Learning with Primal-Dual Spectral Risk Measures: a Fast Incremental Algorithm

Ronak Mehta[1] **Vincent Roulet**[2] **Krishna Pillutla**[2] **Zaid Harchaoui**[1]

Abstract

We consider learning with a generalization of rank-weighted objectives known as *spectral risk measures (SRMs)*. SRMs, like other distributionally robust learning objectives, explicitly capture the worst-case performance over a set of possible deviations from the observed training distribution by introducing dual variables maximizing an adversarial objective. Even when the underlying (regularized) losses are smooth and strongly convex, direct stochastic gradient methods fail to converge to the SRM minimizer due in part to bias. In this work, we introduce a fast incremental optimization algorithm for SRMs that maintains a running estimate of the optimal dual variables by efficiently solving an approximation of the dual problem. Unlike related methods, our approach converges linearly for any smooth SRM and requires tuning a single hyperparameter: the (constant) primal learning rate. Empirically, our optimizer can achieve convergence within 2-3x fewer passes through the training set than recent baselines on distribution shift and fairness benchmarks.

1. Introduction

Consider a loss function $\ell(w, z)$ incurred by a model with parameters $w \in \mathbb{R}^d$ on data instance $z \in \mathcal{Z}$ (e.g. a feature-label pair). Given a collection of training examples $\boldsymbol{z} = (z_1, \ldots, z_n)$, the *empirical risk minimization (ERM)* problem aims to solve

$$\min_{w \in \mathbb{R}^d} \mathbb{E}_{Z \sim P_{\boldsymbol{z}}} [\ell(w, Z)] ,$$

where $P_{\boldsymbol{z}}$ is the empirical distribution of $\boldsymbol{z}$. Because a deployed model often observes data from a distribution other than that on which it was trained, we examine the risk-sensitive learning problem

$$\min_{w \in \mathbb{R}^d} \max_{Q \in \mathcal{U}(P_{\boldsymbol{z}})} \{\mathbb{E}_{Z \sim Q} [\ell(w, Z)] - \nu D(Q \| P_{\boldsymbol{z}})\} , \qquad (1)$$

in which $\mathcal{U}(P_{\boldsymbol{z}})$ is an *uncertainty/ambiguity set*, $\nu \geq 0$ is a hyperparameter, and $D(Q \| P_{\boldsymbol{z}})$ measures the deviation of $P_{\boldsymbol{z}}$ from Q. The objective (1) emulates a game in which for any model setting w, nature pays a price of ν per unit $D(Q \| P_{\boldsymbol{z}})$ to replace the true data-generating distribution with an unfavorably chosen Q. Accordingly, we refer to ν as the *shift cost*, the inner maximization as the *dual problem*, and the outer minimization as the *primal problem*. Canonical choices of D include the χ^2 divergence and Kullback Leibler (KL) divergence (Levy et al., 2020; Mehta et al., 2023) while the uncertainty set $\mathcal{U}(P_{\boldsymbol{z}})$ is often divergence-based (Dommel & Pichler, 2021; Levy et al., 2020; Ben-Tal et al., 2013), transport-based (Blanchet et al., 2019; Esfahani & Kuhn, 2018; Kuhn et al., 2019; Bui et al., 2022), or entropy-based (Pichler & Schlotter, 2020; Ahmadi-Javid, 2012).

In this work, we develop algorithms to optimize *spectral risk measures (SRMs)*, a broad collection of instantiations of (1) which includes the superquantile (a.k.a. conditional value-at-risk), extremile, and exponential spectral risk measure (ESRM) classes of objectives (Laguel et al., 2021; Acerbi & Tasche, 2002; Cotter & Dowd, 2006; Daouia et al., 2019). These (and other) distributionally robust objectives have recently enjoyed a surge in popularity in diverse contexts such as reinforcement learning and control (Liu et al., 2022a; Kallus et al., 2022; Liu et al., 2022b; Xu et al., 2023; Wang et al., 2023; Lotidis et al., 2023), continual learning (Wang et al., 2022), federated learning (Pillutla et al., 2023), dimension reduction (Vu et al., 2022), bandit problems (Yang et al., 2023), Bayesian learning (Tay et al., 2022; Inatsu et al., 2022), and structured prediction (Li et al., 2022). Despite this popularity, the options for stochastic/incremental algorithms (those that require only $O(1)$ number of calls to a function value/gradient oracle per-iterate) are limited for robust objectives. The key challenge is that the weight that the adversary assigns to data point z_i is given by solving the dual problem exactly when fixing the losses $\ell(w, z_1), \ldots, \ell(w, z_n)$ at a particular value

[1] Department of Statistics, University of Washington [2] Google Research. Correspondence to: Ronak Mehta <ronakdm@uw.edu>.

ICML 2023 Workshop on Duality for Modern Machine Learning, Honolulu, Hawaii, USA. Copyright 2023 by the author(s).

of the primal variables (as we discuss in Sec. 2). Computing these losses requires all n oracle calls, leading minibatch estimators to be biased (Mehta et al., 2023; Levy et al., 2020; Rényi, 1953). As such, full-batch approaches endure the computational cost and invoke $O(n)$ oracle calls per-iterate whereas stochastic gradient descent (SGD)-like approaches endure the bias, which along with their inherent variance leads to in poor convergence in practice (Mehta et al., 2023; Levy et al., 2020; Kawaguchi & Lu, 2020).

LSVRG (Mehta et al., 2023) is a recently proposed incremental algorithm that converges linearly for SRMs, given that the shift cost ν is large enough. This method periodically computes full batch gradients by computing the losses $\{\ell(w, z_i)\}_{i=1}^n$ and optimal dual variables at epoch checkpoints every N iterations. The dual variables are held fixed for the entirety of the epoch, so setting $N = O(n)$ amortizes the cost of n oracle calls in theory. In practice, the method requires tuning both a learning rate and an epoch length so the exact number of calls to an oracle per-iterate is n/N, a computational subtlety shared by its namesake SVRG in ERM (Johnson & Zhang, 2013). More pertinent to robust learning, however, is the fact that N also governs the *bias* accrued by using an approximate dual solution within each epoch, causing hyperparameter selection for LSVRG to more be precarious than for vanilla SVRG in ERM. Thus, the opportunity remains for a versatile stochastic optimizer for spectral risk-based objectives.

Contributions. In this paper, we propose SpecSAGA, an incremental algorithm for optimizing SRMs designed in the spirit of SAGA for ERM (Defazio et al., 2014). Our approach, unlike LSVRG, converges linearly for *any* shift cost on regularized convex losses. It only has one tunable hyperparameter (a learning rate), and makes $O(1)$ oracle calls regardless of hyperparameter choices. Experimentally, our method demonstrates equal or faster convergence than competitors on the training objective on nearly all objectives and datasets considered, and exhibits higher stability with respect to external metrics in fairness and distribution shifts. In Sec. 2, we describe our problem and the challenges it poses. In Sec. 3, we propose SpecSAGA and establish its convergence. In Sec. 4 we present extensions of SpecSAGA to tackle generic f-divergences. In Sec. 5, we demonstrate its performance on tabular, vision, and language benchmarks in various supervised learning settings.

Related Work. In the standard ERM setting for convex losses, incremental variance-reduced algorithms have demonstrated faster convergence than both full-batch and direct stochastic gradient methods in theory and practice (Gower et al., 2020). They have been studied in neural network learning as well (Defazio & Bottou, 2019). When the underlying losses are L-smooth and μ-strongly convex, variance-reduced methods such as SVRG (Johnson & Zhang, 2013) and SAGA (Defazio et al., 2014) reach an ε-suboptimal point in $O\left((n + L/\mu)\log(1/\varepsilon)\right)$ steps (whereas full batch gradient descent requires $O\left((nL/\mu)\log(1/\varepsilon)\right)$ iterations), decoupling the sample size n and the condition number L/μ. As the gradient estimates are often unbiased in ERM, this speed is principally due to variance reduction by the addition of *control variates* (Graham & Talay, 2013) to the update direction.

Distributionally robust objectives (Rahimian & Mehrotra, 2022; Michel et al., 2021; 2022; Haddadpour et al., 2022; Piratla et al., 2022) and spectral risk measures (SRMs) in particular (Fan et al., 2017; Kawaguchi & Lu, 2020; Khim et al., 2020; Maurer et al., 2021; Holland & Mehdi Haress, 2022) have been considered to explicitly account for "worst-case" performance on learning tasks. Note that this framework is distinct from adversarial learning (Qian et al., 2022). SRMs may also be called L-risks, based on classical L-estimators (linear combinations of order statistics) from the statistics literature (Shorack, 2017). We refer to an SRM as "smooth" when $\nu > 0$ (Mehta et al., 2023; Levy et al., 2020; Michel et al., 2021), which is a common relaxation considered by Lee et al. (2020); Michel et al. (2021), for example. Optimized certainty equivalent (OCE) and cumulative prospect theory (CPT) measures are motivated similarly but the latter is not characterized as explicitly capturing worst-case expected loss over a set of possible distributions (Leqi et al., 2019; Lee et al., 2020). In light of the convex-concave saddle-point problem interpretation of (1), various incremental "two-loop" methods were developed (Palaniappan & Bach, 2016; Yu et al., 2022; Chavdarova et al., 2019; Thekumparampil et al., 2019; Curi et al., 2020; Yang et al., 2020; Wang & Li, 2020; Yu et al., 2022): one for updating the parameters (primal variables) and one for reweighing the data (dual variables). Unlike these approaches, our algorithm operates in the primal space only, with a single, constant learning rate while enjoying linear convergence to the SRM minimizer.

2. Problem Setup

Our object of study is an instantiation of (1) with spectral risk measures (SRM). SRMs are defined by a collection of non-negative weights $\sigma_1 \leq \cdots \leq \sigma_n$ that sum to 1, called the *spectrum*. The corresponding uncertainty set $\mathcal{U}(P_{\boldsymbol{z}})$ is defined using the permutahedron $\mathcal{P}(\sigma)$ generated by σ,

$$\mathcal{P}(\sigma) = \text{ConvexHull}\left\{\left(\sigma_{\pi(1)}, \ldots, \sigma_{\pi(n)}\right) : \pi \in \Pi_n\right\}$$
$$\Pi_n = \{\text{permutations on } [n]\} .$$

The shifted distribution Q selects some $q = (q_1, \ldots, q_n) \in \mathcal{P}(\sigma)$, a convex combination of reorderings of σ, and assigns weight q_i to datum z_i. This weight can be at most σ_n,

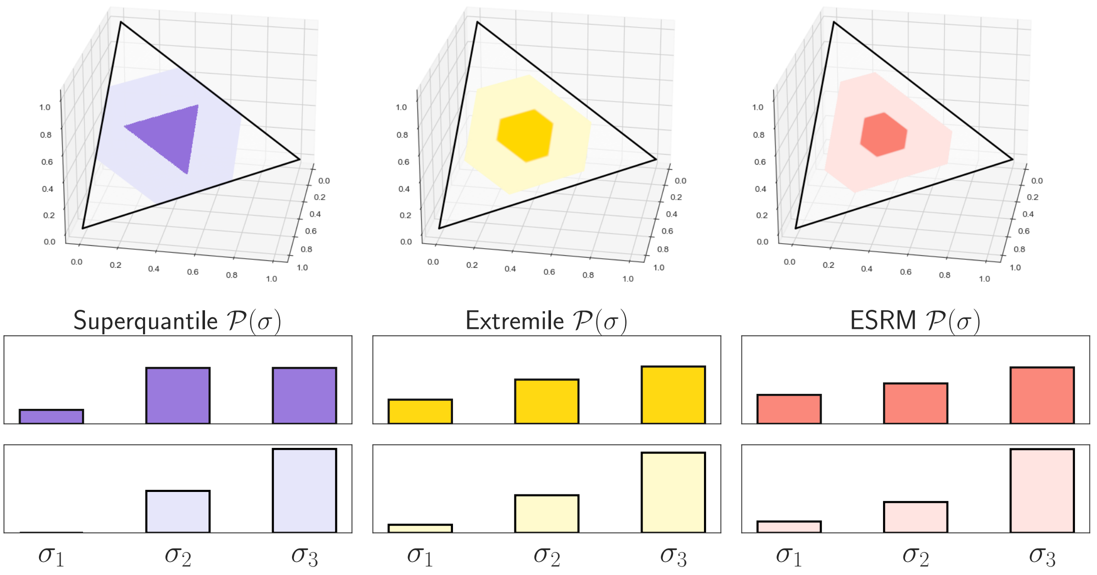

Figure 1: **Geometry of uncertainty sets**. Illustration of the permutahedra $\mathcal{P}(\sigma)$ within the three-dimensional probability simplex for the 0.25 and 0.5-superquantile (**left**), 1.5 and 2.5-extremile (**center**), and 1 and 3-ESRM (**right**). The size of $\mathcal{P}(\sigma)$ increases for more non-uniform spectra σ.

at least σ_1, or any value in between — see Fig. 1 for some examples of SRMs, their spectra, and their associated permutahedra $\mathcal{P}(\sigma)$. For $p \in [0, 1]$, the p-*superquantile*, also known as the conditional value-at-risk (CVaR) or the average top-k loss (Rockafellar & Royset, 2013; Kawaguchi & Lu, 2020; Laguel et al., 2021) requires that $k = np$ elements of σ be non-zero with equal probability and that the remaining $n - k$ are zero. Unlike the hard thresholding of superquantile, the b-*extremile* (Daouia et al., 2019) and γ-*exponential spectral risk measure* (Cotter & Dowd, 2006) define their spectra by $\sigma_i = (i/n)^b - ((i-1)/n)^b$ for $b \geq 1$ and $\sigma_i = \gamma e^{\gamma(i-1)}/(1 - e^{-\gamma})$ for $\gamma > 0$, respectively.

Recalling that $P_{\boldsymbol{z}}$ assigns weight $1/n$ to each of z_i, we use the empirical χ^2-divergence $D_{\chi^2}(Q\|P_{\boldsymbol{z}}) = \sum_{i=1}^n n(q_i - 1/n)^2 = n\left\|q - \mathbf{1}_n/n\right\|_2^2$ shift penalty. Denoting $\ell_i(w) := \ell(w, z_i)$ as the loss function on example i and concatenating $\ell(w) = (\ell_1(w), \dots, \ell_n(w))$, problem (1) reads

$$\begin{aligned} &\max_{Q \in \mathcal{U}(P_{\boldsymbol{z}})} \left\{\mathbb{E}_{Z\sim Q}\left[\ell(w, Z)\right] - \nu D(Q\|P_{\boldsymbol{z}})\right\} \\ &= \max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w) - \nu n\left\|q - \mathbf{1}_n/n\right\|_2^2 . \end{aligned}$$

Using regularization parameter $\mu > 0$, the spectral risk-based objective we wish to minimize is thus

$$\mathcal{L}_\sigma(w) := \max_{q \in \mathcal{P}(\sigma)} \Phi_\nu(\ell(w), q) + \frac{\mu}{2}\|w\|_2^2 , \tag{2}$$

$$\text{where } \Phi_\nu(l, q) := q^\top l - \nu n\left\|q - \mathbf{1}_n/n\right\|_2^2 . \tag{3}$$

Similar to the typical regularization parameter in classical statistical learning, ν controls the conditioning of the primal objective. Indeed, if each ℓ_i is G-Lipschitz and L-smooth, i.e. $w \mapsto \nabla\ell_i(w)$ is L-Lipschitz, w.r.t. $\|\cdot\|_2$, we have that $\mathcal{L}_\sigma$ is $L_\nu := (L + \mu + G^2/\nu)$-smooth and μ-strongly convex. Then, the condition number of $\mathcal{L}_\sigma$ is $L_\nu/\mu = \kappa + G^2/(\mu\nu)$ with $\kappa = 1 + L/\mu$. Thus, a larger shift cost makes (3) better conditioned.

Optimizing Smooth Spectral Risks. The gradient of (3) is given by Danskin's theorem (Bertsekas, 1997) as

$$\nabla\mathcal{L}_\sigma(w) = \sum_{i=1}^n q_i^{\text{opt}}(\ell(w))\nabla\ell_i(w) + \mu w , \tag{4}$$

$$\text{where } q^{\text{opt}}(l) := \mathop{\arg\max}_{q \in \mathcal{P}(\sigma)} \Phi_\nu(l, q), \tag{5}$$

admitting gradient descent as a natural baseline for optimization. The gradient can be seen as a variation of the standard gradient of ERM with an importance weight $q_i^{\text{opt}}(\ell(w))$ assigned to the loss $\ell_i(w)$. The strong concavity of the shift penalty not only ensures the uniqueness of the optimal weights but also their Lipschitz-continuity: $\|q^{\text{opt}}(l) - q^{\text{opt}}(l')\|_2 \leq \|l - l'\|_2/(2n\nu)$ (Nesterov, 2005).

The maximization over q can be expressed by standard con-

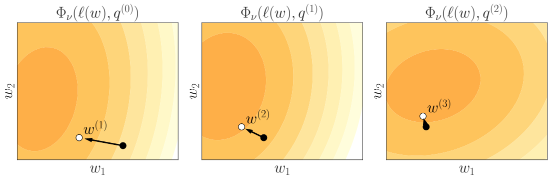

Figure 2: **Game-theoretic interpretation**. At each iteration of the SpecSAGA algorithm, the learner is provided a stochastic gradient of a reweighted objective $\Phi_\nu(w, q^{(t)})$, whose landscape depends on the current weights. For any sequence of "moves" made by the learner $w^{(0)}, w^{(1)}, \ldots$, the losses are reweighed using $q^{(1)}, q^{(2)}, \ldots$ computed by exactly solving an approximation to the dual problem.

vex duality arguments as, see Appx. C,

$$\max_{q \in \mathcal{P}(\sigma)} q^\top l - n\nu \|q - \mathbf{1}_n/n\|_2^2$$
$$= \min_{z \in \mathbb{R}^n : z_{\pi_1} \leq \ldots \leq z_{\pi_n}} \sum_{i=1}^n \sigma_i z_{\pi_i} + \omega^\star(l_i - z_i),$$

where $\omega^\star$ is the convex conjugate of $n\nu\|\cdot - \mathbf{1}_n/n\|_2^2$ and π a permutation on $[n]$ that orders the losses in non-decreasing order: $l_{\pi_1} \leq \ldots \leq l_{\pi_n}$. This is an isotonic regression problem that can be solved exactly by the Pool Adjacent Violators algorithm (Best et al., 2000). It runs in $O(n)$ time when the losses are sorted; this in turn requires $O(n \log n)$ time. The optimal weights $q^{\text{opt}}(l)$ can be obtained from the optimal z^* as $q_i^{\text{opt}}(l) = \nabla\omega^\star(l_i - z_i^*)$ to compute the gradient of the overall objective. Thus, the main bottleneck is computing the n losses at the current iterate w, each one costing $O(d)$ operations.

It is not straightforward to alleviate this call to n losses evaluations as minibatch gradient estimators with $O(1)$-samples are biased estimators of the batch gradient for non-uniform σ. Given a minibatch $S \subset [n]$ with $|S| = m$, the natural stochastic gradient estimator is $\nabla \mathcal{L}_{\hat{\sigma}}(w; S) := \nabla \left[\max_{q \in \mathcal{P}(\hat{\sigma})} \Phi_\nu\left(\{\ell_i(w)\}_{i \in S}, q\right)\right] + \mu w$, where we use the minibatch spectrum $\hat{\sigma}_j = \sigma_{(j-1)n/m+1} + \cdots + \sigma_{jn/m}$ for $j \in [m]$ and considered for simplicity that m divides n. A stochastic gradient algorithm with this estimator *does not minimize* (3); it actually optimizes $\mathbb{E}_S\left[\mathcal{L}_{\hat{\sigma}}(w; S)\right]$, which differs from the true objective $\mathcal{L}_\sigma(w)$ by $O(1/\sqrt{m})$ (Levy et al., 2020; Mehta et al., 2023; Kawaguchi & Lu, 2020). Thus, vanilla stochastic gradient descent cannot be employed to minimize the original objective (3).

3. The SpecSAGA Algorithm

We present the SpecSAGA algorithm to optimize (3), building up the full algorithm in Algorithm 1.

Our starting point is the gradient formula (4). It is expensive because we need to compute the loss vector $\ell(w)$ at a $O(nd)$ cost of retrieving the optimal weights $q^{\text{opt}}(\ell(w))$. However, a reasonable approximation l of the losses $\ell(w)$ can be used to approximate the weights with $q^{\text{opt}}(l)$ in $O(n)$ time. SpecSAGA presented in Algorithm 1 leverages this idea. Equipped with the approximate weights $q = q^{\text{opt}}(l)$ of a running estimate l of the loss vector $\ell(w)$, we consider a stochastic gradient $nq_i(\nabla\ell_i(w) + \mu w)$ of the surrogate objective $\sum_{i=1}^n q_i \ell_i(w) + \mu\|w\|_2^2/2$ using a sample $i \sim \text{Unif}[n]$. Optimizing such a surrogate objective at each step amounts to letting an adversary shift the landscape of the objective at each step as illustrated in Fig. 2.

This stochastic gradient estimator has better bias properties than the one from Sec. 2. Due to the strong concavity of the shift penalty, the approximation of the losses translates directly in terms of approximation of the weights: $\|q - q^{\text{opt}}(\ell(w))\|_2 \leq \|l - \ell(w)\|_2/(2n\nu)$ for q, l computed in Line 8 of Algorithm 1. In particular, if ν is large, the weights remain stable even if the losses can vary. The loss estimate l is driven by the current iterate based on its update in Line 8. The updated loss estimate l^+ satisfies $\mathbb{E}_{i_t}\left[l^+\right] = \frac{1}{n}\ell(w) + (1 - \frac{1}{n})l$. For Lipschitz losses, by appropriately choosing the step size, a faithful approximation of the losses around the current iterate can be accumulated in the loss vector to ensure overall convergence. As w converges to $w^\star$, we expect $l \to \ell(w^\star)$ and $q \to q^{\text{opt}}(\ell(w^\star))$, i.e., our gradient estimator exhibits vanishing bias.

This gradient estimator can have a large variance due to sampling of i even when $w \approx w^\star$. We tackle this with a zero-mean *control variate* of the form $n\rho_{i_t} g_{i_t} - \bar{g}$ using a table $g_1, \ldots, g_n \in \mathbb{R}^d$ of gradients and its weighted mean $\bar{g} = \sum_{i=1}^n \rho_i g_i$. These weights $\rho \in \mathbb{R}_+^n$ are meant as an unnormalized proxy of $q(\ell(w))$. As w converges to $w^\star$, we expect $g_i \to \nabla\ell_i(w^\star) + \mu w^\star$ and $\rho \to q^{\text{opt}}(\ell(w^\star))$, so the control variate $n\rho_i g_i - \bar{g}$ correlates strongly with the estimator $nq_{i_t}(\nabla\ell_{i_t}(w) + \mu w)$. This allows the corrected update in Line 7 to have vanishing variance while retaining the same bias reduction. Crucially, since ρ is not normalized to sum to 1 like q, the update to $\bar{g}$ in Line 9 takes $O(d)$ time, rather than $O(nd)$ if we were to compute $\sum_{i=1}^n q_i g_i$.

The full algorithm is given in Algorithm 1. Unlike LSVRG, there is no epoch length hyperparameter, and the approximation q is updated every iteration. Note that the exposition in Algorithm 1 is intended for conceptual clarity and can be further optimized, as we describe next.

Computational Aspects. The weight update in Line 8 is solved exactly by (i) sorting the vector of losses in $O(n \log n)$, (ii) plugging the sorted loss table l into the Pool Adjacent Violators (PAV) algorithm running in $O(n)$ time, as discussed in Sec. 2. Because only one element of l changes every iterate, we may simply bubble sort l starting from the index that was changed. While in the worst case,

Algorithm 1 SpecSAGA

```
    Inputs: Initial point $w_0$, spectrum $\sigma$, stepsize $\eta > 0$, number of iterations $T$, regularization parameter $\mu > 0$, shift cost $\nu > 0$.
 1: Initialize $w \leftarrow w_0$, $l_i \leftarrow \ell_i(w_0)$, and $g_i \leftarrow \nabla \ell_i(w_0) + \mu w_0$ for $i = 1, \ldots, n$.
 2: Compute $q \leftarrow q^{\text{opt}}(l)$
 3: Set $\rho \leftarrow q$ and $\bar{g} \leftarrow \sum_{i=1}^n \rho_i g_i \in \mathbb{R}^d$
 4: for each iterate do
 5:     Sample $i \sim \mathrm{Unif}[n]$
 6:     $v \leftarrow n q_i (\nabla \ell_i(w) + \mu w) - n \rho_i g_i + \bar{g}$
 7:     $w \leftarrow w - \eta v$                          ▷ Parameter Update
 8:     $l_i \leftarrow \ell_i(w)$ and $q \leftarrow q^{\text{opt}}(l)$    ▷ Adversary Update
 9:     $\bar{g} \leftarrow \bar{g} - \rho_i g_i + q_i (\nabla \ell_i(w) + \mu w)$
10:     $g_i \leftarrow \nabla \ell_i(w) + \mu w$
11:     $\rho_i \leftarrow q_i$.                    ▷ Control Variate Updates
    Output: Final point $w$.
```

this cost is $O(n)$, it is exactly $O(k_t)$ where k_t is the number of swaps needed to sort l from iterate t to $t+1$. We find in experiments that the sorted order of l stabilizes quickly.

The storage of the gradient table g requires $O(nd)$ space in general, but it can be reduced to $O(n)$ for generalized linear models and nonlinear additive models. For losses of the form $\ell_i(w) = h(x_i^\top w, y_i)$, for a differentiable loss h and scalar output y_i, we have $\nabla \ell_i(w) = x_i\, h'(x_i^\top w, y_i)$. We only need to store the scalar $h'(x_i^\top w, y_i)$, so SpecSAGA requires $O(n + d)$ memory.

In terms of time complexity, Lines 7 and 9 require $O(d)$ operations and Line 8 requires at most $O(n)$ operations, so that in total the iteration complexity is $O(n + d)$. In comparison, a full batch gradient descent requires $O(nd)$ operations so SpecSAGA decouples the cost of computing the weights and accessing the losses and gradients.

Convergence Analysis. We assume throughout that each ℓ_i convex, G-Lipschitz, and L-smooth. The convergence guarantees depend on the condition numbers $\kappa = 1 + L/\mu$ of the individual losses, as well as a measure $\kappa_\sigma = n\sigma_n$ of the skewness of the spectrum.

Theorem 1. *By decoupling the sampling of the losses and the gradients as described in Appx. D.3, SpecSAGA with a small enough step size is guaranteed to converge linearly for all $\nu > 0$. If, in addition, the shift cost is $\nu \geq \Omega(G^2/\mu)$, then the sequence of iterates $(w^{(t)})$ generated by SpecSAGA with a single sampling and learning rate $\eta = (6(L + 2\mu)\kappa_\sigma)^{-1}$ converges linearly at a rate $\tau = \max\left\{2n, 12\kappa_\sigma(\kappa+1)^2/\kappa\right\}$, i.e.,*

$$\mathbb{E}\|w^{(t)} - w^\star\|_2^2 \leq (1 + 2n + 2n^2)\exp(-t/\tau)\|w^{(0)} - w^\star\|_2^2 .$$

The number of iterations t required by SpecSAGA to achieve $\mathbb{E}\|w^{(t)} - w^\star\|_2^2 \leq \varepsilon$ (provided that ν is large enough) is $t = O((n + \kappa\kappa_\sigma)\log(1/\varepsilon))$. This exactly matches the rate of LSVRG. Unlike LSVRG, SpecSAGA is guaranteed to converge linearly for any shift cost — this requires two samples per update for technical reasons. Compared to primal-dual SAGA, our algorithm requires only one learning rate, streamlining its implementation.

4. Towards Broader Shifts: f-Divergences and Hidden Smoothness

In this section, we adapt SpecSAGA to apply to other shift penalties based on f-divergences and reduce the requirement on the shift cost ν.

Handling General f-Divergence Shift Penalties. A shift penalty dampens the adversary's power in (3), ensuring they cannot move the unfavorable distribution q too far from the uniform distribution $\mathbf{1}_n/n$. While we focus primarily on the χ^2-divergence, other divergences such as Kullback-Leibler (KL) and squared Hellinger distance are frequently employed as measures of discrepancy between distributions in this context. They have been utilized as convex surrogates to improve statistical generalization (Lam, 2016; Namkoong & Duchi, 2017; Lam, 2019; Duchi et al., 2021), promote fairness (Hashimoto et al., 2018; Williamson & Menon, 2019), and model adversarial games (Bauso et al., 2017; Nowozin et al., 2016).

SpecSAGA can be extended to handle general f-divergences by replacing the shift penalty in Line 8 of Alg. 1. Given a convex function $f : \mathbb{R}_+ \to \mathbb{R}_+$, recall that the associated f-divergence is defined as $D_f(s\|q) = \sum_{i=1}^n q_i f(s_i/q_i)$. The dual maximization $q^*(l) = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l - \nu D_f(q\|\mathbf{1}_n/n)$ can be efficiently solved for many f-divergences via their dual as an isotonic optimization problem. SpecSAGA thus has the same computational complexity as in Sec. 3.

Similarly, we can extend Thm. 1 for α-strongly convex f-divergences. This includes the KL and Jeffreys divergences with $\alpha = 1$ w.r.t. $\|\cdot\|_1$. In general, other common examples such as the Jensen-Shannon and Le Cam divergences are strongly convex with α dependent on κ_σ (Melbourne, 2020).

Theorem 2. *Suppose the f-divergence $q \mapsto D_f(q\,\|\mathbf{1}_n/n)$ is α-strongly convex w.r.t. $\|\cdot\|_p$ for some $p \in [1, 2]$. Then, the f-penalized version of SpecSAGA satisfies the guarantee of Thm. 1 as long as $\nu \geq \Omega\left(\frac{\sqrt{n}G}{\alpha\mu}(\|\nabla \ell(w^\star)\|_{2,p} + G\sqrt{n/\kappa\kappa_\sigma})\right)$, where $\nabla \ell(w^\star) \in \mathbb{R}^{n \times d}$ is the Jacobian of ℓ at $w^\star$.*

The condition on ν depends on the geometry of the shift penalty. For the χ^2-divergence, we have that

$\|\nabla \ell(w^\star)\|_{2,p} \leq \sqrt{n}G$, so this condition is implied by that of Thm. 1. For the KL divergence, we have that $\|\nabla \ell(w^\star)\|_{1,p} \leq G$, so the requirement on ν can be up to $\sqrt{n}$ better when $n < \kappa\kappa_\sigma$.

SpecSAGA with No Shift Penalty. Note that SpecSAGA requires $\nu > 0$ as a condition for its convergence guarantee, particularly due to the smoothness it provides the objective. Historically, however, SRMs such as the conditional value-at-risk have been employed as coherent risk measures for distributions of losses (Acerbi & Tasche, 2002) with no shift penalty, i.e. $\nu = 0$. For a vector $(l_1, \ldots, l_n) \in \mathbb{R}^n$ (corresponding to a discrete empirical distribution of losses), the SRM aggregates the rank statistics $l_{(1)} \leq \cdots \leq l_{(n)}$ with weights supplied by σ as $\max_{q \in \mathcal{P}(\sigma)} q^\top l = \sum_{i=1}^n \sigma_i l_{(i)}$. This form can be identified as an L-estimator in statistics (Shorack, 2017), and for continuous distributions, the weighted sum of order statistics becomes a weighted integral of the quantile function. If these losses are separated at the optimum, we may achieve linear convergence with SpecSAGA even with $\nu = 0$. This behavior can be explained as the algorithm leveraging hidden smoothness in the objective (3), as is differentiable at points satisfying $\ell_{(1)}(w) < \cdots < \ell_{(n)}(w)$, where $\ell_{(i)}(w)$ denotes the i-th smallest loss at w. We assume that $\ell_1, \ldots, \ell_n$ are convex and that $\mu > 0$.

Proposition 3. *Let $w_\nu^\star$ be the unique minimizer of (3) with shift cost $\nu \geq 0$. Assume that the values $\ell_1(w_0^\star), \ldots, \ell_n(w_0^\star)$ are all distinct. Then, there exists a constant $\nu_0 > 0$ such that $w_0^\star = w_\nu^\star$ exactly for all $\nu \leq \nu_0$. Thus, running decoupled SpecSAGA (Appx. D.3) converges to the minimizer $w_0^\star$.*

In particular, ν_0 is chosen so that $\nu_0 (\sigma_{i+1} - \sigma_i) < \ell_{(i+1)}(w_0^\star) - \ell_{(i)}(w_0^\star)$ for each i, or as the multiplicative factor that relates gaps in the spectrum to the gaps in the loss at optimality (see Appx. B).

Smoother Gradient Oracles. SpecSAGA can leverage more information about the losses using the Moreau envelope of each loss ℓ_i and its gradient (Bauschke et al., 2011; Rockafellar, 1976). Specifically, we consider oracles returning $\nabla \operatorname{env}(\ell_i)(w)$ where $\operatorname{env}(\ell_i) = \inf_{v \in \mathbb{R}^d} \ell_i(v) + \|w - v\|_2^2$; this can be expressed in terms of the proximal operators of the losses (Bauschke et al., 2011). Such an approach has been considered for ERM by (Defazio, 2016) to accelerate the SAGA algorithm. These oracles can easily be accessed either in closed form or by efficient subroutines in common machine learning settings (Defazio, 2016; Frerix et al., 2018; Roulet & Harchaoui, 2022), and we can easily adapt Alg. 1 to leverage such oracles. The resulting algorithm enjoys a linear convergence guarantee similar to Thm. 1 with a less restrictive condition on the shift cost ν, while still providing competitive performance in practice. We refer to Appx. F for details.

5. Experiments

In this section, we compare SpecSAGA against competitors in a variety of tasks. While we focus attention on its performance as an optimizer with respect to its training objective, we also highlight metrics of interest on the test set in fairness and distribution shift benchmarks as they constitute common use-cases for distributionally robust objectives.

Setting, Baselines, and Evaluation. We consider supervised learning tasks for which each data point $z_i = (x_i, y_i)$ is an input-label pair. Losses are of the form $\ell_i(w) := h(y_i, w^\top \phi(x_i))$, where $\phi(\cdot)$ is a fixed feature embedding function and h measures the error between the predicted and true labels. We use 3 choices of the spectrum σ: 0.5-superquantile, 2-extremile, and 1-ESRM.

We compare against four baselines: minibatch stochastic gradient descent (SGD), stochastic regularized dual averaging (SRDA) (Xiao, 2009), Saddle-SAGA (Palaniappan & Bach, 2016), and LSVRG (Mehta et al., 2023). For SGD and SRDA, we use a batch size of 64, and for LSVRG we use an epoch length of n. In the case of Saddle-SAGA, we find that allowing different learning rates for the primal and dual variables improves theoretically and experimentally (Appx. E) and compare against the improved heuristic (setting the dual stepsize as $10n$ times smaller than the primal stepsize) in our experiments. We plot

$$\text{Suboptimality}(w) = \frac{\mathcal{L}_\sigma(w) - \mathcal{L}_\sigma(w^\star)}{\mathcal{L}_\sigma(w_0) - \mathcal{L}_\sigma(w^\star)}, \qquad (6)$$

where $w^\star$ is approximated by running LBFGS (Nocedal & Wright, 1999) on the objective until convergence. The x-axis displays the number of calls to any first-order oracle $w \mapsto (\ell_i(w), \nabla \ell_i(w))$ divided by n, i.e. the number of passes through the training set. We fix the shift cost $\nu = 1$ and regularization parameter $\mu = 1/n$. Further details of the setup and additional results are given in Appxs H and I respectively.

5.1. Tabular Least-Squares Regression

We consider five tabular regression benchmarks under square loss. The datasets used are `yacht` ($n = 244$) (Tsanas & Xifara, 2012), `energy` ($n = 614$) (Baressi Segota et al., 2020), `concrete` ($n = 824$) (Yeh, 2006), `kin8nm` ($n = 6553$) (Akujuobi & Zhang, 2017), and `power` ($n = 7654$) (Tüfekci, 2014). The training curves for each optimizer are shown in Fig. 3.

Results. Across datasets and objectives, we find that SpecSAGA exhibits linear convergence at a rate no worse than SaddleSAGA and LSVRG, but one that is often much better. For example, SpecSAGA converges to precision 10^{-8} for the superquantile on `concrete` and the extremile on `power` within half the number of passes that LSVRG takes

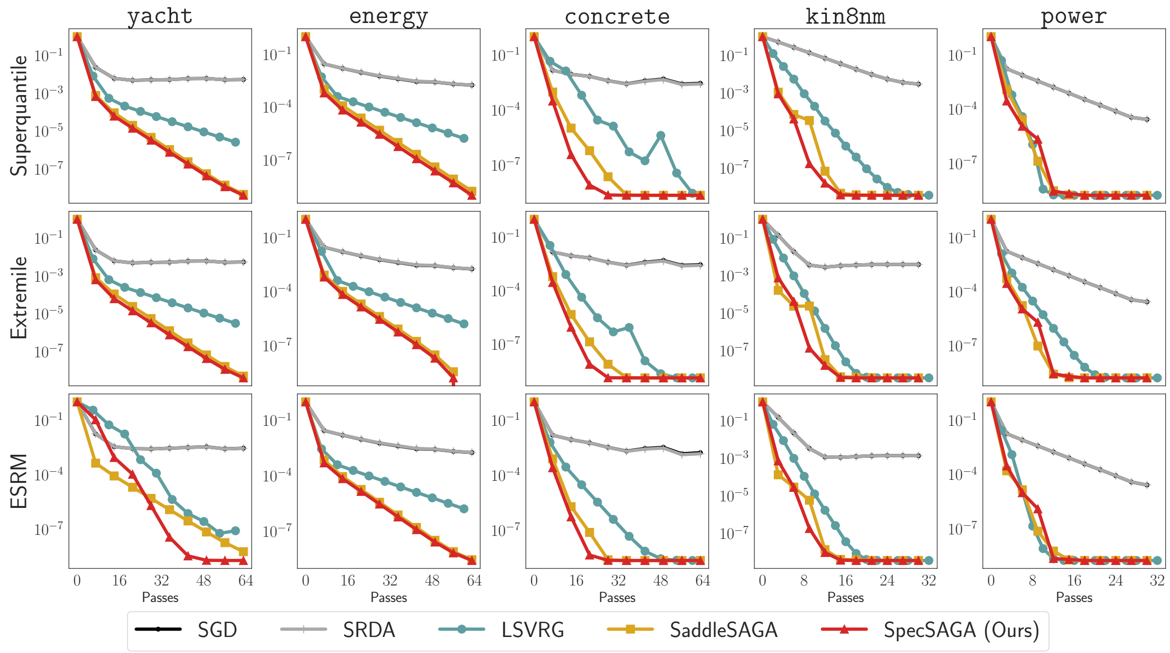

Figure 3: **Regression benchmarks**. The y-axis measures the suboptimality as given by (6), while the x-axis measures the number of calls to the function value/gradient oracle divided by n. Rows indicate different spectral risk objectives and columns indicate datasets.

for the same suboptimality value. Similarly, for the ESRM on yacht, SaddleSAGA requires 64 epochs to reach the same precision as SpecSAGA at 40 epochs. The direct stochastic methods, SGD and SRDA, have incurable bias and variance and fail to converge for any learning rate.

5.2. Fair Classification and Regression

Inspired by (Williamson & Menon, 2019), we explore the relationship between robust learning and group fairness on 2 common tabular benchmarks. **Diabetes 130-Hospitals** (diabetes) is a binary classification task of predicting readmission for diabetes patients based on 10 years worth of clinical data from 130 US hospitals (Rizvi et al., 2014). **Adult Census** (acsincome) is a regression task of predicting income of US adults given features compiled from the American Community Survey (Ding et al., 2021).

Evaluation. We evaluate fairness with the *statistical parity score*, which compares predictive distributions of a model given different values of a particular protected attribute (Agarwal et al., 2018; 2019). Letting $Z = (X, Y, A)$ denote a random (input, label, metadata attribute) triplet, a model g is said to satisfy statistical parity (SP) if the conditional distribution of $g(X)$ over predictions given $A = a$ is equal for any value a. Intuitively, statistical parity scores measure the maximum deviation between these distributions for any over a, so values close to zero indicate SP-fairness. In diabetes, we use gender as the protected attribute A, whereas in acsincome we use race as the protected attribute. Note that the protected attributes are not supplied to the models. The results are given in Fig. 4.

Results. Firstly, we note that SpecSAGA converges rapidly on both datasets while LSVRG fails to converge on diabetes and SaddleSAGA fails to converge on acsincome. Secondly, LSVRG does not stabilize with respect to classification SP, showing a mean/std SP score of $1.38 \pm 0.25\%$ within the final ten passes on the diabetes superquantile, whereas SpecSAGA gives $0.82 \pm 0.00\%$, i.e., a 40% relative improvement with greater stability. While SaddleSAGA does stabilize in SP on diabetes, it fails to qualitatively decrease at all on the acsincome. Interestingly, while suboptimality and SP-fairness are correlated for SpecSAGA, SGD (reaching only 10^{-1} suboptimality with respect to the superquantile objectives on acsincome) achieves a lower fairness score. Again, across both suboptimality and fairness, SpecSAGA is either the best or close to the best.

5.3. Image & Text Classification with Distribution Shift

We consider two tasks from the WILDS distribution shift benchmark (Koh et al., 2021). The **Amazon Reviews** (amazon) task (Ni et al., 2019) consists of classifying text reviews of products to a rating of 1-5, with disjoint train and test reviewers. The **iWildCam** (iwildcam) image classification challenge (Beery et al., 2020) contains labeled im-

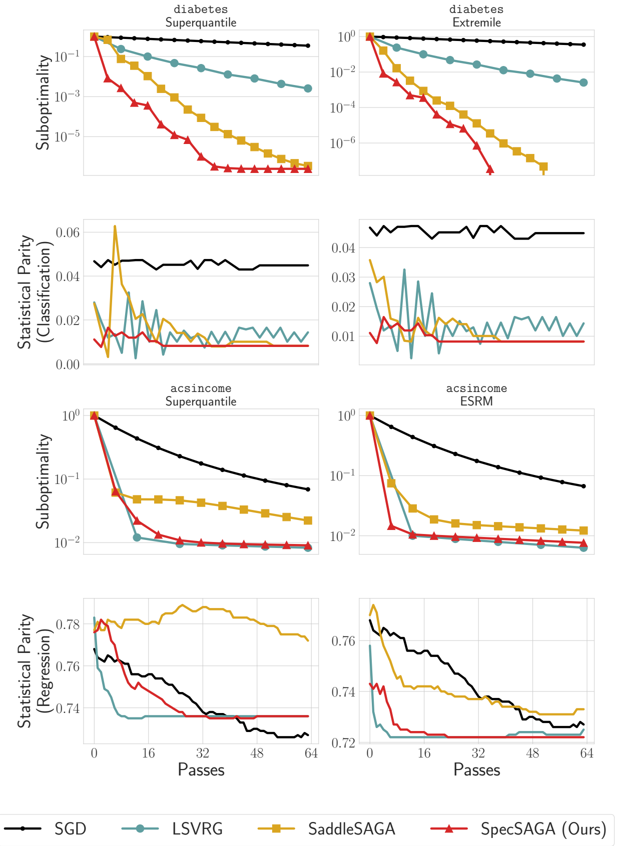

Figure 4: **Fairness benchmarks**. **First row:** Training curves for optimizers on the superquantile and extremile for `diabetes`. **Second row:** Statistical parity scores for the two classification objectives on `diabetes`. **Third row:** Training curves for optimizers on the superquantile and extremile for `acsincome`. **Bottom row:** Statistical parity scores for regression objectives on `acsincome`. Values closer to zero indicate better SP-fairness.

ages of animals, flora, and backgrounds from camera traps placed in wilderness sites. Shifts are caused due to changes in camera angles, locations, lighting, etc. We use a sample of $n = 10000$ and $n = 20000$ examples respectively. For both datasets, we train a *linear probe classifier*, i.e., a linear model over a frozen deep feature representation. For `amazon`, we use a pretrained BERT model (Devlin et al., 2019) fine-tuned on a held-out subset of the Amazon Reviews training set for 2 epochs. For `iwildcam`, we use a ResNet50 pretrained on ImageNet (without fine-tuning).

Evaluation. Apart from the training suboptimality, we evaluate the spectral risk objectives on their robustness to subpopulation shifts. We define each subpopulation group based on the true label. For `amazon`, we use the *worst group misclassification error* on the test set as a robustness measure (Sagawa et al., 2020). For `iwildcam`, we use the *median group error* owning to its larger number of classes.

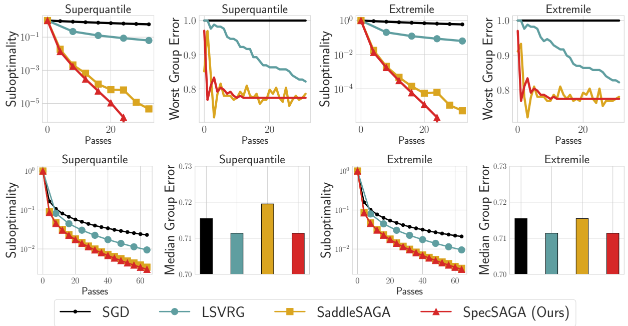

Figure 5: **Distribution shift results**. **Top row:** Training curves and worst group misclassification error on `amazon` test. **Bottom row:** Training curves and median group misclassification error on the `iwildcam` test set. Smaller values indicate better performance for all metrics.

Results. For both `amazon` and `iwildcam`, SpecSAGA and SaddleSAGA (with our heuristic) outperform LSVRG in training suboptimality. We hypothesize that this phenomenon is likely due to checkpoints of LSVRG getting stale over the n-length epochs for these datasets with large n (leading to a slow reduction of bias). In contrast, SpecSAGA and SaddleSAGA avoid this issue by dynamically updating the running estimates of the importance weights.

For the worst group error for `amazon`, SpecSAGA and SaddleSAGA outperform LSVRG. SpecSAGA has a mean/std worst group error of $77.38 \pm 0.00\%$ over the last ten passes on the extremile, whereas SaddleSAGA has a slightly worse $77.53 \pm 1.57\%$. Interestingly, on `iwildcam`, LSVRG and SpecSAGA demonstrate stronger generalization performance, nearly 1pp better, than SaddleSAGA in terms of median group misclassification rate. In summary, across both tasks and all objectives, SpecSAGA demonstrates the best or close to the best performance on both benchmarks.

6. Discussion

In this paper, we introduced SpecSAGA, an algorithm for optimizing smooth spectral risk measures with a linear convergence guarantee. The algorithm demonstrates rapid linear convergence on benchmark examples and has the practical benefits of converging for any shift cost and having a single hyperparameter. While we primarily used the χ^2-shift penalty throughout this work, we derive methods for other penalties derived from taking f-divergences between the shifted distribution Q and the original one P_z. Promising avenues for future work include extensions to the non-convex setting by considering the regular subdifferential and handling missing data problems.

References

Acerbi, C. and Tasche, D. On the Coherence of Expected Shortfall. *Journal of Banking & Finance*, 26, 2002.

Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A Reductions Approach to Fair Classification. In *ICML*, volume 80. PMLR, 2018.

Agarwal, A., Dudik, M., and Wu, Z. S. Fair Regression: Quantitative Definitions and Reduction-Based Algorithms. In *ICML*, volume 97. PMLR, 2019.

Ahmadi-Javid, A. Entropic Value-at-Risk: A New Coherent Risk Measure. *Journal of Optimization Theory and Applications*, 155, 2012.

Akujuobi, U. and Zhang, X. Delve: A Dataset-Driven Scholarly Search and Analysis System. *SIGKDD Explor. Newsl.*, 19, 2017.

Baressi Segota, S., Andelic, N., Kudlacek, J., and Cep, R. Artificial Neural Network for Predicting Values of Residuary Resistance per Unit Weight of Displacement. *Journal of Maritime & Transportation Science*, 57, 2020.

Bauschke, H. H., Combettes, P. L., et al. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, volume 408. Springer, 2011.

Bauso, D., Gao, J., and Tembine, H. Distributionally Robust Games: F-Divergence and Learning. In *Proceedings of the 11th EAI International Conference on Performance Evaluation Methodologies and Tools*. Association for Computing Machinery, 2017.

Beery, S., Cole, E., and Gjoka, A. The iWildCam 2020 Competition Dataset. *arXiv preprint arXiv:2004.10340*, 2020.

Ben-Tal, A., den Hertog, D., Waegenaere, A. D., Melenberg, B., and Rennen, G. Robust Solutions of Optimization Problems Affected by Uncertain Probabilities. *Management Science*, 59, 2013.

Bertsekas, D. P. Nonlinear Programming. *Journal of the Operational Research Society*, 48, 1997.

Best, M. J., Chakravarti, N., and Ubhaya, V. A. Minimizing Separable Convex Functions Subject to Simple Chain Constraints. *SIAM Journal on Optimization*, 10, 2000.

Blanchet, J., Kang, Y., and Murthy, K. Robust Wasserstein Profile Inference and Applications to Machine Learning. *Journal of Applied Probability*, 56, 2019.

Blondel, M., Teboul, O., Berthet, Q., and Djolonga, J. Fast Differentiable Sorting and Ranking. In *ICML*. PMLR, 2020.

Bui, A. T., Le, T., Tran, Q. H., Zhao, H., and Phung, D. A Unified Wasserstein Distributional Robustness Framework for Adversarial Training. In *ICLR*, 2022.

Chavdarova, T., Gidel, G., Fleuret, F., and Lacoste-Julien, S. Reducing Noise in GAN Training with Variance Reduced Rxtragradient. *NeurIPS*, 32, 2019.

Cotter, J. and Dowd, K. Extreme Spectral Risk Measures: an Application to Futures Clearinghouse Margin Requirements. *Journal of Banking & Finance*, 30, 2006.

Curi, S., Levy, K. Y., Jegelka, S., and Krause, A. Adaptive Sampling for Stochastic Risk-Averse Learning. In *NeurIPS*, volume 33. Curran Associates, Inc., 2020.

Daouia, A., Gijbels, I., and Stupfler, G. Extremiles: A New Perspective on Asymmetric Least Squares. *Journal of the American Statistical Association*, 114, 2019.

Defazio, A. A Simple Practical Accelerated Method for Finite Sums. In *NeurIPS*, volume 29, 2016.

Defazio, A. and Bottou, L. On the Ineffectiveness of Variance Reduced Optimization for Deep Learning. In *NeurIPS*, volume 32. Curran Associates, Inc., 2019.

Defazio, A., Bach, F., and Lacoste-Julien, S. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. *NeurIPS*, 27, 2014.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-scale Hierarchical Image Database. In *CVPR*, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 2019.

Ding, F., Hardt, M., Miller, J., and Schmidt, L. Retiring Adult: New Datasets for Fair Machine Learning. In *NeurIPS*, volume 34. Curran Associates, Inc., 2021.

Dommel, P. and Pichler, A. Convex Risk Measures Based on Divergence. *Pure and Applied Functional Analysis*, 6, 2021.

Duchi, J. C., Glynn, P. W., and Namkoong, H. Statistics of Robust Optimization: A Generalized Empirical Likelihood Approach. *Mathematics of Operations Research*, 46, 2021.

Esfahani, P. M. and Kuhn, D. Data-driven Distributionally Robust Optimization using the Wasserstein Metric: Performance Guarantees and Tractable Reformulations. *Mathematical Programming*, 171, 2018.

Fan, Y., Lyu, S., Ying, Y., and Hu, B. Learning with Average Top-k Loss. In *NeurIPS*, volume 30, 2017.

Frerix, T., Möllenhoff, T., Möller, M., and Cremers, D. Proximal Backpropagation. In *ICLR*, 2018.

Gower, R. M., Schmidt, M., Bach, F., and Richtárik, P. Variance-Reduced Methods for Machine Learning. *Proceedings of the IEEE*, 108, 2020.

Graham, C. and Talay, D. *Stochastic Simulation and Monte Carlo Methods*. Springer Berlin, Heidelberg, 2013.

Haddadpour, F., Kamani, M. M., Mahdavi, M., and amin karbasi. Learning Distributionally Robust Models at Scale via Composite Optimization. In *ICLR*, 2022.

Hashimoto, T., Srivastava, M., Namkoong, H., and Liang, P. Fairness without Demographics in Repeated Loss Minimization. In *ICML*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.

Hiriart-Urruty, J.-B. and Lemaréchal, C. *Fundamentals of Convex Analysis*. Springer Science & Business Media, 2004.

Holland, M. J. and Mehdi Haress, E. Spectral Risk-based Learning Using Unbounded Losses. In *AISTATS*, volume 151, 2022.

Inatsu, Y., Takeno, S., Karasuyama, M., and Takeuchi, I. Bayesian Optimization for Distributionally Robust Chance-constrained Problem. In *ICML*, volume 162. PMLR, 2022.

Johnson, R. and Zhang, T. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *NeurIPS*, volume 26, 2013.

Kallus, N., Mao, X., Wang, K., and Zhou, Z. Doubly Robust Distributionally Robust Off-Policy Evaluation and Learning. In *ICML*, volume 162. PMLR, 2022.

Kawaguchi, K. and Lu, H. Ordered SGD: A New Stochastic Optimization Framework for Empirical Risk Minimization. In *AISTATS*, volume 108, 2020.

Khim, J., Leqi, L., Prasad, A., and Ravikumar, P. Uniform Convergence of Rank-weighted Learning. In *ICML*, volume 119, 2020.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B. A., Haque, I. S., Beery, S., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. WILDS: A Benchmark of in-the-Wild Distribution Shifts, 2021.

Kuhn, D., Esfahani, P. M., Nguyen, V. A., and Shafieezadeh-Abadeh, S. Wasserstein Distributionally Robust optimization: Theory and Applications in Machine Learning. *Operations research & management science in the age of analytics*, 2019.

Laguel, Y., Pillutla, K., Malick, J., and Harchaoui, Z. Superquantiles at Work: Machine Learning Applications and Efficient Subgradient Computation. *Set-Valued and Variational Analysis*, 2021.

Lam, H. Robust Sensitivity Analysis for Stochastic Systems. *Mathematics of Operations Research*, 41, 2016.

Lam, H. Recovering Best Statistical Guarantees via the Empirical Divergence-based Distributionally Robust Optimization. *Operations Research*, 67, 2019.

Lee, J., Park, S., and Shin, J. Learning Bounds for Risk-sensitive Learning. In *NeurIPS*, volume 33, 2020.

Leqi, L., Prasad, A., and Ravikumar, P. K. On Human-Aligned Risk Minimization. In *NeurIPS*, volume 32, 2019.

Levy, D., Carmon, Y., Duchi, J., and Sidford, A. Large-Scale Methods for Distributionally Robust Optimization. In *NeurIPS*, volume 33, 2020.

Li, Y., Saeed, D., Zhang, X., Ziebart, B., and Gimpel, K. Moment Distributionally Robust Tree Structured Prediction. In *NeurIPS*, volume 35. Curran Associates, Inc., 2022.

Liu, J., Wu, J., Li, B., and Cui, P. Distributionally Robust Optimization with Data Geometry. In *NeurIPS*, volume 35. Curran Associates, Inc., 2022a.

Liu, Z., Bai, Q., Blanchet, J., Dong, P., Xu, W., Zhou, Z., and Zhou, Z. Distributionally Robust Q-Learning. In *ICML*, volume 162. PMLR, 2022b.

Lotidis, K., Bambos, N., Blanchet, J., and Li, J. Wasserstein Distributionally Robust Linear-Quadratic Estimation under Martingale Constraints. In *AISTATS*, volume 206. PMLR, 2023.

Maurer, A., Parletta, D. A., Paudice, A., and Pontil, M. Robust Unsupervised Learning via L-statistic Minimization. In *ICML*. PMLR, 2021.

Mehta, R., Roulet, V., Pillutla, K., Liu, L., and Harchaoui, Z. Stochastic Optimization for Spectral Risk Measures. In *AISTATS*, 2023.

Melbourne, J. Strongly Convex Divergences. *Entropy*, 22, 2020.

Michel, P., Hashimoto, T., and Neubig, G. Modeling the Second Player in Distributionally Robust Optimization. In *ICLR*, 2021.

Michel, P., Hashimoto, T., and Neubig, G. Distributionally Robust Models with Parametric Likelihood Ratios. In *ICLR*, 2022.

Namkoong, H. and Duchi, J. C. Variance-based Regularization with Convex Objectives. *NeurIPS*, 30, 2017.

Nesterov, Y. Smooth Minimization of Non-Smooth Functions. *Mathematical programming*, 103, 2005.

Nesterov, Y. *Lectures on Convex Optimization*. Springer Publishing Company, Incorporated, 2nd edition, 2018.

Ni, J., Li, J., and McAuley, J. Justifying Recommendations using Distantly-Labeled Reviews and Fine-grained Aspects. In *EMNLP*, 2019.

Nocedal, J. and Wright, S. J. *Numerical optimization*. Springer, 1999.

Nowozin, S., Cseke, B., and Tomioka, R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. In *NeurIPS*, volume 29. Curran Associates, Inc., 2016.

Palaniappan, B. and Bach, F. Stochastic Variance Reduction Methods for Saddle-Point Problems. *NeurIPS*, 29, 2016.

Pichler, A. and Schlotter, R. Entropy Based Risk Measures. *European Journal of Operational Research*, 285, 2020.

Pillutla, K., Laguel, Y., Malick, J., and Harchaoui, Z. Federated Learning with Superquantile Aggregation for Heterogeneous Data. *Mach. Learn.*, 2023.

Piratla, V., Netrapalli, P., and Sarawagi, S. Focus on the Common Good: Group Distributional Robustness Follows. In *ICLR*, 2022.

Qian, Z., Huang, K., Wang, Q., and Zhang, X.-Y. A Survey of Robust Adversarial Training in Pattern Recognition: Fundamental, Theory, and Methodologies. *Pattern Recognit.*, 131, 2022.

Rahimian, H. and Mehrotra, S. Frameworks and Results in Distributionally Robust Optimization. *Open Journal of Mathematical Optimization*, 3, 2022.

Rényi, A. On the Theory of Order Statistics. *Acta Mathematica Academiae Scientiarum Hungarica*, 4, 1953.

Rizvi, A., Strack, B., DeShazo, J. P., Gennings, C., Olmo, J. L., Ventura, S., Cios, K. J., and Clore, J. N. Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records. *BioMed Research International*, 2014, 2014.

Rockafellar, R. T. Monotone Operators and the Proximal Point Algorithm. *SIAM Journal on Control and Optimization*, 14, 1976.

Rockafellar, R. T. and Royset, J. O. Superquantiles and Their Applications to Risk, Random Variables, and Regression. In *Theory Driven by Influential Applications*. Informs, 2013.

Roulet, V. and Harchaoui, Z. Differentiable Programming à la Moreau. In *ICASSP*. IEEE, 2022.

Sagawa, S., Koh*, P. W., Hashimoto, T. B., and Liang, P. Distributionally Robust Neural Networks. In *ICLR*, 2020.

Shorack, G. *Probability for Statisticians*. Springer Texts in Statistics, 2017.

Tay, S. S., Foo, C. S., Daisuke, U., Leong, R., and Low, B. K. H. Efficient Distributionally Robust Bayesian Optimization with Worst-case Sensitivity. In *ICML*, volume 162. PMLR, 2022.

Thekumparampil, K. K., Jain, P., Netrapalli, P., and Oh, S. Efficient Algorithms for Smooth Minimax Optimization. In *NeurIPS*, volume 32. Curran Associates, Inc., 2019.

Tsanas, A. and Xifara, A. Accurate Quantitative Estimation of Energy Performance of Residential Buildings Using Statistical Machine Learning Tools. *Energy and Buildings*, 49, 2012.

Tüfekci, P. Prediction of Full Load Electrical Power Output of a Base Load Operated Combined Cycle Power Plant using Machine Learning Methods. *International Journal of Electrical Power & Energy Systems*, 60, 2014.

Vu, H., Tran, T., Yue, M.-C., and Nguyen, V. A. Distributionally Robust Fair Principal Components via Geodesic Descents. In *ICLR*, 2022.

Wang, S., Si, N., Blanchet, J., and Zhou, Z. A Finite Sample Complexity Bound for Distributionally Robust Q-learning. In *AISTATS*, volume 206. PMLR, 2023.

Wang, Y. and Li, J. Improved Algorithms for Convex-Concave Minimax Optimization. In *NeurIPS*, volume 33. Curran Associates, Inc., 2020.

Wang, Z., Shen, L., Fang, L., Suo, Q., Duan, T., and Gao, M. Improving Task-free Continual Learning by Distributionally Robust Memory Evolution. In *ICML*, volume 162. PMLR, 2022.

Williamson, R. and Menon, A. Fairness Risk Measures. In *ICML*, 2019.

Xiao, L. Dual Averaging Method for Regularized Stochastic Learning and Online Optimization. In *NeurIPS*, volume 22, 2009.

Xu, M., Huang, P., Niu, Y., Kumar, V., Qiu, J., Fang, C., Lee, K.-H., Qi, X., Lam, H., Li, B., and Zhao, D. Group Distributionally Robust Reinforcement Learning with Hierarchical Latent Variables. In *AISTATS*, volume 206. PMLR, 2023.

Yang, J., Zhang, S., Kiyavash, N., and He, N. A Catalyst Framework for Minimax Optimization. In *NeurIPS*, volume 33. Curran Associates, Inc., 2020.

Yang, Z., Guo, Y., Xu, P., Liu, A., and Anandkumar, A. Distributionally Robust Policy Gradient for Offline Contextual Bandits. In *AISTATS*, volume 206. PMLR, 2023.

Yeh, I. Analysis of Strength of Concrete Using Design of Experiments and Neural Networks. *Journal of Materials in Civil Engineering*, 18, 2006.

Yu, Y., Lin, T., Mazumdar, E. V., and Jordan, M. Fast Distributionally Robust Learning with Variance-Reduced Min-Max Optimization. In *AISTATS*. PMLR, 2022.

Appendix

In the appendices, we give summarize notation in Appx. A and provide intuition and results regarding the primal/dual objective function in Appx. B. We describe in detail efficient implementations of the proposed algorithm in Appx. C. In Appx. D, we describe the convergence analyses of the main algorithm. In Appx. E and Appx. F, we describe our saddle point and Moreau-envelope-based variants, respectively. Appx. G contains technical results shared to multiple proofs. We then describe the experimental setup in detail in Appx. H and give additional results in Appx. I.

Table of Contents

A. Summary of Notation

We summarize the notation used throughout in Tab. 1.

Symbol	Description
$\mu \geq 0$	Standard regularization constant.
$\nu \geq 0$	Shift cost.
$\bar{\nu}$	Shorthand $\bar{\nu} = 2n\nu$ (used in the convergence proofs).
$\ell_1(w), \ldots, \ell_n(w)$	Loss functions $\ell_i : \mathbb{R}^d \to \mathbb{R}$.
$\ell(w)$	Vector of losses $\ell(w) = (\ell_1(w), \ldots, \ell_n(w))$ for $w \in \mathbb{R}^d$.
$r_i(w)$	Regularized loss $r_i(w) = \ell_i(w) + \frac{\mu}{2}\|w\|_2^2$.
$r(w)$	Vector of regularized losses $r(w) = (r_1(w), \ldots, r_n(w))$.
$\nabla \ell(w)$	Jacobian matrix of $\ell : \mathbb{R}^d \to \mathbb{R}^n$ at w (shape = $n \times d$).
σ	The vector $\sigma = (\sigma_1, \ldots, \sigma_n) \in [0,1]^n$ where each $\sigma_1 \leq \ldots \leq \sigma_n$ and they sum to 1.
$\mathcal{P}(\sigma)$	The set $\{\Pi\sigma \, : \, \Pi \in [0,1]^{n \times n}, \Pi \mathbf{1}_n = \mathbf{1}_n, \Pi^\top \mathbf{1}_n = \mathbf{1}_n\}$, known as the permutahedron.
f	Convex function $f : [0, \infty) \to \mathbb{R} \cup \{+\infty\}$ generating an f-divergence.
f^*	Convex conjugate $f^*(y) := \sup_{x \in \mathbb{R}} \{xy - f(x)\}$.
Ω_f or Ω	Shift penalty function $\Omega_f : \mathcal{P}(\sigma) \mapsto [0, \infty)$. We consider f-divergence penalties $\Omega_f(q) = D_f(q \| \mathbf{1}_n/n)$.
$\mathcal{L}_\sigma$	Main objective $\mathcal{L}_\sigma(w) = \max_{q \in \mathcal{P}(\sigma)} \left\{q^\top \ell(w) - \nu D_f(q \| \mathbf{1}_n/n)\right\} + \frac{\mu}{2}\left\|w\right\|_2^2$.
$q^{\text{opt}}(l)$	Most unfavorable distribution for a given vector l of losses, i.e., $q^{\text{opt}}(l) = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l - \nu D(q \| \mathbf{1}_n/n)$.
$w^\star$	Optimal weights $\arg\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{P}(\sigma)} q^\top l - \nu D(q \| \mathbf{1}_n/n) + (\mu/2)\left\|w\right\|_2^2$.
$q^\star$	Most unfavorable distribution of $\ell(w^\star)$, i.e., $q^\star = q^{\text{opt}}(\ell(w^\star))$
G	Lipschitz constant of each ℓ_i w.r.t. $\left\|\cdot\right\|_2$.
L	Lipschitz constant of each $\nabla \ell_i$ w.r.t. $\left\|\cdot\right\|_2$.
M	$M = L + \mu$, the Lipschitz constant of each ∇r_i w.r.t. $\left\|\cdot\right\|_2$.
$\mathbb{E}_t\left[\cdot\right]$	Shorthand for $\mathbb{E}\left[\,\cdot \mid w^{(t)}\right]$, i.e., expectation conditioned on $w^{(t)}$.

Table 1: Notation used throughout the paper.

B. Properties of the Primal and Dual Objectives

In this section, we state (and prove) the properties of the objectives we consider. Recall that we are interested in the optimization problem

$$\min_{w \in \mathbb{R}^d} \left[\mathcal{L}_\sigma(w) := \max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w) - \nu D_f(q \| \mathbf{1}_n/n) + \frac{\mu}{2}\left\|w\right\|_2^2\right], \tag{7}$$

where $D_f(q \| \mathbf{1}_n/n)$ denotes an f-divergence between the distribution given by q and the discrete uniform distribution $\mathbf{1}_n/n = (1/n, \ldots, 1/n)$. Note that the formulation (7) is more general than that of (3) in Sec. 2, as we consider generic

f-divergences, as opposed to the χ^2 divergence. The analysis techniques are broadly the same, so we derive them in the general case and describe specific cases such as the χ^2 and Kullback-Liebler (KL) divergences as examples.

Our goal for this section will be to derive properties of the function $\mathcal{L}_\sigma(w)$, or the *primal objective*, as well as the inner maximization problem, which we refer to as the *dual objective*. Both will be useful in motivating and analyzing SpecSAGA (used for the primal minimization) and various subroutines used to compute the maximally unfavorable distribution (i.e., the maximizer over q in the inner maximization).

Review of f-Divergences. Consider a strongly convex function $f : [0, \infty) \mapsto \mathbb{R} \cup \{+\infty\}$ such that $f(1) = 0$. The f*-divergence* from q to p generated by this function f is given by

$$D_f(q \| p) := \sum_{i=1}^{n} f\left(\frac{q_i}{p_i}\right) p_i,$$

where we define $0f\left(0/0\right) := 0$ in the formula above.

The χ^2-divergence is generated by $f_{\chi^2}(x) = x^2 - 1$ and the KL divergence is generated by $f_{\mathrm{KL}}(x) = x \ln x$.

The Dual Problem. We describe the inner maximization first, that is

$$\max_{q \in \mathcal{P}(\sigma)} \left\{ q^\top l - \nu D_f(q \| \mathbf{1}_n) \right\} . \tag{8}$$

Its properties will inform the algorithmic implementation for the minimization over w in (7). In our specific case, since we care about the f-divergence between q and the uniform distribution $\mathbf{1}_n/n$, we have

$$D_f(q \| \mathbf{1}_n/n) := \frac{1}{n} \sum_{i=1}^{n} f\left(nq_i\right). \tag{9}$$

We now derive the dual problem to Equation (8). This will lead to an algorithm to solve the optimization problem efficiently. Throughout, we denote $f^*(y) := \sup_{x \in \mathbb{R}} \left\{ xy - f(x) \right\}$ as the convex conjugate of f.

Proposition 4. *Let $l \in \mathbb{R}^n$ be a vector and π be a permutation that sorts its entries in non-decreasing order, i.e., $\ell_{\pi(1)} \leq \ldots \leq \ell_{\pi(n)}$ Then, the maximization over the permutahedron subject to the shift penalty can be expressed as*

$$\max_{q \in \mathcal{P}(\sigma)} \left\{ q^\top l - \nu D_f(q \| \mathbf{1}_n/n) \right\} = \min_{\substack{c \in \mathbb{R}^n \\ c_1 \leq \ldots \leq c_n}} \sum_{i=1}^{n} g_i(c_i; l), \tag{10}$$

where we define

$$g_i(c_i\,;\,l) := \sigma_i c_i + \frac{\nu}{n}\ f^*\left(\frac{l_{\pi(i)} - c_i}{\nu}\right).$$

Proof. Let $\iota_{\mathcal{P}(\sigma)}$ denote the indicator function of the permutahedron $\mathcal{P}(\sigma)$, which is 0 inside $\mathcal{P}(\sigma)$ and $+\infty$ outside of $\mathcal{P}(\sigma)$. Its convex conjugate is the support function of the permutahedron, i.e.,

$$\iota^*_{\mathcal{P}(\sigma)}(l) = \max_{q \in \mathcal{P}(\sigma)} q^\top l.$$

For two closed convex functions h_1 and h_2 that are bounded from below, the convex conjugate of their sum is the infimal convolution of their conjugate (Hiriart-Urruty & Lemaréchal, 2004, Proposition 6.3.1):

$$(h_1 + h_2)^*(x) = \inf_{y} \left\{ h_1^*(y) + h_2^*(x - y) \right\} .$$

In our context, taking $h_1(q) = \iota_{\mathcal{P}(\sigma)}(q)$ and $h_2(q) = \Omega_f(q) = \nu D_f(q \| \mathbf{1}_n/n)$, we have

$$\begin{aligned}
\sup_{q \in \mathcal{P}(\sigma)} \left\{ q^\top l - \Omega_f(q) \right\} &= \sup_{q \in \mathbb{R}^n} \left\{ q^\top l - (\iota_{\mathcal{P}(\sigma)}(q) + \Omega_f(q)) \right\} \\
&= (\iota_{\mathcal{P}(\sigma)} + \Omega_f)^*(l) \\
&= \inf_{y \in \mathbb{R}^n} \left\{ \iota^*_{\mathcal{P}(\sigma)}(y) + \Omega^*_f(l - y) \right\} \\
&= \inf_{y \in \mathbb{R}^n} \left\{ \max_{q \in \mathcal{P}(\sigma)} q^\top y + \Omega^*_f(l - y) \right\} \\
&= \inf_{y \in \mathbb{R}^n} \left\{ \sum_{i=1}^n \sigma_i y_{(i)} + \Omega^*_f(l - y) \right\}, \qquad (11)
\end{aligned}$$

where $y_{(1)} \leq \ldots \leq y_{(n)}$ are the ordered values of $y \in \mathbb{R}^n$.

Since for any $x \in \mathbb{R}^n$, Ω_f is decomposable into a sum of identical functions evaluated at the coordinates $(x_1, \ldots, x_n)$, that is, $\Omega_f(x) = \sum_{i=1}^n \omega(x_i)$, its convex conjugate is $\Omega^*_f(y) = \sum_{i=1}^n \omega^*(y_i)$. In our case, $\omega(x_i) = \frac{\nu}{n} f(nx_i)$ from Equation (9), so $\omega^*(y_i) = (\nu/n) f^*(y_i/\nu)$.

Next, by convexity of ω^*, we have that if for scalars l_i, l_j, y_i, y_j such that $l_i \leq l_j$ and $y_i \geq y_j$, then using Lem. 44, we have that

$$\omega^*(l_i - y_i) - \omega^*(l_j - y_j) \geq \omega^*(l_i - y_j) - \omega^*(l_j - y_i).$$

Hence for y to minimize $\Omega^*_f(l - y) = \sum_{i=1}^n \omega^*(l_i - y_i)$, the coordinates of y must be ordered as l. That is, if π is an argsort for l, s.t. $l_{\pi(1)} \leq \ldots \leq l_{\pi(n)}$, then $y_{\pi(1)} \leq \ldots \leq y_{\pi(n)}$. Since $\iota^*_{\mathcal{P}(\sigma)}(y) = \sum_{i=1}^n \sigma_i y_{(i)}$ does not depend on the ordering of y, the solution of (11) must also be ordered as l such that the dual problem (11) can be written as

$$\begin{aligned}
\inf_{\substack{y \in \mathbb{R}^n \\ y_{\pi(1)} \leq \cdots \leq y_{\pi(n)}}} \sum_{i=1}^n \sigma_i y_{\pi(i)} + \frac{\nu}{n} f^* \left(\frac{l_{\pi(i)} - y_{\pi(i)}}{\nu} \right) &= \inf_{\substack{y \in \mathbb{R}^n \\ c_1 \leq \ldots \leq c_n}} \sum_{i=1}^n \sigma_i c_i + \frac{\nu}{n} f^* \left(\frac{l_{\pi(i)} - c_i}{\nu} \right) \\
&= \min_{\substack{c \in \mathbb{R}^n \\ c_1 \leq \ldots \leq c_n}} \sum_{i=1}^n g_i(c_i; l).
\end{aligned}$$

□

Because we are interested in computing the maximizer of (8), we denote it as

$$q^{\text{opt}}(l) = \operatorname*{arg\,max}_{q \in \mathcal{P}(\sigma)} \left\{ q^\top l - \nu D_f(q \| \mathbf{1}_n/n) \right\}.$$

The maximizer will exist and be unique as we considered f strongly convex such that $q \mapsto D_f(q \| \mathbf{1}_n/n)$ is also strongly convex. The next result allows use to use a minimizer of (10) to compute the maximizer of (8).

Corollary 5. *In the setting of Prop. 4, if*

$$c^{opt}(l) \in \operatorname*{arg\,min}_{\substack{c \in \mathbb{R}^n \\ c_1 \leq \ldots \leq c_n}} \sum_{i=1}^n g_i(c_i; l),$$

then

$$q_i^{opt}(l) = \frac{1}{n} [f^*]' \left(\tfrac{1}{\nu} (l_i - c^{opt}_{\pi^{-1}(i)}(l)) \right). \qquad (12)$$

The Pool Adjacent Violators (PAV) algorithm is designed exactly for the minimization (10). The algorithm is described for the the χ^2-divergence with implementation steps in Appx. C. Both the argsort π and the inverse argort π^{-1} are mappings from $[n] = \{1, \ldots, n\}$ onto itself, but the interpretation of these indices are different for the input and output spaces $[n]$. The argsort π can be thought of as an *index finder*, in the sense that for a vector $l \in \mathbb{R}^n$, because $l_{\pi(1)} \leq \ldots \leq l_{\pi(n)}$, $\pi(i)$ can be interpreted as the index of an element of l which achieves the rank i in the sorted vector. On the other hand, $\pi^{-1}(i)$ can be thought of as a *rank finder*, in that $\pi^{-1}(i) = \operatorname{rank}(i)$ is the position that l_i takes in the sorted form of l. To summarize

$$\pi : \underbrace{[n]}_{\text{ranks of losses}} \to \underbrace{[n]}_{\text{indices of training examples}} \quad \text{while} \quad \pi^{-1} : \underbrace{[n]}_{\text{indices of training examples}} \to \underbrace{[n]}_{\text{ranks of losses}}$$

We may equivalently write (12) as

$$q_i^{\text{opt}}(l) = \frac{1}{n}[f^*]'\left(\tfrac{1}{\nu}(l_i - c^{\text{opt}}_{\text{rank}(i)}(l))\right). \tag{13}$$

Finally, as seen in Appx. C, it will be helpful to compute q^{opt} in sorted order. Because the f-divergence is agnostic to the ordering of the q vector (as it is being compared to the uniform distribution), q can also be sorted by π. Thus, we may also write

$$q_{(i)}^{\text{opt}}(l) = \frac{1}{n}[f^*]'\left(\tfrac{1}{\nu}(l_{(i)} - c_i^{\text{opt}}(l))\right). \tag{14}$$

In the following, the f-divergences we consider as running examples are:

$$f_{\chi^2}(x) = x^2 - 1 \text{ and } f^*_{\chi^2}(y) = y^2/4 + 1 \qquad (\chi^2\text{-divergence})$$
$$f_{\text{KL}}(x) = x \ln x \text{ and } f^*_{\text{KL}}(y) = e^{-1}\exp(y). \qquad (\text{KL-divergence})$$

The Primal Function. When divergence generator f is strongly convex and the loss function $\ell : \mathbb{R}^d \to \mathbb{R}^n$ is convex and differentiable, we have that Equation (7) is differentiable, as we show next.

Lemma 6. *Let $\ell : \mathbb{R}^d \to \mathbb{R}^n$ be differentiable with Jacobian $w \mapsto \nabla\ell(w) \in \mathbb{R}^{n\times d}$. Let each $\ell_i : \mathbb{R}^d \to \mathbb{R}$ be convex. Let f be α_n-strongly convex on the interval $[0, n]$. Then, the function $\mathcal{L}_\sigma$ from Equation (7) is differentiable with its gradient equal to*

$$\nabla\mathcal{L}_\sigma(w) = (\nabla\ell(w))^\top q^{opt}(\ell(w)) + \mu w.$$

Furthermore $l \mapsto q^{opt}(l)$ is $(\alpha_n n\nu)^{-1}$-Lipschitz continuous w.r.t. $\|\cdot\|_2$.

Proof. First, due to the α_n-strong convexity of f, for any $q, \rho \in [0,1]^n$, we may average the strong convexity inequality evaluated at $n\rho$, nq to write

$$\frac{1}{n}\sum_{i=1}^n f(nq_i) \geq \frac{1}{n}\sum_{i=1}^n f(n\rho_i) + \frac{1}{n}\sum_{i=1}^n f'(n\rho_i)(nq_i - n\rho_i) + \frac{1}{n}\sum_{i=1}^n \frac{\alpha_n}{2}(nq_i - n\rho_i)^2.$$

Defining $\Omega_f(q) := D_f(q\|\mathbf{1}_n/n)$, the statement above can be succinctly written as

$$\Omega_f(q) \geq \Omega_f(\rho) + \nabla\Omega_f(\rho)^\top(q - \rho) + \frac{\alpha_n n}{2}\|q - \rho\|_2^2.$$

Therefore, Ω_f is $(\alpha_n n)$-strongly convex with respect to $\|\cdot\|_2$ on $[0,1]^n$. Next, due to the convexity of each ℓ_i and the non-negativity of any $q \in \mathcal{P}(\sigma)$, we have that

$$w \mapsto \max_{q\in\mathcal{P}(\sigma)}\left\{q^\top\ell(w) - \nu\Omega_f(q)\right\}$$

is convex, as is its pointwise maximum (over q) of a family of convex functions $q^\top \ell(w)$. For any $w \in \mathbb{R}^d$, the function

$$q \mapsto q^\top \ell(w) - \nu \Omega_f(q)$$

is $(\nu \alpha_n n)$-strongly concave if f is α_n-strongly convex. Thus, it admits a unique maximizer. By Danskin's theorem (Bertsekas, 1997, Proposition B.25), we have that $\mathcal{L}_\sigma$ is continuously differentiable with

$$\nabla \mathcal{L}_\sigma(w) = \nabla \ell(w)^\top q^{\text{opt}}(\ell(w)) + \mu w\,.$$

Moreover, by Nesterov (2005, Theorem 1), we have that $l \mapsto q^{\text{opt}}(l)$ is Lipschitz continuous with Lipschitz constant equal to the inverse of the strong convexity constant of $\nu \Omega_f$, which is $\nu \alpha_n n$. □

Returning to our canonical examples, we have that for the χ^2, $f_{\chi^2}(x) = x^2 - 1$ is 2-strongly convex and $\mathbb{R}$ and that $f_{\text{KL}}(x) = x \ln x$ is $(1/n)$-strongly convex on $[0, n]$. Thus, the function $l \mapsto q^{\text{opt}}(l)$ will have Lipschitz constant $2n\nu$ and ν, respectively.

Smoothness Properties. By applying Lem. 6 to Lipschitz continuous losses, we may achieve the following guarantee regarding the changes in q^{opt} with respect to w.

Lemma 7. *Let f be α_n-strongly convex on the interval $[0, n]$. For any $w_1, \ldots, w_n, w'_1, \ldots, w'_n \in \mathbb{R}^d$ construct $\bar{\ell}(w_1, \ldots, w_n) = \left(\ell_i(w_i)\right)_{i=1}^n \in \mathbb{R}^n$, as well as $\bar{\ell}(w'_1, \ldots, w'_n)$ where each ℓ_i is G-Lipschitz w.r.t. $\|\cdot\|_2$. Then, we have*

$$\left\|q^{opt}(\bar{\ell}(w_1, \ldots, w_n)) - q^{opt}(\bar{\ell}(w'_1, \ldots, w'_n))\right\|_2^2 = \frac{G^2}{n^2 \alpha_n^2 \nu^2} \sum_{i=1}^n \|w_i - w'_i\|_2^2\,.$$

Proof. By the Lipschitz property of q^{opt} (Lem. 6), we have,

$$\begin{aligned}
\left\|q^{\text{opt}}(\bar{\ell}(w_1, \ldots, w_n)) - q^{\text{opt}}(\bar{\ell}(w'_1, \ldots, w'_n))\right\|_2^2 &\leq \frac{1}{n^2 \alpha_n^2 \nu^2} \left\|\bar{\ell}(w_1, \ldots, w_n) - \bar{\ell}(w'_1, \ldots, w'_n)\right\|_2^2 \\
&\leq \frac{1}{n^2 \alpha_n^2 \nu^2} \sum_{i=1}^n \left(\ell_i(w_i) - \ell_i(w'_i)\right)_2^2 \\
&\leq \frac{G^2}{n^2 \alpha_n^2 \nu^2} \sum_{i=1}^n \|w_i - w'_i\|_2^2\,.
\end{aligned}$$

□

As a special case of Lem. 7, we may consider $w_1 = \cdots = w_n = w \in \mathbb{R}^d$ and $w'_1 = \cdots = w'_n = w' \in \mathbb{R}^d$, in which case the result reads

$$\left\|q^{\text{opt}}(\ell(w)) - q^{\text{opt}}(\ell(w'))\right\|_2^2 = \frac{G^2}{n \alpha_n^2 \nu^2} \|w - w'\|_2^2\,.$$

Properties under No Shift Penalty. Next, we use the smoothness properties above to prove Prop. 3 by virtue of the following proposition, which states the equivalence of the minimizers of "no-cost" and "low-cost" objectives.

Proposition 8. *Let $w^\star_\nu$ be the unique minimizer of (3) with shift cost $\nu \geq 0$ and χ^2-divergence penalty. Define $\ell_{(1)}(w^\star_0) < \ldots, < \ell_{(n)}(w^\star_0)$ to be the order statistics of $\ell_1(w^\star_0), \ldots, \ell_n(w^\star_0)$, which are assumed to be distinct. Consider ν_0 such that*

$$n\nu_0 \left(\sigma_{i+1} - \sigma_i\right) < \ell_{(i+1)}(w^\star_0) - \ell_{(i)}(w^\star_0) \text{ for } i = 1, \ldots, n. \tag{15}$$

We have that $w^\star_0 = w^\star_\nu$ for all $\nu \leq \nu_0$.

Proof. For a vector $l \in \mathbb{R}^n$ and $\nu \geq 0$, consider

$$\begin{aligned} h_\nu(l) &\coloneqq \max_{q \in \mathcal{P}(\sigma)} q^\top l - \nu n \left\| q - \mathbf{1}_n/n \right\|_2^2 \\ &= \max_{q \in \mathcal{P}(\sigma)} q^\top \left(l + 2\nu \mathbf{1}_n \right) - \nu n \left\| q \right\|_2^2 - (\nu/n) \left\| \mathbf{1}_n \right\|_2^2 \\ &= \max_{q \in \mathcal{P}(\sigma)} q^\top l - \nu n \left\| q \right\|_2^2 + \nu \\ &\coloneqq g_\nu(l) + \nu, \end{aligned}$$

where we used that $q^\top \mathbf{1} = 1$ for all $q \in \mathcal{P}(\sigma)$. For $\nu > 0$, by Danskin's theorem (Bertsekas, 1997, Proposition B.25),

$$\nabla h_\nu(l) = \nabla g_\nu(l) = \operatorname*{arg\,max}_{q \in \mathcal{P}(\sigma)} q^\top l - \nu n \left\| q \right\|_2^2 .$$

By applying Proposition 5 of (Blondel et al., 2020), we have that if

$$n\nu_0 \left(\sigma_{i+1} - \sigma_i \right) < \ell_{(i+1)}(w_0^\star) - \ell_{(i)}(w_0^\star) \text{ for } i = 1, \ldots, n, \tag{16}$$

for some $\nu_0 > 0$, then for any $\nu \leq \nu_0$,

$$\nabla g_\nu(\ell(w_0^\star)) = \nabla g_0(\ell(w_0^\star)).$$

Denote our objective as

$$\mathcal{L}_{\sigma,\nu}(w) = h_\nu(\ell(w)) + \frac{\mu}{2} \left\| w \right\|_2^2 ,$$

where we explicitly show the dependence on the shift cost $\nu \geq 0$. For $\nu = 0$, since the losses are differentiable and $\ell(w_0^\star)$ is composed of distinct coordinates, $\mathcal{L}_{\sigma,0}$ is differentiable at $w_0^\star$ with gradient $\nabla \ell(w_0^\star)^\top \nabla h_0(\ell(w_0^\star)) + \mu w_0^\star$ (Mehta et al., 2023, Proposition 2), where $\nabla \ell(w_0^\star) \in \mathbb{R}^{n \times d}$ denotes the Jacobian of ℓ at $w_0^\star$. Using the chain rule, we successively deduce

$$\begin{aligned} \nabla \mathcal{L}_{\sigma,0}(w_0^\star) = 0 &\iff \nabla \ell(w_0^\star)^\top \nabla h_0(\ell(w_0^\star)) + \mu w_0^\star = 0 \\ &\iff \nabla \ell(w_0^\star)^\top \nabla g_0(\ell(w_0^\star)) + \mu w_0^\star = 0 \\ &\iff \nabla \ell(w_0^\star)^\top \nabla g_\nu(\ell(w_0^\star)) + \mu w_0^\star = 0 \\ &\iff \nabla \ell(w_0^\star)^\top \nabla h_\nu(\ell(w_0^\star)) + \mu w_0^\star = 0 \\ &\iff \nabla \mathcal{L}_{\sigma,\nu}(w_0^\star) = 0. \end{aligned}$$

Applying the first-order optimality conditions of $\mathcal{L}_{\sigma,0}$ and $\mathcal{L}_{\sigma,\nu}$, as well as the uniqueness of $w_0^\star$ completes the proof. □

Prop. 3 of the main paper then follows by combining Prop. 8 above with the convergence guarantee Thm. 20 of decoupled SpecSAGA (Algorithm 9). Indeed, Thm. 20 shows that decoupled SpecSAGA (Algorithm 9) is able to converge linearly for arbitrarily small $\nu > 0$ and as long as $\nu \leq \nu_0$. Under Prop. 8, the minimizer will be equal to $w_0^\star$.

We interpret this phenomenon as the "hidden smoothness" of $\mathcal{L}_\sigma$, in that the non-differentiable points of the map $w \mapsto \max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w)$ are precisely the points at which $\ell_i(w) = \ell_j(w)$ for some $i \neq j$, as the subdifferential may contain multiple elements (Mehta et al., 2023, Proposition 2). Thus, if the losses are well-separated enough (in comparison to the spectrum σ) at the minimizer $w_0^\star$, the objective for the non-smooth setting $\nu = 0$ and regularized setting $\nu > 0$ result in the same minimizer.

C. Efficient Implementation of SpecSAGA

In this section, we describe how to implement SpecSAGA efficiently. A precise version of Algorithm 1 is given in Algorithm 2 We index relevant quantities with the iterate number t to explicitly describe their changes at each step. As in Algorithm 1, we maintain a table of losses $l^{(t)} \in \mathbb{R}^n$, gradients $g^{(t)} \in \mathbb{R}^{n \times d}$, weights $\rho^{(t)} \in \mathbb{R}^n$, and aggregate $\bar{g}^{(t)} = \sum_{i=1}^n \rho_i^{(t)} g_i^{(t)}$ used to construct the control variate. We also maintain the maximizer $q^{(t)} = q^{\text{opt}} \left(l^{(t)} \right)$ used in the

Algorithm 2 SpecSAGA: A precise version on Algorithm 1 with iteration counters specified.

```
Inputs: Initial points $w^{(0)}$, stepsize $\eta > 0$, number of iterations $T$
$q^{(0)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w^{(0)}) - \nu D_f(q \| \mathbf{1}_n/n)$, $\rho^{(0)} = q^{(0)}$.
Set $l^{(0)} = (\ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^n$, $g^{(0)} = (\nabla \ell_i(w^{(0)}) + \mu w^{(0)})_{i=1}^n \in \mathbb{R}^{d \times n}$,
Compute $\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$.
for $t = 0, \ldots, T-1$ do
    $i_t \sim \mathrm{Unif}([n])$.
    $v^{(t)} = nq_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - (n\rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - \bar{g}^{(t)})$.
    $w^{(t+1)} = w^{(t)} - \eta v^{(t)}$.                                             ▷ Update parameter vector.
    $l_{i_t}^{(t+1)} = \ell_{i_t}(w^{(t+1)})$ and $l_i^{(t+1)} = l_i^{(t)}$ for $i \neq i_t$.
    $g_{i_t}^{(t+1)} = \nabla \ell_{i_t}(w^{(t)}) + \mu w^{(t)}$ and $g_i^{(t+1)} = g_i^{(t)}$ for $i \neq i_t$.
    $\rho_{i_t}^{(t+1)} = q_{i_t}^{(t)}$ and $\rho_i^{(t+1)} = \rho_i^{(t)}$ for $i \neq i_t$.
    $q^{(t+1)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(t+1)} - \nu D_f(q \| \mathbf{1}_n/n)$.          ▷ Update data distribution.
    $\bar{g}^{(t+1)} = \bar{g}^{(t)} + (\rho_{i_t}^{(t+1)} g_{i_t}^{(t+1)} - \rho_{i_t}^{(t)} g_{i_t}^{(t)}) = \sum_{i=1}^n \rho_i^{(t+1)} g_i^{(t+1)}$.   ▷ Update control variate.
Output: Final point $w^{(T)}$.
```

stochastic gradient estimate.

Efficient Implementation. For efficiency, we exactly solve the maximization problem

$$q^{(t)} = q^{\mathrm{opt}}\left(l^{(t)}\right) = \underset{q \in \mathcal{P}(\sigma)}{\arg\max} \left\{ q^\top l^{(t)} - (\nu/n) \sum_{i=1}^n f(nq_i) \right\}. \tag{17}$$

by a sequence of three steps:

- **Sorting:** Find π such that $l_{\pi(1)}^{(t)} \leq \ldots \leq l_{\pi(n)}^{(t)}$.
- **Isotonic regression:** Apply Pool Adjacent Violators (PAV) (Algorithm 4) to solve the isotonic regression minimization problem (10), yielding solution $c^{(t)} = c^{\mathrm{opt}}(l^{(t)})$.
- **Conversion:** Use (12) to convert $c^{(t)}$ back to $q^{(t)} = q^{\mathrm{opt}}(l^{(t)})$.

The sorting step runs in $O(n \ln n)$ elementary operations whereas the isotonic regression and conversion steps run in $O(n)$ operations. Crucially, retrieving $q^{(t)}$ from the output $c^{(t)} = c^{\mathrm{opt}}(l^{(t)})$ in the third step can be done by a single $O(n)$-time pass by setting

$$q_{\pi^{(t)}(i)}^{(t)} = \frac{1}{n}[f^*]'\left(\tfrac{1}{\nu}(l_{\pi^{(t)}(i)}^{(t)} - c_i^{(t)})\right)$$

for $i = 1, \ldots, n$, as opposed to computing the inverse π^{-1} and use (12) directly, which in fact requires another sorting operation and can be avoided. Because only one element of $l^{(t)}$ changes on every iteration, we may sort it by simply bubbling the value of the index that changed into its correct position to generate the sorted version of $l^{(t+1)}$. The full algorithm is given Algorithm 3. We give a brief explanation on the PAV algorithm for general f-divergences below.

Pool Adjacent Violators (PAV) Algorithm. First, recall the optimization problem we wish to solve:

$$\min_{\substack{c \in \mathbb{R}^n \\ c_1 \leq \ldots \leq c_n}} \sum_{i=1}^n g_i(c_i; l), \quad \text{where} \quad g_i(c_i; l) := \sigma_i c_i + \frac{\nu}{n} f^*\left(\frac{l_{\pi(i)} - c_i}{\nu}\right). \tag{18}$$

The objective can be thought of as fitting a real-valued monotonic function to the points $(1, l_{\pi(1)}), \ldots, (n, l_{\pi(n)})$, which would require specifying its values $(c_1, \ldots, c_n)$ on $(1, \ldots, n)$ and defining the function as any $x \in [c_j, c_{j+1}]$ on $(j, j+1)$. Because $l_{\pi(1)} \leq \ldots \leq l_{\pi(n)}$, if we evaluated our function $(c_1, \ldots, c_n)$ on a loss such as $\sum_{i=1}^n (l_{\pi(i)} - c_i)^2$, we may easily solve the problem by returning $c_1 = \ell_{\pi(1)}, \dots, c_n = l_{\pi(n)}$. However, by specifying functions $g_1, \dots, g_n$ we allow our loss function to change in different regions of the inputs space $\{1, \dots, n\}$. In such cases, the monotonicity constraint $c_1 \leq \ldots \leq c_n$ is often violated because individually minimizing $g_i(c_i)$ for each c_i has no guarantee of yielding a function that is monotonic.

The idea behind the PAV algorithm is to attempt a pass at minimizing each g_i individually, and correcting *violations* as they appear. To provide intuition, define $c_i^* \in \arg\min_{c_i \in \mathbb{R}} g_i(c_i)$, and consider $i < j$ such that $c_i^* > c_j^*$. If f^* is strictly convex, then $g_i(x) > g_i(c_i^*)$ for any $x < c_i^*$ and similarly $g_j(x) > g_j(c_j^*)$ for any $x > c_j^*$. Thus, to correct the violation, we decrease c_i^* to $\bar{c}_i$ and increase c_j^* to $\bar{c}_j$ until $\bar{c}_i = \bar{c}_j$. We determine this midpoint precisely by

$$\bar{c}_i = \bar{c}_j = \arg\min_{x \in \mathbb{R}} g_i(x) + g_j(x)$$

as these are exactly the contributions made by these terms in the overall objective. The computation above is called *pooling* the indices i and j. We may generalize this viewpoint to *violating chains*, that is collections of contiguous indices $(i, i+1, \dots, i+m)$ such that $c_j^* < c_i^*$ for all $j < i$ and $c_j^* > c_{i+m}^*$ for all $j > i+m$, but $c_i^* > c_{i+m}^*$. One approach is use dynamic programming to identify such chains and then compute the pooled quantities

$$\bar{c}_i = \arg\min_{x \in \mathbb{R}} \sum_{k=1}^{m} g_{i+k}(x).$$

This requires two passes through the vector: one for identifying violators and the other for pooling. The Pool Adjacent Violators algorithm, on the other hand, is able to perform both operations in one pass by greedily pooling violators as they appear. This can be viewed as a meta-algorithm, as it hinges on the notion that the solution of "larger" pooling problems can be easily computed from solutions of "smaller" pooling problems. Precisely, for indices $S \subseteq [n] = \{1, \dots, n\}$ define

$$\mathrm{Sol}(S) = \arg\min_{x \in \mathbb{R}} \sum_{i \in S} g_i(x).$$

We rely on the existence of an operation Pool, such that for any $S, T \subseteq [n]$ such that $S \cap T = \emptyset$, we have that

$$\mathrm{Sol}(S \cup T) = \mathrm{Pool}\left(\mathrm{Sol}(S), m(S), \mathrm{Sol}(T), m(T)\right), \tag{19}$$

where $m(S)$ denotes "metadata" associated to S, and that the number of elementary operations in the Pool function is $O(1)$ with respect to $|S| + |T|$. We review our running examples.

For the χ^2-divergence, we have that $f_{\chi^2}(x) = x^2 - 1$ and $f_{\chi^2}^*(y) = y^2/4 + 1$, so

$$\begin{aligned}
\mathrm{Sol}(S) &= \arg\min_{x \in \mathbb{R}} \left\{ x \left(\sum_{i \in S} \sigma_i \right) + |S| + \frac{\nu}{4n} \sum_{i \in S} (l_{\pi(i)} - x)^2 \right\} \\
&= \frac{1}{|S|} \left[(2n/\nu) \sum_{i \in S} \sigma_i - \sum_{i \in S} l_{\pi(i)} \right] \\
\mathrm{Sol}(S \cup T) &= \frac{1}{|S| + |T|} \left[(2n/\nu) \sum_{i \in S \cup T} \sigma_i - \sum_{i \in S \cup T} l_{\pi(i)} \right] \\
&= \frac{|S| \,\mathrm{Sol}(S) + |T| \,\mathrm{Sol}(T)}{|S| + |T|}.
\end{aligned}$$

Thus, the metadata $m(S) = |S|$ used in the pooling step eq. (19) is the size of each subset.

For the KL divergence, $f_{\text{KL}}(x) = x \ln x$ and $f^*_{\text{KL}}(y) = e^{-1} \exp{(y)}$, so so

$$
\begin{aligned}
\text{Sol}(S) &= \operatorname*{arg\,min}_{x \in \mathbb{R}} \left\{ x \left(\sum_{i \in S} \sigma_i \right) + \frac{\nu}{ne} \sum_{i \in S} \exp\left(l_{\pi(i)}/\nu \right) \exp\left(-x/\nu \right) \right\} \\
&= \nu \left[\ln \sum_{i \in S} \exp\left(l_{\pi(i)}/\nu \right) - \ln \sum_{i \in S} \sigma_i - \ln n - 1 \right] \\
\text{Sol}(S \cup T) &= \nu \left[\ln \sum_{i \in S \cup T} \exp\left(l_{\pi(i)}/\nu \right) - \ln \sum_{i \in S \cup T} \sigma_i - \ln n - 1 \right] \\
&= \nu \left[\ln \left(\sum_{i \in S} \exp\left(l_{\pi(i)}/\nu \right) + \sum_{i \in T} \exp\left(l_{\pi(i)}/\nu \right) \right) - \ln \left(\sum_{i \in S} \sigma_i + \sum_{i \in T} \sigma_i \right) - \ln n - 1 \right].
\end{aligned}
$$

Here, we carry the metadata $m(S) = (\ln \sum_{i \in S} \exp\left(l_{\pi(i)}/\nu \right), \ln \sum_{i \in S} \sigma_i)$, which can easily be combined and plugged into the function

$$
(m_1, m_2), (m_1', m_2') \mapsto \nu \left[\ln\left(\exp m_1 + \exp m_1' \right) - \ln\left(\exp m_2 + \exp m_2' \right) - \ln n - 1 \right]. \tag{20}
$$

for two instances of metadata (m_1, m_2) and (m_1', m_2'). We carry the "logsumexp" instead of just the sum of exponential quantities for numerical stability, and Equation (20) applies this operation as well. It might be that $\sum_{i \in S} \sigma_i = 0$, e.g. for the superquantile. In this case, we may interpret $\text{Sol}(S) = -\infty$ and evaluate $\exp{(-\infty)} = 0$ in the conversion formula (18). Two examples of the PAV algorithm are given in Algorithm 4 and Algorithm 5, respectively. These operate by selecting the unique values of the optimizer and partitions of indices that achieve that value.

Hardware Acceleration. Finally, note that all of the subroutines in Algorithm 3 (Algorithm 4/Algorithm 5, Algorithm 7, and Algorithm 7) all require primitive operations such as control flow and linear scans through vectors. Because these steps are outside of the purview of oracle calls or matrix multiplications, they benefit from just-in-time compilation on the CPU. We accelerate these subroutines using the Numba package in Python and are able to achieve an approximate 50%-60% decrease in runtime across benchmarks.

D. Convergence Analysis of SpecSAGA

Denote the quantity $\Omega_f(q) = D_f\left(q \| \mathbf{1}_n/n \right)$ for the f-divergence of the distribution $q \in \mathcal{P}(\sigma)$ to the uniform distribution and define

$$
r_i(w) = \ell_i(w) + \frac{\mu}{2} \|w\|_2^2, \quad r(w) = (r_i(w))_{i=1}^n \in \mathbb{R}^n.
$$

Our objective of interest can be rewritten

$$
\begin{aligned}
\mathcal{L}_\sigma(w) &= \max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w) - \nu D_f(q \| \mathbf{1}_n/n) + \frac{\mu}{2} \|w\|_2^2 && (21) \\
&= \max_{q \in \mathcal{P}(\sigma)} q^\top r(w) - \nu \Omega_f(q), && (22)
\end{aligned}
$$

which we wish to minimize in $w \in \mathbb{R}^d$. In the main text, we primarily considered $f(x) := x^2 - 1$, generating the χ^2-divergence. We analyze Algorithm 8, which adopts similar notation to Algorithm 3 in Appx. C. That is, at iterate t we have access to a table of losses $l^{(t)} \in \mathbb{R}^n$, (regularized) gradients $g^{(t)} \in \mathbb{R}^{n \times d}$, control variate weights $\rho^{(t)} \in \mathbb{R}^n$, and aggregate $\bar{g}^{(t)} = \sum_{i=1}^n \rho_i^{(t)} g_i^{(t)}$, and maximizer $q^{(t)} = q^{\text{opt}}\left(l^{(t)} \right)$. In addition, however, we keep track of the iterates $z_i^{(t)}$ at which losses and gradients are loaded into the table, so that $l_i^{(t)} = \ell_i(z_i^{(t)})$ and $g_i^{(t)} = \nabla r_i(z_i^{(t)})$. We emphasize that these iterates need not be stored and the presentation is made to clarify steps in the analysis.

D.1. Convergence Analysis for Large Shift Cost

Assumption 9. *We consider the losses $\ell_1, \ldots, \ell_n$ to each be convex, G-Lipschitz continuous, and L-smooth. Let f be α_n-strongly convex on $[0, n]$.*

Algorithm 3 SpecSAGA (Efficient)

```
Inputs: Initial points $w^{(0)}$, spectrum $\sigma$, stepsize $\eta > 0$, number of iterations $T$, regularization parameter $\mu > 0$, shift cost $\nu > 0$, loss and gradient oracles $\ell_1, \ldots, \ell_n$ and $\nabla \ell_1, \ldots, \nabla \ell_n$.
$l^{(0)} = (\ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^n$.
$g^{(0)} = (\nabla \ell_i(w^{(0)}) + \mu w^{(0)})_{i=1}^n \in \mathbb{R}^{n \times d}$.
$\pi^{(0)} = \mathrm{argsort}(l^{(0)})$.
$c^{(0)} = \mathrm{PAV}(l^{(0)}, \pi^{(0)}, \sigma)$.                    ▷ Algorithm 4 or Algorithm 5
$q^{(0)} = \mathrm{Convert}(c^{(0)}, l^{(0)}, \pi^{(0)}, \nu, c^{(0)})$.                ▷ Algorithm 7
Define $q^{(0)}$ by $q^{(0)}_{\pi^{(0)}(i)} = c_i$.
$\rho^{(0)} = q^{(0)}$.
$\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$.
for $t = 0, \ldots, T-1$ do
    Sample $i_t \sim \mathrm{Unif}[n]$.
    $v^{(t)} = n q_{i_t}^{(t)} \nabla \ell_{i_t}(w^{(t)}) - n \rho_{i_t}^{(t)} g_{i_t}^{(t)} - \bar{g}^{(t)}$.
    $w^{(t+1)} = (1 - \eta\mu) w^{(t)} - \eta v^{(t)}$.
    $l_{i_t}^{(t+1)} = \ell_{i_t}(w^{(t)})$ and $l_i^{(t+1)} = l_i^{(t)}$ for $i \neq i_t$.
    $g_{i_t}^{(t+1)} = \nabla \ell_{i_t}(w^{(t)}) + \mu w^{(t)}$ and $g_i^{(t+1)} = g_i^{(t)}$ for $i \neq i_t$.
    $\pi^{(t+1)} = \mathrm{Bubble}(\pi^{(t)}, l^{(t+1)})$.                ▷ Algorithm 6
    $\bar{g}^{(t+1)} = \bar{g}^{(t)} - \rho_{i_t}^{(t)} g_{i_t} + \rho_{i_t}^{(t+1)} g_{i_t}^{(t+1)}$.
    $c^{(t+1)} = \mathrm{PAV}(l^{(t+1)}, \pi^{(t+1)}, \sigma)$.
    $q^{(t+1)} = \mathrm{Convert}(c^{(t+1)}, l^{(t+1)}, \pi^{(t+1)}, \nu, q^{(t)})$.
    $\rho_{i_t}^{(t+1)} = \sigma_{i_t}^{(t+1)}$.
Output: Final point $w^{(T)}$.
```

The assumptions on losses ℓ_i are standard, whereas the second is satisfied by the χ^2 generator with $\alpha_n = 2$ and the KL generator with $\alpha_n = 1/n$.

Let $M = L + \mu$ be the smoothness constant of the regularized losses r_i. Moreover, we write $\mathbb{E}_t[\cdot]$ to denote the expectation conditioned on all the randomness up to iteration t (equivalently, conditioned on $w^{(t)}$, so that the only additional randomness is w.r.t. i_t). Recall that $\kappa_\sigma = n\sigma_n \geq 1$. Finally, define $q^\star = q^{\mathrm{opt}}(\ell(w^\star))$ as the value of the dual variable at optimum.

We have the following guarantee in the large shift cost regime.

Theorem 10. *Suppose the shift cost satisfies $\nu \geq \frac{12G^2}{\mu\alpha_n}$ Then, the sequence of iterates $(w^{(t)})$ produced by SpecSAGA*

Algorithm 4 Pool Adjacent Violators (PAV) Algorithm for χ^2 divergence

```
Inputs: Losses $(\ell_i)_{i \in [n]}$, argsort $\pi$, and spectrum $(\sigma_i)_{i \in [n]}$.
Initialize partition endpoints $(b_0, b_1) = (0, 1)$, partition value $v_1 = 2n/\nu\sigma_1 - l_{\pi(1)}$, number of parts $k = 1$.
for $i = 2, \ldots, n$ do
    Add part $k = k + 1$.
    Compute $v_{k+1} = 2n/\nu\sigma_i - l_{\pi(i)}$.
    while $k \geq 2$ and $v_k \geq v_{k+1}$ do
        $v_k = \frac{(b_k - b_{k-1})v_k + (i - b_k)v_d}{b_k - b_{k-1} + 1}$.
        Set $k = k - 1$.
    $b_k = i$.
Output: Vector $c$ containing $c_i = v_k$ for $b_{k-1} < i \leq b_k$.
```

Algorithm 5 Pool Adjacent Violators (PAV) Algorithm for KL divergence

```
**Inputs:** Losses $(\ell_i)_{i\in[n]}$, argsort $\pi$, and spectrum $(\sigma_i)_{i\in[n]}$.
Initialize partition endpoints $(b_0, b_1) = (0, 1)$, number of parts $k = 1$.
Initialize partition value $v_1 = \nu\left(l_{\pi(1)}/\nu - \ln\sigma_1 - \ln n - 1\right)$.
Initialize metadata $m_1 = \ell_{\pi(1)}/\nu$ and $t_1 = \ln\sigma_1$.
**for** $i = 2, \ldots, n$ **do**
    Add part $k = k + 1$.
    Compute $v_{k+1} = \nu\left(l_{\pi(i)}/\nu - \ln\sigma_i - \ln n - 1\right)$.
    Compute $m_{k+1} = \ell_{\pi(i)}/\nu)$ and $t_{k+1} = \ln\sigma_i$
    **while** $k \geq 2$ and $v_k \geq v_{k+1}$ **do**
        $m_k = \text{logsumexp}(m_k, m_{k+1})$ and $t_k = \text{logsumexp}(t_k, t_{k+1})$.
        $v_k = \nu\left(m_k - t_k - \ln n - 1\right)$.
        Set $k = k - 1$.
    $b_k = i$.
**Output:** Vector $c$ containing $c_i = v_k$ for $b_{k-1} < i \leq b_k$.
```

Algorithm 6 Bubble

```
**Require:** Index $j_{\text{init}}$, sorting permutation $\pi$, loss table $l$.
$j = j_{\text{init}}$.                                   ▷ If $l_{\pi(j_{\text{init}})}$ too small, bubble left.
**while** $j > 1$ and $l_{\pi(j)} < l_{\pi(j-1)}$ **do**
    Swap $\pi(j)$ and $\pi(j-1)$.
$j = j_{\text{init}}$.                                   ▷ If $l_{\pi(j_{\text{init}})}$ too large, bubble right.
**while** $j < n$ and $l_{\pi(j)} > l_{\pi(j+1)}$ **do**
    Swap $\pi(j)$ and $\pi(j+1)$.
**return** $\pi$
```

(Algorithm 8) with learning rate $\eta = (6M(1+\kappa^{-1})\kappa_\sigma)^{-1}$, $\kappa_\sigma = n\sigma_n$, $\kappa = M/\mu$, satisfies

$$\mathbb{E}\left\|w^{(t)} - w^\star\right\|_2^2 \leq \exp(-t/\tau)\left(n^2\left\|\sigma\right\|_2^2 + n\right)\left\|w^{(0)} - w^\star\right\|_2^2,$$

where we have defined

$$\tau = \max\left\{2n, 12\kappa_\sigma(\kappa+1)^2/\kappa\right\}.$$

The smoothness condition can be tightened to

$$\nu \geq \frac{2G(1+\kappa)}{M\alpha_n}\sqrt{\frac{\|\nabla\ell(w^\star)\|_2^2}{n} + \frac{8G^2}{\kappa_\sigma(1+\kappa)}},$$

as we demonstrate in Prop. 15. Here, $\|\nabla\ell(w^\star)\|_2^2$ is the squared spectral norm of the Jacobian of ℓ at $w^\star$. We proceed in several steps to complete the proof.

Evolution of Iterates. By expanding out the updates, we have,

$$\mathbb{E}_t\|w^{(t+1)} - w^\star\|^2 \leq \|w^{(t)} - w^\star\|_2^2 - 2\eta\left\langle\sum_{i=1}^n q_i^{(t)}\nabla r_i(w^{(t)}), w^{(t)} - w^\star\right\rangle + \eta^2\mathbb{E}_t\left[\|v^{(t)}\|_2^2\right]. \tag{23}$$

We analyze each of the two terms in turn.

Analysis of the 1st Order Term. We first analyze the first-order term with a particular focus on the bias. This is achieved in Lem. 13.

We first give two the helper lemmas.

Algorithm 7 Convert

```
Require: Sorted vector $c \in \mathbb{R}$, vector $l \in \mathbb{R}^n$, argsort $\pi$ of $l$, shift cost $\nu \geq 0$, vector $q \in \mathbb{R}^n$.
for $i = 1, \ldots, n$ do
    Set $q_{\pi(i)} = (1/n)[f^*]'\big((l_{\pi(i)} - c_i)/\nu\big)$.
return $q$.
```

Algorithm 8 SpecSAGA: Verbose

```
Inputs: Initial points $w^{(0)}$, stepsize $\eta > 0$, number of iterations $T$
Set $z_i^{(0)} = w^{(0)}$ for all $i \in [n]$, $q^{(0)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w^{(0)}) - \nu D_f(q \| \mathbf{1}_n/n)$, $\rho^{(0)} = q^{(0)}$.
Set $l^{(0)} = (\ell_i(z_i^{(0)}))_{i=1}^n \in \mathbb{R}^n$, $g^{(0)} = (\nabla r_i(z_i^{(0)}))_{i=1}^n \in \mathbb{R}^{d \times n}$, $\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$.
for $t = 0, \ldots, T-1$ do
    $i_t \sim \mathrm{Unif}([n])$.
    $z_{i_t}^{(t+1)} = w^{(t)}$ and $z_i^{(t+1)} = z_i^{(t)}$ for $i \neq i_t$.
    $\rho_{i_t}^{(t+1)} = q_{i_t}^{(t)}$ and $\rho_i^{(t+1)} = \rho_i^{(t)}$ for $i \neq i_t$.
    $v^{(t)} = n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - (n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - \bar{g}^{(t)})$.
    $w^{(t+1)} = w^{(t)} - \eta v^{(t)}$.
    $l^{(t+1)} = \ell(z^{(t+1)})$ .
    $q^{(t+1)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(t+1)} - \nu D_f(q \| \mathbf{1}_n/n)$.
    $g^{(t+1)} = (\nabla r_i(z^{(t+1)}))_{i=1}^n$
    $\bar{g}^{(t+1)} = \bar{g}^{(t)} + (\rho_{i_t}^{(t+1)} g_{i_t}^{(t+1)} - \rho_{i_t}^{(t)} g_{i_t}^{(t)}) = \sum_{i=1}^n \rho_i^{(t+1)} g_i^{(t+1)}$.
Output: Final point $w^{(T)}$.
```

Lemma 11. *For any $q \in \mathcal{P}(\sigma)$ and $w, v \in \mathbb{R}^d$, we have for r_i that is M-smooth and μ-strongly convex, we have*

$$
\begin{aligned}
(\nabla(q^\top r)(w) - \nabla(q^\top r)(v))^\top (w - v) \geq & \frac{\mu M}{\mu + M} \|w - v\|_2^2 \\
& + \frac{1}{(M + \mu)\sigma_n} \sum_{i=1}^n \|q_i \nabla r_i(w) - q_i \nabla r_i(v)\|_2^2.
\end{aligned}
$$

Proof. Consider $i \in [n]$ with $q_i \neq 0$, we have since $q_i r_i$ is $q_i M$-smooth and $q_i \mu$ strongly convex, we have by Thm. 40 that

$$
\begin{aligned}
(q_i \nabla r_i(w) - q_i \nabla r_i(v))^\top (w - v) &\geq \frac{q_i \mu M}{\mu + M} \|w - v\|_2^2 + \frac{1}{(M + \mu) q_i} \|q_i \nabla r_i(w) - q_i \nabla r_i(v)\|_2^2 \\
&\geq \frac{q_i \mu M}{\mu + M} \|w - v\|_2^2 + \frac{1}{(M + \mu)\sigma_n} \|q_i \nabla r_i(w) - q_i \nabla r_i(v)\|_2^2.
\end{aligned}
$$

The last inequality holds naturally for $q_i = 0$. Summing the above and using that $q^\top \mathbf{1} = 1$ since $q \in \mathcal{P}(\sigma)$ gives the result. □

Lemma 12. *Consider the setting of Algorithm 8. We have,*

$$
\frac{1}{n} \sum_{i=1}^n \left\| n(q_i^{(t)} - q_i^*) \nabla r_i(w^\star) \right\|_2^2 \leq 4nG^2 \left\| q_i^{(t)} - q_i^* \right\|_2^2 .
$$

Proof. Since $\nabla r_i(w^\star) = \nabla \ell_i(w^\star) + \mu w^\star$, we have $\|r_i(w^\star)\|_2^2 \leq 2G^2 + 2\|\mu w^\star\|_2^2$. Further, since $\sum_{i=1}^n q_i^\star \nabla r_i(w^\star) = 0$, we have that $\mu w^\star = -\sum_{i=1}^n q_i^* \nabla \ell_i(w^\star)$. Since $\|\nabla \ell_i(w^\star)\|_2 \leq G$, we have that $\|\mu w^\star\|_2 \leq G$, as it is a convex combination of vectors bounded by G. Therefore, we get,

$$\begin{aligned}\|n(q_i^{(t)} - q_i^*)\nabla r_i(w^\star)\|_2^2 &\leq 2n^2(q_i^{(t)} - q_i^*)^2(G^2 + \|\mu w^\star\|_2^2) \\ &\leq 4n^2G^2(q_i^{(t)} - q_i^*)^2\,.\end{aligned}$$

□

The following gives the bound on the first-order descent term. Define $\gamma_* = \|\nabla \ell(w^\star)\|_2$ to be the spectral norm of the Jacobian of ℓ at $w^\star$.

Lemma 13. *Consider the setting of Algorithm 8. For any $a > 0$, we have,*

$$\begin{aligned}-2\nabla({q^{(t)}}^\top r)(w^{(t)})^\top(w^{(t)} - w^\star) \leq &-\frac{2\mu M}{\mu + M}\|w^{(t)} - w^\star\|_2^2 \\ &- \frac{1}{(\mu + M)\kappa_\sigma}\mathbb{E}_t\left[\|nq_{i_t}^{(t)}\nabla r_i(w^{(t)}) - nq_{i_t}^*\nabla r_{i_t}(w^\star)\|_2^2\right] \\ &+ a^{-1}\|w^{(t)} - w^\star\|_2^2 \\ &+ \left(a\gamma_*^2 + \frac{8nG^2}{(\mu + M)\kappa_\sigma}\right)\frac{G^2}{n^2\alpha_n^2\nu^2}\sum_{i=1}^n \|z_i^{(t)} - w^\star\|_2^2.\end{aligned}$$

Proof. We have using Lem. 11

$$\begin{aligned}-\nabla({q^{(t)}}^\top r)(w^{(t)})^\top(w^{(t)} - w^\star) \leq &-\frac{\mu M}{\mu + M}\|w^{(t)} - w^\star\|_2^2 \\ &- \frac{1}{(\mu + M)n\sigma_n}\mathbb{E}_t\left[\|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^{(t)}\nabla r_i(w^\star)\|_2^2\right] \\ &- \nabla({q^{(t)}}^\top r)(w^\star)^\top(w^{(t)} - w^\star).\end{aligned} \tag{24}$$

Using $\|a + b\|_2^2 \leq 2\,\|a\|_2^2 + 2\,\|b\|_2^2$, the second term in (24) can be bounded as

$$\begin{aligned}&-\frac{1}{n}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^{(t)}\nabla r_i(w^\star)\|_2^2 \\ &\qquad\leq -\frac{1}{2n}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2 + \frac{1}{n}\sum_{i=1}^n \|n(q_i^{(t)} - q_i^*)\nabla r_i(w^\star)\|_2^2 \\ &\qquad\leq -\frac{1}{2n}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2 + 4nG^2\left\|q^{(t)} - q^\star\right\|_2^2\,,\end{aligned} \tag{25}$$

where the final inequality follows from Lem. 12. The last term in (24) is a bias term that is zero in the case that $\sigma = \mathbf{1}_n/n$. It can be bounded as

$$\begin{aligned}\nabla({q^{(t)}}^\top r)(w^\star)^\top(w^{(t)} - w^\star) &= (\nabla({q^{(t)}}^\top r)(w^\star) - \nabla({q^\star}^\top r)(w^\star))^\top(w^{(t)} - w^\star) \\ &= (\nabla({q^{(t)}}^\top \ell)(w^\star) - \nabla({q^\star}^\top \ell)(w^\star))^\top(w^{(t)} - w^\star) \\ &\leq \frac{a}{2}\|\nabla\ell(w^\star)^\top(q^{(t)} - q^\star)\|_2^2 + \frac{1}{2a}\|w^{(t)} - w^\star\|_2^2 \\ &\leq \frac{a\gamma_*^2}{2}\|q^{(t)} - q^\star\|_2^2 + \frac{1}{2a}\|w^{(t)} - w^\star\|_2^2,\end{aligned}$$

where we used Young's inequality, $2x^\top y \leq a\|x\|_2^2 + a^{-1}\|y\|_2^2$ for the first inequality, and γ_*^2 is the largest singular value of $\nabla \ell(w^\star) \in \mathbb{R}^{n \times d}$. Putting these together, we get

$$
\begin{aligned}
-\nabla({q^{(t)}}^\top r)(w^{(t)})^\top (w^{(t)} - w^\star) \leq & - \frac{\mu M}{\mu + M} \|w^{(t)} - w^\star\|_2^2 \\
& - \frac{1}{2(\mu + M)\kappa_\sigma} \mathbb{E}_t \left[\|n q_{i_t}^{(t)} \nabla r_i(w^{(t)}) - n q_{i_t}^* \nabla r_i(w^\star)\|_2^2 \right] \\
& + \frac{1}{2a} \|w^{(t)} - w^\star\|_2^2 \\
& + \left(\frac{a\gamma_*^2}{2} + \frac{4nG^2}{(\mu + M)\kappa_\sigma} \right) \|q^{(t)} - q^\star\|_2^2.
\end{aligned}
$$

Now, applying Lem. 6, the result follows from

$$
\|q^{(t)} - q^\star\|_2^2 \leq \frac{1}{n^2 \alpha_n^2 \nu^2} \|l^{(t)} - \ell(w^\star)\|_2^2 \leq \frac{G^2}{n^2 \alpha_n^2 \nu^2} \sum_{i=1}^n \|z_i^{(t)} - w^\star\|_2^2. \tag{26}
$$

□

Analysis of the 2nd Order Term.

Lemma 14. *Consider the notations of Alg. 8, we have for any $\beta > 0$,*

$$
\begin{aligned}
\mathbb{E}_t \|v^{(t)}\|_2^2 \leq (1 + \beta) & \mathbb{E}_t \|n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star)\|_2^2 \\
& + (1 + \beta^{-1}) \mathbb{E}_t \|n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star)\|_2^2.
\end{aligned}
$$

Proof. In the following, we use the identity $\mathbb{E}\|X - E[X]\|_2^2 = \mathbb{E}\|X\|_2^2 - \|\mathbb{E}[X]\|_2^2$ in equations denoted with $(\star)$. We denote by β an arbitrary positive number stemming from using Young's inequality $\|a+b\|_2^2 \leq (1+\beta)\|a\|_2^2 + (1+\beta^{-1})\|b\|_2^2$ in equation $(\circ)$. Noting that $\sum_{i=1}^n q_i^\star \nabla r_i(w^\star) = 0$, we get,

$$
\begin{aligned}
& \mathbb{E}_t \left[\|v^{(t)} - \nabla({q^*}^\top r)(w^\star)\|_2^2 \right] \\
& = \mathbb{E}_t \Big[\|n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star) \\
& \qquad + n q_{i_t}^* \nabla r_{i_t}(w^\star) - n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - (\nabla({q^\star}^\top r)(w^\star) - \bar{g}^{(t)})\|_2^2 \Big] \\
& \overset{(\star)}{=} \|\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^\star}^\top r)(w^\star)\|_2^2 \\
& \quad + \mathbb{E}_t \Big[\|n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star) - (\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^\star}^\top r)(w^\star)) \\
& \qquad + n q_{i_t}^* \nabla r_{i_t}(w^\star) - n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - (\nabla({q^\star}^\top r)(w^\star) - \bar{g}^{(t)})\|_2^2 \Big] \\
& \overset{(\circ)}{\leq} \|\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^\star}^\top r)(w^\star)\|_2^2 \\
& \quad + (1 + \beta) \mathbb{E}_t \left[\|n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star) - (\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^\star}^\top r)(w^\star))\|_2^2 \right] \\
& \quad + (1 + \beta^{-1}) \mathbb{E}_t \left[\|n q_{i_t}^* \nabla r_{i_t}(w^\star) - n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - (\nabla({q^\star}^\top r)(w^\star) - \bar{g}^{(t)})\|_2^2 \right] \\
& \overset{(\star)}{=} -\beta \|\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^\star}^\top r)(w^\star)\|_2^2 \\
& \quad + (1 + \beta) \mathbb{E}_t \left[\|n q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - n q_{i_t}^* \nabla r_{i_t}(w^\star)\|_2^2 \right] \\
& \quad + (1 + \beta^{-1}) \mathbb{E}_t \left[[\|n q_{i_t}^* \nabla r_{i_t}(w^\star) - n \rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)})\|_2^2 \right] \\
& \quad - (1 + \beta^{-1}) \|\nabla({q^\star}^\top r)(w^\star) - \bar{g}^{(t)}\|_2^2.
\end{aligned}
$$

□

Lyapunov Function. We use the Lyapunov function

$$V^{(t)} = \|w^{(t)} - w^\star\|_2^2 + c_1 T^{(t)} + c_2 S^{(t)}, \tag{27}$$

where c_1, c_2 are two constants to be defined later, and

$$T^{(t)} = \frac{1}{n}\sum_{i=1}^n \|n\rho_i^{(t)} \nabla r_i(z_i^{(t)}) - nq_i^* \nabla r_i(w^\star)\|_2^2, \quad S^{(t)} = \sum_{i=1}^n \|z_i^{(t)} - w^\star\|_2^2\,.$$

One-step update. The effect of one step of the algorithm is as follows.

Proposition 15. *Consider the setting of Thm. 10, that is,* $\nu \geq \frac{12G^2}{\mu\alpha_n}$*,* $\eta = (6M(1+\kappa^{-1})\kappa_\sigma)^{-1}$*. For* $c_1 = 3n\eta^2, c_2 = (12(\kappa^{-1}+1)\kappa_\sigma(\kappa+1))^{-1}$*, we have*

$$\mathbb{E}_t\left[V^{(t+1)}\right] \leq (1-\tau^{-1})V^{(t)}\,.$$

Proof. The last two terms of the Lyapunov function are easy to handle since

$$\mathbb{E}_t\left[T^{(t+1)}\right] = \frac{1}{n^2}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2 + \left(1-\frac{1}{n}\right)T^{(t)}$$

$$\mathbb{E}_t\left[S^{(t+1)}\right] = \|w^{(t)} - w^\star\|_2^2 + \left(1-\frac{1}{n}\right)S^{(t)}.$$

The evolution of the first term (23) of the Lyapunov function is given Lem. 13 and Lem. 14 (where we take $\beta = 2$). Plugging these in, we get for any $\tau > 1$,

$$\begin{aligned}\mathbb{E}_t\left[V^{(t+1)}\right] - (1-\tau^{-1})V^{(t)} \leq K_1 &\|w^{(t)} - w^\star\|_2^2 \\ &+ K_2\mathbb{E}_t\left[\|nq_{i_t}^{(t)}\nabla r_{i_t}(w^{(t)}) - nq_{i_t}^*\nabla r_{i_t}(w^\star)\|_2^2\right] \\ &+ K_3 c_1 T^{(t)} \\ &+ K_4 c_2 S^{(t)},\end{aligned}$$

with constants

$$\begin{aligned}K_1 &= \frac{1}{\tau} - \frac{2\eta\mu M}{\mu+M} + c_2 + \eta a^{-1}\,, \\ K_2 &= -\frac{\eta}{(\mu+M)\kappa_\sigma} + 3\eta^2 + \frac{c_1}{n}\,, \\ K_3 &= \frac{3\eta^2}{2c_1} + \frac{1}{\tau} - \frac{1}{n}\,, \\ K_4 &= \left(a\gamma_*^2 + \frac{8nG^2}{(\mu+M)\kappa_\sigma}\right)\frac{\eta G^2}{c_2 n^2\alpha_n^2\nu^2} + \frac{1}{\tau} - \frac{1}{n}\,.\end{aligned}$$

Our goal is to set the various free parameters so that all the coefficients $K_1, \ldots, K_4 \leq 0$. We enforce $\tau \geq 2n$ throughout. By setting

$$\eta = \frac{1}{6(\mu+M)\kappa_\sigma}, \quad \text{and} \quad c_1 = \frac{n\eta}{2(\mu+M)\kappa_\sigma},$$

we have $K_2 \leq 0, K_3 \leq 0$. By also requiring $\tau \geq 12(\kappa^{-1}+1)\kappa_\sigma(\kappa+1)$ and setting

$$a^{-1} = \frac{\mu M}{\mu+M}, \quad \text{and} \quad c_2 = \frac{1}{12(\kappa^{-1}+1)\kappa_\sigma(\kappa+1)},$$

we have $K_1 \leq 0$, as $(\mu M)/(\mu + M)^2 = (\kappa^{-1} + 1)^{-1}(\kappa + 1)^{-1}$. Turning to K_4 next, we get

$$\begin{aligned}
K_4 &= \left[\frac{1+\kappa}{M}\gamma_*^2 + \frac{8nG^2}{M(1+\kappa^{-1}\kappa_\sigma}\right] \frac{2G^2(1+\kappa)}{n^2\alpha_n^2\nu^2 M} + \frac{1}{\tau} - \frac{1}{n} \\
&\leq \frac{2G^2(1+\kappa)}{M^2 n^2 \alpha_n^2 \nu^2}\left[(1+\kappa)\gamma_*^2 + \frac{8nG^2}{\kappa_\sigma}\right] + \frac{1}{\tau} - \frac{1}{n} \\
&\leq \frac{2G^2(1+\kappa)}{M^2 n^2 \alpha_n^2 \nu^2}\left[(1+\kappa)\gamma_*^2 + \frac{8nG^2}{\kappa_\sigma}\right] - \frac{1}{2n},
\end{aligned}$$

which recovers the condition

$$\nu \geq \frac{2G(1+\kappa)}{M\alpha_n}\sqrt{\frac{\gamma_*^2}{n} + \frac{8G^2}{\kappa_\sigma(1+\kappa)}},$$

to achieve $K_4 \leq 0$, which is satisfied when $\nu \geq 12G^2/(\mu\alpha_n)$, completing the proof.

By loosening the condition on ν to

$$\nu \geq \frac{6G^2(1+\kappa)}{M\alpha_n},$$

by using that $\kappa \geq 1$, $\kappa_\sigma \geq 1$, and $\gamma_* \leq \sqrt{n}G$, we get a simpler dependence on the problem parameters. □

Next, we have a condition on the initial value of the Lyapunov function.

Proposition 16. *Consider the setting of Prop. 15. We have*

$$V^{(0)} \leq (1 + 2n + 2n^2)\left\|w^{(0)} - w^\star\right\|_2^2.$$

Proof. We use $c_2 \leq 1$ to bound $c_2 S^{(0)} \leq n\left\|w^{(0)} - w^\star\right\|_2^2$. For the second term, we have

$$\begin{aligned}
T^{(0)} &\leq \frac{2}{n}\sum_{i=1}^n \left\|nq_i^{(0)}(\nabla r_i(w^{(0)}) - \nabla r_i(w^\star))\right\|_2^2 + \frac{2}{n}\sum_{i=1}^n \left\|n(q_i^{(0)} - q_i^*)\nabla r_i(w^\star)\right\|_2^2 \\
&\leq 2n\sum_{i=1}^n (q_i^{(0)})^2 M^2 \left\|w^{(0)} - w^\star\right\|_2^2 + 8nG^2\left\|q^{(0)} - q^\star\right\|_2^2 \\
&\leq 2nM^2\left\|w^{(0)} - w^\star\right\|_2^2 + \frac{8G^4}{\alpha_n^2\nu^2}\left\|w^{(0)} - w^\star\right\|_2^2.
\end{aligned}$$

The first inequality above comes from $\|a+b\|_2^2 \leq 2\|a\|_2^2 + 2\|b\|_2^2$. The second inequality comes from using the M-Lipschitz property of each ∇r_i and Lem. 12. The third inequality comes from $\sum_i (q_i)^2 \leq \sum_i q_i = 1$ and for any $q \in \mathcal{P}(\sigma)$, and the $(n\alpha_n\nu)^{-1}$-Lipschitz property of $l \mapsto q^{\text{opt}}(l)$ (cf. (26)). Because $c_1 = n/(12(\mu + M)^2\kappa_\sigma)$, we have that for the first term,

$$c_1 2nM^2 = \frac{n^2}{6(1+\kappa^{-1})^2\kappa_\sigma^2} \leq 2n^2.$$

and for the second term

$$\frac{8c_1 G^4}{\alpha_n^2\nu^2} = \frac{2nG^4}{3\alpha_n^2\nu^2(\mu+M)^2\kappa_\sigma^2} \leq \frac{2G^4}{3\alpha_n^2\nu^2 M^2\sigma_n} \leq \frac{1}{54(1+\kappa)^2\sigma_n} \leq n.$$

where we used $\nu \geq 12G^2/(\mu\alpha_n) \geq 6G^2(1+\kappa)/(M\alpha_n)$ under the shift cost assumption. Finally, we have that

$$c_1 T^{(0)} \leq (1+n+2n^2) \left\| w^{(0)} - w^\star \right\|_2^2 .$$

□

Completing the Proof. We are now ready to complete the proof of Thm. 10.

Proof of Thm. 10. Let us choose c_1, c_2 as in Prop. 15. By using (a) Prop. 15, (b) $1 - x \leq \exp(-x)$, and (c) Prop. 16, we get

$$\begin{aligned} \mathbb{E} \left\| w^{(t)} - w^\star \right\|_2^2 \leq \mathbb{E}[V^{(t)}] &\overset{(a)}{\leq} (1-\tau^{-1})^t \, V^{(0)} \overset{(b)}{\leq} \exp(-t/\tau) V^{(0)} \\ &\overset{(c)}{\leq} \exp(-t/\tau)(1+2n+2n^2) \left\| w^{(0)} - w^\star \right\|_2^2 . \end{aligned}$$

□

D.2. Convergence Analysis for Alternate Norms

In this section, we provide details regarding an alternate analysis of Algorithm 8. In the previous section, we assume that the generator f is α_n-strongly convex on $[0, n]$, for which we do not need to specify a norm because f is a function on $\mathbb{R}$. Given this, we observe in Lem. 6 that $q \mapsto D_f(q \| \mathbf{1}_n/n)$ is strongly convex with respect to the ℓ_2-norm with constant $n\alpha$. However, we may also place the strong convexity condition directly on $q \mapsto D_f(q \| \mathbf{1}_n/n)$ for which other norms may be suitable for analysis. For example, when $f = f_{\text{KL}}$, we have that $D_f(\cdot \| \mathbf{1}_n/n)$ is 1-strongly convex with respect to the ℓ_1-norm. This section explore the effect of alternate norms on the convergence guarantee.

Notation. We define the constant

$$\gamma_{\star,p} := \|\nabla \ell(w^\star)\|_{2,p} := \max \left\{ \rho^\top \nabla \ell(w^\star) w \, : \, \|w\|_2 = 1, \|\rho\|_p = 1 \right\} . \tag{28}$$

Note that this implies for all $x \in \mathbb{R}^n$ that

$$\left\| \nabla \ell(w^\star)^\top x \right\|_2^2 \leq \|\nabla \ell(w^\star)\|_{2,p}^2 \, \|x\|_p^2 . \tag{29}$$

Theorem 17. *Suppose the f-divergence $q \mapsto D_f(q \, \| \mathbf{1}_n/n)$ is α-strongly convex w.r.t. the L_p norm $\|\cdot\|_p$ for some $p \in [1,2]$. Suppose the shift cost satisfies*

$$\nu \geq \frac{2\sqrt{n}G}{\alpha\mu} \left((1+\kappa^{-1})\gamma_{\star,p} + G\sqrt{\frac{8n}{\kappa\kappa_\sigma}} \right) .$$

Then, the sequence of iterates $(w^{(t)})$ produced by SpecSAGA (Algorithm 8) with learning rate $\eta = (6M(1+\kappa^{-1})\kappa_\sigma)^{-1}$ satisfies

$$\mathbb{E} \left\| w^{(t)} - w^\star \right\|_2^2 \leq \exp(-t/\tau) \, (1+n+3n^2) \left\| w^{(0)} - w^\star \right\|_2^2 ,$$

where we have defined

$$\tau = \max \left\{ 2n, 12(\kappa^{-1}+1)\kappa_\sigma(\kappa+1) \right\} .$$

The proof of this result proceeds similarly to that Thm. 10. We highlight the key differences.

First-order term.

Lemma 18 (Counterpart of Lem. 13)**.** *Consider the setting of Algorithm 8. For any $a > 0$, we have,*

$$\begin{aligned}
-2\nabla(q^{(t)\top} r)(w^{(t)})^\top (w^{(t)} - w^\star) \le &- \frac{2\mu M}{\mu + M}\|w^{(t)} - w^\star\|_2^2 \\
&- \frac{1}{(\mu+M)\kappa_\sigma}\mathbb{E}_t\left[\|nq_{i_t}^{(t)}\nabla r_{i_t}(w^{(t)}) - nq_{i_t}^*\nabla r_i(w^\star)\|_2^2\right] \\
&+ a^{-1}\|w^{(t)} - w^\star\|_2^2 \\
&+ \left(a\gamma_{\star,p}^2 + \frac{8nG^2}{(\mu+M)\kappa_\sigma}\right)\frac{nG^2}{\alpha^2\nu^2}\sum_{i=1}^n \|z_i^{(t)} - w^\star\|_2^2.
\end{aligned}$$

Proof. The proof proceeds similarly to Lem. 13, starting with (24). The last term in (24) can be bounded as

$$\begin{aligned}
\nabla(q^{(t)\top} r)(w^\star)^\top(w^{(t)} - w^\star) &\le \frac{a}{2}\|\nabla\ell(w^\star)^\top(q^{(t)} - q^*)\|_2^2 + \frac{1}{2a}\|w^{(t)} - w^\star\|_2^2 \\
&\le \frac{a\gamma_{\star,p}^2}{2}\|q^{(t)} - q^*\|_p^2 + \frac{1}{2a}\|w^{(t)} - w^\star\|_2^2,
\end{aligned}$$

where we used (29) in the last inequality.

For the 2nd term in (24), we start with (25) and further use $\|x\|_2 \le \|x\|_p$ for $p \in [1, 2]$ to get

$$\begin{aligned}
&-\frac{1}{n}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^{(t)}\nabla r_i(w^\star)\|_2^2 \\
&\qquad \le -\frac{1}{2n}\sum_{i=1}^n \|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2 + 4nG^2\left\|q_i^{(t)} - q_i^*\right\|_p^2 .
\end{aligned}$$

Putting these together and invoking Lem. 7 instead of (26) completes the proof. □

One-step update. The effect of one step of the algorithm is as follows.

Proposition 19 (Counterpart of Prop. 15)**.** *Consider the setting of Thm. 17. For $c_1 = 3n\eta^2, c_2 = (12(\kappa^{-1} + 1)\kappa_\sigma(\kappa + 1))^{-1}$, we have*

$$\mathbb{E}_t\left[V^{(t+1)}\right] \le (1 - \tau^{-1})V^{(t)} .$$

Proof. The proof is identical to that of Prop. 15, except that we have

$$K_4 = \left(a\gamma_{\star,p}^2 + \frac{8nG^2}{(\mu+M)\kappa_\sigma}\right)\frac{\eta G^4}{c_2\alpha^2\nu^2} + \frac{1}{\tau} - \frac{1}{n} .$$

Together with the other parameter choices, we have $K_4 \le 0$ as long as

$$\nu^2 \ge \frac{4nG^2}{\alpha^2\mu^2}\left((1 + \kappa^{-1})^2\gamma_{\star,p}^2 + \frac{8nG^2}{\kappa\kappa_\sigma}\right) .$$

The claimed condition on ν is sufficient to ensure that this is the case given the subadditivity of the square root. □

D.3. Convergence Analysis for Any Shift Cost

We consider in this section a slight variation of Alg. 1 with decoupled samplings of the losses and the gradients presented in Alg. 9. A detailed version in which we keep track of the variables $z_i^{(t)}$, $\zeta_i^{(t)}$ at which gradients and losses are taken respectively is presented in Alg. 10.

For simplicity, we use the shorthand

$$\bar\nu = 2n\nu.$$

Algorithm 9 Decoupled SpecSAGA

```
**Inputs:** Initial points $w^{(0)}$, stepsize $\eta > 0$, number of iterations $T$
Set $l^{(0)} = (\ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^n$, $g^{(0)} = (\nabla r_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^{d \times n}$, $\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$
Initialize $q^{(0)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(0)} - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2$, $\rho^{(0)} = q^{(0)}$
**for** $t = 0, \dots, T-1$ **do**
    $i_t \sim \mathrm{Unif}([n]), j_t \sim \mathrm{Unif}([n])$
    $v^{(t)} = nq_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - (n\rho_{i_t}^{(t)} g_{i_t} - \bar{g}^{(t)})$
    $w^{(t+1)} = w^{(t)} - \eta v^{(t)}$
    $l_{j_t}^{(t+1)} = \ell_{j_t}(w^{(t)})$ and $l_j^{(t+1)} = l_j^{(t)}$ for $j \neq j_t$.
    $q^{(t+1)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(t+1)} - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2$
    $\rho_{i_t}^{(t+1)} = q_{i_t}^{(t)}$ and $\rho_i^{(t+1)} = \rho_i^{(t)}$ for $i \neq i_t$
    $g_{i_t}^{(t+1)} = \nabla r_{i_t}(w^{(t)})$ and $g_i^{(t+1)} = g_i^{(t)}$ for $i \neq i_t$
    $\bar{g}^{(t+1)} = \bar{g}^{(t)} + (q_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - \rho_{i_t}^{(t)} g_{i_t}^{(t)})$
**Output:** Final point $w^{(T)}$
```

Algorithm 10 Decoupled SpecSAGA: Detailed Version for the Proof

```
**Inputs:** Initial points $w^{(0)}$, stepsize $\eta > 0$, number of iterations $T$
Set $z_i^{(0)} = \zeta_i^{(0)} = w^{(0)}$ for all $i \in [n]$, $q^{(0)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w^{(0)}) - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2$, $\rho^{(0)} = q^{(0)}$
Set $l^{(0)} = (\ell_i(\zeta_i^{(0)}))_{i=1}^n \in \mathbb{R}^n$, $g^{(0)} = (\nabla r_i(z_i^{(0)}))_{i=1}^n \in \mathbb{R}^{d \times n}$, $\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$
**for** $t = 0, \dots, T-1$ **do**
    $i_t \sim \mathrm{Unif}([n]), j_t \sim \mathrm{Unif}([n])$
    $z_{i_t}^{(t+1)} = w^{(t)}$ and $z_i^{(t+1)} = z_i^{(t)}$ for $i \neq i_t$
    $\zeta_{j_t}^{(t+1)} = w^{(t)}$ and $\zeta_j^{(t+1)} = \zeta_j^{(t)}$ for $j \neq j_t$
    $\rho_{i_t}^{(t+1)} = q_{i_t}^{(t)}$ and $\rho_i^{(t+1)} = \rho_i^{(t)}$ for $i \neq i_t$
    $v^{(t)} = nq_{i_t}^{(t)} \nabla r_{i_t}(w^{(t)}) - (n\rho_{i_t}^{(t)} \nabla r_{i_t}(z_{i_t}^{(t)}) - \bar{g}^{(t)})$
    $w^{(t+1)} = w^{(t)} - \eta v^{(t)}$
    $l^{(t+1)} = \ell(\zeta^{(t+1)})$
    $q^{(t+1)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(t+1)} - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2$
    $g^{(t+1)} = (\nabla r_i(z^{(t+1)}))_{i=1}^n$
    $\bar{g}^{(t+1)} = \sum_{i=1}^n \rho_i^{(t+1)} g_i^{(t+1)}$
**Output:** Final point $w^{(T)}$
```

Convergence Statement. The convergence of Alg. 9 is presented in Thm. 20

Theorem 20. *Denote $\kappa_1 = 1 + L/\mu, \kappa_2 = G^2 n/(\bar{\nu}\mu), \kappa_3 = \kappa_\sigma(\kappa_1 + 1) + 8\kappa_2$, $\kappa_\sigma = n\sigma_{\max}$. Then, the sequence $(w^{(t)})$ produced by Alg. 9 with a learning rate $\eta \leq \max\left\{24\mu\kappa_3, 4(1 + 4n\kappa_2)\kappa_2\mu \max\{8\kappa_1^2, 2\kappa_2\}\right\}^{-1}$ converges linearly to the $w^\star$ at a rate $\tau = \max\{n, \mu\eta\}$.*

Convergence Proof. Recall that our problem is, for $\mu, \bar{\nu} > 0$, $\ell(w) = (\ell_i(w))_{i=1}^n$

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w) + \frac{\mu}{2}\|w\|_2^2 - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2. \tag{30}$$

where $\mathcal{P}(\sigma) \subseteq \Delta^n = \{p \in \mathbb{R}^n : p \geq 0, p^\top \mathbf{1} = 1\}$ with $\|q\|_\infty \leq \sigma_{\max}$ for all $q \in \mathcal{P}(\sigma)$ and $\sigma_{\max} \geq 1/n$. Denote $r_i(w) = \ell_i(w) + \frac{\mu}{2}\|w\|_2^2, r(w) = (r_i(w))_{i=1}^n \in \mathbb{R}^n$. Throughout the analysis we consider ℓ_i to be G-Lipschitz continuous and L-smooth.

In the following, we denote $M = L + \mu$ the smoothness constant of the regularized losses r_i. We denote $\mathbb{E}_t$ the expectation w.r.t to the randomness induced by picking i_t, j_t *given* $w^{(t)}$, i.e. the conditional expectation given the σ-algebra generated by $w^{(t)}$. The optimum of (30) is denoted $w^\star$ and satisfies

$$\nabla({q^\star}^\top r(w^\star)) = 0, \ \text{for } q^\star = \operatorname*{arg\,max}_{q \in \mathcal{P}(\sigma)} q^\top \ell(w^\star) - \frac{\bar\nu}{2}\|q - \mathbf{1}_n/n\|_2^2. \tag{31}$$

Evolution of the distance to the optimum. We start by analyzing the evolution of the distance to the solution given in Cor. 22 as a corollary of Lem. 21. Compared to the proof for large smoothing we use a cross-product term defined by the differences of weighted losses, that is, we have a term $(q-q^\star)^\top(\ell(w) - \ell(w^\star))$ that replaces the previous cross-product term.

Lemma 21. *Consider $w^\star \in \mathbb{R}^d$ the optimum of (30) satisfying (31) with associated $q^\star \in \mathcal{P}(\sigma)$. For any $w \in \mathbb{R}^d, l \in \mathbb{R}^n$, denoting $q = \operatorname{arg\,max}_{p \in \mathcal{P}(\sigma)} p^\top l - \frac{\bar\nu}{2}\|p - \mathbf{1}/n\|_2^2$, we have for any $\beta_1 \in [0,1]$,*

$$\begin{aligned}
&(q-q^\star)^\top(\ell(w) - \ell(w^\star)) - (\nabla(q^\top r)(w) - \nabla({q^\star}^\top r)(w^\star))^\top(w - w^\star)\\
&\leq -\frac{\mu}{2}\|w - w^\star\|_2^2 - \frac{\beta_1}{4(M+\mu)\sigma_{\max}}\sum_{i=1}^n \|q_i \nabla r_i(w) - q_i^* \nabla r_i(w^\star)\|_2^2\\
&\quad + \frac{2\beta_1 G^2}{\bar\nu(M+\mu)\sigma_{\max}}(q - q^\star)^\top(l - \ell(w^\star)).
\end{aligned}$$

Proof. From Lem. 27 we have that for any $\beta_1 \in [0,1]$,

$$\begin{aligned}
&(q-q^\star)^\top(r(w) - r(w^\star)) - (\nabla(q^\top r)(w) - \nabla({q^\star}^\top r)(w^\star))^\top(w - w^\star)\\
&\leq -\frac{\mu}{2}\|w - w^\star\|_2^2\\
&\quad - \frac{\beta_1}{2(M+\mu)\sigma_{\max}}\left(\sum_{i=1}^n \|q_i^* \nabla r_i(w) - q_i^* \nabla r_i(w^\star)\|_2^2 + \sum_{i=1}^n \|q_i \nabla r_i(w) - q_i \nabla r_i(w^\star)\|_2^2\right).
\end{aligned}$$

We have that

$$\|q_i \nabla r_i(w) - q_i^* \nabla r_i(w^\star)\|_2^2 \leq 2\|q_i \nabla r_i(w) - q_i \nabla r_i(w^\star)\|_2^2 + 2(q_i - q_i^*)^2\|\nabla r_i(w^\star)\|_2^2,$$

which is equivalent to

$$-\|q_i \nabla r_i(w) - q_i \nabla r_i(w^\star)\|_2^2 \leq -\frac{1}{2}\|q_i \nabla r_i(w) - q_i^* \nabla r_i(w^\star)\|_2^2 + (q_i - q_i^*)^2\|\nabla r_i(w^\star)\|_2^2.$$

We have that $\nabla r_i(w^\star) = \nabla \ell_i(w^\star) + \mu w^\star$. Moreover, since $\nabla({q^\star}^\top r)(w^\star) = 0$, we have $\mu w^\star = -\sum_{i=1}^n q_i^* \nabla \ell_i(w^\star)$. Hence, by Jensen's inequality using that $q^\star \in \Delta^n$, we have $\|\mu w^\star\|_2 \leq \sum_{i=1}^n q_i^* \|\nabla \ell_i(w^\star)\|_2 \leq G$ since ℓ_i is G-Lipschitz-continuous. We have then that $\|\nabla r_i(w^\star)\| \leq 2G$ and so

$$\begin{aligned}
&(q-q^\star)^\top(r(w) - r(w^\star)) - (\nabla(q^\top r)(w) - \nabla({q^\star}^\top r)(w^\star))^\top(w - w^\star)\\
&\leq -\frac{\mu}{2}\|w - w^\star\|_2^2\\
&\quad - \frac{\beta_1}{4(M+\mu)\sigma_{\max}}\sum_{i=1}^n \|q_i \nabla r_i(w) - q_i^* \nabla r_i(w^\star)\|_2^2 + \frac{2\beta_1 G^2}{(M+\mu)\sigma_{\max}}\|q - q^\star\|_2^2.
\end{aligned}$$

Since $q, q^\star \in \Delta^n$, we have

$$\begin{aligned}
(q-q^\star)^\top(r(w) - r(w^\star)) &= (q-q^\star)^\top\left(\ell(w) - \ell(w^\star) + \left(\frac{\mu}{2}\|w\|_2^2 - \frac{\mu}{2}\|w^\star\|_2^2\right)\mathbf{1}_n\right)\\
&= (q-q^\star)^\top(\ell(w) - \ell(w^\star)).
\end{aligned}$$

With the notations of Lem. 45, we have that $q = \nabla h(l)$ and $q^\star = \nabla h(\ell(w^\star))$. Hence, by Lem. 45, we have

$$\|q - q^\star\|_2^2 \leq \frac{1}{\bar{\nu}}(q - q^\star)^\top (l - \ell(w^\star)).$$

The result follows. □

Corollary 22. *Consider the setting of Alg. 10, we have for any $\beta_1 \in [0, 1]$, $\beta_2 > 0$, denoting $\kappa_\sigma = n\sigma_{\max}$ and $l^* = \ell(w^\star)$,*

$$\begin{aligned}
\mathbb{E}_t \left[\|w^{(t+1)} - w^\star\|^2\right] \leq{}& (1 - \eta\mu)\|w^{(t)} - w^\star\|_2^2 - 2\eta(q^{(t)} - q^\star)(\ell(w^{(t)}) - l^*) \\
&- \eta\left(\frac{\beta_1}{2(M + \mu)\kappa_\sigma} - \eta(1 + \beta_2)\right)\mathbb{E}_t\left[\|nq_i^{(t)}\nabla r_i(w^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2\right] \\
&+ \frac{4\eta\beta_1 G^2 n}{\bar{\nu}(M + \mu)\kappa_\sigma}(q^{(t)} - q^\star)^\top (l^{(t)} - l^*) \\
&+ \eta^2(1 + \beta_2^{-1})\mathbb{E}_t\left[\|n\rho_{i_t}^{(t)}\nabla r_{i_t}(z_{i_t}^{(t)}) - nq_{i_t}^*\nabla r_{i_t}(w^\star)\|_2^2\right]
\end{aligned}$$

Proof. We have

$$\mathbb{E}_t\left[\|w^{(t+1)} - w^\star\|^2\right] \leq \|w^{(t)} - w^\star\|_2^2 - 2\eta\nabla({q^{(t)}}^\top r)(w^{(t)})^\top (w^{(t)} - w^\star) + \eta^2\mathbb{E}_t\left[\|v^{(t)}\|_2^2\right].$$

We have using that the optimality conditions (31) of $w^\star$ and Lem. 21 that for any $\beta_1 \in [0, 1]$,

$$\begin{aligned}
-2\eta\nabla({q^{(t)}}^\top r)(w^{(t)})^\top (w^{(t)} - w^\star) &= -2\eta(\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla {q^\star}^\top r(w^\star))^\top (w^{(t)} - w^\star) \\
&\leq -2\eta(q^{(t)} - q^\star)(\ell(w^{(t)}) - \ell(w^\star)) - \eta\mu\|w^{(t)} - w^\star\|_2^2 \\
&\quad - \frac{\eta\beta_1}{2(M + \mu)\sigma_{\max}}\sum_{i=1}^n \|q_i^{(t)}\nabla r_i(w^{(t)}) - q_i^*\nabla r_i(w^\star)\|_2^2 \\
&\quad + \frac{4\eta\beta_1 G^2}{\bar{\nu}(M + \mu)\sigma_{\max}}(q^{(t)} - q^\star)^\top (l^{(t)} - \ell(w^\star)).
\end{aligned}$$

Combined with Lem. 14, we get the claim. □

Evolution of weighted losses. To incorporate the term $(q - q^\star)^\top (\ell(w) - \ell(w^\star))$ that appeared in the evolution of the distances, we consider analyzing the term $(q^{(t)} - q^\star)^\top (l^{(t)} - l^*)$ which, in expectation, partly incorporates $(q - q^\star)^\top (\ell(w) - \ell(w^\star))$ while introducing some additional terms from $\mathbb{E}_{j_t}\left[(q^{(t+1)} - q^{(t)})^\top (l^{(t+1)} - l^*)\right]$ that are further bounded by the distance of the current iterate to the solution or the checkpoints in the tables.

Lemma 23. *Consider the setting of Alg. 10. Consider*

$$R^{(t)} = 2\eta n(q^{(t)} - q^\star)^\top (l^{(t)} - l^*) \geq 0$$

for $l^* = \ell(w^\star)$. We have that

$$\begin{aligned}
\mathbb{E}_t\left[R^{(t+1)}\right] &\leq 2\eta(q^{(t)} - q^\star)^\top (\ell(w^{(t)}) - l^*) + \left(1 - \frac{1}{n}\right) R^{(t)} + \mathbb{E}_{j_t}\left[(q^{(t+1)} - q^{(t)})^\top (l^{(t+1)} - l^*)\right] \\
&\leq 2\eta(q^{(t)} - q^\star)(\ell(w^{(t)}) - l^*) + \left(1 - \frac{1}{n}\right) R^{(t)} \\
&\quad + \frac{2\eta G^2 n}{\bar{\nu}}(1 + \beta_3)\frac{1}{n}\sum_{j=1}^n \|w^{(t)} - \zeta_j^{(t)}\|_2^2 + \frac{\eta G^2 n}{2\bar{\nu}}\beta_3^{-1}\sum_{j=1}^n \|\zeta_j^{(t)} - w^\star\|_2^2.
\end{aligned}$$

Proof. The fact that $R^{(t)}$ is non-negative is a consequence of Lem. 45 using that $q^{(t)} = \nabla h(l^{(t)})$, $q^\star = \nabla h(\ell(w^\star))$ for h

defined in Lem. 45. We have for any $\beta_3 > 0$

$$\begin{aligned}(q^{(t+1)} - q^\star)^\top (l^{(t)} - l^*) = (q^{(t)} - q^\star)^\top (l^{(t+1)} - l^*) + (q^{(t+1)} - q^{(t)})^\top (l^{(t+1)} - l^{(t)}) \\ + (q^{(t+1)} - q^{(t)})^\top (l^{(t)} - l^*).\end{aligned}$$

With the notations of Lem. 45, we have $q^{(t)} = \nabla h(l^{(t)})$ and so

$$(q^{(t+1)} - q^{(t)})^\top (l^{(t+1)} - l^{(t)}) \le \frac{1}{\bar\nu} \|l^{(t+1)} - l^{(t)}\|_2^2.$$

Next, using Young's inequality, that is, $a^\top b \le \frac{\beta_3}{2}\|a\|_2^2 + \frac{\beta_3^{-1}}{2}\|b\|_2^2$ for any $\beta_3 > 0$, we have

$$\begin{aligned}(q^{(t+1)} - q^{(t)})^\top (l^{(t)} - l^*) &\le \frac{\beta_3}{2}\|q^{(t+1)} - q^{(t)}\|_2^2 + \frac{\beta_3^{-1}}{2}\|l^{(t)} - l^*\|_2^2 \\ &\le \frac{\beta_3}{2\bar\nu^2}\|l^{(t+1)} - l^{(t)}\|_2^2 + \frac{\beta_3^{-1}}{2}\|l^{(t)} - l^*\|_2^2.\end{aligned}$$

Note that we have

$$\mathbb{E}_{j_t}\left[l^{(t+1)}\right] = \frac{1}{n}\ell(w^{(t)}) + \left(1 - \frac{1}{n}\right) l^{(t)}.$$

Hence, we get, since $\mathbb{E}_t\left[R^{(t+1)}\right] = \mathbb{E}_{j_t}\left[R^{(t+1)}\right]$,

$$\begin{aligned}\frac{1}{2\eta n}\mathbb{E}_t\left[R^{(t+1)}\right] &\le \frac{1}{n}(q^{(t)} - q^\star)(\ell(w^{(t)}) - l^*) + \left(1 - \frac{1}{n}\right)(q^{(t)} - q^\star)^\top (l^{(t)} - l^*) \\ &\quad + \frac{1}{n\bar\nu}\left(1 + \frac{\beta_3}{2\bar\nu}\right)\sum_{j=1}^n (\ell_j(w^{(t)}) - \ell_j(\zeta_j))^2 \\ &\quad + \frac{\beta_3^{-1}}{2}\sum_{j=1}^n (\ell_j(\zeta_j) - \ell_j(w^\star))^2 \\ &\le \frac{1}{n}(q^{(t)} - q^\star)(\ell(w^{(t)}) - l^*) + \left(1 - \frac{1}{n}\right)(q^{(t)} - q^\star)^\top (l^{(t)} - l^*) \\ &\quad + \frac{G^2}{n\bar\nu}\left(1 + \frac{\beta_3}{2\bar\nu}\right)\sum_{j=1}^n \|w^{(t)} - \zeta_j^{(t)}\|_2^2 \\ &\quad + \frac{G^2\beta_3^{-1}}{2}\sum_{j=1}^n \|\zeta_j^{(t)} - w^\star\|_2^2.\end{aligned}$$

Replacing β_3 by $2\bar\nu\beta_3$ gives the claim. □

Lyapunov function and overall convergence. Lem. 23 introduces two terms $\frac{1}{n}\sum_{j=1}^n \|w^{(t)} - \zeta_j^{(t)}\|_2^2$ and $\sum_{j=1}^n \|\zeta_j^{(t)} - w^\star\|_2^2$ whose evolution can further be bounded by Lem. 26. We now define a Lyapunov function that incorporates all these terms with appropriate constants to show convergence of the algorithm.

Theorem 24. *Consider the setting of Alg. 10, and denote* $\kappa_1 = M/\mu, \kappa_2 = G^2 n/(\bar\nu\mu), \kappa_3 = \kappa_\sigma(\kappa_1 + 1) + 8\kappa_2$, $\kappa_\sigma = n\sigma_{\max}$. *Define*

$$V^{(t)} = \|w^{(t)} - w^\star\|_2^2 + R^{(t)} + c_1 S^{(t)} + c_2 T^{(t)} + c_3 U^{(t)}$$

where

$$R^{(t)} = 2\eta n(q^{(t)} - q^\star)^\top(l^{(t)} - l^*), \quad S^{(t)} = \frac{1}{n}\sum_{i=1}^n \|n\rho_i^{(t)}\nabla r_i(z_i^{(t)}) - nq_i^*\nabla r_i(w^\star)\|_2^2,$$

$$T^{(t)} = \sum_{i=1}^n \|\zeta_i^{(t)} - w^\star\|_2^2, \quad U^{(t)} = \frac{1}{n}\sum_{j=1}^n \|w^{(t)} - \zeta_j^{(t)}\|_2^2,$$

and $c_1 = (n\eta)/(4\mu\kappa_3)$, $c_2 = \eta\mu/2$, $c_3 = \max\{8\kappa_1^2, 2\kappa_2\}^{-1}$. *Choosing*

$$\eta \leq \max\left\{24\mu\kappa_3, 4(1+4n\kappa_2)\kappa_2\mu\max\{8\kappa_1^2, 2\kappa_2\}\right\}^{-1},$$

we have

$$\mathbb{E}_t\left[V^{(t+1)}\right] \leq (1-\tau^{-1})V^{(t)},$$

where

$$\tau \geq 2\max\left\{n, a_1^2\kappa_2, 2\kappa_2^2\}\right\}.$$

Proof. The evolution of the terms $S^{(t)}, T^{(t)}$ are simply given by

$$\mathbb{E}_t\left[S^{(t+1)}\right] = \frac{1}{n}Q^{(t)} + \left(1 - \frac{1}{n}\right)S^{(t)}, \quad \mathbb{E}_t\left[T^{(t+1)}\right] = \|w^{(t)} - w^\star\|_2^2 + \left(1 - \frac{1}{n}\right)T^{(t)}$$

for $Q^{(t)} = \mathbb{E}_t\left[\|nq_{i_t}^{(t)}\nabla r_{i_t}(w^{(t)}) - q_{i_t}^*\nabla r_{i_t}(w^\star)\|_2^2\right]$. For any $\tau > 1$, we have by combining Cor. 22, Lem. 23, Lem. 26, that

$$\begin{aligned}
&\mathbb{E}_t\left[V^{(t+1)}\right] - (1-\tau^{-1})V^{(t)} \\
&\leq K_1\|w^{(t)} - w^\star\|_2^2 + K_2R^{(t)} + K_3S^{(t)} + K_4T^{(t)} + K_5U^{(t)} + K_6Q^{(t)}
\end{aligned}$$

for

$$\begin{aligned}
K_1 &= \tau^{-1} - \eta\mu + c_2 \\
K_2 &= \tau^{-1} - \frac{1}{n} + \frac{2\beta_1 G^2}{\bar{\nu}(M+\mu)\kappa_\sigma} + \left(1 - \frac{1}{n}\right)\frac{c_3 G^2}{2\bar{\nu}\mu} \\
K_3 &= \tau^{-1} - \frac{1}{n} + \frac{\eta^2(1+\beta_2^{-1})(1+c_3)}{c_1} \\
K_4 &= \tau^{-1} - \frac{1}{n} + \frac{\eta n G^2\beta_3^{-1}}{2c_2\bar{\nu}} + \left(1 - \frac{1}{n}\right)\frac{c_3\eta M^2}{c_2\mu n} \\
K_5 &= \tau^{-1} - \frac{1}{n} + \frac{2\eta G^2 n}{c_3\bar{\nu}}(1+\beta_3) \\
K_6 &= -\eta\frac{\beta_1}{2(M+\mu)\kappa_\sigma} + \eta^2(1+\beta_2)(1+c_3) + \frac{c_1}{n}.
\end{aligned}$$

Taking

$$\begin{aligned}
\beta_1 &= \frac{(M+\mu)\kappa_\sigma}{8G^2 n/\bar{\nu} + (M+\mu)\kappa_\sigma} \in [0,1], \ \beta_2 = 2, \ \beta_3 = \frac{4n^2G^2}{\mu\bar{\nu}} \\
c_1 &= \frac{n\eta\beta_1}{4(M+\mu)\kappa_\sigma} = \frac{n\eta}{4(8G^2n/\bar{\nu} + (M+\mu)\kappa_\sigma)}, \ c_2 = \frac{\eta\mu}{2},
\end{aligned}$$

we get

$$
\begin{aligned}
K_1 &= \tau^{-1} - \frac{\eta\mu}{2} \\
K_2 &\leq \tau^{-1} - \frac{1}{n} + \frac{1}{4n} + \frac{c_3 G^2 n}{2\bar{\nu}\mu n} \\
K_3 &= \tau^{-1} - \frac{1}{n} + \frac{6\eta(1+c_3)(8G^2 n/\bar{\nu} + (M+\mu)\kappa_\sigma)}{n} \\
K_4 &= \tau^{-1} - \frac{1}{n} + \frac{1}{4n} + \frac{2c_3 M^2}{n\mu^2} \\
K_5 &= \tau^{-1} - \frac{1}{n} + \frac{2\eta G^2 n}{c_3 \bar{\nu}}\left(1 + \frac{4n^2 G^2}{\mu\bar{\nu}}\right) \\
K_6 &= -\frac{\eta}{32G^2 n/\bar{\nu} + 4(M+\mu)\kappa_\sigma} + 3\eta^2(1+c_3).
\end{aligned}
$$

Denoting $\kappa_1 = M/\mu$, $\kappa_2 = G^2 n/(\bar{\nu}\mu)$, $\kappa_3 = ((\kappa_1+1)\kappa_\sigma + 8\kappa_2)$, choosing

$$
c_3 = \min\{(8\kappa_1^2)^{-1}, (2\kappa_2)^{-1}\}
$$

and assuming $\kappa_2 \geq 1$, so that $c_3 \leq 1$, we get

$$
\begin{aligned}
K_1 &= \tau^{-1} - \frac{\eta\mu}{2} \\
K_2 &\leq \tau^{-1} - \frac{1}{2n} \\
K_3 &\leq \tau^{-1} - \frac{1}{n} + \frac{12\eta\mu\kappa_3}{n} \\
K_4 &= \tau^{-1} - \frac{1}{2n} \\
K_5 &= \tau^{-1} - \frac{1}{n} + \frac{2\eta\mu\kappa_2}{c_3}(1 + n\kappa_2) \\
K_6 &= -\frac{\eta}{4\mu\kappa_3} + 6\eta^2
\end{aligned}
$$

Finally, we choose

$$
\eta = \min\left\{\frac{1}{24\mu\kappa_3}, \frac{c_3}{4(1+4n\kappa_2)\kappa_2\mu}\right\}, \quad \tau^{-1} = \frac{1}{2}\min\left\{\frac{1}{n}, \eta\mu\right\},
$$

we get $K_i \leq 0$ for all i and the result follows. □

Corollary 25. *Consider the setting of Thm. 24, the iterates of Alg. 9 satisfy*

$$
\mathbb{E}\left[\|w^{(t)} - w^\star\|_2^2\right] \leq \exp(-t/\tau)\Big((1+nc_2)\|w^{(t)} - w^\star\|_2^2 + nc_1\sum_{i=1}^n \|q_i^{(0)}\nabla r_i(w^{(0)}) - \nabla r_i(w^\star)\|_2^2 \\
+ 2\eta n(q^{(0)} - q^\star)^\top(\ell(w^{(0)}) - \ell(w^\star))\Big).
$$

Proof. Follows from Thm. 24 by taking the expectation over the iterations of the algorithm and using that $R^{(t)}, S^{(t)}, T^{(t)}, U^{(t)}$ are positive. □

Lemma 26. *Consider the setting of Thm. 24. Define*

$$
U^{(t)} = \frac{1}{n}\sum_{j=1}^n \|w^{(t)} - \zeta_j^{(t)}\|_2^2.
$$

We have, denoting $Q^{(t)} = \mathbb{E}_t\left[\|nq^{(t)}_{i_t}\nabla r_{i_t}(w^{(t)}) - q^*_{i_t}\nabla r_{i_t}(w^\star)\|_2^2\right]$,

$$
\begin{aligned}
\mathbb{E}_t\left[U^{(t+1)}\right] &\leq \eta^2(1+\beta_2)Q^{(t)} + \eta^2(1+\beta_2^{-1})S^{(t)} \\
&\quad + \frac{\eta M^2}{\mu n}\left(1-\frac{1}{n}\right)T^{(t)} + \left(1-\frac{1}{n}\right)\frac{G^2}{2\bar{\nu}\mu n}R^{(t)} + \left(1-\frac{1}{n}\right)U^{(t)}.
\end{aligned}
$$

Proof. For $S^{(t+1)}$ and $T^{(t+1)}$, it follows from the definition of $\zeta_i^{(t+1)}$, $z_i^{(t+1)}$ and $\rho^{(t+1)}$ in Alg. 10. For $U^{(t+1)}$, we have

$$
\mathbb{E}_t\left[U^{(t+1)}\right] = \frac{1}{n}\mathbb{E}_t\left[\|w^{(t+1)} - w^{(t)}\|_2^2\right] + \left(1-\frac{1}{n}\right)\mathbb{E}_t\left[\frac{1}{n}\sum_{j=1}^n \|w^{(t+1)} - \zeta_j^{(t)}\|_2^2\right].
$$

We have

$$
\begin{aligned}
&\mathbb{E}_t\left[\sum_{j=1}^n \|w^{(t+1)} - \zeta_j^{(t)}\|_2^2\right] \\
&= \mathbb{E}_t\left[\sum_{j=1}^n \|w^{(t+1)} - w^{(t)}\|_2^2\right] + 2\mathbb{E}_t\left[\sum_{j=1}^n (w^{(t+1)} - w^{(t)})^\top (w^{(t)} - \zeta_j^{(t)})\right] + \mathbb{E}_t\left[\sum_{j=1}^n \|\zeta_j^{(t)} - w^{(t)}\|_2^2\right] \\
&= n\eta^2\mathbb{E}_t\left[\|v^{(t)}\|_2^2\right] - 2\eta\sum_{j=1}^n \nabla({q^{(t)}}^\top r)(w^{(t)})^\top(w^{(t)} - \zeta_j^{(t)}) + \sum_{j=1}^n \|\zeta_j^{(t)} - w^{(t)}\|_2^2.
\end{aligned}
$$

Next, we have

$$
\begin{aligned}
-2\nabla({q^{(t)}}^\top r)(w^{(t)})^\top(w^{(t)} - \zeta_j^{(t)}) &= -2(\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^{(t)}}^\top r)(\zeta_j^{(t)}))^\top(w^{(t)} - \zeta_j^{(t)}) \\
&\quad - 2(\nabla({q^{(t)}}^\top r)(\zeta_j^{(t)}) - \nabla({q^{(t)}}^\top r)(w^\star))^\top(w^{(t)} - \zeta_j^{(t)}) \\
&\quad - 2(\nabla((q^{(t)} - q^\star)^\top \ell)(w^\star))^\top(w^{(t)} - \zeta_j^{(t)}) \\
&\leq \beta_4 M^2\|\zeta_j^{(t)} - w^\star\|_2^2 + (\beta_4^{-1} + \beta_5^{-1} - 2\mu)\|\zeta_j^{(t)} - w^{(t)}\|_2^2 \\
&\quad + \beta_5 G^2\|q^{(t)} - q^\star\|_2^2,
\end{aligned}
$$

because

$$
\begin{aligned}
\nabla({q^{(t)}}^\top r)(w^{(t)}) - \nabla({q^{(t)}}^\top r)(\zeta_j^{(t)})^\top(w^{(t)} - \zeta_j^{(t)}) &\overset{(i)}{\geq} \mu\|\zeta_j^{(t)} - w^{(t)}\|_2^2 \\
-2(\nabla({q^{(t)}}^\top r)(\zeta_j^{(t)}) - \nabla({q^{(t)}}^\top r)(w^\star))^\top(w^{(t)} - \zeta_j^{(t)}) &\overset{(ii)}{\leq} \beta_4\|\nabla({q^{(t)}}^\top r)(\zeta_j^{(t)}) - \nabla({q^{(t)}}^\top r)(w^\star)\|_2^2 \\
&\quad + \beta_4^{-1}\|\zeta_j^{(t)} - w^{(t)}\|_2^2 \\
&\overset{(iii)}{\leq} \beta_4 M^2\|\zeta_j^{(t)} - w^{(t)}\|_2^2 + \beta_4^{-1}\|\zeta_j^{(t)} - w^{(t)}\|_2^2 \\
-2(\nabla((q^{(t)} - q^\star)^\top\ell)(w^\star))^\top(w^{(t)} - \zeta_j^{(t)}) &\overset{(iv)}{\leq} \beta_5\|\nabla((q^{(t)} - q^\star)^\top\ell)(w^\star)\|_2^2 + \beta_5^{-1}\|\zeta_j^{(t)} - w^{(t)}\|_2^2 \\
&\overset{(v)}{\leq} \beta_5 G^2\|q^{(t)} - q\|_2^2 + \beta_5^{-1}\|\zeta_j^{(t)} - w^{(t)}\|_2^2,
\end{aligned}
$$

using in (i) that ${q^{(t)}}^\top r$ is μ-strongly convex, in (ii) and (iv) Young's inequality with parameters β_4, β_5, in (iii) that ${q^{(t)}}^\top r$

is M smooth, and in (v) that the losses ℓ_i are G-Lipschitz. Hence, we get

$$\mathbb{E}_t\left[U^{(t+1)}\right] \leq \eta^2\mathbb{E}_t\left[\|v^{(t)}\|_2^2\right] + \frac{\eta\beta_4 M^2}{n}\left(1-\frac{1}{n}\right)\sum_{i=1}^n \|\zeta_j^{(t)} - w^\star\|_2^2 + \left(1-\frac{1}{n}\right)\beta_5\eta G^2\|q^{(t)} - q^\star\|_2^2$$
$$+ ((\beta_4^{-1} + \beta_5^{-1} - 2\mu)\eta + 1)\left(1-\frac{1}{n}\right)U^{(t)}.$$

Taking $\beta_4 = \beta_5 = \mu^{-1}$, using Lem. 14, and Lem. 45 to bound $\|q^{(t)} - q^\star\|_2^2 \leq \frac{1}{\bar{\nu}}(q^{(t)} - q^\star)(l^{(t)} - l^*)$ gives the claim. □

Proof. The fact that $q^{(t)} = \nabla h(l^{(t)})$ is a consequence of Danskin's theorem (Bertsekas, 1997, Proposition B.25) and the fact that the maximization problem is strongly concave. The function h is $1/\bar{\nu}$-smooth as the convex conjugate of a $\bar{\nu}$-strongly convex function (Nesterov, 2005, Theorem 1). The final inequalities are shown by, e.g., Nesterov (2018, Theorem 2.1.5). □

Lemma 27. *Consider $r : w \mapsto (r_i(w))_{i=1}^n$ with $r_i : \mathbb{R}^n \to \mathbb{R}$ M-smooth and μ-strongly convex. For any $p, q \in \Delta^n = \{x \in \mathbb{R}^n, x \geq 0, x^\top \mathbf{1} = 1\}$ such that $\|p\|_\infty \leq \sigma_{\max}, \|q\|_\infty \leq \sigma_{\max}$, and any $w, v \in \mathbb{R}^d$, we have*

$$-(q-p)^\top(r(v) - r(w)) + (\nabla(q^\top r)(v) - \nabla(p^\top r)(w))^\top(v-w)$$
$$\geq \frac{\mu}{2}\|w-v\|_2^2 + \frac{1}{2(M+\mu)\sigma_{\max}}\left(\sum_{i=1}^n \|p_i\nabla r_i(w) - p_i\nabla r_i(v)\|_2^2 + \sum_{i=1}^n \|q_i\nabla r_i(w) - q_i\nabla r_i(v)\|_2^2\right).$$

Proof. For any $i \in [n]$, such that $p_i > 0$, we have by Lem. 41, that for any $w, v \in \mathbb{R}^d$,

$$p_i r_i(v) \geq p_i r_i(w) + \nabla(p_i r_i)(w)^\top(v-w) + \frac{1}{2p_i(M+\mu)}\|p_i\nabla f(w) - p_i\nabla f(v)\|_2^2 + \frac{p_i\mu}{4}\|w-v\|_2^2$$
$$\geq p_i r_i(w) + \nabla(p_i r_i)(w)^\top(v-w) + \frac{1}{2\sigma_{\max}(M+\mu)}\|p_i\nabla r_i(w) - p_i\nabla r_i(v)\|_2^2 + \frac{p_i\mu}{4}\|w-v\|_2^2.$$

The second inequality holds for $p_i = 0$ too and so for all i since $p_i \geq 0$. Summing from 1 to n and using that $p^\top\mathbf{1} = \mathbf{1}$, we get

$$p^\top r(v) \geq p^\top r(w) + \nabla(p^\top r)(w)^\top(v-w)$$
$$+ \frac{1}{2(M+\mu)\sigma_{\max}}\sum_{i=1}^n \|p_i\nabla r_i(w) - p_i\nabla r_i(v)\|_2^2 + \frac{\mu}{4}\|w-v\|_2^2.$$

Similarly, we have that

$$q^\top r(w) \geq q^\top r(v) + \nabla(q^\top r)(v)^\top(w-v)$$
$$+ \frac{1}{2(M+\mu)\sigma_{\max}}\sum_{i=1}^n \|q_i\nabla r_i(w) - q_i\nabla r_i(v)\|_2^2 + \frac{\mu}{4}\|w-v\|_2^2.$$

Summing both inequalities give the claim. □

E. SaddleSAGA: Tackling the Saddle Point Problem Directly

We give an incremental saddle point algorithm to minimize the objective (3) in its min-max form directly. We build upon the saddle version of the SAGA algorithm (Palaniappan & Bach, 2016) to this end — we call the algorithm *SaddleSAGA*. For simplicity, we denote

$$\bar{\nu} = 2n\nu.$$

We consider directly the min-max problem

$$\min_{w \in \mathbb{R}^d} \max_{q \in \mathcal{P}(\sigma)} \left[\Psi(w, q) := q^\top \ell(w) + \frac{\mu}{2} \|w\|_2^2 - \frac{\bar{\nu}}{2} \|q - \mathbf{1}_n/n\|_2^2 \right]. \tag{32}$$

Note that the function Ψ is strongly convex in its first argument and strongly concave in its second argument. A pair $(w^\star, q^\star)$ is called a saddle point of the convex-concave function Ψ if

$$\max_{q \in \mathcal{P}(\sigma)} \Psi(w^\star, q) \leq \Psi(w^\star, q^\star) \leq \min_{w \in \mathbb{R}^d} \Psi(w, q^\star).$$

In our setting, we can verify that the pair $w^\star = \arg\min \mathcal{L}_\sigma$ and $q^\star = q^{\text{opt}}(\ell(w^\star))$ is the unique saddle point of Ψ.

Algorithm. We present SaddleSAGA in in Algorithm 11. The algorithm takes advantage of the availability of proximal operators, defined for a convex function $f : \mathbb{R}^d \to \mathbb{R}$, and $x \in \mathbb{R}^d$ as

$$\text{prox}_f(x) = \arg\min_{y \in \mathbb{R}^d} f(y) + \frac{1}{2} \|x - y\|_2^2.$$

The proximal update on $w^{(t+1)}$ can be computed in closed form. The proximal update on $q^{(t+1)}$ can be solved with the PAV algorithm, see Appx. C. Overall, the time and space complexity of SaddleSAGA is identical to that of SpecSAGA.

Rate of Convergence. We prove the following rate of convergence for SaddleSAGA.

Theorem 28. *The iterates $(w^{(t)}, q^{(t)})$ of Alg. 11 with learning rates*

$$\eta = \min\left\{\frac{1}{\mu}, \frac{1}{6(L\kappa_\sigma + 2G^2 n/\bar{\nu})}\right\}, \quad \delta = \min\left\{\frac{1}{\bar{\nu}}, \frac{\mu}{8n^2 G^2}\right\}$$

converge linearly to the saddle point of (32). In particular, for non-trivial regularization $\mu\bar{\nu} \leq 8n^2 G^2$ and $\mu \leq 6(L\kappa_\sigma + 2G^2 n/\bar{\nu})$, the number of iterations t to get $\|w^{(t)} - w^\star\|_2^2 + c\|q^{(t)} - q^\star\|_2^2 \leq \varepsilon$ (for some constant c) is at most

$$O\left(\left(n + \kappa\kappa_\sigma + \frac{n^2 G^2}{\mu\bar{\nu}}\right) \ln \frac{1}{\varepsilon}\right).$$

The proof of this statement is given as Cor. 34 later in this section.

Comparison to Previous Work. The version of SAGA adapted to saddle point problems, proposed by Palaniappan & Bach (2016), forms the basis of Algorithm 11. Compared to Algorithm 11, the algorithm of (Palaniappan & Bach, 2016) only uses a single learning rate for both the primal and dual updates. This seemingly simple modification leads to a significant difference in theory and in practice. Theoretically, the rate obtained by (Palaniappan & Bach, 2016) in terms of our problem's constants is

$$O\left(\left(n + \frac{nG^2}{\mu\bar{\nu}} + n\kappa^2\right) \ln \frac{1}{\varepsilon}\right).$$

Compared to this, the rate we prove for SaddleSAGA improves κ^2 to $\kappa\kappa_\sigma$ while suffering an additional factor of n in the $n^2 G^2/(\mu\bar{\nu})$ term. The rate of SaddleSAGA matches that of Thm. 1 when the shift cost $\bar{\nu}$ is large enough, while the rate of (Palaniappan & Bach, 2016) is worse by a factor of κ.

Empirical comparisons between SaddleSAGA and the algorithm of (Palaniappan & Bach, 2016) are given in Appx. I.

E.1. Convergence proof

In the following, we denote by $\mathbb{E}_t[\cdot]$ the expecation of a quantity according to the randomness of i_t conditioned on $w^{(t)}, q^{(t)}$. Throughout the proof, we consider that the losses are L-smooth and G-Lipschitz continuous.

Evolution of the distances to the optimum. We start by using the contraction properties of the proximal operator to bound the evolution of the distances to the saddle point $(w^\star, q^\star)$.

Algorithm 11 SaddleSAGA Algorithm: Solving the Min-Max Problem Directly

```
**Inputs:** Initial points $w^{(0)}$, $q^{(0)} = u_n = \mathbf{1}/n$, stepsizes $\eta > 0, \delta > 0$, number of iterations $T$
Set $\rho^{(0)} = q^{(0)}$, $l^{(0)} = (\ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^n$, $g^{(0)} = (\nabla \ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^{d\times n}$, $\bar{g}^{(0)} = \sum_{i=1}^n \rho_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$
**for** $t = 0, \dots, T-1$ **do**
    Sample $i_t \sim \mathrm{Unif}([n])$
    $v^{(t)} = nq_{i_t}^{(t)} \nabla \ell_{i_t}(w^{(t)}) - (n\rho_{i_t}^{(t)} g_{i_t}^{(t)}) - \bar{g}^{(t)})$
    $w^{(t+1)} = \mathrm{prox}_{\eta\mu\|\cdot\|_2^2}(w^{(t)} - \eta v^{(t)})$
    $\pi^{(t)} = n\ell_{i_t}(w^{(t)})e_{i_t} - (nl_{i_t}^{(t)} e_{i_t} - l^{(t)})$
    $q^{(t+1)} = \mathrm{prox}_{\iota_{\mathcal{P}(\sigma)} + \delta\bar{\nu}\|\cdot - \mathbf{1}_n/n\|_2^2/2}(q^{(t)} - \delta\pi^{(t)})$
    $\rho_{i_t}^{(t+1)} = q_{i_t}^{(t)}$ and $\rho_j^{(t+1)} = \rho_j^{(t)}$ for $j \neq i_t$
    $l_{i_t}^{(t+1)} = \ell_{i_t}(w^{(t)})$ and $l_j^{(t+1)} = l_j^{(t)}$ for $j \neq i_t$
    $g_{i_t}^{(t+1)} = \nabla \ell_{i_t}(w^{(t)})$ and $g_j^{(t+1)} = g_j^{(t)}$ for $j \neq i_t$
    $\bar{g}^{(t+1)} = \sum_{i=1}^n \rho_i^{(t+1)} g_i^{(t+1)} = \rho_{i_t}^{(t+1)} \nabla \ell_{i_t}(w^{(t)}) - \rho_{i_t}^{(t)} g_{i_t} + \bar{g}^{(t)}$
**Output:** Final points $w^{(T)}$, $q^{(T)}$.
```

Lemma 29. *Consider the setting of Alg. 11. We have,*

$$
\begin{aligned}
\mathbb{E}_t\left[\|w^{(t+1)} - w^\star\|_2^2\right] \leq \frac{1}{(1+\eta\mu)^2}\Big(&\|w^{(t)} - w^\star\|_2^2 \\
&- 2\eta(\nabla({q^{(t)}}^\top \ell)(w^{(t)}) - \nabla({q^*}^\top \ell)(w^\star))^\top (w^{(t)} - w^\star) \\
&+ \eta^2 \mathbb{E}_t\left[\|v^{(t)} - \nabla({q^*}^\top \ell)(w^\star)\|_2^2\right]\Big) \\
\mathbb{E}_t\left[\|q^{(t+1)} - q^\star\|_2^2\right] \leq \frac{1}{(1+\delta\bar{\nu})^2}\Big(&\|q^{(t)} - q^\star\|_2^2 \\
&+ 2\delta(\ell(w^{(t)}) - \ell(w^\star))^\top (q^{(t)} - q^\star) \\
&+ \delta^2 \mathbb{E}_t\left[\|\pi^{(t)} - \ell(w^\star)\|_2^2\right]\Big).
\end{aligned}
$$

Proof. By considering the first-order optimality conditions of the problem one verifies that $w^\star, q^\star$ satisfy for any η, δ,

$$
w^\star = \mathrm{prox}_{\eta\mu\|\cdot\|_2^2/2}(w^\star - \eta\nabla({q^\star}^\top \ell)(w^\star)), \quad q^\star = \mathrm{prox}_{\iota_{\mathcal{P}(\sigma)} + \delta\bar{\nu}\|\cdot - \mathbf{1}_n/n\|_2^2/2}(q^\star + \delta\ell(w^\star)).
$$

Recall that the proximal operator of a c-strongly convex function h is contractive such that $\|\mathrm{prox}_h(z) - \mathrm{prox}_h(z')\|_2 \leq \frac{1}{1+c}\|z - z'\|_2$. In our case, it means that

$$
\begin{aligned}
\|w^{(t+1)} - w^\star\|_2 &\leq \frac{1}{1+\eta\mu}\|w^{(t)} - \eta v^{(t)} - (w^\star - \eta\nabla({q^\star}^\top \ell)(w^\star))\|_2, \\
\|q^{(t+1)} - q^\star\|_2 &\leq \frac{1}{1+\delta\bar{\nu}}\|q^{(t)} + \delta\pi^{(t)} - (q^\star + \delta\ell(w^\star))\|_2.
\end{aligned}
$$

By taking the squared norm, the expectation, expanding the squared norms and using that $\mathbb{E}_t\left[v^{(t)}\right] = \nabla({q^{(t)}}^\top \ell)(w^{(t)})$, $\mathbb{E}_t\left[\pi^{(t)}\right] = \ell(w^{(t)})$, we get the result. □

Variance term evolutions. We consider the evolution of the additional variance term added to the dual variables.

Lemma 30. *In the setting of Alg. 11, we have for any $\beta_2 > 0$,*

$$
\begin{aligned}
\mathbb{E}_t\left[\|\pi^{(t)} - \ell(w^\star)\|_2^2\right] \leq\ & (n + (n-1)\beta_2)nG^2\|w^{(t)} - w^\star\|_2^2 \\
&+ (n-1)(1+\beta_2^{-1})\|\ell(w^\star) - l^{(t)}\|_2^2.
\end{aligned}
$$

Proof. As in the proof of Lem. 14, we have for some $\beta_2 > 0$,

$$\begin{aligned}
\mathbb{E}_t\left[\|\pi^{(t)} - \ell(w^\star)\|_2^2\right] &= \mathbb{E}_{i_t}\Big[\|(n\ell_{i_t}(w^{(t)}) - n\ell_{i_t}(w^\star))e_{i_t} \\
&\qquad + (n\ell_{i_t}(w^\star) - n\ell_{i_t}(z_{i_t}^{(t)}))e_{i_t} - (\ell(w^\star) - l^{(t)})\|_2^2\Big] \\
&\leq -\beta_2\|\ell(w^{(t)}) - \ell(w^\star)\|_2^2 \\
&\quad + (1+\beta_2)\mathbb{E}_t\left[\|(n\ell_{i_t}(w^{(t)}) - n\ell_{i_t}(w^\star))e_{i_t}\|_2^2\right] \\
&\quad + (1+\beta_2^{-1})\mathbb{E}_t\left[\|(n\ell_{i_t}(w^\star) - n\ell_{i_t}(z_{i_t}^{(t)}))e_{i_t}\|_2^2\right] \\
&\quad - (1+\beta_2^{-1})\|\ell(w^\star) - l^{(t)}\|_2^2 \\
&= (n + (n-1)\beta_2)\|\ell(w^{(t)}) - \ell(w^\star)\|_2^2 \\
&\quad + (n-1)(1+\beta_2^{-1})\|\ell(w^\star) - l^{(t)}\|_2^2 \\
&\leq (n + (n-1)\beta_2)nG^2\|w^{(t)} - w^\star\|_2^2 \\
&\quad + (n-1)(1+\beta_2^{-1})\|\ell(w^\star) - l^{(t)}\|_2^2.
\end{aligned}$$

□

Incorporating smoothness and convexity of the losses. Our approach differs from (Palaniappan & Bach, 2016) by Cor. 32 stemming from Lem. 31. We exploit the smoothness and convexity of the losses to get a negative term $-\mathbb{E}_t\left[\|nq_{i_1}\nabla\ell_{i_t}(w^{(t)}) - nq_{i_t}^*\nabla\ell_{i_t}(w^\star)\|_2^2\right]$ used to temper the variance of the primal updates at the price of an additional positive term $\|q^{(t)} - q^\star\|_2^2$. The sum of both being positive we can dampen the effect of the additional positive term $\|q^{(t)} - q^\star\|_2^2$ at the price of getting a less negative term $-\mathbb{E}_t\left[\|nq_{i_1}\nabla\ell_{i_t}(w^{(t)}) - nq_{i_t}^*\nabla\ell_{i_t}(w^\star)\|_2^2\right]$. The insights of Cor. 32 inspired Cor. 22 for the proof of Alg. 9 for any shift costs. However, they slightly differ, so we provide their complete statements here.

Lemma 31. *For any $q_1, q_2 \in \mathcal{P}(\sigma)$, $w_1, w_2 \in \mathbb{R}^d$, we have,*

$$\begin{aligned}
&(q_1 - q_2)^\top(\ell(w_1) - \ell(w_2)) - (\nabla(q_1^\top\ell)(w_1) - \nabla(q_2^\top\ell)(w_2))^\top(w_1 - w_2) \\
&\leq -\frac{1}{2Ln\sigma_{\max}}\left(\mathbb{E}_{i\sim\mathrm{Unif}[n]}\left[\|nq_{1,i}\nabla\ell_i(w_1) - nq_{2,i}\nabla\ell_i(w_2)\|_2^2 + \|nq_{1,i}\nabla\ell(w_2) - nq_{2,i}\nabla\ell(w_1)\|_2^2\right]\right) \\
&\quad + \frac{G^2}{L\sigma_{\max}}\|q_1 - q_2\|_2^2.
\end{aligned}$$

Proof. For any $q \in \mathcal{P}(\sigma)$ and any $w, v \in \mathbb{R}^d$, we have by smoothness and convexity of $q_i\ell_i$, for $q_i > 0$

$$q_i\ell_i(v) \geq q_i\ell_i(w) + q_i\nabla\ell_i(w)^\top(v - w) + \frac{1}{2Lq_i}\|q_i\nabla\ell_i(w) - q_i\nabla\ell_i(v)\|_2^2 \tag{33}$$

$$\geq q_i\ell_i(w) + q_i\nabla\ell_i(w)^\top(v - w) + \frac{1}{2Ln^2\sigma_{\max}}\|nq_i\nabla\ell_i(w) - nq_i\nabla\ell_i(v)\|_2^2. \tag{34}$$

Note that the second inequality is then true even if $q_i = 0$, since in that case all terms are 0. Therefore, for any $q_1, q_2 \in \mathcal{P}(\sigma)$, and any w_1, w_2, we have

$$\begin{aligned}
q_1^\top\ell(w_2) &\geq q_1^\top\ell(w_1) + \nabla(q_1^\top\ell)(w_1)^\top(w_2 - w_1) + \frac{1}{2Ln\sigma_{\max}}\mathbb{E}_{i\sim\mathrm{Unif}[n]}\left[\|nq_{1,i}\nabla\ell_i(w_1) - nq_{1,i}\nabla\ell_i(w_2)\|_2^2\right], \\
q_2^\top\ell(w_1) &\geq q_2^\top\ell(w_2) + \nabla(q_2^\top\ell)(w_2)^\top(w_1 - w_2) + \frac{1}{2Ln\sigma_{\max}}\mathbb{E}_{i\sim\mathrm{Unif}[n]}\left[\|nq_{2,i}\nabla\ell(w_1) - nq_{2,i}\nabla\ell(w_2)\|_2^2\right].
\end{aligned}$$

Combining these inequalities, we get

$$
\begin{aligned}
&-(q_1 - q_2)^\top(\ell(w_1) - \ell(w_2)) + (\nabla(q_1^\top \ell)(w_1) - \nabla(q_2^\top \ell)(w_2))^\top (w_1 - w_2)\\
&\geq \frac{1}{2Ln\sigma_{\max}}\left(\mathbb{E}_{i\sim \mathrm{Unif}[n]}\left[\|nq_{1,i}\nabla\ell_i(w_1) - nq_{1,i}\nabla\ell_i(w_2)\|_2^2 + \|nq_{2,i}\nabla\ell(w_1) - nq_{2,i}\nabla\ell(w_2)\|_2^2\right]\right).
\end{aligned}
$$

For any 4 vectors a, b, c, d,

$$
\|a-b\|_2^2 + \|c-d\|_2^2 = \|a-c\|_2^2 + \|b-d\|_2^2 - 2(a-d)^\top(b-c).
$$

Applying this for $a = q_{1,i}\nabla\ell_i(w_1),\ b = q_{i,1}\nabla\ell_i(w_2),\ c = q_{2,i}\nabla\ell_i(w_2),\ d = q_{2,i}\nabla\ell_i(w_1)$, we get

$$
\begin{aligned}
&-(q_1 - q_2)^\top(\ell(w_1) - \ell(w_2)) + (\nabla(q_1^\top \ell)(w_1) - \nabla(q_2^\top \ell)(w_2))^\top (w_1 - w_2)\\
&\geq \frac{1}{2Ln\sigma_{\max}}\Big(\mathbb{E}_{i\sim \mathrm{Unif}[n]}\left[\|nq_{1,i}\nabla\ell_i(w_1) - nq_{2,i}\nabla\ell_i(w_2)\|_2^2 + \|nq_{1,i}\nabla\ell(w_2) - nq_{2,i}\nabla\ell(w_1)\|_2^2\right]\\
&\quad - 2n^2\mathbb{E}_{i\sim \mathrm{Unif}[n]}\left[(q_{1,i} - q_{2,i})^2\nabla\ell_i(w_1)^\top\nabla\ell_i(w_2)\right]\Big).
\end{aligned}
$$

Reorganizing the terms and bounding $\nabla\ell_i(w_1)^\top\nabla\ell_i(w_2)$ by G^2 we get the result. □

Corollary 32. *In the setting of Alg. 11, we have for any $\alpha \in [0,1]$*

$$
\begin{aligned}
&\mathbb{E}_t\left[\frac{(1+\eta\mu)^2}{\eta}\|w^{(t+1)} - w^\star\|_2^2 + \frac{(1+\delta\bar{\nu})^2}{\delta}\|q^{(t+1)} - q^\star\|_2^2\right]\\
&\leq \eta^{-1}\|w^{(t)} - w^\star\|_2^2 + \left(\delta^{-1} + \frac{2\alpha G^2}{L\sigma_{\max}}\right)\|q^{(t)} - q^\star\|_2^2\\
&\quad + \eta\mathbb{E}_t\left[\|v^{(t)} - \nabla({q^*}^\top\ell)(w^\star)\|_2^2\right] + \delta\mathbb{E}_t\left[\|\pi^{(t)} - \ell(w^\star)\|_2^2\right]\\
&\quad - \frac{\alpha}{Ln\sigma_{\max}}\mathbb{E}_t\left[\|nq_{i_1}\nabla\ell_{i_t}(w^{(t)}) - nq_{i_t}^*\nabla\ell_{i_t}(w^\star)\|_2^2\right].
\end{aligned}
$$

Proof. Follows from Lem. 31 □

Lyapunov function and overall convergence. Thm. 33 shows that an appropriately defined Lyapunov function incorporating the distances to the optima, decrease exponentially.

Theorem 33. *Consider the setting of Alg. 11. Define the Lyapunov function*

$$
\begin{aligned}
V^{(t)} = &\frac{(1+\eta\mu)^2}{\eta}\|w^{(t)} - w^\star\|_2^2 + \frac{(1+\delta\bar{\nu})^2}{\delta}\|q^{(t)} - q^\star\|_2^2\\
&+ c_1\sum_{i=1}^n\|n\rho_i^{(t)}\nabla\ell_i(z_i^{(t)}) - nq_i^*\nabla\ell_i(w^\star)\|_2^2 + \frac{c_2}{G^2}\|l^{(t)} - \ell(w^\star)\|_2^2,
\end{aligned}
$$

with $c_1 = \frac{n}{2(L\kappa_\sigma + 2G^2 n/\bar{\nu})}$ and $c_2 = \frac{\mu}{2}$ with $\kappa_\sigma = n\sigma_{\max}$. By taking

$$
\eta = \min\left\{\frac{1}{\mu}, \frac{1}{6(L\kappa_\sigma + 2G^2 n/\bar{\nu})}\right\}, \quad \delta = \min\left\{\frac{1}{\bar{\nu}}, \frac{\mu}{8n^2G^2}\right\},
$$

we have

$$
\mathbb{E}_t\left[V^{(t+1)}\right] \leq (1 - \tau^{-1})V^{(t)},
$$

for some $\tau > 1$. In particular, for small regularizations, i.e., $\mu\bar{\nu} \leq 8n^2G^2$ and $\mu \leq 6(L\kappa_\sigma + 2G^2n/\bar{\nu})$, we have

$$
\tau = \max\left\{2n, 4 + \frac{24L\kappa_\sigma}{\mu} + \frac{48G^2 n}{\mu\bar{\nu}}, 2 + \frac{16G^2n^2}{\bar{\nu}\mu}\right\}.
$$

Proof. Let us denote

$$T^{(t)} = \frac{1}{n}\sum_{i=1}^{n} \|n\rho_i^{(t)}\nabla \ell_i(z_i^{(t)}) - nq_i^{*}\nabla\ell_i(w^\star)\|_2^2, \quad S^{(t)} = \|l^{(t)} - \ell(w^\star)\|_2^2,$$

we have,

$$\begin{aligned}
\mathbb{E}_t\left[T^{(t+1)}\right] &\le \frac{1}{n^2}\sum_{i=1}^{n}\|nq_i^{(t)}\nabla\ell_i(w^{(t)}) - nq_i^{*}\nabla\ell_i(w^\star)\|_2^2 + \left(1-\frac{1}{n}\right)T^{(t)},\\
\mathbb{E}_t\left[S^{(t+1)}\right] &\le G^2\|w^{(t)} - w^\star\|_2^2 + \left(1-\frac{1}{n}\right)S^{(t)}.
\end{aligned}$$

By combining Cor. 32, Lem. 14, Lem. 30 we have, denoting $\kappa_\sigma = n\sigma_{\max}$,

$$\begin{aligned}
\mathbb{E}_t\left[V^{(t+1)}\right] \le{}& \left(\eta^{-1} + \delta(n+(n-1)\beta_2)nG^2 + c_2\right)\|w^{(t)} - w^\star\|_2^2\\
&+ \left(\delta^{-1} + \frac{2\alpha n G^2}{L\kappa_\sigma}\right)\|q^{(t)} - q^\star\|_2^2\\
&+ \left(\eta(1+\beta_1) + \frac{c_1}{n} - \frac{\alpha}{Ln\sigma_{\max}}\right)\mathbb{E}_{i\sim\mathrm{Unif}[n]}\left[\|nq_{i_1}\nabla\ell_{i_t}(w^{(t)}) - nq_{i_t}^{*}\nabla\ell_{i_t}(w^\star)\|_2^2\right]\\
&+ \left(\eta(1+\beta_1^{-1}) + c_1\left(1-\frac{1}{n}\right)\right)\frac{1}{n}\sum_{i=1}^{n}\|n\rho_i^{(t)}\nabla\ell_i(z_i^{(t)}) - nq_i^{*}\nabla\ell_i(w^\star)\|_2^2\\
&+ \left(\delta(n-1)(1+\beta_2^{-1}) + \frac{c_2}{G^2}\left(1-\frac{1}{n}\right)\right)\|\ell(w^\star) - l^{(t)}\|_2^2.
\end{aligned}$$

Therefore for some $\tau > 1$, we have

$$\begin{aligned}
\mathbb{E}_t\left[V^{(t+1)}\right] - (1-\tau^{-1})V^{(t)} \le{}& K_1\|w^{(t)} - w^\star\|_2^2 + K_2\|q^{(t)} - q^\star\|_2^2\\
&+ K_3\mathbb{E}_{i\sim\mathrm{Unif}[n]}\left[\|nq_{i_1}\nabla\ell_{i_t}(w^{(t)}) - nq_{i_t}^{*}\nabla\ell_{i_t}(w^\star)\|_2^2\right]\\
&+ K_4\frac{1}{n}\sum_{i=1}^{n}\|n\rho_i^{(t)}\nabla\ell_i(z_i^{(t)}) - nq_i^{*}\nabla\ell_i(w^\star)\|_2^2 + K_5\|\ell(w^\star) - l^{(t)}\|_2^2,
\end{aligned}$$

with,

$$\begin{aligned}
K_1 &= \frac{(1+\eta\mu)^2}{\eta}\left(\frac{1+\eta\left((n+(n-1)\beta_2)nG^2\delta + c_2\right)}{(1+\eta\mu)^2} - (1-\tau^{-1})\right)\\
K_2 &= \frac{(1+\delta\bar{\nu})^2}{\delta}\left(\frac{1+2\delta\alpha G^2 n/(L\kappa_\sigma)}{(1+\delta\bar{\nu})^2} - (1-\tau^{-1})\right)\\
K_3 &= \eta(1+\beta_1) + \frac{c_1}{n} - \frac{\alpha}{L\kappa_\sigma}\\
K_4 &= c_1\left(\eta(1+\beta_1^{-1})\frac{1}{c_1} + \left(1-\frac{1}{n}\right) - (1-\tau^{-1})\right)\\
K_5 &= \frac{c_2}{G^2}\left(\delta(n-1)(1+\beta_2^{-1})\frac{G^2}{c_2} + \left(1-\frac{1}{n}\right) - (1-\tau^{-1})\right).
\end{aligned}$$

Fix $\beta_1 = 2, \beta_2 = 1$. Denote also $\bar{\eta} = \frac{\eta\mu}{1+\eta\mu} \in (0,1)$ and $\bar{\delta} = \frac{\delta\bar{\nu}}{1+\delta\bar{\nu}} \in (0,1)$ with e.g. $\eta = \frac{\bar{\eta}}{\mu(1+\bar{\eta})}$. We have then for

$c_1/n = \alpha/(2L\kappa_\sigma)$ and $c_2 = \mu/2$,

$$\begin{aligned}
K_1 &\leq \eta\mu^2\bar{\eta}\left(\bar{\eta}^2 - \left(1 - \frac{2n^2G^2\delta}{\mu}\right)\bar{\eta} + \tau^{-1}\right)\\
K_2 &\leq \delta\bar{\nu}^2\bar{\delta}\left(\bar{\delta}^2 - 2\left(1 - \frac{\alpha G^2 n}{L\kappa_\sigma\bar{\nu}}\right)\bar{\delta} + \tau^{-1}\right)\\
K_3 &= 3\eta - \frac{\alpha}{2L\kappa_\sigma}\\
K_4 &= c_1\left(3\eta\frac{L\kappa_\sigma}{n\alpha} - \frac{1}{n} + \tau^{-1}\right)\\
K_5 &\leq \frac{c_2}{G^2}\left(\delta\frac{4nG^2}{\mu} - \frac{1}{n} + \tau^{-1}\right).
\end{aligned}$$

We can further take $3\eta \leq \alpha/(2L\kappa_\sigma)$ and $\delta \leq \mu/(8n^2G^2)$. By imposing the constraint $\tau \geq 2n$, we can simplify

$$\begin{aligned}
K_1 &\leq \eta\mu^2\bar{\eta}\left(\bar{\eta}^2 - \frac{3}{4}\bar{\eta} + \tau^{-1}\right)\\
K_2 &\leq \delta\bar{\nu}^2\bar{\delta}\left(\bar{\delta}^2 - 2\left(1 - \frac{\alpha G^2 n}{L\kappa_\sigma\bar{\nu}}\right)\bar{\delta} + \tau^{-1}\right)\\
K_3 &\leq 0, K_4 \leq 0, K_5 \leq 0.
\end{aligned}$$

Recall that α must be chosen in $[0, 1]$. Taking then

$$\alpha = \frac{L\kappa_\sigma}{L\kappa_\sigma + 2G^2n/\bar{\nu}} \leq \frac{L\kappa_\sigma\bar{\nu}}{2G^2n},$$

we get

$$K_1 \leq \eta\mu^2\bar{\eta}\left(\bar{\eta}^2 - \frac{3}{4}\bar{\eta} + \tau^{-1}\right), \quad K_2 \leq \delta\bar{\nu}^2\bar{\delta}\left(\bar{\delta}^2 - \bar{\delta} + \tau^{-1}\right).$$

By taking $\eta \leq 1/\mu$, $\delta \leq 1/\bar{\nu}$, we get $\bar{\eta} \leq 1/2$, $\bar{\delta} \leq 1/2$ and so $\bar{\eta}^2 - \frac{3}{4}\bar{\eta} \leq -\frac{1}{4}\bar{\eta}$ and $\bar{\delta}^2 - \bar{\delta} \leq -\frac{1}{2}\bar{\delta}$. Therefore taking

$$\eta = \min\left\{\frac{1}{\mu}, \frac{1}{6(L\kappa_\sigma + 2G^2n/\bar{\nu})}\right\}, \quad \delta = \min\left\{\frac{1}{\bar{\nu}}, \frac{\mu}{8n^2G^2}\right\},$$

we get $K_i \leq 0$ for all i as long as $\tau \geq \max\{2n, 4/\bar{\eta}, 2/\bar{\delta}\}$. In our case,

$$\begin{aligned}
\frac{4}{\bar{\eta}} &= \begin{cases} 4\left(1 + \frac{6L\kappa_\sigma}{\mu} + \frac{12G^2n}{\mu\bar{\nu}}\right) & \text{if } \mu \leq 6(L\kappa_\sigma + 2G^2n/\bar{\nu}),\\ 8 & \text{otherwise,}\end{cases}\\
\frac{2}{\bar{\delta}} &= \begin{cases} 2\left(1 + \frac{8G^2n^2}{\bar{\nu}\mu}\right) & \text{if } \mu\bar{\nu} \leq 8n^2G^2,\\ 4 & \text{otherwise.}\end{cases}
\end{aligned}$$

The result follows. □

Corollary 34. *Under the setting of Thm. 33, after t iterations of Alg. 11, we have*

$$
\begin{aligned}
&\mathbb{E}\left[\frac{(1+\eta\mu)^2}{\eta}\|w^{(t)} - w^\star\|_2^2 + \frac{(1+\delta\bar{\nu})^2}{\delta}\|q^{(t)} - q^\star\|_2^2\right] \\
&\leq \exp(-t/\tau)\Bigg(\frac{(1+\eta\mu)^2}{\eta}\|w^{(0)} - w^\star\|_2^2 + \frac{(1+\delta\bar{\nu})^2}{\delta}\|q^{(0)} - q^\star\|_2^2 \\
&\qquad\qquad + c_1 n^2 \sum_{i=1}^{n} \|nq_i^{(0)}\nabla\ell_i(w^{(0)}) - q_i^*\nabla\ell_i(w^\star)\|_2^2 + \frac{c_2}{G^2}\|\ell(w^{(0)}) - \ell(w^\star)\|_2^2\Bigg).
\end{aligned}
$$

F. Improving SpecSAGA with Moreau Envelopes

Notation. The Moreau envelope and the proximal (prox) operator of a convex function $f : \mathbb{R}^d \to \mathbb{R}$ are respectively defined for a constant $\eta > 0$ as

$$
\mathcal{M}_\eta[f](w) = \min_{z\in\mathbb{R}^d}\left\{f(z) + \frac{1}{2\eta}\|w - z\|_2^2\right\}, \tag{35}
$$

$$
\mathrm{prox}_{\eta f}(w) = \arg\min_{z\in\mathbb{R}^d}\left\{f(z) + \frac{1}{2\eta}\|w - z\|_2^2\right\}. \tag{36}
$$

A fundamental property is that the gradient of the Moreau envelope is related to the prox operator:

$$
\nabla\mathcal{M}_\eta[f](w) = \frac{1}{\eta}(w - \mathrm{prox}_{\eta f}(w)). \tag{37}
$$

The algorithm is given in Algorithm 12. For simplicity, we denote

$$
\bar{\nu} = 2n\nu.
$$

Implementation Details. The proximal operators can be computed in closed form or algorithmically for common losses. We list here the implementations for some losses of interest. The proximal operators for the binary or multiclass logistic losses cannot be obtained in closed form, we approximate them by one Newton step.

Squared loss. For the squared loss, defined as $\ell(w) = \frac{1}{2}(w^\top x - y)^2$ for $x \in \mathbb{R}^d, y \in \mathbb{R}$, then

$$
\mathrm{prox}_{\eta\ell}(w) = w - \frac{\eta x}{1 + \eta\|x\|^2}\left(x^\top w - y\right).
$$

Binary logistic loss. For the binary logistic loss defined for $x \in \mathbb{R}^d$, $y \in \{0, 1\}$, $w \in \mathbb{R}^d$ as $\ell(w) = -y\ln(\sigma(x^\top w)) - (1 - y)\ln(1 - \sigma(x^\top w)) = -yx^\top w + \ln(1 + e^{x^\top w})$, we approximate the proximal operator by one Newton step, whose formulation reduces to

$$
\mathrm{prox}_{\eta\ell}(w) \approx w - \frac{\eta g}{1 + \eta q\|x\|_2^2}x
$$

Multinomial logistic loss. For the multinomial logistic loss of a linear model defined by W on a sample (x, y) as $\ell(W) = -y^\top Wx + \ln(\exp(Wx)^\top\mathbf{1})$. for $x \in \mathbb{R}^d$, $y \in \{0, 1\}^k$, $y^\top\mathbf{1} = 1$, $W \in \mathbb{R}^{k\times d}$, we consider approximating the proximal operator by one Newton-step, whose formulation reduces to

$$
\begin{aligned}
\mathrm{prox}_{\eta\ell}(W) &\approx W - \eta z^* x^\top \\
&\qquad z^* = z_1 - \lambda^* z_2, \\
&\qquad z_1 = -y \oslash z_3 + z_2,\ z_2 = \sigma(Wx) \oslash z_3,\ z_3 = (\mathbf{1} + \eta\|x\|_2^2\sigma(Wx)),\ \lambda^* = \frac{z_1^\top\mathbf{1}}{z_2^\top\mathbf{1}}.
\end{aligned}
$$

Algorithm 12 SpecSAGA-Prox

```
**Inputs:** Initial points $w^{(0)}$, spectrum $\sigma$, stepsize $\eta > 0$, number of iterations $T$, regularization parameter $\mu > 0$, shift cost $\bar\nu > 0$, losses $\ell_1, \ldots, \ell_n$.
$l^{(0)} = (\ell_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^n$.
$g^{(0)} = (\nabla r_i(w^{(0)}))_{i=1}^n \in \mathbb{R}^{n \times d}$.
$q^{(0)} = \nabla h_\sigma(l^{(0)})$
$\bar g^{(0)} = \sum_{i=1}^n q_i^{(0)} g_i^{(0)} \in \mathbb{R}^d$.
**for** $t = 0, \ldots, T-1$ **do**
    Sample $i_t \sim q^{(t)}$ and $j_t \sim \mathrm{Unif}([n])$.
    $u^{(t)} = w^{(t)} + \eta(g_{i_t}^{(t)} - \bar g^{(t)})$.                          ▷ Add control variate to $w^{(t)}$.
    $w^{(t+1)} = \mathrm{prox}_{\eta r_{i_t}}(u^{(t)})$.                          ▷ Proximal update on the sampled loss.
    $l_{j_t}^{(t+1)} = \ell_{j_t}(w^{(t)})$ and $l_j^{(t+1)} = l_j^{(t)}$ for $j \neq j_t$.
    $g_{j_t}^{(t+1)} = \nabla \mathcal{M}_\eta[r_{j_t}]\big(w^{(t)} + \eta(g_{j_t}^{(t)} - \bar g^{(t)})\big)$      ▷ Update table with grad. of Moreau env.
    $g_j^{(t+1)} = g_j^{(t)}$ for $j \neq j_t$.
    $q^{(t+1)} = \arg\max_{q \in \mathcal{P}(\sigma)} q^\top l^{(t+1)} - \frac{\bar\nu}{2}\|q - \mathbf{1}_n/n\|_2^2$.
    $\bar g^{(t+1)} = \sum_{i=1}^n q_i^{(t+1)} g_i^{(t+1)} \in \mathbb{R}^d$.
**Output:** Final point $w^{(T)}$.
```

Regularized losses. For a convex $\ell : \mathbb{R}^d \to \mathbb{R}$, define $r(w) = \ell(w) + (\mu/2)\left\|w\right\|^2$. Then, we have,

$$\mathrm{prox}_{\eta r}(w) = \mathrm{prox}_{\frac{\eta\ell}{1+\eta\mu}}\left(\frac{w}{1+\eta\mu}\right) .$$

F.1. Convergence Analysis

Algorithm 12 satisfies the following convergence bound. Recall that $\gamma_\star = \left\|\nabla\ell(w^\star)\right\|_2$.

Theorem 35. *Suppose the smoothing parameter $\bar\nu$ is set large enough as*

$$\bar\nu \geq \frac{\gamma_* G}{M} \min\left\{\sqrt{\frac{2n\kappa}{4\kappa_\sigma^* - 1}}, 2\kappa\right\} ,$$

and define a constant

$$\tau = 2 + \max\{2(n-1),\ \kappa(4\kappa_\sigma^* - 1)\} ,$$

for $\kappa_\sigma^* = \sigma_n/\sigma_1$. Then, the sequence of iterates $(w^{(t)})$ generated by Algorithm 12 with learning rate $\eta = M^{-1}\min\{1/(4\kappa_\sigma^* - 1), \kappa/(n-1)\}$ satisfies

$$\mathbb{E}\left\|w^{(t)} - w^\star\right\|_2^2 \leq (n + 3/2)\exp(-t/\tau)\left\|w^{(0)} - w^\star\right\|_2^2 .$$

We now prove Thm. 35.

Notation for the Proof. We denote $\mathbb{E}_t[\cdot]$ denote the expectation conditioned on the randomness until time t; more precisely, on the sigma-algebra generated by $w^{(t)}$. Further, we define $w_i^* = w^* + \eta\nabla r_i(w^*)$. By analyzing the first-order conditions of the prox, it is easy to see that

$$\mathrm{prox}_{\eta r_i}(w_i^*) = w^* . \tag{38}$$

We will use the Lyapunov function

$$V^{(t)} = \left\| w^{(t)} - w^* \right\|^2 + c_1 \sum_{i=1}^{n} \left\| z_i^{(t)} - w^* \right\|^2 + \frac{c_2}{M^2} \sum_{i=1}^{n} \left\| g_i^{(t)} - \nabla r_i(w^*) \right\|^2 . \tag{39}$$

The first step is to analyze the effect of the update on $w^{(t)}$ as the first term of the Lyapunov function.

Proposition 36. *The iterates of Algorithm 12 satisfy*

$$\begin{aligned} (1 + \mu\eta)\mathbb{E}_t \left\| w^{(t+1)} - w^* \right\|^2 \leq{} & \left\| w^{(t)} - w^* \right\|^2 + 2\eta^2 \sigma_n \sum_{i=1}^{n} \left\| g_i^{(t)} - \nabla r_i(w^*) \right\|^2 \\ & + \frac{2\eta^2 \gamma_*^2 G^2}{\bar{\nu}^2} \sum_{i=1}^{n} \left\| z_i^{(t)} - w^* \right\|^2 \\ & - \eta^2 \left(1 + \frac{1}{M\eta} \right) \sigma_1 \sum_{i=1}^{n} \left\| \nabla \mathcal{M}_\eta [r_i] \big(w^{(t)} - \eta (g_i^{(t)} - \bar{g}^{(t)}) \big) - \nabla r_i(w^*) \right\|^2 . \end{aligned}$$

Proof. We use the co-coercivity of the prox operator (Thm. 42) to get

$$\begin{aligned} (1 + \mu\eta)\, \mathbb{E}_t \left\| w^{(t+1)} - w^* \right\|^2 &= (1 + \mu\eta)\, \mathbb{E}_t \left\| \mathrm{prox}_{\eta r_{i_t}}(u^{(t)}) - \mathrm{prox}_{\eta r_{i_t}}(w_{i_t}^*) \right\|^2 \\ &\leq \mathbb{E}_t \langle u^{(t)} - w_{i_t}^*, \mathrm{prox}_{\eta r_{i_t}}(u^{(t)}) - \mathrm{prox}_{\eta r_{i_t}}(w_{i_t}^*) \rangle \\ &= \mathbb{E}_t \langle u^{(t)} - w_{i_t}^*, w^{(t+1)} - w^* \rangle \\ &= \underbrace{\mathbb{E}_t \langle u^{(t)} - w_{i_t}^{(t)}, w^{(t)} - w^* \rangle}_{=:\mathcal{T}_1} + \underbrace{\mathbb{E}_t \langle u^{(t)} - w_{i_t}^*, w^{(t+1)} - w^{(t)} \rangle}_{=:\mathcal{T}_2} , \end{aligned} \tag{40}$$

where we added and subtracted $w^{(t)}$ in the last step.

For the first term, we observe that $\mathbb{E}_t[u^{(t)}] = w^{(t)}$ and $\mathbb{E}_t[w_{i_t}^*] = w^* + \eta\, \mathbb{E}_t[\nabla r_{i_t}(w^*)]$ so that

$$\mathcal{T}_1 = \left\langle \mathbb{E}_t[u^{(t)} - w_{i_t}^*], w^{(t)} - w^* \right\rangle = \left\| w^{(t)} - w^* \right\|^2 + \eta \left\langle \mathbb{E}_t[\nabla r_{i_t}(w^*)], w^{(t)} - w^* \right\rangle . \tag{41}$$

For $\mathcal{T}_2$, note that

$$w^{(t+1)} - w^{(t)} = -\eta \left(\nabla \mathcal{M}_\eta [r_{i_t}](u^{(t)}) - g_{i_t}^{(t)} + \bar{g}^{(t)} \right) .$$

We manipulate $\mathcal{T}_2$ to set ourselves up to apply co-coercivity of prox-gradient by adding and subtracting $\nabla \mathcal{M}_\eta [r_{i_t}](w_{i_t}^*)$ as follows:

$$\begin{aligned} \mathcal{T}_2 ={} & -\eta\, \mathbb{E}_t \langle u^{(t)} - w_{i_t}^{(t)}, \nabla \mathcal{M}_\eta [r_{i_t}](u^{(t)}) - g_{i_t}^{(t)} + \bar{g}^{(t)} \rangle \\ ={} & \underbrace{-\eta\, \mathbb{E}_t \langle u^{(t)} - w_{i_t}^*, \nabla \mathcal{M}_\eta [r_{i_t}](u^{(t)}) - \nabla \mathcal{M}_\eta [r_{i_t}](w_{i_t}^*) \rangle}_{=:\mathcal{T}_2'} \\ & \underbrace{-\eta\, \mathbb{E}_t \langle u^{(t)} - w_{i_t}^*, \nabla \mathcal{M}_\eta [r_{i_t}](w_{i_t}^*) - g_{i_t}^{(t)} + \bar{g}^{(t)} \rangle}_{=:\mathcal{T}_2''} . \end{aligned}$$

Now, co-coercivity of the prox-gradient (Thm. 43) of the M-smooth function r_{i_t} gives

$$\mathcal{T}_2' \leq -\eta^2 \left(1 + \frac{1}{M\eta} \right) \mathbb{E}_t \left\| \nabla \mathcal{M}_\eta [r_{i_t}](u^{(t)}) - \nabla \mathcal{M}_\eta [r_{i_t}](w_{i_t}^*) \right\|^2 . \tag{42}$$

Next, we use $u^{(t)} = w^{(t)} + \eta(g_{i_t}^{(t)} - \bar{g}^{(t)})$, and $w_i^* = w^* + \eta \nabla r_i(w^*)$ and $\nabla \mathcal{M}_\eta[r_i](w_i^*) = \nabla r_i(w^*)$ to get

$$\begin{aligned}\mathcal{T}_2'' &= -\eta \, \mathbb{E}_t \langle w^{(t)} - w^* - \eta(\nabla r_{i_t}(w^*) - g_{i_t}^{(t)} + \bar{g}^{(t)}), \nabla r_{i_t}(w^*) - g_{i_t}^{(t)} + \bar{g}^{(t)} \rangle \\ &= -\eta \, \langle w^{(t)} - w^*, \mathbb{E}_t[\nabla r_{i_t}(w^*)] \rangle + \eta^2 \, \mathbb{E}_t \left\| g_{i_t}^{(t)} - \bar{g}^{(t)} - \nabla r_{i_t}(w^*) \right\|^2 ,\end{aligned}$$

where we used that $\mathbb{E}_t[g_{i_t}^{(t)}] = \bar{g}^{(t)}$. Next, we use $\|x + y\|^2 \le 2\,\|x\|^2 + 2\,\|y\|^2$ for any vectors x, y and $\mathbb{E}\|X - \mathbb{E}[X]\|^2 \le \mathbb{E}\|X\|^2$ for any random vector X to get

$$\mathcal{T}_2'' \le -\eta \, \langle w^{(t)} - w^*, \mathbb{E}_t[\nabla r_{i_t}(w^*)] \rangle + 2\eta^2 \, \mathbb{E}_t \left\| g_{i_t}^{(t)} - \nabla r_{i_t}(w^*) \right\|^2 + 2\eta^2 \, \left\| \mathbb{E}_t[\nabla r_{i_t}(w^*)] \right\|^2 . \tag{43}$$

Plugging (43), (42), and (43) into (40) gives us

$$\begin{aligned}(1 + \mu\eta)\mathbb{E}_t \left\| w^{(t+1)} - w^* \right\|^2 \le \left\| w^{(t)} - w^* \right\|^2 + 2\eta^2 \, \mathbb{E}_t \left\| g_{i_t}^{(t)} - \nabla r_{i_t}(w^*) \right\|^2 + 2\eta^2 \, \left\| \mathbb{E}_t \left[\nabla r_{i_t}(w^*) \right] \right\|^2 \\ - \eta^2 \left(1 + \frac{1}{M\eta} \right) \mathbb{E}_t \left\| \nabla \mathcal{M}_\eta[r_{i_t}](u^{(t)}) - \nabla r_{i_t}(w^*) \right\|^2 .\end{aligned} \tag{44}$$

Next, we note that $\mathcal{P}(\sigma) \subset [\sigma_1, \sigma_n]^n$ to get,

$$\begin{aligned}\mathbb{E}_t \left\| g_{i_t} - \nabla r_{i_t}(w^*) \right\|^2 &= \sum_{i=1}^n q_i^{(t)} \left\| g_i - \nabla r_i(w^*) \right\|^2 \le \sigma_n \sum_{i=1}^n \left\| g_i - \nabla r_i(w^*) \right\|^2 , \quad \text{and} \\ \mathbb{E}_t \left\| \nabla \mathcal{M}_\eta[r_{i_t}](u^{(t)}) - \nabla r_{i_t}(w^*) \right\|^2 &= \sum_{i=1}^n q_i^{(t)} \left\| \nabla \mathcal{M}_\eta[r_i]\big(w^{(t)} - \eta(g_i^{(t)} - \bar{g}^{(t)})\big) - \nabla r_i(w^*) \right\|^2 \\ &\ge \sigma_1 \sum_{i=1}^n \left\| \nabla \mathcal{M}_\eta[r_i]\big(w^{(t)} - \eta(g_i^{(t)} - \bar{g}^{(t)})\big) - \nabla r_i(w^*) \right\|^2 .\end{aligned}$$

Moreover, we also have that

$$\begin{aligned}\left\| \mathbb{E}_t[\nabla r_{i_t}(w^*)] \right\|^2 &= \left\| \nabla \ell(w^\star)^\top (q^{\text{opt}}(l^{(t)}) - q^{\text{opt}}(\ell(w^\star))) \right] \right\|^2 \\ & \gamma_*^2 \left\| q^{\text{opt}}(l^{(t)}) - q^{\text{opt}}(\ell(w^\star))) \right\|_2^2 \\ &\le \frac{\gamma_*^2 G^2}{\bar{\nu}^2} \sum_{i=1}^n \left\| z_i^{(t)} - w^* \right\|^2 .\end{aligned}$$

Plugging these back into (44) completes the proof. □

Next, we analyze the other two terms of the Lyapunov function. The proof is trivial, so we omit it.

Proposition 37. *We have,*

$$\begin{aligned}\mathbb{E}_t \left[\sum_{i=1}^n \left\| z_i^{(t+1)} - w^* \right\|^2 \right] &= (1 - n^{-1}) \sum_{i=1}^n \left\| z_i^{(t)} - w^* \right\|^2 + \left\| w^{(t)} - w^* \right\|^2 , \\ \mathbb{E}_t \left[\sum_{i=1}^n \left\| g_i^{(t+1)} - \nabla r_i(w^*) \right\|^2 \right] &= (1 - n^{-1}) \sum_{i=1}^n \left\| g_i^{(t)} - \nabla r_i(w^*) \right\|^2 \\ &\quad + \frac{1}{n} \sum_{i=1}^n \left\| \nabla \mathcal{M}_\eta[r_i]\big(w^{(t)} - \eta(g_i^{(t)} - \bar{g}^{(t)})\big) - \nabla r_i(w^*) \right\|^2 .\end{aligned}$$

We are now ready to prove Thm. 35.

Proof of Thm. 35. Let $\tau > 1$ be a constant to be determined later and let $\Gamma := \gamma_*^2 G^2/(M^2\bar\nu^2)$ denote the effect of the smoothing. Combining Props. 36 and 37, we can write

$$
\begin{aligned}
\mathbb{E}_t[V^{(t)}] - (1-\tau^{-1})V^{(t)} \le & -\left\|w^{(t)} - w^*\right\|^2 \left(\frac{\mu\eta}{1+\mu\eta} - c_1 - \tau^{-1}\right) \\
& - \sigma_1 \sum_{i=1}^n \left\|\nabla \mathcal{M}_\eta[r_i]\big(w^{(t)} - \eta(g_i^{(t)} - \bar g^{(t)})\big) - \nabla r_i(w^*)\right\|^2 \left(\frac{\eta^2(1+(M\eta)^{-1})}{1+\mu\eta} - \frac{c_2}{n\sigma_1 M^2}\right) \\
& - \sum_{i=1}^n \left\|z_i^{(t)} - w^*\right\|^2 \left(c_1(n^{-1}-\tau^{-1}) - \frac{2\eta^2\gamma_*^2 G^2}{(1+\mu\eta)\bar\nu^2}\right) \\
& - \sum_{i=1}^n \left\|g_i^{(t)} - \nabla r_i(w^*)\right\|^2 \left(\frac{c_2}{M^2}(n^{-1}-\tau^{-1}) - \frac{2\eta^2\sigma_n}{1+\mu\eta}\right) .
\end{aligned}
\tag{45}
$$

Let $\eta = b/M$. Our goal is to set the constants $b, c_1, c_2, \tau > 0$ so that the right side above is non-positive and τ is as small as possible. We will require $\tau \ge 2n$ so that $n^{-1} - \tau^{-1} \ge (2n)^{-1}$. Thus, we can have the right side nonpositive with

$$\frac{b}{b+\kappa} - c_1 - \tau^{-1} \ge 0 \tag{46a}$$

$$b(b+1) \ge \frac{c_2}{n\sigma_1}\left(1 + \frac{b}{\kappa}\right) \tag{46b}$$

$$\frac{c_1}{2n} - \frac{2b^2\Gamma}{1+b/\kappa} \ge 0 \tag{46c}$$

$$\frac{c_2}{2n} - \frac{2b^2\sigma_n}{1+b/\kappa} \ge 0 \,. \tag{46d}$$

Let us set $c_1 = \tau^{-1}$. By setting $c_2 = 4\kappa n\sigma_n b^2/(b+\kappa)$, we ensure that (46d) is satisfied. Next, we satisfy (46a) with

$$\frac{b}{b+\kappa} = 2\tau^{-1} \qquad \iff \qquad b = \frac{2\kappa}{\tau - 2} \,.$$

Now, (46b) is an inequality only in τ. It is satisfied with

$$\tau \ge \tau_* := 2 + 2\kappa(4\kappa_\sigma^* - 1) \,.$$

This lets us fix $\tau = \max\{2n, \tau_*\}$ throughout, which leads to the value of η as claimed in the theorem statement. Finally, (46c) requires

$$\frac{4n\kappa^2\Gamma}{\tau - 2} \le 1 \qquad \iff \qquad \bar\nu \ge \frac{\sqrt{n}\kappa\gamma_* G}{M} \min\left\{ \sqrt{\frac{2}{\kappa(4\kappa_\sigma^* - 1)}}, \frac{2}{\sqrt{n}} \right\} .$$

Thus, under these conditions, the right-hand side of (45) is non-negative. Iterating (45) over t updates, we get

$$\mathbb{E}[V^{(t)}] = (1-\tau^{-1})^t V^{(0)} \le \exp(-t/\tau) V^{(0)} \,.$$

To complete the proof, we note that $c_1 \le 1/(2n)$ and

$$c_2 = \frac{4\kappa n\sigma_n b^2}{b+\kappa} = 8\frac{\kappa\kappa_\sigma}{\tau} b \le 8\frac{\kappa\kappa_\sigma}{\kappa(4\kappa_\sigma^* - 1)}\frac{1}{\kappa_\sigma^* - 1} \le \frac{8}{9} \,.$$

This lets us use the fact that ∇r_i is M-Lipschitz to bound

$$V^{(0)} = \left\| w^{(0)} - w^* \right\| + c_1 \sum_{i=1}^{n} \left\| w^{(0)} - w^* \right\|^2 + \frac{c_2}{M^2} \sum_{i=1}^{n} \left\| \nabla r_i(w^{(0)}) - \nabla r_i(w^*) \right\|^2$$
$$\leq (n + 3/2) \left\| w^{(0)} - w^* \right\|^2 .$$

□

G. Technical Results from Convex Analysis

In this section, we collect several results, mostly from Nesterov (2018), that are used throughout the manuscript. In the following, let $\|\cdot\|$ denote an arbitrary norm on $\mathbb{R}^d$ and let $\|\cdot\|_*$ denote its associated dual norm.

The first concerns L-smooth function, or those with L-Lipschitz continuous gradient.

Theorem 38. *(Nesterov, 2018, Theorem 2.1.5) The conditions below are considered for any $x, y \in \mathbb{R}^d$ and $\alpha \in [0, 1]$. The following are equivalent for a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$.*

1. *f is convex and L-smooth with respect to $\|\cdot\|$.*
2. $0 \leq f(y) - f(x) - \langle \nabla f(x), y - x \rangle \leq \frac{L}{2} \|x - y\|^2$.
3. $f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2L} \|\nabla f(x) - \nabla f(y)\|_*^2 \leq f(y)$.
4. $\frac{1}{L} \|\nabla f(x) - \nabla f(y)\|_*^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle$.
5. $0 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle \leq L \|x - y\|^2$.

Next, we detail the properties of strongly convex functions.

Theorem 39. *(Nesterov, 2018, Theorem 2.1.10) If $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex, then for any $x, y \in \mathbb{R}^d$,*

- $f(y) \leq f(x) + \langle f(x), y - x \rangle + \frac{1}{2\mu} \|\nabla f(x) - \nabla f(y)\|_*^2$.
- $\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq \frac{1}{\mu} \|\nabla f(x) - \nabla f(y)\|_*^2$.
- $\mu \|x - y\| \leq \|\nabla f(x) - \nabla f(y)\|_*$.

Finally, functions that are both smooth and strongly convex enjoy a number of relevant primal-dual properties.

Theorem 40. *(Nesterov, 2018, Theorem 2.1.12) If f is both L-smooth and μ-strongly convex, then for any $x, y \in \mathbb{R}^d$,*

$$-\langle \nabla f(x), x - y \rangle = -\tfrac{\mu L}{\mu + L} \|x - y\|^2 - \tfrac{1}{\mu + L} \|\nabla f(x) - \nabla f(y)\|^2 - \langle \nabla f(y), x - y \rangle . \tag{47}$$

Lemma 41. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be μ-strongly convex and M-smooth. Then, we have for any $w, v \in \mathbb{R}^d$,*

$$f(v) \geq f(w) + \nabla f(w)^\top (v - w) + \frac{1}{2(M + \mu)} \|\nabla f(w) - \nabla f(v)\|_2^2 + \frac{\mu}{4} \|w - v\|_2^2.$$

Proof. The function $g = f - \mu\| \cdot \|_2^2/2$ is convex and $M - \mu$ smooth. Hence, we have by line 3 of Thm. 38 for any $w, v \in \mathbb{R}^d$,

$$g(v) \geq g(w) + \nabla g(w)^\top (v - w) + \frac{1}{2(M - \mu)} \|\nabla g(v) - \nabla g(w)\|_2^2.$$

Expanding g and ∇g, we get

$$f(v) \geq f(w) + \nabla f(w)^\top (v - w) + \frac{1}{2(M-\mu)} \|\nabla f(w) - \nabla f(v)\|_2^2 \\ + \frac{\mu M}{2(M-\mu)} \|w - v\|_2^2 - \frac{\mu}{M-\mu} (\nabla f(w) - \nabla f(v))^\top (w - v).$$

Using Young's inequality, that is, $a^\top b \leq \frac{\alpha}{2}\|a\|_2^2 + \frac{\alpha^{-1}}{2}\|b\|_2^2$, we have

$$f(v) \geq f(w) + \nabla f(w)^\top (v - w) + \frac{1 - \alpha\mu}{2(M-\mu)} \|\nabla f(w) - \nabla f(v)\|_2^2 \\ + \frac{\mu(M - \alpha^{-1})}{2(M-\mu)} \|w - v\|_2^2.$$

Taking $\alpha = \frac{2}{\mu+M}$ gives the claim. □

We state a few properties of the prox operator.

Theorem 42 (Co-coercivity of the prox)**.** *If $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex, then we have for any constant $\eta > 0$ that*

$$\langle x - y, \mathrm{prox}_{\eta f}(x) - \mathrm{prox}_{\eta f}(y) \rangle \geq (1 + \eta\mu) \left\|\mathrm{prox}_{\eta f}(x) - \mathrm{prox}_{\eta f}(y)\right\|^2 .$$

The same result applied to the convex conjugate $f^\star$ of f and noting that $\nabla \mathcal{M}_\eta[f](x) = \mathrm{prox}_{f^\star/\eta}(x/\eta)$ gives the following result:

Theorem 43 (Co-coercivity of the prox)**.** *If $f : \mathbb{R}^d \to \mathbb{R}$ is L-smooth, then we have for any constant $\eta > 0$ that*

$$\langle x - y, \nabla \mathcal{M}_\eta[f](x) - \nabla \mathcal{M}_\eta[f](y) \rangle \geq \eta \left(1 + \frac{1}{L\eta}\right) \|\nabla \mathcal{M}_\eta[f](x) - \nabla \mathcal{M}_\eta[f](y)\|^2 .$$

Lemma 44 ((Blondel et al., 2020, Lemma 4))**.** *For a convex function $f : \mathbb{R} \to \mathbb{R}$, if $x_1 \geq x_2$ and $y_2 \geq y_1$, then*

$$f(y_1 - x_1) + f(y_2 - x_2) \geq f(y_2 - x_1) + f(y_1 - x_2).$$

Lemma 45. *Define for $l \in \mathbb{R}^n$,*

$$h(l) = \max_{q \in \mathcal{P}(\sigma)} l^\top q - \frac{\bar{\nu}}{2}\|q - \mathbf{1}_n/n\|_2^2.$$

The function h is $1/\bar{\nu}$-smooth and convex such that for any $l, l' \in \mathbb{R}^n$,

$$\bar{\nu}\|\nabla h(l) - \nabla h(l')\|_2^2 \leq (\nabla h(l) - \nabla h(l'))^\top (l - l') \leq \frac{1}{\bar{\nu}}\|l - l'\|_2^2.$$

H. Experimental Details

H.1. Tasks & Objectives

In all settings, we consider supervised learning tasks specified by losses of the form

$$\ell_i(w) = h(y_i, w^\top \varphi(x_i)),$$

where we consider an input $x_i \in \mathcal{X}$, a feature map $\varphi : \mathcal{X} \to \mathbb{R}^d$, and a label $y_i \in \mathcal{Y}$. The function $h : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ measures the error between the true label and another value which is the prediction in regression and the logit probabilities of the associated classes in classification. In the regression tasks, $\mathcal{Y} = \mathbb{R}$ and we used the squared loss

$$\ell_i(w) = \frac{1}{2}(y_i - w^\top \phi(x_i))^2 .$$

Dataset	d	n_{train}	n_{test}	**Task**	**Source**
`yacht`	6	244	62	Regression	UCI
`energy`	8	614	154	Regression	UCI
`concrete`	8	824	206	Regression	UCI
`kin8nm`	8	6,553	1,639	Regression	OpenML
`power`	4	7,654	1,914	Regression	UCI
`diabetes`	33	4,000	1,000	Binary Classification	Fairlearn
`acsincome`	202	4,000	1,000	Regression	Fairlearn
`amazon`	535	10,000	10,000	Multiclass Classification	WILDS
`iwildcam`	9420	20,000	5,000	Multiclass Classification	WILDS

Table 2: Dataset attributes and dimensionality d, train sample size n_{train}, and test sample size n_{test}.

For binary classification, we have $\mathcal{Y} = \{-1, 1\}$, denoting a negative and positive class. We used the binary logistic loss

$$\ell_i(w) = -y_i x_i^\top w + \ln(1 + e^{x_i^\top w}) .$$

For multiclass classification, $\mathcal{Y} = \{1, \ldots, C\}$ where C is the number of classes. We used the multinomial logistic loss:

$$\ell_i(w) = -\ln p_{y_i}(x_i; w), \text{ where } p_{y_i}(x_i; w) := \frac{\exp\left(w_{\cdot y}^\top x_i\right)}{\sum_{y'=1}^{C} \exp\left(w_{\cdot y'}^\top x_i\right)}, \; w \in \mathbb{R}^{d \times C}$$

The design matrix $(\varphi(x_1), \ldots, \varphi(x_n)) \in \mathbb{R}^{n \times d}$ is standardized to have columns with zero mean and unit variance, and the estimated mean and variance from the training set is used to standardize the test sets as well. Our final objectives are of the form

$$\mathcal{L}_\sigma(w) = \max_{q \in \mathcal{P}(\sigma)} \sum_{i=1}^{n} q_i \ell_i(w) - \nu n \left\| q - \mathbf{1}_n / n \right\|_2^2 + \frac{\mu}{2} \left\| w \right\|_2^2$$

for shift cost $\nu \geq 0$ and regularization constant $\mu \geq 0$.

H.2. Datasets

We detail the datasets used in the experiments. If not specified below, the input space $\mathcal{X} = \mathbb{R}^d$ and φ is the identity map. The sample sizes, dimensions, and source of the datasets are summarized in Tab. 2, where d refers to the dimension of each $\varphi(x_i)$.

(a) `yacht`: prediction of the residuary resistance of a sailing yacht based on its physical attributes (Tsanas & Xifara, 2012).

(b) `energy`: prediction of the cooling load of a building based on its physical attributes (Baressi Segota et al., 2020).

(c) `concrete`: prediction of the compressive strength of a concrete type based on its physical and chemical attributes (Yeh, 2006).

(d) `kin8nm`: prediction of the distance of an 8 link all-revolute robot arm to a spatial endpoint (Akujuobi & Zhang, 2017).

(e) `power`: prediction of net hourly electrical energy output of a power plant given environmental factors (Tüfekci, 2014).

(f) `diabetes`: prediction of readmission for diabetes patients based on 10 years worth of clinical care data at 130 US hospitals (Rizvi et al., 2014).

(g) `acsincome`: prediction of income of US adults given features compiled from the American Community Survey (ACS) Public Use Microdata Sample (PUMS) (Ding et al., 2021).

(h) `amazon`: prediction of the review score of a sentence taken from Amazon products. Each input $x \in \mathcal{X}$ is a sentence in natural language and the feature map $\varphi(x) \in \mathbb{R}^d$ is generated by the following steps:

- A BERT neural network (Devlin et al., 2019) (fine-tuned on $10,000$ held-out examples) is applied to the text x_i, resulting in vector x'_i.
- The $x'_1, \ldots, x'_n$ are normalized to have unit norm.
- Principle Components Analysis (PCA) is applied, resulting in 105 components that explain 99% of the variance, resulting in vectors $x''_i \in \mathbb{R}^{105}$. The d in Tab. 2 refers to the total dimension of the parameter vectors for all 5 classes.

(i) `iwildcam`: prediction of an animal or flora in an image from wilderness camera traps, with heterogeneity in illumination, camera angle, background, vegetation, color, and relative animal frequencies (Beery et al., 2020). Each input $x \in \mathcal{X}$ is an image the feature map $\varphi(x) \in \mathbb{R}^d$ is generated by the following steps:

- A ResNet50 neural network (He et al., 2016) that is pretrained on ImageNet (Deng et al., 2009) is applied to the image x_i, resulting in vector x'_i.
- The $x'_1, \ldots, x'_n$ are normalized to have unit norm.
- Principle Components Analysis (PCA) is applied, resulting in $d = 157$ components that explain 99% of the variance. The d in Tab. 2 refers to the total dimension of the parameter vectors for all 60 classes.

H.3. Hyperparameter Selection

We fix a minibatch size of 64 SGD and SRDA and an epoch length of $N = n$ for LSVRG. For SaddleSAGA we consider three schemes for selecting the primal and dual learning rates that reduce to searching for a single parameter $\eta > 0$, as described in Appx. I. In practice, the regularization parameter μ and shift cost ν are tuned by a statistical metric, i.e. generalization error as measured on a validation set. We study the optimization performance of the methods for multiple values of each in Appx. I.

For the tuned hyperparameters, we use the following method. Let $k \in \{1, \ldots, K\}$ be a seed that determines algorithmic randomness. This corresponds to sampling a minibatch without replacement for SGD and SRDA and a single sampled index for SaddleSAGA, LSVRG, and SpecSAGA. Letting $\mathcal{L}_k(\eta)$ denote the average value of the training loss of the last ten passes using learning rate η and seed k, the quantity $\mathcal{L}(\eta) = \frac{1}{K}\sum_{k=1}^K \mathcal{L}_k(\eta)$ was minimized to select η. The learning rate η is chosen in the set $\{1\times10^{-4}, 3\times10^{-4}, 1\times10^{-3}, 3\times10^{-3}, 1\times10^{-2}, 3\times10^{-2}, 1\times10^{-1}, 3\times10^{-1}, 1\times10^{0}, 3\times10^{0}\}$, with two orders of magnitude lower numbers used in `acsincome` due to its sparsity. We discard any learning rates that cause the optimizer to diverge for any seed.

H.4. Compute Environment

No GPUs were used in the study; Experiments were run on a CPU workstation with an Intel i9 processor, a clock speed of 2.80GHz, 32 virtual cores, and 126G of memory. The code used in this project was written in Python 3 using the PyTorch and Numba packages for automatic differentiation and just-in-time compilation, respectively.

I. Additional Experiments

Varying Risk Parameters. We study the effect of varying the risk parameters, that is (p, b, γ) for the p-superquantile, b-extremile, γ-ESRM, choosing spectral to increase the condition number $\kappa_\sigma = n\sigma_n$ compared to the experiments in the main text. We use $p = 0.25$, $b = 2.5$, and $\gamma = 1/e^{-2}$ to generate "hard" version of the superquantile, extremile, and ESRM. Fig. 6 plots the corresponding training curves for four datasets of varying sample sizes: `yacht`, `energy`, `concrete`, and `iwildcam`. We see that the comparison of methods is the same as the original methods, that is that SpecSAGA performs the best or close to best in terms of optimization trajectories. Except on `concrete`, SaddleSAGA generally matches the performance of SpecSAGA. The trajectory of LSVRG is noticeably noisier than on the original settings; we hypothesize that the bias accrued by this epoch-based algorithm is exacerbated by the skewness in the spectrum, as mentioned in Mehta et al. (2023, Proposition 1).

Lowering or Removing Shift Cost. A relevant setting is the low or no shift cost regime, as this allows the adversary to make arbitrary distribution shifts (while still constrained to $\mathcal{P}(\sigma)$). These settings correspond to $\nu = 10^{-3}$ and $\nu = 0$, respectively. The low-cost experiment is displayed in Fig. 7 while Fig. 8 displays these curves for the no-cost experiment.

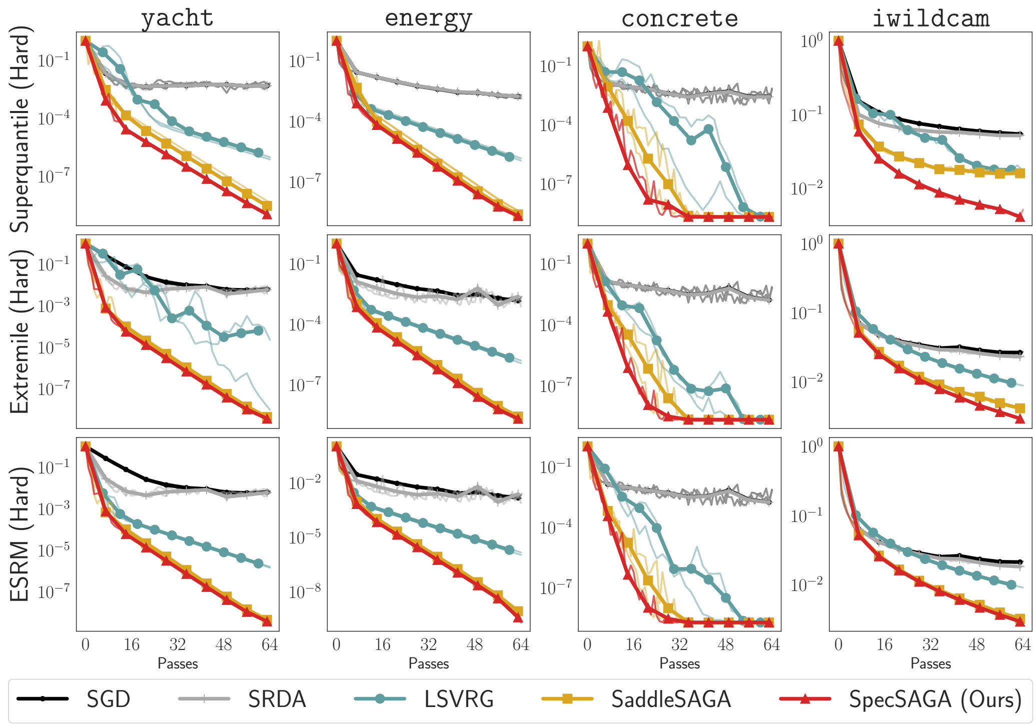

Figure 6: **Harder risk parameter settings.** Each row represents a different "hard" variant of the superquantile, extremile, and ESRM spectra. Columns represent different datasets. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.

When $\nu = 0$, the optimization problem can equivalently be written as

$$\min_{w \in \mathbb{R}^d} \left[\max_{q \in \mathcal{P}(\sigma)} q^\top \ell(w) + \frac{\mu}{2} \|w\|_2^2 = \sum_{i=1}^{n} \sigma_i \ell_{(i)}(w) + \frac{\mu}{2} \|w\|_2^2 \right].$$

In this case, we always have that $q^{\text{opt}}(l) = (\sigma_{\pi^{-1}(1)}, \ldots, \sigma_{\pi^{-1}(n)})$, where π sorts l. Here, w is chosen to optimize a linear combination of order statistics of the losses. In the low shift cost settings, performance trends are qualitatively similar to those seen from $\nu = 1$. Interestingly, for the no-cost setting, LSVRG, SaddleSAGA, and SpecSAGA seem to converge linearly empirically even without smoothness of the objective.

Lowering Regularization. Next, we decrease the ℓ_2-regularization from $\mu = 1/n$ to $\mu = 1/(10n)$ and $\mu = 1/(100n)$. These settings are plotted in Fig. 9 and Fig. 10, respectively. Performance rankings among methods reflect those of the original parameters. For five of the six datasets, that is `yacht`, `energy`, `concrete`, `kin8nm`, and `power`, the regression tasks involve optimizing the squared error. This function is already strongly convex, with constant depending on the smallest eigenvalue of the empirical second moment matrix. When assuming that the input data vectors are bounded, this function is also G-Lipschitz. Thus, if the problem is already well-conditioned, we may observe similar behavior even at negligible regularization ($\mu = 5 \cdot 10^{-7}$ for `iwildcam`, for example).

Comparison of Saddle-Point and Moreau Variants. Finally, observe in Fig. 11 the comparison of SaddleSAGA variants (Appx. E), as well as the Moreau version of SpecSAGA using Moreau envelope-based oracles (Appx. F). There are variants shown.

- **Primal LR = Dual LR:** The original variant of Palaniappan & Bach (2016), in which the primal and dual learning rates are set to be equal and searched as a single hyperparameter.

- **Search Dual LR:** Here, the primal learning rate is fixed as the optimal one for SpecSAGA, and the dual learning rate is searched as a single hyperparameter.

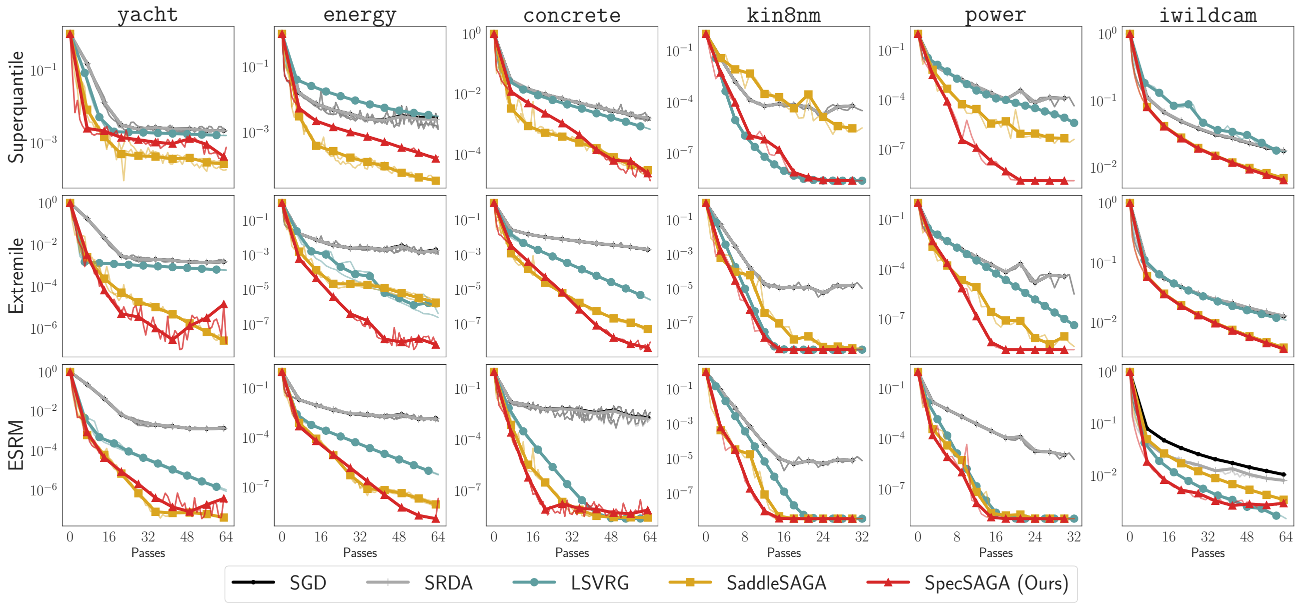

Figure 7: **Low shift cost settings.** Each row represents a different spectral risk objective with $\nu = 10^{-3}$ (instead of $\nu = 1$) while each column represents a different datasets. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.

- **Primal-Dual Heuristic:** In this version, used as the "SaddleSAGA" baseline in the main text, the dual learning rate is set to be $10n$ times smaller than the primal learning rate.
- **SpecSAGA-Moreau:** The Moreau-envelope version of SpecSAGA using proximal oracles.

We find that all methods besides the original variant (primal LR = dual LR) perform comparably on `yacht`, `energy`, `concrete`, `kin8nm`, and `power`. Notably, the ProxSAGA method performs similarly to SpecSAGA and the saddle point-based baselines. While using the Moreau envelope results in accelerated rates in the ERM setting (Defazio, 2016), we find that the convergence rate is the same empirically. This phenomenon is in agreement with Thm. 35, which states that ProxSAGA will achieve the same linear convergence rate as SpecSAGA, but will require a much less stringent condition on the shift cost ν than in the case of SpecSAGA.

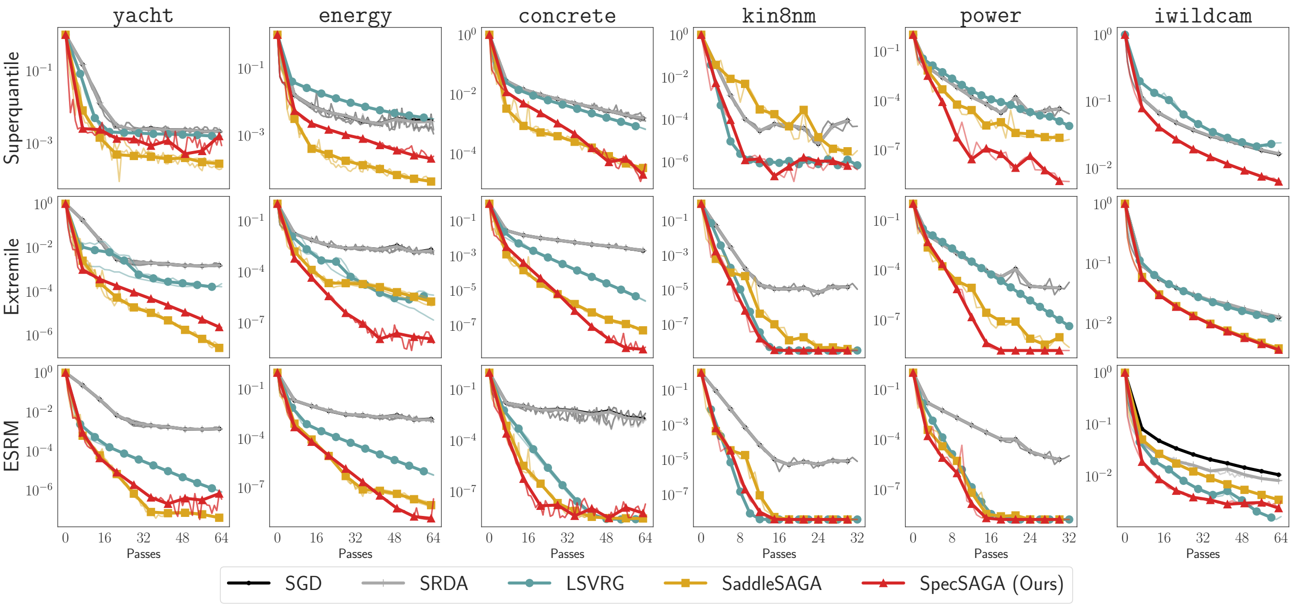

Figure 8: **No shift cost settings.** Each row represents a different spectral risk objective with $\nu = 0$ (instead of $\nu = 1$) while each column represents a different datasets. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.

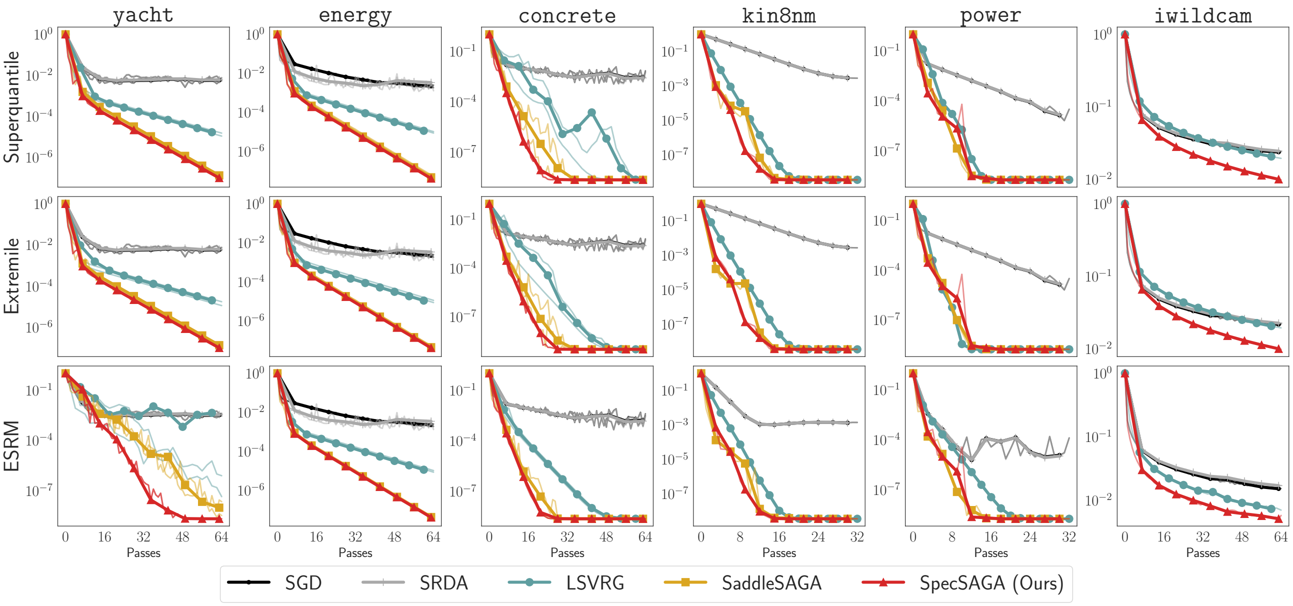

Figure 9: **Reduced ℓ_2-regularization settings** ($\mu = 1/(10n)$**.** Each row represents a different spectral risk objective with $\mu = 1/(10n)$ (instead of $\mu = 1/n$) while each column represents a different dataset. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.

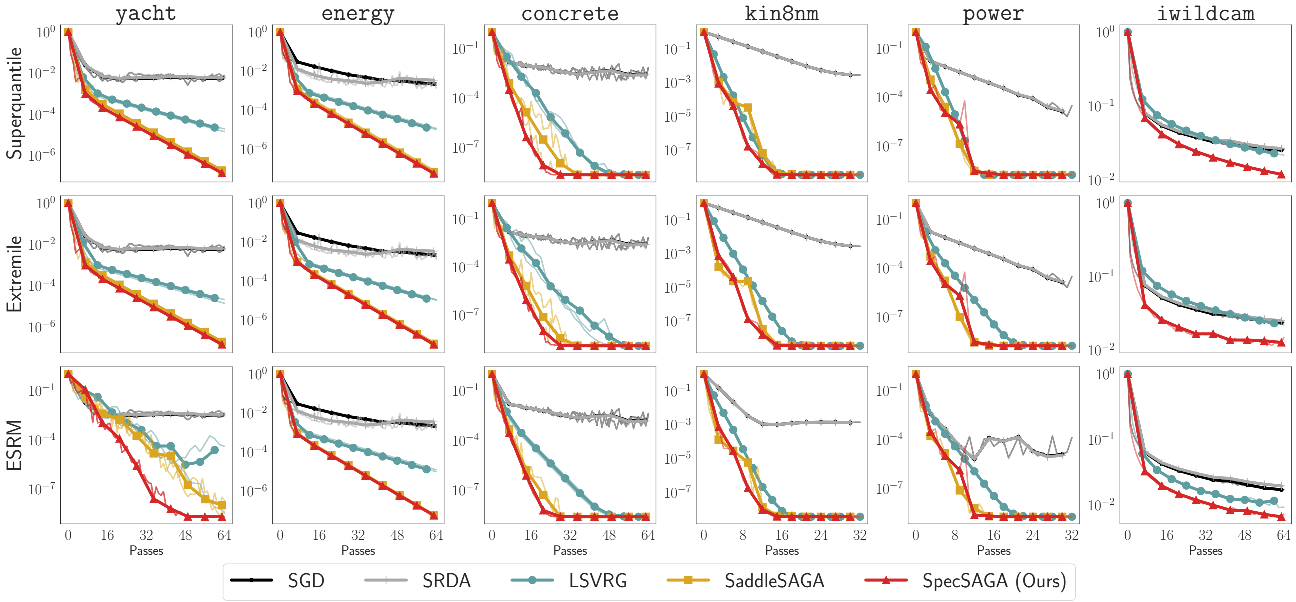

Figure 10: **Low ℓ_2-regularization settings ($\mu = 1/(100n)$.** Each row represents a different spectral risk objective with $\mu = 1/(100n)$ (instead of $\mu = 1/n$) while each column represents a different dataset. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.

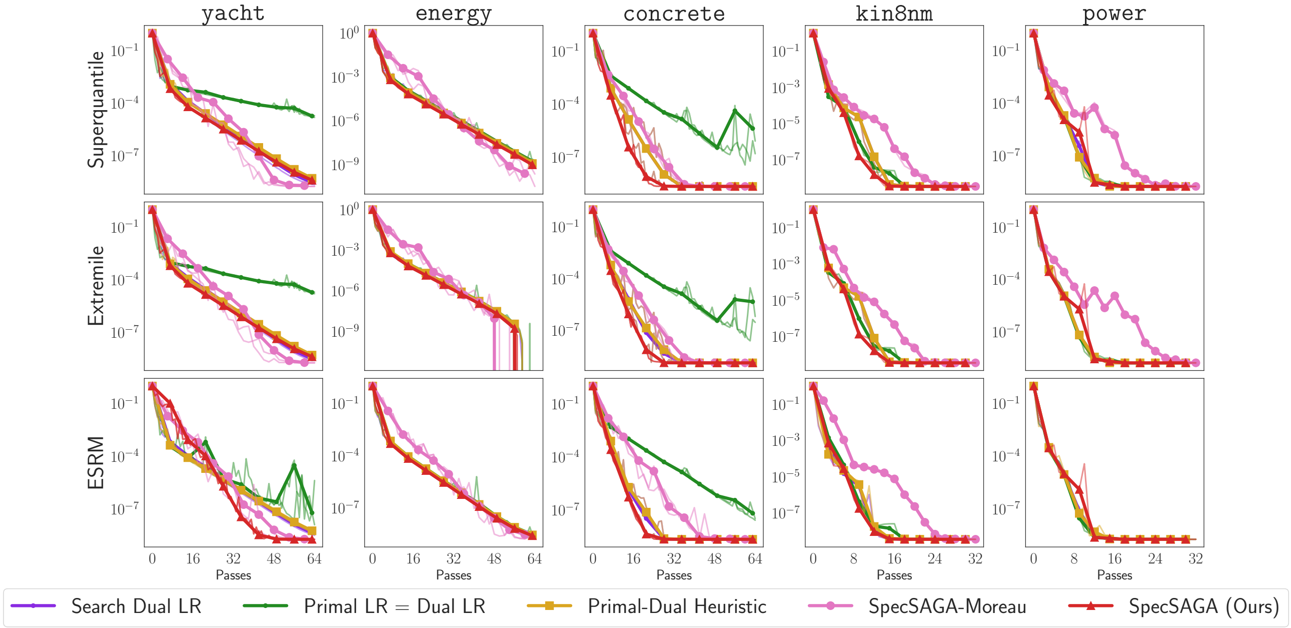

Figure 11: **SaddleSAGA and SpecSAGA-Moreau method comparisons.** Each row represents a different spectral risk objective while each column represents a different dataset. Suboptimality (6) is measured on the y-axis while the x-axis measures the total number of gradient evaluations made divided by n, i.e. the number of passes through the training set.