PILLAR: How to make semi-private learning more effective

Francesco Pinto*
University of Oxford
Oxford, England
Francesco.pinto@eng.ox.ac.uk

Yaxi Hu*
Max Planck Institute for Intelligence Systems
Tübingen, Germany
yaxi.hu@tuebingen.mpg.de

Fanny Yang
ETH Zürich
Zürich, Switzerland
fan.yang@inf.ethz.ch

Amartya Sanyal
Max Planck Institute for Intelligence Systems
Tübingen, Germany
amsa@di.ku.dk

***Abstract*—In Semi-Supervised Semi-Private (SP) learning, the learner has access to both public unlabelled and private labelled data. We propose PILLAR, an easy-to-implement and computationally efficient algorithm that, under mild assumptions on the data, provably achieves significantly lower private labelled sample complexity and can be efficiently run on real-world datasets. The key idea is to use public data to estimate the principal components of the pre-trained features and subsequently project the private dataset onto the top-k Principal Components. We empirically validate the effectiveness of our algorithm in a wide variety of experiments under tight privacy constraints ($\epsilon < 1$) and probe its effectiveness in low-data regimes and when the pre-training distribution significantly differs from the one on which SP learning is performed. Despite its simplicity, our algorithm exhibits significantly improved performance, in all of these settings, over all available baselines that use similar amounts of public data while often being more computationally expensive. For example, in the case of CIFAR-100 for $\epsilon = 0.1$, our algorithm improves over the most competitive baselines by a factor of at least two.**

I. Introduction

In recent years, Machine Learning (ML) models have become an integral part of our daily lives, commonly trained on vast amounts of sensitive private data to offer services better tailored to users' needs. However, this has escalated concerns regarding user privacy. Recent studies [1]–[3] demonstrate the potential for malicious queries to ML models to reveal private information. To address this problem, the de-facto standard remedy is to enforce $(\epsilon, \delta)-$Differential Privacy (DP) guarantees on the ML algorithms [4]. Nonetheless, meeting these guarantees often compromises model utility, unless the volume of available private training data is significantly increased [5]–[9]. In the context of private learning, [10], [11] identified theoretical lower bounds, showing a direct dependence of this cost on data dimensionality, a connection not seen in non-private learning.

To mitigate this degradation of utility, several techniques have been employed. One approach is leveraging feature extractors pre-trained on large-scale datasets (presumed public), even if their data-generating distribution diverges from the private data [12]–[15]. Training a linear classifier atop these pre-trained features has proven to be both cost-efficient and effective [12], [13]. Utility gains can also be achieved by deeming part of the private data public, a scenario known as Semi-Private (SP) learning [16]–[21]. Notably, utilizing public data to assist the optimizer [18], [21], [22] and to reduce problem dimensionality [17], [23]–[25] are among the most effective strategies in this context. However, although these techniques have been shown to be effective for large ϵ values on datasets like CIFAR-10, our experiments over an extensive variety of datasets with varying amounts of training data and classes suggest that the effectiveness of some of these methods is limited in more challenging settings like low data and small ϵ.

In this work, we propose a simple SP algorithm called PILLAR and conduct an extensive empirical study over a wide range of datasets and strict privacy settings to show its effectiveness over existing methods. The key idea is to use public data to estimate the principal components of the pre-trained features and subsequently project the private dataset onto the top-k Principal Components. Despite its simplicity and use of existing techniques like dimensionality reduction [17], [24], it outperforms existing methods in these challenging settings. Beyond its empirical performance, our algorithm also enjoys a provably dimension-independent sample complexity when learning linear halfspaces, and when the distribution satisfies a low-rank separability condition outlined in Definition 3

For practical applications like image classification, we ascertain that pre-trained representations meet this condition across a diverse range of datasets. As suggested by concurrent research [26], we validate our algorithm's efficacy not only against standard benchmarks in DP literature (e.g., CIFAR-10 and CIFAR-100) but also across various datasets (Figure 1) that better represent the challenges and application domains of private training. Remarkably, our experiments reveal that our algorithm surpasses several existing state-of-the-art algorithms [17], [18], [27], [28], with various levels of access to public data, across seven different datasets while remaining

computationally economical.

Unlike previous works, our evaluations particularly concentrate on private data distributions (e.g. traffic signs and medical datasets in addition to object recognition) that significantly deviate from the pre-training one (ImageNet) and focus on low-data regimes. We posit that testing on such pertinent benchmarks is crucial to showcase the practical applicability of our algorithm in privacy-sensitive scenarios. Intriguingly, we observe the benefits of our approach amplify as the privacy guarantees tighten, i.e., when ϵ is lower. Several practical deployments of DP, especially in the query release paradigm, have targeted low ϵ[1] but this remains elusive when deploying machine learning models. We hope our work will accelerate deployment of ML classification models with $\epsilon < 1$.

To summarise, our contributions are the following:

- We introduce PILLAR, a straightforward, readily-implementable, and computationally inexpensive SP algorithm. It enhances classification accuracy compared to several existing competitive algorithms, some of which also exploit dimensionality reduction and semi-private learning principles.
- For learning half-spaces, we establish that our algorithm attains dimension-independent private labelled sample complexity with *large margin low rank distributions*. Significantly, our results are versatile, accommodating distribution shifts between public and private data, and adaptable to multiple loss functions.
- We refine privacy evaluation benchmarks for image classification, concentrating on scenarios that, in our view, hold greater relevance to privacy. These include i) private datasets exhibiting substantial shift from pre-training (and public) datasets, ii) the availability of limited (private and public) training data, and iii) stringent privacy regimes ($\epsilon < 1$)

II. Semi-Private Learning

We begin by defining Differential Privacy (DP). DP ensures that the output distribution of a randomized algorithm remains stable when a single data point is modified. In this paper, a differentially private learning algorithm produces comparable distributions over classifiers when trained on neighbouring datasets. Neighbouring datasets refer to datasets that differ by a single entry. Formally,

Definition 1 (Differential Privacy [4])**.** *A learning algorithm $\mathcal{A}$ is (ϵ, δ)-differential private, if for any two datasets S, S' differing in one entry and for all outputs $\mathcal{Z}$, we have,*

$$\mathbb{P}\left[\mathcal{A}(S) \in \mathcal{Z}\right] \leq e^{\epsilon}\mathbb{P}\left[\mathcal{A}(S) \in \mathcal{Z}\right] + \delta.$$

For $\epsilon < 1$ and $\delta = o(1/n)$, (ϵ, δ)-differential privacy provides valid protection against potential privacy attacks [3].

Differential Privacy and Curse of Dimensionality Similar to non-private learning, the most common approach to DP learning is through Differentially Private Empirical Risk Minimization (DP-ERM), with the most popular optimization procedure being DP-SGD [29] or analogous DP variants of typical optimization algorithms [31]. However, unlike non-private ERM, the sample complexity of DP-ERM suffers from a polynomial dependence on the dimensionality of the problem [10], [11]. Hence, we explore slight relaxations to this definition of privacy to alleviate this problem. We show theoretically (Section III) and through extensive experiments (Section IV and V) that this is indeed possible with some realistic assumptions on the data and a slightly relaxed definition of privacy known as semi-private learning that we describe below. For a discussion of broader impacts and limitations of this setting, please refer to Appendix C.

A. Semi-Private Learning

The concept of semi-private learner was introduced in [16]. In this setting, the learning algorithm is assumed to have access to both a private labelled and a public (labelled or unlabelled) dataset. In this work, we assume the case of only having an *unlabelled* public dataset. This specific setting has been referred to as Semi-Supervised Semi-Private learning in [16]. However, for the sake of brevity, we will refer to it as Semi-Private learning (SPL).

Definition 2 ($(\alpha, \beta, \epsilon, \delta)$-semi-private learner on a family of distributions $\mathcal{D}$)**.** *An algorithm $\mathcal{A}$ is said to $(\alpha, \beta, \epsilon, \delta)$-semi-privately learn a hypothesis class $\mathcal{H}$ on a family of distributions $\mathcal{D}$, if for any distribution $D \in \mathcal{D}$, given a private labelled dataset S^L of size n^L and a public unlabelled dataset S^U of size n^U sampled i.i.d. from D, $\mathcal{A}$ is (ϵ, δ)-DP with respect to S^L and outputs a hypothesis $\hat{h}$ satisfying*

$$\mathbb{P}[\mathbb{P}_{(x,y)\sim D}\left[h(x) \neq y\right] \leq \alpha] \geq 1 - \beta,$$

where the outer probability is over the randomness of S^L, S^U, and $\mathcal{A}$.

Further, the sample complexity n^L and n^U must be polynomial in $\frac{1}{\alpha}, \frac{1}{\beta}$, and the size of the input space. In addition, n^L must also be polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$. The algorithm is said to be efficient if it also runs in time polynomial in $\frac{1}{\alpha}, \frac{1}{\beta}$, and the size of the input domain.

A key distinction between our work and the previous study by [16] is that they examine the distribution-independent agnostic learning setting, whereas we investigate the distribution-specific realisable setting. On the other hand, while their algorithm is computationally inefficient, ours can be run in time polynomial in the relevant parameters and implemented in practice on various datasets with state-of-the-art results. We discuss our algorithm in Section II-B.

Relevance of Semi-Private Learning In various privacy-sensitive domains such as healthcare, legal, social security, and census data, there is often some amounts of publicly available data in addition to the private data. For instance, the U.S. Census Bureau office has partially released historical data before 2020 without enforcing any differential privacy guarantees [2]. It has

[1] https://desfontain.es/privacy/real-world-differential-privacy.html

[2] https://www2.census.gov/library/publications/decennial/2020/census-briefs/c2020br-03.pdf

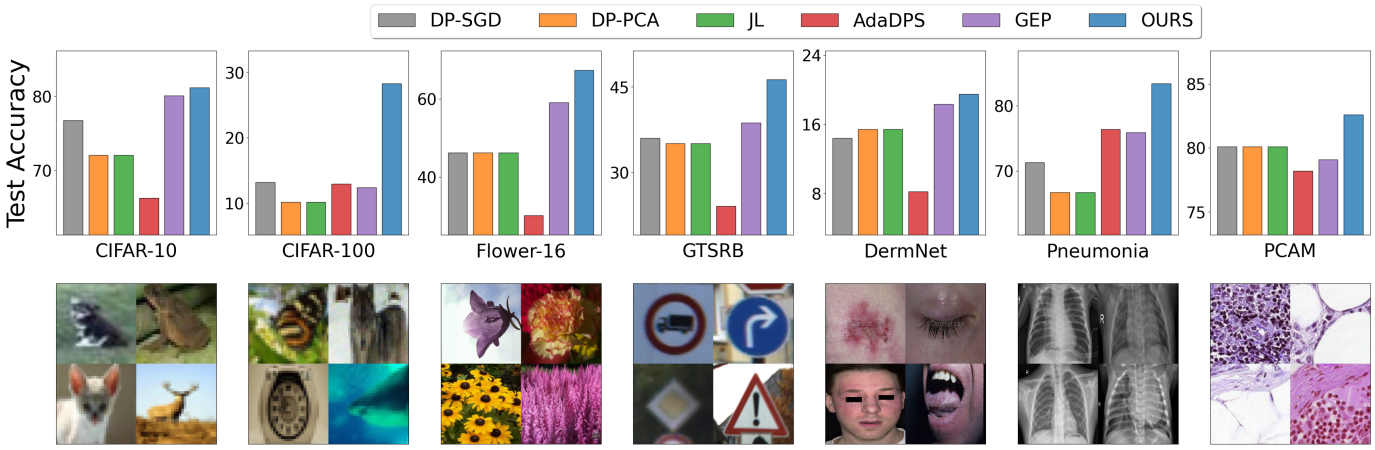


Fig. 1: We compare our algorithm PILLAR with DP-SGD [28], DP-SGD with DP-PCA [29], DP-SGD with JL transformation [27], AdaDPS [18], and GEP [17] on CIFAR-10, CIFAR-100, GTSRB, Flower-16, Dermnet, Pneumonia, and PCAM for $\epsilon = 0.1$. PILLAR consistently outperforms all baselines, often with a large margin. All methods use features extracted from a ResNet-50 pre-trained on ImageNet-1K using either Supervised Learning (SL) or Self-Supervised Learning (BYOL [30])

Algorithm 1 PILLAR $\mathcal{A}_{\epsilon,\delta}(k, \ell)$ for learning halfspaces

1: **Input:** Labelled dataset S^L, Unlabelled dataset S^U, low-dimension k, L-Lipschitz loss function ℓ, high probability parameter β.
2: Using S^U, construct $\widehat{\Sigma} = \sum_{x \in S^U} xx^\top / n^U$.
3: Construct the transformation matrix $\hat{A}_k$ whose i^{th} column is the i^{th} eigenvector of $\widehat{\Sigma}$.
4: Project S^L with the transformation matrix $\hat{A}_k$,

$$S_k^L = \{(\hat{A}_k^\top x, y) : (x, y) \in S^L\}.$$

5: Obtain $v_k = \mathcal{A}_{\text{Noisy-SGD}}(S_k^L, \ell, (\epsilon, \delta), \beta/4)$
6: **Output:** Return $\hat{w} = \hat{A}_k v_k$.

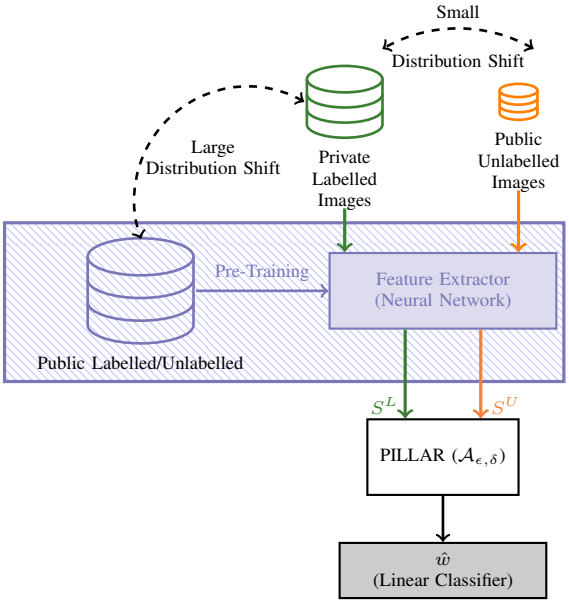


Fig. 2: Diagram describing how PILLAR is applied in image classification (using DP-SGD with cross-entropy loss in Line 4 of Algorithm 1).

also been observed that different data providers may have varying levels of concerns about privacy [32]. In medical data, some patients may consent to render some of their data public to foster research. In other cases, data may become public due to the expiration of the right to privacy after specific time limits [3].

It is also very likely that this public data may be *unlabelled* for the task at hand. For example, if data is collected to train a model to predict a certain disease, the true diagnosis may have been intentionally removed from the available public data to protect sensitive information of the patients. Further, the data may had been collected for a different purpose like a vaccine trial. Finally, the cost of labelling may be prohibitive in some cases. Hence, when public (unlabelled) data is already available, we focus on harnessing this additional data effectively, while safeguarding the privacy of the remaining private data. We hope this can lead to the development of highly performant algorithms which in turn can foster wider adoption of privacy-preserving techniques.

B. PILLAR: An Efficient Semi-Private Learner

In this work, we propose a (semi-supervised) semi-private learning algorithm called PILLAR (PrIvate Learning with Low rAnk Representations), described in Algorithm 1. Before providing formal guarantees in Section III, we first describe how PILLAR is applied in practice. Our algorithm works in two stages.

Leveraging recent practices [12], [13] in DP training with deep neural networks, we first use pre-trained feature extractors

[3]https://www.census.gov/history/www/genealogy/decennial_census_records/the_72_year_rule_1.html

to transform the private labelled and public unlabelled datasets to the representation space to obtain the private and public representations. We use the representations in the penultimate layer of the pre-trained neural network for this purpose. As shown in Figure 2, the feature extractor is trained on large amounts of labelled or unlabelled public data, following whatever training procedure is deemed most suitable. For this paper, we pre-train a ResNet-50 using supervised training (SL), self-supervised training (BYOL [30] and MocoV2+ [33]), and semi-supervised training (SemiSL and Semi-WeakSL [34]) on ImageNet. In the main body, we only focus on SL and BYOL pre-training. As we discuss extensively in Appendix B-H, our algorithm is effective independent of the choice of the pre-training algorithm. In addition, while the private and public datasets are required to be from the same (or similar) distribution, we show that the pre-training dataset can come from a significantly different distribution. In fact, we use ImageNet as the pre-training dataset for all our experiments even when the distributions of the public and private datasets range from CIFAR-10/100 to histological and x-ray images as shown in Figure 1. Recently, [35] have explored the complementary question of how to choose the right pre-training dataset.

In the second stage, PILLAR takes as input the feature representations of the private labelled and public unlabelled datasets, and feeds them to Algorithm 1. We denote these datasets of representations as S^L and S^U respectively. Briefly, Algorithm 1 projects the private dataset S^L onto a low-dimensional space spanned by the top principal components estimated with S^U, and then applies gradient-based private algorithms (e.g. Noisy-SGD [11] in Appendix A-A) to learn a linear classifier on top of the projected features. Algorithm 1 provides an implementation of PILLAR with Noisy-SGD, whereas in our experiments we show that commonly used DP-SGD [29] is also effective [4].

III. Theoretical Results

In this section, we first describe the assumptions under which we provide our theoretical results and show they can be motivated both empirically and theoretically. Then, we show a dimension-independent sample complexity bound for PILLAR under the mentioned assumptions.

A. Problem setting

Our theoretical analysis focuses on learning linear halfspaces $\mathcal{H}^d$ in d dimensions. Consider the instance space $\mathcal{X}_d = B_2^d = \left\{x \in \mathbb{R}^d : \|x\|_2 = 1\right\}$ as the d-dimensional unit sphere and the binary label space $\mathcal{Y} = \{-1, 1\}$. In practice, the instance space is the (normalized) representation space obtained from the pre-trained network. The hypothesis class of linear halfspaces is

$$\mathcal{H}^d = \left\{f_w(x) = \text{sign}\left(\langle w, x\rangle\right) | w \in B_2^d\right\}.$$

[4]Other state-of-art adaptation of DP optimization algorithms, such as DP-SCO [31] and DP-RAFT [36], can also be applied in step 5 of PILLAR for potentially achieving better accuracy (see Appendix B-F for further experiments).

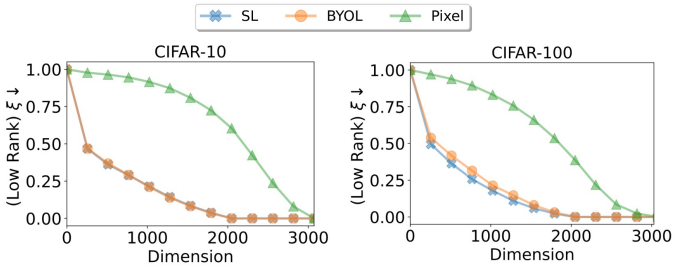


Fig. 3: Estimate of ξ for linear classifiers trained on embeddings of two CIFAR-10 and CIFAR-100 classes, extracted from pre-trained ResNet50s, as well as the raw images (Pixel).

We consider the setting of distribution-specific learning, where our family of distributions admits a large margin linear classifier that contains a significant projection on the top principal components of the population covariance matrix. We formalise this as (γ, ξ_k)-Large margin low rank distributions.In contrast to the usual low rank assumption on the feature matrix [37], large margin low rank distributions can have full rank covariance matrix and generate full rank feature matrix, as long as the true parameter retains its norm in the low dimensional space spanned by the first k eigenvectors of the feature's covariance matrix.

Definition 3 ((γ, ξ_k)-Large margin low rank distribution)**.** *A distribution D over $\mathcal{X}_d \times \mathcal{Y}$ is a (γ, ξ_k)-Large margin low rank distribution if there exists $w^\star \in B_2^d$ such that*

- $\mathbb{P}_{(x,y)\sim D}\left[\frac{y\langle w^\star, x\rangle}{\|w^\star\|_2\|x\|_2} \geq \gamma\right] = 1$ *(Large-margin),*
- $\left\|A_k A_k^\top w^\star\right\|_2 \geq 1 - \xi_k$ *(Low-rank separability).*

where A_k is a $d \times k$ matrix whose columns are the top k eigenvectors of $\mathbb{E}_{X\sim D_X}\left[X^\top X\right]$.

It is worth noting that for every distribution that admits a positive margin γ, the low-rank separability condition is automatically satisfied for all $k \leq d$ with some $\xi_k \geq 0$. Intuitively, this condition requires that there is a large margin classifier with significant projection on the top principal components of the data. However, the low rank separability is helpful for learning, only if it holds for a small k and small ξ_k simultaneously. These assumptions are both theoretically and empirically realisable. Theoretically, we show in Appendix A-E that a class of commonly studied Gaussian mixture distributions with full rank covariance matrices satisfies these properties with $k = 2$ and $\xi_2 = 0$. Empirically, we show in Figure 3 that pre-trained features satisfy these properties with small ξ and k.

Pre-trained features are almost Large-Margin and Low-Rank Figure 3 shows that feature representations of CIFAR-10 and CIFAR-100 obtained by various pre-training strategies approximately satisfy the conditions of Definition 3. To verify the low-rank separability assumption, we first train a binary linear SVM $w^\star$ for a pair of classes on the representation space and estimate $\xi_k = 1 - \left\|A_k A_k^\top w^\star\right\|_2$ as defined in Definition 3.

Loss function ℓ	Formula	Lipschitzness L_ℓ
Cross-entropy loss	$\frac{\log\left(1+e^{-y\langle w,x\rangle}\right)}{\log 2}$	2
Scaled hinge loss	$\max\left\{0, 1-\frac{y\langle w,x\rangle}{0.9\gamma_0(1-\xi_0)}\right\}$	$\frac{1}{0.9\gamma_0(1-\xi_0)}$

TABLE I: Loss functions we consider in Theorem 1, with their expressions and the associated Lipschitz constants

We also compute ξ_k when $w^\star$ is trained on the pixel space [5]. As shown in Figure 3, images in the representation space are better at satisfying the low-rank separability assumption compared to images in the pixel space.

B. Private labelled sample complexity analysis

In this section, we present the theoretical guarantees of PILLAR for Semi-Private learning of linear halfspaces. We prove that for binary cross entropy loss and hinge loss defined in Table I, PILLAR is (ϵ, δ)-DP with respect to the private dataset and achieve high accuracy in learning linear halfspaces with relatively small number of private labelled data samples. Please refer to Appendix A-B for the proof of Theorem 1.

Theorem 1. *Let $k \leq d \in \mathbb{N}, \gamma_0 \in (0,1)$, and $\xi_0 \in (0,1)$. Consider the family of distributions $\mathcal{D}_{\gamma_0,\xi_0}$ which consists of all (γ, ξ_k)-large margin low rank distributions over $\mathcal{X}_d \times \mathcal{Y}$, where $\gamma \geq \gamma_0$ and $\xi_k \leq \xi_0$. For any $\alpha \in (0,1), \beta \in (0, 1/4)$, $\epsilon \in (0, 1/\sqrt{k})$, and $\delta \in (0,1)$, PILLAR with scaled hinge loss or cross entropy loss, is an $(\alpha, \beta, \epsilon, \delta)$-semi-private learner for linear halfspaces $\mathcal{H}^d$ on $\mathcal{D}_{\gamma_0,\xi_0}$ with sample complexity*

$$n^U = O\left(\frac{\log 2/\beta}{(1-\xi_0)^2\Delta_k^2}\right), n^L = \widetilde{O}\left(\frac{L_\ell\sqrt{k}}{\alpha\epsilon}\right)$$

where Δ_k denote the gap between the k^{th} and the $k+1^{\text{th}}$ eigenvalue of the population covariance matrix, and L_ℓ is the Lipschitz coefficient of the loss function ℓ[6].

Table I provides a summary of two loss functions and the associated Lipschitz coefficients. Notably, the Lipschitz coefficient $L_\ell = \frac{1.1}{\gamma_0(1-\xi_0)}$ for the scaled hinge loss varies with the distributional parameters γ_0 and ξ_0. In contrast, for cross-entropy loss, L_ℓ remains fixed at 2. Hence, PILLAR with scaled hinge loss is inherently designed to better harness the large margin property of the distribution with large γ_0 and small ξ_0. On the other hand, PILLAR with cross entropy loss reflects the experiments more closely.

As discussed in Section III-A, the feature representations of images, obtained from pre-trained neural networks, usually satisfy the properties of large-margin low rank distributions (Figure 3). Thus, in practical implementation, the private and public datasets refer to private and public representations, as shown in Figure 2. Note that while Theorem 1 only guarantees (ϵ, δ)-DP on the set of private representations (see Figure 2), this guarantee can also extend to (ϵ, δ)-DP on the private labelled image dataset. See Appendix A-B for more details.

As a concrete instance of the application of Theorem 1, we formally define a family of distributions based on gaussian mixtures, referred to as large margin Gaussian mixture distributions, in Appendix A-E. For this family of distributions, we demonstrate through Theorem 1 that PILLAR significantly reduces the private sample complexity from $O(\sqrt{d})$ to $O(1)$.

C. Distribution shift between private and public datasets

PILLAR also provides theoretical guarantees when the private and public representations come from similar, but not identical distributions. In this case, private sample complexity also depends on the Total Variation (TV) distance, say η between the two distributions. An informal theorem is presented below in Theorem 2, while the formal result can be found in Appendix A-D.

Theorem 2. *Let $k, d, \gamma_0, \xi_0, \mathcal{D}_{\gamma_0,\xi_k}$, $\alpha, \beta, \epsilon, n^L, n^U$ and δ be defined as in Theorem 1. Additionally, consider any $\eta \in [0, 9(1-\xi_0)\Delta_k/140)$. Then PILLAR with scaled hinge loss satisfies the same guarantees as Theorem 1 with $1/L_\ell = \gamma_0\left(1-\xi_0-\frac{140\eta}{9\Delta_k}\right)$ as long as the distributions of the private and public datasets are within η Total variation.*

D. Comparison with existing theoretical results and discussion

Existing works have offered a variety of techniques for achieving dimension-independent sample complexity. In the following, we review these works and compare them with our approach.

a) Generic private algorithms: [38] proposed the Noisy SGD algorithm $\mathcal{A}_{\text{Noisy}-\text{SGD}}$ that can privately learn linear halfspaces with margin γ on a private labelled dataset of size $O(\sqrt{d}/\alpha\epsilon\gamma)$. Recently, [28] showed that DP-SGD, a slightly adapted version of $\mathcal{A}_{\text{Noisy}-\text{SGD}}$, can achieve a dimension independent error bound under a low-dimensionality assumption termed as Restricted Lipschitz Continuity (RLC), which is more restrictive than our low-rank separability assumption. Similar results were showed in [39]. However, these methods cannot utilise public unlabelled data. [21] leverages public data for gradient clipping in DP-SGD. However, their method does not achieve dimension-independent error bound. The generic semi-private learner in [16] leverages unlabelled data to reduce the infinite hypothesis class to a finite α-net and applies exponential mechanism [40] to achieve $(\epsilon, 0)$-DP. Nonetheless, it is not computationally efficient and still requires a dimension-dependent labelled sample complexity $O(d/\alpha\epsilon)$.

b) Dimension reduction based private algorithms: Perhaps, most relevant to our work, [27] applies Johnson-Lindenstrauss (JL) transformation in the input space to reduce the dimension of a linear halfspace with margin γ from d to $O(1/\gamma)$ while preserving the margin in the lower-dimensional space. Private learning in the transformed low-dimensional space requires $O(1/\alpha\epsilon\gamma^2)$ labelled samples. Our algorithm removes the quadratic dependence on the inverse of the margin but pays the price of requiring the linear separator to align

[5]The estimate of ξ_k on pixel space should be taken with caution since classes are not linearly separable in the pixel space thereby only approximately satisfying the Large Margin assumption.

[6]Note that $\widetilde{O}$ neglects the logarithmic terms associated with $\frac{1}{\delta}$ and $\frac{1}{\beta}$.

with the top few principal components of the data. Specifically, the benefit in private sample complexity is significant when $k = o\left(\log(n)/\gamma^2\right)$, which is a realistic condition as k is often independent of n and γ is usually small.

Another approach to circumvent the dependency on the dimension is to apply dimension reduction techniques directly to the gradients. For smooth loss functions with ρ-Lipschitz and G-bounded gradients, [41] showed that applying PCA in the gradient space of DP-SGD [29] achieves dimension-independent labelled sample complexity $O\left(\frac{k\rho G^2}{\alpha\epsilon} + \frac{\rho^2 G^4 \log d}{\alpha}\right)$. However, this algorithm is computationally costly as it applies PCA in every gradient-descent step to a matrix whose size scales with the number of parameters. [24] proposed a computationally efficient method by applying JL transformation in the gradient space. While their method can eliminate the linear dependence of DP-SGD on dimension when the parameter space is the ℓ_1-ball, it leads to no improvement for parameter space being the ℓ_2-ball as in our setting. Gradient Embedding Perturbation (GEP) by [17] is another computationally efficient method that exploit the low-dimensionality of the gradient space with public unlabelled data. However, their analysis yields dimension independent guarantees only when a strict low-rank assumption of the gradient space is satisfied. Similar assumptions were leveraged by [25] who proposed a private adaptive gradient method to achieve dimension independent error bounds. Their final error bounds are very similar to [39]. We compare the assumptions in more detail in Appendix A-F.

c) Private PCA (DP-PCA): Another natural algorithm is to first project the private labelled data to its top k principal components estimated using DP-PCA on both the private and the public data, and then apply DP-SGD to learn a linear classifier in the k-dimensional space [29]. However, estimating the top principal components using DP-PCA on $O(n^U + n^L)$ samples in Theorem 1 introduces an irreducible error of $\Omega\left(\min\left\{\gamma_0^2 d, \frac{d}{\alpha\sqrt{k}}\right\}\right)$ in the estimated space (Theorem 5.4 of [42]), making the lower-dimensional space linearly inseparable for large d. Hence, the classification error of any linear classifier in the low dimensional space does not converge to zero using the same amount of data required for PILLAR.

Importantly, we compare against these algorithms in our experiments and show a consistent improvement, often by a wide margin, on a variety of datasets.

d) Non-private learning and dimensionality reduction: It is interesting to note that our algorithm may not lead to a similar improvement in the non-private case. We show a dimension-independent Rademacher-based labelled sample complexity bound for non-private learning of linear halfspaces. We use a non-private version of Algorithm 1 by replacing Noisy-SGD with Gradient Descent using the same loss function. As before, for any $\gamma_0 \in (0,1), \xi_0 \in (0,1)$, let $\mathcal{D}_{\gamma_0,\xi_0}$ be the family of distributions consisting of all (γ, ξ_k)-large margin low rank distributions with $\gamma \geq \gamma_0$ and $\xi_k \leq \xi_0$.

Proposition 3 (Non-DP learning)**.** *For any $\alpha, \beta \in (0, 1/4)$, and distribution $D \in \mathcal{D}_{\gamma_0,\xi_0}$, given a labelled dataset of size $\tilde{O}\left(1/\zeta\alpha^2\right)$ and unlabelled dataset of size $O\left(\log\frac{2}{\beta}/(\gamma_0\Delta_k)^2\right)$, the non-private version of $\mathcal{A}(k,\zeta)$ produces a linear classifier $\hat{w}$ such that with probability $1-\beta$*

$$\mathbb{P}_D\left[y\left\langle\hat{w}, x\right\rangle < 0\right] < \alpha,$$

where $\zeta = \gamma_0(1-\xi_0)$.

The result follows directly from the uniform convergence of linear halfspaces with Rademacher complexity. For example, refer to Theorem 1 in [43]. The labelled sample complexity in the above result shows that non-private algorithms do not significantly benefit from decreasing dimensionality[7]. We find this trend to be true in all our experiments in Figure 4 and 5.

In summary, our algorithm is computationally efficient and under certain (realistic) assumptions on the data, yields dimension independent private sample complexity. We also show through a wide variety of experiments in the following sections that the results transfer to practice in both common benchmarks as well as many newly designed challenging settings.

IV. Results on Standard Image Classification Benchmarks

In this section, we report performance of PILLAR on two standard benchmarks (CIFAR-10 and CIFAR-100 [44]) for private image classification. We demonstrate that in this setting, PILLAR outperforms all the competing methods. The improvement is especially remarkable for low ϵ values where there is a significant margin for improvement. For moderate values of ϵ, the improvement is more modest.

A. Experimental setting

The resolution difference between ImageNet-1K and CIFAR images can negatively impact the performance of training a linear classifier on pre-trained features. To mitigate this issue, we pre-process the CIFAR images using the ImageNet-1K transformation pipeline, which increases their resolution and leads to significantly improved performance. This technique is consistently applied throughout the paper whenever there is a notable resolution disparity between the pre-training and private datasets. For further details and discussions on pre-training at different resolutions, please refer to Appendix B-B.

We diverge from previous studies in the literature, such as those conducted by [12], [13], [15], by not exclusively focusing on values of $\epsilon > 1$. While a moderately large ϵ can be insightful for assessing the effectiveness of privately training deep neural networks with acceptable levels of accuracy, it is important to acknowledge that a large value of ϵ can result in loose privacy guarantees and consequently lack of willingness to share data [45] . The seminal work of [46] emphasizes that reasonable values of ϵ are expected to be less than 1. Moreover, [47] and [48] have already highlighted that $\epsilon > 1$ leads to loose upper bounds on the success probability of membership inference attacks. Finally, several

[7]However, this bound uses a standard Rademacher complexity result and may be lose. A tighter complexity bound may yield some dependence on the projected dimension.

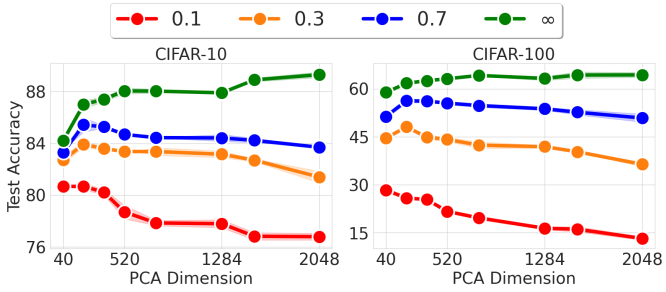


Fig. 4: DP training of linear classifier on SL pre-trained feature using the PRV accountant. For non-DP training ($\epsilon = \infty$), accuracy increases as dimension increases; opposite occurs for DP training ($\epsilon = \{0.1, 0.3, 0.7\}$). For results on additional feature-extractors see Appendix B-H.

recent deployments of DP have use values of ϵ smaller than 1[8]. Consequently, we focus on $\epsilon \in \{0.1, 0.3, 0.7, \infty\}$, where $\epsilon = \infty$ corresponds to the public training of the linear classifier. Nevertheless, for completeness and consistency with the current literature, we also present additional results for higher $\epsilon = \{1, 2\}$ in Appendix B-C.

B. Comparison with Existing Methods

We now compare the performance of PILLAR against several baselines that also leverage either public data or dimensionality reduction or both. We use the same PRV accountant for all methods [49]. For a comprehensive discussion on implementation details and the cross-validation ranges for hyper-parameters across all methods, refer to Appendix B-G.

a) Baselines: We consider the following baselines: i) *DP-SGD* [28], [29]: Trains a linear classifier privately using DP-SGD on the pre-trained features. ii) *JL* [27]: Applies a Johnson-Lindenstrauss (JL) transformation (without utilizing public data) to reduce the dimensionality of the features. We cross-validate various target dimensionalities and report the results for the most accurate one. iii) *AdaDPS* [18]: Utilizes the public *labeled* data to compute the pre-conditioning matrix for adaptive optimization algorithms. Since our algorithm does not require access to labels for the public data, we consider this comparison; nevertheless we report their performance. iv) *GEP* [17]: Employs the public *unlabeled* data to decompose the private gradients into a low-dimensional embedding and a residual component, subsequently perturbing them with noise of different magnitudes. v) *DP-PCA* [29] applies a step of DP-PCA (which consumes a fraction of the privacy budget) to compute the PCA components and then trains a linear classifier. We consider using $1\%, 25\%, 50\%$ of the privacy budget and report the results for the best choice.

Whenever public data is utilized, we employ 10% of the training data as public and remaining data as private. The official implementations of AdaDPS and GEP are used for our comparisons. Compared to baselines like AdaDPS and GEP, PILLAR introduces only one hyperparameter (the dimensionality k), making it less computationally expensive to cross-validate (as discussed in Appendix B-B). It is also extremely simple to implement, and therefore less prone to bugs that may invalidate the privacy guarantees.

In Appendix B-D, we discuss PATE [19], [20] and the reasons for not including it in our comparisons. For a detailed comparison with the work of [13], including the use of a different feature extractor to ensure a fair evaluation, we refer to Appendix B-B, where we demonstrate that our method is competitive, if not superior, while enjoying significantly more computational efficiency.

b) Results: In Table II, we compare our approach with other methods in the literature. Our results suggest that reducing dimensionality by using the JL transformation can only marginally ($\leq 1\%$ for both CIFAR-10 and CIFAR-100) improve over DP-SGD and sometimes even perform worse than DP-SGD. This may be attributed to the higher sample size required for the JL lemma to provide meaningful guarantees. Similarly, employing public data to pre-condition an adaptive optimizer does not result in improved performance for AdaDPS in most settings. The most competitive baseline is often GEP, however *PILLAR consistently outperforms all of them often with large margins.* For instance, consider the challenging setting of CIFAR-100 with $\epsilon = 0.1$. The performance of DP-trained classifiers is particularly low on this dataset because there are only 500 samples for each class. DP-SGD only achieves 13.2% accuracy for $\epsilon = 0.1$ whereas non-private accuracy is more than 80%. In this case no baseline yields performance significantly superior to DP-SGD except PILLAR, which is accurate by more than a factor of two. For $\epsilon = 0.3$, PILLAR outperforms the strongest baseline, GEP, by 6.8%. For $\epsilon = 0.7$, DP-SGD is again the strongest-baseline, and we outperform it by 3.0%.

C. Reducing dimension of projection k helps private learning

In Figure 4, we present the test accuracy of private and non-private trianing on CIFAR-10 and CIFAR-100 as the dimensionality of projection (PCA dimension) varies, with an initial embedding dimension of $k = 2048$. The principal components are computed on a public, unlabelled dataset that constitutes 10% of the full dataset, as allowed by Semi-Private Learning in Definition 4. Our results demonstrate that private training benefits from decreasing dimensionality, while non-private training either suffers in performance or remains stagnant. For example, using the SL feature extractor at $\epsilon = 0.1$ on CIFAR-10, the test accuracy of private training reaches 81.21% when $k = 40$, compared to 76.9% without dimensionality reduction. Similarly, for CIFAR-100 with the SL feature extractor at $\epsilon = 0.7$, the accuracy drops from 53.98% at $k = 200$ to 50.83% for the full dimension.

This observed dichotomy between private and non-private learning in terms of test accuracy and projection dimension aligns with Theorem 1 and Proposition 3. Theorem 1 indicates that the private test accuracy improves as the projection dimension decreases, as depicted in Figure 4. For non-private training with moderately large dimension, ($k \geq 520$), the test

[8]https://desfontain.es/privacy/real-world-differential-privacy.html

	Public Data	SL Pre-training				BYOL Pre-training		
Datasets		CIFAR10	CIFAR100	Flower-16	GTSRB	Dermnet	PCAM	Pneumonia
ϵ		0.1 0.3 0.7	0.1 0.3 0.7	0.1 0.3 0.7	0.1 0.3 0.7	0.1 0.3 0.7	0.1 0.3 0.7	0.1 0.3 0.7
DP-SGD [29]	None	76.8 81.4 83.7	13.2 36.4 50.9	46.2 72.2 82.4	36.0 46.8 59.6	14.4 22.4 28.0	80.1 81.4 81.6	71.3 73.6 79.2
DP-PCA [29]	None	72.1 77.5 81.2	10.2 34.9 48.3	46.2 69.6 76.1	35.1 50.0 58.0	15.4 22.9 27.6	80.1 81.9 81.1	66.7 68.3 79.3
JL [27]	None	76.1 82.1 84.1	13.7 37.6 51.3	43.5 70.4 80.9	36.3 53.3 62.1	14.3 22.3 28.1	78.3 78.7 79.7	65.2 70.0 76.9
GEP [17]	Unlabelled	80.1 83.2 84.5	12.4 41.2 45.2	59.1 78.5 82.8	38.7 58.2 61.2	18.3 24.6 27.7	79.1 82.0 81.7	75.9 78.5 82.9
AdaDPS [18]	Labelled	66.3 80.9 83.2	13.0 33.2 39.4	30.2 69.4 75.9	24.1 49.1 54.4	8.2 21.1 24.6	78.2 79.3 81.4	76.4 74.2 81.3
OURS	Unlabelled	**81.2 84.0 85.5**	**28.3 48.0 53.9**	**67.3 81.8 85.1**	**46.3 59.1 66.0**	**19.5 26.4 29.1**	**82.6 82.6 82.7**	**83.4 84.3 85.7**

TABLE II: Empirical comparison of PILLAR (OURS) against several baselines with different assumptions about the availability of public data. For the first four datasets (CIFAR-10, CIFAR-100, Flower-16, GTSRB), we use a SL pre-trained feature extractor, as it yields the best performance. For the last three datasets (Dermnet, PCAM, Pneumonia) we use a BYOL pre-trained feature extractor. In all cases, PILLAR outperforms all baselines under several levels of tightness of the privacy constraints ($\epsilon = \{0.1, 0.3, 0.7\}$). Baselines are implemented with the official, publicly available implementation when available. We use the PRV accountant. See Appendix B-G for more details.

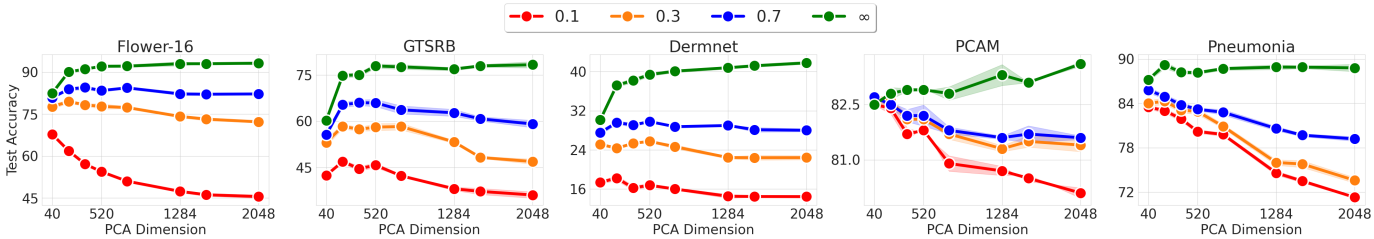


Fig. 5: Test Accuracy of DP classification on Flower-16, GTSRB, Dermnet, PCAM, and Pneumonia for best pre-training algorithm (SL pre-training for Flower-16 and GTSRB and BYOL for the remaining.). For results on additional feature-extractors refer to Appendix B-H.

accuracy remains largely constant. We discuss this theoretically in Proposition 3. The decrease in non-private accuracy for very small values of k is attributed to the increasing approximation error (i.e. how well can the best classifier in k dimensions represent the ground truth). This difference in behaviour between private and non-private learning for decreasing k values consistently holds in all our experiments and is one of the interesting observations of this paper. While we have demonstrated the effectiveness of our algorithm on the CIFAR-10 and CIFAR-100 benchmarks, as discussed in Section V, we acknowledge that this evaluation setting may not fully reflect the actual objectives of private learning.

V. Experimental Results Beyond Standard Benchmarks

In line with concurrent work [26], we raise concerns regarding the current trend of utilizing pre-trained feature extractors for differentially private training [12], [13]. It is common practice to evaluate differentially private algorithms for image classification by pre-training on ImageNet-1K and performing private fine-tuning on CIFAR datasets [12], [13]. However, we argue that this approach may not yield generalisable insights for privacy-sensitive scenarios. Both ImageNet and CIFAR datasets primarily consist of everyday objects, and the label sets of ImageNet are partially included within CIFAR. Such a scenario is unrealistic for many privacy-sensitive applications, such as medical, finance, and satellite data, where a large publicly available pre-training dataset with similar characteristics to the private data may not be accessible.

Moreover, public datasets are typically large-scale and easily scraped from the web, whereas private data is often collected on a smaller scale and subject to legal and competitive constraints, making it difficult to combine with other private datasets. Additionally, labeling private data, particularly in domains such as medical or biochemical datasets, can be costly. Therefore, evaluating the performance of privacy-preserving algorithms requires examining their robustness with respect to small dataset sizes. In order to address these considerations, we assess the performance of our algorithm on five additional datasets that exhibit varying degrees of distribution shift compared to the pre-training set, as described in Section V-A. Furthermore, we also demonstrate the robustness of our algorithm to minor distribution shifts between public unlabeled and private labeled data. In Section V-B, we show our algorithm is also robust to both small-sized private labeled datasets and public unlabeled datasets.

A. Effectiveness under Distribution Shift

a) Distribution Shift between Pre-Training and Private Data: We consider private datasets that exhibit varying levels of dissimilarity compared to the ImageNet pre-training dataset: Flower-16 [50], GTSRB [51], Pneumonia [52], a fraction (12.5%) of PCAM [53], and DermNet [54]. In Figure 1, we provide visual samples from each of these datasets. Flower-16

and GTSRB have minimal overlap with ImageNet-1K, with only one class in Flower-16 and 43 traffic signs aggregated into a single label in ImageNet-1K. The Pneumonia, PCAM, and DermNet datasets do not share any classes with ImageNet-1K. We also observe that, given a fixed pre-training distribution and model, different training procedures can have a different impact in the utility of the extracted features for each downstream classification task. Therefore, for each dataset we report the best performance produced by the most useful pre-training algorithm. Results for all the 5 pre-training strategies we consider and a discussion of how to choose them is relegated to Appendix B-H.

From Table II, we can see PILLAR outperforms all the considered baselines for all the ϵ values on all datasets. Before providing a more detailed discussion of the results, we would like to emphasize that no baseline consistently achieves the best performance across all these settings, in contrast to PILLAR, which proves to be a more consistent and widely applicable algorithm. On Flower-16, PILLAR achieves remarkable improvements. For $\epsilon = 0.1$, it outperforms the strongest baseline (GEP) by 8.2%. Similarly, on GTSRB we attain improvements of 3.9% over the runner-up (JL) for $\epsilon = 0.7$ and 8.4% with respect to GEP for $\epsilon = 0.1$. In the case of PCAM, although the relatively large training set size and the simplicity of the binary classification problem allows all classifiers to produce moderately high levels of accuracy (approximately 80%), our method is the only one to maintain an accuracy of approximately 82.6% across all the considered ϵ values, thus alleviating the utility degradation incurred by imposing tighter privacy constraints. In contrast, the Pneumonia dataset is a binary classification dataset with significantly less training data. In this case, competing techniques incur a significantly larger utility cost. For $\epsilon = 0.1$, the strongest baseline (AdaDPS) achieves 76.4%, while our method achieves 83.4%. *In summary, PILLAR consistently achieves the highest performance, often by a large margin, among all baselines for a wide range of datasets.*

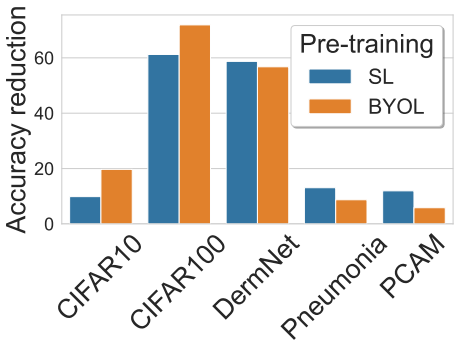


Fig. 6: Comparing the difference between the maximum attainable test accuracy with a publicly trained linear classifier and a DP trained linear classifier between using SL and BYOL pre-trained networks for different datasets. SL suffers a smaller drop in accuracy is more useful when the fine-tuning dataset contains daily-life objects and semantically overlap with ImageNet-1K, BYOL performs better otherwise otherwise.

In Figure 5, we demonstrate that reducing the dimensionality of the pre-trained models enhances differentially private training, irrespective of the private dataset used. Dimensionality reduction has a more pronounced effect on performance when tighter privacy constraints are imposed. It is worth noting that using dimensionality reduction can significantly degrade performance for non-DP training, similar to what we observed in CIFAR-10 and CIFAR-100.

b) When to use labels in pre-training: We also investigate the impact of different pre-training strategies on DP test accuracy. In our experiments, we have observed that some pre-trained models are more effective than others for specific datasets. To measure the maximum attainable accuracy with a publicly trained classifier, we compute the drop in performance, observed by training a DP classifier on BYOL pre-trained features, and the drop in performance for SL pre-trained features. We then plot the fractional reduction for both BYOL and SL across all the datasets for $\epsilon = 0.1$ in Figure 6. In Figure 11 we compare the relative reduction in performance when using Semi-supervised pre-training and BYOL pre-training. We find that datasets with daily-life objects and semantic overlap with ImageNet-1K benefit more from leveraging SL features and thus have a smaller reduction in accuracy for SL features compared to BYOL features. In contrast, datasets with little label overlap with ImageNet-1K benefit more from BYOL features, consistent with findings by [55].

PCA Data \ Pre-training	CIFAR10		CIFAR100	
	SL	BYOL	SL	BYOL
In-distribution	81.21	72.33	28.3	19.98
CIFAR-10v1	81.18	73.24	28.19	19.61

TABLE III: Distribution Shift between public (PCA) and private data: Comparison between using the same amount of in-distribution data (i.e. 10% of CIFAR-10 and CIFAR-100 respectively) and CIFAR-10v1 for computing the PCA projection ($\epsilon = 0.1$).

c) Distribution Shift between S^U and S^L: We demonstrate the effectiveness of our algorithm even when the public unlabeled data (used for computing the PCA projection matrix) is sourced from a slightly different distribution than the private labeled dataset. Specifically, we utilize the CIFAR-10v1 [56] dataset and present the results in Table III.

Notably, CIFAR-10v1 consists of only 2000 samples (4% of the training data), yet the results for both CIFAR-10 and CIFAR-100 remain essentially unchanged. This finding indicates that the data used to compute the PCA projection matrix does not necessarily have to originate from the same distribution as the private data and underscores that large amounts of public data are not required for our method to be effective.

B. Effectiveness in Low-Data Regimes

In privacy-critical settings such as medical contexts, there is often a limited availability of training data. For instance, the DermNet and pneumonia datasets contain only 12,000 and 3,400 training data points, respectively, which is significantly

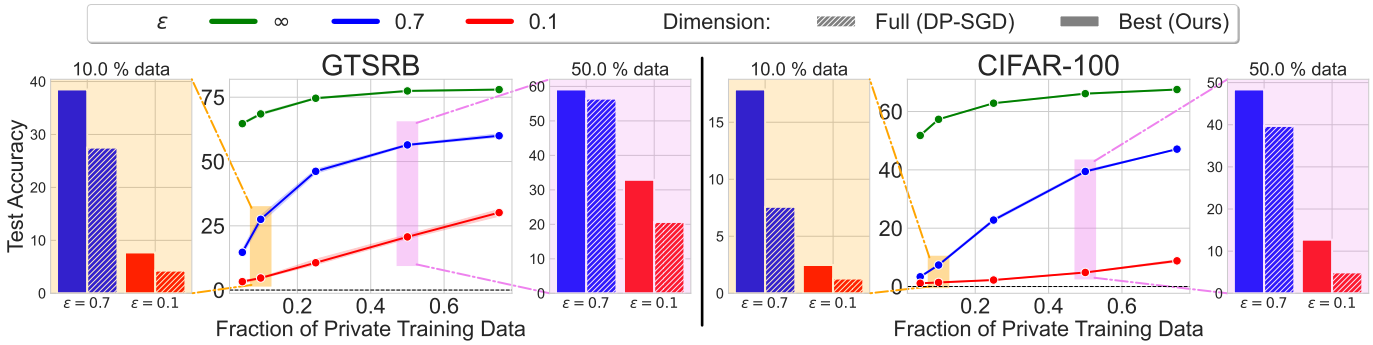


Fig. 7: For the GTSRB and CIFAR-100 datasets, in the central panel we report how the test accuracy varies as the amount of available private training data decreases (fraction of available data in $\{0.05, 0.1, 0.25, 0.5, 0.75\}$) for $\epsilon = 0.1$ and 0.7. We then select the cases in which 10% and 50% of the samples are available (left orange and right pink panels, respectively) and compare how PILLAR (solid bars) behaves with respect to DP-SGD (dashed bars). As it can be seen, PILLAR can alleviate the utility degradation caused by the reduced availability of private training data.

	CIFAR10		GTSRB	
Pre-training / PCA Data	SL	BYOL	SL	BYOL
1%	79.93	72.27	45.59	35.91
5%	81.02	72.33	45.64	35.88
10%	81.21	72.33	46.32	35.97

TABLE IV: Varying amounts of public (PCA) data: Performance of PILLAR with varying amounts of public (in distribution) data for computing the PCA projection ($\epsilon = 0.1$). The amount of public data is presented as a fraction of the whole available dataset.

smaller compared to datasets like CIFAR-10 with 50,000 samples. To examine the impact of reduced data (both private labeled and public unlabeled) on privacy, in this section we conduct ablations using varying fractions of public and private training data.

a) Less public unlabelled data: We demonstrate the robustness of our algorithm to reduced amounts of public unlabeled data used to compute the Principal Components.

In Table IV, we show the results of this ablation. As it can be seen, reducing the available public data does not yield dramatic variations in performance under the tightest privacy guarantees we consider ($\epsilon = 0.1$). For instance, for CIFAR-10 and GTSRB using a BYOL trained feature extractor, we observe the performance does not vary at all when the amount of available public data is reduced from 10% to 5% and 1%. For a SL trained feature extractor, we observe the performance only marginally decreases. For GTSRB, the performance reduces only by 0.93% when passing from 10% to 1% available public data, and of 1.28% on CIFAR-10 in the same setting.

b) Less private labelled data: In Figure 7, we present the performance of private and public training using different percentages of labeled private training data for CIFAR-100 and GTSRB. Our results indicate that under stringent privacy constraints ($\epsilon \in \{0, 7, 0.1\}$), the performance of DP training, without dimensionality reduction (DP-SGD), is considerably low. Conversely, even with a small percentage of training data, non-DP training demonstrates relatively high performance. By applying our algorithm in this scenario, we achieve significant performance improvements compared to using the full-dimensional embeddings. For instance, applying PCA with with $k = 40$ dimensions enhances the accuracy of our proposed algorithm from 7.53% to 18.3% on 10% of CIFAR-100, with $\epsilon = 0.7$ using the SL feature extractor. Similar improvements are also shown for GTSRB: when 10% of the data is available, the test accuracy improves from 27.3% to 38.4% for $\epsilon = 0.7$. To a smaller extent, improvements can be also observed when $\epsilon = 0.1$.

VI. Conclusion

In this paper, we consider the setting of semi-private learning where the learner has access to public unlabelled data in addition to private labelled data. This is a realistic setting in many circumstances e.g. where some people choose to make their data public. Under this setting, we proposed a new algorithm to learn linear halfspaces. Our algorithm uses a mix of PCA on unlabelled data and DP training on private data. Under reasonable theoretical assumptions, we have shown the proposed algorithm is (ϵ, δ)-DP and provably reduces the sample complexity. In practical applications, we performed an extensive set of experiments that show the proposed technique is effective when tight privacy constraints are imposed, even in low-data regimes and with a significant distribution shift between the pre-training and private distribution. In particular our algorithm consistently outperforms existing methods, often by a wide margin.

References

[1] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *IEEE Symposium on Security and Privacy (SP)*, 2017.

[2] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *ACM SIGSAC Conference on Computer and Communications Security*, 2022.

[3] N. Carlini, S. Chien, M. Nasar, S. Song, A. Terzis, and F. Tramèr, "Membership inference attacks from first principles," in *IEEE Symposium on Security and Privacy (SP)*, 2022.

[4] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," *Theory of Cryptography*, 2006.

[5] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?" *SIAM Journal on Computing*, 2011.

[6] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: the SuLQ framework," in *ACM symposium on Principles of database systems*, 2005.

[7] A. Beimel, K. Nissim, and U. Stemmer, "Characterizing the sample complexity of private learners," in *Innovations in Theoretical Computer Science (ITCS)*, 2013.

[8] ——, "Private learning and sanitization: Pure vs. approximate differential privacy," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2013.

[9] V. Feldman and D. Xiao, "Sample complexity bounds on differentially private learning via communication complexity," in *Conference on Learning Theory (COLT)*, 2014.

[10] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization." *Journal of Machine Learning Research (JMLR)*, 2011.

[11] R. Bassily, A. Smith, and A. Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *Annual Symposium on Foundations of Computer Science (FOCS)*, 2014.

[12] F. Tramer and D. Boneh, "Differentially private learning needs better features (or much more data)," in *International Conference on Learning Representations (ICLR)*, 2021.

[13] S. De, L. Berrada, J. Hayes, S. L. Smith, and B. Balle, "Unlocking high-accuracy differentially private image classification through scale," *arXiv:2204.13650*, 2022.

[14] X. Li, F. Tramer, P. Liang, and T. Hashimoto, "Large language models can be strong differentially private learners," in *International Conference on Learning Representations (ICLR)*, 2022.

[15] A. Kurakin, S. Chien, S. Song, R. Geambasu, A. Terzis, and A. Thakurta, "Toward training at ImageNet scale with differential privacy," *arXiv:2201.12328*, 2022.

[16] N. Alon, R. Bassily, and S. Moran, "Limits of private learning with access to public data," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[17] D. Yu, H. Zhang, W. Chen, and T.-Y. Liu, "Do not let privacy overbill utility: Gradient embedding perturbation for private learning," in *International Conference on Learning Representations (ICLR)*, 2021.

[18] T. Li, M. Zaheer, S. Reddi, and V. Smith, "Private adaptive optimization with side information," in *International Conference on Machine Learning (ICML)*, 2022.

[19] N. Papernot, M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," in *International Conference on Learning Representations (ICLR)*, 2017.

[20] N. Papernot, S. Song, I. Mironov, A. Raghunathan, K. Talwar, and U. Erlingsson, "Scalable private learning with PATE," in *International Conference on Learning Representations (ICLR)*, 2018.

[21] M. Nasr, S. Mahloujifar, X. Tang, P. Mittal, and A. Houmansadr, "Effectively using public data in privacy preserving machine learning," in *Proceedings of the 40th International Conference on Machine Learning*, 2023, pp. 25 718–25 732.

[22] E. Amid, A. Ganesh, R. Mathews, S. Ramaswamy, S. Song, T. Steinke, V. M. Suriyakumar, O. Thakkar, and A. Thakurta, "Public data-assisted mirror descent for private model training," in *International Conference on Machine Learning (ICML)*, 2022.

[23] D. Yu, H. Zhang, W. Chen, J. Yin, and T. Liu, "Large scale private learning via low-rank reparametrization," in *International Conference on Machine Learning (ICML)*, 2021.

[24] S. P. Kasiviswanathan, "SGD with low-dimensional gradients with applications to private and distributed learning," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2021.

[25] P. Kairouz, M. Ribero, K. Rush, and A. Thakurta, "Fast dimension independent private adagrad on publicly estimated subspaces," *arXiv:2008.06570*, 2020.

[26] F. Tramèr, G. Kamath, and N. Carlini, "Considerations for differentially private learning with large-scale public pretraining," *arXiv:2212.06470*, 2022.

[27] H. L. Nguyen, J. R. Ullman, and L. Zakynthinou, "Efficient private algorithms for learning large-margin halfspaces," in *Algorithmic Learning Theory (ALT)*, 2020.

[28] X. Li, D. Liu, T. Hashimoto, H. A. Inan, J. Kulkarni, Y. Lee, and A. G. Thakurta, "When does differentially private learning not suffer in high dimensions?" in *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[29] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[30] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[31] H. Asi, J. C. Duchi, A. Fallah, O. Javidbakht, and K. Talwar, "Private adaptive gradient methods for convex optimization," *CoRR*, vol. abs/2106.13756, 2021.

[32] C. Jensen, C. Potts, and C. Jensen, "Privacy practices of internet users: Self-reports versus observed behavior," *International Journal of Human-Computer Studies*, 2005.

[33] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv:2003.04297*, 2020.

[34] I. Z. Yalniz, H. Jégou, K. Chen, M. Paluri, and D. Mahajan, "Billion-scale semi-supervised learning for image classification," *arxiv:1905.00546*, 2019.

[35] X. Gu, G. Kamath, and Z. S. Wu, "Choosing public datasets for private machine learning via gradient subspace distance," *arXiv:2303.01256*, 2023.

[36] A. Panda, X. Tang, V. Sehwag, S. Mahloujifar, and P. Mittal, "Dp-raft: A differentially private recipe for accelerated fine-tuning," *arXiv: 2212.04486*, 2022.

[37] S. Song, O. Thakkar, and A. Thakurta, "Characterizing private clipped gradient descent on convex generalized linear problems," *arxiv:2006.06783*, 2020.

[38] R. Bassily, A. D. Smith, and A. Thakurta, "Private empirical risk minimization, revisited," *ICML Workshop on Learning, Security and Privacy*, 2014.

[39] S. Song, T. Steinke, O. Thakkar, and A. Thakurta, "Evading the curse of dimensionality in unconstrained private glms," 2021.

[40] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Annual Symposium on Foundations of Computer Science (FOCS)*, 2007.

[41] Y. Zhou, S. Wu, and A. Banerjee, "Bypassing the ambient dimension: Private SGD with gradient subspace identification," in *International Conference on Learning Representations (ICLR)*, 2021.

[42] X. Liu, W. Kong, P. Jain, and S. Oh, "Dp-pca: Statistically optimal and differentially private pca," in *Advances in Neural Information Processing Systems*, 2022.

[43] P. Awasthi, N. Frank, and M. Mohri, "On the Rademacher complexity of linear hypothesis sets," *arXiv:2007.11045*, 2020.

[44] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[45] P. Nanayakkara, M. A. Smart, R. Cummings, G. Kaptchuk, and E. M. Redmiles, "What are the chances? explaining the epsilon parameter in differential privacy," in *USENIX Security Symposium (USENIX Security)*, 2023.

[46] C. Dwork, "A firm foundation for private data analysis," *Communications of the ACM*, 2011.

[47] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *IEEE computer security foundations symposium (CSF)*, 2018.

[48] M. Nasr, S. Songi, A. Thakurta, N. Papemoti, and N. Carlin, "Adversary instantiation: Lower bounds for differentially private machine learning," in *IEEE Symposium on Security and Privacy (SP)*, 2021.

[49] S. Gopi, Y. T. Lee, and L. Wutschitz, "Numerical composition of differential privacy," in *Conference on Neural Information Processing Systems (NeurIPS)*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021.

[50] "Flowers dataset," https://tinyurl.com/2p8vpsp2, 2021.

[51] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark," in *International Joint Conference on Neural Networks*, 2013.

[52] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan *et al.*, "Identifying medical

diagnoses and treatable diseases by image-based deep learning," *cell*, 2018.
[53] B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling, "Rotation equivariant CNNs for digital pathology," *arXiv:1806.03962*, 2018.
[54] "Dataset for 23 skin lesions," https://www.kaggle.com/datasets/shubhamgoel27/dermnet, 2019.
[55] Y. Shi, I. Daunhawer, J. E. Vogt, P. H. Torr, and A. Sanyal, "How robust are pre-trained models to distribution shift?" in *International Conference on Learning Representations (ICLR)*, 2023.
[56] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do CIFAR-10 classifiers generalize to CIFAR-10?" *arxiv:1806.00451*, 2018.
[57] L. Zwald and G. Blanchard, "On the convergence of eigenspaces in kernel principal component analysis," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2005.
[58] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. The MIT Press, 2012.
[59] P. Bartlett, M. Jordan, and J. Mcauliffe, "Large margin classifiers: Convex loss, low noise, and convergence rates," in *Advances in Neural Information Processing Systems*, 2003.
[60] M. Anthony and P. L. Bartlett, *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.
[61] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.
[62] S. O. Gharan, "Low rank approximation," Lecture notes for CSE 521: Design and Analysis of Algorithms I, 2017.
[63] A. Yousefpour, I. Shilov, A. Sablayrolles, D. Testuggine, K. Prasad, M. Malek, J. Nguyen, S. Ghosh, A. Bharadwaj, J. Zhao, G. Cormode, and I. Mironov, "Opacus: User-friendly differential privacy library in PyTorch," *arXiv:2109.12298*, 2021.
[64] R. Wightman, "Pytorch image models," https://github.com/rwightman/pytorch-image-models, 2019.
[65] V. G. T. da Costa, E. Fini, M. Nabi, N. Sebe, and E. Ricci, "solo-learn: A library of self-supervised methods for visual representation learning," *Journal of Machine Learning Research (JMLR)*, 2022.
[66] Y. Zhu, X. Yu, M. Chandraker, and Y.-X. Wang, "Private-knn: Practical differential privacy for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
[67] C. Mühl and F. Boenisch, "Personalized PATE: Differential privacy for machine learning with individual privacy guarantees," in *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2022.
[68] F. Boenisch, C. Mühl, A. Dziedzic, R. Rinberg, and N. Papernot, "Have it your way: Individualized privacy assignment for DP-SGD," *arXiv:2303.17046*, 2023.
[69] J. Vanschoren, J. N. Van Rijn, B. Bischl, and L. Torgo, "Openml: networked science in machine learning," *ACM SIGKDD Explorations Newsletter*, 2014.
[70] S. Gopi, Y. T. Lee, and L. Wutschitz, "Numerical composition of differential privacy," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.
[71] W. B. Johnson, "Extensions of lipschitz mappings into hilbert space," *Contemporary Mathematics*, 1984.
[72] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov, "Differential privacy has disparate impact on model accuracy," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.
[73] R. Cummings, V. Gupta, D. Kimpara, and J. Morgenstern, "On the compatibility of privacy and fairness," in *Conference on User Modeling, Adaptation and Personalization*, 2019.
[74] A. Sanyal, Y. Hu, and F. Yang, "How unfair is private learning?" in *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2022.

APPENDIX A
PROOFS

A. Noisy SGD

In this section, we present Algorithm 2, an adapted version of the Noisy SGD algorithm from [38] for d-dimensional linear halfspaces $\mathcal{H}^d$, that is used as a sub-procedure in Algorithm 1. Algorithm 2 first applies a base procedure $\mathcal{A}_{Base}$ on $\mathcal{H}^d$ for k times to generate a set of k results while preserving (ϵ, δ)-DP, and then applies the exponential mechanism $\mathcal{M}_E$ to output one final result from the set.

Algorithm 2 $\mathcal{A}_{Noisy-SGD}(S^L, \ell, (\epsilon, \delta), \beta)$

procedure $\mathcal{A}_{Noisy-SGD}(S^L, \ell, (\epsilon, \delta), \beta)$
  Input: a labelled dataset S^L, a loss function ℓ, privacy parameters ϵ, δ, and the failure probability β.
  Set $k = \lceil \log 1/\beta \rceil$.
  for $i = 1$ to k **do**
    $\hat{w}^{(i)} \leftarrow \mathcal{A}_{Base}(S^L, \ell, (\epsilon/k, \delta/k))$
  end for
  Let $\mathcal{O} \leftarrow \{\hat{w}^{(1)}, \ldots, \hat{w}^{(k)}\}$.
  $\hat{w} \leftarrow \mathcal{M}_E(S^L, -\ell, \mathcal{O}, \epsilon)$.
  Output: $\hat{w}$
end procedure
procedure $\mathcal{A}_{Base}((S^L, \ell, (\epsilon, \delta)))$
  Input: a labelled dataset S^L, a loss function ℓ, privacy parameters ϵ, δ.
  Let $\mathcal{L}$ be the Lipschitz coefficient of the loss function ℓ and n^L be the size of S^L.
  Set noise variance $\sigma^2 \leftarrow \frac{32L^2\left(n^L\right)^2 \log(n^L/\delta)\log(1/\delta)}{\epsilon}$.
  Randomize $\hat{w}^0 \in \mathcal{H}^d$.
  Set the learning rate function $\eta(t) = \frac{1}{\sqrt{t(n^L)^2\mathcal{L}^2 + m\sigma^2}}$.
  for $t = 1$ to $(n^L)^2 - 1$ **do**
    Uniformly choose $(x, y) \in S^L$.
    Update $\hat{w}^{t+1} = \Pi_{\mathcal{W}}\left(\hat{w}_t - \eta(t)[n^L \nabla \ell(\hat{w}^t; (x, y)) + \xi]\right)$ where $\xi \sim N(0, \mathbb{I}_d\sigma^2)$.
  end for
  Output: $\hat{w} = \hat{w}^{(n^L)^2}$
end procedure
procedure $\mathcal{M}_E(S^L, \ell, \mathcal{O}, \epsilon)$
  Input: a dataset S^L, a loss function ℓ, an set of parameters $\mathcal{O}$, and a privacy parameter ϵ.
  Set the global sensitivity as $\Delta_U = \max_{S,S'} \max_{w\in\mathcal{O}} |\ell(S, w) - \ell(S', w)|$, for any S, S' of size $|S^L|$ differing at exactly one entry.
  Output: $w \in \mathcal{O}$ with probability proportional to $\exp\left(\frac{\epsilon\ell(S^L, w)}{2\Delta_U}\right)$.
end procedure

In Lemma 1, we state the privacy guarantee and the high probability upper bound on the excess error of the adpated version of Noisy SGD (Algorithm 2). This is a corollary of Theorem 2.4 in [11], which provides an upper bound on the expected excess risk of $\mathcal{A}_{Base}$. The proof of Lemma 1 follows directly from Markov inequality and the post-processing property of DP (Lemma 3), as described in Appendix D of [11].

Lemma 1 (Theoretical guarantees of Noisy SGD [38])**.** *Let the loss function ℓ be $\mathcal{L}$-Lipschitz and $\mathcal{H}^d$ be the d-dimensional linear halfspace with diameter* 1*. Then $\mathcal{A}_{Noisy-SGD}$ is (ϵ, δ)-DP, and with probability $1 - \beta$, its output $\hat{w}$ satisfies the following upper bound on the excess risk,*

$$\sum_{(x,y)\in S} \ell(\hat{w}, (x, y)) - \sum_{(x,y)\in S} \ell(w^\star, (x, y)) = \frac{\mathcal{L}\sqrt{d}}{\epsilon} \cdot \textit{polylog}(n, \frac{1}{\beta}, \frac{1}{\delta}),$$

for a labelled dataset S of size n. Here, $w^\star$ is the empirical risk minimizer $w^\star = \textit{argmin}_{w\in\mathcal{C}} \sum_{(x,y)\in S} \ell(w, (x, y))$.

B. Theoretical results under no distribution shift and proofs

In this section, we provide a proof for Theorem 1, which demonstrates that PILLAR is (ϵ, δ)-DP with respect to the private dataset and can achieve accuracy with only a modest amount of private data. To establish Theorem 1, we begin by

proving Lemma 2. This lemma shows that PILLAR attains a convergence guarantee in excess loss for all Lipschitz continuous loss functions in learning the linear halfspace $\mathcal{H}^d$.

Lemma 2. *Let $k \leq d \in \mathbb{N}, \gamma_0 \in (0,1)$, and $\xi_0 \in (0,1)$. Consider the family of distributions $\mathcal{D}_{\gamma_0,\xi_0}$ which consists of all (γ, ξ_k)-large margin low rank distributions over $\mathcal{X}_d \times \mathcal{Y}$, where $\gamma \geq \gamma_0$ and $\xi_k \leq \xi_0$. For any $\alpha \in (0,1), \beta \in (0, 1/4)$, $\epsilon \in (0, 1/\sqrt{k})$, and $\delta \in (0,1)$, PILLAR $\mathcal{A}_{\epsilon,\delta}(k,\ell)$, described by Algorithm 1 with an L-Lipschitz loss function ℓ in step 5, is (ϵ,δ)-DP and outputs an estimator $\hat{w}$ satisfying*

$$\mathbb{P}_{(S^U,S^L)\sim D, \hat{w}\sim\mathcal{A}_{\epsilon,\delta}}\left[\mathbb{E}_{(x,y)\sim D}\left[\ell(\hat{w};(x,y))\right] - \min_{w\in B_2^d}\mathbb{E}_{(x,y)\sim D}\left[\ell(w;(x,y))\right] \leq \alpha\right] \geq 1-\beta,$$

given a public unlabelled and private labelled sample S^U, S^L from distribution D of size

$$n^U = O\left(\frac{\log 2/\beta}{(1-\xi_0)^2\Delta_k^2}\right), n^L = \widetilde{O}\left(\left(\frac{1}{\alpha^2}+\frac{1}{\alpha\epsilon}\right)L\sqrt{k}\right).$$

Proof. **Privacy guarantee** Algorithm $\mathcal{A}_{\epsilon,\delta}(k,\zeta)$ computes the transformation matrix $\hat{A}_k$ on the public unlabelled dataset. This step is independent of the labelled data S^L and has no impact on the privacy with respect to S^L. $\mathcal{A}_{\text{Noisy}-\text{SGD}}$ ensures the operations on the labelled dataset S^L to output v_k is (ϵ,δ)-DP with respect to S^L (Lemma 1). The final output $\hat{w} = \hat{A}_k v_k$ is attained by post-processing of v_k and preserves the privacy with respect to S^L by the post-processing property of differential privacy (Lemma 3).

Lemma 3 (Post-processing [4])**.** *For every (ϵ,δ)-DP algorithm $\mathcal{A}: S \to \mathcal{Y}$ and every (possibly random) function $f: \mathcal{Y} \to \mathcal{Y}'$, $f \circ \mathcal{A}$ is (ϵ,δ)-DP.*

Accuracy guarantee By definition, all distributions $D_{\gamma,\xi_k} \in \mathcal{D}_{\gamma_0,\xi_0}$ are (γ,ξ_k)-large margin low rank for some $\gamma \geq \gamma_0, \xi_k \leq \xi_0$. Let the empirical covariance matrix of D_{γ,ξ_k} calculated with the unlabelled dataset S^U be $\widehat{\Sigma} = \frac{1}{n^U}\sum_{x\in S^U}(x-\bar{x})(x-\bar{x})^\top$ and $\hat{A}_k \in \mathbb{R}^{d\times k}$ be the projection matrix whose i^{th} column is the i^{th} eigenvector of $\widehat{\Sigma}$. Let Σ be the population covariance matrix and similarly, let A_k the matrix of top k eigenvectors of Σ.

For any distribution $D_{\gamma,\xi_k} \in \mathcal{D}_{\gamma_0,\xi_0}$, let $D_{X,(\gamma,\xi_k)}$ be the marginal distribution of X and $w^\star$ be the large margin linear classifier that is guaranteed to exist by Definition 3. The margin after projection by $\hat{A}_k$ is lower bounded by $\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2}$ for any $z \in \operatorname{supp}\left(D_{X,(\gamma,\xi_k)}\right)$.

We will first derive a high-probability lower bound for this term to show that, after projection, with high probability, the projected distribution still has a large margin. Then, we will invoke existing algorithms in the literature with the right parameters, to privately learn a large margin classifier in this low-dimensional space.

Let z be any vector in $\operatorname{supp}\left(D_{X,(\gamma,\xi_k)}\right)$. We can write $z = a_z w^\star + b^\perp$ for some a_z where $b^\perp$ is in the nullspace of $w^\star$. Then, it is easy to see that using the large-margin property in Definition 3, we get

$$ya_z = \frac{\langle w^\star, z\rangle}{\|w^\star\|_2\|z\|_2} \geq \gamma \geq \gamma_0. \tag{1}$$

Then, we lower bound $\dfrac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2}$ as

$$\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2} \overset{(a)}{=} \frac{ya_z\left\|\hat{A}_k^\top w^\star\right\|_2^2}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2} \overset{(b)}{\geq} \gamma_0\left\|\hat{A}_k^\top w^\star\right\|_2 \tag{2}$$

where step (a) is due to $\left\langle w^\star, b^\perp\right\rangle = 0$ and step (b) follows from $\left\|\hat{A}_k^\top z\right\|_2 \leq \left\|\hat{A}_k\right\|_{\text{op}}\|z\|_2 \leq 1$ and Equation (1).

To lower bound $\left\|\hat{A}_k^\top w^\star\right\|_2$, note that

$$\begin{aligned}
\left\|\hat{A}_k^\top w^\star\right\|_2 = \left\|\hat{A}_k\hat{A}_k^\top w^\star\right\|_2 &\geq \left\|A_kA_k^\top w^\star\right\|_2 - \left\|\hat{A}_k\hat{A}_k^\top w^\star - A_kA_k^\top w^\star\right\|_2 && \text{by Triangle Inequality}\\
&\geq \left\|A_kA_k^\top w^\star\right\|_2 - \left\|\hat{A}_k\hat{A}_k^\top - A_kA_k^\top\right\|_F\|w^\star\|_2 && \text{by Cauchy Schwarz Inequality}\\
&\geq 1-\xi_k - \left\|\hat{A}_k\hat{A}_k^\top - A_kA_k^\top\right\|_F.
\end{aligned} \tag{3}$$

where the last step follows from the low rank assumption in Definition 3 and observing that $\|w^\star\|_2 = 1$.

To upper bound $\left\|\hat{A}_k\hat{A}_k^\top - A_kA_k^\top\right\|_F$, we use Lemma 4.

Lemma 4 (Theorem 4 in [57]). *Let D be a distribution over $\{x \in \mathbb{R}^d | \ \|x\|^2 \leq 1\}$ with covariance matrix Σ and zero mean $\mathbb{E}_{x\sim D}[x] = 0$. For a sample S of size n from D, let $\widehat{\Sigma} = \frac{1}{n}\sum_{x\in S} xx^T$ be the empirical covariance matrix. Let $A_k, \hat{A}_k$ be the matrices whose columns are the first k eigenvectors of Σ and $\widehat{\Sigma}$ respectively and let $\lambda_1(\Sigma) > \lambda_2(\Sigma) > \ldots > \lambda_d(\Sigma)$ be the ordered eigenvalues of Σ. For any $k > 0, \beta \in (0,1)$ such that $\lambda_k(\Sigma) > 0$ and $n \geq \frac{16\left(1+\sqrt{\beta/2}\right)^2}{(\lambda_k(\Sigma)-\lambda_{k+1}(\Sigma))^2}$, we have that with probability at least $1 - e^{-\beta}$,*

$$\left\|A_kA_k^T - \hat{A}_k\hat{A}_k^T\right\|_F \leq \frac{4\left(1+\sqrt{\frac{\beta}{2}}\right)}{(\lambda_k(\Sigma) - \lambda_{k+1}(\Sigma))\sqrt{n}}.$$

It guarantees that with probability $1 - \frac{\beta}{2}$,

$$\left\|A_kA_k^\top - \hat{A}_k\hat{A}_k^\top\right\|_F \leq \frac{4\left(1+\sqrt{\frac{\log(2/\beta)}{2}}\right)}{(\lambda_k(\Sigma) - \lambda_{k+1}(\Sigma))\sqrt{n^U}} \leq \frac{1-\xi_0}{10}. \tag{4}$$

where the last inequality follows from choosing the size of unlabelled data $n^U \geq \frac{1600\left(1+\sqrt{\frac{\log(2/\beta)}{2}}\right)^2}{(1-\xi_0)^2(\Delta_{\min}\lambda_k)^2}$.

Substituting Equation (4) into Equation (3), we get that with probability $1 - \frac{\beta}{2}$,

$$\left\|\hat{A}_k^\top w^\star\right\|_2 \geq 1 - \xi_k - \frac{1-\xi_0}{10} \geq 1 - \xi_0 - \frac{1-\xi_0}{10} = 0.9(1-\xi_0) \tag{5}$$

Plugging Equation (5) into Equation (2), we derive a high-probability lower bound on the distance of any point to the decision boundary in the transformed space. For all $z \in \operatorname{supp}\left(D_{X,(\gamma,\xi_k)}\right)$,

$$\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2} \geq 0.9\gamma_0(1-\xi_0). \tag{6}$$

This implies that the margin in the transformed space is at least $0.9\gamma_0(1-\xi_0)$.

For a halfspace with parameter $v \in B_2^k$, denote the empirical loss on a dataset S by $\hat{L}(w;S) = \frac{1}{|S|}\sum_{(x,y)\in S}\ell(w,(x,y))$ and the loss on the distribution D by $L(w;D) = \mathbb{P}_{(x,y)\sim D}[\ell(w,(x,y))]$. Let D_k be the k-dimension transformation of the original distribution D obtained by projecting each $x \in \mathcal{X}$ to $\hat{A}_k^\top x$.

By the convergence bound in Lemma 1 for $\mathcal{A}_{\text{Noisy}-\text{SGD}}$, we have with probability $1 - \frac{\beta}{2}$, $\mathcal{A}_{\text{Noisy}-\text{SGD}}$ outputs a hypothesis $v_k \in B_2^k$ and $\hat{w} = A_kv_k \in B_w^d$ such that

$$\hat{L}(\hat{w};S^L) - \hat{L}(w_{ERM};S^L) \overset{(a)}{=} \hat{L}(v_k;S_k^L) - \hat{L}(v_{ERM};D_k) = O\left(\frac{L\sqrt{k}}{n^L\epsilon}\text{polylog}\left(n^L, \frac{1}{\delta}, \frac{1}{\beta}\right)\right) \tag{7}$$

where $w_{ERM} = \text{argmin}_{w\in B_2^d}\hat{L}(w;S^L)$ and $v_{ERM} = \text{argmin}_{v\in B_2^k}\hat{L}(v;S_k^L)$.

Let $w^\star$ be the ground truth of the given distribution. The generalization error can be decomposed as

$$\begin{aligned} L(\hat{w}) - L(w^\star) &= \left(L(\hat{w}) - \hat{L}(\hat{w})\right) + \left(\hat{L}(\hat{w}) - \hat{L}(w_{ERM})\right) + \left(\hat{L}(w_{ERM}) - \hat{L}(w^\star)\right) + \left(\hat{L}(w^\star) - \hat{L}(w^\star)\right) \\ &\overset{(a)}{\leq} \underbrace{\left(L(\hat{w}) - \hat{L}(\hat{w})\right)}_{(a)} + \underbrace{\left(\hat{L}(\hat{w}) - \hat{L}(w_{ERM})\right)}_{(b)} + \underbrace{\left(\hat{L}(w^\star) - \hat{L}(w^\star)\right)}_{(c)} \end{aligned} \tag{8}$$

where step (a) follows as $\hat{L}(w_{ERM}) - \hat{L}(w^\star) \leq 0$ by the definition of w_{ERM}.

We have shown in Equation (7) that the second term (b) is upper bounded by $\frac{\alpha}{2}$ for $n^L = \tilde{O}\left(\frac{L\sqrt{k}}{\alpha\epsilon}\right)$. It remains to bound the generalization error of linear halfspace $\mathcal{H}^d$ for L-Lipschitz loss function, ie. term (a) and term (c). That is, we need to show that the empirical error of a linear halfspace is a good approximation of the error on the distribution. To achieve this, we apply uniform convergence bound using Rademacher complexity [58].

With probability $1 - \frac{\beta}{4}$,

$$\sup_{w\in B_2^d}\left(\mathbb{E}_{x,y\sim D}\ell(w;(x,y)) - \frac{1}{n^L}\sum_{(x,y)\in S^L}\ell(w;(x,y))\right) \leq 2\mathfrak{R}_{S^L}(\mathcal{H}_\ell) + \sqrt{\frac{3\log\frac{8}{\beta}}{2n^L}}, \tag{9}$$

where $\mathcal{H}_\ell = \{h_w(x,y) = \ell(w,(x,y)) | w \in B_2^d\}$ is the composition of the loss function with the linear halfspace.

$$\mathfrak{R}_{S^L}(\mathcal{H}_\ell) \leq L\mathfrak{R}_{S^L}(\mathcal{H}^d) = \frac{L}{n^L}. \tag{10}$$

Substituting Equation (10) into Equation (9), we can upper bound both term (a) and (c) by $\frac{\alpha}{4}$ with probability at least $1 - \frac{\beta}{2}$ for $n^L \geq \frac{L}{\alpha^2}\text{polylog}\left(() \frac{4}{\beta}\right)$, i.e.

$$L(\hat{w}) - \hat{L}(\hat{w}) \leq \frac{\alpha}{4}, \quad \hat{L}(w^\star) - L(w^\star) \leq \frac{\alpha}{4}. \tag{11}$$

Combining Equation (8), Equation (7) and Equation (11) concludes the proof.

□

In the following, we use Lemma 2 to prove Theorem 1. Recall that we define cross entropy loss and scaled hinge loss in Table I.

Theorem 1. *Let $k \leq d \in \mathbb{N}, \gamma_0 \in (0,1)$, and $\xi_0 \in (0,1)$. Consider the family of distributions $\mathcal{D}_{\gamma_0,\xi_0}$ which consists of all (γ, ξ_k)-large margin low rank distributions over $\mathcal{X}_d \times \mathcal{Y}$, where $\gamma \geq \gamma_0$ and $\xi_k \leq \xi_0$. For any $\alpha \in (0,1), \beta \in (0, 1/4)$, $\epsilon \in (0, 1/\sqrt{k})$, and $\delta \in (0,1)$, PILLAR with scaled hinge loss or cross entropy loss, is an $(\alpha, \beta, \epsilon, \delta)$-semi-private learner for linear halfspaces $\mathcal{H}^d$ on $\mathcal{D}_{\gamma_0,\xi_0}$ with sample complexity*

$$n^U = O\left(\frac{\log 2/\beta}{(1-\xi_0)^2\Delta_k^2}\right), n^L = \widetilde{O}\left(\frac{L_\ell\sqrt{k}}{\alpha\epsilon}\right)$$

where Δ_k denote the gap between the k^{th} and the $k+1^{\text{th}}$ eigenvalue of the population covariance matrix, and L_ℓ is the Lipschitz coefficient of the loss function ℓ[9]*.*

Proof. **Guarantees for PILLAR with (scaled) hinge loss function:** Note that the (scaled) hinge loss function ℓ_ζ^h defined in Table I is $\frac{1}{0.9\gamma_0(1-\xi_0)}$-Lipschitz. Substituting $L_\ell = \frac{1}{0.9\gamma_0(1-\xi_0)}$ into the sample complexity in Lemma 2 upper bounds the excess hinge loss of PILLAR's output $\hat{w}$ with probability at least $1-\beta$, i.e.

$$\begin{aligned} &\mathbb{E}_{(x,y)\sim D}\left[\ell(\hat{w};(x,y))\right] - \min_{w\in B_2^d}\mathbb{E}_{(x,y)\sim D}\left[\ell(w;(x,y))\right] \\ \overset{(a)}{=}&\mathbb{E}_{(x,y)\sim D}\left[\ell(v_k;(A_k^\top x,y))\right] - \min_{v\in B_2^k}\mathbb{E}_{(x,y)\sim D}\left[\ell(v;(A_k^\top x,y))\right] \leq \alpha \end{aligned} \tag{12}$$

where step (a) follows from the definition of $\hat{w} = A_k v_k$ by the last step in PILLAR using the same notation as in the proof of Lemma 2.

By Equation (5) following the same argument as in the proof of Lemma 2, the k-dimensional space projected by A_k has a positive margin at least $0.9\gamma_0(1-\xi_0)$. Thus, the empirical risk minimizer in the low-dimensional space is zero, i.e.

$$\mathbb{E}_{(x,y)\sim D}\left[\ell(\hat{w};(x,y))\right] = \min_{v\in B_2^k}\mathbb{E}_{(x,y)\sim D}\left[\ell(v;(A_k^\top x,y))\right] = 0. \tag{13}$$

Then, we can upper bound the empirical 0-1 error by the empirical (scaled) hinge loss in the k-dimensional transformed space, For $n^L = O\left(\frac{\sqrt{k}}{\alpha\epsilon\gamma_0(1-\xi_0-0.1\gamma_0)}\text{polylog}\left(\frac{1}{\delta},\frac{1}{\epsilon},\frac{1}{\beta},\frac{1}{\alpha},\frac{1}{\gamma_0},\frac{1}{\xi_0},k,n^L\right)\right)$, with probability $1-\frac{\beta}{4}$,

$$\begin{aligned} \mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\langle x,\hat{w}\rangle\}\right] &= \mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\left\langle A_k^\top x, v_k\right\rangle\}\right] \\ &\leq \mathbb{E}_{(x,y)\sim D}\left[\ell(v_k;(A_k^\top x,y))\right] \end{aligned} \tag{14}$$

Combining Equation (12), Equation (13) and Equation (14) concludes the proof.

Guarantees for PILLAR with cross entropy loss: As cross entropy loss function ℓ_{CN} defined in Table I is 2-Lipschitz, directly applying Lemma 2 shows that excess cross-entropy loss ℓ_{CN} is upper bounded by $\frac{\alpha}{2}$ with the given public unlabelled and private labelled samples, i.e.

$$\mathbb{P}_{(S^U,S^L)\sim D,\hat{w}\sim\mathcal{A}_{\epsilon,\delta}}\left[\mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(\hat{w};(x,y))\right] - \min_{w\in B_2^d}\mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(w;(x,y))\right] \leq \frac{\alpha}{2}\right] \geq 1-\beta, \tag{15}$$

when $n^U = O\left(\frac{\log 2/\beta}{(1-\xi_0)^2\Delta_k^2}\right), n^L = \widetilde{O}\left(\frac{\sqrt{k}}{\alpha\epsilon}\right)$.

[9]Note that $\widetilde{O}$ neglects the logarithmic terms associated with $\frac{1}{\delta}$ and $\frac{1}{\beta}$.

We apply Theorem 7 in [59] with $\psi(\theta) = \theta$ and $\alpha = 1$ for cross entropy loss to obtain an upper bound on excess 0-1 loss,

$$\begin{aligned}
&\mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(\hat{w};(x,y))\right] - \min_{w\in B_2^d} \mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(w;(x,y))\right] \\
\geq &\frac{1}{2}\left(\mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\,(x,\hat{w})>0\}\right] - \min_{w\in B_2^d} \mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\,(x,\hat{w})>0\}\right]\right)
\end{aligned} \tag{16}$$

Substitute Equation (16) into Equation (15), we obtain the convergence guarantee on 0-1 loss.

$$\begin{aligned}
&\mathbb{P}_{(S^U,S^L)\sim D,\hat{w}\sim\mathcal{A}_{\epsilon,\delta}}\left[\mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(\hat{w};(x,y))\right] - \min_{w\in B_2^d} \mathbb{E}_{(x,y)\sim D}\left[\ell_{CN}(w;(x,y))\right] \leq \frac{\alpha}{2}\right] \\
\leq &\mathbb{P}_{(S^U,S^L)\sim D,\hat{w}\sim\mathcal{A}_{\epsilon,\delta}}\left[\mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\,(x,\hat{w})>0\}\right] - \min_{w\in B_2^d} \mathbb{E}_{(x,y)\sim D}\left[\mathbb{1}\{y\,(x,\hat{w})>0\}\right] \leq \alpha\right] \geq 1-\beta.
\end{aligned} \tag{17}$$

This completes the proof. □

C. Privacy guarantees for PILLAR on the original image dataset

As described in Figure 2, in practice PILLAR is applied on the set of representations obtained by passing the private dataset of images through a pre-trained feature extractor. Therefore, a straightforward application of Theorem 1 yields an (ϵ, δ)-DP guarantee on the set of representations and not on the dataset in the raw pixel space themselves. Here, we show that PILLAR provides (at least) the same DP guarantees on the dataset in the pixel space as long as the pre-training dataset cannot be manipulated by the privacy adversary. One way to achieve this, as we show is possible in this paper, is by using the same pre-trained model across different tasks. Investigating the extent of privacy harm that can be caused by allowing the adversary to manipulate the pre-training data remains an important future direction.

Corollary 1. *Let $f : \mathbb{R}^p \to \mathbb{R}^d$ be a feature extractor pre-trained using any algorithm. Let S_1, S_2 be any two neighbouring datasets of private images in $\mathbb{R}^p$. Then, for any $Q \subseteq \mathcal{H}^d$ where $\mathcal{H}^d$ is the class of linear halfspaces in d dimensions,*

$$\mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\circ f(S_1)}\left[h \in Q\right] \leq e^{\epsilon}\mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\circ f(S_2)}\left[h \in Q\right] + \delta$$

where $\mathcal{A}_{\epsilon,\delta}$ is Algorithm 1 (PILLAR) run with privacy parameters ϵ, δ.

Proof. Note that f is a deterministic many-to-one function from the dataset of images to the dataset of representations [10]. For any two neighbouring datasets S_1, S_2 in the image space, let $S_1^{\mathrm{R}}, S_2^{\mathrm{R}}$ be the corresponding set of representations extracted by f, i.e. $S_1^{\mathrm{R}} = \{f(x) : x \in S_1\}$ and $S_2^{\mathrm{R}} = \{f(x) : x \in S_2\}$. Then for any $Q \subseteq \mathcal{H}^d$

$$\begin{aligned}
\mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\circ f(S_1)}\left[h \in Q\right] &= \mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\left(S_1^{\mathrm{R}}\right)}\left[h \in Q\right] \\
&\leq e^{\epsilon}\mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\left(S_2^{\mathrm{R}}\right)}[h \in Q] + \delta \\
&= e^{\epsilon}\mathbb{P}_{h\sim\mathcal{A}_{\epsilon,\delta}\circ f(S_2)}\left[h \in Q\right] + \delta
\end{aligned}$$

where the first and the last equality follows by using the definition $S_1^{\mathrm{R}}, S_2^{\mathrm{R}}$ and due to the fact that f is a many-to-one function. The second inequality follows from observing that $S_1^{\mathrm{R}}, S_2^{\mathrm{R}}$ can differ on at most one point as f is a deterministic many-to-one function and $\mathcal{A}_{\epsilon,\delta}$ is (ϵ, δ)-DP. □

D. Theoretical results under distribution shifts and proofs

In this section, we provide the theoretical guarantees of PILLAR under distribution shifts. Before that, we formally define η-TV tolerant semi-private learning.

Definition 4 (η-TV tolerant $(\alpha, \beta, \epsilon, \delta)$-semi-private learner on a family of distributions $\mathcal{D}$)**.** *An algorithm $\mathcal{A}$ is an η-TV tolerant $(\alpha, \beta, \epsilon, \delta)$-semi-private learner for a hypothesis class $\mathcal{H}$ on a family of distributions $\mathcal{D}$ if for any distribution $D^L \in \mathcal{D}$, given a labelled dataset S^L of size n^L sampled i.i.d. from D^L and an unlabelled dataset S^U of size n^U sampled i.i.d. from any distribution D^U with η-bounded TV distance from D_X^L as well as third moment bounded by η, $\mathcal{A}$ is (ϵ, δ)-DP with respect to S^L and outputs a hypothesis h satisfying*

$$\mathbb{P}[\mathbb{P}_{(x,y)\sim D}\left[h(x) \neq y\right] \leq \alpha] \geq 1-\beta,$$

where the outer probability is over the randomness of the samples and the intrinsic randomness of the algorithm. In addition, the sample complexity n^L and n^U must be polynomial in $\frac{1}{\alpha}$ and $\frac{1}{\beta}$, and n^L must also be polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$.

[10] f can be designed to normalize the extracted features in a d-dimensional unit ball.

In Theorem 4, we prove a full version of Theorem 2 that demonstrates PILLAR is an η-TV tolerant $(\alpha, \beta, \epsilon, \delta)$-semi-private learner for linear halfspaces $\mathcal{H}^d$. We define the scaled hinge loss that depends on η, ξ_0, γ_0 as

$$\ell(w;(x,y)) = \max\left\{0, 1 - \frac{y\langle w, x\rangle}{\gamma_0\left(0.9(1-\xi_0) - \frac{14\eta}{\Delta_k}\right)}\right\}. \tag{18}$$

Theorem 4. *For $k \leq d \in \mathbb{N}$, $\gamma_0 \in (0,1)$, $\xi_0 \in [0,1)$, let $\mathcal{D}_{\gamma_0,\xi_0}$ be the family of distributions consisting of all (γ, ξ_k)-large margin low rank distributions over $\mathcal{X}_d \times \mathcal{Y}$ with $\gamma \geq \gamma_0$ and $\xi_k \leq \xi_0$ and third moment bounded by η. For any $\alpha \in (0,1), \beta \in (0, 1/4)$, $\epsilon \in (0, 1/\sqrt{k})$, $\delta \in (0,1)$ and $\eta \in [0, 9(1-\xi_0)\Delta_k/140)$, PILLAR with scaled hinge loss ℓ defined in Equation* (18)*, is an η-TV tolerant $(\alpha, \beta, \epsilon, \delta)$-semi-private learner of the linear halfspace $\mathcal{H}^d$ on $\mathcal{D}_{\gamma_0,\xi_0}$ with sample complexity*

$$n^U = O\left(\frac{\log\frac{2}{\beta}}{(\gamma_0\Delta_k)^2}\right), n^L = \tilde{O}\left(\frac{\sqrt{k}}{\alpha\epsilon\zeta}\right)$$

where $\Delta_k = \lambda_k\left(\Sigma^L\right) - \lambda_{k+1}\left(\Sigma^L\right)$ and $\zeta = \gamma_0\left(0.9(1-\xi_0) - 14\eta/\Delta_k\right)$.

Proof. **Privacy Guarantee** A similar argument as the proof of the privacy guarantee in Theorem 1 shows that Algorithm $\mathcal{A}_{\epsilon,\delta}(k,\zeta)$ preserves (ϵ,δ)-DP on the labelled dataset S^L. We now focus on the accuracy guarantee.

Accuracy Guarantee For any unlabelled distribution D^U with η-bouneded TV distance from the labelled distribution D^L_{γ,ξ_k}, let the empirical covariance matrix of the unlabelled dataset S^U be $\widehat{\Sigma}^U = \frac{1}{n^U}\sum_{x\in S^U} xx^\top$ and $\hat{A}_k \in \mathbb{R}^{d\times k}$ be the projection matrix whose i^{th} column is the i^{th} eigenvector of $\widehat{\Sigma}^U$. Let Σ^L and Σ^U be the population covariance matrices of the labelled and unlabelled distributions D^L and D^U. Similarly, let A^L_k and A^U_k be the matrices of top k eigenvectors of Σ^L and Σ^U respectively.

By definition, all distributions $D^L_{\gamma,\xi_k} \in \mathcal{D}_{\gamma_0,\xi_0}$ are (γ,ξ_k)-large margin low rank distribution, as defined in Definition 3, for some $\gamma \geq \gamma_0$, $\xi_k \leq \xi_0$. Let $w^\star$ be the large margin linear classifier that is guaranteed to exist by Definition 3. Then, for all $z \in \operatorname{supp}\left(D^L_{X,(\gamma,\xi_0)}\right)$, where $D^L_{X,(\gamma,\xi_0)}$ is the marginal distribution of D_{γ,ξ_k}, its margin is lower bounded by $\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2}$. Similar to the proof of Lemma 2, we will first lower bound this term to show that, with high probability, the projected dataset still retains a large margin. Then, we will invoke existing algorithms in the literature with scaled hinge loss with the right parameters, to privately learn a large margin classifier in this low dimensional space.

First, let $z = a_z w^\star + b^\perp$ for some a_z where $b^\perp$ is in the nullspace of $w^\star$. Then, it is easy to see that using the large-margin property in Definition 3, we get

$$ya_z = \frac{\langle w^\star, z\rangle}{\|w^\star\|_2\|z\|_2} \geq \gamma \geq \gamma_0. \tag{19}$$

Then, we lower bound $\dfrac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2}$ as

$$\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2} \overset{(a)}{=} \frac{ya_z\left\|\hat{A}_k^\top w^\star\right\|_2}{\left\|\hat{A}_k^\top z\right\|_2} \overset{(b)}{\geq} \gamma_0\left\|\hat{A}_k^\top w^\star\right\|_2, \tag{20}$$

where step (a) is due to $\left\langle w^\star, b^\perp\right\rangle = 0$ and step (b) follows from $\left\|\hat{A}_k^\top z\right\|_2 \leq \left\|\hat{A}_k\right\|_{\text{op}}\|z\|_2 \leq 1$ and Equation (19). To lower bound $\left\|\hat{A}_k^\top w^\star\right\|_2$, we use the triangle inequality to decompose it as follows

$$\begin{aligned}
\left\|\hat{A}_k^\top w^\star\right\|_2 &\geq \left\|A^L_k\left(A^L_k\right)^\top w^\star\right\|_2 - \left\|\left(A^L_k\left(A^L_k\right)^\top - A^U_k\left(A^U_k\right)^\top\right)w^\star\right\|_2 - \left\|\left(A^U_k(A^U_k)^\top - \hat{A}_k(\hat{A}_k)^\top\right)w^\star\right\|_2 \\
&\geq \left\|A^L_k\left(A^L_k\right)^\top w^\star\right\|_2 - \left\|A^L_k(A^L_k)^\top - A^U_k\left(A^U_k\right)^\top\right\|_{\text{op}}\|w^\star\|_2 - \left\|A^U_k(A^U_k)^\top - \hat{A}_k(\hat{A}_k)^\top\right\|_F\|w^\star\|_2 \\
&\geq 1 - \xi_k - \left\|A^L_k(A^L_k)^\top - A^U_k\left(A^U_k\right)^\top\right\|_{\text{op}} - \left\|A^U_k(A^U_k)^\top - \hat{A}_k(\hat{A}_k)^\top\right\|_F
\end{aligned} \tag{21}$$

where the second inequality follows from applying Cauchy-Schwartz inequality on the second and third term and the third step follows from using the low rank separability assumption in Definition 3 on the first term and observing that $\|w^\star\|_2 = 1$.

Now, we need to bound the two terms $\left\|A^L_k(A^L_k)^\top - A^U_k\left(A^U_k\right)^\top\right\|_{\text{op}}$ and $\left\|A^U_k(A^U_k)^\top - \hat{A}_k(\hat{A}_k)^\top\right\|_F$. We bound the first term with Lemma 5.

Lemma 5 (Theorem 3 in [57]). *Let $A \in \mathbb{R}^d$ be a symmetric positive definite matrix with nonzero eigenvalues $\lambda_1 > \lambda_2 > \ldots > \lambda_d$. Let $k > 0$ be an integer such that $\lambda_k > 0$. Let $B \in \mathbb{R}^d$ be another symmetric positive definite matrix such that $\|B\|_F < \frac{1}{4}(\lambda_k - \lambda_{k+1})$ and $A + B$ is still a positive definite matrix. Let $P_k(A), P_k(A+B)$ be the matrices whose columns consists of the first k eigenvectors of $A, A+B$, then*

$$\left\|P_k(A)P_k(A)^T - P_k(A+B)P_k(A+B)^T\right\|_F \leq \frac{2\|B\|_F}{\lambda_k - \lambda_{k+1}}.$$

It guarantees that with probability $1 - \beta/4$,

$$\left\|A_k^L \left(A_k^L\right)^\top - A_k^U \left(A_k^U\right)^\top\right\|_{\text{op}} \leq \frac{2\left\|\Sigma^L - \Sigma^U\right\|_{\text{op}}}{\lambda_k\left(\Sigma^L\right) - \lambda_{k+1}\left(\Sigma^L\right)} = \frac{2\left\|\Sigma^L - \Sigma^U\right\|_{\text{op}}}{\Delta_k}. \tag{22}$$

Then, we bound the term $\left\|\Sigma^L - \Sigma^U\right\|_{\text{op}}$ with Lemma 6.

Lemma 6. *Let f and g be the Probability Density Functions (PDFs) of two zero-mean distributions F and G over $\mathcal{X}$ with covariance matrices Σ_f and Σ_g respectively. Assume the spectral norm of the third moments of both F and G are bounded by η. If the total variation between the two distributions is bounded by η,i.e. $TV(f,g) = \max_{A\subset\mathcal{X}} |f(A) - g(A)| \leq \eta$, then the discrepancy in the covariance matrices is bounded by 7η, i.e.$\|\Sigma_f - \Sigma_g\|_{op} \leq 7\eta$.*

By applying Lemma 6 and the assumption of bounded total variation between the labelled and unlabelled distributions to Equation (22), we get

$$\left\|A_k^L \left(A_k^L\right)^\top - A_k^U \left(A_k^U\right)^\top\right\|_{\text{op}} \leq \frac{14\eta}{\lambda_k\left(\Sigma^L\right) - \lambda_{k+1}\left(\Sigma^L\right)} = \frac{14\eta}{\Delta_k}, \tag{23}$$

where Δ_k is defined as the difference between the k^{th} and $(k+1)^{th}$ eigenvalue of Σ^L.

Similar to the proof for Lemma 2, we upper bound the term $\left\|A_k^U(A_k^U)^\top - \hat{A}_k(\hat{A}_k)^\top\right\|_F$ using Lemma 4, which guarantees that with probability $1 - \beta/4$,

$$\left\|A_k^U \left(A_k^U\right)^\top - \hat{A}_k\hat{A}_k^\top\right\|_F \leq \frac{1-\xi_0}{10}, \tag{24}$$

where the inequality follows from choosing the size of unlabelled data $n^U = O\left(\frac{\log\frac{2}{\beta}}{((1-\xi_0)\Delta_k)^2}\right)$.

Substituting Equations (23) and (24) into Equation (21) and then plugging Equation (21) into Equation (20), we get that with probability at least $1 - \beta/2$, the margin in the projected space is lower bounded as

$$\frac{y\left\langle \hat{A}_k^\top z, \hat{A}_k^\top w^\star\right\rangle}{\left\|\hat{A}_k^\top z\right\|_2\left\|\hat{A}_k^\top w^\star\right\|_2} \geq \gamma_0\left(0.9(1-\xi_0) - \frac{14\eta}{\Delta_k}\right).$$

Thus, the (scaled) hinge loss function ℓ defined in Equation (18) is $\frac{1}{\gamma_0(0.9(1-\xi_0)-14\eta/\Delta_k)}$-Lipschitz. For a halfspace with parameter $v \in B_2^k$, denote the empirical hinge loss on a dataset S by $\hat{L}(w;S) = \frac{1}{|S|}\sum_{(x,y)\in S}\ell(w,(x,y))$ and the loss on the distribution D by $L(w;D) = \mathbb{E}_{(x,y)\sim D}[\ell(w,(x,y))]$. Let D_k be the k-dimension transformation of the original distribution D by projecting each $x \in \mathcal{X}$ to $\hat{A}_k^\top x$. By the convergence bound in Lemma 1 for $\mathcal{A}_{\text{Noisy-SGD}}$, we have with probability $1 - \frac{\beta}{4}$, $\mathcal{A}_{\text{Noisy-SGD}}$ outputs a hypothesis $v_k \in B_2^k$ such that

$$\hat{L}(v_k; S_k^L) - \hat{L}(v_k^\star; D_k) = \hat{L}(v_k; S_k^L) = \tilde{O}\left(\frac{\sqrt{k}}{n^L\epsilon\gamma_0\left(0.9(1-\xi_0) - 14\eta/\Delta_k\right)}\right),$$

where $v_k^\star = \text{argmin}_{v\in B_2^k}\hat{L}(v;S_k^L)$ and $\hat{L}(v_k^\star;S_k^L) = 0$ as the margin in the transformed low-dimensional space is at least $\gamma_0\left(0.9(1-\xi_0) - \frac{14\eta}{\Delta_k}\right) > 0$ for $\eta \leq \frac{9(1-\xi_0)\Delta_k}{140}$. For $n^L = O\left(\frac{\sqrt{k}}{\alpha\beta\gamma_0(0.9(1-\xi_0)-14\eta/\Delta_k)}\text{polylog}\left(\frac{1}{\delta},\frac{1}{\epsilon},\frac{1}{\beta},\frac{1}{\alpha},\frac{1}{\gamma_0},\frac{1}{\xi_0},k,n^L\right)\right)$, we can bound the emiprical 0-1 error with probability $1 - \frac{\beta}{4}$,

$$\frac{1}{n^L}\sum_{(x,y)\in S_k^L}\mathbb{I}\{y\langle v_k, x\rangle < 0\} \leq \hat{L}(v_k;S_k^L) = \tilde{O}\left(\frac{\sqrt{k}}{n^L\epsilon\gamma_0\left(0.9(1-\xi_0) - 14\eta/\Delta_k\right)}\right) \leq \frac{\alpha}{4}. \tag{25}$$

It remains to bound the generalisation error of linear halfspace $\mathcal{H}^k$. We use Lemma 7 for upper bounding this term.

Lemma 7 (Convergence bound on generalisation error [60])**.** *Suppose $\mathcal{H}$ is a hypothesis class with instance space $\mathcal{X}$ and output space $\{-1,1\}$. Let D be a distribution over $\mathcal{X}\times\mathcal{Y}$ and S be a dataset of size n sampled i.i.d. from D. For $\eta\in(0,1),\zeta>0$, we have*

$$\mathbb{P}_{S\sim D^n}\left[\sup_{h\in\mathcal{H}} L(h;D)-(1+\zeta)\hat{L}(h;S)>\eta\right]\leq 4\Pi_{\mathcal{H}}(2n)\exp\left(-\frac{\eta\zeta n}{4\left(\zeta+1\right)}\right),$$

where L and $\hat{L}$ are the population and the empirical 0-1 error and $\Pi_{\mathcal{H}}$ is the growth function of $\mathcal{H}$.

Setting $\zeta=1$ and $\eta=\frac{\alpha}{2}$ in Lemma 7 gives us that with probability $1-\frac{\beta}{4}$,

$$\mathbb{P}_{(x,y)\sim D_k}\left[y\left\langle v_k,x\right\rangle<0\right]-\frac{2}{n^L}\sum_{(x,y)\in S_k^L}\mathbb{I}\{y\left\langle v_k,x\right\rangle<0\}\leq\frac{\alpha}{2}. \tag{26}$$

Thus, combining Equations (25) and (26) we get

$$\mathbb{P}_{(x,y)\sim D}\left[y\left\langle v_k,\hat{A}_k^\top x\right\rangle<0\right]=\mathbb{P}_{(x,y)\sim D_k}\left[y\left\langle v_k,x\right\rangle<0\right]\leq\frac{2}{n^L}\sum_{(x,y)\in S_k^L}\mathbb{I}\{y\left\langle v_k,x\right\rangle<0\}+\frac{\alpha}{2}=\alpha,$$

for $n^L\geq\frac{k}{\alpha}\text{polylog}\left(\frac{1}{\beta},\frac{1}{k}\right)$. This is equivalent as stating that the output of Algorithm 1 $\hat{w}=\hat{A}_k v_k$ satisfies

$$\mathbb{P}_{(x,y)\sim D}\left[y\left\langle\hat{w},x\right\rangle<0\right]=\mathbb{P}_{(x,y)\sim D}\left[y\left\langle\hat{A}_k v_k,x\right\rangle<0\right]=\mathbb{P}_{(x,y)\sim D}\left[y\left\langle v_k,\hat{A}_k^\top x\right\rangle<0\right]\leq\alpha,$$

which concludes the proof. □

Proof of Lemma 6. We first approximate Moment Generating Functions (MGFs) of g and f by their first and second moments. Then, we express the error bound in this approximation by the error bound for Taylor expansion, for any $t\in\mathbb{R}^d$ with $\|t\|_2>0$,

$$\begin{aligned}\left|M_f(t)-\ 1+t^T\mathbb{E}_f\left[X\right]+\frac{t^T\Sigma_f t}{2}\right|&\overset{(a)}{\leq}\frac{\mathbb{E}_f\left[e^{t^Tx}xx^Tx\right]\|t\|_2^3}{3!}\\&\overset{(b)}{\leq}\frac{\mathbb{E}_f\left[xx^Tx\right]e^{\|t\|_2}\|t\|_2^3}{3!}\\&\overset{(c)}{\leq}\eta\|t\|_2^3\end{aligned} \tag{27}$$

where step (a) follows by the error bound of Taylor expansion, step (b) is due to $e^{t^Tx}\leq e^{\|t\|_2\|x\|_2}\leq e^{\|t\|_2}$ for all $x\in B_d^2$, and step (c) follows from $e^{\|t\|_2}\leq 3!$ for $\|t\|_2\leq 1$. Similarly,

$$\left|M_g(t)-\ 1+t^T\mathbb{E}_g\left[X\right]+\frac{t^T\Sigma_g t}{2}\right|\leq\eta\|t\|_2^3. \tag{28}$$

Rewrite Equation (27) and Equation (28) and observing that $\mathbb{E}_g[X]=\mathbb{E}_f[X]=0$, we can bound the terms $\frac{t^T\Sigma_f t}{2}$ and $\frac{t^T\Sigma_g t}{2}$ by

$$\begin{aligned}1-M_f(t)-\eta\|t\|_2^3\leq\frac{t^T\Sigma_f t}{2}\leq 1-M_f(t)+\eta\|t\|_2^3\\1-M_g(t)-\eta\|t\|_2^3\leq\frac{t^T\Sigma_g t}{2}\leq 1-M_g(t)+\eta\|t\|_2^3.\end{aligned} \tag{29}$$

Next, we show that the discrepancy in covariance matrices of distributions G and F are upper bounded by the difference in their MGFs. By Equation (29), for all $t\in\mathbb{R}^d$ and $\|t\|_2\neq 0$,

$$\begin{aligned}\left|\frac{t^T\left(\Sigma_f-\Sigma_g\right)t}{2}\right|&\leq 1-M_f(t)+\eta\|t\|_2^3-1+M_g(t)+\eta\|t\|_2^3\\&=\left|M_g(t)-M_f(t)+2\eta\|t\|_2^3\right|\\&\leq|M_g(t)-M_f(t)|+2\eta\|t\|_2^3\end{aligned} \tag{30}$$

Next, we upper bound the difference between the MGFs of distributions G and F by the TV distance between them.

$$
\begin{aligned}
|M_f(t) - M_g(t)| &= \left| \int_{x \in B_2^d} e^{t^T x} \left[f(x) - g(x) \right] dx \right| \\
&\leq \int_{x \in B_2^d} e^{t^T x} \left| f(x) - g(x) \right| dx \\
&\leq \int_{x \in B_2^d} e^{\|t\|_2 \|x\|_2} \left| f(x) - g(x) \right| dx \leq \frac{e^{\|t\|_2} \eta}{2}
\end{aligned} \tag{31}
$$

where the last inequality follows as $\|x\|_2 = 1$ for $x \in B_2^d$ and $TV(f, g) \leq \eta$.

Combine Equation (30) and Equation (31), we have for all $t \in \mathbb{R}^d$ and $\|t\|_2 \neq 0$,

$$
\left| t^T \left(\Sigma_f - \Sigma_g \right) t \right| \leq e^{\|t\|_2} \eta_1 + 4\eta \|t\|_2^3 \tag{32}
$$

Choose t as a vector in the direction of the first eigenvector (i.e. the eigenvector corresponding to the largest eigenvalue) of $\Sigma_f - \Sigma_g$. For t in this direction, by the definition of operator norm,

$$
\|\Sigma_f - \Sigma_g\|_{\text{op}} = \frac{\left| t^T \left(\Sigma_f - \Sigma_g \right) t \right|}{\|t\|_2}. \tag{33}
$$

Plugging Equation (33) into Equation (32) and choose the norm of t as the minimizer of $e^{\|t\|_2} \eta_1 + 4\eta \|t\|_2^3$, we get

$$
\|\Sigma_f - \Sigma_g\|_{\text{op}} \leq \min_{0 \leq \|t\|_2 \leq 1} \frac{e^{\|t\|_2} \eta}{\|t\|_2^2} + 4\eta \|t\|_2 \leq \frac{\eta(1 + \|t\|_2 + \|t\|_2^2)}{\|t\|_2^2} + 4\eta \|t\|_2 = 7\eta
$$

This conclude the proof. □

E. Large margin Gaussian mixture distributions

In this section, we present in Example 1 a class of Large margin Gaussian mixture distributions that satisfies the large-margin low rank assumption. For any $\theta, \sigma^2 = O(1/\sqrt{d})$, it is easy to see that this family of distributions satisfies the large margin low rank properties in Definition 3 for $k = 2$ and $\xi_k = 0$.

Example 1. *A distribution D over $\mathcal{X} \times \mathcal{Y}$ is a (θ, σ^2)-Large margin Gaussian mixture distribution if there exists $w^\star, \mu \in B_2^d$, such that $\langle \mu, w^\star \rangle = 0$, the conditional random variable $X|y$ is distributed according to a normal distribution with mean μy and covariance matrix $\theta w^\star (w^\star)^\top + \sigma^2 I_d$ and $y \in \{-1, 1\}$ is distributed uniformly.*

We present Corollary 2 following Theorem 1, which shows that for large margin Gaussian mixture distributions, PILLAR leads to a drop in the private sample complexity from $O(\sqrt{d})$ to $O(1)$.

Corollary 2 (Theoretical guarantees for large margin Gaussian mixture distribution)**.** *For $\theta, \sigma^2 = \tilde{O}\left(1/\sqrt{d}\right)$, let $\mathcal{D}_{\theta,\sigma^2}$ be the family of all (θ, σ^2)-large margin Gaussian mixture distribution (Example 1). For any $\alpha \in (0, 1), \beta \in (0, 1/4), \epsilon \in (0, 1/\sqrt{M})$, and $\delta \in (0, 1)$, PILLAR $\mathcal{A}_{\epsilon,\delta}(k, \ell)$ with scaled hinge loss defined in Table I is an $(\alpha, \beta, \epsilon, \delta)$-semi-private learner on D_{θ,σ^2} of linear halfspaces $\mathcal{H}^d$ with sample complexity*

$$
\begin{aligned}
n^U &= O\left(\frac{M^2 \log \frac{2}{\beta}}{\gamma^2 \theta^2} \right), \\
n^L &= \tilde{O}\left(\frac{M\sqrt{k}}{\alpha\epsilon\gamma(1 - 0.1\gamma)} \right)
\end{aligned} \tag{34}
$$

where $\gamma = 1 - \left(4\sqrt{d} + 2\sqrt{\log \frac{2n^L}{\delta}} \right) \left(\sigma^2 + \theta \right)$, $M = 1 + \left(4\sqrt{d} + 2\sqrt{\log \frac{2n^L}{\delta}} \right) \left(\sigma^2 + \theta \right)$.

Here, in line with the notation of Definition 3, γ intuitively represents the margin in the d-dimensional space and M is the upper bound for the radius of the labelled dataset. For $\theta = \sigma^2 = 1/2C\sqrt{d}$ and ignoring the logarithmic terms, we get $M = 1.5$ and $\gamma = 0.5$. Corollary 2 implies the labelled sample complexity $\tilde{O}\left(1/\alpha\epsilon\right)$.

Proof. To prove this result, we first show that all large-margin Gaussian mixture distributions $D_{\theta,\sigma^2} \in \mathcal{D}_{\theta,\sigma^2}$ are (γ_0, ξ)-large margin low rank distribution (Definition 3) after normalization. In particular, we show that the normalized distribution is (γ_0, ξ)-large margin low rank distribution with $\xi = 0$ and margin $\gamma_0 = \gamma/M$, where $\gamma = 1 - \left(4\sqrt{d} + 2\sqrt{\log \frac{2n^L}{\delta}} \right) \left(\sigma^2 + \theta \right)$ and $M = 1 + \left(4\sqrt{d} + 2\sqrt{\log \frac{2n^L}{\delta}} \right) \left(\sigma^2 + \theta \right)$. Then, invoking Theorem 1 gives the desired sample complexity in Equation (34).

To normalize the distribution, we consider the marginal distribution D_X of the mixture distribution $D \in \mathcal{D}_{\theta,\sigma^2}$ and compute its mean and the covariance matrix. By Example 1, D is a mixture of two gaussians with identical covariance matrix $\Sigma = \theta w^\star (w^\star)^\top - \sigma^2 I_d$ and means $\mu_1 = -\mu_2$.With a slight misuse of notation, we denote the probability density function of a normal distribution with mean μ and covariance Σ using $\mathcal{N}(x; \mu, \Sigma)$. Then, we can calculate the mean and covariance matrix as

$$\mathbb{E}_X [X] = \mathbb{E}_y \mathbb{E}_{X|y} [X|y] = \frac{1}{2}\mu_1 + \frac{1}{2}\mu_2 = 0 \tag{35}$$

and

$$\begin{aligned}
\Sigma_X &= \mathbb{E}_X \left[XX^\top\right] - \left(\mathbb{E}_X [X]\right)\left(\mathbb{E}_X [X]\right)^\top \overset{(a)}{=} \mathbb{E}_y \mathbb{E}_{X|y} \left[XX^\top | y\right] \\
&= \frac{1}{2}\int_{B_2^d} xx^\top \mathcal{N}(x; \mu_1, \Sigma) dx + \frac{1}{2}\int_{B_2^d} xx^\top \mathcal{N}(x; \mu_2, \Sigma) dx \\
&\overset{(b)}{=} \frac{1}{2}\left(\Sigma + \mu_1\mu_1^\top\right) + \frac{1}{2}\left(\Sigma + \mu_2\mu_2^\top\right) \\
&\overset{(c)}{=} \theta w^\star \left(w^\star\right)^\top + \mu_1\mu_1^\top + \sigma^2 I_d
\end{aligned}$$

where step (a) follows by Equation (35), step (b) follows by the relationship between covaraince matrix and the second moment $\Sigma = \mathbb{E}_X \left[XX^\top\right] - \mu\mu^\top$, and step (c) follows by the definition of large-margin Gaussian mixture distribution (Example 1) of Σ and μ_1, μ_2.

Then, we show that the first two eigenvectors are μ_1 and $w^\star$ with the corresponding eigenvalues $1 + \sigma^2$ and $\theta + \sigma^2$ for $\theta = O\left(1/\sqrt{d}\right) \leq 1$. The remaining non-spiked eigenvalues are σ^2.

$$\begin{aligned}
\Sigma_X \mu_1 &= \theta w^\star \left(w^\star\right)^\top \mu_1 + \mu_1\mu_1^\top \mu_1 + \sigma^2\mu_1 \\
&\overset{(a)}{=} (\|\mu_1\|_2^2 + \sigma^2)\mu_1 = (1 + \sigma^2)\mu_1 \\
\Sigma_X w^\star &= \theta w^\star \left(w^\star\right)^\top w^\star + \mu_1\mu_1^\top w^\star + \sigma^2 w^\star \\
&\overset{(b)}{=} (\theta + \sigma^2) w^\star,
\end{aligned}$$

where step (a) and (b) both follow from the fact that $\left(w^\star\right)^\top \mu_1 = 0$. For $k = 2$, it follows immediately that $\Delta_k = \theta$ (Equation (36)) and $\xi = 0$ (Equation (37)),

$$\Delta_k = \lambda_k \left(\Sigma_X\right) - \lambda_{k+1} \left(\Sigma_X\right) = \theta + \sigma^2 - \sigma^2 = \theta. \tag{36}$$

$$\begin{aligned}
\frac{\left\|A_k^\top w^\star\right\|_2}{\|w^\star\|_2} &= \frac{1}{\|w^\star\|_2} \begin{bmatrix} \mu_1^\top \\ \left(w^\star\right)^\top \end{bmatrix} w^\star \\
&= \frac{\left|\mu_1^\top w^\star + \left(w^\star\right)^\top w^\star\right|}{\|w^\star\|_2} \\
&\overset{(a)}{=} 1 = 1 - \xi,
\end{aligned} \tag{37}$$

where step (a) follows from $\mu_1^\top w = 0$.

Next, we show that the labelled dataset lies in a ball with bounded radius with high probability, which further implies that original data has a large margin.

Denote the part of the dataset from the gaussian component with $y = 1$ by S_1^L and denote the part from the component with $y = -1$ by S_2^L. We apply the well-known concentration bound on the norm of Gaussian random vectors (Lemma 8) to show a high probability upper bound on the radius of the datasets S_1^L and S_2^L.

Lemma 8 ([61]). *Let $X \sim N(\mu, \Sigma)$, where $v \in B_d^2$. Then, with probability at least $1 - \delta$,*

$$\|X - \mu\|_2 \leq 4\|\Sigma\|_{op}\sqrt{d} + 2\|\Sigma\|_{op}\sqrt{\log\frac{1}{\delta}}.$$

This gives the following high probability upper bound on any $x \in S_i^L$ for $i = 1, 2$ and some $\frac{\beta}{2n^L} > 0$,

$$\mathbb{P}_{S^L \sim D^{n^L}}\left[\|x - \mu_i\|_2 \leq 4\left(\theta + \sigma^2\right)\sqrt{d} + 2\left(\theta + \sigma^2\right)\sqrt{\log\frac{4n^L}{\beta}}\right] \geq 1 - \frac{\beta}{4n^L}$$

	Public Unlabelled Data	Low-rank Assumption	Sample complexity
Generic semi-private learner ([16])	✔	-	$\tilde{O}\left(\frac{\sqrt{d}}{\alpha\epsilon\gamma}\right)$
No Projection			
DP-SGD [28]	✗	Restricted Lipschitz Continuity $\left(\sum_{i=1}^{\log(d/k)+1} G_{2^{s-1}k}^2 \leq c_2\right)$	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha\epsilon\gamma} + \sqrt{\frac{c_2 d}{k}}\right)$
Random Projection (e.g. JL Transform)			
[27]	✗	-	$\tilde{O}\left(\frac{1}{\alpha\epsilon\gamma^2}\right)$
[24]	✗	-	$\tilde{O}\left(\min\left\{\frac{\omega(\mathcal{C})}{\beta}, \sqrt{d}\right\}\frac{1}{\alpha\epsilon\gamma}\right)$
Low Rank Projection Projection			
GEP [17]	✔	Low-rank gradients ($\bar{r} \leq c_1$)	$\tilde{O}\left(\frac{1}{\alpha\epsilon\gamma} + (\sqrt{k} + c_1\sqrt{d})\right)$
OURS	✔	Low Rank Separability (Definition 3)	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha\epsilon\gamma(1-\xi)}\right)$

TABLE V: Comparison with existing works: $\omega(\mathcal{C})$ represents the Gaussian width of the parameter space $\mathcal{C}$, and c_1, c_2 are constants that decrease with the low-rankness of the gradient space of the loss function. G_i represents the projection of the norm of the projection of the gradient onto the null space of a low rank matrix and is formally defined in Equation (38). All remaining notations: $d, k, \xi, \gamma, \alpha, \beta$, and ϵ have the same meaning as the main text. The sample-complexity of DP-SGD is based on hinge loss.

For $i \in \{1, 2\}$, by applying union bound on all $x \in S_i^L$, we can bound maximum distance of a points $x \in S_i^L$ to the center μ_i,

$$\mathbb{P}_{S^L \sim D^{n^L}}\left[\max_{x \in S_i^L} \|x - \mu_i\|_2 \leq \left(\theta + \sigma^2\right)\left(4\sqrt{d} + 2\sqrt{\log\frac{4n^L}{\beta}}\right)\right] \leq 1 - \frac{\beta}{4}.$$

Note that the distance between the two centers μ_1 and μ_2 is 2. Thus, with probability at least $1 - \frac{\beta}{2}$, all points in the labelled dataset S^L lie in a ball centered at 0 having radius

$$M = 1 + \left(4\sqrt{d} + 2\sqrt{\log\frac{4n^L}{\beta}}\right)\left(\sigma^2 + \theta\right).$$

Also, the margin in the original labelled dataset is at least

$$\gamma = 1 - \left(4\sqrt{d} + 2\sqrt{\log\frac{4n^L}{\beta}}\right)\left(\sigma^2 + \theta\right).$$

Normalizing the data by M, it is obvious that the normalized distribution satisfies the definition of (γ, ξ)-large margin low rank distribution with parameters $\xi = 0$, $\Delta_k = \theta/M$ and $\gamma_0 = \gamma/M$, where $\gamma = 1 - \left(4\sqrt{d} + 2\sqrt{\log\frac{2n^L}{\delta}}\right)\left(\sigma^2 + \theta\right)$, $M = 1 + \left(4\sqrt{d} + 2\sqrt{\log\frac{2n^L}{\delta}}\right)\left(\sigma^2 + \theta\right)$. Invoking Theorem 1 concludes the proof. □

F. Discussion of assumptions for existing methods

Table V summarises the comparison of our theoretical results with some existing methods. We describe the notation used in the table below.

a) Analysis of the Restricted Lipschitz Continuity (RLC) assumption [28]: As indicated in Table V, DP-SGD [28] achieves dimension independent sample complexity if the following assumption, known as Restricted Lipschitz Continuity (RLC) is satisfied. For some $k \ll d$,

$$\sum_{i=1}^{\lfloor \log(d/k)+1 \rfloor} G_{2^{i-1}k}^2 \leq O(\sqrt{k/d}), \tag{RLC 1}$$

where $G_0, G_1, ..., G_d$ represent the RLC coefficients. For any $i \in [d]$, the loss function ℓ is said to satisfy RLC with coefficient G_i if

$$G_i \geq \min_{\substack{\text{rank}(P_i)=i \\ P_i \in \Pi}} \|(I - P_i)\nabla \ell(w; (x, y))\|_2 , \tag{38}$$

for all $w, x, y \in \text{domain}\,(\ell)$, where Π is the set of orthogonal projection matrices. Equivalently, assumption RLC 1 states that for some $k \ll d$,

$$\sum_{i=k+1}^{d} G_i^2 \leq O(\sqrt{k/d}). \tag{RLC}$$

In this section, we demonstrate that if we assume the Restricted Lipschitz Continuity (RLC) condition from [28], our low rank separability assumption on $\left\|A_k A_k^\top w^\star\right\|$ holds for large-margin linear halfspaces. However, using the RLC assumption leads to a looser bound compared to our assumption. More specifically, given the RLC assumption and the loss function ℓ defined in Table I, we can show $\left\|A_k A_k^\top w^\star\right\| \geq \gamma$.

Given the parameter ζ in Algorithm 1, for $x, y \in \text{supp}\,(D)$ and w satisfying $y \langle w, x \rangle \leq \zeta$, we can calculate the i^{th} restricted Lipschitz coefficient

$$\begin{aligned} G_i &\geq \min_{\substack{\text{rank}(P_i)=1 \\ P_i \in \Pi}} \|(I - P_i)\nabla \ell(w; (x, y))\|_2 \\ &= \min_{\substack{\text{rank}(P_i)=1 \\ P_i \in \Pi}} \left\| \frac{y}{\zeta}(I - P_i)x \right\|_2 \\ &= \min_{\substack{\text{rank}(P_i)=1 \\ P_i \in \Pi}} \left\| \frac{1}{\zeta}(x - P_i x) \right\|_2 . \end{aligned} \tag{39}$$

Equivalently, we can rewrite Equation (39) as there exists a rank-i orthogonal projection matrix $P_i^{\min}$ such that

$$\left\|x - P_i^{\min} x\right\|_2 \leq \zeta G_i. \tag{40}$$

Thus, for x such that $y \langle w, x \rangle \leq \zeta$,

$$\begin{aligned} \left\|xx^\top - (P_i^{\min}x)(P_i^{\min}x)^\top\right\|_{\text{op}} &\overset{(a)}{=} \left\|(x - P_i^{\min}x)(x + P_i^{\min}x)^\top\right\|_2 \\ &\leq \left\|x + P_i^{\min}x\right\|_2 \left\|x - P_i^{\min}x\right\|_2 \\ &\overset{(b)}{\leq} 2\left\|x - P_i^{\min}x\right\|_2 \\ &\overset{(c)}{\leq} 2G_i\zeta \end{aligned} \tag{41}$$

where step (a) follows from the orthogonality of $P_i^{\min}$, step (b) follows from $\left\|P_i^{\min}x\right\|_2 \leq \|x\|_2 = 1$, and step (c) follows from Equation (40).

Then, we can bound the low-rank approximation error for the covariance matrix of the data distribution.

$$\left\|\Sigma_X - P_i^{\min}\Sigma_X(P_i^{\min})^\top\right\|_{\text{op}} \overset{(a)}{\leq} \mathbb{E}_{x \sim D_X}\left(\left\|xx^\top - (P_i^{\min}x)(P_i^{\min}x)^\top\right\|_{\text{op}}\right) \overset{(b)}{\leq} 2G_i\zeta.$$

where $\Sigma_X = \mathbb{E}_{x \sim D_X}\left[xx^\top\right]$, and step (a) follows from the convexity of the Euclidean norm and step (b) follows from Equation (41).

This further provides an upper bound on the last $d - k$ eigenvalues of the covariance matrix Σ_X of the data distribution D_X. Let λ_i denote the i^{th} eigenvalue of the covariance matrix Σ_X. Then, we apply Lemma 9 that gives an upper bound on the singular values of a matrix in terms of the rank k approximation error of the matrix.

Lemma 9 ([62]). *For any matrix $M \in \mathbb{R}^{m \times n}$,*

$$\inf_{\text{rank}(\hat{M})=k} \left\|M - \hat{M}\right\|_{op} = \sigma_{k+1},$$

where the infimum is over all rank k matrices $\hat{M}$ and σ_{k+1} is the k^{th} singular value of the matrix M.

This gives an upper bound on the i^{th} eigenvalue of the covariance matrix Σ_X in terms of the i^{th} restricted Lipschitz coefficient,

$$\lambda_{i+1} = \sigma_{i+1}^2 = \inf_{\text{rank}\left(\Sigma'_X\right)=i} \|\Sigma_X - \Sigma'_X\|_{\text{op}}^2 \leq \left\|\Sigma_X - P_i^{\min}\Sigma_X(P_i^{\min})^\top\right\|_{\text{op}}^2 \leq 4G_i^2\zeta^2.$$

Thus, for matrix A_k consisting of the first k eigenvectors of Σ_X, we can upper bound the reconstruction error of $A_k^\top x$ with the eigenvalues of the covariance matrix Σ_X,

$$\begin{aligned}\mathbb{E}_{x\sim D_X}\left[\|x\|_2 - \left\|A_k^\top x\right\|_2\right] &= \mathbb{E}_{x\sim D_X}\left[\left\|xx^\top\right\|_{\text{op}} - \left\|(A_kx)(A_kx)^\top\right\|_{\text{op}}\right] \\ &\leq \mathbb{E}_{x\sim D_X}\left[\left\|xx^\top - (A_kx)(A_kx)^\top\right\|\right]_{\text{op}} \leq \sum_{i=k+1}^{d}\lambda_i \leq 4\zeta^2\sum_{i=k+1}^{d}G_i^2.\end{aligned}$$

By Markov's inequality, with probability at least $1-\beta$,

$$\begin{aligned}&\mathbb{P}_{x\sim D_X}\left[\left\|xx^\top\right\|_{\text{op}} - \left\|(A_k^\top x)(A_k^\top x)^\top\right\|_{\text{op}} \geq \frac{4\zeta^2}{\beta}\sum_{i=k+1}^{d}G_i^2\right] \\ &\leq \mathbb{P}_{x\sim D_X}\left[\left\|xx^\top - (A_k^\top x)(A_k^\top x)^\top\right\|_{\text{op}} \leq \frac{4\zeta^2}{\beta}\sum_{i=k+1}^{d}G_i^2\right] \leq \beta.\end{aligned} \tag{42}$$

This implies our assumption with probability at least $1-\beta$,

$$\begin{aligned}\left\|A_kA_k^\top w^\star\right\|_2 &\overset{(a)}{=} \|x\|_2\left\|A_kA_k^\top w^\star\right\|_2 \geq \left|\left\langle A_kA_k^\top x, w^\star\right\rangle\right| \\ &\overset{(b)}{\geq} \left|\langle x, w^\star\rangle\right| - \left|\left\langle x - A_kA_k^\top x, w^\star\right\rangle\right| \\ &\overset{(c)}{\geq} \gamma - \left\|x - A_kA_k^\top x\right\|_2\|w^\star\|_2 \\ &\overset{(d)}{\geq} \gamma - \frac{4\zeta^2}{\beta}\sum_{i=k+1}^{d}G_i^2\end{aligned} \tag{43}$$

where step (a) follows from $\|x\|_2 = 1$, step (b) follows by $\left\langle A_kA_k^\top x, w^\star\right\rangle = \langle x, w^\star\rangle - \left\langle x - A_kA_k^\top x, w^\star\right\rangle$ and the triangle inequality, step (c) follows by the large margin assumption $y\langle x, w^\star\rangle = |\langle x, w^\star\rangle| \geq \gamma$, and step (d) follows by Equation (42) with probability at least $1-\beta$.

The RLC assumption requires the last term in Equation (43) to vanish at the rate of $O(k/d)$. This implies our low-rank assumption holds with $\xi = 1-\gamma$.

b) Analysis on the error bound for GEP: To achieve a dimension-independent sample complexity bound in GEP [17], the gradient space must satisfy a low-rank assumption, which is even stronger than the rapid decay assumption in RLC coefficients (Equation (RLC 1)). By following a similar argument as the analysis for the RLC assumption [28], we can demonstrate that our low-rank assumption is implied by the assumption in GEP.

Privacy	CIFAR10		CIFAR100	
	Ours	**[13]**	**Ours**	**[13]**
$\epsilon = 0.1$	89.4	-	36.1	-
$\epsilon = 0.7$	93.1	-	69.7	-
$\epsilon = 1$	93.5	93.1	71.8	70.3
$\epsilon = 2$	93.9	93.6	74.9	73.9

TABLE VI: Result for our algorithm is with pre-training on ImageNet32x32. Results for [13] is taken from their paper where available.

APPENDIX B
EXPERIMENTAL DETAILS AND ADDITIONAL EXPERIMENTS

A. Details and hyperparameter ranges for our method

Unless stated otherwise, we use the PRV accountant [49] in our experiments. Following [13], we use the validation data for cross-validation of the hyperparameters in all of our experiments and set the clipping constant to 1. We search the learning rate in $\{0.01, 0.1, 1\}$, use no weight decay nor momentum as we have seen it to have little or adverse impact. We search the number of steps in $\{500, 1000, 3000, 5000, 6000\}$ and our batch size in $\{128, 512, 1024\}$. We compute the variance of the noise as a function of the number of steps and the target ϵ using `opacus`. We set $\delta = 1e-5$ in all our experiments. We use the open-source `opacus` [63] library to run DP-SGD with the PRV Accountant efficiently. We use `scikit-learn` to implement PCA. Checkpoints of ResNet-50 are taken or trained using the `timm` [64] and `solo-learn` [65] libraries. Standard ImageNet pre-processing of images is applied, without augmentations.

B. Discrepancy in pre-training resolution

Several works have used different resolutions of ImageNet to pre-train their models. In particular, [13] used ImageNet 32x32 to pre-train their model, which is a non-standard dimensionality of ImageNet, but it matches the dimensionality of their private dataset CIFAR-10. In contrast, we use the standard ImageNet (224x224) for pre-training in all our experiments with both CIFAR datasets as well as other datasets. In this section, we show that using the resolution of 32x32 for pre-training, we can indeed outperform [13] but also highlight why this may not be suitable for privacy applications.

a) Low-resolution (CIFAR specific) pre-training: Different private tasks/datasets may have images of differing resolutions. While all images in CIFAR [44] are 32x32 dimensional, in other datasets, images not only have higher resolution but their resolution varies widely. For example, GTSRB [51] has images of size 222x193 as well as 15x15 , PCAM [53] has 96x96 dimensional images, most images in Dermnet [54] have resolution larger than 720x400, and in Pneumonia [52] most x-rays have a dimension higher than 2000x2000. Therefore, it may not be possible to fine-tune the feature extractor at a single resolution for such datasets.

Identifying the optimal pre-training resolution for each private dataset is beyond the scope of our work and orthogonal to the contributions of our work (as we extensively show, our method PILLAR operates well under several pre-training strategies in Figure 10 and Figure 12). Furthermore, assuming the pre-training and private dataset resolution to be perfectly aligned is a strong assumption.

b) Comparison with [13]: Nevertheless, we compare our approach with [13] pre-training a ResNet50 on a 32x32 rescaled ImageNet version, and obtain a non-private accuracy larger than 94% reported for $\epsilon = 8$ in Table 5 in [13] for *Classifier training*. Note that our approaches is computationally significantly cheaper than theirs as we do no use the tricks proposed in their work (including Augmult, EMA, and extremely large batch sizes ($>$ 16K))

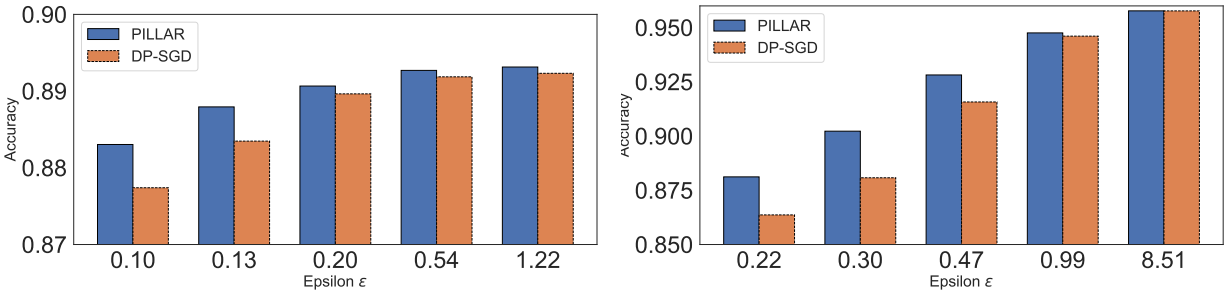
Using ImageNet32x32 for pre-training, we perform slightly better than them in private training. Our results are reported in Table VI. We expect that applying their techniques will result in even higher accuracies at the cost of computational efficiency. Interestingly, Table VI shows that our model's accuracy for $\epsilon = 0.7$ on CIFAR10, is as good as [13] for $\epsilon = 1.0$. This provides evidence that large batch sizes, which is one of the main hurdles in producing deployable private machine learning models, might not be required using our approach.

C. Experiments with large ϵ (≥ 1)

While in most of the paper, we focus on settings with small ϵ, in certain practical settings, the large epsilon regime may also be important. In Table VII, we repeat our experiments for CIFAR10 and CIFAR100 with $\epsilon \in 1, 2$ and report the accuracy for the best projection dimension. Our results show that for $\epsilon \in \{0.1, 0.7, 1, 2\}$ our method can provide significant gains on the challenging dataset of CIFAR-100; however for CIFAR-10 with $\epsilon = 1, 2$ the improvements are more modest.

Privacy	Pre-training	CIFAR10		CIFAR100	
		Ours	No Projection	**Ours**	No Projection
$\epsilon = 1$	SL	86.4	85.4	58.8	54.4
	SSL	81.4	80.5	49.0	45.8
$\epsilon = 2$	SL	86.8	86.4	61.8	60.0
	SSL	82.5	81.9	53.03	50.06

TABLE VII: Experiment with larger ϵ. Pre-training is with ImageNet 224x224.



(a) Image Space

(b) ResNet50 Representations

Fig. 8: DP Training of linear classifier on a) Images and b) representations obtained from pre-trained ResNet-50.

D. Comparison with PATE

We now discuss discuss the *PATE* family of approaches [19], [20], [66], [67]. These methods partition the training set into disjoint subsets, train an ensemble of teacher models on them, and use them to pseudo-label a public dataset using a privacy-preserving mechanism. For PATE to provide tight privacy guarantees, a large number (150-200 [20]) of subsets is needed, which reduces the test accuracy of each teacher. Large amounts of public data are also required. For CIFAR-10, [20], [66] use 29000 examples (58% of training set size), whereas we only use 5000 (10% of training set size) public unlabelled data points (and to retain its accuracy, in Section V-B we show 500 (1%) samples are sufficient). Of these 29000 examples, [66] reports only half of them is labelled due to the private labelling mechanism, further limiting the student's performance in settings with low amounts of public training data. Despite our best attempts, we could not train PATE-based approaches in our challenging setting to satisfactory levels of accuracy on either CIFAR-10 or CIFAR-100.[11]

E. Additional Datasets

In this section, we look at results on the MNIST dataset. We consider the standard train-test split of MNIST and train two types of classifiers. The first classifier is the standard linear classifier with cross-entropy loss. The second is a linear classifier with standard cross entropy loss trained on representations of MNIST images obtained from a Resnet-50, pre-trained on ImageNet. Our results, plotted in Figure 8, shows that PILLAR consistently outperforms DP-SGD and the improvement is more prominent for smaller values of ϵ. We also investigated how the best projection dimension k varies as a function of ϵ. The results are shown in Figure 9. As indicated by Theorem 1, the best k increases as ϵ increases.

In addition to results on MNIST dataset, we also conduct experiments using PILLAR on tabular datasets. We select two datasets: Guillermo and Riccardo from the OpenML [69] respository. Both of these are binary classification datasets with 4096 dimensions and $16,000$ data points. We train logistic regression models on them using both PILLAR and vanilla DP-SGD. The results presented in Table VIII show that PILLAR consistently outperforms DP-SGD on both of these datasets.

Privacy	PILLAR	DP-SGD
$\epsilon = 0.1$	75.25	57.8
$\epsilon = 0.3$	76.5	65.4
$\epsilon = 1.0$	78.2	70.2
$\epsilon = 5.0$	79.3	73.6

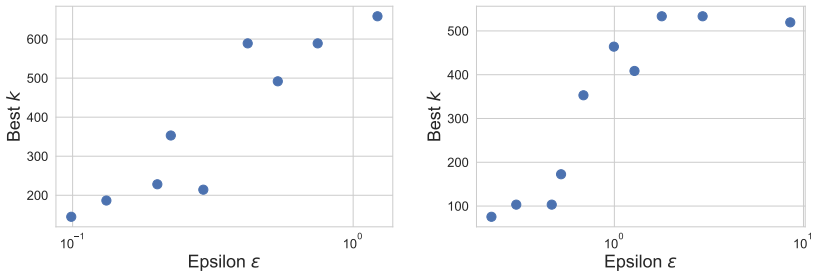
Riccardo

Privacy	PILLAR	DP-SGD
$\epsilon = 0.1$	60.19	52.3
$\epsilon = 0.3$	61.78	54.6
$\epsilon = 1.0$	63.10	59.2
$\epsilon = 5.0$	64.35	61.62

Guillermo

TABLE VIII: Comparison of PILLAR with DP-SGD on Riccardo and Guillermo datasets from the OpenML repository [69].

[11]For reference, we refer the reader to the accuracies reported for the state-of-the-art implementation in [68] (Table 12) and [66] (Table 1), which are less than 40% and 75% respectively, whereas we obtain more than 85% for tighter privacy guarantees.



(a) Image Space (b) ResNet50 Representations

Fig. 9: Best projection dimension k as a function of ϵ on the MNIST dataset.

	Public Data	ResNet50 SL	ConvNeXt-XL
Datasets		CIFAR100	CIFAR100
ϵ		0.1 0.7 1.0	0.1 0.7 1.0
DP-RAFT	Unlabelled	28.80 58.35 61.79	68.38 79.12 83.69
DP-RAFT + PILLAR	None	38.75 62.49 64.32	74.69 82.89 85.12

TABLE IX: Results comparing DP-RAFT and DP-RAFT+PILLAR.

F. DP-RAFT Experiments

In this section we present some results that combine DP-RAFT and PILLAR to yield further accuracy improvements. We perform our experiments on CIFAR100, considering learning rate values in $\{0.1, 0.01, 1\}$, training for a number of epochs in $\{5, 10, 50\}$ and for $\epsilon \in \{0.1, 0.7, 1.0\}$. For PILLAR we consider $k \in \{40, 100, 200, 300, 400\}$. In Table IX we compare the performance of DP-RAFT and the combination of DP-RAFT+PILLAR for ResNet50 and, since the authors of [36] consider also additional backbones, we also show the effectiveness of our method on the ConvNeXt-XL backbone. As it can be seen, in all cases using PILLAR in conjunction with DP-RAFT induces a performance improvement.

G. Experimental details for Section IV-B

In this section, we provide details of the other algorithms we compare our approach with in Section IV-B. We use the PRV accountant [70] for all experiments.

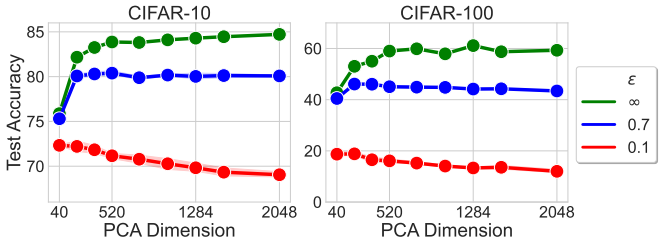
a) JL transformation [27]: [27] uses JL transformation to reduce the dimensionality of the input. For our baseline, we simulate this method by using Random Matrix Projection using Gaussian Random Matrices (GRM) instead of PCA to reduce the dimensionality of the inputs. Our experimental results in Table II show that our method outperforms these approaches. Although this approach does not require the availability of public data, this comparison allows us to conclude that reducing the dimensionality of the input is not sufficient to achieve improved performance. Furthermore, even though the JL Lemma [71] guarantees distances between inputs are preserved up to a certain distortion in the lower-dimensionality space, the dataset size required to guarantee a small distortion is much larger than what is available in practice. We leverage `scikit-learn` to project the data to a target dimension identical to the ones we use for PCA. We similarly search the same hyperparameter space.

b) GEP [17]: We use the code-base[12] released by the authors for implementation of GEP. We conduct hyper-parameter search for the learning rate in $\{0.01, 0.05, 0.1, 1\}$ and the number of steps in $\{500, 1000, 2500, 3000, 5000, 6000, 20000\}$. As recommended by the authors, we set the highest clipping rate to $\{1, 0.1, 0.01\}$ and the lowest clipping rate is obtained by multiplying the highest with 0.20. The anchor dimension ranges in $\{40, 120, 200, 280, 512, 1024, 1580\}$. We try batch sizes in $\{64, 512, 1024\}$. We tried using $\{0.1\%, 0.01\%\}$ of the data as public. Despite this extensive hyperparameter search, we could not manage to make GEP achieve better performance than the DP-SGD baseline (see Table II).

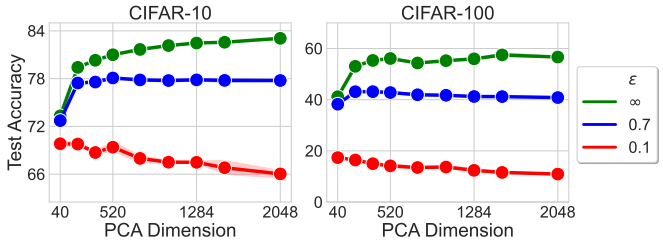
c) AdaDPS [18]: We use the code-base[13] released by the authors of AdaDPS. We estimate the noise variance as a function of the number of steps and the target ϵ using the code of `opacus` under the RDP accountant (whose implementation is the same as the code released by the authors of AdaDPS). We search the learning rate in $\{0.01, 0.1, 1\}$, the number of steps in $\{500, 1000, 2500, 3000, 5000, 6000, 7500, 10000\}$, the batch size in $\{32, 64, 128, 512, 1024\}$, and we tried using $\{0.1\%, 0.01\%\}$ of the data as public, and the ϵ_c (the conditioner hyperparameter) in $\{10, 1, 0.1, 1e-3, 1e-5, 1e-7\}$. of the validation data for the public data conditioning. We applied micro-batching in $\{2, 4, 32\}$. Despite this extensive hyperparameter search, we could not manage to make AdaDPS achieve better performance than the DP-SGD baseline.

[12]https://github.com/dayu11/Gradient-Embedding-Perturbation

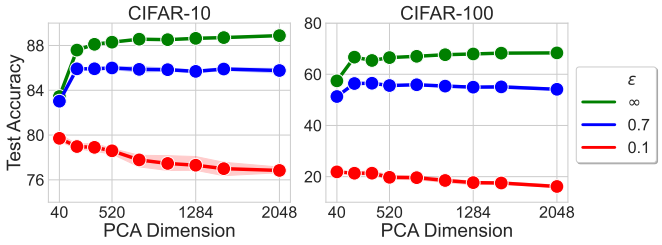
[13]https://github.com/litian96/AdaDPS



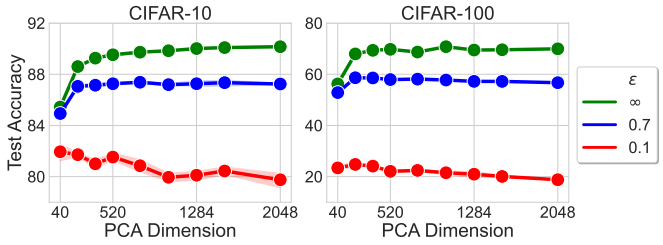
(a) BYOL Pretraining



(b) MoCov2+ Pretraining



(c) SemiSL Pretraining



(d) SemiWeakSL Pretraining

Fig. 10: DP Training of linear classifier on different pre-trained features using the PRV accountant for CIFAR-10 and CIFAR-100.

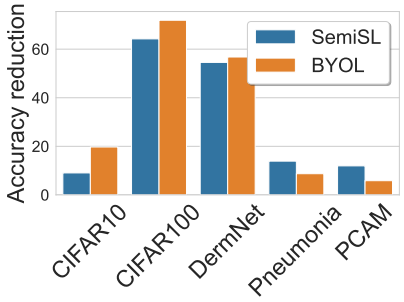


Fig. 11: Comparing reduction in test accuracy for different datasets between using SemiSL and BYOL pre-trained networks.

d) DP-PCA [29]: All settings are the same with respect to PILLAR, except for the additional need of cross-validating the privacy budget consumed by the DP-PCA procedure. We consider $1\%, 25\%, 50\%$. For DP-PCA, we use the `diffprivlib` implementation.

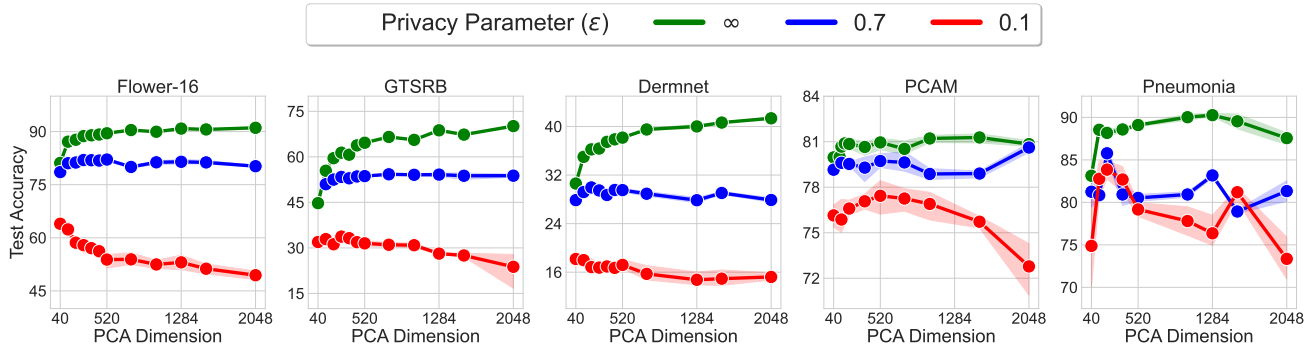
H. Different pre-training algorithms

In Figure 4 and 5 in the main text, we only reported accuracies for the best performing pre-training algorithm. In this, section we report the performance of our algorithm against the remaining pre-training algorithms that we consider in this paper. In particular, we consider two self-supervised pre-training algorithms: BYOL [30] and MoCov2+ [33] and two semi-supervised algorithms [34]. While one of them is a Semi-Supervised (SemiSL) algorithm, the other only uses weak supervision and we refer to it Semi-Weakly Supervised (SemiWeakSL) algorithm. In Figure 10 we report the results on CIFAR-10 and CIFAR-100. In Figure 12 we report the results for Flower-16 [50], GTSRB [51], PCAM [53], Pneumonia [52] and DermNet [54].

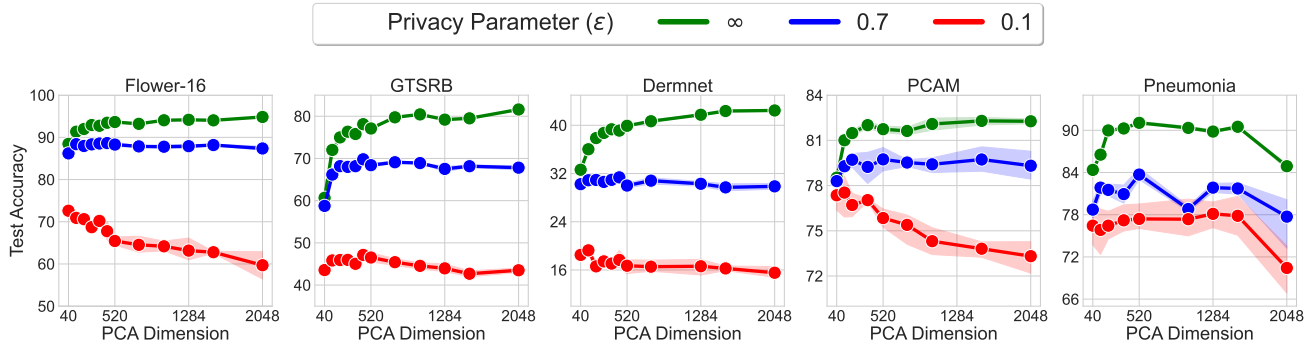
Similar to Figure 6 in the main text, we show the accuracy reduction for Semi-Supervised pre-training vs BYOL (Self-Supervised) pre-trianing in Figure 11. Our results shows similar results as [55] that labels are more useful for pre-training for tasks where there is a significant label overlap between the pre-training and the final task.

APPENDIX C
COMPUTATIONAL COST, BROADER IMPACT AND LIMITATIONS

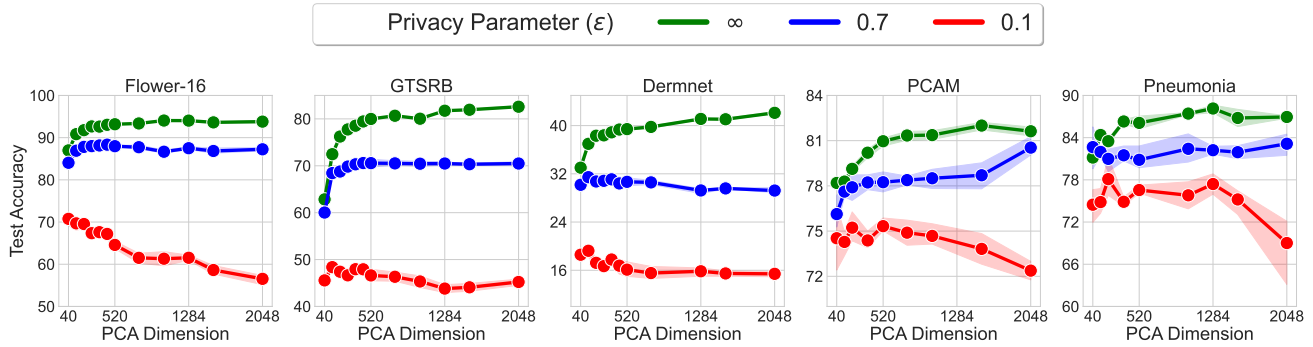
a) Computational cost: Except for the supervised training on ImageNet32x32, we leverage pre-trained models. To optimize the training procedure, we checkpoint feature embeddings for each dataset and pre-trained model. Therefore, training requires loading the checkpoint and training a linear layer via SGD (or DP-SGD), accelerating the training procedure by avoiding the forward pass through the feature encoder. We use a single Tesla M40 (11GB) for each run.



(a) MoCov2+ Pretraining



(b) SemiSL Pretraining



(c) SemiWeakSL Pretraining

Fig. 12: DP Training of linear classifier on different pre-trained features using the PRV accountant for Flower-16, GTSRB, DermNet, PCAM, and Pneumonia.

b) Broader impact and Limitations: In this work we show our method can be used in order to increase the utility of models under tight Differential Privacy constraints. Increasing the utility for low ϵ is crucial to foster the adoption of DP methods that provide provable guarantees for the privacy of users. Further, unlike several recent works that have shown improvement in accuracy for deep neural networks, our algorithm can be run on commonly available computational resources like a Tesla M40 11GB GPU as it does not require large batch sizes. We hope this will make DP training of high-performing classifiers more accessible. Finally, we show our algorithm improves not only on commonly used benchmarks like CIFAR10 and CIFAR100 but also in privacy relevant tasks like medical datasets including Pneumonia, PCAM, and DermNet. We hope this will encourage future works to also consider benchmarking their algorithms on more privacy relevant tasks.

As discussed, the assumption that labelled public data is available may not hold true in several applications. Our algorithm does not require the public data to be labelled, however the distribution shift between the public unlabelled data and the private one should not be too large. We have shown that for relatively small distribution shift our method remains effective. Finally, recent works have suggested that differentially private learning may disparately impact certain subgroups more than others [72]–[74]. It remains to explore whether semi-private learning can help alleviate these disparity.