LARGER MODEL CAUSES LOWER CLASSIFICATION ACCURACY UNDER DIFFERENTIAL PRIVACY: REASON AND SOLUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Differential privacy (DP) is an essential technique for privacy-preserving, which works by adding random noise to the data. In deep learning, DP-stochastic gradient descent (SGD) is a popular technique to build privacy-preserving models. With a small noise, however, the large model (such as ResNet50) trained by DP-SGD cannot perform better than the small model (such as ResNet18). To better understand this phenomenon, we study high dimensional DP learning from the viewpoint of generalization. Theoretically, we first demonstrate that for the Gaussian mixture model with even small DP noise, if excess features are used, classification can be as bad as the random guessing since the noise accumulation for the estimation in high dimensional feature space. Then we propose a robust measure to select the important features, which trades off the model accuracy and privacy preserving. Moreover, the conditions under which important features can be selected by the proposed measure are established. Simulation on the real data (such as CIFAR-10) supports our theoretical results and reveals the advantage of the proposed classification and privacy preserving procedure.

1 INTRODUCTION

Deep neural networks have made a series of remarkable achievements in the field of image recognition and classification, natural language processing. But training deep neural networks typically requires large and representative data to achieve high-performance Gheisari et al. (2017). Since the datasets often contain some sensitive information, such as medical records, location, purchase history, when we use these sensitive data to train a model without specific measures to the secret information, individual privacy can be leaked Fung et al. (2010). Thus, privacy-preserving is a crucial issue in deep learning.

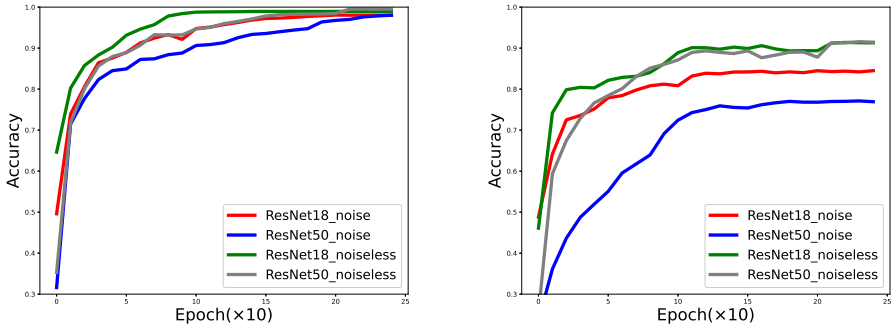
One of the most popular techniques for privacy-preserving is (ε, δ)-DP that was first proposed by Dwork et al. (2014). DP works by adding noise or adding randomness to the data while it provides a rigorous mathematical framework for preserving privacy, then many works with DP have been proposed to protect individual privacy Chen and Lin (2014); Goodfellow et al. (2016).

Recently, deep neural networks with millions of parameters have proven to outperform smaller models (such as GPT-3 Brown et al. (2020)), although the deeper neural networks are more difficult to train. In He et al. (2016), they present a network called ResNet to overcome this problem and it can gain accuracy from the considerably increased depth of networks.

When we consider privacy-preserving, DP-SGD (Abadi et al. (2016)) adding Gaussian noise to gradient is popular to train the neural network (Dupuy et al. (2021)). Then we use DP-SGD to train ResNet50 and ResNet18, respectively. In Fig. 1, we observe that a degradation problem has been exposed by adding noise. However, ResNet50 has lower test accuracy than ResNet18. Similar phenomena on MNIST for CNN is presented in Appendix (see Fig. **??**, Fig. **??**).

The above observation motivates us to ask the following question:

Why do the larger models cause lower classification accuracy under DP?



(a) Accuracy in training set

(b) Accuracy in test set

Figure 1: The performance of ResNet on CIFAR-10 by DP-SGD with $\varepsilon = 2, \delta = 0.0001$. (a) is the result in the training set, we see that both the ResNet 18 and ResNet 50 with noise or without noise obtain 98% classification accuracy, respectively. (b) is the result in the test set, we see that the performance of ResNet 50 and ResNet 18 under noiseless condition is the same but ResNet50 causes much lower test accuracy than ResNet18.

We answer this question from the generalization aspects of differential private learning since the larger model leads to higher test error rather than training error. In addition, we propose to select a subset of features to trade off the classification accuracy and privacy-preserving.

1.1 Our contributions

- **Generalization bound.** We first analyze generalization in a simple GMM model under DP. By focusing on specific Gaussian noise, we can establish information-theoretic upper bounds of the classification error, which depends on the size of dimension and noise. With the increasing of dimension, noise can accumulate on dimension to cause classification error increase, finally, the classifier performs nearly the same as random guessing. It implies that the larger model with high dimension cause lower classification accuracy under DP.
- **Feature selection.** Since models have increasing classification error with an increasing number of features, we use the feature selection technique to reduce the dimension. A novel filter feature selection method is proposed, which uses a distance measure to assign a scoring to each feature. Comparing with t-statistic, the proposed method can obtain the stable and important features under DP.
- **Experiment.** We perform simulation based on synthetic data and common real data such as RCV1 (Lewis et al. (2004)), CIFAR-10. After using the proposed feature selection method, we show that ResNet50 performs better than ResNet18 on CIFAR-10 in terms of DP.

1.2 Outline of the paper

In the next section, we give some definitions and preliminaries. In Section 3, we analyze a simple GMM model and prove that larger models lead to higher error under DP. Then we proposed a feature selection algorithm for dimension reduction in differential privacy. The simulation in Section 4 reveals that feature selection can improve the performance and the proposed method performs better in some real dataset including RCV1 and CIFAR10.

2 Basic Definitions

In this section, we first define (ε, δ)-DP. Moreover, we consider a simple Gaussian mixture model (GMM) under DP. Then, We will analyze a Fisher classifier for this GMM model.

Definition 1. *(Differential Privacy Dwork (2008)) A randomized algorithm $\mathcal{M}$ with domain dataset $\mathcal{D}$ is (ε, δ)-differential private if for all $\mathcal{S} \subseteq$ Range $(\mathcal{M})$ and for all $x, y \in \mathcal{D}$ that $\|x - y\|_1 \leq 1$:*

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq \exp(\varepsilon) \Pr[\mathcal{M}(y) \in \mathcal{S}] + \delta. \tag{1}$$

Since Definition 1 imposes no limitations on randomized algorithm $\mathcal{M}$, we use the following Gaussian mechanism that adding Gaussian noise, which we can create a DP algorithm for function f with sensitivity $\Delta f \triangleq \max \|f(d_i) - f(d_j)\|_1$, where the maximum is over all pairs of datasets d_i and d_j in dataset $\mathcal{D}$ differing in at most one element and $\|\cdot\|_1$ denotes the ℓ_1 norm.

Definition 2. *(Gaussian Mechanism Dwork et al. (2014)) Given any function $f : \mathcal{D} \to \mathbb{R}^k$, the (ϵ, δ)-Gaussian mechanism is defined as:*

$$\mathcal{M}_L(x, f(\cdot), \varepsilon) = f(x) + (Y_1, \ldots, Y_k) \tag{2}$$

where Y_i are i.i.d. random variables drawn from $\mathcal{N}(0, \sigma^2)$ where $\sigma = \Delta f \cdot \ln(1/\delta)/\epsilon$.

Consider the p-dimensional classification problem between two classes C_1 and C_2. Suppose our clean data comes from the Gaussian mixture model (GMM). To analyze the impact of DP, based on the Gaussian mechanism, we can use GMM adding Gaussian noise to achieve (ϵ, δ)-DP.

Definition 3. *(Private GMM) Let $\mu_k \in \mathbb{R}^p$, $k = 1, 2$, be the per-class mean vector and*

$$\mathbf{\Sigma} \triangleq \operatorname{diag}(\sigma_1^2, ..., \sigma_p^2) \tag{3}$$

be the variance parameter. $(\epsilon, \delta)-$private Gaussian mixture model is defined by the following distribution over $(\hat{x}_k, k) \in \mathbb{R}^p \times \{1, 2\}$: First, draw a label k from $\{1, 2\}$ uniformly at random, then sample the data point $x_k \in \mathbb{R}^p$ from $\mathcal{N}(\mu_k, \mathbf{\Sigma})$. Then we get a non-private dataset $\{x_k^i, k\}$, $k = 1, 2$, $i = 1, \ldots, n_k$. Finally, according Gaussian mechanism to obtain dataset $\{\hat{x}_k^i, k\}$, where

$$\hat{x}_k^i = x_k^i + 2C_p \ln(1/\delta)/\epsilon \cdot (\eta_1, ..., \eta_p), \tag{4}$$

where η_i are i.i.d variables $\eta_i \sim \mathcal{N}(0, 1)$ and $C_p \triangleq \max_{k \in \{1,2\}, i \leq n_k} \|x_k^i\|_1$ is a constant depending on dimension p.

From private GMM, we can obtain some training data $\{\hat{x}_k^i, k\}$, $k = 1, 2$, $i = 1, \ldots, n_k$. Let $n = n_1 + n_2$. Using these training data, the parameters μ_k and Σ can be estimated by

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \hat{x}_k^i, k = 1, 2, \hat{\mathbf{\Sigma}} = \operatorname{diag}\left\{ \frac{\left(S_{1j}^2 + S_{2j}^2\right)}{2}, j = 1, \ldots, p \right\}, \tag{5}$$

where $S_{kj}^2 = \frac{1}{(n_k - 1)} \sum_{i=1}^{n_k} \left(\hat{x}_{kj}^i - \bar{x}_{kj}\right)^2$ is the sample variance of the j-th feature in class k and $\bar{x}_{kj} = \frac{1}{n_k} \sum_{i=1}^{n_k} \hat{x}_{kj}^i$.

Consider the following classification rule.

Definition 4. *(Fisher Classifier Hart et al. (2000)) The Fisher classifier is defined as:*

$$\hat{\delta}_n(x) = (x - \hat{\mu})\hat{\mathbf{\Sigma}}^{-1}\hat{\alpha}, \tag{6}$$

where $\hat{\mu} = \frac{1}{2}\left(\hat{\mu}_1 + \hat{\mu}_2\right), \hat{\alpha} = \hat{\mu}_1 - \hat{\mu}_2$.

From Definition 4, it shows that if $\hat{\delta}_n(x) > 0$, which classifies sample x into class C_1. Let us denote the parameter by $\theta = (\mu_1, \mu_2, \Sigma)$, we define the following classification error.

Definition 5. *(Classification Error) If we have a new observation x from class C_1, then the classification error $\mathbf{W}(\hat{\delta}_n, \theta)$ of the Fisher classifier is defined by*

$$\mathbf{W}(\hat{\delta}_n, \theta) \triangleq P(\hat{\delta}_n(x) \leq 0 | \hat{x}_k^i, k = 1, 2, i = 1, \ldots, n_k) = 1 - \Phi(\Psi), \tag{7}$$

where

$$\Psi = \frac{(\mu_1 - \hat{\mu})\prime\hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_1 - \hat{\mu}_2)}{\sqrt{(\hat{\mu}_1 - \hat{\mu}_2)\prime\hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_1 - \hat{\mu}_2)}}, \tag{8}$$

and $\Phi(\cdot)$ is the standard Gaussian distribution function.

3 THEORETICAL RESULTS

In this section, we first prove that with added noise, the error increases as the dimension increases. The intuition is that noise for different features can accumulate to cause large classification error. Then we focus on a criterion suitable for feature selection under DP to reduce dimension. Finally, we give an algorithm to realize our criterion for a dataset.

3.1 IMPACT OF HIGH DIMENSION UNDER DP-MECHANISM

In this part, we first give an upper bound for the classification error. Without loss of generality, the sample data are assumed to be balanced.

Theorem 6. *Suppose the training data comes from private GMM (Definition 2) and $n_1 = n_2$. In addition, assume $\log p = o(n), n = o(p)$. Then the classification error $\mathbf{W}(\hat{\delta}_n, \theta)$ is bounded by*

$$\mathbf{W}(\hat{\delta}_n, \theta) \leq 1 - \Phi\left(\frac{(1 + o_p(1))\,\Gamma}{2\left[\frac{4p}{n} + (1 + o_p(1))\,\Gamma\right]^{\frac{1}{2}}}\right), \tag{9}$$

where δ, ϵ, C_p are defined in equation 4, respectively; $\alpha = \mu_1 - \mu_2$ and μ_1 and μ_2 are the per-class mean vectors; $o_p(1)$ is a variable decreasing when p increasing;

$$\Gamma \triangleq \sum_{j=1}^{p} \frac{\alpha_j^2}{\sigma_j^2 + (2C_p \ln(1/\delta)/\epsilon)^2} \tag{10}$$

where α_j is j-th of α and σ_j^2 is defined in equation 3.

Remark 1. *The condition $\log p = o(n), n = o(p)$ means that n grows much slower than p while $\log p$ grows much slower than n. It is one of the common assumptions to study the high dimensional learning with low sample size Tsybakov (2003).*

Let $\tau \triangleq \ln(1/\delta)/\epsilon$ and $p \to \infty$, we derive an upper bound for Γ defined in equation 10.

$$\Gamma < \frac{\sum_{j=1}^{p} \alpha_j^2}{(2C_p \ln(1/\delta)/\epsilon)^2} \leq \frac{1}{\tau^2} \frac{\sum_{j=1}^{p} |\alpha_j|^2}{\max_{a \leq n_1, b \leq n_2} \|x_1^a - x_2^b\|_1^2} \leq \frac{1}{\tau^2} \frac{\sum_{j=1}^{p} |\alpha_j|^2}{(\sum_{i=1}^{p} |\alpha_i|)^2} \leq \frac{1}{\tau^2}, \tag{11}$$

where the second inequality dues to the definition of C_p. Since C_p is the largest norm of data, the norm of distance between two classes should not be huger than $2C_p$. The third inequality caused by the maximum distance between two classes is no less than the distance of true means of each class with probability 1, i.e., $P\left(\max_{a \leq n_1, b \leq n_2} \|x_1^a - x_2^b\|_1 \geq \sum_{j=1}^{p} |\alpha_j|\right) \xrightarrow{n} 1$.

Note that Γ can be controlled by an upper-bound without p.

Remark 2. *We can see two aspects from this theorem. First, for fixed noise with given ϵ, δ, when $p \to \infty$, denominator in the right side of equation 9 towards infinity. Thus the classification error is*

$$\mathbf{W}(\hat{\delta}_n, \theta) \to 1 - \Phi(\mathcal{O}(\frac{1}{\sqrt{p}})) \to \frac{1}{2}, \tag{12}$$

where $\mathcal{O}(d)$ means that it grows at the order of d. According to equation 12, it shows the Fisher classifier with high dimension performs nearly the same as random guessing, which is similar to the result in Fan and Fan (2008).

However, when we consider perturbation in equation 9 with fixed p and n. When ϵ and δ decrease to 0, i.e., , $\tau \to \infty$, which means the noise is large enough. Thus the classification error is

$$\mathbf{W}(\hat{\delta}_n, \theta) \to 1 - \Phi(\mathcal{O}(\frac{1}{\tau^2})) \to \frac{1}{2}, \tag{13}$$

which is merely random guessing without any ability to classify. Moreover, when $p \to \infty$ and $\epsilon \to 0$ at the same time, then the classification error is

$$\mathbf{W}(\hat{\delta}_n, \theta) \to 1 - \Phi(\mathcal{O}(\frac{1}{\sqrt{p\tau^4}})) \to \frac{1}{2}. \tag{14}$$

Compared equation 14 with equation 12, it reveals that the larger noise can speed up the rate of model degradation.

From Theorem 6 and the above remark, it shows that the larger model with high dimension leads to lower classification accuracy under DP. To trade off the classification accuracy and privacy-preserving, we use the feature selection technique to reduce the dimension.

3.2 FEATURE SELECTION

In this subsection, we use filter feature selection methods, which assign a score (often a statistical measure) to each feature. One typical statistical measure is t-statistics Hua et al. (2009), which is defined as follows

$$T_j = \frac{\bar{x}_{1j} - \bar{x}_{2j}}{\sqrt{S_{1j}^2/n_1 + S_{2j}^2/n_2}} j = 1, \cdots, p, \tag{15}$$

where $\bar{x}_{kj}$ and S_{kj} are defined in equation 5. After computing the values of t-statistic for each feature, we sort these values in descending order and select the important feature. Moreover, under the DP setting, we hope the feature selection result is independent of the perturbation.

When there exists a significant difference between the means of two classes, t-statistic can perform well for finding important features. However, when we add noise to the data, the selected feature using t-statistic is susceptible to perturbation, since the formulation of t-statistic relies on sample variance. Specifically, according to the definition of $S_{k,j}$ defined in equation 5, we calculate the expectation of $\hat{\mathbf{\Sigma}}$ as $\mathbf{E}(\hat{\mathbf{\Sigma}}) = \mathbf{\Sigma} + (2C_p \ln(1/\delta)/\epsilon)^2 * \mathbf{I}_p$. It shows that the DP budget ϵ and δ can influence the value of t-statistic. Here we also give an example to show it.

Example 1. *Consider a binary classification problem based on private GMM (Definition 3). The variance and mean vector set* $\mathbf{\Sigma} = \mathrm{diag}(1, 10)$ *and* $\mu_1 = [0, 0]$ $\mu_2 = [5, 10]$ *(Fig.2a), respectively. We sample* $n_1 = n_2 = 200$ *for each class.*

Firstly, if we use the clean data without adding noise in private GMM, the value of t-statistic, $D = \mu_2 - \mu_1$*,* S_1^2 *and* S_2^2 *are calculated as the following table.*

feature	*1*	*2*
D	*5*	*10*
variance	*1*	*10*
t-statistics	*50*	*31.62*

.

The above table shows that feature 1 has a bigger t-statistics, thus we select feature 1 if we only require one feature.

Secondly, when we add noise with DP budget of $\epsilon = 3$ *in private GMM (Fig.2b), the results are presented as follows*

feature	*1*	*2*
D	*5*	*10*
variance perturbed	*10*	*19*
t-statistics	*15.82*	*22.94*

.

Thus feature 2 is a better result to be selected.

This example shows the best feature or the sort of t-statistic is not stable to perturbation, which means a small noise on data may create a new rank and it is harmful for feature selection in DP. However, numerator of t-statistics is stable since $E(\hat{\mu}_k) = \mu_k$ regardless of perturbation (see the first row of the above tables). It suggests us to consider the following distance criterion for selecting the important feature.

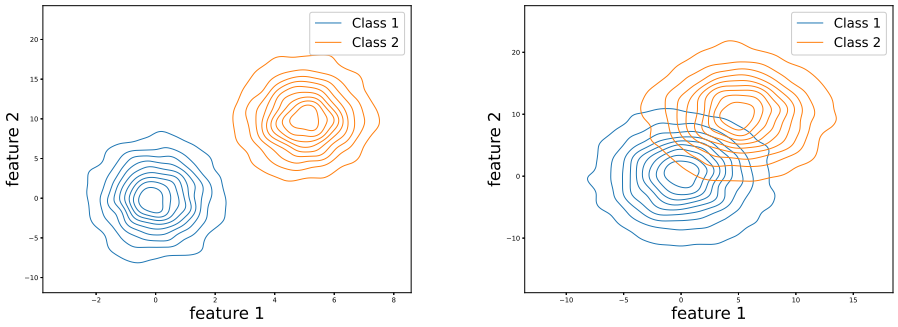
Definition 7. *Distance criterion is defined as follow:*

$$\hat{D}_j = \hat{x}_{1j} - \hat{x}_{2j}, \tag{16}$$

where $\hat{x}_{kj}$ *is the average of class* k*, feature* j*.*

This is a stable criterion since $\mathbf{E}(\hat{D}) = \mu_1 - \mu_2$ whether the data has noise or not. Next, we give a theorem to show that the proposed distance criterion can distinguish those useful features with probability one.

Assumption 1.



(a) Feature distribution before perturbation (b) Feature distribution after perturbation

Figure 2: Distribution for classes in different situations. For the left figure, feature 1 of two distributions is almost no overlap which means this feature is powerful to distinguish class while feature 2 is not so powerful. For the right, both features have overlap. But feature 2 is merely over 1/2 while feature 1 is about 3/4, so feature 2 is more powerful now.

1. *Assume that distance vector $\alpha = \mu_1 - \mu_2$ is sparse and without loss of generality, only the first s entries are nonzero.*
2. *Assume that the elements of both diagonal matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are bounded with upper bound v.*

In high dimension learning with low size data, sparsity is always a consideration (Grčar et al. (2005)). Also, variance is normal to be seen as finite, otherwise, estimation of variance will not be close to true value with a low size of data.

The following theorem describes that all important features can be selected by distance criterion. Recall that $n = n_1 + n_2$ and n_k represents sample size of class k.

Theorem 8. *Let $c_1 \leq n_1/n_2 \leq c_2$, s be a value such that $\log(p-s) = o(n^\gamma)$ and $\log s = o\left(n^{\frac{1}{2}-\gamma}\beta_n\right)$ for some $\beta_n \to \infty$ and $0 < \gamma < \frac{1}{3}$. Suppose that $\min_{j=1,\ldots,p} |\alpha_j| = vn^{-\gamma}\beta_n$. Then under Assumption 1, for $y = cvn^{(\gamma-1)/2}$ with c some positive constant, we have*

$$P\left(\min_{j\leq s}\left|\hat{D}_j\right| \geq y \quad \text{and} \quad \max_{j>s}\left|\hat{D}_j\right| < y\right) \to 1. \tag{17}$$

Remark 3. *From Theorem 8, we observe that the proposed distance criterion can distinguish the non-zero feature with probability one. When these important features are selected, then the dimension can be reduced. Thus, combing with Theorem 6, classification accuracy and privacy-preserving can be traded off by feature selection.*

3.3 DP Feature Selection Algorithm (DFS)

Based on the proposed distance criterion, we design an integral algorithm to select the important feature under DP.

Since we clip feature from p to m ($m < p$), C_p and p in Theorem 6 will become smaller, thus classification error would be reduced.

Remark 4. *This algorithm bases on our private GMM. When we consider a neural network with inputs of image and text which are not vectors, we will use their latent layer of a neural network as features to utilize our algorithm.*

4 Experiment

In this section, we check our theoretical results by performing experiments on multiple common datasets, including synthetic data, RCV1 (Lewis et al. (2004)) and CIFAR-10. For all DP-

Algorithm 1: DP Feature Selection Algorithm

Input:: $[[\mathbf{X}_{11}],...,[\mathbf{X}_{1n_1}]]$ and $[[\mathbf{X}_{21}],...,[\mathbf{X}_{2n_2}]]$
Calculate average of features: $\hat{\mu}_1 = [a_{11}, ..., a_{1p}]$ and $\hat{\mu}_2 = [a_{21}, ..., a_{2p}]$
Calculate distance of features: $D = |\hat{\mu}_1 - \hat{\mu}_2|$
Rank features with distance: X_r = $[[x_{1[1]},...,x_{1[p]}],...,[x_{n[1]},...,x_{n[p]}]]$
Cut the first m features: X_c = $[[x_{1[1]},...,x_{1[m]}],...,[x_{n[1]},...,x_{n[m]}]]$
Calculate the maximum norm in X_c: $N_{max} \triangleq \max_{i \le n, X_i \in X_c} \|X_i\|_1$
Generate noise: $n \times m$ matrix ε with i.i.d. $\varepsilon_{ij} \sim \mathcal{N}(0, 2N_{max} \ln(1/\delta)/\epsilon)$
Add noise to feature: $\hat{X} = X_c + \varepsilon$
Output: feature with noise $\hat{X}$, $Label$

mechanism, choose normal distribution and set $\delta = 0.0001$. Then for different data set, we choose different DP budget of ϵ to protect the data.

4.1 SYNTHETIC DATA

For synthetic data, consider two high dimensional Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, where $\boldsymbol{\Sigma}_k = diag(a_{1_k}, ..., a_{p_k})$ with $p = 3000$ and $a_{i_j} \sim \exp(0.1)$. In addition, $\boldsymbol{\mu}_0 = \mathbf{0} \in \mathbb{R}^p$, $\boldsymbol{\mu}_1$ is a $(1-c)\delta_0 + \frac{1}{2}c\exp(-2|x|)$, where δ_0 means equals to 0 and $c = 0.88$.

Using the GMM parameters above, we generate 30 training data and 200 testing data for each class. To protect privacy, we add some Gaussian noise with $\epsilon = 5$ and $\epsilon = 10$ to all data except labels basing on private GMM definition3. The Fisher classifier is used to separate these two classes. Fig. 3 presents the test accuracy.

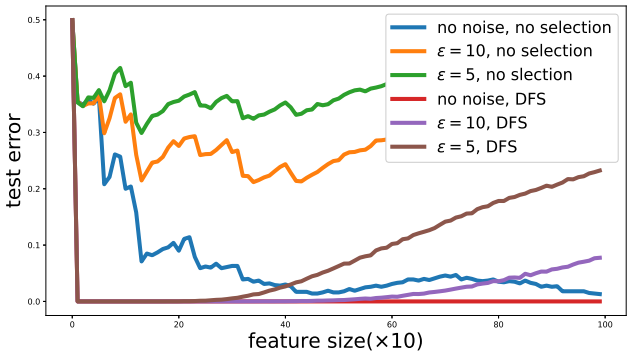
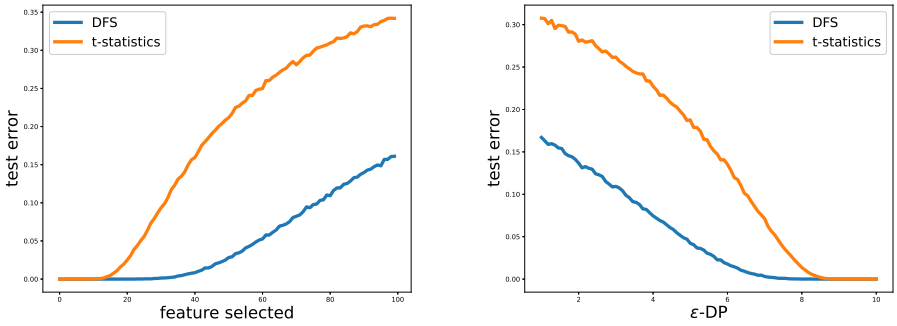


Figure 3: Comparison for different DP parameters and selection. For our algorithm, only 20 features can generate a model with the highest accuracy in all DP settings while no selection model cannot reach the best with noise.

In Fig.3, we observe that the proposed algorithm performs well in feature selection under different noise levels. In noiseless condition, the performance of feature selection and without feature selection is the same. However, within 20 features selected by DFS, our algorithm converges to 0 test error while feature without selection needs more than 1000 features. Moreover, DFS can select features to maintain the best accuracy for more than 500 features. Also, without feature selection, result is not smooth since some perturbing data influence the performance.

In Fig.4a, a comparison of proposed algorithms with t-statistic in terms of test error has been presented. It shows that the increase of dimension leads to the decrease of performance, which is consistent with Theorem 6. However, the curve of the proposed method is below that of t-statistic, thus the proposed method can reduce the influence of high dimension. Since the larger ε means the smaller noise, Fig. 4b shows the test error decreasing with the increasing the ε while the proposed method can reduce the test error much more. In addition, in both figures, our curves maintain paral-



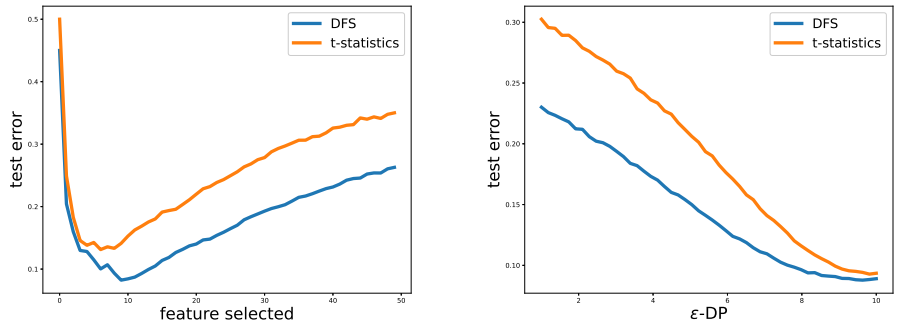
(a) fixed $\varepsilon = 4$ with increasing p (b) fixed $p = 70$ with increasing ε

Figure 4: Results for the numerical dataset. The left figure shows that our DFS maintains stable for $p < 30$ while t-statistic climbing all the time. Right shows that for fixed p, comparing with t-statistic, DFS obtain higher accuracy with the same DP budget ε.

lel for a long time, which means the proposed method is more resilient to dimension increasing and noise accumulation.

4.2 RCV1

RCV1 dataset is a famous embedding set. So we can regard it as features after extractor. Then we set it into a binary classification problem by choosing random 2 classes and draw 40 data each for training and 200 for the test.



(a) fixed $\varepsilon = 6$ with increasing p (b) fixed $p = 20$ with increasing ε

Figure 5: Results for RCV1. The left figure shows when the dimension of feature comes to 9, both algorithm reaches the best accuracy while DFS gets 0.06 test error. Right shows for DP budget ε from 1 to 6, test error of DFS is much smaller than that of t-statistic.

For Fig.5a and 5b, it also shows the test error of DFS is smaller than that of t-statistic. It is convinced that our algorithm can outperform traditional selection methods by t-statistic in the DP condition. Thus the proposed method provides a solution for the issue in the introduction.

4.3 CIFAR10

Recall our original problem in the introduction that ResNet50 draws back more due to noise accumulation, our selection rule helps to reduce this tendency. (Algorithm for multiple classes and details in this experiment is listed in the appendix.)

In this experiment, we use the last but one layer data of ResNet to represent the input data in our algorithm. Then adding perturbation on data according to the largest norm with the definition of

DFS. For a fair comparison, since ResNet18 has 512 features, we select 512 features from 2048 in ResNet50. Then we use multi-layer perceptrons (MLP) to train it with SGD without noise.

	CIFAR-10		
Model	Min	Max	Median
ResNet50	75.5	79.2	77.0
ResNet18	83.6	85.3	84.5
ResNet50+t-sta	78.4	81.3	80.1
ResNet50+DFS	**84.8**	**86.4**	**85.7**

Table 1: Result for features on CIFAR10 with ResNet18/50 under DP condition. We select 512 features from ResNet50 by DFS, then we see ResNet50 performs better than ResNet18. But the test accuracy of ResNet50 by t-statistics is less than that of ResNet18.

In table 1, beyond that our algorithm can raise accuracy for ResNet50, we also show that our method is better than the classic approaches which consider variance like t-statistics.

5 CONCLUSIONS

This paper has studied the phenomenon that the larger model causes lower classification accuracy under DP. To illustrate our idea, we have considered a simple model for analysis. When noise or dimension tends to infinity, the classifier using all features performs nearly the same as random guessing. Hence it is necessary to find a method to reduce the dimension of the data. Based on a robust distance criterion, we can select the important features with probability one. Finally, we propose DFS algorithm to trade off the classification accuracy and privacy-preserving. Simulation reveals that the proposed DFS algorithm enjoys better performance on the real data.

6 RELATED WORK

Differential Privacy: In Xu et al. (2019), it considers both input-DP which adds noise on data processing, and output-DP which perturbs the answer of questions, and propose practical algorithms to show how to deal with two DP mechanisms. For the complicated situation like neural network, DP-SGD has been proved in utility (Chen et al. (2020)) with bounds for convergence after clipping gradient. Considering dimensions, Bassily et al. (2014) points that under assumptions of loss function and parameters, empirical risk can degenerate with dimension increment under differential privacy. Recently on neural network, Tramèr and Boneh (2021) shows that linear models trained on handcrafted features significantly outperform neural networks for moderate privacy budgets. However, they did not consider and set experiment for the affect of the dimension for the same type of model with accuracy instead of empirical risk.

High Dimension Low Sample Size Data: In low sample size n and high dimension p, Hall et al. (2005) studies the impact of the increasing n with fixed p, and they propose a geometric representation method for high-dimension data. For a linear model, Tsybakov (2003) propose a similar assumption with our condition and achieve a risk bound. For a neural network, Liu et al. (2017) propose DNP network to train on low sample by dropouts. DNP trains model by dropping neutrons randomly to minimize model size to increase model stability. Their works are powerful but in clean data, not concerning about privacy which people concerns.

Feature Selection: There are many traditional methods like wrapper and filter (Hart et al. (2000)) to select 'important' features for the clean data. Also, for neural network, an approach named pruning (Han et al. (2015)) come out for cutting neurons in network for maintaining low dimension of a model. Considering utility, the robustness of selection has been considered in Ilyas et al. (2019). They propose an algorithm to separate features with robustness in a certain model by adversary perturbation: changing labels for classes. However, their work either bases on clean data or adversary perturbation, which is not suitable for DP.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016.

R. Bassily, A. Smith, and A. Thakurta. Differentially private empirical risk minimization: Efficient algorithms and tight error bounds. *Computer Science*, 2014.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Xiangyi Chen, Steven Z Wu, and Mingyi Hong. Understanding gradient clipping in private sgd: a geometric perspective. *Advances in Neural Information Processing Systems*, 33, 2020.

Xue-Wen Chen and Xiaotong Lin. Big data deep learning: Challenges and perspectives. *IEEE Access*, 2:514–525, 2014.

Christophe Dupuy, Radhika Arava, Rahul Gupta, and Anna Rumshisky. An efficient dp-sgd mechanism for large scale nlp models. *arXiv preprint arXiv:2107.14586*, 2021.

Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Jianqing Fan and Yingying Fan. High dimensional classification using features annealed independence rules. *Annals of Statistics*, 36(6):2605, 2008.

Benjamin CM Fung, Ke Wang, Rui Chen, and Philip S Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (Csur)*, 42(4):1–53, 2010.

Mehdi Gheisari, Guojun Wang, and Md Zakirul Alam Bhuiyan. A survey on deep learning in big data. In *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, volume 2, pages 173–180. IEEE, 2017.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Miha Grčar, Dunja Mladenič, Blaž Fortuna, and Marko Grobelnik. Data sparsity issues in the collaborative filtering framework. In *International Workshop on Knowledge Discovery on the Web*, pages 58–76. Springer, 2005.

Peter Hall, James Stephen Marron, and Amnon Neeman. Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):427–444, 2005.

Song Han, Jeff Pool, John Tran, and William J Dally. Learning both weights and connections for efficient neural networks. *arXiv preprint arXiv:1506.02626*, 2015.

Peter E Hart, David G Stork, and Richard O Duda. *Pattern classification*. Wiley Hoboken, 2000.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer vVision and Pattern Recognition*, pages 770–778, 2016.

Jianping Hua, Waibhav D Tembe, and Edward R Dougherty. Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*, 42(3):409–424, 2009.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

Bo Liu, Ying Wei, Yu Zhang, and Qiang Yang. Deep neural networks for high dimension, low sample size data. In *International Joint Conference on Artificial Intelligence*, pages 2287–2293, 2017.

Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

A. B. Tsybakov. Optimal rates of aggregation. *Digital Bibliography & Library Project*, 2003.

Yahong Xu, Geng Yang, and Shuangjie Bai. Laplace input and output perturbation for differentially private principal components analysis. *Security and Communication Networks*, 2019, 2019.