Learning to optimize with hidden constraints

Aaron Babier, Timothy C. Y. Chan
Mechanical & Industrial Engineering, University of Toronto, Toronto, Canada, {ababier, tcychan}@mie.utoronto.ca
Adam Diamant
Schulich School of Business, York University, Toronto, Canada, adiamant@schulich.yorku.ca
Rafid Mahmood
Mechanical & Industrial Engineering, University of Toronto, Toronto, Canada, rafid.mahmood@mail.utoronto.ca

We consider a data-driven framework for learning to generate decisions to instances of continuous optimization problems where the feasible set varies with an instance-specific auxiliary input in a way that cannot be formally described. We use a data set of inputs and feasible solutions, as well as an oracle of feasibility to iteratively train two machine learning models. The first model is a binary classifier of feasibility; this model then serves as a barrier function to train the second model via an interior point method. We develop a theory and optimality guarantees for interior point methods when given a barrier that relaxes the feasible set, and extend these results to obtain probabilistic guarantees for our classification and generative models. Finally, we implement our method on a radiation therapy treatment planning problem to predict personalized treatments for head-and-neck cancer patients.

1. Introduction

Consider a decision-maker that regularly solves different instances of a continuous optimization problem. There is a fixed objective function and a common set of constraints over all instances. However, each instance also includes some exogenous input that may change the feasible set in a way that cannot be easily characterized. This input is not a parameter, but rather, auxiliary data that maps to latent constraints that are not amenable to formal mathematical representation.

While conventional optimization methods can produce optimal solutions for the common problem (i.e., the fixed objective and common constraints), these approaches make limited use of the auxiliary data because the latent constraints cannot be described mathematically. For example, a decision-maker may only be able to determine feasibility for any candidate decision via an oracle. The common problem is a relaxation of the instance-specific formulation and if the latent

constraints are not considered, the solution to the common problem may not be optimal for the problem instance. In practice, this approach still provides a starting point for the decision-maker: they can manually modify the output and obtain an instance-specific solution. However, these post-hoc refinements suggest that the final solutions may not be provably optimal decisions.

Systematic approaches may attempt to predict the effect of the auxiliary data on the optimization model. Since these techniques are data-driven, they rely on the collection of past inputs and decisions as well as a prediction model to characterize the latent behavior. In operations research, this has been studied in the context of the auxiliary data affecting the objective. For example, local regression and neighbor or tree-based models can be placed in a conditional stochastic objective so as to penalize decisions predicted to be of poor quality (Bertsimas and Kallus 2019). An alternative is to recast the common problem as a parameterized one and use machine learning to predict the instance-specific parameters from the auxiliary input (Angalakudati et al. 2014, Ferreira et al. 2015, Babier et al. 2018a, Liu et al. 2018). This can be implemented by first predicting the quantities of interest and then optimizing over them or by embedding the machine learning model directly into the optimization problem and solving both simultaneously (Elmachtoub and Grigas 2017, Ban and Rudin 2018). Regardless of the approach, these techniques assume that the feasible region can be described mathematically and thus, the problem reduces to learning an objective function.

Since the common problem is well-specified and both past inputs and implemented decisions exist, it should be possible to learn a representation of the latent constraints in order to directly predict the instance-specific optimal decisions (Larsen et al. 2018). While the notion of using deep learning to predict optimal solutions to an optimization problem is not novel (Hopfield and Tank 1985, Bengio et al. 2018), there are no results that guarantee the optimality of the predicted solutions for a general input. Thus, in this paper, we consider a framework for learning to generate optimal decisions to a continuous constrained optimization problem with a fixed objective and a feasible region characterized by the latent constraints of the decision-maker. By combining techniques from deep learning and operations research, we capture the best of both worlds; learning unstructured mappings between inputs and decisions while also proving mathematical guarantees on solution quality and generalization for out-of-sample problem instances.

1.1. Motivating application

Our work is motivated by the problem of automatically generating personalized radiation therapy (RT) treatment plans for patients diagnosed with cancer. RT is one of the primary methods for cancer treatment and is recommended for over 50% of all diagnoses (Delaney et al. 2005). In RT, a linear accelerator delivers beamlets of radiation from different angles to a tumor. To construct a treatment plan, a dosimetrist solves an optimization problem that is meant to capture the trade-offs between ensuring sufficient dose to the tumor while minimizing the effect on healthy tissue. However, the plan must be approved by an oncologist before administration. This approval is based on its performance across several dosimetric criteria. Since it is impossible to satisfy all criteria simultaneously (e.g., tumor dose may be sacrificed to reduce dose to nearby critical structures or vice versa), oncologists make subjective trade-offs based on their expertise and prior experience. These oncologist-driven trade-offs can be interpreted as latent constraints that are parameterized by the patient's information. The latent constraints are difficult to express a priori but can be learned by examining past decisions (i.e., deliverable treatment plans) approved by the oncologist.

The current clinical practice is an iterative and time-consuming process where the dosimetrist and oncologist work together to generate acceptable treatment plans. The dosimetrist tunes the parameters of a surrogate multi-objective optimization problem and solves it to generate a candidate treatment. The oncologist is a membership oracle who reviews the solution, determines whether it is satisfactory, and if not, suggests areas of improvement. The dosimetrist then re-parametrizes the optimization problem and generates a new solution. Multiple iterations are typically required and it can take several days to generate a single treatment plan, especially for complex cases.

By analyzing past inputs and deliverable treatments, the oncologist's experience in determining satisfactory plans can be automated using predictive modeling. Several recent papers have demonstrated that automated treatment plan generation using machine learning techniques in conjunction with optimization is a viable alternative to the manual process (e.g., Shiraishi et al. 2015, McIntosh and Purdie 2017, Babier et al. 2018a). This area is broadly known as knowledge-based

planning (KBP). First, a machine learning model trained on previously delivered plans predicts an acceptable dose distribution (Shiraishi et al. 2015, McIntosh et al. 2017, Babier et al. 2018b, Mahmood et al. 2018). The prediction is used as input into an optimization model that generates a treatment plan (i.e., the set of beamlets) that yields a similar dose distribution as the prediction.

Since the KBP framework involves two distinct stages, prediction then optimization, there is no guarantee that the final plans satisfy the latent oncologist constraints. This may lead to delays in treatment delivery as the dosimetrist and oncologist may need additional iterations to manually correct the plans. Further, there is no assurance that the final oncologist-approved plans are near optimal. Thus, the final plans prescribed by the oncologist may not be the best possible treatment.

1.2. Contribution

Our approach involves transforming the "true" optimization problem (the model with the latent feasible set) into a prediction problem solved via two machine learning models. First, a binary classifier determines whether a decision is feasible for a specific instance. Then, a generative model navigates the instance-specific feasible region characterized by the support of the classifier to return an optimal solution. The two models are trained sequentially over several iterations; after each iteration, an oracle of feasibility labels the current predictions of the generative model as feasible or infeasible. The newly labelled data then helps train the classifier in the following iteration.

In our training algorithm, the generative model navigates the support of the classifier via an interior point method (IPM). However, IPMs are effective primarily when the feasible set is fully known. As we simultaneously learn feasibility along with optimality, our classifier does not enjoy the conventional properties of a canonical barrier. Therefore, we introduce the notion of a weak barrier function—one that may not perfectly discriminate feasibility. We derive a new ϵ-optimality guarantee for optimization when given only a weak barrier function. We then show that our generative model, which now predicts solutions rather than optimizing for them, enjoys similar guarantees both for in-sample testing and out-of-sample problem instances. Specifically,

1. We introduce the concept of a δ-*barrier*, which is a barrier function for a relaxation of a feasible set. We characterize the (δ, ϵ)-optimality guarantee for solutions from optimization using a δ-barrier generalizing key results of IPMs to the setting of a partially specified feasible set.

2. We introduce Interior Point Methods with Adversarial Networks (IPMAN), an iterative, oracle-guided algorithm for learning to predict optimal solutions when given instance information to problems with latent constraints. The classifier trained in this algorithm approximates a δ-barrier, the generative model generates (δ, ϵ)-optimal solutions on training instances, and the oracle-guided data augmentation guarantees that the classifier improves in every iteration.
3. We prove a generalization bound on the (δ, ϵ)-optimality gap of any model that predicts solutions to a random problem instance. This bound holds for IPMAN, meaning that both in-sample and out-of-sample error from the optimal value can be evaluated.

All proofs are available in the Electronic Companion.

2. Background and related work

This paper brings together ideas from several fields. First, the concept of using two learning models, one to evaluate feasibility and one to generate solutions, is a standard approach in reinforcement learning (e.g., Konda and Tsitsiklis 2000), and deep learning (e.g., Goodfellow et al. 2014). The specific practice of training a machine learning model using an oracle is known as imitation learning (Bain and Sammut 1999). Further, our loss function and optimality guarantees are derived using the theory of interior point methods (Nemirovskii and Nesterov 1994), while our learning guarantees extend recent results on Rademacher complexity for data-driven optimization (Bertsimas and Kallus 2019). Finally, the learning performed by our generative model bears a loose resemblance to estimation of distribution algorithms (EDAs), commonly used in evolutionary and black-box optimization (Pelikan et al. 2002). Because our work is most closely tied to interior point methods and the "learning to optimize" literature, we focus specifically on those two areas below.

2.1. Interior point methods

Interior point methods are among the most popular techniques for solving constrained optimization problems (Nemirovskii and Nesterov 1994). A constrained problem $\min_{\mathbf{x}} \{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{X}\}$ is transformed into an unconstrained problem via a barrier function $B(\mathbf{x})$. The barrier satisfies two properties: (i) $B(\mathbf{x}) = 0$ (and thus $\log B(\mathbf{x}) = -\infty$) when $\mathbf{x}$ is infeasible, and (ii) $B(\mathbf{x}) > 0$ when

$\mathbf{x}$ is strictly feasible. The resulting problem is $\min_{\mathbf{x}}\{f(\mathbf{x}) - \lambda \log B(\mathbf{x})\}$ where $\lambda > 0$ is the dual parameter.

Given a differentiable barrier function and an initial solution $\mathbf{x}^{(0)}$, IPMs use the Newton method to iterate over a sequence $\{(\lambda_j, \mathbf{x}^{(j)})\}_{j=0}^{\infty}$ until convergence to an optimal solution (Boyd and Vandenberghe 2004). These methods have found the most success in linear and quadratic optimization where it is possible to give theoretical guarantees on optimality as well as fast empirical convergence rates (Gondzio 2012). However, recent development of a new class of barrier functions known as entropic barriers have renewed interest in interior point methods for challenging (i.e., arbitrary) convex feasible sets (Bubeck and Eldan 2019). The most similar work to ours is by Badenbroek and de Klerk (2018) who solve a sampling-based IPM using a membership oracle for the feasible set and the aforementioned entropic barrier. IPMs have also been adapted for non-convex optimization problems (Vanderbei and Shanno 1999, Benson et al. 2004, Hinder and Ye 2018) although in this setting, efficiency guarantees generally do not exist.

The previous papers all assume access to either explicit constraints or at least a barrier function for the entire feasible set. In this work, we assume access to only a polyhedral relaxation of the true feasible set and learn feasibility via a classification model trained on labelled decisions. Furthermore, the previous papers focus on determining efficiency and complexity analyses of their algorithm for a single given instance of an optimization problem. Because we consider the problem of learning to optimize over a set of problem instances, we focus instead on out-of-sample guarantees for the model rather than efficiency of the IPM algorithm.

2.2. Learning to construct optimal solutions

2.2.1. The operations research perspective. Learning to construct optimal solutions from auxiliary feature data is, in most cases, performed by embedding the output of a machine learning model as parameters in an optimization model (Angalakudati et al. 2014, Ferreira et al. 2015, Elmachtoub and Grigas 2017, Liu et al. 2018). This approach is particularly effective when there exists an important parameter in the problem and a clear relationship between it and the features (e.g., the demand in a revenue model (Ferreira et al. 2015)).

Non-parametric methods construct functions that map feature data to the optimal solution or the optimal value of the problem. This is often a stochastic optimization problem with a random objective conditioned on the input. Kao et al. (2009), Ban and Rudin (2018), and Bertsimas and Kallus (2019) consider Empirical Risk Minimization (ERM) where the objective is to learn a function that takes auxiliary data as input and outputs an optimal solution. However, it is often difficult to use advanced, expressive models while preserving computational tractability. Consequently, Hannah et al. (2010), Bertsimas and Kallus (2019), and Bertsimas and McCord (2018) consider a weighted learning framework, where the goal is to obtain a function that determines the weights (i.e., the conditional probability terms) using a sample-average approximation of the stochastic optimization problem.

Our work shares a similar approach to Ban and Rudin (2018) who use ERM to construct a function for the optimal solution to the newsvendor problem. As in their paper, we study out-of-sample generalization of the learning model. However, the key difference is that the newsvendor problem is only constrained by the non-negativity of the order quantities. In contrast, our focus on using ERM applies to a more general set of constraints. Our key contribution is to incorporate constraint satisfaction as the output of a binary classification model.

Bertsimas and Kallus (2019) remark on the challenges of constraint satisfaction when using the ERM approach and, consequently, focus on weighted learning. However, they also prove several generalizability results arising from ERM. Our paper further explores the avenue introduced by Bertsimas and Kallus (2019) by extending their generalization bound to our IPM framework.

2.2.2. The deep learning perspective. Recent advances in deep learning have prompted a resurgent interest in using neural networks to solve optimization problems (Bengio et al. 2018). Generally, a model is trained by minimizing a loss function that encourages the model to output optimal solutions to problem instances. While the literature mostly focuses on benchmark combinatorial problems such as the Traveling Salesman Problem (TSP) (Vinyals et al. 2015, Bello et al. 2017, Dai et al. 2017), there is an increasing interest in other operational applications (Donti et al.

2017, Larsen et al. 2018). Methodologically, Vinyals et al. (2015) and Larsen et al. (2018) use supervised learning, where a data set of problem instances and optimal solutions are used to train the model. On the other hand, Bello et al. (2017) and Dai et al. (2017) train a reinforcement learning agent to navigate the space of decisions. Finally, Donti et al. (2017) consider a gradient-descent algorithm that encourages predicting feasible and optimal solutions.

A major challenge in learning to predict optimal solutions is that it is difficult to enforce challenging constraints using a predictive model. The design of the neural network architecture sometimes naturally enforces certain structural constraints. For example, a pointer network is a recurrent neural network that returns permutations of a sequence making it ideal for satisfying tour constraints in a TSP (Vinyals et al. 2015). Alternatively, if the learning process is supervised, simply using a high-quality data set may be empirically sufficient (Larsen et al. 2018). A third approach is to customize the loss function to encourage constraint satisfaction (Donti et al. 2017). Regardless of the approach, a major benefit of learning to predict solutions over optimization is the speed at which solutions are produced. That is, after training has concluded, the model requires a simple function call (e.g., a neural network) to return a solution. Thus, the deep learning approaches produce heuristics that are significantly faster than conventional solvers (Bello et al. 2017, Larsen et al. 2018). The drawback is that they do not admit formal optimality guarantees.

We preserve the efficiency of learning algorithms while addressing the problem of constraint satisfaction. More specifically, we consider constraints that are not formally stated, but rather, are implicitly provided via data and an oracle. Further, we provide optimality guarantees on the solutions generated by the deep learning model as well as characterizing the out-of-sample error.

3. Constrained optimization with a partially specified feasible set

We define the problem of solving a constrained optimization problem when the feasible set is specified only by common constraints and auxiliary data. We introduce notation in Section 3.1, before describing the problem and assumptions in Section 3.2. In Section 3.3, we propose a new barrier problem and prove both optimality and feasibility guarantees for our generalization.

3.1. Notation

We denote vectors by bold and sets by calligraphic script. The interior, boundary, and closure of a set are denoted $\text{int}(\mathcal{X})$, $\text{bd}(\mathcal{X})$, and $\text{cl}(\mathcal{X})$ respectively. The exclusion of $\mathcal{X}_1$ from a superset $\mathcal{X}_2 \supseteq \mathcal{X}_1$ is denoted as $\mathcal{X}_2 \setminus \mathcal{X}_1$. We denote probability distributions using $\mathbb{P}$ and absolutely continuous distributions as $p(\mathbf{x})$. The support of a probability distribution is denoted $\text{supp}(\mathbb{P})$. Samples from a random variable $\mathbf{x} \sim \mathbb{P}$ are accented $\hat{\mathbf{x}} \sim \mathbb{P}$. $\|\cdot\|$ refers to the l_2 norm unless specified otherwise. A function $f(\mathbf{x})$ is L-Lipschitz continuous in $\mathbf{x}$ if there exists a constant $L > 0$ such that $|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq L \|\mathbf{x}_1 - \mathbf{x}_2\|$ for all $\mathbf{x}_1, \mathbf{x}_2$.

3.2. Problem setup

Let $\mathbf{u} \in \mathcal{U}$ denote auxiliary inputs that describe our optimization problem and $\mathbb{P}_{\mathbf{u}}$ the probability distribution of inputs. Let $\mathbf{x} \in \mathbb{R}^n$ denote decisions and for a given $\mathbf{u}$, consider the problem

$$\mathbf{OP}(\mathbf{u}): \quad \min_{\mathbf{x}} \{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u})\}$$

where $f(\mathbf{x})$ is an objective function and $\mathcal{X}(\mathbf{u})$ is a feasible set determined by $\mathbf{u}$. We assume $f(\mathbf{x}) = \mathbf{f}^\intercal \mathbf{x}$ is linear without loss of generality; that is, our results extend to convex objectives with minor modifications. We also make the following assumptions about $\mathcal{X}(\mathbf{u})$:

1. The space of inputs $\mathcal{U}$ is compact, continuous, and has a non-empty interior.
2. For all $\mathbf{u} \in \mathcal{U}$, the feasible set $\mathcal{X}(\mathbf{u})$ is compact, continuous, and has a non-empty interior. Furthermore, the joint set $\{(\mathbf{x}, \mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$ is compact.
3. We know a full-dimensional polyhedral relaxation $\mathcal{P} = \left\{\mathbf{x} \mid \mathbf{a}_m^\intercal \mathbf{x} \leq b_m, m = 1, \ldots, M\right\}$ such that $\mathcal{X}(\mathbf{u}) \subset \text{int}(\mathcal{P})$ for all $\mathbf{u}$.
4. Although we do not know $\mathcal{X}(\mathbf{u})$ a priori, we have access to:
   (a) A data set of feasible decision and input pairs $\mathcal{D} = \{(\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i)\}_{i=1}^{N_{\mathbf{x}}}$, where $\hat{\mathbf{x}}_i \in \mathcal{X}(\hat{\mathbf{u}}_i)$ for all $i \in \{1, \ldots, N_{\mathbf{x}}\}$. In general, the data set includes multiple feasible decisions per input. This data is sampled i.i.d. from a distribution $\mathbb{P}_{(\mathbf{x},\mathbf{u})}$.
   (b) A feasibility oracle $\Psi(\mathbf{x}, \mathbf{u})$ where $\Psi(\mathbf{x}, \mathbf{u}) = 1$ if $\mathbf{x} \in \mathcal{X}(\mathbf{u})$ and $\Psi(\mathbf{x}, \mathbf{u}) = 0$ otherwise.

The first assumption ensures that the auxiliary inputs belong to a well-behaved set (i.e., compact, continuous, and non-empty). The second and third assumptions ensure that the instance-specific feasible sets have similar requirements. Set $\mathcal{P}$ represents the space defined by the common constraints that all instances must satisfy in addition to the instance-specific constraints. Finally, the fourth assumption describes the available data to construct an optimal solution to $\mathbf{OP}(\mathbf{u})$. In practice, the data set of feasible solutions may represent previously implemented decisions which are required to learn an approximation of the feasible sets $\mathcal{X}(\mathbf{u})$. The oracle $\Psi(\mathbf{x}, \mathbf{u})$ helps guide the search for an optimal solution by ensuring that any constructed solution is feasible.

3.3. Optimization with a δ-barrier

Consider an instance $\mathbf{OP}(\mathbf{u})$ for a fixed $\mathbf{u}$. Although $\mathcal{X}(\mathbf{u})$ is unspecified, we know a relaxation $\mathcal{X}(\mathbf{u}) \subset \mathcal{P}$. Were the relaxation tight (i.e., $\mathcal{P} = \mathcal{X}(\mathbf{u})$), then we could consider a canonical barrier function (i.e., defining $B(\mathbf{x})$ such that $\log B(\mathbf{x}) = \sum_{m=1}^{M} \log\left(b_m - \mathbf{a}_m^\intercal \mathbf{x}\right)$) and use an IPM to obtain an optimal solution $\mathbf{x}^*(\mathbf{u})$ to $\mathbf{OP}(\mathbf{u})$ (Nemirovskii and Nesterov 1994). However, if $\mathcal{X}(\mathbf{u}) \subset \mathcal{P}$, then the barrier incorrectly returns $B(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathcal{P} \setminus \mathcal{X}(\mathbf{u})$.

For this problem, the canonical barrier belongs to a class of functions that are barriers over a relaxation or super-set of $\mathcal{X}(\mathbf{u})$. Consider functions $B_\delta(\mathbf{x}, \mathbf{u})$ such that $B_\delta(\mathbf{x}, \mathbf{u}) > 0$ for all $\mathbf{x} \in \mathcal{X}(\mathbf{u})$, and $B_\delta(\mathbf{x}, \mathbf{u}) = 0$ for all $\mathbf{x}$ that are sufficiently far from $\mathcal{X}(\mathbf{u})$. We define these functions as δ-barriers.

Definition 1. For some $\delta > 0$, let $\mathcal{N}_\delta\left(\mathcal{X}(\mathbf{u})\right) = \{\mathbf{x} + \boldsymbol{\epsilon} \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \|\boldsymbol{\epsilon}\| < \delta\}$ be a δ-neighbourhood around $\mathcal{X}(\mathbf{u})$. A δ-barrier $B_\delta(\mathbf{x}, \mathbf{u}) : \mathbb{R}^n \times \mathcal{U} \to [0, 1)$ is a function that satisfies

$$\mathcal{X}(\mathbf{u}) \subset \{\mathbf{x} \mid B_\delta(\mathbf{x}, \mathbf{u}) > 0\} \subseteq \mathcal{N}_\delta\left(\mathcal{X}(\mathbf{u})\right).$$

Note that for a given barrier function $B(\mathbf{x}, \mathbf{u})$ supported over a super-set of $\mathcal{X}(\mathbf{u})$, δ is equivalent to the Hausdorff distance between $\mathcal{X}(\mathbf{u})$ and the support of the function, i.e.,

$$\delta = d_H\big(\mathcal{X}(\mathbf{u}), \{\mathbf{x} \mid B(\mathbf{x}, \mathbf{u}) > 0\}\big) = \min_{\xi \geq 0} \big\{\xi \;\big|\; \{\mathbf{x} \mid B(\mathbf{x}, \mathbf{u}) > 0\} \subseteq \mathcal{N}_\xi\left(\mathcal{X}(\mathbf{u})\right)\big\}. \tag{1}$$

Remark 1. Let $\Delta(\mathbf{u}) = d_H(\mathcal{X}(\mathbf{u}), \mathcal{P})$. A canonical barrier (i.e., $\log B(\mathbf{x}) = \sum_{m=1}^{M} \log\left(b_m - \mathbf{a}_m^\intercal \mathbf{x}\right)$) for $\mathcal{P}$ is a $\Delta(\mathbf{u})$-barrier for $\mathcal{X}(\mathbf{u})$. As $\mathcal{P}$ is known, we assume access to barrier for which $\delta \leq \Delta(\mathbf{u})$.

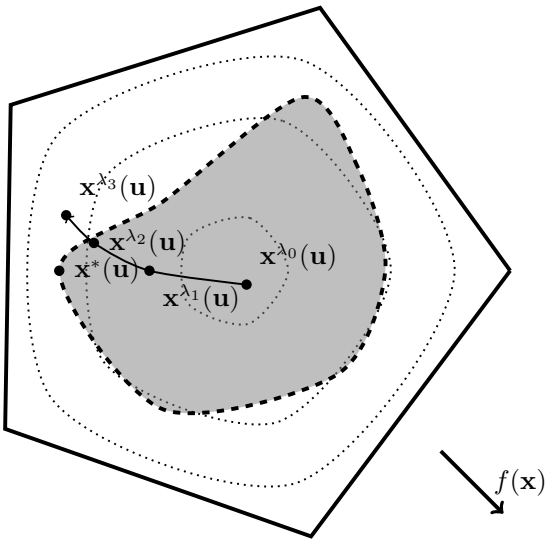


Figure 1 **The bold shape is $\mathcal{P}$ and the filled region is $\mathcal{X}(\mathbf{u})$. We consider a canonical barrier for $\mathcal{P}$. The dotted lines are contours for the barrier. An optimal solution to $\mathbf{OP}(\mathbf{u})$ is $\mathbf{x}^*(\mathbf{u})$.**

Given a δ-barrier $B_\delta(\mathbf{x}, \mathbf{u})$, let $\lambda > 0$ be a constant corresponding to the Lagrangian dual variable. We then define the unconstrained barrier optimization problem

$$\mathbf{BP}(\mathbf{u}, B_\delta, \lambda): \quad \min_{\mathbf{x}} \{f(\mathbf{x}) - \lambda \log B_\delta(\mathbf{x}, \mathbf{u})\} \tag{2}$$

The optimal value of $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is bounded by the optimal value of $\mathbf{OP}(\mathbf{u})$.

THEOREM 1. *Let $\mathbf{x}^*(\mathbf{u})$ be an optimal solution to $\mathbf{OP}(\mathbf{u})$. For any $\lambda > 0$, $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is bounded and feasible. An optimal solution $\mathbf{x}^\lambda(\mathbf{u})$ to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is (δ, ϵ)-optimal for $\mathbf{OP}(\mathbf{u})$:*

$$f\left(\mathbf{x}^\lambda(\mathbf{u})\right) - \epsilon < f\left(\mathbf{x}^*(\mathbf{u})\right) < f\left(\mathbf{x}^\lambda(\mathbf{u})\right) + \delta L, \tag{3}$$

where L is the Lipschitz constant of $f(\mathbf{x})$ and $\epsilon = C\lambda$, where C is a positive constant.

The (δ, ϵ)-optimality inequality proved in Theorem 1 generalizes the classical ϵ-optimality bound of IPMs (Nemirovskii and Nesterov 1994). That is, when $\delta = 0$, we obtain $f(\mathbf{x}^\lambda(\mathbf{u})) - \epsilon < f(\mathbf{x}^*(\mathbf{u})) < f(\mathbf{x}^\lambda(\mathbf{u}))$. Furthermore, similar to classical IPMs, the (δ, ϵ)-optimality of solutions to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ can be controlled by tuning λ. Specifically because $\epsilon = C\lambda$ for a fixed C, as λ goes to 0, so does ϵ.

In a classical IPM, the barrier problem is repeatedly solved over a sequence of decreasing λ (Boyd and Vandenberghe 2004). In that setting, a large λ guarantees that the barrier problem yields

solutions in the interior of the feasible set, while small λ guarantees solutions close to the true optimum. Here however, ϵ may decrease but δ is a property of the barrier itself. Therefore, we can only guarantee that for large λ, an optimal solution $\mathbf{x}^\lambda(\mathbf{u})$ is sub-optimal for $\mathbf{OP}(\mathbf{u})$ but as λ decreases, the optimal solution may have lower objective function value, i.e., become infeasible. Figure 1 shows a sample sequence $\{(\lambda_j, \mathbf{x}^{(j)})\}_{j=0}^{3}$ of decreasing $\lambda_0 > \cdots > \lambda_3$ and corresponding solutions $\mathbf{x}^\lambda(\mathbf{u})$. In the Companion EC.3, we show that this behavior holds more generally and formally prove that the δ-barrier IPM inherits many of the same properties as the classical IPM.

4. Interior Point Methods with Adversarial Networks

In this section, we develop the main contribution of the paper, which is a learning-based approach to solving instances of $\mathbf{OP}(\mathbf{u})$. We first introduce our main algorithm that iteratively trains two machine learning models. The algorithm consists of (i) a classifier that learns to distinguish feasibility for $\mathbf{OP}(\mathbf{u})$, and thus, approximates a δ-barrier for any $\mathbf{u}$; and (ii) a generative model that uses the classifier to learn to predict (δ, ϵ)-optimal solutions to $\mathbf{OP}(\mathbf{u})$ for a given $\mathbf{u}$.

We require a data set of feasible solutions $\mathcal{D}$ as well as an oracle $\Psi(\mathbf{x}, \mathbf{u})$ as introduced in Section 3.2. Furthermore, let $\hat{\mathcal{U}} = \{\hat{\mathbf{u}}_i\}_{i=1}^{N_\mathbf{u}} = \{\hat{\mathbf{u}} \mid \exists \hat{\mathbf{x}} : (\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \mathcal{D}\}$ denote a data set of inputs. In general, $N_\mathbf{u} \leq N_\mathbf{x}$ as this data set is obtained by collecting the unique auxiliary inputs from $\mathcal{D}$. We also assume an additional data set of *infeasible solutions*, $\bar{\mathcal{D}} = \{(\hat{\mathbf{x}}_{\bar{i}}, \hat{\mathbf{u}}_{\bar{i}})\}_{\bar{i}=1}^{\bar{N}_\mathbf{x}}$ where $\hat{\mathbf{x}}_{\bar{i}} \in \mathbb{R}^n \setminus \mathcal{X}(\hat{\mathbf{u}}_{\bar{i}})$. This data set arrives from a distribution $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$, similar to $\mathcal{D} \sim \mathbb{P}_{(\mathbf{x},\mathbf{u})}$. Unlike $\mathcal{D}$, however, $\bar{\mathcal{D}}$ is not assumed a priori, but can be instead generated by sampling outside the relaxation, i.e., $\hat{\mathbf{x}}_{\bar{i}} \in \mathbb{R}^n \setminus \mathcal{P}$.

4.1. Overview of the main algorithm

Let $\mathcal{F} = \{F : \mathcal{U} \to \mathbb{R}^n\}$ denote a class of models that predict solutions to $\mathbf{OP}(\mathbf{u})$. Let $\mathcal{B} = \{B : \mathbb{R}^n \times \mathcal{U} \to [0,1]\}$ denote a class of binary classifiers. In each iteration, we first train $B(\mathbf{x}, \mathbf{u}) \in \mathcal{B}$ to correctly label points in $\mathcal{D}$ and $\bar{\mathcal{D}}$. We then repeatedly re-train $F(\mathbf{u}) \in \mathcal{F}$ over $\hat{\mathcal{U}}$ using an IPM that uses the binary classifier as the barrier function. After each step of the IPM, the oracle $\Psi(\mathbf{x}, \mathbf{u})$ labels the solutions generated by $F(\mathbf{u})$ as either feasible or infeasible. At the end of the iteration, we add these points to $\mathcal{D}$ and $\bar{\mathcal{D}}$. In this way, the training data for the classifier is augmented so

that we iteratively learn a δ-barrier that more closely approximates $\mathcal{X}(\mathbf{u})$ for any $\mathbf{u}$. Let k index the iterations, starting from 0. The k-th iteration proceeds as follows:

1. Train the classifier $B \in \mathcal{B}$ to predict $(\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}$ as feasible and $(\hat{\mathbf{x}}_{\bar{i}}, \hat{\mathbf{u}}_{\bar{i}}) \in \bar{\mathcal{D}}^{(k)}$ as infeasible solution-input pairs by solving the Feasibility Classification Problem:

$$\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)}): \quad \sup_{B \in \mathcal{B}} \left\{ \frac{1}{N_{\mathbf{x}}} \sum_{i=1}^{N_{\mathbf{x}}} \log B(\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) + \frac{1}{\bar{N}_{\mathbf{x}}} \sum_{\bar{i}=1}^{\bar{N}_{\mathbf{x}}} \log\left(1 - B(\hat{\mathbf{x}}_{\bar{i}}, \hat{\mathbf{u}}_{\bar{i}})\right) \right\}. \tag{4}$$

   The objective function is known as the cross-entropy loss function in machine learning (Goodfellow et al. 2016). Let $B^{(k)}$ be an optimal solution to $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$. The optimal value is 0 and achieved when $B^{(k)}$ satisfies $B^{(k)}(\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) = 1$ and $B^{(k)}(\hat{\mathbf{x}}_{\bar{i}}, \hat{\mathbf{u}}_{\bar{i}}) = 0$.

2. Fix an initial dual parameter $\lambda_0 > 0$ and a decay rate $0 < \nu < 1$. Let $M > 0$ denote the number of IPM steps and, for $j \in \{0, \dots, M\}$, let $\lambda_j = \lambda_0 \nu^j$ denote the dual parameter. We train the generative model to predict optimal solutions to $\mathbf{BP}(\hat{\mathbf{u}}_i, B^{(k)}, \lambda_j)$ for all $\hat{\mathbf{u}}_i \in \hat{\mathcal{U}}$. This task is referred to as the Generative Barrier Problem:

$$\mathbf{GBP}(\hat{\mathcal{U}}, B^{(k)}, \lambda_j): \quad \min_{F \in \mathcal{F}} \left\{ \frac{1}{N_{\mathbf{u}}} \sum_{i=1}^{N_{\mathbf{u}}} f\big(F(\hat{\mathbf{u}}_i)\big) - \lambda_j \log B^{(k)}\big(F(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i\big) \right\}. \tag{5}$$

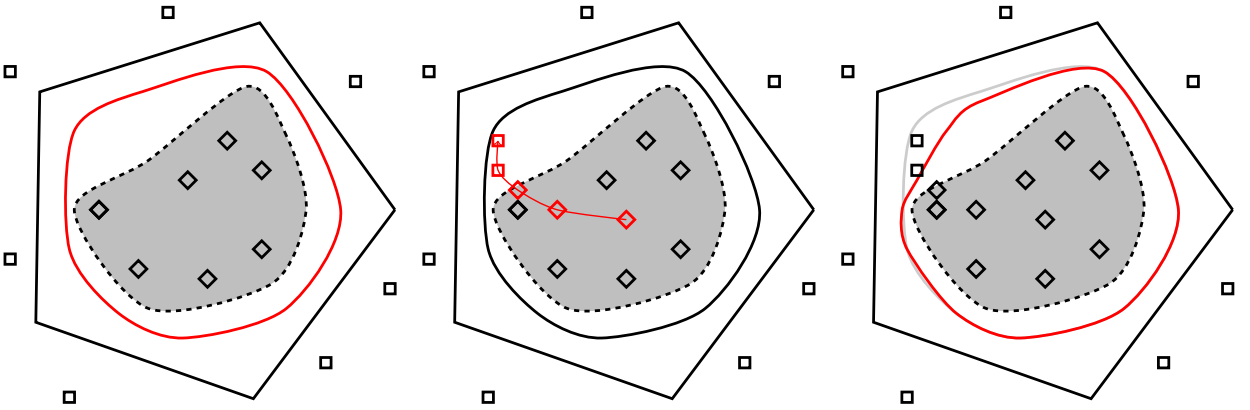
   Let $F^{(j,k)}$ be an optimal solution to $\mathbf{GBP}(\hat{\mathcal{U}}, B^{(k)}, \lambda_j)$. Whereas $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ directly optimizes for a single $\mathbf{u}$, $\mathbf{GBP}(\hat{\mathcal{U}}, B^{(k)}, \lambda_j)$ is an empirical risk minimization problem that trains $F^{(j,k)}(\mathbf{u})$ to predict $\mathbf{x}^{\lambda_j}(\mathbf{u})$. Furthermore, we now use the classifier $B^{(k)}(\mathbf{x}, \mathbf{u})$ as the δ-barrier.

3. For each pair $(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i)$, use the oracle $\Psi(\mathbf{x}, \mathbf{u})$ to validate whether the generative model outputs a feasible or infeasible solution. Append the predicted solution to $\mathcal{D}^{(k)}$ or $\bar{\mathcal{D}}^{(k)}$:

$$\mathcal{D}^{(k+1)} = \mathcal{D}^{(k)} \cup \mathcal{Q}, \text{ where } \mathcal{Q} := \left\{ (F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) \;\middle|\; \Psi(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) = 1, \hat{\mathbf{u}}_i \in \hat{\mathcal{U}} \right\} \tag{6}$$

$$\bar{\mathcal{D}}^{(k+1)} = \bar{\mathcal{D}}^{(k)} \cup \bar{\mathcal{Q}}, \text{ where } \bar{\mathcal{Q}} := \left\{ (F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) \;\middle|\; \Psi(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) = 0, \hat{\mathbf{u}}_i \in \hat{\mathcal{U}} \right\} \tag{7}$$

   Note that for all $\hat{\mathbf{u}}_i \in \hat{\mathcal{U}}$, the generative model will always produce solutions that satisfy $B^{(k)}(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) > 0$. The oracle then checks whether $B^{(k)}(\mathbf{x}, \mathbf{u})$ is correct for each point. Then, the classifier can correct itself in the $k+1$-th iteration.



(a) $B^{(k)}$ learns to classify points in $\mathcal{D}^{(k)}$ and $\bar{\mathcal{D}}^{(k)}$.

(b) $F^{(j,k)}$ is trained to predict a sequence of solutions given λ_j.

(c) After binning, we obtain a tighter barrier in the next iteration.

Figure 2 **One iteration of IPMAN for a single $\hat{\mathbf{u}}_i$. $\Diamond$ and $\square$ are samples of $\mathbb{P}_{(\mathbf{x},\mathbf{u})}$ and $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$, respectively. The filled region is $\mathcal{X}(\mathbf{u})$, the bold line on the outside is $\mathcal{P}$. The inner solid lines show the support of $B^{(k)}(\mathbf{x},\mathbf{u})$.**

Figure 2 shows the steps and the outcome for a single iteration k and a single $\hat{\mathbf{u}}_i$. In the remainder of this section, we describe the algorithm in greater detail and show that it satisfies several desirable properties. Specifically, we first show that the classifier learns to approximate a δ-barrier. We then show that the generative model satisfies a (δ,ϵ)-optimality guarantee on in-sample instances, albeit not as strong as one that would be obtained were we to directly solve $\mathbf{BP}(\hat{\mathbf{u}}_i, B_\delta, \lambda)$. Finally, we show that the data augmentation procedure shrinks the set of optimal solutions to $\mathbf{FCP}(\mathcal{D},\bar{\mathcal{D}})$. Thus, each iteration learns a δ-barrier that more closely approximates $\mathcal{X}(\mathbf{u})$ for any $\mathbf{u}$.

4.2. A data-driven δ-barrier

A perfect barrier function where $\delta = 0$ is a perfect classifier of feasibility. That is, it returns positive values if and only if the input $\mathbf{x}$ is feasible for $\mathbf{OP}(\mathbf{u})$, and zero otherwise. A δ-barrier can perfectly classify feasible points, but potentially incorrectly classify infeasible points. By solving this classification problem, $B(\mathbf{x},\mathbf{u})$ learns to approximate a δ-barrier.

We make an assumption that the classifier is sufficiently parameterized to be able to describe a complex, and potentially non-convex feasible set. A sufficient condition would be that the model class $\mathcal{B}$ satisfies a Universal Approximation property.

ASSUMPTION 1. *$\mathcal{B}$ satisfies a Universal Approximation Theorem (Hornik 1991). That is, for any continuous function $B^*(\mathbf{x},\mathbf{u}):\mathbb{R}^n\times\mathcal{U}\to[0,1]$ and degree of accuracy $\varepsilon>0$, there exists $B\in\mathcal{B}$ such that $|B(\mathbf{x},\mathbf{u})-B^*(\mathbf{x},\mathbf{u})|<\varepsilon$ for all $\mathbf{x}\in\mathbb{R}^n$, $\mathbf{u}\in\mathcal{U}$.*

LEMMA 1 **(Arjovsky and Bottou (2017))**. *Let $\mathcal{B}$ satisfy Assumption 1. If $\mathcal{D}^{(k)}$ and $\bar{\mathcal{D}}^{(k)}$ are closed, then $\mathbf{FCP}(\mathcal{D}^{(k)},\bar{\mathcal{D}}^{(k)})$ is feasible and has an optimal value equal to 0.*

REMARK 2. Arjovsky and Bottou (2017, Theorem 2.1) prove Lemma 1 showing the cross-entropy loss function yields an optimal value of 0 whenever the two classes to be predicted are supported over compact and disjoint sets. We show that closedness, rather than compactness, is sufficient.

Intuitively for $\mathbf{FCP}(\mathcal{D}^{(k)},\bar{\mathcal{D}}^{(k)})$, we require that the two data sets, $\mathcal{D}^{(k)}$ and $\bar{\mathcal{D}}^{(k)}$, be disjoint and sufficiently far from each other to ensure that a classifier can learn a separation between their supports. Our problem naturally provides the disjoint property as the two data sets arise from solutions that are correctly labeled as feasible and infeasible, respectively. Figure 2(a) demonstrates an example of this intuition. We also observe that the optimal solution set to $\mathbf{FCP}(\mathcal{D}^{(k)},\bar{\mathcal{D}}^{(k)})$ is large. As a result, any function that separates the two sets is optimal.

Given a limited data set, solving $\mathbf{FCP}(\mathcal{D}^{(k)},\bar{\mathcal{D}}^{(k)})$ may not yield a *practically useful* classifier. For example, an optimal classifier may over-fit to the data, be unnecessarily complex, or may misclassify regions where data is not available. In order to ensure that the classifier is a δ-barrier, we assume sufficient data so as to be able to solve the stochastic optimization variant of $\mathbf{FCP}(\mathcal{D},\bar{\mathcal{D}})$:

$$\mathbf{FCP}(\mathbb{P}_{(\mathbf{x},\mathbf{u})},\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}):\quad \sup_{B\in\mathcal{B}}\left\{\mathbb{E}_{\mathbf{x},\mathbf{u}\sim\mathbb{P}_{(\mathbf{x},\mathbf{u})}}\Big[\log B(\mathbf{x},\mathbf{u})\Big]+\mathbb{E}_{\mathbf{x},\mathbf{u}\sim\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}}\Big[\log\big(1-B(\mathbf{x},\mathbf{u})\big)\Big]\right\}. \tag{8}$$

REMARK 3. The key difference between $\mathbf{FCP}(\mathcal{D}^{(k)},\bar{\mathcal{D}}^{(k)})$ and $\mathbf{FCP}(\mathbb{P}_{(\mathbf{x},\mathbf{u})},\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$ is that the latter permits arbitrary probability distributions rather than discrete empirical distributions. An optimal solution to $\mathbf{FCP}(\mathbb{P}_{(\mathbf{x},\mathbf{u})},\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$ satisfies $B^{(k)}(\mathbf{x},\mathbf{u})=1$ for all $(\mathbf{x},\mathbf{u})\in\text{supp}(\mathbb{P}_{(\mathbf{x},\mathbf{u})})$ and $B^{(k)}(\mathbf{x},\mathbf{u})=0$ for all $(\mathbf{x},\mathbf{u})\in\text{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$, so long as they are closed and disjoint. Further, because $\bar{\mathcal{D}}^{(k)}$ is initially obtained by sampling from $\mathbb{R}^n\setminus\mathcal{P}$ (see Appendix **??**), we can always assume access to a distribution $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$ for which $\text{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})\supseteq\{(\mathbf{x},\mathbf{u})\mid\mathbf{x}\in\mathbb{R}^n\setminus\mathcal{P},\mathbf{u}\in\mathcal{U}\}$. Thus, the remaining

difference in studying the stochastic versus the data-driven classification problems is that the data set of feasible solutions $\mathcal{D}^{(k)}$ is sufficiently large.

If the supports of $\mathbb{P}_{(\mathbf{x},\mathbf{u})}$ and $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$ are over the feasible and infeasible sets respectively, the Feasibility Classification Problem yields a δ-barrier.

COROLLARY 1. *Let $B^{(k)}(\mathbf{x},\mathbf{u})$ be the optimal solution to $\mathbf{FCP}(\mathbb{P}_{(\mathbf{x},\mathbf{u})},\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$ which achieves an optimal value of 0. At the optimum, the following statements are true:*

1. *If $\operatorname{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}) \supseteq \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{P}, \mathbf{u} \in \mathcal{U}\}$, then for any $\mathbf{u} \in \mathcal{U}$, $\{\mathbf{x} \mid B^{(k)}(\mathbf{x},\mathbf{u}) > 0\} \subseteq \mathcal{P}$.*
2. *If $\operatorname{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}) \supseteq \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{P}, \mathbf{u} \in \mathcal{U}\}$ and $\operatorname{supp}(\mathbb{P}_{(\mathbf{x},\mathbf{u})}) = \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$, then for any $\zeta \in (0,1)$, the scaled classifier $\zeta B^{(k)}(\mathbf{x},\mathbf{u})$ is a δ-barrier for some $\delta \leq \Delta(\mathbf{u})$.*

Corollary 1 states that given access to two disjoint distributions $\mathbb{P}_{(\mathbf{x},\mathbf{u})}$ and $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$ with closed supports, $B^{(k)}(\mathbf{x},\mathbf{u})$ will learn a boundary between them. The first statement ensures that for any $\mathbf{u}$, the optimal classifier has a closed and bounded support that is smaller than $\mathcal{P}$. The second statement is a sufficient condition for $B^{(k)}(\mathbf{x},\mathbf{u})$ to be a δ-barrier. Furthermore, because we initially sample from $\bar{\mathcal{D}}$, we can therefore assume access to a distribution $\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}$ supported over $\{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{P}, \mathbf{u} \in \mathcal{U}\}$ which implies that the first statement is always satisfied in practice. However, satisfying the second statement is contingent on access to a data set whose support is equal to $\{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$. In Step 2 of the IPMAN algorithm, the generated points are used to augment the two data sets and produce a better classifier.

4.3. In-sample optimality guarantees

We train a generative model $F(\mathbf{u})$ to solve $\mathbf{GBP}(\hat{\mathcal{U}}, B^{(k)}, \lambda_j)$ for a decreasing sequence of $\lambda_j > 0$. As a result, $F(\mathbf{u})$ learns to predict optimal solutions to the barrier problem $\mathbf{x}^{\lambda}(\mathbf{u})$ in an unsupervised fashion, i.e., without using $\mathbf{x}^{\lambda}(\mathbf{u})$ or $\mathbf{x}^{*}(\mathbf{u})$. Moreover, the approximation error of $F(\hat{\mathbf{u}}_i)$ versus $\mathbf{x}^{*}(\hat{\mathbf{u}}_i)$ is also bounded, which we characterize below.

THEOREM 2. *Fix $\hat{\mathbf{u}}_i \in \hat{\mathcal{U}}$ and $\lambda_j > 0$ and consider $B^{(k)}(\mathbf{x},\mathbf{u})$ and $F^{(j,k)}(\mathbf{x},\mathbf{u})$. Let $\mathbf{x}^{\lambda_j}$ be an optimal solution to $\mathbf{BP}(\hat{\mathbf{u}}_i, B^{(k)}, \lambda_j)$ Then, there exists $\delta, \epsilon > 0$ such that*

$$\left|f\left(F^{(j,k)}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^{*}(\hat{\mathbf{u}}_i)\right)\right| < \left|f\left(F^{(j,k)}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)\right)\right| + \max(\delta L, \epsilon)$$

Theorem 2 illustrates the key strengths and challenges with the IPMAN algorithm. Intuitively, the proof considers two cases: one where the classifier $B^{(k)}(\mathbf{x}, \mathbf{u})$ is a δ-barrier for a given $\mathbf{OP}(\mathbf{u})$, and one where it isn't. If the classifier is a δ-barrier, then it is straightforward to bound the quality of the model predictions by directly optimizing $\mathbf{BP}(\mathbf{u}, B^{(k)}, \lambda_j)$ and comparing against the optimal solution. However, the learned classifier may not yet be a δ-barrier if there are insufficient points. In this scenario, we can still bound the (δ, ϵ)-optimality of predicted solutions by artificially constructing a δ-barrier from the classifier. Therefore, the in-sample performance of $F^{(j,k)}(\mathbf{u})$ can always be measured at any iteration. Unlike a δ-barrier, however, the (δ, ϵ)-optimality bound using an artificial δ-barrier does not necessarily converge to 0 as we decrease λ_j.

4.4. Data augmentation via the oracle

At every iteration, the oracle evaluates the generative models by labelling the predictions as feasible or infeasible. If $\Psi(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i) = 1$ for any $\hat{\mathbf{u}}_i$, then $(F^{(j,k)}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i)$ is added to the data set of feasible solutions $\mathcal{D}^{(k)}$. Otherwise, it is added to $\bar{\mathcal{D}}^{(k)}$. Consequently, these two data sets grow after each iteration and the augmented sets are used to train $B^{(k+1)}(\mathbf{x}, \mathbf{u})$. This data augmentation procedure implies that the classifier can learn to become a tighter approximation of $\mathcal{X}(\mathbf{u})$.

PROPOSITION 1. *For any k, let $\mathcal{B}^{(k)}$ be the optimal solution set of $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$. If $\mathcal{B}$ satisfies Assumption 1 and the data sets are closed, then $\mathcal{B}^{(k+1)} \subset \mathcal{B}^{(k)}$.*

After each iteration of the IPMAN algorithm, $\mathcal{D}^{(k)}$ and $\bar{\mathcal{D}}^{(k)}$ are augmented. By augmenting $\bar{\mathcal{D}}^{(k)}$, we correct regions that the classifier has incorrectly labelled as feasible. On the other hand, augmenting $\mathcal{D}^{(k)}$ reinforces regions where the classifier has correctly labelled points so that it does not incorrectly mislabel the region in a subsequent iteration.

Because $\mathcal{X}(\mathbf{u})$ is not known, it is difficult to determine whether the classifier is, in fact, a δ-barrier for a given $\mathbf{u}$. From the proof of Theorem 2, we know that even if the classifier is not a δ-barrier, there exists an equivalent δ-barrier for the classifier as well as a fixed $\lambda > 0$. It remains, therefore, to estimate the value of δ for the classifier in order to fully characterize the optimality bound.

COROLLARY 2. *If $B(\mathbf{x}, \hat{\mathbf{u}}_i)$ is a δ-barrier for $\mathbf{OP}(\hat{\mathbf{u}}_i)$, then $\delta \leq d_H(\{\hat{\mathbf{x}}_i \mid (\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}\}, \mathrm{bd}(\mathcal{P}))$.*

By generating additional feasible points after each iteration, we can obtain a smaller δ. Further, to compute the Hausdorff distance, we must calculate the distance of a fixed set of points to each of the facets of the polyhedron. This problem can be solved via a finite number of linear programs.

5. Generalization of (δ, ϵ)-optimality to unseen instances

In this section, we evaluate the potential for a generative model to predict a (δ, ϵ)-optimal solution when given an unseen out-of-sample problem instance $\mathbf{u}$. Our analysis applies on any model and is independent of the IPMAN algorithm itself. However, this analysis, when applied to IPMAN, offers an opportunity to understand the true quality of the trained model at any iteration of the algorithm. Although the optimal solution $\mathbf{x}^*(\mathbf{u})$ is not known, the objective function error between $f(F(\mathbf{u}))$ and the optimal value $f(\mathbf{x}^*(\mathbf{u}))$ can be bounded using the Triangle inequality

$$|f\left(F^*(\mathbf{u})\right) - f\left(\mathbf{x}^*(\mathbf{u})\right)| < \left|f\left(F^*(\mathbf{u})\right) - f\left(\mathbf{x}^{\lambda}(\mathbf{u})\right)\right| + \left|f\left(\mathbf{x}^*(\mathbf{u})\right) - f\left(\mathbf{x}^{\lambda}(\mathbf{u})\right)\right|.$$

From Theorem 1, the second term is bounded above by $\max\{\delta L, \epsilon\}$. Therefore, it only remains to bound the empirical error of $F(\mathbf{u})$ versus $\mathbf{x}^{\lambda}(\mathbf{u})$ (i.e., the first term on the right-hand side).

We use Rademacher complexity theory to obtain a probabilistic bound on the empirical error from an out-of-sample input (Bartlett and Mendelson 2002). While Bertsimas and Kallus (2019) develop generalization bounds for predicting decisions to problems with conditional stochastic optimization objectives, we extend their work by providing a probabilistic bound on (δ, ϵ)-optimality when the feasible set is not fully specified.

Definition 2 (Bertsimas and Kallus (2019)). Let $\mathcal{F} \subset \{F(\mathbf{u}) : \mathcal{U} \to \mathbb{R}^n\}$ be a function class and $\hat{\mathcal{U}} \sim \mathbb{P}_{\mathbf{u}}$ be an i.i.d. data set. The empirical multivariate Rademacher complexity of $\mathcal{F}$ is

$$\hat{\mathfrak{R}}_{N_{\mathbf{u}}}(\mathcal{F}, \hat{\mathcal{U}}) = \mathbb{E}_{\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}}\left[\frac{2}{N_{\mathbf{u}}} \sup_{F \in \mathcal{F}} \sum_{i=1}^{N_{\mathbf{u}}} \boldsymbol{\sigma}_i^{\top} F(\hat{\mathbf{u}}_i) \,\middle|\, \hat{\mathcal{U}} = \{\hat{\mathbf{u}}_i\}_{i=1}^{N_{\mathbf{u}}}\right],$$

where $\boldsymbol{\sigma}_i \sim p_{\boldsymbol{\sigma}}$ is an n-dimensional vector of i.i.d. Rademacher variables. The multivariate Rademacher complexity of $\mathcal{F}$ is $\mathfrak{R}_{N_{\mathbf{u}}}(\mathcal{F}) = \mathbb{E}_{\hat{\mathcal{U}} \sim \mathbb{P}_{\mathbf{u}}}\left[\hat{\mathfrak{R}}_{N_{\mathbf{u}}}(\mathcal{F}, \hat{\mathcal{U}})\right]$.

Rademacher complexities are often used to generate risk bounds in various statistical learning settings (Bartlett and Mendelson 2002). In order to ensure these bounds are practical, it is important to use model classes $\mathcal{F}$ whose Rademacher complexities are bounded above as a function of the data and parametrization. For generality, we leave the choice of model class open.

REMARK 4. Although the literature mostly focuses on the single-variate Rademacher complexity, Bertsimas and Kallus (2019) and Maurer (2016) prove bounds for several linear multivariate classes (e.g., $\mathcal{F}_R = \{\mathbf{W}\mathbf{u} \mid \|\mathbf{W}\| \leq R\}$). In general, if $F(\mathbf{u}) = (F_1(\mathbf{u}), \ldots, F_n(\mathbf{u}))$, then $\mathcal{F} \subset \times_{\ell=1}^{n} \mathcal{F}_\ell$, where $\mathcal{F}_\ell = \{F(\mathbf{u})^\mathsf{T}\mathbf{e}_\ell \mid F \in \mathcal{F}\}$ and $\mathbf{e}_\ell$ is the ℓ-th identity vector. Then, $\hat{\mathfrak{R}}_{N_\mathbf{u}}(\mathcal{F},\hat{\mathcal{U}}) \leq \sum_{\ell=1}^{n} \hat{\mathfrak{R}}_{N_\mathbf{u}}(\mathcal{F}_\ell,\hat{\mathcal{U}})$ decomposes to a sum of single-variate complexities. We refer to Bartlett and Mendelson (2002) for Rademacher bounds for linear models and decision trees and Bartlett and Mendelson (2002), Neyshabur et al. (2015), Foster et al. (2018) for neural networks.

$\mathbf{GBP}(\hat{\mathcal{U}}, B_\delta, \lambda)$ is trained using a finite data set $\hat{\mathcal{U}}$, meaning there is no guarantee whether $F(\mathbf{u})$ will be feasible or even satisfy $B_\delta(F(\mathbf{u}), \mathbf{u}) > 0$ for an arbitrary $\mathbf{u}$. Because $\mathcal{P}$ is available, however, we can always project any generated solution to the polyhedron.

ASSUMPTION 2. *If $F^{(j,k)}$ is an optimal solution to $\mathbf{GBP}(\hat{\mathcal{U}}, B_\delta, \lambda)$, then the projected function $F^*(\mathbf{u}) = \arg\min_\mathbf{x}\{\left\|\mathbf{x} - F^{(j,k)}(\mathbf{u})\right\| \mid \mathbf{x} \in \mathcal{P}\}$ is used at test time.*

Our generalization bound follows from the bound of Bertsimas and Kallus (2019). While they derive an empirical risk bound on an unconstrained stochastic optimization problem, we focus on the (δ, ϵ)-optimality of a constrained continuous optimization problem. The proof is provided in the Electronic Companion EC.2.

THEOREM 3. *Let F^* satisfy Assumption 2. Let K and L_∞ be sufficiently large positive constants. Let $\beta \in (0,1)$ be a constant. Then, for any $\gamma > 0$, the following inequality holds*

$$\mathbb{P}_\mathbf{u}\Big\{f\big(F^*(\mathbf{u})\big) - \epsilon - \gamma < f\big(\mathbf{x}^*(\mathbf{u})\big) < f\big(F^*(\mathbf{u})\big) + \delta L + \gamma\Big\} \geq 1 - \frac{\frac{1}{N_\mathbf{u}}\sum_{i=1}^{N_\mathbf{u}} \left|f\big(F^*(\hat{\mathbf{u}}_i)\big) - f\big(\mathbf{x}^\lambda(\hat{\mathbf{u}}_i)\big)\right| + K\sqrt{\frac{\log(1/\beta)}{2N_\mathbf{u}}} + \sqrt{2n}L_\infty \mathfrak{R}_{N_\mathbf{u}}(\mathcal{F})}{\gamma},$$

with probability at least $1-\beta$ with respect to the sampling of $\hat{\mathcal{U}}$.

The quality of this bound is dependent on the generative model itself. The first term in the numerator, $(\sum_{i=1}^{N_{\mathbf{u}}} |f(F^*(\hat{\mathbf{u}}_i)) - f(\mathbf{x}^\lambda(\hat{\mathbf{u}}_i))|)/N_{\mathbf{u}}$, is the empirical error of solving $\mathbf{GBP}(\hat{\mathcal{U}}, B_\delta, \lambda)$ versus $\mathbf{BP}(\hat{\mathbf{u}}_i)$ for all $\hat{\mathbf{u}}_i \in \hat{\mathcal{U}}$. This effectively measures how well the model performs on in-sample data. The second term is dependent on the constants K and $1/\beta$ and scales with $O(1/N_{\mathbf{u}})$. Finally, the third term is dependent on the Rademacher complexity of $\mathcal{F}$. Thus, in order to obtain a tight and useful bound, we must balance the trade-off between a model class with high complexity versus obtaining a final model with low empirical error.

Theorem 3 is a bound on the (δ, ϵ)-optimality of a random out-of-sample $\mathbf{u}$. Note that we require two distinct probability statements to describe this bound. The explicit statement is calculating, given a γ and F^*, the probability that the model will predict a $(\delta + \gamma/L, \epsilon + \gamma)$-optimal solution for a random $\mathbf{u}_{N_{\mathbf{u}}+1} \sim \mathbb{P}_{\mathbf{u}}$. The probability of this event is bounded from below by the right-hand-side of the relation in Theorem 3. However, the implicit statement is that this bound on the probability will only hold with probability at least $1 - \beta$.

6. Conclusion

Conventional optimization techniques generally require well-structured problem formulations and make limited account of auxiliary data present in problems where different instances must be regularly solved. We propose Interior Point Methods with Adversarial Networks, a learning-based approach for generating solutions to optimization problems whose feasible sets are determined by instance-specific auxiliary information. We develop an unconstrained barrier problem where the barrier is replaced by a classifier trained on historical instances to predict feasibility. Because a classifier is not perfectly accurate, we extend the theory of interior point methods to the setting where only a relaxation of the feasible set is known and develop a corresponding optimality guarantee. Our main algorithm iteratively trains the classifier as well as a generative model via empirical risk minimization of the barrier problem. We demonstrate that the classifier learns to better approximate an effective barrier and the generative model learns to predict solutions with an optimality guarantee for both in-sample and out-of-sample instances. Ultimately, we obtain

a deep learning model that can predict optimal solutions to problems in a fraction of the time that it would take a conventional optimization solver. Furthermore, our predictions account for instance-specific variations in the feasible set that conventional optimization would fail to permit.

Our next steps are to apply IPMAN to predict treatment plans in radiation therapy. Current treatment planning systems generally require multiple stages of prediction and optimization to construct a final plan that can satisfy clinical constraints and minimize radiation dose. Using IPMAN, we can predict clinically feasible treatments that also carry an optimality guarantee in one shot, thereby delivering high quality plans in a fraction of the time that is currently possible.

References

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

Arjovsky M, Bottou L (2017) Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862* .

Babier A, Boutilier JJ, McNiven AL, Chan TCY (2018a) Knowledge-based automated planning for oropharyngeal cancer, accepted to Med. Phys.

Babier A, Boutilier JJ, Sharpe MB, McNiven AL, Chan TCY (2018b) Inverse optimization of objective function weights for treatment planning using clinical dose-volume histograms. *Phys Med Biol* 63(10):105004.

Badenbroek R, de Klerk E (2018) Complexity analysis of a sampling-based interior point method for convex optimization. *arXiv preprint arXiv:1811.07677* .

Bain M, Sammut C (1999) A framework for behavioural cloning. *Machine Intelligence 15, Intelligent Agents*, 103–129.

Ban GY, Rudin C (2018) The big data newsvendor: Practical insights from machine learning. *Operations Research* 67(1):90–108.

Bartlett PL, Mendelson S (2002) Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research* 3(Nov):463–482.

Bello I, Pham H, Le QV, Norouzi M, Bengio S (2017) Neural combinatorial optimization.

Bengio Y, Lodi A, Prouvost A (2018) Machine learning for combinatorial optimization: a methodological tour d'horizon. *arXiv preprint arXiv:1811.06128* .

Benson HY, Shanno DF, Vanderbei RJ (2004) Interior-point methods for nonconvex nonlinear programming: jamming and numerical testing. *Mathematical programming* 99(1):35–48.

Bertsimas D, Kallus N (2019) From predictive to prescriptive analytics. *Management Science* 0(0).

Bertsimas D, McCord C (2018) Optimization over continuous and multi-dimensional decisions with observational data. *Advances in Neural Information Processing Systems*, 2966–2974,.

Boyd S, Vandenberghe L (2004) *Convex optimization* (Cambridge University Press).

Bubeck S, Eldan R (2019) The entropic barrier: Exponential families, log-concave geometry, and self-concordance. *Mathematics of Operations Research* 44(1):264–276.

Dai H, Khalil EB, Zhang Y, Dilkina B, Song L (2017) Learning combinatorial optimization algorithms over graphs. *Advances in Neural Information Processing Systems,*, 6348–6358.

Delaney G, Jacob S, Featherstone C, Barton M (2005) The role of radiotherapy in cancer treatment. *Cancer* 104(6):1129–1137.

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

Elmachtoub AN, Grigas P (2017) Smart "predict, then optimize". *arXiv preprint arXiv:1710.08005* .

Engelking R (1977) *General Topology* (Polish Scientific Publishers).

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

Foster DJ, Sekhari A, Sridharan K (2018) Uniform convergence of gradients for non-convex learning and optimization. *Advances in Neural Information Processing Systems*, 8759–8770.

Gondzio J (2012) Interior point methods 25 years later. *European Journal of Operational Research* 218(3):587–601.

Goodfellow I, Bengio Y, Courville A (2016) *Deep learning*, volume 1 (MIT press Cambridge).

Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. *Advances in neural information processing systems*, 2672–2680.

Hannah L, Powell W, Blei DM (2010) Nonparametric density estimation for stochastic optimization with an observable state variable. *Advances in Neural Information Processing Systems*, 820–828.

Hinder O, Ye Y (2018) A one-phase interior point method for nonconvex optimization. *arXiv preprint arXiv:1801.03072* .

Hopfield JJ, Tank DW (1985) "neural" computation of decisions in optimization problems. *Biological cybernetics* 52(3):141–152.

Hornik K (1991) Approximation capabilities of multilayer feedforward networks. *Neural networks* 4(2):251–257.

Kao YH, Roy BV, Yan X (2009) Directed regression. *Advances in Neural Information Processing Systems*, 889–897.

Konda VR, Tsitsiklis JN (2000) Actor-critic algorithms. *Advances in neural information processing systems*, 1008–1014.

Larsen E, Lachapelle S, Bengio Y, Frejinger E, Lacoste-Julien S, Lodi A (2018) Predicting solution summaries to integer linear programs under imperfect information with machine learning. *arXiv preprint arxiv:1807.11876* .

Liu S, He L, Shen ZJ (2018) Data-driven order assignment for last mile delivery .

Mahmood R, Babier A, McNiven A, Diamant A, Chan TCY (2018) Automated treatment planning in radiation therapy using generative adversarial networks. of Machine Learning Research P, ed., *Machine Learning for Health Care*, volume 85.

Maurer A (2016) A vector-contraction inequality for rademacher complexities. *International Conference on Algorithmic Learning Theory*, 3–17 (Springer).

McIntosh C, Purdie TG (2017) Voxel-based dose prediction with multi-patient atlas selection for automated radiotherapy treatment planning. *Phys Med Biol* 62(2):415–431.

McIntosh C, Welch M, McNiven A, Jaffray DA, Purdie TG (2017) Fully automated treatment planning for head and neck radiotherapy using a voxel-based dose prediction and dose mimicking method. *Phys. Med. Biol.* 62(15):5926–5944.

Nemirovskii A, Nesterov Y (1994) Interior-point polynomial methods in convex programming. *Society for Industrial and Applied Mathematics* .

Neyshabur B, Tomioka R, Srebro N (2015) Norm-based capacity control in neural networks. *Conference on Learning Theory*, 1376–1401.

Pelikan M, Goldberg DE, Lobo FG (2002) A survey of optimization by building and using probabilistic models. *Computational optimization and applications* 21(1):5–20.

Shiraishi S, Tan J, Olsen LA, Moore KL (2015) Knowledge-based prediction of plan quality metrics in intracranial stereotactic radiosurgery. *Med. Phys.* 42(2):908.

Vanderbei RJ, Shanno DF (1999) An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications* 13(1-3):231–252.

Vinyals O, Fortunato M, Jaitly N (2015) Pointer networks. *Advances in Neural Information Processing Systems*, 2692–2700.

Electronic Companion

EC.1. Proofs of statements

Proof of Theorem 1. We first prove that $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is bounded and feasible. Note that for any $\lambda > 0$, the second term, $-\lambda \log B_\delta(\mathbf{x}, \mathbf{u})$, is also bounded below as $B_\delta(\mathbf{x}, \mathbf{u}) \in [0, 1)$. Moreover, $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is only feasible within $\{\mathbf{x} \mid B_\delta(\mathbf{x}, \mathbf{u}) > 0\} \subset \mathcal{P}$. By assumption, $\mathcal{P}$ is closed and bounded, meaning $f(\mathbf{x})$, by assumption of linearity, has a bounded minimum within $\mathcal{P}$. Therefore, $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ must have a bounded optimal solution. To show feasibility, note that by the definition of a δ-barrier, any solution that is feasible for $\mathbf{OP}(\mathbf{u})$ is also feasible for $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$.

Suppose we choose $\epsilon = -\lambda \log B_\delta(\mathbf{x}^*(\mathbf{u}), \mathbf{u}))$. By definition, $0 < B_\delta(\mathbf{x}^*(\mathbf{u}), \mathbf{u}) < 1$, implying that $C := -\log B_\delta(\mathbf{x}^*(\mathbf{u}), \mathbf{u}) > 0$ is valid. Let $\mathbf{x}^\lambda$ be an optimal solution to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$. We first prove that $f(\mathbf{x}^\lambda(\mathbf{u})) - \epsilon < f(\mathbf{x}^*(\mathbf{u}))$:

$$\begin{aligned} f\left(\mathbf{x}^*(\mathbf{u})\right) + \epsilon &= f\left(\mathbf{x}^*(\mathbf{u})\right) - \lambda \log B_\delta\left(\mathbf{x}^*(\mathbf{u}), \mathbf{u}\right) \\ &\geq f\left(\mathbf{x}^\lambda(\mathbf{u})\right) - \lambda \log B_\delta\left(\mathbf{x}^\lambda(\mathbf{u}), \mathbf{u}\right) \\ &> f\left(\mathbf{x}^\lambda(\mathbf{u})\right). \end{aligned}$$

The first inequality follows from the optimality of $\mathbf{x}^\lambda(\mathbf{u})$ for $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ while the second inequality follows from $\log B_\delta(\mathbf{x}^\lambda(\mathbf{u}), \mathbf{u}) < 0$, meaning thtat $-\lambda \log B_\delta(\mathbf{x}^\lambda(\mathbf{u}), \mathbf{u}) > 0$ and can be removed. Moving ϵ to the right-hand-side gives the lower bound.

The proof for $f(\mathbf{x}^*(\mathbf{u})) < f(\mathbf{x}^\lambda(\mathbf{u})) + \delta L$ has two cases. If $\mathbf{x}^\lambda(\mathbf{u}) \in \mathcal{X}(\mathbf{u})$, then by the optimality of $\mathbf{x}^*(\mathbf{u})$ for $\mathbf{OP}(\mathbf{u})$, we have $f(\mathbf{x}^*(\mathbf{u})) \leq f(\mathbf{x}^\lambda(\mathbf{u})) < f(\mathbf{x}^\lambda(\mathbf{u})) + \delta L$. Otherwise if $\mathbf{x}^\lambda(\mathbf{u}) \in \mathcal{N}_\delta\left(\mathcal{X}(\mathbf{u})\right) \setminus \mathcal{X}(\mathbf{u})$, then let $\tilde{\mathbf{x}} \in \arg\min_{\mathbf{x} \in \mathcal{X}(\mathbf{u})} \left\|\mathbf{x}^\lambda(\mathbf{u}) - \mathbf{x}\right\|$ be the projection of $\mathbf{x}^\lambda(\mathbf{u})$ on $\mathcal{X}(\mathbf{u})$. Then,

$$\begin{aligned} f\left(\mathbf{x}^*(\mathbf{u})\right) - f\left(\mathbf{x}^\lambda(\mathbf{u})\right) &\leq f(\tilde{\mathbf{x}}) - f\left(\mathbf{x}^\lambda(\mathbf{u})\right) \\ &\leq \left|f(\tilde{\mathbf{x}}) - f\left(\mathbf{x}^\lambda(\mathbf{u})\right)\right| \\ &\leq \left\|\tilde{\mathbf{x}}(\mathbf{u}) - \mathbf{x}^\lambda(\mathbf{u})\right\| L \\ &< \delta L. \end{aligned}$$

The first inequality follows from the optimality of $\mathbf{x}^*(\mathbf{u})$ over $\tilde{\mathbf{x}}$ for $\mathbf{OP}(\mathbf{u})$. The third inequality follows from the Lipschitz continuity of $f(\mathbf{x})$ and the fourth by definition of the δ-barrier. Therefore, the upper bound in inequality (3) is proved for both cases. □

Proof of Lemma 1. Because $B(\mathbf{x},\mathbf{u}) \in [0,1]$, the optimal value must be 0 and is attained only when $B^{(k)}(\mathbf{x},\mathbf{u}) = 1$ for all $(\mathbf{x},\mathbf{u}) \in \mathcal{D}^{(k)}$ and $B^{(k)}(\mathbf{x},\mathbf{u}) = 0$ for all $(\mathbf{x},\mathbf{u}) \in \bar{\mathcal{D}}^{(k)}$. Then, Urysohn's Smooth Lemma states that given two closed and disjoint sets $\mathcal{A}$ and $\mathcal{A}'$, there exists a continuous function for which $f(\mathcal{A}) = 1$ and $f(\mathcal{A}') = 0$ (Engelking 1977). Now note that for any iteration,

$$\mathcal{D}^{(k)} \subset \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$$
$$\bar{\mathcal{D}}^{(k)} \subset \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\},$$

meaning that they are always disjoint and, by assumption, closed. Therefore, there exists a continuous function that achieves the optimal value to $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$. Thus, by Assumption 1, there exists $B(\mathbf{x},\mathbf{u}) \in \mathcal{B}$ that can approximate the supremum. □

Proof of Corollary 1. $\mathbf{FCP}(\mathbb{P}_{(\mathbf{x},\mathbf{u})}, \bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$ achieves an optimal value of 0 if and only if $B^{(k)}(\mathbf{x},\mathbf{u}) = 1$ for all $(\mathbf{x},\mathbf{u}) \in \text{supp}(\mathbb{P}_{(\mathbf{x},\mathbf{u})})$ and $B^{(k)}(\mathbf{x},\mathbf{u}) = 0$ for all $(\mathbf{x},\mathbf{u}) \in \text{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})})$. Thus,

$$\left\{(\mathbf{x},\mathbf{u}) \,\middle|\, \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{P}, \mathbf{u} \in \mathcal{U}\right\} \subseteq \text{supp}(\bar{\mathbb{P}}_{(\mathbf{x},\mathbf{u})}) \subseteq \left\{(\mathbf{x},\mathbf{u}) \,\middle|\, B^{(k)}(\mathbf{x},\mathbf{u}) = 0, \mathbf{u} \in \mathcal{U}\right\}.$$

For any fixed $\mathbf{u}$, rearranging the set inclusions proves the first statement.

To prove the second statement, note that

$$\text{supp}(\mathbb{P}_{(\mathbf{x},\mathbf{u})}) \subseteq \left\{(\mathbf{x},\mathbf{u}) \,\middle|\, B^{(k)}(\mathbf{x},\mathbf{u}) > 0\right\} \subset \{(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathcal{P}, \mathbf{u} \in \mathcal{U}\},$$

where the first inclusion follows by definition and the second follows from Statement 1 of Corollary 1. We fix $\zeta \in (0,1)$ to ensure that $\zeta B^{(k)}(\mathbf{x},\mathbf{u}) \in [0,1)$. Because a canonical barrier for $\mathcal{P}$ is a $\Delta(\mathbf{u})$-barrier (see Remark 1) and $\zeta B^{(k)}(\mathbf{x},\mathbf{u})$ has smaller support, this classifier is a δ-barrier for $\delta \leq \Delta(\mathbf{u})$. □

Proof of Theorem 2. By the Triangle inequality,

$$\left|f\left(F^{(j,k)}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^*(\hat{\mathbf{u}}_i)\right)\right| \leq \left|f\left(F^{(j,k)}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)\right)\right| + \left|f\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^*(\hat{\mathbf{u}}_i)\right)\right|$$

We consider two cases: when $B^{(k)}(\mathbf{x}^*, \hat{\mathbf{u}}_i) > 0$ and when $B^{(k)}(\mathbf{x}^*, \hat{\mathbf{u}}_i) = 0$.

First, if $B^{(k)}(\mathbf{x}^*, \hat{\mathbf{u}}_i) > 0$, then let $\hat{\mathcal{X}}(\hat{\mathbf{u}}_i) = \mathcal{X}(\hat{\mathbf{u}}_i) \cap \{\mathbf{x} \mid B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\}$ be the correctly classified subset of $\mathcal{X}(\hat{\mathbf{u}}_i)$. We need only consider this subset as the feasible set when solving $\mathbf{OP}(\hat{\mathbf{u}}_i)$, since $\mathbf{x}^*(\hat{\mathbf{u}}_i)$ remains feasible. However, $B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i)$ is a δ-barrier for the subset. From Theorem 1, $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)$ is (δ, ϵ)-optimal and we bound $|f(\mathbf{x}^*(\hat{\mathbf{u}}_i)) - f(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i))| < \max(\delta L, \epsilon)$. If $B^{(k)}(\mathbf{x}^*, \hat{\mathbf{u}}_i) = 0$, then the classifier is not a δ-barrier for $\mathcal{X}(\hat{\mathbf{u}}_i)$. Instead, we will construct a "test" δ-barrier from $B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i)$ and show that $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)$ is still an optimal solution for this artificial barrier and thus, is (δ, ϵ)-optimal.

Fix a constant parameter $\varepsilon > 0$. Let $\mathbf{x}^{\mathcal{P}} \in \arg\min_{\mathbf{x}} \{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{P}\}$ and let $\bar{B}$ be defined as follows:

$$\frac{1}{\bar{B}} = \max\left\{1 + \varepsilon, \exp\left[\left(f\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^{\mathcal{P}}\right)\right)\frac{1}{\lambda_j} - \log B^{(k)}\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i\right)\right]\right\}. \tag{EC.1}$$

Note that $\bar{B} \in (0, 1)$. Then, the following function $B^{\text{Test}}(\mathbf{x})$ is a δ-barrier for $\mathbf{OP}(\hat{\mathbf{u}}_i)$:

$$B^{\text{Test}}(\mathbf{x}) = \begin{cases} B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i), & \forall \mathbf{x} \in \{\mathbf{x} \mid B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\} \\ \bar{B}, & \forall \mathbf{x} \in \mathcal{X}(\hat{\mathbf{u}}_i) \cap \{\mathbf{x} \mid B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i) = 0\}. \end{cases}$$

We show that $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)$ is an optimal solution to $\mathbf{BP}(\hat{\mathbf{u}}_i, B^{\text{Test}}, \lambda_j)$. By definition, $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)$ is optimal in $\{\mathbf{x} \mid B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\}$. To show that it is also optimal in $\mathcal{X}(\hat{\mathbf{u}}_i) \cap \{\mathbf{x} \mid B^{(k)}(\mathbf{x}, \hat{\mathbf{u}}_i) = 0\}$, we observe

$$-\log \bar{B} \geq \left(f\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)\right) - f\left(\mathbf{x}^{\mathcal{P}}\right)\right)\frac{1}{\lambda_j} - \log B^{(k)}\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i\right).$$

The above inequality is obtained by transforming the maximum in (EC.1) to an inequality and taking the logarithm on both sides. Re-arranging this inequality yields

$$f\left(\mathbf{x}^{\lambda}(\hat{\mathbf{u}}_i)\right) - \lambda_j \log B^{(k)}\left(\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i), \hat{\mathbf{u}}_i\right) \leq f\left(\mathbf{x}^{\mathcal{P}}\right) - \lambda_j \log \bar{B} \tag{EC.2}$$

$$\leq f\left(\mathbf{x}\right) - \lambda_j \log \bar{B}, \qquad \forall \mathbf{x} \in \mathcal{X}(\hat{\mathbf{u}}_i). \tag{EC.3}$$

We obtain (EC.3) because $\mathcal{X}(\hat{\mathbf{u}}_i) \subset \mathcal{P}$ and consequently, $f(\mathbf{x}^{\mathcal{P}}) \leq f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}(\hat{\mathbf{u}}_i)$. Therefore, $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i) \in \arg\min_{\mathbf{x}} \{f(\mathbf{x}) - \lambda_j \log B^{\text{Test}}(\mathbf{x})\}$. From Theorem 1, $\mathbf{x}^{\lambda_j}(\hat{\mathbf{u}}_i)$ is (δ, ϵ)-optimal. □

Proof of Proposition 1. From Lemma 1, the optimal value of $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$ is 0, since the data sets are closed and disjoint. The augmentations $\mathcal{Q}$ and $\bar{\mathcal{Q}}$ are also closed and disjoint:

$$\mathcal{Q} \subseteq \{(\mathbf{x}, \mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$$
$$\bar{\mathcal{Q}} \subset \{(\mathbf{x}, \mathbf{u}) \mid \mathbf{x} \in \mathbb{R}^n \setminus \mathcal{X}(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}.$$

Therefore, $\mathcal{D}^{(k+1)} = \mathcal{D}^{(k)} \cup \mathcal{Q}$ and $\bar{\mathcal{D}}^{(k+1)} = \bar{\mathcal{D}}^{(k)} \cup \bar{\mathcal{Q}}$ are both closed and disjoint. From Lemma 1, the optimal value of $\mathbf{FCP}(\mathcal{D}^{(k+1)}, \bar{\mathcal{D}}^{(k+1)})$ is also 0.

To show $\mathcal{B}^{(k+1)} \subset \mathcal{B}^{(k)}$, we first prove $\mathcal{B}^{(k+1)} \subseteq \mathcal{B}^{(k)}$ and then present a counter-example with disproves equivalence. The objective function of $\mathbf{FCP}(\mathcal{D}^{(k+1)}, \bar{\mathcal{D}}^{(k+1)})$ is

$$\begin{aligned}
&\frac{1}{|\mathcal{D}^{(k+1)}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \mathcal{D}^{(k+1)}} \log B(\hat{\mathbf{x}}, \hat{\mathbf{u}}) + \frac{1}{|\bar{\mathcal{D}}^{(k+1)}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \bar{\mathcal{D}}^{(k+1)}} \log\big(1 - B(\hat{\mathbf{x}}, \hat{\mathbf{u}})\big) \\
&\qquad = \frac{\alpha}{|\mathcal{D}^{(k)}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \mathcal{D}^{(k)}} \log B(\hat{\mathbf{x}}, \hat{\mathbf{u}}) + \frac{1-\alpha}{|\mathcal{Q}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \mathcal{Q}} \log B(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \\
&\qquad\quad + \frac{\alpha'}{|\bar{\mathcal{D}}^{(k)}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \bar{\mathcal{D}}^{(k)}} \log\big(1 - B(\hat{\mathbf{x}}, \hat{\mathbf{u}})\big) + \frac{1-\alpha'}{|\bar{\mathcal{Q}}|} \sum_{(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \bar{\mathcal{Q}}} \log\big(1 - B(\hat{\mathbf{x}}, \hat{\mathbf{u}})\big),
\end{aligned}$$

where $\alpha = |\mathcal{D}^{(k)}|/|\mathcal{D}^{(k+1)}|$ and $\alpha' = |\bar{\mathcal{D}}^{(k)}|/|\bar{\mathcal{D}}^{(k+1)}|$ are the mixture weights defining the ratio of existing to new points in each data set. Because the optimal value of $\mathbf{FCP}(\mathcal{D}^{(k+1)}, \bar{\mathcal{D}}^{(k+1)})$ is 0 and $B(\mathbf{x}, \mathbf{u}) \in [0, 1]$, each of the individual terms must be equal to 0 for an optimal solution. However, the first and third terms are the objective function for $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$. Thus, any optimal solution $B^{(k+1)}$ to $\mathbf{FCP}(\mathcal{D}^{(k+1)}, \bar{\mathcal{D}}^{(k+1)})$ must also be optimal for $\mathbf{FCP}(\mathcal{D}^{(k)}, \bar{\mathcal{D}}^{(k)})$ implying $\mathcal{B}^{(k+1)} \subseteq \mathcal{B}^{(k)}$.

To prove the inclusion is strict, consider the closed and disjoint sets $\mathcal{D}^{(k)} \cup \bar{\mathcal{Q}}$ and $\bar{\mathcal{D}}^{(k)}$. By Lemma 1, there exists a function $B^*(\mathbf{x}, \mathbf{u})$ such that $B^*(\mathbf{x}, \mathbf{u}) = 1$ for all $(\mathbf{x}, \mathbf{u}) \in \mathcal{D}^{(k)} \cup \bar{\mathcal{Q}}$ and $B^*(\mathbf{x}, \mathbf{u}) = 0$ for all $(\mathbf{x}, \mathbf{u}) \in \bar{\mathcal{D}}^{(k)}$, i.e., $B^* \in \mathcal{B}^{(k)}$. However, then $B^*(\hat{\mathbf{x}}, \hat{\mathbf{u}}) = 1$ for all $(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \bar{\mathcal{Q}}$ and $B^*(\mathbf{x}, \mathbf{u})$ has an infinite objective function value for $\mathbf{FCP}(\mathcal{D}^{(k+1)}, \bar{\mathcal{D}}^{(k+1)})$. Thus, $B^* \notin \mathcal{B}^{(k+1)}$. □

Proof of Corollary 2. By definition, $\delta = d_H(\mathcal{X}(\hat{\mathbf{u}}_i), \{\mathbf{x} \mid B(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\})$, where $\mathcal{X}(\hat{\mathbf{u}}_i) \subseteq \{\mathbf{x} \mid B(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\}$. Then,

$$\begin{aligned}
d_H\big(\mathcal{X}(\hat{\mathbf{u}}_i), \{\mathbf{x} \mid B(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\}\big) &\leq d_H\big(\{\hat{\mathbf{x}}_i \mid (\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}\}, \{\mathbf{x} \mid B(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\}\big) \\
&\leq d_H\big(\{\hat{\mathbf{x}}_i \mid (\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}\}, \mathrm{bd}(\mathcal{P})\big).
\end{aligned}$$

The first inequality follows from $\{\hat{\mathbf{x}}_i \mid (\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}\} \subset \mathcal{X}(\hat{\mathbf{u}}_i)$, while the second follows from $\{\mathbf{x} \mid B(\mathbf{x}, \hat{\mathbf{u}}_i) > 0\} \subseteq \mathcal{P}$ and that the furthest point in $\mathcal{P}$ from $\{\hat{\mathbf{x}}_i \mid (\hat{\mathbf{x}}_i, \hat{\mathbf{u}}_i) \in \mathcal{D}^{(k)}\}$ is on the boundary. □

EC.2. Proof of the generalization bound (Theorem 3)

The proof of the generalization bound uses a Generalization Lemma of Bertsimas and Kallus (2019) to bound the error in objective function value of $F^*(\mathbf{u})$ versus $\mathbf{x}^\lambda(\mathbf{u})$ and Markov's inequality to translate this bound to a probabilistic (δ, ϵ)-optimality certificate. However, in order to use the lemma in this way, we first require an auxiliary result to relate F^* with $\mathfrak{R}_{N_\mathbf{u}}(\mathcal{F})$.

Assumption 2 states that the generative model F^* is a composition; we project the optimal solution $F^{(k)}$ of $\mathbf{GBP}(\hat{\mathcal{U}}, B_\delta, \lambda)$ to $\mathcal{P}$ whenever $F^{(k)}(\mathbf{u}) \notin \mathcal{P}$. Although $F^{(k)} \in \mathcal{F}$, the final model $F^*(\mathbf{u}) := \text{proj}(F(\mathbf{u})) = \arg\min_\mathbf{x}\{\|\mathbf{x} - F(\mathbf{u})\| \mid \mathbf{x} \in \mathcal{P}\}$ is not a member of $\mathcal{F}$. We first bound the Rademacher complexity of models composed from projection below.

LEMMA EC.1. *Let* $\mathcal{F} = \{F : \mathcal{U} \to \mathbb{R}^n\}$ *be a model class and* $\text{proj}(\mathcal{F}) = \{\text{proj}(F) \mid F \in \mathcal{F}\}$ *be the class of models composed by a projection to a polyhedron* $\mathcal{P}$*. Then for any* $\hat{\mathcal{U}} \sim \mathbb{P}_\mathbf{u}$*,* $\hat{\mathfrak{R}}_{N_\mathbf{u}}(\text{proj}(\mathcal{F}), \hat{\mathcal{U}}) \leq \sqrt{2n}\hat{\mathfrak{R}}_{N_\mathbf{u}}(\mathcal{F}, \hat{\mathcal{U}})$*.*

Proof of Lemma EC.1. We want to show for any fixed $\hat{\mathcal{U}}$ that

$$\mathbb{E}_{\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}}\left[\frac{2}{N_\mathbf{u}} \sup_{F \in \mathcal{F}} \sum_{i=1}^{N_\mathbf{u}} \boldsymbol{\sigma}_i^\intercal \text{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right] \leq \sqrt{2n}\mathbb{E}_{\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}}\left[\frac{2}{N_\mathbf{u}} \sup_{F \in \mathcal{F}} \sum_{i=1}^{N_\mathbf{u}} \boldsymbol{\sigma}_i^\intercal F(\hat{\mathbf{u}}_i)\right]. \tag{EC.4}$$

By conditioning and iterating, it suffices to prove the following inequality for any function $\Xi(F) : \mathcal{F} \to \mathbb{R}$,

$$\mathbb{E}_{\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}}\left[\sup_{F \in \mathcal{F}} \boldsymbol{\sigma}^\intercal \text{proj}(F) + \Xi(F)\right] \leq \mathbb{E}_{\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}}\left[\sup_{F \in \mathcal{F}} \sqrt{2n}\boldsymbol{\sigma}^\intercal F + \Xi(F)\right]. \tag{EC.5}$$

We first prove inequality (EC.5), before returning to the main lemma.

As $\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}$ is a random vector of i.i.d. Rademacher variables, it is supported over the (ordered) set $\{(-1, \ldots, -1, -1), (-1, \ldots, -1, 1), \ldots, (1, \ldots, 1, 1)\}$ all with equal probability. Let $\hat{\boldsymbol{\sigma}}_\ell$ denote the

ℓ-th element of this set. By iterating over all values, we expand the left-hand-side of (EC.5) out to:

$$\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}} \boldsymbol{\sigma}^\intercal \text{proj}(F) + \Xi(F)\right] = \frac{1}{2^n}\sum_{\ell=1}^{2^n}\left(\sup_{F\in\mathcal{F}} \hat{\boldsymbol{\sigma}}_\ell^\intercal \text{proj}(F) + \Xi(F)\right) \tag{EC.6}$$

$$= \frac{1}{2^n}\sum_{\ell=1}^{2^{n-1}}\left(\sup_{F\in\mathcal{F}}\left\{\hat{\boldsymbol{\sigma}}_\ell^\intercal \text{proj}(F) + \Xi(F)\right\} + \sup_{F\in\mathcal{F}}\left\{-\hat{\boldsymbol{\sigma}}_\ell^\intercal \text{proj}(F) + \Xi(F)\right\}\right) \tag{EC.7}$$

$$= \frac{1}{2^n}\sum_{\ell=1}^{2^{n-1}}\left(\sup_{F_1,F_2\in\mathcal{F}} \hat{\boldsymbol{\sigma}}_\ell^\intercal \left(\text{proj}(F_1) - \text{proj}(F_2)\right) + \Xi(F_1) + \Xi(F_2)\right). \tag{EC.8}$$

Equation (EC.6) follows by letting $\hat{\boldsymbol{\sigma}}_\ell$ iterate over the support of the distribution. Equation (EC.7) follows from the symmetry of the Rademacher distribution. That is, for every $\hat{\boldsymbol{\sigma}}_\ell$, there exists $-\hat{\boldsymbol{\sigma}}_\ell$ with equal probability, and we need to only characterize half of the elements in the support. (EC.8) merges the suprema.

By the Obtuse Angle Criterion, projection to a convex set is a non-expansive operation (i.e., $\|\text{proj}(F_1) - \text{proj}(F_2)\| \leq \|F_1 - F_2\|$). We use the Cauchy-Schwarz inequality and the non-expansiveness property (in (EC.9) and (EC.10) below, respectively) to remove the dependency on the projection operator:

$$\text{RHS (EC.8)} \leq \frac{1}{2^n}\sum_{\ell=1}^{2^{n-1}}\left(\sup_{F_1,F_2\in\mathcal{F}} \|\hat{\boldsymbol{\sigma}}_\ell\| \, \|\text{proj}(F_1) - \text{proj}(F_2)\| + \Xi(F_1) + \Xi(F_2)\right) \tag{EC.9}$$

$$\leq \frac{1}{2^n}\sum_{\ell=1}^{2^{n-1}}\left(\sup_{F_1,F_2\in\mathcal{F}} \|\hat{\boldsymbol{\sigma}}_\ell\| \, \|F_1 - F_2\| + \Xi(F_1) + \Xi(F_2)\right) \tag{EC.10}$$

$$\leq \frac{1}{2^n}\sum_{\ell=1}^{2^{n-1}}\left(\sup_{F_1,F_2\in\mathcal{F}} \sqrt{n}\,\|F_1 - F_2\| + \Xi(F_1) + \Xi(F_2)\right) \tag{EC.11}$$

$$\leq \frac{1}{2}\left(\sup_{F_1,F_2\in\mathcal{F}} \sqrt{n}\,\|F_1 - F_2\| + \Xi(F_1) + \Xi(F_2)\right) \tag{EC.12}$$

$$\leq \frac{1}{2}\left(\sqrt{n}\,\|F_1^* - F_2^*\| + \Xi(F_1^*) + \Xi(F_2^*)\right). \tag{EC.13}$$

Inequality (EC.11) follows by noting $\|\boldsymbol{\sigma}\| \leq \sqrt{n}$ for all $\boldsymbol{\sigma} \sim p_{\boldsymbol{\sigma}}$ and (EC.12) from the fact that the dependency on $\hat{\boldsymbol{\sigma}}_\ell$ has been removed. We obtain (EC.13) by letting F_1^* and F_2^* be the two values that attain the supremum.

Finally, we use the Khintchine inequality to bound $\|F_1^* - F_2^*\| \leq \sqrt{2}\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\left|\boldsymbol{\sigma}^\intercal(F_1^* - F_2^*)\right|\right]$. We then rearrange the terms as follows:

$$\begin{aligned}
\text{RHS (EC.13)} &\leq \frac{1}{2}\left(\sqrt{2n}\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\left|\boldsymbol{\sigma}^\intercal(F_1^* - F_2^*)\right|\right] + \Xi(F_1^*) + \Xi(F_2^*)\right) && \text{(EC.14)}\\
&= \frac{1}{2}\left(\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sqrt{2n}\left|\boldsymbol{\sigma}^\intercal(F_1^* - F_2^*)\right| + \Xi(F_1^*) + \Xi(F_2^*)\right]\right) && \text{(EC.15)}\\
&\leq \frac{1}{2}\left(\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F_1,F_2\in\mathcal{F}} \sqrt{2n}\left|\boldsymbol{\sigma}^\intercal(F_1 - F_2)\right| + \Xi(F_1) + \Xi(F_2)\right]\right) && \text{(EC.16)}\\
&= \frac{1}{2}\left(\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\left\{\sqrt{2n}\boldsymbol{\sigma}^\intercal F + \Xi(F)\right\} + \sup_{F\in\mathcal{F}}\left\{-\sqrt{2n}\boldsymbol{\sigma}^\intercal F + \Xi(F)\right\}\right]\right) && \text{(EC.17)}\\
&= \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}} \sqrt{2n}\boldsymbol{\sigma}^\intercal F + \Xi(F)\right]. && \text{(EC.18)}
\end{aligned}$$

Inequality (EC.15) brings all of the terms inside the expectation. (EC.16) upper bounds by the supremum. Because $\Xi(F_1)+\Xi(F_2)$ is invariant under the exchange of F_1 and F_2, the supremum will be obtained when $\boldsymbol{\sigma}^\intercal(F_1 - F_2)$ is positive, meaning we can remove the absolute value and separate the supremum in (EC.17). Finally, the symmetry of the random variable $\boldsymbol{\sigma}$ implies that the two suprema are equal, thereby giving (EC.18).

To complete the proof, we use a standard argument from Maurer (2016) to show how (EC.4) decomposes to (EC.5). For any $0 \leq m \leq N_{\mathbf{u}}$, we prove the following inequality by induction:

$$\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sum_{i=1}^{N_{\mathbf{u}}}\boldsymbol{\sigma}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right] \leq \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sum_{i=1}^{m}\sqrt{2n}\boldsymbol{\sigma}_i^\intercal F(\hat{\mathbf{u}}_i) + \sum_{i=m+1}^{N_{\mathbf{u}}}\boldsymbol{\sigma}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right].$$

The case for $m = 0$ is an identity. Now for fixed values of $\hat{\boldsymbol{\sigma}}_i, \forall i \neq m$, let

$$\Xi(F) = \sum_{i=1}^{m-1}\sqrt{2n}\hat{\boldsymbol{\sigma}}_i^\intercal F(\hat{\mathbf{u}}_i) + \sum_{i=m+1}^{N_{\mathbf{u}}}\hat{\boldsymbol{\sigma}}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big).$$

Then, assuming the inequality holds for $m-1$, we show

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sum_{i=1}^{N_{\mathbf{u}}}\boldsymbol{\sigma}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right] &\leq \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sum_{i=1}^{m-1}\sqrt{2n}\boldsymbol{\sigma}_i^\intercal F(\hat{\mathbf{u}}_i) + \sum_{i=m}^{N_{\mathbf{u}}}\boldsymbol{\sigma}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right]\\
&= \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\mathbb{E}_{\boldsymbol{\sigma}_m\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\boldsymbol{\sigma}_m^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_m)\big) + \Xi(F)\,\middle|\,\{\hat{\boldsymbol{\sigma}}_i, \forall i \neq m\}\right]\right]\\
&\leq \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\mathbb{E}_{\boldsymbol{\sigma}_m\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sqrt{2n}\boldsymbol{\sigma}_m^\intercal F(\hat{\mathbf{u}}_m) + \Xi(F)\,\middle|\,\{\hat{\boldsymbol{\sigma}}_i, \forall i \neq m\}\right]\right]\\
&= \mathbb{E}_{\boldsymbol{\sigma}\sim p_{\boldsymbol{\sigma}}}\left[\sup_{F\in\mathcal{F}}\sum_{i=1}^{m}\sqrt{2n}\boldsymbol{\sigma}_i^\intercal F(\hat{\mathbf{u}}_i) + \sum_{i=m+1}^{N_{\mathbf{u}}}\boldsymbol{\sigma}_i^\intercal \operatorname{proj}\big(F(\hat{\mathbf{u}}_i)\big)\right].
\end{aligned}$$

The second inequality comes from substituting (EC.5). When $m = N_{\mathbf{u}}$, the proof is complete. □

Lemma EC.1 can be seen as an extension of the main theorem of Maurer (2016) and is proved using a similar sequence of steps. There, the authors showed that composition of a Lipschitz scalar-valued vector function onto a vector-valued model class bounds the Rademacher complexity of the composed class by $\sqrt{2L}$. In the above, we compose the projection operator, a vector-valued function, to the vector-valued model class and bound the Rademacher complexity by $\sqrt{2n}$. Although we only specifically consider the projection operator, the proof easily extends to any vector-valued function, so long as it is L-Lipschitz, whereupon we reintroduce L back into the bound.

Before proving Theorem 3, we re-state the Generalization Lemma of Bertsimas and Kallus (2019).

LEMMA EC.2 **(Bertsimas and Kallus (2019))**. *Consider a function $z(\mathbf{x},\mathbf{u}):\mathcal{P}\times\mathcal{U}\to\mathbb{R}$ that is bounded and L_∞-Lipschitz continuous in $\mathbf{x}$ using the $\|\cdot\|_\infty$ norm,*

$$\sup_{\mathbf{x}\in\mathcal{P},\mathbf{u}\in\mathcal{U}} z(\mathbf{x},\mathbf{u}) \leq K, \qquad \sup_{\mathbf{x}_1\neq\mathbf{x}_2\in\mathcal{P},\mathbf{u}\in\mathcal{U}} \frac{z(\mathbf{x}_1,\mathbf{u})-z(\mathbf{x}_2,\mathbf{u})}{\|\mathbf{x}_1-\mathbf{x}_2\|_\infty} \leq L_\infty.$$

Let $F\in\operatorname{proj}(\mathcal{F})$. Then for any $\beta>0$, with probability at least $1-\beta$ with respect to the sampling of $\hat{\mathcal{U}}$,

$$\mathbb{E}_{\mathbf{u}\sim\mathbb{P}_\mathbf{u}}\Big[z\big(F(\mathbf{u}),\mathbf{u}\big)\Big] \leq \frac{1}{N_\mathbf{u}}\sum_{i=1}^{N_\mathbf{u}} z\big(F(\hat{\mathbf{u}}_i),\hat{\mathbf{u}}_i\big) + K\sqrt{\frac{\log(1/\beta)}{2N_\mathbf{u}}} + L_\infty\mathfrak{R}_{N_\mathbf{u}}\big(\operatorname{proj}(\mathcal{F})\big), \quad \forall F\in\operatorname{proj}(\mathcal{F}).$$

We are now ready to prove Theorem 3.

Proof of Theorem 3. The proof follows by first applying Lemma EC.2, before applying Markov's inequality. We let $z(\mathbf{x},\mathbf{u}) = |f(\mathbf{x}) - f(\mathbf{x}^\lambda(\mathbf{u}))|$, as a function of $\mathbf{x}\in\mathcal{P}$ and $\mathbf{u}\in\mathcal{U}$, and show it is bounded from above

$$\sup_{\mathbf{x}\in\mathcal{P},\mathbf{u}\in\mathcal{U}} z(\mathbf{x},\mathbf{u}) = \sup_{\mathbf{x}\in\mathcal{P},\mathbf{u}\in\mathcal{U}} \big|f(\mathbf{x}) - f(\mathbf{x}^\lambda(\mathbf{u}))\big| \tag{EC.19}$$

$$\leq \max_{\mathbf{x}\in\mathcal{P}} f(\mathbf{x}) - \min_{\mathbf{x}\in\mathcal{P}} f(\mathbf{x}) = K. \tag{EC.20}$$

Because $\mathcal{P}$ is a closed and bounded set and $f(\mathbf{x})$ is linear, (EC.20) is bounded. We define K to be equal to RHS (EC.20).

We next show L_∞-Lipschitz continuity,

$$\sup_{\mathbf{x}_1\neq\mathbf{x}_2\in\mathcal{P},\mathbf{u}\in\mathcal{U}} \frac{z(\mathbf{x}_1,\mathbf{u})-z(\mathbf{x}_2,\mathbf{u})}{\|\mathbf{x}_1-\mathbf{x}_2\|_\infty} = \sup_{\mathbf{x}_1\neq\mathbf{x}_2\in\mathcal{P},\mathbf{u}\in\mathcal{U}} \frac{\big|f(\mathbf{x}_1)-f(\mathbf{x}^\lambda(\mathbf{u}))\big|-\big|f(\mathbf{x}_2)-f(\mathbf{x}^\lambda(\mathbf{u}))\big|}{\|\mathbf{x}_1-\mathbf{x}_2\|_\infty} \tag{EC.21}$$

$$\leq \sup_{\mathbf{x}_1\neq\mathbf{x}_2\in\mathcal{P},\mathbf{u}\in\mathcal{U}} \frac{\big|f(\mathbf{x}_1)-f(\mathbf{x}^\lambda(\mathbf{u}))-f(\mathbf{x}_2)+f(\mathbf{x}^\lambda(\mathbf{u}))\big|}{\|\mathbf{x}_1-\mathbf{x}_2\|_\infty} \tag{EC.22}$$

$$= \sup_{\mathbf{x}_1\neq\mathbf{x}_2\in\mathcal{P}} \frac{|f(\mathbf{x}_1)-f(\mathbf{x}_2)|}{\|\mathbf{x}_1-\mathbf{x}_2\|_\infty} = L_\infty \tag{EC.23}$$

Inequality (EC.22) follows from the Reverse Triangle Inequality. (EC.23) follows from the fact that $f(\mathbf{x})$ is linear and therefore, Lipschitz continuous using the $\|\cdot\|_\infty$ norm. We let L_∞ be the Lipschitz constant of $f(\mathbf{x})$.

Because $z(\mathbf{x},\mathbf{u})$ satisfies the bounded and Lipschitz continuity assumptions, we apply Lemma EC.2 to obtain

$$\mathbb{E}_{\mathbf{u}\sim\mathbb{P}_\mathbf{u}}\Big[z\big(F(\mathbf{u}),\mathbf{u}\big)\Big] \leq \frac{1}{N_\mathbf{u}}\sum_{i=1}^{N_\mathbf{u}} z\big(F(\hat{\mathbf{u}}_i),\hat{\mathbf{u}}_i\big) + K\sqrt{\frac{\log(1/\beta)}{2N_\mathbf{u}}} + L_\infty\mathfrak{R}_{N_\mathbf{u}}\big(\mathrm{proj}(\mathcal{F})\big), \quad \forall F\in\mathrm{proj}(\mathcal{F}).$$

Specifically, this bound holds for $F^*\in\mathrm{proj}(\mathcal{F})$. By Lemma EC.1, we can bound $\mathfrak{R}_{N_\mathbf{u}}(\mathrm{proj}(\mathcal{F}))\leq\sqrt{2n}\mathfrak{R}_{N_\mathbf{u}}(\mathcal{F})$.

The remainder of the proof follows from Markov's inequality. For $\gamma>0$,

$$\begin{aligned}\mathbb{P}_\mathbf{u}\Big\{z\big(F^*(\mathbf{u}),\mathbf{u}\big)>\gamma\Big\} &= \mathbb{P}_\mathbf{u}\Big\{\big|f\big(F^*(\mathbf{u})\big)-f\big(\mathbf{x}^\lambda(\mathbf{u})\big)\big|>\gamma\Big\}\\ &\leq \frac{\mathbb{E}_{\mathbf{u}\sim\mathbb{P}_\mathbf{u}}\Big[\big|f\big(F^*(\hat{\mathbf{u}}_i)\big)-f\big(\mathbf{x}^\lambda(\hat{\mathbf{u}}_i)\big)\big|\Big]}{\gamma}.\end{aligned}$$

From the Law of Total Probability, we obtain

$$\begin{aligned}\mathbb{P}_\mathbf{u}\Big\{\big|f\big(F^*(\mathbf{u})\big)-f\big(\mathbf{x}^\lambda(\mathbf{u})\big)\big|\leq\gamma\Big\} &= 1-\mathbb{P}_\mathbf{u}\Big\{\big|f\big(F^*(\mathbf{u})\big)-f\big(\mathbf{x}^\lambda(\mathbf{u})\big)\big|>\gamma\Big\}\\ &\geq 1-\frac{\mathbb{E}_{\mathbf{u}\sim\mathbb{P}_\mathbf{u}}\Big[\big|f\big(F^*(\hat{\mathbf{u}}_i)\big)-f\big(\mathbf{x}^\lambda(\hat{\mathbf{u}}_i)\big)\big|\Big]}{\gamma},\\ &\geq 1-\frac{\frac{1}{N_\mathbf{u}}\sum_{i=1}^{N_\mathbf{u}}\big|f\big(F^*(\hat{\mathbf{u}}_i)\big)-f\big(\mathbf{x}^\lambda(\hat{\mathbf{u}}_i)\big)\big|+K\sqrt{\frac{\log(1/\beta)}{2N_\mathbf{u}}}+\sqrt{2n}L_\infty\mathfrak{R}_{N_\mathbf{u}}(\mathcal{F})}{\gamma},\end{aligned}$$

with probability $1-\beta$. The second and third line follow from Markov's inequality and substituting the bound from Lemma EC.2, respectively. Given that we have a probabilistic bound for the error

of $F^*(\mathbf{u})$ from $\mathbf{x}^\lambda(\mathbf{u})$, we bound the error to $\mathbf{x}^*(\mathbf{u})$. Recall that $f(\mathbf{x}^\lambda, \mathbf{u})$ is (δ, ϵ)-optimal. There are two cases to consider. First, if $f(\mathbf{x}^\lambda(\mathbf{u})) \leq f(F(\mathbf{u})) \leq f(\mathbf{x}^\lambda(\mathbf{u})) + \gamma$, then by substitution,

$$f\big(F^*(\mathbf{u})\big) - \epsilon - \gamma < f\big(\mathbf{x}^*(\mathbf{u})\big) < f\big(F^*(\mathbf{u})\big) + \delta L.$$

Alternatively, if $f(F^*(\mathbf{u})) \leq f(\mathbf{x}^\lambda(\mathbf{u})) \leq f(F^*(\mathbf{u})) + \gamma$, then by substitution,

$$f\big(F^*(\mathbf{u})\big) - \epsilon < f\big(\mathbf{x}^*(\mathbf{u})\big) < f\big(F^*(\mathbf{u})\big) + \delta L + \gamma.$$

Note that both of these events can be covered by adding and subtracting γ to both the upper and lower bounds respectively. Then,

$$\mathbb{P}_{\mathbf{u}}\Big\{ f\big(F^*(\mathbf{u})\big) - \epsilon - \gamma < f\big(\mathbf{x}^*(\mathbf{u})\big) < f\big(F^*(\mathbf{u})\big) + \delta L + \gamma \Big\} \geq \mathbb{P}_{\mathbf{u}}\Big\{ \big| f\big(F^*(\mathbf{u})\big) - f\big(\mathbf{x}^\lambda(\mathbf{u})\big) \big| \leq \gamma \Big\},$$

completing the proof. □

EC.3. Structural properties of (δ, ϵ)-optimality for the barrier problem

The IPMAN algorithm simultaneously trains a classifier and a generative model to learn feasibility and predictive optimal solutions respectively. Alternatively, if we are already given a δ-barrier $B_\delta(\mathbf{x}, \mathbf{u})$, we may consider directly optimizing $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$. In this section, we show how tuning the λ parameter can yield feasible or infeasible solutions of different qualities.

Under a mild regularity assumption, for a sufficiently large λ, an optimal solution $\mathbf{x}^\lambda(\mathbf{u})$ to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is guaranteed to lie inside $\mathcal{X}(\mathbf{u})$. Once λ is sufficiently small, the optimal solutions then enter $\mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$. We first state this assumption before characterizing the trajectory of the sequence of points obtained via an IPM.

Assumption EC.1 **(Regularity of the δ-barrier)**.

1. *There exist* $\tilde{\mathbf{x}} \in \text{int}(\mathcal{X}(\mathbf{u}))$ *such that* $B_\delta(\tilde{\mathbf{x}}, \mathbf{u}) > B_\delta(\mathbf{x}, \mathbf{u})$ *for all* $\mathbf{x} \in \text{cl}(\mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u}))$.
2. *There exist* $\tilde{\mathbf{x}}' \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$ *such that* $f(\tilde{\mathbf{x}}') < f(\mathbf{x}^*(\mathbf{u}))$ *and* $0 < B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) < B_\delta(\mathbf{x}, \mathbf{u})$ *for all* $\mathbf{x} \in \mathcal{X}(\mathbf{u})$.

The first statement implies that there exists a point inside $\mathcal{X}(\mathbf{u})$ for which $B_\delta(\mathbf{x},\mathbf{u})$ is greater than any point outside of $\mathcal{X}(\mathbf{u})$. Similarly, the second statement implies that there exists a point outside of $\mathcal{X}(\mathbf{u})$ for which $B_\delta(\mathbf{x},\mathbf{u})$ is lower than any point inside $\mathcal{X}(\mathbf{u})$. Intuitively, the barrier yields higher values for points inside $\mathcal{X}(\mathbf{u})$ rather than outside. Furthermore, the existence of $\tilde{\mathbf{x}}'$ for which $f(\tilde{\mathbf{x}}) > f(\mathbf{x}^*(\mathbf{u})) > f(\tilde{\mathbf{x}}')$ is a direct consequence of the linear objective. Figure 1(a) shows an example of such points for a feasible set where the δ-barrier is a canonical barrier for $\mathcal{P}$. Given a barrier function satisfying Assumption EC.1, λ controls the feasibility of $\mathbf{x}^\lambda(\mathbf{u})$ for $\mathbf{OP}(\mathbf{u})$.

LEMMA EC.3. *If Assumption EC.1 is satisfied, then there exists $\tilde{\lambda}$ such that for all $\lambda \geq \tilde{\lambda}$, the optimal solution to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ is feasible for $\mathbf{OP}(\mathbf{u})$, i.e., $\mathbf{x}^\lambda(\mathbf{u}) \in \mathcal{X}(\mathbf{u})$.*

Proof of Lemma EC.3. Let $\mathbf{x}^+ \in \arg\sup_{\mathbf{x}} \{B_\delta(\mathbf{x},\mathbf{u}) \mid \mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})\}$ and $\mathbf{x}^- \in \arg\inf_{\mathbf{x}} \{f(\mathbf{x}) \mid B_\delta(\mathbf{x},\mathbf{u}) > 0\}$. Then, for $\tilde{\mathbf{x}}$ satisfying Assumption EC.1 Statement 1, we set

$$\tilde{\lambda} = \frac{f(\tilde{\mathbf{x}}) - f(\mathbf{x}^-)}{\log B_\delta(\tilde{\mathbf{x}},\mathbf{u}) - \log B_\delta(\mathbf{x}^+,\mathbf{u})}. \tag{EC.24}$$

From the optimality of $\mathbf{x}^-$, we have $f(\tilde{\mathbf{x}}) > f(\mathbf{x}^-)$. Then, Assumption EC.1 implies that the denominator is positive, and therefore $\tilde{\lambda} > 0$. Rearranging (EC.24) yields

$$f(\tilde{\mathbf{x}}) - \tilde{\lambda} \log B_\delta(\tilde{\mathbf{x}},\mathbf{u}) = f(\mathbf{x}^-) - \tilde{\lambda} \log B_\delta(\mathbf{x}^+,\mathbf{u}).$$

By optimality of $\mathbf{x}^+$ and $\mathbf{x}^-$, we have $f(\mathbf{x}) \geq f(\mathbf{x}^-)$ and $\log B_\delta(\mathbf{x},\mathbf{u}) \leq \log B_\delta(\mathbf{x}^+,\mathbf{u})$ respectively, for all $\mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$. Therefore, $f(\tilde{\mathbf{x}}) - \tilde{\lambda} \log B_\delta(\tilde{\mathbf{x}},\mathbf{u}) \leq f(\mathbf{x}) - \tilde{\lambda} \log B_\delta(\mathbf{x},\mathbf{u})$ for all $\mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$, concluding that the optimal solution to $\mathbf{BP}(\mathbf{u}, B_\delta, \tilde{\lambda})$ must satisfy $\mathbf{x}^{\tilde{\lambda}}(\mathbf{u}) \in \mathcal{X}(\mathbf{u})$.

Now for any $\varepsilon > 0$, observe that

$$\begin{aligned} f(\tilde{\mathbf{x}}) - (\tilde{\lambda}+\varepsilon) \log B_\delta(\tilde{\mathbf{x}},\mathbf{u}) &\leq f(\mathbf{x}) - \tilde{\lambda} \log B_\delta(\mathbf{x},\mathbf{u}) - \varepsilon \log B_\delta(\tilde{\mathbf{x}},\mathbf{u}), \qquad \forall \mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u}) \\ &< f(\mathbf{x}) - \tilde{\lambda} \log B_\delta(\mathbf{x},\mathbf{u}) - \varepsilon \log B_\delta(\mathbf{x},\mathbf{u}), \qquad \forall \mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u}). \end{aligned}$$

The first line is obtained by adding $\varepsilon \log B_\delta(\tilde{\mathbf{x}},\mathbf{u})$ to both sides, and the second from $B_\delta(\tilde{\mathbf{x}},\mathbf{u}) > B_\delta(\mathbf{x},\mathbf{u})$ for $\mathbf{x} \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$. Thus, $\mathbf{BP}(\mathbf{u}, B_\delta, \tilde{\lambda}+\varepsilon)$ yields feasible solutions to $\mathbf{OP}(\mathbf{u})$. $\square$

LEMMA EC.4. *If Assumption EC.1 is satisfied, then there exists* $\tilde{\lambda}'$ *such that for all* $\lambda \leq \tilde{\lambda}'$*, the optimal solution to* $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda)$ *is infeasible for* $\mathbf{OP}(\mathbf{u})$*, i.e.,* $\mathbf{x}^\lambda(\mathbf{u}) \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$.

Proof of Lemma EC.4. Let $\mathbf{x}^\dagger \in \arg\max_{\mathbf{x}} \{B_\delta(\mathbf{x}, \mathbf{u}) \mid \mathbf{x} \in \mathcal{X}(\mathbf{u})\}$. Then, for $\tilde{\mathbf{x}}'$ satisfying Assumption EC.1 Statement 2, let

$$\tilde{\lambda}' = \frac{f(\mathbf{x}^*(\mathbf{u})) - f(\tilde{\mathbf{x}}')}{\log B_\delta(\mathbf{x}^\dagger, \mathbf{u}) - \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u})}. \tag{EC.25}$$

Assumption EC.1 Statement 2 ensures $f(\mathbf{x}^*(\mathbf{u})) > f(\tilde{\mathbf{x}}')$ and $\log B_\delta(\mathbf{x}^\dagger, \mathbf{u}) > \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u})$. Therefore, $\tilde{\lambda}' > 0$. Rearranging (EC.25) gives us

$$f(\tilde{\mathbf{x}}') - \tilde{\lambda}' \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) = f(\mathbf{x}^*(\mathbf{u})) - \tilde{\lambda}' \log B_\delta(\mathbf{x}^\dagger, \mathbf{u}).$$

By optimality of $\mathbf{x}^*(\mathbf{u})$ and $\mathbf{x}^\dagger$, we have $f(\mathbf{x}) \geq f(\mathbf{x}^*(\mathbf{u}))$ and $\log B_\delta(\mathbf{x}, \mathbf{u}) \leq \log B_\delta(\mathbf{x}^\dagger, \mathbf{u})$ respectively, for all $\mathbf{x} \in \mathcal{X}(\mathbf{u})$. Therefore $f(\tilde{\mathbf{x}}') - \tilde{\lambda}' \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) \leq f(\mathbf{x}) - \tilde{\lambda}' \log B_\delta(\mathbf{x}, \mathbf{u})$ for all $\mathbf{x} \in \mathcal{X}(\mathbf{u})$, concluding that the optimal solution to $\mathbf{BP}(\mathbf{u}, B_\delta, \tilde{\lambda}')$ must satisfy $\mathbf{x}^{\tilde{\lambda}'}(\mathbf{u}) \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$.

Now for any $\varepsilon > 0$, observe that

$$\begin{aligned} f(\tilde{\mathbf{x}}') - (\tilde{\lambda}' - \varepsilon) \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) &\leq f(\mathbf{x}) - \tilde{\lambda}' \log B_\delta(\mathbf{x}, \mathbf{u}) + \varepsilon \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u}), \qquad \forall \mathbf{x} \in \mathcal{X}(\mathbf{u}) \\ &< f(\mathbf{x}) - \tilde{\lambda}' \log B_\delta(\mathbf{x}, \mathbf{u}) + \varepsilon \log B_\delta(\mathbf{x}, \mathbf{u}), \qquad \forall \mathbf{x} \in \mathcal{X}(\mathbf{u}). \end{aligned}$$

The first line is obtained by subtracting $\varepsilon \log B_\delta(\tilde{\mathbf{x}}', \mathbf{u})$ to both sides, and the second from $B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) < B_\delta(\mathbf{x}, \mathbf{u})$ for all $\mathbf{x} \in \mathcal{X}(\mathbf{u})$. Thus, $\mathbf{BP}(\mathbf{u}, B_\delta, \tilde{\lambda}' - \varepsilon)$ yields infeasible solutions to $\mathbf{OP}(\mathbf{u})$. □

Lemma EC.4 and the second statement of Assumption EC.1 give the condition where the barrier problem produces undesirable results. Otherwise, if $f(\tilde{\mathbf{x}}') > f(\mathbf{x}^*(\mathbf{u}))$ and $B_\delta(\tilde{\mathbf{x}}', \mathbf{u}) \geq B_\delta(\mathbf{x}, \mathbf{u})$ for all $\tilde{\mathbf{x}}' \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$ and $\mathbf{x} \in \mathcal{X}(\mathbf{u})$, $\mathbf{OP}(\mathbf{u})$ could be solved by classical IPMs.

Lemmas EC.3 and EC.4 state that when λ is set sufficiently high (or low), the corresponding optimal solution $\mathbf{x}^\lambda(\mathbf{u})$ is a certifiably feasible (or infeasible) solution to $\mathbf{OP}(\mathbf{u})$. Furthermore, there exists a trajectory, i.e., feasibility (or infeasibility) is guaranteed for all λ sufficiently high (or low). Assuming access to an oracle $\Psi(\mathbf{x}, \mathbf{u})$, we con construct a simple IPM to obtain optimal

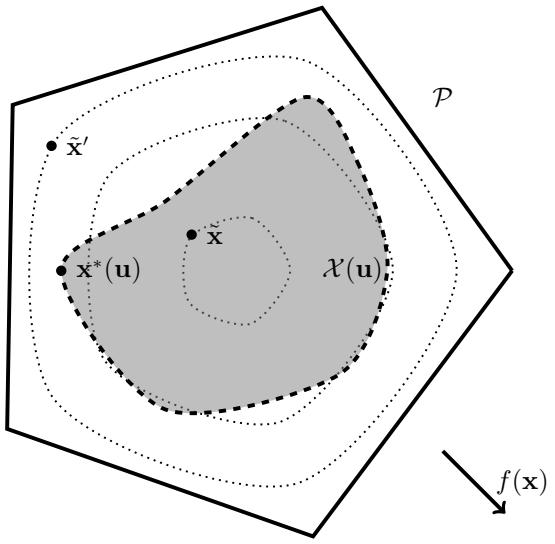


Figure EC.1 **The dotted lines are level sets. $\mathbf{x}^*(\mathbf{u})$ is optimal for $\mathbf{OP}(\mathbf{u})$ while $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}}'$ satisfy Lemmas EC.3 and EC.4 respectively.**

solutions to $\mathbf{OP}(\mathbf{u})$. We initialize with a large λ_0 that satisfies Lemma EC.3. We define a decay rate $\nu < 1$ and a number of iterations $j \in 0, \ldots, M$. Then, for each j, we simply let $\lambda_j = \lambda_0 \nu^j$ and solve $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_j)$ to obtain a new (δ, ϵ)-optimal solution in each iteration. At the end of each iteration, the oracle checks if the solution is still feasible, and terminates when the solution exits the feasible set. We prove the structure of this approach below.

PROPOSITION EC.1. *Suppose that $\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2 \in \mathcal{X}(\mathbf{u})$ and $\tilde{\mathbf{x}}'_1, \tilde{\mathbf{x}}'_2 \in \mathcal{N}_\delta(\mathcal{X}(\mathbf{u})) \setminus \mathcal{X}(\mathbf{u})$ satisfy Statements 1 and 2 of Assumption EC.1, respectively. Assume without loss of generality $B_\delta(\tilde{\mathbf{x}}_1, \mathbf{u}) > B_\delta(\tilde{\mathbf{x}}_2, \mathbf{u})$ and $f(\tilde{\mathbf{x}}'_1) > f(\tilde{\mathbf{x}}'_2)$. Let $\mathbf{x}^{\mathcal{P}} \in \arg\min_{\mathbf{x}} \{f(\mathbf{x}) \,|\, \mathbf{x} \in \mathcal{P}\}$. For $M > 0$ and $j \in \{0, \ldots, M\}$, consider*

$$\lambda_0 = \frac{f(\tilde{\mathbf{x}}_1) - f(\mathbf{x}^{\mathcal{P}})}{\log B_\delta(\tilde{\mathbf{x}}_1, \mathbf{u}) - \log B_\delta(\tilde{\mathbf{x}}_2, \mathbf{u})}, \quad \nu = \left(\frac{f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2)}{-\lambda_0 \log B_\delta(\tilde{\mathbf{x}}'_1, \mathbf{u})} \right)^{1/M}, \quad \lambda_j = \lambda_0 \nu^j$$

Then, the following statements are true:

1. *An optimal solution $\mathbf{x}^{\lambda_0}(\mathbf{u})$ to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_0)$ is a feasible solution for $\mathbf{OP}(\mathbf{u})$.*
2. *There exists $1 \leq j^* \leq M$ such that for all $j < j^*$, an optimal solution $\mathbf{x}^{\lambda_j}(\mathbf{u})$ to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_j)$ is feasible for $\mathbf{OP}(\mathbf{u})$ and for all $j \geq j^*$, $\mathbf{x}^{\lambda_j}(\mathbf{u})$ is infeasible for $\mathbf{OP}(\mathbf{u})$.*

3. *For any $j < j^*$, an optimal solution $\mathbf{x}^{\lambda_j}(\mathbf{u})$ is $(0, \epsilon_j)$-optimal for $\mathbf{OP}(\mathbf{u})$ where*

$$\epsilon_j = (f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2))\nu^{j-M}.$$

Furthermore for any $j \geq j^*$, $\mathbf{x}^{\lambda_j}(\mathbf{u})$ is $(\Delta(\mathbf{u}), \epsilon_j)$-optimal for $\mathbf{OP}(\mathbf{u})$.

Proof of Proposition EC.1. We first make several observations about the parameters. Note that because $\mathcal{X}(\mathbf{u}) \subset \mathcal{P}$ relaxes the feasible set, we have $f(\mathbf{x}^{\mathcal{P}}) \leq f(\mathbf{x}^*(\mathbf{u}))$. Next for all $j \leq M$, $\lambda_j = \lambda_0 \nu^j$ and specifically $\lambda_M = \lambda_0 \nu^M = -(f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2))/\log B_\delta(\tilde{\mathbf{x}}'_1, \mathbf{u})$.

To prove the first statement, we show that $\lambda_0 > \tilde{\lambda}$ where $\tilde{\lambda}$ is defined as in (EC.24) and constructed using $\tilde{\mathbf{x}}_1$. Note that $f(\mathbf{x}^{\mathcal{P}}) \leq f(\mathbf{x}^-)$ and by Assumption EC.1, $\log B_\delta(\tilde{\mathbf{x}}_2, \mathbf{u}) > \log B_\delta(\mathbf{x}^+, \mathbf{u})$. We substitute $f(\mathbf{x}^{\mathcal{P}})$ and $\log B_\delta(\tilde{\mathbf{x}}_2, \mathbf{u})$ in λ_0 and prove $\lambda_0 > \tilde{\lambda}$. By Lemma EC.3, Statement 1 must hold.

We use a similar argument to show $\lambda_M < \tilde{\lambda}'$ as defined in (EC.25) using $\tilde{\mathbf{x}}'_1$. By Lemma EC.4, an optimal solution $\mathbf{x}^{\lambda_M}$ must be infeasible for $\mathbf{OP}(\mathbf{u})$. Given that λ_j decreases every iteration and using the first statement, there must exist a cutoff point $1 \leq j^* \leq M$ for which $\lambda_{j^*} < \tilde{\lambda}'$ and $\lambda_{j^*-1} \geq \tilde{\lambda}'$. Therefore, Statement 2 must also hold.

In order to prove the third statement, recall that we assume $\delta \leq \Delta(\mathbf{u})$ for all j. We first prove $(\Delta(\mathbf{u}), \epsilon_j)$-optimality when $j = M$, and then prove for $j < M$. Let $\epsilon_M = f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2)$. Note that

$$\lambda_M = \frac{f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2)}{-\log B_\delta(\tilde{\mathbf{x}}'_1, \mathbf{u})} = \frac{\epsilon_M}{-\log B_\delta(\tilde{\mathbf{x}}'_1, \mathbf{u})} < \frac{\epsilon_M}{-\log B_\delta(\mathbf{x}^*, \mathbf{u})}.$$

The second equality follows from substituting the value of ϵ_M and the inequality from $B_\delta(\tilde{\mathbf{x}}'_1, \mathbf{u}) < B_\delta(\mathbf{x}^*(\mathbf{u}), \mathbf{u})$ (i.e., Assumption EC.1). We next show that $\mathbf{x}^{\lambda_M}$ satisfies $(\Delta(\mathbf{u}), \epsilon_M)$-optimality,

$$\begin{aligned} f\left(\mathbf{x}^*(\mathbf{u})\right) + \epsilon_M &> f\left(\mathbf{x}^*(\mathbf{u})\right) - \lambda_M \log B_\delta\left(\mathbf{x}^*(\mathbf{u}), \mathbf{u}\right) \\ &\geq f(\mathbf{x}^{\lambda_M}(\mathbf{u})) - \lambda_M \log B_\delta\left(\mathbf{x}^{\lambda_M}(\mathbf{u}), \mathbf{u}\right) \\ &> f\left(\mathbf{x}^{\lambda_M}(\mathbf{u})\right). \end{aligned}$$

The first line follows from substituting the value of ϵ_M and the second from the optimality of $\mathbf{x}^{\lambda_M}(\mathbf{u})$ for $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_M)$. The third line follows from the fact that $-\lambda_M \log B_\delta(\mathbf{x}^{\lambda_M}(\mathbf{u}), \mathbf{u}) > 0$.

For each $j < M$, we have $\lambda_j = \lambda_M \nu^{j-M}$. Then, we write $\epsilon_j = (f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2))\nu^{j-M}$. The same steps used for the $j = M$ case are repeated to obtain $(\Delta(\mathbf{u}), \epsilon_j)$-optimality certificates. Finally, note that from Statement 2, for all $j < j^*$, the optimal solutions $\mathbf{x}^{\lambda_j}(\mathbf{u})$ are feasible for $\mathbf{OP}(\mathbf{u})$. By optimality of $\mathbf{x}^*(\mathbf{u})$ for $\mathbf{OP}(\mathbf{u})$, we have $\delta = 0$ for all $j < j^*$. □

Proposition EC.1 first provides parameters $\lambda_0 > \tilde{\lambda}$ and $\lambda_M < \tilde{\lambda}'$ for which the optimal solutions to $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_0)$ and $\mathbf{BP}(\mathbf{u}, B_\delta, \lambda_M)$ lie inside and outside of $\mathcal{X}(\mathbf{u})$, respectively. Next, it shows that the sequence of λ_j produces a sequence of optimal solutions $\{\mathbf{x}^{\lambda_j}(\mathbf{u})\}$ that start within the feasible set $\mathcal{X}(\mathbf{u})$ and proceed to move outside. Finally, it derives a sequence of corresponding $\{\epsilon_j\}$ such that the sequence of solutions are $(\Delta(\mathbf{u}), \epsilon_j)$-optimal for $\mathbf{OP}(\mathbf{u})$. This implies the final solution is $(0, (f(\tilde{\mathbf{x}}'_1) - f(\tilde{\mathbf{x}}'_2)\nu^{j^*-1-M}))$-optimal for $\mathbf{OP}(\mathbf{u})$.

The above proposition summarizes an IPM for solving $\mathbf{OP}(\mathbf{u})$ when given a δ-barrier and $\Psi(\mathbf{x}, \mathbf{u})$. The IPM behaves in a desirable and predictable fashion; by initializing with large λ, we ensure that we obtain feasible solutions, but by decreasing λ, we know that the solution will ultimately be infeasible. An oracle could identify the point of termination immediately before the IPM leaves the feasible set. We can from here obtain a tight bound on the (δ, ϵ)-optimality of the final solution.

While direct optimization is desirable for its structural properties, this IPM approach is reliant on access to a δ-barrier. On the other hand, IPMAN learns a classifier that approximates a δ-barrier after several iterations. Therefore, unless we are given an a priori δ-barrier (e.g., a canonical barrier for $\mathcal{P}$), this IPM approach is not necessarily feasible from the onset. A potential fix would be to first train IPMAN until a δ-barrier is obtained and then use the δ-barrier IPM to solve subsequent problems. This ties to the second difference between the two approaches; IPMAN is ultimately a predictive model and is therefore subject to prediction error. On the other hand, prediction from a trained model is much faster than direct optimization. Therefore, in cases where the problem is large and an IPM would be difficult to solve or require numerous queries from an oracle, the predictive power of IPMAN yields more practical benefits.