SUFFICIENT DECISION PROXIES FOR DECISION-FOCUSED LEARNING

A PREPRINT

Noah Schutte
Delft University of Technology
n.j.schutte@tudelft.nl

Grigorii Veviurko
Delft University of Technology
g.veviurko@tudelft.nl

Krzysztof Postek
Independent Researcher
krzysztof.postek@gmail.com

Neil Yorke-Smith
Delft University of Technology
n.yorke-smith@tudelft.nl

ABSTRACT

When solving optimization problems under uncertainty with contextual data, utilizing machine learning to predict the uncertain parameters is a popular and effective approach. Decision-focused learning (DFL) aims at learning a predictive model such that decision quality, instead of prediction accuracy, is maximized. Common practice here is to predict a single value for each uncertain parameter, implicitly assuming that there exists a (single-scenario) deterministic problem approximation (proxy) that is sufficient to obtain an optimal decision. Other work assumes the opposite, where the underlying distribution needs to be estimated. However, little is known about when either choice is valid. This paper investigates for the first time problem properties that justify using either assumption. Using this, we present effective decision proxies for DFL, with very limited compromise on the complexity of the learning task. We show the effectiveness of presented approaches in experiments on problems with continuous and discrete variables, as well as uncertainty in the objective function and in the constraints.

1 Introduction

Decision making under uncertainty is encountered everywhere. Imagine packing your suitcase or finding the best travel route when going to a conference. To better make these decisions, we use available data that correlate with what is uncertain: e.g. we use seasonal weather patterns to predict the weather and decide if an umbrella is needed. We are trying to solve a contextual optimization or 'predict and optimize' problem: making predictions based on contextual information that we then use for optimization (decision making). Since our goal is to make the best decision, we do not care about the predictions by themselves, but only about the decisions they lead to. This is the main premise of *decision-focused learning* (DFL): learning a predictive model that is optimized for decision quality instead of prediction accuracy.

Observing contextual information in the form of feature values $z \in \mathbb{R}^n$, the goal of the decision maker is to find optimal decision (solution) x from constrained set X by solving the following problem:

$$x^*(z) = \operatorname*{argmin}_{x \in X} \mathbb{E}_{c \sim \mathcal{C}_z}[f(c, x)|z], \tag{1}$$

where distribution $\mathcal{C}_z$ is unknown. In this setting, we can learn a predictive model $\phi_\theta(\cdot)$ for uncertain c, and use it to get a single-valued prediction, with the 'estimated scenario' $\hat{c} = \phi_\theta(z)$ when contextual information z is observed. We do this because we would like to predict the realized value of c, i.e., the value that c will take in this empirical observation. If we do this successfully, we will get the empirical optimal decision by solving the deterministic approximation (proxy) of the true optimization problem 1: $\arg\min_{x \in X} f(\hat{c}, x)$. However, since the true c is stochastic and its distribution $\mathcal{C}_z$

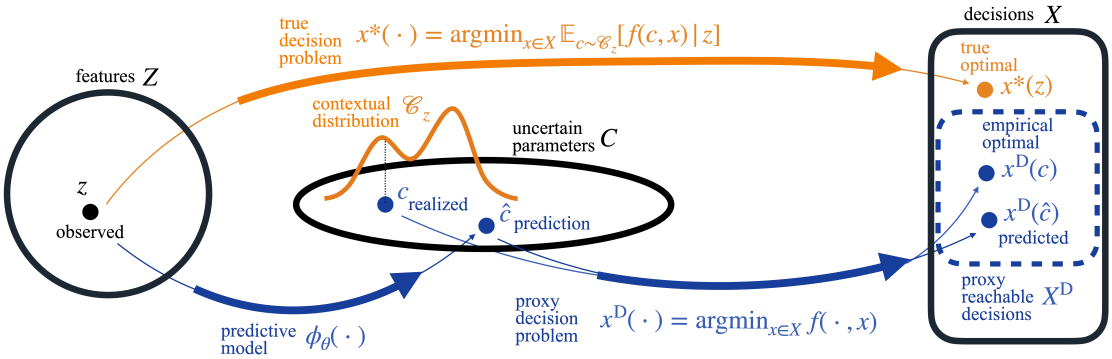


Figure 1: Schematic representation of contextual optimization and the difference between the true optimal decision based on the unknown contextual distribution (orange) and the empirical optimal decision based on the deterministic proxy (blue)

is unknown, the empirical optimal decision is different from the *true* optimal decision based on Equation 1. Still, the deterministic proxy is a natural choice. This is also because it is proven to be able to lead to true optimal decisions, for example, when $\mathbb{E}[f(c,x)] = f(\mathbb{E}[c], x)$, which happens when uncertainty is linear in the objective (Elmachtoub and Grigas, 2022); or, recently shown for certain fixed recourse, fixed costs two-stage stochastic problems (Homem-de-Mello et al., 2024).

We identify the following gap in the literature: *When is a deterministic proxy provably sub-optimal, and how can we efficiently apply DFL in this case?* In this work we contribute the following:

1. We provide sufficient conditions for (deterministic) decision proxies to be able to lead to an optimal decision and present problem-classes where these conditions are violated.
2. We present sufficient and efficient, i.e., computationally cheap, alternative decision proxies.
3. We analyze continuous and discrete problems where the sufficient conditions do not hold for deterministic proxies, and show the effectiveness of the proposed proxies.

The remainder of this paper is organized as follows. We first provide more background on DFL. In Sect. 3 we provide intuition and theory on when a deterministic proxy is sub-optimal. Then we present alternative approaches for when this occurs in Sect. 4, followed by an empirical analysis in Sect. 5.

2 Problem formulation

The aim in a contextual optimization setting is to go from context z to optimal decision $x^*(z)$, by finding some mapping, or policy $\pi : Z \to X$. Due to the structure of the (potentially combinatorial) optimization problem $x^*(\cdot)$, it is challenging to learn this complete end-to-end mapping directly Bengio et al. (2021). Considering it as separate prediction and optimization problems (predict-then-optimize) is practical, as both types of problems have been studied extensively Sadana et al. (2025). From the prediction problem perspective, assuming data with feature-label pairs z, c available, a typical learning approach would be to train predictor $\phi_\theta : Z \to C$ by minimizing the predictive error in the data (e.g., mean squared error). We will refer to this approach as *prediction-focused learning* (PFL). By contrast, DFL aims at learning a predictive model that minimizes decision error, taking the downstream optimization problem into account. Decision error is considered some measure of the difference between empirical optimal decision $x^{\mathrm{D}}(c)$ and the obtained decision based on the prediction $x^{\mathrm{D}}(\hat{c})$ (e.g., regret (Teso et al., 2022)). Here $x^{\mathrm{D}} : C \to X$ is the *deterministic proxy* of the stochastic optimization problem $x^*(\cdot)$ from (1):

$$x^{\mathrm{D}}(c) = \operatorname*{argmin}_{x \in X} f(c, x).$$

Both learning methods lead to a policy $\pi^{\mathrm{D}} = x^{\mathrm{D}} \circ \phi_\theta$. It is natural to consider this deterministic proxy, since in practice we only have data available and therefore do not have $\mathcal{C}_z$ to compute $x^*(z)$ directly. We observe realized pairs z, c, that have an accompanied empirical optimal $x^{\mathrm{D}}(c)$. However, at inference time $x^*(z)$ is the optimal decision, and there is no guarantee that $x^{\mathrm{D}}(\cdot)$ is able to equal $x^*(z)$. Proxy $x^{\mathrm{D}}(\cdot)$ can be limited to a subset of the feasible space $X^{\mathrm{D}} \subset X$ due to its structure, which means that $x^*(z)$ is unattainable if it lies outside X^{D}. Fig. 1 shows a schematic representation of this potential problem. We give an illustrative example.

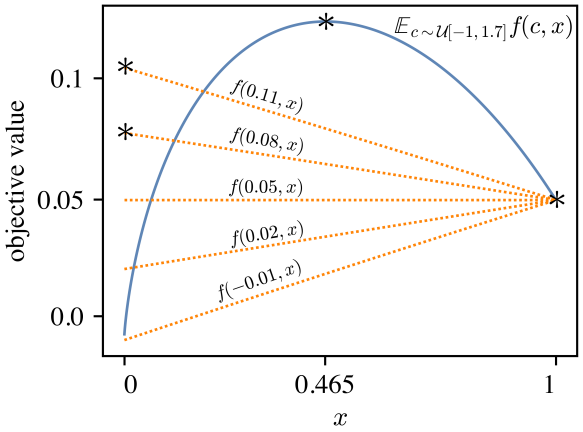


Figure 2: True objective function $\mathbb{E}_{c\sim\mathcal{U}[-1,1.7]}f(c,x)$ compared to deterministic proxy objectives $f(\hat{c},x)$ for Example 1. $*$ denote different function maxima.

Example 1. *(Based on the stock-exchange paradox by Kibzun and Kan (1997)). Assume we have some capital that we want to grow in the coming t years. Every year, we can deposit capital in a bank with fixed rate of return $\beta = 0.05$ and we can purchase shares with an uncertain rate of return $c \sim \mathcal{U}[-1, \alpha = 1.7]$ (continuous uniformly distributed). If our aim is to maximize the next period's return, we will invest our whole capital in shares, since $\mathbb{E}[c] = 0.35 > 0.05$. However, this leads to a negative* ***expected growth rate****:*

$$\mathbb{E}[\ln(1+c)] = \ln(\alpha+1) - 1 = \ln(2.7) - 1 < 0,$$

Because of this, investing all capital in shares every period will make the probability of ruin, i.e., losing all capital at some point, converge to 1.

This is considered the stock-exchange paradox, where the Kelly strategy (Kelly, 1956) is being considered as solving:

$$x^* = \underset{0\leq x\leq 1}{\operatorname{argmax}}\, \mathbb{E}[\ln(1+\beta x + c(1-x))].$$

Given $\alpha = 1.7$ and $\beta = 0.05$, we get $x^* \approx 0.465$.

Now, if we look at the deterministic proxy of the problem, it will soon become clear that we cannot find the optimal decision:

$$x^D(\hat{c}) = \underset{0\leq x\leq 1}{\operatorname{argmax}}\, \ln(1+\beta x + \hat{c}(1-x))].$$

Since $\ln(\cdot)$ is a monotonically increasing function, we get $x^D(\hat{c}) = 1$ when $\hat{c} < \beta$, $x^D(\hat{c}) = 0$ when $\hat{c} > \beta$, and $x^D(\hat{c}) = [0,1]$ when $\hat{c} = \beta$). Hence $\forall \hat{c} : x^D(\hat{c}) \in \{0,1\}$ almost surely, and true optimal decision x^* cannot be obtained. Fig. 2 shows a visual representation of this example.

2.1 The contextual value of the stochastic solution

Before we provide proofs for sub-optimality of deterministic proxies, we introduce the concept of *value of the stochastic solution* Birge (1982) in a contextual optimization setting. This concept is introduced as a measure for the value of solving the true stochastic optimization problem compared to solving the deterministic version with the uncertain variables replaced by their expectation. This is useful because in practice it is both harder to estimate the uncertain variables distribution (compared to its expectation) and more complex to solve a stochastic optimization problem. In this setting it gives us some clear structure to prove strict differences between solving a deterministic or stochastic contextual optimization problem.

First, let us define the value of the true optimal decision based on Equation 1:

$$\mathrm{V}^* = \min_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x)|z] = \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x^*(z))|z]$$

Now we look at the value of a decision obtained by using PFL. The predictive error is minimized in PFL if we are able to exactly estimate $\bar{c}_z = \mathbb{E}_{c\sim\mathcal{C}_z}[c|z]$, and obtain:

$$\mathrm{V}_{\text{PFL}} = \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\bar{c}_z),)|z]$$

The value of the stochastic solution as defined by Birge (1982) is $\mathrm{V}_{\text{PFL}} - \mathrm{V}^*$, with the only difference being the added context z. Taking a DFL perspective, we aim to find predictions that minimize decision error. This means that we aim to find the following value:

$$\mathrm{V}_{\text{DFL}} = \min_{\hat{c}_z \in C} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\hat{c}_z))|z]$$

We have $\mathrm{V}^* \leq \mathrm{V}_{\text{DFL}} \leq \mathrm{V}_{\text{PFL}}$. The first inequality holds because in V^* we directly minimize over the input of $f(c, \cdot)$, $x \in X$, compared to minimizing over the argument of a function with codomain X, resulting into the same input of $f(c, \cdot)$. The second inequality holds because we take minimizer $\hat{c}_z \in C$ instead of $\bar{c}_z \in C$. The value of using DFL over PFL $\mathrm{V}_{\text{PFL}} - \mathrm{V}_{\text{DFL}}$ is highly relevant for predict-then-optimize problems. A general consensus is that this value increases with predictive error Mandi et al. (2024), as the higher the predictive error the more important it is to learn such that decision quality is least affected. Additionally, Cameron et al. (2022) show that $\mathrm{V}_{\text{PFL}} - \mathrm{V}_{\text{DFL}}$ is also dependent on dependencies between multiple uncertain variables: Predicting marginal expectations using PFL can cause theoretically unbounded worse performance compared to DFL.

In this work we are interested in the difference between the value of using DFL with a deterministic proxy V_{DFL} and true optimal value V^*: $\mathrm{V}_{\text{DFL}} - \mathrm{V}^*$. This difference has not been studied in literature. Due to DFL's end-to-end learning approach, proxy $x^{\mathrm{D}}(\cdot)$ has not been considered as limiting in its performance even in more complex stochastic settings Silvestri et al. (2023); van den Houten et al. (2024). In the following we will identify cases when $\mathrm{V}_{\text{DFL}} - \mathrm{V}^* > 0$.

3 Sub-optimality of the deterministic decision proxy

This section presents cases where one can prove that $\mathrm{V}_{\text{DFL}} > \mathrm{V}^*$, i.e., DFL with a deterministic proxy is sub-optimal. We first present theorems for the cases when $\mathrm{V}^* = \mathrm{V}_{\text{DFL}} = \mathrm{V}_{\text{PFL}}$ and $\mathrm{V}^* = \mathrm{V}_{\text{DFL}} < \mathrm{V}_{\text{PFL}}$. Proofs are presented in the Appendix.

Theorem 1. *There exists at least one optimal single-scenario w.r.t. Equation 1 if* $\forall z \in \mathbb{R}^n$, $\mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x)|z] = f(\mathbb{E}_{c\sim\mathcal{C}_z}[c|z], x)$. *An optimal single-scenario is* $\bar{c}_z = \mathbb{E}_{c\sim\mathcal{C}_z}[c|z]$. *We have* $V^* = V_{DFL} = V_{PFL}$.

Theorem 1 shows that $\mathrm{V}^* = \mathrm{V}_{\text{DFL}} = \mathrm{V}_{\text{PFL}}$ when Jensen's inequality renders an equality, for example when $f(c, x)$ is linear in c, $\forall x \in X$. PFL becomes sub-optimal if Jensen's inequality is strict such that both function have different minimizers:

Theorem 2. *If* $\operatorname{argmin}_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x)|z] \neq \operatorname{argmin}_{x\in X} f(\mathbb{E}_{c\sim\mathcal{C}_z}[c|z], x)$, *we have* $V^* < V_{PFL}$[1].

We now present a sufficient condition for when an optimal single-scenario exists.

Theorem 3. *Given* x, *there exists an optimal single-scenario w.r.t. Equation 1 if* $x^D(\cdot) : C \to X$ *is surjective, i.e.,* $\forall x \in X, \exists \hat{c} \in C : x^D(\hat{c}) = x$. *We have* $V^* = V_{DFL}$.

In other words: When x^{D} is surjective, it is flexible enough to attain any feasible decision $x \in X$, so also optimal $x^*(z)$. Theorem 2 and Theorem 3 demonstrate a theoretical case where PFL is inferior to DFL. The result that $\mathrm{V}^* = \mathrm{V}_{\text{DFL}}$, based on $x^{\mathrm{D}}(\cdot)$ being surjective, shows the flexibility of DFL: it allows for learning predictions that work well w.r.t. the true optimization problem. On the other hand, if the deterministic proxy is not able to reach stochastically optimal decisions, using this proxy is sub-optimal. This is the main intuition behind sub-optimality of the deterministic proxy, and occurs when we have decision dominance:

Definition 1. *Decision* $x \in X$ *is dominated by* $\hat{x} \in X$ *if* $\forall c \in C : f(c, \hat{x}) < f(c, x)$.

Note that this definition is based on the objective function $f(\cdot)$, which means that this property does not relate decisions w.r.t. the true stochastic problem in Equation 1.

Lemma 1. *If there exists a dominated decision* $x \in X$, $x^D : C \to X, x^D(c) = \operatorname{argmin}_{x\in X} f(c, x)$ *is not surjective.*

Lemma 1 holds because dominated decisions can not be reached, violating the property of surjectivity. Based on this, we cannot apply Theorem 3 if there exists a dominated decision. However, non-surjectivity is not a sufficient condition for a proxy to be sub-optimal. A true optimal decision $x^*(z)$ being dominated is sufficient, as the next theorem expresses.

[1]With abuse of nation, as the arguments of the minima are not necessarily unique. Technically these have to be disjoint sets.

Theorem 4. *If there exists a decision $\hat{x} \in X$ and a feature value $z \in Z$ for which $\hat{x}$ dominates true optimal $x^*(z)$, deterministic proxy $x^D(\cdot)$ is sub-optimal. We have $V^* < V_{DFL}$.*

Now the question arises in what cases this dominance occurs. In the following section we present cases where this occurs for continuous optimization/decision problems, i.e., when the feasibility set is compact (closed and bounded) and objective function f is continuous. After this we present a class of discrete optimization problems where it is more natural to occur, in Section 3.2.

3.1 Continuous optimization

The main result for *continuous optimization problems* is the following: Depending on the shape of objective function $f(c, x)$ and expected objective function $g(x) := \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x)|z]$, it can occur that $\forall c \in C$ the minimizer of $f(c, x)$ lies on the boundary of X, while the minimizer of $g(x)$ lies in the interior of X. In this case, all interior decisions are dominated by boundary decisions. In Example 1, we observe that the true objective function is convex non-monotonic in x, while the deterministic proxy objective function is strictly monotonic in x. A monotonic function ensures that optimal decision lies on the boundary of the feasibility space, while we see the true optimal decision laying in the interior. Because of this, the optimal decision of the deterministic proxy never coincides with the optimal decision of the true optimization problem. This reasoning holds for single variable functions. Since we are working with multivariate functions in general, we first give a monotonicity definition for multivariate functions.

Definition 2. *Let $f : X \to \mathbb{R}, X \subset \mathbb{R}^n$ be a multivariate function. f is (strictly) coordinate-wise (non-)monotonic if $\forall i \in \{1, \dots, n\}$ we have that for any fixed variables $\{\bar{x}_1, \dots, \bar{x}_{i-1}, \bar{x}_{i+1}, \dots \bar{x}_n\} \in \mathbb{R}^{n-1}$, $f(\bar{x}_1, \dots, x_i, \dots, \bar{x}_n)$ is (strictly) (non-)monotonic in x_i.*

Note that coordinate-wise monotonicity allows the function to be decreasing in some variables while increasing in others.

Theorem 5. *Given two continuous (objective) functions $f : C, X \to \mathbb{R}$, $g : X \to \mathbb{R}$, with $g(x) := \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x)|z]$, and (feasibility set) X is compact (closed and bounded) and has a nonempty interior. If $f(c, x)$ is strictly coordinate-wise monotonic in x for all $c \in C$, and $g(x)$ is convex and coordinate-wise non-monotonic, then $V^* < V_{DFL}$.*

In practice the constraints that define X can lead it to have a lower intrinsic dimension and therefore an empty interior. However the same result holds, we refer to the Appendix for further details and the proof.

Example 2. *Continuous objective functions with no/an optimal deterministic proxy. (Assuming X is compact.)*
Objective $f(c, x) = c^T x$ does not satisfy the conditions of Theorem 5, since $g(x) = \mathbb{E}_{c\sim\mathcal{C}_z}[c^T x|z] = \mathbb{E}_{c\sim\mathcal{C}_z}[c|z]^T x$ is linear in x for any distribution $\mathcal{C}_z$ and therefore not coordinate-wise non-monotonic. Similarly $\frac{1}{c^T x}$ and $(c^T x)^n$ with even n's do not satisfy the conditions of Theorem 5 as they are also not coordinate-wise non-monotonic.
Objective $f(c, x) = -\log(c^T x)$ does satisfy the conditions of Theorem 5 for at least some distributions $\mathcal{C}_z$. This is because $-\log(c^T x)$ is strictly coordinate-wise monotonic $\forall c \in C$, because for any $c_i > 0$, increasing x_i increases $\log(c^T x)$, and for any $c_i < 0$, decreasing x_i decreases $\log(c^T x)$. On the other hand, $g(x)$ can be strictly convex with a minimum in the interior, depending on the distribution $\mathcal{C}_z$ (see Example 1). Similarly $\sqrt{c^T x}, e^{c^T x}$ and $(cx)^n$ with odd n's satisfy the conditions of Theorem 5.

Theorem 5 demonstrates when a deterministic proxy can be sub-optimal. The intuition here is that functions that are monotonic lead to decisions on the boundary of the feasibility set, instead of the interior. Referring back to Example 1: The deterministic model leads to one of two decisions: if we expect our return is positive we invest everything, if we expect it is negative we invest nothing. Reality however is that we are uncertain, and investing partially in multiple shares is less risky and more profitable long-term.

So far we have derived properties for the structure of the optimization problem to show sub-optimality. Based on the function examples, one could argue that the types of functions for which this occurs are not common. However, there is an important class of practical optimization problem that typically have complex, non-smooth objective functions.

3.2 Two-stage stochastic optimization problems

A practical class of optimization problems where a deterministic proxy can be sub-optimal is the class of *two-stage stochastic optimization problems*. This two-stage formulation is of particular interest in optimization literature Birge and Louveaux (2011), as it is effective in modeling real-world problems under uncertainty. These problems consist of making first-stage decisions x that are taken before uncertain parameters c are realized and second-stage decisions y

that denote some recourse action after uncertain parameters c are realized. The general formulation is:

$$\min_{x \in X} f(x) + \mathbb{E}_c[Q(c,x)]$$
$$\text{with} \quad Q(c,x) = \min_{y \in Y(c,x)} g(c,x,y)$$

The practicality of this formulation comes from the fact that in practice the realization of uncertainty leads to some action: if we forgot an umbrella, we can buy a new one. These *recourse actions* include additional costs compared to what was initially decided. One can model the problem in a way it has *relatively complete recourse*, i.e., the second stage is feasible for any feasible first-stage decision. This is one of the main benefits of two-stage stochastic optimization, as other uncertain problem formulations like (distributionally) robust optimization or chance constraints (Charnes and Cooper, 1962) can only ensure feasibility for a certain set of uncertain value realizations or with a certain probability. Specifically, models with a linear second stage are commonly used, i.e.,

$$Q(c,x) = \min_{y \in Y(c,x)} q(c)^T y$$
$$\text{s.t.} \quad T(c)x + U(c)y = h(c).$$

Homem-de-Mello et al. (2024) showed that for these problems with fixed costs and fixed recourse, i.e., $q(c) = q$ and $U(c) = U$ being independent of c, an optimal single-scenario exists. In this work we show that there are problems with fixed costs and fixed recourse for which an optimal scenario does *not* exist. The proof by Homem-de-Mello et al. (2024) is based on the assumption that $h(c)$ is completely stochastic, but when it is partly deterministic an optimal scenario does not necessarily exist. We demonstrate this with an example based on the *weighted-set multi-cover problem* (WSMC).

In the WSMC problem, we have some items that have a coverage requirement, which is attained by choosing sets that each cover a subset of these items. The goal is to minimize the total costs of the chosen sets. A stochastic version is presented in Silvestri et al. (2023), where the coverage requirements are uncertain. When a number of sets is chosen, and the coverage requirements are unmet, an additional cost is paid in the second stage to cover the unmet requirement by adding sets. In the formulation we present here, we allow removing single-item sets in the case of excess coverage. This makes the recourse similar to the common formulation of the two-stage knapsack problem with uncertain weights (Kosuch and Lisser, 2011). Below is the formulation of WSMC with n items and m sets, $m > n$.

For $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$:

$x_j \in \mathbb{N}$:	Number of set j chosen
$c_j \in \mathbb{N}$:	Cost for set j
$\xi_i \in \mathbb{N}$:	Coverage requirement for item i
$a_{ij} \in \{0,1\}$:	If item i is covered by set j, $A = (a_{ij})$, with $a_{ii} = 1, a_{ji} = 0$ for $i, j \in \{1, ..., n\}, i \neq j$
$y_i^+, y_i^- \in \mathbb{N}$:	Unmet/excess coverage item i
$c_i^+, c_i^- \in \mathbb{N}$:	Unmet/excess coverage cost item i

$$\min_x c^T x + \mathbb{E}_\xi[Q(\xi, x)]$$
$$Q(\xi, x) = \min_{y^+, y^-} c^{+T} y^+ - c^{-T} y^-$$
$$Ax + y^+ - y^- \geq \xi$$
$$y_i^- \leq x_i, \quad i \in \{1, \dots, n\}$$

Example 3. *Given the WSMC problem above, with $n = 2$ items, $m = 3$ sets, costs $c = (4, 4, 7)$, $A = \left[\begin{smallmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{smallmatrix}\right]$ and, recourse costs $c^+ = (7, 7)$, $c^- = (3, 3)$. First of all, we note that the optimal decision to the deterministic equivalent of this problem has no recourse, i.e., $y^+ = y^- = 0$, because given $\hat{\xi}$, we select x such that it is exactly met and $c^+ > (c_1, c_2) > c^-$. Therefore, it simply reduces to $\min_{x: Ax \geq \hat{\xi}} c^T x$. $\min_{x: \sum_{j \in M} a_{ij} x_j \geq \hat{\xi}_i} \sum_{j \in M} c_j x_j$ Secondly, we note that decision $x = (1, 1, 0)$ is dominated by $\hat{x} = (0, 0, 1)$, because $\hat{x}$ results into the same exact coverage as x, but with lower objective $f(\hat{x}) = 7 < 8 = f(x)$. However, the optimal decision to the original problem can very well be x. Take for example ξ distributed s.t. $\mathbb{P}(\xi = (1, 1)) = 0.5 = \mathbb{P}(\xi = (0, 0))$, now let $g(x) = c^T x + \mathbb{E}_\xi[Q(\xi, x)]$, we get $g((1, 1, 0)) = 5 < g((1, 0, 0)) = 6 < g((0, 0, 1)) = 7 = g((0, 0, 0))$.*

Section 3 highlighted the shape of a decision problem's objective function being the culprit in making the deterministic proxy sub-optimal. In two-stage stochastic problems, the second stage problem is a term on the objective function and has potentially complex and non-smooth behaviour – especially because practical problems often include integer decisions.

In the next section, we propose efficient alternative proxies.

4 Sufficient decision proxies

In the previous section we saw that a sufficient condition for a proxy to be able to lead to an optimal decision $x^*(z)\ \forall z$ is that it is expressive enough. In this section we present proxies that can adhere to this condition, when a deterministic proxy does not. We also discuss why an intuitive choice, using a (parameterized) distribution proxy, is most often not effective. Recall that in a contextual optimization problem, the input to the chosen proxy problem $x(\cdot)$ is the output of the predictive model $\phi_\theta(\cdot)$, so they go hand in hand. We train our predictive model to get a decision quality maximizing policy $\pi : Z \to X, \pi = x \circ \phi_\theta$ w.r.t. problem 1.

4.1 Scenario-based

Based on Theorem 4 we know that in certain cases proxy $x^{\mathrm{D}}(\cdot)$ is not sufficient. Since this deterministic variant is introduced to approximate decision problem of interest in Equation 1, we consider an alternative approximation based on the principle of *sample average approximation* (SAA) (Kleywegt et al., 2002):

$$x_n^{\mathrm{SA}}(c_1, \ldots, c_n) = \operatorname*{argmin}_{x \in X} \sum_{i=1}^{n} f(c_i, x).$$

SAA is used in stochastic optimization to approximately solve a stochastic optimization problem. When the uncertain parameter distribution is known but does not lead to a solvable optimization problem (non closed-form objective), SAA approximates the stochastic optimization problem. To solve this problem scenarios can be sampled from the uncertain distribution to subsequently solve the SAA. The SAA converges to the true stochastic optimization problem as the samples go to infinity. In the contextual optimization setting this results into the following theorem:

Theorem 6 (Kim et al. (2015))**.** *Suppose $A \subset X$ be nonempty and compact, such that (i) $x^*(z) \in A$, (ii) for sufficiently large n, $x_n^{SA} \in A$ and (iii) suppose that $f(c, \cdot)$ is continuous at x a.s. for any $x \in X$, and that there exists $\delta > 0$ such that the family of random variables $\{f(c, y) : ||y - x|| < \delta\}$ is uniformly integrable. Then $x_n^S(\cdot) \to x^*(z)$*

The difference with the traditional SAA setting is that here we use a predictive model to obtain the scenarios, instead of sampling them. Theoretically one could predict a distribution and sample from it, however this would require a high value of n and/or significantly increases variance during training by introducing randomness. Instead, we predict the scenarios directly, i.e., a discrete distribution with equal probabilities.

This means use a predictive model that predicts scenarios $\phi_{\theta,n}^{\mathrm{S}} : Z \to C^n$, and we get scenario-based policy $\pi_n^{\mathrm{S}} = x_n^{\mathrm{SA}} \circ \phi_{\theta,n}^{\mathrm{S}}$. This means that as long as we pick a high enough value of n such that we have a theoretically optimal predictor, the DFL pipeline could learn to find the right scenarios that lead to optimal decisions. One drawback of this approach is that it requires a high number of parameters to closely approximate continuous probability distributions. However, experimental results in Section 5 that with values of n as low as 2 the model is able to learn strong performing predictors. This is because of one of the main principles of DFL: We do not learn to predict accurately, we learn to predict such that we get high quality decisions.

4.2 Quadratic

Instead of trying to approximate Equation 1 better, we can also look at the problem from the other perspective. Based on the earlier-developed theory on necessary conditions for a predictor to be theoretically optimal, we can design a proxy that adheres to the required properties. We can do this by altering the objective function $f(\cdot)$. First, we generalize the first part of Theorem 3:

Theorem 7. *Consider a function $x(\cdot) : \Xi \to X$. An optimal prediction w.r.t. Equation 1 exists if $x(\cdot)$ is surjective, i.e., $\forall x \in X, \exists \xi \in \Xi : x(\xi) = x$.*

Proof. Take an arbitrary z, with optimal decision $x^*(z)$. Given that $x(\cdot)$ is surjective, $\exists \xi \in \Xi$ s.t. $x(\xi) = x^*(z)$. □

We now introduce a surjective function for this setting, which coincides with the *quadratic approximation* proposed recently by Veviurko et al. (2024).

$$x^{\mathrm{Q}}(\xi) = \operatorname*{argmin}_{x \in X} ||\xi - x||_2^2.$$

This proxy is surjective by design, as for arbitrary $x \in X$, taking $\xi = x$ leads to $x^{\mathrm{Q}}(\xi) = x$. This leads us to using predictive model $\phi_\theta^{\mathrm{Q}} : Z \to \Xi$, where $\Xi = \mathbb{R}^{\dim(X)}$, and quadratic policy $\pi^{\mathrm{Q}} = x^{\mathrm{Q}} \circ \phi_\theta^{\mathrm{Q}}$.

Let us elaborate more on the rationale of this proxy $x^{\mathrm{Q}}(\cdot)$, originally introduced in the context of dealing with zero-gradient problem: it is a function that has a strictly convex objective function, while adhering to the constraints of the problem (feasibility set X). There is no reason to believe this is a reasonable approximation to Equation 1, but this is not necessary. Given the same feasibility set, we will still reach feasible decisions. Based on the assumption of the predictive model being flexible, the objective function being different does not have to be a problem. This is because an optimal decision always lies in some minimum, thus why not approximate this minimum by a quadratic function? The DFL pipeline will ensure the predictor learns to predict in the right area.

4.3 Parameterized distribution

Since we argue that deterministic proxies and therefore single-valued predictions are not always sufficient, an intuitive approach would be to predict a distribution and use this to approximate Equation 1. Donti et al. (2017) assume a Gaussian distribution in one of their experiments, since the problem they present has a closed-form objective if the underlying distribution is Gaussian. This is exactly the use-case where this is helpful. In a general sense this leads to the following decision function, with $\beta \in B$ the parameters of the chosen distribution $\mathbb{P}$ and $p_\beta(\cdot)$ its density function:

$$x^{\mathbb{P}}(\beta) = \operatorname*{argmin}_{x \in X} \int_C f(c, x) p_\beta(c) dc.$$

Leading to using predictive model $\phi^{\mathbb{P}}_\theta : Z \to B$ and policy $\pi^{\mathbb{P}} = x^{\mathbb{P}} \circ \phi^{\mathbb{P}}_\theta$. In practice however, the named class of two-stage stochastic optimization problems is unlikely to have a parameterized distribution that leads to a closed-form objective. If no closed-form exists for a certain distribution, it is still possible to consider the following proxy: Let $F^{-1}_\beta(\cdot)$ be the β-parameterized inverse cumulative probability density function for distribution $\mathbb{P}$:

$$x^{\mathbb{P},\mathrm{SA}}_n(\beta) = \operatorname*{argmin}_{x \in X} \sum_{i=1}^{n} f\big(F^{-1}_\beta\big(\frac{i}{n+1}\big), x\big).$$

Leading to policy $\pi^{\mathbb{P},\mathrm{S}}_n = x^{\mathbb{P},\mathrm{SA}}_n \circ \phi^{\mathbb{P}}_\theta$. The benefit of this approach over π^{S}_n is that even with high n, the number of learnable parameters stays equal to the number of parameters of the predetermined distribution. Additionally, the chosen distribution imposed some structure, which could make the learning easier if this structure is present in the true underlying distribution. However, this is also its main drawback. The approach is less flexible than directly predicting the scenarios. Given some policy $\hat{\pi}^{\mathbb{P},\mathrm{S}}_n$, there always exists a $\hat{\pi}^{\mathrm{S}}_n$ such that $\hat{\pi}^{\mathrm{S}}_n = \hat{\pi}^{\mathbb{P},\mathrm{S}}_n$, because the image of F^{-1}_β is a subset of C. The reverse is most often not true, as any $\hat{\pi}^{\mathbb{P},\mathrm{S}}_n$ is restricted by its distribution. In the experiments we present there is no commonly used parameterized distribution such that a closed-form objective exists. We ran some preliminary experiments using $\pi^{\mathbb{P},\mathrm{S}}_n$, but these were always outperformed by π^{S}_n and therefore omitted them.

4.4 Complexity of policies

All presented policies have a different combination of distributional predictors and decision proxies. Table 1 summarizes the policies by denoting the prediction space and the number of decision variables in a two-stage stochastic optimization setting. This number correlates highly with the number of constraints of the problem, as any constraint involving a second-stage decision variable is multiplied by the number of scenarios. Learning with DFL, the complexity of the decision proxy is highly relevant as it is solved for each data point for each epoch. In general more decision variables and constraints increase the complexity of the decision proxy.

The prediction space has less impact. The size of the prediction space impacts the number of parameters of the predictive model. Since in general we (can) use neural networks with a lot more parameters than the dimension of the prediction space, this does not change the learning problem much. What is more of an open question is if the prediction space influences the complexity of learning a high quality policy. In some problems $\dim(X)$ is significantly different from $\dim(C)$, which leads to the question if a high quality π^{Q} is harder/easier to learn than π^{D}. The experiments we present give some insights regarding this.

5 Experimental evaluation

To experimentally evaluate the proposed approaches as alternatives to a deterministic proxy, we consider three problems. We consider one continuous problem and two discrete two-stage problems. In all cases we use three baselines: deterministic proxy learned using DFL π^{D}, deterministic proxy learned using PFL $\pi^{\mathrm{D}}_{\mathrm{PFL}}$ and a PFL learned model using SAA at inference $\pi^{\mathrm{S}}_{\mathrm{PFL}}$. In the PFL approaches we train on mean squared error, where $\pi^{\mathrm{S}}_{n,\mathrm{PFL}}$ uses the same learned predictive model as PFL ϕ_θ, but evaluates based on an SAA approximation (x^{SA}_n). This is done by using a

policy	prediction space	decision var. #
π^{D}	C	$d_y + d_x$
π^{S}_n	C^n	$nd_y + d_x$
π^{Q}	$\mathbb{R}^{\dim(X)}$	$d_y + d_x$
$\pi^{\mathbb{P}}$	B	$d_y + d_x$
$\pi^{\mathbb{P},\mathrm{S}}_n$	B	$nd_y + d_x$

Table 1: Summary of different policies and the output space of the predictor. The last column denotes the number of decision variables when the problem is two-stage, with d_x and d_y denoting number of first- and second-stage variables respectively.

residual-based distribution (Sadana et al., 2025), using empirical additive errors (Deng and Sen, 2022): taking the predictions as mean, and adding samples from the observed residuals (in the validation set). We use $n = 16$, so $\pi^{\mathrm{S}}_{16,\mathrm{PFL}}$ in all problems. We compare the baselines with our proposed approaches: π^{Q}, π^{S}_2 and π^{S}_8 (except for WSMC, as there π^{S}_2 is already best performing). For the continuous problem we use DFL with exact gradients, using differentiable convex layer from Agrawal et al. (2019). For the other problems we use score function gradient-estimation as proposed by Silvestri et al. (2023). Regarding the predictors we always use neural networks with 2 hidden layers of size 256 and LeakyReLU activation functions. Adam optimizer is used for the gradient descent of the predictive models and *Gurobi* version 10.0.1 (Gurobi Optimization, 2025) as the optimization problem solver. We give each approach the same number of epochs, which we consider enough to converge. In all problems we evaluate on absolute regret at validation and test-time. The best performing model on validation is used for test. For further technical details please refer to the Appendix. Code can be found at *placeholder link Git*.

Portfolio As considered in earlier work on DFL (Wang et al., 2020; Veviurko et al., 2024), we consider a portfolio problem based on data from Quandl (2016). Instead of adding a penalized covariance term to minimize risk, we do this by maximizing the long-term expected value of the portfolio using the Kelly strategy Kelly (1956). This makes the problem a multivariate version of Example 1. We use a fixed rate of return $\beta = 0.08$. Since data is limited (2898 data points) we do not use different data splits, instead we randomly pick 10 different securities for each seed, running 5 different seeds. We use a train, validation, test split of 70%, 15%, 15%.

Weighted-set multi-cover We test on the WSMC problem as introduced in Section 3.2. Contextual data is randomly generated as done by Silvestri et al. (2023) for 7 seeds. For train, validation and test sizes, we use 1000, 250 and 1250.

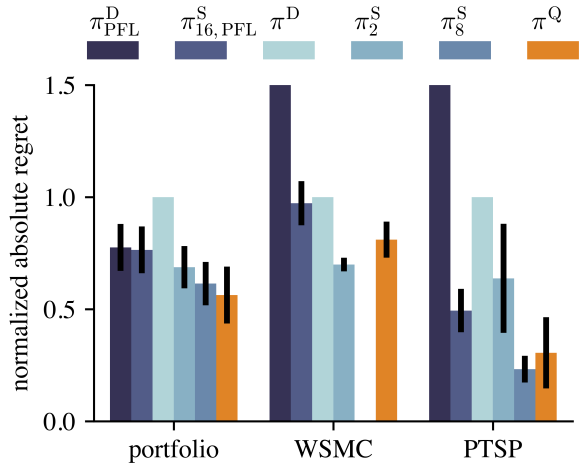
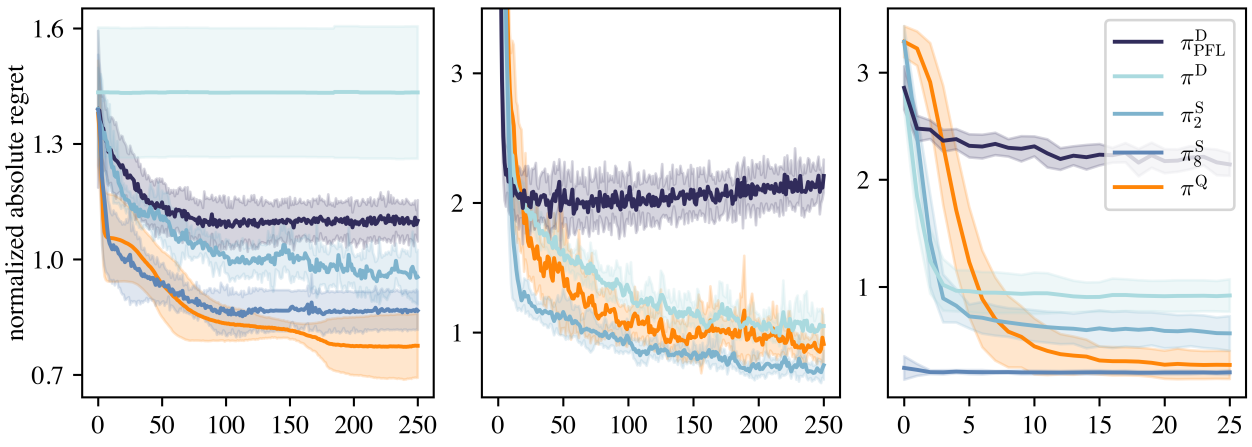


Figure 3: Absolute regret mean on the test set, normalized by π^{D}. Error bars denote one standard deviation. $\pi^{\mathrm{D}}_{\mathrm{PFL}}$ bars for WSMC 1.80 (0.21) and PTSP 2.35 (0.35) were cut off for visibility.

	portfolio	WSMC	PTSP
$\pi^{\mathrm{D}}_{\mathrm{PFL}}$	2.4 (0.3)	551.1 (79.3)	127.9 (20.4)
$\pi^{\mathrm{S}}_{16,\mathrm{PFL}}$	2.3 (0.3)	300.5 (53.0)	26.9 (4.5)
π^{D}	3.0 (0.1)	309.6 (50.0)	55.2 (9.6)
π^{S}_2	2.1 (0.2)	**216.8 (37.1)**	34.0 (11.2)
π^{S}_8	1.9 (0.2)	-	**12.6 (2.6)**
π^{Q}	**1.7 (0.3)**	251.1 (47.3)	17.0 (8.8)

Table 2: Absolute regret mean (standard deviation) on the test set. Best values per problem in **bold**. Portfolio values are in percentages. First 3 policies are baselines, last 3 are proposed.

Probabilistic traveling salesperson Finally we test on a two-stage probabilistic traveling salesperson (PTSP). Probabilistic vehicle routing problems have been studied since Tillman (1969), with PTSP specifically introduced by (Jaillet, 1985), and are still highly relevant (Oyola et al., 2018). The goal of the PTSP is to plan a route along a known set of customers while minimizing transportation costs, but it is uncertain which customers require service. Missing a



(a) Portfolio (b) Weighted-set multi-cover (c) Probabilistic traveling salesperson

Figure 4: Validation learning curves per epoch (x-axis). The average of approaches minimum validation absolute regret was used to scale the absolute regret for each seed. Error bars denote one standard deviation.

customer who requires service incurs a penalty (based on going back-and-forth to the depot, which is a typical recourse choice Oyola et al. (2018)), while visiting a customer that does not require service does nothing out. In this two-stage version, one is allowed to add direct trips to customers, similar to allowing crowdsourcing Santini et al. (2022). After it becomes known which customers require service (second stage), these direct trips can be canceled if they are not required. This makes the problem such that in a given (deterministic) scenario, direct trips are always sub-optimal, but given uncertainty they have have the benefit of incurring low costs when no service is required. We run 10 seeds and for train, validation and test sizes, we use 1000, 250 and 1250.

The problem instances are generated by putting the customers equally spaced on a circle with fixed diameter and then perturbed with a normal distribution. Contextual data is generated similar to (Elmachtoub and Grigas, 2022), but since we are dealing with binary uncertain parameters, we clip generated values and replace a percentage of values with a Bernoulli distribution based on the *noise width* parameter.

5.1 Results

Figure 3 shows the final results, while Figure 4 visualizes the learning. We use a paired t-test (5% significance level) to compare our proposed methods with the PFL, PFL with SAA and deterministic proxy baselines, and in all but one case the proposed approaches performed significantly better (PTSP, π^{S}_{2} compared to $\pi^{\mathrm{S}}_{16,\mathrm{PFL}}$ being the exception). We see that in x^{Q} performs best or second best in all problems, which shows that using an accurate approximation of the objective function is not necessary for DFL. For portfolio and PTSP, we see x^{SA}_{8} outperforming x^{SA}_{2}, suggesting more predicted points could lead to better results. For PTSP this behavior can be explained by the fact that the uncertain parameters can only attain 0 or 1. In this case, the 2-scenario proxy only has a probability of 0, 0.5 or 1 of requiring a visit per customer, which might be insufficient. For portfolio, Figure 4a shows that x^{D} is not able to learn at all. This is primarily caused by having zero-gradients, a common issue in DFL, which goes hand in hand with our argument that deterministic proxies can be limiting. A higher number of scenarios here is more likely to result in a non-zero gradient. The most notable part about the WSMC is that the learning curves have high variance. This is inherent to the used score function gradient estimation approach. For PTSP this is not as pronounced, due to the small problem size and therefore less existing high-quality decisions.

6 Discussion and related work

We formally showed cases when deterministic proxies are sub-optimal, even in a DFL setting, and introduce effective alternative predictors and decision proxies. This is another step in understanding the applicability of DFL, after Cameron et al. (2022) consider when PFL is worse compared to DFL and Homem-de-Mello et al. (2024) show for a certain class of problems that an optimal single-scenario exists. Both works discuss the importance of the decision proxy, which is similarly discussed by Schutte et al. (2024), highlighting the difference between the empirical problem (deterministic proxy) and the true stochastic problem with as main aim to introduce loss functions that are able to generalize better

when few training data is available. Earlier work by (Wilder et al., 2019) introduced a continuous proxy to speed up DFL when the decision problem is discrete.

There are several existing works that do consider the true problem as stochastic and use distribution-based decision proxies. These works differ from our work in that we provide theoretical backing on when a deterministic proxy is non-sufficient, as well as propose methods based on this theory that are efficient, i.e., keep complexity as low as possible while having the guarantee that there exist a predictive model that leads to an optimal solution. Donti et al. (2017) tackle a problem that has a differentiable formulation when assuming the uncertainty is Gaussian. Alternatively they assume the uncertain parameter has finite discrete support, having the predictive model output probabilities for these discrete scenarios. This is a similar assumption as done by Grigas et al. (2021), which is quite limiting as uncertain parameters are often continuous and multi-dimensional, requiring to decide on some discretization that is representative without having too many scenarios. Our scenario-based approach learns representative scenarios and only requires to decide the number of scenarios up front. Elmachtoub et al. (2023) also include a distribution-based approach, assuming a parameterized family of distributions. They show that when the distribution is misspecified, distributional DFL (contextual integrated-estimation-optimization) outperforms distributional PFL (estimate-then-optimize), while in a well-specified distribution case the result is opposite (given enough data).

Another way that stochasticity is considered is when it is used to obtain gradients when they do no exist or are zero, but in these cases the deterministic approximation is used at inference time Berthet et al. (2020); Silvestri et al. (2023). Furthermore, Hu et al. (2023) consider uncertainty in the constraints, modelling infeasibility using penalties. This is similar to a two-stage formulation as presented here, but again a deterministic approximation is considered. We refer the reader to Qi and Shen (2022) for an operations management perspective on DFL, and Mandi et al. (2024) for a general survey on DFL.

7 Conclusions

The prevailing assumption in decision-focused learning is that there exists a single-scenario problem approximation that is sufficient to obtain an optimal decision. While this is a valid assumption in a wide class of problems, this paper investigates for the first time theoretical properties of problems for which this assumption is violated. Based on these properties we derive requirements for problem approximations and predictive models that are sufficient to obtain optimal decisions. On three problems, we demonstrated that the assumption does not hold and empirically showed the value of the proposed approaches. Future work includes studying problems where parameterized distributions can easily be assumed, as well as problems with (highly) dependent uncertain variables. Given the strong performance of the quadratic proxy, it opens the question what the relevance is of the true objective in the training pipeline. With this we are moving a step closer in the direction of direct feature to decision mappings.

8 Acknowledgements

This project has received funding from the EU Horizon 2020 programme under grant number 964505 (Epistemic AI).

References

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize"'. *Management Science*, 68:9–26, 2022. doi:10.1287/mnsc.2020.3922.

Tito Homem-de-Mello, Juan Valencia, Felipe Lagos, and Guido Lagos. Forecasting outside the box: Application-driven optimal pointwise forecasts for stochastic optimization. *CoRR*, abs/2411.03520, 2024. doi:10.48550/arXiv.2411.03520.

Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: A methodological tour d'horizon. *European Journal of Operational Research*, 290(2):405–421, 2021. doi:10.1016/j.ejor.2020.07.063.

Utsav Sadana, Abhilash Chenreddy, Erick Delage, Alexandre Forel, Emma Frejinger, and Thibaut Vidal. A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research*, 320(2):271–289, 2025. doi:10.1016/j.ejor.2024.03.020.

Stefano Teso, Laurens Bliek, Andrea Borghesi, Michele Lombardi, Neil Yorke-Smith, Tias Guns, and Andrea Passerini. Machine learning for combinatorial optimisation of partially-specified problems: Regret minimisation as a unifying lens. *CoRR*, abs/2205.10157, 2022. doi:10.48550/arXiv.2205.10157.

Andrey I Kibzun and Yuri S Kan. Stochastic programming problems with probability and quantile functions. *Journal of the Operational Research Society*, 48(8):849–849, 1997. doi:10.1057/palgrave.jors.2600833.

John L Kelly. A new interpretation of information rate. *The Bell System Technical Journal*, 35(4):917–926, 1956. doi:10.1002/j.1538-7305.1956.tb03809.x.

John R. Birge. The value of the stochastic solution in stochastic linear programs with fixed recourse. *Mathematical Programming*, 24:314–325, 1982. doi:10.1007/BF01585113.

Jayanta Mandi, James Kotary, Senne Berden, Maxime Mulamba, Victor Bucarey, Tias Guns, and Ferdinando Fioretto. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. *Journal of Artificial Intelligence Research*, 80:1623–1701, 2024. doi:10.1613/jair.1.15320.

Chris Cameron, Jason Hartford, Taylor Lundy, and Kevin Leyton-Brown. The perils of learning before optimizing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3708–3715, 2022. doi:10.1609/aaai.v36i4.20284.

Mattia Silvestri, Senne Berden, Jayanta Mandi, Ali İrfan Mahmutoğulları, Maxime Mulamba, Allegra De Filippo, Tias Guns, and Michele Lombardi. Score function gradient estimation to widen the applicability of decision-focused learning. *CoRR, abs/2307.05213*, 2023. doi:10.48550/arXiv.2307.05213.

Kim van den Houten, David M. J. Tax, Esteban Freydell, and Mathijs de Weerdt. Learning from scenarios for repairable stochastic scheduling. In *Proceedings of the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, pages 234–242, 2024. doi:10.1007/978-3-031-60599-4_15.

J R Birge and F Louveaux. *Introduction to Stochastic Programming*. Springer, 2011. doi:10.1007/978-1-4614-0237-4.

A. Charnes and W. W. Cooper. Chance constraints and normal deviates. *Journal of the American Statistical Association*, 57(297):134–148, 1962. doi:10.1080/01621459.1962.10482155.

S. Kosuch and A. Lisser. On two-stage stochastic knapsack problems. *Discrete Applied Mathematics*, 159(16): 1827–1841, 2011. doi:10.1016/j.dam.2010.04.006. 8th Cologne/Twente Workshop on Graphs and Combinatorial Optimization (CTW'09).

Anton J. Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2002. doi:10.1137/S1052623499363220.

Sujin Kim, Raghu Pasupathy, and Shane G Henderson. A guide to sample average approximation. In *Handbook of simulation optimization*, pages 207–243. Springer, 2015. doi:10.1007/978-1-4939-1384-8_8.

Grigorii Veviurko, Wendelin Böhmer, and Mathijs de Weerdt. You shall pass: Dealing with the zero-gradient problem in predict and optimize for convex optimization. *CoRR, abs/2307.16304*, 2024. doi:10.48550/arXiv.2307.16304.

Priya Donti, Brandon Amos, and J. Zico Kolter. Task-based end-to-end model learning in stochastic optimization. *Advances in Neural Information Processing Systems*, 30:5484–5494, 2017. doi:10.48550/arXiv.1703.04529.

Yunxiao Deng and Suvrajeet Sen. Predictive stochastic programming. *Computational Management Science*, 19(1): 65–98, 2022. doi:10.1007/s10287-021-00400-0.

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J Zico Kolter. Differentiable convex optimization layers. *Advances in neural information processing systems*, 32, 2019. doi:10.48550/arXiv.1910.12430.

Gurobi Optimization. Gurobi Optimizer 12.0 reference manual, 2025. URL `www.gurobi.com`. Accessed: 2025-01-25.

Kai Wang, Bryan Wilder, Andrew Perrault, and Milind Tambe. Automatically learning compact quality-aware surrogates for optimization problems. *Advances in Neural Information Processing Systems*, 33:9586–9596, 2020. doi:10.48550/arXiv.2006.10815.

Quandl. WIKI various end-of-day data, 2016. URL `https://www.quandl.com/data/WIKI`.

Frank A Tillman. The multiple terminal delivery problem with probabilistic demands. *Transportation Science*, 3(3): 192–204, 1969. doi:10.1287/trsc.3.3.192.

Patrick Jaillet. *Probabilistic traveling salesman problems*. PhD thesis, Massachusetts Institute of Technology, 1985. URL `https://www.mit.edu/~jaillet/general/jaillet-phd-mit-orc-85.pdf`.

Jorge Oyola, Halvard Arntzen, and David L. Woodruff. The stochastic vehicle routing problem, a literature review, part I: models. *EURO Journal on Transportation and Logistics*, 7(3):193–221, 2018. doi:10.1007/s13676-016-0100-5.

Alberto Santini, Ana Viana, Xenia Klimentova, and João Pedro Pedroso. The probabilistic travelling salesman problem with crowdsourcing. *Computers & Operations Research*, 142:105722, 2022. ISSN 0305-0548. doi:10.1016/j.cor.2022.105722.

Noah Schutte, Krzysztof Postek, and Neil Yorke-Smith. Robust losses for decision-focused learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI'24*, pages 4868–4875, 2024. doi:10.24963/ijcai.2024/538.

Bryan Wilder, Bistra Dilkina, and Milind Tambe. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1658–1665, 2019. doi:10.1609/aaai.v33i01.33011658.

Paul Grigas, Meng Qi, and Zuo-Jun Shen. Integrated conditional estimation-optimization. *CoRR*, abs/2110.12351, 2021. doi:10.48550/arXiv.2110.12351.

Adam N. Elmachtoub, Henry Lam, Haofeng Zhang, and Yunfan Zhao. Estimate-then-optimize versus integrated-estimation-optimization: A stochastic dominance perspective. *CoRR*, abs/2304.06833, 2023. doi:10.48550/arXiv.2304.06833.

Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis Bach. Learning with differentiable perturbed optimizers. *Advances in Neural Information Processing Systems*, 33:9508–9519, 2020. doi:10.48550/arXiv.2002.08676.

Xinyi Hu, Jasper C. H. Lee, and Jimmy H. M. Lee. Two-stage predict+optimize for mixed integer linear programs with unknown parameters in constraints. *CoRR*, abs/2311.08022, 2023. doi:10.48550/ARXIV.2311.08022.

Meng Qi and Zuo-Jun Shen. Integrating prediction/estimation and optimization with applications in operations management. In *Tutorials in operations research: emerging and impactful topics in operations*, pages 36–58. INFORMS, 2022. doi:10.1287/educ.2022.0249.

Appendix

A Proofs

Proof for Theorem 1.

$$\begin{aligned}\mathrm{V}^* &= \min_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x)|z] \\ &= \min_{x\in X} f(\bar{c}_z, x) \\ &= f(\bar{c}_z, x^{\mathrm{D}}(\bar{c}_z)) \\ &= \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\bar{c}_z))|z] = \mathrm{V}_{\text{PFL}}\end{aligned}$$ □

Proof for Theorem 2.

$$\begin{aligned}x^{\mathrm{D}}(\bar{c}_z) &= \operatorname*{argmin}_{x\in X} f(\bar{c}_z, x) \\ &\neq \operatorname*{argmin}_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x)|z] = x^*(z) \\ \implies & \\ \mathrm{V}^* &= \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^*(z))|z] \\ &< \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\bar{c}_z)|z] = \mathrm{V}_{\text{PFL}}\end{aligned}$$ □

Proof for Theorem 3. Due to $x^{\mathrm{D}}(\cdot)$ being surjective, for every $x^*(z)$ there exists $\hat{c}_z \in C$ such that $x^{\mathrm{D}}(\hat{c}_z) = x^*(z)$. We get:

$$\begin{aligned}\mathrm{V}^* &= \min_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x)|z] \\ &= \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^*(z))|z] \\ &= \min_{\hat{c}_z} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\hat{c}_z))|z] = \mathrm{V}_{\text{DFL}}\end{aligned}$$ □

Proof for Lemma 1. Let $x \in X$ be dominated by $\hat{x} \in X$.

$$\begin{aligned}&\min_{x\in X} f(c,x) \leq f(c,\hat{x}) < f(c,x) \quad \forall c \\ &\implies \\ &x^{\mathrm{D}}(c) \neq x \quad \forall c\end{aligned}$$ □

Proof for Theorem 4. Take $z \in Z$ and $\hat{x} \in X$ such that $x^*(z)$ is dominated by $\hat{x}$. We have that:

$$\begin{aligned}& x^{\mathrm{D}}(c) \neq x^*(z) \quad \forall c \\ & \implies \\ \mathrm{V}^* &= \min_{x\in X} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c,x)|z] \\ &= \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^*(z))|z] \\ &< \min_{\hat{c}_z} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\hat{c}_z))|z] = \mathrm{V}_{\text{DFL}}\end{aligned}$$ □

Proof for Theorem 5. Given that for arbitrary $c \in C$, $f(c,x)$ is continuous and strictly coordinate-wise monotone, with X closed and bounded, it attains its minimum on the boundary of X for all $c \in C$.

We use proof by contradiction to show that $g(x)$ does not attain its minimum on the boundary. Assume that minimum of $g(x)$ lies on the boundary: $x^* \in \partial X$. Since $g(x)$ is single coordinate non-monotonic, we know that $\exists i$ such that $g_i(x_i) := g(x_1^*, \ldots, x_i, \ldots, x_n^*)$ is non-monotonic. Assume that x_i^* is on the right boundary, i.e., $x_i^* \geq x_i, \quad \forall x_i \in X_i$, with $X_i \subset \mathbb{R}$ the allowed range for the i-th coordinate. Non-monotonicity means that $\exists x, x' : x < x', g_i(x) < g_i(x')$. If $g(x_i^*) \leq g(x)$ this violates the convexity of $g(x)$, if $g(x_i^*) \geq g(x') > g(x)$ this violates optimality. Flipping

inequalities the same exact argument can be made for the left boundary. So we have that $x^{\mathrm{D}}(c) \in \partial X$ for all $c \in C$ and $x^*(z) \notin \partial X$ and therefore the value at minimum $x^*(z)$ of Equation 1 cannot be attained by $x^{\mathrm{D}}(c)$ for any c, so:

$$\overbrace{\mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^*(z))|z]}^{\mathrm{V}^*} < \overbrace{\min_{\hat{c}} \mathbb{E}_{c\sim\mathcal{C}_z}[f(c, x^{\mathrm{D}}(\hat{c}))]}^{\mathrm{V}^{\mathrm{DFL}}}$$

□

In Section 3, after presenting Theorem 5, we note that X can have a lower intrinsic dimension and therefore an empty interior due to its constraints. If we can reduce to this lower dimension and we have functions with the same properties, the same result holds.

Corollary 8. *If there exists a projection $P \in \mathbb{R}^{m\times n}$, $m < n$ such that PX, $f(c, Px)$ and $g(Px)$ have the same properties as X, $f(c, x)$ and $g(x)$ in Theorem 5, then $V^* < V_{DFL}$.*

This corollary simply holds because we obtain a set and two functions with the required properties to apply Theorem 5.

B Experimental details

B.1 Score function gradient estimation

For two of the experimental problems we use score function gradient estimation as propose by Silvestri et al. (2023). In this approach stochasticity is used to estimate gradients, making it possible to have the optimization model as a complete black box. We use normal distributions in all cases, initializing sigma to be the predictive error we observe obtained from the initialized predictive model. The sigma parameter is also update during back-propagation, but feature independent. Initializing sigma to close to zero does not give any gradients, while initializing it too big will cause a lot of variance and therefore slow convergence.

B.2 Initialization

Initialization of the predictive model is in general important, as it can have significant impact on the results in both quality and convergence time. Another example is that in DFL it is not uncommon to initialize the predictive model as a PFL trained model. We do not do this, as we do not want our models to be biased towards a PFL model, only learning to improve relative to it. However, we do initialize the bias of the predictive model to the mean of the uncertain parameter in the training data, such that predictions are at least in the range of actual realizations. For the scenario-based approach π^{S}_n, we apply the same idea but use quantiles $\frac{i}{n+1}, i \in \{1, \ldots, n\}$. For the quadratic proxy we similarly take the mean of empirical optimal decisions in the training data as bias for Portfolio and WSMC. For PTSP we take a bias of 0, as the other approach sometimes get the model stuck in a minima based on empirically optimal decisions.

C Experimental problem details

For completeness a mathematical formulation of each of the experimental problems is presented in this section.

C.1 Portfolio

For clarity the full problem formulation is shown below, given n securities:

For $i \in \{1, \ldots, n\}$:

$x_i \in [0, 1]$:	Percentage investment in security i
$x_0 \in [0, 1]$:	Percentage investment in the bank
$c_i \in \mathbb{R}$:	Return on investment for security i
$\beta \in \mathbb{R}$:	Return on investment for the bank

$$\max_x \ln(1 + \beta x_0 + c^T x)$$
$$\sum_{i=0}^{n} x_i = 1$$

C.2 Weighted-set multi-cover

For $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$:

$x_j \in \mathbb{N}$:	Number of set j chosen
$c_j \in \mathbb{N}$:	Cost for set j
$\xi_i \in \mathbb{N}$:	Coverage requirement for item i
$a_{ij} \in \{0, 1\}$:	If item i is covered by set j, $A = (a_{ij})$, with $a_{ii} = 1, a_{ji} = 0$ for $i, j \in \{1, ..., n\}, i \neq j$
$y_i^+, y_i^- \in \mathbb{N}$:	Unmet/excess coverage item i
$c_i^+, c_i^- \in \mathbb{N}$:	Unmet/excess coverage cost item i

$$\begin{aligned} &\min_x c^T x + \mathbb{E}_\xi[Q(\xi, x)] \\ Q(\xi, x) = &\min_{y^+, y^-} c^{+T} y^+ - c^{-T} y^- \\ &Ax + y^+ - y^- \geq \xi \\ &y_i^- \leq x_i, \quad i \in \{1, \ldots, n\} \end{aligned}$$

C.3 Probabilistic traveling salesperson

Below is a mathematical formulation for the PTSP with set of nodes $N = \{0, 1, \ldots, n\}$ (depot represented by 0), set of customers $N' = N \setminus \{0\}$.

For $i, j \in N, k \in M, i' \in N'$:

$x_{ij} \in \{0, 1\}$:	When arc (i, j) is traversed
$x_i^d \in \{0, 1\}$	When there is a direct trip to i
$y_i \in \{0, 1\}$:	If direct trip to i is canceled
$d_{ij} \in \mathbb{R}$:	Distance between i and j
$\xi_{i'} \in \{0, 1\}$:	Customer i' requires service
$x_i^v = x_i^d + \sum_{j \in N} x_{ji}$	When i is visited

$$\begin{aligned} &\min_x \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ij} + \sum_{i \in N'} 2 d_{0i} x_i^d + \mathbb{E}_\xi[Q(\xi, x)] \\ &\sum_{j \in N} x_{ij} = x_i^v - x_i^d \\ &\sum_{i \in N} x_{ij} = x_j^v - x_j^d \\ &\sum_{i \in N} \sum_{j \in S} x_{ij} = |S| - 1, \\ &\qquad \forall S \subsetneq \{i \in N' : x_i^v = 1, x_i^d = 0\} \\ Q(\xi, x) = &\min_y \sum_{i \in N'} 2 d_{0i} (\rho \xi_i (1 - x_i^v) - y_i) \\ &y_i \leq x_i^d (1 - \xi_i) \end{aligned}$$