preserving aggregation in smart grid. *IEEE Systems Journal*, 8(2):598–607, June 2014.

Jiang, J. Multivariate functional-coefficient regression models for nonlinear vector time series data. *Biometrika*, 101(3):689–702, September 2014.

Juban, R., Ohlsson, H., Maasoumy, M., Poirier, L., and Kolter, J. Z. A multiple quantile regression approach to the wind, solar, and price tracks of GEFCom2014. *International Journal of Forecasting*, 32(3):1094–1102, 2016.

Karr, A. F., Lin, X., Sanil, A. P., and Reiter, J. P. Privacy-preserving analysis of vertically partitioned data using secure matrix products. *Journal of Official Statistics*, 25(1):125, 2009.

Koenker, R. and Bassett Jr, G. Regression quantiles. *Econometrica: journal of the Econometric Society*, pages 33–50, 1978.

Kou, P., Gao, F., and Guan, X. Sparse online warped gaussian process for wind power probabilistic forecasting. *Applied Energy*, 108(C):410–428, 2013.

Kubáček, L. Multivariate regression model with constraints. *Mathematica Slovaca*, 57(3):271–296, 2007.

Kurtulmus, A. and Daniel, K. Trustless machine learning contracts; evaluating and exchanging machine learning models on the ethereum blockchain. *arXiv:1802.10185*, pages 1–11, 2018.

Langford, J., Li, L., and Zhang, T. Sparse online learning via truncated gradient. *Journal of Machine Learning Research*, 10:777–801, 2009.

Lau, A. and McSharry, P. Approaches for multi-step density forecasts with application to aggregated wind power. *Annals of Applied Statistics*, 4(3):1311–1341, 2010. doi: 10.1214/09-AOAS320.

Lenzi, A., Steinsland, I., and Pinson, P. Benefits of spatio-temporal modeling for short-term wind power forecasting at both individual and aggregated levels. *Environmetrics*, 29(3):e2493, 2018.

Lerch, S., Thorarinsdottir, T. L., Ravazzolo, F., and Gneiting, T. Forecaster's dilemma: Extreme events and forecast evaluation. *Statistical Science*, 32(1):106–127, 2017.

Li, Q. and Cao, G. Efficient and privacy-preserving data aggregation in mobile sensing. In *2012 20th IEEE International Conference on Network Protocols (ICNP)*, pages 1–10. IEEE, 2012.

Li, S., Xue, K., Yang, Q., and Hong, P. PPMA: privacy-preserving multisubset data aggregation in smart grid. *IEEE Transactions on Industrial Informatics*, 14(2):462–471, 2018.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

Li, W., Li, H., and Deng, C. Privacy-preserving horizontally partitioned linear programs with inequality constraints. *Optimization Letters*, pages 137–144, 2013.

Li, X., Chen, J., Liu, W., and Wan, W. An improved AES encryption algorithm. *IeT International Communication Conference on Wireles Mobile and Computing (CCWMC 2009)*, pages 694–698, 2009.

Li, Y. and Genton, M. G. Single-index additive vector autoregressive time series models. *Scandinavian Journal of Statistic*, 36(3):369–388, September 2009.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Li, Y., Jiang, X., Wang, S., Xiong, H., and Ohno-Machado, L. VERTIcal Grid lOgistic regression (VERTIGO). *Journal of the American Medical Informatics Association*, 23(3):570–579, 2015a.

Li, Y., Jiang, X., Wang, S., Xiong, H., and Ohno-Machado, L. Vertical grid logistic regression (vertigo). *Journal of the American Medical Informatics Association*, 23(3):570–579, 2015b.

Liu, K., Giannella, C., and Kargupta, H. A survey of attack techniques on privacy-preserving data perturbation methods. In *Privacy-Preserving Data Mining*, pages 359–381. Springer, 2008.

Liu, Y., Guo, W., Fan, C.-I., Chang, L., and Cheng, C. A practical privacy-preserving data aggregation (3PDA) scheme for smart grid. *IEEE Transactions on Industrial Informatics*, 15(3):1767–1774, 2018a.

Liu, Y., Guo, W., Fan, C.-I., Chang, L., and Cheng, C. A practical privacy-preserving data aggregation (3PDA) scheme for smart grid. *IEEE Transactions on Industrial Informatics*, 15(3):1767–1774, 2018b.

Lu, C.-L., Wang, S., Ji, Z., Wu, Y., Xiong, L., Jiang, X., and Ohno-Machado, L. WebDISCO: a web service for distributed Cox model learning without patient-level data sharing. *Journal of the American Medical Informatics Association*, 22(6):1212–1219, 2015.

Ma, X., Zhu, Y., and Li, X. An efficient and secure ridge regression outsourcing scheme in wearable devices. *Computers & Electrical Engineering*, 63:246–256, 2017.

Mangasarian, O. L. Privacy-preserving linear programming. *Optimization Letters*, 5(1):165–172, 2011.

Mangasarian, O. L. Privacy-preserving horizontally partitioned linear programs. *Optimization Letters*, 6(3):431–436, 2012.

Matamoros, J. Asynchronous online ADMM for consensus problems. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5875–5879, 2017.

Mateos, G., Bazerque, J. A., and Giannakis, G. B. Distributed sparse linear regression. *IEEE Transactions on Signal Processing*, 58(10):5262–5276, 2010a.

Mateos, G., Bazerque, J. A., and Giannakis, G. B. Distributed sparselinear regression. *IEEE Transactions on Signal Processing*, 58(10):5262–5276, 2010b.

Matos, M., Bessa, R., Botterud, A., and Zhou, Z. *Forecasting and setting power system operating reserves*, pages 279–308. Renewable Energy Forecasting: From Models to Applications. Woodhead Publishing, 2017. doi: 10.1016/B978-0-08-100504-0.00011-1.

Matos, M., Bessa, R. J., Gonçalves, C., Cavalcante, L., Miranda, V., Machado, N., Marques, P., and Matos, F. Setting the maximum import net transfer capacity under extreme res integration scenarios. In *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, pages 1–7. IEEE, 2016.

Matos, M. A. and Bessa, R. J. Setting the operating reserve using probabilistic wind power forecasts. *IEEE Transactions on Power Systems*, 26(2):594–603, 2010.

Mazzi, N. and Pinson, P. *Wind power in electricity markets and the value of forecasting*, pages 259–278. Renewable Energy Forecasting: From Models to Applications. Woodhead Publishing, 2017. doi: 10.1016/B978-0-08-100504-0.00010-X.

McNeil, A. J. and Saladin, T. The peaks over thresholds method for estimating high quantiles of loss distributions. In *Proceedings of 28th International ASTIN Colloquium*, pages 23–43, 1997.

Messner, J. W., Pinson, P., Browell, J., Bjerregård, M. B., and Schicker, I. Evaluation of wind power forecasts – an up-to-date view. *Wind Energy*, 23(6):1461–1481, 2020.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Messner, J. W. and Pinson, P. Online adaptive lasso estimation in vector autoregressive models for high dimensional wind power forecasting. *International Journal of Forecasting*, 2018. available online.

Messner, J. W. and Pinson, P. Online adaptive lasso estimation in vector autoregressive models for high dimensional wind power forecasting. *Int. Journal of Forecasting*, 35(4):1485–1498, October 2019.

Messner, J. W., Zeileis, A., Broecker, J., and Mayr, G. J. Probabilistic wind power forecasts with an inverse power curve transformation and censored regression. *Wind Energy*, 17(11):1753–1766, 2013.

Min, W. and Wynter, L. Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, 19(4):606–616, 2011. doi: 10.1016/j.trc.2010.10.002.

Mohassel, P. and Zhang, Y. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 19–38. IEEE, 2017.

Møller, J., Nielsen, H., and Madsen, H. Time-adaptive quantile regression. *Computational Statistics & Data Analysis*, 52(3):1292–1303, 2008. ISSN 0167-9473. doi: 10.1016/j.csda.2007.06.027.

Nemirovski, A. S. and Yudin, D. B. *Problem complexity and method efficiency in optimization*. Wiley, 1983.

Nesterov, Y. Introductory lectures on convex programming volume i: Basic course. *Lecture notes*, 3(4):5, 1998.

Nicholson, W. B., Matteson, D. S., and Bien, J. VARX-L: Structured regularization for large vector autoregressions with exogenous variables. *International Journal of Forecasting*, 33(3):627–651, 2017.

Nikolaenko, V., Weinsberg, U., Ioannidis, S., Joye, M., Boneh, D., and Taft, N. Privacy-preserving ridge regression on hundreds of millions of records. In *2013 IEEE Symposium on Security and Privacy*, pages 334–348. IEEE, 2013.

Nocedal, J. and Wright, S. *Numerical optimization*. Springer Science & Business Media, 2006.

Nogueira, F. Python bayesian optimization implementation. `http://github.com/fmfn/BayesianOptimization`, 2020.

Nystrup, P., Lindström, E., Pinson, P., and Madsen, H. Temporal hierarchies with autocorrelation for load forecasting. *European Journal of Operational Research*, 280(3):876–888, 2020.

Paci, L., Gelfand, A. E., and Holland, D. M. Spatio-temporal modeling for real-time ozone forecasting. *Spatial Statistics*, 4:79–93, 2013.

Papadimitriou, S., Li, F., Kollios, G., and Yu, P. S. Time series compressibility and privacy. In *Proceedings of the 33rd international conference on Very large data bases*, pages 459–470. VLDB Endowment, 2007.

Pinson, P. Very-short-term probabilistic forecasting of wind power with generalized logit—normal distributions. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 61(4):555–576, 2012a. ISSN 00359254, 14679876. URL `http://www.jstor.org/stable/23251160`.

Pinson, P. Very short-term probabilistic forecasting of wind power with generalized logit-normal distributions. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(4):555–576, 2012b.

Pinson, P. Introducing distributed learning approaches in wind power forecasting. In *Proceedings of the 2016 Int. Conf. on Prob. Methods Applied to Power Sys. (PMAPS)*, Beijing, China, October 2016a.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Pinson, P. Introducing distributed learning approaches in wind power forecasting. In *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, pages 1–6. IEEE, 2016b.

Pinson, P. and Madsen, H. Adaptive modelling and forecasting of offshore wind power fluctuations with markov-switching autoregressive models. *Journal of Forecasting*, 31(4):281–313, 2012. ISSN 0277-6693. doi: 10.1002/for.1194.

Pinson, P., Madsen, H., Nielsen, H. A., Papaefthymiou, G., and Klöckl, B. From probabilistic forecasts to statistical scenarios of short-term wind power production. *Wind Energy*, 12(1):51–62, 2009.

Rathore, B. S., Singh, A., and Singh, D. A survey of cryptographic and non-cryptographic techniques for privacy preservation. *International Journal of Computer Applications*, 975:8887, 2015.

Ravi, S. and Al-Deek, H. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *Journal of Intelligent Transportation Systems*, 13(2):53–72, 2009.

Shalev-Shwartz, S. and Tewari, A. Stochastic methods for l_1-regularized loss minimization. *Journal of Machine Learning Research*, 12 (Jun):1865–1892, 2011.

Slavkovic, A. B., Nardi, Y., and Tibbits, M. M. Secure logistic regression of horizontally and vertically partitioned distributed databases. In *icdmw*, pages 723–728. IEEE, 2007.

Sommer, B., Pinson, P., Messner, J., and Obst, D. Online distributed learning in wind power forecasting. *International Journal of Forecasting*, 37(1):205–223, January 2021. doi: 10.1016/j.ijforecast.2020.04.004.

Song, S., Chaudhuri, K., and Sarwate, A. D. Stochastic gradient descent with differentially private updates. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 245–248. IEEE, 2013.

Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., and Megías, D. Individual differential privacy: A utility-preserving formulation of differential privacy guarantees. *IEEE Transactions on Information Forensics and Security*, 12(6):1418–1429, 2017.

Suzuki, T. Dual averaging and proximal gradient descent for online Alternating Direction Multiplier Method. *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 28: 392–400, 2013.

Sweeney, C., Bessa, R. J., Browell, J., and Pinson, P. The future of forecasting for renewable energy. *Wiley Interdisciplinary Reviews: Energy and Environment*, 9(2):e365, March 2020a.

Sweeney, C., Bessa, R. J., Browell, J., and Pinson, P. The future of forecasting for renewable energy. *Wiley Interdisciplinary Reviews: Energy and Environment*, 9(2):e365, 2020b.

Taillardat, M., Fougères, A.-L., Naveau, P., and de Fondeville, R. Extreme events evaluation using CRPS distributions. *arXiv preprint arXiv:1905.04022*, 2019.

Tascikaraoglu, A. Evaluation of spatio-temporal forecasting methods in various smart city applications. *Renewable and Sustainable Energy Reviews*, 82(1):424–435, February 2018.

Tastu, J., Pinson, P., and Madsen, H. 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.

Tastu, J., Pinson, P., Trombe, P., and Madsen, H. Probabilistic forecasts of wind power generation accounting for geographically dispersed information. *IEEE Transactions on Smart Grid*, 5(1):480–489, 2012.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Tastu, J., Pinson, P., Kotwa, E., Madsen, H., and Nielsen, H. A. Spatio-temporal analysis and modeling of short-term wind power forecast errors. *Wind Energy*, 14(1):43–60, January 2011.

Tastu, J., Pinson, P., Trombe, P.-J., and Madsen, H. Probabilistic forecasts of wind power generation accounting for geographically dispersed information. *IEEE Transactions on Smart Grid*, 5(1): 480–489, 2014.

Toda, H. Y. and Phillips, P. C. Vector autoregressions and causality. *Econometrica: Journal of the Econometric Society*, pages 1367–1393, 1993.

Tran, H.-Y. and Hu, J. Privacy-preserving big data analytics a comprehensive survey. *Journal of Parallel and Distributed Computing*, 134:207–218, 2019.

van Erven, T. and Cugliari, J. *Game-theoretically optimal reconciliation of contemporaneous hierarchical time-series forecasts*, pages 297–316. Modeling and Stochastic Learning for Forecasting in High Dimensions. Springer, 2015.

Wang, H. and Banerjee, A. Online alternating direction method. In *Proceedings of the 29th International Conference on Machine Learning*, volume 2, pages 1119–1126, 2012.

Wang, H. J. and Li, D. Estimation of extreme conditional quantiles through power transformation. *Journal of the American Statistical Association*, 108(503):1062–1074, 2013.

Wang, H. J., Li, D., and He, X. Estimation of high conditional quantiles for heavy-tailed distributions. *Journal of the American Statistical Association*, 107(500):1453–1464, 2012.

Wickramasuriya, S. L., Athanasopoulos, G., and Hyndman, R. J. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Society*, 114(526):804–819, 2018.

Wu, Y., Jiang, X., Kim, J., and Ohno-Machado, L. Grid binary LOgistic REgression (GLORE): building shared models without sharing data. *Journal of the American Medical Informatics Association*, 19(5):758–764, 2012.

Yang, Q., Liu, Y., Chen, T., and Tong, Y. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):12, 2019.

Yu, S., Fung, G., Rosales, R., Krishnan, S., Rao, R. B., Dehing-Oberije, C., and Lambin, P. Privacy-preserving cox regression for survival analysis. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1034–1042. ACM, 2008.

Zhang, C., Ahmad, M., and Wang, Y. ADMM based privacy-preserving decentralized optimization. *IEEE Transactions on Information Forensics and Security*, 14(3):565–580, 2019.

Zhang, T. and Zhu, Q. Dynamic differential privacy for ADMM-based distributed classification learning. *IEEE Transactions on Information Forensics and Security*, 12(1):172–187, 2017.

Zhang, X., Khalili, M. M., and Liu, M. Recycled ADMM: Improve privacy and accuracy with less computation in distributed algorithms. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 959–965. IEEE, 2018.

Zhang, Y. and Dong, J. Least squares-based optimal reconciliation method for hierarchical forecasts of wind power generation. *IEEE Transactions on Power Systems*, 2018.

Zhang, Y. and Wang, J. A distributed approach for wind power probabilistic forecasting considering spatio-temporal correlation without direct access to off-site information. *IEEE Transactions on Power Systems*, 2018a.

Zhang, Y. and Wang, J. A distributed approach for wind power probabilistic forecasting considering spatio-temporal correlation without direct access to off-site information. *IEEE Transactions on Power Systems*, 33(5):5714–5726, 2018b.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Zhao, C., Zhao, S., Zhao, M., Chen, Z., Gao, C.-Z., Li, H., and Tan, Y.-a. Secure multi-party computation: Theory, practice and applications. *Information Sciences*, 476:357–372, 2019.

Zhao, Y., Ye, L., Pinson, P., Tang, Y., and Lu, P. Correlation-constrained and sparsity-controlled vector autoregressive model for spatio-temporal wind power forecasting. *IEEE Transactions on Power Systems*, 33(5):5029–5040, September 2018.

Zhou, S., Lafferty, J., and Wasserman, L. Compressed and privacy-sensitive sparse regression. *IEEE Transactions on Information Theory*, 55(2):846–866, 2009.

Zhu, J., He, P., Zheng, Z., and Lyu, M. R. A privacy-preserving qos prediction framework for web service recommendation. In *2015 IEEE International Conference on Web Services*, pages 241–248. IEEE, 2015.

Zhu, Q., Chen, J., Shi, D., Zhu, L., Bai, X., Duan, X., and Liu, Y. Learning temporal and spatial correlations jointly: A unified framework for wind speed prediction. *IEEE Transactions on Sustainable Energy*, 11(1):509–523, January 2020.

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

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337

Smart4RES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864337