Fast and Adversarial Robust Kernelized SDU Learning

Yajing Fan
Nanjing University of Information Science and Technology

Wanli Shi
MBZUAI

Yi Chang
School of Artificial Intelligence
Jilin University

Bin Gu*
MBZUAI
School of Artificial Intelligence
Jilin University

Abstract

SDU learning, a weakly supervised learning problem with only pairwise similarities, dissimilarities data points and unlabeled data available, has many practical applications. However, it is still lacking in defense against adversarial samples, and its learning process can be expensive. To address this gap, we propose a novel adversarial training framework for SDU learning. Our approach reformulates the conventional minimax problem as an equivalent minimization problem based on the kernel perspective, departing from traditional confrontational training methods. Additionally, we employ the random gradient method and random features to accelerate the training process. Theoretical analysis shows that our method can converge to a stationary point at a rate of $\mathcal{O}(1/T^{1/4})$. Our experimental results show that our algorithm is superior to other adversarial training methods in terms of generalization, efficiency and scalability against various adversarial attacks.

1 INTRODUCTION

In supervised classification, the requirement for a large amount of labeled training data to train classifiers poses challenges. The challenge of acquiring labels arises from various factors, such as the substantial costs associated with labeling processes (Chapelle et al., 2010), apprehensions about privacy (Warner, 1965), potential social biases (Nederhof, 1985), and the inherent complexity involved in labeling datasets. Consequently, real-world classification problems often involve scenarios where collecting pairwise similarities (i.e., pairs of samples belonging to the same class) and dissimilarities (i.e., pairs of samples belonging to different classes) may be easier compared to obtaining fully labeled data. For instance, in tasks like protein function prediction (Klein et al., 2002), knowledge about similarities and dissimilarities can be obtained as additional supervision through experimental means. To address the utilization of such pairwise information, the concept of Similar-Dissimilar-Unlabeled (SDU) learning has been proposed (Shimada et al., 2021). The practical value of SDU learning has garnered significant attention within the data mining and machine learning communities.

SDU learning (Shimada et al., 2021) introduces two classification types: Dissimilar-Unlabeled (DU) classification and Similar-Dissimilar (SD) classification. It combines the risks associated with different types of classifications, similar to the approach used in Positive-Negative-Unlabeled classification (Sakai et al., 2017). SDU learning employs the double hinge loss $\ell_{DH} = \max(-tz, \max(0, \frac{1}{2} - \frac{1}{2}tz))$ (Du Plessis et al., 2015), and the optimization problem is tackled using quadratic programming. In recent years, there has been a growing interest in SDU learning. Notable studies in this field include the application of Multiple-Instance Learning to SDU bags (Feng et al., 2023) and the exploration of the relationship between similarity learning and binary classification (Bao et al., 2022). In addition to considering SD, DU classification simultaneously, some studies only focus on a certain part of them. For instance, Bao et al. (2018) centers around similar data and unlabeled data. Wu et al. (2022) introduces noisy data for robustness. Wu et al. (2020) is designed for multi-class classification with noisy similarity labels. Cao et al. (2021) aims for improved accuracy using similarity confidence. Maheshwara and Manwani (2023) proposes robust classifiers with noisy pairwise data, and Dan et al. (2021) addresses pairwise supervision with noisy SD data. These studies contribute to the broader understanding of SDU learning

Proceedings of the 27[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

Table 1: Comparative Analysis of Robustness Factors for Various Algorithms in Kernel Perspective. (S. denotes Samples; C. Rate denotes Convergence Rate; Com. denotes Complexity; n denotes the number of training dataset; T denotes the number of iterations; d denotes the dimension of data; m denotes the number of random features.)

Algorithm	Noisy S.	**Adversarial Samples**	Unlabel S.	C. Rate	Time Com.	Space Com.	Convexity
SU-SL (Bao et al., 2018)	×	✗	✓	-	$\mathcal{O}(n^3)$	$\mathcal{O}(n^2)$	convex
SDU-DH (Shimada et al., 2021)	×	✗	✓	-	$\mathcal{O}(n^3)$	$\mathcal{O}((d+2n)^2)$	convex
nSU (Wu et al., 2022)	✓	✗	✓	-	$\mathcal{O}(Tn^3)$	$\mathcal{O}(n^2)$	convex
Sconf (Cao et al., 2021)	✓	✗	✓	-	$\mathcal{O}(Tn^2)$	$\mathcal{O}(n^2)$	non-convex
S-D (Dan et al., 2021)	✓	✗	×	-	$\mathcal{O}(Tn^2)$	$\mathcal{O}(n^2)$	convex
QSG-ATSDU (Ours)	✓	✓	✓	$\mathcal{O}(1/T^{1/4})$	$\mathcal{O}(T^2)$	$\mathcal{O}(Tm)$	non-convex

by exploring different subsets of the available information and proposing novel techniques to tackle specific challenges within the SDU framework.

While the above methods mainly address the difficulty of noise, it is essential to acknowledge that adversarial noise is common in real-world situations. Adversarial samples are generated by making small perturbations to the input to increase the loss incurred by a machine learning model (Szegedy et al., 2014; Shafahi et al., 2019; Wong et al., 2020). These samples have the ability to misguide models into making confidently incorrect predictions, revealing the sensitivity of model outputs to their inputs and indicating a lack of desired generalization. Detecting and defending against adversarial samples is crucial for ensuring the robustness of machine learning models in real-world applications. SDU learning introduces a fresh weakly supervised classification issue. This makes it especially susceptible to adversarial samples due to its reliance on limited supervision. Incomplete labeling obstructs the model's capacity to effectively distinguish between regular examples and adversarial instances. Thus, it is imperative to investigate the resilience of SDU learning to adversarial samples and develop techniques that enhance the model's robustness against such attacks.

However, existing SDU learning mainly focuses on considering the samples with noise and lack of ability to deal with the adversarial samples as shown in Table 1. Specifically, Bao et al. (2018) and Shimada et al. (2021) considered training a model on clean data, and naturally the model can be easily attacked by adversarial examples. Wu et al. (2022); Cao et al. (2021); Dan et al. (2021) considered training a robust model with noisy data. However, they just use some samples with incorrect labels. As mentioned above, the adversarial samples are carefully designed for fooling the model. Even if the model can deal with some simple noisy data, it is still impossible to classify the adversarial samples. What's worse, from a kernel-focused viewpoint, these algorithms exhibit high time complexity and storing the kernel matrix worsens this, needing at least $O(n^2)$ space. Among them, Bao et al. (2018); Wu et al. (2022) demand matrix inversion, resulting in time complexity up to $O(n^3)$. This means even if we use the minimax method with these methods, which is widely used in traditional adversarial training, it is still very time-consuming. Therefore, it is still an open challenge to design an adversarial training method in SDU learning on the kernel method.

To address the challenge, we propose a new adversarial training framework for SDU learning. Specifically, we formulate the adversarial training problem of SDU learning as a minimax problem. Then, instead of using K-steps protection gradient descent (PGD) (Madry et al., 2017) to train the adversarial samples, we reformulate the minimax problem of adversarial training as a minimization problem by using the kernel method. However, the kernel method usually has high computational complexity. To overcome this problem, we propose a quardruple stochastic gradient based on random Fourier features. Theoretically, we prove our method can converge to a stationary point at the rate of $\mathcal{O}(1/T^{1/4})$. Extensive experimental results revealing superior generalization performance against a range of adversarial attacks. Moreover, they demonstrate efficiency and scalability when compared to alternative methods. The main contributions of this paper are summarized as follows:

1. We propose an innovative objective function for adversarial training of SDU learning based on the kernel method, where we do not need to K-steps PGD to train the adversarial samples.

2. We propose an efficient algorithm for solving the adversarial training of the SDU learning problem based on random Fourier features. We prove our method can converge to a stationary point at the rate of $\mathcal{O}(1/T^{1/4})$ for the non-convex condition.

3. Our experimental results show that our algorithms not only achieve better generalization per-

formance against various adversarial attacks but also enjoy efficiency and scalability compared with other methods.

2 PRELIMINARY

In this section, we introduce our new SDU learning formulation and SDU adversarial training.

2.1 Problem Setting of SDU Learning

Let $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{+1, -1\}$ be a d-dimensional example space and binary label space, respectively. Suppose that each labeled example $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is generated from the joint probability with density $p(x, y)$ independently. For simplicity, let π_+ and π_- be class priors $p(y = +1)$ and $p(y = -1)$, which satisfy the condition $\pi_+ + \pi_- = 1$, $p_+(x)$, and $p_-(x)$ be class conditional densities $p(x|y = +1)$ and $p(x|y = -1)$.

The standard goal of supervised binary classification is to obtain a classifier $f : \mathcal{X} \to \mathbb{R}$ which minimizes the classification risk defined by $R(f) = \mathbb{E}_{(x,y)\sim p(x,y)}[\ell(f(x), y)]$, where $\mathbb{E}_{(x,y)\sim p(x,y)}[\cdot]$ denotes the expected value over joint density $p(x, y)$ and $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_+$ is a loss function.

However, the acquisition of labeled data of this nature often proves to be prohibitively expensive. In real-life scenarios, individuals tend to exhibit hesitation when providing explicit answers and instead prefer responding to indirect questions that capture pairwise similarities and dissimilarities, such as "Which person shares your beliefs?" Recognizing the potential inherent in this type of pairwise information, SDU learning (Shimada et al., 2021) as an innovative approach to solve such problems has been proposed. SDU learning tackles binary classification task by utilizing unlabeled data in conjunction with pairs of samples. These pairs consist of similar instances (i.e., $y = \hat{y}$), indicating that they belong to the same class, and are labeled with $s = +1$. Conversely, dissimilar instances (i.e., $y \neq \hat{y}$) are used to denote pairs belonging to different classes, and are labeled with $s = -1$. In this context, (x, y) and $(\hat{x}, \hat{y})$ represents one single sample respectively. In addition, the concept of pointwise densities (Shimada et al., 2021) has been innovatively introduced to effectively capture the distinguishing characteristics of both similar and dissimilar data. The pointwise densities, labeled as $\widetilde{p}_S(x)$ and $\widetilde{p}_D(x)$, are derived through marginalizing the pairwise densities $p_S(x, \hat{x}) = \frac{\pi_+^2}{\pi_S} p_+(x)p_+(\hat{x}) + \frac{\pi_-^2}{\pi_S} p_-(x)p_-(\hat{x})$ and $p_D(x, \hat{x}) = \frac{1}{2}p_+(x)p_-(\hat{x}) + \frac{1}{2}p_-(x)p_+(\hat{x})$ with respect to the variable $\hat{x}$, where π_S represents the class priors $p(y = +1)p(\hat{y} = +1)+p(y = -1)p(\hat{y} = -1) = \pi_+^2+\pi_-^2$. The definitions of $\widetilde{p}_S(x)$ and $\widetilde{p}_D(x)$ are as follows: $\widetilde{p}_S(x) = \int p_S(x, \hat{x})d\hat{x} = \frac{\pi_+^2}{\pi_+^2+\pi_-^2} p_+(x) + \frac{\pi_-^2}{\pi_+^2+\pi_-^2} p_-(x)$, $\widetilde{p}_D(x) = \int p_D(x, \hat{x})d\hat{x} = \frac{1}{2}p_+(x) + \frac{1}{2}p_-(x)$.

Let us define $\widetilde{\mathcal{D}}_S$ as the set of pointwise samples derived from similar pairs $\mathcal{D}_S \sim p_S$, and $\widetilde{\mathcal{D}}_D$ as the set of pointwise samples obtained from dissimilar pairs $\mathcal{D}_D \sim p_D$. $\mathcal{D}_U$ denotes unlabeled dataset. n_S, n_D, and n_U represent the respective sizes of $\mathcal{D}_S$, $\mathcal{D}_D$, and $\mathcal{D}_U$. More precisely, we express these sets as follows:

$$\begin{aligned}
\widetilde{\mathcal{D}}_S &:= \{\widetilde{x}_{S,i}\}_{i=1}^{2n_S} = \bigcup \{x_S, \hat{x}_S \mid (x_S, \hat{x}_S) \in \mathcal{D}_S\} \\
&\quad \sim \widetilde{p}_S(x), \\
\widetilde{\mathcal{D}}_D &:= \{\widetilde{x}_{D,i}\}_{i=1}^{2n_D} = \bigcup \{x_D, \hat{x}_D \mid (x_D, \hat{x}_D) \in \mathcal{D}_D\} \\
&\quad \sim \widetilde{p}_D(x), \\
\mathcal{D}_U &:= \{x_{U,i}\}_{i=1}^{n_U} \sim p_U(x) = \pi_+ p_+(x) + \pi_- p_-(x).
\end{aligned}$$

2.2 Objective of SDU Learning

SDU classification (Shimada et al., 2021) represents a novel approach that combines DU and SD classification. Its classification risk is specifically defined as follows:

$$R^\gamma(f) = (1 - \gamma)R_{\widetilde{SD}}(f) + \gamma R_{\widetilde{DU}}(f), \tag{1}$$

where $\gamma \in [0, 1]$ is the balance parameter.

The classification risk, denoted as $R_{\widetilde{SD}}(f)$ and $R_{\widetilde{DU}}(f)$, can be estimated based on similar and dissimilar pairs (SD), or dissimilar pairs and unlabeled data (DU). The definitions of these risks are as follows:

$$\begin{aligned}
R_{\widetilde{SD}} =& \pi_S \mathbb{E}_{\widetilde{x}_S}[\hat{\ell}(f(x), +1)] + \pi_D \mathbb{E}_{\widetilde{x}_D}[\hat{\ell}(f(x), -1)], \\
R_{\widetilde{DU}} =& \pi_D \mathbb{E}_{\widetilde{x}_S}[-\widetilde{\ell}(f(x))] + \mathbb{E}_{x_U}[\hat{\ell}(f(x), +1)],
\end{aligned}$$

where $\hat{\ell}(z, t) = \frac{\pi_+}{\pi_+-\pi_-}\ell(z, t) - \frac{\pi_-}{\pi_+-\pi_-}\ell(z, -t)$, $\widetilde{\ell}(z) = \frac{1}{\pi_+-\pi_-}\ell(z, +1) - \frac{1}{\pi_+-\pi_-}\ell(z, -1)$, and $\pi_D = p(y = +1)p(\hat{y} = -1) + p(y = -1)p(\hat{y} = +1) = 2\pi_+\pi_-$.

Then, the expected objective function Eq. (1) reduces to the following:

$$\begin{aligned}
R^\gamma(f) =& \frac{\pi_D}{\pi_c}\mathbb{E}_{\widetilde{x}_D}[\gamma_1 \ell(f(\widetilde{x}_D), -1) - \gamma_2 \ell(f(\widetilde{x}_D), +1)] \\
&+ \frac{\pi_S(1 - \gamma)}{\pi_c}\mathbb{E}_{\widetilde{x}_S}[\pi_+ \ell(f(\widetilde{x}_S), +1) - \pi_- \ell(f(\widetilde{x}_S), -1)] \\
&+ \frac{\gamma}{\pi_c}\mathbb{E}_{x_U}[\pi_+ \ell(f(x_U), +1) - \pi_- \ell(f(x_U), -1)]
\end{aligned} \tag{2}$$

where $\pi_c = \pi_+ - \pi_-, \gamma_1 = \pi_+ + \pi_-\gamma$, and $\gamma_2 = \pi_+\gamma + \pi_-$.

3 PROPOSED METHOD

Standard adversarial training using the minimax formulation is widely recognized for its slow convergence

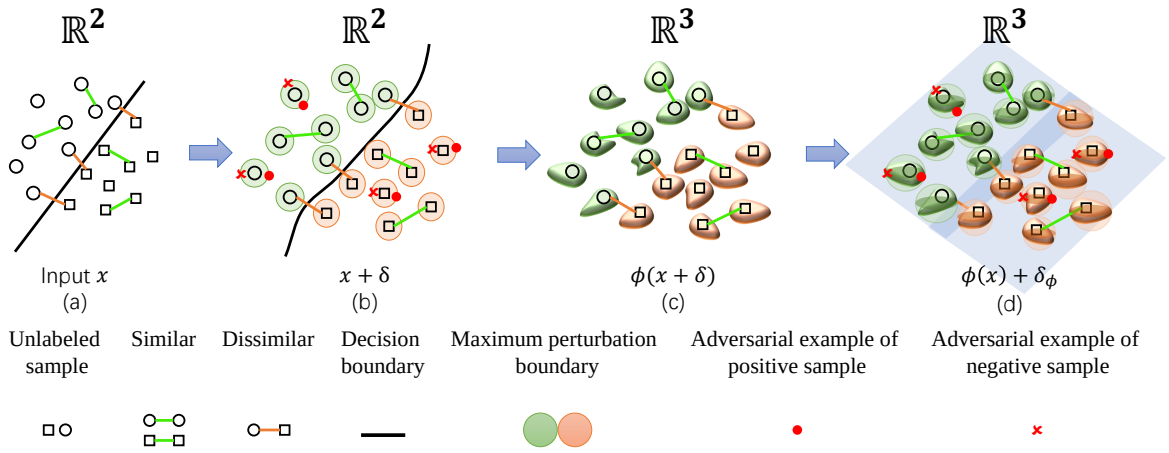


Figure 1: Conceptual illustration of perturbations in the linear and kernel spaces. (Note that one unlabeled sample has two adversarial samples.)

due to the computational expense involved in generating adversarial samples through the K-step PGD attack. Consequently, its practicality is limited when dealing with large-scale problems. To address this challenge, we introduce a novel SDU adversarial strategy for nonlinear model from a kernel perspective. This approach allows us to convert the original minimax problem into a kernel-based minimization problem, thereby enhancing computational efficiency.

3.1 Adversarial Training for SDU Learning

Existing adversarial training methods commonly employ the minimax strategy (Goodfellow et al., 2014). Utilizing Eq. (2), we can readily derive following optimizaton problem for adversarial training in SDU learning as follows:

$$\begin{aligned}\min_f \ \mathcal{L}(f) \stackrel{\text{def}}{=} & \frac{\pi_D}{\pi_c}\mathbb{E}_{\widetilde{x}_D}[M_1] + \frac{\pi_S(1-\gamma)}{\pi_c}\mathbb{E}_{\widetilde{x}_S}[M_2] \\ & + \frac{\gamma}{\pi_c}\mathbb{E}_{x_U}[M_3] \\ s.t.\ & \|\widetilde{x}'_S - \widetilde{x}_S\|_p \le \epsilon;\ \|\widetilde{x}'_D - \widetilde{x}_D\|_p \le \epsilon;\ \|x'_U - x_U\|_p \le \epsilon.\end{aligned} \tag{3}$$

where $\epsilon > 0$ denotes the perturbation on data points, M_1, M_2, and M_3 are defined as follows: $M_1 = \gamma_1 \max_{\widetilde{x}'_D}[\ell(f(\widetilde{x}'_D), -1)] - \gamma_2 \max_{\widetilde{x}'_D}[\ell(f(\widetilde{x}'_D), +1)]$, $M_2 = \pi_+ \max_{\widetilde{x}'_S}[\ell(f(\widetilde{x}'_S), +1)] - \pi_- \max_{\widetilde{x}'_S}[\ell(f(\widetilde{x}'_S), -1)]$, $M_3 = \pi_+ \max_{x'_U}[\ell(f(x'_U), +1)] - \pi_- \max_{x'_U}[\ell(f(x'_U), -1)]$.

The inner maximization problem actually follows the principle of adversarial attack. Its objective is to amplify loss function maximization, thereby crafting the most potent adversarial samples. In essence, this involves seeking a model, denoted as f, that effectively minimizes the value of $\mathcal{L}(f)$.

3.2 Primary Results from the Kernel Perspective

In this subsetion, we first build some primary results for the adversarial training from the kernel perspective. Considering a kernel $k(x, x')$ and its associated Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$, our objective is to identify a function $f^* \in \mathcal{H}$ that resolves the ensuing minimization problem:

$$\min_{f\in\mathcal{H}} \quad \frac{\lambda}{2}\|f\|^2_{\mathcal{H}} + \mathcal{L}(f) \tag{4}$$

where $\lambda > 0$ is the regularization parameter and $\|\cdot\|_{\mathcal{H}}$ denotes the norm on RKHS. In Figure 1 (a) and Figure 1 (b), we illustrate the perturbation δ added to the data samples in the linear space. We can see that a more complicated decision boundary is needed to separate them even if the dataset is linearly separable. Moreover, when the adversarial samples $x+\delta$ are mapped into the kernel space with the kernel mapping $\phi(\cdot)$, $\phi(x+\delta)$ will become unpredictable like Figure 1, which significantly increases the difficulty of data processing and computation. Fortunately, a significant breakthrough was achieved by Xu et al. (2009), who established a theorem establishing the relationship between perturbations in the linear and the kernel space. The theorem can be expressed as follows:

Theorem 1. *Xu et al. (2009) Suppose the kernel function has the form $k(x, x') = h(\|x - x'\|_2)$, with $h : \mathbb{R}^+ \to \mathbb{R}$, a decreasing function. Denote by $\mathcal{H}$ the RKHS space of $k(\cdot,\cdot)$ and $\phi(\cdot)$ the corresponding feature mapping. Then we have for any $x \in \mathbb{R}^d$, $\omega \in \mathcal{H}$, $\epsilon > 0$, and $h_1(\cdot) = \sqrt{2h(0) - 2h(\cdot)}$,*

$$\sup_{\|\delta\|_2\le\epsilon} \langle\omega, \phi(x+\delta)\rangle_{\mathcal{H}} \le \sup_{\|\delta_\phi\|_2\le h_1(\epsilon)} \langle\omega, \phi(x) + \delta_\phi\rangle_{\mathcal{H}}. \tag{5}$$

Since we can use a l_2-norm ball to wrap a l_p-norm ball, e.g., $\{\|\delta\|_\infty \leq c\} \subseteq \{\|\delta\|_2 \leq \sqrt{2}c\}$, Theorem 1 is applicable to other norms as well as follows:

$$\sup_{\|\delta\|_1 \leq c} \langle \omega, \phi(x+\delta) \rangle \leq \sup_{\|\delta_\phi\|_2 \leq h_1(c)} \langle \omega, \phi(x) + \delta_\phi \rangle, \quad (6)$$

$$\sup_{\|\delta\|_\infty \leq c} \langle \omega, \phi(x+\delta) \rangle \leq \sup_{\|\delta_\phi\|_2 \leq h_1(\sqrt{2}c)} \langle \omega, \phi(x) + \delta_\phi \rangle. \quad (7)$$

The perturbation range of $\phi(x) + \delta_\phi$ closely matches $\phi(x+\delta)$, as Figure 1 (d) in the appendix shows. We use $\phi(x)+\delta_\phi$ for subsequent computations, simplifying perturbation handling in kernel space. The objective function (4) is reformulated accordingly.

$$\begin{aligned} \min_{f \in \mathcal{H}} \quad & \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2 + \frac{\pi_D}{\pi_c}\mathbb{E}_{\widetilde{x}_D}[\gamma_1 m\ell_{\Phi_2}(\widetilde{x}_D) - \gamma_2 m\ell_{\Phi_1}(\widetilde{x}_D)] \\ & + \frac{\pi_S(1-\gamma)}{\pi_c}\mathbb{E}_{\widetilde{x}_S}[\mathcal{E}(\widetilde{x}_S)] + \frac{\gamma}{\pi_c}\mathbb{E}_{x_U}[\mathcal{E}(x_U)] \\ s.t. \quad & \|\mathcal{S}(\widetilde{x}_S)\|_2 \leq \epsilon', \|\mathcal{S}(\widetilde{x}_D)\|_2 \leq \epsilon', \|\mathcal{S}(x_U)\|_2 \leq \epsilon'. \end{aligned} \quad (8)$$

where $m\ell_{\Phi_1}(\cdot) = \max_{\Phi(\cdot)}[\ell(\langle f, \Phi(\cdot)\rangle_{\mathcal{H}}, +1)]$, $m\ell_{\Phi_2}(\cdot) = \max_{\Phi(\cdot)}[\ell(\langle f, \Phi(\cdot)\rangle_{\mathcal{H}}, -1)]$, $\Phi(\widetilde{x}_S) = \phi(\widetilde{x}_S) + \delta_\Phi^S$, $\Phi(\widetilde{x}_D) = \phi(\widetilde{x}_D) + \delta_\Phi^D$, $\Phi(x_U) = \phi(x_U) + \delta_\Phi^U$, $\mathcal{E}(\cdot) = \pi_+ m\ell_{\Phi_1}(\cdot) - \pi_- m\ell_{\Phi_2}(\cdot)$, $\mathcal{S}(\cdot) = \Phi(\cdot) - \phi(\cdot)$, and ϵ' is $\sqrt{2h(0) - 2h(\epsilon)}$.

However, Eq. (8) is still a minimax problem that is hard to be solved. To efficiently solve this problem, we further relax the objective function by Theorem 2. The theorem is defined as follows:

Theorem 2. *If f is a function in an RKHS $\mathcal{H}$, the inner maximization problem $\max_{\Phi(\widetilde{x}_S)} \ell(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1)$ is equivalent to the regularized loss function $\ell(f(\widetilde{x}_S) - \epsilon'\|f\|_{\mathcal{H}}, +1)$.*

Detailed proof can be found in our appendix. Based on this theorem, the minimax problem is equivalent to the following minimization problem:

$$\begin{aligned} \min_{f \in \mathcal{H}} \quad & \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2 + \frac{\pi_D}{\pi_c}\mathbb{E}_{\widetilde{x}_D}[\gamma_1 \ell_2(\widetilde{x}_D) - \gamma_2 \ell_1(\widetilde{x}_D)] \\ & + \frac{\pi_S(1-\gamma)}{\pi_c}\mathbb{E}_{\widetilde{x}_S}[\mathcal{E}(\widetilde{x}_S)] + \frac{\gamma}{\pi_c}\mathbb{E}_{x_U}[\mathcal{E}(x_U)] \end{aligned} \quad (9)$$

where $\ell_1(\cdot) = \ell(f(\cdot) - \epsilon'\|f\|_{\mathcal{H}}, +1)$, and $\ell_2(\cdot) = \ell(f(\cdot) - \epsilon'\|f\|_{\mathcal{H}}, -1)$, $\mathcal{E}(\cdot) = \pi_+\ell_1(\cdot) - \pi_-\ell_2(\cdot)$. Obviously, solving problem (9) does not need the K-step PGD attack for generating adversarial samples which will naturally reduce the training time.

3.3 Quadruply Stochastic Gradient Algorithm

In this subsection, we discuss our method to solve the problem (9). Using the kernel method will highly increase the computational complexity. To solve this problem, we propose our quadruply stochastic gradient method based on Random Fourier features (Rahimi and Recht, 2007).

A natural method to solve the problem is using the following full gradient to update the function f:

$$\begin{aligned} \nabla R^\gamma(f) = \lambda f &+ \pi_S(1-\gamma)e_c I_1 \left[k(\widetilde{x}_S, \cdot) - \frac{\epsilon' f(\cdot)}{\|f\|_{\mathcal{H}}}\right] \\ &- \pi_D e_c I_2 \left[k(\widetilde{x}_D, \cdot) - \frac{\epsilon' f(\cdot)}{\|f\|_{\mathcal{H}}}\right] \\ &+ \gamma e_c I_3 \left[k(x_U, \cdot) - \frac{\epsilon' f(\cdot)}{\|f\|_{\mathcal{H}}}\right], \end{aligned} \quad (10)$$

where $e_c = \frac{1}{\pi_+ - \pi_-} = \frac{1}{\pi_c}$, $I_1 = \pi_+\ell'(z)|_{z=f(\widetilde{x}_S)-\epsilon'\|f\|_{\mathcal{H}}} + \pi_-\ell'(z)|_{z=-f(\widetilde{x}_S)+\epsilon'\|f\|_{\mathcal{H}}}$, $I_2 = \gamma_1\ell'(z)|_{z=-f(\widetilde{x}_D)+\epsilon'\|f\|_{\mathcal{H}}} + \gamma_2\ell'(z)|_{f(\widetilde{x}_D)-\epsilon'\|f\|_{\mathcal{H}}}$, $I_3 = \pi_+\ell'(z)|_{z=f(x_U)-\epsilon'\|f\|_{\mathcal{H}}} + \pi_-\ell'(z)|_{z=-f(x_U)+\epsilon'\|f\|_{\mathcal{H}}}$.

However, using the full gradient method is not practical for the large-scale problem. To solve this problem, a natural method is to use the stochastic gradient method. Specifically, we randomly sample one point $\widetilde{x}_S$ from similar dataset $\widetilde{\mathcal{D}}_S$, one point $\widetilde{x}_D$ from dissimilar dataset $\widetilde{\mathcal{D}}_D$, and another point x_U from D_U in each iteration. Then the stochastic functional gradient with these three data points can be achieved as follows: $\xi(\cdot) = e_c[\pi_S(1-\gamma)I_1 k(\widetilde{x}_S, \cdot) - \pi_D I_2 k(\widetilde{x}_D, \cdot) + \gamma I_3 k(x_U, \cdot)]$.

Algorithm 1 $\{\alpha_i\}_{i=1}^t = \text{QSG-ATSDU}(\widetilde{x}_S, \widetilde{x}_D, x_U, \omega)$

Input: $p(\omega), \phi_\omega(\widetilde{x}_S), \phi_\omega(\widetilde{x}_D), \phi_\omega(x_U), \lambda, \pi_S, \pi_D, e, \gamma, \eta, I_1, I_2, I_3$.
Output: $\{\alpha_i\}_{i=1}^t$.

```
for i= 1,··· ,t do
    Sample (x̃_S) from D̃_S, Sample (x̃_D) from D̃_D, Sample x_U from D_U.
    Sample ω_i ∼ p(ω) with seed i.
    f(x_i) =Predict(x_i, {α_i}_{j=1}^{i-1}).
    α_i = −η_i e_c[π_S(1 − γ)I_1 φ_{ω_i}(x̃_S) − π_D I_2 φ_{ω_i}(x̃_D) + γI_3 φ_{ω_i}(x_U)].
    C = λ − (ε' e_c / ‖f‖_H)[π_S (1 − γ) I_1 − π_D I_2 + γI_3].
    α_j = α_j(1 − η_j C) for j = 1,··· ,i − 1.
end for
```

Algorithm 2 $f(x) = \text{Predict } (x, \{\alpha_i\}_{i=1}^t)$

Input: $p(\omega), \phi_\omega(x)$.
Output: $f(x)$.

```
Set f(x) = 0.
for i = 1,··· ,t do
    Sample ω_i ∼ p(ω) with seed i.
    f(x) = f(x) + α_i φ_{ω_i}(x).
end for
```

3.3.1 Random Feature Approximation

Since using kernel functions directly is costly, we use the random feature approximation method Rahimi and Recht (2007) to approximate the kernels. Assume that there exists a *continuous, real-valued, symmetric*, and *shift-invariant* kernel function $k(x, x')$. According to Bochner Theorem (Rudin, 2017), for any shift-invariant kernel $k(x, x') = k(x - x')$, we have $k(x, x') = \int p(\omega) e^{j\omega^\top (x-x')} d\omega$, where $p(\omega)$ is a density function associated with kernel $k(x, x')$ and it can be regarded as the distribution density of ω. Since the distribution density $p(\omega)$ and $k(x - x')$ is real, the integrand $e^{j\omega^\top (x-x')}$ can be replaced with $\cos \omega(x - x')$. Then we can obtain a real-valued feature map $\phi_{\omega_i}(x) = [\cos \omega_i^\top x, \sin \omega_i^\top x]^\top$, where ω_i is randomly sampled according to the distribution density $p(\omega)$. This leads us to acquire the feature map encompassing m random features from a real-valued kernel: $\phi_\omega(x) = \sqrt{\frac{1}{m}}[\cos \omega_1^\top x, \cdots, \cos \omega_m^\top x, \sin \omega_1^\top x, \cdots, \sin \omega_m^\top x]^\top$.

Obviously, $\phi_\omega^\top(x)\phi_\omega(x')$ is an unbiased estimate of $k(x, x')$. Subsequently, We can approximate $\xi(\cdot)$ as follows: $\zeta(\cdot) = e_c[\pi_S(1 - \gamma)I_1\phi_\omega(\widetilde{x}_S, \cdot)\phi_\omega(\cdot) - \pi_D I_2\phi_\omega(\widetilde{x}_D, \cdot)\phi_\omega(\cdot) + \gamma I_3\phi_\omega(x_U, \cdot)\phi_\omega(\cdot)]$. Note that we have $\xi(\cdot) = \mathbb{E}[\zeta(\cdot)]$, which means $\zeta(\cdot)$ is the unbiased estimation of $\xi(\cdot)$. As we randomly sample four variables, i.e., $\widetilde{x}_S, \widetilde{x}_D, x_U, \omega$, we can call our functional gradient $\zeta(\cdot)$ as the quadruply stochastic functional gradient.

3.3.2 Updating Rules

For convenience, the function value is expressed as $h(x)$ if updated by using the exact kernel function and is expressed as $f(x)$ if updated by using random Fourier features. We abbreviate the full gradient of the objective function by using $\xi(\cdot)$ and $\zeta(\cdot)$ as follows:

$$\nabla R^\gamma(h) = \mathbb{E}_S\mathbb{E}_D\mathbb{E}_U[\xi(\cdot)] + Ch(\cdot), \tag{11}$$

$$\nabla R^\gamma(f) = \mathbb{E}_S\mathbb{E}_D\mathbb{E}_U[\mathbb{E}_\omega[\zeta(\cdot)]] + Cf(\cdot), \tag{12}$$

where $C = \lambda - \frac{\epsilon' e_c}{\|f\|_\mathcal{H}}[\pi_S(1 - \gamma)I_1 - \pi_D I_2 + \gamma I_3]$.

Let $h_1(\cdot) = f_1(\cdot) = 0$. Subsequently, we present the update rules using the true stochastic functional gradient $\xi(\cdot)$ at t-th iteration as follows: $h_{t+1}(\cdot) = h_t(\cdot) - \eta_t(\xi(\cdot) + Ch_t(\cdot)) = \sum_{i=1}^t a_t^i \xi_i(\cdot)$, where η_t is the stepsize in the t-th iteration, the initial value $f_1(\cdot) = 0$, the value of a_t^i can be inferred as $a_t^i = -\eta_i \prod_{j=i+1}^t \{1 - \eta_j C\}$.

Since $\zeta(\cdot)$ is an unbiased etimation of $\xi(\cdot)$, the update rule by using $\zeta(\cdot)$ is similar to that by using $\xi(\cdot)$. So the update rule at t-th iteration by using $\zeta(\cdot)$ is $f_{t+1}(\cdot) = f_t(\cdot) - \eta_t(\zeta(\cdot) + Cf_t(\cdot)) = \sum_{i=1}^t a_t^i \zeta_i(\cdot)$. In order to implement the update rules in the computer program, we rewrite the update rule as the iterative update rules with constantly-changing coefficients $\{\alpha_i\}_{i=1}^t$, $f_t(x) = \sum_{i=1}^t \alpha_i \phi_\omega(x)$, $\alpha_i = -\eta_i e_c[\pi_S(1 - \gamma)I_1\phi_{\omega_i}(\widetilde{x}_S) - \pi_D I_2\phi_{\omega_i}(\widetilde{x}_D) + \gamma I_3\phi_{\omega_i}(x_U)]$, and for $j = 1, \cdots, i - 1$, $\alpha_j = [1 - \eta_j C]\alpha_j$.

3.4 Algorithm

Following the update rules, we present the training and prediction algorithms of SDU adversarial training on kernel SVM in Algorithms 1 and 2. Since our method contains four random sources, i.e., similar dataset, dissimilar dataset, unlabeled dataset and random features, we call our method the Quadruply Stochastic Gradient Method for Adversarial Training of SDU learning (QSG-ATSDU).

4 CONVERGENCE ANALYSIS

In this section, we first give some reasonable assumptions for later inferences and then show the convergence rate of SDU adversarial training for estimating the optimal function in the RKHS $\mathcal{H}$. Detailed proof can be found in our appendix.

Assumption 1. *(Bound of kernel function) There exists $\kappa > 0$, such that $k(x, x') \leq \kappa$.*

Assumption 2. *(Bound of random feature norm) There exists $\phi > 0$, such that $|\phi_\omega(x)\phi_\omega(x')| \leq \phi$.*

Assumption 3. *(Bound of derivation) The derivative of ℓ w.r.t the first term u is bounded: $|\ell'(u, v)| < M$.*

Assumption 4. *(Lipschitz continuous) ℓ is L-**Lipschiz continuous.***

Assumption 5. *(Lipschitz gradient) The gradient function $\nabla R^\gamma(f)$ is Lipschitz continuous such that $\|\nabla R^\gamma(f) - \nabla R^\gamma(g)\|_\mathcal{H} \leq \widetilde{L}\|f - g\|_\mathcal{H}, \forall f, g \in \mathcal{H}$.*

Assumption 6. *The spectral radius $\rho(f)$ of a function $f(\cdot)$ has a lower bound that $\rho(f) \geq \epsilon' \geq 0$, where a spectral radius is the maximum moudulus of eigenvalues Mason and Shorten (2007), i.e., $\rho(f) = \max_{1 \leq i \leq \infty}\{\sqrt{|\lambda_i|}\}$.*

Assumption 7. *The objective function $R^\gamma(h_t)$ is bounded below by a scalar R_{inf}^γ.*

These assumptions, crucial for convergence analysis, are referenced in (Mason and Shorten, 2007; Shi et al., 2019, 2021). Due to possible non-convexity in the objective function, we aim to bound $\mathbb{E}[\|\nabla R^\gamma(f_t)\|_\mathcal{H}^2]$. However, with f_t approximating h_t via random Fourier features, it may not reside in RKHS $\mathcal{H}$. Thus, we initially constrain $\mathbb{E}[\|\nabla R^\gamma(h_t)\|_\mathcal{H}^2] < \epsilon_1$ (a small constant) using the exact kernel, then show f_t's proximity to h_t at stationary points. Relevant lemmas for this analysis are detailed in the appendix, and these, along

with our assumptions, underpin the convergence rate theorem we propose.

Theorem 3. *(Convergence in expectation) For a fixed step size* $\eta_t = \overline{\eta} = \frac{\theta}{T^{3/4}}$ *and* $0 < \theta \leq 1$*, we have that*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\|\nabla R^{\gamma}(h_t)\|_{\mathcal{H}}^2\right] \leq \frac{R^{\gamma}(h_1) - R^{\gamma}_{inf}}{\theta T^{1/4}} + \frac{\widetilde{L}\theta}{2T^{3/4}}G_1 + \frac{\theta}{T^{1/4}}G_2 \tag{13}$$

where $G_1 = (1+C)^2 G^2 {e_c}^2 M^2 \kappa$, $G_2 = (1+C)G^3|e_c|^3 M^2 L\kappa(\sqrt{\kappa}+\sqrt{\phi})$*, and* $G = [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]$.

Remark 1. *Even when considering four sources of randomness, this theorem suggests that our approach can achieve convergence to the stationary point at a rate of* $\mathcal{O}(1/T^{1/4})$ *for any given data* x.

5 EXPERIMENT

In this section, we conduct experiment to verify the robustness and efficiency of SDU adversarial training.

5.1 Experimental Setup

5.1.1 Datasets

The experiments on binary classification are conducted on six datasets come from UCI (Asuncion and Newman, 2007) and LIBSVM (Chang and Lin, 2011). Since we focus on binary classification, we select two similar classes to construct the binary classification dataset, like MNIST and CIFAR. We summarize the dataset used in our experiments in Table 2.

Table 2: Datasets used in the experiments.

Dataset	Features	Sizes
a9a	123	48,842
CIFAR10 automobile vs. truck	3,072	20,000
MNIST 6 vs. 8	780	20,000
Acoustic	50	98,528
Combined	100	98,528
Covertype	54	581,012

5.1.2 Compared Methods

In this section, we assess the robustness and training time of the state-of-the-art robust similarity learning algorithms. The summarized methods under consideration are as follows:

1. ***SU-SL***: The method introduced in Bao et al. (2018), learn a classifier from similar and unlabeled data. This method employs squared loss, defined as $\ell(u,v) = \frac{1}{4}(uv-1)^2$.
2. ***SDU-DH***: Proposed in Shimada et al. (2021), this method utilizes double hinge loss, given by $\ell(u,v) = \max\{-uv, \max\{0, \frac{1}{2} - \frac{1}{2}uv\}\}$.
3. ***nSU***: Presented in Wu et al. (2022), this method focuses on learning from noisy similar (nS) data and unlabeled (U) data.
4. ***SGD-SDU (Ours)***: Our algorithm, employing Stochastic Gradient Descent (SGD) Wei and Li (2018) and kernel methods, is specifically designed for our robust SDU classification, i.e., objective function (9). Refer to the appendix for details on the specific algorithm.
5. ***QSG-ATSDU (Ours)***: Our proposed algorithm involves kernelized adversarial training for SDU learning based on a doubly stochastic gradient framework Dai et al. (2014).

5.1.3 Implementation

In addition, SU-SL, SDU-DH, nSU, SGD-SDU, and QSG-ATSDU, we use Gaussian RBF kernel, $k(x,x') = \exp(-\sigma\|x-x'\|^2)$ to build the nonlinear model. We implement all method in Python. We run all the methods for $T = 100$ iterations and $\epsilon = 0.3$. The batch size for similar data is determined as $\frac{n_S}{T}$, while the batch size for dissimilar data is computed as $\frac{n_D}{T}$. Correspondingly, the batch size for unlabeled data is evaluated as $\frac{n_U}{T}$. The value for hyper-parameters (λ, γ, and σ) are selected over $\gamma \in \{0.1, 0.2, \cdots, 1.0\}, \sigma \in \{2^{-10}, 2^{-9}, \cdots, 2^{10}\}, \lambda \in \{2^{-10}, 2^{-9}, \cdots, 2^{10}\}$, via 5-fold cross-validation. Precisely, the similar dataset, dissimilar dataset, and unlabeled dataset were each evenly divided into five segments. Subsequently, one segment was designated as the test set, while the remaining four segments were combined to create the training set. For generating similar and dissimilar pairs from labeled data, we first establish the positive prior π_+. The positive class prior π_+, set at 0.7, reflects the class distribution in the entire training dataset, and estimation methods like Lee and Liu (2003); Blanchard et al. (2010); du Plessis et al. (2015) can be used. Then we randomly extract pairwise similar and dissimilar data, ensuring a ratio as defined by π_S and π_D, respectively. These ratios are calculated based on π_+.

Notice that all experiments were run on a PC with 36 2.2 GHz cores and 80GB RAM. All the experiments were run at 10 times and all the results are the average values.

5.2 Results and Discussions

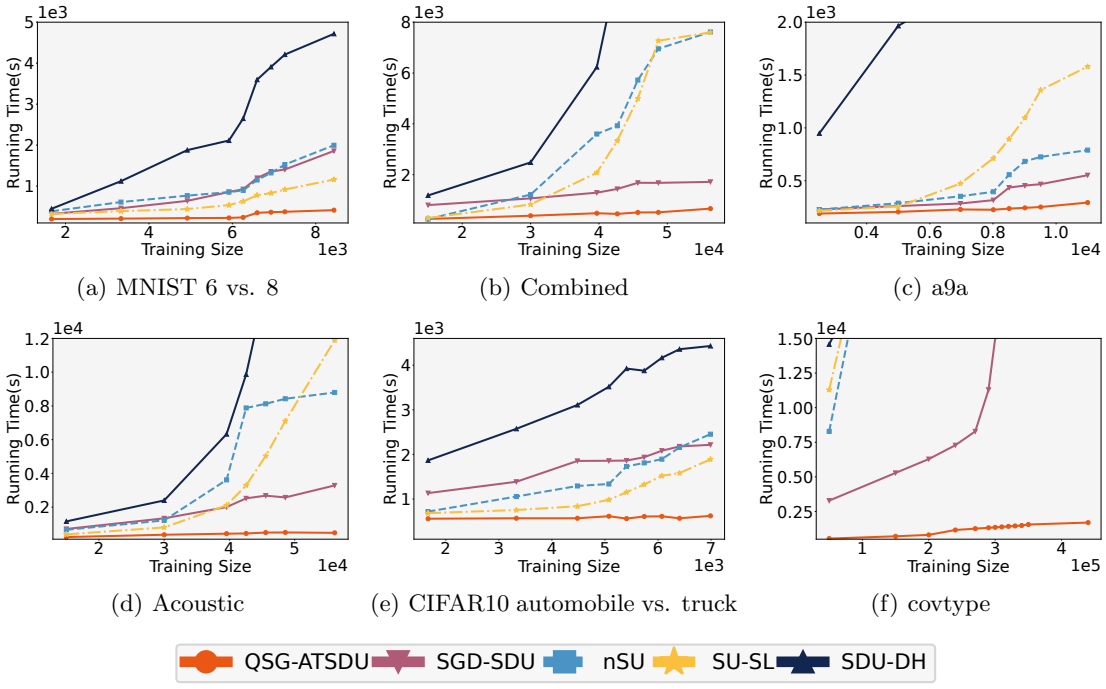
We evaluate the test accuracy of different methods against various attacks, including FGSM (Goodfellow

Table 3: Accuracy on MNIST and CIFAR10 test with various attacks and perturbations constrained by l_2-norm. (- denotes out of memory; † denotes our algorithm.)

	MNIST 6 vs. 8				CIFAR10 automobile vs. truck			
	clean	FGSM	PGD	CW	clean	FGSM	PGD	CW
SU-SL	93.94	50.7	48.65	43.31	**72.64**	59.37	54.77	51.17
SDU-DH	83.45	63.4	59.4	52.1	70.34	56.21	52.38	47.81
nSU	79.08	69.2	70	53.8	71.14	58.36	56.2	55.01
SGD-SDU†	**94.73**	91.88	72.44	65.13	72.5	68.2	67.9	70.24
QSG-ATSDU†	94.07	**93.27**	**75.37**	**69.81**	70.68	**69.95**	**69.51**	**70.38**

Table 4: Accuracy on a9a and Acoustic test with various attacks and perturbations constrained by l_2-norm. (- denotes out of memory; † denotes our algorithm.)

	a9a				Acoustic			
	clean	FGSM	PGD	CW	clean	FGSM	PGD	CW
SU-SL	**83.09**	61.62	60	53.36	**72.07**	59.58	58.25	48.31
SDU-DH	71.23	52.03	51.19	49.87	-	-	-	-
nSU	72.2	64.07	56.9	56.78	70.24	65.6	61.07	55.1
SGD-SDU†	**75**	74.02	72.55	63.66	71.8	67.87	65.93	63.79
QSG-ATSDU†	78.45	**77.33**	**77.19**	**69.99**	**73.35**	**71.08**	**70.72**	**69.97**



(a) MNIST 6 vs. 8 (b) Combined (c) a9a

(d) Acoustic (e) CIFAR10 automobile vs. truck (f) covtype

Figure 2: The training time against the different number of unlabeled samples. (The line of SDU-DH is incomplete because its training time is more than 2000 seconds.)

Table 5: Accuracy on Combined and Covtype test with various attacks and perturbations constrained by l_2-norm. (- denotes out of memory; † denotes our algorithm.)

	Combined				Covtype			
	clean	FGSM	PGD	CW	clean	FGSM	PGD	CW
SU-SL	**77.31**	66.03	63.05	58.48	**-**	-	-	-
SDU-DH	-	-	-	-	-	-	-	-
nSU	76	67.1	64.27	59.06	-	-	-	-
SGD-SDU†	**76.07**	75.93	75.34	74.29	-	-	-	-
QSG-ATSDU†	**80.59**	**78.36**	**76.78**	**74.52**	**73.68**	**71.02**	**70.6**	**67.86**

et al., 2014), 10-step PGD, and C&W (Chen et al., 2017), all under l_2-norm constrained perturbations. The summarized outcomes are presented in Tables 3, 4, 5, and Figure 2. Our approach offers several notable advantages over the results.

Firstly, our approach demonstrates superior defense performance against these attacks. It consistently maintains high accuracy in the presence of different attack methods, all of which yield higher accuracy than other comparative algorithms. Notably, it's important to mention that the kernel map calculation in SDU-DH requires additional memory, which leads to potential memory overflow for certain datasets. Similarly, memory constraints become apparent for SU-SL, nSU, and SGD-SDU when the training dataset size exceeds 400,000.

Additionally, QSG-ATSDU showcases significantly faster execution times in comparison to the adversarial training method SGD-SDU and the noisy SU learning method nSU, as illustrated in Figure 2. QSG-ATSDU's utilization of the quadruply stochastic algorithm accelerates processing by solely requiring a random seed for generating random features, unlike SU-SL, which necessitates kernel map calculations for each sampled data point. We've reformulated the adversarial training minimax problem into a minimization problem, enabling direct resolution through stochastic gradient descent. This eliminates the need for additional perturbation discovery steps in each iteration, a requirement of traditional adversarial training approaches. Our method's efficiency is further enhanced by surpassing SGD-SDU, which demands $\mathcal{O}(bn)$ complexity for calculating the kernel map for a batch of size b, with n representing the entire training set size. In contrast, our method requires only $\mathcal{O}(Tm)$ complexity for feature map calculation ($m < n$), where m is the number of random features.

Finally, a comprehensive analysis of QSG-ATSDU under various parameter configurations is available in the appendix.

6 CONCLUSION

In this paper, an effcient and scalable adversarial training method for SDU learning is proposed. We propose a new minimization objective for adversarial training of SDU learning based on the kernel method, which can naturally ignore the multiple steps of PGD to update the perturbation. We also propose an effcient algorithm using the random Fourier features and stochastic gradient method to solve our new algorithms. We prove that QSG-ATSDU has a convergence rate of $\mathcal{O}(1/T^{1/4})$. The experimental results on various datasets demonstrate the superiority of our proposed algorithms over exiting SDU learning algorithms.

Acknowledgments

Bin Gu was supported by the Natural Science Foundation of China under Grant 62076138. Yi Chang was partially supported by the Natural Science Foundation of China under Grant U2341229, as well as by the Ministry of Science and Technology Key R&D Project of China under the number 2023YFF0905400.

References

Arnab, A., Miksik, O., and Torr, P. H. (2018). On the robustness of semantic segmentation models to adversarial attacks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 888–897.

Asuncion, A. and Newman, D. (2007). Uci machine learning repository.

Bao, H., Niu, G., and Sugiyama, M. (2018). Classification from pairwise similarity and unlabeled data. In *International Conference on Machine Learning*, pages 452–461. PMLR.

Bao, H., Shimada, T., Xu, L., Sato, I., and Sugiyama, M. (2022). Pairwise supervision can provably elicit a decision boundary. PMLR.

Blanchard, G., Lee, G., and Scott, C. (2010). Semi-

supervised novelty detection. *Journal of Machine Learning Research*, 11(Nov):2973–3009.

Cao, Y., Feng, L., Xu, Y., An, B., Niu, G., and Sugiyama, M. (2021). Learning from similarity-confidence data. In *International Conference on Machine Learning*, pages 1272–1282. PMLR.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27.

Chapelle, O., Schölkopf, B., and Zien, A. (2010). Semi-supervised learning. adaptive computation and machine learning. *MIT Press, Cambridge, MA, USA. Cited in page (s)*, 21(1):2.

Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. (2017). Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26.

Dai, B., Xie, B., He, N., Liang, Y., Raj, A., Balcan, M.-F. F., and Song, L. (2014). Scalable kernel methods via doubly stochastic gradients. In *Advances in Neural Information Processing Systems*, pages 3041–3049.

Dan, S., Bao, H., and Sugiyama, M. (2021). Learning from noisy similar and dissimilar data. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*, pages 233–249. Springer.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. (2018). Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193.

Du Plessis, M., Niu, G., and Sugiyama, M. (2015). Convex formulation for learning from positive and unlabeled data. In *International conference on machine learning*, pages 1386–1394. PMLR.

du Plessis, M. C., Niu, G., and Sugiyama, M. (2015). Class-prior estimation for learning from positive and unlabeled data. In *ACML*, pages 221–236.

Feng, L., Shu, S., Cao, Y., Tao, L., Wei, H., Xiang, T., An, B., and Niu, G. (2023). Multiple-instance learning from unlabeled bags with pairwise similarity. *IEEE Transactions on Knowledge and Data Engineering*.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Klein, D., Kamvar, S. D., and Manning, C. D. (2002). From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *ICML*, volume 2, pages 307–314.

Kurakin, A., Goodfellow, I. J., and Bengio, S. (2018). Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC.

Lee, W. S. and Liu, B. (2003). Learning with positive and unlabeled examples using weighted logistic regression. In *ICML*, volume 3, pages 448–455.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Maheshwara, S. S. and Manwani, N. (2023). Rolnip: Robust learning using noisy pairwise comparisons. In *Asian Conference on Machine Learning*, pages 706–721. PMLR.

Mason, O. and Shorten, R. (2007). On linear copositive lyapunov functions and the stability of switched positive linear systems. *IEEE Transactions on Automatic Control*, 52(7):1346–1349.

Mosbach, M., Andriushchenko, M., Trost, T., Hein, M., and Klakow, D. (2018). Logit pairing methods can fool gradient-based attacks. *arXiv preprint arXiv:1810.12042*.

Nederhof, A. J. (1985). Methods of coping with social desirability bias: A review. *European journal of social psychology*, 15(3):263–280.

Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20.

Rudin, W. (2017). *Fourier analysis on groups*. Courier Dover Publications.

Sakai, T., Plessis, M. C., Niu, G., and Sugiyama, M. (2017). Semi-supervised classification based on classification from positive and unlabeled data. In *International conference on machine learning*, pages 2998–3006. PMLR.

Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. (2019). Adversarial training for free! *Advances in Neural Information Processing Systems*, 32.

Shi, W., Gu, B., Li, X., Deng, C., and Huang, H. (2021). Triply stochastic gradient method for large-scale nonlinear similar unlabeled classification. *Machine Learning*, 110:2005–2033.

Shi, W., Gu, B., Li, X., Geng, X., and Huang, H. (2019). Quadruply stochastic gradients for large

scale nonlinear semi-supervised auc optimization. *arXiv preprint arXiv:1907.12416*.

Shimada, T., Bao, H., Sato, I., and Sugiyama, M. (2021). Classification from pairwise similarities/dissimilarities and unlabeled data via empirical risk minimization. *Neural Computation*, 33(5):1234–1268.

Song, D., Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Tramer, F., Prakash, A., and Kohno, T. (2018). Physical adversarial examples for object detectors. In *12th USENIX workshop on offensive technologies (WOOT 18)*.

Su, J., Vargas, D. V., and Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014*.

Tramer, F. and Boneh, D. (2019). Adversarial training and robustness for multiple perturbations. *Advances in Neural Information Processing Systems*, 32.

Warner, S. L. (1965). Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69.

Wei, H. and Li, M. (2018). Positive and unlabeled learning for detecting software functional clones with adversarial training. In *IJCAI*, pages 2840–2846.

Wong, E., Rice, L., and Kolter, J. Z. (2020). Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*.

Wu, S., Liu, T., Han, B., Yu, J., Niu, G., and Sugiyama, M. (2022). Learning from noisy pairwise similarity and unlabeled data. *Journal of Machine Learning Research*, 23(307):1–34.

Wu, S., Xia, X., Liu, T., Han, B., Gong, M., Wang, N., Liu, H., and Niu, G. (2020). Multi-class classification from noisy-similarity-labeled data. *arXiv preprint arXiv:2002.06508*.

Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., and Yuille, A. (2017). Adversarial examples for semantic segmentation and object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1369–1378.

Xu, H., Caramanis, C., and Mannor, S. (2009). Robustness and regularization of support vector machines. *Journal of machine learning research*, 10(7).

Yang, Y., Zhang, G., Katabi, D., and Xu, Z. (2019). Me-net: Towards effective adversarial robustness with matrix estimation. *arXiv preprint arXiv:1905.11971*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [No]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

(d) Information about consent from data providers/curators. [Not Applicable]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary of "Fast and Adversarial Robust Kernelized SDU Learning"

The appendix contains several additional results that were excluded from the main body of the paper due to space constraints, along with the proof process of the lemma and solutions. The organization of the appendix is as follows:

Part A RELATED WORK:

Explores the landscape of adversarial training and similarity learning, emphasizing significant progress and its implications.

Part B Class Prior Estimation: This section provides methods for class prior mation.

Part C PROOF OF ThEOREM 2:

This section provides a detailed proof procedure for Theroem 2.

Part D CONVERGENCE ANALYSIS:

Detailed procedures for demonstrating convergence within the proposed models.

Part E THE SPECIFIC ALGORITHM FOR THE SGD-SDU EXPERIMENT:

Details specific algorithmic procedures for SGD-SDU experiments, enhancing the main text with practical insights.

Part F ADDITIONAL EXPERIMETNS:

Extends the discourse with further experimental insights into adversarial training for SDU Learning, complemented by results from ablation studies, enriching the narrative with empirical evidence and analytical depth.

A RELATE WORK

A.1 Adversarial Training

Adversarial training (AT) is a technique aimed at enhancing the robustness of machine learning models by exposing them to adversarial samples during the training process Goodfellow et al. (2014). This approach has been extensively explored in various supervised classification tasks, including object detection Song et al. (2018); Xie et al. (2017), object segmentation Arnab et al. (2018); Xie et al. (2017) and image classification Goodfellow et al. (2014); Su et al. (2019). Several important methods have been proposed to improve adversarial training.

One notable method introduced by Goodfellow et al. (2014); Su et al. (2019) is the Fast Gradient Sign Method (FGSM), which generates adversarial samples using a single step. This technique perturbs the inputs before updating the models. Building upon FGSM, Goodfellow et al. (2014); Su et al. (2019) enhanced it by adding a randomization step known as R+FGSM. Another improvement came from Kurakin et al. (2018), who proposed the Basic Iterative Method. This method refines FGSM by taking multiple smaller FGSM steps, thereby rendering the earlier FGSM-based adversarial training methods ineffective. Currently, these methods are widely regarded as adversarial training against a projected gradient descent adversary.

Moreover, the PGD attack and its corresponding adversarial training defense have been further augmented with various techniques. Dong et al. (2018) introduced the momentum-based optimization method to enhance the adversary. Additionally, combinations with other heuristic defenses Yang et al. (2019); Mosbach et al. (2018) and generalization to multiple types of adversarial attacks Tramer and Boneh (2019) have been explored to strengthen the PGD attack and improve its corresponding adversarial training defense.

A.2 Similarity Learning

In the context of similarity learning, several algorithms have been proposed. Bao et al. (2018) introduced the SU classification method, which involves using only pairs of similar (S) data points where both examples belong to the same class, along with unlabeled (U) data points, instead of relying on fully labeled data. By utilizing this limited information, the algorithm aims to develop an effective learning approach. To tackle the issue of false information within SU classification, the nSU approach Wu et al. (2022) has been proposed. nSU classification pertains to the creation of a robust and consistent classifier from pairs of noisy similar (nS) data and unlabeled (U) data. This approach effectively enhances the data's resilience against random noise.

Moving forward, Wu et al. (2020) proposed the Multi-class Noisy Similarity (MNS) deep learning system for multi-class classification, addressing the challenge of learning from data labeled with noisy similarity annotations. Notably, MNS not only exhibits a strong ability to generalize to unseen data but also delves into multi-class classification problems, a domain often overshadowed by studies predominantly focusing on binary classification issues.

On the other hand, Cao et al. (2021) introduced a novel Sconf learning framework that incorporates empirical risk minimization (ERM). This framework constructs an unbiased estimator of classification risk by solely utilizing unlabeled data pairs with similarity confidence. The Sconf algorithm employs similarity confidence rather than traditional labels. It provides an unbiased estimator of classification risk and integrates an empirical risk correction scheme to enhance learning performance. Importantly, this approach doesn't make implicit assumptions about models, loss functions, or optimizers in its analysis, rendering it adaptable to both convex and non-convex loss functions, as well as deep and linear models.

Additionally, Maheshwara and Manwani (2023) introduced the RoLNiP approach for learning robust classifiers from noisy pairwise similar-dissimilar data. Unlike altering the loss function, RoLNiP is founded on the principle of robust loss functions.

Furthermore, Dan et al. (2021) addressed the challenge of learning from pairwise supervision, specifically dealing with pairs of similar (S) and dissimilar (D) data, which were provided instead of standard labeled data.

B Class Prior Estimation

Although we have to know the class prior π_+ before training for calculation of empirical risks $\widehat{R}_{SD}$ and $\widehat{R}_{DU}$, π_+ can be estimated from the number of similar pairs n_S and the number of dissimilar pairs n_D. First, π_+ and π_S has following relationship.

$$\pi_+ = \begin{cases} \frac{1+\sqrt{2\pi_S-1}}{2} & (\pi_+ \geq 0.5), \\ \frac{1-\sqrt{2\pi_S-1}}{2} & (\pi_+ < 0.5), \end{cases} \tag{14}$$

The above equality is obtained from $2\pi_S - 1 = \pi_S - \pi_D = (\pi_+ - \pi_-) = (2\pi_+ - 1)^2$. Note that $\widehat{\pi}_S = n_S/(n_S + n_D)$ is an unbiased estimator of π_S. Thus, π_+ can be estimated by plugging $\widehat{\pi}_S$ into Eq. (14).

C PROOF OF THEOREM 2

Proof. Since $\Phi(x) = \phi(x) + \delta_\phi$, the constraint can also be written as $\|\delta_\phi\|_2 \leq \epsilon'$, let $\tau = \{\delta_\phi \mid \|\delta_\phi\|_2 \leq \epsilon'\}$. We define $\nu = l(f(\widetilde{x}_S) - \epsilon'\|f\|_{\mathcal{H}}, +1)$, and $g(x,t) = \max(0, tx)$ is the hinge loss. To prove the theorem, we first prove $\nu \leq \max\, l(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1)$, and then prove $\nu \geq \max\, l(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1)$. In the following, we give the details to prove these two sub-conclusions.

Step 1: We first prove $\nu \leq \max\, l(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1)$.

Since, $\tau = \{\delta_\phi \| \delta_\phi\|_2 \leq \epsilon'\}$, we define two susbsets of τ as $\tau_1' = \{-\epsilon' \frac{f}{\|f\|_{\mathcal{H}}}\}$. Hence,

$$\begin{aligned} &\max_{\delta_\phi^S \in \tau_1'} \; l(\langle f, \phi(\widetilde{x}_S) + \delta_\phi^S\rangle_{\mathcal{H}}, +1) \\ =& \max_{\delta_\phi^S \in \tau_1'} \; l(\langle f, \phi(\widetilde{x}_S)\rangle_{\mathcal{H}} + \langle f, \delta_\phi^S\rangle_{\mathcal{H}}, +1) \\ =& \; l(f(\widetilde{x}_S) - \epsilon'\|f\|_{\mathcal{H}}, +1). \end{aligned} \tag{15}$$

Since $\tau_1' \subseteq \tau$, the first sub-conclusion can be proved.

Step 2: Next we prove $\nu \geq \max\, l(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1)$.

$$\begin{aligned}
&\max_{\delta_\phi^S \in \tau}\ l(\langle f, \phi(\widetilde{x}_S) + \delta_\phi^S\rangle_{\mathcal{H}}, +1) \leq \nu \\
=\ &\max_{\delta_\phi^S \in \tau}\ g(-\langle f, \phi(\widetilde{x}_S)\rangle_{\mathcal{H}} - \langle f, \delta_\phi^S\rangle_{\mathcal{H}}, +1) \\
=\ &g(\max_{\delta_\phi^S \in \tau}\ -\langle f, \phi(\widetilde{x}_S)\rangle_{\mathcal{H}} - \langle f, \delta_\phi^S\rangle_{\mathcal{H}}, +1) \\
\leq\ &g(\max_{\delta_\phi^S \in \tau}\ -\langle f, \phi(\widetilde{x}_S)\rangle_{\mathcal{H}} + \|f\|_{\mathcal{H}}\|\delta_\phi^S\|_2, +1) \\
=\ &l(f(\widetilde{x}_S) - \epsilon'\|f\|_{\mathcal{H}}, +1)
\end{aligned} \tag{16}$$

The first inequality also uses the Cauchy-Schwarz inequality.

Step 3: Combing these two steps, we have :

$$\max_{\|\Phi(\widetilde{x}_S) - \phi(\widetilde{x}_S)\|_2 \leq \epsilon'}\ l(\langle f, \Phi(\widetilde{x}_S)\rangle_{\mathcal{H}}, +1) = l(f(\widetilde{x}_S) - \epsilon'\|f\|_{\mathcal{H}}, +1). \tag{17}$$

□

D CONVERGENCE ANALYSIS

To proof Therom 3, we first give several lemmas and their respective proof useful in our convergence analysis.

Lemma 1. *For all x, for a fixed step size $\eta_t = \overline{\eta} = \frac{\theta}{T^{3/4}}$, $0 < \theta \leq 1$, we have that*

$$\mathbb{E}_{\widetilde{x}_S, \widetilde{x}_D, x_U, \omega}[|f_{T+1} - h_{T+1}|^2] \leq B_{1,T+1}^2,$$

where $B_{1,T+1}^2 = [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2(\sqrt{\phi} + \sqrt{\kappa})^2 \dfrac{\theta^2}{T^{1/2}}$ and $\mathbb{E}_{\widetilde{x}_S, \widetilde{x}_D, x_U, \omega}[\cdot]$ denotes the expectation over the similar dataset, dissimilar dataset, unlabeled dataset and the random features.

Remark 2. *In this leamma, we can find that after T iterations, the function value f will converge to the real function value h.*

D.1 Proof of Lemma 1

Proof. We denote $A_i(x) = A_i(x; \widetilde{x}_S, \widetilde{x}_D, x_U, \omega_i) = a_t^i(\zeta_i(x) - \xi_i(x))$ for the t-th iteration. According to the assumption in section (convergence analysis), $A_i(x)$ have a bound:

$$
\begin{aligned}
|A_i(x)| &\leq |a_t^i|(|\zeta_i(x)| + |\xi_i(x)|) \\
&= |a_t^i|\Big(| - \pi_S (1-\gamma) e_c I_1 \phi_\omega(\widetilde{x}_i^S, x)\phi_\omega(x) \ + \pi_D e_c I_2 \phi_\omega(\widetilde{x}_i^D, x)\phi_\omega(x) - \gamma e_c I_3 \phi_\omega(x_i^U, x)\phi_\omega(x)| \\
&\quad + | - \pi_S(1-\gamma) e_c I_1 k(\widetilde{x}_i^S, x) + \pi_D e_c I_2 k(\widetilde{x}_i^D, x) - \gamma e_c I_3 k(x_i^U, x)| \Big) \\
&\leq |a_t^i|\Big[|\pi_S (1-\gamma) e_c I_1 \phi_\omega(\widetilde{x}_i^S, x)\phi_\omega(x)| + \pi_D e_c I_2 \phi_\omega(\widetilde{x}_i^D, x)\phi_\omega(x)| + |\gamma e_c I_3 \phi_\omega(x_i^U, x)\phi_\omega(x)| \\
&\quad + |\pi_S (1-\gamma) e_c I_1 k(\widetilde{x}_i^S, x)| + |\pi_D e_c I_2 k(\widetilde{x}_i^D, x)| + |\gamma e_c I_3 k(x_i^U, x)| \Big] \\
&\leq |a_t^i|\Big[\pi_S (1-\gamma) |e_c| M\sqrt{\phi} + \pi_D |e_c| (1+\gamma) M\sqrt{\phi} + \gamma |e_c| M\sqrt{\phi} + \pi_S (1-\gamma) |e_c| M\sqrt{\kappa} \\
&\quad + \pi_D |e_c| (1+\gamma) M\sqrt{\kappa} + \gamma |e_c| M\sqrt{\kappa} \Big] \\
&= [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma] |e_c| M(\sqrt{\phi} + \sqrt{\kappa}) |a_t^i| \qquad (18)
\end{aligned}
$$

Then, based on the definition of a_t^t and a fix step size $\eta_t = \overline{\eta}$, we have $a_t^t \leq \overline{\eta}$. Then according to Assumption 6 and the theorem that $\rho(f)$ is the lower bound of any matrix norm of $f(\cdot)$ that $\|f\| \geq \rho(f)$, we can get that $\|f\| \geq \epsilon'$. In addition, for any i we have

$$
|a_t^i| \leq \eta_i \prod_{j=i+1}^{t} \left\{ 1 - \eta_j \big[\lambda + \frac{\epsilon' e_c}{\|f\|_{\mathcal{H}}} (\pi_S (1-\gamma) I_1 - \pi_D I_2 + \gamma I_3) \big] \right\} \leq t\overline{\eta}^2.
$$

Then, for the t-th iteration, we have

$$
\mathbb{E}_{\widetilde{x}_S, \widetilde{x}_D, x_U, \omega}[|f_{t+1}(x) - h_{t+1}(x)|^2] \leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 (\sqrt{\phi} + \sqrt{\kappa})^2 t\overline{\eta}^2. \qquad (19)
$$

Taking the step size $\overline{\eta} = \frac{\theta}{T^{3/4}}$, we have

$$
\begin{aligned}
B_{1,T+1}^2 \ &:= [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 (\sqrt{\phi} + \sqrt{\kappa})^2 t \frac{\theta^2}{T^{3/2}} \\
&\leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 (\sqrt{\phi} + \sqrt{\kappa})^2 T \frac{\theta^2}{T^{3/2}} \\
&\leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 (\sqrt{\phi} + \sqrt{\kappa})^2 \frac{\theta^2}{T^{1/2}} \qquad (20)
\end{aligned}
$$

Thus, for the Tth iteration, we have

$$
\mathbb{E}_{\widetilde{x}_S, \widetilde{x}_D, x_U, \omega}[|f_{T+1} - h_{T+1}|^2] \leq B_{1,T+1}^2,
$$

That completes the proof. □

Lemma 2. *Let us denote* $\mathcal{H}_t = \sqrt{\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2}$, $\mathcal{M}_t = \|g_t\|_{\mathcal{H}}^2$, $\mathcal{N}_t = \langle \nabla R^\gamma(h_t), \nabla R^\gamma(h_t) - \hat{g}_t \rangle$ *and* $\mathcal{R}_t = \langle \nabla R^\gamma(h_t), \hat{g}_t - g_t \rangle$. $\mathcal{H}_t, \mathcal{M}_t, \mathcal{N}_t$ *and* $\mathcal{R}_t$ *are bounded as follows:*

$$
\begin{aligned}
&\mathcal{M}_t \leq (1+2C)^2 [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa, \\
&\mathbb{E}^2[\mathcal{H}_t] \leq (1+C)^2 [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa, \\
&\mathbb{E}[\mathcal{N}_t] = 0, \\
&\mathbb{E}[\mathcal{R}_t] \leq (1+C) [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M L \kappa B_{1,T+1},
\end{aligned}
$$

where $C = \lambda + \frac{\epsilon' e_c}{\|f\|_{\mathcal{H}}} [\pi_S (1-\gamma) I_1 - \pi_D I_2 + \gamma I_3]$.

D.2 Proof of Lemma 2

To establish Lemma 2, it is essential to derive bounds for $\mathcal{M}_t$, $\mathcal{H}_t$, $\mathcal{N}_t$ and $\mathcal{R}_t$ individually. The comprehensive proof is as follows:

D.2.1 Bound of $\mathcal{M}_t$:

Proof. First, we give the bound of $\mathcal{M}_t$.

$$\mathcal{M}_t = \|g_t\|_{\mathcal{H}}^2 = \|\xi_t + Ch_t\|_{\mathcal{H}}^2 \leq (\|\xi_t\|_{\mathcal{H}} + C\|h_t\|_{\mathcal{H}})^2 \tag{21}$$

If we can give the bound of $\|\xi_t\|_{\mathcal{H}}$ and $\|h_t\|_{\mathcal{H}}$, then we can obtain the bound of $\mathcal{M}_t$.

$$\begin{aligned}
\|\xi_t\|_{\mathcal{H}} = \Big\| & - \pi_S(1-\gamma)e_c k(\widetilde{x}_t^S, \cdot)\left[\pi_+ \ell'(f(\widetilde{x}_t^S) - \epsilon'\|f\|_{\mathcal{H}}, +1) + \pi_- \ell'(f(\widetilde{x}_t^S) - \epsilon'\|f\|_{\mathcal{H}}, -1)\right] \\
& + \pi_D e_c k(\widetilde{x}_t^D, \cdot)\left[(\pi_+ + \pi_-\gamma)\ell'(f(\widetilde{x}_t^D) - \epsilon'\|f\|_{\mathcal{H}}, -1) + (\pi_+\gamma + \pi_-)\ell'(f(\widetilde{x}_t^D) - \epsilon'\|f\|_{\mathcal{H}}, +1)\right] \\
& - \gamma e_c k(x_t^U, \cdot)\left[\pi_+ \ell'(f(x_t^U) - \epsilon'\|f\|_{\mathcal{H}}, +1) + \pi_- \ell'(f(x_t^U) - \epsilon'\|f\|_{\mathcal{H}}, -1)\right]\Big\|_{\mathcal{H}} \\
\leq & \left\|\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S, \cdot)\pi_+ \ell'(f(\widetilde{x}_t^S) - \epsilon'\|f\|_{\mathcal{H}}, +1)\right\|_{\mathcal{H}} + \left\|\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S, \cdot)\pi_- \ell'(f(\widetilde{x}_t^S) - \epsilon'\|f\|_{\mathcal{H}}, -1)\right\|_{\mathcal{H}} \\
& + \|\pi_D e_c k(\widetilde{x}_t^D, \cdot)(\pi_+ + \pi_-\gamma)\ell'(f(\widetilde{x}_t^D) - \epsilon'\|f\|_{\mathcal{H}}, -1)\|_{\mathcal{H}} \\
& + \|\pi_D e_c k(\widetilde{x}_t^D, \cdot)(\pi_+\gamma + \pi_-)\ell'(f(\widetilde{x}_t^D) - \epsilon'\|f\|_{\mathcal{H}}, +1)\|_{\mathcal{H}} \qquad (22) \\
& + \|\gamma e_c k(x_t^U, \cdot)\pi_+ \ell'(f(x_t^U) - \epsilon'\|f\|_{\mathcal{H}}, +1)\|_{\mathcal{H}} + \|\gamma e_c k(x_t^U, \cdot)\pi_- \ell'(f(x_t^U) - \epsilon'\|f\|_{\mathcal{H}}, -1)\|_{\mathcal{H}} \\
\leq & \pi_S\pi_+(1-\gamma)|e_c|M\kappa^{\frac{1}{2}} + \pi_S\pi_-(1-\gamma)|e_c|M\kappa^{\frac{1}{2}} + \pi_D(\pi_+ + \pi_-\gamma)|e_c|M\kappa^{\frac{1}{2}} \\
& + \pi_D(\pi_+\gamma + \pi_-)|e_c|M\kappa^{\frac{1}{2}} + \pi_+\gamma|e_c|M\kappa^{\frac{1}{2}} + \pi_-\gamma|e_c|M\kappa^{\frac{1}{2}} \\
= & [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}} \qquad (23)
\end{aligned}$$

For $t = 1$, according to the definition of h_t,we have $h_1 = 0$ and $\|h_1\|_{\mathcal{H}} = 0$. In add assume that $\|h_t\|_{\mathcal{H}} \leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}}$ for any $t = 1, \cdots, T-1$, we have

$$\begin{aligned}
\|h_{t+1}\|_{\mathcal{H}} &= \|h_t(1 - C\eta_t) - \eta_t\xi_t(\cdot)\|_{\mathcal{H}} \\
&\leq (1 - C\eta_t)\|h_t\|_{\mathcal{H}} + \eta_t\|\xi_t(\cdot)\|_{\mathcal{H}} \\
&\leq (1 - C\eta_t)\left[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma\right]|e_c|M\kappa^{\frac{1}{2}} + \eta_t\left[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma\right]|e_c|M\kappa^{\frac{1}{2}} \\
&\leq (1 + \eta_t)[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}} \\
&\leq 2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}} \qquad (24)
\end{aligned}$$

Thus, based on the above two inequalities, we have

$$\mathcal{M}_t \leq (1+2C)^2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]e_c^2M^2\kappa \tag{25}$$

□

D.2.2 Bound of $\mathcal{H}_t$:

Proof. Then, we prove the bound of $\mathcal{H}_t$.

$$\begin{aligned}
\mathbb{E}^2[\mathcal{H}_t] &= \mathbb{E}^2_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\mathcal{H}_t] \\
&= \mathbb{E}^2_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\|R^\gamma(h_t)\|_{\mathcal{H}}^2] \\
&= \|\mathbb{E}^2_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\hat{\xi}_t + Ch_t]\|_{\mathcal{H}}^2 \\
&\leq (\|\mathbb{E}^2_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\hat{\xi}_t]\|_{\mathcal{H}} + C\|h_t\|_{\mathcal{H}})^2 \qquad (26)
\end{aligned}$$

According to the above results, we have

$$\|E^2_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\hat{\xi}_t]\|_{\mathcal{H}} \leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}} \tag{27}$$

and

$$\|h_t\|_{\mathcal{H}} \leq [\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]|e_c|M\kappa^{\frac{1}{2}}. \tag{28}$$

Therefore, we have

$$\mathbb{E}^2[\mathcal{H}_t] \leq (1+C)^2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa \tag{29}$$

□

D.2.3 Bound of $\mathcal{N}_t$:

Proof. Here, we give the bound of $\mathcal{N}_t$.

$$\begin{aligned}
\mathbb{E}[\mathcal{N}_t] &= \mathbb{E}_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}[\mathcal{N}_t] \\
&= \mathbb{E}_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}\left[\mathbb{E}_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U}\left[\langle \nabla R^\gamma(h_t), \nabla R^\gamma(h_t) - \hat{g}_t\rangle | \widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega\right]\right] \\
&= \mathbb{E}_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega}\left[\langle \nabla R^\gamma(h_t), \mathbb{E}_{\widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U}\left[\nabla R^\gamma(h_t) - \hat{g}_t\rangle | \widetilde{x}_t^S, \widetilde{x}_t^D, x_t^U, \omega\right]\right] \\
&= 0
\end{aligned} \tag{30}$$

□

D.2.4 Bound of $\mathcal{R}_t$:

Proof. Finally, we bound $\mathcal{R}_t$.

$$
\begin{aligned}
\mathbb{E}[\mathcal{R}_t] &= \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}[\mathcal{R}_t] \\
&= \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}[\langle \nabla R^\gamma(h_t), \hat{g}_t - g_t\rangle] \\
&= \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\langle \nabla R^\gamma(h_t), -\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)\big[\pi_+\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)\big] \\
&\quad + \pi_D e_c k(\widetilde{x}_t^D,\cdot)\big[(\pi_++\pi_-\gamma)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)+(\pi_+\gamma+\pi_-)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)\big] \\
&\quad - \gamma e_c k(x_t^U,\cdot)\big[\pi_+\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)\big] + Ch_t \\
&\quad + \pi_S(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)\big[\pi_+\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)\big] \\
&\quad - \pi_D e_c k(\widetilde{x}_t^D,\cdot)\big[(\pi_++\pi_-\gamma)\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)+(\pi_+\gamma+\pi_-)\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)\big] \\
&\quad + \gamma e_c k(x_t^U,\cdot)\big[\pi_+\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)\big] - Ch_t\rangle\Big] \\
&\le \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_S\pi_+(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)[\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1) \\
&\quad - \ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)]\|_{\mathcal{H}}\Big] \\
&\quad + \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_S\pi_-(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)[\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1) \\
&\quad - \ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)]\|_{\mathcal{H}}\Big] \\
&\quad + \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_D(\pi_++\pi_-\gamma)e_c k(\widetilde{x}_t^D,\cdot)[\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1) \\
&\quad - \ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)]\|_{\mathcal{H}}\Big] \\
&\quad + \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_D(\pi_+\gamma+\pi_-)e_c k(\widetilde{x}_t^D,\cdot)[\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1) \\
&\quad - \ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)]\|_{\mathcal{H}}\Big] \\
&\quad + \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_+\gamma e_c k(x_t^U,\cdot)[\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)-\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)]\|_{\mathcal{H}}\Big] \\
&\quad + \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot\|\pi_-\gamma e_c k(x_t^U,\cdot)[\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)-\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)]\|_{\mathcal{H}}\Big] \\
&\le \pi_S(1-\gamma)|e_c|\kappa^{\frac{1}{2}}L\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot|f_t(\widetilde{x}_t^S)-h_t(\widetilde{x}_t^S)|\Big] \\
&\quad + \pi_D(1+\gamma)|e_c|\kappa^{\frac{1}{2}}L\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot|f_t(\widetilde{x}_t^D)-h_t(\widetilde{x}_t^D)|\Big] \\
&\quad + \gamma|e_c|\kappa^{\frac{1}{2}}L\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\Big[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}\cdot|f_t(x_t^U)-h_t(x_t^U)|\Big] \\
&\le \pi_S(1-\gamma)|e_c|\kappa^{\frac{1}{2}}L\sqrt{\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2}\cdot\sqrt{\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}[|f_t(\widetilde{x}_t^S)-h_t(\widetilde{x}_t^S)|^2]} \\
&\quad + \pi_D(1+\gamma)|e_c|\kappa^{\frac{1}{2}}L\sqrt{\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2}\cdot\sqrt{\mathbb{E}_{\widetilde{x}_t^D,\widetilde{x}_t^D,x_t^U,\omega}[|f_t(\widetilde{x}_t^D)-h_t(\widetilde{x}_t^D)|^2]} \\
&\quad + \gamma|e_c|\kappa^{\frac{1}{2}}L\sqrt{\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2}\cdot\sqrt{\mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}[|f_t(x_t^U)-h_t(x_t^U)|^2]} \\
&\le (1+C)[\pi_S(1-\gamma)+\pi_D(1+\gamma)+\gamma]^2 e_c^2 ML\kappa B_{1,T+1} \qquad (31)
\end{aligned}
$$

The second inequality is due to Assumption 4. That completes the proof.

□

D.3 Proof of Theorem 3

Based on the lemmas and assumptions mentioned earlier, we can readily obtain Theorem 3.

Proof. For Convenience, we denote the following three different gradient terms,

$$\begin{aligned}
g_t &= \xi_t + Ch_t \\
&= -\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)[\pi_+\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(f_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)] \\
&\quad + \gamma e_c k(x_t^U,\cdot)[\pi_+\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(f_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)] \\
&\quad + \pi_D e_c k(\widetilde{x}_t^D,\cdot)[(\pi_+-\pi_-\gamma)\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)+(\pi_+\gamma+\pi_-)\ell'(f_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)] \qquad (32) \\
\hat{g}_t &= \hat{g}_t + Ch_t \\
&= -\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)[\pi_+\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)] \\
&\quad + \gamma e_c k(x_t^U,\cdot)[\pi_+\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)] \\
&\quad + \pi_D e_c k(\widetilde{x}_t^D,\cdot)[(\pi_+-\pi_-\gamma)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)+(\pi_+\gamma+\pi_-)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)] \qquad (33)
\end{aligned}$$

$$\begin{aligned}
\nabla R(h_t) &= \mathbb{E}_{\widetilde{x}_t^S}\mathbb{E}_{\widetilde{x}_t^D}\mathbb{E}_{x_t^U}[\hat{g}_t] \\
&= Ch_t + \mathbb{E}_{\widetilde{x}_t^S}\mathbb{E}_{\widetilde{x}_t^D}\mathbb{E}_{x_t^U}\Big[-\pi_S(1-\gamma)e_c k(\widetilde{x}_t^S,\cdot)\big[\pi_+\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(\widetilde{x}_t^S)-\epsilon'\|f\|_{\mathcal{H}},-1)\big] \\
&\quad + \pi_D e_c k(\widetilde{x}_t^D,\cdot)\big[(\pi_+-\pi_-\gamma)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},-1)+(\pi_+\gamma+\pi_-)\ell'(h_t(\widetilde{x}_t^D)-\epsilon'\|f\|_{\mathcal{H}},+1)\big] \\
&\quad + \gamma e_c k(x_t^U,\cdot)\big[\pi_+\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},+1)+\pi_-\ell'(h_t(x_t^U)-\epsilon'\|f\|_{\mathcal{H}},-1)\big]\Big] \qquad (34)
\end{aligned}$$

0From our previous definition, we have $h_{t+1} = h_t - \eta_t g_t, \forall t = 1,\cdots,T$, we have

$$\begin{aligned}
R^\gamma(h_{t+1}) &\le R^\gamma(h_t) + \langle \nabla R^\gamma(h_t), h_{t+1}-h_t\rangle + \frac{\widetilde{L}}{2}\|h_{t+1}-h_t\|_{\mathcal{H}}^2 \\
&= R^\gamma(h_t) - \eta_t\langle \nabla R^\gamma(h_t), g_t\rangle + \frac{\widetilde{L}\eta_t^2}{2}\|g_t\|_{\mathcal{H}}^2 \\
&= R^\gamma(h_t) - \eta_t\langle \nabla R^\gamma(h_t), g_t-\hat{g}_t+\hat{g}_t-\nabla R^\gamma(h_t)+\nabla R^\gamma(h_t)\rangle + \frac{\widetilde{L}\eta_t^2}{2}\|g_t\|_{\mathcal{H}}^2 \\
&= R^\gamma(h_t) - \eta_t\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2 + \eta_t\langle \nabla R^\gamma(h_t), \hat{g}_t-g_t\rangle + \eta_t\langle \nabla R^\gamma(h_t), \nabla R^\gamma(h_t)-\hat{g}_t\rangle + \frac{\widetilde{L}\eta_t^2}{2}\|g_t\|_{\mathcal{H}}^2 \qquad (35)
\end{aligned}$$

Taking expectations on both sides, we can obtain

$$\begin{aligned}
&\eta_t\mathbb{E}[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2] \\
&\le \mathbb{E}[R^\gamma(h_t)] - \mathbb{E}[R^\gamma(h_{t+1})] + \eta_t\mathbb{E}[\langle \nabla R^\gamma(h_t), \hat{g}_t-g_t\rangle] + \eta_t\mathbb{E}[\langle \nabla R^\gamma(h_t), \nabla R^\gamma(h_t)-\hat{g}_t\rangle] + \frac{\widetilde{L}\eta_t^2}{2}\mathbb{E}[\|g_t\|_{\mathcal{H}}^2] \qquad (36)
\end{aligned}$$

where R^* denotes the optimal value of $R^\gamma(h)$ and $\mathbb{E}[\cdot] = \mathbb{E}_{\widetilde{x}_t^S,\widetilde{x}_t^D,x_t^U,\omega}[\cdot]$. Let us denote $\mathcal{H}_t = \sqrt{\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2}$, $\mathcal{M}_t = \|g_t\|_{\mathcal{H}}^2$, $\mathcal{N}_t = \langle \nabla R^\gamma(h_t), \nabla R^\gamma(h_t)-\hat{g}_t\rangle$ and $\mathcal{R}_t = \langle \nabla R^\gamma(h_t), \hat{g}_t-g_t\rangle$. For the fixed stepsize $\eta_t = \overline{\eta} = \frac{\theta}{T^{3/4}}$, based on Lemma 2, we have

$$\begin{aligned}
&\overline{\eta}\mathbb{E}[\|\nabla R^\gamma(h_t)\|_{\mathcal{H}}^2] \\
&\le \mathbb{E}[R^\gamma(h_t)] - \mathbb{E}[R^\gamma(h_{t+1})] + \overline{\eta}(1+C)[\pi_S(1-\gamma)+\pi_D(1+\gamma)+\gamma]^2 e_c^2 ML\kappa B_{1,T+1} \\
&\quad + \frac{\widetilde{L}\overline{\eta}^2}{2}(1+C)^2[\pi_S(1-\gamma)+\pi_D(1+\gamma)+\gamma]^2 e_c^2 M^2\kappa \\
&\le \mathbb{E}[R^\gamma(h_t)] - \mathbb{E}[R^\gamma(h_{t+1})] + \overline{\eta}(1+C)[\pi_S(1-\gamma)+\pi_D(1+\gamma)+\gamma]^3 |e_c|^3 M^2 L\kappa(\sqrt{\kappa}+\sqrt{\phi})\frac{\theta}{T^{1/4}} \\
&\quad + \frac{\widetilde{L}\overline{\eta}^2}{2}(1+C)^2[\pi_S(1-\gamma)+\pi_D(1+\gamma)+\gamma]^2 e_c^2 M^2\kappa \qquad (37)
\end{aligned}$$

Summing both side of the inequality for $t \in \{1, \cdots, T\}$ and recall the Assumption, we have

$$\begin{aligned}
&\overline{\eta}\mathbb{E}[\sum_{t=1}^{T} \|\nabla R^{\gamma}(h_t)\|_{\mathcal{H}}^2] \\
&\leq R^{\gamma}(h_1) - R_{inf}^{\gamma} + (1+C)[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^3 |e_c|^3 M^2 L\kappa(\sqrt{\kappa} + \sqrt{\phi})\frac{\theta}{T^{1/4}}T \\
&\quad + \frac{\widetilde{L}\overline{\eta}^2}{2}(1+C)^2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa T
\end{aligned} \tag{38}$$

Rearranging the above inequality and dividing by $\bar{\eta}T$, we have

$$\begin{aligned}
&\mathbb{E}[\frac{1}{T}\sum_{t=1}^{T} \|\nabla R^{\gamma}(h_t)\|_{\mathcal{H}}^2] \\
&\leq \frac{R^{\gamma}(h_1) - R_{inf}^{\gamma}}{\overline{\eta}T} + (1+C)[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^3 |e_c|^3 M^2 L\kappa(\sqrt{\kappa} + \sqrt{\phi})\frac{\theta}{T^{1/4}} \\
&\quad + \frac{\widetilde{L}\overline{\eta}}{2}(1+C)^2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa \\
&\leq \frac{R^{\gamma}(h_1) - R_{inf}^{\gamma}}{\theta T^{1/4}} + (1+C)[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^3 |e_c|^3 M^2 L\kappa(\sqrt{\kappa} + \sqrt{\phi})\frac{\theta}{T^{1/4}} \\
&\quad + \frac{\widetilde{L}\theta}{2T^{3/4}}(1+C)^2[\pi_S(1-\gamma) + \pi_D(1+\gamma) + \gamma]^2 e_c^2 M^2 \kappa
\end{aligned} \tag{39}$$

□

E THE SPECIFIC ALGORITHM FOR THE SGD-SDU EXPERIMENT

Algorithm 3 Stochastic Gradient Descent for our adversarial SDU classification (SGD-SDU)

1: **Input:** Learning rate η, Number of iterations T.
2: Initialize function f in the **RKHS** $\mathcal{H}$.
3: **for** $t = 1$ to T **do**
4: Sample $(\widetilde{x}_S)$ from $\widetilde{D}_S$, Sample $(\widetilde{x}_D)$ from $\widetilde{D}_D$, Sample x_U from D_U.
5: Compute gradients:

$$\begin{aligned}
\nabla_f R_S^{\gamma} &= \frac{\pi_S(1-\gamma)}{\pi_c}\left[\pi_+ \nabla \ell_1(\widetilde{x}_S) - \pi_- \nabla \ell_2(\widetilde{x}_S)\right] \\
\nabla_f R_D^{\gamma} &= \frac{\pi_D}{\pi_c}\left[\gamma_1 \nabla \ell_2(\widetilde{x}_D) - \gamma_2 \nabla \ell_1(\widetilde{x}_D)\right] \\
\nabla_f R_U^{\gamma} &= \frac{\gamma}{\pi_c}\left[\pi_+ \nabla \ell_1(x_U) - \pi_- \nabla \ell_2(x_U)\right]
\end{aligned}$$

6: Update function parameters: $f \leftarrow f - \eta\left(\lambda f + \nabla_f R_S^{\gamma} + \nabla_f R_D^{\gamma} + \nabla_f R_U^{\gamma}\right)$
7: **end for**

F ADDITIONAL EXPERIMENTS

The adversarial techniques employed in the experiments conducted in this paper are detailed below: FGSM with a perturbation parameter of $\epsilon = 0.3$, 10-step PGD with $\epsilon = 0.016$, and C&W attack implemented using the code available at https://github.com/Trusted-AI/adversarial-robustness-toolbox.

F.1 Adversarial Training for SDU Learning

It should be noted that there is no reference for the adversarial training on SDU Learning as far as our knowledge. To solve this concern, we've proposed the corresponding adversarial training algorithm based on the objective

function of this paper, called Adversarial Training for SDU learning (AT-SDU$_{all}$). Additionally, we also extended the adversarial training to specifically target similar, dissimilar, and unlabeled data, called AT-SDU$_S$, AT-SDU$_D$, and AT-SDU$_U$, respectively. The results of these methods can be found in Table 6. (Time is the result of the PGD test.) Our algorithm stands out significantly in terms of efficiency and speed, being **20 times faster** than AT-SDU, although it may be slightly behind AT-SDU in accuracy. This is acceptable compared to the boost in speed. Because we use random features to approximate kernel and minimization to approximate the minimax problem, the complexity of the problem is reduced.

Table 6: Accuracy on MNIST and Combined test with various attacks and perturbations constrained by l_2-norm.

Dataset	MNIST				Time(s)	Combined				Time(s)
Method	clean	FGSM	PGD	CW		clean	FGSM	PGD	CW	
AT-SDU_all	92.84	94.01	77.12	69.88	6580	77.02	78.72	76.93	74.76	26442
AT-SDU_S	95.18	86.4	70.69	64.62	2535	79.21	71.94	75.19	71.95	11714
AT-SDU_D	97.08	93.93	76.02	68.86	2738	81.4	75.38	76.57	73.63	11840
AT-SDU_U	92.98	89.47	70.03	63.96	2441	78.81	70.03	74.84	70.48	11696
QSG-ATSDU	94.07	93.27	75.37	69.81	413	80.59	78.36	76.78	74.52	664

F.2 Ablation Studies

F.2.1 Impact of Constraint on ϵ, σ, λ, and γ

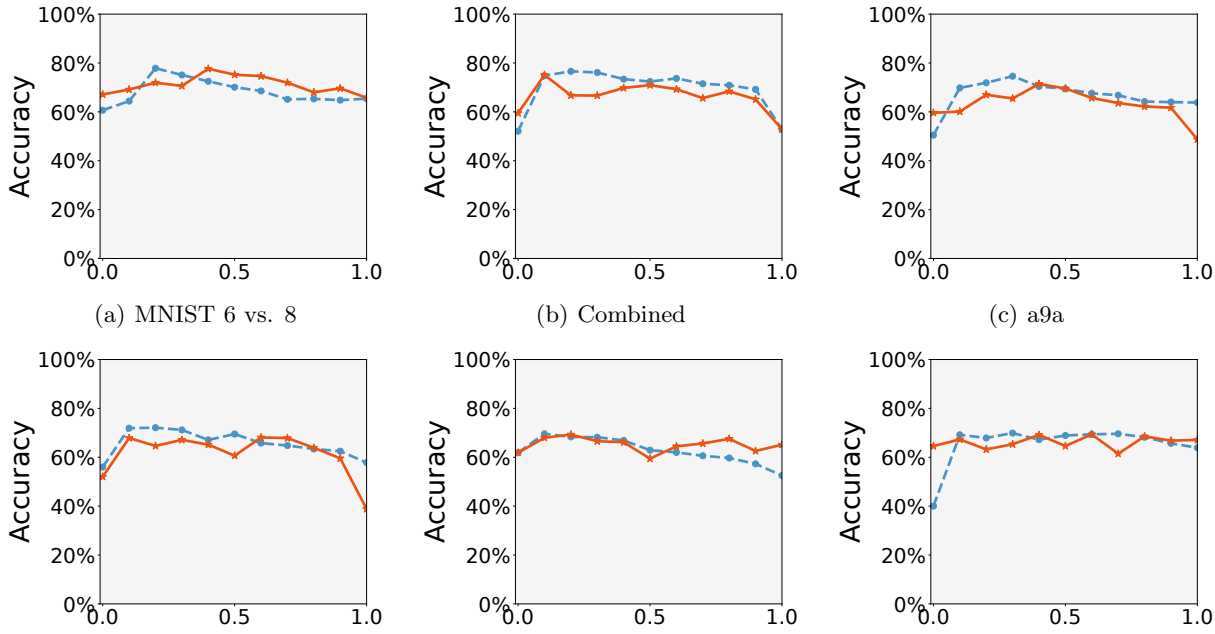
In this series of experiments, we investigate the impact of different parameter settings on the performance of our method. We focus on four hyper-parameters: ϵ, σ, λ, and γ, each of which plays a critical role in shaping the behavior and outcomes of our method. The description of these parameters are as follows:

1. ϵ: Serving as the perturbation radius, ϵ influences the scope of the model's robustness to input variations.
2. σ: The σ parameter, embodying the kernel parameter, plays a fundamental role in shaping the influence of data points within the model. With values ranging from 2^{-10} to 2^{10}, our study uncovers the impact of σ on the learned function's smoothness and sensitivity to input variations.
3. λ: λ assumes the role of the regularization parameter, driving the trade-off between data fitting and the regularization term in the model's optimization process. Our exploration across the range of 2^{-10} to 2^{10} reveals how λ governs the model's capacity to generalize beyond the training data.
4. γ: γ serves as the equilibrium factor between the SD (Source Domain) and DU (Target Domain) classifications in our method. By exploring the spectrum from 0.1 to 1.0 with increments of 0.1, we scrutinize how γ influences the decision boundary balance between the two domains.

Each experiment involved fixing three parameters while exploring different values for the remaining one. This meticulous approach enabled us to thoroughly evaluate the performance and behavior of our method under diverse scenarios. The results are presented in Figure 3 and 4. The experimental findings clearly demonstrate that our method achieves consistently high test accuracy across different ϵ, λ values, with fluctuations remaining within a certain range. These results indicate the robustness and stability of our approach in handling varying conditions of ϵ, σ, σ and γ.

F.2.2 Impact of the Ratio of n_{SD} and n_U

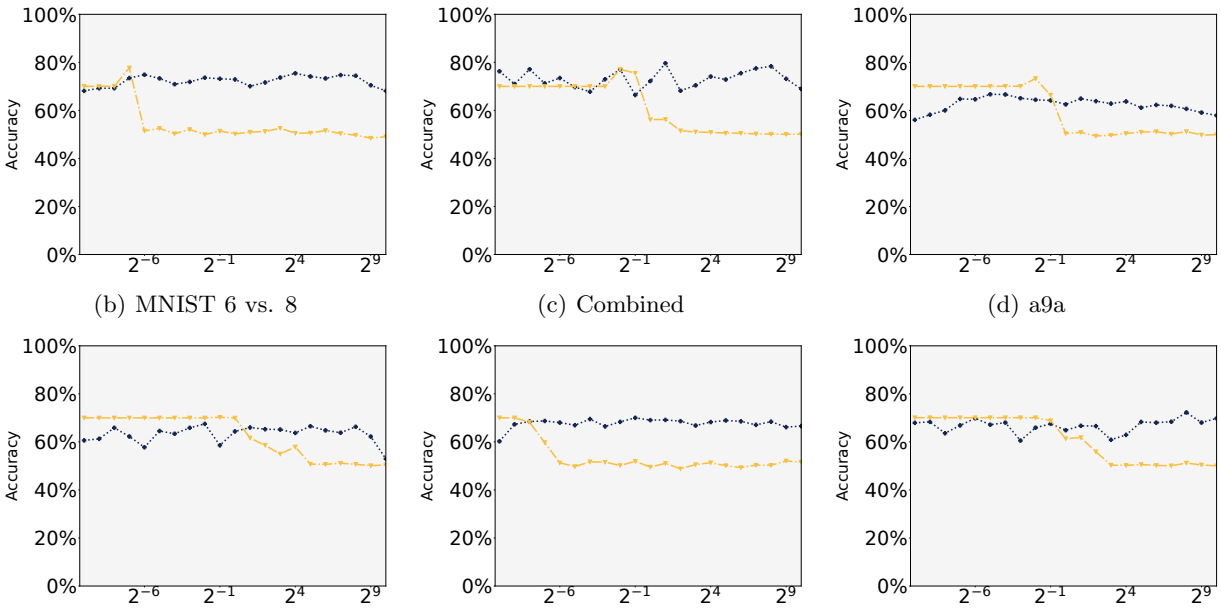
In order to delve into the potential repercussions of different ratios between the number of similar and dissimilar dataset (n_{SD}) and the number of unlabeled dataset (n_U) on the efficacy of our methodology, QSG-ATSDU, a meticulous series of experiments was meticulously carried out. These experiments were executed with varying configurations of n_{SD} and n_U. Through a systematic evaluation of our proposed technique under these diverse ratio settings, the outcomes have been succinctly depicted in Figure 5. It is of significance to note that the empirical insights garnered from these experiments unveil a noteworthy trend. While there is a discernible exception with the extreme case of the $1:5$ ratio, the performance outcomes for the other evaluated ratios exhibit subtle fluctuations. Within a confined range, the behavior of our method seems to exhibit a certain degree of resilience across these varying ratios.



(a) MNIST 6 vs. 8 (b) Combined (c) a9a

(d) Acoustic (e) CIFAR10 automobile vs. truck (f) covtype

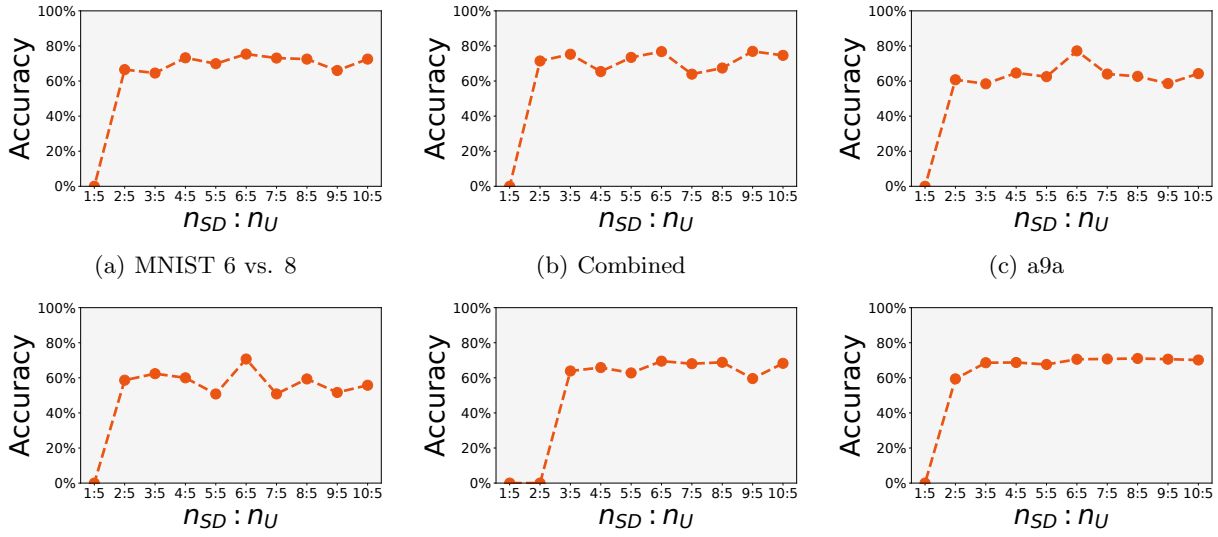
Figure 3: The test accuracy of different ϵ, γ by fixed the others parameters.



(b) MNIST 6 vs. 8 (c) Combined (d) a9a

(e) Acoustic (f) CIFAR10 automobile vs. truck (g) covtype

Figure 4: The test accuracy of different σ, λ by fixed the others parameters.



(a) MNIST 6 vs. 8 (b) Combined (c) a9a

(d) Acoustic (e) CIFAR10 automobile vs. truck (f) covtype

Figure 5: The test accuracy of different ratio of $n_{SD} : n_U$.