Decision-Focused Data Pooling for Contextual Stochastic Optimization

Akylas Stratigakos, Juan Miguel Morales, Salvador Pineda, Georges Kariniotakis

Decision-Focused Data Pooling for Contextual Stochastic Optimization

Akylas Stratigakos[a,1,*], Juan Miguel Morales[b], Salvador Pineda[b], Georges Kariniotakis[a]

[a] *Center PERSEE, Mines Paris, PSL University, Sophia Antipolis, 06904, France*
[b] *OASYS Research Group, University of Málaga, Málaga, 29010, Spain*

Abstract

Data scarcity poses a significant risk that hinders the deployment of advanced data-driven methods. In many cases of practical interest, decision-makers have access to data from similar, potentially unrelated, problem instances. Maximizing the benefits of data-driven methods thus necessitates novel methods to utilize all available data. In this work, we propose two methods to pool data when dealing with multiple contextually-dependent stochastic optimization problems. The first involves naively pooling data and training a global model to derive decisions across all problems, while the second leverages optimal transport for model aggregation. An essential contribution is the development of a decision-focused data pooling algorithm to determine when and how much data to pool. The proposed algorithm leverages tools from ensemble learning to estimate the expected out-of-sample decision cost without sacrificing training data, and effectively interpolates between a local and an anchor distribution. For validation, we examine the problem of stochastic renewable energy sources participating in electricity markets, which is pivotal for their integration into modern power systems. The results demonstrate that data pooling improves overall decision-making when data are scarce. Notably, the proposed decision-focused data pooling algorithm consistently outperforms both local and pooled methods.

*Corresponding author.
Email address: a.stratigakos@imperial.ac.uk (Akylas Stratigakos)
[1]Current address: Department of Electrical and Electronic Engineering, Imperial College London, London, SW7 2AZ, U.K.

Keywords: Decision support systems, contextual stochastic optimization, prescriptive analytics, data pooling

1. Introduction

Data are becoming increasingly important when dealing with challenges that arise in many areas such as supply chains, healthcare, and power systems. Data-driven methods leverage tools across optimization, machine learning, and statistics and enable significant improvements in decision-making under uncertainty (Baardman et al., 2022).

In real-world systems, decision-makers deal with a large number of uncertainties, which are also associated with some contextual information. In turn, these uncertainties can create thousands of potentially unrelated stochastic optimization problems. A prominent example is power systems where power producers manage portfolios of thousands of renewable energy sources, such as wind and solar power plants, whose production depends on the weather at each geographical location. Future power systems and smart grids that integrate a large number of heterogeneous assets, such as small-scale renewable energy sources, flexible loads, storage systems, and electric vehicles, further exacerbate this issue.

In this context, decision-makers often encounter a "large-scale, small-data" regime. That is, while the aggregate volume of data across all problems is large, data at an individual (*local*) problem level might be scarce or even contaminated, which hinders the deployment of data-driven methods. To fully utilize the benefits of available data-driven methods, it is critical to develop effective methods for pooling the available data from different problems, thus enabling improved decision-making.

1.1. Aim and Contribution

In this work, we first propose two methods for data pooling to improve decision performance when dealing with multiple contextually-dependent stochastic optimization problems. The first involves naively pooling all data and training a global model to estimate a conditional distribution of uncertainty as a function of contextual information, which is subsequently used to derive decisions across all problems. The second

approach implicitly performs data pooling through model aggregation using optimal transport (OT) (Peyré and Cuturi, 2019), a mathematical framework that studies similarities of probability distributions, by aggregating a number of models trained locally to solve each separate problem. To determine when and how much data to pool, we further develop a *decision-focused* data pooling algorithm that interpolates between a local and an anchor distribution. The proposed algorithm leverages techniques from ensemble learning, namely the Out-of-Bag (OOB) method (Hastie et al., 2009), to provide an estimation of the expected out-of-sample decision cost without sacrificing training data and avoiding model retraining, which can be computationally costly. We evaluate the effectiveness of the proposed methods in a critical application related to the integration of renewable energy sources in power systems, namely trading in a day-ahead electricity market. Our results show that data pooling leads to better decisions when data are scarce and that the proposed decision-focused algorithm consistently leads to improved performance, even as the number of local training observations increases.

1.2. Related Work

In recent years, there has been a growing interest in solving stochastic optimization problems where the uncertain parameters are associated with some contextual information. The standard data-driven approach consists of two steps. The first step involves forecasting uncertain parameters. In the second step, these forecasts are used as input in an optimization problem. To deal with the induced suboptimality of point forecasts, relevant work focuses on estimating the conditional distribution of uncertainty (Bertsimas and Kallus, 2020), i.e., probabilistic forecasting; training (point or probabilistic) forecasting models to explicitly minimize downstream decision costs (Elmachtoub and Grigas, 2022), (Donti et al., 2017), (Muñoz et al., 2022), (Stratigakos et al., 2022); directly learning the solutions to the optimization problem (Ban and Rudin, 2019); or directly working with the joint distribution of contextual information and uncertainty (Esteban-Pérez and Morales, 2022). Nonetheless, the majority of relevant work focuses on dealing with a single problem and the setting of dealing with multiple contextually-dependent optimization problems simultaneously remains largely unexplored. Gupta

and Kallus (2022) examine data pooling for multiple stochastic optimization problems without contextual information and show that it leads to better decisions owing to the so-called instability versus suboptimality trade-off. Intuitively, data pooling is most useful when data are scarce and the respective local solution, i.e., the solution that leverages only local problem data, is unstable. To determine when and how much data to pool across problems, Gupta and Kallus (2022) further develop an algorithm based on cross-validation that exploits the structure of the optimization problem, which, nonetheless, does not account for contextual information.

Conversely, in the area of time series forecasting, there is a growing interest in developing global forecasting models. The term *global* forecasting model refers to a single univariate model trained by pooling data across a large number of time series, while *local* forecasting model refers to a univariate model trained for a specific time series. Global forecasting models are considered an effective method of simultaneously reducing modeling effort and enabling cross-learning across tasks. For instance, Salinas et al. (2020) propose a global deep learning model for probabilistic demand forecasting, while Montero-Manso and Hyndman (2021) show that global models can perform on par with local models for time series forecasting, but may have a lower representational capacity for regression tasks. Global forecasting is gaining interest in areas where multiple sources of uncertainty appear, such as power systems. Kazmi et al. (2021) propose global models to forecast the uncertain renewable production of multiple plants or the individual consumption at a household level. Balint et al. (2023) examine centralized (i.e., global) and federated learning to forecast the temperature of thermostatically controlled loads using domain-informed data augmentation. Grabner et al. (2022) propose a global model for load forecasting in the distribution grid and a clustering-based localization method to improve performance under data heterogeneity. To cold-start the forecasting problem for a residential solar panel without historical data, Bottieau et al. (2022) train a generic cross-learning model across several series. However, moving from a local to a global model as a function of the volume of available data or the quality of individual models has not received much attention.

In this work, we focus on the case of multiple stochastic problems where, for each problem, the decision-maker uses contextual data to approximate the conditional marginal distribution of uncertainty via probabilistic forecasting. By viewing each model as an expert and aggregating their output leveraging OT (Peyré and Cuturi, 2019), our approach is closely related to model aggregation and forecast combination. Namely, the works by Papayiannis and Yannacopoulos (2015, 2018) are the ones most closely related to ours. Papayiannis and Yannacopoulos (2015) and Papayiannis and Yannacopoulos (2018) leverage OT to combine experts' opinions (i.e., forecasts) of a reference probability distribution, via means of a weighted Wasserstein barycenter (Agueh and Carlier, 2011) and show that, for the univariate case, this is equivalent to quantile averaging (Lichtendahl Jr et al., 2013). Papayiannis et al. (2018) further extend this approach to the linear aggregation of point predictions for wind speed by aggregating forecasts in adjacent spatial locations.

Our work differs in several key aspects. First, we motivate our approach through the lens of stochastic optimization and decision-making. That is, our goal is to combine forecasts (or, equivalently, aggregate models) in a decision-focused way that leads to lower downstream decision costs. While some previous works have considered forecast combinations to minimize decision costs by minimizing the newsvendor loss (Trapero et al., 2019), our setting is more general. Second, we explicitly focus on the case of scarce training data and leverage tools from bootstrapping and cross-validation to estimate the out-of-sample decision performance. Third, we assume that individual models are trained on data from different sources, each one modeling the association between a different pair of uncertain parameters and contextual information. This differs from the typical setting of forecast combination which assumes a different set of forecasts modeling the same distribution.

1.3. Paper Outline

The remainder of this paper is organized as follows. Section 2 presents a short background on OT. Section 3 introduces the main problem. Section 4 develops two data pooling methods and Section 5 formulates the proposed decision-focused algorithm that

determines when and how much data to pool. Section 6 discusses the numerical results, and Section 7 concludes and provides directions for future work.

Notation. Boldfaced lowercase letters, e.g., $\mathbf{x}$, denote vectors, and boldfaced uppercase letters, e.g., $\mathbf{X}$, denote matrices. Sets are denoted with calligraphic font, e.g., $\mathcal{S}$, and $|\mathcal{S}|$ denotes the cardinality (number of elements) of a set $\mathcal{S}$. Scalars are denoted with ordinary letters, either lowercase or uppercase, e.g., n or N, and $\mathbf{1}_n$ denotes an n-size vector of ones.

2. Preliminaries on Optimal Transport

This section provides preliminaries on OT, namely, introduces the OT problem (in Section 2.1) and the Wasserstein barycenter (in Section 2.2).

2.1. Optimal Transport Problem

We consider a histogram $\mathbf{a} \in \Sigma_n$ of n values, where

$$\Sigma_n = \{\mathbf{a} \in \mathbb{R}_+^n \,|\, \mathbf{a}^\top \mathbf{1}_n = 1\}$$

is the standard $(n-1)$-dimensional probability simplex. The terms histogram and probability vector are used interchangeably throughout.

A discrete measure with weights $\mathbf{a}$ and locations $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_n \in \Xi$ reads

$$\alpha = \sum_{i=1}^{n} \mathrm{a}_i \delta_{\boldsymbol{\xi}_i}, \tag{1}$$

where $\delta_{\boldsymbol{\xi}_i}$ is the Dirac delta distribution at position $\boldsymbol{\xi}_i$, intuitively a unit of mass that is concentrated at location $\boldsymbol{\xi}_i$. Such a measure is a probability measure if, additionally, $\mathbf{a} \in \Sigma_n$.

The OT problem seeks to find the best way to transport a given number of goods from a set of sources to a set of destinations, where the cost of transporting each unit of goods from each source to each destination is known. Formally, consider two discrete measures α, β of the form (1) with corresponding histograms $\mathbf{a} \in \Sigma_n$, $\mathbf{b} \in \Sigma_m$ and respective support locations $\boldsymbol{\xi}_i, i = 1, \ldots, n$, and $\boldsymbol{\xi}'_j, j = 1, \ldots, m$. Let $\mathbf{C} \in \mathbb{R}_+^{n \times m}$ be a

known cost matrix, where $c_{i,j}$ stores the cost of transporting a unit of goods from $\boldsymbol{\xi}_i$ to $\boldsymbol{\xi}'_j$. Further, let the polytope of admissible couplings between $\mathbf{a}, \mathbf{b}$ be

$$\mathbf{U}(\mathbf{a}, \mathbf{b}) \stackrel{\text{def}}{=} \{\boldsymbol{\Gamma} \in \mathbb{R}_+^{n \times m} \,|\, \boldsymbol{\Gamma} \mathbf{1}_m = \mathbf{a}, \boldsymbol{\Gamma}^\top \mathbf{1}_n = \mathbf{b}\}. \tag{2}$$

The OT problem between $\mathbf{a}, \mathbf{b}$ is given by

$$W(\mathbf{a}, \mathbf{b}) \stackrel{\text{def}}{=} \min_{\boldsymbol{\Gamma} \in \mathbf{U}(\mathbf{a}, \mathbf{b})} < \boldsymbol{\Gamma}, \mathbf{C} > . \tag{3}$$

where $< \boldsymbol{\Gamma}, \mathbf{C} >= \sum_{i=1}^{n} \sum_{j=1}^{m} \gamma_{i,j} c_{i,j}$. The decision matrix $\boldsymbol{\Gamma}$ is the so-called transportation plan, with $\gamma_{i,j}$ representing the probability mass transported from the i-th source to the j-th destination, with (2) ensuring that the total amount of mass moved satisfies both each source supply and each demand destination and the non-negativity constraints.

If we further assume that $c_{i,j} = \|\boldsymbol{\xi}_i - \boldsymbol{\xi}'_j\|^r$, for some $r \geq 1$, where $\|\cdot\|$ is an arbitrary norm, then the optimal value of (3) is equal to the r-Wasserstein distance between measures α, β, raised to the r-th power. The Wasserstein distance is a distance metric between probability distributions that measures the minimum cost of transforming one distribution into the other and has many applications in different fields, such as computer vision, machine learning, and stochastic programming.

The OT problem (3) is a Linear Programming (LP) problem, which can be solved using off-the-shelf solvers. If α, β are defined on the real line, then a closed-form solution also exists, in the form of averaging quantile functions (Papayiannis and Yannacopoulos, 2018). To deal with the computational challenges associated with large-scale problems that arise in machine learning applications, several specialized algorithms have also been developed, such as entropic regularization schemes (Cuturi, 2013; Cuturi and Peyré, 2016). A comprehensive overview of OT with a focus on numerical methods is given by Peyré and Cuturi (2019).

2.2. Wasserstein Barycenter

We further consider S histograms $\{\mathbf{b}_s\}_{s=1}^S$, where $\mathbf{b}_s \in \Sigma_{n_s}$, and our goal is to estimate an "average histogram" over a grid of n fixed support locations. The Wasserstein barycenter (Agueh and Carlier, 2011), i.e., the generalized mean, is the histogram

$\mathbf{a} \in \Sigma_n$ that minimizes the weighted sum of the Wasserstein distances from $\{\mathbf{b}_s\}_{s=1}^S$. The Wasserstein barycenter $\mathbf{q}^*$ is given by

$$\mathbf{q}^* = \arg\min_{\mathbf{q} \in \Sigma_n} \sum_{s=1}^{S} \lambda_s W(\mathbf{q}, \mathbf{p}_s), \tag{4}$$

and is parameterized by a probability vector of $\boldsymbol{\lambda} \in \Sigma_S$ of known weights, termed *barycentric coordinates*. Note that each Wasserstein distance itself denotes a minimization problem. Evidently, problem (4) is also an LP problem, although its size is much larger than the OT problem (3). The Wasserstein barycenter is a generalization of the Euclidean mean in higher dimensions and computes a representative distribution for a set of distributions, and has found many applications in clustering, classification, model aggregation (Papayiannis and Yannacopoulos, 2018), and variational data assimilation problems (Feyeux et al., 2018). For measures defined on the real line, the Wasserstein barycenter can also be estimated efficiently with a closed-form solution.

3. Problem Formulation

In this section, we introduce the problem of contextual stochastic optimization (in Section 3.1). Then, we consider a scenario of multiple problems each associated with some contextual information (in Section 3.2), and describe the standard solution approach (in Section 3.3).

3.1. Preliminaries on Contextual Stochastic Optimization

We consider a contextual stochastic optimization problem given by

$$\min_{\mathbf{z} \in \mathcal{Z}} \mathbb{E}_{\mathbf{y}}[c(\mathbf{z}; \mathbf{y}) | \mathbf{x} = \mathbf{x}_0], \tag{5}$$

where $\mathbf{y} \in \mathcal{Y}$ denotes the uncertain problem parameters (e.g., uncertain demand or prices), $\mathbf{x} \in \mathcal{X}$ denotes some associated contextual features (e.g., weather or market conditions), $\mathbf{x}_0$ denotes a realization of $\mathbf{x}$, $\mathbf{z}$ denotes the decision variables, $\mathcal{Z}$ denotes the set of feasible solutions, c denotes a convex cost function, and the expectation is taken with respect to (w.r.t.) the conditional distribution of $\mathbf{y}$ given $\mathbf{x} = \mathbf{x}_0$.

We assume that the uncertain parameter $\mathbf{y}$ is a discrete random variable with finite support denoted by $\mathcal{Y} \stackrel{\text{def}}{=} \{\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_K\}$, where K is the number of support locations. For any $\mathbf{x} \in \mathcal{X}$, the true conditional distribution of $\mathbf{y}$ is given by a probability vector $\mathbf{p}(\mathbf{x}) \in \Sigma_K$, where Σ_K is the $(K-1)$-dimensional probability simplex. The k-th component of $\mathbf{p}(\mathbf{x})$ is defined as $p_k(\mathbf{x}) = \mathbb{P}(\mathbf{y} = \tilde{\mathbf{y}}_k|\mathbf{x})$, i.e., the probability of $\mathbf{y} = \tilde{\mathbf{y}}_k$ conditioned on contextual information $\mathbf{x}$. Thus, (5) can be equivalently written as

$$\min_{\mathbf{z}\in\mathcal{Z}} \mathbb{E}_{\mathbf{y}}[c(\mathbf{z};\mathbf{y})|\mathbf{x} = \mathbf{x}_0] = \min_{\mathbf{z}\in\mathcal{Z}} \sum_{k=1}^{K} p_k(\mathbf{x}_0) c(\mathbf{z};\tilde{\mathbf{y}}_k). \tag{6}$$

In practice, instead of the true probability vector $\mathbf{p}(\mathbf{x}_0)$, we have access to a training data set $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i)\}_{i=1}^N$ of N observations, which can be used to approximate (6). In this work, we focus on using a probabilistic forecasting model to estimate the true conditional distribution $\mathbf{p}(\mathbf{x})$. Assume a hypothesis class $\mathcal{H}$ of functions $f : \mathcal{X} \to \Sigma_K$ that map contextual information $\mathbf{x}$ to the conditional distribution of uncertainty $\mathbf{y}$. Note that since $f(\mathbf{x}) \in \Sigma_K$, the output of the learning model needs to satisfy a set of constraints. To keep the notation consistent, we refer to $\widehat{\mathbf{p}} : \mathcal{X} \to \Sigma_K$ as the model trained on available data, and to $\widehat{\mathbf{p}}(\mathbf{x}) \in \Sigma_K$ as the estimated probability vector for any $\mathbf{x} \in \mathcal{X}$.

To measure the decision quality of a model $\widehat{\mathbf{p}}(\mathbf{x}) : \mathcal{X} \to \Sigma_K$, we further define a function that measures the excess cost incurred by using $\widehat{\mathbf{p}}$ to approximate a problem of the form of (6) compared to the perfect foresight solution. To streamline the notation, given any $\mathbf{q} \in \Sigma_K$, we define $\mathbf{z}(\mathbf{q}) = \arg\min_{\mathbf{z}\in\mathcal{Z}} \sum_{k=1}^{K} q_k c(\mathbf{z};\tilde{\mathbf{y}}_k)$. Let

$$D(\widehat{\mathbf{p}}(\mathbf{x}_0), \mathbf{y}_0 \,|\, c, \mathcal{Z}) = c(\mathbf{z}(\widehat{\mathbf{p}}(\mathbf{x}_0)); \mathbf{y}_0) - c(\mathbf{z}^*; \mathbf{y}_0), \tag{7}$$

denote the excess cost incurred using $\widehat{\mathbf{p}}$ estimated with respect to the cost function c and the feasible set $\mathcal{Z}$, where $\mathbf{z}^* = \arg\min_{\mathbf{z}\in\mathcal{Z}} c(\mathbf{z};\mathbf{y}_0)$. Evidently, the cost estimated from D is always non-negative.

Remark 1. *In the special case where $c(\mathbf{z};\mathbf{y}) = \mathbf{y}^\top \mathbf{z}$, i.e., we deal with a linear objective function with unknown cost coefficients, then, for any $\mathbf{x} \in \mathcal{X}$,* (6) *becomes*

$$\min_{\mathbf{z}\in\mathcal{Z}} \sum_{k=1}^{K} p_k(\mathbf{x}_0) \mathbf{z}^\top \tilde{\mathbf{y}}_k = \min_{\mathbf{z}\in\mathcal{Z}} \mathbf{z}^\top \mathbb{E}[\mathbf{y}|\mathbf{x} = \mathbf{x}_0].$$

Thus, we can replace $\mathbf{p}(\mathbf{x}_0)$ *with the conditional expectation of* $\mathbf{y}$ *given* $\mathbf{x}$ *using a point forecasting model.*

A variety of methods can be employed to generate probabilistic forecasts, including parametric models, non-parametric models (Bertsimas and Kallus, 2020), conformal prediction (Angelopoulos and Bates, 2022), or multi-label classification. In this work, we focus on non-parametric estimation methods, specifically on tree-based ensembles like random forests (Breiman, 2001), as they achieve state-of-the-art performance in contextual stochastic optimization problems (Bertsimas and Kallus, 2020) with minimal tuning effort. A number of extensions that embed the downstream optimization problem within tree-based methods also exist—see, e.g., (Kallus and Mao, 2022; Stratigakos et al., 2022; Elmachtoub et al., 2020).

Specifically, non-parametric machine learning models learn a function that assigns weights $\boldsymbol{\omega}(\mathbf{x}) \in \Sigma_N$ to training observations $\mathbf{y}_i$ based on contextual information $\mathbf{x}$. Then, (6) is approximated by

$$\min_{\mathbf{z}\in\mathcal{Z}} \sum_{i=1}^{N} \omega_i(\mathbf{x}_0) c(\mathbf{z}; \mathbf{y}_i). \tag{8}$$

Further consider an ensemble of T decision trees $\{\tau_1, \dots, \tau_T\}$ grown with the random forest method (Breiman, 2001), where $\tau_j : \mathcal{X} \to \{1, \dots, L_j\}$ is a map that corresponds to a disjoint partition of $\mathcal{X}$ into L_j tree leaves and $\tau_j(\mathbf{x})$ is the leaf identity. In this case, the respective weights are given by

$$\omega_i(\mathbf{x}_0) = \frac{1}{T} \sum_{j=1}^{T} \frac{\mathbb{I}\left[\tau_j(\mathbf{x}_i) = \tau_j(\mathbf{x}_0)\right]}{\sum_{i'=1}^{N} \mathbb{I}\left[\tau_j(\mathbf{x}_{i'}) = \tau_j(\mathbf{x}_0)\right]}, \tag{9}$$

where is $\mathbb{I}[\cdot]$ is the indicator function. Evidently, as $\mathbf{y}$ has finite support, we can count the number of times $\tilde{\mathbf{y}}_k$ appears in $\mathcal{D}$ and aggregate the respective weights $\omega_i(\mathbf{x}_0)$ to equivalently write (8) with a probability vector that weighs each support location. That is, the estimated probability of $\mathbf{y} = \tilde{\mathbf{y}}_k$ conditioned on $\mathbf{x} = \mathbf{x}_0$ is given by $\widehat{p}_k(\mathbf{x}) = \sum_{i=1}^{N} \mathbb{I}\left[\mathbf{y}_i = \tilde{\mathbf{y}}_k\right] \omega_i(\mathbf{x})$.

3.2. Dealing with Multiple, Contextually-Dependent Problems

We next discuss the main problem of interest, which is solving a collection of S potentially independent stochastic optimization problems, where each uncertainty is associated with some contextual information, specified by

$$\frac{1}{S}\sum_{s=1}^{S}\min_{\mathbf{z}_s\in\mathcal{Z}_s}\mathbb{E}_{\mathbf{y}_s}[c_s(\mathbf{z}_s;\mathbf{y}_s)|\mathbf{x}_s=\mathbf{x}_{s,0}]=\frac{1}{S}\sum_{s=1}^{S}\min_{\mathbf{z}_s\in\mathcal{Z}_s}\sum_{k=1}^{K_s}p_{s,k}(\mathbf{x}_{s,0})c_s(\mathbf{z}_s;\tilde{\mathbf{y}}_{s,k}),\tag{10}$$

where $\mathbf{y}_s$ represents the uncertain parameters, $\mathcal{Z}_s$ is the set of feasible solutions, $\mathbf{x}_{s,0}$ is a realization of the context $\mathbf{x}_s$, and $\mathbf{p}_s(\mathbf{x}_s)\in\Sigma_{K_s}$ denotes the true conditional distribution of $\mathbf{y}_s$ given $\mathbf{x}_s$. Throughout, subscript s is used to indicate that we are referring to the s-th subproblem[2].

We are particularly interested in the case where the uncertainty $\mathbf{y}_s$ and the contextual information $\mathbf{x}_s$ represent the same variables across all problems, which is a common setting; for instance, $\mathbf{y}_s$ could be the uncertain renewable energy production and $\mathbf{x}_s$ associated weather forecasts, with s indicating a specific geographical location. Thus, we assume that $\tilde{\mathbf{y}}_{s,k}=\tilde{\mathbf{y}}_k$, $K_s=K$, and $\mathcal{X}_s=\mathcal{X}$. To further simplify the notation, we assume, without loss of generality, that $c_s(\mathbf{z};\mathbf{y})=c(\mathbf{z};\mathbf{y})$ and $\mathcal{Z}_s=\mathcal{Z}$. Problem (10) can be equivalently written as

$$\frac{1}{S}\sum_{s=1}^{S}\min_{\mathbf{z}_s\in\mathcal{Z}}\sum_{k=1}^{K}p_{s,k}(\mathbf{x}_{s,0})c(\mathbf{z}_s;\tilde{\mathbf{y}}_k).\tag{11}$$

Note that the true conditional distributions $\mathbf{p}_s(\mathbf{x})$ may differ across problems and are, naturally, unknown. Instead, for each subproblem, we have access to a local training data set $\mathcal{D}_s=\{(\mathbf{y}_{s,i},\mathbf{x}_{s,i})\}_{i=1}^{N_s}$ of N_s observations, with subscript s highlighting that training observations differ across problems; the same also holds true for the out-of-sample realizations $\mathbf{x}_{s,0}$. Similar to the case of the single problem, our goal is to use the available data sets to approximate (11).

[2]For simplicity, we assume that all problems are weighted equally in the objective.

3.3. The Standard Local Solution Approach

In the absence of coupling constraints or variables across the S problems in (11), the standard approach is to decouple them and solve them separately using the local data sets. Specifically, consider a probabilistic forecasting model $\widehat{\mathbf{p}}_s : \mathcal{X} \to \Sigma_K$ trained on the local data set $\mathcal{D}_s$. The decoupled solution of (11) is then given by

$$\left\{ \min_{\mathbf{z} \in \mathcal{Z}} \sum_{k=1}^{K} \widehat{p}_{s,k}(\mathbf{x}_{s,0}) c(\mathbf{z}; \tilde{\mathbf{y}}_k) \right\}_{s=1,\ldots,S}, \tag{12}$$

where $\widehat{\mathbf{p}}_s(\mathbf{x}_{s,0}) \in \Sigma_K$ is an estimated probability vector. We consider (12) to be the standard benchmark of solving (10) and refer to it as the *local* approach, as it relies solely on the local data set $\mathcal{D}_s$ when solving the s-th subproblem.

However, if the local training data sets are scarce, the learned models may incur a high degree of misspecification and lead to poor out-of-sample performance. Therefore, we investigate whether pooling data across the S problems can be beneficial.

4. Data Pooling Methods

In this section, we describe different approaches to leverage data across problems to improve decision performance across a number of problems, namely, a method based on naive data pooling (in Section 4.1) and a method based on OT (in Section 4.2).

4.1. Global Model with Naive Data Pooling

A straightforward approach for data pooling is to combine all local data sets $\{\mathcal{D}_s\}_{s=1}^{S}$ and train a single, centralized probabilistic forecasting model. Let $\mathcal{D}^{\texttt{pool}} = (\mathcal{D}_1, \ldots, \mathcal{D}_S)$ be the concatenation of all data sets, and let $\widehat{\mathbf{p}}^{\texttt{pool}} : \mathcal{X} \to \Sigma_K$ be a global probabilistic forecasting model. Then, problem (10) can be approximated by solving S decoupled problems given by

$$\left\{ \min_{\mathbf{z} \in \mathcal{Z}} \sum_{k=1}^{K} \widehat{p}_k^{\texttt{pool}}(\mathbf{x}_{s,0}) c(\mathbf{z}; \tilde{\mathbf{y}}_k) \right\}_{s=1,\ldots,S}, \tag{13}$$

i.e., the decoupled problems are solved using a common forecasting model. For a non-parametric machine learning model (8), we first train a single model using data set $\mathcal{D}^{\texttt{pool}}$ and then estimate weights $\boldsymbol{\omega}^{\texttt{pool}}(\mathbf{x}) \in \Sigma_{N^{\texttt{pool}}}$, where $N^{\texttt{pool}} = |\mathcal{D}^{\texttt{pool}}|$.

Following the forecasting terminology (Salinas et al., 2020), we refer to this approach as a global model with naive data pooling. In practice, this requires a centralized entity that collects all the data and trains the global model, which may create issues regarding data leakage and raise privacy concerns. Considering a federated learning framework where the global model is trained without sharing data across the S subproblems can ameliorate privacy concerns.

4.2. Optimal Transport-based Data Pooling

In this section, we propose implicit data pooling via means of model aggregation based on OT that does not rely on centralized data collection. Following the standard local approach described in Section 3.2, we assume S local models $\widehat{\mathbf{p}}_s : \mathcal{X} \to \Sigma_K$ that map contextual information to probability vectors $\widehat{\mathbf{p}}_s(\mathbf{x}) \in \Sigma_K$. Our goal is, for each $\mathbf{x} \in \mathcal{X}$, to combine knowledge across the S problems by estimating representative conditional distributions. Let $\mathbf{g} : \mathcal{X} \to \Sigma_K$ be defined as

$$\mathbf{g}(\mathbf{x}) = \arg\min_{\mathbf{q}} \sum_{s=1}^{S} \lambda_s W(\mathbf{q}, \mathbf{p}_s(\mathbf{x})). \tag{14}$$

In words, $\mathbf{g}$ is a function that aggregates the S models by estimating the Wasserstein barycenter of their output for a realization of contextual information $\mathbf{x}$, parameterized by coordinates $\boldsymbol{\lambda} \in \Sigma_S$. Equivalently, we can view this as aggregating S probabilistic forecasting models by minimizing the Wasserstein distance of their outputs. Problem (10) can now be approximated by solving S decoupled problems given by

$$\left\{ \min_{z \in \mathcal{Z}} \sum_{k=1}^{K} g_k(\mathbf{x}_{s,0}) c(\mathbf{z}; \tilde{\mathbf{y}}_k) \right\}_{s=1,\ldots,S}. \tag{15}$$

As in the previous case, all problems leverage the same function to derive conditional distributions. However, unlike the naive data pooling approach for learning a global model, we do not require centralized access to the local training data sets. Rather, we only require access to the conditional marginal distributions, i.e., the outputs of $\widehat{\mathbf{p}}_s$. Hence, the model training phase remains the same as the local approach, and only the inference phase is affected. Also, note that the Wasserstein barycenter can be estimated in a decentralized way to further minimize data leakage across the subproblems.

5. Decision-Focused Data Pooling

We previously presented two approaches for pooling data across multiple subproblems: naive data pooling and OT-based model aggregation. A potential shortcoming associated with both approaches is model misspecification due to data heterogeneity. For instance, concept drift, i.e., the case when the true joint distribution between $\mathbf{y}$ and $\mathbf{x}$ differs across the subproblems, poses a major challenge. A global model may not generalize well to all subproblems, while OT-based model aggregation assumes that all local models are equally informative. To address this issue, we develop a decision-focused data pooling algorithm that interpolates between the local and the global approaches based on the expected out-of-sample cost of the downstream optimization problem.

First, we introduce a procedure to estimate the expected decision cost using the OOB method, which sets the foundation for our method (in Subsection 5.1). Next, we present our decision-focused data pooling algorithm (in Subsection 5.2).

5.1. OOB Estimation of the Decision Cost

This section describes how to estimate the out-of-sample decision cost of a trained model building on the OOB error method, which is a technique used in ensemble learning to estimate model performance without the need for a separate validation set. The reason for building our proposed approach on the OOB method is twofold. First, it allows us to jointly train and test a model, which is considerably less computationally costly than cross-validation. Second, it leverages the full training data set and does not require a separate validation set, making it advantageous when training data are scarce.

We consider an ensemble model of weak base learners trained using bootstrap aggregation (*bagging*), e.g., a random forest model. That is, during the training process, each base learner is trained on a new data set created by subsampling with replacement (bootstrapping) from the original training data set. The predictions of the models inferred from the base learners are then aggregated via, e.g., averaging— see (Hastie et al., 2009, Ch. 8) for details. By evaluating predictions on observations not used in building a specific base learner, bagging allows for evaluating the so-called OOB error, which

provides an estimate of the out-of-sample prediction error. As the number of training observations increases, the OOB error converges to the leave-one-out cross-validation error (Breiman, 1996).

We now describe a novel approach to evaluating the expected out-of-sample decision cost, by adapting the OOB method to a prescriptive context. For simplicity, we consider the case of a single model and drop subscript s. Consider a problem of the form of (6) approximated using an ensemble model $\widehat{\mathbf{p}} : \mathcal{X} \to \Sigma_K$ composed of weak base learners, trained either to minimize prediction error or the downstream decision cost. The decision-focused OOB method is described as follows. For $i = 1, \ldots, N$, we find all the models inferred from the base learners for which the i-th observation was not used for training. These models can be considered a new ensemble model, which we use to estimate a conditional distribution $\widehat{\mathbf{p}}_i^{\texttt{OOB}}(\mathbf{x}_i)$. The OOB estimate of the decision cost is then evaluated as the average difference between the incurred decision cost and the perfect foresight solution, given by

$$\frac{1}{N} \sum_{i=1}^{N} D(\widehat{\mathbf{p}}_i^{\texttt{OOB}}(\mathbf{x}_i), \mathbf{y}_i \,|\, c, \mathcal{Z}). \tag{16}$$

Notably, the key distinction from the standard OOB error method is that the decision-focused OOB method solves a weighted sample average approximation (Bertsimas and Kallus, 2020) of a stochastic optimization problem for each OOB observation and measures the incurred decision cost. In contrast, the standard OOB error method involves averaging the base learner predictions and estimating the prediction error[3]. The decision-focused OOB method also has potential applications in searching for model hyperparameters that lead to the smallest decision cost, similar to the method proposed by Corredera and Ruiz (2023).

We next describe in detail how to estimate $\widehat{\mathbf{p}}_i^{\texttt{OOB}}(\mathbf{x})$ for the case when $\widehat{\mathbf{p}}$ is a random forest model. Consider a random forest composed of T trees $\{\tau_1, \ldots, \tau_T\}$ that outputs weights $\boldsymbol{\omega}(\mathbf{x}) \in \Sigma_N$ of the form (9) for any $\mathbf{x} \in \mathcal{X}$, where τ_j is trained using a bootstrapped version of $\mathcal{D}$. For the i-th observation, let $\mathcal{T} \subseteq [T]$ be the subset of trees that

[3]For simplicity, we assume a regression setting where the target variable is continuous.

did not use that observation for training. Further, let $\mathcal{D}' = \mathcal{D} \setminus \{(\mathbf{y}_i, \mathbf{x}_i)\}$ be a surrogate data set that excludes the i-th training observation from the original data set $\mathcal{D}$. For each observation of $\mathcal{D}'$, indexed by subscript l, we estimate weights

$$\omega_l^{\texttt{OOB}}(\mathbf{x}_i) = \frac{1}{|\mathcal{T}|} \sum_{j \in \mathcal{T}} \frac{\mathbb{I}\left[\tau_j(\mathbf{x}_l) = \tau_j(\mathbf{x}_i)\right]}{\sum_{l'=1}^{N-1} \mathbb{I}\left[\tau_j(\mathbf{x}_{l'}) = \tau_j(\mathbf{x}_i)\right]},$$

which are of the form of (9) but only consider a subset of trees. Note that the i-th observation is removed from the original data set to avoid potential bias. Finally, for $k = 1, \ldots, K$, we estimate $\widehat{p}_{i,k}^{\texttt{OOB}}(\mathbf{x}_i) = \sum_{l=1}^{N-1} \mathbb{I}\left[\mathbf{y}_l = \tilde{\mathbf{y}}_k\right] \omega_l^{\texttt{OOB}}(\mathbf{x}_i)$.

5.2. Decision-Focused Barycentric Interpolation

Algorithm 1 `PrescrInterp`

Input: training data sets $\{\mathcal{D}_s\}_{s=1}^S$, local models $\{\widehat{\mathbf{p}}_s\}_{s=1}^S$, anchor probability vector $\mathbf{p}^{\texttt{anch}}(\mathbf{x})$

Output: hyperparameters $\{\alpha_s\}_{s=1}^S$

1: fix a grid of values, e.g., $\mathcal{A} = \{0.0, 0.1, \ldots, 1.0\}$
2: **for** $s = 1, \ldots, S$ **do**
3: **for** $\alpha \in \mathcal{A}$, $i = 1, \ldots, N_s$ **do**
4: find $\widehat{\mathbf{p}}_{s,i}^{\texttt{OOB}}(\mathbf{x}_{s,i})$ {OOB histogram}
5: $\mathbf{q}_{s,i,\alpha}^{\texttt{OOB}} = \arg\min_{\mathbf{q}} \alpha W(\mathbf{q}, \widehat{\mathbf{p}}_{s,i,k}^{\texttt{OOB}}(\mathbf{x}_{s,i})) + (1-\alpha) W(\mathbf{q}, \mathbf{p}^{\texttt{anch}}(\mathbf{x}_{s,i}))$ {barycentric interpolation}
6: **end for**
7: find $\alpha_s^* = \arg\min_{\alpha \in \mathcal{A}} \frac{1}{N_s} \sum_{i=1}^{N_s} D(\mathbf{q}_{s,i,\alpha}^{\texttt{OOB}}, \mathbf{y}_{s,i})$ {minimizes the OOB prescriptive cost}
8: **end for**

return $\{\alpha_s^*\}_{s=1}^S$

In this section, we propose a decision-focused algorithm to pool data from a collection of S problems with contextual information. Assume access to local data sets $\mathcal{D}_s$ and models $\widehat{\mathbf{p}}_s$, as well as an anchor distribution $\mathbf{p}^{\texttt{anch}}(\mathbf{x})$ estimated from a data pooling procedure, e.g., the output of a global model with naive data pooling or aggregation of

$\widehat{\mathbf{p}}_s$ of the form of (14). Note that it is also possible to consider distributions that are not data-driven, e.g., a distribution provided by a domain expert given the context. Our goal is to determine when and how much data to pool in order to minimize the expected out-of-sample decision cost. Effectively, this can be viewed as a problem of decision-focused forecast combination. To achieve this, we utilize OT and the Wasserstein barycenter once again to interpolate between a local and an anchor distribution, allowing for a flexible combination of information from both of them.

The decision-focused interpolation algorithm is detailed in Algorithm 1. The algorithm begins by fixing a grid of values for hyperparameter $\alpha \in [0, 1]$, which controls the amount of data pooling. For each subproblem s, the algorithm iterates over the values of α and training observations $i = 1, \ldots, N_s$, and estimates a conditional distribution using the decision-focused OOB method. The algorithm then interpolates between the OOB and anchor distributions by estimating a barycenter whose coordinates are determined by α. For clarity, $\widehat{\mathbf{p}}^{\texttt{OOB}}_{s,i}(\mathbf{x}_{s,i})$ is the OOB probability vector given $\mathbf{x} = \mathbf{x}_{s,i}$, estimated from a subset of base learners from the ensemble model $\widehat{\mathbf{p}}_s$ which did not use the i-th observation for training (hence the superscript $\texttt{OOB}$). Further, $\mathbf{q}^{\texttt{OOB}}_{s,i,\alpha}$ is the α-weighted average distribution, in the sense of the Wasserstein distance, between $\widehat{\mathbf{p}}^{\texttt{OOB}}_{s,i}(\mathbf{x}_{s,i})$ and $\mathbf{p}^{\texttt{anchor}}(\mathbf{x}_{s,i})$. Evidently, $\alpha = 1$ retrieves the local solution, while $\alpha = 0$ maximizes the amount of data pooling. Finally, the algorithm finds the value of α that minimizes the OOB decision cost for the training data set.

For an out-of-sample realization of uncertainty, $\mathbf{x}_{s,0}$, we first estimate the α-weighted Wasserstein barycenter of the local and the anchor models and then solve the respective problem. A different hyperparameter α is selected for each problem. This way, problems with high-quality local data sets and, by extension, high-quality forecasting models will converge to the local approach faster, while the rest of the problems may still benefit from data pooling.

Note that we can, alternatively, interpolate between the local and the anchor distribution by minimizing the ℓ_2 distance, instead of the Wasserstein distance, by replacing

Step 5 of Algorithm 1 with

$$\mathbf{q}^{\texttt{OOB}}_{s,i,\alpha} = \alpha \widehat{\mathbf{p}}^{\texttt{OOB}}_{s,i,k}(\mathbf{x}_{s,i}) + (1-\alpha)\mathbf{p}^{\texttt{anch}}(\mathbf{x}_{s,i}),$$

effectively creating a convex combination between the local and the anchor distribution. Nonetheless, the resulting mixture of distributions does not maintain the geometric structure and is less interpretable.

6. Numerical Experiments

In this section, we empirically validate the proposed data pooling methods on a motivating application related to the integration of stochastic renewable energy sources in power systems. We describe the problem (in Section 6.1), discuss our experimental setup and input data (in Section 6.2), and present the numerical results (in Sections 6.3).

6.1. Trading Renewable Energy in Electricity Markets

We consider a set of renewable producers, namely wind power producers, participating as price-takers in a day-ahead electricity market subject to imbalance penalties, assuming a dual-price balancing mechanism (Stratigakos et al., 2022). Prior to market closure and for each market clearing period, producers submit an energy offer based on contextual information regarding future production, e.g., weather conditions. During real-time operation, the system operator activates balancing reserves to ensure a demand-supply equilibrium and proper operation. If the system length is positive, i.e., supply is greater than demand, downward regulation reserves are activated, while upward regulation reserves are activated if the system length is negative. The procurement cost of balancing reserves is retrieved ex-post; namely, producers whose real-time production deviated from the contracted offer in the same direction as the system length are subject to financial penalties.

The problem of minimizing a producer's trading cost under production and price uncertainty can be formulated as a Bernoulli newsvendor problem (Pinson, 2023). Let y be the uncertain renewable production and $h \sim \text{Bern}(\tau)$ be a Bernoulli random variable

that models the system length with τ being the probability of success. Typically, y, h are assumed to be independent. If τ is known, then the optimal solution of the Bernoulli newsvendor problem is the τ-th quantile of the predictive density of y (Pinson, 2023, Proposition 1); conversely, a fully robust solution when τ is unknown is offering the expected value of y (Pinson, 2023, Corollary 2). Here, we propose a hybrid strategy that interpolates between these two extremes by minimizing

$$c(z; y) = (1 - r) \max \big(\frac{\tau}{1 - \tau} (y - z), (z - y) \big) + r(y - z)^2, \tag{17}$$

where z is the energy offer that takes values in the feasible set $\mathcal{Z} = \{z \,|\, 0 \leq z \leq 1\}$, and r is a user-defined parameter that controls the degree of risk-aversion against price uncertainty. Evidently, $r = 0$ retrieves a standard newsvendor loss, while for $r = 1$ we minimize the mean squared error.

6.2. Experimental Setup and Input Data

In the numerical experiments, we compare the following methods:

- `Local`: a standard approach where each subproblem is solved independently without any data pooling.
- `Pool-Naive`: a global model with naive data pooling.
- `Pool-OT`: model aggregation with the Wasserstein barycenter.
- `Interp`: barycentric interpolation between `Local` and `Pool-OT` using the proposed decision-focused data pooling algorithm.

To estimate the conditional distribution of the production, for each problem, we train a random forest model with 100 trees and default hyperparameters. For `Pool-Naive`, we consider the same model and hyperparameters as `Local` but trained on the concatenated data sets. For `Pool-OT`, we use the 1-Wasserstein metric to compute the barycenters, with the barycentric coordinates set at $\lambda_s = \frac{N_s}{\sum_{s=1}^{S} N_s}$. We also considered a modified cross-validation scheme to tune the barycentric coordinates by evaluating the in-sample performance of each model for the subset of problems that did not use its local data

Table 1: Average percentage (%) of task loss improvement over `Local` for $\tau = 0.20$, $r = 0.50$. Parentheses show the standard error.

	`Pool-Naive`	`Pool-OT`	`Interp`
$S = 5$	1.84 (2.03)	2.83 (2.04)	4.09 (1.64)
$S = 10$	2.89 (2.97)	3.86 (2.80)	5.92 (2.15)
$S = 20$	3.26 (0.70)	3.47 (0.64)	5.33 (0.54)
$S = 50$	3.59 (0.70)	4.23 (0.65)	5.52 (0.55)

for training. However, we did not observe significant differences in performance, hence these results are omitted.

For input data, we use power measurements from $S = 50$ wind turbines located in mid-west France, with a total nominal capacity of 100 MW. The available data sets span the period from January 2019 to April 2020 with an hourly resolution. Wind production data are normalized and assumed to take values on the fixed grid $\{0.00, 0.01, \ldots, 0.99, 1.00\}$. We use the data from 2019 to sample training data sets and the remaining 5 months for testing.

For contextual information, we use wind speed and wind direction forecasts from a Numerical Weather Predictions (NWP) model. The NWP model forecasts are issued daily at 00:00 UTC with a spatial resolution of $0.1^{\text{o}} \times 0.1^{\text{o}}$ and a forecast horizon of 96 hours ahead. This setting complies with the requirements of participating in day-ahead electricity markets, where the offers are typically submitted 12 to 36 hours ahead of the actual operation period. For the s-th subproblem, $\mathbf{x}_s$ comprises the NWP model forecasts from the closest grid point in terms of Euclidean distance.

6.3. Results

We examined performance for different values of τ and various degrees of risk aversion r. As results were similar, we focus our discussion on the case of $\tau = 0.20$, i.e., the optimal risk-neutral offer equals the 20-th quantile of the wind production distribution, and risk parameter $r = 0.50$. Additional results are provided as supplementary material.

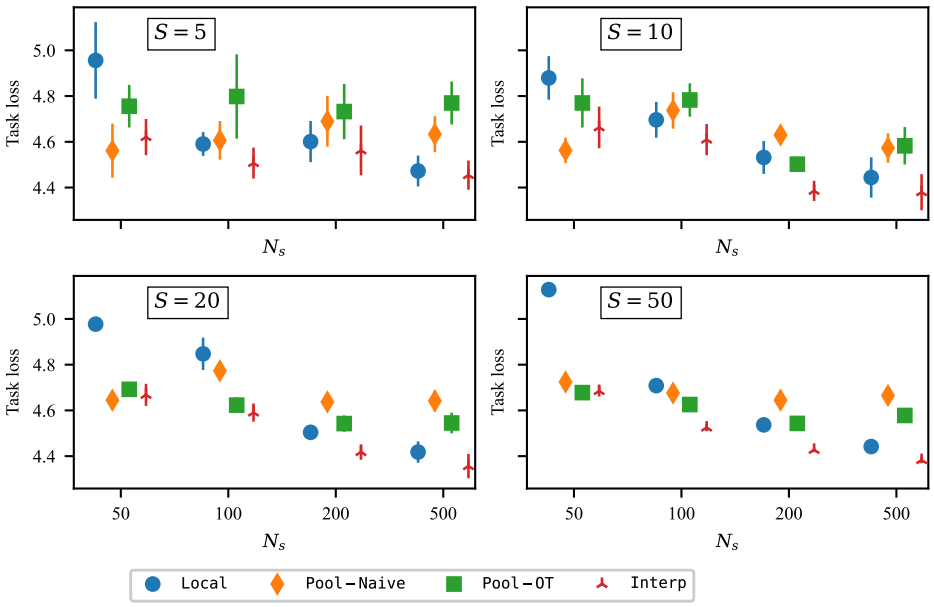

Figure 1: Average task loss for $\tau = 0.20$, $r = 0.50$ versus sample size N_s (same for all subproblems). Error bars show ± 1 standard error.

First, we consider a scenario where the number of training observations N_s is fixed across all problems and investigate the performance of the different methods as a function of N_s, as well as the number of problems (i.e., number of wind turbines) S. To obtain our results, we first sample S wind turbines and N_s training observations for each turbine, train both the local and global models, estimate the Wasserstein barycenters for each $\mathbf{x}$, and run Algorithm 1 for the interpolation method. We then evaluate the performance of each method on the test set. The process is repeated 10 times.

Fig. 1 presents the average task loss over the S subproblems and all the iterations. Overall, the results suggest that data pooling is beneficial when data are scarce, but as the amount of data increases the decisions derived from the local approach, `Local`, become more reliable and the benefits of data pooling are less pronounced. This result is intuitive and corroborates the findings of previous works — see, e.g., the results by Gupta and Kallus (2022). The relative improvement over `Local` is also more pronounced when N_s is small and S becomes larger—see, e.g., the bottom right plot of Fig. 1

for $N_s = 50$. Examining the two data pooling approaches shows that `Pool-Naive` outperforms `Pool-OT` when both the number of observations N_s and the number of turbines S is small—see, e.g., the top plots of Fig. 1 for $N_s = 50$. As S increases, `Pool-OT` improves considerably and for $S \geq 20$ converges to better performance than `Pool-Naive`. Increasing S ameliorates the instability in the barycenter estimation, which helps explain the improved performance of `Pool-OT`.

Importantly, the decision-focused data pooling, `Interp`, performs consistently well and outperforms both `Local` and `Pool-OT` in all cases. When N_s is moderate to small, `Interp` is considerably better than `Local`, while when N_s is larger, `Interp` converges to similar or better performance than `Local`. This result indicates that the decision-focused data pooling algorithm does a very good job of identifying when and how much data to pool in order to minimize the downstream cost, and that a small degree of data pooling offers benefits even for larger training samples.

Next, we repeat the previous experiment but randomly sample the number of training observations, N_s, for each subproblem from a uniform distribution over the interval $[10, 200]$. Table 1 summarizes the expected improvement in terms of decision cost and the standard error of each method. All methods lead to improved performance compared to `Local`, with `Pool-Naive` and `Pool-OT` leading to an expected improvement of 2.90% and 3.60%, respectively. For both approaches, the improvement is within statistical error for smaller values of S and becomes greater as S increases, corroborating the previous results. `Interp` ranks again as the best-performing method with an average improvement of 5.22%, with results being significant for all values of S. This further highlights the benefits of decision-focused interpolation, as it performs consistently well even when the local sample size and, by extension, model quality varies.

7. Conclusions

In this work, we investigated data pooling methods to address data scarcity when dealing with multiple contextually-dependent problems. Two approaches were proposed, namely training a global model with naive data pooling and an OT-based

method for combining estimated conditional distributions. We further developed a decision-focused data pooling algorithm that interpolates between a local and an anchor distribution based on an estimation of the expected downstream decision cost. For validation, we examined a pivotal application related to the integration of weather-dependent renewable energy sources in power systems, namely trading in a day-ahead electricity market. Our empirical results illustrated that data pooling improves overall performance when data are scarce and, perhaps more importantly, our decision-focused data pooling algorithm correctly identifies when and how much data to pool, leading to consistently better performance than standalone and pooled methods. Future work could focus on the case of both scarce and contaminated data, and developing data pooling methods that are robust to local outliers.

Acknowledgements

The work of A. Stratigakos and G. Kariniotakis was supported in part by the Smart4RES Project (Grant No 864337) funded under the Horizon 2020 Framework Program. The work of J. M. Morales and S. Pineda was supported in part by the European Research Council (ERC) funded under the Horizon 2020 Framework Program (Grant No 755705) and in part by the Spanish Ministry of Science and Innovation (AEI/10.13039/501100011033) through project PID2020-115460GB-I00.

References

Agueh, M., Carlier, G., 2011. Barycenters in the wasserstein space. SIAM Journal on Mathematical Analysis 43, 904–924.

Angelopoulos, A.N., Bates, S., 2022. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. `arXiv:2107.07511`.

Baardman, L., Cristian, R., Perakis, G., Singhvi, D., Skali Lami, O., Thayaparan, L., 2022. The role of optimization in some recent advances in data-driven decision-making. Mathematical Programming , 1–35.

Balint, A., Raja, H., Driesen, J., Kazmi, H., 2023. Using domain-augmented federated learning to model thermostatically controlled loads. IEEE Transactions on Smart Grid , 1–1doi:10.1109/TSG.2023.3243467.

Ban, G.Y., Rudin, C., 2019. The big data newsvendor: Practical insights from machine learning. Operations Research 67, 90–108.

Bertsimas, D., Kallus, N., 2020. From predictive to prescriptive analytics. Management Science 66, 1025–1044.

Bottieau, J., De Grève, Z., Piraux, T., Dubois, A., Vallée, F., Toubeau, J.F., 2022. A cross-learning approach for cold-start forecasting of residential photovoltaic generation. Electric Power Systems Research 212, 108415.

Breiman, L., 1996. Out-of-bag estimation. Technical report, Statistics Department, University of California Berkeley. https://www.stat.berkeley.edu/pub/users/breiman/OOBestimation.pdf.

Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

Corredera, A., Ruiz, C., 2023. Prescriptive selection of machine learning hyperparameters with applications in power markets: Retailer's optimal trading. European Journal of Operational Research 306, 370–388.

Cuturi, M., 2013. Sinkhorn distances: Lightspeed computation of optimal transport, in: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc.

Cuturi, M., Peyré, G., 2016. A smoothed dual approach for variational wasserstein problems. SIAM Journal on Imaging Sciences 9, 320–343.

Donti, P.L., Amos, B., Kolter, J.Z., 2017. Task-based end-to-end model learning in stochastic optimization, in: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 5490–5500.

Elmachtoub, A., Liang, J.C.N., McNellis, R., 2020. Decision trees for decision-making under the predict-then-optimize framework, in: International Conference on Machine Learning, pp. 2858–2867.

Elmachtoub, A.N., Grigas, P., 2022. Smart "predict, then optimize". Management Science 68, 9–26.

Esteban-Pérez, A., Morales, J.M., 2022. Distributionally robust stochastic programs with side information based on trimmings. Mathematical Programming 195, 1069–1105.

Feyeux, N., Vidard, A., Nodet, M., 2018. Optimal transport for variational data assimilation. Nonlinear Processes in Geophysics 25, 55–66.

Grabner, M., Wang, Y., Wen, Q., Blažič, B., Štruc, V., 2022. A global modeling approach for load forecasting in distribution networks. arXiv preprint arXiv:2204.00493 .

Gupta, V., Kallus, N., 2022. Data pooling in stochastic optimization. Management Science 68, 1595–1615.

Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H., 2009. The elements of statistical learning: data mining, inference, and prediction. volume 2. Springer.

Kallus, N., Mao, X., 2022. Stochastic optimization forests. Management Science .

Kazmi, H., Munné-Collado, Í., Mehmood, F., Syed, T.A., Driesen, J., 2021. Towards data-driven energy communities: A review of open-source datasets, models and tools. Renewable and Sustainable Energy Reviews 148, 111290.

Lichtendahl Jr, K.C., Grushka-Cockayne, Y., Winkler, R.L., 2013. Is it better to average probabilities or quantiles? Management Science 59, 1594–1611.

Montero-Manso, P., Hyndman, R.J., 2021. Principles and algorithms for forecasting groups of time series: Locality and globality. International Journal of Forecasting

37, 1632–1653. URL: https://www.sciencedirect.com/science/article/pii/S0169207021000558, doi:https://doi.org/10.1016/j.ijforecast.2021.03.004.

Muñoz, M., Pineda, S., Morales, J., 2022. A bilevel framework for decision-making under uncertainty with contextual information. Omega 108, 102575. URL: https://www.sciencedirect.com/science/article/pii/S0305048321001845, doi:https://doi.org/10.1016/j.omega.2021.102575.

Papayiannis, G., Galanis, G., Yannacopoulos, A., 2018. Model aggregation using optimal transport and applications in wind speed forecasting. Environmetrics 29, e2531.

Papayiannis, G., Yannacopoulos, A., 2015. A learning algorithm based on experts' opinions. Available at SSRN 2605905 .

Papayiannis, G.I., Yannacopoulos, A.N., 2018. A learning algorithm for source aggregation. Mathematical Methods in the Applied Sciences 41, 1033–1039.

Peyré, G., Cuturi, M., 2019. Computational optimal transport. Foundations and Trends in Machine Learning 11, 355–607.

Pinson, P., 2023. Distributionally robust trading strategies for renewable energy producers. IEEE Transactions on Energy Markets, Policy and Regulation 1, 37–47.

Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T., 2020. Deepar: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting 36, 1181–1191.

Stratigakos, A., Camal, S., Michiorri, A., Kariniotakis, G., 2022. Prescriptive trees for integrated forecasting and optimization applied in trading of renewable energy. IEEE Transactions on Power Systems 37, 4696–4708. doi:10.1109/TPWRS.2022.3152667.

Trapero, J.R., Cardós, M., Kourentzes, N., 2019. Quantile forecast optimal combination to enhance safety stock estimation. International Journal of Forecasting 35, 239–250.