Neural Collapse Meets Differential Privacy: Curious Behaviors of NoisyGD with Near-perfect Representation Learning

Chendi Wang [*1] **Yuqing Zhu** [*2] **Weijie J. Su** [1] **Yu-Xiang Wang** [3]

Abstract

A recent study by De et al. (2022) shows that large-scale representation learning through pre-training on a public dataset significantly enhances differentially private (DP) learning in downstream tasks. To explain this, we consider a layer-peeled model in representation learning, resulting in Neural Collapse (NC) phenomena. Within NC, we establish that the misclassification error is independent of dimension when the distance between actual and ideal features is below a threshold. We empirically evaluate feature quality in the last layer under different pre-trained models, showing that a more powerful pre-trained model improves feature representation. Moreover, we show that DP fine-tuning is less robust compared to non-DP fine-tuning, especially with perturbations. Supported by theoretical analyses and experiments, we suggest strategies like feature normalization and dimension reduction methods such as PCA to enhance DP fine-tuning robustness. Conducting PCA on last-layer features significantly improves testing accuracy.

1. Introduction

Recently, privately fine-tuning a publicly pre-trained model with differential privacy (DP) has become the workhorse of private deep learning. For example, De et al. (2022) demonstrates that fine-tuning the last-layer of an ImageNet pre-trained Wide-ResNet achieves an accuracy of 95.4% on CIFAR-10 with $(\epsilon = 2.0, \delta = 10^{-5})$-DP, surpassing the 67.0% accuracy from private training from scratch with a three-layer convolutional neural network (Abadi et al., 2016). Additionally, Li et al. (2021); Yu et al. (2021) show that pre-trained BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2018) models achieve near no-privacy utility trade-off when fine-tuned for sentence classification and generation tasks.

However, the empirical success of privately fine-tuning pre-trained large models appears to defy the worst-case dimensionality dependence in private learning problems — noisy stochastic gradient descent (NoisySGD) requires adding noise scaled to $\sqrt{p}$ to each coordinate of the gradient in a model with p parameters, rendering it infeasible for large models with millions of parameters. This suggests that the benefits of pre-training may help mitigate the dimension dependency in NoisySGD. A recent work (Li et al., 2022) makes a first attempt on this problem — they show that if gradient magnitudes projected onto subspaces decay rapidly, the empirical loss of NoisySGD becomes independent to the model dimension. However, the exact behaviors of gradients remain intractable to analyze theoretically, and it remains uncertain whether the "dimension independence" property is robust across different fine-tuning applications.

In this work, we explore private fine-tuning behaviors from an alternative direction — we employ an representation of pre-trained models using the Neural Collapse (NC) theory (Papyan et al., 2020) and study the dimension dependence in a specific private fine-tuning setup — fine-tuning only the last layer of the pre-trained model, a benchmark method in private fine-tuning.

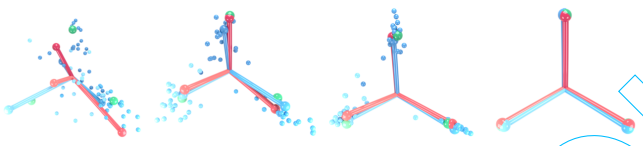


Figure 1. The figure depicts the evolution of the feature layer outputs of a VGG-13 neural network when trained on CIFAR-10 with three randomly selected classes. Each class is represented by a distinct color in the small blue sphere. As the training progresses, the last-layer feature means collapse onto their corresponding classes. Credit to Papyan et al. (2020).

To elaborate, Neural Collapse (Papyan et al., 2020; Fang et al., 2021; He & Su, 2023) is a phenomenon observed in

[*]Equal contribution [1]University of Pennsylvania, Philadelphia, PA, USA. [2]University of California, Santa Barbara, Santa Barbara, CA, USA. [3]University of California, San Diego, La Jolla, CA, USA. Correspondence to: Yu-Xiang Wang <yuxiangw@ucsd.edu>, Chendi Wang <chendi@wharton.upenn.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

the late stage of training deep classifier neural networks. At a high level, NC suggests that when a deep neural network is trained on a K-class classification task, the last-layer feature converges to K distinct points (see Figure 1 for $K = 3$). This insight provides a novel characterization of deep model behaviors, sparking numerous subsequent studies that delve into this phenomenon (Masarczyk et al., 2023; Li et al., 2023a).

To summarize, we aim to answer the following questions about private learning under Neural Collapse.

1. *When fine-tuning the last layer, what specific property of the features of that layer is important to make private learning dimension-independent?*

2. *How robust is this dimension-independence property against perturbations?*

1.1. Contributions of this paper

In essence, our contribution lies in correlating the performance and robustness of privately fine-tuning with the recently proposed Neural Collapse theory.

Theoretically, we identify a key structure of the last-layer feature that leads to a dimension-independent misclassification error (using 0-1 loss) in noisy gradient descent (NoisyGD). We formalize this structure through the feature shift parameter β, which measures the ℓ_∞-distance between the actual last-layer features obtained from the pre-trained model on a private set, and the ideal maximally separable features posited by Neural Collapse theory. A smaller value of β indicates a better representation of the last-layer feature. We show that if the feature shift parameter β remains below a certain threshold related to the model dimension p, the sample complexity bound of achieving a γ misclassification error is dimension-independent.

Empirically, we evaluate the feature shift vector β when fine-tuning different transformers on CIFAR-10. Figure 2 plots the distribution of per-class β when the pre-trained transformer is either the Vision Transformer (ViT) or ResNet-50. Blue and green scatter plots represent the β values for each sample from the two models, while purple and yellow scatter plots denote the median β within each class. The median β is centered around 0.10 for ViT and around 0.20 for ResNet-50. The pre-trained ViT model is known to have better feature representations than the ResNet-50 model. Our results show that: (1) β is bounded for the two pre-trained models, with very few outliers, and (2) the better the pre-trained model, the smaller the β. For example, ViT (with $\beta \approx 0.1$) outperforms ResNet-50 (with $\beta \approx 0.2$) due to its smaller shift parameter. We postpone the details of our experiments to Section A.

Moreover, we study the robustness of the "dimension-independence" property against various types of perturbations, including stochastic, adversarial, and offset perturbations. Each perturbation type will alter the feature presentations, increasing β and potentially making the sample complexity bound dimension-dependent. Notably, we find that the adversarial perturbations impose a stricter limitation on β, lowering the acceptable upper-bound threshold for β from $p^{-\frac{1}{2}}$ to p^{-1}, meaning that adversarial perturbations are more fragile compared to other types of perturbations.

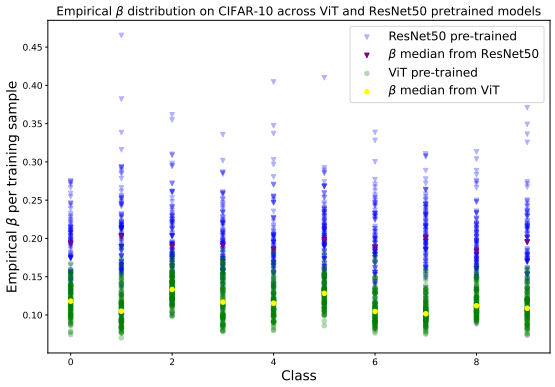


Figure 2. **CIFAR-10.** A figure depicting the feature shift parameter β when fine-tuning different pre-trained models on CIFAR-10. As observed, ViT performs better than ResNet-50, as the shift parameter is much smaller. The feature shift vectors are quite stochastic.

Moving forward, to mitigate the "non-robustness of dimension-independency" issue under perturbations, we propose solutions to enhance the robustness of NoisyGD. We find out that releasing the mean of feature embeddings effectively neutralizes offset perturbation (detailed in Section 4.2), and dimension reduction techniques like PCA (detailed in Section 4.1) reduce the constraint on the feature shift parameter β, thus improve NoisyGD's robustness. Our theoretical results shed light on the recent success of private fine-tuning. Specifically, our analysis on PCA provides a plausible explanation for the effectiveness of employing DP-PCA in the first private deep learning work (Abadi et al., 2016).

We summarize our contributions below:

- We present a direct analysis of the misclassification error (on population) under 0-1 loss. Existing analyses of the excess risk on population (Bassily et al., 2014; 2019; 2020; Feldman et al., 2020) mainly focus on the excess risk under convex surrogate loss, leading to sample complexity bounds (inverse) polynomial in the required excess risk. We show that Neural Collapse theory allows us to directly bound 0-1 loss, which results in a logarithmic sample complexity bound in the misclassification error γ.

- We introduce a feature shift parameter β to measure the discrepancy between actual and ideal last-layer features. Our theoretical findings show that this feature structure is crucial to guarantee NoisyGD's resistance to dimension dependence. In particular, when $\beta \leq p^{-\frac{1}{2}}$, the sample complexity for NoisyGD becomes $O(\log(1/\gamma))$, which is dimension-independent. We also provide an empirical evaluation of this feature shift parameter β on CIFAR-10 using ImageNet pre-trained vision transformer and ResNet-50, which shows that the vision transformer outperforms ResNet-50 as it leads to a smaller β.

- We theoretically analyze the misclassification error and robustness of NoisyGD against several types of perturbations and class-imbalanced scenarios, with sample complexity bounds summarized in Table 1.

- We empirically validate our theoretical findings on private fine-tuning vision models. We show that privately fine-tuning an ImageNet pre-trained vision transformer is not affected by the last-layer dimension on CIFAR-10. However, we observe a degradation in the utility-dimension trade-off when a minor perturbation is introduced, aligning with our theoretical results.

- We propose two solutions to address the non-robustness challenges in DP fine-tuning with theoretical insights. In particular, we empirically apply PCA on the perturbed last-layer feature before private fine-tuning, demonstrating that the utility of NoisySGD remains unaffected by the original feature dimension. We believe these results will provide new insights into enhancing the robustness of private fine-tuning.

Feature	Sample Complexity
Perfect Neural Collapse	$\frac{2\sqrt{\log(1/\gamma)}}{\sqrt{\rho}}$
GD	$p\beta^2 \log \frac{1}{\gamma}$
NoisyGD	$p\beta^2 \log \frac{1}{\gamma} + \frac{p\beta^2\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
stochastic (test)	$\frac{\max\{\sqrt{p}\beta,1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
adversarial (test)	$\frac{\max\{p\beta,1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
offset (training)	$\frac{\max\{\sqrt{p}\beta,1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
offset + Class imbalance	$\frac{\max\{\sqrt{p}\beta,1\}\sqrt{\log(1/\gamma)}}{(1-\beta+2\beta\alpha)\sqrt{2\rho}}$

Table 1. Summary of the sample complexity of achieving a misclassification error γ of private learning under ρ-zCDP. We consider perfect features, actual features (GD and NoisyGD), and perturbed features (stochastic, adversarial, offset perturbations). For the actual features, we assume that the feature shift vectors of traing and testing features are from the same distribution. We have also considered the effects of α-class imbalance. The full version can be found in Table 2 in Appendix.

2. Preliminaries and Problem Setup

In this section, we review the private fine-tuning literature, introduce the Neural Collapse phenomenon, and formally describe the private fine-tuning problem under (approximate) Neural Collapse.

Symbols and notations. Let the data space be $\mathcal{Z}$. A dataset $\mathcal{D}$ is a set of individual data points $\{z_1, z_2, ...\}$ where $z_i \in \mathcal{Z}$. Unless otherwise specified, the size of the data $|\mathcal{D}| := n$. $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the feature space and $\mathcal{Y}$ the label space with $\mathcal{X} \subset \mathbb{R}^p$ and $\mathcal{Y} = \{1, 2, ..., K\}$. A data point z_i is a feature-label pair (x_i, y_i). Occasionally, we overload y_i to also denote the one-hot representation of the label. We use standard probability notations, e.g., $\Pr[\cdot]$ and $\mathbb{E}[\cdot]$ for probabilities and expectations. Other notations will be introduced when they first appear.

Differentially private learning. The general setting of interest is *differentially private learning* where the goal is to train a classifier while satisfying a mathematical definition of privacy known as differential privacy (Dwork, 2006). DP ensures that any individual training data point cannot be identified using the trained model and any additional side information. More formally, we adopt the popular modern variant called zero-centered Concentrated Differential Privacy (zCDP), as defined below.

Definition 2.1 (Zero-Concentrated Differential Privacy, zCDP, Bun & Steinke (2016))**.** Two datasets $\mathcal{D}_0, \mathcal{D}_1$ are neighbors if they can be constructed from each other by adding or removing one data point. A randomized mechanism $\mathcal{A}$ satisfies ρ-zero-concentrated differentially private (ρ-zCDP) if, for all neighboring datasets $\mathcal{D}_0$ and $\mathcal{D}_1$, we have $R_\alpha(\mathcal{A}(\mathcal{D}_0)\|\mathcal{A}(\mathcal{D}_1)) \leq \rho\alpha$, where $R_\alpha(P\|Q) = \frac{1}{\alpha-1}\log\int\left(\frac{p(x)}{q(x)}\right)^\alpha q(x)dx$ is the Rényi divergence between two distributions P and Q.

In the above definition, $\rho \geq 0$ is the privacy loss parameter that measures the strength of the protection. $\rho = 0$ indicates perfect privacy, $\rho = \infty$ means no protection at all.

The goal of differentially private learning is to come up with a differentially private (ρ-zCDP) algorithms that outputs a classifier $f : \mathcal{X} \to \mathcal{Y}$ such that the misclassification error $\mathrm{Err}(f) = \mathbb{E}_{(x,y)\sim\mathcal{P}}\left[\mathbf{1}(f(x) \neq y)\right]$ is minimized (in expectation or with high probability), where $\mathcal{P}$ is the data distribution under which the training data is sampled from i.i.d. For reasons that will become clear soon, we will focus on linear classifiers parameterized by $W \in \mathbb{R}^{K\times p}$ of the form $f_W(x) = \arg\max_{y\in[K]}[Wx]_y$.

Noisy gradient descent. Noisy Gradient Descent or its stochastic version Noisy Stochastic Gradient Descent (Song et al., 2013; Abadi et al., 2016) is a fundamental algorithm

in DP deep learning. To minimize the loss function $\mathcal{L}(\theta) := \sum_{i=1}^{n} \ell(\theta, z_i)$, the NoisyGD algorithm updates the model parameter θ_t by combining the gradient with an isotropic Gaussian.

$$\theta_{t+1} = \theta_t - \eta_t \left(\sum_{i=1}^{n} \nabla \ell(\theta_t, z_i) + \mathcal{N}\left(0, \frac{G^2}{2\rho} I_p\right) \right). \quad (1)$$

Here G is the ℓ_2-sensitivity of the gradient, and the algorithm runs for T iterations that satisfy $T\rho$-zCDP.

However, the excess risk on population of NoisySGD must grow as $\sqrt{p}/\epsilon$ (Bassily et al., 2014; 2019), which limits private deep learning benefit from model scales. To overcome this, DP fine-tuning (De et al., 2022; Li et al., 2021; Bu et al., 2022) is emerging as a promising approach to train large models with privacy guarantee.

Private fine-tuning. In DP fine-tuning, we begin by pre-training a model on a public dataset and then privately fine-tuning the pre-trained model on the private dataset. Our focus is on fine-tuning the last layer of pre-trained models using the NoisySGD/NoisyGD algorithm, which has consistently achieved state-of-the-art results across both vision and language classification tasks (De et al., 2022; Tramer & Boneh, 2020; Bao et al., 2023). However, we acknowledge that in some scenarios, fine-tuning all layers under DP can result in better performance, as demonstrated in the CIFAR-10 task by De et al. (2022). The comprehensive analysis of dimension-dependence in other private fine-tuning benchmarks remains an area for future investigation.

Theoretical setup for private fine-tuning. For a K-class classification task, we rewrite each data point z as $z = (x, y)$ with $x \in \mathbb{R}^p$ being the feature and $y = (y_1, \cdots, y_K) \in \{0,1\}^K$ being the corresponding label generated by the one-hot encoding, that is y belongs to the k-th class if $y_k = 1$ and $y_j = 0$ for $j \neq k$.

When applying NoisyGD for fine-tuning the last-layer parameters, the model is in a linear form. Thus, we consider the linear model $f_W(x) = Wx$ with $W \in \mathbb{R}^{K \times p}$ being the last-layer parameter to be trained and x is the last layer feature of a data point. The parameter θ_t in NoisyGD is the vectorization of W. Let $\ell : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}$ be a loss function that maps $f_W(x) \in \mathbb{R}^K$ and the label y to $\ell(f_W(x), y)$. For example, for the cross-entropy loss, we have $\ell(f_W(x), y) = -\sum_{i=1}^{K} y_i \log[(f_W(x))_i]$.

Misclassification error. For an output $\widehat{W}$ and a testing data point (x, y), the misclassification error we considered is defined as $\Pr\left[y \neq f_{\widehat{W}}(x)\right]$, where the probability is taken with respect to the randomness of $\widehat{W}$ and $(x, y) \sim \mathcal{P}$.

Beyond the distribution-free theory. Distribution-free learning with differential privacy is however known to be statistically intractable even for linear classification in 1-dimension (Chaudhuri & Hsu, 2011; Bun et al., 2015; Wang et al., 2016). Existing work resorts to either proving results about (convex and Lipschitz) surrogate losses (Bassily et al., 2014) or making assumptions on the data distribution (Chaudhuri & Hsu, 2011; Bun et al., 2020). For example, Chaudhuri & Hsu (2011) assumes bounded density, and Bun et al. (2020) shows that linear classifiers are privately learnable if the distribution satisfies a large-margin condition. Our setting, as detailed in Section 3.1.1, can be viewed as a *new family of distributional assumptions* motivated by the recent discovery of the Neural Collapse phenomenon. As we will see, these assumptions not only make private learning statistically and computationally tractable (using NoisyGD), but also produce sample complexity bounds that are *dimension-free* and *exponentially faster* than existing results that are applicable to our setting.

Neural Collapse. Neural Collapse (Papyan et al., 2020; Fang et al., 2021; He & Su, 2023) describes a phenomenon about the last-layer feature structure obtained when a deep classifier neural network converges. It demonstrates that the last-layer feature converges to the column of an equiangular tight frame (ETF). Mathematically, an ETF is a matrix

$$M = \sqrt{\frac{K}{K-1}} P \left(I_K - \frac{1}{K} \mathbf{1}_K \mathbf{1}_K^T \right) \in \mathbb{R}^{p \times K}, \quad (2)$$

where $P = [P_1, \cdots, P_K] \in \mathbb{R}^{p \times K}$ is a partial orthogonal matrix such that $P^T P = I_K$. For a given dimension $d = p$ or K, we denote $I_d \in \mathbb{R}^d$ the identity matrix and denote $\mathbf{1}_d = [1, \cdots, 1]^T \in \mathbb{R}^d$. Rewrite $M = [M_1, \cdots, M_K]$ with M_k being the k-th column of M, that is, the ideal feature of the data belonging to class k.

We adopt the Neural Collapse theory to describe an ideal feature representation of applying the pre-trained model on the private set. However, achieving perfect collapse on the private set is an ambitious assumption, as in practice, the private feature of a given class is distributed around M_k. Therefore, we introduce a feature shift parameter β to measure the discrepancy between the actual feature and the perfect feature M_k.

Definition 2.2 (Feature shift parameter β)**.** For any $1 \leq k \leq K$, given a feature x belonging to the k-th class and the perfect feature M_k, we define $\beta = ||x - M_k||_\infty$ as the feature shift parameter of x that measures the ℓ_∞ distance between x and M_k.

Here, we use the ℓ_∞ norm since it is related to adversarial attacks, which are important in our study of the robustness of NoisyGD. Our numerical results in Figure 2 show that β is bounded on CIFAR-10 if the pretrained model is the vision transformer or ResNet-50.

3. Bounds on misclassification errors and robustness in private fine-tuning

In this section, we establish bounds on the misclassification error for both GD and the NoisyGD .

Section 3.1 aims to delineate the connection between the feature shift parameter β and the misclassification error. Additionally, we derive a threshold for β below which the misclassification error is dimension-independent.

In Section 3.2, our focus is the robustness of private fine-tuning. Specifically, we elucidate how various perturbations impact both β and the misclassification error.

3.1. Bounds on misclassification errors

We consider a binary classification problem with a training set $\{(x_i, y_i)\}_{i=1}^n$, where x_i represents features and $y_i \in \{\pm 1\}$ are the labels. For the broader multi-class scenarios, we state our theory for the perfect case in Section F.1. The rest theory can be extended to the multi-class case similarly.

For binary classification problems, the ETF $M = [M_1, M_2]$ satisfies $M_1 = -M_2$, which is equivalent to $M_1 = e_1$ and $M_2 = -e_1$ up to some rotation map (detailed in Appendix B). For a data point (x, y) with $y = 1$, recall the feature shift parameter $\beta = \|x - e_1\|_\infty$, which is the infinity norm of $v = x - e_1$. We call v a feature shift vector since it is the difference between an actual feature and the perfect one. Similarly, if $y = -1$, the feature shift vector is $v = x + e_1$. For a training set $\{(x_i, y_i)\}_{i=1}^n$, let $\{v_i = x_i - \text{perfect feature}\}_{i=1}^n$ be a sequence of feature shift vectors.

We first consider the scenario where the shift vectors v_i's are i.i.d. copies of a symmetric centered random vector v with $\|v\|_\infty \le \beta$ in Section 3.1.1. This setting is quite practical as can be seen from Figure 1. Furthermore, acknowledging that at times, the feature may be influenced by a fixed vector, such as an offset shift vector which will not change the margin and angles between features, we also investigated the case where v_i is deterministic in Section 3.1.2.

3.1.1. Stochastic shift vectors

For conciseness, we mainly focus on the results for 1-iteration NoisyGD, which is sufficient to ensure the convergence. As presented in Theorem C.1, our theory can be extended to multi-iteration projected NoisyGD. However, the dimension dependency can not be mitigated using multiple iterations. The proofs of all results in this section are given in Appendix C.

For 1-iteration GD without DP guarantee, the output is $\widehat{\theta}_{\mathrm{GD}} = \eta \sum_{i=1}^n y_i x_i$. Moreover, the 1-iteration NoisyGD outputs $\widehat{\theta}_{\mathrm{NoisyGD}} = \eta \sum_{i=1}^n y_i x_i + \mathcal{N}(0, \eta^2 \sigma^2)$. For the private learning problem, the sensitivity of the gradient is $G = \sup_{x_i} \|x_i\|_2 = \sqrt{1 + \beta^2 p}$, which is dimension dependent. If we still want G to be dimension independent, then every data point needs to be shrunk to $(e_1 + v)/\sqrt{1 + \|v\|_2^2}$. In both cases, the error bounds remain the same. In the testing procedure, we consider a testing point $(x, y) \sim \mathcal{P}$.

For an estimate $\widehat{\theta}$ whose randomness is from a training dataset drawn independently from $\mathcal{P}$ and a randomized algorithm $\mathcal{A}$, the misclassification error $\mathbb{E}_{\mathcal{A},\text{data}\sim\mathcal{P}^n}[\mathrm{Err}_{\mathcal{P}}(\mathcal{A}(\text{data}))]$ can be rewritten as $\mathbb{E}_{\mathcal{A},\text{data}\sim\mathcal{P}^n}[\mathrm{Err}_{\mathcal{P}}(\mathcal{A}(\text{data}))] = \Pr\left[y\widehat{\theta}^T x < 0\right]$, where the probability is taken with respect to the randomness of both $(x, y) \sim \mathcal{P}$ and $\widehat{\theta}$.

Theorem 3.1 (misclassification error for GD)**.** *Let $\widehat{\theta}_{\mathrm{GD}}$ be a predictor trained by GD under the cross entropy loss with zero initialization. Then, we have the following error bound on the misclassification error.*

- *If we assume that $\beta^2 p \le 1$, then it holds $\Pr[y\widehat{\theta}_{\mathrm{GD}}^T x < 0] = 0$ for n greater than the number of classes. As a result, to achieve a misclassification error γ, the sample complexity is constant.*
- *In general, if we assume that the sample is i.i.d., then the misclassification error is bounded as $\Pr[y\widehat{\theta}_{\mathrm{GD}}^T x < 0] \le \exp\left(-\frac{n}{2\left(\beta^4 p^2 + \frac{1}{3}\beta^2 p\right)}\right)$. Therefore, to achieve a misclassification error γ, the sample complexity is $O\left(p\beta^2 \log(1/\gamma)\right)$. If we further assume that all p components of v are independent of each other, then, it holds $\Pr[y\widehat{\theta}_{\mathrm{GD}}^T x < 0] \le \exp\left(-\frac{n}{2\left(\beta^4 p + \frac{1}{3}\beta^2\right)}\right)$. Thus, to achieve a misclassification error γ, the sample complexity is $O\left(p\beta^4 \log(1/\gamma)\right)$.*

Theorem 3.2 (misclassification error for NoisyGD)**.** *Let $\widehat{\theta}_{\mathrm{NoisyGD}}$ be a predictor trained by NoisyGD under the cross entropy loss with zero initialization. Then, we have the following error bound on the misclassification error.*

$$\Pr\left[y\widehat{\theta}_{\mathrm{NoisyGD}}^T x < 0\right] \le \exp\left(-\frac{n^2\rho}{2(1+\beta^2 p)^2}\right) + \exp\left(-\frac{n}{8\left(\beta^4 p^2 + \frac{1}{3}\beta^2 p\right)}\right). \quad (3)$$

As a result, to achieve a misclassification error γ, the sample complexity is $O\left(\frac{(1+\beta^2 p)^2\sqrt{\log\frac{1}{\gamma}}}{2\rho} + p\beta^2\log(1/\gamma)\right)$. If we further assume that all p components of v are independent of each other, then, it holds

$$\Pr\left[y\widehat{\theta}_{\mathrm{NoisyGD}}^T x < 0\right] \le \exp\left(-\frac{n^2\rho}{2(1+\beta^2 p)^2}\right) + \exp\left(-\frac{n}{8\left(\beta^4 p + \frac{1}{3}\beta^2\right)}\right).$$

To achieve a misclassification error γ, the sample complexity is $O\left(\frac{(1+\beta^2 p)^2\sqrt{\log\frac{1}{\gamma}}}{2\rho} + 4p\beta^4\log\frac{1}{\gamma}\right)$.

Remark. Note that with further assumptions on feature separability, the second term in Equation 3 (which aligns with GD in Theorem 3.1) can be improved from $\beta^2 p$ to $\beta^4 p$. However, the first term, caused by DP, remains unchanged by this assumption. Thus, NoisyGD has a stricter requirement on feature quality due to the added random noise. Theorems 3.1 and 3.2 indicate that the error bound is exponentially close to 0 under the following conditions: $\beta \le p^{-1/2}$ for both NoisyGD and GD and $\beta \le p^{-1/4}$ for GD under stronger assumptions. This result is dimension-independent when β satisfies the above conditions. Moreover, GD has robustness against larger shift vectors compared to NoisyGD. This aligns with the observations from our experiments detailed in Section 5, where we note a significant decrease in accuracy with increasing dimensionality. In addition, when $\beta \le p^{-1/2}$, the misclassification error for GD is always 0 while that for NoisyGD is $\exp\left(-\frac{n\rho}{1+\beta^2 p}\right)$.

Promising properties for perfect collapse. In the special case $\beta = 0$, all features are equivalent to the perfect feature. For this perfect scenario, numerous promising properties are outlined as follows. The details are discussed in Section F.1.

1. The error bound is exponentially close to 0 if $\rho \gg G^2/n^2$ — very strong privacy and very strong utility at the same time.
2. The result is dimension independent — it doesn't depend on the dimension p.
3. The result is robust to class imbalance for binary classification tasks.
4. The result is independent of the shape of the loss functions. Logistic loss works, while square losses also works.
5. The result does not require careful choice of learning rate. Any learning rate works equally well.

3.1.2. Deterministic shift vectors

We consider the case where each v_i is a fixed vector with $\|v_i\|_\infty \le \beta$. Recall that the 1-iteration NoisyGD outputs $\widehat{\theta}_{\text{NoisyGD}} = \eta\left(ne_1 + \sum_{i=1}^n v_i\right) + \mathcal{N}(0,\sigma^2)$.

Theorem 3.3 (misclassification error for NoisyGD)**.** *Let $\widehat{\theta}_{\text{NoisyGD}}$ be a predictor trained by NoisyGD under the cross entropy loss with zero initialization. Then, for β such that $1-\beta^2 p > 0$, we have* $\Pr[y\widehat{\theta}_{\text{NoisyGD}}^T x < 0] \le \exp\left(-\frac{n^2(1-\beta^2 p)^2}{(1+\beta^2 p)\sigma^2}\right)$. *As a result, to make the misclassification error less than γ, the sample complexity for n is* $O\left(\frac{(1+\beta^2 p)\log\frac{1}{\gamma}}{2\rho(1-\beta^2 p)^2}\right)$.

This misclassification error also corresponds to $\beta^2 p$, which is similar to the stochastic case. When $\beta^2 p < 1$, the misclassification error decays exponentially and the misclassification error is dimension-independent.

3.2. Robustness of NoisyGD under perturbations

In this section, we explore the robustness of NoisyGD against various perturbation types. For the sake of brevity, we focus on perturbations of the perfect feature ($\beta = 0$) by different attackers. The theoretical framework can easily be extended to include perturbations of actual features, following the same proof structure as outlined in Theorem 3.1 and Theorem 3.2. Our findings indicate that each mentioned perturbation type affects the feature shift parameter β, potentially increasing NoisyGD's dimension dependency.

3.2.1. Stochastic attackers

Non-robustness to perturbations in the training time. If the training feature is perturbed by some stochastic perturbation (while the testing feature is perfect), then, the misclassification error for GD is $\exp\left(-\frac{n^2}{\widetilde{\beta}^2}\right)$, which is dimension-independent for any $\widetilde{\beta} > 0$. However, the sample complexity for NoisyGD is $O\left(\sqrt{\frac{\max\{\widetilde{\beta}^2 p,1\}\log(1/\gamma)}{\rho}}\right)$. Thus, the NoisyGD is non-robust even when we only perturb the training feature with attackers that make $\widetilde{\beta} > p^{-1/2}$. We postpone the details to Appendix D.2

Non-robustness to perturbations in the testing time. If we only perturb the testing feature, then we still require $O\left(\frac{\max\{\sqrt{p}\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}\right)$ samples to achieve a misclassification error γ, which is still non-robust when $\widetilde{\beta} > p^{-1/2}$. The technical detail is similar to the proof of Theorem 3.2.

3.2.2. Deterministic attackers

Non robustness to offset perturbations in the training time. Even if we just shift the training feature vectors away by a constant offset (while keeping the same margin and angle between features), it makes DP learning a lot harder. Precisely, for some vector $v \in \mathbb{R}^p$, we consider $v_i = v$ for $y_i = 1$ and $v_i = -v$ for $y_i = -1$. Moreover, this makes absolutely no difference to the gradient, when we start from 0 because

$$\nabla\mathcal{L}(\theta) = \frac{n}{2}\cdot 0.5\cdot -(-e_1+v) + \frac{n}{2}\cdot 0.5\cdot(e_1+v) = \frac{n}{2}e_1.$$

Thus, for GD, the misclassification error is always 0. If all we know is that $\|v\|_\infty \leq \widetilde{\beta}$, the sample complexity for making the classification error less than γ will be $O\left(\frac{\max\{p\widetilde{\beta}^2,1\}\sqrt{\log(1/\gamma)}}{\sqrt{\rho}}\right)$. The details are given in Appendix D.1.

Non-robustness to class imbalance. Note that in the above case, it is quite a coincidence that v gets cancelled out in the non-private gradient. When the class is not balanced, the offset v will be part of the gradient that overwhelms the signal. Consider the case where we have αn data points with label -1 and $(1-\alpha)n$ data points with label 1 for $\alpha \neq 0.5$, and we start at 0, then $\nabla\mathcal{L}(\theta) = \frac{n}{2}e_1 + \frac{(1-2\alpha)n}{2}v$. If we allow the perturbation v to be adversarially chosen, then there exists v satisfying $\|v\|_\infty \leq \widetilde{\beta}$ such that the sample complexity bound to achieve a misclassification error γ is $O\left(\frac{\max\{\sqrt{p}\widetilde{\beta},1\}\sqrt{\log\frac{1}{\gamma}}}{\sqrt{(1-\widetilde{\beta}+2\widetilde{\beta}\alpha)^2\cdot\rho}}\right)$.

Non-robustness to adversarial perturbations in the testing time. When $v_i = 0$ and $\|v\|_\infty \leq \beta$, that is, we only consider perturbations in the testing time, if we allow the perturbation v to be adversarially chosen, then there exists v satisfying $\|v\|_\infty \leq \widetilde{\beta}$ such that the sample complexity bound to achieve misclassification rate γ is $O\left(\frac{G\max\{p\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}\right)$. One may refer to Appendix E.2 for the detail.

4. Solutions for non-robustness issues

In this section, we will explore various solutions for enhancing the robustness of NoisyGD. To deal with random perturbations, we suggest performing dimension reduction to reduce the feature shift parameter β, as detailed in Section 4.1. For offset perturbations, we will consider feature normalization to cancel out the perturbation, as discussed in Section 4.2. The proof of this section can be found in Appendix G.

4.1. Mitigating random perturbation: dimension reduction

In Abadi et al. (2016), dimension reduction methods, such as DP-PCA, were employed to enhance the performance of deep models. In this section, we demonstrate that applying PCA to the private features effectively improves robustness against random perturbations. Since we have a public dataset for pre-training a model, we consider performing dimension reduction with this public dataset. Similar to the PCA method, Pinto et al. (2024); Nguyen et al. (2020) demonstrate that DP learning also achieves dimension-free learning bounds on 0-1 losses by applying random projections or a transformation matrix to the features.

To perform dimension reduction, our goal is to generate a projection matrix $\widehat{P} = [\widehat{P}_1, \ldots, \widehat{P}_{K-1}] \in \mathbb{R}^{p\times(K-1)}$ and train with the dataset $(\widetilde{x}_i = \widehat{P}x_i, y_i)_{i=1}^n$. It is obvious that the "best projection" is one where the space spanned by $\widehat{P}$ matches the space spanned by $\{M_i\}_{i=1}^K$, with M_i being the perfect feature of the i-th class (the i-th column of an ETF).

In practice, it is not possible to obtain $\{M_i\}_{i=1}^K$ directly, and $\widehat{P}$ needs to be generated using another dataset $\{(\widehat{x}_i, \widehat{y}_i)\}_{i=1}^m$.

Recalling the binary classification problem with a training set $\{(x_i, y_i)\}_{i=1}^n$ and v_i as the feature shift vector of x_i, as discussed in Section 3.2, even in the case of class balance, the accuracy is not robust when $\beta^2 \geq 1/p$. Consider a projection vector $\widehat{P} = e_1 + \Delta$ with Δ satisfying $\|\Delta\|_\infty \leq \beta_0$. The following theorem shows that for $\beta\beta_0 < \frac{1}{p}$, the misclassification error decays exponentially and is dimension-independent.

Theorem 4.1. *For the NoisyGD trained with $\{(\widetilde{x}_i, y_i)\}_{i=1}^n$, the sample complexity to achieve a misclassification error γ is $n = O\left(\sqrt{\frac{G^2_{\beta,\beta_0,p}\log\frac{2}{\gamma}}{M_{\beta,\beta_0,p}\rho}}\right)$ with $G_{\beta,\beta_0,p} = 1 + \beta(1 + \beta_0 + p\beta_0)$ and $M_{\beta,\beta_0,p} = (1-\beta_0)^2 - p\beta\beta_0 - (1+\beta_0)(\beta + \beta_0 p) - (\beta + \beta_0 p)(1 + \beta + \beta_0 + \beta\beta_0 p)$.*

Theorem 4.1 suggests that dimension reduction can relax the requirement from $\beta^2 p \leq 1$ to $\beta\beta_0 p \leq 1$. Thus, dimension reduction can enhance robustness whenever $\beta_0 < \beta$. Typically, β_0 is relatively small and tends to 0 as m increases. The next question is how to construct the projection matrix $\widehat{P} = [\widehat{P}_1, \cdots, \widehat{P}_{K-1}] \in \mathbb{R}^{p\times(K-1)}$. We introduce the follwoing two methods.

Principle component analysis. Let $\{\widehat{P}_j\}_{j=1}^{K-1}$ be the the eigenvectors corresponding to $K-1$ largest eigenvalues of $\widehat{\Sigma} = \frac{1}{m}\sum_{i=1}^m \widehat{x}_i\widehat{x}_i^T$. For the binary case ($K = 2$), we have $\widehat{\Sigma}$ converges to $\Sigma = e_1 e_1^T + \widehat{\beta}^2 I_p$ for some constant $\widehat{\beta}$. Note that the eigenvector corresponding to the largest eigenvalue of Σ is the perfect feature e_1. As β_0 is the infinity norm of Δ, we use a bound on the infinity norm of eigenvectors (Fan et al., 2017). We state the results for $K = 2$ that can be extended to $K > 2$. Precisely, for $K = 2$, let $\widehat{P}$ be the eigenvector of $\frac{1}{m}\sum_{i=1}^m \widehat{x}_i\widehat{x}_i^T$ that corresponds to the largest eigenvalue. Then, it holds $\beta_0 = \|\widehat{P} - e_1\|_\infty \leq O\left(\frac{1}{\sqrt{m}}\right)$ with probability $O\left(pe^{-m^2}\right)$.

Releasing the mean of features. Let $X_k = \{\widehat{x}_i : \widehat{y}_i$ belongs to the k-th class$\}$. Let $\widehat{P}_k = \frac{1}{m_k}\sum_{\widehat{x}_i\in X_k}\widehat{x}_i$ with m_k being the size of X_k. Then, we have $\Delta = \widehat{P}_k - M_k = \frac{1}{m_k}\sum_{\widehat{x}_i\in X_k}\widehat{x}_i$. By the concentration inequality, we have $\beta_0 \leq O\left(\frac{\widehat{\beta}}{\sqrt{m_k}}\right)$ with probability $pe^{-m_k^2}$.

4.2. Addressing offset perturbations: normalization

Recall the shift perturbation where, for $x_i \in X_k$, we have $x_i = \widetilde{M}_k := M_k + v$ with some fixed vector v.

To deal with the offset perturbation v, we pre-process the feature as $\widetilde{x}_i = x_i - \frac{1}{n}\sum_{j=1}^{n} x_j$. Then, if the class is balanced, it holds $\widetilde{x}_i = \widetilde{M}_k - \frac{1}{K}\sum_{j=1}^{K} \widetilde{M}_j = M_k$ for $x_i \in X_k$. That is, the perturbations canceled out.

We still need to bound the sensitivity of the gradient when training with $\{(\widetilde{x}_i, y_i)\}_{i=1}^n$. If we delete arbitrary (x_j, y_j) from the dataset, then for the case $K = 2$ with data balance, the sensitivity of the gradient is $G = \frac{n}{n-1} \leq 2$, which is upper bounded by a dimension-independent constant. The sample complexity to achieve the misclassification error γ is $O\left(\frac{\sqrt{\log(1/\gamma)}}{\sqrt{\rho}}\right)$, which is dimension independent.

Note that this normalization method is not robust to class imbalance. In fact, if we consider the class imbalanced case with which we have αn data points with label $+1$ and the rest $(1-\alpha)n$ data points with label -1 for some $\alpha > 0$, then we have $\widetilde{x}_i = 2(1-\alpha)e_1$ for $y_i = 1$ and $\widetilde{x}_i = -2\alpha e_1$ for $y_i = -1$. In this class-imbalance case, one can recover the feature embedding e_1 and $-e_1$ by considering $\frac{\widetilde{x}_i}{\|\widetilde{x}_i\|_2}$. However, in this case, the sensitivity remains a constant G_α which, although independent of the dimension p, still relies on α.

5. Experiments

In this section, we will conduct three sets of experiments to validate our theoretical analysis in Section 3 and Section 4. The first experiment, outlined in Section 5.1, focuses on the synthetic perfect neural collapse scenario. In the second experiment, detailed in Section 5.2, we empirically investigate the behavior of NoisySGD (with fine-tuning the last layer) under different robustness settings and demonstrate that NoisySGD is "almost" dimension-independent, but this "dimension-independency" is not robust to minor perturbations in the last-layer feature. In the last experiment, discussed in Section 5.3, we demonstrate that dimension reduction methods such as PCA effectively enhance the robustness of NoisySGD.

5.1. Fine-tune NoisyGD with synthetic neural collapse feature

We first generate a synthetic data matrix $X \in \mathcal{R}^{n\times d}$ with feature dimension d under perfect neural collapse. The number of classes K is 10 and the sample size is $n = 10^4$. In the default setting, we assume each class draws n/K data from a column of K-ETF, the training starts from a zero weight θ and the testing data are drawn from the same distribution as X. The Gaussian noise is selected such that the NoisyGD is $(1, 10^{-4})$-DP.

In Figure 3(a), we observe that an imbalanced class alone does not affect the utility. However, NoisyGD becomes non-robust to class imbalance when combined with a private feature offset with $||\nu||_\infty = 0.1$. Additionally, it is non-robust to perturbed test data with $||\nu||_\infty = 0.1$.

5.2. Fine-tune NoisySGD with real datasets

In this section, we empirically investigate the non-robustness of neural collapse using real datatsets.

Precisely, we fine-tune NoisySGD with the ImageNet pre-trained vision transformer (Dosovitskiy et al., 2020) on CIFAR-10 for 10 epochs. Test features in the perturb setting are subjected to Gaussian noise with a variance of 0.1. The vision transformer produces a 768-dimensional feature for each image. To simulate different feature dimensions, we randomly sample a subset of coordinates or make copies of the entire feature space.

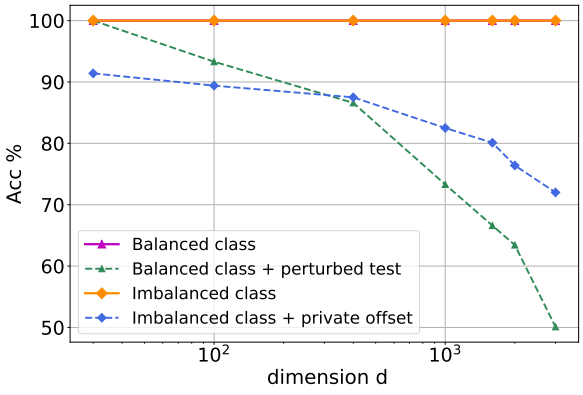
In Figure 3(b), we observe that while perturbing the testing features degrades the utility of both Linear SGD and NoisySGD, Linear SGD is generally unaffected by the increasing dimension. On the other hand, the accuracy of NoisySGD deteriorates significantly as the dimension increases.

5.3. Enhance NoisySGD's robustness with PCA

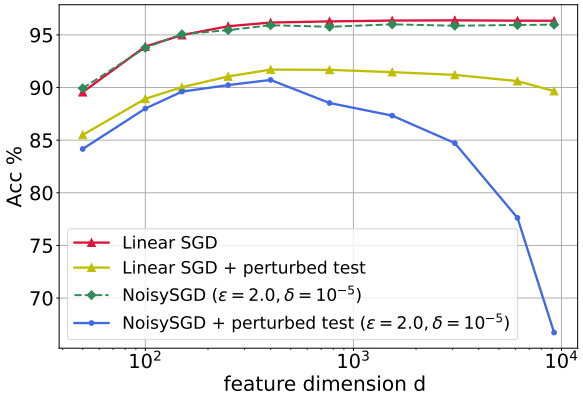
In this experiment, we replicate the set up from Exp 5.2, simulating different feature dimensions and injecting Gaussian noise with a variance of 0.1 to perturb all dimensions of both training and testing features. For simplicity, we apply PCA to the covariance matrix of private feature instead of a DP-PCA, followed by principal component projections on both private and testing features prior to feed them into the neural network. As discussed in Section 4.1, choosing $K-1$ largest eigenvalues is sufficient to improve the robustness of NoisyGD. Therefore, we consider projecting features onto the top $k \in \{10, 50, 100\}$ components. Figure 4 shows that the best utility of NoisyGD is achieved when $k = 10$, aligning with our theoretical findings. Moreover, a larger $k = 100$ fails to improve robustness, likely because the additional 90 principal vectors contribute minimal information and introduce further randomness to the training.

6. Discussions and future work

Most existing theory of DP-learning focuses on suboptimality in surrogate loss of testing data. Our paper studies 0-1 loss directly and observed very different behaviors under perfect and near-perfect neural collapse. In particular, we have $\log(1/\text{error})$ sample complexity rather than $1/\text{error}$ sample complexity. Our theoretical findings shed on light



(a) **Synthetic perfect NC feature** $X \in \mathcal{R}^{n\times d}$ with $K = 10, n = 10^4$ and $(1.0, 10^{-4})$-DP. In the default setting, we assume each class draws $\lfloor n/K \rfloor$ data from a column of K-ETF, the training starts from a zero weight w and the testing data are drawn from the same distribution as X.



(b) **CIFAR-10**: Test features in the perturb setting are subjected to Gaussian noise with a variance of 0.1. The vision transformer produces a 768-dimensional feature for each image. To simulate different feature dimensions, we randomly sample a subset of coordinates or make copies of the entire feature space.

Figure 3. Empirical behaviors of NoisyGD under various robustness setting.

that privacy theorists should look into structures of data and how one can adapt to them. Additionally, our result suggests a number of practical mitigations to make DP-learning more robust in nearly neural collapse settings. It will be useful to investigate whether the same tricks are useful for private learning in general even without neural collapse. Moreover, our results suggest that under neural collapse, choice of loss functions (square loss vs CE loss) do not matter very much for private learning. Square loss has the advantage of having a fixed Hessian independent to the parameter, thus making it easier to adapt to strong convexity parameters like in AdaSSP (Wang, 2018). This is worth exploring.

NoisyGD and NoisySGD theory suggests that one needs $\Omega(n^2)$ time complexity to achieve optimal privacy-utility-trade off in DP-ERM (faster algorithms exist but more complex and they handle only some of the settings). Our results

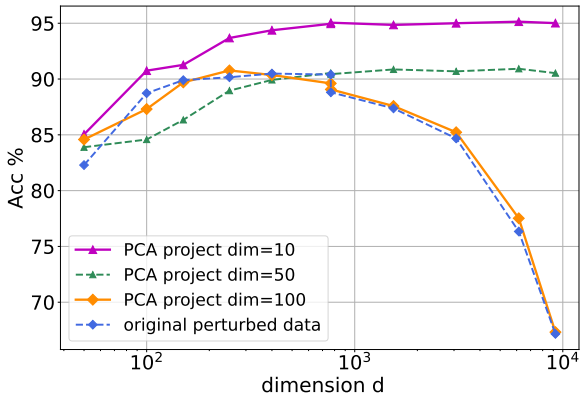


Figure 4. **CIFAR-10.** Apply PCA on both training and testing features before NoisySGD: setting $K-1$ principal components improves NoisySGD's robustness.

on the other hand, suggest that when there are structures in the data, e.g., near-perfect neural collapse, the choice of number of iterations is no longer important, thus making computation easier.

Another perspective in our future study is to investigate when NC will occur in transfer learning. The presence of low-rank structures such as NC in representation learning depends on the pre-trained dataset and the downstream task. Based on our experiments, if the downstream task is CIFAR-10, then ViT performs better than ResNet-50. As NC may not consistently occur, we speculate that the collapse level is more significant when the classes of the downstream task closely resemble those of the pre-trained dataset. For instance, NC may manifest when a model is pre-trained on a broad category such as all animals, and the downstream task involves classifying more specific sub-classes (e.g., different breeds of dogs). This intuition needs further empirical investigation with ample computational resources.

There is another important future topic to consider: what if NC does not occur? When NC cannot be observed, we conjecture that fine-tuning all layers under DP may lead to some low-rank structure of the last layer (potentially inducing some minor collapse phenomenon due to the random noise introduced by DP). Whether NC can be observed after fine-tuning all layers under DP will be validated in our future study.

Acknowledgments

The research was partially supported by NSF Awards No. 2048091 and No. 2134214.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal

consequences of our work, none which we feel must be specifically highlighted here.

References

Abadi, M., Chu, A., Goodfellow, I. J., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In Weippl, E. R., Katzenbeisser, S., Kruegel, C., Myers, A. C., and Halevi, S. (eds.), *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pp. 308–318. ACM, 2016.

Altschuler, J. M. and Talwar, K. Privacy of noisy stochastic gradient descent: More iterations without more privacy loss. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Balle, B., Barthe, G., and Gaboardi, M. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6280–6290, 2018.

Bao, W., Pittaluga, F., b g, V. K., and Bindschaedler, V. DP-mix: Mixup-based data augmentation for differentially private learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Bassily, R., Smith, A. D., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pp. 464–473. IEEE Computer Society, 2014.

Bassily, R., Feldman, V., Talwar, K., and Thakurta, A. G. Private stochastic convex optimization with optimal rates. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 11279–11288, 2019.

Bassily, R., Feldman, V., Guzmán, C., and Talwar, K. Stability of stochastic gradient descent on nonsmooth convex losses. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Bok, J., Su, W., and Altschuler, J. M. Shifted interpolation for differential privacy. *arXiv preprint arXiv:2403.00278*, 2024.

Bu, Z., Dong, J., Long, Q., and Su, W. Deep Learning With Gaussian Differential Privacy. *Harvard Data Science Review*, 2(3), sep 30 2020.

Bu, Z., Wang, Y., Zha, S., and Karypis, G. Differentially private bias-term only fine-tuning of foundation models. *arXiv preprint arXiv:2210.00036*, 2022.

Bun, M. and Steinke, T. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In Hirt, M. and Smith, A. D. (eds.), *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pp. 635–658, 2016.

Bun, M., Nissim, K., Stemmer, U., and Vadhan, S. Differentially private release and learning of threshold functions. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pp. 634–649. IEEE, 2015.

Bun, M., Livni, R., and Moran, S. An equivalence between private classification and online prediction. In Irani, S. (ed.), *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pp. 389–402. IEEE, 2020.

Chaudhuri, K. and Hsu, D. Sample complexity bounds for differentially private learning. In *Proceedings of the 24th Annual Conference on Learning Theory*, pp. 155–186. JMLR Workshop and Conference Proceedings, 2011.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dong, J., Roth, A., and Su, W. J. Gaussian differential privacy. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 84(1): 3–54, 2022. ISSN 1369-7412. With discussions and a reply by the authors.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Dwork, C. Differential privacy. In Bugliesi, M., Preneel, B., Sassone, V., and Wegener, I. (eds.), *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pp. 1–12. Springer, 2006.

Fan, J., Wang, W., and Zhong, Y. An ℓ_{∞} eigenvector perturbation bound and its application. *J. Mach. Learn. Res.*, 18:207:1–207:42, 2017.

Fang, C., He, H., Long, Q., and Su, W. J. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43):e2103091118, 2021.

Feldman, V., Koren, T., and Talwar, K. Private stochastic convex optimization: optimal rates in linear time. In Makarychev, K., Makarychev, Y., Tulsiani, M., Kamath, G., and Chuzhoy, J. (eds.), *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pp. 439–449. ACM, 2020.

Ghalebikesabi, S., Berrada, L., Gowal, S., Ktena, I., Stanforth, R., Hayes, J., De, S., Smith, S. L., Wiles, O., and Balle, B. Differentially private diffusion models generate useful synthetic images. *arXiv preprint arXiv:2302.13861*, 2023.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Commun. ACM*, 63(11): 139–144, oct 2020. doi: 10.1145/3422622.

He, H. and Su, W. J. A law of data separation in deep learning. *Proceedings of the National Academy of Sciences*, 120(36):e2221704120, 2023. doi: 10.1073/pnas.2221704120.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Laurent, B. and Massart, P. Adaptive estimation of a quadratic functional by model selection. *Ann. Statist.*, 28(5):1302–1338, 2000. ISSN 0090-5364. doi: 10.1214/aos/1015957395.

Li, X., Tramer, F., Liang, P., and Hashimoto, T. Large language models can be strong differentially private learners. *arXiv preprint arXiv:2110.05679*, 2021.

Li, X., Liu, D., Hashimoto, T. B., Inan, H. A., Kulkarni, J., Lee, Y.-T., and Guha Thakurta, A. When does differentially private learning not suffer in high dimensions? *Advances in Neural Information Processing Systems*, 35: 28616–28630, 2022.

Li, X., Liu, S., Zhou, J., Fernandez-Granda, C., Zhu, Z., and Qu, Q. What deep representations should we learn? – a neural collapse perspective. *https://openreview.net/forum?id=ZKEhS93FjhR*, 2023a.

Li, X., Wang, C., and Cheng, G. Statistical theory of differentially private marginal-based data synthesis algorithms. In *The Eleventh International Conference on Learning Representations*, 2023b.

Liu, T., Vietri, G., Steinke, T., Ullman, J. R., and Wu, Z. S. Leveraging public data for practical private query release. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6968–6977. PMLR, 2021.

Masarczyk, W., Ostaszewski, M., Imani, E., Pascanu, R., Miłoś, P., and Trzciński, T. The tunnel effect: Building data representations in deep neural networks. *NeurIPS*, 2023.

McKenna, R., Miklau, G., and Sheldon, D. Winning the NIST contest: A scalable and general approach to differentially private synthetic data. *J. Priv. Confidentiality*, 11 (3), 2021.

Nguyen, H. L., Ullman, J. R., and Zakynthinou, L. Efficient private algorithms for learning large-margin halfspaces. In Kontorovich, A. and Neu, G. (eds.), *Algorithmic Learning Theory, ALT 2020, 8-11 February 2020, San Diego, CA, USA*, volume 117 of *Proceedings of Machine Learning Research*, pp. 704–724. PMLR, 2020.

Papyan, V., Han, X. Y., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proc. Natl. Acad. Sci. USA*, 117(40):24652–24663, 2020. ISSN 0027-8424. doi: 10.1073/pnas.2015509117.

Pinto, F., Hu, Y., Yang, F., and Sanyal, A. PILLAR: how to make semi-private learning more effective. In *IEEE Conference on Secure and Trustworthy Machine Learning, SaTML 2024, Toronto, ON, Canada, April 9-11, 2024*, pp. 110–139. IEEE, 2024.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. Improving language understanding by generative pre-training. *OpenAI*, 2018.

Song, S., Chaudhuri, K., and Sarwate, A. D. Stochastic gradient descent with differentially private updates. In *2013 IEEE global conference on signal and information processing*, pp. 245–248. IEEE, 2013.

Sutskever, I., Martens, J., Dahl, G. E., and Hinton, G. E. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 1139–1147. JMLR.org, 2013.

Tramer, F. and Boneh, D. Differentially private learning needs better features (or much more data). *arXiv preprint arXiv:2011.11660*, 2020.

Wang, C., Su, B., Ye, J., Shokri, R., and Su, W. J. Unified enhancement of privacy bounds for mixture mechanisms via f-differential privacy. In *NeurIPS*, 2023.

Wang, H., Gao, S., Zhang, H., Shen, M., and Su, W. J. Analytical composition of differential privacy via the edgeworth accountant, 2022.

Wang, Y. Revisiting differentially private linear regression: optimal and adaptive prediction & estimation in unbounded domain. In Globerson, A. and Silva, R. (eds.), *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pp. 93–103. AUAI Press, 2018.

Wang, Y., Lei, J., and Fienberg, S. E. Learning with differential privacy: stability, learnability and the sufficiency and necessity of ERM principle. *J. Mach. Learn. Res.*, 17:Paper No. 183, 40, 2016. ISSN 1532-4435.

Wang, Y., Balle, B., and Kasiviswanathan, S. P. Subsampled Rényi differential privacy and analytical moments accountant. *J. Priv. Confidentiality*, 10(2), 2020.

Ye, J. and Shokri, R. Differentially private learning needs hidden state (or much faster convergence). In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Ye, J., Zhu, Z., Liu, F., Shokri, R., and Cevher, V. Initialization matters: Privacy-utility analysis of overparameterized neural networks. In *NeurIPS*, 2023.

Yu, D., Naik, S., Backurs, A., Gopi, S., Inan, H. A., Kamath, G., Kulkarni, J., Lee, Y. T., Manoel, A., Wutschitz, L., et al. Differentially private fine-tuning of language models. *arXiv preprint arXiv:2110.06500*, 2021.

Zhu, Y. and Wang, Y. Poission subsampled Rényi differential privacy. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7634–7642. PMLR, 2019.

Zhu, Y., Dong, J., and Wang, Y. Optimal accounting of differential privacy via characteristic function. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event*, volume 151 of *Proceedings of Machine Learning Research*, pp. 4782–4817. PMLR, 2022.

A. Evaluate the feature shift vector

In this section, we empirically investigate the feature shift parameter β if the pre-trained transformer is the Vision Transformer (ViT) or ResNet-50 and the fine-tuned dataset is CIFAR-10. The results are displayed in Figure 2 in the Introduction section.

Recall the feature shift parameter $\beta = \|x - M_k\|_\infty$ that is the ℓ_∞ distance between the feature x and the perfect feature feature mean M_k of the class k. The perfect feature M_k is unknown, thus, we cannot compute the feature shift parameter $\beta = \|x - M_k\|_\infty$ exactly. However, we can approximate M_k by the empirical feature mean $\widehat{M}_k$, which is the average of all features in the k-th class. We use the following steps to evaluate β numerically.

- If $\cos(\widehat{M}_i, \widehat{M}_j) \approx \frac{1}{K-1}$ for all $1 \le i, j \le K$, then we may claim that NC happens and $\widehat{M}_k$ can be regarded as the perfect feature (as implied by the maximal-equiangularity in equation 2).
- Suppose we have n data points with features $\{x_i\}_{i=1}^n$. For x_i in the k-th class, we empirically calculate $\|x_i - \widehat{M}_k\|_\infty$ and plot their distribution.

To instantiate various β distributions, we consider two ImageNet pre-trained models (ResNet-50 and the vision transformer (ViT) model). We apply two models to extract training features, applying standard feature normalizations ($||x||_2 = 1$) and evaluate their empirical β across all training samples.

Specifically, we first calculate the training feature mean per class $\widehat{M}_i$ for $i \in [K]$ and evaluate the cosine matrix of $\cos(\widehat{M}_i, \widehat{M}_j)$ for $1 \le i, j \le K$. We found that all entries of the cosine matrix are close to $1/(K-1)$ (roughly all entries between $[-0.2, 0.2]$ and the cosine median is 0.082 for the ViT model and 0.112 for the ResNet-50 model). Therefore, we next evaluate $||x - \widehat{M}_k||_\infty$ for all training samples.

B. Additional preliminaries

Definition B.1 (Sample complexity for private $\mathfrak{D}$-learnability)**.** For a set of distributions $\mathfrak{D}$, the sample complexity of private $\mathfrak{D}$-learning a hypothesis class $\mathcal{H}$ under ρ-zCDP is defined to be

$$n_{\mathcal{H},\mathfrak{D},\rho}(\gamma) = \min\left\{ n \in \mathbb{N} \;\middle|\; \inf_{\mathcal{A} \text{ satisfies } \rho-\text{zCDP}} \sup_{\mathcal{P}\in\mathfrak{D}} \mathbb{E}_{\mathcal{A},\text{data}\sim\mathcal{P}^n}\left[\text{Err}_{\mathcal{P}}(\mathcal{A}(\text{data})) - \inf_{h\in\mathcal{H}} \text{Err}_{\mathcal{P}}(h)\right] \le \gamma \right\}$$

i.e., the smallest integer such that the minimax excess risk is smaller than γ.

We say $\mathfrak{D}$ is learnable under ρ-zCDP if $n_{\mathcal{H},\mathfrak{D},\rho}$ is finite, meaning there exists an algorithm $\mathcal{A}$ such that the expected excess risk is smaller than γ with finitely many training data. It is obvious that the misclassification error defined above can be written as $\mathbb{E}_{\mathcal{A},\text{data}\sim\mathcal{P}^n}[\text{Err}_{\mathcal{P}}(\mathcal{A}(\text{data}))]$ as the randomness of $\widehat{W}$ comes from the randomized algorithm $\mathcal{A}$ and the training data. Since the second term $\inf_{h\in\mathcal{H}} \text{Err}_{\mathcal{P}}(h)$, which is the misclassification error of the optimal classifier, is non-negative, the misclassification error is always larger than the excess risk. Thus, in Section 3, we investigate the convergence of the misclassification error, which implies an upper bound on the excess risk as well.

NC for binary classification. For the binary classification problem, we have $M_1 = \frac{P_1 - P_2}{\sqrt{2}}$, according to the definition of an ETF in Eq. 2. Similarly, it holds that $M_2 = \frac{P_2 - P_1}{\sqrt{2}}$. Thus, we have $M_1 = -M_2$ and $\|M_1\|_2 = \|M_2\|_2 = 1$. Since our theory depends only on the norm of the feature means M_1 and M_2, and the angle between M_1 and M_2, while the rotation matrix P will not change the misclassification error, without loss of generality, we assume that $M_1 = e_1$ and $M_2 = -e_1$.

C. Additional content about sample complexity in Section 3.1

In this section, we provide additional results for sample complexity in Section 3.1. We first provide a full version of the sample complexity table in Table 2. Then, we discussed the extension of our results to multiple iterations in Section C.4. We also provide the proofs of Section 3.1. Without loss of generality, we let $v_i^T e_1 = 0$. Otherwise, by the orthogonal decomposition, there is a constant $c > 0$ and a shift vector v_i such that $y_i x_i = c(e_1 + v_i)$ and it is obvious that a scalar c will not change the misclassification error based on our proof details.

Setting	Assumption	Sample complexity
Non-private learning with GD	Perfect NC or β-approximate NC with $\beta^2 p \leq 1$	No. of classes
	β-approximate NC	$p\beta^2 \log \frac{1}{\gamma}$
	β-approximate NC (separability)	$p\beta^4 \log \frac{1}{\gamma}$
Private learning (ρ-zCDP) with NoisyGD	Perfect NC	$\frac{2\sqrt{\log(1/\gamma)}}{\sqrt{\rho}}$
	β-approximate NC	$p\beta^2 \log \frac{1}{\gamma} + \frac{\max\{p\beta^2,1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
	β-approximate NC (separability)	$p\beta^4 \log \frac{1}{\gamma} + \frac{\max\{p\beta^2,1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
	$\widetilde{\beta}$-stochastic perturbation (test)	$\frac{\max\{\sqrt{p}\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
	$\widetilde{\beta}$-adversarial perturbation (test)	$\frac{\max\{p\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
	$\widetilde{\beta}$-offset perturbation (training)	$\frac{\max\{\sqrt{p}\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{\sqrt{2\rho}}$
	$\widetilde{\beta}$-offset perturbation + α-Class imbalance	$\frac{\max\{\sqrt{p}\widetilde{\beta},1\}\sqrt{\log(1/\gamma)}}{(1-\widetilde{\beta}+2\widetilde{\beta}\alpha)\sqrt{2\rho}}$

Table 2. Summary of the sample complexity of achieving a misclassification error γ of private learning under ρ-zCDP. We consider perfect features, actual features (GD and NoisyGD), and perturbed features (stochastic, adversarial, offset perturbations). For the actual features, we assume that the feature shift vectors of traing and testing features are from the same distribution. The separability assumption here is that all p components of the feature shift vectors are independent. We have also considered the effects of α-class imbalance.

C.1. Proof of Theorem 3.1

The output of linear GD without DP guarantee is given by

$$\widehat{\theta} = \eta \sum_{i=1}^{n} y_i x_i.$$

For a testing data point (x, y), without loss of generality, we consider $y = 1$ and $x = e_1 + v$ for some vector v. When v is fixed, then, the misclassification error is 0 if $1 - \beta^2 p > 0$.

For v_i being symmetric i.i.d. random vectors, since $v_i^T e_1 = 0$ and $v^T e_1 = 0$, we have the misclassification error is given by

$$\Pr_{v,v_i}\left[-\left(ne_1 - \sum_{i=1}^{n} v_i\right)^T (e_1 + v) > 0\right] \leq \Pr\left[\sum_{i=1}^{n} v_i^T v > n\right].$$

Since $|v_i^T v| \leq \beta^2 p$ and $\mathbb{E}[(v_i^T v)^2] \leq \beta^4 p^2$, by a Bernstein-type inequality, we have

$$\Pr\left[\sum_{i=1}^{n} v_i^T v > n\right] = \mathbb{E}_v \Pr\left[\sum_{i=1}^{n} v_i^T v > n \middle| v\right] \leq \exp\left(-\frac{n^2}{2\left(\beta^4 np^2 + \frac{1}{3}\beta^2 pn\right)}\right).$$

Rewrite $v_i = (v_i^j)_{j=1}^p$ and $v = (v^j)_{j=1}^p$ for $v_i^j, v^j \in [-\beta, \beta]$. Note that $|v_i^j v^j| \leq \beta^2$ and $\mathbb{E}[(v_i^j v^j)^2 | v^j] \leq \beta^4$. If we further assume that and $v_i^j, 1 \leq i \leq n, 1 \leq j \leq p$ are independent random variables, then , by a Bernstein-type inequality, we have

$$\begin{aligned}\Pr\left[\sum_{i=1}^{n} v_i^T v > n\right] &= \mathbb{E}_v \Pr\left[\sum_{i=1}^{n} v_i^T v > n \middle| v\right] = \mathbb{E}_v \Pr\left[\sum_{i=1}^{n}\sum_{j=1}^{p} v_i^j v^j > n \middle| v\right] \\ &\leq \exp\left(-\frac{n^2}{2\left(\beta^4 np + \frac{1}{3}\beta^2 n\right)}\right).\end{aligned}$$

C.2. Proof of Theorem 3.2

The output of NoisyGD is given by

$$\widehat{\theta}_{\text{NoisyGD}} = \eta \left(\sum_{i=1}^{n} y_i x_i + \mathcal{N}(0, \sigma^2 I) \right).$$

For a testing data $x = e_1 + v$, we have

$$y\widehat{\theta}_{\text{NoisyGD}}^T x = \mu_n + \mathcal{N}(0, \eta^2 \|x\|_2^2 \sigma^2),$$

where $\mu_n = \eta(n + \sum_{i=1}^n v_i^T v)$.

By the law of total probability, we have

$$\begin{aligned}
\Pr[y\widehat{\theta}_{\text{NoisyGD}}^T x < 0] &= \Pr\left[\mu_n + \mathcal{N}(0, \eta^2 \|x\|_2^2 \sigma^2) < 0\right] \\
&= \Pr\left[\mu_n + \mathcal{N}(0, \eta^2 \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \cdot \Pr\left[\sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \\
&\quad + \Pr\left[\mu_n + \mathcal{N}(0, \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v > -\frac{n}{2}\right] \cdot \Pr\left[\sum_{i=1}^n v_i^T v > -\frac{n}{2}\right].
\end{aligned}$$

For the first term, if $\sum_{i=1}^n v_i^T v > -\frac{n}{2}$, we have $\mu_n \ge \frac{\eta n}{2}$. As a result, it holds

$$\begin{aligned}
\Pr\left[\mu_n + \mathcal{N}(0, \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v > -\frac{n}{2}\right] &\le \Pr\left[\mathcal{N}(0, \|x\|_2^2 \sigma^2) < -\frac{n}{2} \middle| \sum_{i=1}^n v_i^T v > -\frac{n}{2}\right] \\
&\le \exp\left(-\frac{n^2}{4\|x\|_2^2 \sigma^2}\right) \le \exp\left(-\frac{n^2}{4(1+\beta^2 p)\sigma^2}\right).
\end{aligned}$$

Thus, we have

$$\Pr\left[\mu_n + \mathcal{N}(0, \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v > -\frac{n}{2}\right] \cdot \Pr\left[\sum_{i=1}^n v_i^T v > -\frac{n}{2}\right] \le \exp\left(-\frac{n^2}{4(1+\beta^2 p)\sigma^2}\right).$$

For the second term, similarly to the proof in Section C.1, we have

$$\Pr\left[\sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \le \exp\left(-\frac{n^2}{2\left(\beta^4 np^2 + \frac{1}{3}\beta^2 np\right)}\right),$$

or, under i.i.d. assumptions on the components of v, we have

$$\Pr\left[\sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \le \exp\left(\frac{-n^2}{2\left(\beta^4 np + \frac{1}{3}\beta^2 n\right)}\right).$$

As a result, it holds

$$\Pr\left[\mu_n + \mathcal{N}(0, \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \cdot \Pr\left[\sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \le \exp\left(-\frac{n^2}{2\left(\beta^4 np^2 + \frac{1}{3}\beta^2 np\right)}\right),$$

or, under further i.i.d. assumptions on the components of v, it holds

$$\Pr\left[\mu_n + \mathcal{N}(0, \|x\|_2^2 \sigma^2) < 0 \middle| \sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \cdot \Pr\left[\sum_{i=1}^n v_i^T v \le -\frac{n}{2}\right] \le \exp\left(-\frac{n^2}{2\left(\beta^4 np + \frac{1}{3}\beta^2 n\right)}\right).$$

Over all, we obtain

$$\Pr[y\widehat{\theta}_{\text{NoisyGD}}^T x < 0] \leq \exp\left(-\frac{n^2}{4(1+\beta^2 p)\sigma^2}\right) + \exp\left(-\frac{-n^2}{2\left(\beta^4 np^2 + \frac{1}{3}\beta^2 np\right)}\right),$$

or, under further i.i.d. assumptions on all p components of v, we obtain

$$\Pr[y\widehat{\theta}_{\text{NoisyGD}}^T x < 0] \leq \exp\left(-\frac{n^2}{4(1+\beta^2 p)\sigma^2}\right) + \exp\left(-\frac{-n^2}{2\left(\beta^4 np + \frac{1}{3}\beta^2 n\right)}\right).$$

We finish the proof by noting that $\sigma^2 = (1+\beta^2 p)/2\rho$.

Let the misclassification error less than γ and we have

$$n = O\left(\frac{(1+\beta^2 p)^2 \log\frac{1}{\gamma}}{2\rho} + 4p\beta^2 \log\frac{1}{\gamma}\right),$$

or, under further i.i.d. assumptions on all p components of v, we have

$$n = O\left(\frac{(1+\beta^2 p)^2 \log\frac{1}{\gamma}}{2\rho} + 4p\beta^4 \log\frac{1}{\gamma}\right).$$

C.3. Proof of Theorem 3.3

The output of NoisyGD is given by

$$\widehat{\theta}_{\text{NoisyGD}} = \eta\left(\sum_{i=1}^{n} y_i x_i + \mathcal{N}(0, \sigma^2 I)\right).$$

For a testing data $x = e_1 + v$, we have

$$y\widehat{\theta}_{\text{NoisyGD}}^T x = \eta(\mu_n + (e_1+v)^T \xi),$$

where $\mu_n = n + \sum_{i=1}^{n} v_i^T v$ and $\xi \sim \mathcal{N}(0, \sigma^2 I)$.

Since $\beta^2 p < 1$, we have

$$\mu_n \geq n(1-\beta^2 p) > 0.$$

Thus, we have

$$\begin{aligned}\Pr\left[y\widehat{\theta}_{\text{NoisyGD}}^T x < 0\right] &\leq \Pr\left[\mathcal{N}(0, \|x\|_2^2\sigma^2) > n(1-\beta^2 p)\right] \\ &\leq \exp\left(-\frac{n^2(1-\beta^2 p)^2}{(1+\beta^2 p)\sigma^2}\right).\end{aligned}$$

C.4. Multiple iterations

For the multi-iteration case, we consider the projected NoisyGD to bound the parameters. Precisely, the output is defined iteratively as

$$\theta_{k+1} = \mathcal{P}_{B_2^p(0,R)}\left(\theta_k - \eta\left(g_n(\theta_k) + \xi_k\right)\right), \tag{4}$$

where $B_2^p(0,R) \subset \mathbb{R}^p$ is an ℓ_2-norm ball with radius R, $\xi_k \sim \mathcal{N}(0, \sigma^2 I_p)$, and $\mathcal{P}_A$ is the projection onto a convex set A w.r.t. the Euclidean inner product. Here we take $\sigma^2 = (1+\beta^2 p)/2\rho$ and the overall privacy budget is $k\rho$.

Theorem C.1 (Multiple iterations). *Let θ_{k+1} be the output of projected NoisyGD defined in equation 4. For any $t > 0$, if we take $\eta = \frac{R}{n(1+\beta^2 p)+(p+\sqrt{pt}+t)}$, then, the misclassification error is*

$$\Pr\left[\theta_{k+1}^T(e_1+v) < 0\right] \leq \exp\left(-\frac{n^2}{C_{p,k}^2\sigma^2(1+\beta^2 p)}\right) + ke^{-t},$$

where $C_{p,k} = \frac{1+2^{-k}}{1-2^{-k}} \cdot \frac{1-1/2}{1+1/2} \cdot \frac{\left(1+e^{R(1+\beta^2 p)}\right)^2}{(1-\beta^2 p)^2}$ and $\sigma^2 = (1+\beta^2 p)/2\rho$. Specifically, for $t = n^2 + \log 1/k$, we have $\Pr\left[\theta_{k+1}^T(e_1+v) < 0\right] \leq O\left(e^{-\frac{\rho n^2}{(1+\beta^2 p)^2}}\right)$.

Remark. To make the projected NoisyGD converge exponentially, we still require $\beta^2 < \frac{1}{p}$.

Proof of Theorem C.1. Recall the loss function $\ell(\theta,(x,y)) = \log(1+e^{-y\cdot\theta^T x})$. The gradient is then given by

$$g(\theta,(x,y)) = \frac{e^{-y\cdot\theta^T x}}{1+e^{-y\cdot\theta^T x}}(-yx).$$

Note that $yx = e_1 + v$ for $y = 1$ and $yx = e_1 - v$ for $y = -1$. For $\alpha = 1/2$, we have

$$-g_n(\theta) := \nabla\mathcal{L}(\theta) = \frac{n}{2}\left[\frac{1}{1+e^{\theta^T(e_1+v)}}(e_1+v) + \frac{1}{1+e^{\theta^T(e_1-v)}}(e_1-v)\right].$$

and

$$0 \leq -g_n(\theta)^T e_1 \leq \frac{n(1-\beta)}{2}$$

for any $\theta \in \mathbb{R}^p$. To make the multi-iterations work, we need lower bound $-g_n(\theta)^T e_1$ by a positive constant. Now we consider the two-iteration GD.

$$\theta_2 = \theta_1 - \eta(g_n(\theta_1) + \xi_1), \qquad \xi_1 \sim \mathcal{N}(0,\sigma^2 I_p).$$

Here $\theta_1 = \frac{n}{2}e_1 + \xi_0$. Then, it holds

$$g_n(\theta_1) = \frac{n}{2}\left[\frac{e_1+v}{1+e^{n/2+\xi_{0,1}}} + \frac{e_1-v}{1+e^{n/2-\xi_{0,1}}}\right]$$

To lower bound the gradient, we consider the projected iteration defined as

$$\theta_{k+1} = \mathcal{P}_{B_2^p(0,R)}\left(\theta_k - \eta\left(g_n(\theta_k) + \xi_k\right)\right),$$

where $B_2^p(0,R) \subset \mathbb{R}^p$ is an ℓ_2-norm ball with radius R, $\xi_k \sim \mathcal{N}(0,\sigma^2)$, and $\mathcal{P}_A$ is the projection onto a convex set A w.r.t. the Euclidean inner product.

Then we have

$$\theta_{k+1}^T(e_1+v) = c_{R,k}\left(\theta_k - \eta\left(g_n(\theta_k)+\xi_k\right)\right)^T(e_1+v),$$

where $c_{R,k} = \frac{\min\{R,\|\theta_k - \eta(g_n(\theta_k)+\xi_k)\|_2\}}{\|\theta_k-\eta(g_n(\theta_k)+\xi_k)\|_2}$. Since

$$\eta\left\|(g_n(\theta_k)+\xi_k)\right\|_2 \leq \eta n(1+\beta^2 p) + \eta\|\xi_k\|_2,$$

it is enough to bound $\|\xi_k\|_2$. Note that $\|\xi_k\|_2^2$ is a $\chi^2(p)$ distribution and one (cf., Laurent & Massart (2000)) has the tail bound

$$\Pr\left[\eta^2\|\xi_k\|_2^2 > \eta^2(p+\sqrt{pt}+t)\right] \leq e^{-t}$$

for any $t > 0$. Thus, by the union bound, we have

$$C_{R,k} \geq C_{R,p,t,\eta,n} = \frac{R}{R + \eta n(1+\beta^2 p) + \eta(p + \sqrt{pt} + t)^{1/2}}$$

for any $k \geq 0$ with probability ke^{-t}. Since $-g_n(\theta_k)^T(e_1 + v) \geq n\frac{1-\beta^2 p}{1+e^{R(1+\beta^2 p)}}$, it holds

$$\begin{aligned}\theta_{k+1}^T(e_1+v) &\geq C_{R,p,t,\eta,n}\left(\theta_k^T(e_1+v) + n\eta\frac{1-\beta^2 p}{1+e^{R(1+\beta^2 p)}} - \eta\xi_k^T(e_1+v)\right) \\ &\geq n\eta\frac{1-\beta^2 p}{1+e^{R(1+\beta^2 p)}}\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{k-j+1} + \eta\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{k-j+1}\xi_j^T(e_1+v).\end{aligned}$$

Since $\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{k-j+1}\xi_j^T(e_1+v) \sim \mathcal{N}(0, \sum_{j=1}^{k} C_{R,p,t,\eta,n}^{2(k-j+1)}\sigma^2\|e_1+v\|_2^2)$, we have

$$\Pr\left[\theta_{k+1}^T(e_1+v) < 0\right] \leq \Pr\left[\mathcal{N}\left(n\frac{1-\beta^2 p}{1+e^{R(1+\beta^2 p)}}, \frac{\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{2(k-j+1)}}{\left(\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{k-j+1}\right)^2}\sigma^2(1+\beta^2 p)\right) < 0\right] + ke^{-t}.$$

Since $\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{2(k-j+1)} = \frac{C_{R,p,t,\eta,n}^2(1-C_{R,p,t,\eta,n}^{2k})}{1-C_{R,p,t,\eta,n}^2}$ and $\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{(k-j+1)} = \frac{C_{R,p,t,\eta,n}(1-C_{R,p,t,\eta,n}^{k})}{1-C_{R,p,t,\eta,n}}$, we have

$$\frac{\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{2(k-j+1)}}{\left(\sum_{j=1}^{k} C_{R,p,t,\eta,n}^{(k-j+1)}\right)^2} = \frac{1+C_{R,p,t,\eta,n}^{k}}{1-C_{R,p,t,\eta,n}^{k}} \cdot \frac{1-C_{R,p,t,\eta,n}}{1+C_{R,p,t,\eta,n}}.$$

By taking $\eta = \frac{R}{n(1+\beta^2 p)+(p+\sqrt{pt}+t)}$, we get $C_{R,p,t,\eta,n} = \frac{1}{2}$ and

$$\frac{1+C_{R,p,t,\eta,n}^{k}}{1-C_{R,p,t,\eta,n}^{k}} \cdot \frac{1-C_{R,p,t,\eta,n}}{1+C_{R,p,t,\eta,n}} = \frac{1+2^{-k}}{1-2^{-k}} \cdot \frac{1-1/2}{1+1/2} =: C_k.$$

Thus, it holds

$$\Pr\left[\theta_{k+1}^T(e_1+v) < 0\right] \leq \exp\left(-\frac{n^2}{C_{p,k}^2\sigma^2(1+\beta^2 p)}\right) + ke^{-t}$$

with $C_{p,k}^2 = C_k\left(1+e^{R(1+\beta^2 p)}\right)^2/(1-\beta^2 p)^2$. □

D. Results for perturbing only the training data

D.1. Fixed perturbation

Without loss of generality, we assume $0 < \alpha < 1/2$. Consider the class imbalanced case with $n_{-1} = \alpha n$ and $n_{+1} = (1-\alpha)n$. The gradient for $\theta_0 = 0$ is

$$\nabla\mathcal{L}(\theta_0) = \alpha n \cdot 0.5 \cdot -(-e_1 + v) + (1-\alpha)n \cdot 0.5 \cdot (e_1 + v) = \frac{n}{2}e_1 + \frac{(1-2\alpha)n}{2}v.$$

Thus, the output is

$$\widehat{\theta} = -\eta\left(\frac{n}{2}e_1 + \frac{(1-2\alpha)n}{2}v + \mathcal{N}(0,\sigma^2)\right)$$

The sensitivity is $G = \sqrt{1+\|v\|_2^2}$ and σ^2 is taken to be $G^2/2\rho$ to achieve ρ-zCDP. Moreover, we have

$$\widehat{\theta}^T e_1 = -\frac{n}{2} - \frac{(1-2\alpha)n}{2}v_1 + \mathcal{N}(0,\sigma^2).$$

Thus, the misclassification error is

$$\Pr[\widehat{\theta} e_1 > 0] = \Phi\left(\frac{n\left[1-(1-2\alpha)\, v_1\right]}{2\sigma}\right) \leq e^{-\frac{n^2(1-\beta+2\alpha\beta)^2\rho}{4G^2}}.$$

As a result, the sample complexity to achieve $1-\gamma$ accuracy is

$$n = O\left(\sqrt{\frac{4G^2 \log\frac{1}{\delta}}{(1-\beta+2\beta\alpha)^2 \cdot \rho}}\right)$$

The sensitivity $G = \sqrt{1+\beta^2 p}$ here is dimension-dependent.

D.2. Random perturbation

Now we consider the random perturbation. Denote $\{v_i\}_{i=1}^n \subseteq \mathbb{R}^p$ a sequence of i.i.d. copies of a random vector v. Consider the binary classification problem with training set $\{(x_i, y_i)\}_{i=1}^n$. Here $x_i = e_1 + v_i$ if $y_i = 1$ and $x_i = -e_1 + v_i$ if $y_i = -1$. Then, the loss function is $\mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^n \log\left(1 + e^{-y_i \theta^T x_i}\right)$. The one-step iterate of DP-GD from 0 outputs

$$\widehat{\theta} = -\eta \sum_{i=1}^n (-y_i x_i) + \mathcal{N}(0, \sigma^2 I_p)$$

with $\sigma^2 = G^2/2\rho$ and $G = \sup_{v_i} \sqrt{1+\|v_i\|^2}$. Assume that v_i is symmetric, that is $y_i v_i$ has the same distribution as $-y_i v_i$. Then, it holds

$$\sum_{i=1}^n y_i x_i = n e_1 + \sum_{i=1}^n v_i =: \mu_n.$$

The misclassification error is now given by

$$\Pr[\widehat{\theta}^T e_1 < 0] = \Pr[\mathcal{N}(\mu_n^T e_1, \sigma^2) < 0].$$

Assume that $\|v_i\|_\infty = \beta < 1$. Then, we have $\mu_n^T e_1 \geq n - \beta n$ and the sample complexity is $O\left(\sqrt{\frac{4G^2\log(1/\delta)}{(1-\beta)^2\rho}}\right)$ with $G = \sqrt{1+\beta^2 p}$.

E. Results for perturbing only the testing data

For simplicity, we only consider perturbing the perfect feature. Our results can be extended naturally to the case of actual features.

E.1. Fixed (offset) perturbation

Recall that the output of DP-GD has the form $\widehat{\theta} = \mathcal{N}(-\frac{\eta n}{2}, \sigma^2)$. One has

$$\hat{\theta}^T(e+v) = \frac{n}{2} + \mathcal{N}\left(0, \frac{G^2(p\beta^2+1)}{2\rho}\right).$$

The sample complexity can be derived similarly as previous sections, which is dimension dependent.

E.2. Adversarial perturbation

Let's say in prediction time, the input data point can be perturbed by a small value in ℓ_∞. If we allow the perturbation to be adversarial chosen, then there exits v satisfying $\|v\|_\infty \leq \beta$ such that

$$\hat{\theta}^T(x+v) = \frac{n}{2} + \frac{G}{\sqrt{2\rho}} Z_1 - \sum_{i=1}^p |Z_i| \frac{G\beta}{\sqrt{2\rho}},$$

where $Z_1, ..., Z_p \sim \mathcal{N}(0,1)$ i.i.d. Note that the additional term scales as $O(p\frac{G\beta}{\sqrt{\rho}})$, which can alter the prediction if $p \asymp n$ even if ρ is a constant (weak privacy).

The number of data points needed to achieve $1-\delta$ robust classification under neural collapse is $O\left(\frac{G\max\{p\beta,1\}\sqrt{\log(1/\delta)}}{\sqrt{2\rho}}\right)$.

F. Proofs of Section F.1

F.1. Omitted details for perfect features

In this section, we explore the theoretical scenario of an ideal, perfect neural collapse, where β equals to zero. Although achieving a perfect neural collapse is practically elusive, examining this idealized case is beneficial due to the promising properties it exhibits. However, as we have noted in the previous sections, these advantageous properties are highly fragile to any perturbations in the features. The details of this section is given in Appendix F.

Consider the multi-class classification problem with K classes. Under perfect neural collapse, we have $\Pr[x = M_k | y_k = 1] = 1$. Let $\widehat{\theta} \in \mathbb{R}^{Kp}$ be the 1-step output of the NoisyGD algorithm defined in Equation (1) with 0-initialization and let $\widehat{W} \in \mathbb{R}^{K\times p}$ be the matrix derived from $\widehat{\theta}$. Denote $\widehat{y} = \mathrm{OneHot}(\widehat{W}x) \in \{0,1\}^K$ the predictor after the one-hot encoding, that is $\widehat{y}_i = 1$ if $i = \arg\max_j\{(\widehat{W}x)_j\}$, otherwise $\widehat{y}_i = 0$.

Theorem F.1. *Let $\widehat{y}$ be a predictor trained by NoisyGD under the cross entropy loss with zero initialization. Assume that the training dataset is balanced, that is, the sample size of each class is n/K. For classification problems under neural collapse with K classes, if we take $G \geq 1$, then the misclassification error is*

$$\begin{aligned}\Pr[\widehat{y} \neq y] &= (K-1)\Phi\left(-\frac{n}{K\sigma}\left(1+\frac{K-2}{K(K-1)}\right)\right)\\ &\leq (K-1)e^{-\frac{C_K n^2}{K\sigma^2}}\end{aligned}$$

with $\sigma^2 = \frac{G^2}{2\rho}$ and $C_K = \left(1+\frac{K-2}{K(K-1)}\right)^2$. As a result, to make the misclassification error less than γ, the sample complexity for n is $\frac{GK\sqrt{\log((K-1)/\gamma)}}{C_K\sqrt{2\rho}}$.

The theorem offers several insights, we have

1. The error bound is exponentially close to 0 if $\rho \gg G^2/n^2$ — very strong privacy and very strong utility at the same time.
2. The result is dimension independent — it doesn't depend on the dimension p.
3. Even though here we assume that the training dataset is class-balanced, the result is robust to class imbalance for $K = 2$, if we apply a re-parameterization of private data (see, Section 3.1).
4. The result is independent of the shape of the loss functions. Logistic loss works, while square losses also works.
5. The result does not require careful choice of learning rate. Any learning rate works equally well.

Our theory can be extended to the domain adaptation context, where the model is initially pre-trained on an extensive dataset with K_0 classes, and is subsequently fine-tuned for a downstream task with a smaller number of classes $K \leq K_0$. One may refer to Appendix F.4.

F.2. Proof of Theorem F.1 and corresponding results

Recall an ETF defined by

$$M = \sqrt{\frac{K}{K-1}}P\left(I_K - \frac{1}{K}\mathbf{1}_K\mathbf{1}_K^T\right) = \sqrt{\frac{K}{K-1}}\left(P - \frac{1}{K}\sum_{k=1}^{K}P_k\mathbf{1}_K^T\right),$$

where $P = [P_1, \cdots, P_K] \in \mathbb{R}^{p\times K}$ is a partial orthogonal matrix with $P^TP = I_K$. Rewrite $M = [M_1, \cdots, M_K]$. Let the label $y = (y_1, \cdots, y_K)^T \in \{0,1\}^K$ be represented by the one-hot encoding, that is, $y_k = 1$ and $y_j = 0$ for $j \neq k$ if y belongs to the k-th class.

Definition F.2 (Classification problem under Neural Collapse)**.** Let there be K classes. The distribution $\mathbb{P}[x = M_k|y_k = 1] = 1$ for $k = 1, ..., K$.

Proof of Theorem F.1. Let $W = [W_1, \cdots, W_K]^T \in \mathbb{R}^{K\times p}$. Consider the output function $f_W(x) = Wx \in \mathbb{R}^K$. Suppost that $y_k = 1$. Then, the cross-entropy loss is defined by

$$\ell(f_W(x), y) = -\log\left(\frac{e^{W_k^T x}}{\sum_{k'=1}^K e^{W_{k'}^T x}}\right).$$

The corresponding empirical risk is

$$R_n(M, W) = \sum_{k=1}^K -n_k \log\left(\frac{e^{W_k^T M_k}}{\sum_{k'=1}^K e^{W_{k'}^T M_k}}\right).$$

Note that

$$\nabla_W \ell(f_W(x), y) = (\mathrm{SoftMax}(f_W(x)) - y)\, x^T,$$

where $\mathrm{SoftMax} : \mathbb{R}^K \to \mathbb{R}^K$ is the SoftMax function defined by

$$\mathrm{SoftMax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \qquad \text{for all } z \in \mathbb{R}^K.$$

We obtain

$$\nabla_W R_n(M, W) = \sum_{k=1}^K n_k \left(\mathrm{SoftMax}(f_W(M_k)) - y^k\right) M_k^T,$$

where y^k is the label of the k-th class. For zero initialization, we have

$$\mathrm{SoftMax}(f_{\mathbf{0}}(M_k)) = \frac{1}{K}\mathbf{1}_K$$

and

$$\nabla_W (R_n(M, W))\Big|_{W=\mathbf{0}} = \sum_{k=1}^K n_k \left(\frac{1}{K}\mathbf{1}_K - y^k\right) M_k^T. \tag{5}$$

Now we consider one step NoisyGD from 0 with learning rate $\eta = 1$:

$$\widehat{W} = -\sum_{k=1}^K n_k \left(\frac{1}{K}\mathbf{1}_K - y^k\right) M_k^T + \Xi,$$

where $\Xi \in \mathbb{R}^{K\times p}$ with Ξ_{ij} drawn independently from a normal distribution $\mathcal{N}(0, \sigma^2)$.

Consider $x = M_k$. It holds

$$f_{\widehat{W}}(x) = \widehat{W}M_k = -\sum_{k'=1}^K n_{k'} \left(\frac{1}{K}\mathbf{1}_K - y^{k'}\right) M_{k'}^T M_k + \Xi M_k.$$

Since

$$\Xi M_k \sim \mathcal{N}\left(0, \sigma^2\|M_k\|_2^2 I_K\right) \qquad \text{and} \qquad \|M_k\|_2^2 = 1,$$

we have

$$\widehat{W} M_k \sim \mathcal{N}\left(\boldsymbol{\mu}_{n,K}, \sigma^2 I_K\right),$$

where $\boldsymbol{\mu}_{n,K} = -\sum_{k'=1}^{K} n_{k'} \left(\frac{1}{K}\mathbf{1}_K - y^{k'}\right) M_{k'}^T M_k$. Note that

$$M_{k'}^T M_k = \frac{K}{K-1}\left(\delta_{k,k'} - \frac{1}{K}\right).$$

We obtain

$$\left(\boldsymbol{\mu}_{n,K}\right)_j = \begin{cases} n/K, & j = k, \\ -\frac{n(K-2)}{K^2(K-1)}, & j \neq k, \end{cases}$$

for $n_{k'} = n/K$ (balanced data). By the union bound, the misclassification error is

$$(K-1)\Pr\left[\mathcal{N}(n/K, \sigma^2) < \mathcal{N}(-\frac{n(K-2)}{K^2(K-1)}, \sigma^2)\right] = (K-1)\Phi\left(-\frac{n}{K\sigma}\left(1 + \frac{K-2}{K(K-1)}\right)\right).$$

□

Proof sketches of the insights. Note that in Equation equation 5, the gradient is a linear function of the feature map thanks to the zero-initialization while for least-squares loss, one can derive a similar gradient as Equation 5. Thus, the proof can be extended to the least squares loss directly. Moreover, by replacing n_k with $n_k\eta$ in equation 5, one can extend the results to any η.

F.3. Omited details for binary classification

Recall the re-parameterization for $K = 2$. Precisely, an equivalent neural collapse case gives $M = [-e_1, e_1]$ with $e_1 = [1, 0, \ldots, 0]^T$. Furthermore, we consider the re-parameterization with $y \in \{-1, 1\}$, $\theta \in \mathbb{R}^p$ and the decision rule being $\hat{y} = \operatorname{sign}(\theta^T x)$. Then, the logistic loss is $\log(1 + e^{-y \cdot \theta^T x})$.

Theorem F.3. *For the case $K = 2$, under perfect neural collapse, using the aforementioned re-parameterization, the misclassification error is*

$$\Pr[\widehat{y} \neq y] = \Phi\left(-\frac{n}{2\sigma}\right) \leq e^{-\frac{n^2}{2\sigma^2}}.$$

Moreover, to make the misclassification error less than γ, the sample complexity is $O\left(\frac{\sqrt{\log(1/\gamma)}}{\sqrt{\rho}}\right)$.

Remark. The bound derived here is optimal up to a $\log n$ term. In fact, we have

$$\Pr[\widehat{y} \neq y] = \Phi\left(-\frac{n}{2\sigma}\right) \geq \frac{2\sigma}{n} e^{-\frac{n^2}{2\sigma}} = e^{-\frac{n^2}{2\sigma^2} + \log\frac{2\sigma}{n}}.$$

Proof. According to the re-parameterization, for the class imbalanced case, we have

$$\hat{\theta} = -\eta\left(\frac{n}{2} \cdot 0.5 \cdot (-\begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}) + \frac{n}{2} \cdot 0.5 \cdot \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \mathcal{N}(0, \frac{G^2}{2\rho} I_p)\right) = -\eta\left(\begin{bmatrix} n/2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \mathcal{N}(0, \frac{G^2}{2\rho} I_p)\right).$$

The rest of the proof is similar to that of Theorem F.1.

For the class-imbalanced case, assume that we have αn data points have with label $+1$ while the rest $(1-\alpha)n$ points have label -1. Then, the gradient is

$$\hat{\theta} = -\eta \left(\frac{n\alpha}{2} \cdot\cdot (- \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}) + \frac{n(1-\alpha)}{2} \cdot\cdot \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \mathcal{N}(0, \frac{G^2}{2\rho} I_p) \right) = -\eta \left(\begin{bmatrix} n/2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \mathcal{N}(0, \frac{G^2}{2\rho} I_p) \right).$$

Thus, the same conclusion holds. □

F.4. Domain adaption

Neural collapse in domain adaptation: In many private fine-tuning scenarios, the model is initially pre-trained on an extensive dataset with thousands of classes (e.g., ImageNet), denoted as K_0 class, and is subsequently fine-tuned for a downstream task with a smaller number of classes, denotes as $K \leq K_0$. We formalize it under the neural collapse setting as follows.

Let $P = [P_1, \cdots, P_{K_0}] \in \mathbb{R}^{p \times K_0}$ be a partial orthogonal matrix with $P^T P = I_{K_0}$. Let $\widetilde{M} = [\widetilde{M}_1, \cdots, \widetilde{M}_K]$ be a matrix where each $\widetilde{M}_i$ is a column of an ETF $M \in \mathbb{R}^{p \times K_0}$. With prefect neural collapse, we assume $\Pr[x = \widetilde{M}_k | y_k = 1] = 1$. The following theorem shows that the dimension-independent property still holds when private dataset has a subset classes of the pre-training dataset.

Theorem F.4. *Let $\widehat{y}$ be a predictor trained by NoisyGD under the cross entropy loss with zero initialization. Assume that the training dataset is balanced. For multi-class classification problems under neural collapse with K classes, subset of a gigantic dataset with $K_0 \geq K$ classes, the misclassification error is*

$$\Pr[\widehat{y} \neq y] \leq (K-1)\Phi\left(\frac{nC_{K,K_0}}{\sigma}\right) \leq (K-1)e^{-\frac{n^2 C_{K,K_0}^2}{2\sigma^2}}$$

with $C_{K,K_0} = \frac{1}{K}\left[\frac{K \cdot K_0 - 2}{K^2(K_0-1)}\right]$ and $\sigma^2 = \frac{G^2}{2\rho}$.

Proof. Let

$$M_0 = \sqrt{\frac{K_0}{K_0-1}} P \left(I_{K_0} - \frac{1}{K_0} \mathbf{1}_{K_0} \mathbf{1}_{K_0}^T \right) = \sqrt{\frac{K_0}{K_0-1}} \left(P - \frac{1}{K_0} \sum_{k=1}^{K_0} P_k \mathbf{1}_{K_0}^T \right).$$

Denote $M = [M_1, \cdots, M_K]$ with each M_k being a column of M_0. Note that

$$M_{k'}^T M_k = \frac{K_0}{K_0 - 1} \left(\delta_{k,k'} - \frac{1}{K_0} \right).$$

We have

$$\boldsymbol{\mu}_{n,K} := -\sum_{k'=1}^{K} n_{k'} \left(\frac{1}{K} \mathbf{1}_K - y^{k'} \right) M_{k'}^T M_k.$$

For $j \neq k$, we have

$$(\mu_{n,K})_j = -\frac{n}{K} \left[\frac{1}{K} + \frac{K-1}{K(K_0-1)} - \frac{K-2}{K(K_0-1)} \right] = -\frac{n(K_0-2)}{K^2(K_0-1)}.$$

For $j = k$, it holds

$$(\mu_{n,K})_j = -\frac{n}{K}\left[\frac{1}{K} - 1 - \frac{K-1}{K(K_0-1)}\right] = \frac{n(K-1)K_0}{K^2(K_0-1)}.$$

By the union bound, the misclassification error is

$$(K-1)\Pr\left[\mathcal{N}((\mu_{n,K})_k, \sigma^2) < \mathcal{N}((\mu_{n,K})_1, \sigma^2)\right] = (K-1)\Phi\left(\frac{nC_{K,K_0}}{\sigma}\right)$$

with $C_{K,K_0} = \frac{1}{K}\left[\frac{K\cdot K_0 - 2}{K^2(K_0-1)}\right]$. □

G. Proofs of Section 4

G.1. Details of the normalization

Consider the case where the feature is shifted by a constant offset v. The feature of the k-th class is

$$\widetilde{x}_i = x_i - \frac{1}{n}\sum_{i=1}^{n} x_i = \widetilde{M}_k = M_k + v$$

with M_k being the k-th column of the ETF M.

The offset v can be canceled out by considering the differences between the features. That is, we train with the feature $\widetilde{M}_k - \frac{1}{K}\sum_{j=1}^{K}\widetilde{M}_j$ for the k-th class. In fact, let P_k be the k-th column of P and we have

$$\begin{aligned}
\widetilde{M}_k - \frac{1}{K}\sum_{j=1}^{K}\widetilde{M}_j &= M_k - \frac{1}{K}\sum_{j=1}^{K} M_j \\
&= \sqrt{\frac{K}{K-1}}\left[\left(P_k - \frac{1}{K}\sum_{i=1}^{K} P_i\right) - \frac{1}{K}\sum_{j=1}^{K}\left(P_j - \frac{1}{K}\sum_{i=1}^{K} P_i\right)\right] \\
&= \sqrt{\frac{K}{K-1}}\left(P_k - \frac{1}{K}\sum_{j=1}^{K} P_j\right) = M_k.
\end{aligned}$$

G.2. Proof of Theorem 4.1

Proof of Theorem 4.1. Consider the case with $K = 2$ and a projection vector $\widehat{P} = (e_1 + \Delta)$ with some perturbation $\Delta = (\Delta_1, \cdots, \Delta_p)$ such that $\|\Delta\|_\infty \le \beta_0$ for some $0 < \beta_0 \ll p$. $\widehat{P}$ can be generated by the pre-training dataset or the testing dataset. Consider training with features $\widetilde{x}_i = \widehat{P}x_i$. Then, the sensitivity of the NoisyGD is $G = \sup_v |\widehat{P}^T(e_1+v)| = 1 + \beta + \beta|\Delta_1| + \beta(\sum_{j=1}^{p}|\Delta_j|) \le 1 + \beta(1 + \beta_0 + p\beta_0)$. The output of Noisy-GD is then given by

$$\widehat{\theta} = -\widehat{P}\cdot\left(\sum_{i=1}^{n} y_i\widetilde{x}_i\right) + \mathcal{N}(0, \sigma^2).$$

Moreover, for any testing data point $e_1 + v$, define

$$\widehat{\mu}_n = -\left(\sum_{i=1}^{n} y_i\widetilde{x}_i\right)\widehat{P}^T(e_1+v) = (e_1+V)^T\,\widehat{P}\widehat{P}^T(e_1+v)$$

with $V = \frac{1}{n}\sum_{i=1}^{n} v_i =: (V_1, \cdots, V_p)$.

We now divide $\widehat{\mu}_n$ into four terms and bound each term separately.

For the first term $e_1^T\widehat{P}\widehat{P}^T e_1$, it holds

$$e_1^T\widehat{P}\widehat{P}^T e_1 = (1 + e_1^T\Delta_1)^2 \le (1-\beta_0)^2.$$

For the second term $V^T\widehat{P}\widehat{P}^T e_1$, we have

$$V^T\widehat{P}\widehat{P}^T e_1 = (V_1 + V^T\Delta)(1+\Delta_1) \leq |V_1 + V^T\Delta|(1+\beta_0)$$

Note that V_1 is the average of n i.i.d. random variables bounded by β. By Hoeffding's inequality, we obtain

$$|V_1| \leq \frac{\beta \log\frac{2}{\gamma}}{\sqrt{n}}, \text{ with probability at least } 1-\gamma.$$

Similarly, with confidence $1-\gamma$, it holds

$$|V^T\Delta| \leq \frac{p\beta\beta_0 \log\frac{2}{\gamma}}{\sqrt{n}}.$$

The third term $e_1^T\widehat{P}\widehat{P}^T v$ can be bounded as

$$|e_1^T\widehat{P}\widehat{P}^T v| = (1+\Delta_1)\left(\sum_{j=1}^{p} v_i\,(1+\Delta_i)\right) \leq (1+\beta_0)\left(\beta + \beta_0\sqrt{p\log\frac{2}{\gamma}}\right),$$

where the last inequality is a result of the Hoeffding's inequality by assuming that each coordinate of v are independent of each others. Moreover, without further assumptions on the independence of each coordinate of v, we have

$$|e_1^T\widehat{P}\widehat{P}^T v| = (1+\Delta_1)\left(\sum_{j=1}^{p} v_i\,(1+\Delta_i)\right) \leq (1+\beta_0)\,(\beta+\beta_0 p)\,.$$

Using the Hoeffding's inequality again, for the last term $V^T\widehat{P}\widehat{P}^T(e_1+v)$, it holds

$$|V^T\widehat{P}\widehat{P}^T(e_1+v)| \leq \frac{(\beta+\beta_0\sqrt{p})(1+\beta+\beta_0+\beta\beta_0\sqrt{p})\log\frac{4}{\gamma}}{\sqrt{n}}$$

with confidence $1-\gamma$ if we assume that all coordinates of v are independent of each other. Without further assumptions on the independence of each coordinate of v, we have

$$|V^T\widehat{P}\widehat{P}^T(e_1+v)| \leq \frac{(\beta+\beta_0 p)(1+\beta+\beta_0+\beta\beta_0 p)\log\frac{2}{\gamma}}{\sqrt{n}}.$$

□

H. Some calculations on random Initialization

In machine learning, training a deep neural network using (stochastic) gradient descent combined with random initialization is widely adopted (Sutskever et al., 2013). The significance of random initialization on differential privacy in noisy gradient descent is also emphasized by (Ye et al., 2023; Wang et al., 2023). Extending our theory from zero initialization to random initialization is non-trivial and we discuss some details in this section.

H.1. Gaussian initialization without offset

For Gaussian initialization $\xi = (\xi_1, \cdots, \xi_p) \sim \mathcal{N}(0, I_p)$, we have

$$\begin{aligned}\hat{\theta} &= \xi - \eta\left(\frac{n}{2}\cdot\frac{-e^{-\xi_1}}{1+e^{-\xi_1}}\cdot(-\begin{bmatrix}-1\\0\\\vdots\\0\end{bmatrix}) + \frac{n}{2}\cdot\frac{-e^{-\xi_1}}{1+e^{-\xi_1}}\cdot\begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix} + \mathcal{N}(0,\frac{G^2}{2\rho}I_p)\right)\\ &= \xi + \eta\left(\frac{e^{-\xi_1}}{1+e^{-\xi_1}}\cdot\begin{bmatrix}n\\0\\\vdots\\0\end{bmatrix} + \mathcal{N}(0,\frac{G^2}{2\rho}I_p)\right)\end{aligned}$$

The sensitivity is $\frac{e^{-\xi_1}}{1+e^{-\xi_1}} < 1$. Consider $x = (-1, 0, \cdots, 0)^T$. We have

$$\widehat{\theta}^T x = -\xi_1 + \eta\left(-\frac{ne^{-\xi_1}}{1+e^{-\xi_1}}\right) + \mathcal{N}(0, \frac{G^2}{2\rho}) =: \mu_{\xi_1,n} + \mathcal{N}(0, \frac{G^2}{2\rho}).$$

The misclassification error is

$$\begin{aligned}\Pr[\widehat{\theta}^T x > 0] &= \mathbb{E}_{\xi_1\sim\mathcal{N}(0,1)} \Pr\left[\mathcal{N}\left(\mu_{\xi_1,n}, \frac{G^2}{2\rho}\right) > 0 \middle| \xi_1\right] \\ &= \mathbb{E}_{\xi_1\sim\mathcal{N}(0,1)}\left[\Phi\left(\frac{\sqrt{2\rho}\mu_{\xi_1,n}}{G}\right)\right]\end{aligned}$$

H.2. Gaussian initialization with off-set

Denote $x_1 = -e_1 + v$ and $x_2 = e_1 + v$ with $\|v\|_\infty \le \beta$. For the logistic loss $\ell(y, \theta^T x) = \log(1 + e^{-y\theta^T x})$, we have

$$g(\theta, y\cdot x) := \nabla_\theta \ell(y, \theta^T x) = \frac{e^{-y\theta^T x}}{1+e^{-y\theta^T x}}(-yx).$$

Denote

$$g_1(\theta) = g(\theta, -1\cdot x_1) = \frac{e^{\theta^T x_1}}{1+e^{\theta^T x_1}} x_1$$

and

$$g_2(\theta) = g(\theta, 1\cdot x_2) = \frac{e^{-\theta^T x_2}}{1+e^{-\theta^T x_2}}(-x_2).$$

If we shift the feature by some vector v, then the loss function is

$$R_n = \frac{n}{2}\log(1+e^{\theta^T x_1}) + \frac{n}{2}\log(1+e^{-\theta^T x_2}).$$

And the gradient is

$$\nabla_\theta R_n(\theta) = \frac{n}{2}\left(g_1(\theta) + g_2(\theta)\right).$$

Thus, the output of one-step NoiseGD is given by

$$\widehat{\theta} = \theta_0 - \frac{\eta n}{2}\left[g_1(\theta_0) + g_2(\theta_0) + \mathcal{N}(0, \sigma^2)\right].$$

Let $\mu_\xi = \xi - \frac{\eta n}{2}\left[g_1(\xi) + g_2(\xi)\right]$. Then, we have

$$\mu_\xi^T e_1 = \xi_1 - \frac{\eta n e^{\xi^T x_1}}{2+2e^{\xi^T x_1}}(-1+v_1) + \frac{\eta n e^{\xi^T x_2}}{2+2e^{\xi^T x_2}}(1+v_1).$$

And the misclassification error is

$$\mathbb{E}_\xi\left(\Phi\left(-\frac{\sqrt{2\rho}\mu_\xi^T e_1}{G}\right)\right).$$

I. Further discussions on Gaussian DP, synthetic data, and DP-SGD

It is obvious that our theory for NoisyGD can be further generalized using Gaussian differential privacy (Dong et al., 2022), which leads to better utility analysis of our theory. However, extending our theory to NoisySGD is not straightforward, as the privacy accounting becomes complicated when considering sub-sampling due to mini-batches (Zhu & Wang, 2019; Wang et al., 2020; Balle et al., 2018). If we further consider multiple iterations, the joint effects of sub-sampling and composition on the privacy budget is another challenge (Zhu et al., 2022). Besides using composition and the central limit theorem for Gaussian DP (Bu et al., 2020; Wang et al., 2022), incorporating recently developed privacy analyses of last-iteration output of Noisy(S)GD in terms of the training dynamic (Ye & Shokri, 2022; Altschuler & Talwar, 2022; Bok et al., 2024) to obtain refined privacy analyses is another potential future topic. Existing privacy analyses of the last-iteration output are applied to strongly convex loss functions or convex functions with bounded domain, which is applicable to our setting when fine-tuning the last layer. If we further consider the privacy of NoisySGD under random initialization, the privacy analysis becomes much more complicated, as studied by (Ye et al., 2023; Wang et al., 2023). Moreover, in Appendix H, we discussed some extensions of our utility theory to the random initialization case, which shows that the utility theory is much more sophisticated than in the 0-initialization case.

In addition to DP-ERM, there has been notable recent interest in using public data to enhance the accuracy of DP synthetic data (Ghalebikesabi et al., 2023; Liu et al., 2021). The incorporation of public data, either for traditional query-based synthetic data methods (McKenna et al., 2021; Li et al., 2023b) or more recent techniques such as DP generative adversarial networks or diffusion models (Goodfellow et al., 2020; Ho et al., 2020), has shown promise. The possibility of extending our perspective from Neural Collapse on DP-ERM to DP synthetic data in future research is intriguing.