Beyond Performative Prediction: Open-environment Learning with Presence of Corruptions

Jia-Wei Shan, Peng Zhao, Zhi-Hua Zhou
National Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing 210023, China
{shanjw, zhaop, zhouzh}@lamda.nju.edu.cn

Abstract

Performative prediction is a framework to capture the endogenous distribution changes resulting from the reactions of deployed environments to the learner's decision. Existing results require that the collected data are sampled from the clean observed distribution. However, this is often not the case in real-world applications, and even worse, data collected in open environments may include corruption due to various undesirable factors. In this paper, we study the entanglement of endogenous distribution change and corruption in open environments, where data are obtained from a corrupted decision-dependent distribution. The central challenge in this problem is the entangling effects between changing distributions and corruptions, which impede the use of effective gradient-based updates. To overcome this difficulty, we propose a novel recursive formula that decouples the two sources of effects, which allows us to further exploit suitable techniques for handling two decoupled effects and obtaining favorable guarantees. Theoretically, we prove that our proposed algorithm converges to the desired solution under corrupted observations, and simultaneously it can retain a competitive rate in the uncorrupted case. Experimental results also support our theoretical findings.

1 INTRODUCTION

Distribution change is one of the fundamental challenges in modern machine learning, which has drawn increasing attention in recent years from both empirical and theoretical aspects [Sugiyama and Kawanabe, 2012, Bengio et al., 2021], and is one of the key requirements towards open-environment machine learning [Zhou, 2022]. A variety of factors can affect the data distribution. On the one hand, data distribution *exogenously* evolves due to the non-stationarity exhibited in the data collected environments. On the other hand, the data-generating distribution may also evolve *endogenously* due to the interactions between the learner and the environments. Besides, it is also particularly often the case in real-world applications that the collected data can be *corrupted* due to unexpected accidents. Take traffic prediction as a real-world example: In a travel, tourists may decide the route they take referring to the results of traffic prediction models. On the one hand, the estimated time of arrival may change as a result of exogenous environment change such as weather change. On the other hand, the observed data distribution may also change due to endogenous reasons caused by the decision system itself: if a route is predicted to have a good traffic, more people may decide to take the route, then the mostly-selected route will actually appear to be congested. Besides, the GPS signal sometimes gets lost by accident, leading to extremely erroneous predictions.

Despite numerous progresses have been achieved in dealing with exogenous distribution change via the non-stationary online learning framework [Hazan and Seshadhri, 2009, Daniely et al., 2015, Besbes et al., 2015, Zhang et al., 2018, Zhao et al., 2020, 2021, 2022], the endogenous distribution change is generally less explored. A considerable recent advance called *performative prediction* (or sometimes called *decision-dependent learning*) was proposed by Perdomo et al. [2020], with the endogenous distribution change modeled as a decision-dependent mapping $\mathcal{D}(x)$ from model parameters to data distributions. Those studies offer a simple abstraction for decision-theoretic learning, with a natural assumption which states that the endogenous distribution change is mild with respect to the update of model parameters [Perdomo et al., 2020, Mendler-Dünner et al., 2020]. By such, the formulation shares many benign properties with the problems having unchanged data distributions, so it is possible to adapt

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

the algorithms designed for stationary problems to solve those decision-dependent optimizations [Mendler-Dünner et al., 2020, Drusvyatskiy and Xiao, 2022].

However, performative prediction is far from satisfaction in real-world open environments due to the extensively existence of other sources that affect the data distributions, especially the entanglement of corruptions. Dealing with corruptions is challenging from a theoretical view [Tukey, 1960, Diakonikolas et al., 2021], as corruptions are significantly different from the clean data such that even a tiny fraction of corruptions sharply affect the true data distribution. Furthermore, corruptions can be even more challenging to handle when it is entangled with endogenous distribution change, since the pernicious composition of both factors severely ruins the nice properties of the problem and leads to failure of existed methods. To this end, it is definitely important to design robust open-environment learning methods to deal with endogenous distribution change with presence of corruptions.

In this paper, we study the problem described above, which is formulated as a time-varying stochastic optimization over a distribution consisting of a decision-dependent data distribution and a small fraction of unknown corruptions. The central challenge comes from the entanglement between endogenous distribution change and corruptions, which obstructs the attempts of using effective gradient-based updates. To overcome this difficulty, we present a novel *recursive formula* that disentangles the two sources of effects. In light of that, we can further use robust estimation as a means of resolving the problem of corruptions, and then derive a novel gradient-based algorithm with robust estimation as a sub-routine such that the algorithm updates by the robustly-estimated gradients rather than the observed stochastic gradients. We thus achieve the *first* provably convergent algorithm to deal with endogenous distribution change with presence of corruptions. Moreover, when there is no corruption (which the learner is unaware of), it gracefully achieve the same convergence rate compared to algorithms specifically designed for uncorrupted decision-dependent learning scenarios, up to logarithmic factors in the number of iterations. We also conduct numerical experiments to support our theoretical findings.

We finally highlight our technical innovation. Due to the *entanglement* of endogenous distribution change and corruptions as well as the fact that their effects on the dynamics of gradient-based methods are completely *different*, existing frameworks of performative prediction fail to handle them simultaneously. We break this obstruction by presenting a novel *recursive formula* for the analysis, showing that the effects of endogenous distribution change and corruptions can be *disentangled*. Based on this observation, we are able to adopt recent advances of robust estimation to construct a gradient estimator provably robust to arbitrary corruptions, and then design gradient-based algorithm with convergence guarantees. Our result also enables the usage of other robust estimators and gradient-based optimization techniques, which indicates the generality of our approach.

The rest of the paper is organized as follows. We discuss related work in Section 2, and formulate the problem in Section 3. Our main results are presented in Section 4, with discussions in Section 5. Section 6 reports the empirical results. We finally conclude the paper in Section 7. Due to the page limits, we defer all the proofs to the appendices.

2 RELATED WORK

Our work fits in the broader literature of open-environment machine learning (open ML) [Zhou, 2022], in which developing algorithms robust to distribution changes is one of the key requirements. As our paper focuses on the entangling effects of endogenous distribution changes and corruptions, in the following, we briefly discuss the closely related topics, including performative prediction and learning in the presence of corruptions.

Performative prediction. The performative prediction framework introduced by Perdomo et al. [2020] models an endogenous reason of distribution change, where the data distribution may depend on the current predictive model. They also propose algorithms that converge to *stable points*, which exhibits the equilibrium for performative prediction and thus becomes crucial and desirable. Mendler-Dünner et al. [2020] first prove the convergence of stochastic gradient methods, and Drusvyatskiy and Xiao [2022] subsequently show that a variety of popular gradient-based algorithms in the performative prediction setting can be viewed as solving a stochastic optimization problem with a biased gradient oracle. The relationship between stable points and the optimal solution is studied by Perdomo et al. [2020], and some works explore the conditions under which we can compute such optimal points directly [Miller et al., 2021, Izzo et al., 2021]. There are also explorations on the phenomenon of performativity in different scenarios such as multi-player games [Narang et al., 2022], bandits [Jagadeesan et al., 2022] and state-dependent learning [Brown et al., 2022, Li and Wai, 2022a, Izzo et al., 2022, Mandal et al., 2022]. These works on performative prediction set up an idealized abstraction to study a single source of distribution change, while our current work takes a step beyond and achieves a novel method to handle an instance of entangled factors of distribution change in open environment.

Learning in the presence of corruptions. Learning in the presence of corruptions is a ubiquitous challenging problem in machine learning. The central step in dealing with the problem is to obtain a robust estimation, which has been systematically studied since the pioneering work of Tukey [Tukey, 1960]. While many popular robust estimators has been proposed [Tukey, 1975, Fischler and

Bolles, 1981, Huber, 1981, Hampel et al., 1986, Breunig et al., 2000, Owen, 2007], they either need intractable computation time or can tolerate a very limited fraction of noise in high-dimensional datasets due to their exponential dependency on data dimension. Two celebrated recent advances in this topic independently propose different polynomial-time algorithms for the problem of robust estimation in high dimension [Diakonikolas et al., 2016, Lai et al., 2016], and many follow-up works employ the frameworks to tackle the problem of robust estimation in specific scenarios [Yin et al., 2018, Lykouris et al., 2018, Cheng et al., 2018, Liu et al., 2021, Bakshi et al., 2022].

Among the various applications of the recently proposed frameworks, the ones that mostly related to our work is in a line of research that leverage robust statistics to develop optimization methods which is robust to arbitrary corruptions. Diakonikolas et al. [2019] propose a meta algorithm for stochastic optimization that can convert a black-box stochastic optimization algorithm into a robust algorithm by alternatively invoking the black-box algorithm and applying the robust estimation method to detect and filter the corrupted fraction of data. Prasad et al. [2020] propose a robust gradient descent for the problem of offline optimization, which evenly allocates the samples in hand to each iteration, and uses a blackbox estimator to calculate gradient estimations of each round that are robust to corruptions. Although sharing several ideas with Prasad et al. [2020], our work is significantly different from theirs for the reason that they propose a passive algorithm which is robust to corruptions for offline static optimization problems, and in this paper we design a robust algorithm that interactively runs in a more realistic open-environment setting.

3 PROBLEM FORMULATION

In this section, we present our problem formulation for open-environment learning with presence of corruptions.

Environment model. Denote $\mathcal{Z}$ as the sample space and $\mathcal{X} \subseteq \mathbb{R}^d$ a convex set of model parameters. We formulate the problem *open-environment learning with presence of corruptions* as a T-round interactive protocol. Before the interactions, the learner specifies a loss function $\ell(z; x)$ to evaluate the performance of the deployed model x when working on data point z. At interaction round $t \in [T]$, the learner first *collects* some data $Z_t = \{z_{t,1}, \ldots, z_{t,n_t}\}$ i.i.d. sampled from current data distribution $\mathcal{P}_t$, then uses the information observed so far to obtain a new model x_{t+1}, and *deploys* the model into the environment.

The key ingredient to model the pattern of distribution change of our problem is a sequence of data distributions $\{\mathcal{P}_t\}_{t=1}^T$. We assume that the distributions have the form

$$\mathcal{P}_t = (1-\epsilon)\mathcal{D}(x_t) + \epsilon\mathcal{Q},$$

in which ϵ denotes the proportion of corruptions, $\mathcal{D}(\cdot)$ is a decision-dependent mapping from model parameters to data distributions that characterizes the strength of the endogeneity of the environment, and $\mathcal{Q}$ is an *arbitrary* distribution served as corruptions on observed data distribution.

Performative risk. When a model x is deployed to the environment, the data distribution evolves as a reaction to the deployment. Therefore, the performance of the deployed model x should be evaluated on the uncorrupted distribution $\mathcal{D}(x)$ induced by the deployed model itself, resulting in the following notion called *performative risk*:

$$\mathrm{PR}(x) \stackrel{\text{def}}{=} \mathbb{E}_{z\sim\mathcal{D}(x)}\left[\ell(z; x)\right] .$$

However, the performative risk proposed above is generally hard to optimize since it is generally non-convex, unless making strong assumptions on the distribution mapping [Miller et al., 2021, Izzo et al., 2021].

Remark 1. Note that the risk focuses on $\mathcal{D}(x)$ and rules out the corruptions $\mathcal{Q}$, since we shouldn't blame the algorithm when faults essentially come from the environment. Again take the task of traffic prediction introduced in Section 1 as an example, we would not expect our algorithm to perform well when the GPS signal accidentally lost. Therefore, the proposed risk is natural in real-world applications.

Algorithmic measure. As the minimizer of the performative risk is generally hard to find, we seek to obtain an alternative solution. An appealing solution in the decision-dependent setting is the one that achieves minimal risk on the distribution induced by itself, which leads to the following definition of *performatively stable point* [Perdomo et al., 2020, Mendler-Dünner et al., 2020]:

Definition 1 (Performatively stable point)**.** A point $\bar{x}$ is said to be a *performatively stable point* if

$$\bar{x} \in \arg\min_{x\in\mathcal{X}} \mathbb{E}_{z\sim\mathcal{D}(\bar{x})}\left[\ell(z; x)\right] .$$

It is natural to aim at finding performatively stable points: Once a stable point is deployed, we have no reason to update the model any more, since the collected data already indicates the optimality of current model. Performatively stable point is sure to be uniquely exist under fairly weak assumptions [Perdomo et al., 2020], and the goal of the learner is to find such stable point rather than achieving performatively optimal ones. Prevalent performance metrics in the decision-dependent optimization literature includes *optimizer distance* $\|x_T - \bar{x}\|$ and *function-value distance* $\mathrm{PR}(x_T) - \mathrm{PR}(\bar{x})$, and we focus on the optimizer distance $\|x_T - \bar{x}\|_2$ equipped with the Euclidean norm.

Remark 2. The formulation subsumes the problem formulation of statistical risk minimization and the performative prediction. Specifically, we can recover the performative prediction problem by setting the level of corruptions

$\epsilon = 0$, and can further recover statistical risk minimization problem by letting $\mathcal{D}(\cdot)$ to be the constant mapping.

We finally introduce a couple of assumptions on the loss function and the distribution mapping:

Assumption 1 (Smoothness)**.** The loss $\ell(z; x)$ is L-smooth in x for all $z \in \mathcal{Z}$, and the map $z \mapsto \nabla_x \ell(z; x)$ is β-Lipschitz continuous for all $x \in \mathcal{X}$.

Assumption 2 (Strong convexity)**.** The loss $\ell(z; x)$ is α-strongly convex in x for all $z \in \mathcal{Z}$.

Remark 3. Assumptions 1 and 2 are common in the context of decision-dependent optimization and its applications [Perdomo et al., 2020, Drusvyatskiy and Xiao, 2022, Miller et al., 2021, Narang et al., 2022, Li and Wai, 2022b], and the assumptions hold for most of the commonly-used loss functions including the quadratic loss, the regularized logistic loss, and many other losses with bounded values.

Assumption 3 (Lipschitz distribution)**.** There exists $\gamma > 0$ such that $\forall\, x, y \in \mathcal{X}$,

$$W_1(\mathcal{D}(x), \mathcal{D}(y)) \leq \gamma \cdot \|x - y\|_2 ,$$

where

$$W_1(\mathcal{P}, \mathcal{P}') = \sup_{g \in \mathrm{Lip}_1} \{\mathbb{E}_{z\sim\mathcal{P}}[g(z)] - \mathbb{E}_{z'\sim\mathcal{P}'}[g(z')]\}$$

is the Wasserstein-1 distance.

Remark 4. Assumption 3 is critical yet natural. On the one hand, it is obvious that one cannot obtain any guarantee on the convergence of any algorithm without making regularity assumptions on distribution mapping $\mathcal{D}(\cdot)$. On the other hand, it is reasonable to quantify the regularity of distribution mapping by assuming Lipschitz continuity in the decision-dependent setting, due to the intuition that deploying similar models results in similar observations [Perdomo et al., 2020, Mendler-Dünner et al., 2020].

4 ALGORITHM AND CONVERGENCE GUARANTEES

In this section, we first give an in-depth analysis on the central challenge of the problem, then present our algorithm for the formulated problem with its theoretical guarantees.

4.1 Challenge and Disentangled Recursive Formula

In the rest of the paper, let $f_y(x) = \mathbb{E}_{z\sim\mathcal{D}(y)}[\ell(z; x)]$ denote the expected performance of model x evaluated on the distribution induced by an alternative model y. We always use the notion $\nabla\ell(z; x)$ to denote taking gradient with respect to the model parameters x, then we have $\nabla f_y(x) = \mathbb{E}_{z\sim\mathcal{D}(y)}[\nabla\ell(z; x)]$, and specifically, $\nabla f_x(x)$ is the gradient of function $x \mapsto f_x(w)$ evaluated at $w = x$. Note that $f_y(x)$ is smooth and strongly convex on x as immediate consequences of Assumptions 1 and 2.

At round t, the learner first collects a sample $Z_t = \{z_t^i\}_{i=1}^{n_t}$ from current data distribution $\mathcal{P}_t = (1-\epsilon)\mathcal{D}(x_t) + \epsilon\mathcal{Q}$, then compute the gradients $S_t = \{\nabla\ell(z_t^i; x_t)\}_{i=1}^{n_t}$. Denote $\widetilde{g}_t$ as the gradient estimation according to S_t, we focus on the learner's update rules that have the form

$$x_{t+1} = \arg\min_{x\in\mathcal{X}} \ \langle \widetilde{g}_t, x\rangle + \frac{1}{2\eta_t}\|x - x_t\|_2^2 , \qquad (\star)$$

where η_t is the step size. The goal of the learner is to output a model x_{T+1} after a sequence of interactions, which minimizes the distance between x_{T+1} and the performatively stable model $\bar{x}$ in the sense of the Euclidean norm.

The central challenge in designing algorithms with convergence guarantee is to deal with the entanglement of endogenous distribution change and corruptions, which creates extra obstacles beyond the respective difficulties of handling endogenous distribution change and corruptions: On the one hand, it is generally hard to handle endogenous distribution change under corruptions since even a tiny proportion of corruptions sharply exaggerate the intensity of endogenous distribution change. On the other hand, it is also difficult to deal with corruptions in the decision-dependent setting, as the persistent deviation of observed distributions due to endogenous distribution change flaws the usage of robust estimation which requires unbiased observations on uncorrupted distributions. Therefore, it is necessary to work out the true pattern of the entanglement of two effects at the first step.

Moreover, it is technically non-trivial to derive controllable quantities for both effects simultaneously, due to their significantly different characterizations: The endogenous distribution change is characterized intuitively by the included angle between the observed gradient at current distribution and the gradient at the distribution induced by stable point, and is controlled by a small multiplicative factor related to step size [Mendler-Dünner et al., 2020, Drusvyatskiy and Xiao, 2022]. The effect of corruptions is usually quantified via the level of corruptions, and can be partially dissolved by algorithmic efforts which result in a concentration to the uncorrupted mean with high probability [Diakonikolas et al., 2016, Lai et al., 2016]. Therefore, the technical difficulty in our problem is to seek for quantities that (1) are sufficient to characterize the disparate effects of endogenous distribution change and corruptions, and (2) are able to be controlled within theoretical and algorithmic efforts.

Fortunately, we prove a novel recursive formula that meets both requirements identified above, and even better, our key lemma *disentangles* the two sources of effects to some extent, thus alleviate subsequent algorithmic demands to obtain convergent results. The result is summarized in the following lemma, whose proof is deferred in Appendix B.

Lemma 1 (Disentangled recursive formula)**.** *Suppose Assumptions 1,2 and 3 hold, the updates ($\star$) are applied with*

gradient sequence $\{\widetilde{g}_t\}$*, and the step-size sequence* $\{\eta_t\}$ *satisfies* $\eta_t < 1/(2L)$*. Then for any* $\lambda_t > 0$*, the iterates* $\{x_t\}$ *generated by the updates* ($\star$) *satisfy*

$$
\begin{aligned}
&(1+\alpha\eta_t)\left\|x_{t+1}-\bar{x}\right\|_2^2 \\
&\leq \left(1+\frac{\eta_t}{\lambda_t}-\alpha\eta_t(1-2\rho)\right)\left\|x_t-\bar{x}\right\|_2^2 \qquad \text{(a)} \\
&+\left(\eta_t\lambda_t+\frac{2\eta_t^2}{1-\eta_t L}\right)\left\|\widetilde{g}_t-\nabla f_{x_t}(x_t)\right\|_2^2 \qquad \text{(b)} \\
&+\frac{2\eta_t^2}{1-\eta_t L}\left\|\nabla f_{x_t}(x_t)-\nabla f_{\bar{x}}(x_t)\right\|_2^2 , \qquad \text{(c)}
\end{aligned}
$$

where $\rho = \frac{\beta\gamma}{\alpha}$ *quantifies the strength of endogenous distribution change to some extent.*

Generally speaking, Lemma 1 tells that the deviation of iterate x_{t+1} can be recursively bounded by three terms:

(a) the deviation of the latest iterate x_t multiplies some coefficients related to the strength of endogenous distribution change,

(b) the bias of gradient estimation due to stochastic noise and the existence of corruptions, and

(c) the bias of the observed gradient comparing to the true gradient due to endogenous distribution change.

We can see that the effect of endogenous distribution change and corruptions are now *disentangled* in our Lemma 1, except for an implicit variable λ_t that can be adaptively adjusted in further analysis and the step size η_t that can be carefully tuned. In light of the presented lemma, in order to obtain a convergent algorithm, it suffices to control the bias of gradient estimation in the presence of corruptions in term (b), and then properly set the step-size sequence to balance the effect of endogenous distribution change and unresolved corruptions.

However, the upper bound of gradient estimation is impossible to obtain without extra algorithmic efforts: Due to the arbitrariness of corruptions, the gradient-based method probably diverges if the observations include corruptions at *any* iteration. Since the level of corruption is ϵ, the probability that no corruption is observed during an optimization process with T iterations is only $(1-\epsilon)^{\sum_{t=1}^{T} n_t}$, exponentially decaying with respect to the number of observations, which is far from satisfaction. Therefore, we seek for an estimation $\widetilde{g}_t$ from a set of corrupted observations

$$
S_t = \{\nabla\ell(z_t^1; x_t), \nabla\ell(z_t^2; x_t), \cdots, \nabla\ell(z_t^{n_t}; x_t)\}
$$

where $z_t^i \sim \mathcal{P}_t = (1-\epsilon)\mathcal{D}(x_t) + \epsilon\mathcal{Q}$, which is robust to *arbitrary* kinds of corruptions $\mathcal{Q}$.

4.2 Robust Gradient Estimator

The desired gradient estimator in our solution should have three properties: First, it should be robust against arbitrary corruptions due to the random nature of corruptions in reality. Second, it should be computationally efficient in high dimension due to the practical demand in machine learning literature. Third, it is better to make as relaxed assumptions on uncorrupted distributions as possible, due to a natural expectation of the algorithm to solve as wide a range of problems as possible. We investigate a broad range of robust estimators to seek for the one that is the most suitable to meet our demands.

While extensive studies have shown that median-based robust estimators are effective and efficient in low dimension, they either suffer a high failure probability or are computationally intractable in high-dimensional tasks [Tukey, 1975, Rousseeuw, 1985, Diakonikolas et al., 2021]. Recent advances have design computationally efficient algorithms for mean estimation under the Huber's contamination model [Diakonikolas et al., 2016, Lai et al., 2016], with further elaborations into the scenario of stochastic optimization, which is more relevant to our work [Diakonikolas et al., 2019, Prasad et al., 2020].

Indeed, both works of Prasad et al. [2020] and Diakonikolas et al. [2019] enjoy favorable theoretical guarantees, but the work of Prasad et al. [2020] brings us more inspiration. In decision-dependent literature, the gradient in current round is available only after deployment. The work of Prasad et al. [2020] can serve as a plug-in robust estimator at every round, while the one of Diakonikolas et al. [2019] requires the data to be collected in advance, which does not meet this particular requirement. As a result, we propose Algorithm 1 for gradient estimation task by adopting the robust estimation techniques of Prasad et al. [2020].

The proposed algorithm builds upon the fact that one-dimensional robust estimation is relatively easy, and the insight of Prasad et al. [2020] showing that the main effect of the corruptions are actually low-dimensional, and is exactly lying in the subspace attached with the largest singular values of the covariance matrix. Based on such observations, the algorithm firstly removes the gross corrupted examples with exceptionally large norms, then apply singular value decomposition to identify the directions in which the corruptions take large effect, and recursively invoke the same process to the subspaces that span by the top directions, while using a simple mean estimator in the orthogonal subspaces that span by the remaining directions. Before presenting the theoretical results of Algorithm 1, we introduce the following moment assumptions on the uncorrupted distributions $\mathcal{D}(x)$:

Assumption 4 (Bounded variance)**.** The random variable $G = \nabla\ell(z; x)$ where $z \sim \mathcal{D}(x)$ have uniformly bounded second moments. In other words, there exists a constant σ^2 such that for all $x \in \mathcal{X}$,

$$
\mathbb{E}_{z\sim\mathcal{D}(x)}\left\|\nabla\ell(z;x)-\nabla f_x(x)\right\|_2^2 \leq \sigma^2 .
$$

Assumption 5 (Bounded fourth moment)**.** The random

Algorithm 1: GRADIENTESTIMATOR

Input : gradient sample $S = \{\nabla \ell(z_i; x)\}_{i=1}^n$, level of corruption ϵ, dimension d, confidence δ

for $i = 1$ *to* d **do**
  Let S_i be the samples with only the i-th coordinates;
  Let c_i be the median of S_i;
end
Let B be the smallest ball centered at $(c_i)_{i=1}^d$ that contains $(1-\epsilon)^2$ fraction of S;
Let $\widetilde{S} = S \cap B$;
if $d = 1$ **then**
  return MEAN($\widetilde{S}$)
end
Let Σ be the covariance matrix of $\widetilde{S}$;
Let V be the span of the top $d/2$ principal components of Σ and W be its complement;
Let S_V be the projection of $\widetilde{S}$ onto V;
Let S_W be the projection of $\widetilde{S}$ onto W;
Let $\widehat{\mu}_V$ = GRADIENTESTIMATOR$(S_V, \epsilon, d/2, \delta)$;
Let $\widehat{\mu}_W$ = MEAN(S_W);
return $\widehat{\mu} = (\widehat{\mu}_V, \widehat{\mu}_W)$

variable $G = \nabla \ell(z; x)$ where $z \sim \mathcal{D}(x)$ have uniformly bounded fourth moments. In other words, there exists a constant C_4 such that the following condition

$$\mathbb{E}_{z\sim\mathcal{D}(x)}\left[\langle \nabla \ell(z;x) - \nabla f_x(x), v\rangle^4\right] \le C_4\, \mathbb{E}_{z\sim\mathcal{D}(x)}\left[\langle \nabla \ell(z;x) - \nabla f_x(x), v\rangle^2\right]^2$$

holds uniformly for all unit vector v and $x \in \mathcal{X}$.

Remark 5. The assumptions on the uncorrupted distributions $\mathcal{D}(x)$ are reasonable. Specifically, Assumption 4 is common in the context of stochastic optimization, and Assumption 4 and 5 are in fact weaker assumptions comparing to other common assumptions in this literature [Cutler et al., 2021]. For example, the family of sub-Gaussian distributions all satisfy our moment assumptions.

The following restatement of an important result in Prasad et al. [2020] shows that with appropriate hyperparameters, the output of Algorithm 1 is close to the true gradient mean in the sense of Euclidean norm with high probability.

Lemma 2 (Lemma 1 of Prasad et al. [2020])**.** *Suppose Assumption 4 and 5 hold for $\mathcal{D}(x)$, then there exists a positive constant $C > 0$, such that given $S = \{\nabla \ell(z_i; x)\}_{i=1}^n$ where $\{z_i\}_{i=1}^n$ are i.i.d. sampled from distribution $\mathcal{P} = (1-\epsilon)\mathcal{D}(x) + \epsilon\mathcal{Q}$, Algorithm 1 returns an estimate $\widetilde{g}$ such that with probability at least $1-\delta$,*

$$\left\|\widetilde{g} - \mathbb{E}_{z\sim\mathcal{D}(x)}[\nabla \ell(z;x)]\right\|_2 \le C\left(\sqrt{\epsilon} + \gamma(n,d,\delta,\epsilon)\right)\sigma\sqrt{\log d}\,,$$

where

$$\gamma(n,d,\delta,\epsilon) = \left(\frac{d\log d\log(n/(d\delta))}{n}\right)^{3/8} + \left(\frac{\epsilon d^2 \log d \log\left(\frac{d\log d}{\delta}\right)}{n}\right)^{1/4}.$$

The full proof of Lemma 2 is in [Prasad et al., 2020, Lemma 1] with different notations, and we provide a sketch in Appendix C. Generally speaking, with the help of Lemma 2, we can bound the error of gradient estimation in the sense of Euclidean norm given enough samples.

Moreover, Algorithm 1 is robust to misspecification, i.e., it remains effective even if we only know an upper bound of the true corruption ratio $\epsilon' \ge \epsilon$. Indeed, we can convert the underlying problem $(1-\epsilon)\mathcal{D}(x)+\epsilon\mathcal{Q}$ to $(1-\epsilon')\mathcal{D}(x)+\epsilon'\mathcal{Q}'$ with a new corrupted component $\mathcal{Q}' \triangleq (1-(\epsilon/\epsilon'))\mathcal{D}(x) + (\epsilon/\epsilon')\mathcal{Q}$. It is easy to verify that Algorithm 1 retains convergence when working against $\mathcal{Q}'$ given ϵ', at an expense of higher sample complexity (by replacing ϵ by ϵ' in the theoretical results) due to misspecification. We further conduct experiments to verify our claim in Section F.

Inspired by Lemma 1 and Lemma 2, we can first collect a batch of examples, then invoke Algorithm 1 to get an estimation with small deviation as in term (b) of Lemma 1.

4.3 Overall Algorithm

Now we are ready to propose the overall algorithm RPGD for open-environment learning with presence of corruptions. At each round t, our Algorithm 2 first collects a sample of size n_t from current distribution $\mathcal{P}_t = (1-\epsilon)\mathcal{D}(x_t)+\epsilon\mathcal{Q}$, then applies Algorithm 1 as a sub-routine to compute $\widetilde{g}_t$ as an estimation of current gradient mean, after that conducts gradient descent using the estimated gradient, and finally deploies the updated model x_{t+1} to the environment.

Algorithm 2 enjoys the following convergence guarantee, whose proof is in Appendix D. We will present in-depth discussions on the theorem in next section.

Theorem 3. *Suppose Assumptions 1, 2, 3, 4 and 5 hold, also suppose the level of corruption $\epsilon < 1/(C^2\log d)$ where C is the same constant as in Lemma 2, and we are in the regime $\rho = \beta\gamma/\alpha < 1$. Define $\widehat{\alpha} = \alpha - \beta\gamma$, set $\eta_t = \eta \le 1/\left(4\beta^2\gamma^2/\widehat{\alpha} + 2L\right)$ and set*

$$n_t = n = \frac{16d^2\log d}{A^4}\log\left(\frac{dT\log d}{A^{8/3}\delta}\right)$$

where $A = \frac{1}{2}\left(1/(C\sqrt{\log d}) - \sqrt{\epsilon}\right)$. Then we have with probability at least $1-\delta$,

$$\|x_{T+1} - \bar{x}\|_2^2 \le \left(1 - \frac{\widehat{\alpha}\eta}{3}\right)^T \|x_1 - \bar{x}\|_2^2 + \frac{2+8\eta}{\widehat{\alpha}}\sigma^2\,.$$

Algorithm 2: RPGD

Input : number of iterations T, step size sequence $\{\eta_t\}_{t=1}^T$, sample size sequence $\{n_t\}_{t=1}^T$, contaminated level ϵ, dimension d, confidence δ

Initialize x_1 and deploy x_1 to the environment;

for $t = 1$ *to* T **do**

Collect $Z_t = \{z_t^i\}_{i=1}^{n_t}$ from current $\mathcal{P}_t$;

Let $S_t = \left\{\nabla \ell(z_t^i; x_t)\right\}_{i=1}^{n_t}$;

Let $\widetilde{g}_t = \textsc{GradientEstimator}(S_t, \epsilon, d, \delta)$;

Let $x_{t+1} = \arg\min_{x \in \mathcal{X}} \langle \widetilde{g}_t, x \rangle + \frac{1}{2\eta_t} \|x - x_t\|_2^2$;

Deploy x_{t+1} to the environment;

end

return x_{T+1}

5 DISCUSSIONS

In this section, we discuss our theoretical results at length. In detail, we analyze the dependency of different important quantities that affect the performance of our algorithm, and compare the result to gradient-based algorithms designed for uncorrupted settings. Throughout the section, we use the notation $\widetilde{O}(\cdot)$ to ignore the lower order dependency of variables that appears inside the notation, and specifically, $\widetilde{O}(\log T)$ implies that the dependency on the number of deployments T is no more than $O(\text{polylog}(T))$.

Convergence rate. We first analyze the convergence rate of the algorithm with respect to the number of deployments T. The following corollary is a direct consequence with suitable step-size tuning on Theorem 3, whose proof can be found in Appendix E.

Corollary 4. There exists step size η such that the following holds with probability at least $1 - \delta$:

$$\|x_{T+1} - \bar{x}\|_2^2 \leq \frac{2\sigma^2}{\widehat{\alpha}} + O\left(\frac{\log T}{T}\right) .$$

Corollary 4 tells that the deviation of the last iterate x_{T+1} converges to some *unavoidable* constant $2\sigma^2/\widehat{\alpha}$ in the sense of Euclidean norm at rate $O(\log T/T)$. The unavoidable constant comes from the malignant property of corruptions which leads to a biased gradient estimation, thus cannot be removed by decaying step size nor similar optimization techniques. Since the sample size at each round in Theorem 3 is set to be $O(\log T)$, we deduce that our algorithm achieves $\widetilde{O}(1/N)$ convergence rate provided that a total number of N samples has been collected, up to some logarithmic factors.

Comparison to uncorrupted setting. Stochastic gradient methods have been extensively explored in the context of performative prediction [Mendler-Dünner et al., 2020, Drusvyatskiy and Xiao, 2022], and the main finding is that the iterates converge to the stable point at a rate of $O(1/T)$ in expectation. We give the following delicate comparison between their results and the one proposed in our work.

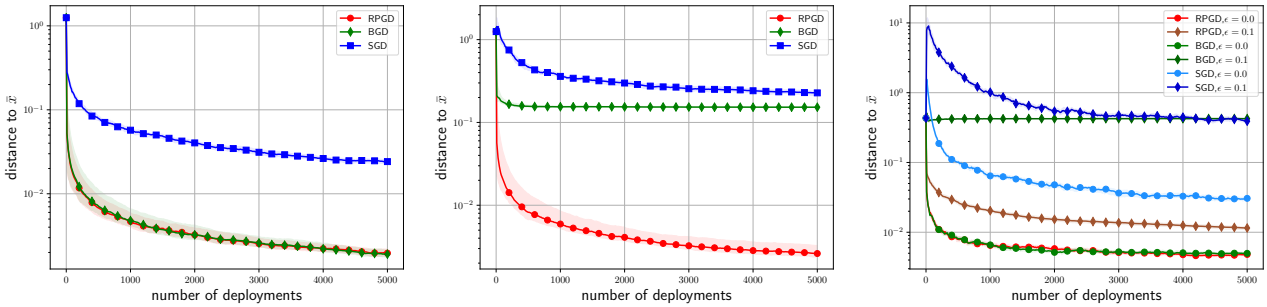
On the one hand, the two bounds are not directly comparable because they lie in different setting and rely on different assumptions. Besides, our method exhibit a different type of convergence: we show in Corollary 4 that the iterates of our method *concentrates* into a small neighbourhood of stable point *with high probability*, while Corollary 7.4 of Drusvyatskiy and Xiao [2022] tells that the iterates of stochastic gradient method *converges* to stable point *in expectation*. On the other hand, the main advantage of our method is that it still works well in uncorrupted setting even if we are unaware of the absence of corruptions, while stochastic gradient methods fails in corrupted problems, which will be empirically verified in next section.

Dependency on d and ϵ. Now we discuss the dependency of sample complexity n_t on the dimension d of the model parameters, and the level of corruptions ϵ. Our Theorem 3 requires $n_t = \widetilde{O}(d^2 A^{-4})$ where $A = \frac{1}{2}\left(1/(C\sqrt{\log d}) - \sqrt{\epsilon}\right)$ is the gap between the level of corruptions and the maximal tolerance of corruptions of the algorithm. We remark that the sample complexity of our algorithm have a low dependency on parameters d and ϵ.

Efficiency. We finally analyze the time complexity of our algorithm with respect to data dimension d and the number of deployments T. Since the algorithm contains a same estimation process (Algorithm 1) applying on $n_t = n = \widetilde{O}(d^2)$ samples and a step of gradient descent that can be efficiently conducted at each round, we only analyze the time complexity of the gradient estimator, and the time complexity of the overall algorithm multiplies by the number of deployments T. The gradient estimation process is recursively executed on the feature spaces with sequentially halving dimensions, and each execution consists of two sub-routines: filtration and eigenvalue decomposition. The filtering operation can be implemented via sortings in each dimension, yielding an $O(n_t \log n_t) = \widetilde{O}(d^2 \log T)$ time complexity, and the cost of the eigenvalue decomposition of a $d \times d$ matrix is well-known to be $O(d^3)$. Therefore, the time complexity of each call of Algorithm 1 is $\widetilde{O}(d^3 \log T)$, which proves that our Algorithm 2 is efficient with respect to data dimension d and the number of deployments T.

6 EXPERIMENTS

We examine the effectiveness of our proposed algorithm and complement our theoretical findings by empirical evaluations. Specifically, we conduct empirical studied on synthetic data to verify that our method: (1) has a competitive performance in absence of corruptions, and (2) concentrates to a neighborhood of stable point while classi-



(a) Mean estimation, $\epsilon = 0.0$ (b) Mean estimation, $\epsilon = 0.1$ (c) Strategic classification

Figure 1: Error versus number of deployments for our algorithm and two benchmark algorithms BGD and SGD on synthetic problems. Specifically, (a) and (b) are the results of mean estimation with $\mu = \mathbf{1}_d/\sqrt{d}$ and $\Sigma = 0.1I_d$, and (c) is the result of regularized logistic regression problem with $\mu = \mathbf{1}_d/\sqrt{d}$ and $\Sigma = 25I_d$, where $\mathbf{1}_d$ and I_d respectively denote all one's vector and identity matrix. Each experiment is repeated 20 times and we display 95% bootstrap confidence intervals.

cal gradient-based methods do not. For simplicity, we focus on the unconstrained problems, in which our gradient-based update formula is identical to a step of gradient descent using $\widetilde{g}_t$. We compare our algorithm and two popular gradient-based algorithms in performative prediction, whose update formulas are listed below:

- RPGD (ours):

$$x_{t+1} = x_t - \eta_t \widetilde{g}_t;$$

- Stochastic gradient descent (SGD):

$$x_{t+1} = x_t - \eta_t g_t;$$

- Batched gradient descent (BGD):

$$x_{t+1} = x_t - \frac{\eta_t}{n_t} \sum_{i=1}^{n_t} g_t^i.$$

Specifically, $\widetilde{g}_t = \textsc{GradientEstimator}(S_t, \epsilon, d, \delta)$ as is declared in Algorithm 2, $g_t^i = \nabla(z_t^i; x_t) \in S_t$ and g_t is a randomly drawed stochastic gradient in S_t.

The experiments are conducted on two different problems with various levels of corruptions ϵ. The step size sequences of all methods are identically set to be $\eta_t = 1/(\gamma t + 8/\gamma)$ according to [Mendler-Dünner et al., 2020, Theorem 3.2], which also satisfies the condition in our Theorem 3. We set the sample size collected by our method at each round according to Lemma 2 to ensure deviation of gradient estimation at each round at most σ with probability at least $1 - \delta/T$, and we choose $\delta = 0.8$ in our method. We only present some of the experimental results due to space limit. More results can be found in Appendix F for similar phenomenon observed in more severe environments, including larger levels of corruptions, heavy-tailed corruptions and the existence of misspecification.

6.1 Mean Estimation

Mean estimation on a decision-dependent Gaussian distribution is a commonly-used benchmark problem in the performative prediction literature, with adaptations in different applications [Mendler-Dünner et al., 2020, Li and Wai, 2022b]. Consider the problem of mean estimation on a corrupted d-dimension Gaussian distribution that evolves with respect to the change of the learner's estimation, and the aim of the learner is to minimize the expected square loss $\ell(z; x) = \frac{1}{2}(z - x)^2$ where $z \sim \mathcal{D}(x) = \mathcal{N}(\mu + \gamma x, \Sigma)$. It is easy to verify that the strongly-convexity and smoothness coefficients are $\alpha = \beta = 1$ in this problem. Simple calculation shows that a unique performatively stable point exists as $\bar{x} = \frac{\mu}{1-\gamma}$ when $\gamma < 1$. We fix $\gamma = 0.2$ in experiment to ensure the existence of a unique stable point. We set $\mathcal{Q} = \mathcal{N}(\mathbf{0}_d, d^2\Sigma)$ to simulate the corruptions, where d is the dimension of Gaussian distribution and $\mathbf{0}_d$ and I_d denote the all zero's vector and the identity matrix. We choose $d = 20$ in our experiment.

We repeatedly run each algorithm for 20 times with identical initial conditions (and independent samples in each trial), and compute the error that measures the distance between iterations x_t of the algorithm and the performatively stable point $\bar{x}$. Figure 1a (for uncorrupted case) and Figure 1b (for corrupted case) demonstrate the average error among repetitions. The results show that our method successfully filter the proportion of corruptions and is convergent when $\epsilon > 0$, and it still retains a competitive performance in the uncorrupted case ($\epsilon = 0$). By contrast, SGD and BGD fail to converge to stable point when $\epsilon > 0$.

We also complement the experimental results on the task of mean estimation by conducting on more levels of corruptions, as well as introducing more severe environments incorporating heavy-tailed corruptions and misspecification, i.e., the input is an upper bound of the true ratio of corrup-

tions $\epsilon' \geq \epsilon$, and the results are reported in appendix (see Figure 2 and 4a respectively in Appendix F).

6.2 Strategic Classification

We further examine the performance of our algorithm in a strategic classification problem in which endogenous distribution change naturally takes place [Mendler-Dünner et al., 2020]. We first construct base distribution $(z_{\text{feat}}, z_{\text{label}}) \sim \mathcal{D}$ where the feature z_{feat} is sampled from some normal distribution $\mathcal{N}(\mu, \Sigma)$ and $z_{\text{label}} = \text{sign}(\langle \theta, z_{\text{feat}} \rangle)$. Then the distribution mapping $\mathcal{D}(x)$ is construct as first get a sample from base distribution, then shift the feature by $z'_{\text{feat}} = z_{\text{feat}} - \gamma x$. In other words, we get $z = (z'_{\text{feat}}, z_{\text{label}})$ every time when we sample from $\mathcal{D}(x)$. It is easy to prove that the constructed distribution mapping satisfies our Assumption 3 [Perdomo et al., 2020]. We set $\gamma = 0.1$ in our experiment. We choose the regularized logistic loss

$$\ell(z; x) = \log(1+\exp(\langle x, z'_{\text{feat}} \rangle)) - z_{\text{label}} \langle x, z'_{\text{feat}} \rangle + \frac{M}{2} \|x\|_2^2,$$

where M is the strength of regularization to ensure strong convexity, which is set to be 1.0. The performatively stable point is hard to be algebraically computed in this problem, so we approximately yield the stable point by running a long iteration of SGD on the uncorrupted problem.

We simulate the corruptions in this problem by firstly defining a feature distribution $\mathcal{N}(\mu, d^2\Sigma)$ with identical mean and increasing variance comparing to the uncorrupted distributions, and then set all the labels to be 1. Such asymmetric corruptions was introduced by Prasad et al. [2020].

Figure 1c demonstrates the mean of the error among 20 repetitions of experiment with various levels of corruptions ϵ. As is shown in Figure 1c, all methods converge to the stable point when $\epsilon = 0.0$, and our algorithm has almost the same convergence rate with BGD. By contrast, BGD and SGD fails to converge even in the presence of corruptions, while our proposed method retain convergence when the level of corruptions increases.

We also complement the experimental results on the task of strategic classification by conducting on more levels of corruptions, as well as introducing more severe environments incorporating heavy-tailed corruptions and misspecification, and the results are reported in appendix (see Figure 3 and Figure 4b respectively in Appendix F).

7 CONCLUSION

In this paper, we formulated the problem of open-environment learning with presence of corruptions. To resolve the entanglement of endogenous distribution change and corruptions, we proposed a novel decomposition of the recursive formula for the gradient-based method, based on which we developed RPGD algorithm to find the performatively stable points via suitable robust gradient estimators. We prove the convergence of our proposed method, showing that the effect of corruptions can be almost entirely dissolved given a small number of observations per round. Empirical results also validate our proposal.

There are a number of interesting directions for future work. First, it is worth investigating the convergent property of gradient-based algorithms under other performance metrics such as the function-value distance, and we also left open the possibility of developing algorithms with a convergence rate exactly matching the best rate in performative prediction literature by removing additional logarithmic factors in our result. Second, endogenous distribution change and corruptions are two key factors to handle on the way to robust learning in open environments, and evidently, more factors, such as the exogenous distribution change, should also be considered. Third, it is also important and much more challenging to incorporate the evolution of different factors in learning process on the way towards robust machine learning methods in open environments.

Acknowledgements

This research was supported by the National Science Foundation of China (61921006, 62206125), and Collaborative Innovation Center of Novel Software Technology and Industrialization. We are grateful for the anonymous reviewers for their helpful comments.

References

Ainesh Bakshi, Ilias Diakonikolas, He Jia, Daniel M Kane, Pravesh K Kothari, and Santosh S Vempala. Robustly learning mixtures of k arbitrary gaussians. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 1234–1247, 2022.

Yoshua Bengio, Yann Lecun, and Geoffrey Hinton. Deep learning for AI. *Communications of the ACM*, 64(7): 58–65, 2021.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

Gavin Brown, Shlomi Hod, and Iden Kalemaj. Performative prediction in a stateful world. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 6045–6061, 2022.

Yu Cheng, Ilias Diakonikolas, Daniel Kane, and Alistair Stewart. Robust learning of fixed-structure bayesian net-

works. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pages 10304–10316, 2018.

Joshua Cutler, Dmitriy Drusvyatskiy, and Zaid Harchaoui. Stochastic optimization under distributional drift. *ArXiv preprint*, arXiv:2108.07356, 2021.

Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. Strongly adaptive online learning. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pages 1405–1411, 2015.

Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 655–664, 2016.

Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 1596–1606, 2019.

Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robustness meets algorithms. *Communications of the ACM*, 64(5): 107–115, 2021.

Dmitriy Drusvyatskiy and Lin Xiao. Stochastic optimization with decision-dependent distributions. *Mathematics of Operations Research*, 2022.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

Frank R Hampel, Elvezio M Ronchetti, Peter Rousseeuw, and Werner A Stahel. *Robust Statistics: The Approach Based on Influence Functions*. Wiley-Interscience; New York, 1986.

Elad Hazan and C. Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pages 393–400, 2009.

Peter J. Huber. *Robust Statistics*. Wiley Series in Probability and Statistics. Wiley, 1981.

Zachary Izzo, Lexing Ying, and James Zou. How to learn when data reacts to your model: performative gradient descent. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 4641–4650, 2021.

Zachary Izzo, James Zou, and Lexing Ying. How to learn when data gradually reacts to your model. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3998–4035, 2022.

Meena Jagadeesan, Tijana Zrnic, and Celestine Mendler-Dünner. Regret minimization with performative feedback. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pages 9760–9785, 2022.

Kevin A Lai, Anup B Rao, and Santosh Vempala. Agnostic estimation of mean and covariance. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 665–674, 2016.

Qiang Li and Hoi-To Wai. State dependent performative prediction with stochastic approximation. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3164–3186, 2022a.

Qiang Li and Hoi-To Wai. State dependent performative prediction with stochastic approximation. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3164–3186, 2022b.

Xiyang Liu, Weihao Kong, Sham Kakade, and Sewoong Oh. Robust and differentially private mean estimation. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pages 3887–3901, 2021.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 114–122, 2018.

Debmalya Mandal, Stelios Triantafyllou, and Goran Radanovic. Performative reinforcement learning. *ArXiv preprint*, arXiv:2207.00046, 2022.

Celestine Mendler-Dünner, Juan Perdomo, Tijana Zrnic, and Moritz Hardt. Stochastic optimization for performative prediction. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pages 4929–4939, 2020.

John P Miller, Juan C Perdomo, and Tijana Zrnic. Outside the echo chamber: Optimizing the performative risk. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 7710–7720, 2021.

Adhyyan Narang, Evan Faulkner, Dmitriy Drusvyatskiy, Maryam Fazel, and Lillian Ratliff. Learning in stochastic monotone games with decision-dependent data. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 5891–5912, 2022.

Art B Owen. A robust hybrid of LASSO and ridge regression. *Contemporary Mathematics*, 443(7):59–72, 2007.

Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pages 7599–7609, 2020.

Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust estimation via robust gradient estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) (JRSSB)*, 82 (3), 2020.

Peter J Rousseeuw. Multivariate estimation with high breakdown point. *Mathematical statistics and applications*, 8(283-297):37, 1985.

Masashi Sugiyama and Motoaki Kawanabe. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT press, 2012.

John W Tukey. A survey of sampling from contaminated distributions. *Contributions to probability and statistics*, pages 448–485, 1960.

John W Tukey. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians, Vancouver, 1975*, volume 2, pages 523–531, 1975.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 5650–5659, 2018.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pages 1323–1333, 2018.

Peng Zhao, Yu-Jie Zhang, Lijun Zhang, and Zhi-Hua Zhou. Dynamic regret of convex and smooth functions. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pages 12510–12520, 2020.

Peng Zhao, Yu-Jie Zhang, Lijun Zhang, and Zhi-Hua Zhou. Adaptivity and non-stationarity: Problem-dependent dynamic regret for online convex optimization. *ArXiv preprint*, arXiv:2112.14368, 2021.

Peng Zhao, Yan-Feng Xie, Lijun Zhang, and Zhi-Hua Zhou. Efficient methods for non-stationary online learning. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, page to appear, 2022.

Zhi-Hua Zhou. Open-environment machine learning. *National Science Review*, 9(8), 2022.

A Useful Lemmas

Lemma 5. *Under Assumptions 1 and 3, the following holds for all $x \in \mathcal{X}$,*

$$\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\|_2 \leq \gamma\beta \cdot \|x - \bar{x}\|_2 ,$$

where $\bar{x} = \arg\min_{x \in \mathcal{X}} f_{\bar{x}}(x)$ is the performatively stable point defined in Section 3.

Proof. Fix a unit vector $v \in \mathbb{R}^d$, and define the function $g(z) = \langle \nabla \ell(z; x), v \rangle$. By Assumption 1, $g(z)$ is β-Lipschitz continuous, so we have for any $x \in \mathcal{X}$:

$$\begin{aligned}
\langle \nabla f_x(x) - \nabla f_{\bar{x}}(x), v \rangle &= \langle \mathbb{E}_{z \sim \mathcal{D}(x)} [\nabla \ell(z; x)] - \mathbb{E}_{z \sim \mathcal{D}(\bar{x})} [\nabla \ell(z; x)], v \rangle \\
&= \mathbb{E}_{z \sim \mathcal{D}(x)} [g(z)] - \mathbb{E}_{z \sim \mathcal{D}(\bar{x})} [g(z)] \\
&\leq \beta \cdot W_1(\mathcal{D}(x), \mathcal{D}(\bar{x})) \\
&\leq \beta\gamma \cdot \|x - \bar{x}\|_2 ,
\end{aligned}$$

where the first inequality holds by definition of Wasserstein-1 distance, and the second inequality is because of Assumption 3. Since the above holds for any unit vector v, it also holds for $v = \frac{\nabla f_x(x) - \nabla f_{\bar{x}}(x)}{\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\|_2}$, so we have

$$\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\|_2 = \left\langle \nabla f_x(x) - \nabla f_{\bar{x}}(x), \frac{\nabla f_x(x) - \nabla f_{\bar{x}}(x)}{\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\|_2} \right\rangle \leq \beta\gamma \cdot \|x - \bar{x}\|_2 ,$$

and this ends the proof. □

Lemma 6. *Under Assumptions 1, 2 and 3, the following holds for all $x \in \mathcal{X}$,*

$$f_{\bar{x}}(\bar{x}) \geq f_{\bar{x}}(x) + \langle \nabla f_x(x), \bar{x} - x \rangle + \frac{\alpha(1 - 2\rho)}{2} \|\bar{x} - x\|_2^2 ,$$

where $\rho = \beta\gamma/\alpha$, and $\bar{x} = \arg\min_{x \in \mathcal{X}} f_{\bar{x}}(x)$ is the performatively stable point defined in Section 3.

Proof. We first prove that $f_y(x)$ is strongly convex for all $y \in \mathcal{X}$. Fix $y \in \mathcal{X}$, by Assumption 2 we have for any $x, x' \in \mathcal{X}$,

$$\begin{aligned}
f_y(x) &= \mathbb{E}_{z \sim \mathcal{D}(y)} [\ell(z; x)] \\
&\geq \mathbb{E}_{z \sim \mathcal{D}(y)} \left[\ell(z; x') + \langle \nabla \ell(z; x'), x - x' \rangle + \frac{\alpha}{2} \|x - x'\|_2^2 \right] \\
&= f_y(x) + \langle \nabla f_y(x'), x - x' \rangle + \frac{\alpha}{2} \|x - x'\|_2^2 .
\end{aligned}$$

Therefore, $f_{\bar{x}}(x)$ is also strongly convex in x, so we have

$$\begin{aligned}
f_{\bar{x}}(\bar{x}) &\geq f_{\bar{x}}(x) + \langle \nabla f_{\bar{x}}(x), \bar{x} - x \rangle + \frac{\alpha}{2} \|\bar{x} - x\|_2^2 \\
&= f(x) + \langle \nabla f_x(x), \bar{x} - x \rangle + \langle \nabla f_{\bar{x}}(x) - \nabla f_x(x), \bar{x} - x \rangle + \frac{\alpha}{2} \|\bar{x} - x\|_2^2 .
\end{aligned}$$

By Cauchy-Schwarz inequality and according to Lemma 5 we have

$$\langle \nabla f_{\bar{x}}(x) - \nabla f_x(x), x - \bar{x} \rangle \leq \|\nabla f_{\bar{x}}(x) - \nabla f_x(x)\|_2 \cdot \|x - \bar{x}\|_2 \leq \beta\gamma \cdot \|x - \bar{x}\|_2^2 .$$

Therefore, $\langle \nabla f_{\bar{x}}(x) - \nabla f_x(x), \bar{x} - x \rangle \geq -\beta\gamma \cdot \|x - \bar{x}\|_2^2$, so

$$\begin{aligned}
f_{\bar{x}}(\bar{x}) &\geq f(x) + \langle \nabla f_x(x), \bar{x} - x \rangle + \left(\frac{\alpha}{2} - \beta\gamma \right) \|\bar{x} - x\|_2^2 \\
&= f_{\bar{x}}(x) + \langle \nabla f_x(x), \bar{x} - x \rangle + \frac{\alpha(1 - 2\rho)}{2} \|\bar{x} - x\|_2^2 ,
\end{aligned}$$

and the proof is finished. □

B Proof of Lemma 1

Proof. Throughout the proof, we slightly abuse the notations by using $f(x)$ to denote $f_{\bar{x}}(x)$, the expected performance of model x evaluated on the distribution induced by the performatively stable point $\bar{x}$.

For any estimated gradient $\widetilde{g}_t$ at round t, by Assumption 1 we have

$$\begin{aligned} f(x_{t+1}) &\leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 \\ &= f(x_t) + \langle \nabla f(x_t) - \widetilde{g}_t, x_{t+1} - x_t \rangle + \langle \widetilde{g}_t, x_{t+1} - x_t \rangle + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 \ . \end{aligned} \tag{1}$$

We first convert the drifting terms containing $x_{t+1} - x_t$. On the one hand, by Young's inequality, for any $\delta_t > 0$ we have

$$\langle \nabla f(x_t) - \widetilde{g}_t, x_{t+1} - x_t \rangle \leq \frac{\delta_t}{2} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 + \frac{\delta_t^{-1}}{2} \|x_{t+1} - x_t\|_2^2 \ . \tag{2}$$

On the other hand, define $G_t(x) = \langle \widetilde{g}_t, x - x_t \rangle + \frac{1}{2\eta_t} \|x - x_t\|_2^2$ which is a $\frac{1}{\eta_t}$-strongly convex function, so we have

$$G_t(\bar{x}) \geq G_t(x_{t+1}) + \langle \nabla G_t(x_{t+1}), \bar{x} - x_{t+1} \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 \ .$$

Further more, according to the update rule ($\star$), x_{t+1} is the minimizer of $G_t(\cdot)$, so $\nabla G_t(x_{t+1}) = \mathbf{0}$, and therefore,

$$\langle \widetilde{g}_t, \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 \geq \langle \widetilde{g}_t, x_{t+1} - x_t \rangle + \frac{1}{2\eta_t} \|x_{t+1} - x_t\|_2^2 + \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 \ .$$

By re-arranging terms we have

$$\langle \widetilde{g}_t, x_{t+1} - x_t \rangle \leq \langle \widetilde{g}_t, \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 - \frac{1}{2\eta_t} \|x_{t+1} - x_t\|_2^2 \ . \tag{3}$$

Substitute the above two inequalities (2) and (3) into inequality (1) yields

$$\begin{aligned} f(x_{t+1}) \leq\ & f(x_t) + \frac{\delta_t}{2} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 + \frac{\delta_t^{-1}}{2} \|x_{t+1} - x_t\|_2^2 + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 \\ & + \langle \widetilde{g}_t, \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 - \frac{1}{2\eta_t} \|x_{t+1} - x_t\|_2^2 \\ =\ & f(x_t) + \langle \widetilde{g}_t, \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 \\ & + \frac{\delta_t}{2} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 + \frac{\delta_t^{-1} - \eta_t^{-1} + L}{2} \|x_{t+1} - x_t\|_2^2 \ . \end{aligned}$$

Choose $\delta_t = \eta_t / (1 - \eta_t L)$ to eliminate the term $\|x_{t+1} - x_t\|_2^2$ and thus we obtain

$$\begin{aligned} f(x_{t+1}) \leq\ & f(x_t) + \langle \widetilde{g}_t, \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 \\ & - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 + \frac{\eta_t}{2(1 - \eta_t L)} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 \ . \end{aligned}$$

Then we seek to control the remaining inner product term. By Lemma 6 we have

$$\begin{aligned} f(x_{t+1}) \leq\ & f(x_t) + \langle \nabla f_{x_t}(x_t), \bar{x} - x_t \rangle + \langle \widetilde{g}_t - \nabla f_{x_t}(x_t), \bar{x} - x_t \rangle + \frac{1}{2\eta_t} \|\bar{x} - x_t\|_2^2 \\ & - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 + \frac{\eta_t}{2(1 - \eta_t L)} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 \\ \leq\ & f(\bar{x}) + \frac{\eta_t^{-1} - \alpha(1 - 2\rho)}{2} \|\bar{x} - x_t\|_2^2 + \langle \widetilde{g}_t - \nabla f_{x_t}(x_t), \bar{x} - x_t \rangle \\ & - \frac{1}{2\eta_t} \|\bar{x} - x_{t+1}\|_2^2 + \frac{\eta_t}{2(1 - \eta_t L)} \|\nabla f(x_t) - \widetilde{g}_t\|_2^2 \ . \end{aligned} \tag{4}$$

Again by Young's inequality we have for any $\lambda_t > 0$,

$$\langle \widetilde{g}_t - \nabla f_{x_t}(x_t), \bar{x} - x_t \rangle \leq \frac{\lambda_t}{2} \left\| \widetilde{g}_t - \nabla f_{x_t}(x_t) \right\|_2^2 + \frac{1}{2\lambda_t} \left\| \bar{x} - x_t \right\|_2^2 . \tag{5}$$

Subtituting (5) into (4) yields

$$\begin{aligned} f(x_{t+1}) \leq\ & f(\bar{x}) + \frac{\eta_t^{-1} - \alpha(1-2\rho)}{2} \left\| \bar{x} - x_t \right\|_2^2 + \frac{\lambda_t}{2} \left\| \widetilde{g}_t - \nabla f_{x_t}(x_t) \right\|_2^2 + \frac{1}{2\lambda_t} \left\| \bar{x} - x_t \right\|_2^2 \\ & - \frac{1}{2\eta_t} \left\| \bar{x} - x_{t+1} \right\|_2^2 + \frac{\eta_t}{2(1-\eta_t L)} \left\| \nabla f(x_t) - \widetilde{g}_t \right\|_2^2 \\ \leq\ & f(\bar{x}) + \frac{\eta_t^{-1} - \alpha(1-2\rho) + \lambda_t^{-1}}{2} \left\| \bar{x} - x_t \right\|_2^2 + \frac{\lambda_t}{2} \left\| \widetilde{g}_t - \nabla f_{x_t}(x_t) \right\|_2^2 \\ & - \frac{1}{2\eta_t} \left\| \bar{x} - x_{t+1} \right\|_2^2 + \frac{\eta_t}{2(1-\eta_t L)} \left\| \nabla f(x_t) - \widetilde{g}_t \right\|_2^2 . \end{aligned}$$

Finally, we convert remaining terms into algorithmically-controllable quantities. By simple calculation we have

$$\left\| \nabla f(x_t) - \widetilde{g}_t \right\|_2^2 \leq\ 2 \left\| \nabla f(x_t) - \nabla f_{x_t}(x_t) \right\|_2^2 + 2 \left\| \nabla f_{x_t}(x_t) - \widetilde{g}_t \right\|_2^2 .$$

Therefore,

$$\begin{aligned} f(x_{t+1}) \leq\ & f(\bar{x}) + \frac{\eta_t^{-1} - \alpha(1-2\rho) + \lambda_t^{-1}}{2} \left\| \bar{x} - x_t \right\|_2^2 - \frac{1}{2\eta_t} \left\| \bar{x} - x_{t+1} \right\|_2^2 \\ & + \left(\frac{\lambda_t}{2} + \frac{\eta_t}{1-\eta_t L} \right) \left\| \widetilde{g}_t - \nabla f_{x_t}(x_t) \right\|_2^2 + \frac{\eta_t}{1-\eta_t L} \left\| \nabla f(x_t) - \nabla f_{x_t}(x_t) \right\|_2^2 . \end{aligned} \tag{6}$$

By definition, we have that $\bar{x}$ is the minimizer of the α-strongly convex function $f(x)$, so $f(x_{t+1}) \geq f(\bar{x}) + \frac{\alpha}{2} \left\| \bar{x} - x_{t+1} \right\|_2^2$, substitute it into inequality (6) yields

$$\begin{aligned} \frac{\alpha}{2} \left\| \bar{x} - x_{t+1} \right\|_2^2 \leq\ & \frac{\eta_t^{-1} - \alpha(1-2\rho) + \lambda_t^{-1}}{2} \left\| \bar{x} - x_t \right\|_2^2 - \frac{1}{2\eta_t} \left\| \bar{x} - x_{t+1} \right\|_2^2 \\ & + \left(\frac{\lambda_t}{2} + \frac{\eta_t}{1-\eta_t L} \right) \left\| \widetilde{g}_t - \nabla f_{x_t}(x_t) \right\|_2^2 + \frac{\eta_t}{1-\eta_t L} \left\| \nabla f(x_t) - \nabla f_{x_t}(x_t) \right\|_2^2 . \end{aligned}$$

Multiply both sides by $2\eta_t$ and re-arranging terms finish the proof. □

C Proof of Lemma 2

Lemma 2 provides an upper bound on the error of the output $\widetilde{g}_t$ of Algorithm 1 for robust gradient estimation, whose full proof can be found in [Prasad et al., 2020, Appendix K]. In this part, we summarize the main ideas. To this end, we first analyze the simple one-dimension case, and then turn to the general case when $d > 1$.

One-dimension case. The first part of the proof analyzes Algorithm 1 in one-dimensional case. The main idea in this part is to firstly identify two events that occurs with high probability, then condition on the events and decompose the total error into separate terms in order to deal with them separately. The two events are the following:

(a) The event of a low-corruption regime. A direct use of Hoeffding's inequality shows that with probability at least $1 - \delta/3$, the fraction of corrupted samples is less than $\epsilon' = \epsilon + \sqrt{\frac{\log(3/\delta)}{2n}}$.

(b) The event of a concentrated interval. Let $I_{1-\epsilon'}$ be the interval around the mean of the uncorrupted distribution containing $1 - \epsilon'$ mass of the uncorrupted distribution. On the one hand, the length of $I_{1-\epsilon'}$ is bounded according to Assumption 5 and Chebyshev's inequality. On the other hand, by classical VC theory we can show that a large fraction of samples lie in this interval with probability at least $1 - \delta/3$.

Conditioned on events (a) and (b), it suffices to bound the total error in the task of mean estimation. The sources of error can be divided into three parts and then separately be controlled:

- The error caused by corruptions. As the fraction of corruptions is small and that we are conditioned in $I_{1-\epsilon'}$ whose length is already controlled, the maximum error in this source can be at most $\epsilon' \cdot \text{length}(I_{1-\epsilon'})$;
- The deviation of empirical mean estimation on uncorrupted fraction of samples to the conditional mean. It is easy to control this term of error with concentration inequalities.
- The deviation between conditional mean and the unconditional mean. These two quantities are intuitively close because of the fact that the probability of the events being conditioned on is large. The formal description of the intuition is in [Lai et al., 2016, Lemma 3.11].

Having obtain the bounds for all sources of error, we can finally achieve the bound of overall error in one-dimension case by a simple summation of bounds on three sources of error and the union bound inequality.

Extension to general case. Algorithm 1 takes a corruption removal step similar to the one-dimension case as a subroutine, and include two new operations: (1) project onto a subspace with reduced dimensions, and (2) recursively apply corruption removal step and projection step until the dimension $d = 1$. We highlight key steps to analyze those operations.

(1) There are two key steps in analyzing the effect of projection operation: (a) Control the deviation between empirical covariance and the covariance of uncorrupted distribution before projection; (b) Control the deviation of empirical mean after projection. Step (a) is obtained by firstly decompose the error into three terms and then separately control them, which is similar to the one-dimension case, and step (b) is finished by firstly decomposing the covariance matrix into the covariance only induced by uncorrupted samples and the rest induced by corruptions, then control the covariance of uncorrupted samples by spectral norm and the corrupted part of covariance by efforts in step (a).

(2) The recursion of Algorithm 1 is unrolled by an observation that the error of running one step of the algorithm on reduced dimensions can be bounded by the error of running one step of the algorithm on initial dimensions with a reduced set of samples. Based on the observations and the achievements in step (1) we finish the proof of Lemma 2.

D Proof of Theorem 3

Proof. Without loss of generality, we assume $\widehat{\alpha} = \alpha - \beta\gamma \geq 1$ throughout the proof, since we can achieve this condition by simply multiplying the loss function by a constant that is large enough, which retains the effectiveness of the conclusion.

First we have for all $t \in [T]$ according to Lemma 1:

$$(1+\alpha\eta_t)\left\|x_{t+1}-\bar{x}\right\|_2^2 \leq \left(1+\frac{\eta_t}{\lambda_t}-\alpha\eta_t(1-2\rho)\right)\left\|x_t-\bar{x}\right\|_2^2+\left(\eta_t\lambda_t+\frac{2\eta_t^2}{1-\eta_t L}\right)\left\|\widetilde{g}_t-\nabla f_{x_t}(x_t)\right\|_2^2 + \frac{2\eta_t^2}{1-\eta_t L}\left\|\nabla f_{x_t}(x_t)-\nabla f_{\bar{x}}(x_t)\right\|_2^2 .$$

We subsequently seek controls for the last two terms. On the one hand, according to the assignment on n_t in Theorem 3, we can verify (by a standard but complicated calculation) that

$$\left(\frac{d\log d\log(nT/(d\delta))}{n}\right)^{3/8} \leq \frac{1}{2C\sqrt{\log d}}-\frac{\sqrt{\epsilon}}{2}, \text{ and}$$
$$\left(\frac{\epsilon d^2\log d\log\left(\frac{dT\log d}{\delta}\right)}{n}\right)^{1/4} \leq \frac{1}{2C\sqrt{\log d}}-\frac{\sqrt{\epsilon}}{2}.$$

Therefore, we have

$$C\left(\sqrt{\epsilon}+\gamma(n_t,d,\delta/T,\epsilon)\right)\sqrt{\log d}\leq 1,$$

where $\gamma(\cdot,\cdot,\cdot,\cdot)$ is defined in Lemma 2, and applying Lemma 2 we have with probability at least $1-\delta/T$,

$$\left\|\widetilde{g}_t-\nabla f_{x_t}(x_t)\right\|_2^2\leq\sigma^2.$$

On the other hand, Lemma 5 directly yield an upper bound of the last term:

$$\left\|\nabla f_{x_t}(x_t)-\nabla f_{\bar{x}}(x_t)\right\|_2^2\leq\gamma^2\beta^2\cdot\left\|x_t-\bar{x}\right\|_2^2.$$

Substitute $\eta_t = \eta$ and $\lambda_t = 1$, by re-arranging terms we obtain that with probability at least $1 - \delta/T$,

$$(1 + \eta\alpha) \left\| x_{t+1} - \bar{x} \right\|_2^2 \leq \left(\eta + \frac{2\eta^2}{1 - \eta L} \right) \sigma^2 + \left(1 + \frac{2\eta^2\beta^2\gamma^2}{1 - \eta L} + \eta - \alpha\eta(1 - 2\rho) \right) \left\| x_t - \bar{x} \right\|_2^2 .$$

Now we simplify the coefficients under condition $\eta \leq 1/\left(4\beta^2\gamma^2/\widehat{\alpha} + 2L\right)$. Since $\eta < 1/(2L)$, we have $\frac{1}{1-\eta L} < 2$, so

$$\frac{\eta + \frac{2\eta^2}{1-\eta L}}{1 + \eta\alpha} \leq \frac{\eta + 4\eta^2}{1 + \eta\alpha} .$$

The second coefficient is more complex to simplify. We start by the conversion

$$\begin{aligned}
\frac{1 + \frac{2\eta^2\beta^2\gamma^2}{1-\eta L} + \eta - \alpha\eta(1 - 2\rho)}{1 + \eta\alpha} &= 1 - \frac{\eta\alpha - \frac{2\eta^2\beta^2\gamma^2}{1-\eta L} - \eta + \alpha\eta(1 - 2\rho)}{1 + \eta\alpha} \\
&= 1 - \frac{\eta(2\alpha - 2\alpha\rho - 1) - \frac{2\eta^2\beta^2\gamma^2}{1-\eta L}}{1 + \eta\alpha} \\
&= 1 - \frac{\eta(2\widehat{\alpha} - 1) - \frac{2\eta^2\beta^2\gamma^2}{1-\eta L}}{1 + \eta\alpha} \\
&\leq 1 - \frac{\eta\widehat{\alpha} - \frac{2\eta^2\beta^2\gamma^2}{1-\eta L}}{1 + \eta\alpha},
\end{aligned}$$

where the last two steps are because the definitions $\rho = \beta\gamma/\alpha$ and $\widehat{\alpha} = \alpha - \beta\gamma \geq 1$. Note that $\frac{2\eta^2\beta^2\gamma^2}{1-\eta L} \leq \frac{\widehat{\alpha}\eta}{2}$, so

$$\begin{aligned}
\frac{1 + \frac{2\eta^2\beta^2\gamma^2}{1-\eta L} + \eta - \alpha\eta(1 - 2\rho)}{1 + \eta\alpha} &\leq 1 - \frac{\eta\widehat{\alpha} - \frac{2\eta^2\beta^2\gamma^2}{1-\eta L}}{1 + \eta\alpha} \\
&\leq 1 - \frac{\eta\widehat{\alpha}}{2(1 + \eta\alpha)}.
\end{aligned}$$

Now we arrive the recursive formula for all $t \in [T]$, with probability at least $1 - \delta/T$,

$$\left\| x_{t+1} - \bar{x} \right\|_2^2 \leq \underbrace{\left(1 - \frac{\widehat{\alpha}\eta}{2(1 + \eta\alpha)} \right)}_{:=q} \left\| x_t - \bar{x} \right\|_2^2 + \underbrace{\frac{\eta + 4\eta^2}{1 + \eta\alpha}\sigma^2}_{:=C} .$$

We finally unroll the recursive formula. Further define $A_t = \left\| x_t - \bar{x} \right\|_2^2$, we have with probability at least $1 - \delta/T$,

$$A_{t+1} \leq q \cdot A_t + C .$$

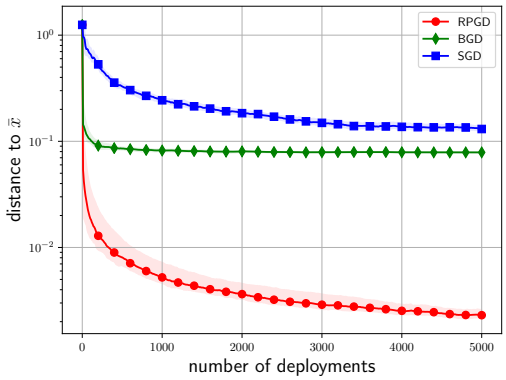
Unroll the recursion, and use the union bound inequality on probabilities, we have with probability at least $1 - \delta$,

$$A_{t+1} \leq q^t A_1 + \frac{1 - q^t}{1 - q} C \leq q^t A_1 + \frac{C}{1 - q}.$$

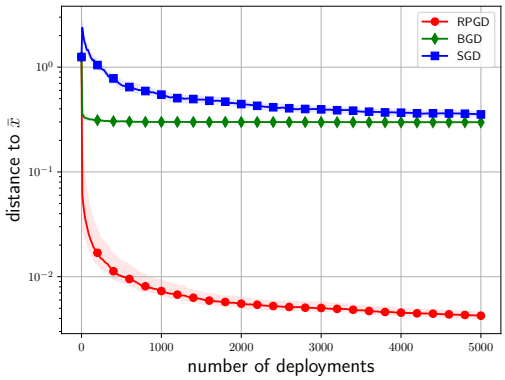
Substitute the symbols back and use the estimate $\eta\alpha \leq \frac{1}{2}$, we finally have with probability at least $1 - \delta$,

$$\begin{aligned}
\left\| x_{t+1} - \bar{x} \right\|_2^2 &\leq \left(1 - \frac{\widehat{\alpha}\eta}{2(1 + \eta\alpha)} \right)^T \left\| x_1 - \bar{x} \right\|_2^2 + \frac{2 + 8\eta}{\widehat{\alpha}}\sigma^2 \\
&\leq \left(1 - \frac{\widehat{\alpha}\eta}{3} \right)^T \left\| x_1 - \bar{x} \right\|_2^2 + \frac{2 + 8\eta}{\widehat{\alpha}}\sigma^2,
\end{aligned}$$

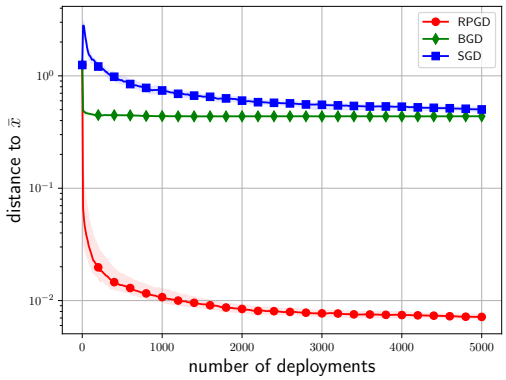
and the proof is completed. □



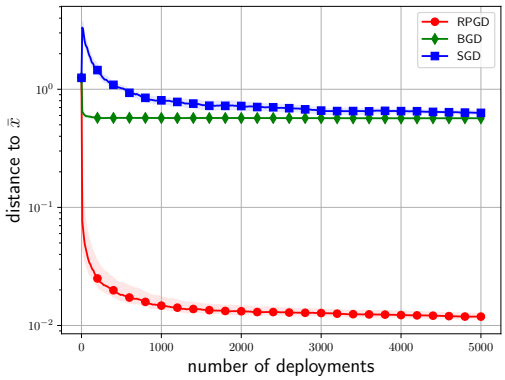
(a) Mean estimation, $\epsilon = 0.05$



(b) Mean estimation, $\epsilon = 0.2$



(c) Mean estimation, $\epsilon = 0.3$



(d) Mean estimation, $\epsilon = 0.4$

Figure 2: Additional results of mean estimation with $\mu = \mathbf{1}_d/\sqrt{d}$ and $\Sigma = 0.1 I_d$, where $\mathbf{1}_d$ and I_d respectively denote all one's vector and identity matrix. Each experiment is repeated 20 times and we display 95% bootstrap confidence intervals.

E Proof of Corollary 4

Proof. Let $\eta = \frac{3}{\widehat{\alpha}}\left[1 - \left(\frac{8\sigma^2}{\widehat{\alpha}T}\right)^{\frac{1}{T-1}}\right]$. By Theorem 3 we directly have with probability at least $1-\delta$:

$$\|x_{T+1} - \bar{x}\|_2^2 \leq \frac{8\sigma^2}{\widehat{\alpha}T} + \frac{2\sigma^2}{\widehat{\alpha}} + \frac{24\sigma^2}{\widehat{\alpha}^2}\left[1 - \left(\frac{8\sigma^2}{\widehat{\alpha}T}\right)^{\frac{1}{T-1}}\right].$$

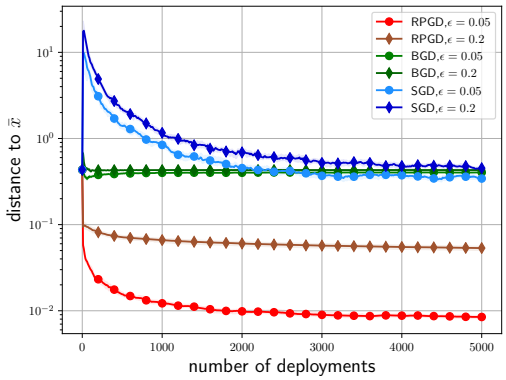
And the equality can be verified by the second order Taylor expansion at $T = \infty$. □

F Additional Experiments

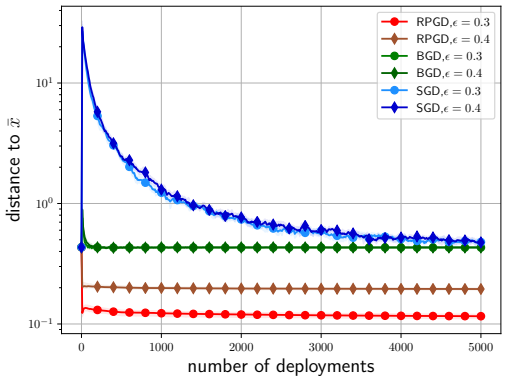
In this section, we provide more experiment results to further support our findings.

Additional Results of Mean Estimation. Figure 2 demonstrates the additional results of the task of mean estimation, in which similar phenomenon is observed: as the level of corruptions increases, our Algorithm 2 retains convergent and suffers little from the corruptions ratio, while BGD and SGD fail to converge even if the level of corruptions is small.

Another interest thing that worth to notice is about the maximum level of corruptions tolerated in our algorithm: In analysis, the threshold is $\epsilon \leq 1/(C^2 \log d)$, where the value is about 0.33 by assuming $C = 1$ when $d = 20$. Nevertheless, our algorithm converges even if the level of corruptions is $\epsilon = 0.4$, as is shown in Figure 2d. Therefore, our Algorithm 2 possibly enjoys stronger theoretical guarantees, which may be established by more sophisticated techniques and is left as another interesting future work.

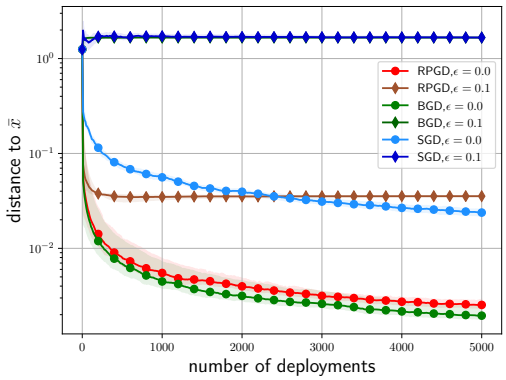


(a) Strategic classification, $\epsilon = 0.05,\ 0.2$

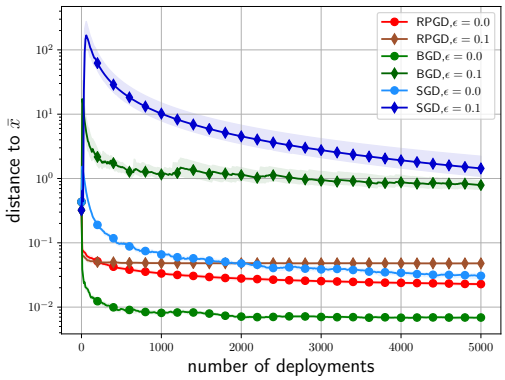


(b) Strategic classification, $\epsilon = 0.3,\ 0.4$

Figure 3: Additional results of strategic classification with $\mu = \mathbf{1}_d/\sqrt{d}$ and $\Sigma = 25I_d$, where $\mathbf{1}_d$ and I_d denote all one's vector and identity matrix. Each experiment is repeated 20 times and we display 95% bootstrap confidence intervals.



(a) Mean estimation



(b) Strategic classification

Figure 4: Error versus number of deployments for our algorithm and two benchmark algorithms BGD and SGD on synthetic problems with heavy-tailed corruptions and misspecification. The corruptions are the *Pareto distribution* which is known as a common heavy-tailed distribution, and we only know an upper bound $\epsilon' = \epsilon + 0.1$ of the true level of corruptions ϵ. Each experiment is repeated 20 times and we display 95% bootstrap confidence intervals.

Additional Results of Strategic Classification. Figure 3 demonstrates the additional results of the task of strategic classification, in which extra levels of corruptions are included. The results support that our algorithm outperforms the two benchmarks BGD and SGD in various levels of corruptions.

Additional Results of Heavy-Tailed Corruptions and Misspecification. We also verify the effectiveness of our algorithm in more severe scenarios. We continue to use mean estimation and strategic classification as basic tasks, while the corruptions are simulated via *Pareto distribution* which is known as a common heavy-tailed distribution. Meanwhile, the true ratio of corruptions ϵ is no longer released to the algorithms, and instead we only know an upper bound $\epsilon' = \epsilon + 0.1$.

Figure 4 demonstrates the mean of the error among 20 repetitions of experiment with various levels of corruptions ϵ. On the one hand, we can see that our algorithm is robust to heavy-tailed corruptions, as the results show that our algorithm exhibits similar convergence properties as the previous results in Figure 1. On the other hand, the effect of misspecification is mild to our algorithm, since with absence of corruption (i.e., $\epsilon = 0.0$), our algorithm (red line) has almost the same convergence rate compared to BGD (green line), which support our claim on misspecification below Lemma 2. In conclusion, the results complement our experimental verification by showing the robustness against malignant corruptions and misspecification.