LEARNING DISTRIBUTIONALLY ROBUST FIRST-ORDER METHODS FOR CONVEX OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a distributionally robust approach for learning hyperparameters (*e.g.*, step sizes and momentum coefficients) for first-order methods in convex optimization. Given a dataset of problem instances, we minimize a Wasserstein distributionally robust version of the performance estimation problem (PEP). Our framework unifies two extremes. As the robustness radius vanishes, we recover the empirical risk minimization problem commonly encountered in learning to optimize (L2O), and as it grows we recover the PEP worst-case algorithm design problem. We solve the resulting problem with stochastic gradient descent, differentiating through the solution of an inner semidefinite program at each step. We prove high-probability bounds showing that the true risk of the learned algorithm is at most the in-sample L2O optimum plus a slack that shrinks with the sample size, and is no worse than the worst-case PEP bound. On logistic regression, LASSO, and linear programming examples, our learned algorithms achieve strong out-of-sample performance with certifiable robustness, outperforming both worst-case optimal and vanilla L2O baselines.

1 INTRODUCTION

Applications in machine learning, signal processing, and optimal control increasingly rely on solving large-scale optimization problems using first-order methods for their low computational cost per iteration and modest memory requirements (Beck, 2017; Ryu & Yin, 2022). Building on classical gradient descent (Boyd & Vandenberghe, 2004), modern variants of first-order methods apply proximal operators (Parikh & Boyd, 2014) and operator splitting techniques (Boyd et al., 2011; Chambolle & Pock, 2011) to solve nonsmooth and constrained problems efficiently. This has led to robust open-source solvers such as PDLP (Applegate et al., 2021; 2023) for linear programs (LPs), OSQP (Stellato et al., 2020) for quadratic programs (QPs), and SCS (O'Donoghue et al., 2016; O'Donoghue, 2021) and COSMO (Garstka et al., 2019) for semidefinite programs (SDPs). Yet a fundamental challenge persists: practical performance depends critically on hyperparameters like step sizes, and tuning them well remains largely a manual effort prone to trial and error.

Classical analysis derives worst-case convergence bounds over a given function class. The performance estimation problem (PEP) (Drori & Teboulle, 2014; Taylor et al., 2017) automates this approach by solving semidefinite programs to obtain tight worst-case guarantees. These bounds yield worst-case optimal constant step sizes (Beck, 2017), while recent work develops time-varying step sizes with improved worst-case rates (Altschuler & Parrilo, 2025). However, worst-case analysis is inherently pessimistic: it guards against pathological instances that rarely arise in practice, yielding conservative predictions of performance on typical problems.

Machine learning offers an alternative by exploiting structure in a dataset of problem instances. Learning to optimize (L2O) (Andrychowicz et al., 2016; Li & Malik, 2017; Chen et al., 2022a) learns algorithm components from such data, spanning amortized optimization (Amos, 2023), algorithm selection (Rice, 1976; Kerschke et al., 2019), and algorithm configuration (Hutter et al., 2011; Balcan, 2021), which tunes the hyperparameters of a fixed algorithmic structure. All three minimize an average loss over the training instances, and often achieve substantial practical speedups over hand-tuned alternatives. However, L2O provides no theoretical guarantees on out-of-distribution performance: learned algorithms may fail on problems outside the training distribution (Chen et al., 2022a). Recent efforts enforce guarantees via nonlinear systems theory (Martin & Furieri, 2024;

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

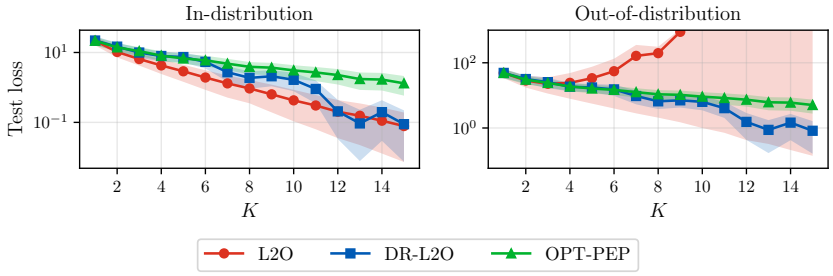


Figure 1: Schematic comparison of L2O (L2O), the worst-case design (OPT-PEP), and (DR-L2O) on a fixed problem class. Solid lines show mean loss across instances; shaded bands show the 10^{th} to 90^{th} quantiles. *Left:* on the training distribution, L2O minimizes empirical loss, the worst-case design is conservative, and DR-L2O lies between (for most K). *Right:* on an out-of-distribution test set, L2O degrades with high variance, the worst-case design remains slow but robust, and DR-L2O dominates both for higher K. Details of this experiment are provided in Section 5.2.

Martin et al., 2025; Martin & Belgioioso, 2026) or statistical bounds, either PAC-Bayes (Dziugaite & Roy, 2017; Sambharya et al., 2024; Sucker et al., 2025) or on held-out validation data (Sambharya & Stellato, 2026). The former reintroduces worst-case conservatism through dynamical systems analysis. PAC-Bayes bounds certify either a Gibbs predictor or a fixed one with a margin penalty, and both kinds of statistical bound still target the training distribution. Appendix A reviews these lines in detail.

In this work, we propose *distributionally robust learning to optimize* (DR-L2O), a framework that combines computer-assisted worst-case analysis with data-driven algorithm configuration. Given a dataset of problem instances, we minimize the worst-case expected loss over algorithm parameters such as step sizes, where the worst case is taken over a Wasserstein ambiguity set centered at the empirical distribution. This builds on the recent distributionally robust PEP of Park et al. (2025), which evaluates this risk for a fixed algorithm. We instead *minimize* it, turning a performance certificate into a learning objective. A single Wasserstein radius controls the trade-off between data-driven and worst-case design: as the radius vanishes we recover L2O; as it grows we recover worst-case optimal design via PEP. We solve the resulting problem with stochastic gradient descent, where each iteration solves an inner SDP and differentiates through its solution via implicit differentiation of the KKT conditions (see Figure 1 for trade-offs). Our contributions are as follows:

1. **DR-L2O framework.** We formulate the distributionally robust learning to optimize problem (DR-L2O) for the scalar parameters of any first-order method satisfying the span condition of Assumption 1, and show that it attains its minimum over the parameter set (Proposition 2). We solve it by stochastic gradient descent, differentiating through the inner SDP (Section 3.4).
2. **Out-of-sample guarantees.** We prove that, for an appropriately chosen radius, the distributionally robust risk upper-bounds the true risk uniformly over algorithm parameters with high probability (Theorem 1). Combined with the interpolation result, the true risk of the learned algorithm is at most the in-sample L2O optimum plus a slack proportional to the radius, and at most the worst-case PEP bound (Theorem 2).
3. **Numerical experiments.** On logistic regression, LASSO, and linear programs, we show that DR-L2O learns algorithms with strong out-of-sample performance and certifiable robustness, outperforming both worst-case optimal and vanilla L2O approaches under distribution shift, while remaining competitive in-distribution.

2 First-order algorithms and data-driven parameter tuning

We consider a *problem instance* $z = (f, x^0)$ consisting of a convex optimization problem to minimize $f(x)$, where $f\colon \mathbf{R}^d \to \mathbf{R} \cup \{\infty\}$ is convex, proper, and lower semi-continuous, and $x^0 \in \mathbf{R}^d$ is an initial point. We assume that the problem has a solution $x^\star$ with optimal value $f^\star = f(x^\star)$.

2.1 First-order algorithms

An *algorithm* $\mathcal{A}_\theta$ maps a problem instance to a sequence of approximate solutions $\mathcal{A}_\theta(z) = \{x^k\}_{k=0,1,\dots}$, where $\theta \in \Theta$ represents the algorithm parameters such as step sizes, also called hyperparameters in the learning-to-optimize literature. We restrict attention to first-order algorithms, which use only gradient information of f.

Assumption 1 (First-order algorithms)**.** *For any $z = (f, x^0)$, the sequence $\{x^k\}_{k=0,1,\dots} = \mathcal{A}_\theta(z)$ satisfies the* span condition

$$x^{k+1} \in x^0 + \mathbf{span}\{g^0, \dots, g^{k-1}, g^k\}, \qquad k = 0, 1, \dots,$$

where $g^i = \nabla f(x^i)$ for $i = 0, \dots, k$. We assume that there exist scalars θ_i^k, called the algorithm parameters, with $x^{k+1} = x^k - \sum_{i=0}^k \theta_i^k g^i$ for $k \geq 0$.

The span condition is the structural requirement of our framework, rather than a design choice, but still encompasses a broad class of first-order methods. We impose the following assumptions on the instance set $\mathcal{Z}$ and the parameter set Θ.

Assumption 2 (Instance and parameter sets)**.**

(i) There exist constants $L > 0$ and $R_0 > 0$ such that, for every instance $z = (f, x^0) \in \mathcal{Z}$ with solution $x^\star$, the function f is L-smooth convex and $\|x^0 - x^\star\| \leq R_0$.

(ii) The parameter set Θ is a box $\Theta = \prod_{j=1}^p [l_j, u_j] \subset \mathbf{R}^p$ with one interval per coordinate of θ and $-\infty < l_j < u_j < \infty$ for every $j = 1, \dots, p$.

We primarily focus on smooth unconstrained minimization in the main text. Our framework also applies to composite problems, which cover a broad range of convex optimization. Section B.3 states the assumptions on the composite setting and the corresponding first-order methods.

2.2 Data-driven parameter tuning

We measure algorithm performance on instance z by a *loss function* $\ell(\mathcal{A}_\theta(z)) \in \mathbf{R}_+$. Given an iteration budget K, common choices of ℓ are the function-value gap $f(x^K) - f^\star$ and the distance to optimality $\|x^K - x^\star\|^2$, evaluated at the K-th iterate x^K.

Let $\mathbf{P}$ be a probability distribution over the set $\mathcal{Z}$ of problem instances. Our goal is to **find the optimal algorithm parameter** $\theta^\star$ minimizing the *risk*, defined as the expected loss under $\mathbf{P}$:

$$\underset{\theta \in \Theta}{\text{minimize}} \quad \mathcal{R}(\theta, \mathbf{P}) := \underset{z \sim \mathbf{P}}{\mathbf{E}} \big(\ell(\mathcal{A}_\theta(z))\big). \tag{1}$$

Since the true distribution $\mathbf{P}$ is unknown, we approximate the risk in (1) from a finite dataset. Given i.i.d. samples $\widehat{\mathcal{D}}_N = \{\hat{z}_i\}_{i=1}^N \sim \mathbf{P}^N$, the empirical distribution is $\widehat{\mathbf{P}}_N = (1/N) \sum_{i=1}^N \delta_{\hat{z}_i}$, where δ_z is the Dirac measure at $z \in \mathcal{Z}$. The corresponding empirical risk is

$$\mathcal{R}(\theta, \widehat{\mathbf{P}}_N) = (1/N) \textstyle\sum_{i=1}^N \ell\big(\mathcal{A}_\theta(\hat{z}_i)\big).$$

The risk minimization problem (1) is the standard L2O formulation, which targets the expected loss under a single fixed but unknown distribution $\mathbf{P}$. This does not guarantee robust performance of the algorithm with the optimal parameter $\theta^\star$ on distributions other than the one that produced $\widehat{\mathcal{D}}_N$.

3 Performance guarantees under distribution shift and DR-L2O

In this section, we highlight the differences between evaluating worst-case guarantees through the performance estimation problem (PEP) (Drori & Teboulle, 2014; Taylor et al., 2017) and its distributionally robust variant (Park et al., 2025). We then define the distributionally robust learning to optimize (DR-L2O) framework and its solution method.

3.1 Worst-case guarantee

The worst-case loss $\sup_{z \in \mathcal{Z}} \ell(\mathcal{A}_\theta(z))$ upper-bounds the risk $\mathcal{R}(\theta, \mathbf{P})$ for any distribution $\mathbf{P}$ supported on $\mathcal{Z}$. It is evaluated as the optimal value of the following performance estimation problem (PEP):

$$\begin{array}{ll} \text{maximize} & \ell(\mathcal{A}_\theta(z)) \\ \text{subject to} & z = (f, x^0) \in \mathcal{Z} \subseteq \mathcal{F} \times \mathcal{X}, \end{array} \tag{PEP}$$

where $\mathcal{F}$ is the function class and $\mathcal{X}$ is the set of initial iterates. When the function class $\mathcal{F}$ admits interpolation conditions, (PEP) admits a tractable formulation as an SDP via a lifted representation (Drori & Teboulle, 2014; Taylor et al., 2017; Ryu et al., 2020). The function classes we consider are convex and L-smooth μ-strongly convex functions (Taylor et al., 2017, Theorems 1 and 4), convex quadratic functions, and linear operators (Bousselmi et al., 2024, Theorems 3.7 and 3.1). Given iterates $\{x^k\}_{k=0}^K = \mathcal{A}_\theta(f, x^0)$, define the Gram matrix

$$G = P^T P \in \mathbf{S}_+^{K+2} \text{ with } P = \begin{bmatrix} x^0 - x^\star & g^0 & \cdots & g^K \end{bmatrix} \in \mathbf{R}^{d\times(K+2)} \text{ where } g^k = \nabla f(x^k),$$

and the function-value vector $F = (f(x^0) - f^\star, \ldots, f(x^K) - f^\star) \in \mathbf{R}^{K+1}$. Each component of (PEP) is linear in (G, F): the loss is $\ell(G, F) = \mathbf{tr}(A_{\text{obj}}^T G) + b_{\text{obj}}^T F$, the initial condition is $\mathbf{tr}(A_0^T G) + b_0^T F + c_0 \le 0$, and the interpolation constraints of $\mathcal{F}$ are $\mathcal{S}_\theta(G, F) \in \mathbf{R}_+^M$ with

$$\big(\mathcal{S}_\theta(G, F)\big)_m = -\,\mathbf{tr}\left(A_m(\theta)^T G\right) - b_m(\theta)^T F, \qquad m = 1, \ldots, M, \tag{2}$$

where the LMI coefficients $A_m(\theta) \in \mathbf{S}^{K+2}$ and $b_m(\theta) \in \mathbf{R}^{K+1}$ depend on the algorithm parameters θ (Kamri et al., 2025). Denote by Ξ^θ the feasible set of (G, F) pairs:

$$\Xi^\theta = \left\{(G, F) \in \mathbf{S}_+^{K+2} \times \mathbf{R}^{K+1} \;\middle|\; \mathcal{S}_\theta(G, F) \in \mathbf{R}_+^M,\ \mathbf{tr}(A_0^T G) + b_0^T F + c_0 \le 0\right\}.$$

This set is closed and convex for each fixed θ. Since every component of (PEP) depends on the algorithm iterates only through their lifted representation (G, F), we identify the output $\mathcal{A}_\theta(z)$ with its lift and write $\mathcal{A}_\theta(z) = (G, F) \in \Xi^\theta$. Any norm applied to $\mathcal{A}_\theta(z)$ is then the pair norm $\|(G, F)\| = \sqrt{\|G\|_F^2 + \|F\|^2}$. We write K_0 for the number of columns of P, which is $K + 2$ for the smooth lifting above and $2K + 3$ for the composite lifting of Section B.3, and F has $K_0 - 1$ entries in both cases. These function classes admit *exact* interpolation conditions, *i.e.*, for every $\theta \in \Theta$ and every $(G, F) \in \Xi^\theta$ there exists $z \in \mathcal{Z}$ with $\mathcal{A}_\theta(z) = (G, F)$, which we assume throughout and which holds when $d \ge K_0$ (Taylor et al., 2017, Theorem 5). The explicit SDP formulation of (PEP) and its Lagrangian dual are given in Section B.1.

3.2 Probabilistic guarantee from data

With finitely many samples, the empirical risk $\mathcal{R}(\theta, \widehat{\mathbf{P}}_N)$ of Section 2.2 may be a poor estimate of the true risk $\mathcal{R}(\theta, \mathbf{P})$. To mitigate the bias of the empirical risk, Park et al. (2025) propose a data-driven framework that yields a probabilistic performance guarantee robust to sampling variability. Running $\mathcal{A}_\theta$ on each sample $\hat{z}_i \in \widehat{\mathcal{D}}_N$ yields a lifted representation $(\widehat{G}_i, \widehat{F}_i) \in \Xi^\theta$, inducing the lifted empirical distribution $\widehat{\mathbf{P}}_N^\theta = (1/N)\sum_{i=1}^N \delta_{(\widehat{G}_i, \widehat{F}_i)}$ over Ξ^θ. The guarantees are derived from a Wasserstein distributionally robust variant of the performance estimation problem (DRO-PEP):

$$\begin{array}{lll} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) := & \text{maximize} & \mathcal{R}(\theta, \mathbf{Q}^\theta) \\ & \text{subject to} & \mathbf{Q}^\theta \in \mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta), \end{array} \tag{DRO-PEP}$$

where the ambiguity set $\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta)$ is the 1-Wasserstein ball of radius ε centered at $\widehat{\mathbf{P}}_N^\theta$:

$$\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta) = \left\{\mathbf{Q}^\theta \;\middle|\; \operatorname{supp}\mathbf{Q}^\theta \subseteq \Xi^\theta,\ W_1(\widehat{\mathbf{P}}_N^\theta, \mathbf{Q}^\theta) \le \varepsilon\right\},$$

with W_1 induced by the norm $\|(G, F)\| = \sqrt{\|G\|_F^2 + \|F\|^2}$. We refer to $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ as the *DRO risk*.

Tractable formulation By (Park et al., 2025, Theorem 2), (DRO-PEP) has a tractable formulation as a convex conic program; see Section B.2 for details. For $\theta \in \Theta$ and $\varepsilon > 0$, (DRO-PEP) also admits the following equivalent robust optimization formulation:

$$\begin{array}{lll} \text{maximize} & (1/N)\sum_{i=1}^N \mathbf{tr}(A_{\text{obj}}^T G_i) + b_{\text{obj}}^T F_i & \\ \text{subject to} & (1/N)\sum_{i=1}^N \|(G_i, F_i) - (\widehat{G}_i, \widehat{F}_i)\| \le \varepsilon & \\ & (G_i, F_i) \in \Xi^\theta, & i = 1, \ldots, N. \end{array} \tag{DRO-PEP-P}$$

The equivalence comes from the strong duality between the dual of (DRO-PEP) and its bi-dual, which is exactly (DRO-PEP-P). The proof of the strong duality result is deferred to Section B.4.

Proposition 1. *Let $\theta \in \Theta$ and $\varepsilon > 0$. Then* (DRO-PEP-P) *is a bi-dual of* (DRO-PEP)*, and their optimal values coincide. Furthermore,* (DRO-PEP-P) *has an optimal solution.*

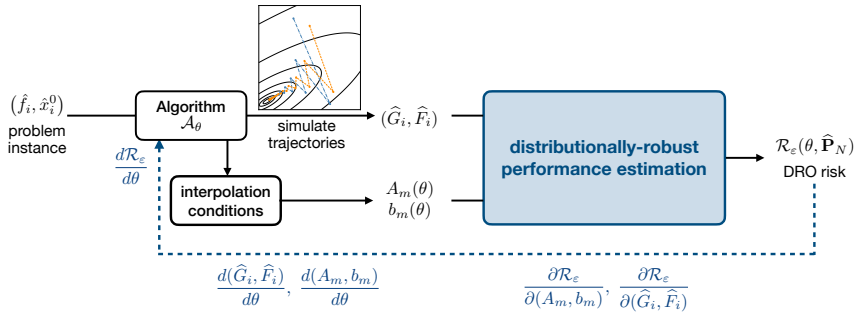


Figure 2: The diagram of the solution method for solving (DR-L2O) with stochastic gradient methods. Given θ, we have interpolation LMI coefficients $A_m(\theta)$ and $b_m(\theta)$, while a sample problem instance $(\hat{f}_i, \hat{x}_i^0)$ is mapped to the Gram matrix and function-value representation $(\widehat{G}_i, \widehat{F}_i) = \mathcal{A}_\theta(\hat{f}_i, \hat{x}_i^0)$. We evaluate the gradient of the DRO risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ w.r.t. θ through backpropagation.

3.3 Distributionally robust learning to optimize

We now minimize the DRO risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ over θ to learn optimal algorithm parameters, rather than evaluating it for a fixed θ as in (Park et al., 2025). When the learning objective is the worst-case loss, the corresponding learning problem is (Kamri et al., 2025)

$$\underset{\theta \in \Theta}{\text{minimize}} \quad \sup_{z \in \mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big). \tag{OPT-PEP}$$

However, worst-case optimal algorithms can underperform on typical instances. At the other extreme, learning to optimize (L2O) minimizes the empirical risk (Chen et al., 2022a):

$$\underset{\theta \in \Theta}{\text{minimize}} \quad \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) = (1/N) \textstyle\sum_{i=1}^N \ell\big(\mathcal{A}_\theta(\hat{z}_i)\big). \tag{L2O}$$

When the loss ℓ is differentiable, (L2O) is solved efficiently with stochastic first-order methods. However, it provides no guarantee under distribution shift: the learned algorithm $\mathcal{A}_{\theta^\star}$ may fail on instances from a different distribution. To balance the two extremes, we minimize the DRO risk, yielding the learning problem we call *distributionally robust learning to optimize* (DR-L2O):

$$\underset{\theta \in \Theta}{\text{minimize}} \quad \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N). \tag{DR-L2O}$$

This learning problem admits a minimizer over Θ. The proof of this fact is deferred to Section C.2.

Proposition 2 (Existence of a minimizer)**.** *Fix $\varepsilon > 0$. Suppose Assumptions 1 and 2 hold, together with Assumption 3 for proximal methods. Then* (DR-L2O) *attains its minimum over Θ.*

3.4 Solution method

Problem (DR-L2O) is generally nonconvex in θ, since the entries of $A_m(\theta)$ and $b_m(\theta)$ are polynomial in θ and those of $\widehat{G}_i$ and $\widehat{F}_i$ are nonlinear. Still, the DRO risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ is the optimal value of the convex conic program (DRO-PEP-D) parametrized by $\{(A_m, b_m)\}_{m=1}^M$ and $\{(\widehat{G}_i, \widehat{F}_i)\}_{i=1}^N$, whose solution map admits a tractable derivative evaluation using the approach of (Agrawal et al., 2019b). We compute the gradient $d\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)/d\theta$ via the chain rule and apply stochastic gradient descent to obtain $\hat{\theta}_\varepsilon$, which may be only a local minimizer (see Section D). See Figure 2 for a summary.

4 Theoretical guarantees of the learned optimizer

In this section, we show that (DR-L2O) not only interpolates between (OPT-PEP) and (L2O), but also yields out-of-sample and out-of-distribution guarantees.

4.1 Robustness certificate

We first show that our learned algorithm enjoys a robust out-of-sample guarantee, a consequence of the data-driven Wasserstein DRO framework (Mohajerin Esfahani & Kuhn, 2018). Here, we

extend the finite-sample guarantee of (Park et al., 2025, Section 3.2) from a fixed $\theta \in \Theta$ to a uniform statement over $\theta \in \Theta$. The proof is provided in Section C.1.

Theorem 1. *Suppose Assumptions 1 and 2 hold, together with Assumption 3 for proximal methods. Let N and K be fixed. For each $\beta \in (0,1)$, there exist $\overline{\varepsilon}(\beta) > 0$ and $\bar{L} > 0$ such that*

$$\mathbf{P}^N \left(\mathop{\mathbf{E}}_{z\sim\mathbf{P}} \big(\ell(\mathcal{A}_\theta(z))\big) \;\leq \sup_{\mathbf{Q}^\theta \in \mathcal{U}_{\overline{\varepsilon}(\beta)}(\widehat{\mathbf{P}}_N^\theta)} \mathop{\mathbf{E}}_{(G,F)\sim\mathbf{Q}^\theta} \big(\ell(G,F)\big), \quad \forall \theta \in \Theta \right) \geq 1-\beta,$$

where q is the dimension of (G,F), namely $q = K_0(K_0+1)/2 + (K_0-1)$, $\mathbf{diam}(\Theta) = \max_{\theta_1,\theta_2\in\Theta} \|\theta_1 - \theta_2\|$ is the diameter of the parameter set, $\bar{L}$ is a Lipschitz constant of the algorithm mapping $\theta \mapsto \mathcal{A}_\theta$, and, for N sufficiently large,

$$\overline{\varepsilon}(\beta) \lesssim \left(\frac{\log(1/\beta) + \dim(\Theta) \log\big(1 + 2N\,\mathbf{diam}(\Theta)\big)}{N} \right)^{1/q} + \frac{2\bar{L}}{N}.$$

Remark 1 (Out-of-distribution (OOD) guarantee). The DRO risk upper-bounds the expected loss under every distribution in the ambiguity set, not only under the true distribution. The guarantee of Theorem 1 therefore extends to test distributions that differ from the training distribution, as long as they are supported on the same class and remain within Wasserstein distance $\overline{\varepsilon}(\beta)$ of $\widehat{\mathbf{P}}_N^\theta$.

Remark 2 (Tightness of the Wasserstein radius). Since $q = O(K^2)$, the bound on $\overline{\varepsilon}(\beta)$ decays as $N^{-1/q}$ and can be conservative for large K, reflecting the Wasserstein curse of dimensionality (Fournier & Guillin, 2015). It is a sufficient radius for the guarantee, whereas in our experiments the operating radius ε is selected by cross-validation (see Section C.1 for sharper rates).

4.2 Interpolating behavior of (DR-L2O)

We now show how (DR-L2O) interpolates between (L2O) and the worst-case optimal design (OPT-PEP) as the radius ε varies. All proofs are presented in Section C.3.

Proposition 3. *Suppose Assumptions 1 and 2 hold, together with Assumption 3 for proximal methods. Let $\varepsilon > 0$, and let $\theta_\varepsilon^\star$ be an optimal solution of (DR-L2O). Then the mapping $\varepsilon \mapsto \mathcal{R}_\varepsilon(\theta_\varepsilon^\star, \widehat{\mathbf{P}}_N)$ is monotonically nondecreasing. Furthermore,*

$$\lim_{\varepsilon\to 0^+} \mathcal{R}_\varepsilon(\theta_\varepsilon^\star, \widehat{\mathbf{P}}_N) = \inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N), \qquad \lim_{\varepsilon\to\infty} \mathcal{R}_\varepsilon(\theta_\varepsilon^\star, \widehat{\mathbf{P}}_N) = \inf_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big).$$

As $\varepsilon \to 0^+$, problem (DR-L2O) loses its distributional robustness and reduces to (L2O). For large enough ε, the optimizer learned via (DR-L2O) is robust to essentially all distributions supported on Ξ^θ, becoming worst-case optimal. We now establish certified performance bounds for the true risk of our learned algorithm.

Theorem 2. *Suppose Assumptions 1 and 2 hold, together with Assumption 3 for proximal methods. Given $\varepsilon > 0$, let $\theta_\varepsilon^\star$ be an optimal solution of (DR-L2O). For any $\beta \in (0,1)$ and $\varepsilon = \overline{\varepsilon}(\beta)$ as in Theorem 1, the true risk $\mathcal{R}(\theta_\varepsilon^\star, \mathbf{P})$ of the optimizer learned via (DR-L2O) satisfies the following with probability at least $1-\beta$:*

$$\mathcal{R}(\theta_\varepsilon^\star, \mathbf{P}) \leq \inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) + \varepsilon\,\mathrm{Lip}(\ell), \qquad \mathcal{R}(\theta_\varepsilon^\star, \mathbf{P}) \leq \inf_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big),$$

where $\mathrm{Lip}(\ell) = \|(A_{\mathrm{obj}}, b_{\mathrm{obj}})\|$ is a Lipschitz constant of the loss function ℓ.

The first bound places the true risk of the optimizer learned via (DR-L2O) within $\varepsilon\,\mathrm{Lip}(\ell)$ of the in-sample optimum of (L2O), uniformly over the ambiguity set. The second bound shows that this true risk is no worse than the worst-case optimal value of (OPT-PEP). Neither bound implies strict empirical superiority over the baselines, which Section 5 demonstrates in practice.

5 Numerical experiments

We now showcase the performance of our learned optimizers[1] on logistic regression, LASSO, and total variation inpainting. For each experiment, we report the out-of-sample test loss, the fraction

[1] Code is available at https://anonymous.4open.science/r/dr-l2o-.

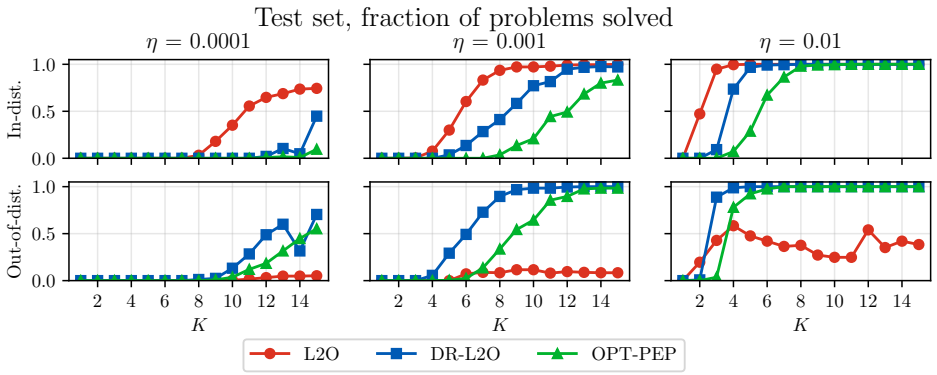


Figure 3: Fractions of logistic regression problems solved to different relative tolerances across horizon K. *Top:* Fraction solved in-distribution. *Bottom:* Fraction solved out-of-distribution.

of test instances solved at relative tolerance $\eta > 0$ (an instance $z = (f, x^0)$ is solved if $\ell(\mathcal{A}_\theta(z)) \leq \eta(1 + |f^\star|)$), and out-of-distribution performance. Implementation details and per-iteration wall-time tables are in Appendix E, while tables of all results are in Appendix F. We compare three learned optimizers with parameters $\theta^\star_\varepsilon$ from (DR-L2O), $\theta^\star_{\text{PEP}}$ from (OPT-PEP), and $\theta^\star_{\text{L2O}}$ from (L2O).

Training and parameter selection We learn a schedule of step sizes (and, for logistic regression, momentum coefficients) $\theta = \{\theta^k\}_{k=0}^{K-1}$ for each K (and learn a separate schedule for each K value as the learning horizon) and each of (DR-L2O), (OPT-PEP), and (L2O) using gradient-based methods. We cross-validate the Wasserstein radius ε, the learning rate, and the AdamW weight decay, and use a held-out test set for final loss evaluations. Full details are in Section D.

5.1 Logistic regression

Consider an unregularized logistic regression problem

$$\underset{x}{\text{minimize}} \quad (1/m)\sum_{i=1}^m \left(\log(1 + \exp(a_i^T x)) - b_i a_i^T x\right),$$

where a_i^T are the rows of the feature matrix $A \in \mathbf{R}^{m\times n}$ and $b \in \{0,1\}^m$ are the labels. The objective is smooth and convex but not strongly convex. We learn the step sizes and momentum coefficients $\theta = \{(\tau^k, \rho^k)\}_{k=0}^{K-1} \in \mathbf{R}^{2K}$ of Nesterov's fast gradient method (FGM) (Nesterov, 1983):

$$x^{k+1} = y^k - \tau^k \nabla f(y^k), \qquad y^{k+1} = x^{k+1} + \rho^k(x^{k+1} - x^k),$$

with $y^0 = x^0$. We consider the performance loss $f(x^K) - f(x^\star)$. Each instance is a uniform random subsample of the rows of the german.numer dataset from LIBSVM (Chang & Lin, 2011); the out-of-distribution shift scales the feature matrix, which increases the Lipschitz constant L (full parameters in Appendix E.1).

Results In Figure 3 we show the fractions of problems solved by the learned schedules, and the losses are provided in Figure 7 (Appendix E.1). In-distribution, we notice that (L2O) solves the most problems at every tolerance and (DR-L2O) lies between (L2O) and (OPT-PEP). Out-of-distribution, the (L2O) fraction solved at $\eta = 10^{-2}$ peaks at a small K and then falls, and at the tighter tolerances it stays near zero for all K. The (L2O) average loss does not decrease with K, so running more learned iterations degrades the out-of-distribution performance. In contrast, both (DR-L2O) and (OPT-PEP) improve with K overall, and (DR-L2O) solves every out-of-distribution instance at $\eta = 10^{-3}$ and solves the largest fraction at $\eta = 10^{-4}$, whereas (OPT-PEP) does not within the horizon.

5.2 LASSO

Consider an ℓ_1-regularized least-squares problem:

$$\underset{x}{\text{minimize}} \quad (1/2)\|Ax - b\|^2 + \lambda\|x\|_1,$$

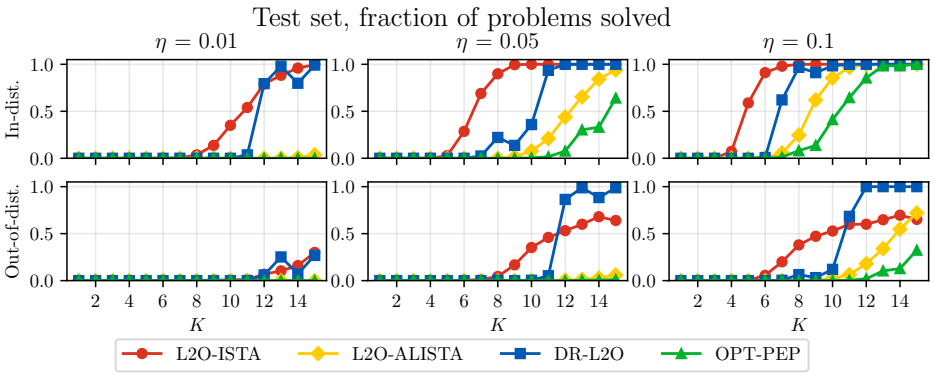


Figure 4: Fractions of LASSO problems solved to different relative tolerances across horizon K. *Top:* Fraction solved in-distribution. *Bottom:* Fraction solved out-of-distribution.

with regularization parameter $\lambda > 0$. We learn the step sizes $\{\theta^k\}_{k=0}^{K-1}$ of ISTA (Beck, 2017):

$$x^{k+1} = \mathbf{prox}_{\lambda\theta^k\|\cdot\|_1}\big(x^k - \theta^k A^T(Ax^k - b)\big),$$

where $\theta = (\theta^k)_{k=0}^{K-1} \in \mathbf{R}_+^K$. We again consider the performance loss $f(x^K) - f(x^\star)$.

Specialized learned optimizers exist for LASSO, all based on empirical risk minimization. Learned ISTA (LISTA) (Gregor & LeCun, 2010) learns weight matrices to speed up the proximal updates over a distribution of instances. For comparison, we label our default (L2O) scheme as L2O-ISTA and compare against analytic LISTA (ALISTA) (Chen et al., 2018; Liu et al., 2019), which matches the performance of LISTA with far fewer learned parameters and much cheaper training. ALISTA restricts the structure of LISTA by making the update rule $x^{k+1} = \mathbf{prox}_{\gamma^k\|\cdot\|_1}\big(x^k - \theta^k W^T(Ax^k - b)\big)$, where W is precomputed as the solution of a convex quadratic program (Liu et al., 2019, Equation 16). We label this baseline L2O-ALISTA. We use a sparse coding setup, recovering sparse vectors $\tilde{x}$ from noisy measurements $b = A\tilde{x} + \delta$ with a fixed dictionary $A \in \mathbf{R}^{m\times n}$ (Chen et al., 2018); the out-of-distribution shift increases the sparse-signal variance σ_x (full parameters in Appendix E.2).

Results We previewed the loss results of this experiment in Figure 1, and the full version with L2O-ALISTA is provided in Figure 8 (Appendix E.2), while the fractions of problems solved are provided in Figure 4. Overall, we notice the performance of L2O-ISTA degrades out of distribution significantly. Given that the 10^{th} quantile is so low and the schedule is able to solve many of the out-of-distribution problems, this implies that the schedule diverges on a small number of instances, while our (DR-L2O) learned schedule is robust against such outliers.

5.3 Total Variation (TV) Inpainting

We consider the ℓ_1 total variation (TV) inpainting problem, which fills in an image that has been corrupted with some pixels blacked out (Diamond & Boyd, 2016). For a grayscale image represented as an $m \times n$ matrix U^{orig} with values in $[0, 255]$ (scaled down to $[0, 1]$ for algorithm stability) and the set $\mathcal{K}$ of known pixel locations, the goal is to reconstruct the image U while matching the known pixels. The corresponding linear program is

$$\underset{U}{\text{minimize}} \quad \sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\left\|\begin{bmatrix} U_{i+1,j} - U_{i,j} \\ U_{i,j+1} - U_{i,j}\end{bmatrix}\right\|_1 \quad \text{subject to} \quad \begin{array}{ll} U_{ij} = U_{ij}^{\text{orig}}, & (i,j) \in \mathcal{K} \\ U_{ij} \in [0,1], & 1 \le i \le m,\ 1 \le j \le n. \end{array}$$

We solve this LP using primal-dual hybrid gradient (PDHG) (Chambolle & Pock, 2011) adjusted for linear programs (PDLP) (Applegate et al., 2023). We first cast the problem in standard LP form by introducing slack variables for the ℓ_1 terms (see Section E.3), and apply PDHG to its convex-concave saddle reformulation

$$\underset{x}{\text{minimize}}\ \underset{u}{\text{maximize}}\ \ L(x,u) = f(x) + \langle u, Mx\rangle - g^*(u),$$

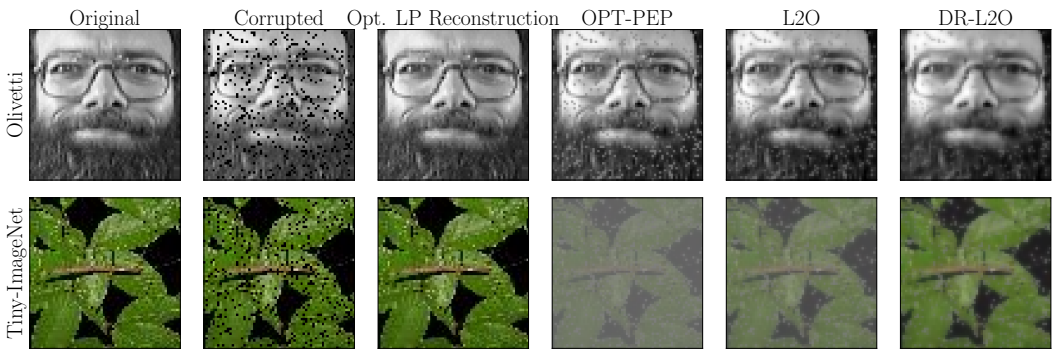


Figure 5: TV inpainting reconstructions ($K = 10$). *Left to right:* original, corrupted (10% missing pixels), optimal LP reconstruction, and reconstructions from (OPT-PEP), (L2O), and (DR-L2O).

where f encodes the LP's linear cost and box constraints, M stacks the equality and inequality constraint matrices, and $g(Mx)$ is the indicator of the constraints; see Section E.3 for the explicit form and the corresponding PDHG iteration, whose step sizes $\theta = \{(\tau^k, \rho^k, \sigma^k)\}_{k=0}^{K-1} \in \mathbf{R}_+^{3K}$ we learn. For the performance loss, we use the Lagrangian duality gap $L(x, u^\star) - L(x^\star, u)$, and since this class falls outside the standard (PEP) interpolation conditions, we extend them to general linear operators in Section B.5 (see Remark 5 for guarantees). For in-distribution we use Olivetti faces (Samaria & Harter, 1994; Cambridge, 1994) with 10% of pixels blacked out and for out-of-distribution we use color images from Tiny ImageNet (Le & Yang, 2015; Deng et al., 2009) (full parameters in Appendix E.3).

Results We show both the fractions of problems solved and the losses in Figures 10 and 9 in Appendix E.3. Here, in Figure 5, we instead show the reconstructions of images from the two datasets with the different learned schedules. The Olivetti reconstruction is not significantly different between (L2O) and (DR-L2O). However, the Tiny ImageNet reconstruction shows a difference where the learned schedules increasingly bring the image into focus and lessen the gray hue. In Figure 11 (Appendix E.3), we provide more reconstructions on extra images from Tiny ImageNet.

6 Conclusion

We presented a distributionally robust algorithm design framework for first-order methods. By minimizing the distributionally robust risk, our learned algorithms enjoy out-of-sample guarantees and a robustness to distribution shift that empirical risk minimization alone does not provide. The framework extends naturally to risk-averse training with other risk measures, such as conditional value-at-risk (Rockafellar & Uryasev, 2001); the corresponding DRO-PEP formulation already appears in (Park et al., 2025, Corollary 2.1). A major limitation is scalability: unrolling algorithms for more iterations requires solving a much larger conic program at each gradient step. A natural remedy is to use GPU-accelerated first-order solvers (Lu & Yang, 2025; Lin et al., 2026; Lu et al., 2025; Chen et al., 2025a;b). A second limitation is scope, since only the scalar span coefficients of Assumption 1 receive a gradient, which leaves learned solution maps, learned projection and preconditioner matrices, and algorithm selection outside the framework. PEP also covers more than our framework learns, since adaptive rules such as exact line search (de Klerk et al., 2017), inexact gradients (de Klerk et al., 2020), and Polyak step sizes (Barré et al., 2020) admit PEP representations but fix their step sizes from the iterates, leaving no free scalar coefficient. We therefore do not claim that fixed learned schedules displace adaptive methods, line search, or silver step sizes (Altschuler & Parrilo, 2025) in general-purpose solving.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, for checking the proofs written by the authors, for providing feedback on the design of the numerical experiments, and for interpreting the results. We have not used generative AI tools for developing the theoretical framework, formulating the mathematical claims, providing critical ingredients for their proofs, proposing or refining the hypotheses, designing the research methodology, generating synthetic data sets, or cleaning or reformatting data sets, and the remaining required-disclosure tasks, namely translation and support for qualitative and thematic data analysis, are not applicable to this work. Additionally, we used generative AI tools for editing software code, for editing the paper to improve readability, for identifying relevant literature, for formatting references, and for creating and modifying figures. We have reviewed all AI-assisted work. All AI-assisted code was verified against reference implementations and tested by the authors, and every AI-edited passage and proof step was checked line-by-line by the authors. The authors made all decisions on the experimental design and on the interpretation of the results, and checked every AI suggestion against the reported numbers. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We have read the ICLR Code of Ethics and find no concerns raised for this work. Our contribution is a training framework for the step size schedules of first-order optimization algorithms, and it involves no human subjects and no newly collected data. The experiments use two publicly available image datasets under their original terms of use: the Olivetti faces dataset (Samaria & Harter, 1994; Cambridge, 1994) and Tiny ImageNet (Le & Yang, 2015; Deng et al., 2009). We use the face images only as test images for the inpainting problem of Section 5.3 and perform no recognition, identification, or any other inference about the individuals depicted.

REPRODUCIBILITY STATEMENT

The assumptions behind our theoretical results are stated in Section 2 and Section 3, and the additional assumptions that the composite minimization setting requires are stated in Assumption 3 of Appendix B.3. The conic reformulations that make our formulations tractable are derived in Appendices B.1 and B.2. Complete proofs of all claims, including those for the composite setting, appear in Appendices B and C. The solution method with its hyperparameter settings is described in Appendix D. Appendix E reports, for every experiment, the sampling procedure and problem data, the training, validation, and test set sizes, the solver and its tolerances, the software versions, and the hardware used. An anonymized implementation that reproduces all figures and tables of the paper is available at `https://anonymous.4open.science/r/dr-l2o-`.

REFERENCES

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable Convex Optimization Layers. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019a.

Akshay Agrawal, Shane Barratt, Stephen Boyd, Enzo Busseti, and Walaa M. Moursi. Differentiating Through a Cone Program. *Journal of Applied & Numerical Optimization*, 1(2):107–115, August 2019b. ISSN 2562-5527. doi: 10.23952/jano.1.2019.2.02.

Jason M. Altschuler and Pablo A. Parrilo. Acceleration by stepsize hedging: Silver Stepsize Schedule for smooth convex optimization. *Mathematical Programming*, 213(1):1105–1118, September 2025. ISSN 1436-4646. doi: 10.1007/s10107-024-02164-2.

Brandon Amos. Tutorial on Amortized Optimization. *Foundations and Trends in Machine Learning*, 16(5):592–732, June 2023. ISSN 1935-8237. doi: 10.1561/2200000102.

Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

David Applegate, Mateo Díaz, Oliver Hinder, Haihao Lu, Miles Lubin, Brendan O'Donoghue, and Warren Schudy. Practical Large-Scale Linear Programming using Primal-Dual Hybrid Gradient. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20243–20257, New York, 2021. Curran Associates, Inc.

David Applegate, Oliver Hinder, Haihao Lu, and Miles Lubin. Faster first-order primal-dual methods for linear programming using restarts and sharpness. *Mathematical Programming*, 201(1):133–184, 2023. ISSN 1436-4646. doi: 10.1007/s10107-022-01901-9.

Maria-Florina Balcan. Data-Driven Algorithm Design. In Tim Roughgarden (ed.), *Beyond the Worst-Case Analysis of Algorithms*, pp. 626–645. Cambridge University Press, 2021. doi: 10.1017/9781108637435.036. URL https://www.cambridge.org/core/books/beyond-the-worstcase-analysis-of-algorithms/datadriven-algorithm-design/E82EB983E99AF747A36047CE982A1A96.

Mathieu Barré, Adrien Taylor, and Alexandre d'Aspremont. Complexity Guarantees for Polyak Steps with Momentum. In *Proceedings of the 33rd Conference on Learning Theory (COLT)*, volume 125 of *Proceedings of Machine Learning Research*, pp. 452–478. PMLR, 2020. URL https://proceedings.mlr.press/v125/barre20a.html.

Peter L. Bartlett, Piotr Indyk, and Tal Wagner. Generalization Bounds for Data-Driven Numerical Linear Algebra. In *Proceedings of the 35th Conference on Learning Theory (COLT)*, volume 178 of *Proceedings of Machine Learning Research*, pp. 2013–2040. PMLR, 2022. URL https://proceedings.mlr.press/v178/bartlett22a.html.

Amir Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017. ISBN 978-1-61197-498-0 978-1-61197-499-7. doi: 10.1137/1.9781611974997.

Claude Berge. *Topological Spaces: Including a Treatment of Multi-valued Functions, Vector Spaces and Convexity*. Oliver & Boyd, 1963.

Nicolas Blin, Stefano Gualandi, Christopher Maes, Andrea Lodi, and Bartolomeo Stellato. Batched First-Order Methods for Parallel LP Solving in MIP. In *International Conference on Machine Learning (ICML)*, 2026.

Nizar Bousselmi, Julien M. Hendrickx, and François Glineur. Interpolation Conditions for Linear Operators and Applications to Performance Estimation Problems. *SIAM Journal on Optimization*, 34(3):3033–3063, September 2024. ISSN 1052-6234, 1095-7189. doi: 10.1137/23M1575391.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004. ISBN 978-0-511-80444-1. doi: 10.1017/cbo9780511804441.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 3:1–122, January 2011. doi: 10.1561/2200000016.

AT&T Laboratories Cambridge. The Olivetti faces dataset. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html, 1994.

Antonin Chambolle and Thomas Pock. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011. ISSN 1573-7683. doi: 10.1007/s10851-010-0251-1.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27, 2011.

Kaihuang Chen, Defeng Sun, Yancheng Yuan, Guojun Zhang, and Xinyuan Zhao. HPR-LP: An implementation of an HPR method for solving linear programming. *Mathematical Programming Computation*, October 2025a. ISSN 1867-2957. doi: 10.1007/s12532-025-00292-0.

Kaihuang Chen, Defeng Sun, Yancheng Yuan, Guojun Zhang, and Xinyuan Zhao. HPR-QP: A dual Halpern Peaceman-Rachford method for solving large-scale convex composite quadratic programming, July 2025b.

Tianlong Chen, Xiaohan Chen, Wuyang Chen, Howard Heaton, Jialin Liu, Zhangyang Wang, and Wotao Yin. Learning to Optimize: A Primer and A Benchmark. *Journal of Machine Learning Research*, 23(189):1–59, 2022a. ISSN 1533-7928.

Xiaohan Chen, Jialin Liu, Zhangyang Wang, and Wotao Yin. Theoretical Linear Convergence of Unfolded ISTA and its Practical Weights and Thresholds. In *Advances in Neural Information Processing Systems*, volume 31, pp. 9079–9089, Canada, 2018. Curran Associates, Inc.

Xuxi Chen, Tianlong Chen, Yu Cheng, Weizhu Chen, Ahmed Awadallah, and Zhangyang Wang. Scalable Learning to Optimize: A Learned Optimizer Can Train Big Models. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIII*, pp. 389–405, Berlin, Heidelberg, October 2022b. Springer-Verlag. ISBN 978-3-031-20049-6. doi: 10.1007/978-3-031-20050-2_23.

Shuvomoy Das Gupta, Bart P. G. Van Parys, and Ernest K. Ryu. Branch-and-bound performance estimation programming: A unified methodology for constructing optimal optimization methods. *Mathematical Programming*, 204(1-2):567–639, March 2024. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-023-01973-1.

Alexandre d'Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. *Foundations and Trends in Optimization*, 5(1–2):1–245, 2021.

Etienne de Klerk, François Glineur, and Adrien B. Taylor. On the worst-case complexity of the gradient method with exact line search for smooth strongly convex functions. *Optimization Letters*, 11(7):1185–1199, 2017. ISSN 1862-4472. doi: 10.1007/s11590-016-1087-4. URL https://link.springer.com/article/10.1007/s11590-016-1087-4.

Etienne de Klerk, François Glineur, and Adrien B. Taylor. Worst-case convergence analysis of inexact gradient and Newton methods through semidefinite programming performance estimation. *SIAM Journal on Optimization*, 30(3):2053–2082, 2020. ISSN 1052-6234. doi: 10.1137/19M1281368. URL https://epubs.siam.org/doi/10.1137/19M1281368.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. IEEE, 2009.

Steven Diamond and Stephen Boyd. Total variation inpainting. CVXPY Examples, 2016. URL https://www.cvxpy.org/examples/applications/tv_inpainting.html. Accessed: 2025-05-01.

Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: A novel approach. *Mathematical Programming*, 145(1):451–482, June 2014. ISSN 1436-4646. doi: 10.1007/s10107-013-0653-0.

Gintare Karolina Dziugaite and Daniel M. Roy. Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence*. AUAI Press, August 2017.

Nicolas Fournier and Arnaud Guillin. On the rate of convergence in Wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3-4):707–738, August 2015. ISSN 0178-8051, 1432-2064. doi: 10.1007/s00440-014-0583-7.

Michael Garstka, Mark Cannon, and Paul Goulart. COSMO: A conic operator splitting method for large convex problems. In *2019 18th European Control Conference (ECC)*, pp. 1951–1956, Naples, Italy, June 2019. IEEE. ISBN 978-3-907144-00-8. doi: 10.23919/ECC.2019.8796161.

Jun-ya Gotoh, Michael Jong Kim, and Andrew E. B. Lim. Calibration of Distributionally Robust Empirical Optimization Models. *Operations Research*, 69(5):1630–1650, September 2021. ISSN 0030-364X. doi: 10.1287/opre.2020.2041.

Paul J. Goulart and Yuwen Chen. Clarabel: An interior-point solver for conic programs with quadratic objectives. *Mathematical Programming Computation*, May 2026. doi: 10.1007/s12532-026-00320-7.

Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on Machine Learning*, ICML'10, pp. 399–406, Madison, WI, USA, June 2010. Omnipress. ISBN 978-1-60558-907-7.

Quill Healey, Parth Nobel, and Stephen Boyd. Differentiating Through a Quadratic Cone Program. *Optimization Letters*, June 2026. doi: 10.1007/s11590-026-02315-8.

Howard Heaton, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Safeguarded Learned Convex Optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):7848–7855, June 2023. ISSN 2374-3468. doi: 10.1609/aaai.v37i6.25950.

Zhaolin Hu and L. Jeff Hong. Kullback-Leibler Divergence Constrained Distributionally Robust Optimization, November 2012.

Jingyi Huang, Paul Goulart, and Kostas Margellos. Data-Driven Performance Guarantees for Parametric Optimization Problems, June 2025.

Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. Sequential Model-Based Optimization for General Algorithm Configuration. In *Learning and Intelligent Optimization (LION 5)*, volume 6683 of *Lecture Notes in Computer Science*, pp. 507–523. Springer, 2011. doi: 10.1007/978-3-642-25566-3_40. URL https://link.springer.com/chapter/10.1007/978-3-642-25566-3_40.

Uijeong Jang, Shuvomoy Das Gupta, and Ernest K. Ryu. Computer-Assisted Design of Accelerated Composite Optimization Methods: OptISTA. *Mathematical Programming*, 219(1–2):391–499, September 2026. doi: 10.1007/s10107-025-02258-5.

Yassine Kamri, Julien M. Hendrickx, and François Glineur. Numerical Design of Optimized First-Order Algorithms, July 2025.

Pascal Kerschke, Holger H. Hoos, Frank Neumann, and Heike Trautmann. Automated Algorithm Selection: Survey and Perspectives. *Evolutionary Computation*, 27(1):3–45, 2019. doi: 10.1162/evco_a_00242. URL https://direct.mit.edu/evco/article/27/1/3/1083/Automated-Algorithm-Selection-Survey-and.

Donghwan Kim. Accelerated proximal point method for maximally monotone operators. *Mathematical Programming*, 190(1):57–87, November 2021. ISSN 1436-4646. doi: 10.1007/s10107-021-01643-0.

Donghwan Kim and Jeffrey A. Fessler. Optimized first-order methods for smooth convex minimization. *Mathematical Programming*, 159(1-2):81–107, September 2016. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-015-0949-3.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations*, 2015.

Daniel Kuhn, Peyman Mohajerin Esfahani, Viet Anh Nguyen, and Soroosh Shafieezadeh-Abadeh. Wasserstein Distributionally Robust Optimization: Theory and Applications in Machine Learning. In Serguei Netessine, Douglas Shier, and Harvey J. Greenberg (eds.), *Operations Research & Management Science in the Age of Analytics*, pp. 130–166. INFORMS, Washington, October 2019. ISBN 978-0-9906153-3-0. doi: 10.1287/educ.2019.0198.

Daniel Kuhn, Soroosh Shafiee, and Wolfram Wiesemann. Distributionally Robust Optimization. *Acta Numerica*, 34:579–804, 2025. doi: 10.1017/S0962492924000084.

Tung Quoc Le, Anh Tuan Nguyen, and Viet Anh Nguyen. Provably Data-Driven Lagrangian Relaxation for Mixed Integer Linear Programming. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026. URL https://arxiv.org/abs/2605.19052.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. CS 231N course report, Stanford University, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and Design of Optimization Algorithms via Integral Quadratic Constraints. *SIAM Journal on Optimization*, 26(1):57–95, January 2016. ISSN 1052-6234. doi: 10.1137/15M1009597.

Ke Li and Jitendra Malik. Learning to Optimize. In *International Conference on Learning Representations*, February 2017.

Zhenwei Lin, Zikai Xiong, Dongdong Ge, and Yinyu Ye. A Practical GPU-Enhanced Matrix-Free Primal-Dual Method for Large-Scale Conic Programs, April 2026.

Jialin Liu, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. ALISTA: Analytic Weights Are As Good As Learned Weights in LISTA. In *International Conference on Learning Representations*, May 2019.

Jialin Liu, Xiaohan Chen, Zhangyang Wang, Wotao Yin, and Hanqin Cai. Towards Constituting Mathematical Structures for Learning to Optimize. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 21426–21449. PMLR, July 2023.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic Gradient Descent with Warm Restarts. In *International Conference on Learning Representations*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, May 2019.

Haihao Lu and Jinwen Yang. cuPDLP.jl: A GPU Implementation of Restarted Primal-Dual Hybrid Gradient for Linear Programming in Julia. *Operations Research*, 73(6):3440–3452, November 2025. doi: 10.1287/opre.2024.1069.

Haihao Lu, Zedong Peng, and Jinwen Yang. cuPDLPx: A Further Enhanced GPU-Based First-Order Solver for Linear Programming, September 2025.

Andrea Martin and Giuseppe Belgioioso. Learning to accelerate Krasnosel'skii-Mann fixed-point iterations with guarantees, January 2026.

Andrea Martin and Luca Furieri. Learning to Optimize With Convergence Guarantees Using Nonlinear System Theory. *IEEE Control Systems Letters*, 8:1355–1360, 2024. ISSN 2475-1456. doi: 10.1109/LCSYS.2024.3406967.

Andrea Martin, Ian R. Manchester, and Luca Furieri. Learning to optimize with guarantees: A complete characterization of linearly convergent algorithms, August 2025.

Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1-2):115–166, September 2018. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-017-1172-1.

Yu. E. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983.

Anh Tuan Nguyen and Viet Anh Nguyen. Provably Data-Driven Projection Method for Quadratic Programming. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence*, volume 40, pp. 24541–24548, 2026. doi: 10.1609/aaai.v40i29.39637. URL https://ojs.aaai.org/index.php/AAAI/article/view/39637.

Brendan O'Donoghue. Operator splitting for a homogeneous embedding of the linear complementarity problem. *SIAM Journal on Optimization*, 31(3):1999–2023, 2021. doi: 10.1137/20M1366307.

Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016. doi: 10.1007/s10957-016-0892-3.

Neal Parikh and Stephen Boyd. Proximal Algorithms. *Foundations and Trends® in Optimization*, 1 (3):127–239, 2014. ISSN 2167-3888, 2167-3918. doi: 10.1561/2400000003.

Jisun Park, Vinit Ranjan, and Bartolomeo Stellato. Data-driven Analysis of First-Order Methods via Distributionally Robust Optimization, December 2025.

Isabeau Prémont-Schwarz, Jaroslav Vítků, and Jan Feyereisl. A Simple Guard for Learned Optimizers. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 17910–17925. PMLR, July 2022.

Vinit Ranjan and Bartolomeo Stellato. Verification of first-order methods for parametric quadratic optimization. *Mathematical Programming*, July 2025. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-025-02261-w.

Vinit Ranjan, Jisun Park, Stefano Gualandi, Andrea Lodi, and Bartolomeo Stellato. Exact Verification of First-Order Methods via Mixed-Integer Linear Programming, May 2025.

John R. Rice. The Algorithm Selection Problem. *Advances in Computers*, 15:65–118, 1976. doi: 10.1016/S0065-2458(08)60520-3. URL https://www.sciencedirect.com/science/article/pii/S0065245808605203.

R. Tyrrell Rockafellar and Stanislav Uryasev. Conditional Value-at-Risk: Optimization Approach. In Stanislav Uryasev and Panos M. Pardalos (eds.), *Stochastic Optimization: Algorithms and Applications*, pp. 411–435. Springer US, Boston, MA, 2001. ISBN 978-1-4757-6594-6. doi: 10.1007/978-1-4757-6594-6_17.

Ernest K. Ryu and Wotao Yin. *Large-Scale Convex Optimization: Algorithms & Analyses via Monotone Operators*. Cambridge University Press, Cambridge, UK, 1 edition, 2022. ISBN 978-1-009-16086-5 978-1-009-16085-8. doi: 10.1017/9781009160865.

Ernest K. Ryu, Adrien B. Taylor, Carolina Bergeling, and Pontus Giselsson. Operator Splitting Performance Estimation: Tight Contraction Factors and Optimal Parameter Selection. *SIAM Journal on Optimization*, 30(3):2251–2271, January 2020. ISSN 1052-6234, 1095-7189. doi: 10.1137/19M1304854.

Shinsaku Sakaue and Taihei Oki. Generalization Bound and Learning Methods for Data-Driven Projections in Linear Programming. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024. doi: 10.52202/079017-0408. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/1780db9272f81961459f4906036ebc6f-Abstract-Conference.html.

F.S. Samaria and A.C. Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, pp. 138–142, 1994. doi: 10.1109/ACV.1994.341300.

Rajiv Sambharya and Bartolomeo Stellato. Data-Driven Performance Guarantees for Classical and Learned Optimizers. *Journal of Machine Learning Research*, 26(171):1–49, 2025. ISSN 1533-7928.

Rajiv Sambharya and Bartolomeo Stellato. Learning Algorithm Hyperparameters for Fast Parametric Convex Optimization. *SIAM Journal on Mathematics of Data Science*, 8(3):649–676, September 2026. doi: 10.1137/24M1712242.

Rajiv Sambharya, Georgina Hall, Brandon Amos, and Bartolomeo Stellato. Learning to Warm-Start Fixed-Point Optimization Algorithms. *Journal of Machine Learning Research*, 25(166):1–46, 2024. ISSN 1533-7928.

Rajiv Sambharya, Jinho Bok, Nikolai Matni, and George Pappas. Learning Acceleration Algorithms for Fast Parametric Convex Optimization with Certified Robustness, October 2025.

Steve Smale. On the average number of steps of the simplex method of linear programming. *Mathematical Programming*, 27(3):241–262, October 1983. ISSN 1436-4646. doi: 10.1007/BF02591902.

Qingyu Song, Wei Lin, Juncheng Wang, and Hong Xu. Towards Robust Learning to Optimize with Theoretical Guarantees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27498–27506, 2024.

Bartolomeo Stellato, Goran Banjac, Paul Goulart, Alberto Bemporad, and Stephen Boyd. OSQP: An operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4): 637–672, 2020. ISSN 1867-2957. doi: 10.1007/s12532-020-00179-2.

Michael Sucker, Jalal Fadili, and Peter Ochs. Learning-to-Optimize with PAC-Bayesian Guarantees: Theoretical Considerations and Practical Implementation. *Journal of Machine Learning Research*, 26(211):1–53, 2025. ISSN 1533-7928.

Adrien B. Taylor, Julien M. Hendrickx, and François Glineur. Smooth strongly convex interpolation and exact worst-case performance of first-order methods. *Mathematical Programming*, 161(1-2): 307–345, January 2017. ISSN 0025-5610, 1436-4646. doi: 10.1007/s10107-016-1009-3.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge, 2018. doi: 10.1017/9781108231596.

Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance. *Bernoulli*, 25(4A):2620–2648, 2019. doi: 10.3150/18-BEJ1065.

Wolfram Wiesemann, Daniel Kuhn, and Melvyn Sim. Distributionally Robust Convex Optimization. *Operations Research*, 62(6):1358–1376, December 2014. ISSN 0030-364X. doi: 10.1287/opre.2014.1314.

Zikai Xiong. High-Probability Polynomial-Time Complexity of Restarted PDHG for Linear Programming, January 2025.

Junjie Yang, Xuxi Chen, Tianlong Chen, Zhangyang Wang, and Yingbin Liang. M-L2O: Towards Generalizable Learning-to-Optimize by Test-Time Fast Self-Adaptation. In *The Eleventh International Conference on Learning Representations*, May 2023.

A Related works

A.1 Worst-case analysis and its limitations

The performance estimation problem (PEP) (Drori & Teboulle, 2014; Taylor et al., 2017) and control-theoretic analysis (Lessard et al., 2016) automate worst-case convergence analysis for first-order methods in convex optimization, while verification frameworks (Ranjan & Stellato, 2025; Ranjan et al., 2025) extend these ideas to parametric quadratic and linear optimization. These approaches have enabled the discovery of worst-case optimal algorithms, including the optimized gradient method (OGM) (Kim & Fessler, 2016), OptISTA (Jang et al., 2026), and the accelerated proximal point method (APPM) (Kim, 2021), as well as accelerated gradient methods based on silver step-size schedules (Das Gupta et al., 2024; Altschuler & Parrilo, 2025). Kamri et al. (2025) design such methods numerically by differentiating the PEP worst-case bound with respect to the step sizes. However, worst-case bounds are often far from typical performance. Average-case analysis (Smale, 1983; Xiong, 2025) offers tighter characterizations when problem distributions are known, while data-driven evaluation (Sambharya & Stellato, 2025; Huang et al., 2025; Park et al., 2025) provides performance bounds on observed instances. Yet worst-case design optimizes hyperparameters only for robustness, while average-case and data-driven analyses only evaluate the performance of a given method.

A.2 Learning to optimize (L2O)

L2O (Andrychowicz et al., 2016; Li & Malik, 2017; Chen et al., 2022a) learns algorithm components from a dataset of problem instances. Its paradigms include amortized optimization (Amos, 2023), algorithm selection (Rice, 1976; Kerschke et al., 2019), and algorithm configuration (Hutter et al., 2011; Balcan, 2021), the last of which learns hyperparameters within a fixed algorithmic structure and is the setting of this paper. This has achieved practical success, from learned proximal methods that recover sparse signals in only a few iterations (Liu et al., 2019) to scalable approaches for neural network training (Chen et al., 2022b). However, L2O is essentially empirical risk minimization, which alone does not ensure robustness to a new distribution.

Two strategies have emerged to address this gap. The first imposes structural constraints that ensure convergence: Chen et al. (2018) identified algorithm structures that are both convergent and trainable, Prémont-Schwarz et al. (2022) and Heaton et al. (2023) developed safeguarded formulations guaranteeing convergence for every instance, Liu et al. (2023) derived necessary conditions for convergent learned algorithms, and Martin & Furieri (2024); Martin et al. (2025) developed parametrizations of learned algorithms with convergence guarantees, including a complete characterization of convergence-preserving modifications for several classes of linearly convergent algorithms. The second strategy incorporates distribution-aware objectives. Some improve generalization through meta-learning (Yang et al., 2023) or data alignment for out-of-distribution robustness (Song et al., 2024). Others certify the learned algorithm, either by regularizing the learning objective with worst-case convergence rates (Sambharya et al., 2025) or through statistical bounds from PAC-Bayes (Sambharya et al., 2024; Sucker et al., 2025) and held-out validation data (Sambharya & Stellato, 2026).

However, these strategies have limitations. Structural constraints reduce learning capacity, while distribution-aware objectives either reintroduce worst-case conservatism or certify a quantity other than the one deployed. The PAC-Bayes bound of Sambharya et al. (2024) holds for the fixed learned weights, at the price of a margin term that takes the worst case over a perturbation of the warm start. The PAC-Bayes guarantee of Sucker et al. (2025) bounds the loss of a Gibbs predictor, namely hyperparameters drawn afresh from a learned posterior at every instance, whereas one deploys a single fixed choice. More fundamentally, these bounds and the validation bound of Sambharya & Stellato (2026) certify performance only under the single distribution that generated the training instances, which is the target we replace by a Wasserstein ball around the empirical distribution.

A parallel line in data-driven algorithm design certifies empirical risk minimization itself in the standard PAC sense, bounding the gap between empirical and expected performance uniformly over the hyperparameters with pseudo-dimension (Sakaue & Oki, 2024; Nguyen & Nguyen, 2026) or fat-shattering (Bartlett et al., 2022) arguments and Rademacher complexity with matching minimax lower bounds (Le et al., 2026). As with the bounds above, those works take the expectation at the

training distribution. This line therefore offers complementary frameworks rather than a bound established for our L2O baseline. Le et al. (2026) name shifts in the problem distribution an important open problem, which is the problem we take up.

A.3 Distributionally robust optimization (DRO)

DRO (Kuhn et al., 2025) handles uncertainty by optimizing over ambiguity sets, which are collections of distributions consistent with partial information. When only data samples are available, data-driven DRO (Wiesemann et al., 2014; Kuhn et al., 2025) constructs ambiguity sets as metric balls centered at the empirical distribution, using Wasserstein distance (Mohajerin Esfahani & Kuhn, 2018; Kuhn et al., 2019) or divergences such as Kullback–Leibler (KL) divergence (Hu & Hong, 2012). This framework naturally quantifies distributional uncertainty from finite samples, providing the mechanism we exploit to interpolate between L2O and worst-case PEP.

B Additional details and proofs for Section 3

Risk on lifted distributions For a distribution $\mathbf{Q}$ over $\mathcal{Z}$, let $\mathbf{Q}^\theta$ denote its pushforward under $\mathcal{A}_\theta$, which is a distribution over Ξ^θ. The loss depends on z only through the lift $\mathcal{A}_\theta(z) = (G, F)$, so the risk (1) satisfies $\mathcal{R}(\theta, \mathbf{Q}) = \mathbf{E}_{(G,F)\sim\mathbf{Q}^\theta}\big(\ell(G, F)\big)$. We use the right-hand side to define $\mathcal{R}(\theta, \mathbf{Q}^\theta)$ for any distribution $\mathbf{Q}^\theta$ supported on Ξ^θ, which is how $\mathcal{R}$ enters (DRO-PEP). The Wasserstein ball $\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta)$ therefore lives in the lifted space Ξ^θ, which depends on the algorithm through θ. The empirical distribution $\widehat{\mathbf{P}}_N^\theta$ is the pushforward of $\widehat{\mathbf{P}}_N$, so $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ is a function of $\widehat{\mathbf{P}}_N$ through $\widehat{\mathbf{P}}_N^\theta$.

B.1 Tractable convex conic formulation of (PEP)

The performance estimation problem (PEP) admits the following tractable formulation:

$$\begin{array}{lllll} \sup\limits_{z\in\mathcal{Z}} & \ell(\mathcal{A}_\theta(z)) & = & \text{maximize} & \ell(G,F)\left(:= \mathbf{tr}(A_{\text{obj}}^T G) + b_{\text{obj}}^T F\right) \\ & & & \text{subject to} & \mathcal{S}_\theta(G,F) \in \mathbf{R}_+^M \\ & & & & \mathbf{tr}(A_0^T G) + b_0^T F + c_0 \le 0 \\ & & & & G \in \mathbf{S}_+^{K+2},\ F \in \mathbf{R}^{K+1}. \end{array}$$

One instance of the initial condition is the bound of Assumption 2(*i*). The choice $A_0 = e_1 e_1^T$, $b_0 = 0$, and $c_0 = -R_0^2$, with e_1 being the first standard basis vector of $\mathbf{R}^{K+2}$, makes $\mathbf{tr}(A_0^T G) + b_0^T F + c_0 \le 0$ equivalent to $\|x^0 - x^\star\|^2 - R_0^2 \le 0$. The Lagrangian dual of the problem above writes as

$$\begin{array}{ll} \text{minimize} & -c_0\tau \\ \text{subject to} & -\mathcal{S}_\theta^*(y) + \tau(A_0, b_0) - (A_{\text{obj}}, b_{\text{obj}}) \in \mathbf{S}_+^{K+2} \times \{0\} \\ & \tau \ge 0,\ y \in \mathbf{R}_+^M, \end{array}$$

where $\mathcal{S}_\theta^*\colon \mathbf{R}^M \to \mathbf{S}^{K+2} \times \mathbf{R}^{K+1}$ is an adjoint mapping of $\mathcal{S}_\theta$ defined as $\mathcal{S}_\theta^*(y) = -\sum_{m=1}^M y_m(A_m, b_m)$. For most first-order methods, strong duality holds true from (Taylor et al., 2017, Theorem 6).

B.2 Tractable form of (DRO-PEP) with expected loss

The problem (DRO-PEP) with expected loss admits a tractable convex conic formulation (Park et al., 2025):

$$\begin{array}{ll} \text{minimize} & (1/N)\sum_{i=1}^N s_i \\ \text{subject to} & -c_0\tau_i - \big\langle (X_i, Y_i),\, (\widehat{G}_i, \widehat{F}_i)\big\rangle + \lambda\varepsilon \le s_i \\ & -\mathcal{S}_\theta^*(y_i) - (X_i, Y_i) + \tau_i(A_0, b_0) - (A_{\text{obj}}, b_{\text{obj}}) \in \mathbf{S}_+^{K+2} \times \{0\} \\ & \|(X_i, Y_i)\| \le \lambda, \qquad\qquad i = 1, \dots, N, \end{array} \tag{DRO-PEP-D}$$

with optimization variables $s_i \in \mathbf{R}$, $\tau_i \in \mathbf{R}_+$, $(X_i, Y_i) \in \mathbf{S}^{K+2} \times \mathbf{R}^{K+1}$, $y_i \in \mathbf{R}_+^M$ for $i = 1, \dots, N$, and $\lambda \in \mathbf{R}_+$, where $\mathcal{S}_\theta^*$ is an adjoint mapping of $\mathcal{S}_\theta$.

B.3 Composite minimization

In this subsection, we provide a detailed treatment of the *composite minimization* setting, where the objective function is split into a smooth part f and a nonsmooth part h:

$$\underset{x}{\text{minimize}} \quad f(x) + h(x), \tag{3}$$

where f is L-smooth convex as in Assumption 2(i) and $h\colon \mathbf{R}^d \to \mathbf{R} \cup \{\infty\}$ is convex, proper, and lower semi-continuous. The set of instances $\mathcal{Z}$ is now a set of triples (f, h, x^0), where x^0 is the initial point.

Given that (3) attains its minimum, for any solution $x^\star$ we have $0 \in \nabla f(x^\star) + \partial h(x^\star)$, where $\partial h(x^\star)$ is the subdifferential of h at $x^\star$. The gradient $\nabla f(x^\star)$ therefore need not vanish, which is in contrast to the unconstrained minimization setting where $h \equiv 0$ and $\nabla f(x^\star) = 0$.

Proximal-gradient methods We consider the class of proximal-gradient methods, which is defined by the sequence of iterations generated by the following update rules:

$$\begin{aligned} y^k &= x^k - \textstyle\sum_{i=0}^{k-1} \phi_i^k \nabla f(y^i) - \sum_{i=0}^{k-1} \psi_i^k s^i, \\ x^{k+1} &= \mathbf{prox}_{\theta_{\text{prox}}^k h}\left(y^k - \textstyle\sum_{i=0}^{k} \theta_i^k \nabla f(y^i)\right), \end{aligned} \tag{4}$$

for $k = 0, \ldots, K-1$. This algorithm uses two oracle calls: the gradient of smooth f and the proximal operator of nonsmooth h. Denoting by w^k the intermediate point

$$w^k = y^k - \sum_{i=0}^{k} \theta_i^k \nabla f(y^i), \qquad k = 0, 1, \ldots, K-1,$$

at which the proximal operator is evaluated, we may extract the subgradient s^k as

$$s^k = \frac{w^k - x^{k+1}}{\theta_{\text{prox}}^k} \in \partial h(x^{k+1}), \qquad k = 0, 1, \ldots, K-1. \tag{5}$$

The empty sums make $y^0 = x^0$. The parameter θ collects the scalars ϕ_i^k, ψ_i^k, θ_i^k, and θ_{prox}^k, so every coefficient in (4) is a coordinate of θ.

Note that this proximal-gradient method definition includes the standard proximal-gradient method or ISTA, and also its accelerated variant FISTA (Beck, 2017). Setting ϕ_i^k and ψ_i^k to zero gives the single-sequence iteration $y^k = x^k$, which is ISTA when $\theta_k^k = \theta_{\text{prox}}^k$ and the other θ_i^k are zero. The FISTA iteration extrapolates $y^k = x^k + \beta_k(x^k - x^{k-1})$ before querying the gradient (Beck, 2017, Chapter 10), and the difference $x^k - x^{k-1}$ is a linear combination of the oracle outputs already produced, so this y^k is also of the form (4).

Assumption 3 (Proximal parameter and composite instances)**.** *Suppose that the algorithm $\mathcal{A}_\theta$ is of the form* (4)*.*

(i) There exist constants $0 < \theta_{\min} \le \theta_{\max} < \infty$ such that the proximal parameters satisfy $\theta_{\text{prox}}^k \in [\theta_{\min}, \theta_{\max}]$ for every $\theta \in \Theta$ and every $k = 0, \ldots, K-1$.

(ii) There exists a constant R_∇ such that

$$\|\nabla f(x^\star)\| \le R_\nabla \tag{6}$$

for every composite instance $z = (f, h, x^0) \in \mathcal{Z}$ with solution $x^\star$.

Part (i) gives a uniform lower bound on the proximal parameters, leaving the gradient step sizes free as in Assumption 2(ii). The proximal-residual bound needed below follows from part (ii) rather than requiring an independent assumption. Since $x^\star$ solves (3), we have $-\nabla f(x^\star) \in \partial h(x^\star)$, or equivalently, $\mathbf{prox}_{\eta h}\big(x^\star - \eta \nabla f(x^\star)\big) = x^\star$ for every $\eta > 0$. Nonexpansiveness of the proximal operator therefore gives

$$\left\|x^\star - \mathbf{prox}_{\eta h}(x^\star)\right\| \le \eta \left\|\nabla f(x^\star)\right\| \le \theta_{\max} R_\nabla =: R_{\text{prox}}, \qquad \eta \in [\theta_{\min}, \theta_{\max}]. \tag{7}$$

Remark 3 (Necessity of the proximal lower bound). The lower bound of Assumption 3(i) is not an artifact of the arguments below. Take $h = |\cdot|$ on $\mathbf{R}$, whose proximal operator is scalar soft thresholding. The subgradient (5) contributed by a proximal step at input v and parameter α is $\mathbf{sign}(v)\min\{|v|/\alpha, 1\}$. The derivative of this expression in α is $-v/\alpha^2$ on $|v| < \alpha$, whose supremum over v grows like $1/\alpha$. This subgradient is therefore not Lipschitz in θ uniformly over instances as the proximal parameter approaches zero.

Remark 4. One way to verify Assumption 3(ii) is to compare the composite solution with a minimizer of the smooth part. Suppose in addition that f attains its minimum at $x_f^\star$. Then

$$\|\nabla f(x^\star)\| = \left\|\nabla f(x^\star) - \nabla f(x_f^\star)\right\| \le L\left\|x^\star - x_f^\star\right\|,$$

by Assumption 2(i), so it suffices that one can choose $x_f^\star$ within a uniformly bounded distance of $x^\star$ over $z \in \mathcal{Z}$.

We now define the lifted representation (G, F) of a composite instance.

Definition 1 (Lifted representation of a composite instance)**.** Let $z = (f, h, x^0) \in \mathcal{Z}$ be a composite instance with solution $x^\star$, let $\{x^k\}_{k=0}^K = \mathcal{A}_\theta(z)$ be the first $K+1$ iterates, and let $\{y^k\}_{k=0}^{K-1}$ be the gradient-query points of (4). The *lifted representation* of z under $\mathcal{A}_\theta$ is the pair $(G, F) \in \mathbf{S}_+^{2K+3} \times \mathbf{R}^{2K+2}$, where $G = P^T P$ is the Gram matrix of

$$P = \begin{bmatrix} x^0 - x^\star & \nabla f(y^0) & \cdots & \nabla f(y^{K-1}) & \nabla f(x^K) & s^\star & s^0 & \cdots & s^{K-1} \end{bmatrix} \in \mathbf{R}^{d\times(2K+3)}, \tag{8}$$

with $s^\star = -\nabla f(x^\star) \in \partial h(x^\star)$ the optimality column and s^k the subgradient (5) contributed by the k-th proximal step, and where

$$\begin{aligned} F = \big(&f(y^0) - (f+h)^\star,\ \ldots,\ f(y^{K-1}) - (f+h)^\star,\ f(x^K) - (f+h)^\star, \\ &h(x^\star),\ h(x^1) - (f+h)^\star,\ \ldots,\ h(x^K) - (f+h)^\star\big) \in \mathbf{R}^{2K+2}, \end{aligned} \tag{9}$$

with $(f+h)^\star = f(x^\star) + h(x^\star)$ the optimal value of (3), normalized to $f(x^\star) = h(x^\star) = 0$ so that the entry $h(x^\star)$ vanishes. This is without loss of generality, since replacing (f, h) by $(f - f(x^\star), h - h(x^\star))$ changes no iterate, oracle output, or gap loss, and without it Ξ^θ is unbounded along the shift $(f - c, h + c)$.

The smooth part is interpolated over the gradient-query points $y^0, \ldots, y^{K-1}$ together with the final iterate x^K and the solution $x^\star$, and the nonsmooth part over $x^1, \ldots, x^K$ and $x^\star$. The matrix P therefore has $K_0 = 2K + 3$ columns here, compared with $K_0 = K + 2$ for the smooth lifting of Section 3.1, and F has $K_0 - 1$ entries in both cases. In both cases P carries one column per interpolation point together with the initial-distance column $x^0 - x^\star$, while F carries one entry per interpolation point, each value measured relative to the optimal value $(f+h)^\star$ of the sum. Smooth minimization is the case $h \equiv 0$, where the optimality column, the proximal columns, and the h-value entries are all absent, so that (8) and (9) reduce to the smooth lifting. Both the interpolation conditions and the loss of (PEP) are linear in this pair, so the tractable formulations of Section B.1 apply with $K_0 = 2K + 3$ in place of $K + 2$ and $F \in \mathbf{R}^{2K+2}$ in place of $F \in \mathbf{R}^{K+1}$ (Park et al., 2025, Appendix A). The support set Ξ^θ also contains the equality $F_{K+2} = 0$ and the bound (6) as $G_{K+3,K+3} \le R_\nabla^2$, each handled like the initial condition. Our implementation omits the latter bound, which enlarges the support set, keeps the certificate of Theorem 1 valid, and is needed only for the compactness of Lemma 2 and the large-radius limit of Proposition 3.

Neither the iterates nor the gradient-query points are themselves columns of P. The update rules (4) write each of them as an affine combination of the columns of (8). Only the $K+1$ gradient columns and the K proximal columns depend on θ, since x^0, $x^\star$, and $s^\star$ belong to the instance z.

B.4 Proof of Proposition 1

Proof of Proposition 1. The dual derivation in (Park et al., 2025, Theorem 2) shows that (DRO-PEP-P) is a dual of (DRO-PEP-D), and hence a bi-dual of (DRO-PEP). It remains to prove that (DRO-PEP) and (DRO-PEP-P) have the same optimal value and that the latter attains its optimum.

Write $Z = (G, F)$ and $\widehat{Z}_i = (\widehat{G}_i, \widehat{F}_i)$, and denote the linear loss by

$$\ell(Z) = \mathbf{tr}(A_{\mathrm{obj}}^T G) + b_{\mathrm{obj}}^T F.$$

Recall that Ξ^θ is closed and convex and that $\widehat{Z}_i \in \Xi^\theta$ for every $i = 1, \ldots, N$.

First, let $\mathbf{Q}^\theta \in \mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta)$ be feasible for (DRO-PEP). Since the first marginal is the empirical distribution $\widehat{\mathbf{P}}_N^\theta = (1/N)\sum_{i=1}^N \delta_{\widehat{Z}_i}$, an optimal coupling between $\widehat{\mathbf{P}}_N^\theta$ and $\mathbf{Q}^\theta$ can be written as

$$\pi = \frac{1}{N}\sum_{i=1}^N \delta_{\widehat{Z}_i} \otimes \mathbf{Q}_i, \qquad \mathbf{Q}^\theta = \frac{1}{N}\sum_{i=1}^N \mathbf{Q}_i,$$

where each $\mathbf{Q}_i$ is supported on Ξ^θ. Define the barycenter

$$\overline{Z}_i = \int_{\Xi^\theta} Z \, d\mathbf{Q}_i(Z).$$

Since Ξ^θ is closed and convex, the point $\overline{Z}_i$ belongs to Ξ^θ. Jensen's inequality gives

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^N \|\overline{Z}_i - \widehat{Z}_i\| &\le \frac{1}{N}\sum_{i=1}^N \int_{\Xi^\theta} \|Z - \widehat{Z}_i\| \, d\mathbf{Q}_i(Z) \\ &= W_1(\widehat{\mathbf{P}}_N^\theta, \mathbf{Q}^\theta) \le \varepsilon.\end{aligned}$$

Thus, $(\overline{Z}_1, \ldots, \overline{Z}_N)$ is feasible for (DRO-PEP-P). Since ℓ is linear,

$$\begin{aligned}\mathop{\mathbf{E}}_{Z\sim\mathbf{Q}^\theta}\big(\ell(Z)\big) &= \frac{1}{N}\sum_{i=1}^N \int_{\Xi^\theta} \ell(Z) \, d\mathbf{Q}_i(Z) \\ &= \frac{1}{N}\sum_{i=1}^N \ell(\overline{Z}_i).\end{aligned}$$

Every feasible distribution for (DRO-PEP) therefore induces a feasible solution of (DRO-PEP-P) with the same objective value, so

$$\text{val}(\text{DRO-PEP}) \le \text{val}(\text{DRO-PEP-P}). \tag{10}$$

Conversely, let $(Z_1, \ldots, Z_N)$ be feasible for (DRO-PEP-P), and define

$$\mathbf{Q}^\theta = \frac{1}{N}\sum_{i=1}^N \delta_{Z_i}.$$

The coupling $\pi = (1/N)\sum_{i=1}^N \delta_{(\widehat{Z}_i, Z_i)}$ satisfies

$$W_1(\widehat{\mathbf{P}}_N^\theta, \mathbf{Q}^\theta) \le \frac{1}{N}\sum_{i=1}^N \|Z_i - \widehat{Z}_i\| \le \varepsilon.$$

Thus, $\mathbf{Q}^\theta$ is feasible for (DRO-PEP), and

$$\mathop{\mathbf{E}}_{Z\sim\mathbf{Q}^\theta}\big(\ell(Z)\big) = \frac{1}{N}\sum_{i=1}^N \ell(Z_i).$$

This construction gives the reverse inequality to (10), and hence the two optimal values coincide.

It remains to prove attainment. Every feasible point of (DRO-PEP-P) satisfies

$$\|Z_i - \widehat{Z}_i\| \le N\varepsilon, \qquad i = 1, \ldots, N.$$

Its feasible set is therefore a closed subset of

$$\prod_{i=1}^N \left(\Xi^\theta \cap \{Z \mid \|Z - \widehat{Z}_i\| \le N\varepsilon\}\right),$$

which is compact because Ξ^θ is closed and the lifted space is finite-dimensional. The feasible set is nonempty because $Z_i = \widehat{Z}_i$ for every i is feasible. The objective is continuous, so (DRO-PEP-P) attains its optimal value. ■

B.5 Extended dual formulation of (DRO-PEP) with linear operator interpolations

Consider the case where the support set Ξ is defined with an additional positive semidefinite constraint as follows.

$$\Xi = \left\{ (G,F) \in \mathbf{S}_+^{K+2} \times \mathbf{R}^{K+1} \;\middle|\; \begin{array}{l} \mathbf{tr}(A_0^T G) + b_0^T F + c_0 \le 0 \\ \mathbf{tr}(A_m^T G) + b_m^T F \le 0,\ m = 1, \dots, M, \\ H \in \mathbf{S}_+^{K+1} \text{ where } (H)_{kl} = \mathbf{tr}(C_{kl}^T G) + d_{kl}^T F \end{array} \right\}.$$

In this subsection, Ξ denotes a support set of this form. The classes of linear operators with bounded spectral norm and convex quadratic functions are such cases, as the following example illustrates.

Example 1 (Interpolation condition of linear operators). We state (Bousselmi et al., 2024, Theorem 3.1), which provides interpolation conditions for linear operators, along with convex quadratic functions as a special case. Let $X \in \mathbf{R}^{n \times N_1}$, $Y \in \mathbf{R}^{m \times N_1}$, $U \in \mathbf{R}^{m \times N_2}$, $V \in \mathbf{R}^{n \times N_2}$ and $L \ge 0$. Then there exists $M \in \mathbf{R}^{m \times n}$ with $\sigma_{\max}(M) \le L$ such that

$$Y = MX, \qquad V = M^T U,$$

if and only if

$$X^T V = Y^T U, \qquad Y^T Y \preceq L^2 X^T X, \qquad V^T V \preceq L^2 U^T U.$$

Note that the first condition can be written as $\mathbf{tr}(A_m^T G) + b_m^T F \le 0$ and $-\mathbf{tr}(A_m^T G) - b_m^T F \le 0$ by writing each entry of the matrix $X^T V - Y^T U$ as the left-hand side of these inequalities. For the second constraint, let $H = L^2 X^T X - Y^T Y$ and $H_2 = L^2 U^T U - V^T V$. Then $H \succeq 0$ and each entry of H is a linear combination of bilinear terms of (X, Y). Therefore, it is of the form

$$(H)_{kl} = \mathbf{tr}(C_{kl}^T G) + d_{kl}^T F, \qquad 1 \le k, l \le K+1,$$

where $C_{kl} \in \mathbf{S}^{K+2}$ and $d_{kl} \in \mathbf{R}^{K+1}$. We can do similarly for the last constraint.

The conditions above can be specialized to a symmetric positive semidefinite matrix Q such that $\mu I \preceq Q \preceq LI$, in the sense that $g^k = Qx^k$ and $f^k = (1/2)(x^k)^T Q x^k$ for $k = 0, \dots, K$ if and only if, for $X = (x^0, \dots, x^K) \in \mathbf{R}^{d \times (K+1)}$, $P = (g^0, \dots, g^K) \in \mathbf{R}^{d \times (K+1)}$, and $F = (f^0, \dots, f^K) \in \mathbf{R}^{K+1}$, (X, P, F) is an element of the following set:

$$\mathcal{Q}_{\mu,L} = \left\{ (X,P,F) \in \mathbf{R}^{d \times (K+1)} \times \mathbf{R}^{d \times (K+1)} \times \mathbf{R}^{K+1} \;\middle|\; \begin{array}{l} X^T P = P^T X \\ (P - \mu X)^T (LX - P) \succeq 0 \\ f^i = \frac{1}{2}(x^i)^T g^i,\ i = 0, \dots, K \end{array} \right\}. \tag{11}$$

Remark 5 (Extension to PDHG). The PDHG iteration of Section E.3 is not of the form (4), but the proofs of Lemmas 2 and 3 carry over as follows, with the lifted representation of the Chambolle–Pock method in (Bousselmi et al., 2024, Section 4.2). Suppose $\|M\|_2 \le M_{\max}$ and $\|(M^T u^\star, M x^\star)\| \le R_\nabla$ for every instance. The oracles M and M^T are then Lipschitz with constant $M_{\max}$ and are applied to affine combinations of past columns, so $M_{\max}$ replaces L. The two proximal maps are the blocks of the proximal map of the separable function $(x, u) \mapsto f(x) + g^*(u)$ with block parameters (τ^k, σ^k), so $\min\{\tau_{\min}, \sigma_{\min}\}$ replaces $\theta_{\min}$ and the extrapolation $\bar{x}^{k+1}$ plays the role of the second sequence. The functions f and g^* are convex, and $\|(x^0 - x^\star, u^0 - u^\star)\| \le R_0$ replaces the initial condition. The interpolation conditions of Example 1 combined with those of f and g^* are exact when the dimensions n and m are large enough, since (Bousselmi et al., 2024, Theorem 3.1) is an equivalence and the convex interpolation of f and g^* constrains only their own triples (Taylor et al., 2017, Theorem 4). Therefore, both lemmas hold for PDHG with (τ, ρ, σ) in a box with $\tau_{\min}, \sigma_{\min} > 0$, and so do Theorems 1 and 2 and Propositions 2 and 3.

Lemma 1. *Let Ξ be of the form above. The worst-case expected loss*

$$\begin{array}{ll} \text{maximize} & \mathop{\mathbf{E}}_{z \sim \mathbf{Q}} \big(\ell(\mathcal{A}_\theta(z)) \big) \\ \text{subject to} & \mathbf{Q} \in \mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N), \quad \operatorname{supp} \mathbf{Q} \subseteq \Xi, \end{array}$$

is upper-bounded by the optimal value of the problem

$$\begin{array}{ll} \text{minimize} & (1/N) \sum_{i=1}^N s_i \\ \text{subject to} & -c_0 \tau_i - \mathbf{tr}(X_i^T \widehat{G}_i) - Y_i^T \widehat{F}_i + \lambda \varepsilon \le s_i \\ & \tau_i A_0 + \sum_{m=1}^M (y_i)_m A_m - \sum_{k,l=1}^{K+1} (\widetilde{H}_i)_{kl} C_{kl} - A_{\text{obj}} - X_i \in \mathbf{S}_+^{K+2} \\ & \tau_i b_0 + \sum_{m=1}^M (y_i)_m b_m - \sum_{k,l=1}^{K+1} (\widetilde{H}_i)_{kl} d_{kl} - b_{\text{obj}} - Y_i = 0 \\ & \|(X_i, Y_i)\| \le \lambda, \qquad\qquad i = 1, \dots, N, \end{array}$$

with variables $s_i \in \mathbf{R}$, $\tau_i \in \mathbf{R}_+$, $y_i \in \mathbf{R}_+^M$, $X_i \in \mathbf{S}^{K+2}$, $Y_i \in \mathbf{R}^{K+1}$, $\widetilde{H}_i \in \mathbf{S}_+^{K+1}$, *and* $\lambda \in \mathbf{R}_+$. *Equality holds if* Ξ *contains a point with* $G \succ 0$ *and* $H \succ 0$.

Proof. The worst-case expected loss given can be written as

$$\sup_{\mathbf{Q} \in \mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N)} \mathop{\mathbf{E}}_{z \sim \mathbf{Q}} \big(\ell(\mathcal{A}_\theta(z))\big) \;=\; \sup_{\mathbf{Q}^\theta \in \mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta)} \mathop{\mathbf{E}}_{(G,F) \sim \mathbf{Q}^\theta} \big(\ell(G,F)\big).$$

Circling back to the proof of (Park et al., 2025, Theorem 2), this is equivalent to

$$\begin{array}{ll} \text{minimize} & (1/N)\sum_{i=1}^N s_i \\ \text{subject to} & (-\ell)^*\big((X_i,Y_i)-(U_i,V_i)\big) + \sigma_\Xi(U_i,V_i) - \mathbf{tr}(X_i^T\widehat{G}_i) - Y_i^T\widehat{F}_i + \lambda\varepsilon \le s_i \\ & \|(X_i,Y_i)\| \le \lambda, \qquad\qquad i=1,\ldots,N, \end{array}$$

with variables $s_i \in \mathbf{R}$, $\lambda \in \mathbf{R}_+$, $X_i, U_i \in \mathbf{S}^{K+2}$, and $Y_i, V_i \in \mathbf{R}^{K+1}$ for $i = 1, \ldots, N$. First of all,

$$\begin{aligned} (-\ell)^*(X,Y) &= \sup_{(G,F)\in\mathbf{S}^{K+2}\times\mathbf{R}^{K+1}} \big(\mathbf{tr}(X^TG) + Y^TF + \ell(G,F)\big) \\ &= \sup_{(G,F)\in\mathbf{S}^{K+2}\times\mathbf{R}^{K+1}} \big(\mathbf{tr}\left((X+A_{\text{obj}})^TG\right) + (Y+b_{\text{obj}})^TF\big) \\ &= \begin{cases} 0 & \text{if } (X,Y)+(A_{\text{obj}},b_{\text{obj}}) = 0 \\ \infty & \text{otherwise.} \end{cases} \end{aligned}$$

Furthermore, the support function σ_Ξ of Ξ is

$$\begin{aligned} \sigma_\Xi(U,V) = \;& \sup \quad \mathbf{tr}(U^TG) + V^TF \\ & \text{s.t.} \quad (G,F) \in \Xi \\ = \;& \sup \quad \mathbf{tr}(U^TG) + V^TF \\ & \text{s.t.} \quad \mathbf{tr}(A_0^TG) + b_0^TF + c_0 \le 0 \\ & \qquad \mathbf{tr}(A_m^TG) + b_m^TF \le 0, \quad m = 1,\ldots,M \\ & \qquad H \succeq 0, \quad (H)_{kl} = \mathbf{tr}(C_{kl}^TG) + d_{kl}^TF \\ & \qquad (G,F) \in \mathbf{S}_+^{K+2} \times \mathbf{R}^{K+1} \\ \le \;& \inf \quad -c_0\tau \\ & \text{s.t.} \quad \tau A_0 + \textstyle\sum_{m=1}^M y_m A_m - \sum_{k,l=1}^{K+1} C_{kl}\widetilde{H}_{kl} - U \in \mathbf{S}_+^{K+2} \\ & \qquad \tau b_0 + \textstyle\sum_{m=1}^M y_m b_m - \sum_{k,l=1}^{K+1} d_{kl}\widetilde{H}_{kl} - V = 0, \end{aligned}$$

with variables $\tau \in \mathbf{R}_+$, $y \in \mathbf{R}_+^M$, and $\widetilde{H} \in \mathbf{S}_+^{K+1}$.

Weak duality gives the inequality above, so the minimization problem below upper-bounds the worst-case expected loss, which suffices for the certificate of Theorem 1 and for training on an upper bound. Equality holds if Ξ contains a point with $G \succ 0$ and $H \succ 0$, since the remaining constraints are affine in (G, F) and the refined Slater condition applies (Boyd & Vandenberghe, 2004, Section 5.2.3).

Combining the results above, the worst-case expected loss is upper-bounded by the optimal value of the problem

$$\begin{array}{ll} \text{minimize} & (1/N)\sum_{i=1}^N s_i \\ \text{subject to} & -c_0\tau_i - \mathbf{tr}(X_i^T\widehat{G}_i) - Y_i^T\widehat{F}_i + \lambda\varepsilon \le s_i \\ & \tau_i A_0 + \sum_{m=1}^M (y_i)_m A_m - \sum_{k,l=1}^{K+1} (\widetilde{H}_i)_{kl} C_{kl} - A_{\text{obj}} - X_i \in \mathbf{S}_+^{K+2} \\ & \tau_i b_0 + \sum_{m=1}^M (y_i)_m b_m - \sum_{k,l=1}^{K+1} (\widetilde{H}_i)_{kl} d_{kl} - b_{\text{obj}} - Y_i = 0 \\ & \|(X_i,Y_i)\| \le \lambda, \qquad\qquad i=1,\ldots,N, \end{array}$$

with variables $s_i \in \mathbf{R}$, $\tau_i \in \mathbf{R}_+$, $y_i \in \mathbf{R}_+^M$, $X_i \in \mathbf{S}^{K+2}$, $Y_i \in \mathbf{R}^{K+1}$, $\widetilde{H}_i \in \mathbf{S}_+^{K+1}$, and $\lambda \in \mathbf{R}_+$, and equality under the Slater condition. ∎

Remark 6 (Slater point for convex quadratic functions). The condition of the lemma holds for $\mathcal{Q}_{\mu,L}$ with $\mu < L$ and $d \ge K+2$ whenever $\theta_k^k \ne 0$ for every k in Assumption 1. Take $Q = \mathbf{diag}(\lambda_1, \ldots, \lambda_d)$ with $\mu < \lambda_i < L$ and distinct $\lambda_1, \ldots, \lambda_{K+2}$, and take x^0 with nonzero first $K+2$ entries and zero remaining entries, so that $x^\star = 0$. Then $x^k = p_k(Q)x^0$ for a polynomial p_k of degree exactly k, and $g^k = Qp_k(Q)x^0$ has degree exactly $k+1$. The first $K+2$

rows of $P = (x^0, g^0, \ldots, g^K)$ are therefore the product of the diagonal matrix of the nonzero entries of x^0, a Vandermonde matrix with distinct nodes, and an upper triangular matrix with nonzero diagonal, so P has full column rank and $G = P^T P \succ 0$. For the same reason $X = (x^0, \ldots, x^K)$ has full column rank, and $H = X^T(Q - \mu I)(LI - Q)X \succ 0$ since $\mu I \prec Q \prec LI$.

C Proofs for Sections 3 and 4

C.1 Proof of Theorem 1

Define the union of the support sets Ξ^θ over Θ as

$$\Xi^\Theta = \bigcup_{\theta \in \Theta} \Xi^\theta.$$

Throughout this section, $(G, F) = \mathcal{A}_\theta(z)$ is the lifted representation of Definition 1, so that $G = P^T P \in \mathbf{S}_+^{K_0}$ is the Gram matrix of the K_0 columns (8). We write $G = G(\theta)$, $F = F(\theta)$, $P = P(\theta)$, $x^i = x^i(\theta)$, $y^i = y^i(\theta)$, $w^i = w^i(\theta)$, and $s^k = s^k(\theta)$ to emphasize the θ-dependency of each term, where y^i and w^i are the gradient-query point and the proximal input of (4). The smooth part is interpolated at $y^0, \ldots, y^{K-1}$ and at x^K, and we write u^i for the i-th of these points, so that $u^i = y^i$ for $i \leq K - 1$ and $u^K = x^K$. Without loss of generality, let $x^\star = 0$. Both proofs below use the largest parameter norm

$$M_\Theta := \max_{\theta \in \Theta} \|\theta\| = \sqrt{\textstyle\sum_{j=1}^p \max\{l_j^2, u_j^2\}}, \tag{12}$$

which is finite and attained because Θ is a box by Assumption 2(ii).

We first bound the points that the algorithm produces, uniformly over $\theta \in \Theta$. These bounds depend on the structure of the algorithm, and the induction that establishes them also produces the constants used in the proof of Lemma 3. The compactness of Ξ^Θ then follows from them, together with the closedness that Assumption 2(ii) gives on its own.

Lemma 2 (Compactness of the lifted support set)**.** *Let K be fixed and suppose Assumptions 1 and 2 hold. If a proximal step is present, suppose in addition that z is a composite instance and that Assumption 3 holds. Then there exist finite constants R_K, M_h, and M_P such that*

$$\begin{aligned} \sup_{\theta \in \Theta} \max_{0 \leq i \leq K} \max \left\{ \left\|x^i(\theta) - x^\star\right\|, \left\|y^i(\theta) - x^\star\right\|, \left\|w^i(\theta) - x^\star\right\| \right\} &\leq R_K, \\ \sup_{\theta \in \Theta} \max_{0 \leq k \leq K-1} \left\|s^k(\theta)\right\| &\leq M_h, \end{aligned} \tag{13}$$

for every instance $z \in \mathcal{Z}$ with solution $x^\star$, and every column of $P(\theta)$ has norm at most M_P for every $\theta \in \Theta$. Here y^i and w^i are read as x^i for $i = K$, where the algorithm takes no further step. Consequently, the set Ξ^Θ is compact.

Proof. First of all, the set $\{(\theta, G, F) \mid \theta \in \Theta,\ (G, F) \in \Xi^\theta\}$ is closed from the fact that Θ is closed and the inequalities defining Ξ^θ are jointly continuous in (θ, G, F), since the entries of $A_m(\theta)$ and $b_m(\theta)$ are polynomial in θ. The second sequence of (4) does not affect this, since every coefficient there is a coordinate of θ, so unrolling the two sequences leaves each entry polynomial in θ and only raises its degree. Then as Θ is compact, the projection mapping $(\theta, G, F) \mapsto (G, F)$ is closed, so the image of the set above, which is Ξ^Θ, is closed. It remains to bound the set, which is where the structure of the algorithm enters.

The points the algorithm produces are bounded by a recursion on their joint radius

$$R_k = \sup_{\theta \in \Theta} \max \left\{ \max_{i \leq k} \left\|x^i(\theta) - x^\star\right\|, \max_{j \leq k-1} \left\|y^j(\theta) - x^\star\right\|, \max_{j \leq k-1} \left\|w^j(\theta) - x^\star\right\| \right\},$$

which collects the iterates, the gradient-query points, and the proximal inputs of (4). The k-th step produces $y^k(\theta)$, then $w^k(\theta)$, then $x^{k+1}(\theta)$, and we bound each in turn by an affine function of R_k. A pure gradient step is the case $h \equiv 0$, where the proximal operator is the identity, and a pure proximal step is the case of vanishing gradient coefficients. Note that we impose the initial condition

that $x^0(\theta) - x^\star$ is uniformly bounded with $\theta \in \Theta$, so that R_0 is the constant of Assumption 2(i). Also, from L-Lipschitzness of ∇f and (6), every gradient the algorithm queries obeys

$$\|\nabla f(u)\| \le \|\nabla f(u) - \nabla f(x^\star)\| + \|\nabla f(x^\star)\| \le L\,\|u - x^\star\| + R_\nabla, \tag{14}$$

so it is bounded whenever its query point is. The Cauchy–Schwarz inequality bounds the coefficient sums as

$$\left|\sum_{i=0}^{k} \theta_i^k\right| \le \sqrt{k+1}\sqrt{\sum_{i=0}^{k}(\theta_i^k)^2} \le \sqrt{K+1}\,M_\Theta, \qquad k = 0, \dots, K-1. \tag{15}$$

The subgradients already produced are controlled by R_k as well. For $i \le k-1$, the definition (5) and the lower bound $\theta^i_{\text{prox}} \ge \theta_{\min}$ of Assumption 3(i) give

$$\left\|s^i(\theta)\right\| = \frac{\left\|w^i(\theta) - x^{i+1}(\theta)\right\|}{\theta^i_{\text{prox}}} \le \frac{\left\|w^i(\theta) - x^\star\right\| + \left\|x^{i+1}(\theta) - x^\star\right\|}{\theta_{\min}} \le \frac{2R_k}{\theta_{\min}}. \tag{16}$$

For the extrapolation, we have

$$\begin{aligned}
\left\|y^k(\theta) - x^\star\right\| &\le \left\|x^k(\theta) - x^\star\right\| + \left\|\sum_{i=0}^{k-1} \phi_i^k \nabla f(y^i(\theta))\right\| + \left\|\sum_{i=0}^{k-1} \psi_i^k s^i(\theta)\right\| \\
&\le R_k + M_\Theta \sqrt{\sum_{i=0}^{k-1} \|\nabla f(y^i(\theta))\|^2} + M_\Theta \sqrt{\sum_{i=0}^{k-1} \|s^i(\theta)\|^2} \\
&\le R_k + \sqrt{K}\,M_\Theta\,(LR_k + R_\nabla) + \sqrt{K}\,M_\Theta\,\frac{2R_k}{\theta_{\min}}.
\end{aligned}$$

The second inequality is the Cauchy–Schwarz inequality together with $\sqrt{\sum_i (\phi_i^k)^2} \le \|\theta\| \le M_\Theta$ and the same bound on the ψ_i^k, and the third uses (14) and (16) over the at most K terms of each sum. Writing the result as

$$\left\|y^k(\theta) - x^\star\right\| \le aR_k + b, \qquad a = 1 + \sqrt{K}\,M_\Theta\left(L + \frac{2}{\theta_{\min}}\right), \qquad b = \sqrt{K}\,M_\Theta R_\nabla, \tag{17}$$

we have $a \ge 1$ and $b \ge 0$, so (17) bounds $\|y^i(\theta) - x^\star\|$ for every $i \le k$. A single-sequence method has $y^k = x^k$, where the two extrapolation sums are absent and (17) holds with $a = 1$ and $b = 0$.

For the gradient part, we have

$$\begin{aligned}
&\left\|w^k(\theta) - x^\star\right\| \\
&= \left\|\left(y^k(\theta) - x^\star\right) - \sum_{i=0}^{k} \theta_i^k \left(\nabla f(y^i(\theta)) - \nabla f(x^\star)\right) - \sum_{i=0}^{k} \theta_i^k \nabla f(x^\star)\right\| \\
&\le \left\|y^k(\theta) - x^\star\right\| + \left\|\sum_{i=0}^{k} \theta_i^k \left(\nabla f(y^i(\theta)) - \nabla f(x^\star)\right)\right\| + \left|\sum_{i=0}^{k} \theta_i^k\right| \|\nabla f(x^\star)\| \\
&\le \left\|y^k(\theta) - x^\star\right\| + \sqrt{\sum_{i=0}^{k}(\theta_i^k)^2}\sqrt{\sum_{i=0}^{k} \|\nabla f(y^i(\theta)) - \nabla f(x^\star)\|^2} + \left|\sum_{i=0}^{k} \theta_i^k\right| \|\nabla f(x^\star)\| \\
&\le \left\|y^k(\theta) - x^\star\right\| + M_\Theta \sqrt{\sum_{i=0}^{k} \|\nabla f(y^i(\theta)) - \nabla f(x^\star)\|^2} + \left|\sum_{i=0}^{k} \theta_i^k\right| \|\nabla f(x^\star)\| \\
&\le \left\|y^k(\theta) - x^\star\right\| + L\,M_\Theta \sqrt{\sum_{i=0}^{k} \|y^i(\theta) - x^\star\|^2} + \left|\sum_{i=0}^{k} \theta_i^k\right| \|\nabla f(x^\star)\|\,.
\end{aligned}$$

The third inequality uses $\sqrt{\sum_{i=0}^{k}(\theta_i^k)^2} \le \|\theta\| \le M_\Theta$, and the last uses L-Lipschitzness of ∇f over the query points. Bounding every $\|y^i(\theta) - x^\star\|$ by (17) and the coefficient sum by (15) gives

$$\left\|w^k(\theta) - x^\star\right\| \le c\,(aR_k + b) + d, \qquad c = 1 + \sqrt{K+1}\,L\,M_\Theta, \qquad d = \sqrt{K+1}\,M_\Theta R_\nabla. \tag{18}$$

For the proximal part,

$$\begin{aligned}\left\|x^{k+1}(\theta) - x^\star\right\| &= \left\|\left(\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(w^k(\theta)) - \mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star)\right) + \left(\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star) - x^\star\right)\right\| \\ &\le \left\|\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(w^k(\theta)) - \mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star)\right\| + \left\|\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star) - x^\star\right\| \\ &\le \left\|w^k(\theta) - x^\star\right\| + \left\|\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star) - x^\star\right\|,\end{aligned}$$

where the inequality comes from $\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}$ being a firmly nonexpansive operator. The proximal part is closed by $\|\mathbf{prox}_{\theta^k_{\mathrm{prox}}h}(x^\star) - x^\star\| \le R_{\mathrm{prox}}$ from (7), so that $\|x^{k+1}(\theta) - x^\star\| \le \|w^k(\theta) - x^\star\| + R_{\mathrm{prox}}$. Chaining this with (18) bounds each of the three points the k-th step produces by an affine function of R_k, and the iterate just displayed is the largest of the three. This gives the affine recursion

$$R_{k+1} \le \nu R_k + \omega_R, \qquad \nu = c\,a, \qquad \omega_R = c\,b + d + R_{\mathrm{prox}}, \tag{19}$$

with a, b from (17) and c, d from (18), which gives $R_K \le \nu^K R_0 + \omega_R(\nu^K - 1)/(\nu - 1)$. The radius grows exponentially in the horizon K, as one expects of a composition of K Lipschitz maps, and is finite for every fixed K. A single-sequence method has $a = 1$ and $b = 0$, where the recursion reduces to $\nu = c$ and $\omega_R = d + R_{\mathrm{prox}}$. The second sequence therefore costs a constant factor in the radius, and that factor degrades as $\theta_{\min} \to 0$ through the a of (17).

The subgradient columns $s^k(\theta)$ of (8) are bounded as well. Taking $k = K - 1$ in (16) and bounding R_{K-1} by R_K gives

$$M_h = \sup_{\theta\in\Theta} \max_k \left\|s^k(\theta)\right\| \le \frac{2R_K}{\theta_{\min}} < \infty. \tag{20}$$

Every column of $P(\theta)$ is therefore bounded, with

$$M_P \le \max\left\{R_0,\ LR_K + R_\nabla,\ M_h\right\},$$

the middle entry coming from (14) together with the fact that R_K bounds every gradient-query point. The optimality column needs no entry of its own, since $\|s^\star\| = \|\nabla f(x^\star)\| \le R_\nabla$. In particular, the column bound M_P is finite. The entries of F are bounded as well. Convexity gives $|f(u^i(\theta)) - f(x^\star)| \le \max\{\|\nabla f(u^i(\theta))\|, \|\nabla f(x^\star)\|\}\,\|u^i(\theta) - x^\star\| \le M_P R_K$, and likewise $|h(x^k(\theta)) - h(x^\star)| \le M_P R_K$ with the subgradients $s^{k-1}(\theta)$ and $s^\star$. Hence $\|F(\theta)\| \le \sqrt{K_0 - 1}\,M_P R_K$. Since $G = P^T P$ with P stacking these bounded columns, the lifted outputs $\mathcal{A}_\theta(z)$ are bounded over $\theta \in \Theta$ and $z \in \mathcal{Z}$. By exactness (Section 3.1), every element of Ξ^Θ is such an output, so Ξ^Θ is bounded. Being closed and bounded, the set Ξ^Θ is compact. ■

We now show the Lipschitz property of the algorithm mapping $\theta \mapsto \mathcal{A}_\theta$.

Lemma 3. *Suppose the hypotheses of Lemma 2 hold. Then the mapping $\theta \mapsto \mathcal{A}_\theta$ is Lipschitz, i.e., there exists $\bar{L} > 0$ such that*

$$\|\mathcal{A}_{\theta_1}(z) - \mathcal{A}_{\theta_2}(z)\| \le \bar{L}\,\|\theta_1 - \theta_2\|,$$

for all $\theta_1, \theta_2 \in \Theta$ and $z \in \mathcal{Z}$, where the norm on the left is the pair norm $\|(G,F)\| = \sqrt{\|G\|_F^2 + \|F\|^2}$ of Section 3.1 applied to the lifted representation $\mathcal{A}_\theta(z) = (G,F)$ of Definition 1, whose matrix $P(\theta)$ lies in $\mathbf{R}^{d\times K_0}$.

Proof. We use the notation introduced before Lemma 2, together with its constants R_K, M_h, and M_P of (13). Since

$$G(\theta_1) - G(\theta_2) = P_1^T P_1 - P_2^T P_2 = \frac{1}{2}\big((P_1 - P_2)^T(P_1 + P_2) + (P_1 + P_2)^T(P_1 - P_2)\big),$$

we have

$$\|G(\theta_1) - G(\theta_2)\|_F$$

$$\leq \|P(\theta_1) - P(\theta_2)\|_F \, \|P(\theta_1) + P(\theta_2)\|_2$$
$$\leq 2\sqrt{K_0} \, \|P(\theta_1) - P(\theta_2)\|_F \max_{\theta \in \{\theta_1, \theta_2\}} \max_{i,k} \left\{ \left\|x^0 - x^\star\right\|, \|s^\star\|, \left\|\nabla f(u^i(\theta))\right\|, \left\|s^k(\theta)\right\| \right\}.$$

The last factor in the bound above is a column norm of $P(\theta_1)$ or $P(\theta_2)$, hence at most the column bound M_P of Lemma 2.

Regarding the F-vector, its f-entries sit at the points u^i by (9). The convexity of f implies

$$\nabla f(u^i(\theta_2))^T \big(u^i(\theta_1) - u^i(\theta_2)\big) \leq f(u^i(\theta_1)) - f(u^i(\theta_2)) \leq \nabla f(u^i(\theta_1))^T \big(u^i(\theta_1) - u^i(\theta_2)\big),$$

so the Cauchy–Schwarz inequality gives

$$\left|f(u^i(\theta_1)) - f(u^i(\theta_2))\right| \leq \max_{\theta \in \{\theta_1, \theta_2\}} \left\|\nabla f(u^i(\theta))\right\| \, \left\|u^i(\theta_1) - u^i(\theta_2)\right\|.$$

The two components are therefore controlled by the same two quantities, the column bound M_P and the distance between the points at which the two parameters evaluate the oracles.

It remains to show that $\theta \mapsto P(\theta)$ is Lipschitz continuous. We use induction to prove that, if $\|x^k(\theta_1) - x^k(\theta_2)\| \leq L_k \|\theta_1 - \theta_2\|$ for all $\theta_1, \theta_2 \in \Theta$ and any choice of f and h, then there exists $L_{k+1} > 0$ such that $\|x^{k+1}(\theta_1) - x^{k+1}(\theta_2)\| \leq L_{k+1} \|\theta_1 - \theta_2\|$ for all $\theta_1, \theta_2 \in \Theta$ and any f and h as well. The induction starts at $L_0 = 0$, since the initial iterate x^0 belongs to the instance z and carries no θ-dependency.

The k-th step of (4) produces three points and one subgradient column, so the induction carries four constants,

$$\left\|y^k(\theta_1) - y^k(\theta_2)\right\| \leq L_{y,k} \, \|\theta_1 - \theta_2\|, \qquad \left\|w^k(\theta_1) - w^k(\theta_2)\right\| \leq L_{w,k} \, \|\theta_1 - \theta_2\|,$$

together with L_{k+1} for the iterate and $L_{s,k}$ for the column s^k. Step k uses only L_k and the constants $L_{y,i}$, $L_{s,i}$ of the earlier steps $i \leq k-1$, so the four are well defined by simultaneous induction. All four are nondecreasing in k, which we use to replace sums of squares by their largest term.

For the extrapolation, splitting each sum into a coefficient difference against a bounded column and a bounded coefficient against a column difference gives

$$\begin{aligned}
&\left\|y^k(\theta_1) - y^k(\theta_2)\right\| \\
&\leq \left\|x^k(\theta_1) - x^k(\theta_2)\right\| + \left\|\sum_{i=0}^{k-1} \big((\phi_1)_i^k - (\phi_2)_i^k\big) \nabla f(y^i(\theta_1))\right\| \\
&\quad + \left\|\sum_{i=0}^{k-1} (\phi_2)_i^k \big(\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\big)\right\| + \left\|\sum_{i=0}^{k-1} \big((\psi_1)_i^k - (\psi_2)_i^k\big) s^i(\theta_1)\right\| \\
&\quad + \left\|\sum_{i=0}^{k-1} (\psi_2)_i^k \big(s^i(\theta_1) - s^i(\theta_2)\big)\right\| \\
&\leq \Big(L_k + \sqrt{K}\,(M_P + M_h) + \sqrt{K}\, M_\Theta \,(L\, L_{y,k-1} + L_{s,k-1})\Big) \, \|\theta_1 - \theta_2\|.
\end{aligned}$$

Here $(\phi_1)_i^k$ is the coordinate ϕ_i^k of θ_1, following the convention used for $(\theta_1)_i^k$ below. The coefficient differences are bounded by $\|\theta_1 - \theta_2\|$ and the coefficients themselves by M_Θ, both by the Cauchy–Schwarz inequality, while $\|\nabla f(y^i(\theta_1))\| \leq M_P$ and $\|s^i(\theta_1)\| \leq M_h$ hold by (13), since both are columns of $P(\theta_1)$. The column differences use L-Lipschitzness of ∇f with the induction hypothesis $L_{y,i}$, and the induction hypothesis $L_{s,i}$ directly. This defines $L_{y,k}$ as the bracket above, with $L_{y,0} = L_0 = 0$ because the two sums are empty. A single-sequence method has $L_{y,k} = L_k$.

For the gradient part, we have

$$\begin{aligned}
&w^k(\theta_1) - w^k(\theta_2) \\
&= y^k(\theta_1) - y^k(\theta_2) - \sum_{i=0}^{k} \Big((\theta_1)_i^k \nabla f(y^i(\theta_1)) - (\theta_2)_i^k \nabla f(y^i(\theta_2))\Big)
\end{aligned}$$

$$= y^k(\theta_1) - y^k(\theta_2) - \sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)\nabla f(y^i(\theta_1)) - \sum_{i=0}^{k}(\theta_2)_i^k\big(\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\big)$$

$$= y^k(\theta_1) - y^k(\theta_2) - \sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)\big(\nabla f(y^i(\theta_1)) - \nabla f(x^\star)\big)$$

$$- \Big(\sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)\Big)\nabla f(x^\star) - \sum_{i=0}^{k}(\theta_2)_i^k\big(\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\big).$$

Therefore,

$$\begin{aligned}
\big\|w^k(\theta_1) - w^k(\theta_2)\big\| &\le \big\|y^k(\theta_1) - y^k(\theta_2)\big\| \\
&\quad + \left\|\sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)\big(\nabla f(y^i(\theta_1)) - \nabla f(x^\star)\big)\right\| \\
&\quad + \left\|\sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)\nabla f(x^\star)\right\| \\
&\quad + \left\|\sum_{i=0}^{k}(\theta_2)_i^k\big(\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\big)\right\| \\
&\le \big\|y^k(\theta_1) - y^k(\theta_2)\big\| \\
&\quad + \sqrt{\sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)^2}\sqrt{\sum_{i=0}^{k} \big\|\nabla f(y^i(\theta_1)) - \nabla f(x^\star)\big\|^2} \\
&\quad + \sqrt{K+1}\sqrt{\sum_{i=0}^{k} \big((\theta_1)_i^k - (\theta_2)_i^k\big)^2}\,\big\|\nabla f(x^\star)\big\| \\
&\quad + \sqrt{\sum_{i=0}^{k}((\theta_2)_i^k)^2}\sqrt{\sum_{i=0}^{k} \big\|\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\big\|^2} \\
&\le \big\|y^k(\theta_1) - y^k(\theta_2)\big\| + L\,\|\theta_1 - \theta_2\|\sqrt{\sum_{i=0}^{k} \big\|y^i(\theta_1) - x^\star\big\|^2} \\
&\quad + \sqrt{K+1}\,\|\theta_1 - \theta_2\|\|\nabla f(x^\star)\| + L\,M_\Theta\sqrt{\sum_{i=0}^{k} \big\|y^i(\theta_1) - y^i(\theta_2)\big\|^2}.
\end{aligned}$$

The third term involves the sum of the coefficient differences rather than their norm, so it costs an extra factor $\sqrt{K+1}$ by the same Cauchy–Schwarz bound as (15).

For the proximal part, we have

$$\begin{aligned}
&\big\|x^{k+1}(\theta_1) - x^{k+1}(\theta_2)\big\| \\
&= \Big\|\mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_1)) - \mathbf{prox}_{(\theta_2)^k_{\mathrm{prox}}h}(w^k(\theta_2))\Big\| \\
&\le \Big\|\mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_1)) - \mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_2))\Big\| \\
&\quad + \Big\|\mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_2)) - \mathbf{prox}_{(\theta_2)^k_{\mathrm{prox}}h}(w^k(\theta_2))\Big\| \\
&\le \big\|w^k(\theta_1) - w^k(\theta_2)\big\| + \Big\|\mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_2)) - \mathbf{prox}_{(\theta_2)^k_{\mathrm{prox}}h}(w^k(\theta_2))\Big\|.
\end{aligned}$$

The second inequality uses nonexpansiveness of the proximal operator. For the remaining term, we have

$$\Big\|\mathbf{prox}_{(\theta_1)^k_{\mathrm{prox}}h}(w^k(\theta_2)) - \mathbf{prox}_{(\theta_2)^k_{\mathrm{prox}}h}(w^k(\theta_2))\Big\|$$

$$\leq |(\theta_1)^k_{\text{prox}} - (\theta_2)^k_{\text{prox}}| \left\| \frac{w^k(\theta_2) - \mathbf{prox}_{(\theta_2)^k_{\text{prox}} h}(w^k(\theta_2))}{(\theta_2)^k_{\text{prox}}} \right\|.$$

The last factor is the norm of the subgradient column $s^k(\theta_2)$ of (8), at most M_h by (13), and the coordinate difference satisfies $|(\theta_1)^k_{\text{prox}} - (\theta_2)^k_{\text{prox}}| \leq \|\theta_1 - \theta_2\|$. Therefore,

$$\left\|x^{k+1}(\theta_1) - x^{k+1}(\theta_2)\right\| \leq \left\|w^k(\theta_1) - w^k(\theta_2)\right\| + M_h \|\theta_1 - \theta_2\|.$$

We may conclude that there exists $L_{k+1} > 0$ such that, for all $\theta_1, \theta_2 \in \Theta$ and functions f, h,

$$\left\|x^{k+1}(\theta_1) - x^{k+1}(\theta_2)\right\| \leq L_{k+1} \|\theta_1 - \theta_2\|,$$

given $\|x^k(\theta_1) - x^k(\theta_2)\| \leq L_k \|\theta_1 - \theta_2\|$. Collecting the three displays gives the per-step recursion

$$L_{w,k} \leq c\, L_{y,k} + \sqrt{K+1}\,(L\, R_K + R_\nabla), \qquad L_{k+1} \leq L_{w,k} + M_h, \tag{21}$$

with the multiplier $c = 1 + \sqrt{K+1}\, L\, M_\Theta$ of (18). The gradient part uses $\|y^i(\theta_1) - x^\star\| \leq R_K$ from (13) and $\sqrt{\sum_{i=0}^k L_{y,i}^2} \leq \sqrt{K+1}\, L_{y,k}$, the latter from the sequence $(L_{y,i})$ being nondecreasing in i. The proximal part adds M_h on top of it, by the same argument as in the boundedness induction in the proof of Lemma 2. Together with the extrapolation constant $L_{y,k}$ above and the column constant $L_{s,k}$ below, this closes the induction, and every constant is finite for a fixed horizon K. A single-sequence method has $L_{y,k} = L_k$, where (21) collapses to the single recursion $L_{k+1} \leq c\, L_k + \sqrt{K+1}(LR_K + R_\nabla) + M_h$.

We now prove the Lipschitz continuity of the mapping $\theta \mapsto P(\theta)$ regarding (8). This reduces to proving that each column is Lipschitz continuous, since the squared Frobenius norm of a matrix is bounded by the sum of the squares of its columns. First of all, columns $x^0 - x^\star$ and $s^\star$ are θ-independent. Regarding gradient columns, the first K sit at the gradient-query points and the last at x^K, so

$$\left\|\nabla f(y^i(\theta_1)) - \nabla f(y^i(\theta_2))\right\| \leq L \left\|y^i(\theta_1) - y^i(\theta_2)\right\| \leq L L_{y,i} \|\theta_1 - \theta_2\|, \qquad i = 0, \ldots, K-1,$$

and the same bound with L_K in place of $L_{y,i}$ for the column at x^K. We write $L_{y,K} = L_K$ so that the two cases read alike. Now, regarding the subgradient columns related to proximal operators, we have

$$s^k(\theta_1) - s^k(\theta_2) = \frac{\left(w^k(\theta_1) - x^{k+1}(\theta_1)\right) - \left(w^k(\theta_2) - x^{k+1}(\theta_2)\right)}{(\theta_1)^k_{\text{prox}}} + s^k(\theta_2)\, \frac{(\theta_2)^k_{\text{prox}} - (\theta_1)^k_{\text{prox}}}{(\theta_1)^k_{\text{prox}}}.$$

Bounding the two denominators by $\theta_{\min}$ and the second numerator by $M_h \|\theta_1 - \theta_2\|$ gives

$$\left\|s^k(\theta_1) - s^k(\theta_2)\right\| \leq L_{s,k} \|\theta_1 - \theta_2\|, \qquad L_{s,k} = \frac{L_{w,k} + L_{k+1} + M_h}{\theta_{\min}}, \qquad k = 0, \ldots, K-1,$$

which is the fourth constant of the induction, with $L_{w,k}$ from (21). This is the only place where $\theta_{\min}$ enters the proof. Summing the squares of the column bounds gives

$$\|P(\theta_1) - P(\theta_2)\|_F \leq \bar{L}_P \|\theta_1 - \theta_2\|, \qquad \bar{L}_P = \sqrt{L^2 \textstyle\sum_{i=0}^K L_{y,i}^2 + \sum_{k=0}^{K-1} L_{s,k}^2}, \tag{22}$$

the second term vanishes for the smooth case.

The F-component also uses a similar induction step. The gradients $\nabla f(u^i(\theta))$ are columns of $P(\theta)$ and are therefore bounded by M_P, and the induction above gives $\|u^i(\theta_1) - u^i(\theta_2)\| \leq L_{y,i} \|\theta_1 - \theta_2\|$, with the convention $L_{y,K} = L_K$ of the column bounds. The convexity bound at the beginning then writes as

$$\left|f(u^i(\theta_1)) - f(u^i(\theta_2))\right| \leq M_P L_{y,i} \|\theta_1 - \theta_2\|, \qquad i = 0, \ldots, K.$$

Similarly, using (8) and convexity of h, whose entries sit at the iterates,

$$\left|h(x^i(\theta_1)) - h(x^i(\theta_2))\right| \leq M_h L_i \|\theta_1 - \theta_2\|, \qquad i = 1, \ldots, K.$$

Summing the squares over all entries and using $M_h \leq M_P$ together with $L_i \leq L_{y,i}$ gives

$$\|F(\theta_1) - F(\theta_2)\| \leq \sqrt{2}\, M_P \sqrt{\textstyle\sum_{i=0}^K L_{y,i}^2}\ \|\theta_1 - \theta_2\|.$$

These result in

$$\|\mathcal{A}_{\theta_1}(z) - \mathcal{A}_{\theta_2}(z)\| = \sqrt{\|G(\theta_1) - G(\theta_2)\|_F^2 + \|F(\theta_1) - F(\theta_2)\|^2} \\ \leq M_P \sqrt{4K_0 \bar{L}_P^2 + 2\sum_{i=0}^{K} L_{y,i}^2}\ \|\theta_1 - \theta_2\|\,.$$

The gradient columns alone make $\bar{L}_P^2 \geq L^2 \sum_{i=0}^{K} L_{y,i}^2$ in (22), so the claim holds with

$$\bar{L} = M_P \bar{L}_P \sqrt{4K_0 + 2L^{-2}}, \tag{23}$$

which is finite for every fixed horizon K. ■

The lemma covers logistic regression, which takes no proximal step and uses the second sequence, and LASSO, which is single-sequence with a real-valued h. The lower bound $\theta_{\min}$ is a live hypothesis for LASSO, since the subgradient column constant $L_{s,k}$ divides by it, and Remark 3 shows that this dependence is unavoidable. The PDHG iteration of Section 5.3 is not of the form (4), and Remark 5 extends both lemmas to it.

The constant $\bar{L}$ enters the finite-sample radius $\bar{\varepsilon}(\beta)$ of Theorem 1 only through its finiteness. A larger $\bar{L}$ therefore enlarges $\bar{\varepsilon}(\beta)$ by a constant factor, given $\theta_{\min}$, and leaves the N-dependence of Theorems 1 and 2 unchanged.

We follow the measure concentration result of (Fournier & Guillin, 2015) to derive the radius ε guaranteeing that DRO loss upper-bounds true risk with probability at least $1-\beta$, as in (Park et al., 2025). Choose $\varepsilon = \varepsilon(\beta, \theta)$ as follows:

$$\varepsilon(\beta,\theta) = \begin{cases} \left(\frac{\log(c_1\beta^{-1})}{c_2 N}\right)^{1/q} & N \geq \frac{\log(c_1\beta^{-1})}{c_2} \\ \left(\frac{\log(c_1\beta^{-1})}{c_2 N}\right)^{1/a} & N < \frac{\log(c_1\beta^{-1})}{c_2}, \end{cases} \tag{24}$$

where $c_1 > 0$ and $c_2 > 0$ are constants that depend on the pushforward $\mathbf{P}^\theta$ of Section B only through q and an exponential moment of $\|(G,F)\|$, which is bounded uniformly in θ because $\mathbf{P}^\theta$ is supported in the compact set Ξ^Θ of Lemma 2. Hence $\varepsilon(\beta,\theta) = \varepsilon(\beta)$ does not depend on θ. Here $q = K_0(K_0+1)/2 + (K_0-1)$ is the dimension of the lifted space $\mathbf{S}^{K_0} \times \mathbf{R}^{K_0-1}$, which is $(K+2)(K+3)/2 + (K+1)$ for the smooth lifting of Section 3.1, and a is the light-tail exponent of the concentration result. From this, we get the following measure concentration inequality for each $\theta \in \Theta$:

Lemma 4. *Let $\beta \in (0,1)$. For each $\theta \in \Theta$, there exists $\varepsilon(\beta,\theta) > 0$ as in (24) such that*

$$\mathbf{P}^N\left(\mathbf{P}^\theta \in \mathcal{U}_{\varepsilon(\beta,\theta)}\left(\widehat{\mathbf{P}}_N^\theta\right)\right) \geq 1-\beta.$$

Proof. The proof follows directly from (Fournier & Guillin, 2015, Theorem 2). ■

We now prove Theorem 1 using the covering number argument on compact Θ to extract a θ-independent radius ε of the Wasserstein ambiguity set.

Proof of Theorem 1. For any $\delta > 0$, there exists a finite covering C_δ of the compact set Θ in the sense that $|C_\delta| < \infty$ and for any $\theta \in \Theta$, there exists $\theta^{(j)} \in C_\delta$ such that $\|\theta - \theta^{(j)}\| \leq \delta$. According to Lemma 4, the radius $\varepsilon(\beta/|C_\delta|) > 0$ satisfies

$$\mathbf{P}^N\left(\mathbf{P}^{\theta^{(j)}} \in \mathcal{U}_{\varepsilon(\beta/|C_\delta|)}\left(\widehat{\mathbf{P}}_N^{\theta^{(j)}}\right)\right) \geq 1-\beta/|C_\delta|, \qquad \forall \theta^{(j)} \in C_\delta.$$

From the union bound, we have

$$\mathbf{P}^N\left(\mathbf{P}^{\theta^{(j)}} \in \mathcal{U}_{\varepsilon(\beta/|C_\delta|)}\left(\widehat{\mathbf{P}}_N^{\theta^{(j)}}\right),\ \forall \theta^{(j)} \in C_\delta\right) \geq 1-\beta.$$

Now, define

$$\bar{\varepsilon}(\beta,\delta) = \varepsilon(\beta/|C_\delta|) + 2\bar{L}\delta.$$

For any $\theta \in \Theta$, there exists $\theta^{(j)} \in C_\delta$ such that $\|\theta - \theta^{(j)}\| \leq \delta$. From Lemma 3, we have

$$W_1\left(\widehat{\mathbf{P}}_N^\theta, \widehat{\mathbf{P}}_N^{\theta^{(j)}}\right) \leq \bar{L}\left\|\theta - \theta^{(j)}\right\| \leq \bar{L}\delta,$$

and

$$W_1\left(\mathbf{P}^\theta, \mathbf{P}^{\theta^{(j)}}\right) \leq \bar{L}\left\|\theta - \theta^{(j)}\right\| \leq \bar{L}\delta.$$

From the triangle inequality of the Wasserstein distance,

$$\begin{aligned} &W_1\left(\widehat{\mathbf{P}}_N^\theta, \mathbf{P}^\theta\right) \\ &\leq W_1\left(\widehat{\mathbf{P}}_N^\theta, \widehat{\mathbf{P}}_N^{\theta^{(j)}}\right) + W_1\left(\widehat{\mathbf{P}}_N^{\theta^{(j)}}, \mathbf{P}^{\theta^{(j)}}\right) + W_1\left(\mathbf{P}^{\theta^{(j)}}, \mathbf{P}^\theta\right) \\ &\leq \bar{L}\delta + \varepsilon(\beta/|C_\delta|) + \bar{L}\delta = \overline{\varepsilon}(\beta, \delta). \end{aligned}$$

Therefore, we get

$$\mathbf{P}^N\left(\mathbf{P}^\theta \in \mathcal{U}_{\overline{\varepsilon}(\beta,\delta)}(\widehat{\mathbf{P}}_N^\theta),\ \forall \theta \in \Theta\right) \geq 1 - \beta.$$

Finally, for any $\delta > 0$, we get

$$\mathbf{P}^N\left(\mathbf{P}^\theta \in \mathcal{U}_{\overline{\varepsilon}(\beta)}(\widehat{\mathbf{P}}_N^\theta),\ \forall \theta \in \Theta\right) \geq \mathbf{P}^N\left(\mathbf{P}^{\theta^{(j)}} \in \mathcal{U}_{\varepsilon(\beta/|C_\delta|)}(\widehat{\mathbf{P}}_N^{\theta^{(j)}}),\ \forall \theta^{(j)} \in C_\delta\right) \geq 1 - \beta,$$

for $\overline{\varepsilon}(\beta, \delta) = \varepsilon(\beta/|C_\delta|) + 2\bar{L}\delta$. The radius of the theorem is $\overline{\varepsilon}(\beta) := \overline{\varepsilon}(\beta, \delta)$ for a single fixed $\delta > 0$, chosen below.

From (Vershynin, 2018, Corollary 4.2.13), the covering number $|C_\delta|$ of Θ is upper bounded as

$$|C_\delta| \leq B_\delta := \left(\frac{2\ \mathbf{diam}(\Theta)}{\delta} + 1\right)^{\dim(\Theta)}.$$

By (24) the map $\alpha \mapsto \varepsilon(\alpha)$ is decreasing, and $\beta/|C_\delta| \geq \beta/B_\delta$, so

$$\overline{\varepsilon}(\beta, \delta) \leq \varepsilon\left(\beta/B_\delta\right) + 2\bar{L}\delta.$$

This holds for every fixed $\delta > 0$, so any choice of δ gives a valid radius, and the δ minimizing the right-hand side gives the tightest one. Substituting the first expression of (24), valid for N large enough, and absorbing c_1, c_2, and the factor $\sqrt{2}$ between the pair norm and the Euclidean norm on $\mathbf{R}^q$,

$$\overline{\varepsilon}(\beta, \delta) \lesssim \left(\frac{\log(1/\beta) + \dim(\Theta)\log(2\ \mathbf{diam}(\Theta)/\delta + 1)}{N}\right)^{1/q} + 2\bar{L}\delta,$$

where q is the dimension of (G, F). Naively choosing $\delta = 1/N$ yields

$$\overline{\varepsilon}(\beta, 1/N) \lesssim \left(\frac{\log(1/\beta) + \dim(\Theta)\log\left(1 + 2N\ \mathbf{diam}(\Theta)\right)}{N}\right)^{1/q} + \frac{2\bar{L}}{N}.$$

The first branch of (24) applies whenever $c_2 N \geq \log(c_1 B_{1/N}/\beta)$, which holds for all sufficiently large N because the right-hand side grows only logarithmically in N. ■

The rate $N^{-1/q}$ with $q = O(K^2)$ is conservative for large K and moderate N. Sharper rates can depend on the intrinsic dimension of the lifted distribution (Weed & Bach, 2019), but establishing such rates uniformly over Θ is beyond the scope of this paper.

C.2 Well-posedness of (DR-L2O)

We now prove Proposition 2, *i.e.*, the learning problem (DR-L2O) attains its minimum over Θ. The parameter set Θ is compact by Assumption 2(*ii*), so the only requirement is continuity of the DRO risk $\theta \mapsto \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$.

Write $Z = (G, F)$ for a lifted pair and $\widehat{Z}_i(\theta) = \mathcal{A}_\theta(\hat{z}_i)$ for the i-th lifted sample. Given $\varepsilon > 0$ and $\theta \in \Theta$, the feasible set of (DRO-PEP-P) is defined as

$$C(\theta) = \left\{Z = (Z_1, \ldots, Z_N) \in (\Xi^\theta)^N \;\middle|\; (1/N)\textstyle\sum_{i=1}^N \left\|Z_i - \widehat{Z}_i(\theta)\right\| \leq \varepsilon\right\}, \tag{25}$$

and Proposition 1 identifies the DRO risk with the following optimal value:

$$\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) = \max_{Z \in C(\theta)} (1/N)\textstyle\sum_{i=1}^N \ell(Z_i). \tag{26}$$

Continuity of $\theta \mapsto \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ is not a trivial fact. The objective of (26) is linear in Z and free of θ, but the feasible set $C(\theta)$ moves with θ through both the support set Ξ^θ and the lifted samples $\widehat{Z}_i(\theta)$. Therefore, when the parameter θ changes, the feasible set $C(\theta)$ can jump, and the optimal value may do so. We therefore consider C as a set-valued mapping and appeal to Berge's maximum theorem (Berge, 1963, Chapter 6); this gives continuity of the optimal value whenever the objective is continuous and the mapping is nonempty-valued, compact-valued, upper hemicontinuous, and lower hemicontinuous.

Proposition 4 (Well-posedness of (DR-L2O))**.** *Fix $\varepsilon > 0$. Suppose the hypotheses of Lemma 2 hold. Then the map $\theta \mapsto \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ is continuous on Θ, and* (DR-L2O) *attains its minimum over Θ.*

Proof. The objective of (26) is linear in Z, hence continuous, and does not depend on θ. We now prove the required properties of the mapping $\theta \mapsto C(\theta)$ for Berge's maximum theorem to hold.

Let $\theta \in \Theta$. The lifted samples satisfy $\widehat{Z}_i(\theta) \in \Xi^\theta$, so the tuple $(\widehat{Z}_1(\theta), \ldots, \widehat{Z}_N(\theta))$ lies in $C(\theta)$. Each Ξ^θ is closed and $Z \mapsto (1/N)\sum_{i=1}^N \|Z_i - \widehat{Z}_i(\theta)\|$ is continuous, so $C(\theta)$ is closed. Furthermore, $C(\theta)$ is contained in $(\Xi^\Theta)^N$, which is compact by Lemma 2; therefore, it is a compact set.

We now show the upper hemicontinuity of C. All values of C lie in the fixed compact set $(\Xi^\Theta)^N$, so it suffices to show that any $\theta_n \to \bar{\theta}$ in Θ and any $Z^n \in C(\theta_n)$ admit a subsequence of (Z^n) converging to a point of $C(\bar{\theta})$. First, compactness gives a subsequence $Z^{n_k} \to \bar{Z}$ with $Z_i^{n_k} \to \bar{Z}_i$ for each i. Second, the graph of $\theta \mapsto \Xi^\theta$ is closed by the proof of Lemma 2, so each limit $\bar{Z}_i$ lies in $\Xi^{\bar{\theta}}$. Finally, since $\theta \mapsto \widehat{Z}_i(\theta) = \mathcal{A}_\theta(\hat{z}_i)$ is Lipschitz by Lemma 3, the mapping $(\theta, Z) \mapsto (1/N)\sum_{i=1}^N \|Z_i - \widehat{Z}_i(\theta)\|$ is jointly continuous, so the Wasserstein inequality is preserved in the limit. Therefore, $\bar{Z}$ lies in $C(\bar{\theta})$.

It remains to show the lower hemicontinuity of C. Let $\theta_n \to \bar{\theta}$ in Θ and $\bar{Z} \in C(\bar{\theta})$. By the exactness of the interpolation conditions (Section 3.1), pick $z_i \in \mathcal{Z}$ with $\bar{Z}_i = \mathcal{A}_{\bar{\theta}}(z_i)$ and define $Z_i^n = \mathcal{A}_{\theta_n}(z_i) \in \Xi^{\theta_n}$. Lemma 3 gives $Z_i^n \to \bar{Z}_i$ and $\widehat{Z}_i(\theta_n) \to \widehat{Z}_i(\bar{\theta})$, so

$$d_n := (1/N)\textstyle\sum_{i=1}^N \left\| Z_i^n - \widehat{Z}_i(\theta_n) \right\| \;\longrightarrow\; \bar{d} = (1/N)\textstyle\sum_{i=1}^N \left\| \bar{Z}_i - \widehat{Z}_i(\bar{\theta}) \right\| \le \varepsilon.$$

The tuple $Z^n = (Z_1^n, \ldots, Z_N^n)$ may exceed the Wasserstein budget ε, so we pull it back toward the lifted samples. Set $t_n = 1$ if $d_n \le \varepsilon$ and $t_n = \varepsilon/d_n$ otherwise, and let $\widetilde{Z}_i^n = (1 - t_n)\widehat{Z}_i(\theta_n) + t_n Z_i^n$. Each Ξ^{θ_n} is convex, so $\widetilde{Z}_i^n \in \Xi^{\theta_n}$, and

$$(1/N)\textstyle\sum_{i=1}^N \left\| \widetilde{Z}_i^n - \widehat{Z}_i(\theta_n) \right\| = t_n d_n \le \varepsilon,$$

which gives $\widetilde{Z}^n \in C(\theta_n)$. From $d_n \to \bar{d} \le \varepsilon$ we get $t_n \to 1$, and the differences $\widehat{Z}_i(\theta_n) - Z_i^n$ are bounded, so

$$\widetilde{Z}_i^n - \bar{Z}_i = (1 - t_n)\big(\widehat{Z}_i(\theta_n) - Z_i^n\big) + \big(Z_i^n - \bar{Z}_i\big) \longrightarrow 0.$$

Hence every $\bar{Z} \in C(\bar{\theta})$ is the limit of a sequence $\widetilde{Z}^n \in C(\theta_n)$, proving lower hemicontinuity.

Berge's maximum theorem now gives continuity of the optimal value of (26), which is exactly $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$. The parameter set Θ is compact by Assumption 2(*ii*), so a continuous function attains its minimum over it. ∎

A large enough radius makes the Wasserstein constraint in (25) inactive, which reduces C, with $N = 1$, to the correspondence $\theta \mapsto \Xi^\theta$ and turns the optimal value into the worst-case loss. The following is therefore a special case of Proposition 4.

Proposition 5 (Continuity of the worst-case loss)**.** *Under the hypotheses of Proposition 4, the worst-case loss*

$$\theta \mapsto \sup_{z \in \mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big) = \sup_{(G,F) \in \Xi^\theta} \ell(G, F),$$

is continuous on Θ, and (OPT-PEP) *attains its minimum over Θ.*

Proof. The set Ξ^Θ is compact by Lemma 2, so its diameter is finite. We apply the proof of Proposition 4 with $N = 1$ and any radius $\varepsilon > \mathbf{diam}(\Xi^\Theta)$. Both Z_1 and $\widehat{Z}_1(\theta)$ lie in $\Xi^\theta \subseteq \Xi^\Theta$, so the budget constraint of (25) is inactive and $C(\theta) = \Xi^\theta$ for every $\theta \in \Theta$. Then (26) becomes $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_1) = \max_{(G,F)\in\Xi^\theta} \ell(G, F)$, which is continuous in θ and attains its minimum over Θ from Proposition 4. ■

C.3 Proof of Proposition 3

Throughout this subsection, we omit θ-dependency and write $\mathbf{P}^\theta$ and $\mathbf{Q}^\theta$ as $\mathbf{P}$ and $\mathbf{Q}$, and $\mathcal{R}$ and $\mathcal{R}_\varepsilon$ are evaluated on lifted distributions as in Section B. First, we state the proposition from (Kuhn et al., 2025), which will be useful in our proofs.

Proposition 6 (Proposition 8.5 of (Kuhn et al., 2025))**.** *Consider the* 1*-Wasserstein ambiguity set*

$$\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N) = \left\{ \mathbf{Q} \mid W_1(\mathbf{Q}, \widehat{\mathbf{P}}_N) \le \varepsilon \right\},$$

where $\mathbf{Q}$ *is a probability measure supported on a closed set* Ξ*. Then, if* ℓ *is a Lipschitz continuous function with* $\mathbf{E}_{Y\sim\widehat{\mathbf{P}}_N}(|\ell(Y)|) < \infty$*,*

$$\sup_{\mathbf{Q}\in\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N)} \mathop{\mathbf{E}}_{Y\sim\mathbf{Q}}\big(\ell(Y)\big) \le \mathop{\mathbf{E}}_{Y\sim\widehat{\mathbf{P}}_N}\big(\ell(Y)\big) + \varepsilon \,\mathrm{Lip}(\ell).$$

Note that from (Park et al., 2025), DRO loss interpolates (OPT-PEP) and (L2O) in the following sense: given a fixed $\theta \in \Theta$, the risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ tends to the worst-case loss of (OPT-PEP) when $\varepsilon \to \infty$ and equals the empirical risk of (L2O) when $\varepsilon = 0$. The following lemma states the monotonicity and upper bound, and the proof of Proposition 3 recovers the two extremes as $\varepsilon \to 0^+$ and $\varepsilon \to \infty$.

Lemma 5. *Suppose the hypotheses of Lemma 2 hold. For any algorithm parameter* $\theta \in \Theta$ *and probability distribution* $\mathbf{Q}$ *supported on* Ξ^θ*,*

$$\mathcal{R}(\theta, \mathbf{Q}) \le \mathcal{R}_\varepsilon(\theta, \mathbf{Q}) \le \mathcal{R}_{\varepsilon^+}(\theta, \mathbf{Q}) \le \sup_{(G,F)\in\Xi^\theta} \ell(G, F),$$

for any $\varepsilon^+ \ge \varepsilon \ge 0$*.*

Proof. From $\{\mathbf{Q}\} \subseteq \mathcal{U}_\varepsilon(\mathbf{Q}) \subseteq \mathcal{U}_{\varepsilon^+}(\mathbf{Q})$, we get

$$\mathcal{R}(\theta, \mathbf{Q}) \le \mathcal{R}_\varepsilon(\theta, \mathbf{Q}) \le \mathcal{R}_{\varepsilon^+}(\theta, \mathbf{Q}).$$

The set Ξ^Θ is compact by Lemma 2, and Ξ^θ is a closed subset of it, so Ξ^θ is compact as well. The loss ℓ is linear and hence continuous, so its supremum over Ξ^θ is finite and attained. Therefore, for every $\varepsilon > 0$,

$$\mathcal{R}_\varepsilon(\theta, \mathbf{Q}) = \sup_{\widetilde{\mathbf{Q}}\in\mathcal{U}_\varepsilon(\mathbf{Q})} \mathop{\mathbf{E}}_{(G,F)\sim\widetilde{\mathbf{Q}}}\big(\ell(G, F)\big) \le \sup_{\widetilde{\mathbf{Q}}\in\mathcal{U}_\varepsilon(\mathbf{Q})} \sup_{(G,F)\in\Xi^\theta} \ell(G, F) = \sup_{(G,F)\in\Xi^\theta} \ell(G, F).$$

■

We now prove Proposition 3.

Proof of Proposition 3. First, from Lemma 5,

$$\mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \le \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) \le \mathcal{R}_{\varepsilon^+}(\theta, \widehat{\mathbf{P}}_N) \le \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big),$$

for any $\theta \in \Theta$ and $\varepsilon^+ \ge \varepsilon > 0$. Each term in the inequality above is continuous in θ. The empirical risk is the linear loss ℓ composed with the algorithm map, which is Lipschitz by Lemma 3, the two DRO risks are continuous by Proposition 4, and the worst-case loss is continuous by Proposition 5. Since Θ is compact, each term attains its minimum over Θ. First of all,

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \le \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \le \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) \le \mathcal{R}_{\varepsilon^+}(\theta, \widehat{\mathbf{P}}_N), \qquad \forall\theta \in \Theta.$$

For $\theta^\star_\varepsilon \in \operatorname{argmin}_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$, we have

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) = \inf_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N), \qquad \forall \theta \in \Theta.$$

Similarly, with respect to $\theta^\star_{\varepsilon^+} \in \operatorname{argmin}_{\theta\in\Theta} \mathcal{R}_{\varepsilon^+}(\theta, \widehat{\mathbf{P}}_N)$,

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_{\varepsilon^+}(\theta^\star_{\varepsilon^+}, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_{\varepsilon^+}(\theta, \widehat{\mathbf{P}}_N), \qquad \forall \theta \in \Theta.$$

Lastly,

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_{\varepsilon^+}(\theta^\star_{\varepsilon^+}, \widehat{\mathbf{P}}_N) \leq \inf_{\theta\in\Theta} \sup_{(G,F)\in\Xi^\theta} \ell(G, F),$$

we observe that $\varepsilon \mapsto \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N)$ is monotonically nondecreasing and upper-bounded by $\inf_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big)$.

The hypotheses of Proposition 6 hold for every $\theta \in \Theta$. The lifted loss $\ell(G, F) = \mathbf{tr}(A_{\text{obj}}^T G) + b_{\text{obj}}^T F$ is linear, so Cauchy–Schwarz in the pair norm $\|(G, F)\| = \sqrt{\|G\|_F^2 + \|F\|^2}$ inducing W_1 gives the Lipschitz constant $\operatorname{Lip}(\ell) = \|(A_{\text{obj}}, b_{\text{obj}})\|$. The empirical distribution $\widehat{\mathbf{P}}_N^\theta$ has finite support, so $\mathbf{E}_{Y\sim\widehat{\mathbf{P}}_N^\theta} |\ell(Y)| < \infty$. The upper bound in the following display then follows by applying Proposition 6 to each θ and taking the infimum over Θ,

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) + \varepsilon \operatorname{Lip}(\ell).$$

Applying $\varepsilon \to 0^+$, we get the desired result.

For $\varepsilon \geq \mathbf{diam}(\Xi^\Theta)$, which is finite by Lemma 2, every distribution on Ξ^θ lies in $\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N^\theta)$, so $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) = \sup_{z\in\mathcal{Z}} \ell(\mathcal{A}_\theta(z))$ for all $\theta \in \Theta$ by Lemma 5. Taking the infimum over Θ gives the second limit, which is attained for every such ε. ■

C.4 Proof of Theorem 2

We now prove Theorem 2.

Proof of Theorem 2. According to Lemma 5,

$$\mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) \leq \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big),$$

holds true for all $\theta \in \Theta$. The minimizer $\theta^\star_\varepsilon \in \operatorname{argmin}_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$ exists by Proposition 4, and $\theta^\star_{\text{PEP}} \in \operatorname{argmin}_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell(\mathcal{A}_\theta(z))$ exists by Proposition 5. From these,

$$\mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \mathcal{R}_\varepsilon(\theta^\star_{\text{PEP}}, \widehat{\mathbf{P}}_N) \leq \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_{\theta^\star_{\text{PEP}}}(z)\big) = \inf_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big).$$

Taking $\inf_{\theta\in\Theta}$ consecutively from smaller to larger terms, we get

$$\inf_{\theta\in\Theta} \mathcal{R}(\theta, \widehat{\mathbf{P}}_N) \leq \inf_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) = \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \inf_{\theta\in\Theta} \sup_{z\in\mathcal{Z}} \ell\big(\mathcal{A}_\theta(z)\big).$$

The hypotheses of Proposition 6 hold here, as verified in Section C.3. Applying Proposition 6 with $Y = \mathcal{A}_\theta(z)$,

$$\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) = \sup_{\mathbf{Q}\in\mathcal{U}_\varepsilon(\widehat{\mathbf{P}}_N)} \mathop{\mathbf{E}}_{z\sim\mathbf{Q}} \big(\ell(\mathcal{A}_\theta(z))\big) \leq \mathop{\mathbf{E}}_{z\sim\widehat{\mathbf{P}}_N} \big(\ell(\mathcal{A}_\theta(z))\big) + \varepsilon \operatorname{Lip}(\ell),$$

holds for every $\theta \in \Theta$. Take $\inf_{\theta\in\Theta}$ on the left-hand side of the inequality to have

$$\mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) = \inf_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N) \leq \mathop{\mathbf{E}}_{z\sim\widehat{\mathbf{P}}_N} \big(\ell(\mathcal{A}_\theta(z))\big) + \varepsilon \operatorname{Lip}(\ell),$$

and also take $\inf_{\theta\in\Theta}$ on the right-hand side to get

$$\mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \leq \inf_{\theta\in\Theta} \mathop{\mathbf{E}}_{z\sim\widehat{\mathbf{P}}_N} \big(\ell(\mathcal{A}_\theta(z))\big) + \varepsilon \operatorname{Lip}(\ell).$$

We conclude the proof by upper-bounding the true risk $\mathcal{R}(\theta^\star_\varepsilon, \mathbf{P})$ by the DRO risk $\mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N)$ with probability at least $1 - \beta$, using Theorem 1. ■

D ADDITIONAL DETAILS OF SECTION 3.4

Algorithm 1 describes the solution method for the learning problem (DR-L2O).

Algorithm 1 Stochastic Gradient Method for Distributionally Robust L2O (DR-L2O)

Input: training data $\widehat{\mathcal{D}}_N$, parameter initialization θ^0, mini-batch size n, iteration counter $k = 0$.
repeat
  Sample mini-batch $\widehat{\mathcal{D}}_n$ from training data $\widehat{\mathcal{D}}_N$
  Define mini-batched empirical distribution $\widehat{\mathbf{P}}_n$ of $\widehat{\mathcal{D}}_n$
  Evaluate $d\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)/d\theta$ using Lemma 6.
  Update $\theta^{k+1} \leftarrow \texttt{AdamWUpdate}(\theta^k)$.
  $k \leftarrow k + 1$.
until θ^k converges

Since (DR-L2O) is nonconvex, the parameter $\hat{\theta}_\varepsilon$ returned by Algorithm 1 may be a local minimizer rather than a global minimizer $\theta^\star_\varepsilon$. Theorem 1 holds uniformly over Θ, so the DRO risk $\mathcal{R}_\varepsilon(\hat{\theta}_\varepsilon, \widehat{\mathbf{P}}_N)$ still upper-bounds the true risk $\mathcal{R}(\hat{\theta}_\varepsilon, \mathbf{P})$ with probability at least $1 - \beta$. Theorem 2 uses global optimality only through $\mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) = \inf_{\theta\in\Theta} \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_N)$. For $\hat{\theta}_\varepsilon$, both of its bounds therefore hold with the optimality gap $\mathcal{R}_\varepsilon(\hat{\theta}_\varepsilon, \widehat{\mathbf{P}}_N) - \mathcal{R}_\varepsilon(\theta^\star_\varepsilon, \widehat{\mathbf{P}}_N) \geq 0$ added to the right-hand side.

In order to evaluate the gradient of $\theta \mapsto \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)$ given a dataset $\widehat{\mathcal{D}}_n$ of n samples, we use the approach of (Agrawal et al., 2019b;a; Healey et al., 2026) to differentiate the solution mapping of the conic linear problem in terms of problem parameters. Let $\mathcal{T}\big(\{(A_m, b_m)\}_{m=0}^M, \{(\widehat{G}_i, \widehat{F}_i)\}_{i=1}^n\big)$ be a solution mapping of the problem (DRO-PEP-D) with N replaced by n, and let $\mathcal{L}$ be its objective function. Then the DRO risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)$ is composed of the solution mapping and $\mathcal{L}$ as follows:

$$\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n) = \mathcal{L}\Big(\mathcal{T}\big(\mathcal{S}_\theta, \mathcal{A}_\theta(\widehat{\mathcal{D}}_n)\big)\Big). \tag{27}$$

We finally apply the chain rule to obtain the gradient with respect to θ, as follows:

Lemma 6. *Given $\widehat{\mathcal{D}}_n$, the gradient of $\theta \mapsto \mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)$ evaluates as*

$$\frac{d\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)}{d\theta} = \left.\frac{d\mathcal{L}}{dt}\right|_{t=\mathcal{T}(\mathcal{S}_\theta, \mathcal{A}_\theta(\widehat{\mathcal{D}}_n))} \times \left(\sum_{m=0}^M \left.\frac{\partial \mathcal{T}}{\partial (A_m, b_m)}\right|_{(\mathcal{S}_\theta, \mathcal{A}_\theta(\widehat{\mathcal{D}}_n))} \frac{d(A_m(\theta), b_m(\theta))}{d\theta} + \sum_{i=1}^n \left.\frac{\partial \mathcal{T}}{\partial (\widehat{G}_i, \widehat{F}_i)}\right|_{(\mathcal{S}_\theta, \mathcal{A}_\theta(\widehat{\mathcal{D}}_n))} \frac{d\mathcal{A}_\theta(\hat{z}_i)}{d\theta} \right).$$

Proof. We interpret the risk $\mathcal{R}_\varepsilon(\theta, \widehat{\mathbf{P}}_n)$ as the parametric conic program (DRO-PEP-D) with parameters $(\widehat{G}_i, \widehat{F}_i)$ for $i = 1, \ldots, n$ and (A_m, b_m) for $m = 0, 1, \ldots, M$. Applying the chain rule to the composition (27) gives the result. ■

Remark 7 (Differentiability of the solution map). Lemma 6 requires the solution map $\mathcal{T}$ to be differentiable at the parameters it is evaluated at. The solution map of a semidefinite program is differentiable at parameters whose primal and dual solutions are unique and satisfy strict complementarity together with primal and dual nondegeneracy. These conditions make the Jacobian of the optimality conditions nonsingular, so the implicit function theorem applies. They also hold for generic problem data, which leaves the parameters where $\mathcal{T}$ fails to be differentiable in a thin set. At such parameters the approach of (Agrawal et al., 2019b) returns a heuristic quantity in place of the derivative.

Training procedure For each experiment, we fix the algorithm structure and learn the schedule of step sizes (and, for logistic regression, momentum coefficients) $\theta = \{\theta^k\}_{k=0}^{K-1}$ for each horizon K and each of (DR-L2O), (OPT-PEP), and (L2O). For (DR-L2O) and (L2O), we use mini-batched stochastic gradient descent with the AdamW optimizer (Kingma & Ba, 2015; Loshchilov & Hutter, 2019), with weight decay cross-validated over $\{0, 10^{-5}, 10^{-4}, 10^{-3}\}$, combined with a linear

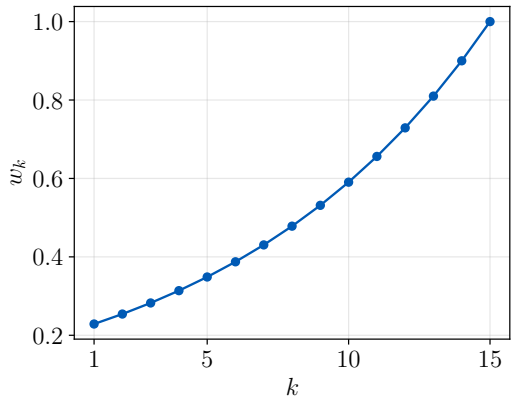


Figure 6: A plot showing the exponential weight decay for our training objective with weights $w_k = 0.9^{K-k}$ for $K = 15$.

warm-up followed by a cosine annealing learning rate schedule (Loshchilov & Hutter, 2017). We acknowledge that, since the DRO risk is not an average over samples, the mini-batch gradient is not an unbiased estimate of the gradient of (DR-L2O) on the full training set. For (OPT-PEP), we follow the approach of (Kamri et al., 2025): their (Kamri et al., 2025, Theorem 3.2) gives an explicit gradient of the worst-case rate with respect to the step size, and we apply gradient descent with the same cosine annealing schedule. All experiments run for 500 iterations, with 10% of the iterations used for a linear warm-up to the maximum learning rate, which is cross-validated from $\{10^{-5}, 10^{-4}, 10^{-3}\}$.

When learning the schedule, we do not use the performance loss at the last iterate as the training objective as this can hinder gradient information flow to earlier step sizes (Chen et al., 2022a). Instead, we use an exponentially decaying weighted summation for the training loss. Letting ℓ^k represent the loss at iterate k, we minimize the sample analogue of the surrogate objective:

$$\mathop{\mathbf{E}}_{z\sim\mathbf{P}}\left(\sum_{k=1}^{K} w_k \ell^k(\mathcal{A}_\theta(z))\right).$$

We choose $w_k = 0.9^{K-k}$, and show a plot of the exponential decay in Figure 6. Since $w_K = 1$ and every ℓ^k is nonnegative, the final-iterate loss ℓ^K reported in our figures and tables is bounded by this weighted loss, so the guarantees of Section 4, which apply to the weighted loss we train on, also bound the final-iterate risk.

Additionally, during the learning procedure, in order to remove numerical issues with any step size component becoming negative, we *square* the step size before plugging it into the first-order method. We initialize the values so that the square of the input step size is our desired value but we backpropagate the gradients to the square root. This squaring enforces nonnegative step sizes, the lower bound of the parameter box Θ in Assumption 2; we do not need to enforce an upper bound, since we optimize for convergent algorithms, which naturally keeps the step sizes bounded.

Parameter selection For each experiment, we set a maximum value of K and learn a separate schedule for each $k = 1, \ldots, K$ used as the learning horizon. The Wasserstein radius $\varepsilon > 0$ from Section 3.2 controls the probability of constraint satisfaction; in practice it is treated as a hyperparameter tuned by cross-validation (Gotoh et al., 2021; Park et al., 2025). We cross-validate over $\varepsilon \in \{10^{-2}, 10^{-1}, 1, 5, 10\}$ on a validation set and choose the schedule minimizing the empirical risk. A separate test set is used for final loss evaluations. As noted in Remark 2, this cross-validated value is the operating radius, whereas $\bar{\varepsilon}(\beta)$ in Theorem 1 is the radius certified a priori.

Offline learning versus online deployment The inner conic program and the stochastic gradient method of Algorithm 1 run once per task and horizon, offline. The conic program evaluates the DRO risk exactly, so (DR-L2O) minimizes a certified quantity rather than the empirical average of (L2O). At deployment the learned optimizer is the underlying first-order method with fixed scalars θ, so its per-iteration cost equals that of the baseline method. Adaptive rules move this tuning cost online

rather than avoiding it. Backtracking line search, for instance, has its own hyperparameters and spends several oracle calls per accepted step, so at an equal oracle budget it completes fewer steps than a fixed schedule.

The size of the conic program (DRO-PEP-D) is set by the horizon and the mini-batch size, not by the instance dimension, since the lifted data are inner products of iterates and oracle outputs. The instance dimension ranges from 25 variables in Section 5.1 to more than 10^4 in Section 5.3 without changing the program. The horizon is the binding cost parameter, since the lifted variables carry $O(K^2)$ entries per sample and generic conic solvers scale worse than that. We therefore claim practicality only for moderate horizons, which is the scalability limitation of Section 6.

E Further experiment details

All examples are written in Python 3.13 with JAX version 0.9.0; the code is available at `https://anonymous.4open.science/r/dr-l2o-`.The SDPs are solved using the Clarabel solver (Goulart & Chen, 2026) with primal residual, dual residual, and duality gap tolerances all set to 10^{-5}. For the derivatives, we use the diffcp package (Agrawal et al., 2019a;b) in order to differentiate through the KKT conditions of the inner SDP. All computations were run on a high-performance computing cluster with 4 CPU cores and 64GB of RAM.

In all plots, shaded bars show the 10^{th} to 90^{th} quantile ranges. For each experiment, we also tabulate the plotted values for every horizon K: the loss tables report the mean, the 10^{th} quantile, the median, and the 90^{th} quantile of the test loss, and the fraction-solved tables use the same tolerances η as the figures. Furthermore, in all SGD iteration timing tables, our code with JAX uses just-in-time (JIT) techniques for computational efficiency, and we noticed that the first few SGD iterations take significantly longer because of the compilation time. Since this is an initial startup cost, these initial times become outliers for the full distribution of times so we discard the first 5 iteration times before computing the averages and standard deviations.

In these experimental families, since the optimal values are not always at 0 with objective value 0, we use a separate set of 1000 instances, solve all problem instances to optimality, and compute the maximum distance to optimality from the initial points. We then add a 10% buffer to ensure that all sample values in the training set satisfy the interpolation conditions with this computed sample distance.

E.1 Further details of logistic regression experiment in Section 5.1

We use the german.numer dataset from LIBSVM (Chang & Lin, 2011), which contains 1000 examples with 24 features. We standardize each feature over the full dataset and append an intercept column of ones, so that $n = 25$. Each problem instance is a uniform random subsample of $m = 300$ rows, and we reject subsamples that are linearly separable, since the unregularized problem then has no minimizer. We initialize all x^0 values at 0. We compute L as the maximum of $\lambda_{\max}(A_i^T A_i)/(4m)$ over the training instances and R as the maximum of $\|x_i^\star\|$ over the training instances. For the out-of-distribution set, we multiply the feature matrix of each instance by 1.25 and hold all other parameters constant, which increases L by a factor of 1.25^2. Every out-of-distribution instance then has a smoothness constant between 0.91 and 1.20, above the training bound $L = 0.80$, so these tests probe robustness beyond what Theorem 1 certifies, and the out-of-distribution performance of (DR-L2O) shows a strength of our method.

We use a training set of size 1000; the validation set and both in-distribution and out-of-distribution test sets are size 250. We use a mini-batch of size 20 for the SGD iterations. We initialize the step sizes with constant value $1/L$ and the momentum coefficients with the standard FGM sequence for smooth convex functions (d'Aspremont et al., 2021). The final momentum coefficient ρ^{K-1} does not affect x^K, so only the first $K-1$ coefficients are effectively learned. In Figure 7 we show the test losses of all the learned schedules while in Table 1 we provide average times per SGD iteration for select values of K. Tables 4 and 5 list the values plotted in Figures 7 and 3 for every horizon K.

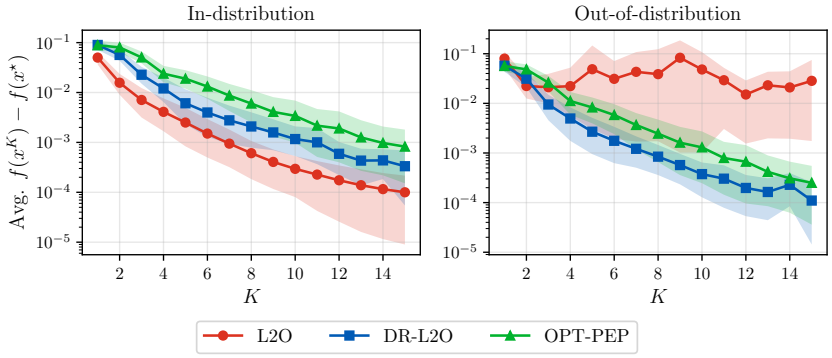


Figure 7: Logistic regression experiment (Section 5.1) test losses against horizon K on the *Left:* in-distribution test set and *Right:* out-of-distribution test set.

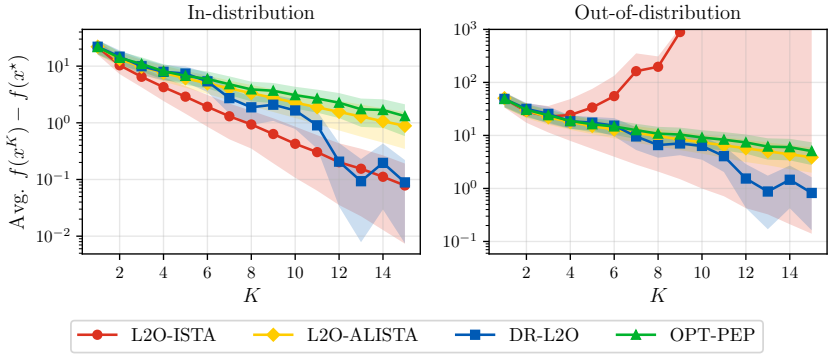


Figure 8: LASSO experiment (Section 5.2) test losses against horizon K on the *Left:* in-distribution test set and *Right:* out-of-distribution test set.

E.2 Further details of LASSO experiment in Section 5.2

We use a sparse coding example, or the problem of recovering a sparse vector $\tilde{x}$ from noisy measurements through a sparse dictionary matrix $A \in \mathbf{R}^{m \times n}$ (Chen et al., 2018). We form A by sampling each entry $A_{ij} \sim \mathcal{N}(0, 1/m)$ and normalizing each column to have unit 2-norm; we fix A for both the in-distribution and out-of-distribution sets and compute L as the largest eigenvalue of $A^T A$. We sample a sparse vector $\tilde{x}_j \in \mathbf{R}^n$ as $\tilde{x}_j \sim \mathcal{N}(0, \sigma_x^2)$ with probability p_{mask} and 0 otherwise. We then construct $b_j = A\tilde{x}_j + \delta_j$ where $\delta_j \sim \mathcal{N}(0, \sigma_{\text{err}}^2)$. In the experiment, we set $(m, n) = (250, 500)$, $\lambda = 0.4$, $\sigma_x = 2.0$ for the in-distribution set and $\sigma_x = 3.0$ for the out-of-distribution set, $\sigma_{\text{err}} = 0.01$, and $p_{\text{mask}} = 0.1$.

The training set, validation set, and test set sizes are 1000, 250, and 248, respectively. We use a size 10 mini-batch for SGD and initialize the step size schedules with constant value $1/L$. We initialize all x^0 values at 0. In Figure 8 we show the test losses of all the learned schedules while in Table 2 we provide average times per SGD iteration for select values of K. Tables 6 and 8 list the values plotted in Figures 8 and 4 for every horizon K.

E.3 Further details of TV inpainting experiment in Section 5.3

For the in-distribution dataset, we use the Olivetti faces dataset (Samaria & Harter, 1994; Cambridge, 1994), which contains 40 faces with 10 grayscale images each (64×64 pixels) of different angles. For the corruption, we choose 10% of pixels of each image uniformly at random to black out (*i.e.* set to 0). We randomly pick 28 faces for the training set, 4 faces for the validation set, 8 faces for the test set, and ensure that the images are separated in a stratified manner to prevent data leakage during

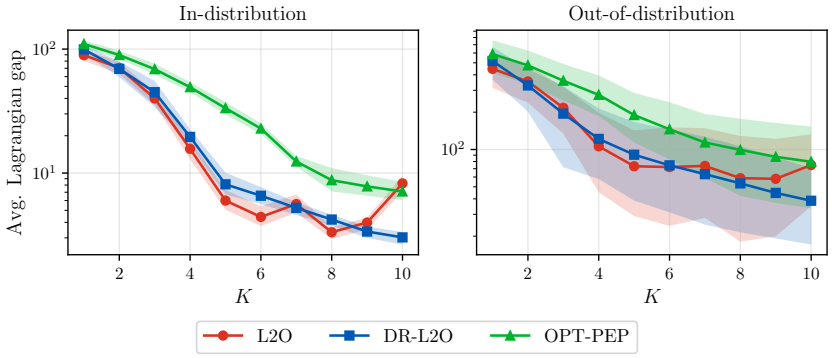


Figure 9: TV inpainting experiment (Section 5.3) test losses against horizon K on the *Left:* Olivetti in-distribution test set and *Right:* Tiny ImageNet out-of-distribution test set.

training. For the out-of-distribution set, we pick 40 random *color* images from Tiny ImageNet (Le & Yang, 2015; Deng et al., 2009), where each $U_{i,j} \in \mathbf{R}^3$ is now a triplet of RGB values.

We reformulate the TV LP in the standard LP form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & Gx \leq h \\ & \underline{x} \leq x \leq \overline{x}, \end{array}$$

equivalent to the formulation used in (Lu & Yang, 2025, Equation 1) and (Blin et al., 2026), and its saddle-point reformulation. To obtain this form from the TV LP in Section 5.3, we vectorize the image U into $\mathrm{vec}(U) \in \mathbf{R}^{mn}$ and introduce nonnegative slacks $t \in \mathbf{R}^{2(m-1)(n-1)}$ that upper-bound each absolute-difference term in the objective. Let D stack the horizontal and vertical pixel-difference operators and E select the known-pixel coordinates. Writing $x = (\mathrm{vec}(U), t)$, the data of the standard LP are

$$c = \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix}, \quad A = \begin{bmatrix} E & 0 \end{bmatrix}, \quad b = \mathrm{vec}(U^{\mathrm{orig}})_{\mathcal{K}}, \quad G = \begin{bmatrix} D & -I \\ -D & -I \end{bmatrix}, \quad h = 0,$$

with box bounds $\underline{x} = 0$ and $\overline{x} = (\mathbf{1}, +\infty)$ on $(\mathrm{vec}(U), t)$. The corresponding PDHG iterations are

$$\begin{aligned} x^{k+1} &= \mathbf{prox}_{\tau^k f}\left(x^k - \tau^k M^T u^k\right), \\ \bar{x}^{k+1} &= x^{k+1} + \rho^k (x^{k+1} - x^k), \\ u^{k+1} &= \mathbf{prox}_{\sigma^k g^*}\left(u^k + \sigma^k M \bar{x}^{k+1}\right). \end{aligned}$$

For each sample LP, we define the stacked constraint matrix $M^T = \begin{bmatrix} A^T & -G^T \end{bmatrix}$ and compute the maximum value of $\|M\|_2$ across the samples, denoted $M_{\max}$. We initialize the training procedures with $(\tau^k, \rho^k, \sigma^k) = (0.5/M_{\max}, 1, 0.5/M_{\max})$. In order for the Lagrangian duality gap to make sense as a metric, we need to initialize the algorithm from a feasible point, so we choose to initialize $x^0 = 1/2$ and $u^0 = 1$. In Figure 9 we show the test losses of all the learned schedules and in Figure 10 we show the fractions of problems solved. Lastly, in Table 3 we provide average times per SGD iteration for select values of K. Tables 9 and 10 list the values plotted in Figures 9 and 10 for every horizon K.

E.4 Adaptive comparison

Since the learned schedules are fixed ahead of time, it is natural to compare them against a step size that adapts to each instance. Such a comparison is not straightforward to make fair: line searches use extra oracle calls within each iteration, and standard techniques require the performance metric to be the objective value. For the TV inpainting experiment, our performance metric is the Lagrangian duality gap, so objective-based backtracking and Armijo conditions do not apply, and we omit this experiment.

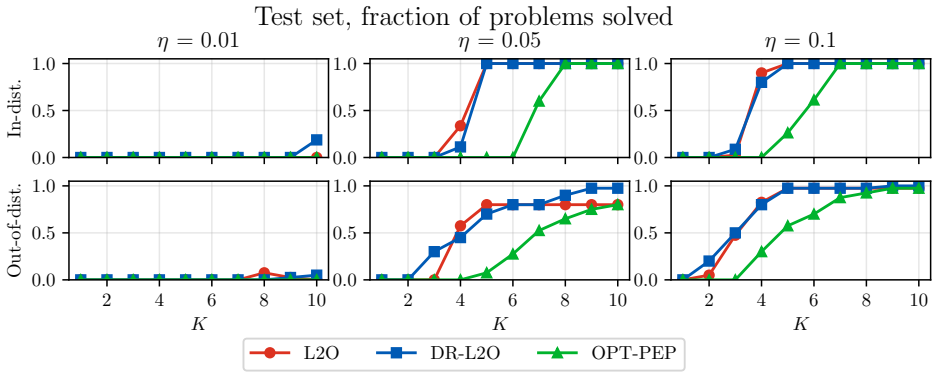


Figure 10: Fractions of TV inpainting problems solved to different relative tolerances against horizon K. *Top:* Fractions solved on the Olivetti in-distribution set. *Bottom:* Fractions solved on the Tiny ImageNet out-of-distribution set.



Figure 11: More reconstructions on images from the Tiny ImageNet dataset. The images are ordered in the same way as in Figure 5.

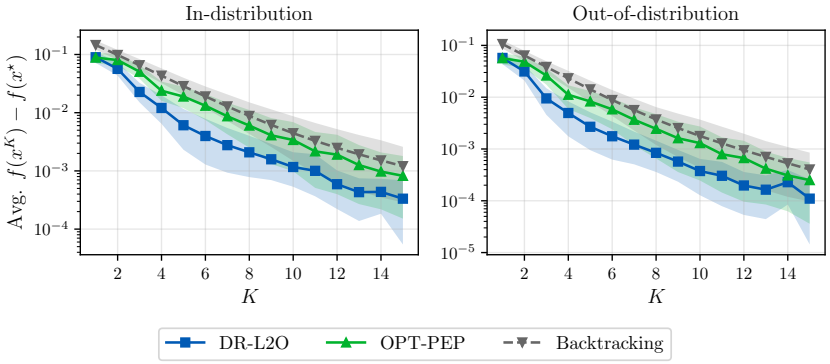


Figure 12: Logistic regression experiment (Section 5.1) test losses of (DR-L2O), (OPT-PEP), and backtracking line search against horizon K on the *Left:* in-distribution test set and *Right:* out-of-distribution test set.

As the adaptive baseline we use backtracking line search (Boyd & Vandenberghe, 2004, Algorithm 9.2): each iteration starts from the trial step $t = 1$ and halves t until a sufficient decrease condition holds. For logistic regression we use the Armijo condition $f(y^k - t\nabla f(y^k)) \leq f(y^k) - 0.1\, t\, \|\nabla f(y^k)\|^2$ at the extrapolated point y^k, with the standard FGM momentum sequence. Each trial reuses the product $A\nabla f(y^k)$, so backtracking uses two matrix-vector products per iteration, the same as the learned schedules, and comparing by horizon K is fair (Figure 12). For LASSO we use the proximal gradient variant (Beck, 2017), which accepts $x^+ = \mathbf{prox}_{t\lambda\|\cdot\|_1}(x^k - t\nabla f(x^k))$ if $f(x^+) \leq f(x^k) + \nabla f(x^k)^T(x^+ - x^k) + \|x^+ - x^k\|^2/(2t)$, with f the least-squares term. Here each trial needs a new product with A, so an iteration can cost more than the two products of a learned iteration: at $K = 15$, backtracking uses on average 36.5 in-distribution and 39.2 out-of-distribution products instead of 30. We therefore compare against both the horizon K and the number of matrix-vector products (Figure 13). Tables 4 and 7 list the plotted values.

F TABLES

Table 1: Logistic regression experiment per-SGD-iteration wall time (mean $\pm$ 2σ, in seconds) for the validation-best schedule used by the losses figure.

Framework	K	Time (s)
L2O	1	0.001 ± 0.000
	5	0.002 ± 0.000
	10	0.006 ± 0.001
	15	0.009 ± 0.001
DR-L2O	1	0.062 ± 0.045
	5	0.602 ± 0.157
	10	2.635 ± 0.594
	15	7.402 ± 0.706
OPT-PEP	1	0.002 ± 0.000
	5	0.006 ± 0.001
	10	0.018 ± 0.002
	15	0.070 ± 0.005

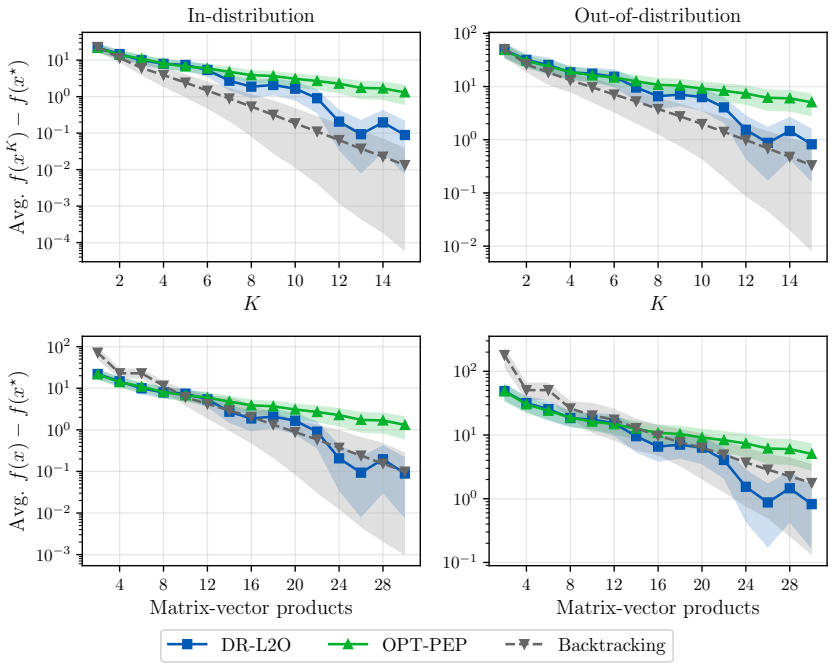


Figure 13: LASSO experiment (Section 5.2) test losses of (DR-L2O), (OPT-PEP), and backtracking line search on the *Left:* in-distribution test set and *Right:* out-of-distribution test set. *Top:* against horizon K. *Bottom:* against the number of matrix-vector products.

Table 2: LASSO experiment per-SGD-iteration wall time (mean $\pm$ 2σ, in seconds) for the validation-best schedule used by the losses figure.

Framework	K	Time (s)
L2O-ISTA	1	0.006 ± 0.000
	5	0.015 ± 0.001
	10	0.025 ± 0.002
	15	0.037 ± 0.002
L2O-ALISTA	1	0.004 ± 0.000
	5	0.011 ± 0.001
	10	0.020 ± 0.001
	15	0.029 ± 0.001
DR-L2O	1	0.144 ± 0.027
	5	0.825 ± 0.129
	10	2.993 ± 0.327
	15	14.697 ± 8.933
OPT-PEP	1	0.015 ± 0.001
	5	0.050 ± 0.009
	10	0.364 ± 0.041
	15	2.008 ± 0.424

Table 3: TV inpainting experiment per-SGD-iteration wall time (mean $\pm$ 2σ, in seconds) for the validation-best schedule used by the losses figure.

Framework	K	Time (s)
L2O	1	0.119 ± 0.014
	5	0.450 ± 0.016
	10	0.845 ± 0.019
DR-L2O	1	0.410 ± 0.018
	5	3.755 ± 0.154
	10	57.125 ± 27.471
OPT-PEP	1	0.037 ± 0.012
	5	0.371 ± 0.021
	10	3.954 ± 0.309

Table 4: Logistic regression experiment test losses plotted in Figures 7 and 12: mean, 10^{th} quantile, median, and 90^{th} quantile of $f(x^K) - f(x^\star)$ over the in-distribution and out-of-distribution test sets for every horizon K. Backtracking line search uses two matrix-vector products per iteration, as the learned schedules do, so its values at a budget of $2K$ products equal its values at horizon K.

		In-distribution				Out-of-distribution			
Framework	K	Mean	10^{th}	Median	90^{th}	Mean	10^{th}	Median	90^{th}
L2O	1	5.01e-2	3.64e-2	4.93e-2	6.43e-2	7.98e-2	5.14e-2	7.97e-2	1.10e-1
	2	1.57e-2	9.03e-3	1.50e-2	2.37e-2	2.24e-2	1.25e-2	2.00e-2	3.37e-2
	3	7.09e-3	3.15e-3	6.34e-3	1.23e-2	2.12e-2	9.98e-3	1.58e-2	3.93e-2
	4	4.10e-3	1.67e-3	3.54e-3	7.62e-3	2.22e-2	5.56e-3	1.10e-2	5.24e-2
	5	2.50e-3	8.34e-4	2.10e-3	4.93e-3	4.87e-2	3.04e-3	1.57e-2	1.47e-1
	6	1.50e-3	4.98e-4	1.18e-3	2.93e-3	3.12e-2	1.93e-3	2.21e-2	7.12e-2
	7	9.45e-4	3.19e-4	7.61e-4	1.81e-3	4.32e-2	1.70e-3	2.88e-2	1.08e-1
	8	6.08e-4	1.84e-4	4.41e-4	1.07e-3	3.88e-2	1.72e-3	2.13e-2	1.23e-1
	9	4.06e-4	1.16e-4	2.75e-4	7.22e-4	8.30e-2	1.02e-3	7.00e-2	1.86e-1
	10	2.95e-4	8.01e-5	1.87e-4	5.30e-4	4.82e-2	1.02e-3	4.14e-2	1.08e-1
	11	2.26e-4	4.23e-5	1.19e-4	4.23e-4	2.93e-2	3.05e-3	2.96e-2	5.20e-2
	12	1.74e-4	2.57e-5	8.04e-5	3.45e-4	1.51e-2	1.55e-3	1.36e-2	2.90e-2
	13	1.39e-4	1.60e-5	5.51e-5	2.89e-4	2.31e-2	1.97e-3	1.92e-2	4.34e-2
	14	1.16e-4	1.16e-5	3.57e-5	2.43e-4	2.09e-2	1.94e-3	1.67e-2	4.42e-2
	15	1.00e-4	8.98e-6	2.90e-5	2.16e-4	2.84e-2	1.74e-3	1.83e-2	7.49e-2
DR-L2O	1	8.97e-2	7.18e-2	8.74e-2	1.11e-1	5.73e-2	4.35e-2	5.68e-2	7.10e-2
	2	5.66e-2	4.33e-2	5.50e-2	7.34e-2	3.14e-2	2.09e-2	3.05e-2	4.22e-2
	3	2.27e-2	1.50e-2	2.12e-2	3.26e-2	9.48e-3	4.57e-3	8.89e-3	1.49e-2
	4	1.21e-2	6.44e-3	1.07e-2	1.99e-2	4.95e-3	1.75e-3	4.43e-3	8.29e-3
	5	6.09e-3	2.31e-3	5.06e-3	1.16e-2	2.68e-3	9.49e-4	2.33e-3	5.11e-3
	6	3.98e-3	1.29e-3	3.25e-3	7.87e-3	1.76e-3	6.24e-4	1.44e-3	3.37e-3
	7	2.77e-3	9.32e-4	2.28e-3	5.47e-3	1.21e-3	5.05e-4	1.02e-3	2.17e-3
	8	2.09e-3	7.82e-4	1.65e-3	4.01e-3	8.43e-4	3.59e-4	7.11e-4	1.45e-3
	9	1.58e-3	6.95e-4	1.26e-3	2.91e-3	5.71e-4	2.33e-4	4.89e-4	9.61e-4
	10	1.17e-3	5.34e-4	9.28e-4	2.09e-3	3.75e-4	1.28e-4	3.27e-4	6.44e-4
	11	1.00e-3	3.66e-4	7.46e-4	1.93e-3	3.04e-4	7.86e-5	2.33e-4	5.64e-4
	12	5.91e-4	2.21e-4	4.51e-4	1.02e-3	1.97e-4	5.33e-5	1.47e-4	3.60e-4
	13	4.32e-4	1.38e-4	3.12e-4	7.47e-4	1.64e-4	4.42e-5	1.09e-4	3.09e-4
	14	4.36e-4	1.84e-4	3.78e-4	7.31e-4	2.29e-4	8.45e-5	1.91e-4	3.94e-4
	15	3.33e-4	5.44e-5	1.65e-4	6.90e-4	1.10e-4	1.43e-5	3.83e-5	2.58e-4
OPT-PEP	1	8.85e-2	7.08e-2	8.62e-2	1.09e-1	5.66e-2	4.29e-2	5.61e-2	7.03e-2
	2	8.02e-2	6.42e-2	7.73e-2	1.00e-1	4.84e-2	3.65e-2	4.75e-2	6.11e-2
	3	5.05e-2	3.85e-2	4.84e-2	6.63e-2	2.62e-2	1.75e-2	2.52e-2	3.59e-2
	4	2.39e-2	1.60e-2	2.23e-2	3.42e-2	1.11e-2	5.91e-3	1.06e-2	1.70e-2
	5	1.89e-2	1.20e-2	1.75e-2	2.80e-2	8.27e-3	3.95e-3	7.79e-3	1.31e-2
	6	1.32e-2	7.49e-3	1.18e-2	2.10e-2	5.82e-3	2.40e-3	5.34e-3	9.64e-3
	7	8.68e-3	4.25e-3	7.58e-3	1.49e-2	3.67e-3	1.18e-3	3.17e-3	6.37e-3
	8	6.01e-3	2.51e-3	5.05e-3	1.11e-2	2.45e-3	6.55e-4	2.03e-3	4.71e-3
	9	4.09e-3	1.32e-3	3.33e-3	8.08e-3	1.61e-3	3.59e-4	1.30e-3	3.36e-3
	10	3.40e-3	9.86e-4	2.77e-3	6.89e-3	1.30e-3	2.46e-4	1.03e-3	2.72e-3
	11	2.17e-3	5.11e-4	1.72e-3	4.67e-3	7.93e-4	1.44e-4	5.96e-4	1.68e-3
	12	1.90e-3	3.99e-4	1.49e-3	4.18e-3	6.67e-4	9.60e-5	4.76e-4	1.44e-3
	13	1.26e-3	2.69e-4	9.31e-4	2.78e-3	4.17e-4	8.47e-5	2.64e-4	8.93e-4
	14	9.75e-4	2.19e-4	6.92e-4	2.08e-3	3.09e-4	6.26e-5	1.71e-4	6.70e-4
	15	8.23e-4	1.51e-4	5.43e-4	1.80e-3	2.50e-4	3.60e-5	1.12e-4	5.50e-4
Backtracking	1	1.45e-1	1.21e-1	1.43e-1	1.75e-1	1.06e-1	8.71e-2	1.06e-1	1.25e-1
	2	9.81e-2	8.01e-2	9.52e-2	1.20e-1	6.40e-2	4.95e-2	6.34e-2	7.89e-2
	3	6.53e-2	5.13e-2	6.27e-2	8.33e-2	3.84e-2	2.82e-2	3.75e-2	4.94e-2
	4	4.33e-2	3.23e-2	4.13e-2	5.78e-2	2.32e-2	1.52e-2	2.22e-2	3.23e-2
	5	2.86e-2	2.01e-2	2.68e-2	3.99e-2	1.42e-2	8.12e-3	1.35e-2	2.09e-2
	6	1.91e-2	1.21e-2	1.77e-2	2.83e-2	8.82e-3	4.34e-3	8.18e-3	1.37e-2
	7	1.28e-2	7.21e-3	1.15e-2	2.06e-2	5.64e-3	2.27e-3	5.17e-3	9.35e-3
	8	8.80e-3	4.32e-3	7.68e-3	1.51e-2	3.72e-3	1.20e-3	3.22e-3	6.44e-3
	9	6.17e-3	2.58e-3	5.20e-3	1.14e-2	2.53e-3	6.99e-4	2.11e-3	4.82e-3
	10	4.44e-3	1.48e-3	3.62e-3	8.69e-3	1.78e-3	4.30e-4	1.45e-3	3.70e-3
	11	3.28e-3	9.25e-4	2.70e-3	6.67e-3	1.28e-3	2.87e-4	1.02e-3	2.69e-3
	12	2.49e-3	6.32e-4	2.00e-3	5.22e-3	9.45e-4	1.87e-4	7.32e-4	1.97e-3
	13	1.93e-3	4.51e-4	1.53e-3	4.14e-3	7.04e-4	1.59e-4	5.29e-4	1.42e-3
	14	1.52e-3	3.49e-4	1.16e-3	3.36e-3	5.30e-4	1.19e-4	3.62e-4	1.09e-3
	15	1.21e-3	2.88e-4	8.89e-4	2.60e-3	4.02e-4	9.68e-5	2.53e-4	8.52e-4

Table 5: Fractions of logistic regression problems solved plotted in Figure 3, at each relative tolerance η, on the in-distribution and out-of-distribution test sets for every horizon K.

		In-distribution			Out-of-distribution		
Framework	K	$\eta = 10^{-4}$	$\eta = 10^{-3}$	$\eta = 10^{-2}$	$\eta = 10^{-4}$	$\eta = 10^{-3}$	$\eta = 10^{-2}$
L2O	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.472	0.000	0.000	0.196
	3	0.000	0.004	0.948	0.000	0.000	0.428
	4	0.000	0.076	0.996	0.000	0.000	0.584
	5	0.000	0.300	1.000	0.000	0.000	0.476
	6	0.000	0.604	1.000	0.000	0.072	0.420
	7	0.000	0.832	1.000	0.000	0.088	0.364
	8	0.032	0.936	1.000	0.000	0.084	0.376
	9	0.180	0.972	1.000	0.012	0.116	0.272
	10	0.352	0.972	1.000	0.004	0.116	0.248
	11	0.556	0.980	1.000	0.016	0.080	0.248
	12	0.648	0.992	1.000	0.032	0.096	0.540
	13	0.688	0.996	1.000	0.048	0.088	0.352
	14	0.736	0.996	1.000	0.048	0.084	0.420
	15	0.744	1.000	1.000	0.052	0.084	0.384
DR-L2O	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.008
	3	0.000	0.000	0.092	0.000	0.000	0.888
	4	0.000	0.000	0.736	0.000	0.056	0.988
	5	0.000	0.036	0.968	0.000	0.292	1.000
	6	0.000	0.136	0.992	0.000	0.492	1.000
	7	0.000	0.284	0.996	0.000	0.728	1.000
	8	0.000	0.412	1.000	0.012	0.896	1.000
	9	0.000	0.584	1.000	0.024	0.968	1.000
	10	0.000	0.772	1.000	0.132	0.984	1.000
	11	0.000	0.816	1.000	0.284	0.984	1.000
	12	0.020	0.948	1.000	0.488	0.996	1.000
	13	0.104	0.968	1.000	0.600	1.000	1.000
	14	0.048	0.976	1.000	0.316	1.000	1.000
	15	0.448	0.972	1.000	0.704	1.000	1.000
OPT-PEP	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.036
	4	0.000	0.000	0.072	0.000	0.000	0.780
	5	0.000	0.000	0.288	0.000	0.004	0.924
	6	0.000	0.000	0.672	0.000	0.024	0.976
	7	0.000	0.000	0.864	0.000	0.136	1.000
	8	0.000	0.036	0.980	0.000	0.336	1.000
	9	0.000	0.136	0.992	0.008	0.544	1.000
	10	0.000	0.208	0.996	0.036	0.644	1.000
	11	0.000	0.444	1.000	0.120	0.856	1.000
	12	0.008	0.492	1.000	0.184	0.896	1.000
	13	0.016	0.684	1.000	0.316	0.976	1.000
	14	0.008	0.800	1.000	0.448	0.984	1.000
	15	0.096	0.832	1.000	0.552	0.984	1.000

Table 6: LASSO experiment test losses plotted in Figure 8: mean, 10^{th} quantile, median, and 90^{th} quantile of $f(x^K) - f(x^\star)$ over the in-distribution and out-of-distribution test sets for every horizon K.

		In-distribution				Out-of-distribution			
Framework	K	Mean	10^{th}	Median	90^{th}	Mean	10^{th}	Median	90^{th}
L2O-ISTA	1	2.17e1	1.60e1	2.10e1	2.85e1	4.87e1	3.36e1	4.91e1	6.37e1
	2	1.03e1	7.23e0	9.73e0	1.44e1	2.85e1	1.73e1	2.79e1	4.00e1
	3	6.49e0	4.23e0	6.08e0	9.29e0	2.27e1	1.14e1	2.02e1	3.59e1
	4	4.26e0	2.43e0	3.92e0	6.39e0	2.43e1	7.90e0	1.77e1	4.84e1
	5	2.89e0	1.45e0	2.57e0	4.45e0	3.37e1	5.58e0	1.62e1	7.55e1
	6	1.92e0	8.53e-1	1.74e0	3.10e0	5.51e1	3.96e0	1.25e1	1.33e2
	7	1.30e0	5.12e-1	1.17e0	2.19e0	1.63e2	2.81e0	1.10e1	3.53e2
	8	9.35e-1	3.51e-1	8.80e-1	1.63e0	1.97e2	2.08e0	6.04e0	3.09e2
	9	6.35e-1	1.93e-1	5.86e-1	1.20e0	8.91e2	1.50e0	4.79e0	1.07e3
	10	4.26e-1	1.07e-1	3.82e-1	8.55e-1	5.44e3	9.98e-1	3.50e0	5.86e3
	11	3.04e-1	6.35e-2	2.62e-1	6.07e-1	1.65e4	7.13e-1	2.46e0	1.35e4
	12	2.02e-1	3.54e-2	1.58e-1	4.35e-1	1.21e5	4.24e-1	1.83e0	9.50e4
	13	1.55e-1	2.24e-2	1.16e-1	3.49e-1	3.06e5	3.25e-1	1.38e0	2.08e5
	14	1.12e-1	1.28e-2	7.99e-2	2.67e-1	7.20e5	2.11e-1	9.84e-1	2.93e5
	15	7.85e-2	7.36e-3	5.18e-2	1.91e-1	6.04e6	1.40e-1	7.82e-1	4.03e6
L2O-ALISTA	1	2.20e1	1.61e1	2.11e1	2.94e1	5.04e1	3.37e1	5.09e1	6.63e1
	2	1.33e1	9.83e0	1.29e1	1.75e1	2.98e1	2.11e1	3.00e1	3.91e1
	3	9.73e0	7.08e0	9.49e0	1.29e1	2.20e1	1.58e1	2.22e1	2.84e1
	4	7.62e0	5.35e0	7.39e0	1.02e1	1.79e1	1.28e1	1.82e1	2.31e1
	5	6.11e0	4.16e0	5.96e0	8.27e0	1.51e1	1.05e1	1.52e1	1.97e1
	6	4.96e0	3.22e0	4.80e0	6.81e0	1.29e1	8.82e0	1.29e1	1.70e1
	7	4.06e0	2.51e0	3.92e0	5.57e0	1.11e1	7.30e0	1.11e1	1.50e1
	8	3.33e0	1.97e0	3.22e0	4.64e0	9.69e0	6.19e0	9.59e0	1.32e1
	9	2.75e0	1.55e0	2.65e0	3.93e0	8.46e0	5.28e0	8.33e0	1.17e1
	10	2.27e0	1.22e0	2.19e0	3.37e0	7.41e0	4.55e0	7.25e0	1.05e1
	11	1.88e0	9.42e-1	1.81e0	2.85e0	6.50e0	3.79e0	6.32e0	9.26e0
	12	1.56e0	7.33e-1	1.51e0	2.44e0	5.71e0	3.15e0	5.54e0	8.30e0
	13	1.29e0	5.68e-1	1.25e0	2.10e0	5.01e0	2.74e0	4.88e0	7.37e0
	14	1.07e0	4.47e-1	1.03e0	1.81e0	4.41e0	2.31e0	4.25e0	6.58e0
	15	8.84e-1	3.46e-1	8.51e-1	1.54e0	3.87e0	1.98e0	3.71e0	5.94e0
DR-L2O	1	2.20e1	1.63e1	2.13e1	2.88e1	4.89e1	3.43e1	4.90e1	6.44e1
	2	1.47e1	1.10e1	1.44e1	1.90e1	3.17e1	2.30e1	3.19e1	4.10e1
	3	9.96e0	7.03e0	9.49e0	1.35e1	2.55e1	1.67e1	2.53e1	3.49e1
	4	7.94e0	5.61e0	7.72e0	1.06e1	1.85e1	1.32e1	1.88e1	2.38e1
	5	7.40e0	5.18e0	7.16e0	9.90e0	1.74e1	1.24e1	1.77e1	2.24e1
	6	5.39e0	3.49e0	5.09e0	7.59e0	1.53e1	9.66e0	1.53e1	2.12e1
	7	2.73e0	1.46e0	2.58e0	4.01e0	9.58e0	5.31e0	9.32e0	1.42e1
	8	1.88e0	9.18e-1	1.79e0	2.87e0	6.58e0	3.80e0	6.44e0	9.40e0
	9	2.09e0	1.08e0	1.99e0	3.13e0	7.06e0	4.24e0	6.92e0	1.01e1
	10	1.65e0	7.79e-1	1.58e0	2.61e0	6.34e0	3.48e0	6.15e0	9.52e0
	11	9.03e-1	3.50e-1	8.69e-1	1.59e0	4.06e0	2.04e0	3.90e0	6.33e0
	12	2.06e-1	3.35e-2	1.60e-1	4.40e-1	1.54e0	4.30e-1	1.42e0	2.88e0
	13	9.33e-2	7.82e-3	6.05e-2	2.30e-1	8.75e-1	1.70e-1	7.39e-1	1.74e0
	14	1.96e-1	2.96e-2	1.51e-1	4.31e-1	1.46e0	4.23e-1	1.33e0	2.70e0
	15	8.89e-2	7.44e-3	5.77e-2	2.20e-1	8.22e-1	1.62e-1	7.05e-1	1.62e0
OPT-PEP	1	2.17e1	1.60e1	2.10e1	2.85e1	4.87e1	3.36e1	4.91e1	6.37e1
	2	1.40e1	1.05e1	1.37e1	1.82e1	3.03e1	2.19e1	3.06e1	3.90e1
	3	1.09e1	8.04e0	1.07e1	1.43e1	2.41e1	1.75e1	2.42e1	3.09e1
	4	8.12e0	5.76e0	7.92e0	1.08e1	1.88e1	1.35e1	1.91e1	2.42e1
	5	6.74e0	4.68e0	6.53e0	9.09e0	1.62e1	1.14e1	1.64e1	2.09e1
	6	5.95e0	4.03e0	5.80e0	8.07e0	1.48e1	1.03e1	1.49e1	1.93e1
	7	4.79e0	3.09e0	4.65e0	6.60e0	1.26e1	8.53e0	1.26e1	1.66e1
	8	3.89e0	2.38e0	3.76e0	5.33e0	1.08e1	7.02e0	1.08e1	1.45e1
	9	3.68e0	2.23e0	3.56e0	5.04e0	1.04e1	6.70e0	1.03e1	1.40e1
	10	3.07e0	1.78e0	2.96e0	4.33e0	9.14e0	5.77e0	9.04e0	1.26e1
	11	2.69e0	1.51e0	2.59e0	3.85e0	8.31e0	5.19e0	8.17e0	1.16e1
	12	2.26e0	1.21e0	2.18e0	3.35e0	7.38e0	4.53e0	7.21e0	1.04e1
	13	1.74e0	8.49e-1	1.67e0	2.68e0	6.15e0	3.49e0	5.96e0	8.82e0
	14	1.68e0	8.15e-1	1.62e0	2.60e0	6.00e0	3.39e0	5.80e0	8.63e0
	15	1.31e0	5.82e-1	1.27e0	2.12e0	5.06e0	2.76e0	4.94e0	7.43e0

Table 7: LASSO experiment test losses of backtracking line search plotted in Figure 13: mean, 10^{th} quantile, median, and 90^{th} quantile of $f(x) - f(x^\star)$ over the in-distribution and out-of-distribution test sets. *Iterations:* the iterate x^K after K iterations. *Products:* the last iterate whose cumulative number of matrix-vector products is within the budget, or x^0 if no iteration fits. The learned schedules use two products per iteration, so their values at a budget of $2K$ products are those at horizon K in Table 6.

		In-distribution				Out-of-distribution			
Cost	Budget	Mean	10^{th}	Median	90^{th}	Mean	10^{th}	Median	90^{th}
Iterations	1	2.29e1	1.69e1	2.23e1	2.99e1	5.07e1	3.54e1	5.09e1	6.65e1
	2	1.15e1	8.52e0	1.13e1	1.51e1	2.61e1	1.84e1	2.62e1	3.44e1
	3	6.12e0	3.84e0	5.63e0	9.20e0	1.85e1	1.07e1	1.87e1	2.60e1
	4	3.90e0	1.84e0	3.68e0	6.26e0	1.31e1	7.72e0	1.33e1	1.82e1
	5	2.43e0	9.43e-1	2.23e0	4.15e0	9.68e0	5.00e0	9.92e0	1.42e1
	6	1.46e0	4.67e-1	1.33e0	2.59e0	7.07e0	3.23e0	7.09e0	1.09e1
	7	8.90e-1	2.09e-1	7.63e-1	1.76e0	5.26e0	2.00e0	5.24e0	8.27e0
	8	5.34e-1	8.01e-2	4.27e-1	1.16e0	3.79e0	1.25e0	3.80e0	6.08e0
	9	3.18e-1	2.74e-2	2.27e-1	7.96e-1	2.76e0	7.52e-1	2.65e0	4.78e0
	10	1.86e-1	1.11e-2	1.11e-1	4.88e-1	1.97e0	4.25e-1	1.82e0	3.52e0
	11	1.10e-1	4.06e-3	5.84e-2	2.90e-1	1.41e0	1.94e-1	1.24e0	2.65e0
	12	6.45e-2	1.16e-3	2.78e-2	1.85e-1	9.88e-1	8.48e-2	8.09e-1	1.97e0
	13	3.75e-2	4.30e-4	1.46e-2	1.17e-1	6.88e-1	4.53e-2	5.14e-1	1.45e0
	14	2.26e-2	1.80e-4	7.14e-3	7.18e-2	4.77e-1	1.96e-2	3.09e-1	1.08e0
	15	1.33e-2	5.76e-5	3.38e-3	3.87e-2	3.31e-1	7.98e-3	1.84e-1	8.43e-1
Products	2	7.18e1	4.82e1	6.94e1	9.97e1	1.80e2	1.13e2	1.83e2	2.43e2
	4	2.29e1	1.69e1	2.23e1	2.99e1	5.07e1	3.54e1	5.09e1	6.65e1
	6	2.29e1	1.69e1	2.23e1	2.99e1	5.07e1	3.54e1	5.09e1	6.65e1
	8	1.15e1	8.52e0	1.13e1	1.51e1	2.61e1	1.84e1	2.62e1	3.44e1
	10	6.12e0	3.84e0	5.63e0	9.20e0	2.03e1	1.07e1	1.91e1	3.20e1
	12	4.19e0	1.84e0	3.76e0	6.78e0	1.74e1	8.14e0	1.79e1	2.60e1
	14	2.93e0	9.43e-1	2.38e0	6.12e0	1.28e1	5.72e0	1.33e1	1.82e1
	16	2.04e0	4.67e-1	1.66e0	4.15e0	9.74e0	4.76e0	9.94e0	1.45e1
	18	1.33e0	2.09e-1	1.13e0	2.59e0	7.73e0	3.23e0	7.42e0	1.25e1
	20	8.84e-1	8.35e-2	7.43e-1	1.79e0	6.32e0	2.00e0	6.41e0	1.06e1
	22	5.83e-1	3.00e-2	4.27e-1	1.33e0	4.93e0	1.25e0	4.99e0	8.27e0
	24	3.70e-1	1.23e-2	2.30e-1	9.85e-1	3.70e0	7.52e-1	3.80e0	6.30e0
	26	2.40e-1	4.52e-3	1.26e-1	6.79e-1	2.87e0	4.90e-1	2.67e0	5.07e0
	28	1.54e-1	2.04e-3	6.32e-2	4.83e-1	2.26e0	2.55e-1	1.87e0	4.44e0
	30	9.88e-2	9.65e-4	3.50e-2	2.86e-1	1.77e0	1.30e-1	1.46e0	3.50e0

Table 8: Fractions of LASSO problems solved plotted in Figure 4, at each relative tolerance η, on the in-distribution and out-of-distribution test sets for every horizon K.

		In-distribution			Out-of-distribution		
Framework	K	$\eta = 10^{-2}$	$\eta = 5 \cdot 10^{-2}$	$\eta = 10^{-1}$	$\eta = 10^{-2}$	$\eta = 5 \cdot 10^{-2}$	$\eta = 10^{-1}$
L2O-ISTA	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.000
	4	0.000	0.000	0.073	0.000	0.000	0.000
	5	0.000	0.028	0.589	0.000	0.000	0.004
	6	0.000	0.286	0.911	0.000	0.004	0.056
	7	0.004	0.690	0.980	0.000	0.012	0.200
	8	0.036	0.899	1.000	0.004	0.044	0.380
	9	0.137	0.996	1.000	0.004	0.168	0.472
	10	0.351	1.000	1.000	0.008	0.352	0.528
	11	0.540	1.000	1.000	0.012	0.460	0.600
	12	0.790	1.000	1.000	0.068	0.532	0.600
	13	0.883	1.000	1.000	0.104	0.600	0.648
	14	0.960	1.000	1.000	0.160	0.680	0.696
	15	0.996	1.000	1.000	0.300	0.640	0.652
L2O-ALISTA	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.000
	4	0.000	0.000	0.000	0.000	0.000	0.000
	5	0.000	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.004	0.000	0.000	0.000
	7	0.000	0.000	0.056	0.000	0.000	0.000
	8	0.000	0.000	0.246	0.000	0.000	0.000
	9	0.000	0.016	0.621	0.000	0.000	0.004
	10	0.000	0.077	0.855	0.000	0.000	0.020
	11	0.000	0.210	0.968	0.000	0.004	0.064
	12	0.000	0.440	0.996	0.000	0.004	0.180
	13	0.000	0.653	1.000	0.000	0.012	0.340
	14	0.008	0.843	1.000	0.000	0.028	0.548
	15	0.036	0.940	1.000	0.000	0.060	0.720
DR-L2O	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.000
	4	0.000	0.000	0.000	0.000	0.000	0.000
	5	0.000	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.008	0.000	0.000	0.000
	7	0.000	0.024	0.621	0.000	0.000	0.004
	8	0.000	0.222	0.968	0.000	0.004	0.064
	9	0.000	0.137	0.911	0.000	0.000	0.032
	10	0.000	0.359	0.984	0.000	0.004	0.120
	11	0.036	0.935	1.000	0.000	0.052	0.684
	12	0.794	1.000	1.000	0.060	0.864	1.000
	13	0.980	1.000	1.000	0.252	0.988	1.000
	14	0.798	1.000	1.000	0.072	0.884	1.000
	15	0.992	1.000	1.000	0.268	0.988	1.000
OPT-PEP	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.000
	4	0.000	0.000	0.000	0.000	0.000	0.000
	5	0.000	0.000	0.000	0.000	0.000	0.000
	6	0.000	0.000	0.000	0.000	0.000	0.000
	7	0.000	0.000	0.012	0.000	0.000	0.000
	8	0.000	0.000	0.081	0.000	0.000	0.000
	9	0.000	0.000	0.137	0.000	0.000	0.000
	10	0.000	0.004	0.411	0.000	0.000	0.000
	11	0.000	0.016	0.645	0.000	0.000	0.004
	12	0.000	0.077	0.855	0.000	0.000	0.020
	13	0.000	0.302	0.984	0.000	0.004	0.104
	14	0.000	0.331	0.984	0.000	0.004	0.128
	15	0.000	0.641	1.000	0.000	0.012	0.324

Table 9: TV inpainting experiment test losses plotted in Figure 9: mean, 10^{th} quantile, median, and 90^{th} quantile of the Lagrangian duality gap over the Olivetti in-distribution and Tiny ImageNet out-of-distribution test sets for every horizon K.

		In-distribution				Out-of-distribution			
Framework	K	Mean	10^{th}	Median	90^{th}	Mean	10^{th}	Median	90^{th}
L2O	1	8.93e1	8.07e1	8.66e1	9.92e1	4.47e2	3.14e2	4.23e2	5.84e2
	2	7.05e1	6.29e1	6.86e1	7.86e1	3.54e2	2.38e2	3.30e2	4.78e2
	3	4.00e1	3.55e1	3.90e1	4.47e1	2.17e2	1.34e2	1.97e2	3.20e2
	4	1.57e1	1.29e1	1.57e1	1.84e1	1.06e2	4.53e1	8.53e1	1.97e2
	5	6.00e0	5.06e0	5.96e0	7.33e0	7.33e1	2.91e1	5.66e1	1.43e2
	6	4.42e0	3.73e0	4.40e0	5.39e0	7.22e1	2.42e1	5.15e1	1.51e2
	7	5.62e0	4.79e0	5.62e0	6.72e0	7.37e1	2.81e1	5.56e1	1.48e2
	8	3.33e0	2.94e0	3.37e0	3.66e0	5.87e1	1.81e1	3.95e1	1.29e2
	9	3.99e0	3.51e0	4.02e0	4.37e0	5.80e1	2.00e1	4.15e1	1.22e2
	10	8.25e0	7.06e0	8.31e0	9.52e0	7.49e1	3.49e1	6.19e1	1.33e2
DR-L2O	1	9.93e1	9.21e1	9.74e1	1.08e2	5.17e2	3.66e2	4.96e2	6.72e2
	2	6.93e1	5.94e1	6.81e1	7.94e1	3.29e2	2.02e2	2.85e2	4.47e2
	3	4.49e1	3.41e1	4.51e1	5.54e1	1.95e2	7.18e1	1.49e2	3.23e2
	4	1.96e1	1.66e1	1.95e1	2.26e1	1.22e2	5.79e1	1.00e2	2.14e2
	5	8.11e0	6.72e0	8.02e0	1.01e1	9.08e1	3.87e1	7.16e1	1.67e2
	6	6.56e0	5.61e0	6.56e0	7.70e0	7.45e1	3.11e1	5.84e1	1.45e2
	7	5.24e0	4.59e0	5.28e0	5.80e0	6.33e1	2.48e1	4.81e1	1.27e2
	8	4.21e0	3.76e0	4.24e0	4.61e0	5.33e1	2.15e1	4.08e1	1.06e2
	9	3.37e0	2.99e0	3.41e0	3.70e0	4.47e1	1.92e1	3.48e1	8.63e1
	10	3.03e0	2.65e0	3.06e0	3.35e0	3.86e1	1.71e1	3.11e1	7.11e1
OPT-PEP	1	1.10e2	1.03e2	1.08e2	1.18e2	5.88e2	4.16e2	5.45e2	7.58e2
	2	8.96e1	8.32e1	8.79e1	9.69e1	4.77e2	3.37e2	4.50e2	6.29e2
	3	6.89e1	6.23e1	6.69e1	7.64e1	3.59e2	2.49e2	3.46e2	4.89e2
	4	4.93e1	4.50e1	4.82e1	5.42e1	2.76e2	1.87e2	2.60e2	3.96e2
	5	3.34e1	2.96e1	3.26e1	3.74e1	1.90e2	1.14e2	1.71e2	2.86e2
	6	2.29e1	2.07e1	2.23e1	2.52e1	1.46e2	8.31e1	1.23e2	2.40e2
	7	1.24e1	1.13e1	1.24e1	1.37e1	1.14e2	5.91e1	8.84e1	1.93e2
	8	8.76e0	7.11e0	8.60e0	1.10e1	9.93e1	4.23e1	8.04e1	1.77e2
	9	7.80e0	6.56e0	7.74e0	9.58e0	8.73e1	3.71e1	6.88e1	1.64e2
	10	7.10e0	6.04e0	7.10e0	8.48e0	7.96e1	3.35e1	6.26e1	1.53e2

Table 10: Fractions of TV inpainting problems solved plotted in Figure 10, at each relative tolerance η, on the Olivetti in-distribution and Tiny ImageNet out-of-distribution sets for every horizon K.

		In-distribution			Out-of-distribution		
Framework	K	$\eta = 10^{-2}$	$\eta = 5 \cdot 10^{-2}$	$\eta = 10^{-1}$	$\eta = 10^{-2}$	$\eta = 5 \cdot 10^{-2}$	$\eta = 10^{-1}$
L2O	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.050
	3	0.000	0.000	0.025	0.000	0.000	0.475
	4	0.000	0.338	0.900	0.000	0.575	0.825
	5	0.000	1.000	1.000	0.000	0.800	0.975
	6	0.000	1.000	1.000	0.000	0.800	0.975
	7	0.000	1.000	1.000	0.000	0.800	0.975
	8	0.000	1.000	1.000	0.075	0.800	0.975
	9	0.000	1.000	1.000	0.025	0.800	0.975
	10	0.000	1.000	1.000	0.000	0.800	0.975
DR-L2O	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.200
	3	0.000	0.000	0.087	0.000	0.300	0.500
	4	0.000	0.113	0.800	0.000	0.450	0.800
	5	0.000	1.000	1.000	0.000	0.700	0.975
	6	0.000	1.000	1.000	0.000	0.800	0.975
	7	0.000	1.000	1.000	0.000	0.800	0.975
	8	0.000	1.000	1.000	0.000	0.900	0.975
	9	0.000	1.000	1.000	0.025	0.975	1.000
	10	0.188	1.000	1.000	0.050	0.975	1.000
OPT-PEP	1	0.000	0.000	0.000	0.000	0.000	0.000
	2	0.000	0.000	0.000	0.000	0.000	0.000
	3	0.000	0.000	0.000	0.000	0.000	0.000
	4	0.000	0.000	0.000	0.000	0.000	0.300
	5	0.000	0.000	0.263	0.000	0.075	0.575
	6	0.000	0.000	0.613	0.000	0.275	0.700
	7	0.000	0.600	1.000	0.000	0.525	0.875
	8	0.000	1.000	1.000	0.000	0.650	0.925
	9	0.000	1.000	1.000	0.000	0.750	0.975
	10	0.000	1.000	1.000	0.000	0.800	0.975