Manuscript

Decision-Driven Regularization

Harmonizing the Predictive and Prescriptive

Gar Goei Loke
Department of Analytics & Operations, NUS Business School, National University of Singapore, Singapore 119245, gargoei@nus.edu.sg

Qinshen Tang
Nanyang Business School, Nanyang Technological University, Singapore 639798, qinshen.tang@gmail.com

Yangge Xiao
Institute of Operations Research and Analytics, National University of Singapore, Singapore 117602, yangge_xiao@u.nus.edu

Joint prediction and optimization problems are common in many business applications ranging from customer relationship management and marketing to revenue and retail operations management. These problems involve a first-stage *learning* model, where outcomes are predicted from features, and a second-stage *decision* process, which selects the optimal decisions based on these outcomes. In practice, these two stages are conducted separately, but is sub-optimal. In this work, we propose a novel model that solves both parts as a whole, but is tractable under many circumstances. Specifically, we introduce the notion of a regularizer that measures the value of a predictive model in terms of the cost incurred in the decision process. We term this *decision-driven regularization*, and it is centred on the premise that the bias-variance trade-off in the learning problem is not transformed linearly by the subsequent decision problem. Additionally, we identify an ambiguity in the definition of the cost function and propose a form to deal with this. We prove that our model is consistent. We also examine special cases under which we draw links to existing models in the literature, propose hybrid models and are able to describe their effectiveness using our framework as a theoretical basis. In our numerical experiments, we illustrate the behaviour of our model, and its performance against other models in the literature.

Key words: joint prediction and optimization, regularization, machine learning, decision-making under uncertainty, robust optimization

1. Introduction

Making predictions lies at the core of Machine Learning. Nonetheless, this is rarely the end point of business decision-making; predictions are often utilized to estimate outcomes that guide subsequent decisions. Often, these decisions will encompass identifying areas or individuals for intervention, such as customer churn management, or allocating of resources across components, such as purchasing. When considered together, the problem falls within the realm of joint prediction and optimization problems. Such settings involve a two-stage process where some intermediate quantity, termed *outcomes*, is to be estimated from the data, and then an optimization performed, with or without constraints, that takes these outcomes as inputs. For example in the context of customer churn management, the decision-maker has to decide upon a subset of customers, whom are provided a retention benefit, in order to discourage churn. First, she obtains some data, comprising past churn behaviour of customers and possible features that predict them, and develops a model that estimates the likelihood of each customer churning. Subsequently, by assuming this model, an optimization is then performed in order to decide upon the customers to avail the incentive.

For a while, it has been assumed that as long as predictions were accurate, whatever decisions made upon them would be optimal. As such, the literature is proliferate with approaches that conduct the prediction and optimization components separately, in areas such as pricing (Ferreira et al. 2016, Perakis et al. 2018), assortment (Fisher and Vaidyanathan 2014), and facility location (Huang et al. 2019, Glaeser et al. 2019). Specifically, in these approaches, the prediction is done without taking into consideration the nature of the decisions to be made later. The decisions are also optimized independently of the learning process, in the sense that it assumes that the predictive model generated is the true model.

Recent studies, however, have shown that this could not be further from the truth. Even in situations, which seem the natural domain of making predictions, it can be unreasonable to use the predictions as if they represented the core business decisions. Returning to the churn management example, in the field experiment by Ascarza (2018), the author illustrates that if customers, who were identified to have the highest likelihoods of churn, were made targets for intervention, then it can result in even worse retention rates than had this prediction not been available to the decision-maker.

There are a few reasons for why this occurs. First, there is a clear causal relationship between the outcomes (churn probabilities) and the predictors (observed behaviours). However, this causality is often de-emphasized or rarely compatible with the learning approach (Athey et al. 2019). While we agree that there is such an effect in this case, addressing matters of causality is not the preoccupation of our paper. Second, there is a subtle difference between the decisions to be made and the act of forming the predictions. As Ascarza (2018) argues, the decision is in fact the confluence of two predictions, the likelihood of churn, and the likelihood of retention if offered the incentive. The latter involves some degree of allocation, a concept sitting squarely with the notion of optimization, across different customers. Hence omitting this fundamentally misses the point. Third, as Van Parys et al. (2020) explains, the calibration of bias-variance trade-off for the goal of maximizing prediction accuracy may be fundamentally incompatible with the optimization problem. Specifically, we posit that this is because any non-linearity in the optimization can potentially impart a certain bias to even unbiased estimators. In this paper, we focus on the second and third reasons, and how they form two of three key tenets in our approach.

The complexity involved in the interplay of predictions and decision-making has led to a growing call for deeper inspection into what lies between. In Bertsimas and Kallus (2020), the authors espouse that Machine Learning techniques do not readily extend to the setting of decision-making under uncertainty. This was echoed by den Hertog and Postek (2016), who reason that the process of decision-making would greatly benefit from better understanding of what connects the predictive and the prescriptive.

Before moving to discuss that, we pay homage to the limited examples in the literature that attempt to address joint prediction and optimization problems under specific business contexts. In place of the two-step process of first estimating a demand distribution then optimizing for the optimal order quantity, Ban and Rudin (2019) propose a pair of single-step machine learning algorithms motivated from the empirical risk minimization and kernel-weights optimization perspectives. Liyanage and Shanthikumar (2005) study a new parametric approach called operational statistics to simultaneously optimize the Newsvendor model and estimate the parameter of the underlying exponential demand distribution. In the area of choice models, Yan et al. (2019) capture customer diversification using a regularization function and are able to achieve a separable form, from which closed-form relationships were derived. Liu et al. (2020) propose a model that integrates the travel time predictors with the order assignment optimization and illustrated an efficient solution methodology via branch-and-price.

Our goal in this paper is not to delve into the specific structure of these business contexts, but to derive a general framework for performing joint prediction and optimization. We also differentiate our work from recent attempts to incorporate ideas in optimization into the machine learning process (such as Bertsimas and Dunn 2017, which deals with the setting of classification trees), and existing methods to incorporate data into robust optimization frameworks (*e.g.* Delage and Ye 2010, Mohajerin Esfahani and Kuhn 2018, Bertsimas et al. 2018, Wiesemann et al. 2014).

1.1. Frameworks for Joint Prediction and Optimization

The earliest attempt to perform joint learning and optimization is the stream known as Empirical Optimization (EO) (see *e.g.*, Vapnik 1992, Shapiro 2003, Nemirovski and Shapiro 2006), which is posed as a general two-stage problem where the weights are first solved, then the optimal decisions are decided. Two assumptions characterize this approach. First, the objective function is approximated by the outcomes in the dataset. Second, the optimal decision for each data point depends only on the predicted outcome for that data point.

Kao et al. (2009) reason that EO can potentially overfit when the data set is small, as the learning process is not captured explicitly in the objective function. Instead, they examine a linear regression framework, and propose, as the solution, a convex combination of the best estimate and the EO solution. They argue that this induces a bias-variance trade-off into the optimization objective that would pull it away from either extremes of learning or optimization focused. A second attempt was made by Elmachtoub and Grigas (2017) to address the shortcomings of EO. In this paper, the authors propose a model, which decomposes the empirical objective into its best estimate under the learning model and an approximation error term, which is minimized. Also, the work is notable for their illustration that separating the learning and optimization could arrive at sub-optimal solutions. This is echoed by the earlier mentioned Liyanage and Shanthikumar (2005) in the context of inventory control.

Unfortunately, extending these works to the general setting can encounter tractability issues. To make further progress, we need to appeal to the ability of Robust Optimization in retaining the complexity of problems. This is illustrated as a possibility when deep connections between the learning and optimization problems were illustrated. In Xu et al. (2010), the authors characterize the duality between robustness in the decision setting and regularization in the learning setting, for the specific example of lasso regression. They illustrate that the regularization term could be interpreted as an uncertainty set under which the distribution of the data was uncertain. This was generalized by Bertsimas and Copenhaver (2018), though the authors identify situations with a potential duality gap.

The first such model to exploit this duality, and the only one that we know to the best of our abilities, is Zhu et al. (2019). They propose a Joint Estimation and Robustness Optimization (JERO) model that seeks the optimal decision variables, under the assumption that the true prediction parameters lie within some neighbourhood of the estimated parameters, whose radius they attempt to shrink. They term this the *robustness* of the solution.

1.2. Contributions of this Paper

In our work, we focus on three tenets that underpin our model, which best reflect the decision environment we are considering. To the best of our knowledge, we are first to consider such a context.

(A) **Joint decisions**: Decision problems entail an element of allocation, *i.e.*, decisions are made across the predicted outcomes of all data points, rather than on individual data points.

(B) **Bias-variance alteration**: Non-linear decision problems can impart non-zero bias to bias-free estimates, hence, improving prediction accuracy alone can never be optimal.

(C) **Cost function ambiguity**: If information about decisions is used in the learning, the cost function involves outcomes that are yet to be learnt, hence it is necessarily ambiguous.

In the next Section, we justify these tenets. They, especially **(B)**, motivate the key notion that information regarding the decision must be incorporated in the learning process. This motivates our approach of a *decision-driven regularization*. Specifically, we consider the following learning problem for some non-negative $\lambda \geq 0$,

$$\underset{\boldsymbol{w}}{\operatorname{argmin}}\ L(\boldsymbol{w}) + \lambda R(\boldsymbol{w}), \tag{DDR}$$

where L is the loss function associated with the learning process, capturing the fidelity of the model to the data, and R is some regularization that describes the 'value' of making the choice of weights $\boldsymbol{w}$ on the eventual optimization objective.

In this regard, our paper makes important contributions to the domain of joint prediction and optimization problems. We list them down specifically as follows:

1. We describe and justify these three tenets. These concepts are not immediately apparent, for example, it is still widely accepted that optimality in the decision problem can be achieved through improving predictions.
2. We introduce the idea of a decision-driven regularization within a novel framework for solving a broad class of joint prediction and optimization problems. We also prove its consistency (Theorem 1) and illustrate that it has a robust interpretation in terms of wrong estimation from the data (Theorem 2).
3. We prove that our model is indeed general in the sense that two existing works in the literature, namely Elmachtoub and Grigas (2017) and Zhu et al. (2019), reduces to special cases of our model (Propositions 3 and 4). We also propose new hybrid models under our framework that we numerically illustrate outperform these models.
4. We present numerical simulation results that illustrate the effectiveness of our model in addressing the challenges presented by the tenets.

Our theoretical framework provides us footing to analyze joint prediction and optimization problems, and any model or methodology proposed to solve such problems. We posit that it also provides new inroads into specific Machine Learning problems, such as learning under structure.

Organization of Paper

After the Introduction, Section 2 is devoted to the description of the decision-driven regularization model. Section 3 illustrates numerically the behaviour that we described in our model (DDR). We wrap up with some comments in the Conclusion in Section 4. To facilitate easy reading, we have deferred all proofs to Appendix A.

2. Decision-Driven Regularization

Let $\mathcal{D} := \{(\boldsymbol{x}_n, z_n)\}_n$ be a dataset of N data-points of predictors $\boldsymbol{x} \in \mathbb{R}^p$ and their outcomes z. The decision-maker is keen on considering a class of explanatory models,

$$z = f(\boldsymbol{x}; \boldsymbol{w}) + \epsilon, \tag{1}$$

where the functional family f is known, but the weights $\boldsymbol{w} \in \mathcal{W} \subseteq \mathbb{R}^p$ are not, and are to be inferred from the data via the minimization of some loss function L:

$$\operatorname*{argmin}_{\boldsymbol{w}} L(\boldsymbol{w}). \tag{2}$$

Notationally, we let $\boldsymbol{X} := (\boldsymbol{x}_n)_n$ represent the collection of predictor data over all the data points. Similarly, we collect all the noisy observations into $\tilde{\boldsymbol{z}} := (\tilde{z}_n)_n$. We assume that the loss function L is convex in the weights $\boldsymbol{w}$. In practice, it could be any norm on the error of the predictions $\|\tilde{\boldsymbol{z}} - \boldsymbol{f}(\boldsymbol{X};\boldsymbol{w})\|_q$, $q \geq 1$, over the dataset (*e.g.* the mean squared error, when $q = 2$, and accordingly, this loss function becomes ordinary least squares (OLS)). Another common option is to consider the log-likelihood function, given by $L(\boldsymbol{w}) = -log\Big(\prod_n f_n(\boldsymbol{x}_n;\boldsymbol{w})\Big)$. It may also contain a regularization term, specific to this loss function, such as LASSO, $L(\boldsymbol{w}) = \|\tilde{\boldsymbol{z}} - \boldsymbol{f}(\boldsymbol{X};\boldsymbol{w})\|_2^2 + \theta\|\boldsymbol{w}\|_1$ or ridge regression, $L(\boldsymbol{w}) = \|\tilde{\boldsymbol{z}} - \boldsymbol{f}(\boldsymbol{X};\boldsymbol{w})\|_2^2 + \theta\|\boldsymbol{w}\|_2$. In short, we interpret L as some measure of the fidelity of $\boldsymbol{w}$ to the data.

We refer to this as the *learning* problem, where the goal is to determine the best weights $\tilde{\boldsymbol{w}}$ under the loss function. Notationally, all variables pertaining to the learning process are denoted with a tilde.

Definition 1 (Learning optimal weights). The weights $\tilde{\boldsymbol{w}}$ are *learning optimal* if they are a minimizer to (2); they do not incorporate any information about the subsequent decision problem.

Post-learning, in the *decision* problem, the predicted outcomes $\hat{\boldsymbol{z}} := (\hat{z}_n)_n$ are derived $\hat{z}_n := f(\boldsymbol{x}_n;\boldsymbol{w})$ as an approximation for the true outcomes $\check{\boldsymbol{z}}$. Had the decision-maker already known the true outcomes, which we denote by $\check{\boldsymbol{z}}$ (hereafter, all *true* outcomes of variables are accented with the check sign $\check{}$), then she would be able to make a decision $\boldsymbol{y} := (y_n)_n := \boldsymbol{y}(\check{\boldsymbol{z}})$, as a function of the outcomes. This may be subjected to constraints on $\boldsymbol{y} \in \mathcal{Y} \subseteq \mathbb{R}^N$, with the objective to minimize some cost function $C(\boldsymbol{y};\check{\boldsymbol{z}})$, *i.e.*,

$$\min_{\boldsymbol{y}\in\mathcal{Y}} C(\boldsymbol{y};\check{\boldsymbol{z}}). \tag{3}$$

Usually, this cost is accrued for each data points separately and instead can be written as $C(\boldsymbol{y};\check{\boldsymbol{z}}) := \sum_n \alpha_n c(y_n;\check{z}_n)$, under some scaling α_n for each data point. For brevity, let $\alpha_n \equiv 1$. Our notation is summarized in Table 1.

At this point, we take a few moments to explain why the decision problem is defined as a joint optimization over all data points $\boldsymbol{y}$, as opposed to individually:

$$\sum_n \min_{y_n} c(y_n;\check{z}_n). \tag{4}$$

The central thesis of this paper is that making predictions solely does not suffice in most practical situations. To take the example of churn management raised by Ascarza (2018), upon predicting

		Dataset notations
$\mathcal{D}$	:	The (training) dataset
n	:	Index used to denote each data point in the dataset
$\boldsymbol{X}$	:	Collection of all feature data $(\boldsymbol{x}_n)_n$ used as predictors
$\tilde{\boldsymbol{z}}$	:	Collection of all *observed* outcomes $(\tilde{z}_n)_n$ in the dataset
		Learning Process
$f(\boldsymbol{x};\boldsymbol{w})$	:	Functional family parametrized by weights $\boldsymbol{w}$ explaining how outcomes z arise from predictors $\boldsymbol{x}$
$L(\boldsymbol{w})$	:	Loss function correspond to the functional family f
$\tilde{\boldsymbol{w}}$	:	Learning optimal weights; minimizer of the loss function L
$\hat{\boldsymbol{z}}$	:	Collection of all *predicted* outcomes $(\hat{z}_n)_n$, implicitly depending on weights $\boldsymbol{w}$
		Decision Process
$\boldsymbol{y}$	:	Collection of all decisions $(y_n)_n$.
$C(\boldsymbol{y};\tilde{\boldsymbol{z}})$	:	Estimate of the true cost function by using observed outcomes $\tilde{\boldsymbol{z}}$, termed *empirical cost function*
$C(\boldsymbol{y};\hat{\boldsymbol{z}})$	:	Estimate of the true cost function by using predicted outcomes $\hat{\boldsymbol{z}}$, termed *estimated cost function*
$c(\cdot;\cdot)$	:	Cost function accrued to each data point

Table 1 List of Parameters and Variables

the churn probabilities of a customer $\check{z}_n$, from the customer's history and demographics $\boldsymbol{x}_n$, the decision-maker would need to decide upon which customers to provide a retention incentive y_n, represented by (3). Notice that this is different from the question of whether customer n ought to be provided a retention incentive, represented by (4). As Ascarza (2018) reasons, the latter quickly runs into the problem that the customers with the highest churn probabilities are often the wrong targets for intervention, because they are more likely to have already made up their minds to leave and hence are less likely to be persuaded by the retention incentives. In other words, their cost-to-retain is very high. This implies that the decision should involve an element of *allocation* – given a total budget, how should the retention benefits be allocated amongst the customers to be retained? Such a question not just fits more reasonably within the structure of (3), the feasibility space $\mathcal{Y}$, also allows for constraints, say on the budget, to be imposed across data points. We enshrine this understanding in the following tenet:

Tenet A (Joint decisions). Decision problems entail an element of allocation, *i.e.*, decisions are made across the predicted outcomes of all data points, rather than on individual data points.

As mentioned in the Introduction, it is often observed in the literature, that the learning and decision problems are handled separately. The reader might question why it is necessary to do joint prediction and optimization. For example, why would it not be possible to simply pursue ever more precise predictions, and in what forms would that jeopardize the optimization? In the following Illustration, we point out exactly why.

Illustration 1 *Suppose z_1 and z_2 are both identically distributed random variables with mean 0, simulating the fact that their true values were indeed 0, but there is some noise in the estimation. Suppose also that the decision problem involves minimizing costs $C(\boldsymbol{y};\boldsymbol{z}) = y_1\min\{z_1,\eta\} + y_2\min\{z_2,\eta\}$ subject to $y_1+y_2=1$ and $y_1,y_2\in[0,1]$. In this case, the constraints effectively require that we pick the smaller of either $\min\{z_1,\eta\}$ or $\min\{z_2,\eta\}$. Then,*

$$\begin{aligned}\mathbb{E}\left[\inf_{y_1,y_2} C(\boldsymbol{y};\boldsymbol{z})\right] &= \int_{z_1,z_2\geq\eta}\eta d\boldsymbol{z} + \int_{z_1\leq\min\{z_2,\eta\}} z_1 d\boldsymbol{z} + \int_{z_2\leq\min\{z_1,\eta\}} z_2 d\boldsymbol{z}\\ &< \int_{\eta\leq z_1\leq z_2} z_1 d\boldsymbol{z} + \int_{\eta\leq z_2\leq z_1} z_2 d\boldsymbol{z} + \int_{z_1\leq\min\{z_2,\eta\}} z_1 d\boldsymbol{z} + \int_{z_2\leq\min\{z_1,\eta\}} z_2 d\boldsymbol{z}\\ &= 0.5\cdot\mathbb{E}[z_1] + 0.5\cdot\mathbb{E}[z_2] = 0.\end{aligned}$$

The inequality is obtained by replacing η with z_1 and z_2, which we can assume to be chosen such that there is a set where $z_1,z_2>\eta$ with positive measure. The final equality comes about by combining the first and third integrals and the second and fourth integrals, and then using symmetry between z_1 and z_2.

Had we had the true values of z_1 and z_2, i.e.*, that they were both 0, then as long as $\eta>0$, $\inf_{y_1,y_2} C(\boldsymbol{y};\boldsymbol{z})$ should in fact be 0. Hence, the above computation illustrates that the definition of the cost function itself, has imparted a non-zero bias into the problem. One can actually do a similar computation on the variance to find that the inherent bias has shifted going from the z's to the cost function. In fact, our illustration here that the curvature of the objective function can skew the performance and robustness frontier, has been seen in other contexts (* e.g. *data-driven stochastic optimization, Gotoh et al. 2017).*

In summary, using zero-bias predictors does not guarantee that one's conduct of the optimization in the decision problem would be bias-free, and presumably optimal. Illustration 1 clearly identifies that the functional form of a non-linear cost function can change the properties of the bias and variance, regardless of whether the learning problem was unbiased or not. Hence, to control the bias in the objective function of the decision problem itself, we must involve in the learning problem, information regarding this functional form. We enshrine this in the following tenet:

Tenet B (Bias-variance alteration). Non-linear decision problems can impart non-zero bias to bias-free estimates, hence, improving prediction accuracy alone can never be optimal.

Before we move on, let us take this opportunity to clarify about the flow of events during the training and testing phase. In the training phase, the main goal is to decide upon the weights $\boldsymbol{w}$. The decision-maker is privy to information regarding the observed outcomes $\tilde{\boldsymbol{z}}$. The solved decisions $\boldsymbol{y}$ in the training phase, do not have any real interpretation. In the testing phase, the decision-maker does not observe $\check{\boldsymbol{z}}$, but will instead be able to estimate that from $\boldsymbol{w}$ and the testing dataset. The decisions $\boldsymbol{y}$ now have a real interpretation in the sense that they are the decisions to be implemented, and will be used in the computation of the actual costs for performance comparisons.

2.1. The Decision-Driven Regularizer

The way through which we incorporate information about the decision problem is via a *decision-driven regularization* R, on the learning problem:

$$\underset{\boldsymbol{w}}{\operatorname{argmin}}\, L(\boldsymbol{w}) + \lambda R(\boldsymbol{w}).$$

This is motivated by the interpretation of a regularization as an energy functional pushing away from the learning optimal solution $\tilde{\boldsymbol{w}}$, to one which has desirable properties. For example, in LASSO, the regularization penalizes non-zero values of the weights vector $\boldsymbol{w}$, hence resulting in sparse solutions (Meinshausen et al. 2006, Zhao and Yu 2006). In ridge regression, the regularization penalizes large components of $\boldsymbol{w}$, hence averting instability as a result of multi-collinearity (Guilkey and Murphy 1975). Both of these examples indicate that there is an inherent property of the problem that which defines a 'desirable' situation the regularization advocates, namely sparsity and reduction in amplitudes respectively. Nonetheless, given a problem, it can be debatable as to what this desirable quality is. As such, in this paper, we propose that the eventual decision problem dictates desirability, since the predicted outcomes are only an intermediary to the act of making the decisions. Formally, 'desirability' means possessing a low-cost solution to the decision problem. In other words, we seek to relate R to $\min\limits_{\boldsymbol{y}} C(\boldsymbol{y}; \check{\boldsymbol{z}})$ for true outcomes $\check{\boldsymbol{z}}$.

From Cost Function Ambiguity to Regularizer

At this point, we would like to be able to define the cost function $C(\boldsymbol{y}; \check{\boldsymbol{z}})$, in terms of the true costs $\check{\boldsymbol{z}}$. However, in reality, at the point of inference, *i.e.*, the application of the model to new data, the true costs will never be known, even if we do know the functional form for C. Only $\boldsymbol{X}$ is observed. Hence, this leads to the conundrum: How should cost function C be defined? Specifically, which value of $\boldsymbol{z}$ should be used in the definition of C? This is not a foreign concept in the wider Machine Learning literature, *e.g.* learning the reward function in Reinforcement Learning. In joint prediction and optimization, while it is not a new question (though only briefly described in Elmachtoub and Grigas 2017), the literature is *laissez faire* in its handling.

Broadly, there are two choices for defining the cost function. The first is to utilize the observed outcomes $\tilde{\boldsymbol{z}}$ in place of the true outcomes $\check{\boldsymbol{z}}$. This leads to the definition of the cost as $C(\boldsymbol{y}; \tilde{\boldsymbol{z}})$, which we term the *empirical cost.* The second option is to utilize the estimated or predicted outcomes $\hat{\boldsymbol{z}}$, which are a function of the weights $\boldsymbol{w}$, in place of the true outcomes. This leads to the definition of $C(\boldsymbol{y}; \hat{\boldsymbol{z}})$, which we term the *estimated cost.*

Notice that both approaches impart an error to the cost function. In using observed outcomes $\tilde{\boldsymbol{z}}$, the noise in the observation is also imparted to the cost function C. In using estimated outcomes $\hat{\boldsymbol{z}}$, the error in the estimation of the weights $\boldsymbol{w}$ is transferred onto the cost function C. Either way, this error is consequential. If C is non-linear in the argument of the outcomes $\boldsymbol{z}$, a non-zero bias will be introduced, as implied by Illustration 1. This has profound implications on the optimization performed on the decision problem. As such, we enshrine this in the following tenet:

Tenet C (Cost function ambiguity). If information about decisions is used in the learning, the cost function involves outcomes that are yet to be learnt, hence it is necessarily ambiguous.

To address this problem, we propose the following formulation for the decision problem: For a given $\boldsymbol{w}$, the decision-maker minimizes the estimated cost function, while keeping it close to the empirical cost function, to control estimation error.

$$\begin{aligned} &\inf_{\boldsymbol{y}\in\mathcal{Y}} C\Big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\Big) := \sum_n c\Big(y_n; f(\boldsymbol{x}_n;\boldsymbol{w})\Big) \\ &\text{s.t. } \gamma_1 \le C(\boldsymbol{y};\tilde{\boldsymbol{z}}) - C(\boldsymbol{y};\hat{\boldsymbol{z}}) \le \gamma_2, \\ \text{where } &\gamma_1 \le 0 \le \gamma_2. \end{aligned} \tag{5}$$

Proposition 1. *The formulation (5) has Langragian relaxation:*

$$\inf_{\boldsymbol{y}\in\mathcal{Y}} \sum_n \Big[\mu\cdot c(y_n;\tilde{z}_n) + (1-\mu)\cdot c(y_n; f(\boldsymbol{x}_i;\boldsymbol{w}))\Big].$$

Proposition 1 can be thought of as a combination of the empirical cost and estimated cost functions, if $\mu\in[0,1]$, which leans towards the estimated cost if μ is small and the empirical cost if μ is close to 1. Of course, μ need not be in this range due to the double-sided inequality in (5). However, any μ outside of $[0,1]$ runs the risk losing properties, in particular, convexity that is later discussed in Proposition 2.

This formulation allows us to define the valuation function below:

Definition 2 (Valuation Function). For a given *calibration parameter* μ, the valuation function, $v_\mu(\cdot):\mathbb{R}^p\to\mathbb{R}$, maps the weights $\boldsymbol{w}$ to the space of objective values in the decision problem, given by,

$$v_\mu(\boldsymbol{w}) := \inf_{\boldsymbol{y}\in\mathcal{Y}} \Big\{\mu\, C(\boldsymbol{y};\tilde{\boldsymbol{z}}) + (1-\mu)\, C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)\Big\}. \tag{6}$$

In other words, the valuation function is the lowest possible cost that can be attained with the choice of weights $\boldsymbol{w}$, where the empirical and estimated cost functions are sufficiently close.

Assumption 1. $c\big(y; f(\boldsymbol{x};\boldsymbol{w})\big)$ *is convex in* y *and concave in* $\boldsymbol{w}$ *for all* $\boldsymbol{x}$ *over its range and* $c(y;z)$ *is convex in* y *over the range of all possible outcomes* z.

PROPOSITION 2. *Under Assumption 1, if $\mu \in [0,1]$, then the valuation function is concave in $\boldsymbol{w}$.*

We take a moment to explain the suitability of Assumption 1 here. First, convexity is assumed in the argument $\boldsymbol{y}$. This is because $\boldsymbol{y}$ represents the decisions. Hence, convexity of $\boldsymbol{y}$ is consistent with the law of diminishing marginal returns, as in the literature. Next, concavity is assumed in the argument $\boldsymbol{w}$. This is because $\boldsymbol{w}$, which represents the coefficients pertaining to the factors of production, should benefit from economies of scale. Moreover, it is contained in the argument of the outcomes, which are assumed to be counteracting the decisions $\boldsymbol{y}$. Critically, note also that Proposition 2 makes no assumptions on the feasible space $\mathcal{Y}$. This means that $\mathcal{Y}$ could technically be the intersection of a polyhedra or the interior of a convex cone and the lattice grid, as is common of mixed integer formulations.

DEFINITION 3 (DECISION-DRIVEN REGULARIZER). We say that a regularizer $R(\cdot)$ is a *decision-driven regularizer* (DDR) for the joint learning and decision problem if and only if there exists some convex non-increasing function $r(\cdot)$ such that

$$R(\boldsymbol{w}) = r\big(v_\mu(\boldsymbol{w})\big). \tag{7}$$

Moreover, we call the weights $\boldsymbol{w}$ that are obtained from the optimization of (DDR), the *decision-driven regularized* weights, or as being *decision optimal* for R.

To state in full, the decision-driven regularization (DDR) model is defined for some $\lambda \geq 0$ and $\mu \leq 1$:

$$\operatorname*{argmin}_{\boldsymbol{w}} L(\boldsymbol{w}) + \lambda r\left(\inf_{\boldsymbol{y}\in\mathcal{Y}} \left\{\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}}) + (1-\mu)\, C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)\right\}\right). \tag{DDR}$$

When $\lambda = 0$, we recover the original learning problem without considering decisions.

To clarify, (DDR) involves the observed outcomes $\tilde{\boldsymbol{z}}$, though not seen by the decision-maker when new data is availed for inference, but is in fact privy to her in the training data for the purposes of learning the weights $\boldsymbol{w}$, Hence its usage is legitimate in the model. However, when faced with the decision task on a new dataset, the estimated costs, derived via the DDR weights, are used.

$$\inf_{\boldsymbol{y}\in\mathcal{Y}} C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big).$$

Consistency of DDR

We also prove that our model (DDR) is consistent, *i.e.* that the estimated weights under our model converges as the number of data samples increases. This type of result is classic in Machine Learning, and our proof traces the same strategy in the literature (cf. Knight and Fu 2000).

DEFINITION 4. Let $\Lambda^s(\boldsymbol{w})$ be the random function, defined on the random dataset of size s, $\mathcal{D}^s := \{(\boldsymbol{x}_n^s, \tilde{z}_n^s)\}_n$ of correctly specified but noisy observations $\tilde{z}_n^s = f(x_n^s; \check{\boldsymbol{w}}) + \epsilon_n^s$, given by,

$$\Lambda^s(\boldsymbol{w}) = L(\boldsymbol{w}) + \lambda r\left(\inf_{\boldsymbol{y}\in\mathcal{Y}} \left\{\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}}^s) + (1-\mu)\, C\big(\boldsymbol{y};\boldsymbol{f}(\boldsymbol{x}^s;\boldsymbol{w})\big)\right\}\right). \tag{8}$$

Define $\boldsymbol{w}^s$ as the random variable that is the minimizer of Λ^s.

Theorem 1. *Suppose that the following conditions hold:*

1. *(DDR) is consistent when $\lambda = 0$,*
2. *r is continuously differentiable with bounded first derivatives,*
3. *C obeys Assumption 1, and,*
4. *$\mu C(\boldsymbol{y}; \tilde{\boldsymbol{z}}^s) + (1-\mu)\, C\big(\boldsymbol{y}; \boldsymbol{f}(\boldsymbol{x}^s; \boldsymbol{w})\big)$ is bounded above over the feasibility set $\boldsymbol{y} \in \mathcal{Y}$, and $\forall \boldsymbol{w} \in \mathcal{K}$ for any compact set $\mathcal{K}$ in the domain of $\boldsymbol{w}$.*

Then (DDR) is consistent for $\mu \in [0, 1)$, that is, $\boldsymbol{w}^s$ converges as the number of samples tend to infinity, $s \to \infty$, in probability.

The assumptions of this Theorem are reasonable. The first simply states that the original learning problem under loss function L is consistent. In most cases, we will be using affine functions for r, hence the second assumption is also reasonable. The third simply mirrors the assumptions in Proposition 2, which were explained earlier. The last constraint assumes bounded costs over the feasibility set $\mathcal{Y}$. This assumption is equivalent to ensuring that the set of feasible decisions considered by the decision-maker would not lead to infinite costs, which is also reasonable.

Robust interpretation

The other perspective to understand (DDR) is via the paradigm of Xu et al. (2010), which states that the regularization on the learning problem is dual to a decision problem with a defined uncertainty set controlling the level of robustness to variations in data. In our case, we also see a similar result in the form of the following theorem.

Theorem 2 **(Robustness of DDR)**. *Assume that $\mathcal{Y}$ is convex, closed and compact, and that Assumption 1 holds. Given any $\lambda > 0$, there exists some $\rho > 0$ such that the minimizer achieved on weights $\boldsymbol{w}$ in*

$$\begin{aligned} \sup_{\boldsymbol{y} \in \mathcal{Y}} \inf_{\boldsymbol{w}} \; & r\Big(\mu C(\boldsymbol{y}; \tilde{\boldsymbol{z}}) + (1-\mu) C(\boldsymbol{y}; \hat{\boldsymbol{z}}(\boldsymbol{w}))\Big) \\ s.t. \; & L(\boldsymbol{w}) - L(\tilde{\boldsymbol{w}}) \leq \rho \end{aligned} \tag{9}$$

coincides with the solutions of DDR, when $\mu \in [0, 1)$ and $r(v) := -v$.

The Theorem can be interpreted as stating that the dual to (DDR) is a problem where the objective is to minimize the valuation function, subject to the weights $\boldsymbol{w}$ living in an uncertainty set that relates to the geometry of the loss function. In other words, our model seeks the best approximation for the cost function while being robust to potential mis-specification of both the weights and the loss function. In the numerical section later (Section 3.3), we shall see that once mis-specification is supplied to the predictive model, DDR preserves a good degree of its performance when compared against the oracle.

The other important thing to mention is that, in the literature, the construction of the dual uncertainty set (cis the regularization) does not often have an intuitive form (*e.g.*, see Gao et al. 2017). In our construction, the uncertainty sets and the regularizers are both very intuitive. The uncertainty set is the loss function; and the regularizer is the infimum of the estimate of the cost. To the best of our understanding, our work is the only one to identify a pair of regularizer and uncertainty set where both are intuitively derived from the problem setting.

2.2. Special Cases

In this section, we examine special cases when the calibration parameter μ takes specific values of $\mu = 0$ and $\mu = -1$. We also make a short comment about $\mu = 1$ at the end.

Case of $\mu = 0$. First, we examine the case when $\mu = 0$. Here, (DDR) reduces to the following:

$$\underset{\boldsymbol{w}}{\operatorname{argmin}}\ L(\boldsymbol{w}) + \lambda r\left(\inf_{\boldsymbol{y}\in\mathcal{Y}} C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)\right). \tag{10}$$

In this case, none of the information relating to the observed outcomes $\tilde{\boldsymbol{z}}$ is used in the definition of the costs. In the sense of (5), this is akin to having no regards to whether the estimated cost is close to the empirical cost. Without a mixture of the empirical costs in the valuation function to balance the empirical cost, if λ is large, the model is not prevented from pursuing the $\boldsymbol{w}$ which forces down the estimated costs without giving sufficient regard to the fidelity of the data. Nonetheless, for small λ, this model still returns reasonable solutions.

This model relates to the work on JERO by Zhu et al. (2019).

Proposition 3 **(JERO as a DDR)**. *Define the model (JERO) as follows, which aims to maximize the amount of robustness on the mis-estimation of the loss function within the limit of meeting some target τ on the estimated costs:*

$$\begin{aligned} \max_{\rho>0,\,\boldsymbol{y}\in\mathcal{Y}} \quad & \rho \\ \textit{s.t.} \quad & C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})) \leq \tau \qquad \forall \boldsymbol{w}\in\mathcal{U}(\rho) := \{\boldsymbol{w} : L(\boldsymbol{w}) - L(\tilde{\boldsymbol{w}}) \leq \rho\}. \end{aligned} \tag{JERO}$$

Assume that $\mathcal{Y}$ is convex, closed and compact, and that Assumption 1 holds. Then for all τ for which $\exists \boldsymbol{y}\in\mathcal{Y}$ such that the target is attained $C(\boldsymbol{y};\hat{\boldsymbol{z}}(\tilde{\boldsymbol{w}})) \leq \tau$, under learning optimal weights $\tilde{\boldsymbol{w}}$, there exists some $\lambda := \lambda(\tau) \geq 0$ for which the solutions of JERO and DDR coincide, when $\mu = 0$ and $r(v) := \tau - v$.

This Proposition points to the fact that the model proposed by Zhu et al. (2019) can be cast as a DDR, under the special case of $\mu = 0$ and r is chosen in a specific manner. More critically, our model allows us to utilize our framework to explain the conservativeness of JERO, which happens when τ is chosen loosely. This can be explained via the form of r – the negative coefficient on v_μ in R means that JERO minimizes its savings. In other words, the model avoids reaping additional savings, unless it can ensure more than proportionate fidelity to data. When τ is set with large slack, the model provides no incentive to improve its performance. Such a setting actually corresponds to an extremely large λ. In the numerical simulations later, we shall this one-to-one correspondence between τ and λ (Section 3.4).

Case of $\mu = -1$. When μ is selected as -1, our model (DDR) reduces to the following for some $\lambda \le 0$:

$$\operatorname*{argmin}_{\boldsymbol{w}} L(\boldsymbol{w}) + \lambda r \left(\inf_{\boldsymbol{y} \in \mathcal{Y}} \left\{ 2\, C\big(\boldsymbol{y}; \hat{\boldsymbol{z}}(\boldsymbol{w})\big) - C(\boldsymbol{y}; \tilde{\boldsymbol{z}}) \right\} \right). \tag{11}$$

One way to understand the motivation of such a model, is to notice that the expression within the infimum has the following decomposition into the estimated cost and its estimation error when compared against the empirical cost:

$$\underbrace{C(\boldsymbol{y}; \hat{\boldsymbol{z}}(\boldsymbol{w})) - C(\boldsymbol{y}; \tilde{\boldsymbol{z}})}_{\text{estimation error}} + \underbrace{C(\boldsymbol{y}; \hat{\boldsymbol{z}}(\boldsymbol{w}))}_{\text{best estimate}}. \tag{12}$$

In the language of model (5), this corresponds to only having the one-sided constraint $\gamma_1 \le C(\boldsymbol{y}; \tilde{\boldsymbol{z}}) - C(\boldsymbol{y}; \hat{\boldsymbol{z}})$. This means that we only permit estimated costs that are higher than the empirical costs by some bound. In such a case, one can imagine that if the noise in the empirical costs are very high, then the resultant weights will have the potential to be significantly departed from the real $\breve{\boldsymbol{w}}$. Nonetheless, as it is anchored on the empirical costs, there is the possibility that it would be able to handle mis-specification in the cost function. In exchange, the interior of the infimum is not guaranteed to be concave, and hence the full problem is not guaranteed to be convex. If C is chosen to be affine in $\boldsymbol{y}$, however, the problem will still turn out to be convex and hence results like Theorem 1 will continue to hold true.

Like the case when $\mu = 0$, this model is also loosely related to another model in the literature:

PROPOSITION 4 **(SPO+ as a DDR)**. *Define the model (SPO+) as follows:*

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & 2 \sum_n c(y_n^*; \hat{z}_n) + \sum_n \sup_{y_n} \{ c(y_n; \tilde{z}_n) - 2c(y_n; \hat{z}_n) \} \\ \textit{with} \quad & y_n^* = \arg\min_{y_n} c(y_n; \tilde{z}_n) \qquad \forall n \in \{1, \ldots, N\}. \end{aligned} \tag{SPO+}$$

Assume that there are no constraints across data points y_n, then the solution of SPO+ coincides with DDR for $\lambda = 1$, $\mu = -1$, $r(v) := -v$ and the loss function is chosen as $L(\boldsymbol{w}) = 2 \sum_n c(y_n^*; f(\boldsymbol{x}_n; \boldsymbol{w}))$, where $y_n^* = \arg\min_{y_n} c(y_n; z_n), \forall n \in \{1, \ldots, N\}$.

Notice that SPO+ (Elmachtoub and Grigas 2017) is a very specific example of DDR in this case, where not just the loss function is specified, but the actual Lagrange multiplier λ is also specified.

This does mean that there is significantly less flexibility in SPO+. In the numerical simulations later (Section 3.2), we shall see that this particular choice of loss function in SPO+ leads to a degree of over-fitting when noise is introduced into the observations $\tilde{\boldsymbol{z}}$, from which the very critical $\boldsymbol{y}^*$ in the loss function is computed from. Nonetheless, this choice of loss function turns out to be more general, in the sense that because $\boldsymbol{y}^*$ depends directly on the outcomes $\tilde{\boldsymbol{z}}$, the definition of the loss function itself captures information about $\tilde{\boldsymbol{z}}$, such as mis-specification in the assumed learner f, that cannot otherwise be inferred from the estimated outcomes $\hat{\boldsymbol{z}}$.

There is another subtlety in that the order of summation and supremum/infimum is swapped around in (SPO+) and (11). This is a consequence of Tenet (A) – in SPO+, this concept is fundamentally absent and SPO+ is only concerned about the prediction accuracy, under their loss function. As such, (11) can also be thought of as the extension of SPO+ under Tenet (A) where joint constraints across y_i are permitted. In this regard, we term the family of models defined by (11) as 'SPO+'-hybrid models.

Case when $\mu = 1$. Finally, before moving on, we make a quick comment about the limiting case $\mu = 1$. Notice that if $\mu = 1$, the regularization term does not involve the weights $\boldsymbol{w}$ in any form, hence the argmin obtained would coincide with $\arg\min_{\boldsymbol{w}} L(\boldsymbol{w})$, which is just $\tilde{\boldsymbol{w}}$. However, it is not clear, that given a fixed λ bounded away from 0, that as $\mu \nearrow 1$, it is necessary for $\boldsymbol{w}(\mu) \to \tilde{\boldsymbol{w}}$ uniformly. This is because the behaviour of μ is asymptotic at $\mu = 1$; beyond $\mu = 1$, there are no consistent ways for ensuring that Assumption 1 holds. Indeed, we see from the numerical simulations that as μ gets closer to 1, we do not recover $\tilde{\boldsymbol{w}}$, as long as λ does not also uniformly decrease to 0.

2.3. Solving the DDR Model

In this section we propose two ways to solve our model. This stems from having to calibrate the Lagrange multiplier λ as is conventional for Machine Learning problems. Either way, if both the loss function L and the cost function C are second-order cone representable, which is a modest requirement for a large class of loss functions, then (DDR) is also second-order cone representable.

Calibration of λ via cross-validation. The most natural manner is to perform cross-validation on λ. This can be done by segmenting out a portion of the data to function as a validation set, or to simply perform cross-validation on λ, *e.g.* under k-fold cross-validation. Additionally, we can first use the OLS solution to find the rough ratio between the loss function and the cost function, $\bar{\lambda}$. It is then possible to initiate the search for λ within a rough neighbourhood of $\bar{\lambda}$.

Notice that if $r(v) = -v$, then (DDR) reduces to

$$\arg\min_{\boldsymbol{w}} \; L(\boldsymbol{w}) + \lambda \sup_{\boldsymbol{y}\in\mathcal{Y}} \Big\{ -\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}}) - (1-\mu)\, C\big(\boldsymbol{y}; \boldsymbol{f}(\boldsymbol{x};\boldsymbol{w})\big) \Big\}. \tag{13}$$

We can compute the dual on the inner supremum, especially for the linear case where $C(\boldsymbol{y};\boldsymbol{z}) = \boldsymbol{y}^\top \boldsymbol{z}$, the constraint set $\mathcal{Y}$ is a polyhedra represented by $\{\boldsymbol{A}\boldsymbol{y} \le \boldsymbol{b}, \boldsymbol{y} \ge \boldsymbol{0}\}$ and f remains general:

Proposition 5. *In the case when $r(v) = -v$, C is bilinear in $\boldsymbol{y}$ and $\boldsymbol{z}$, and $\mathcal{Y}$ is a polyhedra, then (13) has the following reformulation:*

$$\begin{aligned} \min_{\boldsymbol{w},\boldsymbol{\beta}} \ & L(\boldsymbol{w}) + \lambda \boldsymbol{\beta}^\top \boldsymbol{b} \\ s.t. \ & \boldsymbol{A}^\top \boldsymbol{\beta} \geq -\mu \tilde{\boldsymbol{z}} - (1-\mu)\boldsymbol{f}(\boldsymbol{x};\boldsymbol{w}); \\ & \boldsymbol{\beta} \geq \boldsymbol{0}. \end{aligned}$$

Once again, if L and f are second-order cone representable, this model can be solved as a second-order conic program. We adopt this strategy when solving (DDR) in the numerical simulations of the next section.

Robustness reformulation. The dual form in Theorem 2 in the case where $r(v) = -v$, can also lead to a separate solution methodology, where the robust counterpart is taken over the loss function, as opposed to the cost function in Proposition 5 above. This can be useful if the robust counterpart of the loss function can be easily deduced.

The slight subtlety is that the optimal weights $\boldsymbol{w}$ are in the inner problem. As such, one needs to determine the optimal weights via the risk level ρ corresponding to λ and decisions $\boldsymbol{y} = \boldsymbol{y}^*$. Hence, the optimal weights are computed as follows:

$$\begin{aligned} \arg\max_{\boldsymbol{w}} \ & \mu\, C(\boldsymbol{y}^*; \tilde{\boldsymbol{z}}) + (1-\mu)\, C\big(\boldsymbol{y}^*; \hat{\boldsymbol{z}}(\boldsymbol{w})\big) \\ \text{s.t. } & \boldsymbol{w} \in \mathcal{U}(\rho). \end{aligned} \tag{14}$$

The reader is referred to Appendix B for more details on this computation methodology.

3. Numerical Illustration

Here, we illustrate the behaviour of DDR within an actual problem context. We consider a simplified version of the vehicle routing problem in Elmachtoub and Grigas (2017): The decision-maker is faced with a network of d routes, which she intends to choose from. Each route $j \in \{1, \dots, d\}$ incurs a cost of $z_j(\boldsymbol{x})$ to traverse, aggregated as $\boldsymbol{z} := (z_j)_j$, where $\boldsymbol{x} := (x_i)_i$, $i \in \{1, \dots, p\}$ is a p-dimensional feature vector of predictors, that is assumed to be varying. In this regard, there is no real routing involved, and the problem could be thought of as a knapsack problem with prediction. Apart from comparability against models in the literature, we choose this problem because it is fundamentally a prediction problem – the decision-maker simply selects the route with the least predicted cost. This guarantees that any predictive model is *optimal*, if there is no mis-specification on the data-generating model or endogeneity with decisions. Hence, we can investigate the impact of our decision-driven regularizer once we layer this problem context with specific decision structures.

At the onset, the decision-maker possesses a *training dataset*, $\mathcal{D} := \{(\boldsymbol{x}^n, \tilde{\boldsymbol{z}}^n)\}_{n\in\mathcal{N}}$, where $|\mathcal{N}| = N$, to perform the learning. When faced with a new *testing dataset* $\overline{\mathcal{D}} := \{(\boldsymbol{x}^n, \tilde{\boldsymbol{z}}^n)\}_{n\in\overline{\mathcal{N}}}$, where $|\overline{\mathcal{N}}| = \overline{N}$, the decision-maker seeks a decision vector $\boldsymbol{y}^n := (y^n_j)_j$ representing whether or not route j is picked for data point n. Her goal is to minimize the total cost in the following linear program:

$$
\begin{aligned}
\min\ & \sum_n \sum_{j=1}^d z^n_j y^n_j && \\
\text{s.t.}\ & \sum_{j=1}^d y^n_j = 1 && \forall n \in \mathcal{N}; \\
& y^n_j \geq 0 && \forall n \in \mathcal{N}, \forall j \in \{1, \ldots, d\}.
\end{aligned}
\tag{15}
$$

$\boldsymbol{y}^n$ does not need to be binary, as this is satisfied at the extreme points of our polyhedra.

3.1. Data and its Handling

We generate a synthetic dataset, as it facilitates the construction of different test cases and parameters to cross-compare the models, as follows. First, the learning model is generated. To ensure that comparisons between different methodologies are due to the model construct, we choose to have no inherent mis-specification in the learning process. As such, we use a linear regression model for learning $\boldsymbol{z}$ from $\boldsymbol{x}$, specifically, $\boldsymbol{z}(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{w}_0$, where $\boldsymbol{w} = (w_{i,j})_{i,j}$ is a $p \times d$-matrix of coefficients for predictor i and route j, and $\boldsymbol{w}_0 := (w_{0,j})_j$ is a vector of intercepts for each route j. We assume that the true model is indeed linear with $\boldsymbol{w}_0 = \boldsymbol{0}$. We denote the true $\boldsymbol{w}$ as $\breve{\boldsymbol{w}}$. This departs from Elmachtoub and Grigas (2017), where outcomes are assumed to be quadratic.

In each simulation, we generate $\breve{w}_{i,j}$, each independently, from a uniform distribution with non-negative support (specifically $[0, 3]$). Separately, we generate N and $\overline{N}$ samples of p-dimensional feature vectors $\boldsymbol{x}^n$ from a standard multivariate normal distribution to form the training $\mathcal{D}$ and testing $\overline{\mathcal{D}}$ datasets. From these, we obtain true cost vectors $\breve{\boldsymbol{z}}^n := \breve{\boldsymbol{z}}(\boldsymbol{x}^n) = \breve{\boldsymbol{w}}^\top \boldsymbol{x}^n$ and noisy estimates of the costs, $\tilde{\boldsymbol{z}}^n := \tilde{\boldsymbol{z}}(\boldsymbol{x}^n) = \breve{\boldsymbol{w}}^\top \boldsymbol{x}^n + \boldsymbol{\epsilon}^n$, with components of $\boldsymbol{\epsilon}^n$ sampled independently from a uniform distribution with support $[-\alpha, \alpha]$.

Solving and Testing

In solving for $\boldsymbol{w}$ and $\boldsymbol{w}_0$, only the features and noisy costs $\tilde{\boldsymbol{z}}^n$ are observed in the training dataset $\mathcal{D}$ and not the true costs $\breve{\boldsymbol{z}}^n$. If the model requires cross-validation, then the partitioning will be done on $\mathcal{D}$. We denote as $\hat{\boldsymbol{w}}$ and $\hat{\boldsymbol{w}}_0$ the estimates of the weights.

We evaluate the performance of models as follows. First, given $\tilde{\boldsymbol{w}}$ and $\hat{\boldsymbol{w}}_0$, model estimates for the costs are computed $\hat{\boldsymbol{z}}^n := \hat{\boldsymbol{z}}(\boldsymbol{x}^n) = \hat{\boldsymbol{w}}^\top \boldsymbol{x}^n + \hat{\boldsymbol{w}}_0$ for each testing data points, from which optimal decisions $\hat{\boldsymbol{y}}^n$ are solved via (15). The performance, in terms of the mean cost, is then computed under true costs:

$$P(\hat{\boldsymbol{w}}, \hat{\boldsymbol{w}}_0) := \sum_{n:\text{test}} \breve{\boldsymbol{z}}^{n\top} \hat{\boldsymbol{y}}^n \Big/ \overline{N}.$$

We also solve for the decision of the oracle, $\breve{\boldsymbol{y}}^n$, who having access to the true costs $\breve{\boldsymbol{z}}^n$ makes the best possible decisions by definition. The oracle's performance is denoted $P^* := \sum_{n:\text{test}} \breve{\boldsymbol{z}}^{n\top} \breve{\boldsymbol{y}}^n \Big/ \overline{N}$, representing a lower bound that can never be crossed. By and large, whenever comparing the performance of any two models $\left(\hat{\boldsymbol{w}}^A, \hat{\boldsymbol{w}}_0^A\right)$ and $\left(\hat{\boldsymbol{w}}^B, \hat{\boldsymbol{w}}_0^B\right)$, we reflect the following metrics:

1. Mean cost improvement: This reflects the difference in mean costs incurred by the two models over the test dataset, with the base taken as the mean cost incurred by the oracle, $\Delta P(A,B) := \left[P\left(\hat{\boldsymbol{w}}^A, \hat{\boldsymbol{w}}_0^A\right) - P\left(\hat{\boldsymbol{w}}^B, \hat{\boldsymbol{w}}_0^B\right)\right] \Big/ P^*$.
2. Discordance: This metric measures the proportion of the data points $\boldsymbol{x}^n$ in the test dataset where the optimal decisions $\hat{\boldsymbol{y}}^n$ disagree across the two models, $D(A,B) = \left[\sum_{n:\text{test}} \mathbb{1}\left\{\hat{\boldsymbol{y}}_A^n \neq \hat{\boldsymbol{y}}_B^n\right\}\right] \Big/ \overline{N}$.
3. Head-to-head: This metric measures the proportion of data points $\boldsymbol{x}^n$ for which the first model achieves a lower cost than the other, out of the base of all data points where the two models disagree, $H(A,B) = \dfrac{1}{D(A,B)} \left[\sum_{n:\text{test}} \mathbb{1}\left\{\breve{\boldsymbol{z}}^{n\top} \hat{\boldsymbol{y}}_A^n < \breve{\boldsymbol{z}}^{n\top} \hat{\boldsymbol{y}}_B^n\right\}\right]$. $H(A,B) = 0$ if $D(A,B) = 0$.

3.2. Comparisons against SPO+

In Proposition 4, we have seen that SPO+ is a specific case of DDR with $\mu = -1$, $\lambda = 1$ and where the loss function is twice the empirical cost, as solved by using the optimal pointwise decisions. Here, we examine the situation of using the same selection of μ and λ, but we change the loss function. In particular, we could consider the version where the mean squared error (MSE) is adopted as the loss function. We call this the OLS-SPO+ hybrid, specifically,

$$\underset{\boldsymbol{w}}{\operatorname{argmin}} \; ||\tilde{\boldsymbol{z}} - \hat{\boldsymbol{z}}||_2^2 + \lambda r \left(\inf_{\boldsymbol{y} \in \mathcal{Y}} \left\{ 2\, C\big(\boldsymbol{y}; \hat{\boldsymbol{z}}(\boldsymbol{w})\big) - C(\boldsymbol{y}; \tilde{\boldsymbol{z}}) \right\} \right). \tag{OLS-SPO+}$$

We illustrate the performance of the OLS-SPO+ hybrid against SPO+ over 100 simulations. By this, we mean that we set up a training and testing dataset, and compute the abovementioned metrics comparing the two models, for a total of 100 times. This allows us to see the distribution of these metrics over random realizations of the data. For each realization, we plot the head-to-head and mean cost improvement metrics in Figure 1 (left). For a particular set of parameters, we can see that, the OLS-SPO+ hybrid disagrees with SPO+ around 39.3% of the time on average. Amongst them, the OLS-SPO+ hybrid picks the better decision 68.0% of the time on average, leading to an average 16.8% cost reductions over SPO+. We simulate for other parameters and the trend holds, but for brevity, we only present one separate configuration of parameters in Figure 1 (right).

More peculiarly, our results here indicate that SPO+ could do with a boost by changing its loss function to the MSE, seemingly contradicting the conclusions in Elmachtoub and Grigas (2017), when they compare their model against OLS. There are simple explanations for this. First, as SPO+ utilizes in its loss function, optimal decisions that depend on noisy data, it is, like EO,

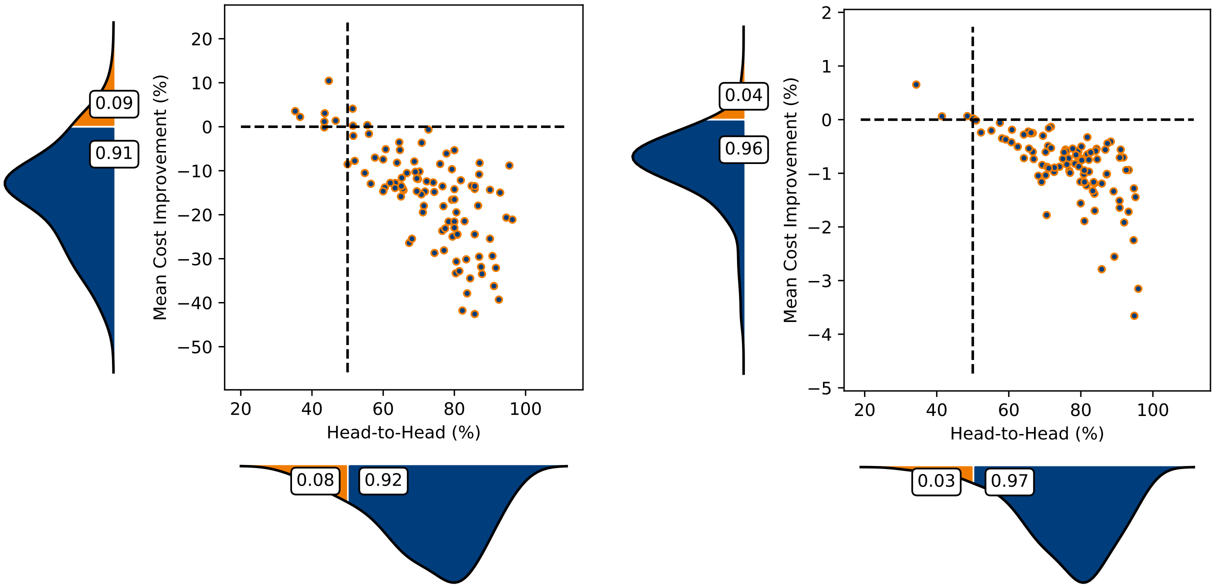

Figure 1 Mean cost improvement and head-to-head metrics of OLS-SPO+ hybrid against SPO+ for $d = 5$, $p = 4$, $N = 25$ and $\alpha = 3$ on $\overline{N} = 100$ test samples (left) and for $d = 3$, $p = 4$, $N = 100$ and $\alpha = 1$ on $\overline{N} = 5000$ test samples (right).

susceptible to overfitting under noise. Second, they assume mis-specification in the true model, in other words, OLS is mis-specified. While they reason that this illustrated SPO+'s robustness, we do note that SPO+'s loss function is more general than MSE. This is because the loss function is a function of the observed data $\tilde{\boldsymbol{z}}$ (which is quadratic and hence lying beyond the linear realm), hence the family of functions it can describe is larger than the family of linear functions in OLS.

To illustrate our point, suppose we allow the costs to be mis-specified. Specifically, let the truth be $\check{z}_j = (\check{\boldsymbol{w}}_j^\top \boldsymbol{x})^\beta$ and observations be $\tilde{z}_j^n = (\check{\boldsymbol{w}}_j^\top \boldsymbol{x}^n)^\beta + \epsilon_j^n$, but still compel the model to remain linear. Here, $\beta > 0$ represents the level of mis-specification. Figure 2 illustrates the results as β is varied. Indeed, when the amount of mis-specification is low, *i.e.*, when β is close to 1, OLS-SPO+ hybrid continues to out-perform SPO+. The differences are quickly eroded once the mis-specification increases, until eventually the performance of SPO+ outstrips its hybrid version. This justifies our claim that it is the generality of the loss function in SPO+ that dictates its performance under mis-specification. Nonetheless, our results indicate that when mis-specification is low, the proper loss function corresponding to the family of predictors should instead be used. For extremely general families, such as neural networks, where mis-specification in the functional form is not expected, it would be strongly advised to use the SPO+-hybrid versions (11).

3.3. Altering the Performance-Robustness Trade-off

Here, we study the behaviour of DDR in altering the bias-variance frontier of OLS through the decision problem, which is shown, in Illustration 1, to occur if the cost function was non-linear.

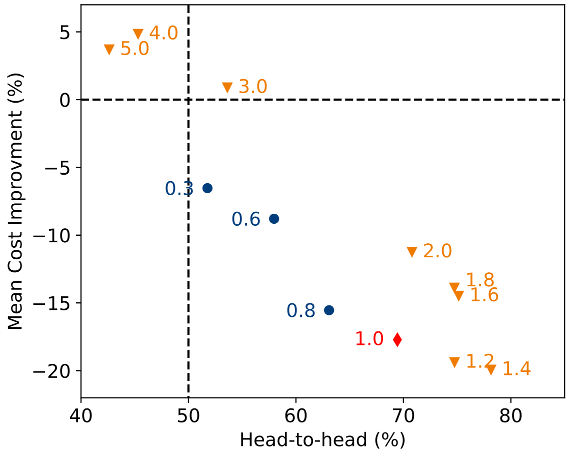

Figure 2 Mean cost improvement and head-to-head metrics of OLS-SPO+ hybrid against SPO+ under different levels of mis-specification β for $d=5$, $p=4$, $N=25$ and $\alpha=3$ on $\overline{N}=100$ test samples.

In this illustration, we assume the following form for the cost function, while keeping the learning problem, as well as the true underlying data-generating model, linear:

$$C(\boldsymbol{y};\boldsymbol{z}) = \sum_{n}\sum_{j=1}^{d} Q(z_j^n) y_j^n,$$

where Q is some concave function. Specifically, we consider the case of $Q(z) = \min\{z, \eta\}$ that represents a threshold at η, as in Illustration 1. Such may represent, for example, the pay rate, capped at a maximum after some number of hours, on travel times z_j^n.

OLS should still perform well here – thresholding should not hinder the selection $\boldsymbol{y}$ of the lowest cost. Nonetheless, DDR can perform better than OLS even under such circumstances, especially when the data is scant and the robustness of DDR to variations in data is able to kick in.

Calibration of λ and μ

Let us first illustrate how the out-of-sample performance, in terms of the mean cost metric, varies as μ and λ varies. As the cost function is linear in decisions $\boldsymbol{y}$, the cost function remains convex in $\boldsymbol{y}$ under negative values of μ, hence, we also illustrate the model under $\mu < 0$. We remind the reader that $\lambda = 0$ represents the OLS solution for any μ, while $\mu = 0$ represents JERO, where each λ corresponds to some target τ specified on the cost function. This is plotted in Figure 3, where the threshold of η is selected at -0.25, for some choice of parameters. First, note that under calibration of λ, DDR will outperform OLS. While at first glance, the improvement does not seem significant, the mean cost improvement of the oracle over OLS is 2.4%, hence DDR closes about a fifth of the gap to the oracle cost for selected choices of μ. The reader is reminded that this gap cannot ever be fully closed due to the irreducible error that arises in out-of-sample testing. Now, the shape of

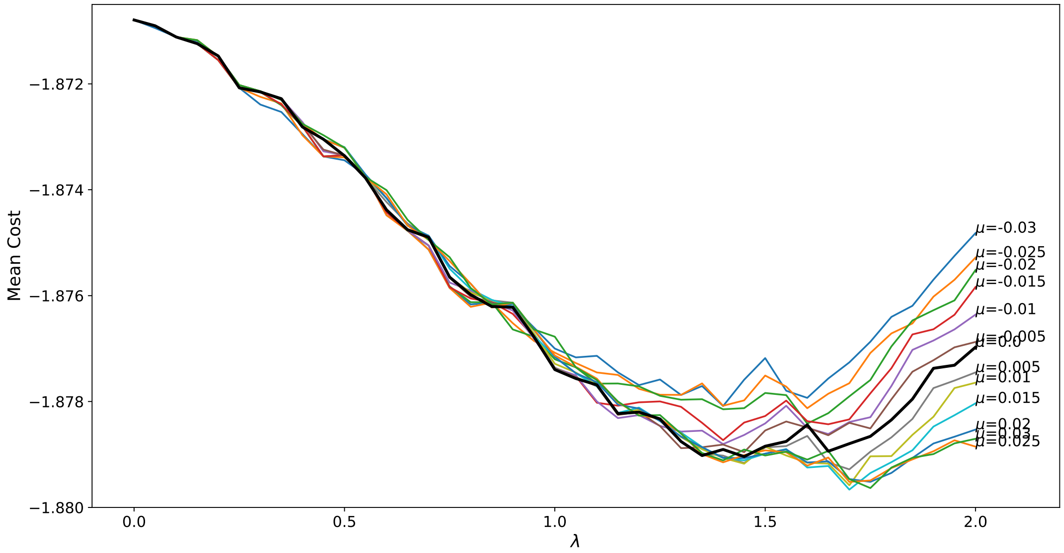

Figure 3 **Mean cost against different λ and μ for $d = 3$, $p = 4$, $N = 25$, $\alpha = 3$, $\eta = -0.25$, on $\overline{N} = 10000$ samples.**

the mean cost metric traces the usual convex structure as λ increases. Such a shape is consistent against performance plots of other regularizers.

The second observation to be made from Figure 3 is the behaviour of the curves as μ varies. Here, we made $\mu = 0$ bold, for reference. As μ increases, performance increases until around $\mu = 0.025$, at which point, the performance starts to dip again. In many of our experiments, we have seen that best performing μ tends to be around a small radius of $\mu = 0$. In the interpretation of (5), this means that we are not too strict on requiring the empirical costs to be close to the estimated costs, but nonetheless require that they are still bounded. This allows the overfitting due to the estimation of $\boldsymbol{w}$ to be controlled.

In Figure 4, we choose, for each μ, the best λ in Figure 3 and represent their head-to-head ratio against the mean cost improvement, as a proportion of the gap that DDR closes from OLS to the oracle. We do this for two choices of thresholds η. Notice that as the threshold is brought down, the head-to-head performance of DDR against OLS improves. This happens because a lower threshold also represents a greater distortion of the costs, which we had explained in Illustration 1, imparts a bias to OLS. One might question why the mean cost improvement does not move in the same direction as the head-to-head performance. It turns out, as the threshold is lowered, the gap between OLS and the oracle also closes, because a greater number of points have been thresholded and hence perfectly estimated. This reduces the error in the prediction on the overall. If we also factor this into consideration, as we see on the right-side chart in Figure 4, then the head-to-head ratio and the mean cost improvement move in the same direction.

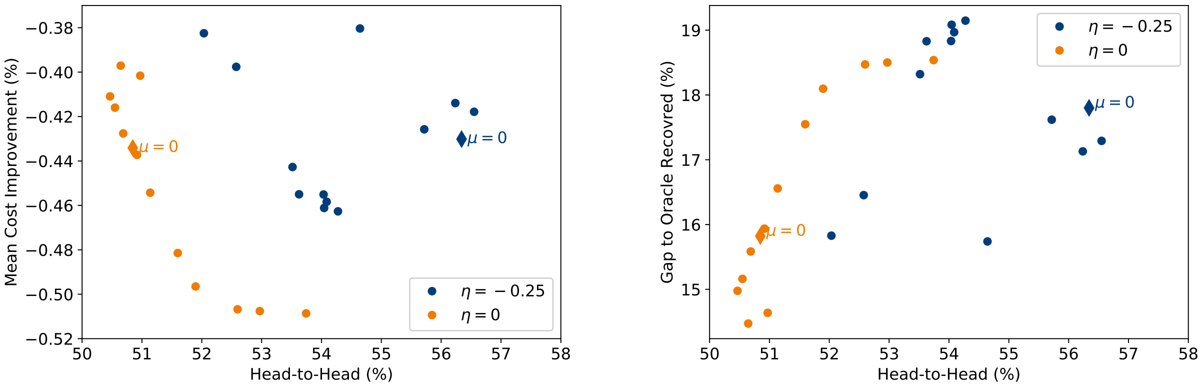

Figure 4 **Mean cost improvement (left), and as a proportion of gap to oracle (right) against the head-to-head ratio for two separate instances of thresholding for $d=3$, $p=4$, $N=25$ and $\alpha=3$ on $\overline{N}=10000$ test samples.**

Mis-specification

One possible question that could be asked at this point, is what if we had treated the outcomes as being the thresholded values $\boldsymbol{z}=\min\{f(\boldsymbol{x};\boldsymbol{w}),\eta\}$ instead, but allowed the costs to be bilinear, namely $C(\boldsymbol{y};\boldsymbol{z})=\boldsymbol{y}^\top\boldsymbol{z}$, in other words, treating the non-linear thresholding as a misspecification instead. Now, this is significantly different from before, as the noise in $\tilde{\boldsymbol{z}}$ is also thresholded prior to the learning, hence the noise is now no longer zero mean.

In Figure 5 below, we examine an analogous chart to Figure 3. In this case, we can see that the performance of DDR over OLS is much more significant. This is echoed also by the fact that the gap between OLS and the oracle have increased to 6.3% from 2.4% previously, while the gap between DDR, under best choice of λ, and the oracle remains roughly the same. This illustrates the effect of Theorem 2, which guarantees a level of robustness even if the estimation in the learning problem is done poorly. The stability of DDR to mis-specification also mirrors the results of SPO+, where we see that OLS performs badly under misspecification, however, in our case, the strong performance of our model vis-a-vis OLS has come about through our regularization that has a robust interpretation, and not the change in the loss function.

3.4. Comparison against JERO

The astute reader would have noticed in Figure 4, that the points for $\mu=0$ actually correspond to JERO. As such, Figure 4 already indicates that DDR permits many choices for μ that would outperform JERO in terms of mean cost improvement.

Here, we take a step further to examine the comparison against JERO more specifically. In particular, Proposition 3 points to JERO being a special choice of λ in the family of DDR models corresponding to $\mu=0$. In the following Figure 6, we zoom in to examine the performance of $\mu=0$ under different choices of λ. The target τ would correspond to some particular λ, with the best

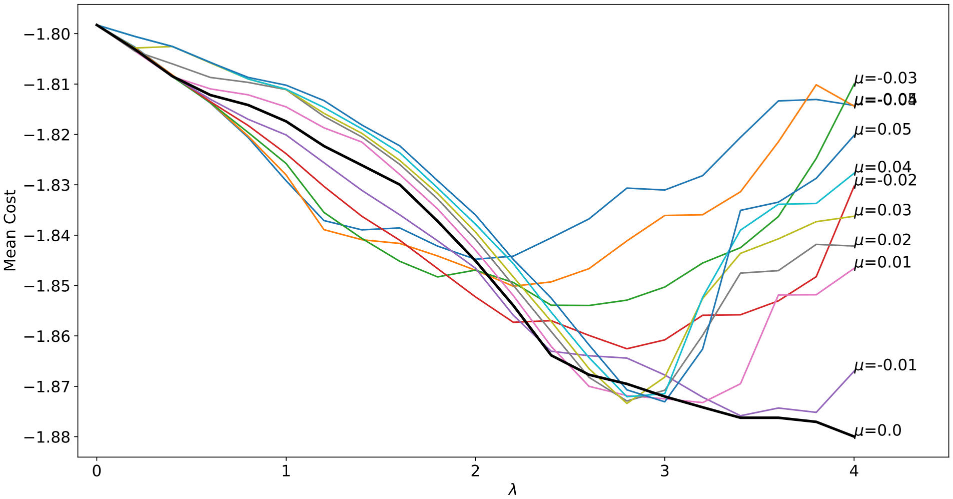

Figure 5 **Mean cost against different λ and μ for $d=3$, $p=4$, $N=25$, $\alpha=3$, $\eta=-0.25$, on $\overline{N}=10000$ samples.**

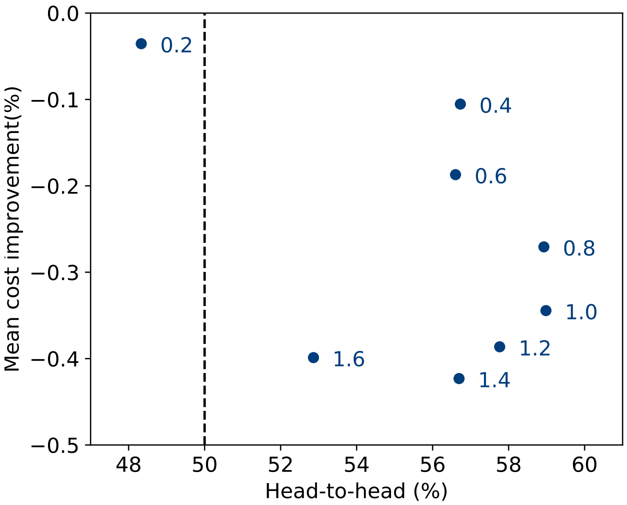

Figure 6 **Mean cost improvement and head-to-head metrics of DDR against JERO for $d=3$, $p=4$, $N=25$ and $\alpha=3$ on $\overline{N}=10000$ test samples.**

performing λ being around 1.4. Hence, if the target was set in such a way that does not correspond to $\lambda=1.4$, then JERO would be sub-optimal.

A closer inspection of the duality of Proposition 3 indicates a one-to-one correspondence between τ (when it is well-defined) and λ. Similarly, we can attempt a recovery of τ given λ in DDR. We do so in Figure 7. From Figure 6, the best value for λ occurs at 1.4. This corresponds to the target of around 89% of the OLS cost. The point we want to emphasize is that without a full calibration of the target in JERO, it would be difficult to motivate such a figure from the normative perspective, even if the target is in the units of the cost. In this regard, the motivation of JERO in the sense of trying to meet a pre-specified target, may not be the most suitable way to think about the joint prediction and optimization problem.

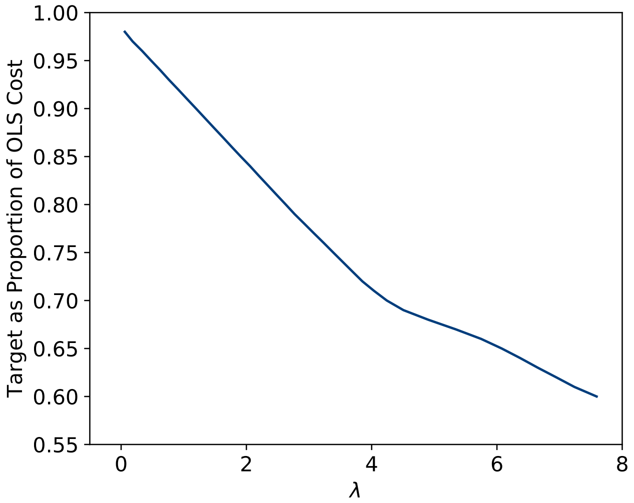

Figure 7 **Target as proportion of OLS cost, plotted against its corresponding λ**

4. Conclusion

We propose a general framework, which we call decision-driven regularization, for the joint prediction and optimization problem. We are, to the best of our knowledge, the first to present major key tenets when considering such approaches, and to justify how they lead to sub-optimality when the learning and decisions are conducted separately. The presence of a framework, under which two models in the literature are generalized (JERO in Zhu et al. 2019 and SPO+ in Elmachtoub and Grigas 2017), allows us to make comparisons and inferences upon their performance in a manner that is supported by theory.

In our framework, we introduce both the notion of a decision-driven regularizer and allowed it to be defined along the ambiguity in the cost function. This notion of cost function ambiguity links to similar existing notions in other areas of Machine Learning, particularly, Reinforcement Learning. Additionally, the presence of a decision problem that which can be used to shape the bias-variance trade-off, provides new technology to examine learning under structure, where it is likely more convenient to encode the structure as the decision problem. These connections yield tantalizing opportunities for future study, a note we hope to end this paper on.

References

Ascarza, E. 2018. Retention futility: Targeting high-risk customers might be ineffective. *Journal of Marketing Research* **55**(1) 80–98.

Athey, S., J. Tibshirani, S. Wager. 2019. Generalized random forests. *Annals of Statistics* **47**(2) 1148–78.

Ban, G.Y., C. Rudin. 2019. The big data newsvendor: Practical insights from machine learning. *Operations Research* **67**(1) 90–108.

Bertsimas, D., M.S. Copenhaver. 2018. Characterization of the equivalence of robustification and regularization in linear and matrix regression. *European Journal of Operational Research* **270**(3) 931–942.

Bertsimas, D., J. Dunn. 2017. Optimal classification trees. *Machine Learning* **106**(7) 1039–82.

Bertsimas, D., V. Gupta, N. Kallus. 2018. Data-driven robust optimization. *Mathematical Programming* **167**(2) 235–292.

Bertsimas, D., N. Kallus. 2020. From predictive to prescriptive analytics. *Management Science* **66**(3) 1025–1044.

Delage, E., Y. Ye. 2010. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research* **58**(3) 595–612.

den Hertog, D., K. Postek. 2016. Bridging the gap between predictive and prescriptive analytics-new optimization methodology needed. Extracted from optimization-online.

Elmachtoub, A.N., P. Grigas. 2017. Smart “predict, then optimize”. Extracted from arxiv: 1710.08005.

Ferreira, K.J., B.H.A. Lee, D. Simchi-Levi. 2016. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management* **18**(1) 69–88.

Fisher, M., R. Vaidyanathan. 2014. A demand estimation procedure for retail assortment optimization with results from implementations. *Management Science* **60**(10) 2401–2415.

Gao, R., X. Chen, A.J. Kleywegt. 2017. Wasserstein distributional robustness and regularization in statistical learning. *Extracted from arXiv:1712.06050* .

Glaeser, C.K., M. Fisher, X. Su. 2019. Optimal retail location: Empirical methodology and application to practice. *Manufacturing & Service Operations Management* **21**(1) 86–102.

Gotoh, J-Y., M.J. Kim, A.E.B. Lim. 2017. Calibration of distributionally robust empirical optimization models. *arXiv preprint arXiv:1711.06565* .

Guilkey, D.K., J.L. Murphy. 1975. Directed ridge regression techniques in cases of multicollinearity. *Journal of the American Statistical Association* **70**(352) 769–775.

Huang, T., D. Bergman, R. Gopal. 2019. Predictive and prescriptive analytics for location selection of add-on retail products. *Production and Operations Management* **28**(7) 1858–1877.

Kao, Y.H., B. Van Roy, X. Yan. 2009. Directed regression. *Advances in Neural Information Processing Systems* **22** 889–897.

Knight, K., W. Fu. 2000. Asymptotics for LASSO-type estimators. *Annals of Statistics* **28**(5) 1356–1378.

Liu, S., L. He, Z.J.M. Shen. 2020. On-time last mile delivery: Order assignment with travel time predictors. *Management Science* .

Liyanage, L.H., G. Shanthikumar. 2005. A practical inventory control policy using operational statistics. *Operations Research Letters* **33**(4) 341–348.

Meinshausen, N., P. Bühlmann, et al. 2006. High-dimensional graphs and variable selection with the LASSO. *Annals of Statistics* **34**(3) 1436–1462.

Mohajerin Esfahani, P., D. Kuhn. 2018. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming* .

Nemirovski, A., A. Shapiro. 2006. Scenario approximations of chance constraints. *Probabilistic and randomized methods for design under uncertainty*. Springer, 3–47.

Perakis, G., M. Sim, Q. Tang, P. Xiong. 2018. Robust pricing and production with information partitioning and adaptation. Available at SSRN 3305039.

Pollard, D. 1991. Asymptotics for least absolute deviation regression estimators. *Econometric Theory* **7**(2) 186–199.

Shapiro, A. 2003. Monte Carlo sampling methods. *Handbooks in operations research and management science* **10** 353–425.

Van Parys, B.P., P. Mohajerin Esfahani, D. Kuhn. 2020. From data to decisions: Distributionally robust optimization is optimal. *Management Science. Forthcoming* .

Vapnik, V. 1992. Principles of risk minimization for learning theory. *Advances in neural information processing systems*. 831–838.

Wiesemann, W., D. Kuhn, M. Sim. 2014. Distributionally robust convex optimization. *Operations Research* **62**(6) 1358–1376.

Xu, H., C. Caramanis, S. Mannor. 2010. Robust regression and LASSO. *IEEE Transactions on Information Theory* **56**(7) 3561–74.

Yan, Z., C. Cheng, K. Natarajan, C. Teo. 2019. A representative consumer model in data-driven multi-product pricing optimization. Available at SSRN 2832385.

Zhao, P., B. Yu. 2006. On model selection consistency of LASSO. *Journal of Machine Learning Research* **7**(Nov) 2541–2563.

Zhu, T., J. Xie, M. Sim. 2019. Joint estimation and robustness optimization. Available at SSRN 3335889.

A. Omitted Proofs

In this segment, we present all deferred proofs from the main text.

A.1. Proof of Proposition 1

Let the dual variable of the RHS inequality be α and of the LHS be β, then the Lagrangian relaxation of the problem is given by:

$$\begin{aligned}&\inf_{\boldsymbol{y}\in\mathcal{Y}} \quad C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)+\alpha(C(\boldsymbol{y};\tilde{\boldsymbol{z}})-C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))-\gamma_2)+\beta(\gamma_1-C(\boldsymbol{y};\tilde{\boldsymbol{z}})+C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})))\\ =&\inf_{\boldsymbol{y}\in\mathcal{Y}} \quad (\alpha-\beta)C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\alpha+\beta)C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)-\alpha\gamma_2+\beta\gamma_1.\end{aligned}$$

Setting $\mu=\alpha-\beta$, we recover the statement of the Proposition, where the γ-terms are dropped as it still results in the same choice of decisions $\boldsymbol{y}$. □

A.2. Proof of Proposition 2

Because $C(\boldsymbol{y};f(\boldsymbol{x},\boldsymbol{w}))$ is convex in $\boldsymbol{y}$ and concave in $\boldsymbol{w}$ for all $\boldsymbol{x}$, $C(\boldsymbol{y};\tilde{\boldsymbol{z}})$ is convex in $\boldsymbol{y}$, and $\mu\in[0,1]$, then by linearity, the objective function $\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))$ is convex in $\boldsymbol{y}$ and concave in $\boldsymbol{w}$. The concavity of $v_\mu(\boldsymbol{w})$ over $\boldsymbol{w}$ arises from the infimum operator over $\boldsymbol{y}$. □

A.3. Proof of Theorem 1

To prove consistency, the common approach is to define $\Lambda(\boldsymbol{w})$, which is the limiting function of Λ^s as $s\to\infty$. As opposed to proving convergence to some unknown limit, the more common strategy is to prove that $\sup_{\boldsymbol{w}\in\mathcal{K}}|\Lambda^s(\boldsymbol{w})-\Lambda(\boldsymbol{w})|\to 0$ in probability for all compact sets $\mathcal{K}$ in the domain of $\boldsymbol{w}$. The handy result by Pollard (1991) allows for simplifying this task to proving that $\Lambda^s(\boldsymbol{w})\to\Lambda(\boldsymbol{w})$ pointwise in probability, if Λ^s is convex. This condition holds due to the combination of r being convex non-increasing and Proposition 2. Finally, as it is assumed that (DDR) is consistent when $\lambda=0$, this solves the matter for the term involving the loss function L. Moreover, as r is continuously differentiable with bounded first derivatives, uniform continuity is guaranteed. As such, it suffices to prove that $J^s(\boldsymbol{w}):=\inf_{\boldsymbol{y}\in\mathcal{Y}}\Big\{\mu\,C(\boldsymbol{y};\tilde{\boldsymbol{z}}^s)+(1-\mu)\,C\big(\boldsymbol{y};\boldsymbol{f}(\boldsymbol{x}^s;\boldsymbol{w})\big)\Big\}$ converges pointwise in probability.

Now, it is clear that the expression within the infimum, $\mu\,C(\boldsymbol{y};\tilde{\boldsymbol{z}}^s)+(1-\mu)\,C\big(\boldsymbol{y};\boldsymbol{f}(\boldsymbol{x}^s;\boldsymbol{w})\big)$, converges as $s\to\infty$ in probability. As it is also bounded above within the feasibility set $\mathcal{Y}$, the combination of triangle inequality and Fatou's Lemma yields that the infimum itself converges in probability. This completes the proof. □

A.4. Proof of Theorem 2

Suppose $\mu\in[0,1)$ and $r(v):=-v$, it follows that the DDR probem now becomes

$$\inf_{\boldsymbol{w}}\sup_{\boldsymbol{y}\in\mathcal{Y}} L(\boldsymbol{w})-\lambda\Big(\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\Big).$$

By introducing a dual variable α, we can write the Lagrangian of (9) as:

$$\begin{aligned}
&\sup_{\boldsymbol{y}\in\mathcal{Y},\alpha>0}\ \inf_{\boldsymbol{w}} \quad \alpha\Big(L(\boldsymbol{w})-L(\tilde{\boldsymbol{w}})-\rho\Big)-\Big(\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\Big)\\
=\ &\inf_{\boldsymbol{w}}\ \sup_{\boldsymbol{y}\in\mathcal{Y},\alpha>0} \quad \alpha\Big(L(\boldsymbol{w})-L(\tilde{\boldsymbol{w}})-\rho\Big)-\Big(\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\Big).
\end{aligned}$$

Notice that $\boldsymbol{w}=\tilde{\boldsymbol{w}}$ is always a feasible solution by assumption. Moreover, $\rho>0$, hence $\boldsymbol{w}=\tilde{\boldsymbol{w}}$ is an interior point in the feasibility region. This achieves Slater's condition, and therefore strong duality holds. Moreover, in general, this would be an inequality due to the min-max inequality. However, in this case, the equality condition is satisfied because Assumption 1 permits convexity in $\boldsymbol{w}$, and concavity in $(\boldsymbol{y},\alpha)$. By comparing, one can now see that the minimizers of $\boldsymbol{w}$ and $\boldsymbol{y}$ in both problems coincide, by setting $\alpha^*=1/\lambda$. □

A.5. Proof of Proposition 3 and Corollary 1

Let us prove both the proposition and the corollary simultaneously. Recall the formulation for the RDDR problem over the range of $\mu\in[0,1)$:

$$\begin{aligned}
\max_{\rho>0,\,\boldsymbol{y}\in\mathcal{Y}} \quad & \rho\\
\text{s.t.} \quad & \mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\le\tau, \qquad \forall \boldsymbol{w}\in\mathcal{U}(\rho):=\{\boldsymbol{w}: L(\boldsymbol{w})-L(\tilde{\boldsymbol{w}})\le\rho\}.
\end{aligned}$$

Once again, notice that when $\mu=0$, JERO is recovered. For brevity, denote by $\sigma=L(\tilde{\boldsymbol{w}})$, which is a known constant given the data. The constraint therefore has robust counterpart:

$$\begin{aligned}
\sup_{\boldsymbol{w}} \quad & \mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\\
\text{s.t.} \quad & L(\boldsymbol{w})\le\sigma+\rho.
\end{aligned}$$

We can compute the dual of this problem as follows, which will have the same objective value should strong duality holds. However, notice that $\boldsymbol{w}=\tilde{\boldsymbol{w}}$ is always a feasible solution by assumption. Moreover, $\rho>0$, hence $\boldsymbol{w}=\tilde{\boldsymbol{w}}$ is an interior point in the feasibility region. This achieves Slater's condition, and therefore strong duality holds.

$$\begin{aligned}
&\inf_{\alpha>0}\sup_{\boldsymbol{w}} \quad \mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))+\alpha(\sigma+\rho-L(\boldsymbol{w}))\le\tau\\
\Leftrightarrow &\sup_{\beta>0,\boldsymbol{w}} \quad \beta\left[\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\right]+\sigma+\rho-L(\boldsymbol{w})-\beta\tau\le 0,
\end{aligned}$$

where α is the dual variable for the constraint on the loss function, and β is a change of variables.

Hence, in fact, ρ can be extracted as

$$\rho\le\inf_{\boldsymbol{w},\beta>0} -\beta\left[\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\right]-\sigma+L(\boldsymbol{w})+\beta\tau,$$

which will allow us to re-formulate the problem as:

$$\max_{\boldsymbol{y}\in\mathcal{Y}} \inf_{\boldsymbol{w},\beta>0} -\beta\left[\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)C(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w}))\right]-\sigma+L(\boldsymbol{w})+\beta\tau.$$

However, as shown in Theorem 2, the DDR problem can be reformulated as:

$$\begin{aligned}&\inf_{\boldsymbol{w}}\sup_{\boldsymbol{y}\in\mathcal{Y}}\{L(\boldsymbol{w})+\lambda(\tau-\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})-(1-\mu)C(\boldsymbol{y};f(\boldsymbol{x};\boldsymbol{w})))\}\\ =&\sup_{\boldsymbol{y}\in\mathcal{Y}}\inf_{\boldsymbol{w}}\{L(\boldsymbol{w})+\lambda(\tau-\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}})-(1-\mu)C(\boldsymbol{y};f(\boldsymbol{x};\boldsymbol{w})))\},\end{aligned}$$

where the last equality holds due to the equality condition on the min-max inequality. This is justified by the objective function being jointly concave in $\boldsymbol{y}$ and convex in $\boldsymbol{w}$. Moreover, by assumption, there exists some $\boldsymbol{y}$ under which the supermum is attained.

It is now easy to visually verify, therefore, by setting $\lambda=\beta^*$, the optimal β attained under the reformulated problem that there is some λ under which both formulations coincide. □

A.6. Proof of Proposition 4

By setting the loss function as $L(\boldsymbol{w})=(1-\mu)C(\boldsymbol{y}^*(\tilde{\boldsymbol{z}});f(\boldsymbol{x};\boldsymbol{w}))$ and $\mu=-1$, we can almost recover the result in SPO+. The only key difference is that in SPO+, the regularization term is phrased as $\sum_n \sup_{y_n}\{c(y_n;\tilde{z}_n)-2c(y_n;\hat{z}_n)\}$, whereas it is $\sup_{\boldsymbol{y}}\sum_n\{c(y_n;\tilde{z}_n)-2c(y_n;\hat{z}_n)\}$ in DDR. In general, the former is larger than the latter. However, by assuming that there are no constraints relating the y_n's, each summand on y_n is independent of the other and hence can be separated under the supremum. Hence, these two terms are equivalent. □

Remark 1. It is worthwhile to take a moment and describe the subtlety involved with the swapping of the summation and the supremum. SPO+, and its choice to order the summation and supremum as such, renders it fundamentally unable to accommodate any form of joint constraints on y_i. While swapping the summation and the supremum appears to solve this matter trivially, it is not trivial to appreciate why this is necessary or desirable without the awareness of a 'decision-driven regularizer'. Indeed, it is not mentioned in Elmachtoub and Grigas (2017) their intent to do so either. In this regard, DDR is an important extension of SPO+ as it enables SPO+ to incorporate constraints on the decisions $\boldsymbol{y}$.

B. Pseudo-code for the Solving (RDDR)

Definition 5 (Robustness DDR). The Robustness Decision-driven Regularization (RDDR, for short) model is defined as the following problem for $\mu\in[0,1)$:

$$\begin{aligned}\max_{\rho>0,\,\boldsymbol{y}\in\mathcal{Y}}\ &\rho \qquad\qquad \text{(RDDR)}\\ \text{s.t. }\ &\mu\, C(\boldsymbol{y};\tilde{\boldsymbol{z}})+(1-\mu)\,C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big)\le\tau \qquad \forall \boldsymbol{w}\in\mathcal{U}(\rho),\\ \text{where }\ &\mathcal{U}(\rho):=\{\boldsymbol{w}:L(\boldsymbol{w})-L(\tilde{\boldsymbol{w}})\le\rho\}.\end{aligned}$$

COROLLARY 1. *Assume that $\mathcal{Y}$ is convex, closed and compact, and that Assumption 1 holds. Then for all τ for which $\exists \boldsymbol{y} \in \mathcal{Y}$ such that the target is attained $\mu C(\boldsymbol{y};\tilde{\boldsymbol{z}}) + (1-\mu) C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\tilde{\boldsymbol{w}})\big) \le \tau$, under learning optimal weights $\tilde{\boldsymbol{w}}$, there exists some $\lambda := \lambda(\tau) \ge 0$ for which the solutions of DDR and RDDR coincide, when $\mu \in [0,1)$ and $r(v) := \tau - v$.*

In short, we seek for the best radius of the uncertainty set ρ, via bisection search, wherein each iteration, we solve a feasibility sub-problem. Here, we present the pseudo-code for the algorithm, however, for more details, the reader can be directed to Zhu et al. (2019), as the procedure for solving the model follows identically.

Algorithm 1 RDDR

```
Input $\tau$ and $\tilde{\boldsymbol{z}}$.
Initialization: $\rho_1 = 0, \rho_2 = \bar{\rho}$, where $\bar{\rho}$ is a sufficiently large number.
while $\rho_2 - \rho_1 > \epsilon$ do
    $\rho := (\rho_1 + r_2)/2$
    Solve the subproblem Sub-RDDR, obtain optimal value $\delta^*$ and optimal decision $\boldsymbol{y}^*$.
    if $\delta^* > 0$ then
        $\rho_2 = \rho$
    else
        $\rho_1 = \rho$
    end if
end while
Output: Optimal $\rho^* = (\rho_1 + \rho_2)/2$ and optimal decision $\boldsymbol{y}_{\text{RDDR}} = \boldsymbol{y}^*$.
Input Optimal $\rho = \rho^*$ and optimal decision $\boldsymbol{y} = \boldsymbol{y}^*$
Do Solve the problem (ROBUSTW), and obtain $\boldsymbol{w}^*$
Output Worst scenario $\hat{\boldsymbol{w}}_{\text{RDDR}} = \boldsymbol{w}^*$
```

In this algorithm, the sub-problem is described as:

$$\begin{aligned} \min_{t} \quad & t - \tau \\ \text{s.t.} \quad & \mu\, C(\boldsymbol{y};\tilde{\boldsymbol{z}}) + (1-\mu)\, C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big) \le t, \qquad \forall \boldsymbol{w} \in \mathcal{U}(\rho) \\ \text{where} \quad & \mathcal{U}(\rho) := \{\boldsymbol{w} : L(\boldsymbol{w}) - L(\tilde{\boldsymbol{w}}) \le \rho\}. \end{aligned} \tag{Sub-RDDR}$$

We finally utilize ρ^* and $\boldsymbol{y}^*$ derived from the overarching problem to solve for the worst case $\boldsymbol{w}$, the problem is

$$\begin{aligned} \max_{\boldsymbol{w}} \quad & (1-\mu)\, C\big(\boldsymbol{y};\hat{\boldsymbol{z}}(\boldsymbol{w})\big) \\ \text{s.t.} \quad & L(\boldsymbol{w}) - L(\tilde{\boldsymbol{w}}) \le \rho. \end{aligned} \tag{ROBUSTW}$$

C. Additional Simulation Results

We attach here some additional simulation results of OLS-SPO+ hybrid as α, N, d and p vary. Specifically, we let $d = 3$, $p = 4$, $\alpha = 1$, and $N = 100$ on $\overline{N} = 5000$ be the basic setting, and vary α over the set $\{0.1, 0.25, 0.5, 1.0, 2.0, 3.0\}$, N over the set $\{50, 100, 500, 1000, 5000\}$, and (d, p) over the set $\{(3,3), (3,4), (3,5), (4,3), (4,4), (4,5)\}$. Hence, in total, we have $6 + 5 + 6 = 17$ instances. For each instance, we average each performance metrics over 100 differently generated simulations of the true weights. Figure 8 displays how the relationship between the mean cost improvement and head-to-head metrics varies under different selections of α (top left), N (top right), and (d, p) (bottom).

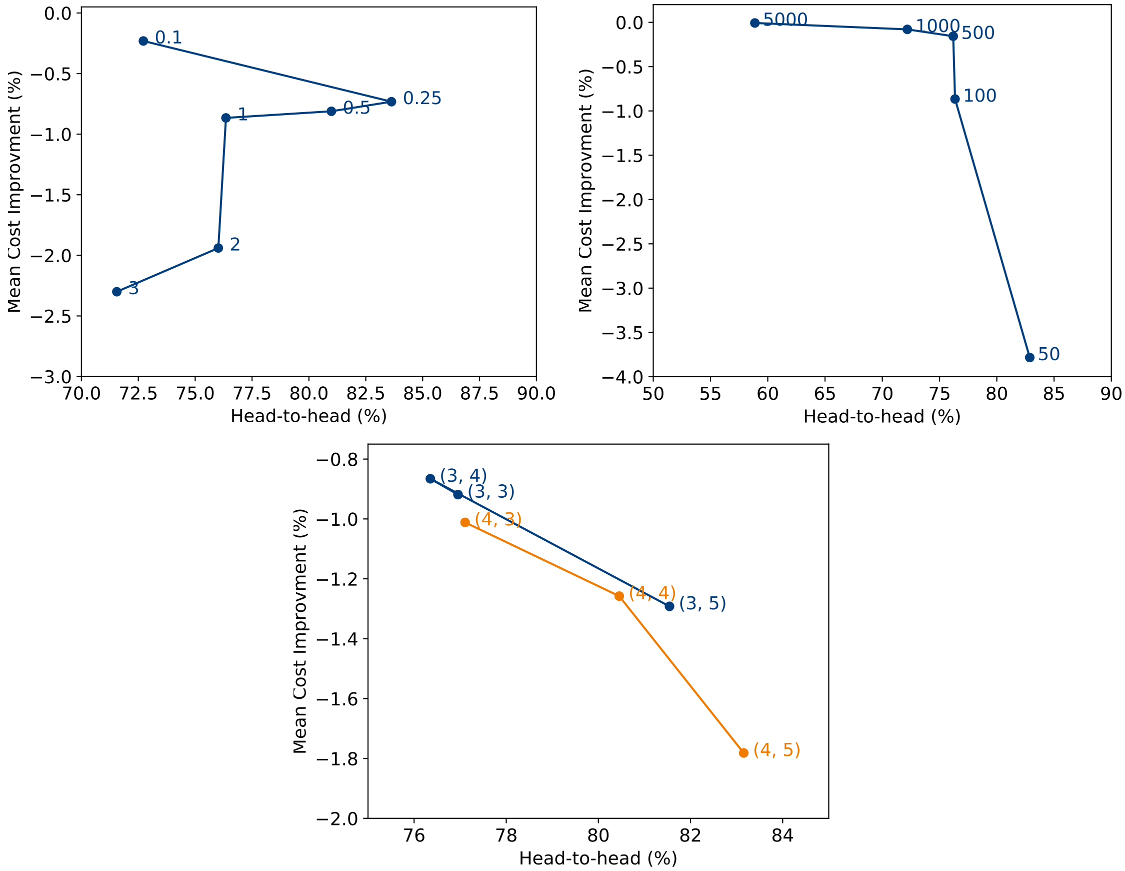

Figure 8 Mean cost improvement and head-to-head metrics w.r.t. α (top left), N (top right), and (d, p) (bottom)

Figure 8 aligns with our intuition: The less the noise (α), the less the overfitting by SPO+. Also, the greater the number of data point (N), the nearer the models converge to the true model, so the difference in their performances vanishes. In addition, note that p and d represent number of features and routes, both of which if increased, has the potential to raise the potential overfitting of SPO+, leading to poorer performance, vis-a-vis our hybrid model.