Distributionally Robust Optimization via Ball Oracle Acceleration

Yair Carmon
Tel Aviv University
ycarmon@tauex.tau.ac.il

Danielle Hausler
Tel Aviv University
hausler@mail.tau.ac.il

Abstract

We develop and analyze algorithms for distributionally robust optimization (DRO) of convex losses. In particular, we consider group-structured and bounded f-divergence uncertainty sets. Our approach relies on an accelerated method that queries a ball optimization oracle, i.e., a subroutine that minimizes the objective within a small ball around the query point. Our main contribution is efficient implementations of this oracle for DRO objectives. For DRO with N non-smooth loss functions, the resulting algorithms find an ϵ-accurate solution with $\widetilde{O}\left(N\epsilon^{-2/3} + \epsilon^{-2}\right)$ first-order oracle queries to individual loss functions. Compared to existing algorithms for this problem, we improve complexity by a factor of up to $\epsilon^{-4/3}$.

1 Introduction

The increasing use of machine learning models in high-stakes applications highlights the importance of reliable performance across changing domains and populations [11, 46, 35]. An emerging body of research addresses this challenge by replacing Empirical Risk Minimization (ERM) with Distributionally Robust Optimization (DRO) [6, 50, 49, 22, 40], with applications in natural language processing [47, 61, 35], reinforcement learning [17, 54] and algorithmic fairness [27, 56]. While ERM minimizes the average training loss, DRO minimizes the worst-case expected loss over all probability distributions in an *uncertainty set* $\mathcal{U}$, that is, it minimizes

$$\mathcal{L}_{\text{DRO}}(x) \coloneqq \sup_{Q\in\mathcal{U}} \mathbb{E}_{S\sim Q}[\ell_S(x)], \tag{1}$$

where $\ell_S(x)$ is the loss a model $x \in \mathcal{X}$ incurs on a sample S and $\mathcal{X}$ is a closed convex set with bounded Euclidean diameter. This work develops new algorithms for DRO, focusing on formulations where $\mathcal{U}$ contains distributions supported on N training points, where N is potentially large. We consider two well-studied DRO variants: (1) Group DRO [57, 31, 49], and (2) f-divergence DRO [19, 6, 22].

Group DRO Machine learning models may rely on spurious correlations (that hold for most training examples but are wrongly linked to the target) and therefore suffer high loss on minority groups where these correlations do not hold [30, 27, 11]. To obtain high performance across all groups, Group DRO minimizes the worst-case loss over groups. Given a set $\mathcal{U} = \{w_1, \ldots, w_M\}$ of M distributions over $[N]$, the Group DRO objective is[1]

$$\mathcal{L}_{\text{g-DRO}}(x) \coloneqq \max_{i\in[M]} \mathbb{E}_{j\sim w_i}\ell_j(x) = \max_{i\in[M]} \sum_{j=1}^{N} w_{ij}\ell_j(x). \tag{2}$$

[1]Typically, each w_i is uniform over a subset ("group") of the N training points. However, most approaches (and ours included) extend to the setting of arbitrary w_i's, which was previously considered in [57].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Smoothness	Method	Group DRO (2)	f-divergence DRO (3)
None ($L=\infty$)	Subgradient method [45]	$N\epsilon^{-2}$	$N\epsilon^{-2}$
	Stoch. primal-dual [43] *	$M\epsilon^{-2}$	$N\epsilon^{-2}$
	MLMC stoch. gradient [39]	-	$\rho\epsilon^{-3}$ or $\alpha^{-1}\epsilon^{-2}$ †
	Ours	$N\epsilon^{-2/3}+\epsilon^{-2}$	$N\epsilon^{-2/3}+\epsilon^{-2}$
Weak ($L\approx 1/\epsilon$)	AGD on softmax [44]	$N\epsilon^{-1}$	$N\epsilon^{-1}$
	Ours	$N\epsilon^{-2/3}+N^{3/4}\epsilon^{-1}$	$N\epsilon^{-2/3}+\sqrt{N}\epsilon^{-1}$

Table 1. Number of $\nabla\ell_i$ and ℓ_i evaluations to obtain $\mathbb{E}[\mathcal{L}_{\text{DRO}}(x)]-\min_{x_\star\in\mathcal{X}}\mathcal{L}_{\text{DRO}}(x_\star)\le\epsilon$, where N is the number of training points and (for Group DRO) M is the number of groups. The stated rates omit constant and polylogarithmic factors. * Requires an additional uniform bound on losses (see Appendix A.1). † These rates hold only for specific f-divergences: CVaR at level α or χ^2-divergence with size ρ, respectively.

If we define the loss of group i as $\mathcal{L}_i(x) := \sum_{j=1}^N w_{ij}\ell_j(x)$ then objective (2) is equivalent to $\max_{q\in\Delta^M}\sum_{i\in[M]} q_i\mathcal{L}_i(x)$ with $\Delta^M := \{q\in\mathbb{R}^M_{\ge 0} \mid \vec{1}^T q = 1\}$. Note that, unlike ERM, Group DRO requires additional supervision in the form of subgroup identities encoded by $\{w_i\}$.

DRO with f-divergence Another approach to DRO, which requires only as much supervision as ERM, takes $\mathcal{U}$ to be an f-divergence ball around the empirical (training) distribution. For every convex function $f:\mathbb{R}_+\to\mathbb{R}\cup\{+\infty\}$ such that $f(1)=0$, $f(0/0)=0$ and the f-divergence between distributions q and p over $[N]$ is $D_f(q,p) := \sum_{i\in[N]} p_i f(q_i/p_i)$. The f-divergence DRO problem corresponds to the uncertainty set $\mathcal{U}=\{q\in\Delta^N : D_f(q,\frac{1}{N}\mathbf{1}))\le 1\}$, i.e.,

$$\mathcal{L}_{f\text{-div}}(x) := \max_{q\in\Delta^N : \frac{1}{N}\sum_{i\in[N]} f(Nq_i)\le 1} \sum_{i\in[N]} q_i\ell_i(x). \tag{3}$$

Several well-studied instances of DRO are a special case of this formulation, with the two most notable examples being conditional value at risk (CVaR) and χ^2 uncertainty sets. CVaR at level α corresponds to $f(x)=\mathbb{1}_{\{x<\frac{1}{\alpha}\}}$ such that $\mathcal{U}=\left\{q\in\Delta^N \text{ s.t } \|q\|_\infty\le 1/(\alpha N)\right\}$, and has many applications in finance such as portfolio optimization and credit risk evaluation [48, 37] as well as in machine learning [47, 39, 20, 60, 17, 54]. The χ^2 uncertainty set with size $\rho>0$ corresponds to $f(x) := \frac{1}{2\rho}(x-1)^2$ and the resulting DRO problem is closely linked to variance regularization [21] and has been extensively studied in statistics and machine learning [42, 27, 21, 39, 61].

Complexity notion In this paper, we design improved-complexity methods for solving the convex problems (2) and (3) under the assumption that the loss ℓ_i is convex and Lipschitz for all i. We measure complexity by the (expected) required number of $\ell_i(x)$ and $\nabla\ell_i(x)$ evaluations to obtain ϵ-suboptimal solution, i.e., return x such that $\mathcal{L}_{\text{DRO}}(x)-\min_{x_\star\in\mathcal{X}}\mathcal{L}_{\text{DRO}}(x_\star)\le\epsilon$ with constant probability. Table 1 summarizes our complexity bounds and compares them to prior art. Throughout the introduction we assume (for simplicity) a unit domain size ($\|x-y\|\le 1$ for all $x,y\in\mathcal{X}$) and that each loss is 1-Lipschitz.

Prior art Let us review existing methods that solve Group DRO and f-divergence DRO problems (see Section 1.1 for extended discussion). For a dataset with N training points, the subgradient method [45] finds an ϵ approximate solution in ϵ^{-2} iterations. Computing a single subgradient costs N functions evaluations (since we need to find the maximizing q). Therefore, the complexity of this method is $O(N\epsilon^{-2})$.

DRO can also be viewed as a game between a minimizing x-player and a maximizing q-player, which makes it amenable to primal-dual methods [43, 42, 49]. If we further assume that the losses are bounded then, for $q\in\Delta^m$, stochastic mirror descent with local norms obtains a regret bound of $O(\sqrt{m\log(m)/T})$ (see Appendix A.1). As a consequence, for Group DRO (where $m=M$) the complexity is $\widetilde{O}(M\epsilon^{-2})$, and for f-divergence DRO ($m=N$) the complexity is $\widetilde{O}(N\epsilon^{-2})$.

Levy et al. [39] studied χ^2-divergence and CVaR DRO problems, and proposed using standard gradient methods with a gradient estimator based on multilevel Monte Carlo (MLMC) [9]. For χ^2-divergence with ball of size ρ they proved a complexity bound of $\widetilde{O}(\rho\epsilon^{-3})$, and for CVaR at level α they established complexity $\widetilde{O}\big(\alpha^{-1}\epsilon^{-2}\big)$. However, for large uncertainty sets (when ρ or α^{-1} approach N) their method does not improve over the subgradient method.

Stronger complexity bounds are available under the weak smoothness assumption that each ℓ_i has $O\big(\epsilon^{-1}\big)$-Lipschitz gradient. Note that this is a weak assumption since if a function ℓ is not continuously differentiable, it is possible to approximate ℓ with additive error at most $\epsilon/2$, by its Moreau envelope $\widetilde{\ell}(x) = \min_{y\in\mathcal{X}}\Big\{\ell(y) + \frac{G^2}{2\epsilon}\|x-y\|^2\Big\}$ (see [15, Appendix A.1] for more details). In particular, we can apply Nesterov's accelerated gradient descent method [44] on an entropy-regularized version of our objective to solve the problem with complexity $\widetilde{O}\big(N\epsilon^{-1}\big)$; see Appendix A.2 for more details.

Our contribution We propose algorithms that solve the problems (2) and (3) with complexity $\widetilde{O}\big(N\epsilon^{-2/3} + \epsilon^{-2}\big)$. Compared to previous works, we obtain better convergence rates for DRO with general f-divergence when $N \gg 1$ and for Group DRO when $M \gg N\epsilon^{4/3}$. When the losses have $O(\epsilon^{-1})$-Lipschitz gradient, we solve f-divergence DRO with complexity $\widetilde{O}\big(N\epsilon^{-2/3} + \sqrt{N}\epsilon^{-1}\big)$, and, under an even weaker mean-square smoothness assumption ($\mathbb{E}_{j\sim w_i}\|\nabla\ell_j(x) - \nabla\ell_j(y)\|^2 \le O(\epsilon^{-2})\|x-y\|^2$ for all x, y and i), we solve Group DRO with complexity $\widetilde{O}\big(N\epsilon^{-2/3} + N^{3/4}\epsilon^{-1}\big)$.

Our complexity bounds are independent of the structure of f and $\{w_i\}$, allowing us to consider arbitrarily f-divergence balls and support a large number of (potentially overlapping) groups. Our rates are optimal up to logarithmic factors for the special case of minimizing $\max_{i\in[N]}\ell_i(x)$, which corresponds to Group DRO with N distinct groups and f-divergence DRO with $f = 0$ [58, 62, 15].

Our approach Our algorithms are based on a technique for acceleration with a *ball optimization oracle*, introduced by Carmon et al. [13] and further developed in [15, 3]. Given a function F and a query point x, the ball optimization oracle returns an approximate minimizer of F inside a ball around x with radius r; the works [13, 15, 3] show how to minimize F using $\widetilde{O}(r^{-2/3})$ oracle calls. Our development consists of efficiently implementing ball oracles with radius $r = \widetilde{O}(\epsilon)$ for the DRO problems (2) and (3), leveraging the small ball constraint to apply stochastic gradient estimators that would have exponential variance and/or cost without it.

Carmon et al. [15] previously executed this strategy for minimizing the maximum loss, i.e., $\max_{q\in\Delta^N}\sum_i q_i\ell_i(x)$, which is a special case of both Group DRO and f-divergence DRO. However, the ball-oracle implementations of [15] do not directly apply to the DRO problems that we consider; our oracle implementations differ significantly and intimately rely on the Group DRO and f-divergence problem structure. We now briefly review the main differences between our approach and [15], highlighting our key technical innovations along the way.

Since the Group DRO objective is $\max_{q\in\Delta^M}\sum_i q_i\mathcal{L}_i(x)$ for $\mathcal{L}_i(x) = \sum_{j\in[N]} w_{ij}\ell_j(x)$, one may naively apply the technique of [15] with $\mathcal{L}_i$ replacing ℓ_i. However, every step of such a method would involve computing quantities of the form $e^{\mathcal{L}_i(x)/\epsilon'}$ (for some $\epsilon' = \widetilde{\Theta}(\epsilon)$), which can be up to N times more expensive than computing $e^{\ell_j(x)/\epsilon'}$ for a single j. To avoid such expensive computation we use MLMC [9] to obtain an unbiased estimate of $e^{\mathcal{L}_i(x)/\epsilon}$ with complexity $O(1)$ and appropriately bounded variance. In the weakly-smooth case we also adapt our estimator to facilitate variance reduction [33, 2].

For f-divergence we consider the well-known dual form [6, 50]

$$\max_{q\in\Delta^N}\sum_{i\in[N]}\{q_i\ell_i(x) - \psi(q_i)\} = \min_{y\in\mathbb{R}}\left\{\Upsilon(x,y) \coloneqq \sum_{i\in[N]}\psi^*(\ell_i(x) - y) + y\right\}$$

where $\psi^*(v) = \max_{t\ge 0}\{vt - \psi(t)\}$ is the Fenchel dual of ψ. Stochastic gradient methods applied directly on the dual formulation are notoriously unstable (see e.g., [42]). This is due to the fact that the Fenchel dual ψ^* can be very badly behaved even for standard f-divergences. We solve this problem by (a) considering a small ball, (b) entropy-regularizing ψ. Our techniques rely on two technical observations: (i) $\log({\psi_\epsilon^*}'(\cdot))$ is $1/\epsilon'$-Lipschitz for all convex ψ, and (ii) for 1-Lipschitz losses, $y^\star(x) = \operatorname{argmin}_{y\in\mathbb{R}}\Upsilon(x,y)$ satisfies $|y^\star(x) - y^\star(x')| \le \|x - x'\|$ for all x, x'. To the best of our knowledge, these observations are new and potentially of independent interest.

Paper organization. Section 2 provides notation and a concise summary of the ball acceleration framework (largely taken from prior work) on which we build our algorithms. Sections 3 and 4 present our main contributions in the Group and f-divergence DRO settings, respectively. Finally, Section 5 concludes with discussion on the limitations and possible extensions of our work.

1.1 Additional related work

MLMC estimators The multilevel Monte Carlo (MLMC) technique was introduced by Giles [26] and Heinrich [29] in order to reduce the computational cost of Monte Carlo estimation of integrals. Blanchet and Glynn [9] extended this technique to estimating functions of expectation and proposed several applications, including stochastic optimization [10]. In this work we use their estimator for two distinct purposes: (1) obtaining unbiased Moreau envelope gradient estimates for ball oracle acceleration as proposed by Asi et al. [3], and (2) estimating the exponential of an expectation for Group DRO. Levy et al. [39] also rely on MLMC for DRO, but quite differently than we do: they directly estimate the DRO objective gradient via MLMC, while we estimate different quantities.

Other DRO methods Several additional works proposed algorithms with theoretical guarantees for f-divergence DRO. Jin et al. [32] considered non-convex and smooth losses. Song et al. [53] proposed an algorithm for linear models with complexity comparable to the "AGD on softmax" approach (Appendix A.2). Namkoong and Duchi [42] proposed a primal-dual algorithm that is suitable for small uncertainty χ^2 sets (with size $\rho \ll \frac{1}{N}$) and Curi et al. [20] proposed a primal-dual algorithm specialized for CVaR. Other works consider DRO with uncertainty sets defined using the Wasserstein distance [24, 23, 51, 34]. Another relevant line of works proposes refinements for DRO that address some of the challenges in applying it to learning problems [60, 59, 55].

2 Preliminaries

Notation We write $\|\cdot\|$ for the Euclidean norm. We denote by $\mathbb{B}_r(x_0)$ the Euclidean ball of radius r around x_0. We let $\Delta^n := \{q \in \mathbb{R}^n_{\geq 0} \mid \mathbf{1}^T q = 1\}$ denote the probability simplex in $\mathbb{R}^n$. For the sequence $z_m, \ldots, z_n$ we use the shorthand z_m^n. Using F as a generic placeholder (typically for a loss function ℓ_i), we make frequent use of the following assumption.

Assumption 1. *The function $F : \mathcal{X} \to \mathbb{R}$ is convex and G-Lipschitz, i.e., for all $x, y \in \mathcal{X}$ we have $|F(x) - F(y)| \leq G\|x - y\|$. In addition, the domain $\mathcal{X}$ is a closed and convex set, and it has Euclidean diameter at most R.*

Throughout, N denotes the number of losses and, in Section 3, M denotes the number of groups. We use ϵ for our target accuracy and $r_\epsilon := \epsilon'/G$ for the ball radius, where $\epsilon' = \epsilon/(2\log M)$ for Group-DRO (Section 3) and $\epsilon' = \epsilon/(2\log N)$ for f-divergence DRO (Section 4).

Complexity model We measure an algorithm's complexity by its *expected* number of ℓ_i and $\nabla \ell_i$ evaluations; bounds on expected evaluation number can be readily converted to more standard probability 1 bounds [see 3, Appendix A.3]. Moreover if $\mathcal{X} \subset \mathbb{R}^d$, $d = \Omega(\log N)$,[2] and the time to evaluate ℓ_i and $\nabla \ell_i$ is $O(d)$, the expected runtime of all the algorithms we consider is at most d times the evaluation complexity.

2.1 Ball oracle acceleration

We now briefly summarize the complexity bounds given by the framework of [13, 15, 3] for accelerated minimization using queries to (inexact) ball optimization oracles, defined as follows.

Definition 1. *An algorithm is a Ball Regularized Optimization Oracle of radius r (r-BROO) for function $F : \mathcal{X} \to \mathbb{R}$ if for query point $\bar{x} \in \mathcal{X}$, regularization parameter $\lambda > 0$ and desired accuracy $\delta > 0$ it returns $\mathcal{O}_{\lambda,\delta}(\bar{x}) \in \mathcal{X}$ satisfying*

$$\mathbb{E}\left[F(\mathcal{O}_{\lambda,\delta}(\bar{x})) + \frac{\lambda}{2}\|\mathcal{O}_{\lambda,\delta}(\bar{x}) - \bar{x}\|^2\right] \leq \min_{x \in \mathbb{B}_r(\bar{x}) \cap \mathcal{X}} \left\{F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2\right\} + \frac{\lambda}{2}\delta^2. \tag{4}$$

Proposition 1. *Let F satisfy Assumption 1, let $\mathcal{C}_F$ be the complexity of evaluating F exactly, and let $\mathcal{C}_\lambda(\delta)$ bound the complexity of an r-BROO query with δ, λ. Assume that $\mathcal{C}_\lambda(\delta)$ is non-increasing in λ and at most polynomial in $1/\delta$. For any $\epsilon > 0$, Algorithm 1 returns x such that*

[2] The assumption $d = \Omega(\log N)$ is only necessary for our results on f-divergence DRO (Section 4), where the runtime of computing $\operatorname{argmin}_{y \in \mathbb{R}} \Upsilon(x, y)$ is $O(Nd + N\log N)$ due to the need to sort the losses.

$F(x) - \min_{x_\star \in \mathcal{X}} F(x_\star) \le \epsilon$ *with probability at least* $\frac{1}{2}$. *For* $m_\epsilon = O\big(\log \frac{GR^2}{\epsilon r}\big)$ *and* $\lambda_{\mathrm{m}} = O\big(\frac{m_\epsilon^2 \epsilon}{r^{4/3} R^{2/3}}\big)$, *the complexity of the algorithm is*

$$O\left(\left(\frac{R}{r}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon} \frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r}{2^{j/2} m_\epsilon^2}\right)\right) m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r) + \mathcal{C}_F) m_\epsilon^3,\right]\right). \tag{5}$$

Informally, the proposition shows that $\widetilde{O}((R/r)^{2/3})$ BROO calls with $\lambda = \widetilde{\Omega}(\epsilon/(r^{3/4}R^{2/3}))$ and accuracy $\delta = \widetilde{O}(r)$ suffice to find an ϵ-accurate solution. As we show in the sequel, for $\mathcal{C}_\lambda(\delta) = \widetilde{O}\big(N + (\frac{G}{\lambda\delta})^2\big)$ the resulting complexity bound is $\widetilde{O}\big(N(\frac{GR}{\epsilon})^{2/3} + (\frac{GR}{\epsilon})^2\big)$. The summation over j in bound (5) stems from the use of MLMC to de-bias the BROO output (i.e., make it exact in expectation): compared to the original proposal of Asi et al. [3], our version of the procedure in Appendix B slightly alters this MLMC scheme by de-biasing one accurate BROO call instead of averaging many inaccurate de-biased calls, improving our bounds by logarithmic factors.

3 Group DRO

In this section we develop our BROO implementations for the Group DRO objective (2). In Section 3.1 we describe an "exponentiated group-softmax" function that approximates $\mathcal{L}_{\text{g-DRO}}$ with additive error at most $\epsilon/2$. We then apply stochastic gradient methods on this function to obtain BROO implementations that yield improved rates for Group DRO via Proposition 1: we first consider the non-smooth case in Section 3.2 and then the weakly-smooth case in Section 3.3.

3.1 Exponentiated group-softmax

Given a cheap and unbiased stochastic gradient estimator of $\nabla\mathcal{L}_{\text{g-DRO}}$, we could use a variant of SGD and minimize $\mathcal{L}_{\text{g-DRO}}$ to ϵ-suboptimal solution using $O(\epsilon^{-2})$ steps. However, obtaining an unbiased estimator is challenging due to the maximum operator in $\mathcal{L}_{\text{g-DRO}}$. As a first step we use entropy smoothing [7, 8, 5, 4] to replace the maximum in $\mathcal{L}_{\text{g-DRO}}$ with the softmax operation. More specifically, we use the trick from [15] and minimize the "exponentiated softmax" (that has the form of a weighted finite sum) within a small ball. For target accuracy ϵ, regularization parameter $\lambda \ge 0$, center point $\bar{x} \in \mathcal{X}$ and $\epsilon' = \epsilon/(2\log M) > 0$, the (regularized) group-softmax function is

$$\mathcal{L}_{\mathrm{smax},\epsilon,\lambda}(x) \coloneqq \epsilon' \log\left(\sum_{i\in[M]} e^{\frac{\mathcal{L}_i(x)}{\epsilon'}}\right) + \frac{\lambda}{2}\|x-\bar{x}\|^2 \ \text{ where } \ \mathcal{L}_i(x) = \sum_{j\in[N]} w_{ij}\ell_j(x). \tag{6}$$

We will implement a BROO for $\mathcal{L}_{\mathrm{smax},\epsilon} \coloneqq \mathcal{L}_{\mathrm{smax},\epsilon,0}$, which is a uniform approximation of $\mathcal{L}_{\text{g-DRO}}$: $|\mathcal{L}_{\text{g-DRO}}(x) - \mathcal{L}_{\mathrm{smax},\epsilon}(x)| \le \epsilon/2$ for all $x \in \mathcal{X}$; see Appendix C.1 for details.

The (regularized) exponentiated group-softmax is

$$\Gamma_{\epsilon,\lambda}(x) \coloneqq \sum_{i\in[M]} \bar{p}_i\gamma_i(x) \ \text{ where } \ \gamma_i(x) = \epsilon' e^{\frac{\mathcal{L}_i(x)-\mathcal{L}_i(\bar{x})+\frac{\lambda}{2}\|x-\bar{x}\|^2}{\epsilon'}} \ \text{ and } \ \bar{p}_i = \frac{e^{\frac{\mathcal{L}_i(\bar{x})}{\epsilon'}}}{\sum_{i\in[M]} e^{\frac{\mathcal{L}_i(\bar{x})}{\epsilon'}}}. \tag{7}$$

In the following lemma we (easily) extend Carmon et al. [15, Lemma 1] to exponentiated group-softmax, showing that $\Gamma_{\epsilon,\lambda}$ is well-behaved inside a ball of (appropriately small) radius r around $\bar{x}$ and facilitates minimizing $\mathcal{L}_{\mathrm{smax},\epsilon,\lambda}$ in that ball; see Appendix C.1 for the proof.

Lemma 1. *Let each* ℓ_i *satisfy Assumption 1, and consider the restriction of* $\mathcal{L}_{\mathrm{smax},\epsilon,\lambda}$ (6) *and* $\Gamma_{\epsilon,\lambda}$ (7) *to* $\mathbb{B}_r(\bar{x})$. *Then the functions have the same minimizer* $x_\star \in \mathbb{B}_r(\bar{x})$ *and, if* $\lambda \le O(G/r)$ *and* $r \le O(\epsilon'/G)$, *then (a)* $\Gamma_{\epsilon,\lambda}$ *is* $\Omega(\lambda)$*-strongly convex, (b) each* γ_i *is* $O(G)$*-Lipschitz and (c) for every* $x \in \mathbb{B}_r(\bar{x})$ *we have* $\mathcal{L}_{\mathrm{smax},\epsilon,\lambda}(x) - \mathcal{L}_{\mathrm{smax},\epsilon,\lambda}(x_\star) \le O(\Gamma_{\epsilon,\lambda}(x) - \Gamma_{\epsilon,\lambda}(x_\star))$.

3.2 BROO implementation for Group DRO non-smooth losses

To motivate our BROO implementation, let us review how [15] use the exponentiated softmax in the special case of size-1 groups, i.e., $\mathcal{L}_i = \ell_i$, and explain the difficulty that their approach faces when the group structure is introduced. The BROO implementation in [15] is based on SGD variant with the stochastic gradient estimator $\hat{g}(x) = e^{(\mathcal{L}_i(x)-\mathcal{L}_i(\bar{x}))/\epsilon'}\nabla\mathcal{L}_i(x)$ where $i \sim \bar{p}_i$. However, for

Group DRO where $\mathcal{L}_i = \sum_{j\in[N]} w_{ij}\ell_j$, the estimator $\hat{g}(x)$ can be up to N times more expensive to compute. Approximating $\hat{g}(x)$ by drawing $j, j' \sim w_i$ and taking $e^{(\ell_j(x)-\ell_j(\bar{x}))/\epsilon'}\nabla\ell_{j'}(x)$ will result in a biased estimator since $\mathbb{E}_{j\sim w_i}e^{(\ell_j(x)-\ell_j(\bar{x}))/\epsilon'} \neq e^{(\mathcal{L}_i(x)-\mathcal{L}_i(\bar{x}))/\epsilon'}$. To address this challenge we propose a new gradient estimator based on the multilevel Monte Carlo (MLMC) method [9].

The MLMC unbiased estimator for $\gamma_i(x) = \epsilon' e^{(\mathcal{L}_i(x)-\mathcal{L}_i(\bar{x}))/\epsilon'}$, which we denote by $\widehat{\mathcal{M}}[\gamma_i(x)]$, is defined as follows:

$$\text{Draw } J \sim \mathrm{Geom}\left(1-\tfrac{1}{\sqrt{8}}\right), S_1, \ldots, S_n \overset{\text{iid}}{\sim} w_i \text{ and let } \widehat{\mathcal{M}}[\gamma_i(x)] \coloneqq \widehat{\gamma}(x; S_1) + \frac{\widehat{\mathcal{D}}_{2^J}}{p_J},$$

where $p_j \coloneqq \mathbb{P}(J=j) = \left(1/\sqrt{8}\right)^j\left(1-\tfrac{1}{\sqrt{8}}\right)$ and, for $n \in 2\mathbb{N}$, we define

$$\widehat{\mathcal{D}}_n \coloneqq \widehat{\gamma}(x; S_1^n) - \frac{\widehat{\gamma}\left(x; S_1^{n/2}\right) + \widehat{\gamma}\left(x; S_{n/2+1}^n\right)}{2} \text{ and } \widehat{\gamma}(x; S_1^n) \coloneqq \epsilon' e^{\frac{1}{n}\sum_{j=1}^n \frac{\ell_{S_j}(x)-\ell_{S_j}(\bar{x})+\frac{\lambda}{2}\|x-\bar{x}\|^2}{\epsilon'}}.$$

With the MLMC estimator for γ_i in hand, we estimate the gradient of $\Gamma_{\epsilon,\lambda}$ as follows:

$$\text{Draw } i \sim p(\bar{x}), j \sim w_i \text{ and set } \hat{g}(x) = \frac{1}{\epsilon'}\widehat{\mathcal{M}}[\gamma_i(x)](\nabla\ell_j(x) + \lambda(x-\bar{x})). \tag{8}$$

In the following lemma we summarize the important properties of the MLMC and gradient estimators; see Appendix C.2 for the proof.

Lemma 2. *Let each ℓ_i satisfy Assumption 1, and let $r \le \frac{\epsilon'}{G}$, $\lambda \le \frac{G}{r}$ and $x \in \mathbb{B}_r(\bar{x})$. Then $\widehat{\mathcal{M}}[\gamma_i(x)]$ and $\hat{g}(x)$ are unbiased for $\gamma_i(x)$ and $\nabla\Gamma_{\epsilon,\lambda}(x)$, respectively, and have bounded second moments: $\mathbb{E}\big[\widehat{\mathcal{M}}[\gamma_i(x)]\big]^2 \le O\Big(\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^2} + \epsilon'^2\Big)$ and $\mathbb{E}\|\hat{g}(x)\|^2 \le O(G^2)$. In addition, the complexity of computing $\widehat{\mathcal{M}}[\gamma_i(x)]$ and $\hat{g}(x)$ is $O(1)$.*

Due to Lemma 2 and since $\Gamma_{\epsilon,\lambda}$ is $\Omega(\lambda)$-strongly convex, we can use the Epoch-SGD algorithm of Hazan and Kale [28] with our gradient estimator (8). This algorithm has rate of convergence $O\big(G^2/(\lambda T)\big)$ and our gradient estimator requires additional N function evaluations for precomputing the sampling probabilities $\{\bar{p}_i\}$. We thus arrive at the following complexity bound.

Theorem 1. *Let each ℓ_j satisfy Assumption 1, let $\epsilon, \delta, \lambda > 0$ and let $r_\epsilon = \epsilon/(2G\log M)$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(G/r_\epsilon)$ and accuracy δ, EpochSGD [28, Algorithm 1]) with the gradient estimator (8) outputs a valid r_ϵ-BROO response and has complexity $\mathcal{C}_\lambda(\delta) = O\big(N + \frac{G^2}{\lambda^2\delta^2}\big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_{\text{g-DRO}}$ (2) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{11/3} H + \left(\frac{GR}{\epsilon}\right)^2 \log^2 H\right) \text{ where } H \coloneqq M\frac{GR}{\epsilon}.$$

We provide the proof for Theorem 1 in Appendix C.3; the final complexity bound follows from straightforward calculations which we now briefly outline. According to Proposition 1, finding an $\frac{\epsilon}{2}$-suboptimal solution for $\mathcal{L}_{\mathrm{smax},\epsilon}$ (and consequently an ϵ-suboptimal solution for $\mathcal{L}_{\text{g-DRO}}$) involves $\widetilde{O}\big((R/r_\epsilon)^{2/3}\big)$ BROO calls with accuracy $\delta = \widetilde{\Omega}\big(r_\epsilon 2^{-J/2}\big)$ and regularization strength $\lambda \ge \lambda_{\mathrm{m}}$, where $J = \min\{\mathrm{Geom}(\frac{1}{2}), m\}$. We may therefore bound the complexity of each such call by

$$\sum_{j=0}^m 2^{-j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(r_\epsilon 2^{-j/2}\right) = \sum_{j=0}^m 2^{-j}\widetilde{O}\left(N + \frac{2^j G^2}{\lambda_{\mathrm{m}}^2 r_\epsilon^2}\right) \overset{(\star)}{=} \widetilde{O}\left(N + \left(\frac{GR}{\epsilon}\right)^2\left(\frac{r_\epsilon}{R}\right)^{2/3}\right),$$

where $(\star)$ follows from substituting $\lambda_{\mathrm{m}} = \widetilde{\Omega}\big(\epsilon r_\epsilon^{-4/3}R^{-2/3}\big)$ and $m = \widetilde{O}(1)$. Multiplying this bound by $\widetilde{O}\big((R/r_\epsilon)^{2/3}\big)$ yields (up to polylogarithmic factors) the conclusion of Theorem 1.

3.3 Accelerated variance reduction for mean-square smooth losses

In this section we provide an algorithm with an improved rate of convergence under the following mean-square smoothness assumption.

Assumption 2. *For all $x, x' \in \mathbb{B}_r(\bar{x})$ and $i \in [M]$, $\mathbb{E}_{j\sim w_i}\|\nabla \ell_j(x) - \nabla \ell_j(x')\|^2 \le L^2\|x - x'\|^2$.*

Note that assuming L-Lipschitz gradient for each ℓ_i implies Assumption 2, but not the other way around. To take advantage of Assumption 2, we first rewrite the function $\Gamma_{\epsilon,\lambda}(x)$ in a way that is more amenable to variance reduction:

$$\Gamma_{\epsilon,\lambda}(x) \coloneqq \sum_{i\in[M]} c_{x',\bar{x}} p_i(x')\gamma_i(x,x'), \text{ where } \gamma_i(x,x') \coloneqq \epsilon' e^{\frac{\mathcal{L}_i(x)-\mathcal{L}_i(x')+\frac{\lambda}{2}\|x-\bar{x}\|^2}{\epsilon'}},$$

$$c_{x',\bar{x}} = \left(\frac{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(x')}{\epsilon'}}}{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(\bar{x})}{\epsilon'}}}\right) \text{ and } p_i(x') \coloneqq \frac{e^{\frac{\mathcal{L}_i(x')}{\epsilon'}}}{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(x')}{\epsilon'}}}.$$

(Note that $\gamma_i(x,\bar{x}) = \gamma_i(x)$). Given a reference point x', to compute a reduced-variance estimator of $\nabla\Gamma_{\epsilon,\lambda}(x)$, we draw $i \sim p_i(x')$ and $j \sim w_i$, and set:

$$\hat{g}_{x'}(x) \coloneqq \nabla\Gamma_{\epsilon,\lambda}(x') + \frac{c_{x',\bar{x}}}{\epsilon'}\left[\widehat{\mathcal{M}}[\gamma_i(x,x')]\nabla\ell_j^\lambda(x) - \gamma_i(x',x')\nabla\ell_j^\lambda(x')\right] \tag{9}$$

where $\nabla\ell_j^\lambda(x) \coloneqq \nabla\ell_j(x) + \lambda(x-\bar{x})$ and $\widehat{\mathcal{M}}[\gamma_i(x,x')]$ is an MLMC estimator for $\gamma_i(x,x')$ defined analogously to $\widehat{\mathcal{M}}[\gamma_i(x)]$ (see details in Appendix C.4). The estimator (9) is not precisely standard SVRG [33] since we use $\widehat{\mathcal{M}}[\gamma_i(x,x')]$ as an estimator for $\gamma_i(x,x')$. Simple calculations show that $\mathbb{E}\hat{g}_{x'}(x) = \nabla\Gamma_{\epsilon,\lambda}(x)$ and the following lemma shows that $\hat{g}$ satisfies a type of variance bound conducive to variance-reduction schemes; see Appendix C.4 for the proof.

Lemma 3. *Let each ℓ_j satisfy Assumptions 1 and 2. For any $\lambda \le \frac{G}{r}$, $r = \frac{\epsilon'}{G}$ and $x, x' \in \mathbb{B}_r(\bar{x})$, the variance of $\hat{g}_{x'}(x)$ is bounded by $\mathrm{Var}(\hat{g}_{x'}(x)) \le O\Big(\big(L + \lambda + \frac{G^2}{\epsilon'}\big)^2\|x - x'\|^2\Big)$.*

Accelerated variance reduction methods for convex functions typically require a stronger variance bound of the form $\mathrm{Var}(\hat{g}_{x'}(x)) \le 2L(F(x') - F(x) - \langle\nabla F(x), x' - x\rangle)$ for every x [cf. 1, Lemma 2.4]. The guarantee of Lemma 3 is weaker, but still allows for certain accelerated rates via, e.g., the Katyusha X algorithm [2]. With it, we obtain the following guarantee.

Theorem 2. *Let each ℓ_j satisfy Assumptions 1 and 2. Let $\epsilon > 0$, $\epsilon' = \epsilon/(2\log M)$ and $r_\epsilon = \epsilon'/G$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(G/r_\epsilon)$ and accuracy δ, KatyushaX^s [2, Algorithm 2] with the gradient estimator (9) outputs a valid r_ϵ-BROO response and has complexity $\mathcal{C}_\lambda(\delta) = O\Big(\Big(N + \frac{N^{3/4}(G+\sqrt{\epsilon' L})}{\sqrt{\lambda\epsilon'}}\Big)\log\big(\frac{Gr_\epsilon}{\lambda\delta^2}\big)\Big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_{\text{g-DRO}}$ (2) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3} H + N^{3/4}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{7/2} H\right) \text{ where } H \coloneqq M\frac{GR}{\epsilon}.$$

We provide the proof of Theorem 2 in Appendix C.5. For the special case of Group DRO with a single group satisfying Assumption 2 with $L = \Theta(G^2/\epsilon)$, i.e. minimizing the average loss, we have the lower bound $\widetilde{\Omega}\big(N + N^{3/4}\frac{GR}{\epsilon}\big)$ [62] and for the case of N distinct groups, i.e. minimizing the maximal loss, we have the lower bound $\widetilde{\Omega}\big(N\epsilon^{-2/3}\big)$ [15]. This implies that in the weakly mean-square smooth setting the term scaling as $N^{3/4}\epsilon^{-1}$ and the term scaling as $N\epsilon^{-2/3}$ are unimprovable.

4 DRO with f-divergence

In this section we develop our BROO implementation for the f-divergence objective (3). In Section 4.1 we reduce the original DRO problem to a regularized form using Lagrange multipliers. Next, in Section 4.2 we show that adding negative entropy regularization to the objective produces the stability properties necessary for efficient ball optimization. In Section 4.3 we describe a BROO implementation for the non-smooth case using a variant of Epoch-SGD [28], and in Section 4.4 we implement the BROO under a weak-smoothness assumption by carefully restarting an accelerated variance reduction method [1].

4.1 The dual problem

We first note that (due to Slater's condition), by Lagrange duality, the objective (3) is equivalent to

$$\mathcal{L}_{f\text{-div}}(x) \coloneqq \max_{q\in\Delta^N:\sum_{i\in[N]}\frac{f(Nq_i)}{N}\le 1} \sum_{i\in[N]} q_i\ell_i(x) = \min_{\nu\ge 0}\left\{\nu + \max_{q\in\Delta^N}\sum_{i\in[N]}\Big(q_i\ell_i(x) - \frac{\nu}{N}f(Nq_i)\Big)\right\}.$$

Writing $\psi(s) \coloneqq \frac{\nu}{N}f(Ns)$ for $\psi : \mathbb{R}_+ \to \mathbb{R}$, we therefore consider objectives of the form

$$\mathcal{L}_\psi(x) \coloneqq \max_{q\in\Delta^N}\sum_{i\in[N]}(q_i\ell_i(x) - \psi(q_i)) = \min_{y\in\mathbb{R}}\left\{\Upsilon(x,y) \coloneqq \sum_{i\in[N]}\psi^*(\ell_i(x) - Gy) + Gy\right\} \tag{10}$$

where G is the Lipschitz constant of each loss ℓ_i and for the last equality we use Lagrange duality, with $\psi^*(v) \coloneqq \max_{t\in\mathrm{dom}(\psi)}\{vt - \psi(t)\}$ the Fenchel dual of ψ (for more details see Appendix D.1). We show that under weak assumptions (introducing logarithmic dependence on bounds on f and the losses) we can solve the constrained problem (3) to accuracy ϵ by computing a polylogarithmic number of $O(\epsilon)$-accurate minimizers of (10); see Appendix D.2 for details. Since the complexity of solving (10) holds for any $\nu > 0$, and we have a lower bound for the required ν, for the remainder of this section we focus on minimizing $\mathcal{L}_\psi$ for *arbitrary* convex ψ.

4.2 Stabilizing the gradient estimator

While minimizing (10) can be viewed as ERM (over x and y), straightforward application of SGD does not solve it efficiently. To see this, consider the standard gradient estimator formed by sampling $i \sim \mathsf{Unif}([N])$ and taking $\hat{g}^{\mathsf{x}} = N\psi^{*\prime}(\ell_i(x) - Gy)\nabla\ell_i(x)$ and $\hat{g}^{\mathsf{y}} = G\big(1 - N\psi^{*\prime}(\ell_i(x) - Gy)\big)$. For general ψ, this estimator will have unbounded second moments, and therefore SGD using them would lack a convergence guarantee. As an extreme example, consider $\psi = 0$ (corresponding to minimizing the maximum loss) whose conjugate function $\psi^*(v)$ is 0 for $v \le 0$ and ∞ for $v > 0$, leading to meaningless stochastic gradients.

We obtain bounded gradient estimates in two steps. First, we find a better distribution for i using a reference point $\bar{x} \in \mathcal{X}$ with corresponding $\bar{y} = \operatorname{argmin}_{y\in\mathbb{R}}\Upsilon(\bar{x}, y)$. Namely, we note that the optimality condition for $\bar{y}$ implies that $\psi^{*\prime}(\ell_i(\bar{x}) - G\bar{y})$ is a pmf over $[N]$. Therefore, we may sample $i \sim \psi^{*\prime}(\ell_i(\bar{x}) - G\bar{y})$ and estimate the gradient of Υ at (x, y) using $\hat{g}^{\mathsf{x}} = \rho_i(x, y)\nabla\ell_i(x)$ and $\hat{g}^{\mathsf{y}} = G\big(1 - \rho_i(x, y)\big)$, where $\rho_i(x, y) = \frac{\psi^{*\prime}(\ell_i(x) - Gy)}{\psi^{*\prime}(\ell_i(\bar{x}) - G\bar{y})}$. However, for general ψ (and $\psi = 0$ in particular), the ratio $\rho_i(x, y)$ can be unbounded even when x, y are arbitrarily close to $\bar{x}, \bar{y}$.

Our second step ensures that $\rho_i(x, y)$ is bounded around $\bar{x}, \bar{y}$ by adding a small negative entropy term to ψ, defining

$$\psi_\epsilon(q) \coloneqq \psi(q) + \epsilon' q\log q \;\text{ where }\; \epsilon' \coloneqq \frac{\epsilon}{2\log N}, \tag{11}$$

and

$$\mathcal{L}_{\psi,\epsilon}(x) = \min_{y\in\mathbb{R}}\Upsilon_\epsilon(x, y) \;\text{ with }\; \Upsilon_\epsilon(x, y) \coloneqq \sum_{i\in[N]}\psi_\epsilon^*(\ell_i(x) - Gy) + Gy. \tag{12}$$

Due to our choice of ϵ', we have $|\mathcal{L}_\psi(x) - \mathcal{L}_{\psi,\epsilon}(x)| \le \epsilon/2$ for all $x \in \mathbb{R}^d$, and consequently an $\epsilon/2$-accurate minimizer of $\mathcal{L}_{\psi,\epsilon}$ is also an ϵ-accurate for $\mathcal{L}_\psi$ (see Lemma 18 in Appendix D.3). When $\psi = 0$ we have $\psi_\epsilon^*(v) = e^{(v-1)/\epsilon'}$ and therefore the corresponding $\rho_i(x, y) = e^{(\ell_i(x) - \ell_i(\bar{x}) - G(y-\bar{y}))/\epsilon'}$.The following lemma, which might be of independent interest, shows that the same conclusion holds for any convex ψ.

Lemma 4. *For any convex $\psi : \mathbb{R}_+ \to \mathbb{R}$ and ψ_ϵ defined in (10), $\log\big(\psi_\epsilon^{*\prime}(\cdot)\big)$ is $\frac{1}{\epsilon'}$-Lipschitz.*

See proof in Appendix D.4. Thus, $\psi_\epsilon^{*\prime}(v)/\psi_\epsilon^{*\prime}(\bar{v}) = e^{\log\psi_\epsilon^{*\prime}(v) - \log\psi_\epsilon^{*\prime}(\bar{v})} \le e^{(v-\bar{v})/\epsilon'}$ and $\rho_i(x, y) \le e^{(\ell_i(x) - \ell_i(\bar{x}) - G(y-\bar{y}))/\epsilon'}$ continues to hold. Therefore, if $|y - \bar{y}| \le \epsilon'/G = r_\epsilon$ and $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ (so that $|\ell_i(x) - \ell_i(\bar{x})| \le \epsilon'$ if ℓ_i satisfies Assumption 1), we have the bound $\rho_i(x, y) \le e^2$.

It remains to show that we may indeed restrict y to be within distance r_ϵ from $\bar{y}$. To this end, we make the following observation which plays a key part in our analysis and might also be of independent interest (see proof in Appendix D.4).

Lemma 5. *For $G > 0$, $\ell(x) = (\ell_1(x), \dots, \ell_N(x))$ and $y^\star(x) = \operatorname{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(x,y)$, we have $|y^\star(x) - y^\star(x')| \le \frac{1}{G}\|\ell(x) - \ell(x')\|_\infty$ for all $x, x' \in \mathcal{X}$. Moreover, if each ℓ_i is G-Lipschitz, we have $|y^\star(x) - y^\star(x')| \le \|x - x'\|$.*

Lemma 5 implies that $x^\star, y^\star = \operatorname{argmin}_{x\in\mathbb{B}_{r_\epsilon}(\bar{x}), y\in\mathbb{R}} \Upsilon_\epsilon(x,y)$ satisfy $|y^\star - \bar{y}| \le \|x^\star - \bar{x}\| \le r_\epsilon$. Therefore, when minimizing Υ_ϵ (or any regularized version of it) inside the ball $\mathbb{B}_{r_\epsilon}(\bar{x})$, we may restrict y to $[\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$ without loss of generality. We also note that Lemma 5 holds for all values of ϵ and is therefore valid even without entropy regularization (as long as ψ^* is strongly convex $y^\star$ is unique, and if $y^\star$ is not unique then we can still choose $y^\star$ such that the bound of this lemma holds).

4.3 BROO implementation for f-divergence DRO with non-smooth losses

By the discussion above, to implement a BROO for $\mathcal{L}_{\psi,\epsilon}(x)$ (with radius $r_\epsilon = \epsilon'/G$, regularization λ, and query $\bar{x} \in \mathcal{X}$) it suffices to minimize $\Upsilon_{\epsilon,\lambda}(x,y) := \Upsilon_\epsilon(x,y) + \frac{\lambda}{2}\|x - \bar{x}\|^2$ over $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ and $y \in [\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$, where $\bar{y} = \operatorname{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(\bar{x}, y)$. To that end we estimate the gradient of $\Upsilon_{\epsilon,\lambda}(x,y)$ as follows. Letting $\bar{p}_i = \psi_\epsilon^{*\prime}(\ell_i(\bar{x}) - G\bar{y})$ (making $\bar{p}$ a pmf by optimality of $\bar{y}$), we sample $i \sim \bar{p}$ and set

$$\hat{g}^{\mathrm{x}}(x,y) = \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i}\nabla\ell_i(x,y) \;\text{ and }\; \hat{g}^{\mathrm{y}}(x,y) = G\left(1 - \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i}\right). \tag{13}$$

Lemma 4 implies the following bounds on our gradient estimator; see proof in Appendix D.5.

Lemma 6. *Let each ℓ_i be G-Lipschitz, let $\bar{x} \in \mathcal{X}$ and $\bar{y} = \operatorname{argmin}_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(\bar{x}, y)$. Let $r_\epsilon = \frac{\epsilon'}{G}$, then for all $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ and $y \in [\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$, the gradient estimators $\hat{g}^{\mathrm{x}}$ and $\hat{g}^{\mathrm{y}}$ satisfy the following:*

1. $\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{x}}(x,y)] = \nabla_x \Upsilon_\epsilon(x,y)$ *and* $\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{y}}(x,y)] = \nabla_y \Upsilon_\epsilon(x,y)$.
2. $\mathbb{E}_{i\sim\bar{p}_i}\|\hat{g}^{\mathrm{x}}(x,y)\|^2 \le e^4 G^2$ *and* $\mathbb{E}_{i\sim\bar{p}_i}|\hat{g}^{\mathrm{y}}(x,y)|^2 \le e^4 G^2$.

To implement the BROO using our gradient estimator we develop a variant of the Epoch-SGD algorithm of Hazan and Kale [28] (Algorithm 3 in Appendix D.5). Similarly to Epoch-SGD, we apply standard SGD on $\Upsilon_{\epsilon,\lambda}$ (with gradient estimator (13)) in "epochs" whose length doubles in every repetition. Our algorithm differs slightly in how each epoch is initialized. Standard Epoch-SGD initializes with the average of the previous epoch's iterates, and strong convexity shows that the suboptimality and distance to the optimum shrink by a constant factor after every epoch. However, since $\Upsilon_{\epsilon,\lambda}$ is strongly convex only in x and not in y, we cannot directly use this scheme. Instead, we set the initial y variable to be $\operatorname{argmin}_y \Upsilon_{\epsilon,\lambda}(x', y)$, where x' is the initial x variable still defined as the previous epoch's average; this initialization has complexity N, but we only preform it a logarithmic number of times. Using our initialization scheme and Lemma 5, we recover the original Epoch-SGD contraction argument, yielding the following complexity bound (see proof in Appendix D.5).

Theorem 3. *Let each ℓ_i satisfy Assumption 1. Let $\epsilon, \lambda, \delta > 0$, and $r_\epsilon = \epsilon/(2G\log N)$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(G/r_\epsilon)$ and accuracy $\delta < r_\epsilon/2$, Algorithm 3 outputs a valid r_ϵ-BROO response for $\mathcal{L}_{\psi,\epsilon}$ and has complexity $\mathcal{C}_\lambda(\delta) = O\big(\frac{G^2}{\lambda^2\delta^2} + N\log\big(\frac{r_\epsilon}{\delta}\big)\big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$ (10) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{11/3} H + \left(\frac{GR}{\epsilon}\right)^2\log^2 H\right) \;\text{ where }\; H := N\frac{GR}{\epsilon}.$$

4.4 Accelerated variance reduction for smooth losses

In this section, we take advantage of the following smoothness assumption.

Assumption 3. *For every $i \in [N]$ the loss ℓ_i is L-smooth, i.e., has L-Lipschitz gradient.*

Let us rewrite $\Upsilon_{\epsilon,\lambda}$ in a form that is more amenable to variance reduction techniques:

$$\Upsilon_{\epsilon,\lambda}(x,y) = \sum_{i\in[N]} \bar{p}_i \upsilon_i(x,y) \;\text{ where }\; \upsilon_i(x,y) := \frac{\psi_\epsilon^*(\ell_i(x) - Gy)}{\bar{p}_i} + Gy + \frac{\lambda}{2}\|x - \bar{x}\|^2$$

and, as before $\bar{p}_i = \psi_\epsilon^*(\ell_i(\bar{x}) - G\bar{y})$ for some ball center $\bar{x} \in \mathcal{X}$ and $\bar{y} = \operatorname{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(\bar{x}, y)$. In the following lemma, we bound the smoothness of the functions υ_i, deferring the proof to Appendix D.6.

Lemma 7. *For any $i \in [N]$, let ℓ_i be G-Lipschitz and L -smooth, let $r_\epsilon = \frac{\epsilon'}{G}$ and $\lambda = O\big(\frac{G}{r_\epsilon}\big)$. The restriction of υ_i to $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ and $y \in [\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$ is $O(G)$-Lipschitz and $O\big(L + \frac{G^2}{\epsilon'}\big)$-smooth.*

Since $\Upsilon_{\epsilon,\lambda}$ is a finite sum of smooth functions, we can obtain reduced-variance gradient estimates by the standard SVRG technique [33]. For any reference point x', y', the estimator is

$$\hat{g}_{x',y'}(x, y) = \nabla\Upsilon_\epsilon(x', y') + \nabla\upsilon_i(x, y) - \nabla\upsilon_i(x', y'), \tag{14}$$

where ∇ is with respect to the vector $[x, y]$. Similar to non-smooth case, obtaining an efficient BROO implementation is complicated by the fact that $\Upsilon_{\epsilon,\lambda}$ is strongly-convex in x but not in y. Our solution is also similar: we propose a restart scheme and minimize over y exactly between restarts (Algorithm 4 in Appendix D.6), that gives the following complexity bound (see proof in Appendix D.6).

Theorem 4. *Let each ℓ_i satisfy Assumptions 1 and 3, let $\epsilon, \lambda, \delta > 0$, and $r_\epsilon = \frac{\epsilon}{2G\log N}$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(\frac{G}{r_\epsilon})$ and accuracy δ, Algorithm 4 outputs a valid r_ϵ-BROO response for $\mathcal{L}_{\psi,\epsilon}$ and has complexity $\mathcal{C}_\lambda(\delta) = O\Big(\Big(N + \frac{\sqrt{N}(G+\sqrt{\epsilon' L})}{\sqrt{\lambda\epsilon'}}\Big)\log\frac{G r_\epsilon}{\lambda\delta^2}\Big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$ (10) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3} H + \sqrt{N}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{5/2} H\right) \text{ where } H \coloneqq N\frac{GR}{\epsilon}.$$

5 Discussion

Limitations While our work indicates that the ball optimization approach offers significant complexity gains for DRO, we note that turning the algorithms we propose into practical DRO methods faces several challenges. A main challenge is the costly bisection procedure common to all Monteiro-Svaiter-type acceleration schemes [41, 25, 13, 52]. Fortunately, very recently, two works [16, 36] (the former partially motivated by our paper) have shown how to remove the bisection from Monteiro-Svaiter schemes, significantly improving the practical potential of the methods we propose. However, another practical limitation of our approach is the need to tune many parameters that are not known in advance, such as those relating to the ball radius r_ϵ and smoothing level ϵ', as well as step sizes and number of iterations of oracle implementations; a more adaptive setting for these parameters is likely important.

Extensions First, it would be interesting to extend our approach to DRO objectives $\max_{q\in\mathcal{U}} \sum_{i\in[N]} q_i\ell_i(x)$ with uncertainty set $\mathcal{U}$ that is an arbitrary subset of the simplex. While the subgradient method, the primal-dual method (Appendix A.1), and "AGD on softmax" (Appendix A.2) all apply to any $\mathcal{U} \subseteq \Delta^N$, our methods strongly rely on the structure of $\mathcal{U}$ induced by Group- f-divergence DRO, and extending them to unstructured $\mathcal{U}$'s seems challenging.

Second, it would be interesting to generalize our results in the "opposite" direction of getting better complexity bounds for problems with additional structure. For Group-DRO our bounds are essentially optimal when the number of groups $M = \Omega(N)$, but are suboptimal when $M = O(1)$. We leave it as a question for further research if it is possible to obtain a stronger bound such as $\widetilde{O}\big(N + M\epsilon^{-2/3} + \epsilon^{-2}\big)$, which recovers our result for $M = N$ but improves on it for smaller values of M. Taking CVaR at level α as a special case of f-divergence DRO, our bounds are optimal when α is close to $1/N$ but suboptimal for larger value of α; it would be interesting to obtain bounds such as $\widetilde{O}\big(N + \alpha^{-1}\epsilon^{-2/3} + \epsilon^{-2}\big)$.

A third possible extension of our research is DRO in the non-convex setting. For this purpose, it might be possible to use the technique of Carmon et al. [12] for turning accelerated convex optimization algorithms to improved-complexity methods for smooth non-convex optimization.

Acknowledgments

We thank Marc Teboulle for detailed and helpful feedback. This work was supported by Israeli Science Foundation (ISF) grant no. 2486/21, the Len Blavatnik and the Blavatnik Family foundation and the Adelis Foundation.

References

[1] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *Journal of Machine Learning Research*, 18(221):1–51, 2017.

[2] Z. Allen-Zhu. Katyusha X: Practical momentum method for stochastic sum-of-nonconvex optimization. In *International Conference on Machine Learning (ICML)*, 2018.

[3] H. Asi, Y. Carmon, A. Jambulapati, Y. Jin, and A. Sidford. Stochastic bias-reduced gradient methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[4] A. Beck and M. Teboulle. Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization*, 22(2):557–580, 2012.

[5] A. Ben-Tal and M. Teboulle. A smoothing technique for nondifferentiable optimization problems. In *Optimization*, pages 1–11. Springer, 1989.

[6] A. Ben-Tal, D. den Hertog, A. De Waegenaere, B. Melenberg, and G. Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2): 341–357, 2013.

[7] D. P. Bertsekas. *Nondifferentiable optimization via approximation*. Springer, 1975.

[8] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

[9] J. H. Blanchet and P. W. Glynn. Unbiased Monte Carlo for optimization and functions of expectations via multi-level randomization. In *2015 Winter Simulation Conference (WSC)*, 2015.

[10] J. H. Blanchet, P. W. Glynn, and Y. Pei. Unbiased multilevel Monte Carlo: Stochastic optimization, steady-state simulation, quantiles, and other applications. *arXiv:1904.09929*, 2019.

[11] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[12] Y. Carmon, J. C. Duchi, O. Hinder, and A. Sidford. "Convex until proven guilty": Dimension-free acceleration of gradient descent on non-convex functions. In *International Conference on Machine Learning (ICML)*, 2017.

[13] Y. Carmon, A. Jambulapati, Q. Jiang, Y. Jin, Y. T. Lee, A. Sidford, and K. Tian. Acceleration with a ball optimization oracle. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[14] Y. Carmon, Y. Jin, A. Sidford, and K. Tian. Coordinate methods for matrix games. In *Foundations of Computer Science (FOCS)*, 2020.

[15] Y. Carmon, A. Jambulapati, Y. Jin, and A. Sidford. Thinking inside the ball: Near-optimal minimization of the maximal loss. In *Conference on Learning Theory (COLT)*, 2021.

[16] Y. Carmon, D. Hausler, A. Jambulapati, Y. Jin, and A. Sidford. Optimal and adaptive Monteiro-Svaiter acceleration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[17] Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in neural information processing systems (NeurIPS)*, 2015.

[18] M. B. Cohen, Y. T. Lee, G. Miller, J. Pachocki, and A. Sidford. Geometric median in nearly linear time. In *ACM symposium on Theory of Computing*, 2016.

[19] I. Csiszár. On topological properties of f-divergences. *Studia Math. Hungar.*, 2:329–339, 1967.

[20] S. Curi, K. Y. Levy, S. Jegelka, and A. Krause. Adaptive sampling for stochastic risk-averse learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[21] J. Duchi and H. Namkoong. Variance-based regularization with convex objectives. *Journal of Machine Learning Research*, 20(68):1–55, 2019.

[22] J. C. Duchi and H. Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

[23] P. M. Esfahani and D. Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming, Series A*, 171(1):115–166, 2018.

[24] R. Gao and A. J. Kleywegt. Distributionally robust stochastic optimization with Wasserstein distance. *arXiv:1604.02199*, 2016.

[25] A. Gasnikov, P. Dvurechensky, E. Gorbunov, E. Vorontsova, D. Selikhanovych, C. A. Uribe, B. Jiang, H. Wang, S. Zhang, S. Bubeck, Q. Jiang, Y. T. Lee, Y. Li, and A. Sidford. Near optimal methods for minimizing convex functions with Lipschitz p-th derivatives. In *Conference on Learning Theory (COLT)*, 2019.

[26] M. B. Giles. Multilevel monte carlo path simulation. *Operations research*, 56(3):607–617, 2008.

[27] T. Hashimoto, M. Srivastava, H. Namkoong, and P. Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, 2018.

[28] E. Hazan and S. Kale. Beyond the regret minimization barrier: Optimal algorithms for stochastic strongly-convex optimization. *Journal of Machine Learning Research*, 15(71):2489–2512, 2014.

[29] S. Heinrich. Multilevel Monte Carlo methods. In *International Conference on Large-Scale Scientific Computing*, 2001.

[30] D. Hovy and A. Søgaard. Tagging performance correlates with author age. In *Association for Computational Linguistics (ACL)*, pages 483–488, 2015.

[31] W. Hu, G. Niu, I. Sato, and M. Sugiyama. Does distributionally robust supervised learning give robust classifiers? In *International Conference on Machine Learning (ICML)*, 2018.

[32] J. Jin, B. Zhang, H. Wang, and L. Wang. Non-convex distributionally robust optimization: Non-asymptotic analysis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[33] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems (NeurIPS)*, 2013.

[34] C. Kent, J. Blanchet, and P. Glynn. Frank-wolfe methods in probability space. *arXiv:2105.05352*, 2021.

[35] P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao, T. Lee, E. David, I. Stavness, W. Guo, B. A. Earnshaw, I. S. Haque, S. Beery, J. Leskovec, A. Kundaje, E. Pierson, S. Levine, C. Finn, and P. Liang. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

[36] D. Kovalev and A. Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[37] P. Krokhmal, J. Palmquist, and S. Uryasev. Portfolio optimization with conditional value-at-risk objective and constraints. *Journal of risk*, 4:43–68, 2002.

[38] G. Lan, Z. Li, and Y. Zhou. A unified variance-reduced accelerated gradient method for convex optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[39] D. Levy, Y. Carmon, J. C. Duchi, and A. Sidford. Large-scale methods for distributionally robust optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[40] F. Lin, X. Fang, and Z. Gao. Distributionally robust optimization: a review on theory and applications. *Numerical Algebra, Control & Optimization*, 12(1):159, 2022.

[41] R. D. Monteiro and B. F. Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2):1092–1125, 2013.

[42] H. Namkoong and J. C. Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[43] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[44] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming, Series A*, 103(1):127–152, 2005.

[45] Y. Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

[46] L. Oakden-Rayner, J. Dunnmon, G. Carneiro, and C. Ré. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *ACM conference on health, inference, and learning*, 2020.

[47] Y. Oren, S. Sagawa, T. B. Hashimoto, and P. Liang. Distributionally robust language modeling. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[48] R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2: 21–41, 2000.

[49] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang. Distributionally robust neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

[50] A. Shapiro. Distributionally robust stochastic programming. *SIAM Journal on Optimization*, 27 (4):2258–2275, 2017.

[51] A. Sinha, H. Namkoong, and J. Duchi. Certifiable distributional robustness with principled adversarial training. In *International Conference on Learning Representations (ICLR)*, 2018.

[52] C. Song, Y. Jiang, and Y. Ma. Unified acceleration of high-order algorithms under Hölder continuity and uniform convexity. *SIAM journal of optimization*, 2021.

[53] C. Song, C. Y. Lin, S. J. Wright, and J. Diakonikolas. Coordinate linear variance reduction for generalized linear programming. *arXiv:2111.01842*, 2021.

[54] N. A. Urpí, S. Curi, and A. Krause. Risk-averse offline reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[55] K. A. Wang, N. S. Chatterji, S. Haque, and T. Hashimoto. Is importance weighting incompatible with interpolating classifiers? In *International Conference on Learning Representations*, 2022.

[56] S. Wang, W. Guo, H. Narasimhan, A. Cotter, M. R. Gupta, and M. I. Jordan. Robust optimization for fairness with noisy protected groups. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[57] J. Wen, C.-N. Yu, and R. Greiner. Robust learning under uncertain test distributions: Relating covariate shift to model misspecification. In *International Conference on Machine Learning (ICML)*, pages 631–639, 2014.

[58] B. E. Woodworth and N. Srebro. Tight complexity bounds for optimizing composite objectives. In *Advances in neural information processing systems (NeurIPS)*, 2016.

[59] R. Zhai, C. Dan, Z. Kolter, and P. Ravikumar. Doro: Distributional and outlier robust optimization. In *International Conference on Machine Learning (ICML)*, 2021.

[60] R. Zhai, C. Dan, A. Suggala, J. Z. Kolter, and P. Ravikumar. Boosted CVaR classification. In *Advances in neural information processing systems (NeurIPS)*, 2021.

[61] C. Zhou, D. Levy, X. Li, M. Ghazvininejad, and G. Neubig. Distributionally robust multilingual machine translation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

[62] D. Zhou and Q. Gu. Lower bounds for smooth nonconvex finite-sum optimization. In *International Conference on Machine Learning (ICML)*, 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Theorem 1, Theorem 2, Theorem 3 and Theorem 4.
   (b) Did you describe the limitations of your work? [Yes] See "Limitations" paragraph in Section 1.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Assumption 1 Assumption 2 and Assumption 3.
   (b) Did you include complete proofs of all theoretical results? [Yes] See supplementary material.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Alternative algorithms for solving DRO problems

A.1 Primal-dual stochastic mirror descent

In this section, we present a primal-dual method capable of solving all the DRO problems our paper considers, under an additional assumption of bounded losses: for every j and x we assume $|\ell_j(x)| \le B_\ell$. Consider the primal-dual problem

$$\underset{x\in\mathcal{X}}{\text{minimize}} \max_{q\in\mathcal{U}} \left\{ \mathcal{L}_{\text{pd}}(x,q) \coloneqq \sum_{i\in[m]} q_i \mathcal{L}_i(x) \right\} \tag{15}$$

where $\mathcal{X} \subseteq B_R(x_0)$ is a closed convex set as before and $\mathcal{U}$ is now an arbitrary closed convex subset of the simplex Δ^m and $\mathcal{L}_i(x) = \sum_{j\in[N]} w_{ij}\ell_j(x)$ are "group losses" with $w_i \in \Delta^N$ for every $i \in [m]$. This formulation subsumes both Group DRO (where $m = M$ and $\mathcal{U} = \Delta^M$) and f-divergence DRO (where $m = N$, $\mathcal{L}_i(x) = \ell_i(x)$, and $\mathcal{U}$ is an f-divergence ball).

As discussed in the introduction, several works have proposed primal-dual methods for DRO, but we could not find in these works the precise rate we prove here (in Proposition 2 below) in its full generality. Our proof is a straightforward specialization of the more general results of Carmon et al. [14].

The particular algorithm we consider is primal-dual stochastic mirror descent, with distances generated by the squared Euclidean norm on $\mathcal{X}$ and entropy on $\mathcal{U}$ and gradient clipping for the $\mathcal{U}$ iterates, corresponding to the following recursion:

$$\begin{aligned} x_{t+1} &= \operatorname*{argmin}_{x\in\mathcal{X}} \left\{ \langle \eta \hat{g}^{\mathrm{x}}(x_t,q_t), x\rangle + \frac{\log m}{R^2}\|x-x_t\|^2 \right\} \text{ and} \\ q_{t+1} &= \operatorname*{argmax}_{x\in\mathcal{X}} \left\{ \left\langle \Pi_{[-1,1]^m}(\eta \hat{g}^{\mathrm{q}}(x_t,q_t)), q \right\rangle + \sum_{i\in[m]} [q]_i \log\frac{[q]_i}{[q_t]_i} \right\}, \end{aligned} \tag{16}$$

where η is a step-size parameter, $\Pi_{[-1,1]^m}$ is the Euclidean projection to the unit box (i.e., entry-wise clipping to $[-1,1]$), and $\hat{g}^{\mathrm{x}}$ and $\hat{g}^{\mathrm{q}}$ are unbiased estimators for $\nabla_x \mathcal{L}_{\text{pd}}$ and $\nabla_q \mathcal{L}_{\text{pd}}$, respectively, given by

$$\begin{aligned} \hat{g}^{\mathrm{x}}(z) &\coloneqq \nabla \ell_j(z^{\mathrm{x}}) \ \text{ with } i \sim z^q \text{ and } j \sim w_i \\ \hat{g}^{\mathrm{q}}(z) &\coloneqq m\ell_j(z^{\mathrm{x}})e_i \ \text{ with } i \sim \mathsf{Unif}([m]) \text{ and } j \sim w_i, \end{aligned} \tag{17}$$

with $e_i \in \mathbb{R}^m$ being the ith standard basis vector in $\mathbb{R}^m$.

This method yields the following convergence guarantees.

Proposition 2. *Assume that each ℓ_j convex and G-Lipschitz and satisfies $|\ell_j(x)| \le B_\ell$ for every $x \in \mathcal{X}$. For $T \in \mathbb{N}$ let $\bar{x}_T = \frac{1}{T}\sum_{t=0}^T x_t$ and $\bar{q}_T = \frac{1}{T}\sum_{t=0}^T q_t$, where $\{x_t, q_t\}$ are the iterates defined in (16), with $\eta = O\Big(\frac{\epsilon \log m}{G^2R^2 + mB_\ell^2}\Big)$. Then, for any $\epsilon > 0$, if $T \ge O\Big(\frac{G^2R^2 + B_\ell^2 m \log m}{\epsilon^2}\Big)$ we have that*

$$\mathbb{E}\mathcal{L}_{\text{DRO}}(\bar{x}_T) - \min_{x_\star\in\mathcal{X}} \mathcal{L}_{\text{DRO}}(x_\star) \le \mathbb{E}\max_{q\in\mathcal{U}} \mathcal{L}_{\text{pd}}(\bar{x}_T, q) - \mathbb{E}\min_{x\in\mathcal{X}} \mathcal{L}_{\text{pd}}(x, \bar{q}_T) \le \epsilon,$$

where $\mathcal{L}_{\text{DRO}}(x) = \max_{q\in\mathcal{U}} \mathcal{L}_{\text{pd}}(x,q)$.

Proof. The proposition is a direct corollary of a more general result by Carmon et al. [14]. To show this, we rewrite the iterations (16) using "local norm setup" notation of [14]. In particular, let $\mathcal{U} = \mathcal{X} \times \mathcal{U}$ and for every $z = (z^{\mathrm{x}}, z^{\mathrm{q}}) \in \mathcal{U}$ define the local norm of $\delta \in \mathcal{U}^*$ at z as

$$\|\delta\|_z \coloneqq \sqrt{\frac{R^2}{\log m}\|\delta^{\mathrm{x}}\|_2^2 + \sum_{i\in[m]} [z^{\mathrm{q}}]_i [\delta^{\mathrm{q}}]_i^2}.$$

In addition, we define the generating distance function

$$r(z) = r(z^{\mathrm{x}}, z^{\mathrm{q}}) \coloneqq \frac{\log m}{R^2}\|z^{\mathrm{x}}\|_2^2 + \sum_{i\in[m]} [z^{\mathrm{q}}]_i \log[z^{\mathrm{q}}]_i$$

and write its associated Bregman divergence as $V_z(z') = r(z') - r(z) - \langle \nabla r(z), z' - z\rangle$. Next, we let $\Theta \coloneqq \max_{z,z' \in \mathcal{Z}}\{r(z) - r(z')\}$ and observe that since $z^{\mathrm{x}} \in \mathcal{X} \subset \mathbb{B}_R(x_0)$ and $z^{\mathrm{q}} \in \mathcal{U} \subset \Delta^m$, then $\Theta = 2\log m$. Last, we define the function clip: $\mathcal{Z}^* \to \mathcal{Z}^*$ as follows:

$$\mathrm{clip}(\delta^{\mathrm{x}}, \delta^{\mathrm{q}}) \coloneqq \big(\delta^{\mathrm{x}}, \Pi_{[-1,1]^m}(\delta^{\mathrm{q}})\big),$$

where $\Pi_{[-1,1]^m}$ denotes entry-wise clipping to $[-1,1]$. By an argument directly analogous to [14, Proposition 1], the quintuplet $(\mathcal{Z}, \|\cdot\|_{\cdot}, r, \Theta, \mathrm{clip})$ forms a valid local norm setup [14, Definition 1].

With this notation, the iterations (16) have the concise form

$$z_{t+1} = \operatorname*{argmin}_{w \in \mathcal{Z}}\{\langle \mathrm{clip}(\eta \hat{g}(z_t)), w\rangle + V_{z_t}(w)\}, \tag{18}$$

where $\hat{g}(z) \coloneqq (\hat{g}^{\mathrm{x}}, -\hat{g}^{\mathrm{q}})$, with $\hat{g}^{\mathrm{x}}$ and $\hat{g}^{\mathrm{q}}$ as defined in (17) above. It is then straight-forward to verify that $\mathbb{E}\hat{g}(z) = (\nabla_x \mathcal{L}_{\mathrm{pd}}(z), -\nabla_q \mathcal{L}_{\mathrm{pd}}(z))$ for every $z \in \mathcal{U}$, and that

$$\mathbb{E}\big[\|\hat{g}(z)\|_w^2\big] = \mathbb{E}_{i\sim z^{\mathrm{q}}, j\sim w_i}\left[\frac{R^2}{\log m}\|\nabla \ell_j(z^{\mathrm{x}})\|^2\right] + \mathbb{E}_{i\sim \mathrm{Unif}([m]), j\sim w_i}\big[[z^{\mathrm{q}}]_i m^2 \ell_j^2(z^{\mathrm{x}})\big] \le \frac{G^2R^2}{\log m} + mB_\ell^2$$

for every $z, w \in \mathcal{U}$. Therefore, $\hat{g}$ is an L-local estimator [14, Definition 3] with $L^2 = G^2R^2/(logm) + mB_\ell^2$ so that $L^2\Theta = 2G^2R^2 + 2B_\ell^2 m\log m$. Proposition 2 now follows immediately from [14, Proposition 2]. $\square$

A.2 AGD on the softmax: complexity bound

In this appendix we briefly develop the complexity guarantees of the "AGD on softmax" approach mentioned in the introduction. While the idea is well known, we could not find in the literature an analysis of the method for the general DRO setting (i.e., maximization over q in arbitrary subsets of the simplex), so we provide it here.

We wish to minimize, over $x \in \mathcal{X}$,

$$\mathcal{L}_{\mathrm{DRO}}(x) \coloneqq \max_{q\in\mathcal{U}} \sum_{i\in[N]} q_i \ell_i(x)$$

where each ℓ_i is convex, G-Lipschitz and L-smooth and $\mathcal{U}$ is an arbitrary closed convex subset of the simplex Δ^N; note that this includes both Group DRO and f-divergence DRO as special cases. We define the approximation

$$\widetilde{\mathcal{L}}_{\mathrm{DRO}}(x) \coloneqq \max_{q\in\mathcal{U}\subset\Delta^N}\left\{\sum_{i\in[N]} q_i\ell_i(x) - \epsilon' q_i \log q_i\right\}$$

with $\epsilon' = \frac{\epsilon}{2\log N}$. In addition, since $\sum_{i\in[N]} q_i \log q_i \in [-\log N, 0]$, for $q \in \Delta^N$ we have that

$$\begin{aligned}\left|\mathcal{L}_{\mathrm{DRO}}(x) - \widetilde{\mathcal{L}}_{\mathrm{DRO}}(x)\right| &= \left|\max_{q\in\mathcal{U}\subset\Delta^N}\left\{\sum_{i\in[N]} q_i\ell_i(x)\right\} - \max_{q\in\mathcal{U}\subset\Delta^N}\left\{\sum_{i\in[N]} q_i\ell_i(x) - \epsilon' q_i\log q_i\right\}\right| \\ &\le \left|\sum_{i\in[N]} \epsilon' q_i \log q_i\right| \le \epsilon/2.\end{aligned}$$

Thus for x satisfying $\widetilde{\mathcal{L}}_{\mathrm{DRO}}(x) - \min_{x\in\mathcal{X}} \widetilde{\mathcal{L}}_{\mathrm{DRO}}(x) \le \epsilon/2$ we have that $\mathcal{L}_{\mathrm{DRO}}(x) - \min_{x_\star\in\mathcal{X}} \mathcal{L}_{\mathrm{DRO}}(x_\star) \le \epsilon$ as well.

Next, we show that $\widetilde{\mathcal{L}}_{\mathrm{DRO}}$ is $\widetilde{O}(1/\epsilon)$-smooth when each ℓ_i is $O(1/\epsilon)$-smooth. For $q \in \mathcal{U}$, the function $\Psi(q) = \sum_{i\in[N]} \epsilon' q_i \log(q_i)$ is ϵ'-strongly convex w.r.t to the $\|\cdot\|_1$ norm, therefore the conjugate function $\Psi^*(\cdot)$ is $\frac{1}{\epsilon'}$-smooth w.r.t to the dual norm $\|\cdot\|_\infty$, such that

$$\|\nabla\Psi^*(v) - \nabla\Psi^*(v')\|_1 \le \frac{1}{\epsilon'}\|v - v'\|_\infty \tag{19}$$

Algorithm 1: Stochastic accelerated proximal point method

Input: BROO $\mathcal{O}_{\lambda,\delta}(\cdot)$, $T_{\max}$, initialization $x_0 = v_0$ and $A_0 \geq 0$
Parameters: Approximation parameters $\{\delta_k, \beta_k, \sigma_k\}$, stopping parameters $A_{\max}$ and $K_{\max}$
for $k = 0, 1, 2, \cdots$ **do**
  $\lambda_{k+1} = \lambda\text{-}\textsc{Bisection}(x_k, v_k, A_k)$
  $a_{k+1} = \frac{1}{2\lambda_{k+1}}\sqrt{1 + 4\lambda_{k+1}A_k}$ and $A_{k+1} = A_k + a_{k+1}$
  $y_k = \frac{A_k}{A_{k+1}}x_k + \frac{a_{k+1}}{A_{k+1}}v_k$
  $x_{k+1} = \mathcal{O}_{\lambda_{k+1},\delta_{k+1}}(y_k)$
  $g_{k+1} = \textsc{MorGradEst}\Big(\mathcal{O}_{\lambda,\delta}(\cdot), y_k, \lambda_k, \frac{\beta_{k+1}}{\lambda_{k+1}}, \frac{\sigma_k^2}{\lambda_{k+1}}\Big)$
  $v_{k+1} = \mathrm{Proj}_{\mathcal{X}}\big(v_k - \frac{1}{2}a_{k+1}g_{k+1}\big)$
  if $A_{k+1} \geq A_{\max}$ **or** $k + 1 = K_{\max}$ **then**
    return x_{k+1}

In addition, let $q^\star(\ell(x)) = \nabla\Psi^*(\ell(x)) = \mathrm{argmax}_{q\in\mathcal{U}}\{\ell(x) - \Psi(q)\}$ and note that $\widetilde{\mathcal{L}}_{\text{DRO}}(x) = \Psi^*(\ell(x))$. Using this for every $x, y \in \mathcal{X}$ we have

$$\begin{aligned}
\left\|\nabla\widetilde{\mathcal{L}}_{\text{DRO}}(x) - \nabla\widetilde{\mathcal{L}}_{\text{DRO}}(y)\right\| &= \left\|\sum_{i\in[N]} \nabla\ell_i(x)q_i^\star(\ell(x)) - \nabla\ell_i(y)q_i^\star(\ell(y))\right\| \\
&\leq \left\|\sum_{i\in[N]} \nabla\ell_i(x)[q_i^\star(\ell(x)) - q_i^\star(\ell(y))]\right\| + \left\|\sum_{i\in[N]} q_i^\star(\ell(y))[\nabla\ell_i(x) - \nabla\ell_i(y)]\right\| \\
&\overset{(i)}{\leq} G\|q^\star(\ell(x)) - q^\star(\ell(y))\|_1 + L\|x - y\| \\
&\overset{(ii)}{\leq} \frac{G}{\epsilon'}\|\ell(x) - \ell(y)\|_\infty + L\|x - y\| \\
&\overset{(iii)}{\leq} \left(\frac{G^2}{\epsilon'} + L\right)\|x - y\|
\end{aligned}$$

where (i) follows since every ℓ_i is G Lipschitz and L smooth, in addition for every $\ell_i(x) \in \mathbb{R}$ we have that $q^\star(\ell(x)) \in \mathcal{U} \subset \Delta^N$, therefore $\sum_{i\in[N]} q_i^\star(\ell(x)) = 1$, (ii) follows from the inequality in (19) and (iii) follows since each ℓ_i is G-Lipschitz.

Since $\widetilde{\mathcal{L}}_{\text{DRO}}$ is $\widetilde{L} = L + \frac{G^2}{\epsilon'}$-smooth, Nesterov's accelerated gradient descent [44] method is efficient for minimizing it. This method finds x such that $\widetilde{\mathcal{L}}_{\text{DRO}}(x) - \min_x \widetilde{\mathcal{L}}_{\text{DRO}}(x) \leq \epsilon/2$ with $O\Big(\sqrt{\widetilde{L}R^2/\epsilon}\Big) = \widetilde{O}\big(\frac{GR}{\epsilon}\big)$ iterations when $L = O(G^2/\epsilon)$. Note that in every iteration we need to compute the full gradient of $\widetilde{\mathcal{L}}_{\text{DRO}}$, which requires to evaluate each $\nabla\ell_i$ and ℓ_i. Therefore, in the weakly smooth setting the complexity of this method is $\widetilde{O}(\frac{NGR}{\epsilon})$.

B Proof of Proposition 1

In this section we provide the proof for Proposition 1 that follows from the analysis of Algorithm 1; this section closely follows Asi et al. [3], and we refer the readers to that paper for a more detailed exposition.

We begin with a short description of Algorithm 1. This algorithm iteratively computes a $\frac{\lambda\delta^2}{2}$-approximate minimizer of $F_\lambda(x) = F(x) + \frac{\lambda}{2}\|x - y\|^2$ within a small ball of radius r around y. To keep the ball constraint inactive it uses a bisection procedure that outputs the regularization strength value λ, such that for the minimizer $\hat{x} = \mathrm{argmin}_{x\in\mathcal{X}} F_\lambda(x)$ with high probability we have $\|\hat{x} - y\| \leq r$. It then computes a (nearly) unbiased gradient estimator of the Moreau envelope $M_\lambda(y) = \min_{x\in\mathcal{X}} F_\lambda(x)$ and uses a momentum-like scheme to compute the next ball center.

Algorithm 2: MorGradEst

Input: BROO $\mathcal{O}_{\lambda,\delta}(\cdot)$, query point y, regularization λ, bias β, and variance σ^2.

Set $T_{\max} = \frac{2G^2}{\lambda^2 \min\left\{\beta^2, \frac{1}{2}\sigma^2\right\}}$, $T_0 = \frac{14G^2 \log T_{\max}}{\sigma^2}$, and $\delta_0 = \frac{G}{\lambda\sqrt{T_0}}$

$x_0 = \mathcal{O}_{\lambda,\delta_0}(y_k)$

Sample $J \sim \mathrm{Geom}\left(\frac{1}{2}\right)$

if $2^J \le T_{\max}$ **then**

  $\delta_J = \frac{G}{\lambda\sqrt{2^J T_0}}$

  $x_J, x_{J-1} = \mathcal{O}_{\lambda,\delta_J}(y), \mathcal{O}_{\lambda,\delta_{J-1}}(y)$

  $\hat{x} = x_0 + 2^J(x_J - x_{J-1})$

else

  $\hat{x} = x_0$

return $\lambda(y - \hat{x})$

Algorithm 1 is a variant of the accelerated proximal point method in [3, Algorithm 4], and we now describe the differences between the two. The main difference is that the bias reduction scheme of Asi et al. [3] averages $\widetilde{O}\left(\frac{G^2}{\sigma^2} + 1\right)$ calls to an estimator with accuracy $\delta'_J = O\left(\frac{G}{\lambda 2^{J/2}}\right)$ where $J \sim \mathrm{Geom}\left(\frac{1}{2}, T_{\max}\right)$ and G is the Lipschitz constant of F. In contrast, we use a single call to an estimator with higher accuracy $\delta_J = \widetilde{O}\left(\delta'_J / \sqrt{G^2/\sigma^2}\right)$; cf. the implementations of MorGradEst subroutine in each algorithm. There are additional differences between the error tolerance settings of our algorithms, as described below.

The guarantees of [3, Proposition 2] require the following choice of approximation parameters:

$$\varphi_k = \frac{\lambda_k \delta_k^2}{2} = \frac{\epsilon}{60\lambda_k a_k}, \ \beta_k = \frac{\epsilon}{120R} \text{ and } \sigma_k^2 = \frac{\epsilon}{60a_k}$$

which implies $\max_{k \le K_{\max}}\left\{\lambda_k a_k \varphi_k + a_k \sigma_k^2 + 2R\beta_k\right\} \le \frac{\epsilon}{20}$ (where β_k in our notation is δ_k in the notation of [3]). These parameters were chosen so that together with [3, Lemma 6] they give the following bound

$$\mathbb{E}\left[A_K(F(x_K) - F(x_\star)) + \frac{1}{6}\sum_{i \le K} \lambda_i A_i \|\hat{x}_i - y_{i-1}\|^2\right] \le A_0(F(x_0) - F(x_\star)) + \frac{\epsilon}{20}\mathbb{E}A_K + R^2, \tag{20}$$

which is a key component in the proof of [3, Proposition 2]. However, for improved efficiency we set different parameters:

$$\varphi_k = \frac{\lambda_k \delta_k^2}{2} = \frac{\lambda_k r^2}{900 \log^3\left(\frac{GR^2}{\epsilon r}\right)} \text{ and } \sigma_k^2 = \frac{\lambda_k^2 r^2}{900 \log^3\left(\frac{GR^2}{\epsilon r}\right)}.$$

To obtain the guarantees of [3, Proposition 2] for our implementation, in the following lemma we reprove [3, Lemma 6] with our parameters and show the same bound as in (20) (with a slightly different constant factor).

Lemma 8 (modification of [3, Lemma 6])**.** *Let F satisfy Assumption 1 with a minimizer $x_\star$. Let*

$$\varphi_k = \frac{\lambda_k r^2}{900 \log^3\left(\frac{GR^2}{r\epsilon}\right)}, \ \sigma_k^2 = \frac{\lambda_k^2 r^2}{900 \log^3\left(\frac{GR^2}{r\epsilon}\right)}, \ \beta_k = \frac{\epsilon}{120R}, \ A_0 = \frac{R}{G} \text{ and } A_{\max} = \frac{9R^2}{\epsilon}.$$

Define $\hat{x}_k := \mathrm{argmin}_{x \in \mathcal{X}}\left\{F(x) + \frac{\lambda}{2}\|x - y_k\|^2\right\}$ and assume that for each k we have $\|\hat{x}_k - y_{k-1}\| \le r$ and that one of the following must occur

1. *$\lambda_k < 2\lambda_{\mathrm{m}} = \frac{2\epsilon}{r^{4/3}R^{2/3}} \log^2\left(\frac{GR^2}{r\epsilon}\right)$, or*
2. *$\|\hat{x}_k - y_{k-1}\| \ge \frac{3}{4}r$.*

Then we have that

$$\mathbb{E}\left[A_K\Big(F(x_K)-F(x_\star)-\frac{\epsilon}{20}\Big)+\frac{1}{12}\sum_{i\le K}\lambda_i A_i\|\hat{x}_i-y_{i-1}\|^2\right]\le A_0(F(x_0)-F(x_\star))+R^2.$$

Proof. Define the filtration

$$\mathcal{F}_k=\sigma(x_1,v_1,A_1,\zeta_1,\ldots,x_k,v_k,A_k,\zeta_k)$$

where ζ_i is the internal randomness of λ-BISECTION(x_k,v_k,A_k) and note that $A_{k+1},y_k,\hat{x}_{k+1}$ are deterministic when conditioned on x_k,v_k,A_k,ζ_k. Following the proof of [3, Lemma 6], we define

$$M_k=A_k\Big(F(x_k)-F(x_\star)-\frac{\epsilon}{20}\Big)+\frac{1}{12}\sum_{i\le k}\lambda_i A_i\|\hat{x}_i-y_{i-1}\|^2+\|v_k-x_\star\|^2$$

and show it is a supermartingle adapted to filteration $\mathcal{F}_k$. From [3, Lemma 5] we have

$$\begin{aligned}\mathbb{E}[M_{k+1}|\mathcal{F}_k]&\le A_k(F(x_k)-F(x_\star))+\|v_k-x_\star\|^2-\frac{1}{6}\lambda_{k+1}A_{k+1}\|\hat{x}_{k+1}-y_k\|^2\\&\quad+\mu_{k+1}-A_{k+1}\frac{\epsilon}{c}+\frac{1}{12}\sum_{i\le k+1}\lambda_i A_i\|\hat{x}_i-y_{i-1}\|^2\end{aligned}\tag{21}$$

where

$$\mu_{k+1}\coloneqq\lambda_{k+1}a_{k+1}^2\varphi_{k+1}+a_{k+1}^2\sigma_{k+1}^2+2Ra_{k+1}\beta_{k+1}.$$

Substituting the values of $\varphi_{k+1},\sigma_{k+1}^2,\beta_{k+1}$ into the definition of μ_{k+1} gives

$$\mu_{k+1}=\frac{2\lambda_{k+1}^2a_{k+1}^2r^2}{900\log^3\big(\frac{GR^2}{r\epsilon}\big)}+a_{k+1}\frac{\epsilon}{60}.$$

Recall that $A_{k+1}=a_{k+1}^2\lambda_{k+1}$, therefore

$$\mu_{k+1}=\frac{2\lambda_{k+1}A_{k+1}r^2}{900\log^3\big(\frac{GR^2}{r\epsilon}\big)}+a_{k+1}\frac{\epsilon}{60}=\frac{2a_{k+1}\lambda_{k+1}^{3/2}\sqrt{A_{k+1}}r^2}{900\log^3\big(\frac{GR^2}{r\epsilon}\big)}+a_{k+1}\frac{\epsilon}{60}.$$

If $\|\hat{x}_k-y_{k-1}\|\ge\frac{3}{4}r$, we have that

$$\mu_{k+1}\le\frac{1}{12}\lambda_{k+1}A_{k+1}\|\hat{x}_{k+1}-y_k\|^2+a_{k+1}\frac{\epsilon}{60}.$$

Else, if $\lambda_k<2\lambda_{\mathrm{m}}=\frac{2\epsilon}{r^{4/3}R^{2/3}}\log^2\Big(\frac{GR^2}{r\epsilon}\Big)$, we have

$$\mu_{k+1}<\frac{2a_{k+1}2\lambda_{\mathrm{m}}^{3/2}\sqrt{A_{k+1}}r^2}{900\log^3\big(\frac{GR^2}{r\epsilon}\big)}+a_{k+1}\frac{\epsilon}{60}=\frac{4a_{k+1}\epsilon^{3/2}\sqrt{A_{k+1}}}{900R}+a_{k+1}\frac{\epsilon}{60}.$$

Now, note that $\lambda_{\mathrm{m}}\ge\widetilde{O}\big(\frac{\epsilon}{R^2}\big)\ge\frac{1}{A_{\max}}=\frac{\epsilon}{9R^2}$ and therefore $a_K=\sqrt{\frac{1}{\lambda_K^2}+\frac{4A_{K-1}}{\lambda_K}}\le1.2A_{\max}$. From the definition of A_k we have that $A_{K_{\max}-1}\le A_{\max}$, therefore for every $k\le K$

$$A_k\le A_K=a_K+A_{K-1}\le2.2A_{\max}.$$

Thus, when $\lambda_k<2\lambda_{\mathrm{m}}$ the bound on μ_{k+1} becomes

$$\mu_{k+1}<\frac{4a_{k+1}\epsilon^{3/2}\sqrt{2.2\frac{R^2}{\epsilon}}}{900R}+a_{k+1}\frac{\epsilon}{60}\le a_{k+1}\frac{\epsilon}{20}\le\frac{1}{12}\lambda_{k+1}A_{k+1}\|\hat{x}_{k+1}-y_k\|^2+a_{k+1}\frac{\epsilon}{20}$$

where the last inequality follows since $A_k\ge0$ and $\lambda_k\ge0$. Therefore, for every $k\le K$ we have that

$$\mu_{k+1}\le\frac{1}{12}\lambda_{k+1}A_{k+1}\|\hat{x}_{k+1}-y_k\|^2+a_{k+1}\frac{\epsilon}{20}.$$

Noting that $\mathbb{E}|M_k| < \infty$ and substituting the bound on μ_{k+1} into (21) we get

$$\mathbb{E}[M_{k+1}|\mathcal{F}_k] \le A_k\Big(F(x_k) - F(x_\star) - \frac{\epsilon}{20}\Big) + \|v_k - x_\star\|^2 + \frac{1}{12}\sum_{i\le k}\lambda_i A_i\|\hat{x}_i - y_{i-1}\|^2 = M_k.$$

Therefore M_k is a supermartingle adapted to filtration $\mathcal{F}_k$. Since K is a stopping time adapted to filtration $\mathcal{F}_k$, by the optional stopping theorem for supermartingles we have

$$\mathbb{E}M_K \le M_0 = A_0\Big(F(x_0) - F(x_\star) - \frac{\epsilon}{20}\Big) + \|v_0 - x_\star\|^2 \le A_0(F(x_0) - F(x_\star)) + R^2.$$

□

For line 2 of Algorithm 1, we use the same λ-BISECTION implementation of [15]. This implementation requires calling to a *high-probability Ball Regularized Optimization Oracle* (high-probability BROO) and we give the definition of it bellow.

Definition 2. *An algorithm is a probability $1-p$ Ball Regularized Optimization Oracle of radius r (r-BROO) for function $F : \mathcal{X} \to \mathbb{R}$ if for query point $\bar{x} \in \mathcal{X}$,probability p, regularization parameter $\lambda > 0$ and desired accuracy $\delta > 0$ it returns $\mathcal{O}_{\lambda,\delta}(\bar{x}) \in \mathcal{X}$ that with probability at least $1-p$ satisfies*

$$F(\mathcal{O}_{\lambda,\delta}(\bar{x})) + \frac{\lambda}{2}\|\mathcal{O}_{\lambda,\delta}(\bar{x}) - \bar{x}\|^2 \le \min_{x\in\mathbb{B}_r(\bar{x})\cap\mathcal{X}}\Big\{F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2\Big\} + \frac{\lambda}{2}\delta^2. \tag{22}$$

In the following lemma we give the complexity guarantee for a high probability r-BROO.

Lemma 9. *Let $\mathcal{C}_F$ be the complexity of evaluating F exactly, and $\mathcal{C}_\lambda(\delta)$ be an r-BROO implementation complexity. Then the complexity of implementing a probability $1-p$ r-BROO of Definition 2 is*

$$\log\Big(\frac{1}{p}\Big)\Big[\mathcal{C}_\lambda\Big(\frac{\delta}{\sqrt{2}}\Big) + \mathcal{C}_F\Big].$$

Proof. To obtain a high-probability r-BROO we run $\log_2\big(\frac{1}{p}\big)$ copies of r-BROO with query point $\bar{x}$, regularization strength λ and accuracy $\delta/\sqrt{2}$ and take the best output, i.e., the output with the minimal value of F. Applying Markov's inequality to a single run of r-BROO with output x, gives

$$\mathbb{P}\Big(F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2 - \min_{x\in\mathbb{B}_r(\bar{x})\cap\mathcal{X}}\Big\{F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2\Big\} \ge \frac{\lambda}{2}\delta^2\Big) \le \frac{1}{2}$$

therefore with probability at least $1 - \big(\frac{1}{2}\big)^{\log_2\left(\frac{1}{p}\right)} = 1 - p$, the best output x' satisfies

$$F(x') + \frac{\lambda}{2}\|x' - \bar{x}\|^2 - \min_{x\in\mathbb{B}_r(\bar{x})\cap\mathcal{X}}\Big\{F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2\Big\} \le \frac{\lambda}{2}\delta^2.$$

Procedure require $\log(1/p)$ BROO calls and the same number of exact function evaluation (to choose the best BROO output), resulting in the claimed complexity bound $\log\big(\frac{1}{p}\big)\Big[\mathcal{C}_\lambda\Big(\frac{\delta}{\sqrt{2}}\Big) + \mathcal{C}_F\Big]$. □

To compute the gradient estimator in line 6 of Algorithm 1, we use Algorithm 2. Our implementation is slightly different than [3] and in the following lemma we show it produces an estimator with the same bias and variance guarantees of [3].

Lemma 10. *Let $F : \mathcal{X} \to \mathbb{R}$ satisfy Assumption 1 and for query point $y \in \mathcal{X}$ and regularization strength $\lambda > 0$ define $x' = \mathrm{argmin}_{x\in\mathcal{X}}\{F(x) + \frac{\lambda}{2}\|x - y\|^2\}$ and $g = \lambda(y - x')$. Then, for any bias and variance parameters $\beta, \sigma > 0$ Algorithm 2 outputs $\hat{g} = \lambda(y - \hat{x})$ satisfying*

$$\|\mathbb{E}\hat{g} - g\| \le \beta \;\text{ and }\; \mathbb{E}\|\hat{g} - \mathbb{E}\hat{g}\|^2 \le \sigma^2.$$

Proof. First note that if $x = \mathcal{O}_{\lambda,\delta}(\bar{x})$ is the output of an r-BROO with accuracy δ and if $x' = \mathrm{argmin}_{x\in\mathbb{B}_r(\bar{x})} F(x)$, from Definition 1 and strong convexity (of $F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2$) we have:

$$\frac{\lambda}{2}\mathbb{E}\|x - x'\|^2 \le \mathbb{E}\Big[F(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2\Big] - \Big[F(x') + \frac{\lambda}{2}\|x' - \bar{x}\|^2\Big] \le \frac{\lambda\delta^2}{2}$$

giving

$$\mathbb{E}\|x - x'\|^2 \le \delta^2. \tag{23}$$

Let $j_{\max} = \lfloor \log_2 T_{\max} \rfloor$, then from the definition of $\hat{x}$ in Algorithm 2 we have that

$$\mathbb{E}\hat{x} = \mathbb{E}x_0 + \sum_{j=1}^{j_{\max}} \mathbb{P}(J = j) 2^j (\mathbb{E}x_j - \mathbb{E}x_{j-1}) = \mathbb{E}x_{j_{\max}}.$$

Therefore

$$\|\mathbb{E}\hat{g} - g\| = \lambda \|\mathbb{E}x_{j_{\max}} - x'\| \overset{(i)}{\le} \lambda \sqrt{\mathbb{E}\|x_{j_{\max}} - x'\|^2} \overset{(ii)}{\le} \lambda \delta_{j_{\max}} = \lambda \frac{G\sqrt{\min\{\beta^2, \frac{1}{2}\sigma^2\}}}{\lambda\sqrt{2G^2 T_0}} \le \min\left\{\beta, \frac{1}{2}\sigma\right\}$$

with (i) following from Jensen inequality and (ii) following from the guarantee in (23). To bound the variance of $\hat{g}$ note that

$$\mathbb{E}\|\hat{g} - \mathbb{E}\hat{g}\|^2 = \lambda^2 \mathbb{E}\|\hat{x} - \mathbb{E}\hat{x}\|^2 \le \lambda^2 \mathbb{E}\|\hat{x} - x'\|^2 \le \lambda^2 \left(2\mathbb{E}\|\hat{x} - x_0\|^2 + 2\mathbb{E}\|x_0 - x'\|^2\right)$$

where the last inequality follows from $\|a + b\|^2 \le 2\|a\|^2 + 2\|b\|^2$. The definition of $\hat{x}$ gives

$$\mathbb{E}\|\hat{x} - x_0\|^2 = \sum_{j=1}^{j_{\max}} 2^j \mathbb{E}\|x_j - x_{j-1}\|^2$$

and from the guarantee in (23) we get

$$\mathbb{E}\|x_j - x_{j-1}\|^2 \le 2\mathbb{E}\|x_j - x'\|^2 + 2\mathbb{E}\|x_{j-1} - x'\|^2 \le 6\delta_j^2 = \frac{6G^2}{\lambda^2 T_0 2^j}$$

thus

$$\mathbb{E}\|\hat{x} - x_0\|^2 \le j_{\max} \frac{6G^2}{\lambda^2 T_0}.$$

In addition we have $\mathbb{E}\|x_0 - x'\|^2 \le \delta_0^2 = \frac{G^2}{\lambda^2 T_0}$ and substituting back we get

$$\mathbb{E}\|\hat{g} - \mathbb{E}\hat{g}\|^2 = \lambda^2 \mathbb{E}\|\hat{x} - \mathbb{E}\hat{x}\|^2 \le \lambda^2 \left(12 j_{\max} \frac{G^2}{\lambda^2 T_0} + 2\frac{G^2}{\lambda^2 T_0}\right) \le 14 j_{\max} \frac{G^2}{T_0} \le \sigma^2.$$

$\square$

Combining the previous statements, we prove our main proposition.

Proposition 1. *Let F satisfy Assumption 1, let $\mathcal{C}_F$ be the complexity of evaluating F exactly, and let $\mathcal{C}_\lambda(\delta)$ bound the complexity of an r-BROO query with δ, λ. Assume that $\mathcal{C}_\lambda(\delta)$ is non-increasing in λ and at most polynomial in $1/\delta$. For any $\epsilon > 0$, Algorithm 1 returns x such that $F(x) - \min_{x_\star \in \mathcal{X}} F(x_\star) \le \epsilon$ with probability at least $\frac{1}{2}$. For $m_\epsilon = O\big(\log \frac{GR^2}{\epsilon r}\big)$ and $\lambda_{\mathrm{m}} = O\big(\frac{m_\epsilon^2 \epsilon}{r^{4/3} R^{2/3}}\big)$, the complexity of the algorithm is*

$$O\left(\left(\frac{R}{r}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon} \frac{1}{2^j} \mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r}{2^{j/2} m_\epsilon^2}\right)\right) m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r) + \mathcal{C}_F) m_\epsilon^3, \right]\right). \tag{5}$$

Proof. We divide the proof into a correctness argument and a complexity calculation.

Correctness. To prove the correctness of Proposition 1 we first need to show that the guarantees of [3, Proposition 2] still hold for our implementation that includes different parameters φ_k and σ_k^2:

$$\varphi_k = \frac{\lambda_k r^2}{900 \log^3\left(\frac{GR^2}{\epsilon r}\right)} \text{ and } \sigma_k^2 = \frac{\lambda_k^2 r^2}{900 \log^3\left(\frac{GR^2}{\epsilon r}\right)}$$

and different implementation of lines 5 and 6.

Following Lemma 8, the guarantees of [3, Proposition 2] are still valid with our different choice of φ_k and σ_k^2. In addition, following Lemma 10, our implementation of line 6 is valid since it produces the same guarantees that the implementation in [3] gives. Last, if the implementation of line 5 is valid it needs to satisfy

$$\mathbb{E}\left[F(x_{k+1}) + \frac{\lambda_{k+1}}{2}\|x_{k+1} - y_k\|^2\right] - \min_{x\in\mathcal{X}}\left\{F(x) + \frac{\lambda_{k+1}}{2}\|x - y_k\|^2\right\} \le \varphi_{k+1}.$$

Note that x_{k+1} in our implementation is the output of r-BROO with accuracy $\delta_k \le \frac{r}{\sqrt{14\cdot 900}\log^{3/2}\left(\frac{GR^2}{\epsilon r}\right)}$, therefore by Definition 1 it satisfies

$$\begin{aligned}\mathbb{E}\left[F(x_{k+1}) + \frac{\lambda_{k+1}}{2}\|x_{k+1} - y_k\|^2\right] - \min_{x\in\mathbb{B}_r(y_k)}\left\{F(x) + \frac{\lambda_{k+1}}{2}\|x - y_k\|^2\right\} &\le \frac{\lambda_{k+1}\delta_{k+1}^2}{2} \\ &\le \frac{\lambda_{k+1} r^2}{900\log^3\left(\frac{GR^2}{\epsilon r}\right)} = \varphi_{k+1}\end{aligned}$$

and for valid output of λ-BISECTION we have

$$\min_{x\in\mathbb{B}_r(y_k)}\left\{F(x) + \frac{\lambda_{k+1}}{2}\|x - y_k\|^2\right\} = \min_{x\in\mathcal{X}}\left\{F(x) + \frac{\lambda_{k+1}}{2}\|x - y_k\|^2\right\}$$

implying that line 5 is valid. Now let p_{BROO} be the probability that all calls to r-BROO result in a valid output. Following [3, Proposition 2], for $K_{\max} = O\left(\left(\frac{R}{r}\right)^{2/3} m_\epsilon\right)$ with probability at least $1 - \left(1 - \frac{2}{3}\right) - (1 - p_{\text{BROO}}) = p_{\text{BROO}} - \frac{1}{3}$ the algorithm outputs x that satisfies

$$F(x) - F(\hat{x}) \le \epsilon/2.$$

Let p be the probability that a single BROO implementation produce invalid output and let $K_{\text{bisect-max}}$ be the maximal number of calls to high-probability r-BROO within line 2. Then, $p_{\text{BROO}} \ge 1 - K_{\max}K_{\text{bisect-max}}p$ and for

$$p \le \frac{1}{6K_{\max}K_{\text{bisect-max}}}$$

with probability at least $\frac{1}{2}$ Algorithm 1 outputs $\frac{\epsilon}{2}$-suboptimal minimizer of F.

Complexity. To bound the complexity of Algorithm 1 we first bound the complexity of line 2 and the complexity of line 6 in the k-th iteration of Algorithm 1. Note that, for x_{k+1} in line 5 we can use x_0 from Algorithm 2, and therefore the complexity of line 6 already includes the complexity of line 5.

Following [15, Proposition 2], λ-BISECTION in line 2 requires $m_\epsilon = O\left(\log\left(\frac{GR^2}{\epsilon r}\right)\right)$ calls to a high-probability r-BROO with accuracy $\delta = \frac{r}{30}$. From Lemma 9 the complexity of a single call to a probability $1-p$ r-BROO is $O\left(\log\left(\frac{1}{p}\right)[\mathcal{C}_\lambda(r) + \mathcal{C}_F]\right)$. We set $p = \frac{1}{6K_{\max}K_{\text{bisect-max}}}$, and since $K_{\max} = O\left(\left(\frac{R}{r}\right)^{2/3} m_\epsilon\right)$ and $K_{\text{bisect-max}} = m_\epsilon$ we get $\log\left(\frac{1}{p}\right) = O\left(\log\left(\frac{R}{r}m_\epsilon^2\right)\right) \le m_\epsilon$. Therefore, the total complexity of λ-BISECTION is $O\left(m_\epsilon^2[\mathcal{C}_\lambda(r) + \mathcal{C}_F]\right)$.

For the complexity of Line 6 note that Algorithm 2 calls to an r-BROO with accuracy $\delta_J \ge \frac{r}{30\sqrt{14}2^{J/2}m_\epsilon^2}$ where $J \sim \text{Geom}\left(\frac{1}{2}, j_{\max}\right)$ and $j_{\max} \le m_\epsilon$. Therefore, we can bound the complexity of Algorithm 2 by

$$O\left(\sum_{j=0}^{m_\epsilon} \frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r}{2^{j/2}m_\epsilon^2}\right)\right).$$

The complexity of the entire algorithm is at most $K_{\max}$ times the complexity of a single iteration. Using $K_{\max} = O\Big(\big(\frac{R}{r}\big)^{2/3} m_\epsilon\Big)$, the total complexity becomes

$$O\left(\left(\frac{R}{r}\right)^{2/3}\left(m_\epsilon \sum_{j=0}^{m_\epsilon} \frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r}{2^{j/2}m_\epsilon^2}\right) + m_\epsilon^3[\mathcal{C}_\lambda(r) + \mathcal{C}_F]\right)\right).$$

□

C Group DRO

In this section we provide the proofs for the results of Section 3. In Appendix C.1 we first prove that the group-softmax is a uniform approximation of $\mathcal{L}_{\text{g-DRO}}$, then, through extension of [15], we show that we can approximate $\mathcal{L}_{\text{g-DRO}}$ using the group-exponentiated softmax instead. Next, in Appendix C.2 we prove Lemma 2 and bound the moments of the MLMC and gradient estimators. We then prove the complexity guarantees of Theorem 1 in Appendix C.3. Last, in Appendix C.4 under the mean-square smoothness assumption, we provide the properties of the gradient estimator in (9) and in Appendix C.5 we prove the complexity guarantees of Theorem 2.

C.1 Exponentiated group-softmax

Recall the definition of the (regularized) group-softmax

$$\mathcal{L}_{\text{smax},\epsilon,\lambda}(x) \coloneqq \epsilon' \log\left(\sum_{i\in[M]} e^{\frac{\mathcal{L}_i(x)}{\epsilon'}}\right) + \frac{\lambda}{2}\|x-\bar{x}\|^2 \;\text{ where }\; \mathcal{L}_i(x) = \sum_{j\in[N]} w_{ij}\ell_j(x)$$

with $\mathcal{L}_{\text{smax},\epsilon,0}(x) = \mathcal{L}_{\text{smax},\epsilon}(x)$. In addition, recall the definition of the (regularized) group-exponentiated softmax

$$\Gamma_{\epsilon,\lambda}(x) \coloneqq \sum_{i\in[M]} \bar{p}_i\gamma_i(x) \;\text{ where }\; \gamma_i(x) = \epsilon' e^{\frac{\mathcal{L}_i(x)-\mathcal{L}_i(\bar{x})+\frac{\lambda}{2}\|x-\bar{x}\|^2}{\epsilon'}} \;\text{ and }\; \bar{p}_i = \frac{e^{\frac{\mathcal{L}_i(\bar{x})}{\epsilon'}}}{\sum_{i\in[M]} e^{\frac{\mathcal{L}_i(\bar{x})}{\epsilon'}}}.$$

Lemma 11. *Let $\mathcal{L}_{\text{smax},\epsilon}$ be the group-softmax defined in eq. (6) and $\mathcal{L}_{\text{g-DRO}}$ be the Group DRO objective defined in (2). Let $\epsilon > 0$ and $\epsilon' = \epsilon/(2\log M) > 0$. Then for all $x \in \mathcal{X}$ we have that*

$$|\mathcal{L}_{\text{g-DRO}}(x) - \mathcal{L}_{\text{smax},\epsilon}(x)| \le \epsilon/2$$

Proof. First note that

$$\mathcal{L}_{\text{g-DRO}}(x) \coloneqq \max_{i\in[M]} \sum_{j=1}^{N} w_{ij}\ell_j(x) = \max_{q\in\Delta^M} \sum_{i\in[M]} q_i\mathcal{L}_i(x)$$

and

$$\mathcal{L}_{\text{smax},\epsilon}(x) = \epsilon' \log\left(\sum_{i\in[M]} e^{\frac{\mathcal{L}_i(x)}{\epsilon'}}\right) = \max_{q\in\Delta^M}\left\{\sum_{i\in[M]} q_i\mathcal{L}_i(x) - \epsilon' q_i \log q_i\right\}.$$

In addition, for $q \in \Delta^M$ we have that $\sum_{i\in[M]} q_i \log qi \in [-\log M, 0]$. Combining these facts gives

$$\begin{aligned}|\mathcal{L}_{\text{smax},\epsilon}(x) - \mathcal{L}_{\text{g-DRO}}(x)| &= \left|\max_{q\in\Delta^M}\left\{\sum_{i\in[M]} q_i\mathcal{L}_i(x) - \epsilon' q_i\log q_i\right\} - \max_{q\in\Delta^M}\sum_{i\in[M]} q_i\mathcal{L}_i(x)\right| \\ &\le \left|\epsilon' \sum_{i\in[M]} q_i \log q_i\right| \le \epsilon' \log M = \epsilon/2\end{aligned}$$

□

Lemma 1. *Let each ℓ_i satisfy Assumption 1, and consider the restriction of $\mathcal{L}_{\text{smax},\epsilon,\lambda}$ (6) and $\Gamma_{\epsilon,\lambda}$ (7) to $\mathbb{B}_r(\bar{x})$. Then the functions have the same minimizer $x_\star \in \mathbb{B}_r(\bar{x})$ and, if $\lambda \le O(G/r)$ and $r \le O(\epsilon'/G)$, then (a) $\Gamma_{\epsilon,\lambda}$ is $\Omega(\lambda)$-strongly convex, (b) each γ_i is $O(G)$-Lipschitz and (c) for every $x \in \mathbb{B}_r(\bar{x})$ we have $\mathcal{L}_{\text{smax},\epsilon,\lambda}(x) - \mathcal{L}_{\text{smax},\epsilon,\lambda}(x_\star) \le O(\Gamma_{\epsilon,\lambda}(x) - \Gamma_{\epsilon,\lambda}(x_\star))$.*

Proof. This lemma is a simple extension of [15, Lemma 1], that considers the exponentiated-softmax:

$$\sum_{i \in M} \epsilon' \frac{e^{l_i(\bar{x})/\epsilon'}}{\sum_{j \in M} e^{f_j(\bar{x})/\epsilon'}} e^{\frac{l_i(x) - l_i(\bar{x}) + \lambda \|x - \bar{x}\|}{\epsilon'}},$$

for some $l_1, \dots, l_N$. The only assumption that [15] have on l_i (for the guarantees we state in Lemma 1) is that each l_i is G-Lipschitz. Note that each $\mathcal{L}_i$ is G-Lipschitz since it is a weighted average of G-Lipschitz functions. Therefore, we can replace l_i in [15, Lemma 1] with the group average $\mathcal{L}_i$ and obtain Lemma 1. $\square$

C.2 MLMC estimator moment bounds

To make the MLMC estimator suitable for both Epoch-SGD and variance reduction methods, we rewrite its definition using more general notation. Specifically, for every $x, x' \in \mathcal{X}$ and $S_1^n \in [N]^n$, let

$$\widehat{\gamma}(x, x'; S_1^n) := \epsilon' e^{\frac{1}{n} \sum_{j=1}^n \frac{\ell_{S_j}(x) - \ell_{S_j}(x') + \frac{\lambda}{2} \|x - \bar{x}\|^2}{\epsilon'}} \tag{24}$$

so that $\widehat{\gamma}(x, \bar{x}; S_1^n) = \widehat{\gamma}(x; S_1^n)$. The MLMC estimator is

$$\text{Draw } J \sim \text{Geom}\left(1 - \tfrac{1}{\sqrt{8}}\right), S_1, \dots, S_n \overset{\text{iid}}{\sim} w_i \text{ and let } \widehat{\mathcal{M}}[\gamma_i(x)] := \widehat{\gamma}(x, \bar{x}; S_1) + \frac{\widehat{\mathcal{D}}_{2^J}}{p_J},$$

where $p_j := \mathbb{P}(J = j) = \left(1/\sqrt{8}\right)^j \left(1 - \frac{1}{\sqrt{8}}\right)$ and, for $n \in 2\mathbb{N}$ we define

$$\widehat{\mathcal{D}}_n := \widehat{\gamma}(x, x'; S_1^n) - \frac{\widehat{\gamma}\left(x, x'; S_1^{\frac{n}{2}}\right) + \widehat{\gamma}\left(x, x'; S_{\frac{n}{2}+1}^n\right)}{2}. \tag{25}$$

Lemma 12. *Let each ℓ_i satisfy Assumption 1, and let $r \le \frac{\epsilon'}{G}$, $\lambda \le \frac{G}{r}$, $\|x - x'\| \le 2r$ and $\|x - \bar{x}\| \le r$. For $\widehat{\mathcal{D}}_n$ defined in (25) we have $\mathbb{E}\left|\widehat{\mathcal{D}}_n\right|^2 \le O\left(\frac{G^4 \|x - x'\|^4}{n^2 \epsilon'^2}\right)$.*

Proof. For abbreviation let $M = \frac{1}{n} \sum_{j \in [n]} \frac{\hat{\ell}_{S_j}(x) + \frac{\lambda}{2} \|x - \bar{x}\|^2}{\epsilon'}$ and $\delta = \frac{1}{n} \sum_{j \in [n/2]} \left(\frac{\hat{\ell}_{S_j}(x) - \hat{\ell}_{S_{j+n/2}}(x)}{\epsilon'} \right)$ where $\hat{\ell}_{S_j}(x) = \ell_{S_j}(x) - \ell_{S_j}(x')$. We have the following bound on $|M|$:

$$|M| \overset{(i)}{\le} \frac{1}{n} \sum_{j \in [n]} \frac{G\|x - x'\| + \frac{G}{2r} \|x - \bar{x}\|^2}{\epsilon'} \overset{(ii)}{\le} \frac{2Gr + Gr/2}{\epsilon'} \overset{(iii)}{\le} 2.5 \tag{26}$$

with (i) following since each ℓ_j is G-Lipschitz and $\lambda \le G/r$, (ii) follows since $\|x - x'\| \le 2r$ and $\|x - \bar{x}\| \le r$, and (iii) since $r \le \epsilon'/G$. For $|\delta|$ we have the bound,

$$|\delta| \le \left| \frac{1}{n} \sum_{j \in [n/2]} \frac{\hat{\ell}_{S_j}(x)}{\epsilon'} \right| + \left| \frac{1}{n} \sum_{j \in [n/2]} \frac{\hat{\ell}_{S_{j+n/2}}(x)}{\epsilon'} \right| \le \frac{G\|x - x'\|}{\epsilon'} \le 2, \tag{27}$$

with the second inequality following since each ℓ_j is G-Lipschitz and the last inequality since $\|x - x'\| \le 2r$ and $r \le \epsilon'/G$. We bound $\left|\widehat{\mathcal{D}}_n\right|$ using the previous guarantees on $|M|$ and $|\delta|$:

$$\left|\widehat{\mathcal{D}}_n\right| = \epsilon' \left| e^M - \frac{e^{M+\delta} + e^{M-\delta}}{2} \right| \le \epsilon' e^M \left(\frac{e^\delta + e^{-\delta}}{2} - 1 \right) \overset{(i)}{\le} \epsilon' e^{2.5} \left(\frac{e^\delta + e^{-\delta}}{2} - 1 \right) \overset{(ii)}{\le} 2e^{2.5} \epsilon' \delta^2$$

where (i) follows from (26) and (ii) from (27) and the inequality $e^x \le 1 + x + 2x^2$ for all $x \le 3$ with $x = \delta$. Therefore, we have that

$$\mathbb{E}\left|\widehat{\mathcal{D}}_n\right|^2 \le 4e^5\epsilon'^2\mathbb{E}\left[\delta^4\right].$$

Let $Y_i = \frac{\hat{\ell}_{S_i}(x)-\hat{\ell}_{S_{i+n/2}}(x)}{n\epsilon'}$ and note that the Lipschitz property of each ℓ_j gives $|Y_i| \le 2\frac{G\|x-x'\|}{n\epsilon'}$, in addition, since the samples $S_1, \ldots, S_n$ are i.i.d we have $\mathbb{E}\frac{\hat{\ell}_{S_i}(x)}{n\epsilon'} = \mathbb{E}\frac{\hat{\ell}_{S_{i+n/2}}(x)}{n\epsilon'}$ and therefore $\mathbb{E}Y_i = 0$. Thus, we can use Lemma 16 below, with $Y_i = \frac{\hat{\ell}_{S_i}(x)-\hat{\ell}_{S_{i+n/2}}(x)}{n\epsilon'}$ and $c = 2\frac{G\|x-x'\|}{n\epsilon'}$, to obtain the bound $\mathbb{E}[\delta]^4 \le O\left(\frac{G^4\|x-x'\|^4}{n^2\epsilon'^4}\right)$. Therefore,

$$\mathbb{E}\left|\widehat{\mathcal{D}}_n\right|^2 \le O\left(\frac{G^4\|x-x'\|^4}{n^2\epsilon'^2}\right).$$

$\square$

Lemma 2. *Let each ℓ_i satisfy Assumption 1, and let $r \le \frac{\epsilon'}{G}$, $\lambda \le \frac{G}{r}$ and $x \in \mathbb{B}_r(\bar{x})$. Then $\widehat{\mathcal{M}}[\gamma_i(x)]$ and $\hat{g}(x)$ are unbiased for $\gamma_i(x)$ and $\nabla\Gamma_{\epsilon,\lambda}(x)$, respectively, and have bounded second moments: $\mathbb{E}\left[\widehat{\mathcal{M}}[\gamma_i(x)]\right]^2 \le O\left(\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^2} + \epsilon'^2\right)$ and $\mathbb{E}\|\hat{g}(x)\|^2 \le O\left(G^2\right)$. In addition, the complexity of computing $\widehat{\mathcal{M}}[\gamma_i(x)]$ and $\hat{g}(x)$ is $O(1)$.*

Proof. We first prove the bias and moment bounds, and then address complexity.

Properties of the MLMC estimator. We first show that the MLMC estimator is unbiased. For every $n \in 2\mathbb{N}$ we have that $\mathbb{E}\widehat{\gamma}(x; S_1^{n/2}) = \mathbb{E}\widehat{\gamma}(x; S_{n/2+1}^n)$, therefore $\mathbb{E}\widehat{\mathcal{D}}_n = \mathbb{E}\widehat{\gamma}(x; S_1^n) - \mathbb{E}\widehat{\gamma}(x; S_1^{n/2})$ and we get

$$\mathbb{E}\left[\widehat{\mathcal{M}}[\gamma_i(x)]\right] = \mathbb{E}\widehat{\gamma}(x; S_1) + \sum_{j=1}^{\infty}\left(\mathbb{E}\widehat{\gamma}(x; S_1^{2^j}) - \mathbb{E}\widehat{\gamma}(x; S_1^{2^{j-1}})\right) = \mathbb{E}\widehat{\gamma}(x; S_1^{\infty}) = \gamma_i(x). \quad (28)$$

To bound the second moment of the estimator we use the inequality $(a+b)^2 \le 2a^2 + 2b^2$, yielding

$$\mathbb{E}\left|\widehat{\mathcal{M}}[\gamma_i(x)]\right|^2 \le 2\mathbb{E}\left|\widehat{\gamma}(x; S_1)\right|^2 + 2\sum_{j=1}^{\infty}\frac{1}{p_j}\mathbb{E}\left|\widehat{\mathcal{D}}_{2^j}\right|^2. \quad (29)$$

Lemma 12 with $x' = \bar{x}$ gives $\mathbb{E}\left|\widehat{\mathcal{D}}_n\right|^2 \le O\left(\frac{G^4\|x-\bar{x}\|^4}{n^2\epsilon'^2}\right)$ and substituting this bound into (29) while noting that $\mathbb{E}\left|\widehat{\gamma}(x; S_1)\right|^2 \le \epsilon'^2 e^3$ gives

$$\mathbb{E}\left|\widehat{\mathcal{M}}[\gamma_i(x)]\right|^2 \le O\left(\epsilon'^2 + \frac{G^4\|x-\bar{x}\|^4}{\epsilon'^2}\sum_{j=1}^{\infty}\left(1 - \frac{1}{\sqrt{8}}\right)\frac{2^{1.5j}}{2^{2j}}\right) = O\left(\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^2} + \epsilon'^2\right). \quad (30)$$

Properties of the gradient estimator. We use the fact that $\widehat{\mathcal{M}}[\gamma_i(x)]$ is unbiased for $\gamma_i(x)$ (shown in eq. (28) above) to argue that gradient estimator is also unbiased:

$$\mathbb{E}[\hat{g}(x)] = \mathbb{E}\left[\frac{1}{\epsilon'}\widehat{\mathcal{M}}[\gamma_i(x)](\nabla\ell_j(x) + \lambda(x-\bar{x}))\right] = \frac{1}{\epsilon'}\sum_{i\in[M]}\sum_{j\in[N]}\bar{p}_i w_{ij}\mathbb{E}\widehat{\mathcal{M}}[\gamma_i(x)](\nabla\ell_j(x) + \lambda(x-\bar{x}))$$

$$= \frac{1}{\epsilon'}\sum_{i\in[M]}\bar{p}_i\gamma_i(x)(\nabla\mathcal{L}_i(x) + \lambda(x-\bar{x})) = \sum_{i\in[M]}\bar{p}_i\nabla\gamma_i(x) = \nabla\Gamma_{\epsilon,\lambda}(x).$$

Next we bound the second moment of the gradient estimator

$$\begin{aligned}
\mathbb{E}\|\hat{g}(x)\|^2 &= \frac{1}{\epsilon'^2}\sum_{i\in[M]}\bar{p}_i\mathbb{E}\Big(\widehat{\mathcal{M}}[\gamma_i(x)]\Big)^2\sum_{j\in[N]}w_{ij}\|\nabla\ell_j(x)+\lambda(x-\bar{x})\|^2 \\
&\stackrel{(i)}{\le} O\left(\left(\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^4}+1\right)\sum_{i\in[M]}\bar{p}_i\sum_{j\in[N]}w_{ij}\|\nabla\ell_j(x)+\lambda(x-\bar{x})\|^2\right) \\
&\stackrel{(ii)}{\le} O\left(\left(\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^4}+1\right)G^2\right)\stackrel{(iii)}{\le} O\big(G^2\big)
\end{aligned}$$

where (i) follows from (30), (ii) follows since each ℓ_j is G-Lipschitz, $\lambda \le \frac{G}{r}$ and $\|x-\bar{x}\|\le r$ and (iii) since $\frac{G^4\|x-\bar{x}\|^4}{\epsilon'^4}\le\frac{G^4r^4}{\epsilon'^4}\le 1$.

Complexity of the MLMC and gradient estimators. $J\sim\mathrm{Geom}(1-\frac{1}{\sqrt{8}})$, therefore

$$\mathbb{E}\big[2^J\big]=\sum_{j=1}^{\infty}\frac{1}{1-\frac{1}{\sqrt{8}}}\left(\frac{1}{\sqrt{8}}\right)^j 2^j = O(1).$$

Note that the estimator $\widehat{\mathcal{M}}[\gamma_i(x)]=\widehat{\gamma}(x,S_1)+\frac{1}{P_J}\widehat{\mathcal{D}}_{2^J}$ requires a single function evaluation for $\widehat{\gamma}(x,S_1)$ and 2^J function evaluations for the term $\widehat{\mathcal{D}}_{2^J}$. As a consequence the computation of $\widehat{\mathcal{M}}[\gamma_i(x)]$ requires only $O(1)$ function evaluations in expectation. To compute $\hat{g}(x)$ we need to compute $\widehat{\mathcal{M}}[\gamma_i(x)]$ and a single sub-gradient, hence, the complexity of computing $\hat{g}(x)$ is also $O(1)$ in expectation. □

C.3 Epoch-SGD BROO implementation

We state below the convergence rate of the Epoch-SGD algorithm.

Lemma 13 (Theorem 5, [28])**.** *Let $F:\mathcal{X}\to\mathbb{R}$ be λ-strongly convex with an unbiased stochastic gradient estimator $\hat{g}$ satisfying $\mathbb{E}\|\hat{g}(x)\|^2\le O\big(G^2\big)$ for all $x\in\mathcal{X}$, and let $x_\star=\mathrm{argmin}_{x\in\mathcal{X}}F(x)$. Epoch-SGD finds an approximate minimizer x that satisfies*

$$\mathbb{E}F(x)-F(x_\star)\le O\left(\frac{G^2}{\lambda T}\right)$$

using T stochastic gradient queries.

Applying this lemma with $F=\Gamma_{\epsilon,\lambda}$, $x\in\mathbb{B}_{r_\epsilon}(\bar{x})$ and $T=O\Big(\frac{G^2}{\lambda^2\delta^2}\Big)$ immediately gives the following guarantee on the BROO implementation complexity.

Theorem 1. *Let each ℓ_j satisfy Assumption 1, let $\epsilon,\delta,\lambda>0$ and let $r_\epsilon=\epsilon/(2G\log M)$. For any query point $\bar{x}\in\mathbb{R}^d$, regularization strength $\lambda\le O(G/r_\epsilon)$ and accuracy δ, EpochSGD [28, Algorithm 1]) with the gradient estimator (8) outputs a valid r_ϵ-BROO response and has complexity $\mathcal{C}_\lambda(\delta)=O\big(N+\frac{G^2}{\lambda^2\delta^2}\big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_{\text{g-DRO}}$ (2) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{11/3}H+\left(\frac{GR}{\epsilon}\right)^2\log^2 H\right)\ \text{where}\ \ H:=M\frac{GR}{\epsilon}.$$

Proof. We divide the proof into correctness and complexity arguments, addressing the BROO implementation and then the overall algorithm.

BROO implementation: correctness. Following Lemma 1 we have that $\Gamma_{\epsilon,\lambda}$ is $\Omega(\lambda)$-strongly convex and Lemma 2 gives $\mathbb{E}\|\hat{g}(x)\|^2\le O\big(G^2\big)$. Thus, we can directly apply Lemma 13 with $F=\Gamma_{\epsilon,\lambda}$, $\mathcal{X}=\mathbb{B}_{r_\epsilon}(\bar{x})$, the gradient estimator $\hat{g}(x)$ defined in (8) and $T=\frac{2c^2G^2}{\lambda^2\delta^2}$ for a constant $c>0$ for which Epoch-SGD outputs x that satisfies

$$\mathbb{E}\Gamma_{\epsilon,\lambda}(x)-\Gamma_{\epsilon,\lambda}(x_\star)\le\frac{cG^2}{\lambda T}\le\frac{\lambda\delta^2}{2c}. \tag{31}$$

Following Lemma 1 there is a value of c such that $\mathbb{E}\mathcal{L}^{\lambda}_{\mathrm{smax},\epsilon}(x) - \mathcal{L}^{\lambda}_{\mathrm{smax},\epsilon}(x_\star) \le c(\mathbb{E}\Gamma_{\epsilon,\lambda}(x) - \Gamma_{\epsilon,\lambda}(x_\star))$ and from (31) we obtain

$$\mathbb{E}\mathcal{L}^{\lambda}_{\mathrm{smax},\epsilon}(x) - \mathcal{L}^{\lambda}_{\mathrm{smax},\epsilon}(x_\star) \le c(\mathbb{E}\Gamma_{\epsilon,\lambda}(x) - \Gamma_{\epsilon,\lambda}(x_\star)) \le \frac{\lambda\delta^2}{2}.$$

Therefore, Epoch-SGD outputs a valid r_ϵ-BROO response for $\mathcal{L}_{\mathrm{smax},\epsilon}$.

BROO implementation: complexity. For the BROO implementation we run Epoch-SGD with the gradient estimator $\hat{g}(x)$ defined in (8) and computation budget $T = O\Big(\frac{G^2}{\lambda^2\delta^2}\Big)$. Therefore, we need to evaluate $O\Big(\frac{G^2}{\lambda^2\delta^2}\Big)$ stochastic gradient estimators with complexity $O(1)$, and our gradient estimator requires additional N functions evaluations for precomputing the sampling probabilities $\{\bar{p}_i\}$. Thus, the total complexity of the BROO implementation is

$$O\bigg(\frac{G^2}{\lambda^2\delta^2} + N\bigg). \tag{32}$$

Minimizing $\mathcal{L}_{\text{g-DRO}}$: correctness. For any $q \in \Delta^M$ note that $\mathcal{L}_q(x) := \sum_{i\in[M]} q_i\mathcal{L}_i(x) - \epsilon' q_i \log q_i$ is G-Lipschitz, since $\mathcal{L}_i$ is G-Lipschitz for all $i \in [M]$ and therefore for all $x \in \mathcal{X}$ we have $\|\nabla\mathcal{L}_q(x)\| = \Big\|\sum_{i\in[M]} q_i \nabla\mathcal{L}_i(x)\Big\| \le G$. Maximum operations preserve the Lipschitz continuity and therefore $\mathcal{L}_{\mathrm{smax},\epsilon}(x) = \max_{q\in\Delta^M}\mathcal{L}_q(x)$ is also G-Lipschitz. Thus, we can use Proposition 1 with $F = \mathcal{L}_{\mathrm{smax},\epsilon}$ and obtain that the output $\bar{x}$ of Algorithm 1 with probability at least $\frac{1}{2}$ will satisfy $\mathcal{L}_{\mathrm{smax},\epsilon}(\bar{x}) - \min_{x_\star\in\mathcal{X}}\mathcal{L}_{\mathrm{smax},\epsilon}(x_\star) \le \epsilon/2$. In addition, from Lemma 11 for every $x \in \mathcal{X}$ we have that $|\mathcal{L}_{\text{g-DRO}}(x) - \mathcal{L}_{\mathrm{smax},\epsilon}(x)| \le \epsilon/2$. Therefore, with probability at least $\frac{1}{2}$

$$\mathcal{L}_{\text{g-DRO}}(\bar{x}) - \min_{x_\star\in\mathcal{X}}\mathcal{L}_{\text{g-DRO}}(x_\star) \le \mathcal{L}_{\mathrm{smax},\epsilon}(\bar{x}) - \min_{x_\star\in\mathcal{X}}\mathcal{L}_{\mathrm{smax},\epsilon}(x_\star) + \epsilon/2 \le \epsilon.$$

Minimizing $\mathcal{L}_{\text{g-DRO}}$: complexity. The complexity of finding an $\epsilon/2$-suboptimal solution for $\mathcal{L}_{\mathrm{smax},\epsilon}$ (and therefore an ϵ-suboptimal solution for $\mathcal{L}_{\text{g-DRO}}$) is bounded by Proposition 1 as:

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right)m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon) + N)m_\epsilon^3\right]\right)$$

where $m_\epsilon = O\Big(\log\Big(\frac{GR^2}{\epsilon r_\epsilon}\Big)\Big) = O\big(\log\big(\frac{GR}{\epsilon}\log M\big)\big)$. To obtain the total complexity we evaluate the complexity of running r_ϵ-BROO with accuracy $\delta_j = \frac{r}{2^{j/2}m_\epsilon^2}$ (for the MLMC implementation), and accuracy $\delta_{\text{Bisection}} = \frac{r_\epsilon}{30}$ (for the bisection procedure). Using (32) and noting that $\lambda_{\mathrm{m}} = \frac{\epsilon}{r_\epsilon^{4/3}R^{2/3}}m_\epsilon^2$ we get the following BROO complexities:

1. $\mathcal{C}_{\lambda_{\mathrm{m}}}\Big(\frac{r_\epsilon}{m_\epsilon^2 2^{j/2}}\Big) = O\Big(\frac{G^2 2^j m_\epsilon^4}{\lambda_{\mathrm{m}}^2 r_\epsilon^2} + N\Big) = O\bigg(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{(\log M)^{2/3}}2^j + N\bigg)$

2. $\mathcal{C}_{\lambda_{\mathrm{m}}}\big(\frac{r_\epsilon}{30}\big) = O\Big(\frac{G^2}{\lambda_{\mathrm{m}}^2 r_\epsilon^2} + N\Big) = O\bigg(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{m_\epsilon^4(\log M)^{2/3}} + N\bigg)$.

Therefore

$$O\left(m_\epsilon\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right) = O\left(m_\epsilon\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\left(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{(\log M)^{2/3}}2^j + N\right)\right) \le O\left(m_\epsilon^2\left(\left(\frac{GR}{\epsilon}\right)^{4/3} + N\right)\right)$$

and

$$O\big(m_\epsilon^3(\mathcal{C}_{\lambda_k}(r_\epsilon/30) + N)\big) \le O\left(\left(\frac{GR}{\epsilon}\right)^{4/3} + Nm_\epsilon^3\right).$$

Substituting the bounds into Proposition 1 with $m_\epsilon = \log\left(\frac{GR}{\epsilon}M\right)$ and $r_\epsilon = \frac{\epsilon}{2G\log M}$, the total complexity is

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[Nm_\epsilon^3 + m_\epsilon^2\left(\frac{GR}{\epsilon}\right)^{4/3}\right]\right) \le O\left(\left(\frac{GR}{\epsilon}\right)^{2/3} N\log^{11/3}\left(\frac{GR}{\epsilon}M\right) + \left(\frac{GR}{\epsilon}\right)^2\log^2\left(\frac{GR}{\epsilon}M\right)\right).$$

□

C.4 SVRG-like estimator properties

We first give a definition of $\Gamma_{\epsilon,\lambda}$ that is more conducive to formulating variance reduction methods:

$$\Gamma_{\epsilon,\lambda}(x) := \sum_{i\in[M]} c_{x',\bar{x}} p_i(x')\gamma_i(x,x'),$$

where $\gamma_i(x,x') := \epsilon' e^{\frac{\mathcal{L}_i(x)-\mathcal{L}_i(x')+\frac{\lambda}{2}\|x-\bar{x}\|^2}{\epsilon'}}$, $c_{x',\bar{x}} := \left(\frac{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(x')}{\epsilon'}}}{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(\bar{x})}{\epsilon'}}}\right)$ and $p_i(x') := \frac{e^{\frac{\mathcal{L}_i(x')}{\epsilon'}}}{\sum_{j\in[M]} e^{\frac{\mathcal{L}_j(x')}{\epsilon'}}}$. Therefore, the MLMC estimator of $\gamma_i(x,x')$ is

$$\text{Draw } J \sim \text{Geom}\left(1-\tfrac{1}{\sqrt{8}}\right),\ S_1,\ldots,S_n \overset{\text{iid}}{\sim} w_i \text{ and let } \widehat{\mathcal{M}}[\gamma_i(x,x')] := \widehat{\gamma}(x,x';S_1) + \frac{\widehat{\mathcal{D}}_{2^J}}{p_J}$$

with $\widehat{\mathcal{D}}_n$ defined in (25) and $\widehat{\gamma}(x,x';S_1^n)$ defined in (24).

Lemma 14. *The SVRG-like estimator (9) is unbiased*

$$\mathbb{E}[\hat{g}_{x'}(x)] = \nabla\Gamma_{\epsilon,\lambda}(x)$$

Proof.

$$\begin{aligned}
\mathbb{E}[\hat{g}_{\bar{x}}(x)] &= \nabla\Gamma_{\epsilon,\lambda}(x') + \sum_{i\in[M]}\sum_{j\in[N]} p_i(x')w_{ij}\frac{1}{\epsilon'}\left[\mathbb{E}\left(\widehat{\mathcal{M}}[\gamma_i(x,x')]\right)\nabla\ell_j^\lambda(x) - \gamma_i(x',x')\nabla\ell_j^\lambda(x')\right]\\
&\overset{(i)}{=} \nabla\Gamma_{\epsilon,\lambda}(x') + \sum_{i\in[M]}\sum_{j\in[N]} p_i(x')w_{ij}\frac{1}{\epsilon'}\left[\gamma_i(x,x')\nabla\ell_j^\lambda(x) - \gamma_i(x',x')\nabla\ell_j^\lambda(x')\right]\\
&= \nabla\Gamma_{\epsilon,\lambda}(x') + \sum_{i\in[M]} p_i(x')[\nabla\gamma_i(x,x') - \nabla\gamma_i(x',x')]\\
&= \nabla\Gamma_{\epsilon,\lambda}(x)
\end{aligned}$$

with (i) following from the unbiased property of the MLMC estimator stated in Lemma 2 (that still holds for $\widehat{\mathcal{M}}[\gamma_i(x,x')]$). □

Lemma 3. *Let each ℓ_j satisfy Assumptions 1 and 2. For any $\lambda \le \frac{G}{r}$, $r = \frac{\epsilon'}{G}$ and $x,x' \in \mathbb{B}_r(\bar{x})$, the variance of $\hat{g}_{x'}(x)$ is bounded by* $\mathrm{Var}(\hat{g}_{x'}(x)) \le O\left(\left(L+\lambda+\frac{G^2}{\epsilon'}\right)^2\|x-x'\|^2\right)$.

Proof.

$$\begin{aligned}
\mathrm{Var}(\hat{g}_{x'}(x)) &= \mathbb{E}\|\hat{g}_{x'}(x) - \mathbb{E}\hat{g}_{x'}(x)\|^2\\
&= \mathbb{E}\left\|\nabla\Gamma_{\epsilon,\lambda}(x') + \frac{c_{x',\bar{x}}}{\epsilon'}\left[\widehat{\mathcal{M}}[\gamma_i(x,x')]\nabla\ell_j^\lambda(x) - \gamma_i(x',x')\nabla\ell_j^\lambda(x')\right] - \nabla\Gamma_{\epsilon,\lambda}(x)\right\|^2\\
&\overset{(i)}{\le} \frac{c_{x',\bar{x}}^2}{\epsilon'^2}\mathbb{E}\left\|\widehat{\mathcal{M}}[\gamma_i(x,x')]\nabla\ell_j^\lambda(x) - \gamma_i(x',x')\nabla\ell_j^\lambda(x')\right\|^2\\
&= \frac{c_{x',\bar{x}}^2}{\epsilon'^2}\mathbb{E}\left\|\left(\widehat{\mathcal{M}}[\gamma_i(x,x')] - \gamma_i(x',x')\right)\nabla\ell_j^\lambda(x) + \gamma_i(x',x')\left(\nabla\ell_j^\lambda(x) - \nabla\ell_j^\lambda(x')\right)\right\|^2\\
&\overset{(ii)}{\le} \frac{c_{x',\bar{x}}^2}{\epsilon'^2}\left[2\mathbb{E}\left\|\left(\widehat{\mathcal{M}}[\gamma_i(x,x')] - \gamma_i(x',x')\right)\nabla\ell_j^\lambda(x)\right\|^2 + 2\mathbb{E}\left\|\gamma_i(x',x')\left(\nabla\ell_j^\lambda(x) - \nabla\ell_j^\lambda(x')\right)\right\|^2\right]
\end{aligned} \tag{33}$$

where(i) follows from the inequality $\mathbb{E}[X-\mathbb{E}X]^2=\mathbb{E}[X^2]-[\mathbb{E}X]^2\le\mathbb{E}[X^2]$ and (ii) from the inequality $(a+b)^2\le 2a^2+2b^2$. Next we bound separately each of the expectation terms. Note that the ball constraint $x\in\mathbb{B}_r(\bar{x})$ with $r=\frac{\epsilon'}{G}$ and $\lambda\le\frac{G}{r}$ gives:

$$\gamma_i(x',x')=\epsilon' e^{\frac{\lambda}{2\epsilon'}\|x-\bar{x}\|^2}\le e^{\frac{Gr}{2\epsilon'}}=O(\epsilon')$$

therefore

$$\begin{aligned}\mathbb{E}\|\gamma_i(x',x')[\nabla\ell_j(x)-\nabla\ell_j(x')+\lambda(x-x')]\|^2 &\overset{(i)}{\le} O\big(\epsilon'^2\big)\Big(2\mathbb{E}\|\nabla\ell_j(x)-\nabla\ell_j(x')\|^2+2\|\lambda(x-x')\|^2\Big)\\ &\overset{(ii)}{\le} O\Big(\epsilon'^2\big(\lambda^2+L^2\big)\|x-x'\|^2\Big)\end{aligned}$$

with (i) following from the inequality $(a+b)^2\le 2a^2+2b^2$ and (ii) from Assumption 2. For the second expectation term we use the fact that each ℓ_j is G-Lipschitz, $\lambda\le\frac{G}{r}$ and $\|x-\bar{x}\|\le r$ and thus $\|\nabla\ell_j(x)+\lambda(x-\bar{x})\|\le\|\nabla\ell_j(x)\|+\|\lambda(x-\bar{x})\|\le 2G$. Therefore,

$$\mathbb{E}\Big\|\Big(\widehat{\mathcal{M}}[\gamma_i(x,x')]-\gamma_i(x',x')\Big)(\nabla\ell_j(x)+\lambda(x-\bar{x}))\Big\|^2\le O\bigg(G^2\bigg(\mathbb{E}\Big|\widehat{\mathcal{M}}[\gamma_i(x,x')]-\gamma_i(x',x')\Big|^2\bigg)\bigg).$$

From the definition of $\widehat{\mathcal{M}}[\gamma_i(x,x')]$ we get:

$$\begin{aligned}\mathbb{E}\Big|\widehat{\mathcal{M}}[\gamma_i(x,x')]-\gamma_i(x',x')\Big|^2 &\le 2\mathbb{E}|\widehat{\gamma}(x,x';S_1)-\gamma_i(x',x')|^2+2\sum_{j=1}^{\infty}\left(1-\frac{1}{\sqrt{8}}\right)2^{1.5j}\mathbb{E}\Big|\widehat{\mathcal{D}}_{2^j}\Big|^2\\ &\overset{(i)}{\le} 2\mathbb{E}|\widehat{\gamma}(x,x';S_1)-\gamma_i(x',x')|^2+O\left(\frac{G^4\|x-x'\|^4}{\epsilon'^2}\right)\\ &= O\left(\epsilon'^2 e^{\frac{\lambda\|x-\bar{x}\|^2}{\epsilon'}}\mathbb{E}\left(e^{\frac{\ell_{S_1}(x)-\ell_{S_1}(x')}{\epsilon'}}-1\right)^2+\left(\frac{G^4\|x-x'\|^4}{\epsilon'^2}\right)\right)\\ &\overset{(ii)}{\le} O\left(\epsilon'^2 e^{\frac{\lambda\|x-\bar{x}\|^2}{\epsilon'}}\mathbb{E}\left(\frac{\ell_{S_1}(x)-\ell_{S_1}(x')}{\epsilon'}\right)^4+\left(\frac{G^4\|x-x'\|^4}{\epsilon'^2}\right)\right)\\ &\overset{(iii)}{\le} O\left(\frac{G^4\|x-x'\|^4}{\epsilon'^2}\right)\\ &\overset{(iv)}{\le} O\Big(G^2\|x-x'\|^2\Big)\end{aligned}$$

with (i) following from Lemma 12, (ii) follows from the inequality $e^x-1\le x+2x^2=O(x^2)$ for $x\le 3$ with $x=\frac{\ell_{S_1}(x)-\ell_{S_1}(x')}{\epsilon'}\le 2$, (iii) follows since each ℓ_j is G-Lipschitz and since $e^{\frac{\lambda\|x-\bar{x}\|^2}{\epsilon'}}=O(1)$ and (iv) since $\frac{G^2\|x-x'\|^2}{\epsilon'^2}\le\frac{G^2 4r^2}{\epsilon'^2}=4$. Therefore,

$$\mathbb{E}\Big\|\Big(\widehat{\mathcal{M}}[\gamma_i(x,x')]-\gamma_i(x',x')\Big)(\nabla\ell_j(x)+\lambda(x-\bar{x}))\Big\|^2\le O\Big(G^4\|x-x'\|^2\Big).$$

Finally we bound $c_{x',\bar{x}}$ using the ball constraint $x'\in\mathbb{B}_r(\bar{x})$ and the fact that each $\mathcal{L}_i$ is G-Lipschitz, therefore

$$c_{x',\bar{x}}=\frac{\sum_{j\in[M]}e^{\mathcal{L}_j(x')/\epsilon'}}{\sum_{j\in[M]}e^{\mathcal{L}_j(\bar{x})/\epsilon'}}=\frac{\sum_{j\in[M]}e^{\frac{\mathcal{L}_j(x')-\mathcal{L}_j(\bar{x})}{\epsilon'}}e^{\frac{\mathcal{L}_j(\bar{x})}{\epsilon'}}}{\sum_{j\in[M]}e^{\frac{\mathcal{L}_j(\bar{x})}{\epsilon'}}}\le e.$$

Substituting back the bounds on each expectaion term and the bound on $c_{x',\bar{x}}$ into (33) we get

$$\mathrm{Var}(\hat{g}_{\bar{x}}(x))\le O\left(L^2+\lambda^2+\frac{G^4}{\epsilon'^2}\right)\|x-x'\|^2\le O\left(\left(L+\lambda+\frac{G^2}{\epsilon'}\right)^2\|x-x'\|^2\right).$$

□

C.5 Complexity of the reduced-variance BROO implementation

We first state the complexity bounds of KatyushaX^s [2]

Lemma 15 ([2, Theorems 1 and 4.3])**.** *Let F be a λ-strongly convex function with minimizer $x_\star$ and let $\hat{g}_{x'}(x)$ be a stochastic gradient estimator satisfying the properties*

1. $\mathbb{E}[\hat{g}_{x'}(x)] = \mathbb{E}\nabla F(x)$
2. $\mathbb{E}[\hat{g}_{x'}(x) - \nabla F(x)]^2 \le \widetilde{L}^2\|x - \bar{x}\|^2$
3. *$\hat{g}_{x'}(\cdot)$ has evaluation complexity $O(1)$ and preprocessing complexity $O(N)$,*

then KatyushaX^s with the stochastic gradient estimator $\hat{g}_{x'}$ finds a point x satisfying $\mathbb{E}[F(x) - F(x_\star)] \le \epsilon$ with complexity

$$O\left(\left(N + \frac{N^{3/4}\sqrt{\widetilde{L}}}{\sqrt{\lambda}}\right)\log\left(\frac{F(x_0) - F(x_\star)}{\epsilon}\right)\right).$$

Applying Lemma 15 with $\widetilde{L} = O\left(L + \frac{G^2}{\epsilon}\right)$ and accuracy $\frac{\lambda\delta^2}{2}$ gives the following result.

Theorem 2. *Let each ℓ_j satisfy Assumptions 1 and 2. Let $\epsilon > 0$, $\epsilon' = \epsilon/(2\log M)$ and $r_\epsilon = \epsilon'/G$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(G/r_\epsilon)$ and accuracy δ, KatyushaX^s [2, Algorithm 2] with the gradient estimator (9) outputs a valid r_ϵ-BROO response and has complexity $\mathcal{C}_\lambda(\delta) = O\Big(\Big(N + \frac{N^{3/4}(G+\sqrt{\epsilon' L})}{\sqrt{\lambda\epsilon'}}\Big)\log\big(\frac{Gr_\epsilon}{\lambda\delta^2}\big)\Big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_{\text{g-DRO}}$ (2) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3} H + N^{3/4}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{7/2} H\right) \text{ where } H \coloneqq M\frac{GR}{\epsilon}.$$

Proof. The proof is structured similarly to the proof of Theorem 1.

BROO implementation: correctness. From Lemma 1 we have that $\Gamma_{\epsilon,\lambda}$ is $\Omega(\lambda)$-strongly convex, in addition, Lemma 3 and Lemma 14 show that the stochastic gradient estimator defined in (9) is unbiased with $\mathrm{Var}(\hat{g}_{x'}(x)) \le \widetilde{L}^2\|x - x'\|^2$. Thus, we can directly apply Lemma 15 with $F = \Gamma_{\epsilon,\lambda}$ and accuracy $\frac{\lambda\delta^2}{2}$ and obtain a valid BROO response.

BROO implementation: complexity. We use KatyushaX^s [2] for the BROO implementation. Applying Lemma 15 with $F = \Gamma_{\epsilon,\lambda}$, $x_0 = \bar{x}$, $\widetilde{L} = O\Big(L + \lambda + \frac{G^2}{\epsilon'}\Big) \le O\Big(L + \frac{G^2}{\epsilon'}\Big)$ and accuracy $\frac{\lambda\delta^2}{2}$ the complexity of our implementation is

$$O\left(\left(N + N^{3/4}\frac{\sqrt{L\epsilon'} + G}{\sqrt{\epsilon'}}\right)\log\left(\frac{\Gamma_{\epsilon,\lambda}(\bar{x}) - \min_{x_\star \in \mathbb{B}_{r_\epsilon}(\bar{x})}\Gamma_{\epsilon,\lambda}(x_\star)}{\lambda\delta^2}\right)\right) \tag{34}$$

and note that $\Gamma_{\epsilon,\lambda}(\bar{x}) - \min_{x_\star \in \mathbb{B}_{r_\epsilon}(\bar{x})}\Gamma_{\epsilon,\lambda}(x_\star) \le Gr_\epsilon$, since (from Lemma 1) $\Gamma_{\epsilon,\lambda}$ is $O(G)$-Lipschitz.

Minimizing $\mathcal{L}_{\text{g-DRO}}$: correctness. Similarly to the proof of Theorem 1, combining the guarantees of Proposition 1 and Lemma 11, with probability at least $\frac{1}{2}$ the output $\bar{x}$ of Algorithm 1 satisfies $\mathcal{L}_{\text{g-DRO}}(\bar{x}) - \min_{x_\star \in \mathcal{X}}\mathcal{L}_{\text{g-DRO}}(x_\star) \le \epsilon$.

Minimizing $\mathcal{L}_{\text{g-DRO}}$: complexity. The complexity of finding $\epsilon/2$-suboptimal solution for $\mathcal{L}_{\text{smax},\epsilon}$ and therefore an ϵ-suboptimal solution for $\mathcal{L}_{\text{g-DRO}}$, is bounded by Proposition 1 as:

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right)m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon) + N)m_\epsilon^3\right]\right) \tag{35}$$

where $m_\epsilon = O\big(\log\frac{GR^2}{\epsilon r_\epsilon}\big) = \log\big(\frac{GR}{\epsilon}\log M\big)$ and $\lambda_{\mathrm{m}} = O\big(\frac{m_\epsilon^2\epsilon}{r^{4/3}R^{2/3}}\big)$. We first show the complexity of the BROO implementation with $\delta_j = \frac{r_\epsilon}{2^{j/2}m_\epsilon^2}$ (for the MLMC implementation) and with $\delta = \big(\frac{r_\epsilon}{30}\big)$ for the bisection procedure. Using (34) we get:

1. $\mathcal{C}_{\lambda_{\rm m}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right) = O\Big(\Big(N + N^{3/4}\Big(\frac{G\sqrt{\log M}}{\sqrt{\epsilon}} + \sqrt{L}\Big)\frac{1}{\sqrt{\lambda_{\rm m}}}\Big)\log\Big(\frac{\epsilon' 2^j m_\epsilon^4}{\lambda_{\rm m} r_\epsilon^2}\Big)\Big)$
2. $\mathcal{C}_{\lambda_{\rm m}}\left(\frac{r_\epsilon}{30}\right) = O\Big(\Big(N + N^{3/4}\Big(\frac{G\sqrt{\log M}}{\sqrt{\epsilon}} + \sqrt{L}\Big)\frac{1}{\sqrt{\lambda_k}}\Big)\log\Big(\frac{\epsilon'}{\lambda_{\rm m} r_\epsilon^2}\Big)\Big)$

From the definitions of $\lambda_{\rm m}$ and r_ϵ we have $\frac{\epsilon'}{\lambda_{\rm m} r_\epsilon^2} = O\Big(\big(\frac{GR}{\epsilon}\big)^{2/3}\frac{1}{m_\epsilon^2}\Big)$ and $\frac{1}{\sqrt{\lambda_{\rm m}}} = O\Big(\frac{R^{1/3} r_\epsilon^{2/3}}{m_\epsilon\sqrt{\epsilon}}\Big)$, therefore,

$$m_\epsilon \sum_{j=0}^{m_\epsilon} \frac{1}{2^j}\mathcal{C}_{\lambda_{\rm m}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right) = O\left(m_\epsilon \sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\left(N + N^{3/4}\left(\frac{G\sqrt{\log M}+\sqrt{L\epsilon}}{\sqrt{\epsilon}}\right)\sqrt{\frac{R^{2/3}r_\epsilon^{4/3}}{\epsilon m_\epsilon^2}}\right)\log\left(\frac{GRm_\epsilon^2 2^j}{\epsilon}\right)\right)$$
$$\le O\left(m_\epsilon^2\left[N + N^{3/4}\left(\frac{GR^{1/3}}{\epsilon} + \sqrt{\frac{LR^{2/3}}{\epsilon}}\right)r_\epsilon^{2/3}\right]\right).$$

Similarly, we have that

$$O\big((\mathcal{C}_{\lambda_{\rm m}}(r_\epsilon) + N)m_\epsilon^3\big) = O\left(m_\epsilon^3\left(N + N^{3/4}\left(\frac{G\sqrt{\log M}+\sqrt{L}}{\sqrt{\epsilon}}\sqrt{\frac{R^{2/3}r_\epsilon^{4/3}}{\epsilon m_\epsilon^2}}\right)\right)\log\left(\left(\frac{GR}{\epsilon}\right)\frac{1}{m_\epsilon^2}\right)\right)$$
$$\le O\left(m_\epsilon^4 N + m_\epsilon^{3.5}N^{3/4}\left(\frac{GR^{1/3}}{\epsilon} + \sqrt{\frac{LR^{2/3}}{\epsilon}}\right)r_\epsilon^{2/3}\right).$$

Substituting the bounds into Proposition 1 with $m_\epsilon = \log\big(\frac{GR}{\epsilon}\log M\big)$ and $r_\epsilon = \frac{\epsilon}{2\log M}$ the total complexity is

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3}\left(\frac{GR}{\epsilon}\log M\right) + N^{3/4}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{7/2}\left(\frac{GR}{\epsilon}\log M\right)\right).$$

□

C.6 Helper lemmas

Lemma 16. *Let $Y_1, \ldots, Y_n$ be a sequence of random i.i.d variables such that for every $i \in [n]$ and a constant $c > 0$ we have that $\mathbb{E}[Y_i] = 0$ and $|Y_i| \le c$ with probability 1. Then*

$$\mathbb{E}\left(\sum_{i=1}^n Y_i\right)^4 \le O\big(n^2 c^4\big).$$

Proof. For $i \neq j$ we have that $\mathbb{E}[Y_iY_j] = \mathbb{E}[Y_i]\mathbb{E}[Y_j] = 0$, therefore

$$\mathbb{E}\left(\sum_{i=1}^n Y_i\right)^4 = \sum_{i=1}^n \mathbb{E}\big[Y_i^4\big] + 3\sum_{i=1}^n\sum_{j\neq i}\mathbb{E}\big[Y_i^2\big]\mathbb{E}\big[Y_j^2\big]$$
$$\le nc^4 + 3n\binom{n-1}{2}c^4 = O\big(n^2c^4\big)$$

□

D DRO with f-divergence

In this section we provide the proofs for the results in Section 4. In Appendix D.1 we provide the derivation of the dual formulation in (10). In Appendix D.2 we show how to reduce the constrained problem (3) to a regularized problem of the form (10), then in Appendix D.3 we describe the properties of $\mathcal{L}_{\psi,\epsilon}$, the approximation of (10). In Appendix D.4 we provide the proofs for our main technical contribution and give guarantees on the stability of the gradient estimators. Last, in Appendices D.5 and D.6 we give the complexity guarantees of our implementation in the non-smooth and slightly smooth cases.

D.1 Dual formulation of DRO with f-divergence

Here we give the derivation of the objective in (10), also considered in prior work [e.g., 42, 39, 32]. Recall the DRO with f-divergence objective:

$$\mathcal{L}_{f\text{-div}}(x) \coloneqq \max_{q\in\Delta^N : \sum_{i\in[N]} \frac{f(Nq_i)}{N} \le 1} \sum_{i\in[N]} q_i \ell_i(x).$$

We first show the relation between $\mathcal{L}_{f\text{-div}}$ and its regularized form (10). Using the Lagrange multiplier ν for the constraint $\sum_{i\in[N]} \frac{f(Nq_i)}{N} \le 1$ and strong duality we get

$$\mathcal{L}_{f\text{-div}}(x) = \min_{\nu\ge 0}\left\{\nu + \max_{q\in\Delta^N} \sum_{i\in[N]} \Big(q_i\ell_i(x) - \frac{\nu}{N} f(Nq_i)\Big)\right\} = \min_{\nu\ge 0}\left\{\nu + \mathcal{L}_{\nu\cdot f}(x)\right\},$$

where $\mathcal{L}_{\nu\cdot f}(x)$ is the regularized form of $\mathcal{L}_{f\text{-div}}$: writing $\psi(x) = \frac{\nu}{N} f(Nx)$, with slight abuse of notation we have

$$\mathcal{L}_{\nu\cdot f}(x) = \mathcal{L}_{\psi}(x) = \max_{q\in\Delta^N}\left\{\sum_{i\in[N]} (q_i\ell_i(x) - \psi(q_i))\right\}.$$

Adding a Lagrange multiplier y for the constraint that $q \in \Delta^N$ and using strong duality again gives

$$\begin{aligned}\mathcal{L}_{\psi}(x) &= \max_{q\in\mathbb{R}_+^N} \min_{y\in\mathbb{R}}\left\{\sum_{i\in[N]} (q_i\ell_i(x) - \psi(q_i) - Gy\cdot q_i) + Gy\right\} \\ &= \min_{y\in\mathbb{R}}\left\{\sum_{i\in[N]} \max_{q_i\in\mathbb{R}_+} (q_i\ell_i(x) - \psi(q_i) - Gy\cdot q_i) + Gy\right\}.\end{aligned}$$

Finally, using $\psi^*(v) \coloneqq \max_{t\in\mathrm{dom}(\psi)}\{vt - \psi t\}$ (the Fenchel conjugate of ψ), we have

$$\mathcal{L}_{\psi}(x) = \min_{y\in\mathbb{R}}\left\{\sum_{i\in[N]} \psi^*(\ell_i(x) - Gy) + Gy\right\},$$

and note that $\psi^*(\ell_i(x) - Gy)$ is equivalent to $\nu f^*\Big(\frac{\ell_i(x)-Gy}{\nu}\Big)$.

D.2 Minimizing the constrained objective using the regularized objective

In this section, we show that under the following Assumptions 4 and 5 we can reduce the constrained problem of minimizing (3) to the regularized problem of minimizing (10) by computing a polylogarithmic number of $O(\epsilon)$-accurate minimizers of (10).

Assumption 4. *Each loss function ℓ_i is bounded, i.e., $\ell_i : \mathcal{X} \to [0, B_\ell]$ for every $i \in [N]$.*

Assumption 5. *For any uncertainty set of the form $\mathcal{U} = \{q \in \Delta^N : D_f(q,p) \le 1\}$, the divergence function f is bounded, i.e., $f : \mathbb{R}_+ \to [0, B_f]$ for some $B_f \ge 1$.*

We note that the above assumptions are weak since the complexity of our approach only depends logarithmically on on $\frac{B_f B_\ell}{\epsilon}$.

We first cite a result on noisy one dimensional bisection, as given by a guarantees on the OneDimMinimizer algorithm in Cohen et al. [18].

Lemma 17 (Lemma 33, Cohen et al. [18]). *let $f : \mathbb{R} \to \mathbb{R}$ be a B-Lipschitz convex function defined on the interval $[\ell, u]$, and let $\mathcal{G} : \mathbb{R} \to \mathbb{R}$ be an oracle such that $|\mathcal{G}(y) - f(y)| \le \widetilde{\epsilon}$ for all y. With $O\Big(\log\Big(\frac{B(u-\ell)}{\widetilde{\epsilon}}\Big)\Big)$ calls to $\mathcal{G}$, the algorithm* OneDimMinimizer *[18, Algorithm 8] outputs y' such that*

$$f(y') - \min_y f(y) \le 4\widetilde{\epsilon}$$

We specialize Lemma 17 to our settings and provide the complexity guarantees of minimizing (3) to an ϵ-accurate solution using a noisy oracle $\mathcal{G}$.

Proposition 3. *Let each ℓ_i satisfy Assumption 4 and let f satisfy Assumption 5. Define the function $h(\nu) := \min_{x\in\mathcal{X}} \mathcal{L}_{\nu\cdot f}(x)+\nu$ with $\mathcal{L}_{\nu\cdot f}$ defined in (10) and let $\mathcal{G}$ be an oracle such that $\mathcal{G}(\nu) \geq h(\nu)$ with probability 1 and $\mathcal{G}(\nu)-h(\nu) \leq \frac{\epsilon}{5}$ with probability at least $\frac{1}{2}$. Then applying* OneDimMinimizer *[18, Algorithm 8] on the interval $[0, B_\ell]$ outputs ν' that with probability at leat $\frac{99}{100}$ satisfies*

$$\mathcal{G}(\nu') - \min_{\nu\geq 0} h(\nu) = \mathcal{G}(\nu') - \min_{x\in\mathcal{X}} \mathcal{L}_{\mathrm{f-div}}(x) \leq \epsilon$$

using $O(\log(H)\log(\log H))$ calls to $\mathcal{G}$, where $H = \frac{B_f B_\ell}{\epsilon}$.

Proof. Let $\hat{h}_{q,x}(\nu) := \sum_{i\in[N]} q_i\ell_i(x) - \nu\Big(\frac{1}{N}\sum_{i\in[N]} f(Nq_i) - 1\Big)$ and note that for any q and x the function $\hat{h}_{q,x}$ is B_f-Lipschitz, since it is linear in ν and $\Big|\frac{1}{N}\sum_{i\in[N]} f(Nq_i) - 1\Big| \leq B_f$. Minimization and maximization operations preserve the Lipschitz continuity and therefore the function $h(\nu) = \min_x \mathcal{L}_{\nu\cdot f}(x) = \min_x \max_q \hat{h}_{q,x}(\nu)$ is also B_f-Lipschitz continuous. In addition for the $q^\star \in \Delta^N$ that maximizes $\mathcal{L}_{\nu\cdot f}(x)$ we have that

$$\sum_{i\in[N]}\left[q_i^\star\ell_i(x) - \nu\frac{1}{N}f(Nq_i^\star)\right] \geq \sum_{i\in[N]}\frac{1}{N}\ell_i(x)$$

and rearranging gives

$$\frac{1}{N}\sum_{i\in[N]} f(Nq_i^\star) \leq \frac{\sum_{i\in[N]}[q_i^\star\ell_i(x) - \frac{1}{N}\ell_i(x)]}{\nu} \leq \frac{B_\ell}{\nu}.$$

Therefore, for all $\nu > B_\ell$ we have $h'(\nu) = 1 - \frac{1}{N}\sum_{i\in[N]} f(Nq_i^\star) > 0$ and therefore it suffices to restrict $h(\nu)$ to $[0, B_\ell]$. Next, to turn $\mathcal{G}$ into a high-probability oracle, we call it $\log_2\Big(100\log\Big(\frac{B_f B_\ell}{\epsilon}\Big)\Big)$ times and choose the smallest output. Therefore, with probability at least $1 - \left(\frac{1}{2}\right)^{\log\left(100\log\left(\frac{B_f B_\ell}{\epsilon}\right)\right)} = 1 - 1/\Big(100\log\Big(\frac{B_f B_\ell}{\epsilon}\Big)\Big)$ the result is within $\frac{\epsilon}{5}$ of $h(\nu)$. Since h is B_f-Lipschitz and defined on $[0, B_\ell]$ we can use Lemma 17 with $\ell = 0$, $u = B_f$, $\widetilde{\epsilon} = \epsilon/5$, $B = B_f$ and the high-probability version of $\mathcal{G}$. Therefore, using $O\Big(\log\Big(\frac{B_f B_\ell}{\epsilon}\Big)\Big)$ calls to the high probability version of $\mathcal{G}$ and applying the union bound, we obtain that with probability at least $\frac{99}{100}$ OneDimMinimizer outputs ν' that satisfies

$$h(\nu') - \min_\nu h(v) \leq 4\epsilon/5$$

and therefore

$$\mathcal{G}(\nu') - \min_\nu h(v) = \mathcal{G}(\nu') - h(\nu') + h(\nu') - \min_\nu h(v) \leq \epsilon.$$

□

Finally, in the following corollary we show that finding an ϵ-suboptimal solution for (3) requires a polylogarithmic number of $O(\epsilon)$-accurate minimizers of (10) and a polylogarithmic number of evaluations of (10).

Corollary 5. *Let each ℓ_i satisfy Assumption 4 and let f satisfy Assumption 5, then minimizing (3) to accuracy ϵ with probability at least $\frac{99}{100}$ requires $O(\log(H)\log(\log H))$ evaluations of (10) and $O(\log(H)\log(\log H))$ calls to an algorithm that with probability at least $\frac{1}{2}$ returns an $O(\epsilon)$-suboptimal point of (10), where $H = \frac{B_f B_\ell}{\epsilon}$.*

Proof. Note that $\mathcal{L}_{\nu\cdot f}$ is defined on (10). Let $\widetilde{\mathcal{G}}(\nu) := \mathcal{L}_{\nu\cdot f}(\widetilde{x}) + \nu$ where $\widetilde{x}$ is the output of an algorithm that with probability at least $\frac{1}{2}$ returns an $\frac{\epsilon}{5}$-suboptimal point of $\mathcal{L}_{\nu\cdot f}$ and let $h(\nu) := \min_{x\in\mathcal{X}} \mathcal{L}_{\nu\cdot f}(x) + \nu$. We have that $\widetilde{\mathcal{G}}(\nu) - h(\nu) \leq \frac{\epsilon}{5}$ with probability at least $\frac{1}{2}$, therefore, we can apply Proposition 3 with $\mathcal{G} = \widetilde{\mathcal{G}}$, and obtain that with $O\Big(\log\Big(\frac{B_f\cdot B_\ell}{\epsilon}\Big)\log\Big(\log\frac{B_f\cdot B_\ell}{\epsilon}\Big)\Big)$ calls to $\widetilde{\mathcal{G}}$ (i.e., to an algorithm that outputs $\frac{\epsilon}{5}$-suboptimal minimizer of $\mathcal{L}_{\nu \cdot f}$ with probability at least $\frac{1}{2}$), OneDimMinimizer outputs ν' that satisfies with probability at least $\frac{99}{100}$

$$\widetilde{\mathcal{G}}(\nu') - \min_{\nu} h(\nu) = \widetilde{\mathcal{G}}(\nu') - \min_{x} \mathcal{L}_{f\text{-div}}(x) \le \epsilon.$$

Noting that $\mathcal{L}_{f\text{-div}}(x) = \min_{\nu \ge 0}\{\mathcal{L}_{\nu \cdot f}(x) + \nu\}$ we obtain

$$\mathcal{L}_{f\text{-div}}(\widetilde{x}) - \min_{x} \mathcal{L}_{f\text{-div}}(x) \le \mathcal{L}_{\nu' \cdot f}(\widetilde{x}) + \nu' - \min_{x} \mathcal{L}_{f\text{-div}}(x) = \widetilde{\mathcal{G}}(\nu') - \min_{x} \mathcal{L}_{f\text{-div}}(x) \le \epsilon.$$

□

Corollary 5 means that the complexity bounds for approximately minimizing the objective $\mathcal{L}_\psi$ established by Theorems 3 and 4 also apply (with slightly larger logarithmic factors) to approximately minimizing the constrained f-divergence objective $\mathcal{L}_{f\text{-div}}$.

D.3 Properties of $\mathcal{L}_\psi$ and $\mathcal{L}_{\psi,\epsilon}$

Lemma 18. *For $\mathcal{L}_\psi$ defined in (10) and $\mathcal{L}_{\psi,\epsilon}$ defined in (12) we have that*

$$|\mathcal{L}_{\psi,\epsilon}(x) - \mathcal{L}_\psi(x)| \le \frac{\epsilon}{2} \text{ for all } x \in \mathbb{R}^d.$$

Proof. Recall that $\epsilon' = \frac{\epsilon}{2\log N}$ and for $q \in \Delta^N$ we have that $\sum_{i \in N} q_i \log q_i \in [-\log N, 0]$, therefore:

$$\begin{aligned}|\mathcal{L}_{\psi,\epsilon}(x) - \mathcal{L}_\psi(x)| &= \left| \max_{q \in \Delta^N} \left\{ \sum_{i \in [N]} (q_i \ell_i(x) - \psi(q_i) - \epsilon' q_i \log q_i) \right\} - \max_{q \in \Delta^N} \left\{ \sum_{i \in [N]} (q_i \ell_i(x) - \psi(q_i)) \right\} \right| \\ &\le \left| \epsilon' \sum_{i \in [N]} q_i \log q_i \right| \le \epsilon' \log N = \epsilon/2.\end{aligned}$$

□

D.4 Gradient estimator stability proofs

Lemma 4. *For any convex $\psi : \mathbb{R}_+ \to \mathbb{R}$ and ψ_ϵ defined in (10), $\log\big(\psi_\epsilon^{*\prime}(\cdot)\big)$ is $\frac{1}{\epsilon'}$-Lipschitz.*

Note that from the definition of $\psi_\epsilon^{*\prime}$ we have that ${\psi_\epsilon}'(0) \to -\infty$, in addition since $q \in \mathbb{R}_+^N$ then $\psi_\epsilon^{*\prime}$ is non-negative and $\log\big(\psi_\epsilon^{*\prime}\big)$ is well-defined. We now give the proof of Lemma 4.

Proof. While we write the proof as though the function ψ is differentiable with derivative ψ', one may readily interpret ψ' as an element in the subdifferential of ψ and the proof continues to hold.

Let $\phi_\epsilon = \epsilon' q \log q$ and recall that $\psi_\epsilon(q) = \psi(q) + \phi_\epsilon(q)$. Fix any two numbers $v_1, v_2 \in \mathbb{R}$ and assume without loss of generality that $v_2 > v_1$. For $i = 1, 2$, let

$$q_i \coloneqq \psi_\epsilon^{*\prime}(v_i) \text{ and } p_i \coloneqq \phi_\epsilon^{*\prime}(v_i) = e^{v_i/\epsilon' - 1}. \tag{36}$$

Note that by definition of the Fenchel dual (and strict convexity of ψ_ϵ), q_i is the unique solution to

$$v_i = \psi_\epsilon'(q_i) = \psi'(q_i) + \phi_\epsilon'(q_i)$$

and moreover that $q_2 \ge q_1$ since ψ_ϵ^* is convex and therefore $\psi_\epsilon^{*\prime}$ is non-deceasing. Similarly, p_i is the unique solution to

$$v_i = \phi_\epsilon'(p_i)$$

and $p_2 > p_1$. Combining the two equalities yields

$$v_2 - v_1 = \psi'(q_2) + \phi_\epsilon'(q_2) - \psi'(q_1) - \phi_\epsilon'(q_1) = \phi_\epsilon'(p_2) - \phi_\epsilon'(p_1).$$

Rearranging, we find that

$$0 \le \phi_\epsilon'(q_2) - \phi_\epsilon'(q_1) = \phi_\epsilon'(p_2) - \phi_\epsilon'(p_1) - [\psi'(q_2) - \psi'(q_1)] \le \phi_\epsilon'(p_2) - \phi_\epsilon'(p_1).$$

where $\psi'(q_2) - \psi'(q_1) \geq 0$ holds by convexity of ψ and $q_2 \geq q_1$. Recalling that $\phi'_\epsilon(q) = \epsilon' + \epsilon' \log q$, we have

$$0 \leq \log q_2 - \log q_1 \leq \log p_2 - \log p_1 = \frac{1}{\epsilon'}(v_2 - v_1),$$

where the last equality follows by substituting the definition of p_i. The proof is complete upon recalling that $q_i = {\psi^*_\epsilon}'(v_i)$. ☐

Lemma 5. *For $G > 0$, $\ell(x) = (\ell_1(x), \dots, \ell_N(x))$ and $y^\star(x) = \operatorname{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(x, y)$, we have $|y^\star(x) - y^\star(x')| \leq \frac{1}{G}\|\ell(x) - \ell(x')\|_\infty$ for all $x, x' \in \mathcal{X}$. Moreover, if each ℓ_i is G-Lipschitz, we have $|y^\star(x) - y^\star(x')| \leq \|x - x'\|$.*

Proof. For $x, x' \in \mathcal{X}$ w.l.o.g. assume that $y^\star(x) \leq y^\star(x')$ and observe that for every $u \in \mathcal{X}$

$$\sum_{i\in[N]} {\psi^*_\epsilon}'(\ell_i(u) - Gy^\star(u)) = 1. \tag{37}$$

Let $\widetilde{\ell}_i(x) = \ell_i(x) + \delta$ with $\delta := \|\ell(x') - \ell(x)\|_\infty$ and $\widetilde{y}(x) := \operatorname{argmin}_{y\in\mathbb{R}}\left\{\sum_{i\in[N]} \psi^*_\epsilon\left(\widetilde{\ell}_i(x) - Gy\right) + Gy\right\}$. Then, according to (37)

$$\sum_{i\in[N]} {\psi^*_\epsilon}'\left(\widetilde{\ell}_i(x) - G\widetilde{y}(x)\right) = \sum_{i\in[N]} {\psi^*_\epsilon}'(\ell_i(x) + \delta - G\widetilde{y}(x)) \overset{(i)}{=} \sum_{i\in[N]} {\psi^*_\epsilon}'(\ell_i(x) - Gy^\star(x)) = 1$$

and due to the monotonicity of ${\psi^*_\epsilon}'$ and (i) we get

$$G\widetilde{y}(x) = Gy^\star(x) + \delta.$$

By convexity, ${\psi^*_\epsilon}'$ is monotonically non decreasing, thus

$$\sum_{i\in[N]} {\psi^*_\epsilon}'(\ell_i(x') - G\widetilde{y}(x)) \overset{(i)}{\leq} \sum_{i\in[N]} {\psi^*_\epsilon}'\left(\widetilde{\ell}_i(x) - G\widetilde{y}(x)\right) = \sum_{i\in[N]} {\psi^*_\epsilon}'(\ell_i(x') - Gy^\star(x')) = 1 \tag{38}$$

where (i) follows from noting that $\ell_i(x') \leq \ell_i(x) + \max_{i\in[N]}|\ell_i(x') - \ell_i(x)| = \widetilde{\ell}_i(x)$. Therefore, $Gy^\star(x') \leq G\widetilde{y}(x) = Gy^\star(x) + \delta$ giving

$$G|y^\star(x') - y^\star(x)| \leq \|\ell(x') - \ell(x)\|_\infty.$$

In addition, if each ℓ_i is G-Lipschitz we have

$$G|y^\star(x') - y^\star(x)| \leq \|\ell(x') - \ell(x)\|_\infty \leq G\|x' - x\|.$$

☐

D.5 Epoch-SGD BROO implementation

In this section we provide the analysis of our algorithm in the non-smooth case, which consists of combining our general BROO acceleration scheme (Algorithm 1) with a variant of Epoch-SGD [28] that we specialize in order to implement a BROO for $\Upsilon_{\epsilon,\lambda}$ (Algorithm 3). We organize this section as follows. First, we prove Lemma 6 showing that our gradient estimators are unbiased with bounded second moment, and therefore can be used in Algorithm 3. Then, in Proposition 4 we give the convergence rate of Algorithm 3. Combining the previous statements with the guarantees of Proposition 1 we prove Theorem 3.

For convenience, we restate the definitions of Υ_ϵ and our stochastic estimators for $\nabla_x \Upsilon_\epsilon(x, y)$ and $\nabla_y \Upsilon_\epsilon(x, y)$:

$$\Upsilon_\epsilon(x, y) := \sum_{i\in[N]} \psi^*_\epsilon(\ell_i(x) - Gy) + Gy,$$

and

$$\hat{g}^{\mathrm{x}}(x, y) = \frac{{\psi^*_\epsilon}'(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x, y) \ \ , \ \hat{g}^{\mathrm{y}}(x, y) = G\left(1 - \frac{{\psi^*_\epsilon}'(\ell_i(x) - Gy)}{\bar{p}_i}\right)$$

where $\bar{p}_i = {\psi^*_\epsilon}'(\ell_i(\bar{x}) - G\bar{y})$.

Lemma 6. *Let each ℓ_i be G-Lipschitz, let $\bar{x} \in \mathcal{X}$ and $\bar{y} = \mathrm{argmin}_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(\bar{x}, y)$. Let $r_\epsilon = \frac{\epsilon'}{G}$, then for all $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ and $y \in [\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$, the gradient estimators $\hat{g}^{\mathrm{x}}$ and $\hat{g}^{\mathrm{y}}$ satisfy the following:*

1. $\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{x}}(x, y)] = \nabla_x \Upsilon_\epsilon(x, y)$ *and* $\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{y}}(x, y)] = \nabla_y \Upsilon_\epsilon(x, y)$.
2. $\mathbb{E}_{i\sim\bar{p}_i}\|\hat{g}^{\mathrm{x}}(x, y)\|^2 \le e^4G^2$ *and* $\mathbb{E}_{i\sim\bar{p}_i}|\hat{g}^{\mathrm{y}}(x, y)|^2 \le e^4G^2$.

Proof. We first show that the stochastic gradients $\hat{g}^{\mathrm{x}}, \hat{g}^{\mathrm{y}}$ are unbiased

$$\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{x}}(x, y)] = \sum_{i\in[N]} \bar{p}_i \cdot \frac{{\psi_\epsilon^*}'(\ell_i(x) - Gy)}{\bar{p}_i} \nabla\ell_i(x) = \nabla_x\Upsilon_\epsilon(x, y),$$

and

$$\mathbb{E}_{i\sim\bar{p}_i}[\hat{g}^{\mathrm{y}}(x, y)] = \sum_{i\in[N]} \bar{p}_i \cdot \left(G\left(1 - \frac{{\psi_\epsilon^*}'(\ell_i(x) - Gy)}{\bar{p}_i}\right)\right) = G - G\sum_{i\in[N]} {\psi_\epsilon^*}'(\ell_i(x) - Gy) = \nabla_y\Upsilon_\epsilon(x, y).$$

Next, we bound the second moment of the stochastic gradients. For any i we have

$$\|\hat{g}^{\mathrm{x}}(x, y)\| = \frac{{\psi_\epsilon^*}'(\ell_i(x) - Gy)}{{\psi_\epsilon^*}'(\ell_i(\bar{x}) - G\bar{y})}\|\nabla\ell_i(x)\| \overset{(i)}{\le} e^{\frac{\ell_i(x) - Gy - (\ell_i(\bar{x}) - G\bar{y})}{\epsilon'}} G \overset{(ii)}{\le} e^2G$$

where (i) follows from Lemma 4 and the fact that ℓ_i is G-Lipschitz and (ii) uses G-Lipschitzness again together with $x \in \mathbb{B}_{r_\epsilon}(\bar{x})$ and $y \in [\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$ to deduce that $\ell_i(x) - Gy - (\ell_i(\bar{x}) - G\bar{y}) \le 2Gr_\epsilon \le 2\epsilon'$. Therefore, we have $\mathbb{E}\|\hat{g}^{\mathrm{x}}(x, y)\|^2 \le e^4G^2$ as required. The second moment bound on $\hat{g}^{\mathrm{y}}(x, y)$ follows similarly, since

$$|\hat{g}^{\mathrm{y}}(x, y)| \le G\max\left\{1, \frac{{\psi_\epsilon^*}'(\ell_i(x) - Gy)}{{\psi_\epsilon^*}'(\ell_i(\bar{x}) - G\bar{y})}\right\} \le e^2G.$$

□

Proposition 4. *Let $\epsilon, \lambda > 0$, $\epsilon' = \frac{\epsilon}{2\log N}$ and $r_\epsilon = \frac{\epsilon'}{G}$. For any query point $\bar{x}$ let $\bar{y} = \mathrm{argmin}_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(\bar{x}, y)$ and let $x_\star, y_\star = \mathrm{argmin}_{x\in\mathbb{B}_{r_\epsilon}(\bar{x}), y\in[\bar{y}-r_\epsilon, \bar{y}+r_\epsilon]} \Upsilon_{\epsilon,\lambda}(x, y)$. For $\gamma_k = \frac{1}{8\lambda 2^k}$, $T \ge 1$ and threshold $T_{\text{threshold}} = \frac{G^4}{\lambda^2\epsilon'^2}$ the output (x, y) of Algorithm 3 satisfies*

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(x, y) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le O\left(\frac{G^2}{\lambda T}\right).$$

Proof. For convenience let $x_k = x_k^{(0)}$ and $y_k = y_k^{(0)}$, and in addition let $\bar{p}_i = {\psi'_\epsilon}^*(\ell_i(\bar{x}) - G\bar{y})$ be the sampling probability from Algorithm 3. We use induction to prove that

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(x_k, y_k) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le \frac{e^4G^2}{\lambda 2^k}$$

for all k. We start with the base case (k=1).

$$\begin{aligned}\Upsilon_{\epsilon,\lambda}(x_\star, y_\star) - \Upsilon_{\epsilon,\lambda}(x_1, y_1) &= \Upsilon_\epsilon(x_\star, y_\star) - \Upsilon_\epsilon(x_1, y_1) + \frac{\lambda}{2}\|x_\star - x_1\|^2 \\ &\overset{(i)}{\ge} \langle\nabla_x\Upsilon_\epsilon(x_1, y_1), x_\star - x_1\rangle + \langle\nabla_y\Upsilon_\epsilon(x_1, y_1), y_\star - y_1\rangle + \frac{\lambda}{2}\|x_\star - x_1\|^2 \\ &\overset{(ii)}{\ge} \langle\nabla_x\Upsilon_\epsilon(x_1, y_1), x_\star - x_1\rangle + \frac{\lambda}{2}\|x_\star - x_1\|^2\end{aligned}$$

where (i) follows from convexity of Υ_ϵ and (ii) since due to optimality conditions we have that $\langle\nabla_y\Upsilon_\epsilon(x_1, y_1), y_\star - y_1\rangle \ge 0$. Therefore,

$$\begin{aligned}\Upsilon_{\epsilon,\lambda}(x_1, y_1) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) &\le -\langle\nabla_x\Upsilon_\epsilon(x_1, y_1), x_\star - x_1\rangle - \frac{\lambda}{2}\|x_\star - x_1\|^2 \\ &\le \max_{x\in\mathbb{B}_{r_\epsilon}(\bar{x})}\left\{-\langle\nabla_x\Upsilon_\epsilon(x_1, y_1), x - x_1\rangle - \frac{\lambda}{2}\|x - x_1\|^2\right\} \\ &= \frac{\|\nabla_x\Upsilon_\epsilon(x_1, y_1)\|^2}{2\lambda} \le \frac{e^4G^2}{2\lambda}\end{aligned}$$

Algorithm 3: Dual EpochSGD

Input: The function Υ_ϵ defined in (12), ball center $\bar{x}$, ball radius r_ϵ, regularization parameter λ, smoothing parameter ϵ' and iteration budget T.

Parameters: Initial step size $\gamma_1 = 1/(16\lambda)$, epoch length $T_1 = 128$ and threshold $T_{\text{threshold}} = \frac{G^4}{\lambda^2 \epsilon'^2}$.

Initialize $x_1^{(0)} = \bar{x}$

Initialize $y_1^{(0)} = \bar{y} = \operatorname{argmin}_{y \in \mathbb{R}} \Upsilon_\epsilon(\bar{x}, y)$

Precomupte sampling probabilities $\bar{p}_i = {\psi'_\epsilon}^*(\ell_i(\bar{x}) - G\bar{y})$

for $k = 1, \ldots, \lceil \log(T/128 + 1) \rceil$ **do**

  for $t = 0, 2, \cdots T_k - 1$ **do**

    Sample $i \sim \bar{p}_i$

    Query stochastic gradients $\hat{g}^{\mathrm{x}}\left(x_k^{(t)}, y_k^{(t)}\right)$ and $\hat{g}^{\mathrm{y}}\left(x_k^{(t)}, y_k^{(t)}\right)$ defined in (13)

    Update $x_k^{(t+1)} = \operatorname{argmin}_{x \in \mathbb{B}_{r_\epsilon}(\bar{x})} \left\{ \gamma_k \left(\langle \hat{g}^{\mathrm{x}}, x \rangle + \frac{\lambda}{2} \|\bar{x} - x\|^2 \right) + \frac{1}{2} \|x_k^{(t)} - x\|^2 \right\}$

    Update $y_k^{(t+1)} = \operatorname{argmin}_{y \in [\bar{y} - \frac{\epsilon'}{G}, \bar{y} + \frac{\epsilon'}{G}]} \left\{ \gamma_k (\hat{g}^{\mathrm{y}} \cdot y) + \frac{1}{2} \left(y_k^{(t)} - y \right)^2 \right\}$

  Set $x_{k+1}^{(0)} = \frac{1}{T_k} \sum_{t \in [T_k]} x_k^{(t)}$

  Set $y_{k+1}^{(0)} = \frac{1}{T_k} \sum_{t \in [T_k]} y_k^{(t)}$

  Update $T_{k+1} = 2T_k$

  Update $\gamma_{k+1} = \gamma_k / 2$

  $k \leftarrow k + 1$

  if $T_k \geq T_{\text{threshold}}$ **then**

    Recompute $y_{k+1}^{(0)} = \operatorname{argmin}_{y \in \mathbb{R}} \Upsilon_\epsilon(x_{k+1}^{(0)}, y)$

return $x = x_k^{(0)}$

where the last inequality follows from Jensen's inequality and Lemma 6. This gives the base case of our induction. Let $V_x(x') = \frac{1}{2}\|x - x'\|^2$ and $V_y(y') = \frac{1}{2}|y - y'|^2$ and suppose that there is a k such that $\mathbb{E}\Upsilon_{\epsilon,\lambda}(x_k, y_k) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \leq \frac{e^4 G^2}{\lambda 2^k}$. Then, using the mirror descent regret bound [see, e.g., 3, Lemma 3] for $k+1$ we get

$$\begin{aligned}
\mathbb{E}\Upsilon_{\epsilon,\lambda}(x_{k+1}, y_{k+1}) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) &\leq \frac{\mathbb{E}V_{x_k}(x_\star)}{\gamma_k T_k} + \frac{\mathbb{E}V_{y_k}(y_\star)}{\gamma_k T_k} \\
&\quad + \frac{\gamma_k}{2} \frac{1}{T_k} \sum_{t=1}^{T_k} \mathbb{E}\|\hat{g}^{\mathrm{x}}(x_k, y_k)\|^2 + \frac{\gamma_k}{2} \frac{1}{T_k} \sum_{t=1}^{T_k} \mathbb{E}\|\hat{g}^{\mathrm{y}}(x_{k,k})\|^2 \\
&\stackrel{(i)}{\leq} \frac{\mathbb{E}V_{x_k}(x_\star)}{\gamma_k T_k} + \frac{\mathbb{E}V_{y_k}(y_\star)}{\gamma_k T_k} + 2e^4 \gamma_k G^2 \\
&\stackrel{(ii)}{\leq} \frac{\lambda \mathbb{E}V_{x_k}(x_\star)}{8} + \frac{\lambda \mathbb{E}V_{y_k}(y_\star)}{8} + \frac{e^4 G^2}{2\lambda 2^{k+1}}
\end{aligned}$$

with (i) following from Lemma 6 and (ii) from the choice of $T_k = 64 \cdot 2^k$ and $\gamma_k = \frac{1}{\lambda 2^{k+3}}$. Next, note that the strong convexity of $\Upsilon_\epsilon(x, y)$ in x implies $\lambda \mathbb{E}V_{x_k}(x_\star) \leq \frac{e^4 G^2}{\lambda 2^k}$ since $\lambda \mathbb{E}V_{x_k}(x_\star) \leq \mathbb{E}\lambda\Upsilon_{\epsilon,\lambda}(x_k, y_k) - \mathbb{E}\Upsilon_{\epsilon,\lambda}(x_\star, y_k) \leq \mathbb{E}\lambda\Upsilon_{\epsilon,\lambda}(x_k, y_k) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \leq \frac{e^4 G^2}{\lambda 2^k}$ by the induction hypothesis. In addition from Lemma 5 we have

$$G|y_k - y_\star| \leq G r_\epsilon = \epsilon'$$

by the constraint on y. We now bound $\mathbb{E}V_{y_k}(y_\star)$ in each scenario $T_k \leq T_{\text{threshold}} = \frac{G^4}{\lambda^2 \epsilon'^2}$ or $T_k > T_{\text{threshold}}$.

1. If $T_k \leq T_{\text{threshold}}$ we have that $|y_k - y_\star| \leq \frac{\epsilon'}{G} \leq \frac{G}{\lambda 2^{k/2+2}}$ and thus $V_{y_k}(y_\star) \leq \frac{G^2}{\lambda^2 2^k}$.

2. If $T_k > T_{\text{threshold}}$ Algorithm 3 will recompute the optimal y_k and using Lemma 5 we have $|y_k - y_\star| \le \|x_k - x_\star\|$ and as a result $V_{y_k}(y_\star) \le V_{x_k}(x_\star)$.

Therefore $\mathbb{E}V_{y_k}(y_\star) \le \frac{e^4 G^2}{\lambda^2 2^k}$ and substituting back the bounds on $\mathbb{E}V_{y_k}(y_\star)$ and $\mathbb{E}V_{x_k}(x_\star)$ we obtain

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(x_{k+1}, y_{k+1}) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le \frac{e^4 G^2}{\lambda 2^{k+1}}$$

which completes the induction. Let K be the iteration where the algorithm outputs $x = x_K^{(0)}$ and let $y = y_K^{(0)}$. Noting that $T = O(2^K)$, we have

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(x, y) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le O\left(\frac{G^2}{\lambda T}\right).$$

□

Theorem 3. *Let each ℓ_i satisfy Assumption 1. Let $\epsilon, \lambda, \delta > 0$, and $r_\epsilon = \epsilon/(2G\log N)$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \le O(G/r_\epsilon)$ and accuracy $\delta < r_\epsilon/2$, Algorithm 3 outputs a valid r_ϵ-BROO response for $\mathcal{L}_{\psi,\epsilon}$ and has complexity $\mathcal{C}_\lambda(\delta) = O\big(\frac{G^2}{\lambda^2\delta^2} + N\log\big(\frac{r_\epsilon}{\delta}\big)\big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$ (10) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{11/3} H + \left(\frac{GR}{\epsilon}\right)^2 \log^2 H\right) \text{ where } H \coloneqq N\frac{GR}{\epsilon}.$$

Proof. We divide the proof into correctness arguments and complexity bounds.

BROO implementation: correctness. We use Algorithm 3 with $T = O\Big(\frac{G^2}{\lambda^2\delta^2}\Big)$ for the BROO implementation. Applying Proposition 4 the output (x, y) of Algorithm 3 satisfies

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(x, y) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le O\big(\lambda\delta^2\big).$$

Therefore, there is a constant $c > 0$ for which the output (x, y) of Algorithm 3 with $T = \frac{cG^2}{\lambda^2\delta^2}$ satisfies $\mathbb{E}\Upsilon_{\epsilon,\lambda}(x, y) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star) \le \frac{\lambda\delta^2}{2}$, and since

$$\mathbb{E}\mathcal{L}_{\psi,\epsilon}(x) - \min_x \mathcal{L}_{\psi,\epsilon}(x) = \mathbb{E}\min_y \Upsilon_\epsilon(x, y) - \Upsilon_\epsilon(x_\star, y_\star) \le \mathbb{E}\Upsilon_{\epsilon,\lambda}(x, y) - \Upsilon_{\epsilon,\lambda}(x_\star, y_\star),$$

Algorithm 3 returns a valid BROO response for $\mathcal{L}_{\psi,\epsilon}$.

BROO implementation: complexity. The total number of epochs that Algorithm 3 performs is $K = O\Big(\log\big(\frac{G^2}{\delta^2\lambda^2}\big)\Big)$. In $O\Big(\log\Big(\frac{G^2}{\lambda^2\delta^2}\Big) - \log\Big(\frac{G^2}{\lambda^2 r_\epsilon^2}\Big)\Big) = O\big(\log\big(\frac{r_\epsilon}{\delta}\big)\big)$ epochs with $T \ge \frac{G^4}{\lambda^2\epsilon'^2} = \frac{G^2}{r_\epsilon^2\lambda^2}$ the algorithm performs $O(N)$ function evaluations to recompute the optimal y. In addition, the algorithm performs $O\Big(\frac{G^2}{\lambda^2\delta^2}\Big)$ stochastic gradient computations (each computation involves a single loss function ℓ_i and a single sub-gradient $\nabla\ell_i$ evaluation). Therefore the total complexity of the BROO implementation is

$$O\left(\frac{G^2}{\lambda^2\delta^2} + N\log\left(\frac{r_\epsilon}{\delta}\right)\right). \tag{39}$$

Minimizing $\mathcal{L}_\psi$: correctness. For any $q \in \Delta^N$ let $\mathcal{L}_q(x) \coloneqq \sum_{i\in[N]} q_i\ell_i(x) - \psi_\epsilon(q_i)$ and note that $\mathcal{L}_q$ is G-Lipschitz, since for all $x \in \mathcal{X}$ we have $\|\nabla\mathcal{L}_q(x)\| = \Big\|\sum_{i\in[N]} q_i\nabla\ell_i(x)\Big\| \le G$. Maximum operations preserve the Lipschitz continuity and therefore $\mathcal{L}_{\psi,\epsilon}(x) = \max_{q\in\Delta^N}\mathcal{L}_q(x)$ is also G-Lipschitz. Since $\mathcal{L}_{\psi,\epsilon}$ is G-Lipschitz, we can use Proposition 1 with $F = \mathcal{L}_{\psi,\epsilon}$ and obtain that with probability at least $\frac{1}{2}$ Algorithm 1 outputs a point x such that $\mathcal{L}_{\psi,\epsilon}(x) - \min_x \mathcal{L}_{\psi,\epsilon}(x) \le \epsilon/2$. In addition, from Lemma 18 we have

$$\mathcal{L}_\psi(x) - \min_x \mathcal{L}_\psi(x) \le \mathcal{L}_{\psi,\epsilon}(x) - \min_x \mathcal{L}_{\psi,\epsilon}(x) + \frac{\epsilon}{2} \le \epsilon.$$

Minimizing $\mathcal{L}_\psi$: complexity. We apply Proposition 1 with $F = \Upsilon_{\epsilon,\lambda}$ and $r_\epsilon = \epsilon/(2\log N \cdot G)$, thus the complexity of finding an $\epsilon/2$-suboptimal minimizer of $\mathcal{L}_{\psi,\epsilon}$ (and therefore an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$) is bounded as:

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right)m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon)+N)m_\epsilon^3,\right]\right)$$

where $m_\epsilon = O\left(\log\frac{GR^2}{\epsilon r_\epsilon}\right) = O\left(\log\frac{GR}{\epsilon}\log N\right)$ and $\lambda_{\mathrm{m}} = O\left(\frac{m_\epsilon^2\epsilon}{r_\epsilon^{4/3}R^{2/3}}\right)$. To obtain the total complexity bound, we evaluate the complexity of running r_ϵ-BROO with accuracy $\delta_j = \frac{r_\epsilon}{2^{j/2}m_\epsilon^2}$ and $\delta = \frac{r_\epsilon}{30}$. Using (39) we get

1. $\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{m_\epsilon^2 2^{j/2}}\right) = O\left(\frac{G^2 2^j m_\epsilon^4}{\lambda_{\mathrm{m}}^2 r_\epsilon^2} + N\log\left(m_\epsilon^2 2^{j/2}\right)\right) = O\left(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{(\log N)^{2/3}}2^j + N\left(m_\epsilon + \log\left(2^{j/2}\right)\right)\right)$
2. $\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{30}\right) = O\left(\frac{G^2}{\lambda_{\mathrm{m}}^2 r_\epsilon^2} + N\right) = O\left(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{m_\epsilon^4(\log N)^{2/3}} + N\right)$.

Thus

$$O\left(m_\epsilon\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right) \le O\left(m_\epsilon^2\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{(\log N)^{2/3}} + m_\epsilon^2 N\right)$$

and

$$O\left((\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon)+N)m_\epsilon^3\right) \le O\left(\frac{\left(\frac{GR}{\epsilon}\right)^{4/3}}{m_\epsilon(\log N)^{2/3}} + Nm_\epsilon^3\right).$$

Substituting the bounds into Proposition 1, the total complexity is bounded as

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[Nm_\epsilon^3 + \frac{m_\epsilon^2\left(\frac{GR}{\epsilon}\right)^{4/3}}{(\log N)^{2/3}}\right]\right) \le O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{11/3}\left(\frac{GR}{\epsilon}\log N\right) + \left(\frac{GR}{\epsilon}\right)^2\log^2\left(\frac{GR}{\epsilon}\log N\right)\right)$$

□

D.6 Accelerated variance reduction BROO implementation

In this section we prove the complexity guarantees of our BROO implementation for (potentially only slightly) smooth losses. We first prove Lemma 7, showing that $\Upsilon_{\epsilon,\lambda}$ (the approximation of our objective (10)) is a finite sum of smooth functions, and thus for the BROO implementation we can use a variance reduction method for a finite (weighted) sums. Then, we give Definition 3 of a "valid accelerated variance reduction" (VR) method, and in Lemma 20 we prove the convergence rate of our BROO implementation (Algorithm 4) which is simply a restart scheme utilizing any valid accelerated VR method. We then combine the guarantees of Lemma 20 and Proposition 1 to prove Theorem 4, our final complexity guarantee in the smooth.

To begin, let us restate here the definition of $\Upsilon_{\epsilon,\lambda}$ (that has the form of a weighted finite sum):

$$\Upsilon_{\epsilon,\lambda}(x,y) = \sum_{i\in[N]}\bar{p}_i\upsilon_i(x,y) \;\text{ where }\; \upsilon_i(x,y) \coloneqq \frac{\psi_\epsilon^*(\ell_i(x)-Gy)}{\bar{p}_i} + Gy + \frac{\lambda}{2}\|x-\bar{x}\|^2 \tag{40}$$

and $\bar{p}_i = \psi_\epsilon^{*\prime}(\ell_i(\bar{x}) - G\bar{y})$.

Lemma 7. *For any $i\in[N]$, let ℓ_i be G-Lipschitz and L -smooth, let $r_\epsilon = \frac{\epsilon'}{G}$ and $\lambda = O\left(\frac{G}{r_\epsilon}\right)$. The restriction of υ_i to $x\in\mathbb{B}_{r_\epsilon}(\bar{x})$ and $y\in[\bar{y}-r_\epsilon,\bar{y}+r_\epsilon]$ is $O(G)$-Lipschitz and $O\left(L+\frac{G^2}{\epsilon'}\right)$-smooth.*

Proof. To show the Lipschitz property we compute the gradient of $\upsilon_i(x, y)$ with respect to x and y.

$$\nabla_x \upsilon_i(x, y) = \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x) + \lambda(x - \bar{x})$$

$$\nabla_y \upsilon_i(x, y) = G - G\frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i}$$

Similarly to the proof of Lemma 6, we have that $\frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \leq e^2$ and therefore $\|\nabla_x \upsilon_i(x, y)\| \leq e^2 G = O(G)$ and $|\nabla_y \upsilon_i(x, y)| \leq e^2 G = O(G)$, giving the first statement. To bound the smoothness of υ_i, we compute the Hessian of $\upsilon_i(x, y)$.

$$\nabla_x^2 \upsilon_i(x, y) = \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla^2 \ell_i(x) + \frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x) \nabla \ell_i(x)^T + \lambda I$$

$$\nabla_y^2 \upsilon_i(x, y) = G^2 \frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\bar{p}_i}$$

$$\nabla_{xy} \upsilon_i(x, y) - G\frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x).$$

Lemma 4 implies that, for all v, $\frac{\psi_\epsilon^{*\prime\prime}(v)}{\psi_\epsilon^{*\prime}(v)} = \left(\log \psi_\epsilon^{*\prime}(v)\right)' \leq \frac{1}{\epsilon'}$ and $\frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \leq e^2$. In addition note that each ℓ_i is L-smooth and G-Lipschitz and $\lambda = O\left(\frac{G}{r_\epsilon}\right) = O\left(\frac{G^2}{\epsilon'}\right)$, therefore

$$\begin{aligned}
\|\nabla_x^2 \upsilon_i(x, y)\|_{\text{op}} &= \left\| \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla^2 \ell_i(x) + \frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)} \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x) \nabla \ell_i(x)^T + \lambda I \right\|_{\text{op}} \\
&\leq O\left(e^2\left(L + 2\frac{G^2}{\epsilon'}\right)\right)
\end{aligned}$$

$$\|\nabla_{xy} \upsilon_i(x, y)\| = \left\| G\frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)} \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x) \right\| \leq e^2 \frac{G^2}{\epsilon'}$$

$$\nabla_y^2 \upsilon_i(x, y) = G^2 \frac{\psi_\epsilon^{*\prime\prime}(\ell_i(x) - Gy)}{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)} \frac{\psi_\epsilon^{*\prime}(\ell_i(x) - Gy)}{\bar{p}_i} \nabla \ell_i(x) \leq e^2 \frac{G^2}{\epsilon'}.$$

Applying Lemma 21 with $h = \upsilon_i$ we get that

$$\left\|\nabla^2 \upsilon_i(x, y)\right\|_{\text{op}} \leq e^2\left(L + 2\frac{G^2}{\epsilon'}\right),$$

proving that each $\upsilon_i(x, y)$ is $O\left(L + \frac{G^2}{\epsilon'}\right)$-smooth. $\square$

Definition 3. *For a given ball center $\bar{x} \in \mathcal{X}$, let $z_\star \in \mathbb{B}_{r_\epsilon}(\bar{x}) \times \mathbb{R}$ minimize the function $\Upsilon_{\epsilon,\lambda} : \mathbb{B}_{r_\epsilon}(\bar{x}) \times \mathbb{R} \to \mathbb{R}$, and let $\bar{y} = \operatorname{argmin}_{y \in \mathbb{R}} \Upsilon_{\epsilon,\lambda}(\bar{x}, y)$. Let* VarianceReduction *be a procedure that takes in $z \in \mathcal{X} \times \mathbb{R}$ and complexity budget T, and outputs $z' \in \mathcal{X} \times \mathbb{R}$. We say that* VarianceReduction *is* a valid accelerated VR method *if it has complexity T and satisfies the following: there a constant C such that for any α, input z, and*

$$T \geq C\left(N\frac{\Upsilon_{\epsilon,\lambda}(z) - \Upsilon_{\epsilon,\lambda}(z_\star)}{\alpha} + \sqrt{\frac{\widetilde{L}\|z - z_\star\|^2}{\alpha}}\right) \text{ for } \widetilde{L} = L + \frac{G^2}{\epsilon'},$$

the output z' of VarianceReduction$(z; T)$ *satisfies*

$$\mathbb{E}\Upsilon_{\epsilon,\lambda}(z') - \Upsilon_{\epsilon,\lambda}(z_\star) \leq \alpha.$$

Algorithm 4: Restarting Accelerated Variance Reduction with Optimal Dual Values

Input: The function $\Upsilon_{\epsilon,\lambda}(x,y) = \sum_{i\in[N]} \bar{p}_i \upsilon_i(x,y)$ defined in (40), number of total restarts K, and an algorithm VarianceReduction that takes in $x, y \in \mathcal{X} \times \mathbb{R}$ and complexity budget T, and outputs $x', y' \in \mathcal{X} \times \mathbb{R}$.

$x_0, y_0 = \bar{x}, \bar{y} = \bar{x}, \mathrm{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(\bar{x}, y)$
for $k = 1, \ldots, K$ **do**
  $x_k, y'_k = \textsc{VarianceReduction}(x_{k-1}, y_{k-1}; T)$
  $y_k = \mathrm{argmin}_{y\in\mathbb{R}} \Upsilon_\epsilon(x_k, y)$
return x_K

Lemma 19. Katyusha[sf] *[38] is a valid accelerated VR method for some $C = O(1)$.*

Proof. Immediate from [1, Theorem 4.1] and Lemma 7. (We note that the theorem is stated for a finite sum with uniform weights, but the extension of the theorem and the method to non-uniform sampling is standard). □

The following lemma shows that Algorithm 4, when coupled with any valid accelerated VR method yields a BROO implementation for

$$\mathcal{L}_{\psi,\epsilon} = \max_{q\in\Delta^N} \sum_{i\in[N]} (q_i \ell_i(x) - \psi_\epsilon(q_i)) = \min_{y\in\mathbb{R}} \Upsilon_\epsilon(x,y),$$

i.e., it outputs an approximate minimizer of

$$\mathcal{L}_{\psi,\epsilon,\lambda}(x) \coloneqq \mathcal{L}_{\psi,\epsilon}(x) + \frac{\lambda}{2}\|x - \bar{x}\|^2 = \min_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(x,y)$$

in $\mathbb{B}_{r_\epsilon}(\bar{x})$.

Lemma 20. *Let Assumptions 1 and 3 hold, and suppose Algorithm 4 uses a valid accelerated VR method with constant C (defined above). Then, for $\widetilde{L} = L + G^2/\epsilon'$ and $T \ge 2C(N + \sqrt{N\widetilde{L}/\lambda})$, for any $K \ge 0$ the output x of Algorithm 4 satisfies*

$$\mathbb{E}\mathcal{L}_{\psi,\epsilon,\lambda}(x) - \min_{x_\star\in\mathbb{B}_{r_\epsilon}(\bar{x})} \mathbb{E}\mathcal{L}_{\psi,\epsilon,\lambda}(x_\star) \le 2^{-K}\left(\mathcal{L}_{\psi,\epsilon,\lambda}(\bar{x}) - \min_{x_\star\in\mathbb{B}_{r_\epsilon}(\bar{x})} \mathbb{E}\mathcal{L}_{\psi,\epsilon,\lambda}(x_\star)\right).$$

Moreover, the complexity of Algorithm 4 is $K(N+T) = O\left(K(N + \sqrt{N\widetilde{L}/\lambda})\right)$

Proof. Let $z_\star = (x_\star, y_\star)$ minimize $\Upsilon_{\epsilon,\lambda}$ in $\mathbb{B}_{r_\epsilon}(\bar{x}) \times \mathbb{R}$, so that $x_\star = \mathrm{argmin}_{x\in\mathbb{B}_{r_\epsilon}(\bar{x})} \mathcal{L}_{\psi,\epsilon,\lambda}(x)$ as well. Note that for all of the outer loop iterations $z_k = (x_k, y_k)$ we have, by the optimality of y_k and Lemma 5,

$$\|z_k - z_\star\|^2 = \|x_k - x_\star\|^2 + (y_k - y_\star)^2 \le 2\|x_k - x_\star\|^2.$$

Moreover, the λ-strong convexity of $\mathcal{L}_{\psi,\epsilon,\lambda}$ implies that

$$\|x_k - x_\star\|^2 \le \frac{2(\mathcal{L}_{\psi,\epsilon,\lambda}(x_k) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star))}{\lambda}.$$

Furthermore, note that $\Upsilon_{\epsilon,\lambda}(z_k) = \min_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(x_k, y) = \mathcal{L}_{\psi,\epsilon,\lambda}(x_k)$. Recalling that by Lemma 5 restricting the domain of y to $[\bar{y} - r_\epsilon, \bar{y} + r_\epsilon]$ does not change the optimal y, we conclude that for VarianceReduction to guarantee $\Upsilon_{\epsilon,\lambda}(x_{k+1}, y'_{k+1}) - \Upsilon_{\epsilon,\lambda}(z_\star) \le \alpha$ it suffices to choose

$$T \ge C\left(N\frac{\mathcal{L}_{\psi,\epsilon,\lambda}(x_k) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star)}{\alpha} + \sqrt{\frac{4\widetilde{L}(\mathcal{L}_{\psi,\epsilon,\lambda}(x_k) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star))}{\lambda\alpha}}\right)$$

In particular, we see that $T \geq 2C\left(N + \sqrt{N\widetilde{L}/\lambda}\right)$ suffices for $\alpha = \frac{\mathcal{L}_{\psi,\epsilon,\lambda}(x_k) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star)}{2}$, from which we conclude that

$$\begin{aligned}\mathcal{L}_{\psi,\epsilon,\lambda}(x_{k+1}) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star) &= \Upsilon_{\epsilon,\lambda}(x_{k+1}, y_{k+1}) - \Upsilon_{\epsilon,\lambda}(z_\star) \\ &\leq \Upsilon_{\epsilon,\lambda}(x_{k+1}, y'_{k+1}) - \Upsilon_{\epsilon,\lambda}(z_\star) \leq \frac{\mathcal{L}_{\psi,\epsilon,\lambda}(x_k) - \mathcal{L}_{\psi,\epsilon,\lambda}(x_\star)}{2},\end{aligned}$$

giving the claimed optimality bound. Finally, the complexity of the method is clearly $K(T+N)$ since we make K calls to VarianceReduction with complexity T and K exact minimizations over y with complexity N. □

To efficiently implement the BROO , we repeatedly apply an accelerated variance reduction scheme that does not require strong convexity, such as Katyusha^{ns} [2], each time with complexity budget $\widetilde{O}(N + \sqrt{NL'/\lambda})$, where $L' = L + G^2/\epsilon'$. We start each repetition by the x variable output by the previous Katyusha^{ns} call, and with $y = \mathrm{argmin}_{y\in\mathbb{R}} \Upsilon_{\epsilon,\lambda}(y, x)$ for that x. Using Lemma 5 in lieu of strong-convexity in y, we show that error halves after each restart, and therefore a logarithmic number of restarts suffices. We arrive at the following complexity bound.

Theorem 4. *Let each ℓ_i satisfy Assumptions 1 and 3, let $\epsilon, \lambda, \delta > 0$, and $r_\epsilon = \frac{\epsilon}{2G\log N}$. For any query point $\bar{x} \in \mathbb{R}^d$, regularization strength $\lambda \leq O(\frac{G}{r_\epsilon})$ and accuracy δ, Algorithm 4 outputs a valid r_ϵ-BROO response for $\mathcal{L}_{\psi,\epsilon}$ and has complexity $\mathcal{C}_\lambda(\delta) = O\Big(\Big(N + \frac{\sqrt{N}(G+\sqrt{\epsilon' L})}{\sqrt{\lambda\epsilon'}}\Big)\log\frac{Gr_\epsilon}{\lambda\delta^2}\Big)$. Consequently, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$ (10) with probability at least $\frac{1}{2}$ is*

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3} H + \sqrt{N}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{5/2} H\right) \text{ where } H := N\frac{GR}{\epsilon}.$$

Proof. We first show correctness and complexity of the BROO implementation and then argue the same points for the overall method.

BROO implementation: correctness and complexity. Combining Lemmas 19 and 20 and setting $K = \log_2 \frac{\Upsilon_\epsilon(x_0,y_0) - \mathrm{argmin}_{x\in\mathbb{B}_{r_\epsilon}(\bar{x}),\, y\,\in\mathbb{R}} \Upsilon_\epsilon(x,y)}{\lambda\delta^2}$, we obain a valid BROO implementation with complexity

$$O\left(\left(N + \frac{\sqrt{N}\left(\sqrt{L\epsilon'} + G\right)}{\sqrt{\epsilon'\lambda}}\right)\log\left(\frac{\Upsilon_\epsilon(x_0,y_0) - \mathrm{argmin}_{x\in\mathbb{B}_{r_\epsilon}(\bar{x}), y\in\mathbb{R}} \Upsilon_\epsilon(x,y)}{\lambda\delta^2}\right)\right) \tag{41}$$

Furthermore, we note that Υ_ϵ is $O(G)$-Lipschitz and therefore $\Upsilon_\epsilon(x_0,y_0) - \mathrm{argmin}_{x\in\mathbb{B}_{r_\epsilon}(x), y\in\mathbb{R}} \Upsilon_\epsilon(x,y) \leq O(Gr_\epsilon)$.

Minimizing $\mathcal{L}_\psi$: correctness. Similarly to the proof of Theorem 3, we note that $\mathcal{L}_{\psi,\epsilon}$ is G-Lipschitz, and therefore we can apply Proposition 1 with $F = \mathcal{L}_{\psi,\epsilon}$ and obtain that with probability at least $\frac{1}{2}$ Algorithm 1 outputs x such that $\mathcal{L}_{\psi,\epsilon}(x) - \min_x \mathcal{L}_{\psi,\epsilon}(x) \leq \epsilon/2$ and Lemma 18 gives

$$\mathcal{L}_\psi(x) - \min_x \mathcal{L}_\psi(x) \leq \mathcal{L}_{\psi,\epsilon}(x) - \min_x \mathcal{L}_{\psi,\epsilon}(x) + \frac{\epsilon}{2} \leq \epsilon.$$

Minimizing $\mathcal{L}_\psi$: complexity. Applying Proposition 1 with $F = \Upsilon_\epsilon$ and $r_\epsilon = \frac{\epsilon}{2G\log N}$, the complexity of finding an ϵ-suboptimal minimizer of $\mathcal{L}_\psi$ is

$$O\left(\left(\frac{R}{r_\epsilon}\right)^{2/3}\left[\left(\sum_{j=0}^{m_\epsilon}\frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right)\right)m_\epsilon + (\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon) + N)m_\epsilon^3\right]\right) \tag{42}$$

where $m_\epsilon = O\big(\log\frac{GR^2}{\epsilon r_\epsilon}\big) = \log\big(\frac{GR}{\epsilon}\log N\big)$ and $\lambda_{\mathrm{m}} = O\big(\frac{m_\epsilon^2\epsilon}{r^{4/3}R^{2/3}}\big)$. Using similar calculations to the proof of Theorem 2 we obtain that

1. $\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right) = O\Big(\Big(N + N^{1/2}\Big(\frac{G\sqrt{\log N}}{\sqrt{\epsilon}} + \sqrt{L}\Big)\frac{1}{\sqrt{\lambda_{\mathrm{m}}}}\Big)\log\Big(\frac{\epsilon' 2^j m_\epsilon^4}{\lambda_{\mathrm{m}} r_\epsilon^2}\Big)\Big)$
2. $\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{30}\right) = O\Big(\Big(N + N^{1/2}\Big(\frac{G\sqrt{\log N}}{\sqrt{\epsilon}} + \sqrt{L}\Big)\frac{1}{\sqrt{\lambda_{\mathrm{m}}}}\Big)\log\Big(\frac{\epsilon'}{\lambda_{\mathrm{m}} r_\epsilon^2}\Big)\Big)$

with $\frac{\epsilon'}{\lambda_{\mathrm{m}} r_\epsilon^2} = O\Big(\big(\frac{GR}{\epsilon}\big)^{2/3}\frac{1}{m_\epsilon^2}\Big)$ and $\frac{1}{\sqrt{\lambda_{\mathrm{m}}}} = O\Big(\frac{R^{1/3} r_\epsilon^{2/3}}{m_\epsilon\sqrt{\epsilon}}\Big)$. Therefore,

$$m_\epsilon \sum_{j=0}^{\infty} \frac{1}{2^j}\mathcal{C}_{\lambda_{\mathrm{m}}}\left(\frac{r_\epsilon}{2^{j/2}m_\epsilon^2}\right) \le O\left(m_\epsilon^2\left[N + N^{1/2}\left(\frac{GR^{1/3}}{\epsilon} + \sqrt{\frac{LR^{2/3}}{\epsilon}}\right) r_\epsilon^{2/3}\right]\right).$$

and

$$O\big((\mathcal{C}_{\lambda_{\mathrm{m}}}(r_\epsilon) + N)m_\epsilon^3\big) \le O\left(m_\epsilon^4 N + m_\epsilon^{3.5} N^{1/2}\left(\frac{GR^{1/3}}{\epsilon} + \sqrt{\frac{LR^{2/3}}{\epsilon}}\right) r_\epsilon^{2/3}\right).$$

Substituting the bounds into Proposition 1 with $m_\epsilon = \log\big(\frac{GR}{\epsilon}\log N\big)$ and $r_\epsilon = \frac{\epsilon}{2\log N}$ the total complexity is

$$O\left(N\left(\frac{GR}{\epsilon}\right)^{2/3}\log^{14/3}\left(\frac{GR}{\epsilon}\log N\right) + N^{1/2}\left(\frac{GR}{\epsilon} + \sqrt{\frac{LR^2}{\epsilon}}\right)\log^{7/2}\left(\frac{GR}{\epsilon}\log N\right)\right).$$

□

D.7 Helper lemmas

Lemma 21. *Let $x \in \mathbb{R}^d$, $y \in \mathbb{R}$, then for any $h : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ we have*

$$\left\|\nabla^2 h(x,y)\right\|_{\text{op}} \le \max\Big\{\nabla_y^2 h(x,y) + \|\nabla_{xy} h(x,y)\|, \left\|\nabla_x^2 h(x,y)\right\|_{\text{op}} + \|\nabla_{xy} h(x,y)\|\Big\}$$

Proof.

$$\begin{aligned}
\left\|\nabla^2 h(x,y)\right\|_{\text{op}} &= \sup_{\|v\|^2+u^2=1} v^T \nabla_x^2 h(x,y) v + 2u(\nabla_{xy} h(x,y))^T v + u^2 \nabla_y^2 h(x,y) \\
&\overset{(i)}{\le} \|v\|^2 \left\|\nabla_x^2 h(x,y)\right\|_{\text{op}} + 2u\|\nabla_{xy} h(x,y)\|\|v\| + u^2 \nabla_y^2 h(x,y) \\
&\overset{(ii)}{\le} \|v\|^2 \left\|\nabla_x^2 h(x,y)\right\|_{\text{op}} + \Big(u^2 + \|v\|^2\Big)\|\nabla_{xy} h(x,y)\| + u^2 \nabla_y^2 h(x,y) \\
&= u^2\big(\nabla_y^2 h(x,y) + \|\nabla_{xy} h(x,y)\|\big) + \big(1-u^2\big)\Big(\left\|\nabla_x^2 h(x,y)\right\|_{\text{op}} + \|\nabla_{xy} h(x,y)\|\Big) \\
&\le \max\Big\{\nabla_y^2 h(x,y) + \|\nabla_{xy} h(x,y)\|, \left\|\nabla_x^2 h(x,y)\right\|_{\text{op}} + \|\nabla_{xy} h(x,y)\|\Big\}
\end{aligned}$$

with (i) following due to Hölder's inequality and (ii) follows from the inequality $2ab \le a^2 + b^2$. □