Antonis Kokossis, Michael C. Georgiadis, Efstratios N. Pistikopoulos (Eds.)
PROCEEDINGS OF THE 33rd European Symposium on Computer Aided Process Engineering (ESCAPE33), June 18-21, 2023, Athens, Greece
 http://dx.doi.org/10.1016/B978-0-443-15274-0.50216-X

An inverse optimization approach to decision-focused learning

Rishabh Gupta[a], Qi Zhang*[a]

[a]*Department of Chemical Engineering and Materials Science, University of Minnesota, Minneapolis, MN 55455, USA*
** qizh@umn.edu*

Abstract

Decision-focused learning is an emerging paradigm specifically aimed at improving the data-driven learning of input parameters to optimization models. The main idea is to learn predictive models that result in the best decisions rather than focusing on minimizing the parameter estimation error. Virtually all existing works on decision-focused learning only consider the case where the unknown model parameters merely affect the objective function. In this work, extend the framework to also consider unknown parameters in the constraints, where feasibility becomes a major concern. We address the problem by leveraging recently developed methods in data-driven inverse optimization, specifically applying a penalty-based block coordinate descent algorithm to solve the resulting large-scale bilevel optimization problem. The results from our computational case study demonstrate the effectiveness of the proposed approach and highlight its benefits compared with the conventional predict-then-optimize approach, which treats the prediction and optimization steps separately.

Keywords: Decision-focused learning, inverse optimization, constraint learning.

1. Introduction

In traditional data-driven optimization, we often follow a two-step predict-then-optimize approach, i.e. we first predict the unknown model parameters from data with external features and then solve the optimization problem with those predicted inputs. Here, the learning step focuses on minimizing the parameter estimation error; however, this does not necessarily lead to the best decisions (evaluated with the true parameter values) in the optimization step. In contrast, decision-focused learning (Wilder et al., 2019), also known as smart predict-then-optimize (Elmachtoub and Grigas, 2022), integrates the two steps to explicitly account for the quality of the optimization solution in the learning of the model parameters (i.e. minimize the decision error).

Existing works on decision-focused learning, many of which are based on deep learning with differentiable optimization layers (Amos and Kolter, 2017), have shown that significantly improved solutions can be achieved compared to the traditional predict-then-optimize approach. However, virtually all of them consider the case where the unknown model parameters only affect the objective function, which simplifies the problem considerably since feasibility is not a concern. Yet in many applications, we also need to use data to predict parameters in the constraints; the treatment of this case is in theory possible but difficult using existing methods. In this work, we address this problem by leveraging methods that we recently developed for data-driven inverse optimization (Gupta and Zhang, 2022a), which provide a natural way of incorporating constraints with unknown parameters.

The goal of inverse optimization is to infer an unknown optimization model from decisions that are assumed to be optimal solutions to that optimization problem (Chan et al., 2021). While early works primarily addressed the deterministic setting in which observations are assumed to be exactly optimal solutions of the optimization model, more recent contributions focus on the case with multiple noisy observations (Aswani et al., 2018; Chan et al., 2019; Gupta and Zhang, 2022b). Decision-focused learning can be viewed as a data-driven inverse optimization problem by treating the predictive model for the input parameters as the unknown part of the overall optimization model.

In the remainder of this paper, we first present the mathematical formulation of the decision-focused learning problem where we explicitly incorporate constraints that ensure feasibility of the optimal solutions obtained from the model with the estimated input parameters. To solve the resulting large-scale bilevel optimization problem, we apply our recently proposed penalty-based block coordinate descent algorithm. In a computational case study, we demonstrate the effectiveness of the proposed approach and highlight its benefits compared with the conventional predict-then-optimize approach.

2. Mathematical formulation

We assume that the optimization problem to be solved can be generally formulated in the following compact form:

$$\begin{array}{ll} \text{minimize} & f(x,u) \\ \text{subject to} & g(x,u) \leq 0, \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$ are the decision variables, and the model parameters are denoted by u. In this work, we assume that problem (1) is convex, with f and g being differentiable and convex in x. The model parameters u (or a subset of them) change with some external features r and are not exactly known; hence, they need to be estimated from data. The goal is to construct a predictive model $u = m(r)$ given a set of N data points, where each data point i corresponds to a feature-output pair $(\bar{r}_i, \bar{u}_i)$. Given a new r, problem (1) will then be solved using the predicted values $u = m(r)$.

2.1. Conventional predict-then-optimize approach

In the conventional two-step process, the prediction of the model parameters is carried out independent from the later optimization. To obtain a predictive model m, one typically solves an empirical risk minimization problem of the following form:

$$\text{minimize} \quad \frac{1}{N}\sum_{i\in I} \ell(\bar{u}_i, m(\bar{r}_i)), \tag{2}$$

where $I = \{1, \dots, N\}$ denotes the set of data points, and the loss function ℓ is some measure of the difference between the true output $\bar{u}_i$ and the prediction $m(\bar{r}_i)$. The underlying assumption is that if we minimize the difference between the true and predicted parameter values, this will also lead to optimal solutions to problem (1) that are the closest possible to the solutions we would obtain if we knew the true parameter values.

2.2. Decision-focused learning

In decision-focused learning, we integrate the prediction and optimization steps to construct a predictive model that directly takes the quality of the resulting optimization solution into account. We do so by solving the following problem:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{N}\sum_{i\in I} f(\hat{x}_i, \bar{u}_i) && (3a)\\ \text{subject to} \quad & \hat{x}_i \in \arg\min_{\tilde{x}}\{f(\tilde{x}, u) : g(\tilde{x}, u) \le 0, u = m(\bar{r}_i)\} \quad \forall\, i \in I && (3b)\\ & g(\hat{x}_i, \bar{u}_i) \le 0 \quad \forall\, i \in I, && (3c) \end{aligned}$$

where per constraints (3b), $\hat{x}_i$ is an optimal solution to problem (1) with $u = m(\bar{r}_i)$. The objective is to minimize the true cost averaged over the training set I, i.e. it considers the cost of $\hat{x}_i$ evaluated at the true parameter values $\bar{u}_i$ for each $i \in I$. Importantly, constraints (3c) ensure feasibility of each $\hat{x}_i$ given $\bar{u}_i$. These last set of constraints are omitted in virtually all existing works on decision-focused learning since they consider the case in which only the objective function f depends on u such that feasibility is not an issue.

3. Solution approach

The decision-focused learning problem (3) is a bilevel optimization problem with $|I|$ convex optimization problems in its lower-level. We reformulate (3) into a single-level problem by replacing the lower-level problems with their KKT conditions. This results in a nonconvex nonlinear optimization problem which generally lacks regularization. To address the convergence difficulties of standard nonlinear solvers on this problem, we consider a penalty reformulation and apply an efficient block coordinate descent (BCD) algorithm. We do not provide more details about our solution algorithm here but refer the reader to Gupta and Zhang (2022a) for more details. We end this section by highlighting the fact that our approach is restricted to the case where problem (1) is a strictly convex problem and satisfies Slater's condition.

4. Case study

In this section, we apply the proposed decision-focused learning approach to a (single-period) production planning problem for a small interconnected process network. This network, as depicted in Figure 1, consists of 5 materials and 3 processes. The goal is to determine the optimal quantities of raw materials to purchase and the amounts of products to manufacture to satisfy a given demands. This problem can be formulated as follows:

$$\begin{aligned} \text{minimize} \quad & z = \textstyle\sum_{m\in\mathcal{M}}\left(\sum_{p\in\mathcal{P}} c_p y_p^2 + f_m w_m^2\right) && (4a)\\ \text{subject to} \quad & q_m^{\min} \le \mathrm{q}_\mathrm{m}^0 + \left(\sum_{p\in\hat{\mathcal{P}}_m} \mu_{pm} y_p - \sum_{p\in\bar{\mathcal{P}}_m} \mu_{pm} y_p + w_m - d_m\right) && (4b)\\ & \le q_m^{\max} \quad \forall\, m \in \mathcal{M} \\ & 0 \le w_m \le w_m^{\max} \quad \forall\, m \in \mathcal{M} && (4c)\\ & 0 \le y_p \le y_p^{\max} \quad \forall\, p \in \mathcal{P}, && (4d) \end{aligned}$$

where $\mathcal{M}$ and $\mathcal{P}$ are the sets of materials and processes, respectively. Further, the set $\hat{\mathcal{P}}_m$ consists of the processes that can produce material m, and the set $\bar{\mathcal{P}}_m$ contains the processes that consume m. The amount of a reference material produced by process p is denoted by y_p and we use w_m to specify the amount of material m purchased from the market. The conversion factor μ_{pm} determines the amount of a material m produced or consumed by process p for one unit of the reference material. Constraints (4b) restrict the inventory levels while accounting for product demand represented by d_m, (4c) limit the amount of a material that can be acquired from the market, and (4d) set the capacities of processes. The objective is to minimize the total production and material purchasing cost.

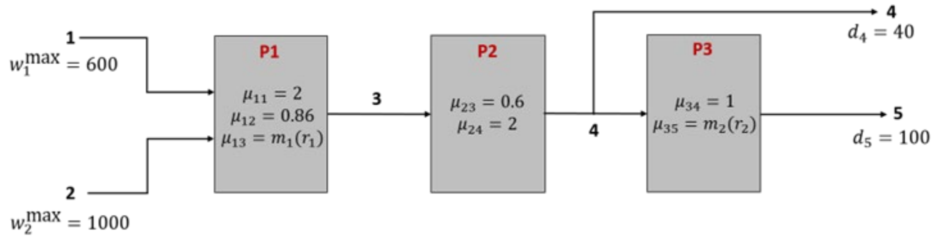

Figure 1 Process network for the production planning problem (4). The minimum and maximum allowed inventory values are 0 and 200, respectively, for all materials. For all processes, the values of $y_p^{\max}$ is set to 400.

For this case study, we consider a scenario where the conversion factors vary based on some observable external feature r. For the sake of simplicity, we assume that the change in most of the conversion factors is negligible; only μ_{13} and μ_{35} deviate significantly enough from their nominal values to affect optimal production decisions. Our goal is to build predictive models for these two uncertain parameters using a data set containing observed (r_1, μ_{13}) and (r_2, μ_{35}) values.

4.1. Synthetic data generation

We now describe the process used to generate the training data set for the case study. We start by assigning models (5a) and (5b) to the uncertain parameters. These are the underlying true models which are assumed to be unknown. To obtain the training data, we sample $|I|$ values of the features r_1 and r_2 from the uniform distributions $\mathbb{U}(0.1, 0.45)$ and $\mathbb{U}(-2, 1)$, respectively. Following that, we evaluate the models for μ_{13} and μ_{35} at each of the sampled feature values to complete the training data set.

$$\mu_{13}(r_1) = \frac{1}{10}(\sin(20\pi r_1) + 7r_1) + 2 \tag{5a}$$

$$\mu_{35}(r_2) = 2 + \frac{1}{10}\left((r_2 - 1)\, r_2\, (r_2 + 2)^2\right) \tag{5b}$$

In order to estimate predictive models m_1 and m_2 with the proposed approach, we assume a hypothesis class consisting of cubic polynomials. We solve problem (3) using our BCD algorithm with training data sets of four different sizes: 10, 25, 50, and 100. The quality of the resulting model estimates is evaluated through a test data set of 100 unseen data points, which is generated using the same parameter generation scheme as the training data set. A model is considered good if the produced $\hat{\mu}$ values result in production decisions that are not only close to the true optimal decisions but also feasible for the true model (i.e., problem (4) with the true μ values).

In addition to the decision-focused approach, we also estimate cubic models for m_1 and m_2 using the traditional two-stage predict-then-optimize approach. Specifically, we use least squares regression to fit a cubic polynomial to the observed data. This estimated model is then used to solve problem (4) for the points in the test data set.

4.2. Results and discussion

Here we compare the performance of the models estimated using the decision-focused and two-stage approaches. Figure 2 compares the plots of the true μ models with their estimates obtained using the two approaches. In all cases, we find that the decision-focused approach constructs an underestimator function for the training data points. This happens because in problem (4), an optimal solution will always be such that inventories of all materials are close to or at their minimum values. If the estimated $\hat{\mu}$ values are such that production gets overestimated, then there is a high probability that the inventory will fall below its permissible value when the process is actually run with a lower conversion value, resulting in an infeasible operation. Therefore, decision-focused learning obtains an underestimate of μ to avoid violating the lower bound on the inventory constraint. From Figure 2, we find that as we provide more training data, the proposed approach finds better underestimators. With 50 data points, it is able to find almost perfect underestimators for both μ parameters.

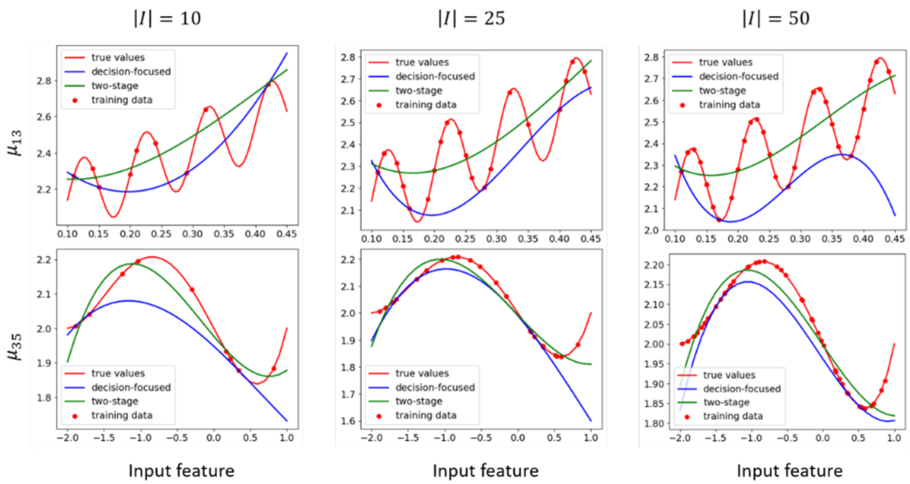

Figure 2 True μ models compared with their approximations estimated using the decision-focused and two-stage learning approaches

In contrast, the goal of the two-stage approach is to build the best approximation of the function itself using the provided data. The estimated model tries to closely mimic the behavior of the actual function to the extent that the assumed hypothesis class allows. This difference in approach leads to differences in performance, as seen in Table 1. The "feasible fraction" column indicates the fraction of the points in the test data set for which the estimated μ models produced a feasible decision. The data in this column shows that the decision-focused approach significantly outperforms the two-stage approach. Moreover, one can see that for the decision-focused case, the better the estimated function underestimates the true function, the higher the fraction of the feasible points. Here since the two-stage approach does not focus on yielding underestimators, the fraction of feasible decisions is very low.

For the test data points that yield feasible decisions, Table 1 also compares the distance of those decisions from the true optimal solutions. While the decision-focused learning generates feasible solutions with a high degree of confidence, these solutions are slightly more different from the true optimal solutions compared to the two-stage approach. However, as can be seen from the last column in the table, which compares the optimality gap of the decisions generated by the two approaches, the mean optimality gap of the decision-focused approach is still less than 10% (compared to ~5% in the two-stage case). This suggests that the decision-focused approach produces high quality decisions while almost guaranteeing their feasibility.

$\lvert I \rvert$	feasible fraction		mean $\frac{\lVert y(\mu)-y(\hat{\mu})\rVert_1}{\lVert y(\mu)\rVert_1}$		mean $\frac{\lVert w(\mu)-w(\hat{\mu})\rVert_1}{\lVert w(\mu)\rVert_1}$		mean $\frac{\lvert z(\mu)-z(\hat{\mu})\rvert}{z(\mu)}$	
	decision -focused	two-stage	decision-focused	two-stage	decision-focused	two-stage	decision-focused	two-stage
10	0.54	0.21	0.04	0.02	0.09	0.06	0.08	0.04
25	0.7	0.32	0.04	0.02	0.11	0.06	0.08	0.05
50	0.9	0.25	0.03	0.03	0.13	0.07	0.07	0.06
100	0.91	0.27	0.03	0.02	0.12	0.06	0.07	0.04

Table 1. A comparison of the performance of the models estimated using the decision-focused and two-stage approaches

5. Conclusions

In this work, we extended the decision-focused learning framework to include cases where the unknown parameters are in the constraints. We used an inverse optimization approach in which the problem is formulated as a bilevel program. Our approach allows inclusion of constraints that, with a high degree of confidence, ensure that the estimated model produces decisions that remain feasible for the true model. We illustrated our approach by applying it on a production planning problem with unknown process parameters. Our results show that the models obtained using decision-focused learning produce feasible decisions at a significantly higher rate compared to traditional two-stage learning without substantially sacrificing the optimality of these decisions.

Acknowledgements

The authors gratefully acknowledge the financial support from the National Science Foundation under Grant #2044077. R.G. acknowledges financial support from a departmental fellowship sponsored by 3M and a Doctoral Dissertation Fellowship from the University of Minnesota.

References

Amos, B. and Kolter, J.Z., 2017. Optnet: Differentiable optimization as a layer in neural networks. *Proceedings of the International Conference on Machine Learning*, pp. 136-145.

Aswani, A., Shen, Z.-J. M., and Siddiq, A., 2018. Inverse optimization with noisy data. *Operations Research, 66(3),* 870–892.

Chan, T. C., Lee, T., and Terekhov, D., 2019. Inverse optimization: Closed-form solutions, geometry, and goodness of fit. *Management Science, 65(3)*, 1115–1135.

Chan, T.C., Mahmood, R., and Zhu, I.Y., 2021. Inverse optimization: Theory and applications. arXiv:2109.03920.

Elmachtoub, A.N. and Grigas, P., 2022. Smart “predict, then optimize”. *Management Science*, *68*(1), pp. 9-26.

Gupta, R. and Zhang, Q., 2022a. Efficient learning of decision-making models: A penalty block coordinate descent algorithm for data-driven inverse optimization. arXiv:2210.15393.

Gupta, R. and Zhang, Q., 2022b. Decomposition and adaptive sampling for data-driven inverse linear optimization. *INFORMS Journal on Computing*.

Wilder, B., Dilkina, B., and Tambe, M., 2019. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1658-1665.