Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Smart "Predict, then Optimize"

Adam N. Elmachtoub, Paul Grigas

To cite this article:
Adam N. Elmachtoub, Paul Grigas (2022) Smart "Predict, then Optimize". Management Science 68(1):9-26. https://doi.org/10.1287/mnsc.2020.3922

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

http://pubsonline.informs.org/journal/mnsc

MANAGEMENT SCIENCE
Vol. 68, No. 1, January 2022, pp. 9–26
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

Smart "Predict, then Optimize"

Adam N. Elmachtoub,[a] Paul Grigas[b]

[a] Department of Industrial Engineering and Operations Research and Data Science Institute, Columbia University, New York, New York 10027; [b] Department of Industrial Engineering and Operations Research, University of California, Berkeley, Berkeley, California 94720

Contact: adam@ieor.columbia.edu, https://orcid.org/0000-0003-0729-4999 (ANE); pgrigas@berkeley.edu, https://orcid.org/0000-0002-5617-1058 (PG)

Received: December 14, 2017
Revised: July 18, 2019; July 8, 2020
Accepted: November 3, 2020
Published Online in Articles in Advance: March 12, 2021

https://doi.org/10.1287/mnsc.2020.3922

Abstract. Many real-world analytics problems involve two significant challenges: prediction and optimization. Because of the typically complex nature of each challenge, the standard paradigm is predict-then-optimize. By and large, machine learning tools are intended to minimize prediction error and do not account for how the predictions will be used in the downstream optimization problem. In contrast, we propose a new and very general framework, called Smart "Predict, then Optimize" (SPO), which directly leverages the optimization problem structure—that is, its objective and constraints—for designing better prediction models. A key component of our framework is the SPO loss function, which measures the decision error induced by a prediction. Training a prediction model with respect to the SPO loss is computationally challenging, and, thus, we derive, using duality theory, a convex surrogate loss function, which we call the SPO+ loss. Most importantly, we prove that the SPO+ loss is statistically consistent with respect to the SPO loss under mild conditions. Our SPO+ loss function can tractably handle any polyhedral, convex, or even mixed-integer optimization problem with a linear objective. Numerical experiments on shortest-path and portfolio-optimization problems show that the SPO framework can lead to significant improvement under the predict-then-optimize paradigm, in particular, when the prediction model being trained is misspecified. We find that linear models trained using SPO+ loss tend to dominate random-forest algorithms, even when the ground truth is highly nonlinear.

History: Accepted by Yinyu Ye, optimization.
Funding: Financial support from the National Science Foundation Division of Computing and Communication Foundation [Award CCF-1755705] and Division of Civil, Mechanical, and Manufacturing Innovation [Awards CMMI-1762744 and CMMI-1763000] is gratefully acknowledged.
Supplemental Material: Data and the online appendix are available at https://doi.org/10.1287/mnsc.2020.3922

Keywords: prescriptive analytics • data-driven optimization • machine learning • linear regression

1. Introduction

In many real-world analytics applications of operations research, a combination of both machine learning and optimization are used to make decisions. Typically, the optimization model is used to generate decisions, while a machine learning tool is used to generate a prediction model that predicts key unknown parameters of the optimization model. Because of the inherent complexity of both tasks, a broad-purpose approach that is often employed in analytics practice is the *predict-then-optimize* paradigm.

For example, consider a vehicle-routing problem that may be solved several times a day. First, a previously trained prediction model provides predictions for the travel time on all edges of a road network based on current traffic, weather, holidays, time, etc. Then, an optimization solver provides near-optimal routes using the predicted travel times as input. We emphasize that most solution systems for real-world analytics problems involve some component of both prediction and optimization (see Mehrotra et al. 2011, Chan et al. 2012, Chan et al. 2013, Angalakudati et al. 2014, Besbes et al. 2015, Deo et al. 2015, Ferreira et al. 2015, Gallien et al. 2015, and Cohen et al. 2017 for recent examples and recent expositions by Simchi-Levi 2013, den Hertog and Postek 2016, Deng et al. 2018, and Mišić and Perakis 2020). Except for a few limited options, machine learning tools do not effectively account for how the predictions will be used in a downstream optimization problem. In this paper, we provide a general framework called Smart "Predict, then Optimize" (SPO) for training prediction models that effectively utilizes the structure of the nominal optimization problem—that is, its constraints and objective. Our SPO framework is fundamentally designed to generate prediction models that aim to minimize decision error, not prediction error.

One key benefit of our SPO approach is that it maintains the decision paradigm of sequentially predicting and then optimizing. However, when training

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

our prediction model, the structure of the nominal optimization problem is explicitly used. The quality of a prediction is *not* measured based on prediction error, such as least-squares loss or other popular loss functions. Instead, in the SPO framework, the quality of a prediction is measured by the decision error. That is, suppose a prediction model is trained using historical feature data $(x_1, \ldots, x_n)$ and associated parameter data $(c_1, \ldots, c_n)$. Let $(\hat{c}_1, \ldots, \hat{c}_n)$ denote the predictions of the parameters under the trained model. The least-squares (LS) loss, for example, measures error with the squared norm $\|c_i - \hat{c}_i\|_2^2$, completely ignoring the decisions induced by the predictions. In contrast, the *SPO loss* is the true cost of the decision induced by $\hat{c}_i$ minus the optimal cost under the true parameter c_i. In the context of vehicle routing, the SPO loss measures the extra travel time incurred due to solving the routing problem on the predicted, rather than true, edge cost parameters.

In this paper, we focus on predicting unknown parameters of a contextual stochastic optimization problem, where the parameters appear linearly in the objective function—that is, the cost vector of any linear, convex, or integer optimization problem. The core of our SPO framework is a new loss function for training prediction models. Because the SPO loss function is difficult to work with, significant effort revolves around deriving a surrogate loss function, SPO+, that is convex and, therefore, can be optimized efficiently. To show the validity of the surrogate SPO+ loss, we prove a highly desirable statistical consistency property and show that it performs well empirically compared with standard predict-then-optimize approaches. In essence, we prove that the function that minimizes the Bayes risk associated to the SPO+ loss is the regression function $\mathbb{E}[c|x]$, which also minimizes the Bayes risk of the SPO loss (under mild assumptions). Interestingly, $\mathbb{E}[c|x]$ also minimizes the Bayes risk associated with the LS loss under the same conditions. Thus, SPO+ and LS (or any convex combination of the two) are essentially on "equal footing"—they are both theoretically valid (consistent) and computationally tractable choices for the loss function. However, when the ultimate goal is to solve a downstream optimization task, the SPO+ loss is the natural choice, as it is tailored to the optimization problem and works significantly better in practice than LS.

Empirically, we observe that, even when the prediction task is challenging due to model misspecification, the SPO framework can still yield near-optimal decisions. We note that a fundamental property of the SPO framework is the requirement that the prediction is directly "plugged in" to the downstream optimization problem. An alternative procedure may alter the decision-making process in some way, such as by adding robustness or by taking into account the entire data set (instead of just the prediction). A strong advantage of our SPO approach is that it has good performance, even when the naive prediction problem is challenging; see the illustrative example in Section 3.1. Another advantage is that the downstream optimization problem is typically more computationally tractable and more attractive to practitioners than a more complex alternative procedure. On the other hand, alternative decision-making procedures may provide other advantages, such as improved generalization performance via the introduction of bias and/or robustness. However, designing such procedures is more challenging in the presence of contextual data, and combining them with the SPO approach would be worthwhile of future research. Overall, we believe our SPO framework provides a clear foundation for designing operations-driven machine learning (ML) tools that can be leveraged in real-world optimization settings.

Our contributions may be summarized as follows:

1. We first formally define a new loss function, which we call the SPO loss, that measures the error in predicting the cost vector of a nominal optimization problem with linear, convex, or integer constraints. The loss corresponds to the suboptimality gap—with respect to the true/historical cost vector—due to implementing a possibly incorrect decision induced by the predicted cost vector. Unfortunately, the SPO loss function can be nonconvex and discontinuous in the predictions, implying that training ML models under the SPO loss may be challenging.

2. Given the intractability of the SPO loss function, we develop a surrogate loss function, which we call the SPO+ loss. This surrogate loss function is derived by using a sequence of steps motivated by duality theory (Proposition 2), a data-scaling approximation, and a first-order approximation. The resulting SPO+ loss function is convex in the predictions (Proposition 3), which allows us to design an algorithm based on stochastic gradient descent for minimizing SPO+ loss (Proposition 8). Moreover, when training a linear regression model to predict the objective coefficients of a linear program, only a linear optimization problem needs be solved to minimize the SPO+ loss (Proposition 7).

3. We prove a fundamental connection to classical machine learning under a very simple and special instance of our SPO framework. Namely, under this instance, the SPO loss is exactly the 0-1 classification loss (Proposition 1), and the SPO+ loss is exactly the hinge loss (Proposition 4). The hinge loss is the basis of the popular support-vector machine (SVM) method and is a surrogate loss to approximately minimize the 0-1 loss, and, thus, our framework generalizes this concept to a very wide family of optimization problems with constraints.

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

4. We prove a key consistency result of the SPO+ loss function (Theorem 1, Proposition 5, and Proposition 6), which further motivates its use. Namely, under full distributional knowledge, minimizing the SPO+ loss function is, in fact, equivalent to minimizing the SPO loss if two mild conditions hold: The distribution of the cost vector (given the features) is continuous and symmetric about its mean. For example, these assumptions are satisfied by the standard Gaussian noise approximation. This consistency property is widely regarded as an essential property of any surrogate loss function across the statistics and machine learning literature. For example, the famous hinge loss and logistic loss functions are consistent with the 0-1 classification loss.

5. Finally, we validate our framework through numerical experiments on the shortest-path and portfolio-optimization problem. We test our SPO framework against standard predict-then-optimize approaches and evaluate the out-of-sample performance with respect to the SPO loss. Generally, the value of our SPO framework increases as the degree of model misspecification increases. This is precisely due to the fact the SPO framework makes "better" wrong predictions, essentially "tricking" the optimization problem into finding near-optimal solutions. Remarkably, a linear model trained using SPO+ even dominates a state-of-the-art random-forests algorithm, even when the ground truth is highly nonlinear.

1.1. Applications

Settings where the input parameters (cost vectors) of an optimization problem need to be predicted from contextual (feature) data are numerous. Let us now highlight a few, of potentially many, application areas for the SPO framework.

1.1.1. Vehicle Routing. In numerous applications, the cost of each edge of a graph needs to be predicted before making a routing decision. The cost of an edge typically corresponds to the expected length of time a vehicle would need to traverse the corresponding edge. For clarity, let us focus on one important example—the shortest-path problem. In the shortest-path problem, one is given a weighted directed graph, along with an origin node and destination node, and the goal is to find a sequence of edges from the origin to the destination at minimum possible cost. A well-known fact is that the shortest-path problem can be formulated as a linear optimization problem, but there are also alternative specialized algorithms, such as the famous Dijkstra's algorithm (see, e.g., Ahuja et al. 1993). The data used to predict the cost of the edges may incorporate the length, speed limit, weather, season, day, and real-time data from mobile applications, such as Google Maps and Waze. Simply minimizing prediction error may not suffice or be appropriate, as overpredictions or underpredictions have starkly different effects across the network. The SPO framework would ensure that the predicted weights lead to shortest paths and would naturally emphasize the estimation of edges that are critical to this decision. See Section 3.1 for an in-depth example.

1.1.2. Inventory Management. In inventory-planning problems, such as the economic lot-sizing problem (Wagner and Whitin 1958) or the joint-replenishment problem (Levi et al. 2006), the demand is the key input into the optimization model. In practical settings, demand is highly nonstationary and can depend on historical and contextual data, such as weather, seasonality, and competitor sales. The decisions of when to order inventory are captured by a linear- or integer-optimization model, depending on the complexity of the problem. Under a common formulation (see Levi et al. 2006 and Cheung et al. 2016), the demand appears linearly in the objective, which is convenient for the SPO framework. The goal is to design a prediction model that maps feature data to demand predictions, which, in turn, lead to good inventory plans.

1.1.3. Portfolio Optimization. In financial-services applications, the returns of potential investments need to be somehow estimated from data and can depend on many features, which typically include historical returns, news, economic factors, social media, and others. In portfolio optimization, the goal is to find a portfolio with the highest return subject to a constraint on the total risk, or variance, of the portfolio. Although the returns are often highly dependent on auxiliary feature information, the variances are typically much more stable and are not as difficult or sensitive to predict. Our SPO framework would result in predictions that lead to high-performance investments that satisfy the desired level of risk. A least-squares loss approach places higher emphasis on estimating higher valued investments, even if the corresponding risk may not be ideal. In contrast, the SPO framework directly accounts for the risk of each investment when training the prediction model.

1.2. Related Literature

Perhaps the most related work is that of Kao et al. (2009), who also directly seek to train a machine learning model that minimizes loss with respect to a nominal optimization problem. In their framework, the nominal problem is an unconstrained quadratic optimization problem, where the unknown parameters appear in the linear portion of the objective. Their work does not extend to settings where the nominal optimization

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

problem has constraints, which our framework does. Donti et al. (2017) proposes a heuristic to address a more general setting than that of Kao et al. (2009) and also focus on the case of quadratic optimization. These works also bypass issues of nonuniqueness of solutions of the nominal problem (because their problem is strongly convex), which must be addressed in our setting to avoid degenerate prediction models.

In Ban and Rudin (2019), ML models are trained to directly predict the optimal solution of a newsvendor problem from data. Tractability and statistical properties of the method are shown, as well as its effectiveness in practice. However, it is not clear how this approach can be used when there are constraints, because feasibility issues may arise.

The general approach in Bertsimas and Kallus (2020) considers the problem of accurately estimating an unknown optimization objective using ML models where the predictions can be described as a weighted combination of training samples—for example, nearest neighbors and decision trees. In their approach, they estimate the objective of an instance by applying the same weights generated by the ML model to the corresponding objective functions of those samples. This approach differs from standard predict-then-optimize *only* when the objective function is nonlinear in the unknown parameter. Note that the unknown parameters of all the applications mentioned in Section 1.1 appear linearly in the objective. Moreover, the training of the ML models does not rely on the structure of the nominal optimization problem, in contrast to the SPO framework.

The approach in Tulabandhula and Rudin (2013) relies on minimizing a loss function that combines the prediction error with the operational cost of the model on an unlabeled data set. However, the operational cost is with respect to the predicted parameters, and not the true parameters. Gupta and Rusmevichientong (2017) consider combining estimation and optimization in a setting without features/contexts. We also note that our SPO loss, although mathematically different, is similar in spirit to the notion of relative regret introduced in Lim et al. (2012) in the specific context of portfolio optimization with historical return data and without features. Other approaches for finding near-optimal solutions from data include operational statistics (Liyanage and Shanthikumar 2005, Chu et al. 2008), sample average approximation (Kleywegt et al. 2002, Schütz et al. 2009, Bertsimas et al. 2018b), and robust optimization (Bertsimas and Thiele 2006, Wang et al. 2016, Bertsimas et al. 2018a). There has also been some recent progress on submodular optimization from samples (Balkanski et al. 2016, 2017). These approaches typically do not have a clear way of using feature data, nor do they directly consider how to train a machine learning model to predict optimization parameters.

Another related stream of work is in data-driven inverse optimization, where feasible or optimal solutions to an optimization problem are observed, and the objective function has to be learned (Keshavarz et al. 2011, Chan et al. 2014, Bertsimas et al. 2015, Aswani et al. 2018, Esfahani et al. 2018). In these problems, there is typically a single unknown objective, and no previous samples of the objective are provided. We also note there have been recent approaches for regularization (Ban et al. 2018) and model selection (Besbes et al. 2010, Den Boer and Sierag 2020, Sen and Deng 2017) in the context of an optimization problem.

Lastly, we note that our framework is related to the general setting of structured prediction (see, e.g., Taskar et al. 2005, Tsochantaridis et al. 2005, Nowozin et al. 2011, Osokin et al. 2017, and the references therein). Motivated by problems in computer vision and natural language processing, structured prediction is a version of multiclass classification that is concerned with predicting structured objects, such as sequences or graphs, from feature data. The SPO+ loss is similar in spirit to that of the structured SVM (SSVM) and is, indeed a convex, upper bound on the SPO loss, akin to the SSVM. However, there are fundamental differences with our approach and the SSVM approach. In the SSVM approach, the structured object one would be predicting is the decision w directly from the feature x (Taskar et al. 2005). In our setting, we have access to historical data on c, which are richer than observations of decisions, because cost vectors induce optimal decisions naturally. Under one special case of our framework, we prove that the SPO loss is equivalent to 0/1 loss, whereas the SPO+ loss is equivalent to the hinge loss. Thus, our framework can be seen as a type of generalization of the SSVM. Finally, we remark that our derivation of the surrogate SPO+ loss relies on completely new ideas using duality theory, which help explain the strong empirical performance.

2. "Predict, then Optimize" Framework

We now describe the "Predict, then Optimize" (PO) framework, which is central to many applications of optimization in practice. Specifically, we assume that there is a nominal optimization problem of interest with a linear objective, where the decision variable $w \in \mathbb{R}^d$ and feasible region $S \subseteq \mathbb{R}^d$ are well defined and known with certainty. However, the cost vector of the objective, $c \in \mathbb{R}^d$, is not available at the time the decision must be made; instead, an associated feature vector $x \in \mathbb{R}^p$ is available. Let $\mathcal{D}_x$ be the conditional distribution of c given x. The goal for the decision

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

maker is to solve, for any new instance characterized by x, the contextual stochastic optimization problem

$$\min_{w\in S} \mathbb{E}_{c\sim\mathcal{D}_x}[c^\top w \,|\, x] \;=\; \min_{w\in S} \mathbb{E}_{c\sim\mathcal{D}_x}[c \,|\, x]^\top w. \tag{1}$$

The predict-then-optimize framework relies on using a prediction for $\mathbb{E}_{c\sim\mathcal{D}_x}[c \,|\, x]$, which we denote by $\hat{c}$, and solving the deterministic version of the optimization problem based on $\hat{c}$—that is, $\min_{w\in S} \hat{c}^\top w$. Our primary interests in this paper concern defining suitable loss functions for the predict-then-optimize framework, examining their properties, and developing algorithms for training prediction models using these loss functions.

We now formally list the key ingredients of our framework:

1. Nominal (downstream) optimization problem, which is of the form

$$P(c): \quad z^*(c) := \min_{w} \; c^T w \quad \text{s.t. } w \in S, \tag{2}$$

where $w \in \mathbb{R}^d$ are the decision variables, $c \in \mathbb{R}^d$ is the problem data describing the linear objective function, and $S \subseteq \mathbb{R}^d$ is a nonempty, compact (i.e., closed and bounded), and convex set representing the feasible region. Because we are focusing on *linear* optimization problems herein, the assumptions that S is convex and closed are without loss of generality. Indeed, if S in (2) is, instead, possibly nonconvex or nonclosed, then replacing S by its closed convex hull does not change the optimal value $z^*(c)$ (lemma 8 in Jaggi 2011). Thus, this basic equivalence for linear optimization problems implies that our methodology can be applied to combinatorial and mixed-integer optimization problems, which we elaborate on further in Section 3.2. Because S is assumed to be fixed and known with certainty, every problem instance can be described by the corresponding cost vector, hence, the dependence on c in (2). When solving a particular instance where c is unknown, a prediction for c is used instead. We assume access to a practically efficient optimization oracle, $w^*(c)$, that returns a solution of $P(c)$ for any input cost vector. For instance, if (2) corresponds to a linear, conic, or mixed-integer optimization problem, then a commercial optimization solver or a specialized algorithm suffices for $w^*(c)$.

2. *Training data* of the form $(x_1, c_1), (x_2, c_2), \ldots, (x_n, c_n)$, where $x_i \in \mathcal{X}$ is a feature vector representing contextual information associated with c_i.

3. A *hypothesis class* $\mathcal{H}$ of cost-vector prediction models $f : \mathcal{X} \to \mathbb{R}^d$, where $\hat{c} := f(x)$ is interpreted as the predicted cost vector associated with feature vector x.

4. A *loss function* $\ell(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$, whereby $\ell(\hat{c}, c)$ quantifies the error in making prediction $\hat{c}$ when the realized (true) cost vector is actually c.

Given the loss function $\ell(\cdot,\cdot)$ and the training data $(x_1, c_1), \ldots, (x_n, c_n)$, the empirical risk minimization (ERM) principle states that we should determine a prediction model $f^* \in \mathcal{H}$ by solving the optimization problem

$$\min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ell\big(f(x_i), c_i\big). \tag{3}$$

Provided with the prediction model f^* and given a feature vector x, the predict-then-optimize decision rule is to choose the optimal solution with respect to the predicted cost vector—that is, $w^*(f^*(x))$. Example 1 in Online Appendix A contextualizes our framework in the context of a network optimization problem.

In standard applications of the "Predict, then Optimize" framework, as in Example 1, the loss function that is used is completely independent of the nominal optimization problem. In other words, the underlying structure of the optimization problem $P(\cdot)$ does not factor into the loss function and, therefore, the training of the prediction model. For example, when $\ell(\hat{c}, c) = \frac{1}{2}\|\hat{c} - c\|_2^2$, this corresponds to the least-squares loss function. Moreover, if $\mathcal{H}$ is a set of linear predictors, then (3) reduces to a standard least-squares linear regression problem. In contrast, our focus in Section 3 is on the construction of loss functions that measure decision errors in predicting cost vectors by leveraging problem structure.

2.1. Useful Notation

Let p be the dimension of a feature vector, d be the dimension of a decision vector, and n be the number of training samples. Let $W^*(c) := \arg\min_{w\in S}\{c^T w\}$ denote the set of optimal solutions of $P(\cdot)$, and let $w^*(\cdot) : \mathbb{R}^d \to S$ denote a particular *oracle* for solving $P(\cdot)$. That is, $w^*(\cdot)$ is a fixed deterministic mapping such that $w^*(c) \in W^*(c)$. Note that nothing special is assumed about the mapping $w^*(\cdot)$; hence, $w^*(c)$ may be regarded as an arbitrary element of $W^*(c)$. Let $\xi_s(\cdot) : \mathbb{R}^d \to \mathbb{R}$ denote the support function of S, which is defined by $\text{supp}(c) := \max_{w\in S}\{c^T w\}$. Because S is compact, $\xi_s(\cdot)$ is finite everywhere, the maximum in the definition is attained for every $c \in \mathbb{R}^d$, and note that $\text{supp}(c) = -z^*(-c) = c^T w^*(-c)$ for all $c \in \mathbb{R}^d$. Recall also that $\text{supp}(\cdot)$ is a convex function. For a given convex function $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$, recall that $g \in \mathbb{R}^d$ is a subgradient of $h(\cdot)$ at $c \in \mathbb{R}^d$ if $h(c') \geq h(c) + g^T(c' - c)$ for all $c' \in \mathbb{R}^d$, and the set of subgradients of $h(\cdot)$ at c is denoted by $\partial h(c)$. For two matrices $B_1, B_2 \in \mathbb{R}^{d\times p}$, the trace inner product is denoted by $B_1 \bullet B_2 := \text{trace}(B_1^T B_2)$. Finally, we note that the name of the framework is inspired by Farias (2007).

3. SPO Loss Functions

Herein, we introduce several loss functions that fall into the predict-then-optimize paradigm, but that are

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

also *smart* in that they take the nominal optimization problem $P(\cdot)$ into account when measuring errors in predictions. We refer to these loss functions as Smart "Predict, then Optimize" loss functions. As a starting point, let us consider a true SPO loss function that exactly measures the excess cost incurred when making a suboptimal decision due to an imprecise cost-vector prediction. Following the PO paradigm, given a cost-vector prediction $\hat{c}$, a decision $w^*(\hat{c})$ is implemented based on solving $P(\hat{c})$. After the decision $w^*(\hat{c})$ is implemented, the cost incurred is with respect to the cost vector c that is *actually realized*. The excess cost due to the fact that $w^*(\hat{c})$ may be suboptimal with respect to c is then $c^T w^*(\hat{c}) - z^*(c)$, which we call the SPO loss.

In Figure 1 *fig* : *geometry*, we show how two predicted values of c with the same prediction error can result in different decisions and different SPO losses. We consider a two-dimensional polyhedron and ellipse for the feasible region S. We plot the (negative of the) true cost vector c, as well as two candidate predictions $\hat{c}_A$ and $\hat{c}_B$ that are equidistant from c and thus have equivalent LS loss. One can see that the optimal decision for $\hat{c}_A$ coincides with that of c, since $w^*(\hat{c}_A) = w^*(c)$, and thus the SPO loss is zero. In contrast, we see that $w^*(\hat{c}_B) \neq w^*(c)$ and thus results in positive SPO loss. In the polyhedron example, any predicted cost vector whose negative is not in the gray region will result in a positive SPO loss, where as in the ellipse example any predicted cost vector that is not exactly parallel with c results in a positive SPO loss. Definition $true_{def}$ formalizes this true SPO loss associated with making the prediction $\hat{c}$ when the actual cost vector is c, given a particular oracle $w^*(\cdot)$ for $P(\cdot)$.

Definition 1 (SPO Loss)**.** Given a cost-vector prediction $\hat{c}$ and a realized cost vector c, the *true SPO loss* $\ell^{w^*}_{\text{SPO}}(\hat{c}, c)$ with respect to optimization oracle $w^*(\cdot)$ is defined as $\ell^{w^*}_{\text{SPO}}(\hat{c}, c) := c^T w^*(\hat{c}) - z^*(c)$.

Note that there is an unfortunate deficiency in Definition 1, which is the dependence on the particular oracle $w^*(\cdot)$ used to solve (2). Practically speaking, this deficiency is not a major issue because we should usually expect $w^*(\hat{c})$ to be a unique optimal solution—that is, we should expect $W^*(\hat{c})$ to be a singleton. Note that if any solution from $W^*(\hat{c})$ may be used by the loss function, then the loss function essentially becomes $\min_{w \in W^*(\hat{c})} c^T w - z^*(c)$. Thus, a prediction model would then be incentivized to always make the degenerate prediction $\hat{c} = 0$ because $W^*(0) = S$. This would then imply that the SPO loss is zero.

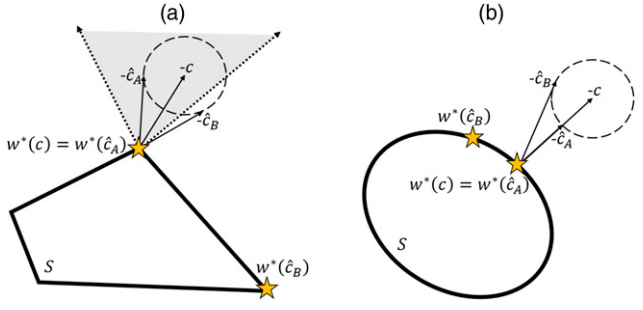
In any case, if one wishes to address the dependence on the particular oracle $w^*(\cdot)$ in Definition 1, then it is most natural to "break ties" by presuming that the implemented decision has worst-case behavior with respect to c. Definition 2 is an alternative SPO loss function that does not depend on the particular choice of the optimization oracle $w^*(\cdot)$.

Definition 2 (Unambiguous SPO Loss)**.** Given a cost-vector prediction $\hat{c}$ and a realized cost vector c, the (unambiguous) *true SPO loss* $\ell_{\text{SPO}}(\hat{c}, c)$ is defined as $\ell_{\text{SPO}}(\hat{c}, c) := \max_{w \in W^*(\hat{c})} \{c^T w\} - z^*(c)$.

Note that Definition 2 presents a version of the true SPO loss that upper bounds the version from Definition 1—that is, it holds that $\ell^{w^*}_{\text{SPO}}(\hat{c}, c) \leq \ell_{\text{SPO}}(\hat{c}, c)$ for all $\hat{c}, c \in \mathbb{R}^d$. As mentioned previously, the distinction between Definitions 1 and 2 is only relevant in degenerate cases. In the results and discussion herein, we work with the unambiguous true SPO loss given by Definition 2. Related results may often be inferred for the version of the true SPO loss given by Definition 1 by recalling that Definition 2 upper bounds Definition 1 and that the two loss functions are almost always equal, except for degenerate cases, where $W^*(\hat{c})$ has multiple optimal solutions.

Notice that $\ell_{\text{SPO}}(\hat{c}, c)$ is impervious to the scaling of $\hat{c}$; in other words, it holds that $\ell_{\text{SPO}}(\alpha\hat{c}, c) = \ell_{\text{SPO}}(\hat{c}, c)$ for all $\alpha > 0$. This property is intuitive because the true loss associated with prediction $\hat{c}$ should only depend on the optimal *solution* of $P(\cdot)$, which does not depend on the scaling of $\hat{c}$. Moreover, this property is also shared by the 0-1 loss function in binary classification problems. Namely, labels can take values in the set $\{-1, +1\}$, and the prediction model predicts values in $\mathbb{R}$. If the predicted value has the same sign as the true value, the loss is zero, and otherwise the loss is one. That is, given a predicted value $\hat{c} \in \mathbb{R}$ and a label $c \in \{-1, +1\}$, the 0-1 loss function is defined by $\ell_{0-1}(\hat{c}, c) := \mathbf{1}(\text{sgn}(\hat{c}) = c)$, where $\text{sgn}(\cdot)$ is the sign function and $\mathbf{1}(\cdot)$ is an indicator function equal to one if its input is true and zero otherwise. Therefore, the 0-1 loss function is also independent of the scale on the predictions. This similarity is not a coincidence; in fact, Proposition 1 illustrates that binary classification is a special case of the SPO framework. All proofs can be found in Online Appendix B.

Figure 1. Geometric Illustration of SPO Loss



Notes. (a) Polyhedral feasible region. (b) Elliptic feasible region.

Proposition 1 (SPO Loss Generalizes 0-1 Loss)**.** *When* $S = [-1/2, +1/2]$ *and* $c \in \{-1, +1\}$, *then* $\ell_{\text{SPO}}(\hat{c}, c) = \mathbf{1}(sgn(\hat{c}) = c)$*—that is, the SPO loss function exactly matches the* 0-1 *loss function associated with binary classification.*

Now, given the training data, we are interested in determining a cost-vector prediction model with minimal true SPO loss. Therefore, given the previous definition of the true SPO loss $\ell_{\text{SPO}}(\cdot, \cdot)$, the prediction model would be determined by following the empirical risk minimization principle as in (3), which leads to the following optimization problem:

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell_{\text{SPO}}\big(f(x_i), c_i\big). \tag{4}$$

Unfortunately, the above optimization problem is difficult to solve, both in theory and in practice. Indeed, for a fixed c, $\ell_{\text{SPO}}(\cdot, c)$ may not even be continuous in $\hat{c}$ because $w^*(\hat{c})$ (and the entire set $W^*(\hat{c})$) may not be continuous in $\hat{c}$. Moreover, because Proposition 1 demonstrates that our framework captures binary classification, solving (4) is at least as difficult as optimizing the 0-1 loss function, which may be NP-hard in many cases (Ben-David et al. 2003). We are therefore motivated to develop approaches for producing "reasonable" approximate solutions to (4) that (i) outperform standard PO approaches, and (ii) are applicable to large-scale problems where the number of training samples n and/or the dimension of the hypothesis class $\mathcal{H}$ may be very large.

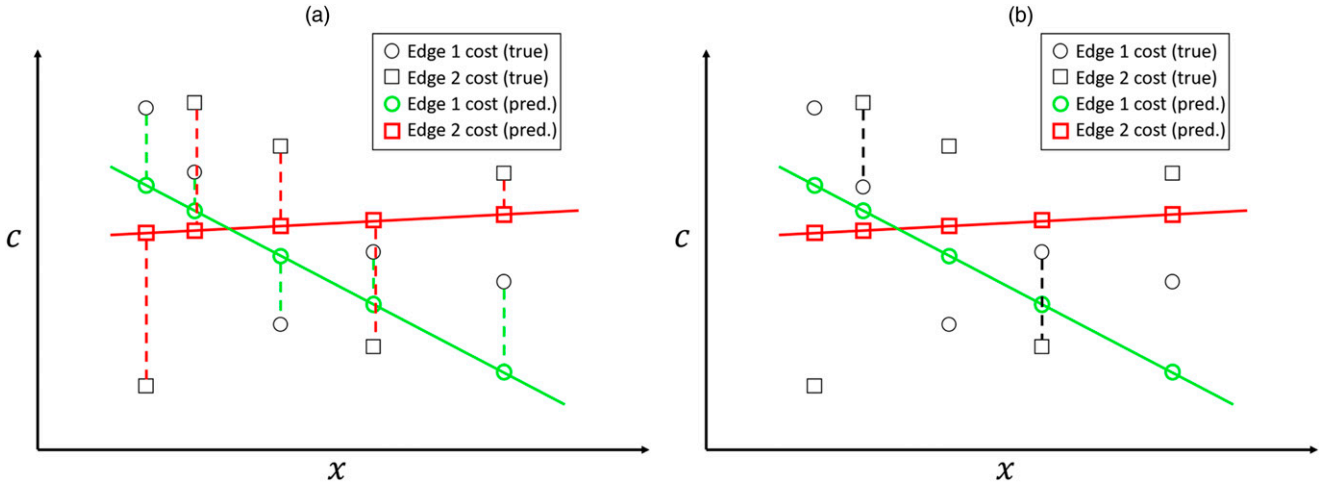
3.1. An Illustrative Example

In order to build intuition, we now compare the SPO loss against the classical least-squares loss function via an illustrative example. Consider a very simple shortest-path problem with two nodes, s and t. There are two edges that go from s to t, edge 1 and edge 2. Thus, a cost vector c is two-dimensional in this setting, and the goal is to simply choose the edge with the lower cost. We shall not observe c directly at the decision-making time, but, rather, just a one-dimensional feature x associated with the vector c. Our data consist of (x_i, c_i) pairs, and c_i are generated nonlinearly as a function of x_i.

In Figure 2(a), the residuals for the LS loss function are marked by the dashed lines. The residual is the distance between the prediction and the true value. In Figure 2(b), the residuals for the SPO loss function are marked by the dashed black lines. The residual is zero when the predicted values are in the right order. Otherwise, the residual is the distance between the true values.

The goal of the decision maker is to predict the cost of each edge from the feature by using a simple linear regression model. The intersection of the two lines (corresponding to each edge) will signal the decision boundary in the predict-then-optimize framework. The decision maker shall try both the SPO and LS loss functions to do the linear regression. In Figure 2, we illustrate the difference between LS and SPO by visualizing the residuals for one particular data set and linear models for predicting the edge 1 and edge 2 costs. In LS regression, one minimizes the sum of the residuals squared, which is denoted by the dashed green and red lines in Figure 2(a). When using SPO loss, we consider "decision residuals," which only occur when the predictions result in choosing the wrong edge. In these cases, the SPO cost is the magnitude difference between the two true costs of edge 1 and edge 2, as depicted by the dashed black lines in Figure 2(b).

Figure 2. Difference Between Prediction and Decision Residuals



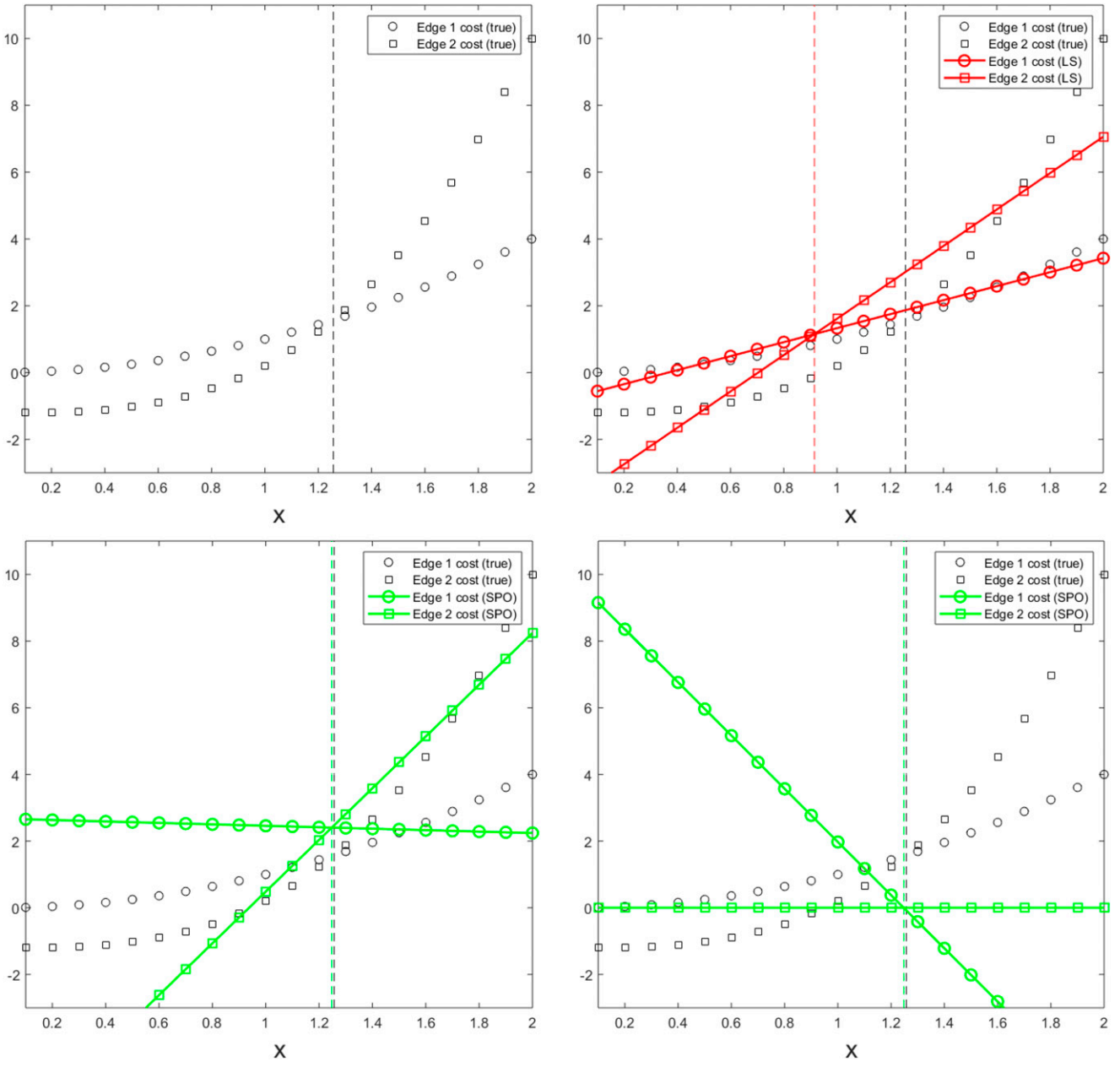
Notes. (a) Prediction residuals. (b) Decision residuals. Pred., prediction.

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

In Figure 3, we consider another data set, but this time plot the optimal LS and SPO linear regression models. In the upper left panel of Figure 3, we plot the data set and the optimal decision boundary. In the upper right panel, we plot the best LS fit to the data, and in the lower two panels, we plot two different optimal solutions to the SPO linear regression. (In fact, the SPO fitted models are also optimal for SPO+ loss, which we derive in Section 3.2.) The vertical dotted lines correspond to the decision boundaries under the true and prediction models. Note that the SPO loss in Figure 3 is zero, as there are no decision errors as described in Figure 2.

One can see from Figure 3 that the LS lines very closely approximate the nonlinear data, although the decision boundary for LS is quite far from the optimal decision boundary. For any value of x between the dotted black and red lines, the decision maker will choose the wrong edge. In contrast, the SPO lines need not approximate the data well at all, yet their decision boundary is nearly optimal. In fact, the SPO lines have zero training error, despite not fitting the data at all. The key intuition is that the SPO loss is incurred any time the wrong edge is chosen, and in this example, one can construct lines that cross at the right decision boundary, so that the wrong edge is never chosen,

Figure 3. Illustrative Example



Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

resulting in zero SPO loss. Note that the only important consideration is where the lines intersect, and, thus, the SPO linear regression does not necessarily minimize prediction error. Of course, a convex combination of SPO and LS loss may be used to overcome the unusual-looking lines generated. In fact, there are infinitely many optimal solutions to the ERM problem for the SPO loss, all of which just require that the intersection of the lines occurs between the x values of 1.2 and 1.3.

3.2. The SPO+ Loss Function

In this section, we focus on deriving a tractable surrogate loss function that reasonably approximates $\ell_{\text{SPO}}(\cdot,\cdot)$. Our surrogate function $\ell_{\text{SPO+}}(\cdot,\cdot)$, which we call the SPO+ loss function, can be derived in a few steps that we shall carefully justify below. Ideally, when finding the prediction model that minimizes the empirical risk using the SPO+ loss, this prediction model will also approximately minimize (4), the empirical risk using the SPO loss.

To begin the derivation of the SPO+ loss, we first observe that, for any $\alpha \in \mathbb{R}$, the SPO loss can be written as

$$\ell_{\text{SPO}}(\hat{c},c) = \max_{w\in W^*(\hat{c})} \{c^T w - \alpha\hat{c}^T w\} + \alpha z^*(\hat{c}) - z^*(c), \quad (5)$$

because $z^*(\hat{c}) = \hat{c}^T w$ for all $w \in W^*(\hat{c})$. Clearly, replacing the constraint $w \in W^*(\hat{c})$ with $w \in S$ in (5) results in an upper bound. Because this is true for all values of α, then

$$\ell_{\text{SPO}}(\hat{c},c) \le \inf_{\alpha}\left\{\max_{w\in S}\{c^T w - \alpha\hat{c}^T w\} + \alpha z^*(\hat{c})\right\} - z^*(c). \quad (6)$$

In fact, one can show that Inequality (6) is actually an equality using duality theory, and, moreover, the optimal value of α tends to ∞. Intuitively, one can see that as α gets large, then the term $c^T w$ in the inner maximization objective becomes negligible and the solution tends to $w^*(\alpha\hat{c}) = w^*(\hat{c})$. Thus, as α tends to ∞, the inner maximization over S can be replaced with maximization over $W^*(\hat{c})$, which recovers (5). We formalize this equivalence in Proposition 2 below.

Proposition 2 (Dual Representation of SPO Loss)**.** *For any cost-vector prediction $\hat{c} \in \mathbb{R}^d$ and realized cost vector $c \in \mathbb{R}^d$, the function $\alpha \mapsto \max_{w\in S}\{c^T w - \alpha\hat{c}^T w\} + \alpha z^*(\hat{c})$ is monotone decreasing on $\mathbb{R}$, and the true SPO loss function may be expressed as*

$$\ell_{\text{SPO}}(\hat{c},c) = \lim_{\alpha\to\infty}\left\{\max_{w\in S}\{c^T w - \alpha\hat{c}^T w\} + \alpha z^*(\hat{c})\right\} - z^*(c). \quad (7)$$

Using Proposition 2, we shall now revisit the SPO ERM Problem (4), which can be written as

$$\begin{aligned}
&\min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^n \lim_{\alpha_i\to\infty}\left\{\max_{w\in S}\{c_i^T w - \alpha_i f(x_i)^T w\}\right.\\
&\qquad\left. + \alpha_i z^*\big(f(x_i)\big)\right\} - z^*(c_i)\\
&= \min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^n \lim_{\alpha_i\to\infty}\left\{\max_{w\in S}\{c_i^T w - \alpha_i f(x_i)^T w\}\right.\\
&\qquad\left. + \alpha_i f(x_i)^T w^*\big(\alpha_i f(x_i)\big)\right\} - z^*(c_i)\\
&= \min_{f\in\mathcal{H}} \frac{1}{n}\lim_{\alpha\to\infty}\left\{\sum_{i=1}^n \max_{w\in S}\{c_i^T w - \alpha f(x_i)^T w\}\right.\\
&\qquad\left. + \alpha f(x_i)^T w^*\big(\alpha f(x_i)\big) - z^*(c_i)\right\}\\
&\le \min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^n \max_{w\in S}\{c_i^T w - 2f(x_i)^T w\}\\
&\qquad + 2f(x_i)^T w^*\big(2f(x_i)\big) - z^*(c_i), \quad (8)\\
&\le \min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^n \max_{w\in S}\{c_i^T w - 2f(x_i)^T w\} +\\
&\qquad 2f(x_i)^T w^*(c_i) - z^*(c_i). \quad (9)
\end{aligned}$$

The first equality follows from the fact that $z^*(\alpha_i f(x_i)) = \alpha_i z^*(f(x_i))$ for any $\alpha_i > 0$. The second equality follows from the observation that all of the α_i variables are tending to the same value, so we can replace them with one variable, which we call α. The first inequality follows from Proposition 2, in particular, that setting $\alpha = 2$ in (6) results in an upper bound on the SPO loss (we shall revisit this specific choice below). Finally, the second inequality follows from the fact that $w^*(c_i)$ is a feasible solution of $P(2f(x_i))$.

The summand expression in (9) is exactly what we refer to as the SPO+ loss function, which we formally state in Definition 3.

Definition 3 (SPO+ Loss)**.** Given a cost-vector prediction $\hat{c}$ and a realized cost vector c, the *SPO+ loss* is defined as $\ell_{\text{SPO+}}(\hat{c},c) := \max_{w\in S}\{c^T w - 2\hat{c}^T w\} + 2\hat{c}^T w^*(c) - z^*(c)$.

Recall that $\xi_S(\cdot)$ is the support function of S—that is, $\xi_S(c) := \max_{w\in S}\{c^T w\}$. Using this notation, the SPO+ loss may be equivalently expressed as $\ell_{\text{SPO+}}(\hat{c},c) = \xi_S(c - 2\hat{c}) + 2\hat{c}^T w^*(c) - z^*(c)$.

Before proceeding, we shall provide reasoning as to why Inequalities (8) and (9), which were used to derive SPO+, are indeed reasonable approximations. Although Inequality (8) could have been derived without the intermediary steps before it, we now claim that this inequality is actually an equality for

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

many hypothesis classes. Namely, for any hypothesis class $\mathcal{H}$, where $f \in \mathcal{H}$ implies $\alpha f \in \mathcal{H}$ for all $\alpha \geq 0$, then the inequality is tight because minimizing over αf is equivalent to minimizing over $2f$. For example, the hypothesis class of linear models satisfies this property because all scalar multiples of linear models are also linear. Note that α being absorbed into the hypothesis class was possible because the α_i terms in each summand can be replaced by a single α because they all tend to infinity. We specifically choose $\alpha = 2$ (rather than any other positive scalar) because the Bayes risk minimizer of the SPO+ loss (under some conditions) is exactly $\mathbb{E}[c|x]$ rather than a multiple of $\mathbb{E}[c|x]$. This notion will be formalized in Section 4.

The final step, (9), in the derivation of our convex surrogate SPO+ loss function involves approximating the concave (nonconvex) function $z^*(\cdot)$ with a first-order expansion. Namely, we apply the bound $z^*(2f(x_i)) = 2z^*(f(x_i)) \leq 2f(x_i)^T w^*(c_i)$, which can be viewed as a first-order approximation of $z^*(f(x_i))$ based on a supergradient computed at c_i (i.e., it holds that $w^*(c_i) \in \partial z^*(c_i)$). Note that if $f(x_i) = c_i$, then $\ell_{\text{SPO}}(f(x_i), c_i) = \ell_{\text{SPO+}}(f(x_i), c_i) = 0$, which implies that when minimizing SPO+, intuitively, we are trying to get $f(x_i)$ to be close to c_i. Therefore, one might expect $w^*(c_i)$ to be a near-optimal solution to $P(2f(x_i))$, and, thus, Inequality (9) would be a reasonable approximation. In fact, Section 4 provides a consistency property under some assumptions that would suggest the prediction $f(x_i)$ is, indeed, reasonably close to the expected value of c_i if the prediction model is trained on a sufficiently large data set.

Next, we state the following proposition, which formally shows that the SPO+ loss is an upper bound on the SPO loss, and it is convex in $\hat{c}$. Note that, although the SPO+ loss is convex in $\hat{c}$, in general, it is not differentiable because $\xi_S(\cdot)$ is not generally differentiable. However, Proposition 3 also shows that $2(w^*(c) - w^*(2\hat{c} - c))$ is a subgradient of the SPO+ loss, which is utilized in developing computational approaches in Section 5.

Proposition 3 (SPO+ Loss Properties)**.** *Given a fixed realized cost vector c, it holds that*:

1. $\ell_{\text{SPO}}(\hat{c}, c) \leq \ell_{\text{SPO+}}(\hat{c}, c)$ *for all* $\hat{c} \in \mathbb{R}^d$,
2. $\ell_{\text{SPO+}}(\hat{c}, c)$ *is a convex function of the cost-vector prediction* $\hat{c}$, *and*
3. *For any given* $\hat{c}$, $2(w^*(c) - w^*(2\hat{c} - c))$ *is a subgradient of* $\ell_{\text{SPO+}}(\cdot)$ *at* $\hat{c}$*—that is*, $2(w^*(c) - w^*(2\hat{c} - c)) \in \partial \ell_{\text{SPO+}}(\hat{c}, c)$.

The convexity of the SPO+ loss function is also shared by the hinge loss function, which is a convex upper bound for the 0-1 loss function. Recall that the hinge loss given a prediction $\hat{c}$ is $\max\{0, 1 - \hat{c}\}$ if the true label is 1 and $\max\{0, 1 + \hat{c}\}$ if the true label is -1. More concisely, the hinge loss can be written as $\max\{0, 1 - c\hat{c}\}$, where $c \in \{-1, +1\}$ is the true label. The hinge loss is central to the support-vector machine method, where it is used as a convex surrogate to minimize 0-1 loss. Recall that, in this setting of binary classification, the SPO loss exactly captures the 0-1 loss, as formalized in Proposition 1. In the same setting, it turns out that the SPO+ loss is equal to the hinge loss evaluated at $2\hat{c}$—that is, twice the predicted value—which is formalized below in Proposition 4. This mild discrepancy is due to our choice of $\alpha = 2$ in the above derivation of the SPO+ loss; the alternative choice of $\alpha = 1$ would yield the hinge loss exactly.

Proposition 4 (SPO+ Loss Generalizes Hinge Loss)**.** *Under the same conditions as Proposition 1—namely, when* $S = [-1/2, +1/2]$ *and* $c \in \{-1, +1\}$*—it holds that* $\ell_{\text{SPO+}}(\hat{c}, c) = \max\{0, 1 - 2c\hat{c}\}$*—that is, the SPO+ loss function is equivalent to the hinge loss function associated with binary classification.*

Remark 1 (Connection to Structured Prediction)**.** It is worth pointing out that the previously described construction of the SPO+ loss bears some resemblance to the construction of the structured hinge loss (Taskar et al. 2004, 2005; Tsochantaridis et al. 2005; Nowozin and Lampert 2011) in structured support vector machines. Moreover, our problem setting expands upon that of structured prediction by utilizing the objective cost of the nominal optimization problem to naturally define the SPO loss function. That is, if we define $w_i^* := w^*(c_i)$, then the modified data set $(x_1, w_1^*), (x_2, w_2^*), \ldots, (x_n, w_n^*)$ may be regarded as the training data of a structured prediction problem. However, this reduction throws away valuable information about the cost vectors c_i, whereas the SPO+ loss function naturally exploits this information and upper bounds the SPO loss. Hence, our framework (and the surrogate SPO+ loss function) may be viewed as a type of refinement of the SSVM problem (and the structured hinge loss) to settings where there is a natural cost structure. Note that both the SPO+ loss and the structured hinge loss recover the regular hinge loss of binary classification as a special case. The hinge loss satisfies a key consistency property with respect to the 0-1 loss (Steinwart 2002), which justifies its use in practice. In Section 4, we show a similar consistency result for the SPO+ loss with respect to the SPO loss under some mild conditions. On the other hand, the structured hinge loss is often *inconsistent* (see, e.g., the discussion around equation (11) in Zhang 2004), although there have been results on characterizing properties of consistent loss function in multiclass classification and structured prediction (Zhang 2004, Tewari and Bartlett 2007, Osokin et al. 2017). □

Remark 2 (When $P(\cdot)$ Is a Combinatorial or Mixed-Integer Problem)**.** As mentioned previously, the assumptions

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

that S is convex and closed are without loss of generality because one can simply replace a possibly nonconvex or nonclosed set with its closed convex hull in (2) without changing the optimal value $z^*(c)$. To be more concrete, suppose that $\tilde{S} \subseteq \mathbb{R}^d$ is a bounded, but possibly nonconvex or nonclosed, set and that S is the closed convex hull of $\tilde{S}$. Suppose further that the oracle $w^*(\cdot)$ returns an optimal solution in $\tilde{S}$—that is, $w^*(c) \in \arg\min_{w\in\tilde{S}} c^T w \subseteq \arg\min_{w\in S} c^T w$ for all $c \in \mathbb{R}^d$. For example, if $\tilde{S}$ represents the feasible region of a combinatorial or mixed-integer optimization problem, then the oracle would correspond to a practically efficient algorithm for this problem. Then, using the fact that linear optimization on $\tilde{S}$ is equivalent to linear optimization on S, it is easy to see that the SPO and SPO+ loss functions defined with respect to $\tilde{S}$ exactly equal the corresponding loss functions defined with respect to S. Finally, using Proposition 3, one can use the oracle $w^*(c) \in \arg\min_{w\in\tilde{S}} c^T w$ to compute subgradients of the SPO+ loss function, which can be utilized in computational approaches, as described in Section 5. □

Applying the ERM principle as in (4) to the SPO+ loss yields the following optimization problem for selecting the prediction model:

$$\min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \ell_{\text{SPO+}}\big(f(x_i), c_i\big). \tag{10}$$

Much of the remainder of the paper describes results concerning Problem (10). In Section 4, we demonstrate the aforementioned Fisher consistency result; in Section 5, we describe several computational approaches for solving Problem (10); and in Section 6, we demonstrate that (10) often offers superior practical performance over standard PO approaches. Next, we provide a theoretically motivated justification for using the SPO+ loss.

4. Consistency of the SPO+ Loss Function

In this section, we prove a fundamental consistency property, known as *Fisher consistency*, to describe when minimizing the SPO+ loss is equivalent to minimizing the SPO loss. The Fisher consistency of a surrogate loss function means that, under full knowledge of the data distribution and no restriction on the hypothesis class, the function that minimizes the surrogate loss also minimizes the true loss (Lin 2004, Zou et al. 2008). One may also say that the surrogate loss is calibrated with the true loss (Bartlett et al. 2006). Our result is analogous to the well-known consistency results of the hinge loss and logistic loss functions with respect to the 0-1 loss—minimizing hinge and logistic loss under full knowledge also minimizes the 0-1 loss—and provides theoretical motivation for their success in practice.

More formally, we let $\mathcal{D}$ denote the distribution of (x, c)—that is, $(x, c) \sim \mathcal{D}$—and consider the population version of the true SPO risk (Bayes risk) minimization problem:

$$\min_f \ \mathbb{E}_{(x,c)\sim\mathcal{D}}\big[\ell_{\text{SPO}}\big(f(x), c\big)\big], \tag{11}$$

and the population version of the SPO+ risk-minimization problem:

$$\min_f \ \mathbb{E}_{(x,c)\sim\mathcal{D}}\big[\ell_{\text{SPO+}}\big(f(x), c\big)\big]. \tag{12}$$

Note here that we place no restrictions on $f(\cdot)$, meaning that $\mathcal{H}$ consists of any measurable function mapping features to cost vectors.

Definition 4 (Fisher Consistency)**.** A loss function $\ell(\cdot,\cdot)$ is said to be *Fisher consistent* with respect to the SPO loss if $\arg\min_f \mathbb{E}_{(x,c)\sim\mathcal{D}}[\ell(f(x), c)]$ (the set of minimizers of the Bayes risk of ℓ) also minimizes (11).

To gain some intuition, let f^*_{SPO} and $f^*_{\text{SPO+}}$ denote any optimal solution of (11) and (12), respectively. From (1), one can see that an ideal value for $f^*_{\text{SPO}}(x)$ is simply $\mathbb{E}[c|x]$. In fact, as long as the optimal solution of $P(\mathbb{E}[c|x])$ is unique with probability one (over the distribution of $x \in \mathcal{X}$)—that is, almost surely—then it is, indeed, the case that $\mathbb{E}[c|x]$ is a minimizer of (11) (see Proposition 5). Moreover, any function that is almost surely equal to $\mathbb{E}[c|x]$ is also a minimizer of (11). In Theorem 1, we show that under Assumption 1, any minimizer of the SPO+ population risk (12) must satisfy $f^*_{\text{SPO+}}(x) = \mathbb{E}[c|x]$ almost surely and, therefore, also minimizes the SPO risk (11). In summary, the SPO+ loss is Fisher-consistent with the SPO loss, under Assumption 1.

Assumption 1. *These assumptions imply Fisher consistency of the SPO+ loss function*:

1. *Almost surely, $W^*(\mathbb{E}[c|x])$ is a singleton—that is, $\mathbb{P}_x(|W^*(\mathbb{E}[c|x])| = 1) = 1$.*
2. *For all $x \in \mathcal{X}$, the distribution of $c|x$ is centrally symmetric about its mean $\mathbb{E}[c|x]$.*
3. *For all $x \in \mathcal{X}$, the distribution of $c|x$ is continuous on all of $\mathbb{R}^d$.*
4. *The interior of the feasible region S is nonempty.*

Theorem 1 (Fisher Consistency of SPO+)**.** *Suppose Assumption 1 holds. Then, any minimizer of the SPO+ risk (12) is almost surely (over the distribution of $x \in \mathcal{X}$) equal to $\mathbb{E}[c|x]$ and is also a minimizer of the SPO risk (11). Thus, the SPO+ loss function is Fisher consistent with respect to the SPO loss.*

The key results to prove Theorem 1 are provided in Section 4.1, and the final proof is given in the online appendix. We remark that Assumption 1(1) is only needed to show that $\mathbb{E}[c|x]$ is a minimizer of the SPO risk.

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

This assumption is rather mild, as the set of points with multiple optimal solutions typically has measure zero. In fact, Assumption 1(1) can be removed if one uses Definition 1 of the SPO loss, which uses a given optimization oracle. Assumption 1(2) ensures that $\mathbb{E}[c|x]$ is a minimizer of the SPO+ risk. Note that a random vector d is centrally symmetric about its mean if $d - \mathbb{E}[d]$ is equal in distribution to $\mathbb{E}[d] - d$, or, equivalently, d is equal in distribution to $2\mathbb{E}[d] - d$. This symmetry condition is satisfied, for instance, when the data are assumed to be of the form $f(x) + \epsilon$, where ϵ is a zero-mean Gaussian distribution with a positive semidefinite covariance matrix. Finally, Assumption 1(3) and Assumption 1(4), both of which are standard, are used to show that $\mathbb{E}[c|x]$ uniquely minimizes the SPO+ risk, except possibly on a set of probability measure zero. Note that Assumption 1(2) and Assumption 1(3) may be relaxed to hold almost surely with respect to the probability measure of $x \in \mathcal{X}$, but for ease of presentation, we state them for all $x \in \mathcal{X}$. In Section 4.1, we discuss examples (provided in the online appendix) that show how our result may not hold if one of the assumptions are violated.

As mentioned previously, any minimizer for the least-squares risk is also almost surely equal to $\mathbb{E}[c|x]$, and, thus, the least-squares loss is also Fisher consistent with respect to the SPO loss. Thus, a priori, one cannot claim LS or SPO+ to be better than the other. Indeed, we have derived a natural surrogate loss function, SPO+, directly from the SPO loss that maintains a fundamental consistency property of the de facto standard LS loss function. In fact, it is easy to see that under Assumption 1, any convex combination of the LS and SPO+ loss functions is Fisher consistent. Because this consistency property applies under full distributional information and no model misspecification (no restriction on hypothesis class), we show in Section 6 that SPO+ indeed outperforms LS in several experimental settings, due to its ability to tailor the prediction to the optimization task.

4.1. Key Results to Prove Fisher Consistency

Throughout this section, we consider a nonparametric setup, where the dependence on the features x is dropped without loss of generality. To see this, first observe that the SPO risk satisfies $\mathbb{E}_{(x,c)\sim\mathcal{D}}[\ell_{\text{SPO}}(f(x), c)] = \mathbb{E}_x[\mathbb{E}_c[\ell_{\text{SPO}}(f(x), c) \mid x]]$ and likewise for the SPO+ risk. Because there is no constraint on $f(\cdot)$ (the hypothesis class consists of all prediction models), solving Problems (11) and (12) is equivalent to optimizing each function value $f(x)$ individually for all $x \in \mathcal{X}$. Therefore, for the remainder of the section, unless otherwise noted, we drop the dependence on x. Thus, we now assume that the distribution $\mathcal{D}$ is only over c, and the SPO and SPO+ risk is defined as $R_{\text{SPO}}(\hat{c}) := \mathbb{E}_c[\ell_{\text{SPO}}(\hat{c}, c)]$ and $R_{\text{SPO+}}(\hat{c}) := \mathbb{E}_c[\ell_{\text{SPO+}}(\hat{c}, c)]$, respectively. For convenience, let us define $\bar{c} := \mathbb{E}_c[c]$ (note that we are implicitly assuming that $\bar{c}$ is finite).

Next, we fully characterize the minimizers of the true SPO risk Problem (11) in this setting. Proposition 5 demonstrates that for any minimizer c^* of $R_{\text{SPO}}(\cdot)$, all of its corresponding solutions with respect to the nominal problem, $W^*(c^*)$, are also optimal solutions for $P(\bar{c})$. In other words, minimizing the true SPO risk also optimizes for the expected cost in the nominal problem (because the objective function is linear). Proposition 5 also demonstrates that the converse is true—namely, any cost-vector prediction with a unique optimal solution that also optimizes for the expected cost is also a minimizer of the true SPO risk.

Proposition 5 (SPO Minimizer)**.** *If a cost vector c^* is a minimizer of $R_{\text{SPO}}(\cdot)$, then $W^*(c^*) \subseteq W^*(\bar{c})$. Conversely, if c^* is a cost vector such that $W^*(c^*)$ is a singleton and $W^*(c^*) \subseteq W^*(\bar{c})$, then c^* is a minimizer of $R_{\text{SPO}}(\cdot)$.*

Example 2 in Online Appendix A demonstrates that, in order to ensure that c^* is a minimizer of $R_{\text{SPO}}(\cdot)$, it is not sufficient to allow c^* to be any cost vector such that $W^*(c^*) \subseteq W^*(\bar{c})$. In fact, it may not be sufficient for c^* to be $\bar{c}$. This follows from the unambiguity of the SPO loss function, which chooses a worst-case optimal solution in the event that the prediction allows for more than one optimal solution.

Next, we provide Proposition 6, which shows sufficient conditions for $\bar{c}$ to be the minimizer of the SPO+ risk and, therefore, the minimizer of the SPO risk, implying Fisher consistency. We also provide conditions for when $\bar{c}$ is the unique minimizer of the SPO+ risk, which alleviates any concern that there may be alternate minimizers of the SPO+ risk that are not Fisher consistent.

Proposition 6 (SPO+ Minimizer)**.** *Suppose that the distribution $\mathcal{D}$ of c is continuous and centrally symmetric about its mean $\bar{c}$ (i.e., c is equal in distribution to $2\bar{c} - c$).*

a. *Then, $\bar{c}$ minimizes $R_{\text{SPO+}}(\cdot)$.*

b. *In addition, suppose the interior of S is nonempty. Then, $\bar{c}$ is the unique minimizer of $R_{\text{SPO+}}(\cdot)$.*

The two important assumptions in Proposition 6 are that $\mathcal{D}$ is centrally symmetric about its mean and continuous, both of which are not individually sufficient to ensure consistency on their own. Example 3 in Online Appendix A demonstrates a situation where c is continuous on $\mathbb{R}^d$ and the minimizer of SPO+ is unique, but it does not minimize the SPO risk. Example 4 in Online Appendix A demonstrates a situation where the distribution of c is symmetric about its mean, but there exists a minimizer of the SPO+ risk that does not minimize the SPO risk. Example 5 in Online Appendix A demonstrates a case where the

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

minimizer of SPO+ is not unique if S is empty, while c is continuous and centrally symmetric about its mean.

5. Computational Approaches

In this section, we consider computational approaches for solving the SPO+ ERM Problem (10). Herein, we focus on the case of linear predictors, $\mathcal{H} = \{f : f(x) = Bx \text{ for some } B \in \mathbb{R}^{d\times p}\}$, with regularization possibly incorporated into the objective function, using the regularizer $\Omega(\cdot) : \mathbb{R}^{d\times p} \to \mathbb{R}$. (This is equivalent to working with the hypothesis class $\mathcal{H} = \{f : f(x) = Bx \text{ for some } B \in \mathbb{R}^{d\times p}, \Omega(B) \le \rho\}$ for some $\rho > 0$.) For example, we may use the ridge penalty $\Omega(B) = \frac{1}{2}\|B\|_F^2$, where $\|B\|_F$ denotes the Frobenius norm of B—that is, the entry-wise ℓ_2 norm. Other possibilities include an entry-wise ℓ_1 penalty or the nuclear norm penalty—that is, an ℓ_1 penalty on the singular values of B. In any case, these presumptions lead to the following version of (10):

$$\min_{B\in\mathbb{R}^{d\times p}} \frac{1}{n}\sum_{i=1}^{n} \ell_{\text{SPO+}}(Bx_i, c_i) + \lambda\Omega(B), \tag{13}$$

where $\lambda \ge 0$ is a regularization parameter. Because the SPO loss is convex, as stated in Proposition 3, then the above problem is a convex optimization problem as long as $\Omega(\cdot)$ is a convex function.

We mainly consider two approaches for solving Problem (13): (i) reformulations based on modeling $\ell_{\text{SPO+}}(\cdot, c)$ using duality, and (ii) stochastic gradient-based methods that instead rely only on an optimization oracle for Problem (2). The reformulation-based approach (i) requires an explicit description of the feasible region S—for example, if S is a polytope, then this approach necessitates working with an explicit list of inequality constraints describing S. On the other hand, the stochastic gradient-based approach (ii) does not require an explicit description of S and, instead, *only* relies on iteratively calling the optimization oracle $w^*(\cdot)$ in order to compute stochastic subgradients of the SPO+ loss (see Proposition 3). Therefore, it is much more straightforward to apply the stochastic gradient-descent approach to problems with complicated constraints, such as nonlinear problems, as well as combinatorial and mixed-integer problems, as mentioned in Remark 2. Although approach (i) is more restrictive in its requirements, it does offer a few advantages. Depending on the structure of S—for example, if S is a polytope with known linear inequality constraints—then approach (i) may be able to utilize off-the-shelf conic optimization solvers, such as CPLEX and Gurobi, that are capable of producing high-accuracy solutions for small to medium-sized problem instances (see Section 5.1). However, for large-scale instances, where d, p, and n might be very large, conic solvers based on interior point methods do not scale as well. Stochastic-gradient methods, on the other hand, scale much better to instances where n may be extremely large, and possibly also to instances where d and p are large, but the optimization oracle $w^*(\cdot)$ is efficiently computable due to the special structure of S. The details of the approach (ii) can be found in Online Appendix C.

5.1. Reformulation Approach

We now discuss the reformulation approach (i), which aims to recast Problem (13) in a form that is amenable to popular optimization solvers. To describe this approach, we presume that S is a polytope described by known linear inequalities—that is, $S = \{w : Aw \ge b\}$ for some given problem data $A \in \mathbb{R}^{m\times d}$ and $b \in \mathbb{R}^m$. The same approach may also be applied to particular classes of nonlinear feasible regions, although the complexity of the resulting reformulated problem will be different. The key idea is that when S is a polytope, then $\ell_{\text{SPO+}}(\cdot, c)$ is a (piecewise linear) convex function of the prediction $\hat{c}$, and, therefore, the epigraph of $\ell_{\text{SPO+}}(\cdot, c)$ can be tractably modeled with linear constraints by employing linear-programming duality. Proposition 7 formalizes this approach. (Recall that, for $w \in \mathbb{R}^d$ and $x \in \mathbb{R}^p$, wx^T denotes $d \times p$ outer product matrix where $(wx^T)_{ij} = w_i x_j$.)

Proposition 7 (Reformulation of ERM for SPO+)**.** *Suppose $S = \{w : Aw \ge b\}$ is a polytope. Then, the regularized SPO+ ERM Problem (13) is equivalent to the following optimization problem:*

$$\begin{aligned} \min_{B,p} \quad & \frac{1}{n}\sum_{i=1}^{n}\left[-b^T p_i + 2\left(w^*(c_i)x_i^T\right) \bullet B - z^*(c_i)\right] + \lambda\Omega(B) \\ \text{s.t.} \quad & A^T p_i = 2Bx_i - c_i \qquad \text{for all } i \in \{1,\dots,n\} \\ & p_i \in \mathbb{R}^m, p_i \ge 0 \qquad \text{for all } i \in \{1,\dots,n\} \\ & B \in \mathbb{R}^{d\times p}. \end{aligned} \tag{14}$$

Thus, as we can see, Problem (14) is almost a linear optimization problem—the only part that may be nonlinear is the regularizer $\Omega(\cdot)$. For several natural choices of $\Omega(\cdot)$, Problem (7) may be cast as a conic optimization problem that can be solved efficiently with interior point methods. For instance, for the LASSO penalty, where $\Omega(B) = \|B\|_1$, then (14) is equivalent to a linear program. If $\Omega(\cdot)$ is the ridge penalty, $\Omega(B) = \frac{1}{2}\|B\|_F^2$, then (14) is equivalent to a quadratic program. If $\Omega(\cdot)$ is the nuclear norm penalty, $\Omega(B) = \|B\|_*$, then (14) is equivalent to a semidefinite program.

6. Computational Experiments

In this section, we present computational results of synthetic data experiments, wherein we empirically examine the quality of the SPO+ loss function for training prediction models, using the shortest-path problem and portfolio optimization as our exemplary

Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

problem classes. Following Section 5, we focus on linear prediction models, possibly with either ridge or entrywise ℓ_1 regularization. We compare the performance of four different methods:

1. The previously described SPO+ method, (13).
2. The least-squares method that replaces the SPO+ loss function in (13) with $\ell(\hat{c}, c) = \frac{1}{2}\|\hat{c} - c\|_2^2$ and also uses regularization whenever SPO+ does.
3. An absolute loss function (i.e., ℓ_1) approach that replaces the SPO+ loss function in (13) with $\ell(\hat{c}, c) = \|\hat{c} - c\|_1$ and also uses regularization whenever SPO+ does.
4. A random-forests approach that independently trains d different random-forest models for each component of the cost vector, using standard parameter settings of $\lceil p/3 \rceil$ random features at each split and 100 trees.

Note that methods (2), (3), and (4) above do not utilize the structure of S in any way and, hence, may be viewed as independent learning algorithms with respect to each of the components of the cost vector. For methods (1), (2), and (3) above, we include an intercept column in B that is not regularized. In order to ultimately measure and compare the performance of the four different methods, we compute a "normalized" version of the SPO loss of each of the four previously trained models on an independent test set of size 10,000. Specifically, if $(\tilde{x}_1, \tilde{c}_1), (\tilde{x}_2, \tilde{c}_2), \ldots, (\tilde{x}_{n_{\text{test}}}, \tilde{c}_{n_{\text{test}}})$ denotes the test set, then we define the normalized test SPO loss of a previously trained model $\hat{f}$ by $\text{NormSPOTest}(\hat{f}) := \frac{\sum_{i=1}^{n_{\text{test}}} \ell_{\text{SPO}}(\hat{f}(\tilde{x}_i), \tilde{c}_i)}{\sum_{i=1}^{n_{\text{test}}} z^*(\tilde{c}_i)}$. Note that we naturally normalize by the total optimal cost of the test set given full information, which with high probability will be a positive number for the examples studied herein.

6.1. Shortest-Path Problem

We consider a shortest-path problem on a 5×5 grid network, where the goal is to go from the northwest corner to the southeast corner, and the edges only go south or east. In this case, the feasible region S can be modeled by using network flow constraints, as in Example 1. We utilize the reformulation approach given by Proposition 7 to solve the SPO+ training problem (13). Specifically, we use the JuMP package in Julia (Dunning et al. 2017) with the Gurobi solver to implement Problem (14). The optimization problems required in methods (2) and (3) are also solved directly by using Gurobi. In some cases, we use ℓ_1 regularization for methods (1), (2), and (3), in which case, in order to tune the regularization parameter λ, we try 10 different values of λ evenly spaced on the logarithmic scale between 10^{-6} and 100. Furthermore, we use a validation-set approach, where we train the 10 different models on a training set of size n and then use an independent validation set of size $n/4$ to pick the model that performs best with respect to the SPO loss.

6.1.1. Synthetic Data-Generation Process. Let us now describe the process used for generating the synthetic experimental data instances for both problem classes. Note that the dimension of the cost vector $d = 40$ corresponds to the total number of edges in the 5×5 grid network and that p is a given number of features. First, we generate a random matrix $B^* \in \mathbb{R}^{d \times p}$ that encodes the parameters of the true model, whereby each entry of B^* is a Bernoulli random variable that is equal to 1 with probability 0.5. We generate the training data $(x_1, c_1), (x_2, c_2), \ldots, (x_n, c_n)$ and the testing data $(\tilde{x}_1, \tilde{c}_1), (\tilde{x}_2, \tilde{c}_2), \ldots, (\tilde{x}_n, \tilde{c}_n)$ according to the following generative model:

1. First, the feature vector $x_i \in \mathbb{R}^p$ is generated from a multivariate Gaussian distribution with independent and identically distributed standard normal entries—that is, $x_i \sim N(0, I_p)$.
2. Then, the cost vector c_i is generated according to $c_{ij} = [(\frac{1}{\sqrt{p}}(B^* x_i)_j + 3)^{\text{deg}} + 1] \cdot \varepsilon_i^j$ for $j = 1, \ldots, d$, and where c_{ij} denotes the j^{th} component of c_i and $(B^* x_i)_j$ denotes the j^{th} component of $B^* x_i$. Here, deg is a fixed positive integer parameter and ε_i^j is a multiplicative noise term that is generated independently at random from the uniform distribution on $[1 - \bar{\varepsilon}, 1 + \bar{\varepsilon}]$ for some parameter $\bar{\varepsilon} \geq 0$.

Note that the model for generating the cost vectors employs a polynomial kernel function (see, e.g., Hofmann et al. 2008), whereby the regression function for the cost vector given the features—that is, $\mathbb{E}[c|x]$—is a polynomial function of x, and the parameter deg dictates the degree of the polynomial. Importantly, we still employ a linear hypothesis class for methods (1)–(3) above; hence, the parameter deg controls the amount of *model misspecification*, and, as deg increases, we expect the performance of the SPO+ approach to improve relative to methods (2) and (3). When deg = 1, the expected value of c is indeed linear in x. Furthermore, for large values of deg, the least-squares method will be sensitive to outliers in the cost-vector-generation process, which is our main motivation for also comparing against the absolute loss approach that is less sensitive to outliers. On the other hand, the random-forests method is a nonparametric learning algorithm and will accurately learn the regression function for any value of deg. However, the practical performance of random forests depends heavily on the sample size n, and, for relatively small values of n, random forests may perform poorly.

6.1.2. Results. In the following set of experiments on the shortest-path problem we described, we fix the

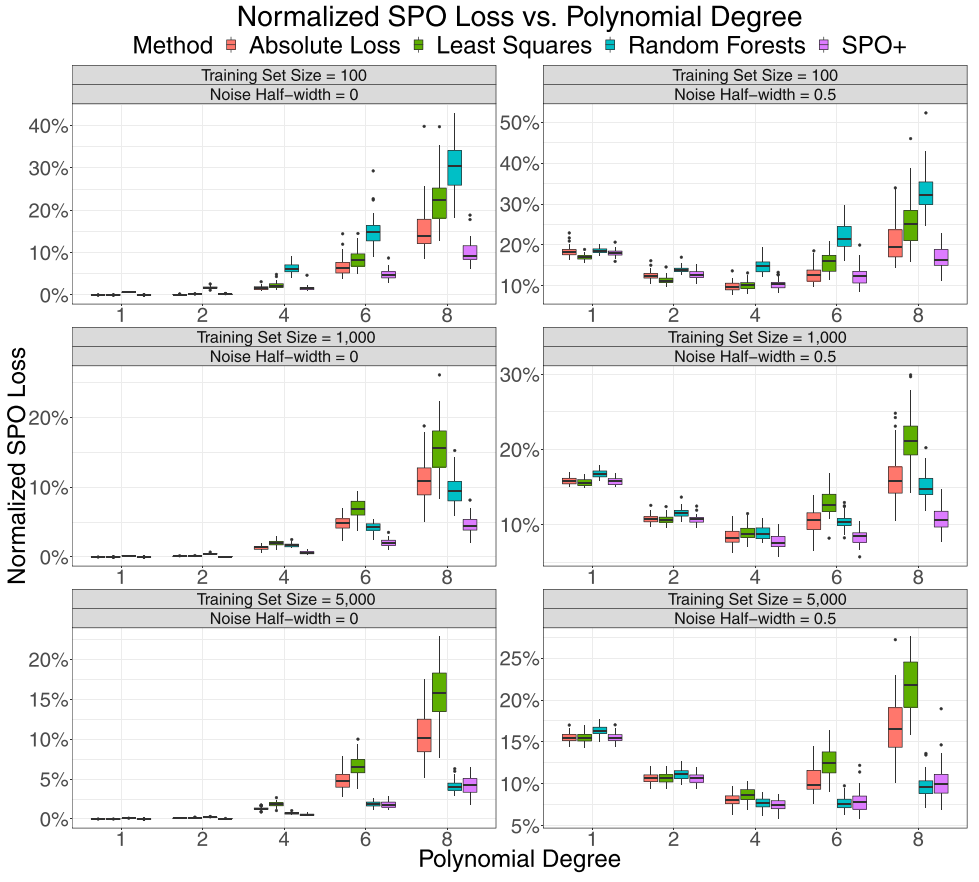
Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

number of features at $p = 5$ throughout and, as previously mentioned, use a 5×5 grid network, which implies that $d = 40$. Hence, in total, there are $pd = 200$ parameters to estimate. We vary the training-set size $n \in \{100, 1{,}000, 5{,}000\}$, we vary the parameter $\deg \in \{1, 2, 4, 6, 8\}$, and we vary the noise half-width parameter $\bar{\varepsilon} \in \{0, 0.5\}$. For every value of n, deg, and $\bar{\varepsilon}$, we run 50 simulations, each of which has a different B^*, and, therefore different ground-truth model. For the cases where $n \in \{100, 1{,}000\}$, we employ ℓ_1 regularization for methods (1)–(3), as previously described. When $n = 5{,}000$, we do not use any regularization (because it did not appear to provide any value). As mentioned previously, for each simulation, we evaluate the performance of the trained models by computing the normalized SPO loss on a test set of 10,000 samples. The computation time for solving one ERM problem using the SPO+ loss is approximately 0.5–1.0 seconds, 5–30 seconds, and 1–15 minutes for $n \in \{100, 1{,}000, 5{,}000\}$, respectively. The other methods can be solved in a few seconds by using well-developed packages. Figure 4 summarizes our findings, and note that the box plot for each configuration of the parameters is across the 50 independent trials.

From Figure 4, we can see that for small values of the deg parameter—that is, $\deg \in \{1, 2\}$—the absolute-loss, least-squares, and SPO+ methods perform comparably, with the least-squares method slightly dominating in the case of noise with $\bar{\varepsilon} = 0.5$. The slight dominance of least squares (and sometimes the absolute loss as well) in these cases might be explained by some inherent robustness properties of the least-squares loss. It is also plausible that, because the SPO+ loss function is more intricate than the "simple" least-squares loss function, it may overfit in situations with noise and a small training-set size. On the other hand, as the parameter deg grows and the degree of model misspecification increases, then the SPO+ approach generally begins to perform best across all instances, except when $n = 5{,}000$, in which case random forests performs comparably to SPO+. This behavior suggests that the SPO+ loss is better than the competitors at leveraging additional data and stronger nonlinear signals.

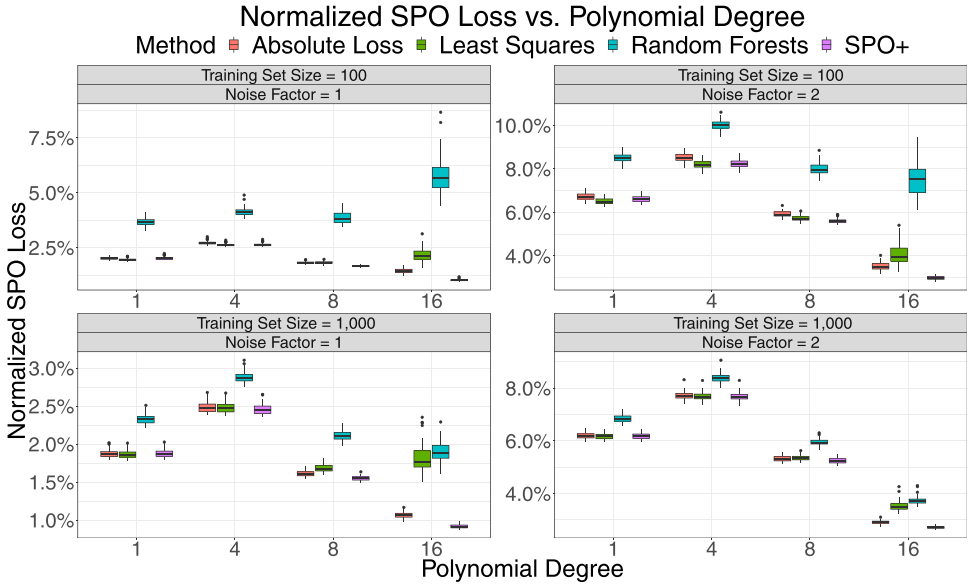
It is interesting to point out that random forests generally does not perform well, except when $n = 5{,}000$, in which case it performs comparably to SPO+, which uses a much simpler linear hypothesis class.

Figure 4. Normalized Test Set SPO Loss for the SPO+, Least Squares, Absolute Loss, and Random-Forests Methods on Shortest-Path Problem Instances



Published in Management Science on March 12, 2021 as DOI: 10.1287/mnsc.2020.3922.
This article has not been copyedited or formatted. The final version may differ from this version.

Figure 5. Normalized Test Set SPO Loss for the SPO+, Least-Squares, Absolute-Loss, and Random-Forests Methods on Portfolio-Optimization Instances



Indeed, when $n \in \{100, 1{,}000\}$, random forests almost always performs worst, except for when $n = 1{,}000$ and $\deg \in \{6, 8\}$, in which case random forests outperforms least squares, performs comparably to the absolute-loss method, and is strongly dominated by SPO+. Indeed, the cases where $n \in \{1{,}000, 5{,}000\}$ and $\deg \in \{6, 8\}$ suggest that least squares is prone to outliers, whereas the absolute loss is not, random forests is slow to converge due to its nonparametric nature, and SPO+ is best able to adapt to the large degree of model misspecification, even with a modest amount of data (i.e., $n = 1{,}000$).

6.2. Portfolio Optimization

Here, we consider a simple portfolio-selection problem based on the classical Markowitz model (Markowitz 1952). As discussed in Section 1, we presume that there are auxiliary features that may be used to predict the returns of d different assets, but that the covariance matrix of the asset returns *does not depend* on the auxiliary features. Therefore, we consider a model with a constraint that bounds the overall variance of the portfolio. Specifically, if $\Sigma \in \mathbb{R}^{d \times d}$ denotes the (positive semidefinite) covariance matrix of the asset returns and $\gamma \geq 0$ is the desired bound on the overall variance (risk level) of the portfolio, then the feasible region S in (2) is given by $S := \{w : w^T \Sigma w \leq \gamma, e^T w \leq 1, w \geq 0\}$. Here, e denotes the vector of all ones and because we only require that $e^T w \leq 1$, the cost vector c in (2) represents the negative of the incremental returns of the assets above the risk-free rate. In other words, it holds that $c = -\tilde{r}$, where $\tilde{r} = r - r_{\text{RF}} e$, r represents the vector of asset returns, and r_{RF} is the risk-free rate. We use the SGD approach (Algorithm 1 of Online Appendix C) for training the SPO+ model of method (1). Training the SPO+ model takes three to five minutes for each ERM instance, whereas the other methods typically take less than a second. For brevity, we defer the details of the experimental setup to Online Appendix D.

Figure 5 displays our results for this experiment. Generally, we observe similar patterns as in the shortest-path experiment, although comparatively larger values of deg are needed to demonstrate the relative superiority of SPO+. In summary, across all of our experiments, our results indicate that as long as there is some degree of model misspecification, then SPO+ tends to offer significant value over competing approaches, and this value is further strengthened in cases where more data are available. The SPO+ approach is either always close to the best approach or dominating all other approaches, making it a fairly suitable choice across all parameter regimes.

7. Conclusion

In this paper, we provide a new framework for developing prediction models under the predict-then-optimize paradigm. Our SPO framework relies on new types of loss functions that explicitly incorporate the problem structure of the optimization problem of interest. Our framework applies for any problem with a linear objective, even when there are integer constraints.

Because the SPO loss function is nonconvex, we also derived the convex SPO+ loss function using several logical steps based on duality theory. Moreover, we prove that the SPO+ loss is consistent with respect to the SPO loss, which is a fundamental property of any loss function. In fact, our results also

directly imply that the least-squares loss function is also consistent with respect to the SPO loss. Thus, least squares performs well when the ground truth is near linear, although, at least empirically, SPO+ strongly outperforms all approaches when there is model misspecification. In subsequent work, we have shown how to train decision trees with SPO loss (Elmachtoub et al. 2020) and developed generalization bounds of the SPO loss function (El Balghiti et al. 2019). Naturally, there are many important directions to consider for future work, including more empirical testing and case studies, handling unknown parameters in the constraints, and dealing with nonlinear objectives.

References

Ahuja RK, Magnanti TL, Orlin JB (1993) *Network Flows: Theory, Algorithms, and Applications* (Pearson, Upper Saddle River, NJ).

Angalakudati M, Balwani S, Calzada J, Chatterjee B, Perakis G, Raad N, Uichanco J (2014) Business analytics for flexible resource allocation under random emergencies. *Management Sci.* 60(6): 1552–1573.

Aswani A, Shen Z-J, Siddiq A (2018) Inverse optimization with noisy data. *Oper. Res.* 66(3):870–892.

Balkanski E, Rubinstein A, Singer Y (2016) The power of optimization from samples. *Adv. Neural Inform. Processing Systems* 29: 4017–4025.

Balkanski E, Rubinstein A, Singer Y (2017) The limitations of optimization from samples. *Proc. 49th Annu. ACM SIGACT Sympos. Theory Comput.* (Association for Computing Machinery, New York), 1016–1027.

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

Ban G-Y, El Karoui N, Lim AEB (2018) Machine learning and portfolio optimization. *Management Sci.* 64(3):1136–1154.

Bartlett PL, Jordan MI, McAuliffe JD (2006) Convexity, classification, and risk bounds. *J. Amer. Statist. Assoc.* 101(473):138–156.

Ben-David S, Eiron N, Long PM (2003) On the difficulty of approximately maximizing agreements. *J. Comput. System Sci.* 66(3): 496–514.

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

Bertsimas D, Thiele A (2006) Robust and data-driven optimization: modern decision making under uncertainty. Johnson MP, Norman B, Secomandi N, eds. *Models, Methods, and Applications for Innovative Decision Making*, INFORMS TutORials in Operations Research (INFORMS, Catonsville, MD), 95–122.

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

Bertsimas D, Gupta V, Kallus N (2018b) Robust sample average approximation. *Math. Programming* 171(1-2):217–282.

Bertsimas D, Gupta V, Paschalidis IC (2015) Data-driven estimation in equilibrium using inverse optimization. *Math. Programming* 153(2):595–633.

Besbes O, Gur Y, Zeevi A (2015) Optimization in online content recommendation services: Beyond click-through rates. *Manufacturing Service Oper. Management* 18(1):15–33.

Besbes O, Phillips R, Zeevi A (2010) Testing the validity of a demand model: An operations perspective. *Manufacturing Service Oper. Management* 12(1):162–183.

Chan CW, Farias VF, Bambos N, Escobar GJ (2012) Optimizing intensive care unit discharge decisions with patient readmissions. *Oper. Res.* 60(6):1323–1341.

Chan CW, Green LV, Lu Y, Leahy N, Yurt R (2013) Prioritizing burn-injured patients during a disaster. *Manufacturing Service Oper. Management* 15(2):170–190.

Chan TCY, Craig T, Lee T, Sharpe MB (2014) Generalized inverse multiobjective optimization with application to cancer therapy. *Oper. Res.* 62(3):680–695.

Cheung M, Elmachtoub AN, Levi R, Shmoys DB (2016) The submodular joint replenishment problem. *Math. Programming* 158(1-2): 207–233.

Chu LY, Shanthikumar JG, Shen Z-JM (2008) Solving operational statistics via a Bayesian analysis. *Oper. Res. Lett.* 36(1):110–116.

Cohen MC, Leung N-HZ, Panchamgam K, Perakis G, Smith A (2017) The impact of linear optimization on promotion planning. *Oper. Res.* 65(2):446–468.

Den Boer AV, Sierag DD (2020) Decision-based model selection. *Eur. J. Oper. Res.* 290(2):671–686.

den Hertog D, Postek K (2016) Bridging the gap between predictive and prescriptive analytics-new optimization methodology needed. Preprint, submitted December 9, http://www.optimization-online.org/DB_HTML/2016/12/5779.html.

Deng Y, Liu J, Sen S (2018) Coalescing data and decision sciences for analytics. Gel E, Ntaimo L, eds. *Recent Advances in Optimization and Modeling of Contemporary Problems*, INFORMS TutORials in Operations Research (INFORMS, Catonsville, MD), 20–49.

Deo S, Rajaram K, Rath S, Karmarkar US, Goetz MB (2015) Planning for HIV screening, testing, and care at the Veterans Health Administration. *Oper. Res.* 63(2):287–304.

Donti P, Amos B, Kolter JZ (2017) Task-based end-to-end model learning in stochastic optimization. *Adv. Neural Inform. Processing Systems.* 30:5484–5494.

Dunning I, Huchette J, Lubin M (2017) Jump: A modeling language for mathematical optimization. *SIAM Rev.* 59(2):295–320.

El Balghiti O, Elmachtoub AN, Grigas P, Tewari A (2019) Generalization bounds in the predict-then-optimize framework. *Adv. Neural Inform. Processing Systems* 32:14412–14421.

Elmachtoub AN, Jason CNL, McNellis R (2020) Decision trees for decision-making under the predict-then-optimize framework. *Proc. 37th Internat. Conf. Machine Learn.* (PMLR), 2858–2867.

Esfahani PM, Shafieezadeh-Abadeh S, Grani A, Hanasusanto DK (2018) Data-driven inverse optimization with imperfect information. *Math. Programming* 167(1):191–234.

Farias V (2007) Revenue management beyond estimate, then optimize. Unpublished doctoral thesis, Stanford University, Stanford, CA.

Ferreira KJ, Bin HAL, Simchi-Levi D (2015) Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing Service Oper. Management* 18(1):69–88.

Gallien J, Mersereau AJ, Garro A, Mora AD, Vidal MN (2015) Initial shipment decisions for new products at Zara. *Oper. Res.* 63(2): 269–286.

Gupta V, Rusmevichientong P (2017) Small-data, large-scale linear optimization with uncertain objectives. Preprint, submitted October 31, https://dx.doi.org/10.2139/ssrn.3065655.

Hofmann T, Schölkopf B, Smola AJ (2008) Kernel methods in machine learning. *Ann. Statist.* 36(3):1171–1220.

Jaggi M (2011) Convex optimization without projection steps. Preprint, submitted August 4, https://arxiv.org/abs/1108.1170.

Kao Y-h, Roy BV, Yan X (2009) Directed regression. *Adv. Neural Inform. Processing Systems* 22:889–897.

Keshavarz A, Wang Y, Boyd S (2011) Imputing a convex objective function. *2011 IEEE Internat. Sympos. Intelligent Control (ISIC)* (IEEE, Piscataway, NJ), 613–619.

Kleywegt AJ, Shapiro A, Homem-de Mello T (2002) The sample average approximation method for stochastic discrete optimization. *SIAM J. Optim.* 12(2):479–502.

Levi R, Roundy RO, Shmoys DB (2006) Primal-dual algorithms for deterministic inventory problems. *Math. Oper. Res.* 31(2):267–284.

Lim AEB, Shanthikumar JG, Vahn G-Y (2012) Robust portfolio choice with learning in the framework of regret: Single-period case. *Management Sci.* 58(9):1732–1746.

Lin Y (2004) A note on margin-based loss functions in classification. *Statist. Probab. Lett.* 68(1):73–82.

Liyanage LH, Shanthikumar JG (2005) A practical inventory control policy using operational statistics. *Oper. Res. Lett.* 33(4):341–348.

Markowitz H (1952) Portfolio selection. *J. Finance* 7(1):77–91.

Mehrotra M, Dawande M, Gavirneni S, Demirci M, Tayur S (2011) OR practice—production planning with patterns: A problem from processed food manufacturing. *Oper. Res.* 59(2):267–282.

Mišić VV, Perakis G (2020) Data analytics in operations management: A review. *Manufacturing Service Oper. Management* 22(1): 158–169.

Nowozin S, Lampert CH (2011) Structured learning and prediction in computer vision. *Foundations Trends Comput. Graphics Vision.* 6(3–4):185–365.

Osokin A, Bach F, Lacoste-Julien S (2017) On structured prediction theory with calibrated convex surrogate losses. Von Luxburg U, Guyon I, Bengio S, Wallach H, Fergus R, eds. *NIPS'17 Proc. 31st Internat. Conf. Neural Inform. Processing Systems* (Curran Associates, Red Hook, NY), 302–313.

Schütz P, Tomasgard A, Ahmed S (2009) Supply chain design under uncertainty using sample average approximation and dual decomposition. *Eur. J. Oper. Res.* 199(2):409–419.

Sen S, Deng Y (2017) Learning enabled optimization: Toward a fusion of statistical learning and stochastic optimization. Preprint, submitted March 14, http://www.optimization-online.org/DB_HTML/2017/03/5904.html.

Simchi-Levi D (2013) OM forum—OM research: From problem-driven to data-driven research. *Manufacturing Service Oper. Management* 16(1):2–10.

Steinwart I (2002) Support vector machines are universally consistent. *J. Complexity* 18(3):768–791.

Taskar B, Guestrin C, Koller D (2004) Max-margin Markov networks. *Adv. Neural Inform. Processing Systems* 16:25–32.

Taskar B, Chatalbashev V, Koller D, Guestrin C (2005) Learning structured prediction models: A large margin approach. *Proc. 22nd Internat. Conf. Machine Learn.* (Association for Computing Machinery, New York), 896–903.

Tewari A, Bartlett PL (2007) On the consistency of multiclass classification methods. *J. Machine Learn. Res.* 8(May):1007–1025.

Tsochantaridis I, Joachims T, Hofmann T, Altun Y (2005) Large margin methods for structured and interdependent output variables. *J. Machine Learn. Res.* 6(Sep):1453–1484.

Tulabandhula T, Rudin C (2013) Machine learning with operational costs. *J. Machine Learn. Res.* 14(1):1989–2028.

Wagner HM, Whitin TM (1958) Dynamic version of the economic lot size model. *Management Sci.* 5(1):89–96.

Wang Z, Glynn PW, Ye Y (2016) Likelihood robust optimization for data-driven problems. *Comput. Management Sci.* 13(2): 241–261.

Zhang T (2004) Statistical analysis of some multi-category large margin classification methods. *J. Machine Learn. Res.* 5(Oct): 1225–1251.

Zou H, Zhu J, Hastie T (2008) New multicategory boosting algorithms based on multicategory Fisher-consistent losses. *Ann. Appl. Statist.* 2(4):1290–1306.