# Randomness Beyond Noise: Differentially Private Optimization Improvement through Mixup

**Hanshen Xiao**
MIT
hsxiao@mit.edu

**Srinivas Devadas**
MIT
devadas@mit.edu

## Abstract

Information-theoretical privacy relies on randomness. Representatively, Differential Privacy has emerged as the gold standard to quantify the individual privacy preservation provided by given randomness. However, almost all randomness in existing differentially private optimization and learning algorithms is restricted to noise perturbation. In this paper, we point out another simple randomization technique, *mixup*: a random linear combination of inputs, with applications in (locally) private (decentralized) optimization and learning. Our contributions are twofold: first, we provide a rigorous analysis on the privacy amplification provided by *mixup*; second, both empirically and theoretically, we show that proper *mixup* comes almost free of utility compromise in optimization.

## 1 Introduction

Differential privacy (DP) has emerged as the standard measure of the individual-level privacy risk during an aggregate analysis on a dataset. Informally, a differentially private algorithm maps any two close datasets to similar probability distributions over outputs and thus, from outputs observed, it is hard to distinguish the participation of an individual. Initially, in Dwork et al.'s pioneering work [1], such indistinguishability is parameterized by a positive real number $\epsilon$ in a multiplicative manner:

**Definition 1** (Pure $\epsilon$-DP)**.** *A randomized algorithm $\mathscr{A}: \mathcal{X}^* \rightarrow \mathcal{O}$, achieves $\epsilon$-DP if for any adjacent datasets $\mathcal{D}$ and $\mathcal{D}'$ in $\mathcal{X}^*$, and any set $S$ in the output domain $\mathcal{O}$ of $\mathscr{A}(\cdot)$,*

$$\Pr[\mathscr{A}(\mathcal{D}) \in S] \leq e^{\epsilon} \Pr[\mathscr{A}(\mathcal{D}') \in S]. \tag{1}$$

Here, we call two datasets $\mathcal{D}$ and $\mathcal{D}'$ adjacent if $\mathcal{D}$ and $\mathcal{D}'$ only differ in one data point, denoted by $\mathcal{D} \sim \mathcal{D}'$ in the following. Stemming from (1), there is a long line of works to relax the original metric to measure the difference between the distributions of $\mathscr{A}(\mathcal{D})$ and $\mathscr{A}(\mathcal{D}')$ in Definition 1, for example, $(\epsilon, \delta)$-DP [2], where under the same setup a failure probability at most $\delta$ of (1) is admitted:

$$\Pr[\mathscr{A}(\mathcal{D}) \in S] \leq e^{\epsilon} \Pr[\mathscr{A}(\mathcal{D}') \in S] + \delta.$$

Other variants include concentrated DP [3, 4, 5], Renyi DP [6] and the recently proposed f-DP [7, 8]. Those relaxations provide versatile frameworks to analyze a larger class of randomized algorithms with tighter bounds to handle composition, i.e., the cumulative privacy risk under repetition of mechanisms on one dataset. However, compared to sharpened composition control, another important issue usually gets overlooked is **how to introduce randomness for privacy preservation?**

**Randomness beyond Noise**: The simplest way to randomize an algorithm is perturbation. For example, to make a deterministic algorithm $\mathscr{A}$ satisfy $\epsilon$-DP, one can add Laplace noise in a scale of the sensitivity, i.e., $\max_{\mathcal{D},\mathcal{D}'} \|\mathscr{A}(\mathcal{D}) - \mathscr{A}(\mathcal{D}')\|$, to the output [1]. In general, DP does not come for free: lower bounds of utility loss in many tasks are known, for example, (strongly) convex optimization [9, 10] and Principal Components Analysis (PCA) [11, 12], etc. However, this is not an end to the study on the efficiency of randomness more practically in an non-asymptotic view.

Preprint. Under review.

Though DP is not free of utility loss, it does **not** mean randomness will always come with a performance compromise. The purposes to introduce randomness in optimization and learning are far more than privacy, for example, stochastic gradient Langevin dynamics (SGLD) [13, 14] for nonconvex optimization, uniform noise perturbed gradient descent to escape saddle points [15]. Randomness can even strengthen the training performance such as random dropout [16] and data augmentation [17]. Generally speaking, data augmentation represents a large class of methods to improve robustness and reduce memorization (instead of generalization), especially in neural nets: Training is conducted on similar but different virtual examples compared to the raw data through random cropping [18], erasing [19] and mixup [20], etc. However, compared to simple noise perturbation, those algorithm-oriented randomnesses do **not** always lead to a DP guarantee. To this end, a natural approach is a hybrid structure of both kinds of randomness, for example, Laplace noise and *mixup*.

**Mixup**: In this paper, *mixup* denotes the simple aggregation structure with random weights. Given $N$ inputs $x_1, x_2, ..., x_N$, *mixup* outputs $\sum_{i=1}^{N} \omega_i x_i$, with random $\omega_i \in (0,1)$ and $\sum_{i=1}^{N} \omega_i = 1$. One successful example of *mixup* is [20], where a surprisingly simple data augmentation is described: Given the raw data $(b_i, y_i)$, $i = 1, 2, ..., n$, where $b_i$ is the observation and $y_i$ is the associated label, a virtual training sample $(\tilde{b}, \tilde{y})$ is constructed where

$$\tilde{b} = \lambda b_{i_1} + (1-\lambda) b_{i_2}, \tilde{y} = \lambda y_{i_1} + (1-\lambda) y_{i_2}. \tag{2}$$

Here, $(b_{i_1}, y_{i_1})$ and $(b_{i_2}, y_{i_2})$ are randomly drawn while $\lambda \in (0,1)$ is a random variable selected from a Beta distribution. Though *mixup* based data augmentation has been shown to be powerful in thorough experiments and subsequent works, it is an empirical result. This raises two interesting questions: **On privacy, what kind of privacy amplification is provided by *mixup*? On utility, what theoretical performance guarantees can we provide about the applications of *mixup*?** We set out to answer the two questions.

## 2 Hybrid Architecture of Mixup and Noise

Differentially private (Stochastic) Gradient Descent ((S)GD) and its variants have been extensively studied[9, 21, 22, 23, 24, 25, 26]. A common strategy is to perturb the gradient in each iteration with well-scaled noise to keep track of the cumulative privacy loss. In the following, we describe two scenarios to apply *mixup*. The first is a continuing analysis from (2): Imagine we run SGD on the empirical loss of samples $\mathcal{S} = \{\boldsymbol{s}_i, i = 1, 2, ..., n\}$ with *mixup*, where for simplicity we use $\boldsymbol{s}_i$ to denote $(b_i, y_i)$. At iteration $k$, the protocol to privately update $\boldsymbol{x}$ becomes:

$$\boldsymbol{x}^k = \boldsymbol{x}^{k-1} - \eta_k \underline{\nabla f(\boldsymbol{x}^{k-1}, \lambda_k \boldsymbol{s}_{i_{k,1}} + (1-\lambda_k)\boldsymbol{s}_{i_{k,2}})} + \boldsymbol{\Delta}^k. \tag{3}$$

Here, $i_{k,1}$ and $i_{k,2}$ are two random indexes sampled from $[1:n]$, $\lambda_k$ is randomly selected from $(0,1)$, $f(\cdot)$ denotes the loss function selected and $\boldsymbol{\Delta}^k$ is the noise added in iteration $k$. *Mixup* can also be applied to the aggregation of parameter $\boldsymbol{x}$. Consider a distributed optimization of $N$ agents, where the goal is to collaboratively minimize the sum of their loss functions $\sum_{i=1}^{N} f_i(\boldsymbol{x})$. GD can also be generalized into a distributed form where the updating protocol of agent $i$ becomes

$$\boldsymbol{x}_i^k = \underline{\sum_{i=1}^{N} w_{ij}^k \boldsymbol{x}_j^{k-1}} - \eta_k \nabla f_i(\boldsymbol{x}_i^k) + \boldsymbol{\Delta}_i^k \tag{4}$$

Here, $w_{ij} \in [0,1]$ is the weight assigned to $\boldsymbol{x}_j^k$ such that $\sum_{j=1}^{N} w_{ij} = 1$. Now we compare (3) and (4). Assuming the noise follows a Laplace distribution, the distribution of produced updates $\boldsymbol{x}^k$ from (3) and (4) have something in common: resorting to the random weights in *mixup*, the underlining parts in (3) and (4) are randomly distributed in a convex hull of samples $\boldsymbol{s}_i$ and earlier updates $\boldsymbol{x}_i^{k-1}$, respectively. Thus, the distribution of $\boldsymbol{x}^k$ ($\boldsymbol{x}_i^k$) is indeed a mixture of a Laplace noise and a bounded random variable. It is well-known that the Laplace Mechanism can provide pure $(\epsilon, 0)$-DP guarantee. As for the above-mentioned hybrid structure, we have the following observation: The additional randomness from *mixup* makes the mixture distribution more smooth and under the same setup, a pure DP of same $\epsilon$ is provided at the very least, compared to the pure Laplace Mechanism.

But what is the exact privacy amplification from *mixup*? To formalize the extra gain, we introduce an alternative definition: *ex-post* local privacy loss. The privacy loss $\epsilon(o)$ for an algorithm $\mathscr{A}$

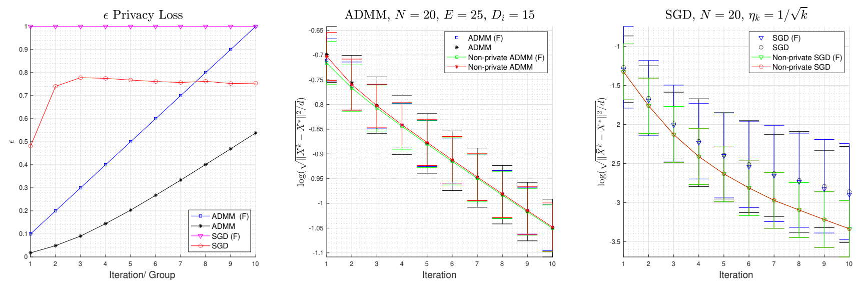

Figure 1: Comparison between ADMM & SGD (with *mixup*), ADMM & SGD (F) (without *mixup*) and their non-private versions without Laplace noise perturbation on logistic regression over $Adult$.

on an output $o$ is defined as $\mathbb{P}(\mathscr{A}(\mathcal{D}) = o) \leq e^{\epsilon(o)}\mathbb{P}(\mathscr{A}(\mathcal{D}') = o)$. Pure $\epsilon$-DP is equivalent to $\sup_o \epsilon(o) \leq \epsilon$.

To compare the pure Laplace Mechanism and the proposed hybrid structure, without loss of generality, we consider the following two distributions: The first is a pure Laplace distribution $\mathrm{Lap}(0,\beta)$ with probability density $\mathbb{P}(z) = \beta/2e^{-\beta|z|}$; The other is the sum of $\mathrm{Lap}(0,\beta)$ and an independent uniform distribution within $[0,\omega]$ denoted by $U[0,\omega]$, whose probability density is then a convolution of $\mathrm{Lap}(0,\beta)$ and $U[0,\omega]$. Given sensitivity bound $\mathscr{B}$, a pure Laplace Mechanism provides a $(\beta\mathscr{B},0)$-DP. Specifically, let $\epsilon_p(o)$ denote the privacy loss from a pure Laplace Mechanism. It is not hard to verify that $\epsilon_p(o)$ is a constant equaling $\beta\mathscr{B}$ for arbitrary $o$. In comparison, the privacy loss $\epsilon_m(o)$ from the mixture distribution shares the same worst case, i.e, $\epsilon_m(o) \leq \beta\mathscr{B}$, but once $\omega > 0$, there always exists some $\hat{o}$ of a strictly smaller privacy loss, i.e., $\epsilon_m(\hat{o}) < \beta\mathscr{B}$. We provide the following upper bound of privacy loss $\epsilon_m(o)$ from the mixture distribution $\mathrm{Lap}(0,\beta) * U[0,\omega]$.

**Theorem 2.1.** *With sensitivity bound $\mathcal{B}$, under the distribution $Lap(0,\beta) * U[0,\omega]$,*

$$\epsilon_m(o) \leq \max_{t=\pm\mathscr{B}} \left| \log\left[\int_0^{\omega} e^{-\beta|o-x|}dx\right] - \log\left[\int_t^{t+\omega} e^{-\beta|o-x|}dx\right]\right|. \tag{5}$$

To have a clearer picture of how much privacy loss is saved from *mixup*, we consider the ratio between the privacy loss of the hybrid structure and the pure Laplace Mechanism, i.e.,

$$\gamma(o) = \frac{\epsilon_m(o)}{\epsilon_p(o)} = \frac{\epsilon(o)}{\beta\mathscr{B}} \leq 1.$$

The following Theorem shows such privacy amplification from *Mixup* in expectation.

**Theorem 2.2.** *Let $\Phi(x,z) = \frac{\beta}{2\omega}e^{-\beta|x-z|}$, when $\omega > \mathscr{B}$, we have*

$$\mathbb{E}_{o\sim Lap(0,\beta)*U[0,\omega]}[\gamma(o)] \leq \frac{1}{\beta\mathscr{B}} \log\left\{2\int_0^{\frac{\omega-\mathscr{B}}{2}}\int_{-\mathscr{B}}^{\omega-\mathscr{B}} \Phi(x,z)dxdz + e^{\beta\mathscr{B}}\left[1-2\int_0^{\frac{\omega-\mathscr{B}}{2}}\int_0^{\omega}\Phi(x,z)dxdz\right]\right\}.$$

Following this idea, we incorporate ADMM and SGD with *mixup*. The detailed description of the two modified algorithms can be found in the full version of our paper. We test such ADMM and SGD with the hybrid randomization structure and their corresponding variants with pure Laplace Mechanism on logistic regression of the *Adult* dataset (from the UCI machine learning repository [27]), shown in Fig.1. For simplicity, in the following, $(F)$ denotes the latter case. In the experiment of ADMM, the communication graphs are randomly generated across 100 trials, where $N$ and $E$ are the number of agents and edges amongst them, respectively. In addition, we assume each agent holds a dataset of size 1000 and 200 in ADMM and SGD, respectively. A full description and similar tests over synthetic datasets (including those generated from heavy-tailed distribution) is included in the full version. One of the key observations is, *with the same setup, the hybrid randomization achieves almost the same utility loss in optimization accuracy as that of the regular Laplace Mechanism*. For the privacy side, the same Laplace noise is applied in both cases where we fix the $\epsilon$ privacy budget to be 1 to guarantee the same worst case. *mixup* renders a sharpened privacy amplification, where the privacy loss is reduced empirically ranging from 30% to 50%, and performs even better when the graph is sparser. Also, earlier iterations enjoy better privacy amplification since the divergence amongst $\boldsymbol{x}^k_{[1:N]}$ (or $\boldsymbol{y}^k_{[1:2]}$ in Algorithm 2) is larger. This is consistent with Theorem 2.2 where a larger interval length $\omega$ renders smaller $\gamma$.

## 3 Utility Analysis from A Random Stochastic Matrix View

In the full version, under proper parameter selection, we prove ADMM and SGD with hybrid *mixup* structure achieve an asymptotically tight optimal utility bound [28]. In the following, we aim to explain why *mixup* is almost free of utility compromise in a *non-asymptotic view*. In general, the framework of the proposed private Distributed GD can be expressed as

$$\boldsymbol{X}_{k+1} = W_{k+1}\boldsymbol{X}_k - \xi_{k+1}\nabla F(\mathbb{E}[W_{k+1}]\boldsymbol{X}_k) + \boldsymbol{\Delta}^{k+1}, \tag{6}$$

Here $\boldsymbol{X}_k = (\boldsymbol{x}_1^k, \boldsymbol{x}_2^k, ..., \boldsymbol{x}_N^k)$, $\boldsymbol{x}_i^k \in \mathbb{R}^d$, and $F(\boldsymbol{X}_k) = \sum_{i=1}^N f_i(\boldsymbol{x}_i^k)$, and accordingly $\nabla F(\boldsymbol{X}_k) = (\nabla f_1(\boldsymbol{x}_1^k), ..., \nabla f_N(\boldsymbol{x}_N^k))$. Recalling $w_{ij}$ in (4), $W_{k+1}(i,j) = w_{ij} \cdot \boldsymbol{I}$ is a random stochastic matrix determined by the weights selected by each agent. Here, $\boldsymbol{I}$ is the $d \times d$ identity matrix and $W(i,j)$ denotes the element of $W$ at the crossing of the $i^{th}$ row and $j^{th}$ column (in a block sense). When $\mathbb{E}[W_k]$ is doubly stochastic, we have the following upper bound of utility-privacy tradeoff for (6),

**Theorem 3.1.** *Selecting $\xi_k = O(\frac{1}{\sqrt{k}})$, when $f_{[1:N]}(\cdot)$ are L-Lipschitz, and $\mathbb{E}[W_k]$ is doubly stochastic,*

$$|F(\frac{\sum_{k=0}^{K-1}\boldsymbol{X}_k}{K}) - F(\boldsymbol{X}_*)| \leq \frac{\bar{\mathcal{R}}}{\sqrt{K}} + L\sum_{i=1}^{N}\mathbb{E}[\mathcal{T}_i^K] \tag{7}$$

*where $W_0 = \boldsymbol{I}$, $\bar{\mathcal{R}}$ is a term invariant to the randomness of $W_k$, specified in the full version, and* $\mathcal{T}_i^K = \|\frac{\sum_{k=0}^{K-1}\sum_{l=1}^{N}\mathbb{E}[W_{k+1}(l,:)]^T\boldsymbol{X}_k}{KN} - \frac{\sum_{k=0}^{K-1}\mathbb{E}[W_{k+1}(i,:)]^T\boldsymbol{X}_k}{K}\| + \|\frac{\sum_{k=0}^{K-1}(\boldsymbol{x}_i^k - \mathbb{E}[W_{k+1}(i,:)]^T\boldsymbol{X}_k)}{K}\|$.

Theorem 3.1 indicates that, to study the utility loss, it suffices to consider *the rate of $\boldsymbol{X}_k$ towards the consensus*, i.e., $\boldsymbol{x}_i^k = \boldsymbol{x}_j^k$ for $i,j \in [1:N]$, where the deviation amongst $\boldsymbol{X}_k$ controls $\mathcal{T}_i^K$ on the right hand of (7). This is consistent with intuition. In the non-private case, where the convergence proofs in [29, 30] guarantee $\boldsymbol{X}_k$ approaches the unique consensus optima $\boldsymbol{X}_*$, the excess loss is then proportional to the divergence among $\boldsymbol{X}_k$ in expectation. For quantification, we introduce the following metric $\phi(\boldsymbol{X})$, which denotes the largest deviation between any two elements of $\boldsymbol{X}$ in $l_2$ norm. For example, $\phi(\boldsymbol{X}_k) = \max_{i,j}\|\boldsymbol{x}_i^k - \boldsymbol{x}_j^k\|$. With the above understanding, we move our focus to $\phi(\sum_{k=0}^{K-1}\boldsymbol{X}_k/K)$. To proceed, we rewrite $\sum_{k=0}^{K-1}\boldsymbol{X}_k/K$ in the following form,

$$\sum_{k=0}^{K-1}\boldsymbol{X}_k/K = \Big(\sum_{k=0}^{K-1}\Big(\prod_{j=1}^{k}W_j\boldsymbol{X}_0 + \sum_{j=1}^{k}\prod_{l=j+1}^{k}W_lR_j\Big)\Big)/K \tag{8}$$

where for simplicity we rewrite $\boldsymbol{X}_{k+1} = W_{k+1}\boldsymbol{X}_k + R_{k+1}$ for some remainder term $R_{k+1}$ and $\prod_{l=j}^{k}W_k = \boldsymbol{I}$ if $j > k$. Now, we measure the impact of random aggregation, i.e., random stochastic $W_k$ applied in (6), on the consensus rate of $\sum_{k=0}^{K-1}\boldsymbol{X}_k/K$, compared to the fixed $W_k = W$ case, where $W(i,j) = \frac{1}{N}\cdot\boldsymbol{I}$. The justification of this choice is as follows. Such fixed weight matrix $W$ with identical rows has the property that for any $\boldsymbol{X}$, elements in $W\boldsymbol{X}$ are identical, i.e., $\phi(W\boldsymbol{X}) = 0$. Let $W_k = W$ in (8), then all the terms that are a multiple of $W$ reach consensus and thus $\phi(\sum_{k=0}^{K-1}\boldsymbol{X}_k/K) = \phi((\boldsymbol{X}_0 + \sum_{k=1}^{K-1}R_k)/K)$. In contrast, for randomized $W_k$, those terms of multiples of $W_k$, such as $\prod_k W_k\boldsymbol{X}_0$ in (8), do not necessarily reach consensus, which produces the gap between the *mixup* and fixed weight cases. For simplicity, we assume $N$ is even and consider the following way to randomize $W_k$: $W_k(i,j) = r_i^k \times \frac{2}{N}\cdot\boldsymbol{I}$ if $j \leq \frac{N}{2}$, otherwise $W_k(i,j) = (1 - r_i^k)\times\frac{2}{N}\cdot\boldsymbol{I}$. $r_i^k$ are i.i.d. random variables in $(0,1)$. Clearly, $\mathbb{E}[W_{k+1}] = W$. Then, we have:

**Theorem 3.2.** *Under $(\epsilon,\delta)$-Local DP and sensitivity in infinity norm bounded by $\mathscr{B}_\infty$, for $W_k$ fixed to $W$, $\phi(\sum_{k=0}^{K-1}\boldsymbol{X}_k/K) = O(\frac{1}{\sqrt{K}} + \frac{1}{K} + \frac{d\mathscr{B}_\infty}{\epsilon})$; as for randomized $W_k$ described above, $\phi(\sum_{k=0}^{K-1}\boldsymbol{X}_k/K) = O(\frac{1}{\sqrt{K}} + \frac{1}{K} + (1 + \frac{1}{\sqrt{N}})\frac{d\mathscr{B}_\infty}{\epsilon})$. Here, $d$ the dimensionality of $\boldsymbol{x}_i$.*

Theorem 3.2 shows that, with either fixed or randomized $W_k$, the consensus distance in the average $\sum_{k=0}^{K-1}\boldsymbol{X}_k/K$ measured with $\phi$ gradually approaches $d\mathscr{B}/\epsilon$ as $N$ and $k$ increase. Such a bound is consistent with our experimental observations.

As a conclusion, in this paper we take *mixup* as a concrete example to show how randomness beyond noise perturbation can be used to amplify privacy. Though *mixup* itself cannot provide a nontrivial DP guarantee, we provide rigorous analysis to quantify the privacy gain in a hybrid structure of *mixup* and a Laplace Mechanism. In addition, we develop a series of utility studies, which explains that why *mixup* is almost free of compromise in optimization. We believe the techniques developed may be of independent interest in robust optimization with Byzantine faults.

## References


[1] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[2] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer, 2006.

[3] Cynthia Dwork and Guy N Rothblum. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887*, 2016.

[4] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pages 635–658. Springer, 2016.

[5] Mark Bun, Cynthia Dwork, Guy N Rothblum, and Thomas Steinke. Composable and versatile privacy via truncated cdp. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 74–86, 2018.

[6] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[7] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

[8] Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. Deep learning with gaussian differential privacy. *arXiv preprint arXiv:1911.11607*, 2019.

[9] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473. IEEE, 2014.

[10] Kunal Talwar, Abhradeep Guha Thakurta, and Li Zhang. Nearly optimal private lasso. In *Advances in Neural Information Processing Systems*, pages 3025–3033, 2015.

[11] Kamalika Chaudhuri, Anand Sarwate, and Kaushik Sinha. Near-optimal differentially private principal components. In *Advances in Neural Information Processing Systems*, pages 989–997, 2012.

[12] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 11–20, 2014.

[13] Wenlong Mou, Liwei Wang, Xiyu Zhai, and Kai Zheng. Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints. *Proceedings of Machine Learning Research vol*, 75:1–34, 2018.

[14] Bai Li, Changyou Chen, Hao Liu, and Lawrence Carin. On connecting stochastic gradient mcmc and differential privacy. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 557–566, 2019.

[15] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1724–1732. JMLR. org, 2017.

[16] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[17] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[19] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. 2020.

[20] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[21] Katrina Ligett, Seth Neel, Aaron Roth, Bo Waggoner, and Steven Z Wu. Accuracy first: Selecting a differential privacy level for accuracy constrained erm. In *Advances in Neural Information Processing Systems*, pages 2566–2576, 2017.

[22] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[23] Prateek Jain and Abhradeep Thakurta. Differentially private learning with kernels. *Journal of Machine Learning Research*, 2013.

[24] Di Wang, Marco Gaboardi, and Jinhui Xu. Empirical risk minimization in non-interactive local differential privacy revisited. In *Advances in Neural Information Processing Systems*, pages 973–982, 2018.

[25] Di Wang, Minwei Ye, and Jinhui Xu. Differentially private empirical risk minimization revisited: Faster and more general. In *Advances in Neural Information Processing Systems*, pages 2722–2731, 2017.

[26] Di Wang and Jinhui Xu. Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view. In *AAAI Conference on Artificial Intelligence*, 2019.

[27] Xueru Zhang, Mohammad Mahdi Khalili, and Mingyan Liu. Improving the privacy and accuracy of ADMM-based distributed algorithms. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5796–5805, 10–15 Jul 2018.

[28] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. In *Advances in neural information processing systems*, pages 2879–2887, 2014.

[29] Ali Makhdoumi and Asuman Ozdaglar. Convergence rate of distributed admm over networks. *IEEE Transactions on Automatic Control*, 62(10):5082–5095, 2017.

[30] Wei Shi, Qing Ling, Kun Yuan, Gang Wu, and Wotao Yin. On the linear convergence of the admm in decentralized consensus optimization. *IEEE Trans. Signal Processing*, 62(7):1750–1761, 2014.