TRAINING WITH ML-LABELED DATA VIA STOCHASTIC FIRST- AND ZEROTH-ORDER OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Training a machine learning (ML) model is an optimization process that typically requires a set of examples with ground-truth labels beforehand. Obtaining these ground truths usually requires laborious human effort and can be costly. The scalability of human labeling becomes a concern when there are a large number of data points that need to be labeled. To address this practical issue, we consider an optimization setting where a subset of samples has ground-truth labels, while the remaining data points were labeled by an ML predictor, which could potentially assign incorrect labels to those points. We propose both first-order and zeroth-order stochastic optimization methods for training on datasets consisting of a mix of true-labeled and ML-labeled samples. We further conduct comprehensive empirical studies from training a linear regression model to fine-tuning a large language model (LLM) to showcase the effectiveness of the proposed algorithms for training with ML-labeled data.

1 INTRODUCTION

Stochastic optimization is the de facto approach for training modern ML models (Robbins & Monro, 1951; Zhang, 2004; Bottou & Bousquet, 2007; Rakhlin et al., 2012; Le Roux et al., 2012; Shamir & Zhang, 2013; Ghadimi & Lan, 2013; Needell et al., 2014; Jain et al., 2018; Nguyen et al., 2018; Gower et al., 2019; Cutkosky & Orabona, 2019; Li et al., 2021). The common machinery of first-order stochastic methods involves using an easy-to-compute estimate of the gradient (a.k.a. a stochastic gradient) to update the underlying ML model. This results in a much lower iteration cost than deterministic methods such as gradient descent, which require gradient computations for all data points—a requirement that can become impractical at scale. The idea of using a mini-batch of samples to construct a gradient estimate can be naturally extended to zeroth-order optimization (Ghadimi & Lan, 2013; Duchi et al., 2015; Nesterov & Spokoiny, 2017; Balasubramanian & Ghadimi, 2022; Zhang et al., 2025a), where only function-value evaluations are used to construct an estimate. Recent work (Malladi et al., 2023) has observed that the memory footprint of a zeroth-order method is a few times smaller than that of its first-order counterpart in practice, which offers opportunities for fine-tuning an LLM using a single modern GPU.

In this work, we investigate stochastic optimization methods for datasets that consist of human-labeled and machine-labeled data. Specifically, we assume access to a set of samples with ground-truth labels, denoted as $\{(\mathbf{x}_i, y_i)\}_{i\geq 1}$, where i is the sample index, and $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ are the feature vector and the true label for point i, respectively. Moreover, ML labels are also available for these data points with true labels, denoted as $\{f(\mathbf{x}_i)\}_{i\geq 1}$. In addition, we have a set of data without ground-truth labels but with ML-generated labels, denoted as $\{(\tilde{\mathbf{x}}_j, f(\tilde{\mathbf{x}}_j))\}_{j\geq 1}$, where j is the sample index for ML-labeled data and that $f(\tilde{\mathbf{x}}_j)$ represents the ML-label for feature vector $\tilde{\mathbf{x}}_j \in \mathbb{R}^d$. We will assume that the feature vectors $\{\mathbf{x}_i\}_{i\geq 1}$ and $\{\tilde{\mathbf{x}}_j\}_{j\geq 1}$ are drawn i.i.d. from the same unknown distribution and that the ML-labeling rule $f(\cdot)$ is independent from these samples.

This scenario is relevant given that modern large-scale datasets typically demand substantial human labeling and collaborative crowdsourcing (Deng et al., 2009; Lin et al., 2014; Rajpurkar et al., 2016; Mayo et al., 2023). Hence, the use of an ML predictor in labeling can help significantly reduce the associated costs and labor. While recent trends in active learning focus on designing methods to draw human attention to points where ML predictors lack confidence (Beygelzimer et al., 2009),

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

this work considers a different and complementary direction. Specifically, we would like to answer the following question: *Can we leverage the ML-labels to accelerate the optimization process?*

In ML, a common objective function takes the following form:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \mathfrak{L}(\mathbf{w}), \quad \text{where} \quad \mathfrak{L}(\mathbf{w}) := \mathbb{E}_{(\mathbf{x},y)}[\ell(\mathbf{w};\mathbf{x},y)]. \tag{1}$$

Here, $\ell(\cdot;\mathbf{x},y)$ is the loss function that evaluates the predictive performance of a model $\mathbf{w}$ on a sample $(\mathbf{x},y)$. In particular, the objective function encompasses the finite-sum problem, i.e., $\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n \ell(\mathbf{w};\mathbf{x}_i,y_i)$, which fits into (1) by interpreting the expectation as being taken with respect to a uniform draw from the training data.

We aim to leverage ML-labeled data to more efficiently solving stochastic optimization problem (1), compared to the setting without access to ML-labeled data. Our results can be viewed as a counterpart of prediction-powered inference (PPI) in statistics (Angelopoulos et al., 2023a;b; Sfyraki & Wang, 2026b). In PPI, the primary objective is to construct confidence intervals for parameter estimation by leveraging ML-labeled data. More specifically, the statistical inference problem of PPI shares a similar setting in which a small gold-standard dataset consisting of feature and outcome pairs $\{(\mathbf{x}_i,y_i)\}_{i=1}^n$ is available, along with a large collection of unlabeled samples without true outcomes $\{\tilde{\mathbf{x}}_j\}_{j=1}^N$, i.e., $N \gg n$. Additionally, an ML model provides its predicted-labels for all the data points, denoted as $\{f(\mathbf{x}_i)\}_{i=1}^n$ and $\{f(\tilde{\mathbf{x}}_j)\}_{j=1}^N$. On the other hand, their goal is to achieve statistical efficiency in estimating a parameter of interest, such as the population mean of the true outcomes $\mathbb{E}[y_i]$, while incorporating information from ML-labeled data with potentially biased predictions. To this end, Angelopoulos et al. (2023a;b) propose unbiased (or bias-corrected) estimators that combine gold-standard and ML-labeled data points. They show that the resulting confidence intervals are asymptotically valid and, in certain regimes where the predictor is sufficiently accurate and the number of ML-labeled samples is large, can be strictly shorter than those constructed using only data with true labels.

Our work is inspired by both the emergence of PPI in statistics (Angelopoulos et al., 2023a;b; Sfyraki & Wang, 2026b) and a recent work by Shoham et al. (2025), who extended the idea of leveraging ML-labeled data for statistical inference to the semi-supervised model training by proposing a gradient estimator that incorporates ML-labeled data points. Specifically, given n samples with true labels and N ML-labeled samples, one can construct a *prediction-powered* gradient estimate:

$$g_{\mathbf{w}}^{\mathrm{PP}} := \tfrac{1}{N}\textstyle\sum_{j=1}^N \nabla\ell\big(\mathbf{w};\tilde{\mathbf{x}}_j,f(\tilde{\mathbf{x}}_j)\big) + \tfrac{1}{n}\sum_{i=1}^n \Big(\nabla\ell(\mathbf{w};\mathbf{x}_i,y_i) - \nabla\ell(\mathbf{w};\mathbf{x}_i,f(\mathbf{x}_i))\Big).$$

Under the assumption that all the samples $\{\tilde{\mathbf{x}}_j\}_{j=1}^N$ and $\{\mathbf{x}_i\}_{i=1}^n$ are drawn i.i.d. from the same distribution, together with the condition that the ML-labeling rule $f(\cdot)$ is independent of the observed samples $\{\tilde{\mathbf{x}}_j\}_{j=1}^N$ and $\{\mathbf{x}_i\}_{i=1}^n$, the gradient estimator $g_{\mathbf{w}}^{\mathrm{PP}}$ is unbiased for $\nabla\mathfrak{L}(\mathbf{w})$. Motivated by PPI++ (Angelopoulos et al., 2023b), Shoham et al. (2025) further consider a tuning parameter that controls the contribution of the ML-labeled components in the prediction-powered gradient estimate.

They further combine AdaGrad (Duchi et al., 2011; Levy, 2017) with the prediction-powered gradient estimate and an online learning algorithm (Hazan, 2022; Orabona, 2026) that dynamically tunes λ during optimization. They evaluate their method against a couple of baselines, including one using the best fixed λ chosen in hindsight, and demonstrate its effectiveness in terms of final mean absolute error and final mean squared error on test sets for some regression and classification tasks. However, their empirical evaluation focuses mainly on the predictive performance of the trained model rather than potential optimization speedups. Nevertheless, a plot of training loss versus epoch for CIFAR-10 in their supplementary (Figure S12 in Shoham et al. (2025)) suggests that incorporating ML-labeled data might accelerate training. Intrigued by their plot, we investigate further to answer the aforementioned question, with the aim of strengthening the foundation of optimization with ML-labeled data.

To give the reader a flavor of our contributions, we summarize them here. We first study standard stochastic gradient descent (SGD), a classical, general-purpose first-order stochastic optimization algorithm, and propose two theoretically grounded schemes that allow SGD leverage ML-labels for variance reduction. Let T be the number of iterations. In terms of the common stationarity metric in general smooth nonconvex optimization, i.e., $\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]$, we provide convergence rate results that characterize the potential improvement in SGD achieved by incorporating ML labels, compared to using only true labels. The proposed schemes aim to dynamically update the

Algorithm 1 SGD with ML-labeled data

Require: Initial point $\mathbf{w}_1 \in \mathbb{R}^d$ and step sizes $\{\eta_t\}_{t=1}^T$.

1: **for** $t = 1, \ldots, T$ **do**

2: Sample a minibatch of points with true labels B_t and compute $g^y_{t,\mathbf{w}_t} := \frac{1}{|B_t|} \sum_{(\mathbf{x},y)\in B_t} \nabla \ell\left(\mathbf{w}_t; \mathbf{x}, y\right)$ and $g^f_{t,\mathbf{w}_t} := \frac{1}{|B_t|} \sum_{(\mathbf{x},y)\in B_t} \nabla \ell\left(\mathbf{w}_t; \mathbf{x}, f(\mathbf{x})\right)$.

3: Sample an ML-labeled minibatch $\tilde{B}_t$ and compute $\tilde{g}^f_{t,\mathbf{w}_t} := \frac{1}{|\tilde{B}_t|} \sum_{\tilde{\mathbf{x}}\in \tilde{B}_t} \nabla \ell\left(\mathbf{w}_t; \tilde{\mathbf{x}}, f\left(\tilde{\mathbf{x}}\right)\right)$.

4: Update $\boldsymbol{\lambda}_t$ using either the scalar tuning scheme (2) or the vector tuning scheme (3).

5: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \, \mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$, where $\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} := g^y_{t,\mathbf{w}_t} + \boldsymbol{\lambda}_t \odot \left(\tilde{g}^f_{t,\mathbf{w}_t} - g^f_{t,\mathbf{w}_t}\right)$.

6: **end for**

contribution of ML labels when constructing a prediction-powered gradient estimate at each step and to advance the perspective of Shoham et al. (2025). Specifically, unlike prior work, we do not assume that the loss value $\ell(\mathbf{w}; \mathbf{x}, z)$ or the size of the stochastic gradient $\nabla_{\mathbf{w}} \ell(\mathbf{w}; \mathbf{x}, z)$ is bounded, where z here is either the true label y or the ML label $f(\mathbf{x})$. At the same time, each of our proposed schemes still admits a lightweight closed-form update obtained by minimizing a quadratic function. Furthermore, whereas prior works (Angelopoulos et al., 2023b; Shoham et al., 2025) restrict the weights on the ML-labeled components in the estimates to the interval $[0, 1]$, we lift this constraint so that the weights determining the contribution of these components at each step can take any real value, including those outside this interval. This allows the underlying optimization methods to fully exploit the ML-labeled data to reduce variance and further accelerate the optimization process. Of the two schemes that we propose, one updates online a single tuning parameter shared across all dimensions, as in prior work, while the other uses different weights to combine the components of the gradient estimate in different dimensions.

We then extend the machinery that we develop for first-order optimization to zeroth-order stochastic optimization, where we build upon a common stochastic zeroth-order oracle based on Gaussian perturbations, which was also considered in prior seminal works, e.g., Ghadimi & Lan (2013); Balasubramanian & Ghadimi (2022). Interestingly, we find a similar variance reduction mechanism in the iteration complexity analysis of the stochastic gradient-free method with the two proposed online tuning schemes adapted to zeroth-order optimization. To our knowledge, this is the first result on stochastic gradient-free optimization with ML-labeled data in the literature. Finally, to corroborate our theoretical results, we conduct experiments to showcase the effectiveness of the proposed method in accelerating optimization with ML-labeled data. These experiments range from fitting a linear regression model to training a image classifier and fine-tuning an LLM.

2 First-Order Stochastic Optimization with ML-labeled Data

Notations. We begin by introducing some notations first. Throughout this paper, we use boldface $\boldsymbol{\lambda} \in \mathbb{R}^d$ to denote a vector of tuning parameters, while $\lambda \in \mathbb{R}$ denotes a scalar tuning parameter. We use $\|\cdot\|$ to denote the Euclidean norm throughout. For any vector $\mathbf{v}$, $\mathbf{v}[i]$ denotes its i_{th}-element Furthermore, $\odot$ represents the Hadamard product; specifically, for any pair of vectors $\mathbf{w}, \mathbf{v} \in \mathbb{R}^d$, $\mathbf{w} \odot \mathbf{v} \in \mathbb{R}^d$ is their element-wise product. We further note that, given a scalar tuning parameter $\lambda \in \mathbb{R}$, a scalar-vector multiplication $\lambda \mathbf{v}$ can be rewritten as $\lambda \mathbf{v} = \lambda \mathbf{1} \odot \mathbf{v}$, where $\mathbf{1} \in \mathbb{R}^d$ is the all-ones vector. For conciseness, we will use this fact to present the proposed online tuning schemes of this work in a single algorithmic description in the following, i.e., Algorithm 1 and Algorithm 2. The distinction between scalar and vector tuning parameters should be clear from the context and will be highlighted when necessary. We will let $\mathcal{F}_t$ denote the filtration representing the information available up to and including step t. For any collection of random variables or random elements, $\sigma(\cdot)$ denotes their generated sigma-algebra. Also, we use $\Delta_1 := \mathfrak{L}(\mathbf{w}_1) - \inf_{\mathbf{w}\in\mathbb{R}^d} \mathfrak{L}(\mathbf{w})$ to represent the initial optimality gap, where $\mathbf{w}_1$ is an initial point.

2.1 Our Methods

Algorithm 1 describes the proposed SGD with ML-labeled data. For further notation brevity, in the following, we will also denote

$$\mathbf{g}_t := g^y_{t,\mathbf{w}_t}, \qquad \mathbf{d}_t := \tilde{g}^f_{t,\mathbf{w}_t} - g^f_{t,\mathbf{w}_t}.$$

Hence the prediction-powered gradient estimate at t in Algorithm 1 can be rewritten as $\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} = \mathbf{g}_t + \lambda \mathbf{d}_t$.

The critical step for effectively leveraging the ML-labeled data lies in how to update the parameter $\boldsymbol{\lambda}_t$ at each time t, i.e., Step 4 in Algorithm 1, which determines how much contribution from the stochastic gradients based on ML-labels should be weighted. We propose two online update schemes: one uses a single scalar tuning parameter λ, and the other weights the contribution differently across dimensions.

For the scalar case, we let $\boldsymbol{\lambda}_t \leftarrow \lambda_t \odot \mathbf{1}$ in Algorithm 1, where $\mathbf{1}$ is the all-one-vector; consequently one can write $\mathbf{g}_t + \boldsymbol{\lambda}_t \odot \mathbf{d}_t = \mathbf{g}_t + \lambda_t \mathbf{d}_t$ for this case. This suggests that only a single scalar parameter λ_t is required to be maintained in each step. We propose the following update scheme:

$$\textbf{(Scalar tuning)} \qquad \lambda_t = -\frac{\sum_{s=1}^{t-1} \langle \mathbf{g}_s, \mathbf{d}_s \rangle}{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}, \quad \text{if } \sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 > 0, \tag{2}$$

where we let $\lambda_t = 0$ when $\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 = 0$, such as the initial step. We remark that as long as the **d**ifference of the stochastic gradient components $\mathbf{d}_t$ at any time point is non-zero, the sum is evidently nonzero as well, which is likely to hold as early as $t = 2$ in practice.

The other scheme uses the full vector tuning parameter $\boldsymbol{\lambda}_t$. Specifically, for each $i \in [d]$, we update

$$\textbf{(Vector tuning)} \quad \boldsymbol{\lambda}_t[i] = -\frac{\sum_{s=1}^{t-1} \mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t-1} \mathbf{d}_s[i]^2}, \quad \text{if } \sum_{s=1}^{t-1} \mathbf{d}_s[i]^2 > 0, \tag{3}$$

and we let $\boldsymbol{\lambda}_t[i] = 0$, otherwise.

Algorithm Design Principle. We now explain the mechanism underlying the proposed online tuning schemes.

Recall the standard SGD update: $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{G}_t$, where $\mathcal{G}_t$ is a stochastic gradient. The machinery of standard SGD is usually built on the premise that $\mathcal{G}_t$ is a conditionally unbiased estimator, i.e., $\mathbb{E}[\mathcal{G}_t \mid \mathcal{F}_{t-1}] = \nabla \mathfrak{L}(\mathbf{w}_t)$. In our problem, $\mathcal{G}_t = \mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$. If the weight $\boldsymbol{\lambda}_t$ in $\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$ is not $\mathcal{F}_{t-1}$-measurable, then in general, $\mathbb{E}[\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \mid \mathcal{F}_{t-1}] \neq \nabla \mathfrak{L}(\mathbf{w}_t)$. In other words, the prediction-powered gradient estimate $\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$ is not necessarily unbiased unless $\boldsymbol{\lambda}_t$ is $\mathcal{F}_{t-1}$-measurable. While convergence results for SGD with biased gradient estimates are established in the literature, such methods typically converge only to a neighborhood, with a nonvanishing bias term in the bound of the iteration complexity, see e.g., Demidovich et al. (2023). Hence, we do not pursue biased SGD in this work and instead require $\boldsymbol{\lambda}_t$ to be $\mathcal{F}_{t-1}$-measurable. To satisfy this requirement, the update of $\boldsymbol{\lambda}_t$ cannot use stochastic gradient information from step t; only information available up to the previous step can be used. Therefore, the problem is inherently an *online learning* problem, where the learner chooses an action using only past information before observing the current round's loss function.

Scalar Tuning as a Game

for $t = 1, \ldots, T$ **do**
  Play $\lambda_t \in \mathbb{R}$.
  Observe the loss function $\phi_t(\lambda) := \|\mathbf{g}_t + \lambda \mathbf{d}_t\|^2$
  Incur loss $\phi_t(\lambda_t)$.
end for

Vector Tuning as a Game

for $t = 1, \ldots, T$ **do**
  Play $\boldsymbol{\lambda}_t \in \mathbb{R}^d$.
  Observe the loss function $\phi_t(\boldsymbol{\lambda}_t) := \|\mathbf{g}_t + \boldsymbol{\lambda}_t \odot \mathbf{d}_t\|^2$
  Incur loss $\phi_t(\boldsymbol{\lambda}_t)$.
end for

Figure 1: Game-theoretic interpretation of the online tuning schemes

Figure 1 presents the game-theoretic interpretation of the online tuning schemes. The loss functions in these games arise naturally from an intermediate step in the standard convergence analysis of SGD with unbiased gradient estimates, which yields

$$\sum_{t=1}^{T} \left(\eta_t - \tfrac{\beta}{2}\eta_t^2\right) \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \Delta_1 + \tfrac{\beta}{2} \sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\|^2\right] - \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right],$$

where we replicate the detailed derivation in Appendix B. Hence, it is clear that one should update the weight $\boldsymbol{\lambda}_t$ to make this bound small, where $\boldsymbol{\lambda}_t$ enters the bound through $\mathbb{E}_{t-1}\left[\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\|^2\right]$. This motivates using the squared norm of the gradient estimator as the loss function in each of our online learning problems, i.e., $\phi_t(\cdot)$ in Figure 1.

The goal of online learning is to minimize regret (Hazan, 2022; Orabona, 2026), i.e., minimizing $\mathrm{Regret}^{\phi}_T(\lambda^*) := \sum_{t=1}^{T} \phi_t(\lambda_t) - \sum_{t=1}^{T} \phi_t(\lambda^*)$, where λ^* is a comparator in the same decision space

as the online learner, namely $\mathbb{R}$ in the scalar tuning. Their counterparts in the vector tuning game are defined analogously. In both games, the quadratic loss functions motivate the use of the Follow-the-Leader (FTL) strategy, which is known to enjoy a logarithmic regret; see e.g., Section 2.2 in Shalev-Shwartz (2011) and Section 3.3 in Wang et al. (2024). FTL selects a minimizer of the cumulative loss observed so far. In the scalar tuning game, this gives $\lambda_t \in \arg\min_{\lambda\in\mathbb{R}} \sum_{s=1}^{t-1} \phi_s(\lambda)$, while in the vector tuning game, the update is $\boldsymbol{\lambda}_t \in \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^d} \sum_{s=1}^{t-1} \phi_s(\boldsymbol{\lambda})$. Here, we have overloaded the notation $\phi_t(\cdot)$ to denote the function in scalar tuning or vector tuning, as determined by context; see Figure 1. Routine calculations show that these lead to the closed-form updates (2) and (3), respectively, where we provide the details in Appendix B.

2.2 Theoretical analysis

We first recall a couple of notions in optimization.

Gradient Lipschitzness and Smoothness. A differentiable function $\ell(\cdot)$ has β-Lipschitz gradients if $\|\nabla\ell(\mathbf{w}) - \nabla\ell(\mathbf{v})\| \le \beta\|\mathbf{w}-\mathbf{v}\|$. Furthermore, a differentiable function $\ell(\cdot)$ is β-smooth if for any pair of points $\mathbf{w}, \mathbf{v} \in \mathbb{R}^d$, it holds that $\ell(\mathbf{w}) \le \ell(\mathbf{v}) + \langle\nabla\ell(\mathbf{v}), \mathbf{w}-\mathbf{v}\rangle + \frac{\beta}{2}\|\mathbf{w}-\mathbf{v}\|^2$.

We remark that β-smoothness implies β-Lipschitz gradients (see e.g., Zhou (2018)). Having introduced these notions, we are ready to state the assumptions that are used later to analyze the convergence behavior of the proposed methods in this work.

Assumption 1. *(Bounded Infimum) The infimum is bounded below, i.e.,* $\inf_{\mathbf{w}\in\mathbb{R}^d} \mathfrak{L}(\mathbf{w}) > -\infty$.

Assumption 2. *(Smoothness) The loss functions $\ell(\mathbf{w};\mathbf{x},z)$ satisfy the β-smoothness, where z is either the ground truth y or the ML-label $f(\mathbf{x})$. Consequently, $\mathfrak{L}(\mathbf{w})$ is β-smooth.*

Assumption 3. *(Finite variance) The stochastic gradients $\nabla_{\mathbf{w}}\ell(\mathbf{w};\mathbf{x},y)$ have bounded variance, i.e.,* $\mathbb{E}\left[\|\nabla_{\mathbf{w}}\ell(\mathbf{w};\mathbf{x},y) - \nabla\mathfrak{L}(\mathbf{w})\|^2\right] \le \sigma^2$.

We note that lower-boundedness, smoothness, and finite variance are common conditions in the analysis of nonconvex optimization, e.g., (Ghadimi & Lan, 2013; Sfyraki & Wang, 2026a). Here, we impose smoothness on the individual loss function for the simplicity of the analysis and presentation in zeroth-order optimization later on. We further note that the same set of assumptions was also considered in an elegant work of Cutkosky & Orabona (2019) for stochastic optimization.

In the following, we denote $\lambda_t^* := \arg\min_{\lambda\in\mathbb{R}} \phi_t(\lambda)$, the minimizer of the instantaneous loss function at t in the scalar tuning game (Figure 1), $\Phi_{1:T}(\lambda) := \sum_{t=1}^T (\phi_t(\lambda) - \phi_t(0))$, the cumulative loss, and $\lambda^* := \arg\min_{\lambda\in\mathbb{R}} \mathbb{E}\left[\Phi_{1:T}(\lambda)\right]$.

Theorem 1. *Denote* $C_0 := \left(\mathbb{E}\left[\max_{1\le t\le T} |\lambda_t^*|^4\right]\right)^{1/4}$, $C_1 := \left(\mathbb{E}\left[\max_{1\le t\le T} \|\mathbf{d}_t\|^4\right]\right)^{1/4}$. *Then, under Assumptions 1,2, 3, Algorithm 1 with scalar tuning and $\eta_t = \frac{1}{\beta\sqrt{T}}$ satisfies*

$$\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8C_0^2C_1^2\,(1+\log T)}{T^{3/2}} + \frac{\mathbb{E}[\Phi_{1:T}(\lambda^*)]}{T^{3/2}} + \frac{\sigma^2}{\sqrt{T}}, \qquad (4)$$

where $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] = -\frac{\left(\sum_{t=1}^T \mathbb{E}[\langle\mathbf{g}_t,\mathbf{d}_t\rangle]\right)^2}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}$, if $\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] > 0$, *and* $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] = 0$ *otherwise. In particular, it holds that* $-T\sigma^2 \le \mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] \le 0$.

We remark that, compared to the standard SGD without using ML-labeled data (Ghadimi & Lan, 2013), there are two additional middle terms in the upper bound in (4). The first middle term arises from the regret incurred by the online FTL tuning procedure and is of order $O\left(\frac{\log T}{T^{3/2}}\right)$, and hence is negligible relative to the $O\left(\frac{1}{\sqrt{T}}\right)$ terms when T is sufficiently large. Under the reasonable condition that $\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]$ is nonzero for at least one $t \in [T]$, the last two terms in (4) can be combined as $\frac{\mathbb{E}[\Phi_{1:T}(\lambda^*)]}{T^{3/2}} + \frac{\sigma^2}{\sqrt{T}} = \frac{1}{\sqrt{T}}\left(\sigma^2 - \frac{\left(\frac{1}{T}\sum_{t=1}^T \mathbb{E}[\langle\mathbf{g}_t,\mathbf{d}_t\rangle]\right)^2}{\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}\right)$, which is no greater than the standard $\frac{\sigma^2}{\sqrt{T}}$ term and demonstrates the benefit of exploiting ML-labeled data.

To further understand the potential improvement in the convergence rate from using ML-labeled data, we present the following corollary, where we consider constant batch sizes such that $|B_t| = n$ and $\left|\tilde{B}_t\right| = N$ for all $t \in [T]$. Below, we define $\mathfrak{L}^f(\mathbf{w}) := \mathbb{E}\left[\ell\left(\mathbf{w}; \mathbf{x}, f(\mathbf{x})\right)\right]$.

Corollary 1. *Suppose the variance of a gradient computed from a single ML-labeled sample is uniformly bounded by σ_f^2, i.e., $\mathbb{E}\left[\left\|\nabla\ell\left(\mathbf{w}; \mathbf{x}, f(\mathbf{x})\right) - \nabla\mathfrak{L}^f(\mathbf{w})\right\|^2\right] \leq \sigma_f^2$, while σ_y^2 represents its counterpart using true labels. Additionally, let $\bar{\Xi}_T := \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]$ and $\Sigma_t^{y,f} := \mathrm{Cov}\left(\nabla\ell\left(\mathbf{w}_t; \mathbf{x}, y\right), \nabla\ell\left(\mathbf{w}_t; \mathbf{x}, f(\mathbf{x})\right) \middle| \mathcal{F}_{t-1}\right)$ denote the conditional covariance between stochastic gradients computed using true and ML labels at step t. Consider the case that $\exists t \in [T] : \mathbb{E}[\|\mathbf{d}_t\|^2] > 0$. Under the assumptions of Theorem 1, Algorithm 1 satisfies*

$$\bar{\Xi}_T \leq \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8C_0^2C_1^2\left(1+\log T\right)}{T^{3/2}} + \frac{1}{\sqrt{T}}\frac{\sigma_y^2}{n} - \frac{1}{\sqrt{T}}\frac{N}{n(n+N)}\frac{\left(\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\mathrm{tr}\left(\Sigma_t^{y,f}\right)\right]\right)^2}{\sigma_f^2}.$$

Corollary 1 suggests that the variance reduction is larger when the deviations of the true-label and ML-label gradients from their respective population gradients are more strongly correlated in expectation, relative to the standard deviation of a stochastic gradient computed from an ML-labeled sample. Intuitively, the variance reduction and hence the speedup are more significant when stochastic gradients computed from ML-labeled data are more "predictive" of those computed from data with true labels. We further note that the unlabeled batch size N affects the variance-reduction term through the factor $\frac{N}{n(n+N)}$ from the corollary, which suggests that while the overall improvement continues to increase with the number of ML-labeled samples N, the additional improvement obtained by increasing N diminishes as N grows, with n fixed.

Due to space limitations, we provide the analysis of SGD with vector tuning in Theorem 3 of Appendix B. As in the scalar tuning case, the bound contains a term that reflects the variance reduction enabled by incorporating ML-labeled data. We refer the reader to Appendix B for details.

3 Zeroth-Order Stochastic Optimization with ML-labeled Data

3.1 Our Methods

The machinery we developed in the previous section can be naturally extended to zeroth-order stochastic optimization. We propose Algorithm 2 with the two online tuning schemes for leveraging ML-labeled data effectively. We emphasize that only one random Gaussian perturbation needs to be sampled at each step and that it is shared across the different components used to construct the zeroth-order prediction-powered gradient estimate. For further notation brevity, we will denote $\mathbf{q}_t := q^y_{t,\mathbf{w}_t}$ and $\mathbf{d}_t^0 := \tilde{q}^f_{t,\mathbf{w}_t} - q^f_{t,\mathbf{w}_t}$ below.

Our corresponding tuning schemes in zeroth-order optimization are:

$$\textbf{(Scalar tuning)} \qquad \lambda_t = -\frac{\sum_{s=1}^{t-1}\langle \mathbf{q}_s, \mathbf{d}_s^0\rangle}{\sum_{s=1}^{t-1}\|\mathbf{d}_s^0\|^2}, \quad \text{if } \sum_{s=1}^{t-1}\left\|\mathbf{d}_s^0\right\|^2 > 0, \tag{5}$$

where we let $\lambda_t = 0$ when $\sum_{s=1}^{t-1}\left\|\mathbf{d}_s^0\right\|^2 = 0$ such as the initial step. The other scheme uses the full vector tuning parameter $\boldsymbol{\lambda}_t$; specifically, for each $i \in [d]$, we update

$$\textbf{(Vector tuning)} \quad \boldsymbol{\lambda}_t[i] = -\frac{\sum_{s=1}^{t-1}\mathbf{q}_s[i]\mathbf{d}_s^0[i]}{\sum_{s=1}^{t-1}(\mathbf{d}_s^0[i])^2}, \quad \text{if } \sum_{s=1}^{t-1}(\mathbf{d}_s^0[i])^2 > 0, \tag{6}$$

and we let $\boldsymbol{\lambda}_t[i] = 0$, otherwise. These updates are derived using the same principle of online tuning as a game as illustrated in the previous section, except that one observes a sequence of loss functions based on zeroth-order gradient estimates here. We refer the reader to Appendix C for details.

3.2 Theoretical analysis

In the following, we denote $\lambda_t^* := \arg\min_{\lambda\in\mathbb{R}}\psi_t(\lambda)$, where $\psi_t(\lambda) := \left\|\mathcal{Q}^{\lambda}_{t,\mathbf{w}_t}\right\|^2 = \left\|\mathbf{q}_t + \lambda\mathbf{d}_t^0\right\|^2$ is the instantaneous loss function at t in the corresponding scalar tuning game, $\Psi_{1:T}(\lambda) := \sum_{t=1}^{T}\left(\psi_t(\lambda) - \psi_t(0)\right)$, the cumulative loss, and $\lambda^* := \arg\min_{\lambda\in\mathbb{R}}\mathbb{E}\left[\Psi_{1:T}(\lambda)\right]$.

Algorithm 2 Stochastic Gradient-Free Method with ML-labeled data

Require: Initial point $\mathbf{w}_1 \in \mathbb{R}^d$, smoothing parameter μ, and step sizes $\{\eta_t\}_{t=1}^T$.

1: **for** $t = 1, \ldots, T$ **do**

2: Sample $\zeta_t \sim \mathcal{N}(0, I_d)$.

3: Compute $q^y_{t,\mathbf{w}_t} := \frac{1}{|B_t|}\sum_{(\mathbf{x},y)\in B_t} \frac{\ell(\mathbf{w}_t+\mu\zeta_t;\mathbf{x},y)-\ell(\mathbf{w}_t;\mathbf{x},y)}{\mu}\zeta_t$ by sampling a mini-batch B_t of points with true labels and also compute $q^f_{t,\mathbf{w}_t} := \frac{1}{|B_t|}\sum_{(\mathbf{x},y)\in B_t} \frac{\ell(\mathbf{w}_t+\mu\zeta_t;\mathbf{x},f(\mathbf{x}))-\ell(\mathbf{w}_t;\mathbf{x},f(\mathbf{x}))}{\mu}\zeta_t$.

4: Compute $\tilde{q}^f_{t,\mathbf{w}_t} := \frac{1}{|\tilde{B}_t|}\sum_{\tilde{\mathbf{x}}\in \tilde{B}_t} \frac{\ell(\mathbf{w}_t+\mu\zeta_t;\tilde{\mathbf{x}},f(\tilde{\mathbf{x}}))-\ell(\mathbf{w}_t;\tilde{\mathbf{x}},f(\tilde{\mathbf{x}}))}{\mu}\zeta_t$ from a batch $\tilde{B}_t$ of ML-labeled points.

5: Obtain $\boldsymbol{\lambda}_t$ using either the scalar tuning scheme (5) or the vector tuning scheme (6).

6: Update $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t\, \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$, where $\mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} := q^y_{t,\mathbf{w}_t} + \boldsymbol{\lambda}_t \odot \left(\tilde{q}^f_{t,\mathbf{w}_t} - q^f_{t,\mathbf{w}_t}\right)$.

7: **end for**

Theorem 2. *Denote* $D_0 := \left(\mathbb{E}\left[\max_{1\le t\le T} |\lambda^*_t|^4\right]\right)^{1/4}$, $D_1 := \left(\mathbb{E}\left[\max_{1\le t\le T} \left\|\mathbf{d}^0_t\right\|^4\right]\right)^{1/4}$, *and* $\bar{\Xi}_T := \frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\left\|\nabla \mathfrak{L}(\mathbf{w}_t)\right\|^2\right]$. *Then, under Assumptions 1, 2, 3, Algorithm 2 with scalar tuning,* $\eta = \min\left\{\frac{1}{4\beta(d+2)}, \frac{1}{4\beta\sqrt{(d+2)T}}\right\}$, *and* $\mu = \frac{1}{\beta(d+3)\sqrt{T}}$ *satisfies*

$$\bar{\Xi}_T \le 16\beta\Delta_1\left(\tfrac{d+2}{T} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\right) + \tfrac{16d}{(d+3)^2}\left(\tfrac{d+2}{T^2} + \tfrac{\sqrt{d+2}}{T^{3/2}}\right) + \tfrac{d+3}{T} + \tfrac{4D_0^2 D_1^2(1+\log T)}{\sqrt{d+2}\,T^{3/2}} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \tfrac{d(d+4)}{4(d+3)^2T} + \tfrac{\mathbb{E}[\Psi_{1:T}(\lambda^*)]}{2(d+2)T}\right), \tag{7}$$

where $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = -\frac{\left(\sum_{t=1}^T \mathbb{E}\left[\left\langle \mathbf{q}_t, \mathbf{d}^0_t\right\rangle\right]\right)^2}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}^0_t\|^2\right]}$ *if* $\sum_{t=1}^T \mathbb{E}\left[\left\|\mathbf{d}^0_t\right\|^2\right] > 0$, *and* $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = 0$ *otherwise. In particular, it holds that* $-2(d+2)T\sigma^2 - \frac{d(d+2)(d+4)}{2(d+3)^2} \le \mathbb{E}[\Psi_{1:T}(\lambda^*)] \le 0$.

One of the technical results used to prove Theorem 2 is Lemma 10 in Appendix C, which relates the expected squared norm of the zeroth-order gradient estimator $\mathbf{q}_t$ to that of the corresponding first-order stochastic gradient $\mathbf{g}_t$. This connection allows us to express the zeroth-order convergence bound in terms of the variance σ^2 of stochastic gradients and an additional finite-difference remainder term. This analysis strategy is inspired from that adopted in the seminal work of Ghadimi & Lan (2013) on standard stochastic gradient-free methods without ML-labeled data; see the proof of Theorem 3.2 therein. Compared to their result, we obtain two additional terms in the upper bound in (7). The first is the leading term in the second row in (7), which corresponds to the regret incurred by the online FTL tuning procedure. As in the first-order result, this term is $O\left(\frac{\log T}{T^{3/2}}\right)$, and therefore negligible for sufficiently large T. The second additional contribution is $\frac{\mathbb{E}[\Psi_{1:T}(\lambda^*)]}{2(d+2)T}$. Under the condition that $\mathbb{E}\left[\|\mathbf{d}^0_t\|^2\right]$ is nonzero for at least one $t \in [T]$, the terms in the final parenthesis of (7) can be written as $\sigma^2 + \frac{d(d+4)}{4(d+3)^2T} + \frac{\mathbb{E}[\Psi_{1:T}(\lambda^*)]}{2(d+2)T} = \sigma^2 + \frac{d(d+4)}{4(d+3)^2T} - \frac{1}{2(d+2)}\frac{\left(\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\left\langle \mathbf{q}_t, \mathbf{d}^0_t\right\rangle\right]\right)^2}{\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}^0_t\|^2\right]}$. Since the final term is nonnegative and is subtracted, this quantity is no greater than $\sigma^2 + \frac{d(d+4)}{4(d+3)^2T}$, which is the corresponding contribution from stochastic-gradient variance and the finite-difference remainder in the standard zeroth-order analysis. Hence, our results suggest that a similar benefit also arises in zeroth-order optimization.

We defer the analysis of Algorithm 2 with vector tuning to Appendix C.

4 Experiments

In this section, we evaluate the proposed first- and zeroth-order methods through a series of experiments on image- and text-classification tasks. Due to space limitations, we present a condensed summary here and provide detailed experimental results in Appendix D, where we also include additional experiments with linear models on regression and multiclass classification tasks in Appendix D.3 and Appendix D.4 respectively.

4.1 IMAGE CLASSIFICATION

Dataset and setup. We evaluate our first-order methods on CIFAR-10 (Krizhevsky, 2009) by training a linear classifier on top of a frozen ImageNet-pretrained ResNet-50 (He et al., 2016; Deng et al., 2009). Specifically, we remove the final classification layer of ResNet-50 and use its $2{,}048$-dimensional penultimate-layer representations as fixed input features for the linear classifier. We partition the original CIFAR-10 training set into $5{,}000$ examples for training the ML-labeler $f(\cdot)$, $4{,}500$ labeled examples for training the classifier, $5{,}000$ validation examples, and $35{,}500$ examples that will be served as unlabeled samples and will be labeled by the ML-labeler, with the official $10{,}000$-example test split used for evaluation. The ML-labeler is a separate linear classifier trained on the disjoint training set, obtaining a test accuracy of 86.47%. The ML-labeler is then kept fixed and used to generate the ML labels for both the labeled and unlabeled examples.

We evaluate four training methods for 100 epochs: standard SGD using only true-labeled examples, PP-SSL with Online Tuning (Shoham et al., 2025), and Algorithm 1 with scalar and vector tuning via FTL. For brevity, we refer to the latter two methods as Scalar FTL and Vector FTL when presenting the experimental results below. We consider several (n, N) configurations, with n and N denoting the labeled and unlabeled batch sizes, respectively. We use a constant learning rate throughout training, which we tune independently for each method and batch-size configuration.

Results. Figure 2 presents test-accuracy over training epochs for three batch-size configurations. Results for other batch-size configurations are available in Appendix D.2.1. We find that across all configurations that we try, Scalar FTL provides the most consistent improvement, achieving the highest mean test accuracy in all configurations. PP-SSL also generally improves over the labeled-only SGD baseline, although by a smaller margin, compared to our method (see also Table 3 in Appendix D.2.1). We observe that the behavior of Vector FTL is less consistent across batch-size configurations: it performs comparably to Scalar FTL for some choices of (n, N), but exhibits a noticeable degradation for others (Table 3). The tuning trajectories help explain these differences (Figure 9 in Appendix D.2.1). The Scalar FTL tuning parameter λ decreases smoothly throughout training, while the PP-SSL tuning parameter rapidly approaches zero, thereby substantially reducing the contribution of the ML-labeled correction after the early stages of training. We also find that although the mean of the Vector FTL parameters, i.e., $\mathrm{mean}(\boldsymbol{\lambda}_t)$, follows a similarly smooth decreasing trend as in scalar tuning, its coordinatewise parameters exhibit substantial dispersion, with some coordinates receiving large positive weights and others negative weights throughout training. Additional results for the zeroth-order methods for CIFAR-10 are provided in Appendix D.2.2.

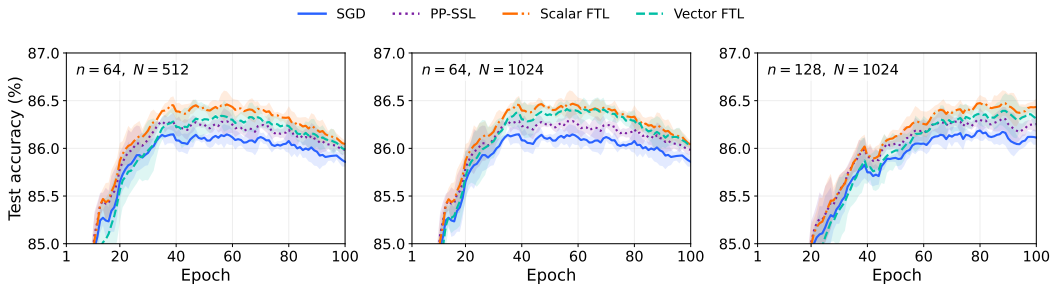


Figure 2: Test accuracy (%) over training epochs for the first-order methods on CIFAR-10 with $n = 64, N = 512$ (left), $n = 64, N = 1024$ (middle), and $n = 128, N = 1024$ (right). Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-accuracy curves are smoothed using a five-epoch moving average.

4.2 TEXT CLASSIFICATION

Dataset and setup. We evaluate our zeroth-order methods on text classification using RoBERTa-large (355M parameters) (Liu et al., 2019b). Following Malladi et al. (2023); Zhang et al. (2025a), we consider few-shot fine-tuning with $K = 32$ labeled examples per class on SST-2 (Socher et al., 2013), a binary sentiment classification task, and SST-5 (Socher et al., 2013), a five-class fine-grained sentiment classification task. This gives 64 labeled training examples for SST-2 and 160 for SST-5, with equally sized validation sets. For each task, we fine-tune a separate RoBERTa-large

ML-labeler $f(\cdot)$ on 128 labeled examples per class, obtaining test accuracies of 92.78% on SST-2 and 58.30% on SST-5. The ML-labeler are then kept fixed and used to generate ML-labels. The remaining training examples are treated as unlabeled data. We fine-tune the underlying model for $10{,}000$ optimization steps and compare three methods: standard stochastic gradient-free method without using ML-labeled data and our proposed zeroth-order methods (Algorithm 2) with scalar and vector tuning, which we denote as "ZO Scalar FTL" and "ZO Vector FTL" respectively in the following. We tune a constant learning rate for each method, and we use a labeled batch size of $n = 32$ and an unlabeled batch size of $N = 64$ for our methods.

Results. Figure 3 illustrates test-accuracy over training steps for Algorithm 2 with scalar and vector tuning on SST-2 and SST-5. On both SST-2 and SST-5, the proposed methods consistently improve over the ZO baseline after the early stages of training, as the tuning parameters become more stable after aggregating information over more optimization steps. In particular, we find that Algorithm 2 with scalar tuning performs similarly to its vector-tuning counterpart throughout most of training. The corresponding trajectories of the scalar tuning parameter and the mean of the vector tuning parameters are smooth and closely aligned. At the same time, the coordinatewise vector tuning parameters $\boldsymbol{\lambda}_t$ span a wide range, indicating that different coordinates continue to place different weights on the ML-labeled correction even when the mean of the vector tuning parameters evolves similarly to the scalar one (Figure 7 in Appendix D.1.2). Additional results on fine-tuning with the first-order methods are provided in Appendix D.1.1.

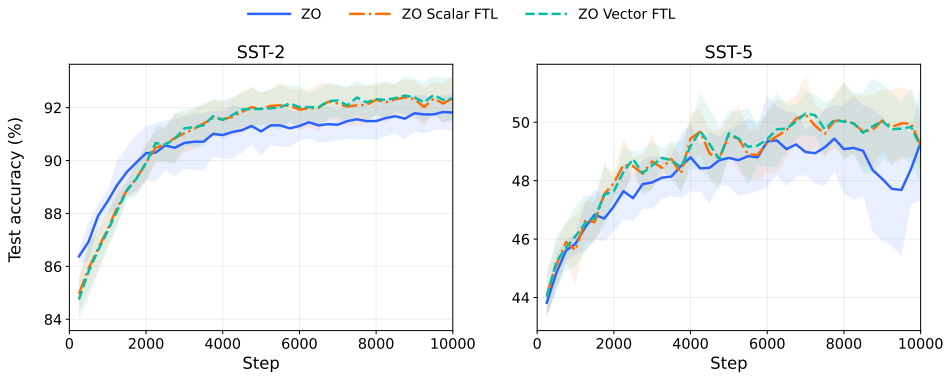


Figure 3: Test accuracy (%) over training steps for the zeroth-order methods on SST-2 (left) and SST-5 (right). Test accuracy is evaluated every 250 optimization steps. Solid curves denote the mean across five random seeds, and shaded regions indicate ± 1 standard deviation.

5 Discussion and Future Directions

In this work, we propose stochastic first- and zeroth-order optimization methods that use true-labeled and ML-labeled data to accelerate optimization. We consider both online scalar and vector tuning schemes that control the contribution of the ML-labeled gradient correction while preserving the convergence rate of standard stochastic optimization and enabling variance reduction. Our results reveal a trade-off between the proposed tuning schemes. Although both may achieve similar convergence rates, vector tuning offers greater coordinatewise flexibility but exhibits greater sensitivity to batch sizes in our experiments, whereas scalar tuning tends to be more stable across configurations. A potential direction for future work is therefore to identify classes of objective functions and data conditions under which vector tuning provably outperforms scalar tuning, as well as settings in which its additional computational overhead outweighs the gains. Another related direction is to investigate intermediate levels of tuning granularity, such as layer-wise or tensor-wise tuning. Such schemes could better account for heterogeneity in gradient estimates across different blocks of model parameters while retaining some of the stability benefits of sharing a tuning parameter across multiple coordinates, which we plan to explore in future work. Finally, our work assumes that ML labels are produced by a fixed ML model and accommodates settings in which only its labels, rather than the ML-labeler itself, are accessible. An interesting direction is to consider settings in which the ML-labeler is accessible and can also be updated on the fly.

AI USE STATEMENT

In this work, we used generative AI tools to automate routine computations and assist with proof checking. We have not used generative AI tools to develop the theoretical results, formulate mathematical claims, generate critical ingredients in the proof ideas, propose or refine hypotheses, design the research methodology or experiments, assist with translation, clean and reformat the datasets, or interpret the results. Qualitative or thematic data analysis and the generation of synthetic datasets are not applicable to this work. Additionally, we used generative AI tools to assist with polishing the paper for readability, implementing experimental code, and creating scientific figures. We have reviewed all AI-assisted work. In particular, all theoretical arguments and proof steps were written, and verified by the authors, and all AI-assisted code was reviewed and validated for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Alon Albalak, Colin Raffel, and William Yang Wang. Improving few-shot generalization by exploring and exploiting auxiliary data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Anastasios N Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 382(6671):669–674, 2023a.

Anastasios N Angelopoulos, John C Duchi, and Tijana Zrnic. PPI++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*, 2023b.

Lei Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems*, 2014.

Krishnakumar Balasubramanian and Saeed Ghadimi. Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points. *Foundations of Computational Mathematics*, 22(1):35–76, 2022.

Ejafa Bassam, Dawei Zhu, and Kaigui Bian. PLD: A choice-theoretic list-wise knowledge distillation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

David Berthelot, Nicholas Carlini, Ekin Dogus Cubuk, Alexey Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations (ICLR)*, 2020.

Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *International Conference on Machine Learning (ICML)*, 2009.

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007.

Cristian Bucilă, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

Sungmin Cha and Kyunghyun Cho. Why knowledge distillation works in generative models: A minimal working explanation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011.

Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien (eds.). *Semi-Supervised Learning*. The MIT Press, 2006.

Baixu Chen, Junguang Jiang, Ximei Wang, Pengfei Wan, Jianmin Wang, and Mingsheng Long. Debiased self-training for semi-supervised learning. In *Advances in Neural Information Processing Systems*, 2022.

Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems (NeurIPS) 32*, 2019.

Rudrajit Das and Sujay Sanghavi. Understanding self-distillation in the presence of label noise. In *International Conference on Machine Learning (ICML)*, 2023.

Philip de Rijk, Lukas Schneider, Marius Cordts, and Dariu Gavrila. Structural knowledge distillation for object detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Yury Demidovich, Grigory Malinovsky, Igor Sokolov, and Peter Richtárik. A guide through the zoo of biased SGD. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Xiang Deng and Zhongfei Zhang. Comprehensive knowledge distillation with causal intervention. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. In *International Conference on Machine Learning (ICML)*, 2019.

Elad Hazan. *Introduction to online convex optimization*. MIT Press, 2nd edition, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015.

Kun Huang, Xin Guo, and Meng Wang. Towards efficient pre-trained language model via feature correlation distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Ayush Jain, Andrea Montanari, and Eren Sasoglu. Scaling laws for learning with real and surrogate data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Prateek Jain, Sham M Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Accelerating stochastic gradient descent for least squares regression. In *Conference on Learning Theory (COLT)*, 2018.

Svante Janson. *Gaussian Hilbert Spaces*. Cambridge Tracts in Mathematics. Cambridge University Press, 1997.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Nicolas Le Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems (NeurIPS) 25*, 2012.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, 2013.

Kfir Y. Levy. Online to offline conversions, universality and adaptive minibatch sizes. In *Advances in Neural Information Processing Systems (NeurIPS) 30*, 2017.

Muyang Li, Runze Wu, Haoyu Liu, Jun Yu, Xun Yang, Bo Han, and Tongliang Liu. Instant: Semi-supervised learning with instance-dependent thresholds. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Zhize Li, Hongyan Bao, Xiangliang Zhang, and Peter Richtárik. PAGE: A simple and optimal probabilistic gradient estimator for nonconvex optimization. In *International Conference on Machine Learning (ICML)*, 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Shikun Liu, Andrew J. Davison, and Edward Johns. Self-supervised generalisation with meta auxiliary learning. In *Advances in Neural Information Processing Systems*, 2019a.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692*, 2019b.

Jiaming Lv, Haoyuan Yang, and Peihua Li. Wasserstein distance rivals Kullback-Leibler divergence for knowledge distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

David Mayo, Jesse Cummings, Xinyu Lin, Dan Gutfreund, Boris Katz, and Andrei Barbu. How hard are computer vision datasets? Calibrating dataset difficulty to viewing time. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Vaishnavh Nagarajan, Aditya Krishna Menon, Srinadh Bhojanapalli, Hossein Mobahi, and Sanjiv Kumar. On student-teacher deviations in distillation: does it pay to disobey? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Deanna Needell, Nathan Srebro, and Rachel Ward. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Lam Nguyen, Phuong Ha Nguyen, Marten van Dijk, Peter Richtárik, Katya Scheinberg, and Martin Takác. SGD and Hogwild! convergence without the bounded gradients assumption. In *International Conference on Machine Learning (ICML)*, 2018.

Youngtaek Oh, Dong-Jin Kim, and In So Kweon. DASO: Distribution-aware semantics-oriented pseudo-label for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Utkarsh Ojha, Yuheng Li, Anirudh Sundara Rajan, Yingyu Liang, and Yong Jae Lee. What knowledge gets distilled in knowledge distillation? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Francesco Orabona. Online learning: A modern introduction using convex optimization. *arXiv:1912.13213v10*, 2026.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *International Conference on Machine Learning (ICML)*, 2012.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

Mher Safaryan, Alexandra Peste, and Dan Alistarh. Knowledge distillation performs partial variance reduction. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Maria-Eleni Sfyraki and Jun-Kun Wang. Lions and Muons: Optimization via Stochastic Frank-Wolfe under Heavy-Tailed Noise. In *International Conference on Machine Learning (ICML)*, 2026a.

Maria-Eleni Sfyraki and Jun-Kun Wang. Revisiting active sequential prediction-powered mean estimation. In *International Conference on Learning Representations (ICLR)*, 2026b.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2011.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning (ICML)*, 2013.

Baifeng Shi, Judy Hoffman, Kate Saenko, Trevor Darrell, and Huijuan Xu. Auxiliary task reweighting for minimum-data learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Noa Shoham, Ron Dorfman, Shalev Shaer, Kfir Y. Levy, and Yaniv Romano. Prediction-powered semi-supervised learning with online power tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Oct. 2013.

Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:596–608, 2020.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi, and Andrew Gordon Wilson. Does knowledge distillation really work? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Jesper E. van Engelen and Holger H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019.

Jun-Kun Wang, Jacob Abernethy, and Kfir Y Levy. No-regret dynamics in the Fenchel game: A unified framework for algorithmic convex optimization. *Mathematical Programming*, 205:203–268, 2024.

Xudong Wang, Zhirong Wu, Long Lian, and Stella X. Yu. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. Freematch: Self-adaptive thresholding for semi-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. CReST: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Blake Woodworth, Konstantin Mishchenko, and Francis Bach. Two losses are better than one: Faster optimization using a cheaper proxy. In *International Conference on Machine Learning (ICML)*, 2023.

Stephen J. Wright and Benjamin Recht. *Optimization for Data Analysis*. Cambridge University Press, 2022.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *International Conference on Machine Learning (ICML)*, 2021.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:18408–18419, 2021.

Liang Zhang, Bingcong Li, Kiran Koshy Thekumparampil, Sewoong Oh, Michael Muehlebach, and Niao He. Zeroth-order optimization finds flat minima. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025a.

Ping Zhang, Zheda Mai, Quang-Huy Nguyen, and Wei-Lun Chao. Revisiting semi-supervised learning in the era of foundation models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b.

Tong Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *International Conference on Machine Learning (ICML)*, 2004.

Xingyu Zhou. On the Fenchel duality between strong convexity and Lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, University of Wisconsin-Madison Department of Computer Sciences, 2005.

A More Related Works

Semi-Supervised Learning. Our setting is closely related to Semi-Supervised Learning (SSL), which extends classical supervised learning by leveraging abundant unlabeled data alongside a limited amount of labeled data, thus offering a practical approach to improving learning performance while reducing the reliance on costly data annotation (Zhu, 2005; Chapelle et al., 2006; van Engelen & Hoos, 2020). Modern SSL methods include self-training approaches (Xie et al., 2020; Wei et al., 2021; Chen et al., 2022), which iteratively use model predictions on unlabeled data to construct additional supervision and retrain the model, and more broadly pseudo-labeling techniques (Lee, 2013; Oh et al., 2022; Wang et al., 2022; Zhang et al., 2025b), which use model-generated predictions as supervisory targets for unlabeled examples. Some popular approaches combine label guessing or pseudo-labeling with consistency regularization and data augmentation, including MixMatch (Berthelot et al., 2019) and ReMixMatch (Berthelot et al., 2020). A more recent line of work on pseudo-labeling methods controls the contribution of unlabeled examples based on model confidence. In particular, FixMatch (Sohn et al., 2020) uses a fixed confidence threshold, while subsequent methods such as FlexMatch (Zhang et al., 2021) and FreeMatch (Wang et al., 2023) adapt the threshold during training, and SoftMatch (Chen et al., 2023) replaces hard thresholding with confidence-based soft weighting. Other works employ alternative selection criteria: Dash (Xu et al., 2021) dynamically adjusts a loss-based selection threshold, instead of a confidence threshold, while InstanT (Li et al., 2023) introduces instance-dependent thresholds for pseudo-label selection. In contrast to the SSL approaches, our work utilizes ML-labels not as direct supervisory targets in the training objective, but as auxiliary information for constructing a stochastic gradient estimator whose contribution is tuned online to reduce variance.

Knowledge Distillation. Our work is also connected to Knowledge Distillation (KD), which focuses on transferring knowledge from a teacher model to a student model, typically by matching the teacher's output logits (Bucilă et al., 2006; Ba & Caruana, 2014) or softened predictive distribution (Hinton et al., 2015). Unlike SSL, whose goal is to utilize unlabeled data to improve learning, KD is primarily concerned with teacher-to-student knowledge transfer and does not inherently require unlabeled data. Subsequent work has extended KD beyond output-logit matching by transferring intermediate feature representations (Tian et al., 2020; Deng & Zhang, 2021; Huang et al., 2023), and relational or structural information (Tian et al., 2020; de Rijk et al., 2022). Other works explore alternative distillation objectives based on contrastive, correlation-based, Wasserstein, and ranking losses (Huang et al., 2022; Lv et al., 2024; Bassam et al., 2025). Beyond the design of distillation objectives and transferred representations, recent work has examined why and when distillation is effective, including optimization difficulties that can prevent students from matching the teacher's predictive distribution (Stanton et al., 2021), bias-variance tradeoffs arising in self-distillation under label noise (Das & Sanghavi, 2023), the transfer of teacher-specific properties to the student (Ojha et al., 2023), and the potential benefits of allowing students to deviate from their teachers (Nagarajan et al., 2023). More recently, KD has also been studied in modern generative-model settings, where distillation can induce a precision-recall tradeoff in the student's learned distribution (Cha & Cho, 2025). Most closely related to our work, Safaryan et al. (2023) study KD through the lens of stochastic variance reduction, showing that it can partially reduce gradient noise in linear and deep linear models. Our setting differs from KD in that we do not optimize a teacher-matching objective, but rather the ML-labels are used to construct an unbiased gradient estimator for the original population objective.

Optimization with Surrogate Data. Another closely related line of research considers learning with surrogate data, where auxiliary data sources or proxy objectives are used to improve optimization or learning on a target task. Woodworth et al. (2023) consider optimization with an expensive target loss and a cheaper proxy loss, and develop an algorithm that exploits curvature information from the proxy loss to reduce the number of stochastic gradient evaluations of the target objective. Jain et al. (2024) study learning from mixtures of real and surrogate data and characterize how surrogate samples should be weighted to improve performance on the target distribution. Related work on auxiliary learning similarly investigates how additional tasks or datasets can be constructed, selected, or weighted to improve learning on a primary task (Shi et al., 2020; Albalak et al., 2023; Liu et al., 2019a).

B Proofs of the Theoretical Results in Section 2

Before proving the theorems, we need a series of lemmas.

We denote $\delta_t := \mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} - \nabla\mathfrak{L}(\mathbf{w})$ in the following and use $\mathbb{E}_{t-1}[\cdot] := \mathbb{E}[\cdot \mid \mathcal{F}_{t-1}]$ for brevity.

Lemma 1. *(Adapted from (2.9) in Ghadimi & Lan (2013)) Consider $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$. Under Assumption 1 and Assumption 2, it holds that*

$$\sum_{t=1}^{T}\left(\eta_t - \frac{\beta}{2}\eta_t^2\right)\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2 \le \Delta_1 - \sum_{t=1}^{T}\left(\eta_t - \beta\eta_t^2\right)\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \frac{\beta}{2}\sum_{t=1}^{T}\eta_t^2\|\delta_t\|^2. \quad (8)$$

Proof. For reader's convenience, we replicate the analysis of Ghadimi & Lan (2013) below.

By β-smoothness,

$$\begin{aligned}
\mathfrak{L}(\mathbf{w}_{t+1}) &\le \mathfrak{L}(\mathbf{w}_t) + \langle\nabla\mathfrak{L}(\mathbf{w}_t), \mathbf{w}_{t+1} - \mathbf{w}_t\rangle + \frac{\beta}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 \\
&= \mathfrak{L}(\mathbf{w}_t) - \eta_t\langle\nabla\mathfrak{L}(\mathbf{w}_t), \nabla\mathfrak{L}(\mathbf{w}_t) + \delta_t\rangle + \frac{\beta}{2}\eta_t^2\|\nabla\mathfrak{L}(\mathbf{w}_t) + \delta_t\|^2 \\
&= \mathfrak{L}(\mathbf{w}_t) - \eta_t\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2 - \eta_t\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \frac{\beta}{2}\eta_t^2\left(\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2 + 2\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \|\delta_t\|^2\right) \\
&= \mathfrak{L}(\mathbf{w}_t) - \left(\eta_t - \frac{\beta}{2}\eta_t^2\right)\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2 - \left(\eta_t - \beta\eta_t^2\right)\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \frac{\beta}{2}\eta_t^2\|\delta_t\|^2.
\end{aligned}$$

Rearranging and summing over $t = 1, \ldots, T$, the loss differences telescope, which gives,

$$\begin{aligned}
\sum_{t=1}^{T}\left(\eta_t - \frac{\beta}{2}\eta_t^2\right)\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2 &\le \mathfrak{L}(\mathbf{w}_1) - \mathfrak{L}(\mathbf{w}_{T+1}) - \sum_{t=1}^{T}\left(\eta_t - \beta\eta_t^2\right)\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \frac{\beta}{2}\sum_{t=1}^{T}\eta_t^2\|\delta_t\|^2 \\
&\le \Delta_1 - \sum_{t=1}^{T}\left(\eta_t - \beta\eta_t^2\right)\langle\nabla\mathfrak{L}(\mathbf{w}_t), \delta_t\rangle + \frac{\beta}{2}\sum_{t=1}^{T}\eta_t^2\|\delta_t\|^2.
\end{aligned}$$

□

Lemma 2. *Assume that $\boldsymbol{\lambda}_t$ is $\mathcal{F}_{t-1}$-measurable and hence $\mathbb{E}_{t-1}[\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}] = \nabla\mathfrak{L}(\mathbf{w}_t)$. Then,*

$$\mathbb{E}_{t-1}\left[\|\delta_t\|^2\right] = \mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2.$$

Proof.

$$\begin{aligned}
\mathbb{E}_{t-1}\left[\|\delta_t\|^2\right] &= \mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \mathbb{E}_{t-1}\left[2\langle\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}, \nabla\mathfrak{L}(\mathbf{w}_t)\rangle\right] + \mathbb{E}_{t-1}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \quad (9) \\
&= \mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2, \quad (10)
\end{aligned}$$

where the last equality uses that $\mathbb{E}_{t-1}[\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}] = \nabla\mathfrak{L}(\mathbf{w}_t)$.

□

Lemma 3. *Under Assumption 1 and Assumption 2. Suppose that $\boldsymbol{\lambda}_t$ is $\mathcal{F}_{t-1}$-measurable and hence $\mathbb{E}_{t-1}[\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}] = \nabla L(\mathbf{w}_t)$. Then,*

$$\sum_{t=1}^{T}\left(\eta_t - \frac{\beta}{2}\eta_t^2\right)\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T}\eta_t^2\,\mathbb{E}\left[\mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right].$$

Proof. Taking expectation on both sides of (8), noting that $\mathbb{E}[\langle \nabla \mathfrak{L}(\mathbf{w}_t), \delta_t \rangle] = 0$, we have

$$\begin{aligned}\sum_{t=1}^{T} \left(\eta_t - \frac{\beta}{2}\eta_t^2\right) \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] &\leq \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\|\delta_t\|^2\right] \\ &= \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\|\delta_t\|^2\right]\right] \\ &= \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right], \quad (11)\end{aligned}$$

where (11) uses Lemma 2. □

Now we recall the notion of vector tuning as a game introduced in Subsection 2.1, where the sequence of loss functions are $\phi_t(\boldsymbol{\lambda}) := \left\|\mathcal{G}^{\boldsymbol{\lambda}}_{t,\mathbf{w}_t}\right\|^2 = \|\mathbf{g}_t + \boldsymbol{\lambda} \odot \mathbf{d}_t\|^2$. For this, we define

$$\mathrm{Regret}^{\phi}_T(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T} \phi_t(\boldsymbol{\lambda}_t) - \sum_{t=1}^{T} \phi_t(\boldsymbol{\lambda}^*),$$

where $\boldsymbol{\lambda}_t \in \mathbb{R}^d$ is the action at t committed by the online learner and $\boldsymbol{\lambda}^* \in \mathbb{R}^d$ is a comparator.

We also recall that the case of vector tuning actually covers the case of scalar tuning as a special case. Specifically, given a scalar tuning parameter $\lambda \in \mathbb{R}$, a scalar-vector multiplication $\lambda \mathbf{v}$ can be rewritten as $\lambda \mathbf{v} = \lambda \mathbf{1} \odot \mathbf{v}$, where $\mathbf{1} \in \mathbb{R}^d$ is the all-ones vector. This allows us to invoke Lemma 3 and Lemma 4 below by letting $\boldsymbol{\lambda}_t \leftarrow \lambda_t \mathbf{1}$ and $\boldsymbol{\lambda}^* \leftarrow \lambda^* \mathbf{1}$ in the lemmas when analyzing the case of scalar tuning.

Lemma 4. *Denote* $\Phi_{1:T}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T} \left\|\mathcal{G}^{\boldsymbol{\lambda}^*}_{t,\mathbf{w}_t}\right\|^2 - \sum_{t=1}^{T} \left\|\mathcal{G}^{0}_{t,\mathbf{w}_t}\right\|^2$. *Set* $\eta_t = \eta$, *under Assumption 1, 2, and 3, we have*

$$\sum_{t=1}^{T} \left(\eta - \frac{\beta}{2}\eta^2\right) \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \Delta_1 + \frac{\beta\eta^2}{2}\,\mathbb{E}\left[\mathrm{Regret}^{\phi}_T(\boldsymbol{\lambda}^*)\right] + \frac{\beta\eta^2}{2}\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)] + \frac{\beta\eta^2 T}{2}\sigma^2. \quad (12)$$

Proof. From Lemma 3,

$$\begin{aligned}\sum_{t=1}^{T} \left(\eta_t - \frac{\beta}{2}\eta_t^2\right)\mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] &\leq \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\left\|\mathcal{G}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right] - \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \quad (13) \\ &= \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\phi_t(\boldsymbol{\lambda}_t) - \phi_t(\boldsymbol{\lambda}^*) + \phi_t(\boldsymbol{\lambda}^*) - \phi_t(0) + \phi_t(0)\right] - \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \quad (14) \\ &\leq \Delta_1 + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2 \, \mathbb{E}\left[\mathbb{E}_{t-1}\left[\phi_t(\boldsymbol{\lambda}_t) - \phi_t(\boldsymbol{\lambda}^*) + \phi_t(\boldsymbol{\lambda}^*) - \phi_t(0)\right]\right] + \frac{\beta}{2}\sum_{t=1}^{T} \eta_t^2\sigma^2, \quad (15)\end{aligned}$$

where the last inequality is due to Assumption 3, i.e., $\mathbb{E}\left[\left\|\mathbf{g}_t - \nabla \mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq \sigma^2$.

Now set $\eta_t = \eta$. We further obtain

$$\sum_{t=1}^{T} \left(\eta - \frac{\beta}{2}\eta^2\right) \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \Delta_1 + \frac{\beta}{2}\eta^2\,\mathbb{E}\left[\mathrm{Regert}^{\phi}_T(\boldsymbol{\lambda}^*)\right] + \frac{\beta}{2}\eta^2\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)] + \frac{\beta T}{2}\eta^2\sigma^2. \quad (16)$$

□

As described in the main text, we propose using the Follow-the-Leader (FTL) strategy in the online tuning as a game, i.e., initialize $\lambda_1 \leftarrow 0$, and for all $t > 1$, play

$$\lambda_t \leftarrow \arg\min_{\lambda \in \mathbb{R}} \sum_{s=1}^{t-1} \phi_s(\lambda), \text{ where } \phi_s(\lambda) := \|\mathbf{g}_s + \lambda \mathbf{d}_s\|^2 . \tag{17}$$

Lemma 5. *(Lemma 2.1 in Shalev-Shwartz (2011); Be-the-Leader lemma) Consider a learner that receives a sequence of loss functions $h_1(\cdot), h_2(\cdot), \ldots, h_T(\cdot)$ in an online learning setting. Then, FTL, i.e., $\theta_t \in \arg\min_{\theta \in \Theta} \sum_{s=1}^{t-1} h(\theta)$, satisfies*

$$\text{Regret}_T^h(\theta^*) := \sum_{t=1}^{T} h_t(\theta_t) - \sum_{t=1}^{T} \phi_t(\theta^*) \le \sum_{t=1}^{T} \left(h_t(\theta_t) - h_t(\theta_{t+1})\right),$$

where $\theta^* \in \Theta$ is a comparator in the same decision space Θ as FTL.

We observe that expanding the cumulative loss gives

$$\sum_{s=1}^{t-1} \phi_s(\lambda) = \sum_{s=1}^{t-1} \|\mathbf{g}_s + \lambda \mathbf{d}_s\|^2 = \sum_{s=1}^{t-1} \|\mathbf{g}_s\|^2 + 2\lambda \sum_{s=1}^{t-1} \langle \mathbf{g}_s, \mathbf{d}_s \rangle + \lambda^2 \sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 .$$

For $\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 > 0$, the cumulative loss function is quadratic in λ. By setting $\frac{\partial}{\partial \lambda} \sum_{s=1}^{t-1} \phi_s(\lambda) = 0$, we obtain $\lambda_t = -\frac{\sum_{s=1}^{t-1} \langle \mathbf{g}_s, \mathbf{d}_s \rangle}{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}$. For $\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 = 0$, we have $\mathbf{d}_s = 0$, for all $s < t$. Therefore, we also have $\sum_{s=1}^{t-1} \langle \mathbf{g}_s, \mathbf{d}_s \rangle = 0$, and the cumulative loss function does not depend on λ. In this case, we use the convention $\lambda_t = 0$. To summarize, FTL (17) gives

$$\lambda_t = \begin{cases} -\frac{\sum_{s=1}^{t-1} \langle \mathbf{g}_s, \mathbf{d}_s \rangle}{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}, & \sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 > 0, \\ 0, & \sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 = 0, \end{cases}$$

which is (2). Next, observe that the minimizer of the instantaneous loss function $\phi_t(\cdot)$ at t has a closed-form expression. Specifically, denote $\lambda_t^* := \arg\min_{\lambda \in \mathbb{R}} \phi_t(\lambda)$. We have

$$\lambda_t^* = \begin{cases} -\frac{\langle \mathbf{g}_t, \mathbf{d}_t \rangle}{\|\mathbf{d}_t\|^2}, & \|\mathbf{d}_t\|^2 > 0, \\ 0, & \|\mathbf{d}_t\|^2 = 0. \end{cases} \tag{18}$$

Lemma 6. *Let $q_1, \ldots, q_T$ be an arbitrary sequence of nonnegative numbers, and define*

$$Q := \max_{1 \le t \le T} q_t \quad , \quad r_t := \begin{cases} \frac{q_t^2}{\sum_{s=1}^{t} q_s}, & \sum_{s=1}^{t} q_s > 0, \\ 0, & \sum_{s=1}^{t} q_s = 0. \end{cases}$$

Then, for $Q > 0$,

$$\sum_{t=1}^{T} r_t \le Q\,(1 + \log T)\,.$$

For $Q = 0$, we have $\sum_{t=1}^{T} r_t = 0$.

Proof. If $Q = 0$, then $q_t = 0$, for all t, and the result is immediate. Suppose that $Q > 0$. Define the function

$$\Gamma(v) := \begin{cases} v, & 0 \le v \le Q, \\ Q\,(1 + \log\,(v/Q))\,, & v \ge Q. \end{cases}$$

The function $\Gamma(\cdot)$ is concave and continuously differentiable with derivative

$$\Gamma'(v) := \begin{cases} 1, & 0 \le v \le Q, \\ Q/v, & v \ge Q. \end{cases}$$

The derivative is nonincreasing. Therefore, for every $v \geq 0$ and every $q \in [0, Q]$ we have

$$\Gamma(v+q) - \Gamma(v) = \int_v^{v+q} \Gamma'(u)\, du \geq q\Gamma'(v+q).$$

If $0 < v + q \leq Q$, then

$$q\Gamma'(v+q) = q \geq \frac{q^2}{v+q}.$$

If $v + q > Q$, then

$$q\Gamma'(v+q) = \frac{qQ}{v+q} \geq \frac{q^2}{v+q}.$$

Consequently, in both cases we have

$$\Gamma(v+q) - \Gamma(v) \geq \frac{q^2}{v+q}.$$

Take $v = \sum_{s=1}^{t-1} q_s$ and $q = q_t$, so that $v + q = \sum_{s=1}^{t} q_s$. If $\sum_{s=1}^{t} q_s > 0$, the above inequalities yield

$$\frac{q_t^2}{\sum_{s=1}^{t} q_s} \leq \Gamma\left(\sum_{s=1}^{t} q_s\right) - \Gamma\left(\sum_{s=1}^{t-1} q_s\right).$$

If $\sum_{s=1}^{t} q_s = 0$, then since $\sum_{s=1}^{t} q_s = \sum_{s=1}^{t-1} q_s + q_t$, we also have $\sum_{s=1}^{t-1} q_s = 0$ and $q_t = 0$. Hence, by definition

$$0 = \Gamma\left(\sum_{s=1}^{t} q_s\right) - \Gamma\left(\sum_{s=1}^{t-1} q_s\right).$$

Thus, for every $t = 1, \ldots, T$ we have

$$r_t \leq \Gamma\left(\sum_{s=1}^{t} q_s\right) - \Gamma\left(\sum_{s=1}^{t-1} q_s\right).$$

Summing over t we obtain

$$\sum_{t=1}^{T} r_t \leq \Gamma\left(\sum_{s=1}^{T} q_s\right) - \Gamma(0) = \Gamma\left(\sum_{s=1}^{T} q_s\right).$$

Since $\sum_{s=1}^{T} q_s \geq Q$, by definition of $\Gamma(\cdot)$ we have

$$\Gamma\left(\sum_{s=1}^{T} q_s\right) = Q\left(1 + \log \frac{\sum_{s=1}^{T} q_s}{Q}\right).$$

Finally, since

$$\sum_{s=1}^{T} q_s \leq TQ,$$

we have

$$\log \frac{\sum_{s=1}^{T} q_s}{Q} \leq \log T.$$

□

Theorem 1. *Denote* $C_0 := \left(\mathbb{E}\left[\max_{1\le t\le T}|\lambda_t^*|^4\right]\right)^{1/4}$, $C_1 := \left(\mathbb{E}\left[\max_{1\le t\le T}\|\mathbf{d}_t\|^4\right]\right)^{1/4}$. *Then, under Assumptions 1,2, 3, Algorithm 1 with scalar tuning and* $\eta_t = \frac{1}{\beta\sqrt{T}}$ *satisfies*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8C_0^2C_1^2\left(1+\log T\right)}{T^{3/2}} + \frac{\mathbb{E}[\Phi_{1:T}(\lambda^*)]}{T^{3/2}} + \frac{\sigma^2}{\sqrt{T}}, \tag{19}$$

where $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] = -\frac{\left(\sum_{t=1}^{T}\mathbb{E}[\langle\mathbf{g}_t,\mathbf{d}_t\rangle]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}$, if $\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] > 0$, *and* $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] = 0$ *otherwise. In particular, it holds that* $-T\sigma^2 \le \mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] \le 0$.

Proof. Using (12) with $\boldsymbol{\lambda}^* \leftarrow \lambda^*\mathbf{1}$, we have

$$\sum_{t=1}^{T}\left(\eta - \frac{\beta}{2}\eta^2\right)\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le \Delta_1 + \frac{\beta\eta^2}{2}\,\mathbb{E}\left[\mathrm{Regret}_T^{\phi}(\lambda^*)\right] + \frac{\beta\eta^2}{2}\mathbb{E}[\Phi_{1:T}(\lambda^*)] + \frac{\beta\eta^2 T}{2}\sigma^2, \tag{20}$$

where we will specify λ^* in the later part of this proof. We first bound $\mathbb{E}\left[\mathrm{Regret}_T^{\phi}(\lambda^*)\right]$. From Lemma 5 we have

$$\mathrm{Regret}_T^{\phi}(\lambda^*) := \sum_{t=1}^{T}\phi_t(\lambda_t) - \sum_{t=1}^{T}\phi_t(\lambda^*) \le \sum_{t=1}^{T}\left(\phi_t(\lambda_t) - \phi_t(\lambda_{t+1})\right).$$

If $\|\mathbf{d}_t\|^2 > 0$, we can rewrite ϕ_t as

$$\phi_t(\lambda) = \|\mathbf{g}_t\|^2 - \frac{\langle\mathbf{g}_t,\mathbf{d}_t\rangle^2}{\|\mathbf{d}_t\|^2} + (\lambda-\lambda_t^*)^2\|\mathbf{d}_t\|^2. \tag{21}$$

Suppose $\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2 > 0$. Then,

$$\begin{aligned}\lambda_t &= -\frac{\sum_{s=1}^{t-1}\langle\mathbf{g}_s,\mathbf{d}_s\rangle}{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}\\ &= \frac{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2\lambda_s^*}{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2},\end{aligned}$$

where λ_s^* is defined in (18). Thus, λ_t is a convex combination of the previous instantaneous minimizers $\left\{\lambda_s^*\right\}_{s=1}^{t-1}$. Then, since $|\lambda_t^*| \le \max_{1\le t\le T}|\lambda_t^*|$, for all t, we also have $|\lambda_t| \le \max_{1\le t\le T}|\lambda_t^*|$. When $\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2 = 0$, we have $\lambda_t = 0$. Hence, in either case,

$$|\lambda_t - \lambda_t^*| \le |\lambda_t| + |\lambda_t^*| \le 2\max_{1\le t\le T}|\lambda_t^*|.$$

Furthermore,

$$\begin{aligned}\lambda_{t+1} &= -\frac{\sum_{s=1}^{t}\langle\mathbf{g}_s,\mathbf{d}_s\rangle}{\sum_{s=1}^{t}\|\mathbf{d}_s\|^2}\\ &= \frac{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}{\sum_{s=1}^{t}\|\mathbf{d}_s\|^2}\left(-\frac{\sum_{s=1}^{t}\langle\mathbf{g}_s,\mathbf{d}_s\rangle}{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}\right)\\ &= \frac{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}{\sum_{s=1}^{t}\|\mathbf{d}_s\|^2}\left(-\frac{\sum_{s=1}^{t-1}\langle\mathbf{g}_s,\mathbf{d}_s\rangle + \langle\mathbf{g}_t,\mathbf{d}_t\rangle}{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}\right)\\ &= \frac{\sum_{s=1}^{t-1}\|\mathbf{d}_s\|^2}{\sum_{s=1}^{t}\|\mathbf{d}_s\|^2}\lambda_t + \frac{\|\mathbf{d}_t\|^2}{\sum_{s=1}^{t}\|\mathbf{d}_s\|^2}\lambda_t^*.\end{aligned}$$

Hence,

$$\lambda_{t+1} - \lambda_t^* = \frac{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} \lambda_t + \left(\frac{\|\mathbf{d}_t\|^2}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} - 1 \right) \lambda_t^*$$
$$= \frac{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} (\lambda_t - \lambda_t^*). \tag{22}$$

Using (21), (22), we have

$$\begin{aligned}
\phi_t(\lambda_t) - \phi_t(\lambda_{t+1}) &= \|\mathbf{d}_t\|^2 \left[(\lambda_t - \lambda_t^*)^2 - (\lambda_{t+1} - \lambda_t^*)^2 \right] \\
&= \|\mathbf{d}_t\|^2 \left[1 - \left(\frac{\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} \right)^2 \right] (\lambda_t - \lambda_t^*)^2 \\
&= \|\mathbf{d}_t\|^2 \left[\frac{\left(\sum_{s=1}^{t} \|\mathbf{d}_s\|^2 \right)^2 - \left(\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 \right)^2}{\left(\sum_{s=1}^{t} \|\mathbf{d}_s\|^2 \right)^2} \right] (\lambda_t - \lambda_t^*)^2 \\
&= \frac{\|\mathbf{d}_t\|^4 \left(\sum_{s=1}^{t} \|\mathbf{d}_s\|^2 + \sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 \right)}{\left(\sum_{s=1}^{t} \|\mathbf{d}_s\|^2 \right)^2} (\lambda_t - \lambda_t^*)^2 \\
&\leq 2 \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} (\lambda_t - \lambda_t^*)^2 \\
&\leq 8 \left(\max_{1 \leq t \leq T} |\lambda_t^*| \right)^2 \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2},
\end{aligned}$$

where the second-to-last inequality follows from $\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 \leq \sum_{s=1}^{t} \|\mathbf{d}_s\|^2$. Whenever $\|\mathbf{d}_t\|^2 = 0$, we have $\phi_t(\lambda_t) - \phi_t(\lambda_{t+1}) = 0$ and the above inequality is trivially true.
Next, suppose $\sum_{s=1}^{t-1} \|\mathbf{d}_s\|^2 = 0$. Then, $\lambda_t = 0$. If $\|\mathbf{d}_t\|^2 > 0$, then we have $\lambda_{t+1} = \lambda_t^*$, and thus from (21) we get

$$\begin{aligned}
\phi_t(\lambda_t) - \phi_t(\lambda_{t+1}) &= (\lambda_t^*)^2 \|\mathbf{d}_t\|^2 \\
&= (\lambda_t^*)^2 \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} \\
&\leq \left(\max_{1 \leq t \leq T} |\lambda_t^*| \right)^2 \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2} \\
&\leq 8 \left(\max_{1 \leq t \leq T} |\lambda_t^*| \right)^2 \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2}.
\end{aligned}$$

If $\|\mathbf{d}_t\|^2 = 0$, then $\phi_t(\lambda_t) - \phi_t(\lambda_{t+1}) = 0$. Hence, in all cases

$$(\phi_t(\lambda_t) - \phi_t(\lambda_{t+1})) \leq 8 \left(\max_{1 \leq t \leq T} |\lambda_t^*| \right)^2 r_t, \tag{23}$$

where

$$r_t := \begin{cases} \frac{\|\mathbf{d}_t\|^4}{\sum_{s=1}^{t} \|\mathbf{d}_s\|^2}, & \sum_{s=1}^{t} \|\mathbf{d}_s\|^2 > 0, \\ 0, & \sum_{s=1}^{t} \|\mathbf{d}_s\|^2 = 0. \end{cases}$$

Summing (23) for every $t = 1, \ldots, T$ and applying Lemma 6 with

$$q_t \leftarrow \|\mathbf{d}_t\|^2 \quad , \quad Q \leftarrow \max_{1 \leq t \leq T} \|\mathbf{d}_t\|^2$$

yields

$$\sum_{t=1}^{T}\left(\phi_t(\lambda_t)-\phi_t(\lambda_{t+1})\right)\le 8\left(\max_{1\le t\le T}|\lambda_t^*|\right)^2(1+\log T)\max_{1\le t\le T}\|\mathbf{d}_t\|^2.$$

Consequently,

$$\begin{aligned}\mathbb{E}\left[\mathrm{Regret}_T^{\phi}(\lambda^*)\right]&\le 8\,(1+\log T)\,\mathbb{E}\left[\left(\max_{1\le t\le T}|\lambda_t^*|\right)^2\max_{1\le t\le T}\|\mathbf{d}_t\|^2\right]\\&\le 8\,(1+\log T)\left(\mathbb{E}\left[\max_{1\le t\le T}|\lambda_t^*|^4\right]\right)^{1/2}\left(\mathbb{E}\left[\max_{1\le t\le T}\|\mathbf{d}_t\|^4\right]\right)^{1/2}\\&= 8\,(1+\log T)\,C_0^2C_1^2.\end{aligned}$$

Substituting the regret bound in (20) we obtain

$$\sum_{t=1}^{T}\left(\eta-\frac{\beta}{2}\eta^2\right)\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]\le\Delta_1+4\beta\eta^2C_0^2C_1^2\,(1+\log T)+\frac{\beta\eta^2}{2}\mathbb{E}[\Phi_{1:T}(\lambda^*)]+\frac{\beta\eta^2T}{2}\sigma^2. \tag{24}$$

Let $\eta=\frac{1}{\beta\sqrt{T}}$. Then, since $\eta\le\frac{1}{\beta}$, we have

$$\eta-\frac{\beta}{2}\eta^2\ge\frac{\eta}{2}.$$

Using the above inequality and substituting the choice of η in (24) yields

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]\le\frac{2\beta\Delta_1}{\sqrt{T}}+\frac{8C_0^2C_1^2\,(1+\log T)}{T^{3/2}}+\frac{\mathbb{E}[\Phi_{1:T}(\lambda^*)]}{T^{3/2}}+\frac{\sigma^2}{\sqrt{T}},$$

which is (4).

Next, let us analyze $\mathbb{E}[\Phi_{1:T}(\lambda^*)]$. We observe that $\Phi_{1:T}(\lambda)$ can be expressed as

$$\Phi_{1:T}(\lambda)=\sum_{t=1}^{T}\|\mathbf{g}_t+\lambda\mathbf{d}_t\|^2-\sum_{t=1}^{T}\|\mathbf{g}_t\|^2=2\lambda\sum_{t=1}^{T}\langle\mathbf{g}_t,\mathbf{d}_t\rangle+\lambda^2\sum_{t=1}^{T}\|\mathbf{d}_t\|^2.$$

Therefore,

$$\mathbb{E}\left[\Phi_{1:T}(\lambda)\right]=2\lambda\sum_{t=1}^{T}\mathbb{E}\left[\langle\mathbf{g}_t,\mathbf{d}_t\rangle\right]+\lambda^2\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right].$$

First, consider the case $\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]>0$. In this case, the coefficient of λ^2 is strictly positive, hence, $\mathbb{E}\left[\Phi_{1:T}(\lambda)\right]$ is convex in λ. By setting $\frac{\partial}{\partial\lambda}\mathbb{E}\left[\Phi_{1:T}(\lambda)\right]=0$, we obtain

$$\lambda^*=-\frac{\sum_{t=1}^{T}\mathbb{E}\left[\langle\mathbf{g}_t,\mathbf{d}_t\rangle\right]}{\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}.$$

Substituting this expression into $\mathbb{E}\left[\Phi_{1:T}(\lambda)\right]$ yields

$$\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right]=-\frac{\left(\sum_{t=1}^{T}\mathbb{E}\left[\langle\mathbf{g}_t,\mathbf{d}_t\rangle\right]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}\le 0.$$

Next, suppose that $\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]=0$. Since each term in the sum is non-negative, this implies $\mathbf{d}_t=0$ almost surely, for all $t\in[T]$. Consequently, $\langle\mathbf{g}_t,\mathbf{d}_t\rangle=0$ almost surely, for all $t\in[T]$, and

thus $\mathbb{E}\left[\Phi_{1:T}(\lambda)\right] = 0$, for all $\lambda \in \mathbb{R}$. In this case, we use the convention $\lambda^* = 0$. Therefore, we define the fixed comparator as the minimizer of $\mathbb{E}\left[\Phi_{1:T}(\lambda)\right]$, given by

$$\lambda^* = \begin{cases} -\frac{\sum_{t=1}^T \mathbb{E}[\langle \mathbf{g}_t, \mathbf{d}_t \rangle]}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}, & \sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] > 0, \\ 0, & \sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] = 0. \end{cases}$$

The corresponding value of the objective is

$$\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] = \begin{cases} -\frac{\left(\sum_{t=1}^T \mathbb{E}[\langle \mathbf{g}_t, \mathbf{d}_t \rangle]\right)^2}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}, & \sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] > 0, \\ 0, & \sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] = 0. \end{cases} \tag{25}$$

Hence, in either case we have $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] \leq 0$.

For the last part, since $\mathbb{E}\left[\mathbf{d}_t \middle| \mathcal{F}_{t-1}\right] = 0$, we have

$$\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t \rangle\right] = \mathbb{E}\left[\langle \mathbf{g}_t - \nabla \mathfrak{L}(\mathbf{w}_t), \mathbf{d}_t \rangle\right].$$

By Cauchy-Schwarz

$$\begin{aligned} \left(\sum_{t=1}^T \mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t \rangle\right]\right)^2 &\leq \left(\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{g}_t - \nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right]\right)\left(\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]\right) \\ &\leq T\sigma^2 \left(\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]\right). \end{aligned}$$

Therefore,

$$0 \leq \frac{\left(\sum_{t=1}^T \mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t \rangle\right]\right)^2}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]} \leq T\sigma^2.$$

Equivalently, using the expression for $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right]$ gives

$$-T\sigma^2 \leq \mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] \leq 0.$$

□

Recall in Corollary 1, we consider constant batch sizes such that $|B_t| = n$ and $\left|\tilde{B}_t\right| = N$ for all $t \in [T]$. We defined $\mathfrak{L}^f(\mathbf{w}) := \mathbb{E}\left[\ell\left(\mathbf{w}; \mathbf{x}, f(\mathbf{x})\right)\right]$, and we note that, conditional on $\mathcal{F}_{t-1}$,

$$\mathbb{E}\left[g^f_{t,\mathbf{w}_t} \middle| \mathcal{F}_{t-1}\right] = \nabla \mathfrak{L}^f(\mathbf{w}_t) \quad , \quad \mathbb{E}\left[g^y_{t,\mathbf{w}_t} \middle| \mathcal{F}_{t-1}\right] = \nabla \mathfrak{L}(\mathbf{w}_t).$$

Corollary 1. *Suppose the variance of a gradient computed from a single ML-labeled sample is uniformly bounded by σ_f^2, i.e., $\mathbb{E}\left[\left\|\nabla \ell\left(\mathbf{w}; \mathbf{x}, f(\mathbf{x})\right) - \nabla \mathfrak{L}^f(\mathbf{w})\right\|^2\right] \leq \sigma_f^2$, while σ_y^2 represents its counterpart using true labels. Additionally, let $\bar{\Xi}_T := \frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right]$ and $\Sigma_t^{y,f} := \mathrm{Cov}\left(\nabla \ell\left(\mathbf{w}_t; \mathbf{x}, y\right), \nabla \ell\left(\mathbf{w}_t; \mathbf{x}, f(\mathbf{x})\right) \middle| \mathcal{F}_{t-1}\right)$ denote the conditional covariance between stochastic gradients computed using true and ML labels at step t. Consider the case that $\exists t \in [T] : \mathbb{E}[\|\mathbf{d}_t\|^2] > 0$. Under the assumptions of Theorem 1, Algorithm 1 satisfies*

$$\bar{\Xi}_T \leq \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8C_0^2C_1^2\left(1+\log T\right)}{T^{3/2}} + \frac{1}{\sqrt{T}}\frac{\sigma_y^2}{n} - \frac{1}{\sqrt{T}}\frac{N}{n(n+N)}\frac{\left(\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\mathrm{tr}\left(\Sigma_t^{y,f}\right)\right]\right)^2}{\sigma_f^2}.$$

Proof. We have, for all $t \in [T]$, $\mathbb{E}\left[\|\mathbf{g}_t - \nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \frac{\sigma_y^2}{n}$, as n samples with true labels are used for the mini-batch stochastic gradient $\mathbf{g}_t$. Also, conditional on $\mathcal{F}_{t-1}$, the two ML-label gradient

estimates are independent and have the common mean $\nabla\mathfrak{L}^f(\mathbf{w}_t)$. Hence,

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] &= \mathbb{E}\left[\left\|\tilde{g}^f_{t,\mathbf{w}_t} - g^f_{t,\mathbf{w}_t}\right\|^2\right] \\ &= \mathbb{E}\left[\left\|(\tilde{g}^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w})) - (g^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w}))\right\|^2\right] \\ &= \mathbb{E}\left[\left\|\tilde{g}^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\|^2\right] + \mathbb{E}\left[\left\|g^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\|^2\right] \\ &\leq \frac{\sigma_f^2}{N} + \frac{\sigma_f^2}{n} = \left(\frac{1}{n} + \frac{1}{N}\right)\sigma_f^2. \end{aligned} \tag{26}$$

The third equality follows because the centered cross term has conditional expectation zero.

Furthermore,

$$\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t\rangle\right] = \mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t}, \tilde{g}^f_{t,\mathbf{w}_t} - g^f_{t,\mathbf{w}_t}\right\rangle\right] = \mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t}, \tilde{g}^f_{t,\mathbf{w}_t}\right\rangle\right] - \mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t}, g^f_{t,\mathbf{w}_t}\right\rangle\right]. \tag{27}$$

Since $\tilde{g}^f_{t,\mathbf{w}_t}$ is computed from the independent unlabeled batch, conditional on $\mathcal{F}_{t-1}$ it is independent of the labeled estimator $g^y_{t,\mathbf{w}_t}$. Hence, for the first term in (27) we have

$$\mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t}, \tilde{g}^f_{t,\mathbf{w}_t}\right\rangle\right] = \mathbb{E}\left[\left\langle \nabla\mathfrak{L}(\mathbf{w}_t), \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right]. \tag{28}$$

For the second term in (27), $g^y_{t,\mathbf{w}_t}$ and $g^f_{t,\mathbf{w}_t}$ are not independent, because they are computed using the same labeled batch. We can express the second term as follows

$$\mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t}, g^f_{t,\mathbf{w}_t}\right\rangle\right] = \mathbb{E}\left[\left\langle \nabla\mathfrak{L}(\mathbf{w}_t), \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right] + \mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t} - \nabla\mathfrak{L}(\mathbf{w}_t), g^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right]. \tag{29}$$

Substituting (28) and (29) into (27), we get

$$\begin{aligned}\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t\rangle\right] &= -\mathbb{E}\left[\left\langle g^y_{t,\mathbf{w}_t} - \nabla\mathfrak{L}(\mathbf{w}_t), g^f_{t,\mathbf{w}_t} - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right] \\ &= -\mathbb{E}\left[\left\langle \frac{1}{n}\sum_{(\mathbf{x},y)\in B_t}\nabla\ell\left(\mathbf{w}_t;\mathbf{x},y\right) - \nabla\mathfrak{L}(\mathbf{w}_t), \frac{1}{n}\sum_{(\mathbf{x},y)\in B_t}\nabla\ell\left(\mathbf{w}_t;\mathbf{x},f(\mathbf{x})\right) - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right] \\ &= -\frac{1}{n}\mathbb{E}\left[\left\langle \nabla\ell\left(\mathbf{w}_t;\mathbf{x},y\right) - \nabla\mathfrak{L}(\mathbf{w}_t), \nabla\ell\left(\mathbf{w}_t;\mathbf{x},f(\mathbf{x})\right) - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle\right] \\ &= -\frac{1}{n}\mathbb{E}\left[\mathbb{E}\left[\left\langle \nabla\ell\left(\mathbf{w}_t;\mathbf{x},y\right) - \nabla\mathfrak{L}(\mathbf{w}_t), \nabla\ell\left(\mathbf{w}_t;\mathbf{x},f(\mathbf{x})\right) - \nabla\mathfrak{L}^f(\mathbf{w}_t)\right\rangle \middle| \mathcal{F}_{t-1}\right]\right] \\ &= -\frac{1}{n}\mathbb{E}\left[\operatorname{tr}\left(\operatorname{Cov}\left(\nabla\ell\left(\mathbf{w}_t;\mathbf{x},y\right), \nabla\ell\left(\mathbf{w}_t;\mathbf{x},f(\mathbf{x})\right) \middle| \mathcal{F}_{t-1}\right)\right)\right]. \end{aligned}$$

Define $\Sigma_t^{y,f} := \operatorname{Cov}\left(\nabla\ell\left(\mathbf{w}_t;\mathbf{x},y\right), \nabla\ell\left(\mathbf{w}_t;\mathbf{x},f(\mathbf{x})\right) \middle| \mathcal{F}_{t-1}\right)$. Then, we can write

$$\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{d}_t\rangle\right] = -\frac{1}{n}\mathbb{E}\left[\operatorname{tr}\left(\Sigma_t^{y,f}\right)\right]. \tag{30}$$

For $\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] > 0$, substituting (26), (30) into the expression for $\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right]$ (25), we obtain

$$\mathbb{E}\left[\Phi_{1:T}(\lambda^*)\right] \leq -\frac{1}{T}\frac{N}{n(n+N)}\frac{\left(\sum_{t=1}^T \mathbb{E}\left[\operatorname{tr}\left(\Sigma_t^{y,f}\right)\right]\right)^2}{\sigma_f^2}.$$

Therefore, using the above bound and letting $\sigma^2 = \frac{\sigma_y^2}{n}$ in (24), we obtain

$$\bar{\Xi}_T \leq \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8C_0^2C_1^2\left(1+\log T\right)}{T^{3/2}} + \frac{1}{\sqrt{T}}\frac{\sigma_y^2}{n} - \frac{1}{\sqrt{T}}\frac{N}{n(n+N)}\frac{\left(\frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\operatorname{tr}\left(\Sigma_t^{y,f}\right)\right]\right)^2}{\sigma_f^2}.$$

□

Having finished the analysis of the scalar tuning case, we turn to vector tuning below. In vector tuning, we also adopt FTL, i.e., initialize $\boldsymbol{\lambda}_1 \leftarrow \mathbf{0}$ and for all $t > 1$, play

$$\boldsymbol{\lambda}_t \leftarrow \arg\min_{\boldsymbol{\lambda} \in \mathbb{R}^d} \sum_{s=1}^{t-1} \phi_s(\boldsymbol{\lambda}), \text{ where } \phi_s(\boldsymbol{\lambda}) := \|\mathbf{g}_s + \boldsymbol{\lambda} \odot \mathbf{d}_s\|^2 . \tag{31}$$

Because the loss function is separable, we have for all $i \in [d]$:

$$\boldsymbol{\lambda}_t[i] \leftarrow \arg\min_{\lambda \in \mathbb{R}} \sum_{s=1}^{t-1} \phi_t^{(i)}(\lambda), \text{ where } \phi_t^{(i)}(\lambda) := (\mathbf{g}_t[i] + \lambda \mathbf{d}_t[i])^2 .$$

Expanding the cumulative loss function gives

$$\begin{aligned}\sum_{s=1}^{t-1} \phi_t^{(i)}(\lambda) &= \sum_{s=1}^{t-1} (\mathbf{g}_s[i] + \lambda \mathbf{d}_s[i])^2 \\ &= \sum_{s=1}^{t-1} \mathbf{g}_s[i]^2 + 2\lambda \sum_{s=1}^{t-1} \mathbf{g}_s[i]\mathbf{d}_s[i] + \lambda^2 \sum_{s=1}^{t-1} \mathbf{d}_s[i]^2\end{aligned}$$

For $\sum_{s=1}^{t-1} \mathbf{d}_s[i]^2 > 0$, the objective is quadratic in λ. By setting $\frac{\partial}{\partial \lambda} \sum_{s=1}^{t-1} \phi_t^{(i)}(\lambda) = 0$, we obtain $\boldsymbol{\lambda}_t[i] = -\frac{\sum_{s=1}^{t-1} \mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t-1} \mathbf{d}_s[i]^2}$. Similarly to the case of scalar tuning, for $\sum_{s=1}^{t-1} \mathbf{d}_s[i]^2 = 0$ the cumulative loss function is independent in λ, and thus in that case we use $\boldsymbol{\lambda}_t[i] = 0$. Therefore, for all $i \in [d]$ we have

$$\boldsymbol{\lambda}_t[i] = \begin{cases} -\frac{\sum_{s=1}^{t-1} \mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t-1} \mathbf{d}_s[i]^2}, & \sum_{s=1}^{t-1} \mathbf{d}_s[i]^2 > 0, \\ 0, & \sum_{s=1}^{t-1} \mathbf{d}_s[i]^2 = 0, \end{cases}$$

which is (3). In the following, we denote

$$\text{Regret}_T^{\phi}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T} \phi_t(\boldsymbol{\lambda}_t) - \sum_{t=1}^{T} \phi_t(\boldsymbol{\lambda}^*). \tag{32}$$

For every $i \in [d]$, define the minimizer of the instantaneous loss function $\phi_t^{(i)}(\cdot)$ as

$$\boldsymbol{\lambda}_t^*[i] := \begin{cases} -\frac{\mathbf{g}_t[i]}{\mathbf{d}_t[i]}, & \mathbf{d}_t[i] \neq 0, \\ 0, & \mathbf{d}_t[i] = 0. \end{cases} \tag{33}$$

Additionally, define $\Phi_{1:T}(\boldsymbol{\lambda}) := \sum_{t=1}^{T} \left\|\mathcal{G}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}}\right\|^2 - \sum_{t=1}^{T} \left\|\mathcal{G}_{t,\mathbf{w}_t}^{\mathbf{0}}\right\|^2$. We are now ready to prove Theorem 3.

Theorem 3. *Suppose that Assumption 1, 2, 3 holds. Denote* $\tilde{C}_0 := \left(\mathbb{E}\left[\max_{t,i} |\boldsymbol{\lambda}_t^*[i]|^4\right]\right)^{1/4}$, $\tilde{C}_1 := \left(\mathbb{E}\left[\left(\sum_{i=1}^{d} \max_t \mathbf{d}_t[i]^2\right)^2\right]\right)^{1/4}$. *Then, Algorithm 1 with vector tuning and* $\eta_t = \frac{1}{\beta\sqrt{T}}$ *satisfies*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \frac{2\beta\Delta_1}{\sqrt{T}} + \frac{8\tilde{C}_0^2\tilde{C}_1^2(1+\log T)}{T^{3/2}} + \frac{\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)]}{T^{3/2}} + \frac{\sigma^2}{\sqrt{T}},$$

where $\mathbb{E}\left[\Phi_{1:T}(\boldsymbol{\lambda}^*)\right] = -\sum_{i \in \mathfrak{I}} \frac{\left(\sum_{t=1}^{T} \mathbb{E}[\mathbf{g}_t[i]\mathbf{d}_t[i]]\right)^2}{\sum_{t=1}^{T} \mathbb{E}[\mathbf{d}_t[i]^2]} \leq 0$, *when the set* $\mathfrak{I} := \{i \in [d] : \sum_{t=1}^{T} \mathbb{E}\left[\mathbf{d}_t[i]^2\right] > 0\}$ *is nonempty; otherwise,* $\mathbb{E}\left[\Phi_{1:T}(\boldsymbol{\lambda}^*)\right] = 0$. *In particular, it holds that* $-T\sigma^2 \leq \mathbb{E}\left[\Phi_{1:T}(\boldsymbol{\lambda}^*)\right] \leq 0$.

Proof. From (12), we have

$$\sum_{t=1}^{T} \left(\eta - \frac{\beta}{2}\eta^2\right) \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \Delta_1 + \frac{\beta\eta^2}{2}\mathbb{E}\left[\text{Regret}_T^{\phi}(\boldsymbol{\lambda}^*)\right] + \frac{\beta\eta^2}{2}\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)] + \frac{\beta\eta^2 T}{2}\sigma^2, \tag{34}$$

where we will specify $\boldsymbol{\lambda}^*$ in the later part of the proof. We first bound $\mathbb{E}\left[\mathrm{Regret}_T^{\phi}(\boldsymbol{\lambda}^*)\right]$. From the definition of regret (32) we have

$$\begin{aligned}\mathrm{Regret}_T^{\phi}(\boldsymbol{\lambda}^*) &:= \sum_{t=1}^{T}\phi_t(\boldsymbol{\lambda}_t) - \sum_{t=1}^{T}\phi_t(\boldsymbol{\lambda}^*)\\ &= \sum_{t=1}^{T}\sum_{i=1}^{d}\phi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \sum_{t=1}^{T}\sum_{i=1}^{d}\phi_t^{(i)}(\boldsymbol{\lambda}^*[i])\\ &= \sum_{i=1}^{d}\underbrace{\left(\sum_{t=1}^{T}\phi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \sum_{t=1}^{T}\phi_t^{(i)}(\boldsymbol{\lambda}^*[i])\right)}_{:=\mathrm{Regret}_T^{\phi^{(i)}}(\boldsymbol{\lambda}^*)}. \end{aligned} \tag{35}$$

Using Lemma 5 we have

$$\mathrm{Regret}_T^{\phi^{(i)}}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T}\phi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \sum_{t=1}^{T}\phi_t^{(i)}(\boldsymbol{\lambda}^*[i]) \le \sum_{t=1}^{T}\left(\phi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \phi_t^{(i)}(\boldsymbol{\lambda}_{t+1}[i])\right), \tag{36}$$

for all $i \in [d]$. Combining (35) and (36) we obtain

$$\mathrm{Regret}_T(\boldsymbol{\lambda}^*) \le \sum_{i=1}^{d}\sum_{t=1}^{T}\left(\phi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \phi_t^{(i)}(\boldsymbol{\lambda}_{t+1}[i])\right). \tag{37}$$

Fix some $i \in [d]$.
If $\mathbf{d}_t[i] \neq 0$ we can rewrite $\phi_t^{(i)}(\lambda)$ as

$$\phi_t^{(i)}(\lambda) = \left(\lambda - \boldsymbol{\lambda}_t^*[i]\right)^2 \mathbf{d}_t[i]^2. \tag{38}$$

Suppose $\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2 > 0$. Then,

$$\begin{aligned}\boldsymbol{\lambda}_t[i] &= -\frac{\sum_{s=1}^{t-1}\mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}\\ &= \frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2\boldsymbol{\lambda}_s^*[i]}{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2},\end{aligned}$$

where we used the expression of $\boldsymbol{\lambda}_s^*[i]$ (33). Thus, $\boldsymbol{\lambda}_t[i]$ is a convex combination of the previous instantaneous minimizers $\left\{\boldsymbol{\lambda}_s^*[i]\right\}_{s=1}^{t-1}$. Since $|\boldsymbol{\lambda}_t^*[i]| \le \max_{t,i}|\boldsymbol{\lambda}_t^*[i]|$, we also have $|\boldsymbol{\lambda}_t[i]| \le \max_{t,i}|\boldsymbol{\lambda}_t^*[i]|$. Therefore,

$$|\boldsymbol{\lambda}_t[i] - \boldsymbol{\lambda}_t^*[i]| \le |\boldsymbol{\lambda}_t[i]| + |\boldsymbol{\lambda}_t^*[i]| \le 2\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|.$$

Furthermore,

$$\begin{aligned}\boldsymbol{\lambda}_{t+1}[i] &= -\frac{\sum_{s=1}^{t}\mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\\ &= -\frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\frac{\sum_{s=1}^{t}\mathbf{g}_s[i]\mathbf{d}_s[i]}{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}\\ &= -\frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\frac{\sum_{s=1}^{t-1}\mathbf{g}_s[i]\mathbf{d}_s[i] + \mathbf{g}_t[i]\mathbf{d}_t[i]}{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}\\ &= \frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\boldsymbol{\lambda}_t[i] + \frac{\mathbf{d}_t[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\boldsymbol{\lambda}_t^*[i].\end{aligned}$$

Hence,

$$\begin{aligned}\boldsymbol{\lambda}_{t+1}[i]-\boldsymbol{\lambda}_t^*[i] &= \frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\boldsymbol{\lambda}_t[i]+\left(\frac{\mathbf{d}_t[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}-1\right)\boldsymbol{\lambda}_t^*[i]\\ &= \frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\left(\boldsymbol{\lambda}_t[i]-\boldsymbol{\lambda}_t^*[i]\right). \end{aligned} \tag{39}$$

From (38) and (39) we get

$$\begin{aligned}\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right) &= \mathbf{d}_t[i]^2\left[\left(\boldsymbol{\lambda}_t[i]-\boldsymbol{\lambda}_t^*[i]\right)^2-\left(\boldsymbol{\lambda}_{t+1}[i]-\boldsymbol{\lambda}_t^*[i]\right)^2\right]\\ &= \mathbf{d}_t[i]^2\left[1-\left(\frac{\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\right)^2\right]\left(\boldsymbol{\lambda}_t[i]-\boldsymbol{\lambda}_t^*[i]\right)^2\\ &= \frac{\mathbf{d}_t[i]^4\left(\sum_{s=1}^{t}\mathbf{d}_s[i]^2+\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2\right)}{\left(\sum_{s=1}^{t}\mathbf{d}_s[i]^2\right)^2}\left(\boldsymbol{\lambda}_t[i]-\boldsymbol{\lambda}_t^*[i]\right)^2\\ &\le 2\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\left(\boldsymbol{\lambda}_t[i]-\boldsymbol{\lambda}_t^*[i]\right)^2\\ &\le 8\left(\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|\right)^2\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2},\end{aligned}$$

where the second-to-last inequality follows from $\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2\le\sum_{s=1}^{t}\mathbf{d}_s[i]^2$. Whenever $\mathbf{d}_t[i]=0$, we have $\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right)=0$ and the above inequality is trivially true.
Next, suppose $\sum_{s=1}^{t-1}\mathbf{d}_s[i]^2=0$. Then, $\boldsymbol{\lambda}_t[i]=0$. If $\mathbf{d}_t[i]\neq 0$, we have $\boldsymbol{\lambda}_{t+1}[i]=\boldsymbol{\lambda}_t^*[i]$, and thus from (38) we get

$$\begin{aligned}\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right) &= \mathbf{d}_t[i]^2\boldsymbol{\lambda}_t^*[i]^2\\ &= \boldsymbol{\lambda}_t^*[i]^2\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\\ &\le \left(\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|\right)^2\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}\\ &\le 8\left(\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|\right)^2\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}.\end{aligned}$$

If, instead, $\mathbf{d}_t[i]=0$, then $\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right)=0$. Hence, in all cases

$$\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right)\le 8\left(\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|\right)^2 u_{t,i}, \tag{40}$$

where

$$u_{t,i}:=\begin{cases}\frac{\mathbf{d}_t[i]^4}{\sum_{s=1}^{t}\mathbf{d}_s[i]^2}, & \sum_{s=1}^{t}\mathbf{d}_s[i]^2>0,\\ 0, & \sum_{s=1}^{t}\mathbf{d}_s[i]^2=0.\end{cases}$$

Summing (40) for every $t=1,\dots,T$ and applying Lemma 6 with

$$q_t\leftarrow\mathbf{d}_t[i]^2\quad,\quad Q\leftarrow\max_t\mathbf{d}_t[i]^2$$

yields

$$\mathrm{Regret}_T^{\phi^{(i)}}(\boldsymbol{\lambda}^*)\le\sum_{t=1}^{T}\left(\phi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right)-\phi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right)\right)\le 8\left(1+\log T\right)\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|^2\max_t\mathbf{d}_t[i]^2.$$

Consequently,

$$\begin{aligned}\mathbb{E}\left[\mathrm{Regret}_T^{\phi}(\boldsymbol{\lambda}^*)\right] &= \sum_{i=1}^{d}\mathbb{E}\left[\mathrm{Regret}_T^{\phi^{(i)}}(\boldsymbol{\lambda}^*)\right] \\ &\leq 8\left(1+\log T\right)\mathbb{E}\left[\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|^2\sum_{i=1}^{d}\max_t \mathbf{d}_t[i]^2\right] \\ &\leq 8\left(1+\log T\right)\left(\mathbb{E}\left[\max_{t,i}|\boldsymbol{\lambda}_t^*[i]|^4\right]\right)^{1/2}\left(\mathbb{E}\left[\left(\sum_{i=1}^{d}\max_t \mathbf{d}_t[i]^2\right)^2\right]\right)^{1/2} \\ &= 8\left(1+\log T\right)\tilde{C}_0^2\tilde{C}_1^2.\end{aligned}$$

Substituting the regret bound in (34) we obtain

$$\sum_{t=1}^{T}\left(\eta-\frac{\beta}{2}\eta^2\right)\mathbb{E}\left[\|\nabla L(\mathbf{w}_t)\|^2\right] \leq \Delta_1 + 4\beta\eta^2\tilde{C}_0^2\tilde{C}_1^2\left(1+\log T\right)+\frac{\beta\eta^2}{2}\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)]+\frac{\beta\eta^2 T}{2}\sigma^2. \tag{41}$$

Let $\eta = \frac{1}{\beta\sqrt{T}}$. Then, since $\eta \leq \frac{1}{\beta}$, we have

$$\eta-\frac{\beta}{2}\eta^2 \geq \frac{\eta}{2}.$$

Using the above inequality and substituting the choice of η in (41) yields

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \leq \frac{2\beta\Delta_1}{\sqrt{T}}+\frac{8\tilde{C}_0^2\tilde{C}_1^2\left(1+\log T\right)}{T^{3/2}}+\frac{\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)]}{T^{3/2}}+\frac{\sigma^2}{\sqrt{T}}.$$

We continue by analyzing $\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)]$. We first observe that $\Phi_{1:T}\left(\boldsymbol{\lambda}\right)$ can be expressed as

$$\begin{aligned}\Phi_{1:T}\left(\boldsymbol{\lambda}\right) &= \sum_{t=1}^{T}\|\mathbf{g}_t+\boldsymbol{\lambda}\odot\mathbf{d}_t\|^2-\sum_{t=1}^{T}\|\mathbf{g}_t\|^2 \\ &= 2\sum_{t=1}^{T}\langle\mathbf{g}_t,\boldsymbol{\lambda}\odot\mathbf{d}_t\rangle+\sum_{t=1}^{T}\|\boldsymbol{\lambda}\odot\mathbf{d}_t\|^2 \\ &= 2\sum_{t=1}^{T}\sum_{i=1}^{d}\boldsymbol{\lambda}[i]\mathbf{g}_t[i]\mathbf{d}_t[i]+\sum_{t=1}^{T}\sum_{i=1}^{d}\boldsymbol{\lambda}[i]^2\mathbf{d}_t[i]^2 \\ &= \sum_{i=1}^{d}\left(2\boldsymbol{\lambda}[i]\sum_{t=1}^{T}\mathbf{g}_t[i]\mathbf{d}_t[i]+\boldsymbol{\lambda}[i]^2\sum_{t=1}^{T}\mathbf{d}_t[i]^2\right).\end{aligned}$$

Therefore,

$$\mathbb{E}\left[\Phi_{1:T}\left(\boldsymbol{\lambda}\right)\right] = \sum_{i=1}^{d}\left(2\boldsymbol{\lambda}[i]\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]+\boldsymbol{\lambda}[i]^2\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t[i]^2\right]\right). \tag{42}$$

Observe that $\mathbb{E}\left[\Phi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ is separable. We now define the comparator $\boldsymbol{\lambda}^*$ as the minimizer of $\mathbb{E}\left[\Phi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ given by

$$\boldsymbol{\lambda}^*[i] = \begin{cases}-\frac{\sum_{t=1}^{T}\mathbb{E}[\mathbf{g}_t[i]\mathbf{d}_t[i]]}{\sum_{t=1}^{T}\mathbb{E}[\mathbf{d}_t[i]^2]}, & \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t[i]^2\right]>0,\\ 0, & \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t[i]^2\right]=0,\end{cases}$$

for all $i\in[d]$. Substituting $\boldsymbol{\lambda}\leftarrow\boldsymbol{\lambda}^*$ into the expression for $\mathbb{E}\left[\Phi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ (42), we obtain

$$\mathbb{E}\left[\Phi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] = -\sum_{\substack{i\in[d]\\ \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t[i]^2\right]>0}}\frac{\left(\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t[i]^2\right]}\leq 0.$$

For the last part, define the centered stochastic gradient $\boldsymbol{\xi}_t := \mathbf{g}_t - \nabla \mathfrak{L}(\mathbf{w}_t)$. Then, Cauchy–Schwarz gives

$$\left(\sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]\mathbf{d}_t[i]\right]\right)^2 \leq \left(\sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]^2\right]\right)\left(\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{d}_t[i]^2\right]\right). \tag{43}$$

Also, we observe that

$$\mathbb{E}\left[\nabla \mathfrak{L}(\mathbf{w}_t)[i]\, \mathbf{d}_t[i]\right] = \mathbb{E}\left[\nabla \mathfrak{L}(\mathbf{w}_t)[i]\, \mathbb{E}\left[\mathbf{d}_t[i] \mid \mathcal{F}_{t-1}\right]\right] = 0,$$

which implies that

$$\sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]\mathbf{d}_t[i]\right] = \sum_{t=1}^{T} \mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]. \tag{44}$$

Combining (43) and (44), we obtain

$$\left(\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]\right)^2 \leq \left(\sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]^2\right]\right)\left(\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{d}_t[i]^2\right]\right).$$

Hence, for every $i \in \mathfrak{I}$,

$$\frac{\left(\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]\right)^2}{\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{d}_t[i]^2\right]} \leq \sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]^2\right].$$

Summing over the active coordinates, we obtain

$$0 \leq \underbrace{\sum_{i \in \mathfrak{I}} \frac{\left(\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{g}_t[i]\mathbf{d}_t[i]\right]\right)^2}{\sum_{t=1}^{T} \mathbb{E}\left[\mathbf{d}_t[i]^2\right]}}_{=-\mathbb{E}[\Phi_{1:T}(\boldsymbol{\lambda}^*)]} \leq \sum_{i \in \mathfrak{I}} \sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{\xi}_t[i]^2\right] \leq \sum_{t=1}^{T} \mathbb{E}\left[\|\boldsymbol{\xi}_t\|^2\right] \leq T\sigma^2.$$

This shows that $-T\sigma^2 \leq \mathbb{E}\left[\Phi_{1:T}(\boldsymbol{\lambda}^*)\right] \leq 0$.

□

C PROOFS OF THE THEORETICAL RESULTS IN SECTION 3

We will need a series of lemmas first. In the following, we denote

$$\delta_t := \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} - \nabla \mathfrak{L}_\mu(\mathbf{w}_t)$$

and define $\mathfrak{L}_\mu(\mathbf{w}) := \mathbb{E}_\zeta \left[\mathfrak{L}(\mathbf{w} + \mu\zeta)\right]$. Additionally, we denote

$$\mathbf{q}_t := q^y_{t,\mathbf{w}_t}, \qquad \mathbf{d}^0_t := \tilde{q}^{f_t}_{t,\mathbf{w}_t} - q^{f_t}_{t,\mathbf{w}_t}.$$

We note that for the scalar case, one can let $\boldsymbol{\lambda}_t \leftarrow \lambda_t \odot \mathbf{1}$, where $\mathbf{1}$ is the all-one-vector; consequently one can write $\mathbf{q}_t + \boldsymbol{\lambda}_t \odot \mathbf{d}^0_t = \mathbf{q}_t + \lambda_t \mathbf{d}^0_t$. Hence, the following lemmas also apply to the case of scalar tuning.

Lemma 7. *(Adapted from the proof of Theorem 3.2 (a) in Ghadimi & Lan (2013)) Consider $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}$. Under Assumption 1 and Assumption 2, it holds that*

$$\eta \sum_{t=1}^{T} \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2 \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T - 2\eta \sum_{t=1}^{T} \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \delta_t \rangle + \beta\eta^2 \sum_{t=1}^{T} \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2.$$

Proof. Denote $\Delta^\mu_1 := \mathfrak{L}_\mu(\mathbf{w}_1) - \inf_{\mathbf{w}} \mathfrak{L}_\mu(\mathbf{w})$. We have for every $t \in [T]$,

$$\mathfrak{L}_\mu(\mathbf{w}_{t+1}) \le \mathfrak{L}_\mu(\mathbf{w}_t) - \eta \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \rangle + \frac{\beta}{2}\eta^2 \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2 \tag{45}$$

$$= \mathfrak{L}_\mu(\mathbf{w}_t) - \eta \|\nabla \mathfrak{L}_\mu(\mathbf{w}_t)\|^2 - \eta \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \delta_t \rangle + \frac{\beta}{2}\eta^2 \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2. \tag{46}$$

A telescoping sum of the inequalities for all $t \in [T]$ above leads to

$$\sum_{t=1}^{T} \eta \|\nabla \mathfrak{L}_\mu(\mathbf{w}_t)\|^2 \le \Delta^\mu_1 - \sum_{t=1}^{T} \eta \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \delta_t \rangle + \frac{\beta}{2} \sum_{t=1}^{T} \eta^2 \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2$$

$$\le \Delta_1 + \mu^2\beta d - \sum_{t=1}^{T} \eta \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \delta_t \rangle + \frac{\beta}{2} \sum_{t=1}^{T} \eta^2 \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2, \tag{47}$$

where the last inequality follows from the known result in the literature that connects the initial gap of the smoothed objective and the initial gap of the original objective function, i.e., $\Delta^\mu_1 \le \Delta_1 + \mu^2\beta d$, see e.g., (3.11) in Ghadimi & Lan (2013).

Furthermore, (3.9) in Theorem 3.1 of Ghadimi & Lan (2013) provides the following lower bound relating the gradient norm of the smoothed objective to that of the original objective:

$$\frac{1}{2}\left(\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2 - \frac{\mu^2}{2}\beta^2(d+3)^3 \right) \le \|\nabla \mathfrak{L}_\mu(\mathbf{w}_t)\|^2. \tag{48}$$

Combining (47) and (48), we obtain

$$\eta \sum_{t=1}^{T} \|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2 \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T - 2\eta \sum_{t=1}^{T} \langle \nabla \mathfrak{L}_\mu(\mathbf{w}_t), \delta_t \rangle + \beta\eta^2 \sum_{t=1}^{T} \left\| \mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t} \right\|^2. \tag{49}$$

□

Lemma 8. *Assume that $\boldsymbol{\lambda}_t$ and $\mathbf{w}$ are $\mathcal{F}_{t-1}$-measurable. Assume also that the labeled observations in B_t and the unlabeled covariates in $\widetilde{B}_t$ have the same covariate marginal, and that $\zeta_t \sim \mathcal{N}(0, I_d)$ is independent of $\mathcal{H}_t := \sigma(\mathcal{F}_{t-1}, B_t, \widetilde{B}_t)$. Then, $\mathbb{E}_{t-1}\left[\mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}}\right] = \nabla \mathfrak{L}_\mu(\mathbf{w})$.*

Proof. We have

$$\begin{aligned}\mathbb{E}_{t-1}\left[q^y_{t,\mathbf{w}}\right] &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\mathbb{E}_{(\mathbf{x},y)}\left[\frac{\ell(\mathbf{w}+\mu\zeta_t;\mathbf{x},y)-\ell(\mathbf{w};\mathbf{x},y)}{\mu}\zeta_t\middle|\zeta_t\right]\right]\\ &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\frac{\mathfrak{L}(\mathbf{w}+\mu\zeta_t)-\mathfrak{L}(\mathbf{w})}{\mu}\zeta_t\right]\\ &= \nabla\mathfrak{L}_\mu(\mathbf{w}),\end{aligned} \tag{50}$$

where (50) follows from the standard identity for the Gaussian zeroth-order gradient estimator; see, e.g., (21) in Nesterov & Spokoiny (2017).

At the same time, we also have

$$\begin{aligned}\mathbb{E}_{t-1}\left[q^f_{t,\mathbf{w}}\right] &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\mathbb{E}_{\mathbf{x}}\left[\frac{\ell(\mathbf{w}+\mu\zeta_t;\mathbf{x},f(\mathbf{x}))-\ell(\mathbf{w};\mathbf{x},f(\mathbf{x}))}{\mu}\zeta_t\middle|\zeta_t\right]\right]\\ &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\frac{\mathfrak{L}^f(\mathbf{w}+\mu\zeta_t)-\mathfrak{L}^f(\mathbf{w})}{\mu}\zeta_t\right]\\ &= \nabla\mathfrak{L}^f_\mu(\mathbf{w}),\end{aligned}$$

where we denote $\mathfrak{L}^f(\mathbf{w}) := \mathbb{E}_{\mathbf{x}}[\ell(\mathbf{w};\mathbf{x},f(\mathbf{x}))]$ and $\mathfrak{L}^f_\mu(\mathbf{w})$ is the objective function after smoothing. Furthermore, under the assumption that the labeled observations in B_t and the unlabeled covariates in $\widetilde{B}_t$ have the same covariate marginal, we also have

$$\begin{aligned}\mathbb{E}_{t-1}\left[\tilde{q}^f_{t,\mathbf{w}}\right] &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\mathbb{E}_{\tilde{\mathbf{x}}}\left[\frac{\ell(\mathbf{w}+\mu\zeta_t;\tilde{\mathbf{x}},f(\tilde{\mathbf{x}}))-\ell(\mathbf{w};\tilde{\mathbf{x}},f(\tilde{\mathbf{x}}))}{\mu}\zeta_t\middle|\zeta_t\right]\right]\\ &= \mathbb{E}_{\zeta_t\sim\mathcal{N}(0,I_d)}\left[\frac{\mathfrak{L}^f(\mathbf{w}+\mu\zeta_t)-\mathfrak{L}^f(\mathbf{w})}{\mu}\zeta_t\right]\\ &= \nabla\mathfrak{L}^f_\mu(\mathbf{w}).\end{aligned}$$

Combining all the above, we obtain

$$\mathbb{E}_{t-1}\left[\mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}}\right] = \nabla\mathfrak{L}_\mu(\mathbf{w}) + \boldsymbol{\lambda}_t\odot\left(\nabla\mathfrak{L}^f_\mu(\mathbf{w})-\nabla\mathfrak{L}^f_\mu(\mathbf{w})\right) = \nabla\mathfrak{L}_\mu(\mathbf{w}).$$

□

Lemma 9. *Follow the assumptions and notations in Lemma 7 and Lemma 8, it holds that*

$$\eta\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T + \beta\eta^2\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right]. \tag{51}$$

Proof. Taking expectation on both sides of the inequality in Lemma 7 and noting that, by Lemma 8, $\mathbb{E}[\langle\nabla\mathfrak{L}_\mu(\mathbf{w}_t),\delta_t\rangle] = 0$, we obtain

$$\eta\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T + \beta\eta^2\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathcal{Q}^{\boldsymbol{\lambda}_t}_{t,\mathbf{w}_t}\right\|^2\right].$$

□

Lemma 10. *Let $\zeta_t \sim N(0,I_d)$ be a common Gaussian perturbation used in every function-value difference appearing in $\mathcal{Q}^{\boldsymbol{\lambda}}_{t,\mathbf{w}_t}$ at iteration t, and assume that ζ_t is independent of $\mathcal{H}_t := \sigma(\mathcal{F}_{t-1},B_t,\widetilde{B}_t)$. Define $\overline{\mathbf{q}}_t := \mathbb{E}\left[\mathbf{q}_t\middle|\mathcal{F}_{t-1},\zeta_t\right]$. Then, under Assumptions 2 and 3, for every $t\in[T]$,*

$$\mathbb{E}\left[\|\mathbf{q}_t\|^2\right] \le 2(d+2)\mathbb{E}\left[\|\mathbf{g}_t\|^2\right] + \frac{\beta^2\mu^2}{2}d(d+2)(d+4),$$

and

$$\mathbb{E}\left[\|\mathbf{q}_t - \overline{\mathbf{q}}_t\|^2\right] \le 2(d+2)\sigma^2 + \frac{\beta^2\mu^2}{2}d(d+2)(d+4).$$

Proof. Define the empirical loss function

$$\widehat{L}_{t,y}(\mathbf{w}) := \frac{1}{|B_t|}\sum_{(\mathbf{x},y)\in B_t}\ell(\mathbf{w};\mathbf{x},y),$$

whose gradient satisfies

$$\nabla\widehat{L}_{t,y}(\mathbf{w}) = \frac{1}{|B_t|}\sum_{(\mathbf{x},y)\in B_t}\nabla\ell(\mathbf{w};\mathbf{x},y) = g^y_{t,\mathbf{w}}.$$

Then, by definition of $\mathbf{q}_t$,

$$\mathbf{q}_t = q^y_{t,\mathbf{w}_t} = \frac{1}{|B_t|}\sum_{(\mathbf{x},y)\in B_t}\frac{\ell(\mathbf{w}_t+\mu\zeta_t;\mathbf{x},y)-\ell(\mathbf{w}_t;\mathbf{x},y)}{\mu}\zeta_t = \frac{\widehat{L}_{t,y}(\mathbf{w}_t+\mu\zeta_t)-\widehat{L}_{t,y}(\mathbf{w}_t)}{\mu}\zeta_t.$$

By the integral form of Taylor's theorem (e.g., Theorem 2.1 in Wright & Recht (2022)), we have

$$\begin{aligned}\widehat{L}_{t,y}(\mathbf{w}_t+\mu\zeta_t)-\widehat{L}_{t,y}(\mathbf{w}_t) &= \mu\left\langle\nabla\widehat{L}_{t,y}(\mathbf{w}_t),\zeta_t\right\rangle + \underbrace{\int_0^1\left\langle\nabla\widehat{L}_{t,y}(\mathbf{w}_t+s\mu\zeta_t)-\nabla\widehat{L}_{t,y}(\mathbf{w}_t),\mu\zeta_t\right\rangle ds}_{:=R_{t,y}}\\ &= \mu\left\langle g^y_{t,\mathbf{w}_t},\zeta_t\right\rangle + R_{t,y}\\ &= \mu\left\langle\mathbf{g}_t,\zeta_t\right\rangle + R_{t,y}.\end{aligned}$$

Therefore,

$$\mathbf{q}_t = \left\langle\mathbf{g}_t,\zeta_t\right\rangle\zeta_t + \frac{R_{t,y}}{\mu}\zeta_t. \tag{52}$$

Using the inequality $\|\mathbf{a}+\mathbf{b}\|^2 \le 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$, for any two vectors $\mathbf{a}$ and $\mathbf{b}$, we obtain

$$\|\mathbf{q}_t\|^2 = 2\left\langle\mathbf{g}_t,\zeta_t\right\rangle^2\|\zeta_t\|^2 + \frac{2R^2_{t,y}}{\mu^2}\|\zeta_t\|^2.$$

Taking expectation,

$$\mathbb{E}\left[\|\mathbf{q}_t\|^2\right] = 2\mathbb{E}\left[\left\langle\mathbf{g}_t,\zeta_t\right\rangle^2\|\zeta_t\|^2\right] + \frac{2}{\mu^2}\mathbb{E}\left[R^2_{t,y}\|\zeta_t\|^2\right]. \tag{53}$$

We now bound the two terms separately. Let $\mathcal{H}_t := \sigma\left(\mathcal{F}_{t-1}, B_t, \widetilde{B}_t\right)$. Conditional on $\mathcal{H}_t$, the stochastic gradient $\mathbf{g}_t$ is fixed, while the search direction remains independent and satisfies $\zeta_t \mid \mathcal{H}_t \sim \mathcal{N}(0, I_d)$. Let δ_{ij} denote the Kronecker delta, i.e.,

$$\delta_{ij} = \begin{cases}1, & i=j,\\ 0, & i\ne j.\end{cases}$$

Then, for standard Gaussian variables $\zeta \sim \mathcal{N}(0, I_d)$, Isserlis's theorem (e.g., Theorem 1.28 in Janson (1997)) gives

$$\mathbb{E}\Big[\zeta[i]\zeta[j]\zeta[k]\zeta[l]\Big] = \delta_{ij}\delta_{kl} + \delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}. \tag{54}$$

Expanding the first term in (53) and using the above identity for $\zeta_t \mid \mathcal{H}_t \sim \mathcal{N}(0, I_d)$, yields

$$\begin{aligned}\mathbb{E}\left[\left\langle\mathbf{g}_t,\zeta_t\right\rangle^2\|\zeta_t\|^2\,\middle|\,\mathcal{H}_t\right] &= \sum_{i,j,k=1}^d \mathbf{g}_t[i]\mathbf{g}_t[j]\mathbb{E}\Big[\zeta_t[i]\zeta_t[j]\zeta_t[k]^2\Big|\mathcal{H}_t\Big]\\ &= \sum_{i,j,k=1}^d \mathbf{g}_t[i]\mathbf{g}_t[j]\left(\delta_{ij}+2\delta_{ik}\delta_{jk}\right)\\ &= d\sum_{i=1}^d \mathbf{g}_t[i]^2 + 2\sum_{i=1}^d \mathbf{g}_t[i]^2\\ &= (d+2)\|\mathbf{g}_t\|^2.\end{aligned}$$

Therefore, the first term in (53) satisfies

$$2\mathbb{E}\left[\langle \mathbf{g}_t, \zeta_t\rangle^2 \|\zeta_t\|^2\right] = 2(d+2)\mathbb{E}\left[\|\mathbf{g}_t\|^2\right] \tag{55}$$

We next bound the second term in (53). We observe that the remainder satisfies

$$\begin{aligned}
|R_{t,y}| &= \left| \int_0^1 \left\langle \nabla \widehat{L}_{t,y}(\mathbf{w}_t + s\mu\zeta_t) - \nabla \widehat{L}_{t,y}(\mathbf{w}_t), \mu\zeta_t \right\rangle ds \right| \\
&\leq \int_0^1 \left| \left\langle \nabla \widehat{L}_{t,y}(\mathbf{w}_t + s\mu\zeta_t) - \nabla \widehat{L}_{t,y}(\mathbf{w}_t), \mu\zeta_t \right\rangle \right| ds \\
&\leq \int_0^1 \left\| \nabla \widehat{L}_{t,y}(\mathbf{w}_t + s\mu\zeta_t) - \nabla \widehat{L}_{t,y}(\mathbf{w}_t) \right\| \mu \|\zeta_t\| ds \\
&\leq \int_0^1 \beta s\mu \|\zeta_t\| \mu \|\zeta_t\| ds \\
&= \frac{\beta\mu^2}{2} \|\zeta_t\|^2,
\end{aligned}$$

where the first inequality follows from the triangle inequality for integrals, the second from the Cauchy-Schwarz inequality, and the third holds since the function $\widehat{L}_{t,y}(\mathbf{w})$ has β-Lipschitz continuous gradients by Assumption 2. Squaring the above and multiplying both sides by $\|\zeta_t\|^2$ yields

$$R_{t,y}^2 \|\zeta_t\|^2 \leq \frac{\beta^2\mu^4}{4} \|\zeta_t\|^6.$$

Consequently,

$$\begin{aligned}
\frac{2}{\mu^2}\mathbb{E}\left[R_{t,y}^2 \|\zeta_t\|^2\right] &\leq \frac{2}{\mu^2}\frac{\beta^2\mu^4}{4}\mathbb{E}\left[\|\zeta_t\|^6\right] \\
&= \frac{\beta^2\mu^2}{2}\mathbb{E}\left[\|\zeta_t\|^6\right] \\
&= \frac{\beta^2\mu^2}{2} d(d+2)(d+4),
\end{aligned} \tag{56}$$

where the last equality is obtained because $\|\zeta_t\|^2$ conditionally follows a chi-squared distribution with d degrees of freedom, i.e., $\|\zeta_t\|^2 \mid \mathcal{H}_t \sim \chi_d^2$. Substituting (55) and (56) into (53) gives

$$\mathbb{E}\left[\|\mathbf{q}_t\|^2\right] \leq 2(d+2)\mathbb{E}\left[\|\mathbf{g}_t\|^2\right] + \frac{\beta^2\mu^2}{2} d(d+2)(d+4),$$

which proves the first result. For the second result, we first observe that since $\mathbb{E}\left[\mathbf{g}_t \middle| \mathcal{F}_{t-1}\right] = \nabla\mathfrak{L}(\mathbf{w}_t)$ we have

$$\begin{aligned}
\overline{\mathbf{q}}_t &= \mathbb{E}\left[\mathbf{q}_t \middle| \mathcal{F}_{t-1}, \zeta_t\right] \\
&= \mathbb{E}\left[\langle \mathbf{g}_t, \zeta_t\rangle \zeta_t + \frac{R_{t,y}}{\mu}\zeta_t \middle| \mathcal{F}_{t-1}, \zeta_t\right] \\
&= \langle \nabla\mathfrak{L}(\mathbf{w}_t), \zeta_t\rangle \zeta_t + \frac{\mathbb{E}\left[R_{t,y} \middle| \mathcal{F}_{t-1}, \zeta_t\right]}{\mu}\zeta_t.
\end{aligned} \tag{57}$$

Subtracting (57) from (52) yields

$$\mathbf{q}_t - \overline{\mathbf{q}}_t = \langle \mathbf{g}_t - \nabla\mathfrak{L}(\mathbf{w}_t), \zeta_t\rangle \zeta_t + \frac{R_{t,y} - \mathbb{E}\left[R_{t,y} \middle| \mathcal{F}_{t-1}, \zeta_t\right]}{\mu}\zeta_t.$$

Hence,

$$\mathbb{E}\left[\|\mathbf{q}_t - \overline{\mathbf{q}}_t\|^2\right] \leq 2\mathbb{E}\left[\langle \mathbf{g}_t - \nabla\mathfrak{L}(\mathbf{w}_t), \zeta_t\rangle^2 \|\zeta_t\|^2\right] + \frac{2}{\mu^2}\mathbb{E}\left[\left(R_{t,y} - \mathbb{E}\left[R_{t,y} \middle| \mathcal{F}_{t-1}, \zeta_t\right]\right)^2 \|\zeta_t\|^2\right], \tag{58}$$

where the above inequality uses that $\|\mathbf{a}+\mathbf{b}\|^2 \le 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$, for any two vectors $\mathbf{a}$ and $\mathbf{b}$. We now bound the two terms in (58) separately. Let $\mathbf{v}_t := \mathbf{g}_t - \nabla\mathfrak{L}(\mathbf{w}_t)$. Expanding the first term in (58) and using the identity (58) for $\zeta_t \mid \mathcal{H}_t \sim \mathcal{N}(0, I_d)$, yields

$$\begin{aligned}
\mathbb{E}\left[\langle \mathbf{v}_t, \zeta_t\rangle^2 \|\zeta_t\|^2 \big| \mathcal{H}_t\right] &= \sum_{i,j,k=1}^{d} \mathbf{v}_t[i]\mathbf{v}_t[j]\mathbb{E}\Big[\zeta_t[i]\zeta_t[j]\zeta_t[k]^2\big|\mathcal{H}_t\Big] \\
&= \sum_{i,j,k=1}^{d} \mathbf{v}_t[i]\mathbf{v}_t[j]\left(\delta_{ij} + 2\delta_{ik}\delta_{jk}\right) \\
&= d\sum_{i=1}^{d}\mathbf{v}_t[i]^2 + 2\sum_{i=1}^{d}\mathbf{g}_t[i]^2 \\
&= (d+2)\|\mathbf{v}_t\|^2.
\end{aligned}$$

Therefore, the first term in (58) satisfies

$$\begin{aligned}
2\mathbb{E}\left[\langle \mathbf{g}_t - \nabla\mathfrak{L}(\mathbf{w}_t), \zeta_t\rangle^2 \|\zeta_t\|^2\right] &= 2(d+2)\mathbb{E}\left[\|\mathbf{g}_t - \nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \\
&\le 2(d+2)\sigma^2, \qquad (59)
\end{aligned}$$

where the inequality follows from Assumption 3. We next bound the second term in (58). We observe that

$$\begin{aligned}
\mathbb{E}\left[\left(R_{t,y} - \mathbb{E}\left[R_{t,y}\middle|\mathcal{F}_{t-1}, \zeta_t\right]\right)^2 \middle|\mathcal{F}_{t-1}, \zeta_t\right] &= \mathrm{Var}\left(R_{t,y}\middle|\mathcal{F}_{t-1}, \zeta_t\right) \\
&\le \mathbb{E}\left[R_{t,y}^2\middle|\mathcal{F}_{t-1}, \zeta_t\right]
\end{aligned}$$

Hence,

$$\mathbb{E}\left[\left(R_{t,y} - \mathbb{E}\left[R_{t,y}\middle|\mathcal{F}_{t-1}, \zeta_t\right]\right)^2 \|\zeta_t\|^2\right] \le \mathbb{E}\left[R_{t,y}^2\|\zeta_t\|^2\right].$$

Multiplying the above with $\frac{2}{\mu^2}$ and using (56) we further obtain

$$\begin{aligned}
\frac{2}{\mu^2}\mathbb{E}\left[\left(R_{t,y} - \mathbb{E}\left[R_{t,y}\middle|\mathcal{F}_{t-1}, \zeta_t\right]\right)^2 \|\zeta_t\|^2\right] &\le \frac{2}{\mu^2}\mathbb{E}\left[R_{t,y}^2\|\zeta_t\|^2\right] \\
&\le \frac{\beta^2\mu^2}{2}d(d+2)(d+4). \qquad (60)
\end{aligned}$$

Substituting (59) and (60) into (58) yields

$$\mathbb{E}\left[\|\mathbf{q}_t - \overline{\mathbf{q}}_t\|^2\right] \le 2(d+2)\sigma^2 + \frac{\beta^2\mu^2}{2}d(d+2)(d+4),$$

which proves the second result. □

Similar to our results in stochastic first-order optimization, we consider updating $\boldsymbol{\lambda}_t$ as a game, where in each round t, the online learner

- Plays $\boldsymbol{\lambda}_t \in \mathbb{R}^d$.
- After that, receives the loss function $\psi_t(\boldsymbol{\lambda}) := \left\|\mathcal{Q}^{\lambda}_{t,\mathbf{w}_t}\right\|^2 = \left\|\mathbf{q}_t + \boldsymbol{\lambda}\odot\mathbf{d}_t^0\right\|^2$.
- Incurs loss $\psi_t(\boldsymbol{\lambda}_t)$.

Therefore, below, we denote:

$$\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T}\psi_t(\boldsymbol{\lambda}_t) - \sum_{t=1}^{T}\psi_t(\boldsymbol{\lambda}^*),$$

where $\boldsymbol{\lambda}^* \in \mathbb{R}^d$ is a comparator. We reiterate that the case of vector tuning covers the case of scalar tuning as a special case. Specifically, given a scalar tuning parameter $\lambda \in \mathbb{R}$, a scalar-vector multiplication $\lambda\mathbf{v}$ can be rewritten as $\lambda\mathbf{v} = \lambda\mathbf{1}\odot\mathbf{v}$, where $\mathbf{1} \in \mathbb{R}^d$ is the all-ones vector.

Similar to the first-order case, the FTL strategy under scalar tuning solves the following:

$$\lambda_t \leftarrow \arg\min_{\lambda\in\mathbb{R}} \sum_{s=1}^{t-1} \psi_s(\lambda), \text{ where } \psi_s(\lambda) := \left\|\mathbf{q}_s + \lambda \mathbf{d}_s^0\right\|^2,$$

which gives

$$\lambda_t = \begin{cases} -\frac{\sum_{s=1}^{t-1}\langle \mathbf{q}_s, \mathbf{d}_s^0\rangle}{\sum_{s=1}^{t-1}\|\mathbf{d}_s^0\|^2}, & \sum_{s=1}^{t-1}\left\|\mathbf{d}_s^0\right\|^2 > 0, \\ 0, & \sum_{s=1}^{t-1}\left\|\mathbf{d}_s^0\right\|^2 = 0. \end{cases}$$

Define the minimizer of the instantaneous loss function $\psi_t(\cdot)$ as

$$\lambda_t^* := \begin{cases} -\frac{\langle \mathbf{q}_t, \mathbf{d}_t^0\rangle}{\|\mathbf{d}_t^0\|^2}, & \left\|\mathbf{d}_t^0\right\|^2 > 0, \\ 0, & \left\|\mathbf{d}_t^0\right\|^2 = 0. \end{cases}$$

Additionally, define $\Psi_{1:T}(\lambda) := \sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda}\right\|^2 - \sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{0}\right\|^2$. We are now ready to prove Theorem 2.

Theorem 2. *Denote* $D_0 := \left(\mathbb{E}\left[\max_{1\le t\le T}|\lambda_t^*|^4\right]\right)^{1/4}$, $D_1 := \left(\mathbb{E}\left[\max_{1\le t\le T}\left\|\mathbf{d}_t^0\right\|^4\right]\right)^{1/4}$, *and* $\bar{\Xi}_T := \frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]$. *Then, under Assumptions 1, 2, 3, Algorithm 2 with scalar tuning,* $\eta = \min\left\{\frac{1}{4\beta(d+2)}, \frac{1}{4\beta\sqrt{(d+2)T}}\right\}$, *and* $\mu = \frac{1}{\beta(d+3)\sqrt{T}}$ *satisfies*

$$\bar{\Xi}_T \le 16\beta\Delta_1\left(\tfrac{d+2}{T} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\right) + \tfrac{16d}{(d+3)^2}\left(\tfrac{d+2}{T^2} + \tfrac{\sqrt{d+2}}{T^{3/2}}\right) + \tfrac{d+3}{T} + \tfrac{4D_0^2D_1^2(1+\log T)}{\sqrt{d+2}\,T^{3/2}} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \tfrac{d(d+4)}{4(d+3)^2T} + \tfrac{\mathbb{E}[\Psi_{1:T}(\lambda^*)]}{2(d+2)T}\right),$$

where $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = -\frac{\left(\sum_{t=1}^T \mathbb{E}\left[\langle \mathbf{q}_t, \mathbf{d}_t^0\rangle\right]\right)^2}{\sum_{t=1}^T \mathbb{E}\left[\|\mathbf{d}_t^0\|^2\right]}$ *if* $\sum_{t=1}^T \mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] > 0$, *and* $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = 0$ *otherwise. In particular, it holds that*

$$-2(d+2)T\sigma^2 - \tfrac{d(d+2)(d+4)}{2(d+3)^2} \le \mathbb{E}[\Psi_{1:T}(\lambda^*)] \le 0.$$

Proof. From Lemma 9 we have

$$\eta\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T + \beta\eta^2\sum_{t=1}^T \mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda_t}\right\|^2\right]. \tag{61}$$

It remains to bound $\sum_{t=1}^T \mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda_t}\right\|^2\right]$. We observe that for any comparator $\lambda^* \in \mathbb{R}$ we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda_t}\right\|^2\right] \\
&= \mathbb{E}\left[\sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda_t}\right\|^2 - \sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda^*}\right\|^2\right] + \mathbb{E}\left[\sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda^*}\right\|^2\right] \\
&= \mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\lambda^*)\right] + \sum_{t=1}^T \mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\lambda^*}\right\|^2\right] \\
&= \mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\lambda^*)\right] + \sum_{t=1}^T \mathbb{E}\left[\left\|\mathbf{q}_t + \lambda^*\mathbf{d}_t^0\right\|^2\right] \\
&= \mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\lambda^*)\right] + \sum_{t=1}^T \mathbb{E}\left[\|\mathbf{q}_t\|^2\right] + \mathbb{E}\left[\underbrace{\sum_{t=1}^T \left\|\mathbf{q}_t + \lambda^*\mathbf{d}_t^0\right\|^2 - \sum_{t=1}^T \|\mathbf{q}_t\|^2}_{:=\Psi_{1:T}(\lambda^*)}\right]
\end{aligned} \tag{62}$$

Furthermore, by Lemma 10 we have

$$\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{q_t}\|^2\right] \le 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{g_t}\|^2\right] + \frac{\beta^2\mu^2}{2}d\,(d+2)\,(d+4)\,T. \tag{63}$$

Combining (62) with (63), we obtain

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T}\left\|\mathcal{Q}^{\lambda_t}_{t,\mathbf{w}_t}\right\|^2\right] &\le \mathbb{E}\left[\mathrm{Regret}^{\psi}_{T}(\lambda^*)\right] + 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{g_t}\|^2\right] \\
&\quad + \frac{\mu^2\beta^2}{2}d(d+2)(d+4)T + \mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right] \\
&= \mathbb{E}\left[\mathrm{Regret}^{\psi}_{T}(\lambda^*)\right] + 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}\left(\mathbf{w}_t\right)\|^2\right] \\
&\quad + 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{g}_t - \nabla\mathcal{L}\left(\mathbf{w}_t\right)\|^2\right] \\
&\quad + \frac{\mu^2\beta^2}{2}d(d+2)(d+4)T + \mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right] \\
&\le \mathbb{E}\left[\mathrm{Regret}^{\psi}_{T}(\lambda^*)\right] + 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}\left(\mathbf{w}_t\right)\|^2\right] + 2(d+2)\sigma^2T \\
&\quad + \frac{\mu^2\beta^2}{2}d(d+2)(d+4)T + \mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right],
\end{aligned} \tag{64}$$

where the last inequality follows from Assumption 3. We next bound $\mathbb{E}\left[\mathrm{Regret}^{\psi}_{T}\left(\lambda^*\right)\right]$. From Lemma 5 we have

$$\mathrm{Regret}^{\psi}_{T}\left(\lambda^*\right) := \sum_{t=1}^{T}\psi_t(\lambda_t) - \sum_{t=1}^{T}\psi_t(\lambda^*) \le \sum_{t=1}^{T}\left(\psi_t(\lambda_t) - \psi_t(\lambda_{t+1})\right).$$

Proceeding as in the proof of Theorem 1, we obtain

$$\sum_{t=1}^{T}\left(\psi_t(\lambda_t) - \psi_t(\lambda_{t+1})\right) \le \sum_{t=1}^{T} 8\left(\max_{1\le t\le T}|\lambda^*_t|\right)^2 r^0_t, \tag{65}$$

where

$$r^0_t := \begin{cases} \frac{\left\|\mathbf{d}^0_t\right\|^4}{\sum_{s=1}^{t}\left\|\mathbf{d}^0_s\right\|^2}, & \sum_{s=1}^{t}\left\|\mathbf{d}^0_s\right\|^2 > 0, \\ 0, & \sum_{s=1}^{t}\left\|\mathbf{d}^0_s\right\|^2 = 0. \end{cases}$$

From the r.h.s. (65), applying Lemma 6 with

$$q_t \leftarrow \left\|\mathbf{d}^0_t\right\|^2 \quad , \quad Q \leftarrow \max_{1\le t\le T}\left\|\mathbf{d}^0_t\right\|^2, \quad r_t \leftarrow r^0_t$$

yields

$$\sum_{t=1}^{T}\left(\psi_t(\lambda_t) - \psi_t(\lambda_{t+1})\right) \le 8\left(\max_{1\le t\le T}|\lambda^*_t|\right)^2(1+\log T)\max_{1\le t\le T}\left\|\mathbf{d}^0_t\right\|^2.$$

Consequently,

$$\begin{aligned}
\mathbb{E}\left[\mathrm{Regret}^{\psi}_{T}\left(\lambda^*\right)\right] &\le 8\,(1+\log T)\,\mathbb{E}\left[\left(\max_{1\le t\le T}|\lambda^*_t|\right)^2\max_{1\le t\le T}\left\|\mathbf{d}^0_t\right\|^2\right] \\
&\le 8\,(1+\log T)\left(\mathbb{E}\left[\max_{1\le t\le T}|\lambda^*_t|^4\right]\right)^{1/2}\left(\mathbb{E}\left[\max_{1\le t\le T}\left\|\mathbf{d}^0_t\right\|^4\right]\right)^{1/2} \\
&= 8\,(1+\log T)\,D_0^2D_1^2.
\end{aligned} \tag{66}$$

Combining (66), (64), and (61) yields

$$\begin{aligned}\left(\eta - 2\beta\left(d+2\right)\eta^2\right)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T \\ + 8\beta\eta^2\left(1+\log T\right)D_0^2D_1^2 + 2\beta\eta^2(d+2)\sigma^2T \\ + \beta\eta^2\mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right] + \frac{\mu^2\beta^3}{2}\eta^2 d(d+2)(d+4)T.\end{aligned} \tag{67}$$

We continue by analyzing $\mathbb{E}[\Psi_{1:T}(\lambda^*)]$. We observe that $\Psi_{1:T}(\lambda)$ can be expressed as

$$\Psi_{1:T}(\lambda) = \sum_{t=1}^{T}\left\|\mathbf{q}_t + \lambda\mathbf{d}_t^0\right\|^2 - \sum_{t=1}^{T}\left\|\mathbf{q}_t\right\|^2 = 2\lambda\sum_{t=1}^{T}\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle + \lambda^2\sum_{t=1}^{T}\left\|\mathbf{d}_t^0\right\|^2.$$

Therefore,

$$\mathbb{E}\left[\Psi_{1:T}(\lambda)\right] = 2\lambda\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right] + \lambda^2\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right].$$

First, consider the case $\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] > 0$. In this case, the coefficient of λ^2 is strictly positive, hence, $\mathbb{E}\left[\Psi_{1:T}(\lambda)\right]$ is convex in λ. By setting $\frac{\partial}{\partial\lambda}\mathbb{E}\left[\Psi_{1:T}(\lambda)\right] = 0$, we obtain

$$\lambda^* = -\frac{\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right]}{\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right]}.$$

Substituting this expression into $\mathbb{E}\left[\Psi_{1:T}(\lambda)\right]$ yields

$$\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = -\frac{\left(\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right]} \leq 0.$$

Next, suppose that $\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] = 0$. Since each term in the sum is non-negative, this implies $\mathbf{d}_t^0 = 0$ almost surely, for all $t \in [T]$. Consequently, $\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle = 0$ almost surely, for all $t \in [T]$, and thus $\mathbb{E}\left[\Psi_{1:T}(\lambda)\right] = 0$, for all $\lambda \in \mathbb{R}$. In this case, we use the convention $\lambda^* = 0$. Therefore, we define the fixed comparator as the minimizer of $\mathbb{E}\left[\Psi_{1:T}(\lambda)\right]$, given by

$$\lambda^* = \begin{cases} -\frac{\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right]}{\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right]}, & \sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] > 0, \\ 0, & \sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] = 0. \end{cases}$$

The corresponding value of $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right]$ is

$$\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] = \begin{cases} -\frac{\left(\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right]}, & \sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] > 0, \\ 0, & \sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right] = 0. \end{cases}$$

Hence, we have $\mathbb{E}\left[\Psi_{1:T}(\lambda^*)\right] \leq 0$. Next, we define

$$\overline{\mathbf{q}}_t := \mathbb{E}\left[\mathbf{q}_t\middle|\mathcal{F}_{t-1}, \zeta_t\right] = \frac{\mathcal{L}\left(\mathbf{w}_t + \mu\zeta_t\right) - \mathcal{L}\left(\mathbf{w}_t\right)}{\mu}\zeta_t.$$

Since $\mathbb{E}\left[\mathbf{d}_t^0\middle|\mathcal{F}_{t-1}, \zeta_t\right] = 0$, we have

$$\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right] = \mathbb{E}\left[\left\langle\mathbf{q}_t - \overline{\mathbf{q}}_t, \mathbf{d}_t^0\right\rangle\right].$$

By Cauchy-Schwarz

$$\left(\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t, \mathbf{d}_t^0\right\rangle\right]\right)^2 = \left(\sum_{t=1}^{T}\mathbb{E}\left[\left\langle\mathbf{q}_t - \overline{\mathbf{q}}_t, \mathbf{d}_t^0\right\rangle\right]\right)^2 \leq \left(\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q}_t - \overline{\mathbf{q}}_t\right\|^2\right]\right)\left(\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{d}_t^0\right\|^2\right]\right).$$

Therefore,

$$\mathbb{E}[\Psi_{1:T}(\lambda^*)] \geq -\sum_{t=1}^{T} \mathbb{E}\left[\left\|\mathbf{q}_t - \overline{\mathbf{q}}_t\right\|^2\right].$$

Additionally, by Lemma 10, we have

$$\sum_{t=1}^{T} \mathbb{E}\left[\left\|\mathbf{q}_t - \overline{\mathbf{q}}_t\right\|^2\right] \leq 2(d+2)T\sigma^2 + \frac{\beta^2\mu^2}{2} d(d+2)(d+4)T.$$

Consequently, we obtain the following bounds for $\mathbb{E}[\Psi_{1:T}(\lambda^*)]$,

$$-2(d+2)T\sigma^2 - \frac{\beta^2\mu^2}{2} d(d+2)(d+4)T \leq \mathbb{E}[\Psi_{1:T}(\lambda^*)] \leq 0. \tag{68}$$

Next, set $\eta = \min\left\{\frac{1}{4\beta(d+2)}, \frac{1}{4\beta\sqrt{(d+2)T}}\right\}$. Then,

$$2\beta\left(d+2\right)\eta \leq \frac{1}{2},$$

and hence

$$\left(\eta - 2\beta\left(d+2\right)\eta^2\right) \geq \frac{\eta}{2}.$$

Let $M := \max\left\{d+2, \sqrt{(d+2)T}\right\}$. Using the choice of η in (67) and rearranging, we obtain

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq \left(16\beta\Delta_1 + 16\mu^2\beta^2 d\right)\frac{M}{T} + \mu^2\beta^2(d+3)^3 + \frac{4D_0^2D_1^2\left(1+\log T\right)}{MT} \\ + \frac{d+2}{M}\left(\sigma^2 + \frac{1}{2(d+2)T}\mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right] + \frac{\mu^2\beta^2 d(d+4)}{4}\right).\end{aligned} \tag{69}$$

Using (68) we have

$$0 \leq \sigma^2 + \frac{1}{2(d+2)T}\mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right] + \frac{\mu^2\beta^2 d(d+4)}{4} \leq \sigma^2 + \frac{\mu^2\beta^2 d(d+4)}{4},$$

and thus the last term in (69) is nonnegative. Using the fact that

$$\sqrt{(d+2)T} \leq M \leq d+2+\sqrt{(d+2)T},$$

the bound (69) further implies

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq \left(16\beta\Delta_1 + 16\mu^2\beta^2 d\right)\left(\frac{d+2}{T} + \frac{\sqrt{d+2}}{\sqrt{T}}\right) \\ + \mu^2\beta^2(d+3)^3 + \frac{4D_0^2D_1^2\left(1+\log T\right)}{\sqrt{d+2}\,T^{3/2}} \\ + \frac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \frac{\mathbb{E}[\Psi_{1:T}(\lambda^*)]}{2(d+2)T} + \frac{\mu^2\beta^2 d(d+4)}{4}\right).\end{aligned}$$

Furthermore, we can choose $\mu = \frac{1}{\beta(d+3)\sqrt{T}}$, for which we obtain

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq 16\beta\Delta_1\left(\frac{d+2}{T} + \frac{\sqrt{d+2}}{\sqrt{T}}\right) \\ + \frac{16d}{(d+3)^2}\left(\frac{d+2}{T^2} + \frac{\sqrt{d+2}}{T^{3/2}}\right) \\ + \frac{d+3}{T} + \frac{4D_0^2D_1^2\left(1+\log T\right)}{\sqrt{d+2}\,T^{3/2}} \\ + \frac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \frac{\mathbb{E}\left[\Psi_{1:T}\left(\lambda^*\right)\right]}{2(d+2)T} + \frac{d(d+4)}{4(d+3)^2T}\right).\end{aligned}$$

Moreover, substituting the choice of μ in (68), we obtain

$$-2(d+2)T\sigma^2 - \frac{d(d+2)(d+4)}{2(d+3)^2} \le \mathbb{E}[\Psi_{1:T}(\lambda^*)] \le 0.$$

□

We now turn to vector tuning below. In vector tuning, we also adopt FTL, i.e., initialize $\boldsymbol{\lambda}_1 \leftarrow \mathbf{0}$ and for all $t > 1$, play

$$\boldsymbol{\lambda}_t \leftarrow \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^d} \sum_{s=1}^{t-1} \psi_s(\boldsymbol{\lambda}), \text{ where } \psi_s(\boldsymbol{\lambda}) := \left\|\mathbf{q}_s + \boldsymbol{\lambda}\odot \mathbf{d}_s^0\right\|^2. \tag{70}$$

Because the loss function is separable, we have for all $i \in [d]$:

$$\boldsymbol{\lambda}_t[i] \leftarrow \arg\min_{\lambda\in\mathbb{R}} \sum_{s=1}^{t-1} \psi_t^{(i)}(\lambda), \text{ where } \psi_t^{(i)}(\lambda) := \left(\mathbf{q}_t[i] + \lambda \mathbf{d}_t^0[i]\right)^2.$$

Similar to the first-order case, the FTL strategy under vector tuning gives for all $i \in [d]$

$$\boldsymbol{\lambda}_t[i] = \begin{cases} -\frac{\sum_{s=1}^{t-1} \mathbf{q}_s[i]\mathbf{d}_s^0[i]}{\sum_{s=1}^{t-1} \mathbf{d}_s^0[i]^2}, & \sum_{s=1}^{t-1} \mathbf{d}_s^0[i]^2 > 0, \\ 0, & \sum_{s=1}^{t-1} \mathbf{d}_s^0[i]^2 = 0, \end{cases}$$

which is (6). In the following, we denote

$$\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T} \psi_t(\boldsymbol{\lambda}_t) - \sum_{t=1}^{T} \psi_t(\boldsymbol{\lambda}^*). \tag{71}$$

For every $i \in [d]$, define the minimizer of the instantaneous loss function $\psi_t^{(i)}(\cdot)$ as

$$\boldsymbol{\lambda}_t^*[i] := \begin{cases} -\frac{\mathbf{q}_t[i]}{\mathbf{d}_t^0[i]}, & \mathbf{d}_t^0[i] \neq 0, \\ 0, & \mathbf{d}_t^0[i] = 0. \end{cases} \tag{72}$$

Additionally, define $\Psi_{1:T}(\boldsymbol{\lambda}) := \sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}}\right\|^2 - \sum_{t=1}^T \left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\mathbf{0}}\right\|^2$. We are now ready to prove Theorem 4.

Theorem 4. *Denote* $\tilde{D}_0 := \left(\mathbb{E}\left[\max_{t,i} |\boldsymbol{\lambda}_t^*[i]|^4\right]\right)^{1/4}$, $\tilde{D}_1 := \left(\mathbb{E}\left[\left(\sum_{i=1}^d \max_t \mathbf{d}_t^0[i]^2\right)^2\right]\right)^{1/4}$, *and* $\bar{\Xi}_T := \frac{1}{T}\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right]$. *Then, under Assumptions 1, 2, 3, Algorithm 2 with vector tuning,* $\eta = \min\left\{\frac{1}{4\beta(d+2)}, \frac{1}{4\beta\sqrt{(d+2)T}}\right\}$, *and* $\mu = \frac{1}{\beta(d+3)\sqrt{T}}$ *satisfies*

$$\bar{\Xi}_T \le 16\beta\Delta_1\left(\tfrac{d+2}{T} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\right) + \tfrac{16d}{(d+3)^2}\left(\tfrac{d+2}{T^2} + \tfrac{\sqrt{d+2}}{T^{3/2}}\right) + \tfrac{d+3}{T} + \tfrac{4D_0^2D_1^2(1+\log T)}{\sqrt{d+2}\,T^{3/2}} + \tfrac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \tfrac{d(d+4)}{4(d+3)^2T} + \tfrac{\mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)]}{2(d+2)T}\right),$$

where $\mathbb{E}\left[\Psi_{1:T}(\boldsymbol{\lambda}^*)\right] = -\sum_{i\in\mathfrak{J}} \frac{\left(\sum_{t=1}^T \mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right]\right)^2}{\sum_{t=1}^T \mathbb{E}[\mathbf{d}_t^0[i]^2]} \le 0$, *when the set* $\mathfrak{J} := \{i \in [d] : \sum_{t=1}^T \mathbb{E}\left[\mathbf{d}_t^0[i]^2\right] > 0\}$ *is nonempty; otherwise,* $\mathbb{E}\left[\Psi_{1:T}(\boldsymbol{\lambda}^*)\right] = 0$. *In particular, it holds that*

$$-2(d+2)T\sigma^2 - \tfrac{d(d+2)(d+4)}{2(d+3)^2} \le \mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)] \le 0.$$

Proof. From Lemma 9 we have

$$\eta\sum_{t=1}^T \mathbb{E}\left[\|\nabla\mathfrak{L}(\mathbf{w}_t)\|^2\right] \le 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T + \beta\eta^2\sum_{t=1}^T \mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}_t}\right\|^2\right]. \tag{73}$$

It remains to bound $\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}_t}\right\|^2\right]$. We observe that for any comparator $\boldsymbol{\lambda}^*\in\mathbb{R}^d$ we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}_t}\right\|^2\right]\\
&=\mathbb{E}\left[\sum_{t=1}^{T}\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}_t}\right\|^2-\sum_{t=1}^{T}\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}^*}\right\|^2\right]+\mathbb{E}\left[\sum_{t=1}^{T}\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}^*}\right\|^2\right]\\
&=\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}^*}\right\|^2\right]\\
&=\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q}_t+\boldsymbol{\lambda}^*\odot\mathbf{d}_t^0\right\|^2\right]\\
&=\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q}_t\right\|^2\right]+\mathbb{E}\left[\underbrace{\sum_{t=1}^{T}\left\|\mathbf{q}_t+\boldsymbol{\lambda}^*\odot\mathbf{d}_t^0\right\|^2-\sum_{t=1}^{T}\left\|\mathbf{q}_t\right\|^2}_{:=\Psi_{1:T}(\boldsymbol{\lambda}^*)}\right] \qquad (74)
\end{aligned}$$

Furthermore, by Lemma 10 we have

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q_t}\right\|^2\right]\le 2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{g_t}\right\|^2\right]+\frac{\beta^2\mu^2}{2}d\left(d+2\right)\left(d+4\right)T. \qquad (75)$$

Combining (74) with (75), we obtain

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T}\left\|\mathcal{Q}_{t,\mathbf{w}_t}^{\boldsymbol{\lambda}_t}\right\|^2\right]&\le\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{g_t}\right\|^2\right]\\
&\quad+\frac{\mu^2\beta^2}{2}d(d+2)(d+4)T+\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right]\\
&=\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}\left(\mathbf{w}_t\right)\right\|^2\right]\\
&\quad+2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{g}_t-\nabla\mathcal{L}\left(\mathbf{w}_t\right)\right\|^2\right]\\
&\quad+\frac{\mu^2\beta^2}{2}d(d+2)(d+4)T+\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right]\\
&\le\mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right]+2(d+2)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}\left(\mathbf{w}_t\right)\right\|^2\right]+2(d+2)\sigma^2T\\
&\quad+\frac{\mu^2\beta^2}{2}d(d+2)(d+4)T+\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right], \qquad (76)
\end{aligned}$$

We next bound $\mathbb{E}\left[\mathrm{Regret}_T^{\psi}\left(\boldsymbol{\lambda}^*\right)\right]$. From the definition of regret in (71) we have

$$\begin{aligned}
\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)&:=\sum_{t=1}^{T}\psi_t(\boldsymbol{\lambda}_t)-\sum_{t=1}^{T}\psi_t(\boldsymbol{\lambda}^*)\\
&=\sum_{t=1}^{T}\sum_{i=1}^{d}\psi_t^{(i)}(\boldsymbol{\lambda}_t[i])-\sum_{t=1}^{T}\sum_{i=1}^{d}\psi_t^{(i)}(\boldsymbol{\lambda}^*[i])\\
&=\sum_{i=1}^{d}\underbrace{\left(\sum_{t=1}^{T}\psi_t^{(i)}(\boldsymbol{\lambda}_t[i])-\sum_{t=1}^{T}\psi_t^{(i)}(\boldsymbol{\lambda}^*[i])\right)}_{:=\mathrm{Regret}_T^{\psi^{(i)}}(\boldsymbol{\lambda}^*)}. \qquad (77)
\end{aligned}$$

Using Lemma 5 we have

$$\mathrm{Regret}_T^{\psi^{(i)}}(\boldsymbol{\lambda}^*) := \sum_{t=1}^{T} \psi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \sum_{t=1}^{T} \psi_t^{(i)}(\boldsymbol{\lambda}^*[i]) \le \sum_{t=1}^{T} \left(\psi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \psi_t^{(i)}(\boldsymbol{\lambda}_{t+1}[i]) \right), \quad (78)$$

for all $i \in [d]$. Proceeding as in the proof of Theorem 3, we obtain

$$\psi_t^{(i)}\left(\boldsymbol{\lambda}_t[i]\right) - \psi_t^{(i)}\left(\boldsymbol{\lambda}_{t+1}[i]\right) \le 8 \left(\max_{t,i} |\boldsymbol{\lambda}_t^*[i]| \right)^2 u_{t,i}^0, \quad (79)$$

where

$$u_{t,i}^0 := \begin{cases} \frac{\mathbf{d}_t^0[i]^4}{\sum_{s=1}^t \mathbf{d}_s^0[i]^2}, & \sum_{s=1}^t \mathbf{d}_s^0[i]^2 > 0, \\ 0, & \sum_{s=1}^t \mathbf{d}_s^0[i]^2 = 0. \end{cases}$$

Summing (79) for every $t = 1, \ldots, T$ and applying Lemma 6 with

$$q_t \leftarrow \mathbf{d}_t^0[i]^2 \quad , \quad Q \leftarrow \max_t \mathbf{d}_t^0[i]^2$$

yields

$$\sum_{t=1}^{T} \left(\psi_t^{(i)}(\boldsymbol{\lambda}_t[i]) - \psi_t^{(i)}(\boldsymbol{\lambda}^*[i]) \right) \le 8 \left(1 + \log T\right) \max_{t,i} |\boldsymbol{\lambda}_t^*[i]|^2 \max_t \mathbf{d}_t^0[i]^2.$$

Consequently,

$$\begin{aligned} \mathbb{E}\left[\mathrm{Regret}_T^{\psi}(\boldsymbol{\lambda}^*)\right] &= \sum_{i=1}^{d} \mathbb{E}\left[\mathrm{Regret}_T^{\psi^{(i)}}(\boldsymbol{\lambda}^*)\right] \\ &\le 8\left(1+\log T\right) \mathbb{E}\left[\max_{t,i} |\boldsymbol{\lambda}_t^*[i]|^2 \sum_{i=1}^{d} \max_t \mathbf{d}_t^0[i]^2 \right] \\ &\le 8\left(1+\log T\right) \left(\mathbb{E}\left[\max_{t,i} |\boldsymbol{\lambda}_t^*[i]|^4 \right] \right)^{1/2} \left(\mathbb{E}\left[\left(\sum_{i=1}^{d} \max_t \mathbf{d}_t^0[i]^2 \right)^2 \right] \right)^{1/2} \\ &= 8\left(1+\log T\right) \tilde{D}_0^2 \tilde{D}_1^2. \end{aligned} \quad (80)$$

Combining (73), (76), and (80) yields

$$\begin{aligned} \left(\eta - 2\beta\left(d+2\right)\eta^2\right) \sum_{t=1}^{T} \mathbb{E}\left[\|\nabla \mathfrak{L}(\mathbf{w}_t)\|^2 \right] \le\ & 2\Delta_1 + 2\mu^2\beta d + \frac{\mu^2}{2}\beta^2(d+3)^3\eta T \\ & + 8\beta\eta^2\left(1+\log T\right)\tilde{D}_0^2\tilde{D}_1^2 + 2\beta\eta^2(d+2)T\sigma^2 \\ & + \beta\eta^2 \mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] + \frac{\mu^2\beta^3}{2}\eta^2 d(d+2)(d+4)T \end{aligned} \quad (81)$$

We continue by analyzing $\mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)]$. We observe that $\Psi_{1:T}(\boldsymbol{\lambda})$ can be expressed as

$$\begin{aligned} \Psi_{1:T}\left(\boldsymbol{\lambda}\right) &= \sum_{t=1}^{T} \left\| \mathbf{q}_t + \boldsymbol{\lambda} \odot \mathbf{d}_t^0 \right\|^2 - \sum_{t=1}^{T} \|\mathbf{q}_t\|^2 \\ &= 2\sum_{t=1}^{T} \langle \mathbf{q}_t, \boldsymbol{\lambda} \odot \mathbf{d}_t^0 \rangle + \sum_{t=1}^{T} \left\| \boldsymbol{\lambda} \odot \mathbf{d}_t^0 \right\|^2 \\ &= 2\sum_{t=1}^{T}\sum_{i=1}^{d} \boldsymbol{\lambda}[i]\mathbf{q}_t[i]\mathbf{d}_t^0[i] + \sum_{t=1}^{T}\sum_{i=1}^{d} \boldsymbol{\lambda}[i]^2 \mathbf{d}_t^0[i]^2 \\ &= \sum_{i=1}^{d} \left(2\boldsymbol{\lambda}[i] \sum_{t=1}^{T} \mathbf{q}_t[i]\mathbf{d}_t^0[i] + \boldsymbol{\lambda}[i]^2 \sum_{t=1}^{T} \mathbf{d}_t^0[i]^2 \right). \end{aligned}$$

Therefore,

$$\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}\right)\right] = \sum_{i=1}^{d}\left(2\boldsymbol{\lambda}[i]\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right] + \boldsymbol{\lambda}[i]^2\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t^0[i]^2\right]\right). \tag{82}$$

Hence, $\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ is separable. Similar to the first-order case, we define the comparator $\boldsymbol{\lambda}^*$ as the minimizer of $\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ given by

$$\boldsymbol{\lambda}^*[i] = \begin{cases} -\frac{\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right]}{\sum_{t=1}^{T}\mathbb{E}[\mathbf{d}_t^0[i]^2]}, & \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t^0[i]^2\right] > 0, \\ 0, & \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t^0[i]^2\right] = 0, \end{cases}$$

for all $i \in [d]$. Substituting $\boldsymbol{\lambda}^*$ into the expression for $\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}\right)\right]$ (82), we obtain

$$\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] = -\sum_{\substack{i\in[d] \\ \sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t^0[i]^2\right]>0}} \frac{\left(\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right]\right)^2}{\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{d}_t^0[i]^2\right]} \leq 0.$$

Next, we define

$$\overline{\mathbf{q}}_t := \mathbb{E}\left[\mathbf{q}_t\middle|\mathcal{F}_{t-1}, \zeta_t\right] = \frac{\mathcal{L}\left(\mathbf{w}_t + \mu\zeta_t\right) - \mathcal{L}\left(\mathbf{w}_t\right)}{\mu}\zeta_t.$$

Since $\mathbb{E}\left[\mathbf{d}_t^0\middle|\mathcal{F}_{t-1}, \zeta_t\right] = 0$, we have

$$\mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right] = \mathbb{E}\left[\left(\mathbf{q}_t[i] - \overline{\mathbf{q}}_t[i]\right)\mathbf{d}_t^0[i]\right].$$

By Cauchy-Schwarz

$$\begin{aligned}\left(\sum_{t=1}^{T}\mathbb{E}\left[\mathbf{q}_t[i]\mathbf{d}_t^0[i]\right]\right)^2 &= \left(\sum_{t=1}^{T}\mathbb{E}\left[\left(\mathbf{q}_t[i] - \overline{\mathbf{q}}_t[i]\right)\mathbf{d}_t^0[i]\right]\right)^2 \\ &\leq \left(\sum_{t=1}^{T}\mathbb{E}\left[\left(\mathbf{q}_t[i] - \overline{\mathbf{q}}_t[i]\right)^2\right]\right)\left(\sum_{t=1}^{T}\mathbb{E}\left[\left(\mathbf{d}_t^0[i]\right)^2\right]\right).\end{aligned}$$

Therefore,

$$\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] \geq -\sum_{i=1}^{d}\sum_{t=1}^{T}\mathbb{E}\left[\left(\mathbf{q}_t[i] - \overline{\mathbf{q}}_t[i]\right)^2\right] = -\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q}_t - \overline{\mathbf{q}}_t\right\|^2\right].$$

Additionally, by Lemma 10, we have

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{q}_t - \overline{\mathbf{q}}_t\right\|^2\right] \leq 2(d+2)T\sigma^2 + \frac{\beta^2\mu^2}{2}d(d+2)(d+4)T.$$

Consequently, we obtain the following bounds for $\mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)]$,

$$-2(d+2)T\sigma^2 - \frac{\beta^2\mu^2}{2}d(d+2)(d+4)T \leq \mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)] \leq 0. \tag{83}$$

Next, set $\eta = \min\left\{\frac{1}{4\beta(d+2)}, \frac{1}{4\beta\sqrt{(d+2)T}}\right\}$. Then,

$$2\beta\left(d+2\right)\eta \leq \frac{1}{2},$$

and hence

$$\left(\eta - 2\beta\left(d+2\right)\eta^2\right) \geq \frac{\eta}{2}.$$

Let $M := \max\left\{d+2, \sqrt{(d+2)T}\right\}$. Using the choice of η in (81) and rearranging, we obtain

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq \left(16\beta\Delta_1 + 16\mu^2\beta^2 d\right)\frac{M}{T} + \mu^2\beta^2(d+3)^3 + \frac{4\tilde{D}_0^2\tilde{D}_1^2\left(1+\log T\right)}{MT} + \frac{d+2}{M}\left(\sigma^2 + \frac{1}{2(d+2)T}\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] + \frac{\mu^2\beta^2 d(d+4)}{4}\right). \quad (84)$$

Using (83) we have

$$0 \leq \sigma^2 + \frac{1}{2(d+2)T}\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right] + \frac{\mu^2\beta^2 d(d+4)}{4} \leq \sigma^2 + \frac{\mu^2\beta^2 d(d+4)}{4},$$

and thus the last term in (84) is nonnegative. Using the fact that

$$\sqrt{(d+2)T} \leq M \leq d+2+\sqrt{(d+2)T},$$

the bound (84) further implies

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq \left(16\beta\Delta_1 + 16\mu^2\beta^2 d\right)\left(\frac{d+2}{T} + \frac{\sqrt{d+2}}{\sqrt{T}}\right) + \mu^2\beta^2(d+3)^3 + \frac{4\tilde{D}_0^2\tilde{D}_1^2\left(1+\log T\right)}{\sqrt{d+2}\,T^{3/2}} + \frac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \frac{\mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)]}{2(d+2)T} + \frac{\mu^2\beta^2 d(d+4)}{4}\right).$$

Furthermore, we can choose $\mu = \frac{1}{\beta(d+3)\sqrt{T}}$, for which we obtain

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathfrak{L}(\mathbf{w}_t)\right\|^2\right] \leq 16\beta\Delta_1\left(\frac{d+2}{T} + \frac{\sqrt{d+2}}{\sqrt{T}}\right) + \frac{16d}{(d+3)^2}\left(\frac{d+2}{T^2} + \frac{\sqrt{d+2}}{T^{3/2}}\right) + \frac{d+3}{T} + \frac{4\tilde{D}_0^2\tilde{D}_1^2\left(1+\log T\right)}{\sqrt{d+2}\,T^{3/2}} + \frac{\sqrt{d+2}}{\sqrt{T}}\left(\sigma^2 + \frac{\mathbb{E}\left[\Psi_{1:T}\left(\boldsymbol{\lambda}^*\right)\right]}{2(d+2)T} + \frac{d(d+4)}{4(d+3)^2T}\right).$$

Moreover, substituting the choice of μ in (83), we obtain

$$-2(d+2)T\sigma^2 - \frac{d(d+2)(d+4)}{2(d+3)^2} \leq \mathbb{E}[\Psi_{1:T}(\boldsymbol{\lambda}^*)] \leq 0.$$

□

D EXPERIMENTAL SETUP

D.1 TEXT CLASSIFICATION ON SST-2 AND SST-5

Tasks and model. We conduct experiments with RoBERTa-large (355M parameters) (Liu et al., 2019b) on text classification. Following the experimental setups of Malladi et al. (2023); Zhang et al. (2025a), we consider few-shot fine-tuning with $K = 32$ labeled examples per class on three text classification benchmarks: SST-2 (Socher et al., 2013; Wang et al., 2019), a binary sentiment classification task, and SST-5 (Socher et al., 2013), a five-class fine-grained sentiment classification task. For a detailed description of the downstream tasks and prompts we refer the reader to Malladi et al. (2023).

Dataset. For each task, we construct the labeled training and validation sets by independently sampling $K = 32$ examples per class from the original training split. Thus, SST-2 contains a total of 64 examples in each of the training and validation sets, whereas SST-5 contains a total of 160 examples in each set. For the test set, we use the official SST-2 validation split of 872 examples, since the labels for the GLUE test split are not publicly available. For SST-5, we use a class-balanced subset of $1{,}000$ examples drawn from the official labeled test split.

ML-labeler. To construct the ML-labeler $f(\cdot)$, we fine-tune a separate RoBERTa-large model for each task using a disjoint subset of the original training data, with 128 labeled examples per class. The labeler is trained for 500 optimization steps using Adam with a batch size of 8, a learning rate of $1e-5$, and a linear learning-rate decay schedule. The resulting labelers achieve test accuracies of 92.78% on SST-2, and 58.30% on SST-5. All remaining examples from the original SST-2 and SST-5 training sets, excluding those used for the training, validation, test, and labeler-training sets, are treated as unlabeled data.

Loss functions. For each labeled and unlabeled input, we pass the example through the corresponding fixed ML-labeler model and extract its logits. We apply a softmax to the ML-labeler logits to obtain a probability distribution over the task labels, which we cache and use as soft targets throughout fine-tuning. On labeled examples, the supervised loss is computed using the standard cross-entropy loss between the model predictions and the ground-truth class labels. The labeler-based loss terms, computed on both labeled and unlabeled examples, use the KL divergence between the ML-labeler distribution and the predictive distribution of the model being fine-tuned. Since the ML-labeler remains fixed throughout fine-tuning, its predictions can be precomputed and cached, avoiding additional forward passes through the ML-labeler during optimization.

Randomness. We report results across five random seeds. Randomness arises from the sampling of the training and validation sets, the sampling of labeled and unlabeled mini-batches during fine-tuning, the initialization of task-specific parameters, and, for zeroth-order methods, the Gaussian search directions $\zeta_t \sim N(0, I_d)$.

D.1.1 FIRST-ORDER METHODS

For the first-order experiments, we fine-tune the model for $1{,}000$ steps and compare four methods: standard SGD using only labeled data, PP-SSL with Online Tuning (Shoham et al., 2025), and the proposed Algorithm 1 with the proposed scalar tuning and vector tuning schemes. For brevity, we refer to the latter two methods as Scalar FTL and Vector FTL when presenting the experimental results below. We use a labeled batch size of $n = 16$ for all methods. For PP-SSL and our methods, we additionally use an unlabeled batch size of $N = 32$. Following Malladi et al. (2023), we use a linear learning-rate decay schedule for all methods. For each method, we independently tune the initial learning rate over $\{1e-5, 5e-5, 1e-4, 5e-4\}$. Learning rates are selected based on the mean validation accuracy across seeds and are reported in Table 1.

Results. Figures 4 and 5 report the test-accuracy results and corresponding power-tuning parameter trajectories for the first-order methods on SST-2 and SST-5.

On SST-2, PP-SSL, Scalar FTL, and Vector FTL all improve over the SGD baseline, with Scalar FTL achieving slightly higher mean test accuracy over most of the later stages of training. We find

that the values of the tuning parameters of our methods generally decrease smoothly over training. In contrast, the PP-SSL tuning parameter exhibits noticeably larger fluctuations throughout training, despite achieving comparable test accuracy.

On SST-5, the differences between the methods are slightly more pronounced. Vector FTL achieves the strongest performance overall, whereas Scalar FTL remains comparable to SGD without a noticeable improvement. This behavior is consistent with the tuning trajectories: the Scalar FTL tuning parameter quickly decays toward zero, effectively reducing the contribution of the ML-labeled correction, while the Vector FTL tuning parameter retains substantial coordinatewise variation for much longer, allowing it to continue exploiting informative directions.

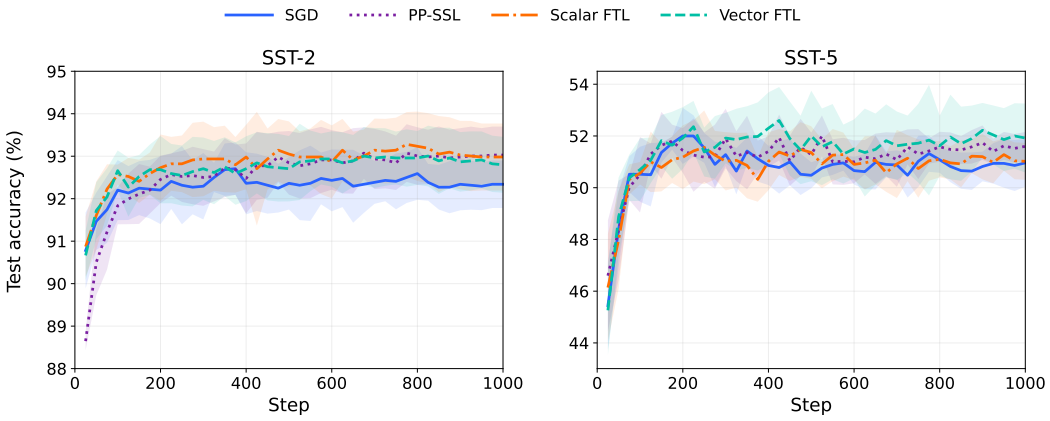


Figure 4: Test accuracy (%) over training steps for the first-order methods on SST-2 (left) and SST-5 (right). Test accuracy is evaluated every 25 optimization steps. Solid curves denote the mean across five random seeds, and shaded regions indicate ± 1 standard deviation.

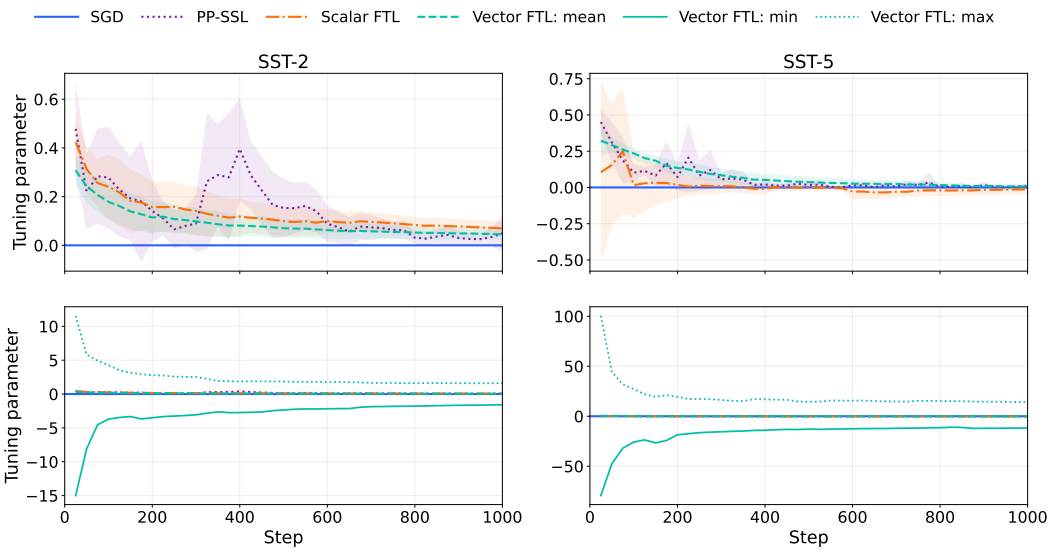


Figure 5: Tuning parameter trajectories for the first-order methods on SST-2 (left) and SST-5 (right). Tuning parameters are recorded every 25 optimization steps. **Top row**: Power-tuning parameter trajectories over optimization steps. **Bottom row**: Power-tuning parameter trajectories together with the across-seed means of the minimum and maximum coordinatewise Vector FTL parameters. Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation.

Method	SST-2	SST-5
SGD	$1e-4$	$1e-4$
PP-SSL	$5e-5$	$1e-4$
Scalar FTL	$1e-4$	$1e-4$
Vector FTL	$1e-4$	$1e-4$

Table 1: Learning rates selected using mean validation accuracy for each first-order method on SST-2 and SST-5.

D.1.2 Zeroth-order methods

For the zeroth-order experiments, we fine-tune the model for $10{,}000$ steps to account for the higher variance of the zeroth-order gradient estimates. Despite the larger number of optimization steps, we note that zeroth-order updates can be more computationally efficient per step, as they require only forward passes and avoid backpropagation. We compare three methods: standard stochastic gradient-free method without using ML-labeled data as well as Algorithm 2 with scalar tuning and vector tuning, which we denote as "ZO Scalar FTL" and "ZO Vector FTL" respectively in the following. We use a labeled batch size of $n = 32$ for all methods. For Algorithm 2, we additionally use an unlabeled batch size of $N = 64$. We set the perturbation parameter to $\mu = 1e-3$. For each method, we independently tune the learning rate over $\{5e-7, 1e-6, 5e-6\}$ and use a constant learning rate throughout fine-tuning. Learning rates are selected based on the mean validation accuracy across seeds and are reported in Table 2.

Results. Figures 6 and 7 report the test-accuracy results and corresponding tuning parameter trajectories for the zeroth-order methods on SST-2 and SST-5. On both SST-2 and SST-5, our methods consistently improve over the ZO baseline after the early stages of training, as the tuning parameters become more stable after aggregating information over more optimization steps. ZO Scalar FTL and ZO Vector FTL perform similarly throughout most of training. The corresponding tuning trajectories are smooth and closely aligned for the scalar and the mean of vector parameters. At the same time, each entry of $\boldsymbol{\lambda}_t$ in vector tuning retains a wider range, indicating that different coordinates continue to place different weights on the ML-labeled correction even when the average tuning parameter evolves similarly to the scalar one.

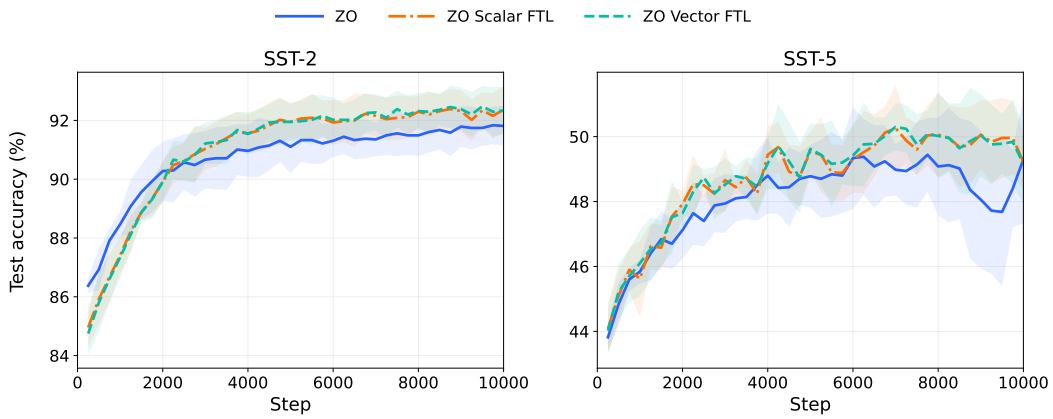


Figure 6: Test accuracy (%) over training steps for the zeroth-order methods on SST-2 (left) and SST-5 (right). Test accuracy is evaluated every 250 optimization steps. Solid curves denote the mean across five random seeds, and shaded regions indicate ± 1 standard deviation.

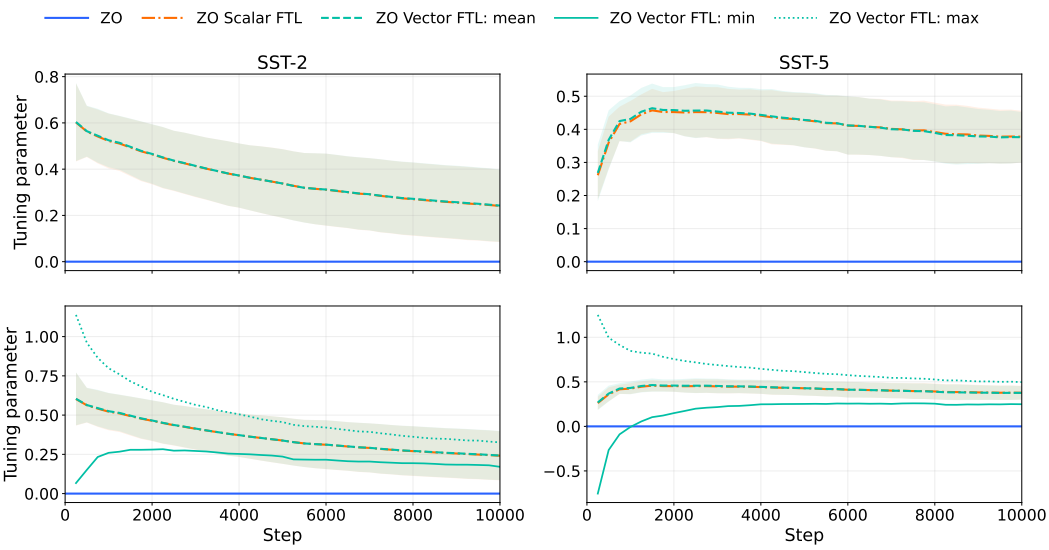


Figure 7: Tuning parameter trajectories for the zeroth-order methods on SST-2 (left) and SST-5 (right). Tuning parameters are recorded every 250 optimization steps. **Top row**: Power-tuning parameter trajectories over optimization steps. **Bottom row**: Power-tuning parameter trajectories together with the across-seed means of the minimum and maximum coordinatewise ZO Vector FTL parameters. Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation.

Method	SST-2	SST-5
ZO	$1e-6$	$1e-6$
ZO Scalar FTL	$1e-6$	$1e-6$
ZO Vector FTL	$1e-6$	$1e-6$

Table 2: Learning rates selected using mean validation accuracy for each zeroth-order method on SST-2 and SST-5.

D.2 Image Classification on CIFAR-10

Task and model. We conduct experiments on image classification using CIFAR-10 (Krizhevsky, 2009). We train a linear classification layer on top of a fixed ResNet-50 (He et al., 2016) pretrained on ImageNet (Deng et al., 2009). CIFAR-10 contains 50,000 training images and 10,000 test images distributed uniformly across 10 classes. Each image is resized and normalized using the preprocessing associated with the ImageNet-pretrained ResNet-50. We remove the final classification layer of the network and use the resulting 2,048-dimensional penultimate-layer representation as input to a linear classifier with 10 outputs. The ResNet-50 feature extractor remains fixed throughout all labeler- and model-training experiments.

Dataset. We partition the original CIFAR-10 training set into four mutually disjoint, class-balanced subsets. We use 5,000 examples to train the labeler model, 4,500 examples as the true-labeled training set, 5,000 examples as the validation set, and the remaining 35,500 examples as unlabeled data. The official CIFAR-10 test split of 10,000 images is used for final evaluation. The same data partition is used for all methods and random seeds. For the labeled, unlabeled, and labeler-training subsets, we apply a random resized crop to 224×224 pixels and a random horizontal flip before extracting the ResNet-50 features. Validation and test images are resized to 256×256 pixels and center-cropped to 224×224 pixels. Because the extracted representations are cached, the training augmentations are sampled once during feature extraction.

ML-labeler. To construct the ML-labeler model $f(\cdot)$, we train a linear classifier on top of the fixed $2{,}048$-dimensional features extracted by the ImageNet-pretrained ResNet-50. The classifier maps the feature representation to the 10 CIFAR-10 classes and is trained exclusively on the disjoint labeler-training set. We train the labeler for 50 epochs using AdamW with a learning rate of 1e-3, weight decay 0.01, and batch size 256. The resulting labeler achieves a test accuracy of 86.47% on CIFAR-10 and is subsequently kept fixed and shared across all experimental configurations.

Loss functions. For every labeled and unlabeled feature vector, we pass the representation through the fixed labeler to obtain its logits. A softmax converts these logits into a probability distribution over the 10 CIFAR-10 classes. On labeled examples, the supervised loss is the standard cross-entropy loss between the underlying model's prediction and the ground-truth label. The labeler-based losses on the labeled and unlabeled examples use the KL divergence between the labeler distribution and the predictive distribution of the model being trained.

Randomness. We report results across five random seeds. Randomness across seeds arises from the initialization of the classification layer and the ordering of the labeled and unlabeled mini-batches. For zeroth-order methods, it additionally arises from the Gaussian search directions $\zeta_t \sim \mathcal{N}(0, I_d)$, with a new direction sampled at every optimization step.

D.2.1 First-order methods

For the first-order experiments, we train each model for 100 epochs and compare four methods: standard SGD using only labeled examples, PP-SSL with Online Tuning (Shoham et al., 2025), and our method with scalar and vector-valued tuning parameters (Algorithm 1). We refer to ours as Scalar FTL and Vector FTL, respectively, when presenting the experimental results below for brevity. We tune the learning rate independently for each method and batch-size configuration over $\{5e-3,\ 1e-2,\ 5e-2,\ 1e-1\}$ and use a constant learning rate throughout training. For each method and batch-size configuration, the learning rate is selected based on the mean validation accuracy across five random seeds. The selected learning rates are reported in Table 4. To study the effect of labeled and unlabeled batch sizes, we consider several (n, N) configurations, with n and N denoting the labeled and unlabeled batch sizes, respectively. The full set of configurations is shown in Table 3.

n	N	SGD	PP-SSL	Scalar FTL	Vector FTL
64	128	86.29 ± 0.08	86.38 ± 0.10	$\mathbf{86.59 \pm 0.05}$	85.78 ± 0.47
64	256	86.29 ± 0.08	86.44 ± 0.05	$\mathbf{86.63 \pm 0.08}$	86.57 ± 0.04
64	512	86.29 ± 0.08	86.43 ± 0.05	$\mathbf{86.63 \pm 0.06}$	86.48 ± 0.17
64	1024	86.29 ± 0.08	86.43 ± 0.05	$\mathbf{86.63 \pm 0.06}$	$\mathbf{86.63 \pm 0.12}$
128	256	86.26 ± 0.07	86.39 ± 0.05	$\mathbf{86.47 \pm 0.06}$	85.97 ± 0.20
128	512	86.26 ± 0.07	86.40 ± 0.04	$\mathbf{86.53 \pm 0.06}$	86.32 ± 0.09
128	1024	86.26 ± 0.07	86.43 ± 0.05	$\mathbf{86.65 \pm 0.05}$	86.40 ± 0.09
256	512	86.43 ± 0.07	86.52 ± 0.05	$\mathbf{86.52 \pm 0.08}$	86.24 ± 0.06
256	1024	86.43 ± 0.07	86.55 ± 0.05	$\mathbf{86.64 \pm 0.05}$	86.25 ± 0.13
512	1024	86.08 ± 0.07	86.12 ± 0.08	$\mathbf{86.38 \pm 0.03}$	85.84 ± 0.09

Table 3: Test accuracy (%) on CIFAR-10. For each method and batch configuration, the learning rate and training checkpoint are selected based on mean validation accuracy, and test accuracy is reported at the selected checkpoint. Results report the mean $\pm$ standard error over five random seeds. For each batch configuration, the highest test accuracy is shown in bold.

Results. Table 3 reports the test accuracy obtained after selecting the learning rate and training checkpoint based on mean validation accuracy, while Figures 8 and 9 illustrate the corresponding test-accuracy and tuning-parameter trajectories for chosen batch-size configurations. Across the considered configurations, Scalar FTL provides the most consistent improvement, achieving the highest mean test accuracy in all configurations. PP-SSL also generally improves over the labeled-only SGD baseline, although by a smaller margin. In contrast, Vector FTL is more sensitive to the

batch-size configuration: it performs comparably to Scalar FTL for some choices of (n, N), but exhibits a noticeable degradation for others.

The tuning trajectories help explain these differences. The Scalar FTL tuning parameter decreases smoothly throughout training, while the PP-SSL tuning parameter rapidly approaches zero, thereby substantially reducing the contribution of the ML-labeled correction after the early stages of training. Although the mean of the Vector FTL parameters follows a similarly smooth decreasing trend, its coordinatewise parameters exhibit substantial dispersion, with some coordinates receiving large positive weights and others negative weights throughout training. This additional flexibility can be beneficial, but the pronounced coordinatewise variation is consistent with the greater variability of Vector FTL across batch-size configurations. Overall, these results suggest that the scalar parameter provides a favorable tradeoff between exploiting the ML-labeled data and maintaining stability across different labeled and unlabeled batch sizes.

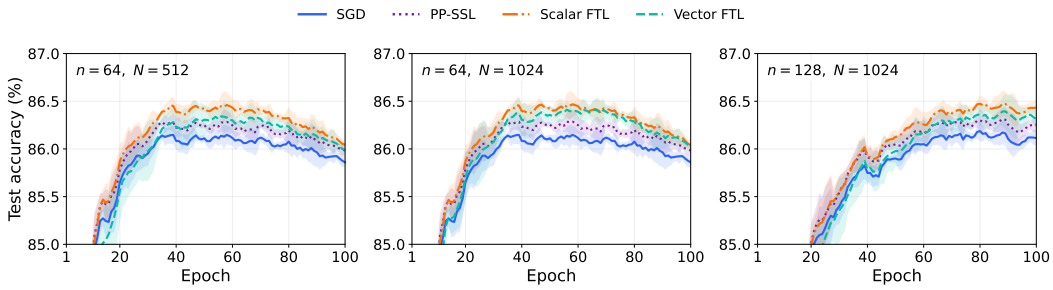


Figure 8: Test accuracy (%) over training epochs for the first-order methods on CIFAR-10 with $n = 64, N = 512$ (left), $n = 64, N = 1024$ (middle), and $n = 128, N = 1024$ (right). Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-accuracy curves are smoothed using a five-epoch moving average.

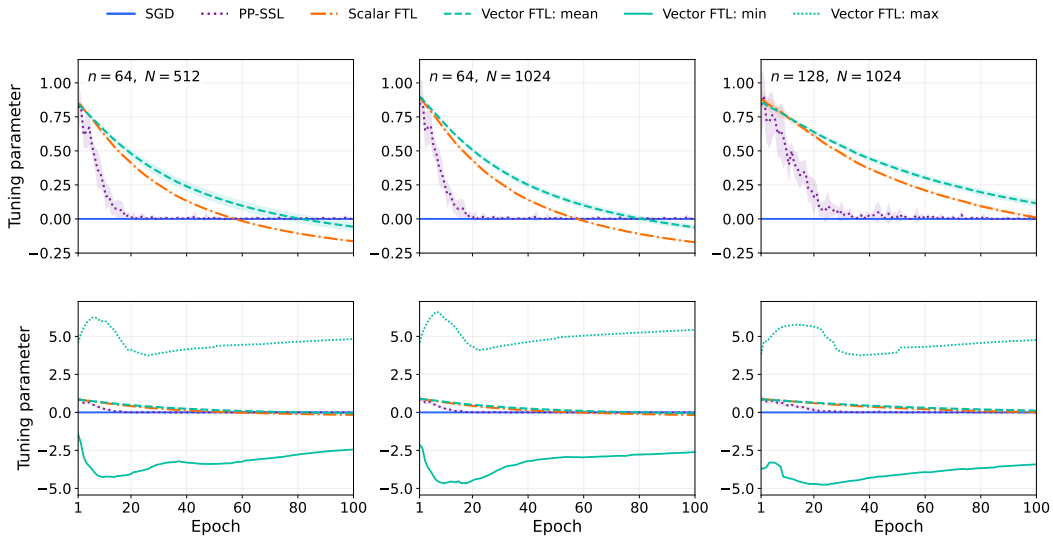


Figure 9: Tuning parameter trajectories for the first-order methods on CIFAR-10 with $n = 64, N = 512$ (left), $n = 64, N = 1024$ (middle), and $n = 128, N = 1024$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise Vector FTL parameters at each epoch. Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation.

n	N	SGD	PP-SSL	Scalar FTL	Vector FTL
64	128	$5e-2$	$5e-2$	$5e-2$	$5e-2$
64	256	$5e-2$	$5e-2$	$5e-2$	$5e-2$
64	512	$5e-2$	$5e-2$	$5e-2$	$5e-2$
64	1024	$5e-2$	$5e-2$	$5e-2$	$5e-2$
128	256	$5e-2$	$5e-2$	$5e-2$	$5e-2$
128	512	$5e-2$	$5e-2$	$5e-2$	$5e-2$
128	1024	$5e-2$	$5e-2$	$5e-2$	$5e-2$
256	512	$1e-1$	$1e-1$	$1e-1$	$1e-1$
256	1024	$1e-1$	$1e-1$	$1e-1$	$1e-1$
512	1024	$1e-1$	$1e-1$	$1e-1$	$1e-1$

Table 4: Learning rates selected using mean validation accuracy for each first-order method and batch-size configuration on CIFAR-10.

D.2.2 Zeroth-order methods

For the zeroth-order experiments, we train each model for $2{,}000$ epochs and compare three methods: standard stochastic gradient-free method without using ML-labeled data and the proposed Algorithm 2 with scalar tuning and vector tuning, which we denote as "ZO Scalar FTL" and "ZO Vector FTL" respectively in the following. We tune the learning rate independently for each method and batch-size configuration over $\{5e-4,\ 1e-3,\ 5e-3\}$ and use a constant learning rate throughout training. The selected learning rates are reported in Table 5. We set the perturbation parameter to $\mu = 1e-3$. To study the effect of labeled and unlabeled batch sizes, we consider $(n, N) \in \{(256, 512), (256, 1024), (512, 1024)\}$, where n and N denote the labeled and unlabeled batch sizes, respectively.

Results. Figures 10 and 11 show the test-accuracy results and corresponding tuning parameter trajectories for the zeroth-order methods on CIFAR-10. Similar to previous subsection, Algorithm 2 consistently improve over the ZO baseline across all three batch-size configurations, with ZO Scalar FTL and ZO Vector FTL achieving similar test accuracy throughout training. The improvement is sustained throughout the training, and the relative behavior is stable across the different choices of (n, N). The corresponding tuning trajectories are also notably smooth: the ZO Scalar FTL parameter and the mean of the tuning parameters of ZO Vector FTL remain closely aligned and gradually decrease over training. For a fixed labeled batch size $n = 256$, we observe both reduced variability in the tuning-parameter trajectories and larger tuning-parameter values as the unlabeled batch size N increases from 512 to 1024. Although the coordinatewise ZO Vector FTL parameters initially exhibit a wider range, this dispersion is reduced substantially as the training progresses. Overall, in the zeroth-order setting, both methods appear to exploit the ML-labeled correction in a stable manner across batch-size configurations, with little difference in predictive performance between the scalar and vector schemes.

n	N	ZO	ZO Scalar FTL	ZO Vector FTL
256	512	$5e-4$	$5e-4$	$5e-4$
256	1024	$5e-4$	$5e-4$	$5e-4$
512	1024	$1e-3$	$1e-3$	$1e-3$

Table 5: Learning rates selected using mean validation accuracy for each zeroth-order method and batch-size configuration on CIFAR-10.

D.3 Linear Regression on California Housing

Task and model. We conduct experiments on linear regression using the California Housing dataset from the LIBSVM repository (Chang & Lin, 2011). The task is to predict the median house value of a California census block from eight numerical features. We use a linear regression model

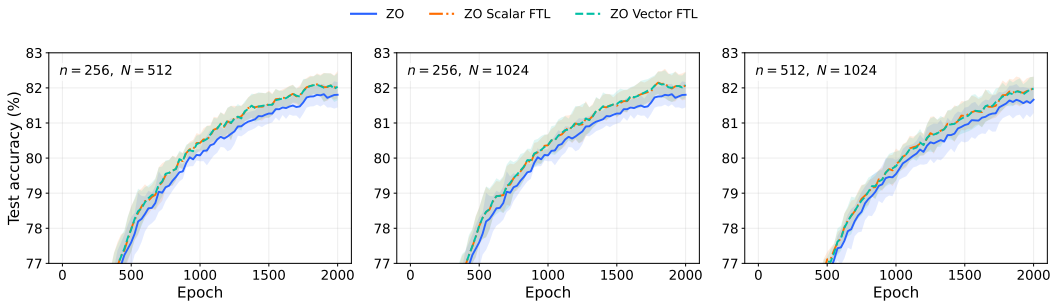


Figure 10: Test accuracy (%) over training epochs for the zeroth-order methods on CIFAR-10 with $n = 256, N = 512$ (left), $n = 256, N = 1024$ (middle), and $n = 512, N = 1024$ (right). Values are reported every 25 epochs. Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation.

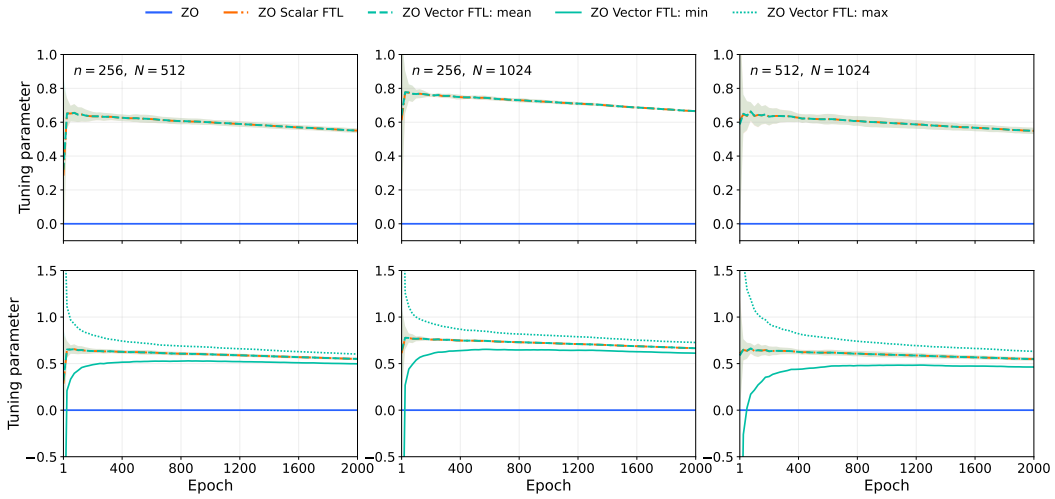


Figure 11: Tuning parameter trajectories for the zeroth-order methods on CIFAR-10 with $n = 256, N = 512$ (left), $n = 256, N = 1024$ (middle), and $n = 512, N = 1024$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise ZO Vector FTL parameters at each epoch. Values are reported every 25 epochs. Curves show the mean across five random seeds, with shaded regions indicating ± 1 standard deviation.

that maps the standardized eight-dimensional input vector to a single standardized target value. The same model architecture is used for the labeler and all models.

Dataset. The complete dataset contains $20{,}640$ examples. Because the dataset does not provide an official test split, we construct a fixed random split. We reserve $4{,}128$ examples, corresponding to 20% of the dataset, for testing. From the remaining examples, we use 500 examples to train the labeler, 80 examples as the true-labeled training set, and 80 examples as the validation set. The remaining $15{,}852$ examples are treated as unlabeled data. These subsets are mutually disjoint, and the same partition is used for all methods and random seeds.

ML-labeler. To construct the labeler model $f(\cdot)$, we train a linear regressor on the disjoint labeler-training set of 500 examples. The labeler is trained for 10 epochs using AdamW with a learning rate of $1e-2$, weight decay $1e-4$, and batch size 64. The resulting labeler achieves a standardized test MSE of 0.4191. After training, the labeler remains fixed and is shared across all experimental configurations.

Loss functions. For each labeled and unlabeled input, we pass the standardized feature vector through the fixed labeler and cache its scalar prediction. On labeled examples, the supervised loss is the mean squared error between the model's prediction and the ground-truth standardized target. The labeler-based losses on both labeled and unlabeled examples are mean squared errors between the model's prediction and the corresponding cached labeler prediction. Thus, the labeler-based terms use continuous soft regression targets. Because the labeler remains fixed, its predictions are computed once and reused throughout optimization.

Randomness. We report results across ten random seeds. The dataset partition and labeler model are held fixed across seeds. Randomness arises from the initialization of the linear model and the ordering and sampling of the labeled and unlabeled mini-batches. For zeroth-order methods, additional randomness arises from the Gaussian search directions $\zeta_t \sim \mathcal{N}(0, I_d)$.

D.3.1 First-order methods

For the first-order experiments, we train the linear regression model for 100 optimization steps and compare four methods: standard SGD using only labeled data, PP-SSL with Online Tuning (Shoham et al., 2025), and SGD with scalar tuning and vector tuning (i.e., Algorithm 1). We refer to the latter two as Scalar FTL and Vector FTL respectively below. We use a labeled batch size of $n = 16$ for all methods and consider unlabeled batch sizes $N \in \{64, 128, 256\}$ for PP-SSL and our methods. Since SGD does not use unlabeled examples, its trajectory is shared across the three batch-size configurations. We use a constant learning rate throughout training and consider the grid $\{5e-3, 1e-2, 5e-2, 1e-1\}$. Based on mean validation performance across seeds, we use a learning rate of $5e-2$ for all methods in the final comparison. The reported curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation.

Results. Figures 12 and 13 show the test MSE and corresponding tuning-parameter trajectories for the first-order methods on California Housing. Across all three unlabeled batch sizes, the methods that incorporate ML-labeled data improve upon the labeled-only SGD baseline after the initial optimization steps. PP-SSL and Scalar FTL perform comparably across configurations and generally achieve the lowest test MSE, while Vector FTL approaches similar performance later in training but exhibits greater variability early in optimization. The relative behavior of the methods remains broadly consistent as the unlabeled batch size increases from $N = 64$ to $N = 256$. The tuning-parameter trajectories stabilize quickly after the first few steps. Compared to our methods, the PP-SSL tuning parameter exhibits more pronounced fluctuations, particularly for smaller N. The PP-SSL, Scalar FTL, and the mean of Vector FTL parameters remain in a similar positive range throughout training, although their relative magnitudes vary across configurations. We also observe a slightly reduced across-seed variability in the tuning parameters as N increases, particularly after the first few oprimization steps. The coordinatewise Vector FTL parameters exhibit substantial dispersion, with some coordinates receiving negative weights and others considerably larger positive weights. This coordinatewise spread decreases over training but remains visible at the end of optimization, even as the mean of the Vector FTL tuning parameters stays close to the scalar tuning parameters.

D.3.2 Zeroth-order methods

For the zeroth-order experiments, we train the linear regression model for 1,000 optimization steps and compare three methods: Standard stochastic gradient-free method without using ML-labeled data and Algorithm 2 with scalar tuning and vector tuning, denoted as "ZO Scalar FTL" and "ZO Vector FTL" respectively. We tune the learning rate independently for each method and batch-size configuration over $\{1e-3, 5e-3,\ 1e-2\}$ and use a constant learning rate throughout training. Learning rates are selected based on the mean validation MSE across ten random seeds and are reported in Table 6. We set the perturbation parameter to $\mu = 1e$-3 and sample a new Gaussian search direction at every optimization step. To study the effect of the unlabeled batch size, we fix the labeled batch size to $n = 32$ and consider $(n, N) \in \{(32, 128), (32, 256), (32, 512)\}$, where N denotes the unlabeled batch size.

Results. Figures 14 and 15 present the test MSE and corresponding tuning-parameter trajectories for the zeroth-order methods on California Housing. Across all three unlabeled batch sizes, the

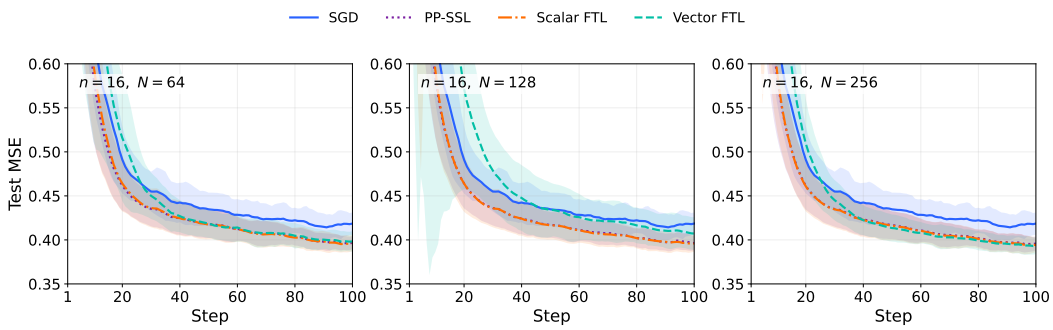


Figure 12: Test MSE over training steps for the first-order methods on California Housing with $n = 16, N = 64$ (left), $n = 16, N = 128$ (middle), and $n = 16, N = 256$ (right). Values are reported every 25 epochs. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-MSE curves are smoothed using a five-step moving average.

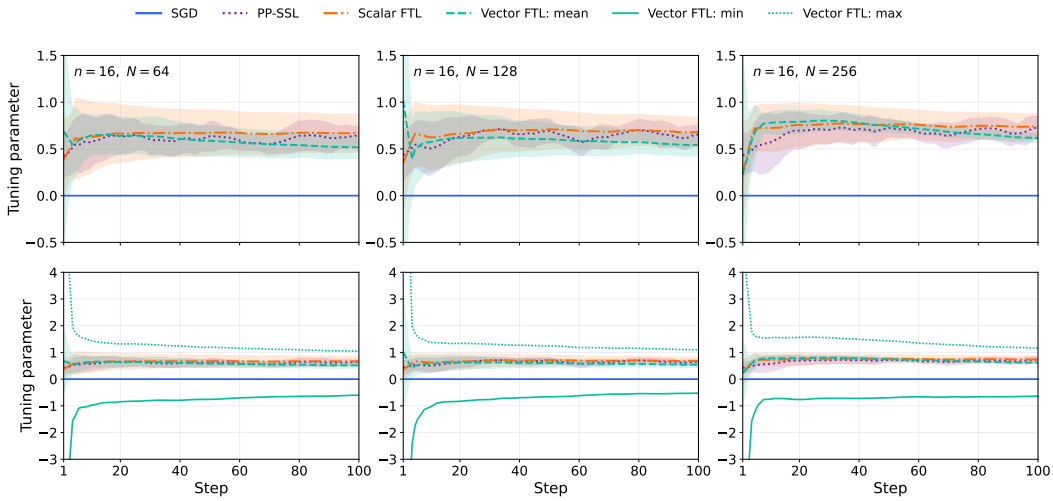


Figure 13: Tuning parameter trajectories for the first-order methods on California Housing with $n = 16, N = 64$ (left), $n = 16, N = 128$ (middle), and $n = 16, N = 256$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise Vector FTL parameters at each epoch. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, the tuning parameter curves are smoothed using a five-step moving average.

proposed Algorithm 2 improve over the true-labeled-only ZO baseline after the initial optimization period, with ZO Scalar FTL and ZO Vector FTL achieving similar test MSE throughout most of training. The improvement becomes particularly clear later in optimization and remains consistent as the unlabeled batch size increases from $N = 128$ to $N = 512$. The test-MSE trajectories also become somewhat less variable for larger N for the proposed methods. The tuning parameters stabilize rapidly after the initial optimization steps. The ZO Scalar FTL parameter and the mean of the ZO Vector FTL parameters remain closely aligned throughout training. For fixed n, the across-seed variability of these tuning parameters also tends to decrease as N increases. The coordinatewise ZO Vector FTL parameters initially exhibit a substantially wider range, but this dispersion is reduced quickly and continues to decrease throughout training. Despite this coordinatewise flexibility, the close agreement between the mean vector and scalar tuning parameters is consistent with the similar predictive performance.

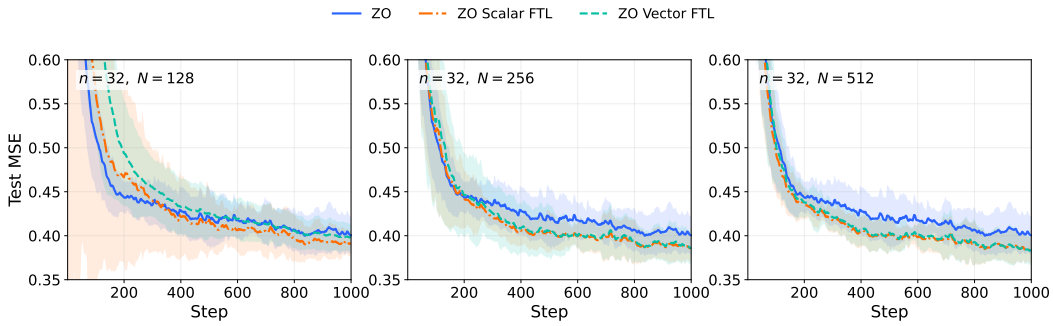


Figure 14: Test MSE over training steps for the zeroth-order methods on California Housing with $n = 32, N = 128$ (left), $n = 32, N = 256$ (middle), and $n = 32, N = 512$ (right). Values are reported every 25 epochs. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-MSE curves are smoothed using a five-step moving average.

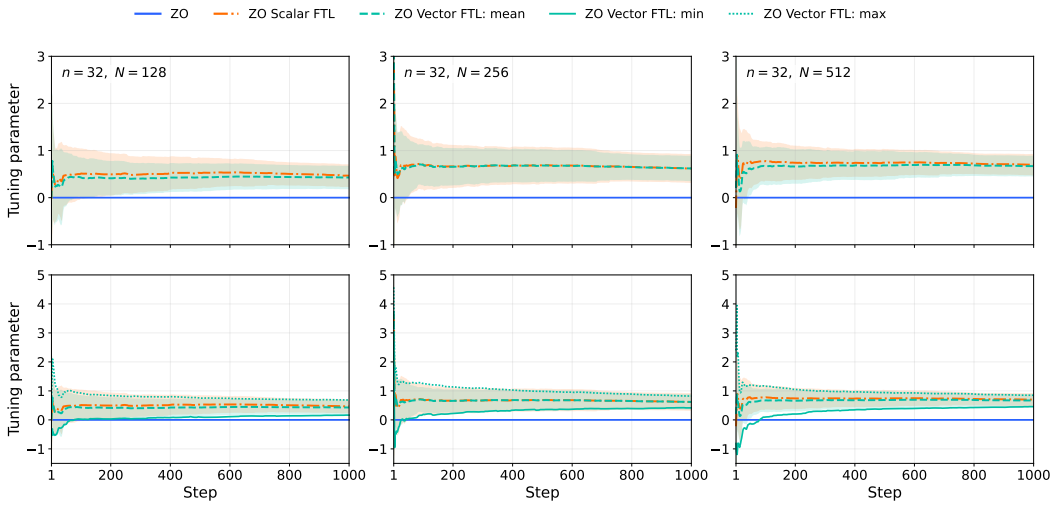


Figure 15: Tuning parameter trajectories for the first-order methods on California Housing with $n = 32, N = 128$ (left), $n = 32, N = 256$ (middle), and $n = 32, N = 512$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise Vector FTL parameters at each epoch. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, the tuning parameter curves are smoothed using a five-step moving average.

n	N	ZO	ZO Scalar FTL	ZO Vector FTL
32	128	$1e-2$	$1e-2$	$5e-3$
32	256	$1e-2$	$1e-2$	$1e-2$
32	1024	$1e-2$	$1e-2$	$1e-2$

Table 6: Learning rates selected using mean validation accuracy for each zeroth-order method and batch-size configuration on California Housing.

D.4 Classification on Satimage

Task and model. We conduct experiments on multiclass classification using the Statlog (Landsat Satellite) dataset, distributed as `satimage.scale` in the LIBSVM repository (Chang & Lin,

2011). The task is to classify the soil type associated with the center pixel of a satellite-image neighborhood. Each example contains 36 numerical features, corresponding to four spectral measurements for each pixel in a 3×3 neighborhood, and belongs to one of six classes. We train a linear classifier that maps the 36-dimensional input vector to six output logits. The dataset features are taken from the scaled version provided by LIBSVM.

Dataset. The dataset provides an official training split containing $4{,}435$ examples and an official test split containing $2{,}000$ examples. We partition the official training split into four mutually disjoint subsets. We use $1{,}200$ examples to train the labeler, 100 examples as the true-labeled training set, and 100 examples as the validation set. The remaining $3{,}035$ examples are treated as unlabeled data. The split approximately preserves the original class proportions in each subset. The official test split is used for final evaluation. The partition is generated once and is shared across all methods and training seeds.

ML-labeler. To construct the labeler model $f(\cdot)$, we train a two-layer multilayer perceptron on the disjoint labeler-training set of $1{,}200$ examples. The labeler consists of a linear layer mapping the 36 input features to 128 hidden units, followed by a ReLU activation and a linear output layer with six logits. We train the labeler for $2{,}000$ optimization steps using AdamW with a learning rate of $1e$-3, weight decay $1e$-4, and batch size 128. The labeler is trained using the standard cross-entropy loss and achieves a test accuracy of 87.55%. After training, the labeler remains fixed and is shared across all experimental configurations.

Loss functions. For every labeled and unlabeled input, we pass the feature vector through the fixed labeler and apply a softmax to its logits to obtain a probability distribution over the six classes. These labeler distributions are cached before training. On labeled examples, the supervised loss is the standard cross-entropy loss between the prediction of the model being trained and the ground-truth class label. The labeler-based losses on both labeled and unlabeled examples use the KL divergence between the cached labeler distribution and the predictive distribution from the model being trained. Because the labeler remains fixed, its predictions are computed once and reused throughout optimization.

Randomness. We report results across ten random seeds. Randomness arises from the initialization of the linear classifier and the sampling of labeled and unlabeled mini-batches. For the zeroth-order methods, additional randomness arises from the Gaussian search directions $\zeta_t \sim \mathcal{N}(0, I_d)$.

D.4.1 First-order methods

For the first-order experiments, we train the linear classifier for 500 optimization steps and compare four methods: standard SGD using only labeled data, PP-SSL with Online Tuning (Shoham et al., 2025), and our proposed methods with scalar tuning and vector tuning (i.e., Algorithm 1). We use a labeled batch size of $n = 16$ for all methods and consider unlabeled batch sizes $N \in \{64, 128, 256\}$ for PP-SSL and our methods. We use a constant learning rate throughout training and independently tune it for each method and batch-size configuration over $\{5e-3, 1e-2, 5e-2, 1e-1\}$. Based on the mean validation accuracy across the initial seeds, a learning rate of $1e-1$ is selected for all methods and configurations in the final comparison.

Results. Figures 16 and 17 present the test-accuracy results and corresponding tuning-parameter trajectories for the first-order methods on Satimage. Across all three unlabeled batch sizes, PP-SSL, Scalar FTL, and Vector FTL consistently outperform the labeled-only SGD baseline after the initial optimization period. The three methods that leverage ML-labeled data achieve similar test accuracy overall. The performance of Vector FTL improves as the unlabeled batch size increases, with larger N yielding higher test accuracy in the later stages of training. The tuning-parameter trajectories stabilize rapidly after the first few optimization steps. The Scalar FTL and mean of Vector FTL parameters remain close to one another throughout training and are relatively smooth, whereas the PP-SSL tuning parameter exhibits somewhat larger fluctuations. The coordinatewise Vector FTL parameters show a great dispersion, with some coordinates initially receiving negative weights and others much larger positive weights. This dispersion reduces sharply early in training and then remains relatively stable, while the mean of the Vector FTL parameters stays close to the scalar tuning parameter.

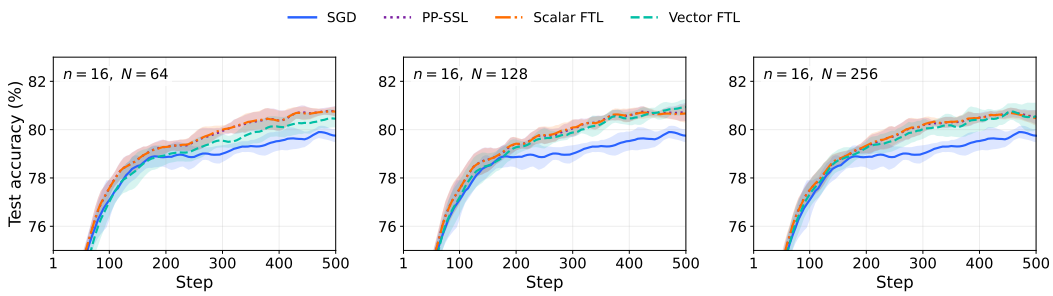


Figure 16: Test Accuracy (%) over training steps for the first-order methods on Satimage with $n = 16, N = 64$ (left), $n = 16, N = 128$ (middle), and $n = 16, N = 256$ (right). Values are reported every 25 epochs. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-MSE curves are smoothed using a five-step moving average.

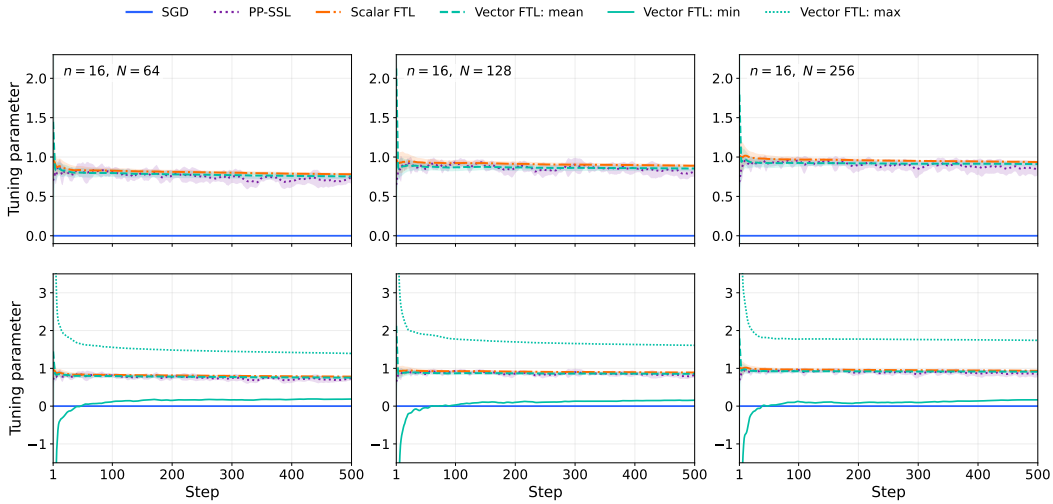


Figure 17: Tuning parameter trajectories for the first-order methods on Satimage with $n = 16, N = 64$ (left), $n = 16, N = 128$ (middle), and $n = 16, N = 256$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise Vector FTL parameters at each epoch. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, the tuning parameter curves are smoothed using a five-step moving average.

D.4.2 Zeroth-order methods

For the zeroth-order experiments, we train the linear classifier for 5,000 optimization steps and compare three methods: standard stochastic zeroth-order method without using ML-labeled data and our proposed Algorithm 2 with scalar tuning and vector tuning. We use a labeled batch size of $n = 32$ for all methods and consider unlabeled batch sizes $N \in \{128, 256, 512\}$ for our methods that use ML-labeled data. We set the perturbation parameter to $\mu = 1e\text{-}3$. We use a constant learning rate throughout training and independently tune it for each method and batch-size configuration over $\{1e-3, 5e-3, 1e-2\}$. Based on mean validation accuracy across seeds, a learning rate of $1e\text{-}2$ is selected for all methods and configurations in the final comparison.

Results. Figures 18 and 19 present the test-accuracy results and corresponding tuning-parameter trajectories for the zeroth-order methods on Satimage. Across all three unlabeled batch sizes, Algorithm 2 with either of the tuning schemes consistently outperform the ZO baseline without using ML-labeled data after the initial stages of training. ZO Scalar FTL and ZO Vector FTL achieve

nearly identical test accuracy throughout training, and their improvement over the baseline is sustained through the later optimization steps. This behavior remains consistent as the unlabeled batch size increases from $N = 128$ to $N = 512$. The corresponding tuning trajectories stabilize rapidly after the initial optimization steps, with the ZO Scalar FTL parameter and the mean of ZO Vector FTL parameter remaining closely aligned throughout training. As N increases for fixed n, the tuning parameters tend to stabilize at larger values, indicating that the methods place greater weight on the ML-labeled correction when it is estimated from a larger unlabeled batch. The coordinatewise ZO Vector FTL parameters initially span a substantially wider range, but this dispersion reduces quickly and continues to decrease over training. Despite this additional coordinatewise flexibility, the close agreement between the scalar and the mean of vector tuning parameters is consistent with the nearly identical predictive performance of the two methods

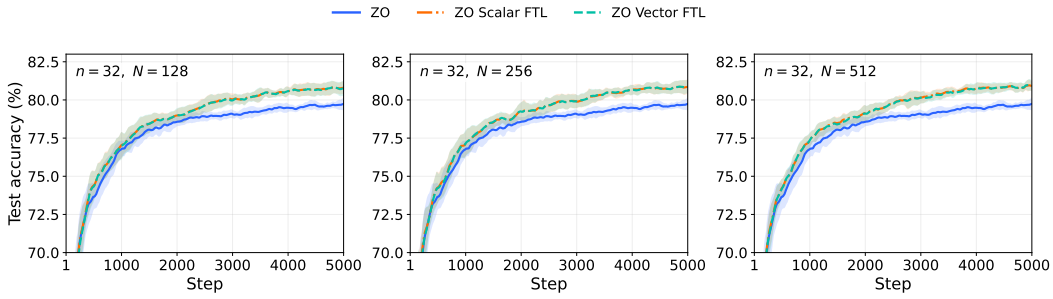


Figure 18: Test Accuracy (%) over training steps for the zeroth-order methods on Satimage with $n = 32, N = 128$ (left), $n = 16, N = 256$ (middle), and $n = 16, N = 512$ (right). Values are reported every 25 epochs. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, test-MSE curves are smoothed using a ten-step moving average.

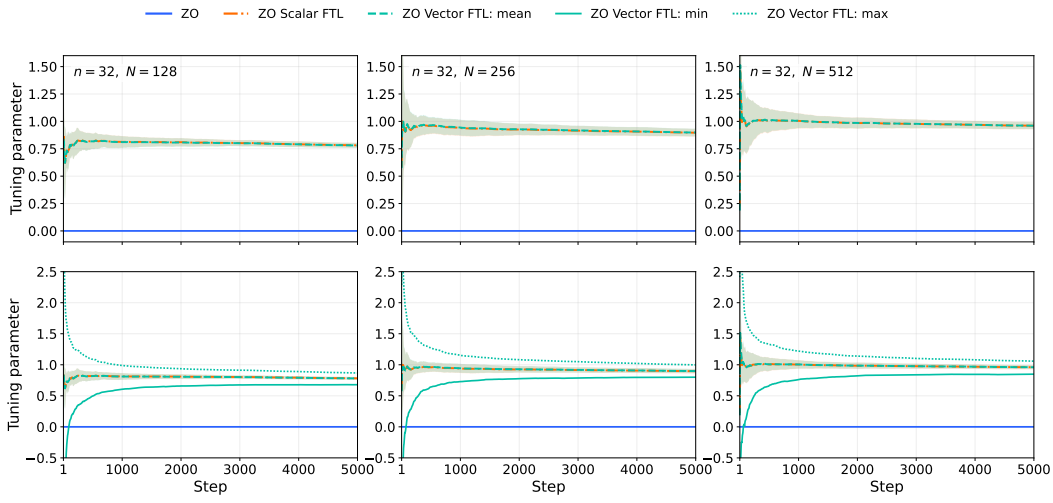


Figure 19: Tuning parameter trajectories for the first-order methods on Satimage with $n = 32, N = 128$ (left), $n = 16, N = 256$ (middle), and $n = 16, N = 512$ (right). **Top row**: Tuning parameter trajectories over training epochs. **Bottom row**: Tuning parameter trajectories together with the across-seed mean of the minimum and maximum coordinatewise ZO Vector FTL parameters at each epoch. Curves show the mean across ten random seeds, with shaded regions indicating ± 1 standard deviation. For visualization, the tuning parameter curves are smoothed using a ten-step moving average.