CAN STOCHASTIC WEIGHT AVERAGING IMPROVE GENERALIZATION IN PRIVATE LEARNING?

Patrick Indri*, **Tamara Drucks*** **& Thomas Gärtner**
Research Unit Machine Learning, TU Wien, Vienna, Austria
{patrick.indri, tamara.drucks, thomas.gaertner}@tuwien.ac.at

ABSTRACT

We investigate stochastic weight averaging (SWA) for private learning in the context of generalization and model performance. Differentially private (DP) optimizers are known to suffer from reduced performance and high variance in comparison to non-private learning. However, the generalization properties of DP optimizers have not been studied much, in particular for large-scale machine learning models. SWA is variant of stochastic gradient descent (SGD) which averages the weights along the SGD trajectory. We consider a DP adaptation of SWA (DP-SWA) which incurs no additional privacy cost and has little computational overhead. For quadratic objective functions, we show that DP-SWA converges to the optimum at the same rate as non-private SGD, which implies convergence to zero for the excess risk. For non-convex objective functions, we observe throughout multiple experiments on standard benchmark datasets that averaging model weights improves generalization, model accuracy, and performance variance.

1 INTRODUCTION

Machine learning models that provide guarantees against data leakage are critical for many applications as they can be more safely trained on sensitive data. Differential privacy (DP; Dwork et al., 2014) offers formal privacy guarantees and protects information about individual training points. Differentially private stochastic gradient descent (DP-SGD), first introduced by Abadi et al. (2016), is the state-of-the-art algorithm to privately train models using SGD. Models trained with DP-SGD, however, suffer from a degradation of performance and often provide less stable solutions (Wang et al., 2021). Moreover, although DP is known to enjoy good generalization properties (Wang et al., 2016), prior work has shown that privately training larger models can be detrimental to generalization (Papernot et al., 2021). In non-private settings, Izmailov et al. (2018) demonstrated empirically that stochastic weight averaging (SWA), i.e., aggregating SGD iterates as originally proposed in Polyak & Juditsky (1992), leads to wider optima, better generalization, and improved performance. Only few research efforts have been made to analyze the impact of weight averaging in DP settings. De et al. (2022) first consider it for large-scale vision tasks, and Shejwalkar et al. (2022) explore different weight averaging techniques for improving model accuracy. Albeit convergence guarantees for averaging schemes for SGD have been previously analyzed (Shamir & Zhang, 2013), there is little theoretical investigation on the benefits of weight averaging for DP-SGD.

Motivated by the lack of theoretical and experimental results on SWA in a private setting, we investigate whether aggregating DP-SGD iterates with SWA (DP-SWA) improves generalization and model performance. Specifically, we consider the wideness of solutions as a proxy for generalization (Cha et al., 2021), and study whether these effects can be observed across different scales of models. We perform several case studies for the optimization of machine learning models with DP-SWA: for quadratic objective functions, we show that DP-SWA converges to the optimum at the same rate as standard SGD, thus showing that the excess risk converges to zero as $1/T$ for T iterations; for strongly convex functions, we empirically corroborate existing convergence bounds for suffix averaging in DP-SGD (Shejwalkar et al., 2022) with logistic regression experiments on Fashion-MNIST; for non-convex objective functions, we experiment with large-scale vision models

*Equal contribution.

on common benchmark datasets (Fashion-MNIST, CIFAR-10, CIFAR-100). We find that DP-SWA yields wider and more stable solutions with higher accuracy at a negligible computational impact.

2 PRELIMINARIES

In this section we recall the definition of differential privacy (DP; Dwork et al., 2014) and present DP-SWA, a DP adaptation of stochastic weight averaging (SWA; Izmailov et al., 2018).

Differential privacy DP offers formal privacy guarantees about individual training points and is defined in terms of *adjacent* datasets. Two datasets are adjacent if they differ in a single point, that is, if one point is present in one dataset but not in the other. In the following, we use the notion of (ϵ, δ)-DP as introduced in Dwork et al. (2006). Consider a randomized algorithm $\mathcal{M} : \mathcal{D} \rightarrow \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$. $\mathcal{M}$ satisfies (ϵ, δ)-DP if for any two adjacent datasets $D, D' \in \mathcal{D}$ and for any subset $S \subseteq R$ the following holds:

$$\mathbb{P}\left[\mathcal{M}(D) \in S\right] \leq e^{\epsilon}\, \mathbb{P}\left[\mathcal{M}(D') \in S\right] + \delta$$

We refer to ϵ as the *privacy budget* of the algorithm. DP can be achieved by adding noise to a function with bounded sensitivity, i.e., where the difference between function values on adjacent inputs D and D' is bounded. Following this approach, Abadi et al. (2016) propose DP-SGD, a DP version of SGD, where the sensitivity of gradients is bounded by clipping them up to a maximum norm C, which is referred to as the *clipping norm*. See Appendix A for more details on DP-SGD. It is important to note that any operation performed on the outputs of a private algorithm $\mathcal{M}$ without additional access to the private dataset does not worsen privacy guarantees of $\mathcal{M}$ (Dwork et al., 2014). We refer to this as the *post-processing* property of DP.

DP-SWA SWA takes averages of multiple weights along the SGD trajectory and has been shown to improve performance and generalization of machine learning models optimized with SGD (Izmailov et al., 2018). To adapt SWA for private learning, we consider DP-SWA, displayed in Algorithm 1, where gradient steps are taken using DP-SGD. Thanks to the post-processing property of DP, DP-SWA can aggregate intermediate DP-SGD weights withough additional privacy cost. Specifically, DP-SWA averages intermediate weights every c steps; we refer to c as the *cycle length* of DP-SWA. In practical applications, it is recommended to start averaging after a warm-up phase where the optimization procedure approaches an optimum (Izmailov et al., 2018; Panda et al., 2022).

Algorithm 1 DP-SWA, DP adaptation of SWA from Izmailov et al. (2018)

Input: initial weights θ_0, cycle length c, number of iterations T, learning rate α, loss function f, clipping norm C, noise scale σ
Output: $\bar{\theta}_T$

```
θ̄_T ← θ_0
for i ∈ 1 ... T do
    b_i ~ N(0, σ²)
    θ_i ← θ_{i−1} − α(clip_C ∇f(θ_{i−1}) + b_i)    (DP-SGD)
    if c = 0 (mod i) then
        n_models ← i/c    (number of models)
        θ̄_T ← (θ̄_T · n_models + θ_i) / (n_models + 1)    (update average)
    end if
end for
return θ̄_T
```

3 CONVERGENCE FOR QUADRATIC AND CONVEX OBJECTIVE FUNCTIONS

In this section, we theoretically investigate DP-SWA for (i) quadratic and (ii) convex objective functions. For (i), we provide an upper bound on the convergence of the parameter estimates, for (ii) we discuss existing results for weight averaging in a private setting.

Quadratic objective functions Consider DP-SWA with cycle length c and T iterations for the optimization of a quadratic objective function with d-dimensional parameters. We show that the distance between the output of DP-SWA and the optimum goes to zero in expectation as $\mathcal{O}(cd/T\epsilon^2)$.

More formally, let f be a quadratic objective function of the form $f(\theta) = (\theta - \theta^*)^T A(\theta - \theta^*)/2$ with $A \succeq \mu I \in \mathbb{R}^{d\times d}$ for some $\mu > 0$ and let θ^* be the optimum of f. Assume that the sample gradients $\nabla \tilde{f}(\theta)$ are equal to the full gradient in expectation, i.e., $\mathbb{E}[\nabla \tilde{f}(\theta)] = \nabla f(\theta) = A(\theta - \theta^*)$. For simplicity, assume a compact domain for the parameters θ and choose the clipping norm to be the implied bound on the gradient norm, i.e., we perform no clipping. For the gradient updates, take $b \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$ with $\sigma^2 = \frac{2\log 1.25/\delta}{\epsilon^2}$ to guarantee DP (Dwork et al., 2014; Abadi et al., 2016). Finally, make the standard assumption that $\mathbb{E}\|\nabla \tilde{f}(\theta) - \nabla f(\theta)\|^2 \leq \varsigma^2$ for some $\varsigma > 0$.

Theorem 1 *Consider a cycle length c and a learning rate α with $0 < \alpha < \|A\|/2$. In expectation, the squared distance of the output $\bar{\theta}_T$ of DP-SWA at iteration T with respect to the optimum θ^* is upper bounded as:*

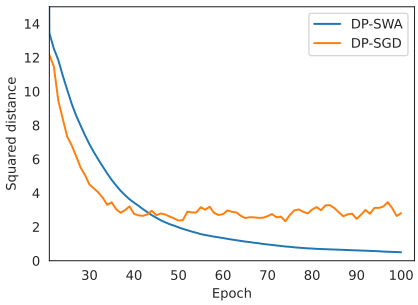
$$\mathbb{E}\|\bar{\theta}_T - \theta^*\|^2 \leq \frac{\|\theta_0 - \theta^*\|^2}{T^2\alpha^2\mu^2} + \frac{c}{T\alpha^2\mu^2}\left(\frac{2d\log\frac{1.25}{\delta}}{\epsilon^2} + \alpha^2\varsigma^2\right) \sim \mathcal{O}\left(\frac{cd}{T\epsilon^2}\right)$$

See Appendix A.2 for the proof, which is adapted from Yang et al. (2019). According to Theorem 1, DP-SWA converges to the optimal solution for a given privacy budget ϵ at the same convergence rate as standard SGD (Varre et al., 2021). It should be noted that the privacy budget ϵ is to be interpreted as the budget for each descent step. As it is often useful to consider the distance to the optimum after T iterations instead, we can also interpret our bound as a $\mathcal{O}(cd/\epsilon^2)$ utility bound after T iterations. In the following, we will consider linear regression as a case study to investigate the empirical behavior in comparison with this bound. For a similar problem setting (linear regression for bounded inputs and bounded moments), Cai et al. (2021) obtain an upper bound on the (non-averaged) last iterate parameters as $\mathbb{E}\|\theta_T - \theta^*\|^2 \leq \mathcal{O}(d^2/\epsilon^2)$. Varshney et al. (2022) obtain a nearly optimal utility bound on averaged iterates using an adaptive clipping technique, but do not consider different cycle lengths. We refer the reader to, e.g., Wang (2018) for a more detailed overview on research efforts in DP linear regression.

Convex objective functions In the non-private stochastic setting, Rakhlin et al. (2012) and Shamir & Zhang (2013) show that averaging all iterates provides only sub-optimal convergence rates in (strongly) convex optimization; however, the authors prove that optimal rates can be recovered by averaging a suffix of the iterates only. This approach corresponds to running SWA with cycle length $c = 1$ after a warm-up phase where no averages are collected. In a private setting and for convex objective functions, Shejwalkar et al. (2022) have recently shown a better upper bound on the excess empirical risk for suffix averaging, when compared to the last, non-averaged iterate. The authors do not, however, provide a lower bound on the excess risk, and thus no statement can be made about whether the last iterate bound could match the suffix averaging bound. Nevertheless, their result indicates that DP-SWA with $c = 1$ may be beneficial for convex optimization. As the authors do not perform experiments in the convex setting we empirically investigate this case with a logistic regression task.

4 EXPERIMENTS

In this section, we empirically corroborate our theoretical findings, and investigate whether weight averaging improves generalization in private learning for non-convex objective functions. Specifically, we (i) solve a linear regression task for quadratic optimization and validate Theorem 1; (ii) experiment on logistic regression for convex optimization; and (iii) conduct several experiments with *ResNet18*, *WRN-16-4*, and *ResNet50* on standard benchmark datasets. We focus on assessing the impact of DP-SWA on (a) generalization, (b) accuracy, and (c) stability of the solutions in comparison to DP-SGD. We measure generalization (a) as the average accuracy resulting from moving away from the weight vectors θ_T and $\bar{\theta}_T$ obtained by DP-SGD and DP-SWA, respectively; larger averages denote wider solutions. Specifically, we take steps along 10 random rays starting from θ_T and $\bar{\theta}_T$. We report the signed percentage gain of DP-SWA over DP-SGD as $\%_{\text{GAIN}}$, with positive values favoring DP-SWA. We measure accuracy (b) by means of distance to the optimum for linear



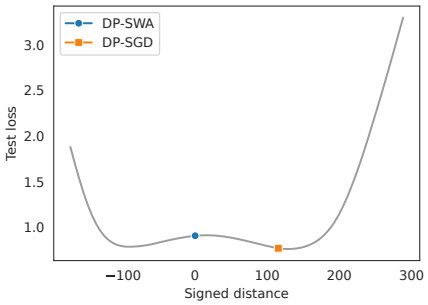


Figure 1: Visualizations for $\epsilon = 8$. (*Left*) Linear regression. DP-SWA is in agreement with Theorem 1 and compatible with convergence to the optimum at a rate (at least as fast as) $1/T$ (see also Appendix C.1). (*Right*) *ResNet-18* on Fashion-MNIST. The solution output by DP-SWA is preferable to DP-SGD with respect to wideness. The plot is obtained by computing the loss function on points which lay on the line connecting the DP-SGD and DP-SWA weights.

regression, and classification accuracy for all other tasks. For a given algorithm ALG, we measure stability (c) as the variance $\mathrm{Var}_{\mathrm{ALG}}$ of the accuracy during the last 10 iterations, or all available iterates if DP-SWA was run for less than 10 iterations. For all experimental settings, we use DP-SGD with constant learning rate as an optimizer and start averaging weights after an initial warm-up phase. We use the same values for the privacy parameters as De et al. (2022) and Shejwalkar et al. (2022): we run experiments with $\epsilon = \{1, 8\}$ and $\delta = 10^{-5}$, unless noted otherwise. Details on the experimental setup can be found in Appendix B.

Quadratic objective functions We use DP-SWA to solve a linear regression task with mean squared error loss on synthetic data and compare the empirical behavior with the bound we derived in Theorem 1; refer to Appendix B for details on data generation. Table 1 shows that DP-SWA performs better than DP-SGD, both in terms of distance to the optimum and stability. In line with our theoretical results, we experimentally observe the convergence of DP-SWA to the optimum in $\mathcal{O}(1/T)$ in Figure 1 (Left). Additional results in Appendix C.1 show the $\mathcal{O}(1/\epsilon^2)$ dependency and faster convergence with more frequent averages (i.e., smaller c).

Table 1: Squared distance to the optimum and squared distance variance over the last iterates for linear regression. Mean and standard deviation across 5 runs. Results for $\delta = 1/n^2$ and $n = 4096$ data points.

ϵ	$\lVert\theta_T - \theta^*\rVert^2_{(\text{DP-SGD})}$	$\lVert\bar{\theta}_T - \theta^*\rVert^2_{(\text{DP-SWA})}$	$\mathrm{Var}_{\text{DP-SGD}}$	$\mathrm{Var}_{\text{DP-SWA}}$
1	22 ± 2	$\mathbf{7.4 \pm 0.9}$	1.1 ± 0.5	$\mathbf{0.11 \pm 0.08}$
8	2.6 ± 0.2	$\mathbf{0.52 \pm 0.01}$	0.2 ± 0.1	$\mathbf{0.03 \pm 0.03}$

Convex objective functions To empirically investigate the results of Shejwalkar et al. (2022) for convex objective functions, we use DP-SWA with logistic regression on Fashion-MNIST. We find that DP-SWA offers significantly better accuracy and stability in comparison to DP-SGD (see Table 2); for instance, DP-SWA for $\epsilon = 1$ reaches an accuracy of 74.7%, which is not significantly different to the DP-SGD result for $\epsilon = 8$ of 75.3%. That is, DP-SWA obtains an accuracy similar to that of DP-SGD, but with stronger privacy guarantees. Additional results are in Appendix C.2.

Table 2: Accuracy and accuracy variance over last iterates for logistic regression. Mean and standard deviation across 5 runs.

ϵ	DP-SGD	DP-SWA	$\mathrm{Var}_{\text{DP-SGD}}$	$\mathrm{Var}_{\text{DP-SWA}}$
1	68.4 ± 0.3	$\mathbf{74.7 \pm 0.6}$	1.0 ± 0.2	$\mathbf{0.6 \pm 0.1}$
8	75.3 ± 0.7	$\mathbf{78.9 \pm 0.3}$	0.4 ± 0.3	$\mathbf{0.3 \pm 0.1}$

Non-convex objective functions In order to assess the impact of weight averaging in private learning for non-convex objectives, we experiment with *ResNet-18*, *WRN-16-4* and *ResNet-50* on Fashion-MNIST (F-MNIST), CIFAR-10 and CIFAR-100. As we are mainly concerned with evaluating whether DP-SWA improves on DP-SGD, we do not perform data augmentation and choose hyperparameters based on related work (De et al., 2022; Shejwalkar et al., 2022; Panda et al., 2022). Refer to **??** for more details. We conduct experiments with *ResNet-18* and *WRN-16-4* on Fashion-MNIST and CIFAR-10; see Table 3 for mean and standard deviation over 5 random repeats. Additional results on CIFAR-10 and CIFAR-100 with *ResNet-18* and *Resnet-50* pre-trained on ImageNet are available in Appendix C.

Table 3: Accuracy, accuracy variance over the last iterates, and $\%_{\text{GAIN}}$ for non-convex objective functions. Mean and standard deviation across 5 runs. For compactness, we only report the order of magnitude for some of the results.

Experiment	ϵ	DP-SGD	DP-SWA	$\text{Var}_{\text{DP-SGD}}$ $(\times 10^{-3})$	$\text{Var}_{\text{DP-SWA}}$ $(\times 10^{-3})$	$\%_{\text{GAIN}}$
F-MNIST *ResNet-18*	1	79.1 ± 0.8	$\mathbf{79.9 \pm 0.1}$	1.3 ± 0.1	$\mathbf{0.3 \pm 0.2}$	$+2.4 \pm 0.4$
	8	84.9 ± 0.2	$\mathbf{85.6 \pm 0.1}$	1.5 ± 0.5	$\mathbf{0.1 \pm 0.1}$	$+4.6 \pm 0.9$
F-MNIST *WRN-16-4*	1	80.1 ± 0.4	$\mathbf{80.2 \pm 0.2}$	0.05 ± 0.04	$\mathbf{0.02 \pm 0.02}$	$+6.5 \pm 1.9$
	8	84.6 ± 0.1	$\mathbf{84.8 \pm 0.1}$	0.01 ± 0.01	$\mathbf{\leq 10^{-2}}$	$+9.7 \pm 2.9$
CIFAR-10 *ResNet-18*	1	38.8 ± 0.5	$\mathbf{41.0 \pm 0.9}$	10 ± 6	$\mathbf{0.4 \pm 0.4}$	$+6.2 \pm 0.7$
	8	60.8 ± 0.5	$\mathbf{62.1 \pm 0.5}$	3 ± 3	$\mathbf{0.2 \pm 0.1}$	$+2.9 \pm 0.5$
CIFAR-10 *WRN-16-4*	1	44.7 ± 1.2	$\mathbf{46.5 \pm 0.3}$	0.04 ± 0.01	$\mathbf{\leq 10^{-2}}$	$+4.3 \pm 2.9$
	8	66.7 ± 0.1	$\mathbf{67.6 \pm 0.1}$	0.04 ± 0.02	$\mathbf{\leq 10^{-3}}$	$+1.6 \pm 1.2$

With reference to the results presented in Table 3, DP-SWA performs better than DP-SGD, as it achieves superior accuracy, stability, and wider solutions. Accuracy improvements for other averaging schemes have also been recently observed in Shejwalkar et al. (2022) and, for large-scale models, De et al. (2022) were the first to achieve state-of-the-art accuracy with weight averaging. In particular, training without data augmentation, De et al. (2022) achieve a (validation) accuracy of $\approx 70\%$ on CIFAR-10 with $\epsilon = 8$ and $\delta = 10^{-5}$, which is comparable to our results. However, previous work does not consider generalization and the stability of the solutions, which we therefore investigate. In our experiments DP-SWA solutions are preferable because they are not only more accurate, but also wider and thus provide better generalization. See Figure 1 (Right) for a visualization. Wide minima are especially desirable in a DP setting as the noisy DP-SGD updates can favor escaping sharp minima (Wang et al., 2021). Panda et al. (2022) briefly address weight averaging for DP for fine-tuning pre-trained models and conclude that, in their experiments, it performs worse than the last iterate. While we observe diminishing performance benefits for pre-trained tasks, we nevertheless consider the DP-SWA solutions to be preferable also in these cases, for their stability and wideness properties.

5 Conclusion

Motivated by the lack of theoretical and experimental results on SWA in a private setting, we investigated the effects of DP-SWA for quadratic, convex, and non-convex objective functions. We (i) obtained a convergence bound for the quadratic case, (ii) showed empirical results for convex objective functions which exhibit a similar behavior as indicated by existing upper bounds, and (iii) found that **DP-SWA achieves better generalization, accuracy, and more stable solutions** in comparison to DP-SGD for the non-convex setting on standard benchmark datasets. In future work, we will consider a more extensive analysis of the benefits and limitations of DP-SWA and, e.g., investigate whether DP-SWA can perform well on language tasks where SWA has previously been shown to under-perform (Kaddour et al., 2022). Additionally, theoretical understanding of the effects of weight averaging in DP is currently lacking, particularly so for non-convex objective functions and different cycle lengths. In this sense, obtaining a better understanding of the loss surface for DP-SGD is a promising research direction.

ACKNOWLEDGMENTS

This work was funded in part by the TU Wien DK SecInt. The computational results presented have been achieved in part using the Vienna Scientific Cluster (VSC).

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*, pp. 308–318, 2016.

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL `https://www.wandb.com/`. Software available from wandb.com.

T Tony Cai, Yichen Wang, and Linjun Zhang. The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *The Annals of Statistics*, 49(5):2825–2850, 2021.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34:22405–22418, 2021.

Soham De, Leonard Berrada, Jamie Hayes, Samuel L Smith, and Borja Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology-EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28-June 1, 2006. Proceedings 25*, pp. 486–503. Springer, 2006.

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.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pp. 876–885, 2018.

Jean Kaddour, Linqing Liu, Ricardo Silva, and Matt J Kusner. When do flat minima optimizers work? *Advances in Neural Information Processing Systems*, 35:16577–16595, 2022.

Ashwinee Panda, Xinyu Tang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Dp-raft: A differentially private recipe for accelerated fine-tuning. *arXiv preprint arXiv:2212.04486*, 2022.

Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Úlfar Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9312–9321, 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. 2012.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, pp. 71–79. PMLR, 2013.

Virat Shejwalkar, Arun Ganesh, Rajiv Mathews, Om Thakkar, and Abhradeep Thakurta. Recycling scraps: Improving private learning by leveraging intermediate checkpoints. *arXiv preprint arXiv:2210.01864*, 2022.

Aditya Vardhan Varre, Loucas Pillaud-Vivien, and Nicolas Flammarion. Last iterate convergence of sgd for least-squares in the interpolation regime. *Advances in Neural Information Processing Systems*, 34:21581–21591, 2021.

Prateek Varshney, Abhradeep Thakurta, and Prateek Jain. (nearly) optimal private linear regression for sub-gaussian data via adaptive clipping. In *Conference on Learning Theory*, pp. 1126–1166. PMLR, 2022.

Wenxiao Wang, Tianhao Wang, Lun Wang, Nanqing Luo, Pan Zhou, Dawn Song, and Ruoxi Jia. Dplis: Boosting utility of differentially private deep learning via randomized smoothing. *Proceedings on Privacy Enhancing Technologies*, 2021(4):163–183, 2021.

Yu-Xiang Wang. Revisiting differentially private linear regression: optimal and adaptive prediction & estimation in unbounded domain. *arXiv preprint arXiv:1803.02596*, 2018.

Yu-Xiang Wang, Jing Lei, and Stephen E Fienberg. Learning with differential privacy: Stability, learnability and the sufficiency and necessity of erm principle. *The Journal of Machine Learning Research*, 17(1):6353–6392, 2016.

Guandao Yang, Tianyi Zhang, Polina Kirichenko, Junwen Bai, Andrew Gordon Wilson, and Chris De Sa. Swalp: Stochastic weight averaging in low precision training. In *International Conference on Machine Learning*, pp. 7015–7024. PMLR, 2019.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, et al. Opacus: User-friendly differential privacy library in pytorch. In *NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021.

A Additional preliminaries and proof of Theorem 1

A.1 DP-SGD

A DP version of SGD was introduced by Abadi et al. (2016). In comparison to standard SGD, in DP-SGD, during the update step, gradients are clipped to a maximum norm C to bound sensitivity (as there is no a priori bound on the size of the gradients) and Gaussian noise b is added before the descent step. For a loss function f with weights θ and learning rate α, we can write the gradient update (for a single data point to simplify notation) as

$$\theta_i = \theta_{i-1} - \alpha(\text{clip}_C \nabla f(\theta_{i-1}) + b_i),$$

where gradients are clipped to a maximum norm of C and $b_i \sim \mathcal{N}(0, \sigma^2)$.

A.2 Proof of Theorem 1

Let f be a quadratic objective function of the form $f(\theta) = (\theta - \theta^*)^T A(\theta - \theta^*)/2$ with $A \succeq \mu I \in \mathbb{R}^{d\times d}$, $\mu > 0$, and let θ^* be the optimum of f. Assume that the sample gradients $\nabla \tilde{f}(\theta)$ are, in expectation, equal to the full gradient, i.e., $\mathbb{E}[\nabla \tilde{f}(\theta)] = \nabla f(\theta) = A(\theta - \theta^*)$. For simplicity, assume a compact domain for the parameters θ to bound the gradient norm, and take this to be the clipping norm, which will therefore be omitted. For the gradient updates, take $b \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$ with $\sigma^2 = \frac{2\log 1.25/\delta}{\epsilon^2}$ to guarantee DP (Dwork et al., 2014; Abadi et al., 2016). Finally, make the standard assumption that $\mathbb{E}\|\nabla \tilde{f}(\theta) - \nabla f(\theta)\|^2 \leq \varsigma^2$ for some $\varsigma > 0$.

Theorem 1 (Adapted from Theorem 1 of Yang et al. (2019)) *Consider a cycle length c and a learning rate α with $0 < \alpha < \frac{\|A\|_2}{2}$. In expectation, the squared distance of the output $\bar{\theta}_T$ of DP-SWA at iteration T with respect to the optimum θ^* is upper bounded as:*

$$\mathbb{E}\|\bar{\theta}_T - \theta^*\|^2 \leq \frac{\|\theta_0 - \theta^*\|^2}{T^2\alpha^2\mu^2} + \frac{c}{T\alpha^2\mu^2}\left(\frac{2d\log\frac{1.25}{\delta}}{\epsilon^2} + \alpha^2\varsigma^2\right) \sim \mathcal{O}\left(\frac{cd}{T\epsilon^2}\right)$$

Following a similar approach to that of Theorem 1 in Yang et al. (2019), let us consider the update step

$$\begin{aligned}
\theta_{t+1} &= \theta_t - \alpha\nabla\tilde{f}(\theta) + b_t \\
\theta_{t+1} &= \theta_t - \alpha A(\theta_t - \theta^*) + \xi_t \\
\theta_{t+1} - \theta^* &= \theta_t - \theta^* - \alpha A(\theta_t - \theta^*) + \xi_t \\
&= (I - \alpha A)(\theta_t - \theta^*) + \xi_t
\end{aligned}$$

where the term ξ_t corresponds to

$$\xi_t = \alpha(A(\theta_t - \theta^*) - \nabla\tilde{f}(\theta_t)) + b_t.$$

Since sample gradients are, in expectation, equal to the full gradient and the Gaussian term b_t has mean zero, it follows that $\mathbb{E}[\xi_t] = 0$. Additionally, the variance of ξ_t can be bounded as

$$\mathbb{E}[\|\xi_t\|^2] = \mathbb{E}[\|\alpha(A(\theta_t - \theta^*) - \nabla\tilde{f}(\theta_t))\|^2] + \mathbb{E}[\|b_t\|^2] \leq \alpha^2\varsigma^2 + d\sigma^2.$$

We now expand on the time steps

$$\theta_t - \theta^* = (I - \alpha A)^t(\theta_0 - \theta^*) + \sum_{i=0}^{t-1}(I - \alpha A)^{t-i-1}\xi_i$$

and consider the distance from the optimum and the average $\bar{\theta}_K$ of K terms:

$$\begin{aligned}
\bar{\theta}_K - \theta^* &= \frac{1}{K}\sum_{t=1}^{K}\theta_{ct} - \theta^* \\
&= \frac{1}{K}\sum_{t=1}^{K}\left((I - \alpha A)^{ct}(\theta_0 - \theta^*) + \sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\xi_i\right) \\
&= \frac{1}{K}\left(\sum_{t=1}^{K}(I - \alpha A)^{ct}\right)(\theta_0 - \theta^*) + \frac{1}{K}\sum_{t=1}^{K}\sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\xi_i.
\end{aligned}$$

The first term of the right hand side is a constant which we denote by X_K. We now consider expectation of the squared norm:

$$\begin{aligned}
\mathbb{E}\left[\|\bar{\theta}_K - \theta^*\|^2\right] &= \mathbb{E}\left[\left\|X_K + \frac{1}{K}\sum_{t=1}^{K}\sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\xi_i\right\|^2\right] \\
&= \|X_K\|^2 + \mathbb{E}\left[\left\|\frac{1}{K}\sum_{t=1}^{K}\sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\xi_i\right\|^2\right] \\
&\quad + \frac{2}{K}X_K^T\sum_{t=1}^{K}\sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\mathbb{E}[\xi_i] \\
&= \|X_K\|^2 + \mathbb{E}\left[\left\|\frac{1}{K}\sum_{t=1}^{K}\sum_{i=0}^{ct-1}(I - \alpha A)^{ct-i-1}\xi_i\right\|^2\right] \\
&= \|X_K\|^2 + \frac{1}{K^2}\mathbb{E}\left[\left\|\sum_{i=0}^{cK-1}\sum_{t=\lfloor i/c\rfloor+1}^{K}(I - \alpha A)^{ct-i-1}\xi_i\right\|^2\right]
\end{aligned}$$

Since the ξ_i terms have mean zero and are independent, we can upper bound the variance as

$$\begin{aligned}
\mathbb{E}\left[\|\bar{\theta}_K - \theta^*\|^2\right] &= \|X_K\|^2 + \frac{1}{K^2}\sum_{i=0}^{cK-1}\mathbb{E}\left[\left\|\sum_{t=\lfloor i/c\rfloor+1}^{K}(I-\alpha A)^{ct-i-1}\xi_i\right\|^2\right] \\
&\leq \|X_K\|^2 + \frac{1}{K^2}\sum_{i=0}^{cK-1}\left\|\sum_{t=\lfloor i/c\rfloor+1}^{K}(I-\alpha A)^{ct-i-1}\right\|^2\mathbb{E}\left[\|\xi_i\|^2\right] \\
&\leq \|X_K\|^2 + \frac{1}{K^2}\sum_{i=0}^{cK-1}\left\|\sum_{j=0}^{\infty}(I-\alpha A)^{j}\right\|^2\mathbb{E}\left[\|\xi_i\|^2\right].
\end{aligned}$$

$\sum_{j=0}^{\infty}(I-\alpha A)^j$ converges as $0 < \alpha < \frac{1}{2}\|A\|_2$ and thus:

$$\left\|\sum_{j=0}^{\infty}(I-\alpha A)^j\right\|^2 = \left\|(I-(I-\alpha A))^{-1}\right\|^2 = \left\|\frac{1}{\alpha}A^{-1}\right\|^2 \leq \frac{1}{\alpha^2\mu^2}$$

Rewriting, we obtain

$$\mathbb{E}[||\bar{\theta}_K - \theta^*||^2] \leq \|X_K\|^2 + \frac{1}{K^2\alpha^2\mu^2}\sum_{i=0}^{cK-1}\mathbb{E}[\|\xi_i\|^2] \leq \|X_K\|^2 + \frac{c}{K\alpha^2\mu^2}(\alpha^2\varsigma^2 + d\sigma^2).$$

We now analyze the constant term $\|X_K\|^2$:

$$\begin{aligned}
\|X_K\|^2 &= \left\|\frac{1}{K}\Big(\sum_{t=1}^{K}(I-\alpha A)^{ct}\Big)(\theta_0-\theta^*)\right\|^2 \\
&\leq \frac{1}{K^2}\left\|\sum_{t=1}^{K}(I-\alpha A)^{ct}\right\|^2\|\theta_0-\theta^*\|^2 \\
&\leq \frac{1}{K^2}\left\|\sum_{t=1}^{\infty}(I-\alpha A)^{t}\right\|^2\|\theta_0-\theta^*\|^2 \\
&= \frac{1}{K^2}\left\|(I-(I-\alpha A))^{-1}\right\|^2\|\theta_0-\theta^*\|^2 \\
&= \frac{1}{K^2}\left\|\frac{1}{\alpha}A^{-1}\right\|^2\|\theta_0-\theta^*\|^2 \\
&\leq \frac{\|\theta_0-\theta^*\|^2}{K^2\alpha^2\mu^2}
\end{aligned}$$

If we run DP-SWA for $T \sim \mathcal{O}(K)$ iterations, we obtain the following bound:

$$\mathbb{E}\|\bar{\theta}_T - \theta^*\|^2 \leq \frac{\|\theta_0-\theta^*\|^2}{T^2\alpha^2\mu^2} + \frac{c}{T\alpha^2\mu^2}\left(\frac{2d\log\frac{1.25}{\delta}}{\epsilon^2} + \alpha^2\varsigma^2\right) \sim \mathcal{O}\left(\frac{cd}{T\epsilon^2}\right)$$

Note that the squared distance to the optimum and the objective gap $f(\bar{\theta}_T) - f(\theta^*)$ can be directly compared, as they differ at most by a factor of μ since $f(\bar{\theta}_T) - f(\theta^*) \geq \frac{\mu}{2}\|\bar{\theta}_T - \theta^*\|^2$. This observation provides convergence to zero for the excess risk.

B EXPERIMENTAL SETUP

We use PyTorch (Paszke et al., 2019) (version 1.13.1), Opacus (Yousefpour et al., 2021) (version 1.3.0) and Weights and Biases (version 0.15.0) (Biewald, 2020) with Python 3.10.9 for all of our experiments. With the exception of experiments with *ResNet-50*, we conduct experiments on a single PC with an RTX-3080 GPU and Intel Core i9-11900KF CPU. For *ResNet-50*, we conduct experiments on a node of the Vienna Scientific Cluster (VCS) equipped with an A100 Tensor Core GPU and an AMD EPYC CPU.

For all our experiments, we rely on a conventional implementation of DP-SGD, using a constant learning rate and no momentum. We use a clipping norm of 1.

We train linear regression for 100 epochs with a warm-up phase of 0.2, a constant learning rate of 10^{-4} and a batch size of 1. The data generation follows that of Yang et al. (2019). We use the mean squared error loss. Data points x_i are sampled as $x_i \sim \mathcal{N}(0, \sigma_x^2)$, initial weights θ_0 are sampled uniformly from $[0, 1]$, labels are sampled as $y_i \sim \mathcal{N}(\theta_0^T x_i, \sigma_y^2)$. We generate 256-dimensional points, with variances of 1, and sample $n = 4096$ points.

For Fashion-MNIST, we train logistic regression for 20 epochs with a warm-up phase of 0.6, a constant learning rate of 0.1 and a batch size of 8.

For all experiments with deep learning models (cf. Section 4), we replace batch normalization layers with group normalization layers to allow for private training. For Fashion-MNIST, we train *ResNet-18* and *WRN-16-4* for 20 epochs with a warm-up phase of 0.7, a constant learning rate of 2 and a logical batch size of 512. For CIFAR-10, we use a constant learning rate of 4 and a logical batch size of 4096, and train *ResNet-18* for 100 epochs with a warm-up phase of 0.7, and *WRN-16-4* for 300 epochs with a warm-up phase of 0.9. For all pre-trained models, we train for 50 epochs with a warm-up phase of 0.9, a constant learning rate of 2 for $\epsilon = 1$ and 4 for $\epsilon = 8$, using a batch size of 4096. For Fashion-MNIST, we replace the first convolutional layer of *ResNet-18* and *WRN-16-4* to allow for grey-scale images as input (i.e., 1 channel instead of 3 channels). For the models pre-trained on ImageNet, we replace the last (fully-connected) layer of *ResNet-18* and *ResNet-50* to allow for 10 and 100 classes, respectively for CIFAR-10 and CIFAR-100. We train only the last layer and keep the weights of the remaining layers frozen.

The code used to perform the experiments and obtain the results presented in this work is available at https://github.com/pindri/dp-swa.

C ADDITIONAL RESULTS

In the following sections we provide additional results and visualizations from our experiments.

C.1 LINEAR REGRESSION

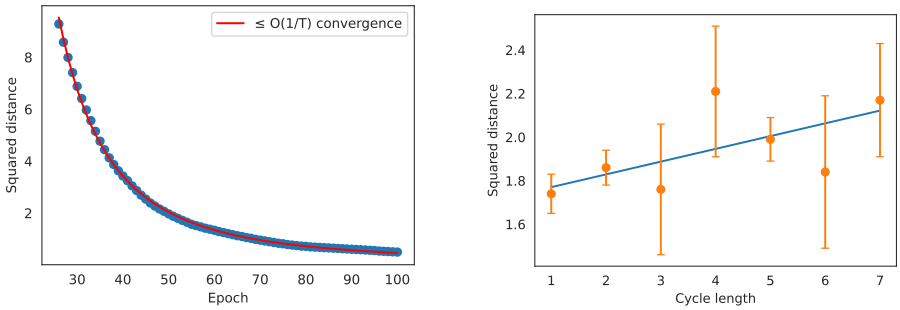


Figure 2: Results for linear regression with $\epsilon = 8$. (*Left*) Squared distance to the optimum, in agreement with convergence faster than $1/T$. (*Right*) Dependency of squared distance to the optimum on cycle length, compatible with a decrease in performance for larger cycle lengths; aggregate results for 3 runs.

C.2 LOGISTIC REGRESSION

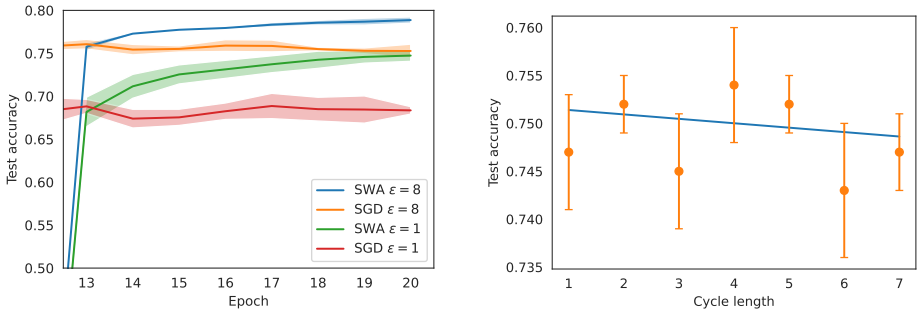


Figure 3: Results for logistic regression. (*Left*) DP-SWA achieves better test accuracy and lower solution variance in comparison with DP-SGD; aggregate results for 5 runs. (*Right*) Dependency of test accuracy on cycle length. Results for $\epsilon = 1$, compatible with a decrease in performance for larger cycle lengths; aggregate results for 3 runs.

C.3 NON-CONVEX OBJECTIVE FUNCTIONS

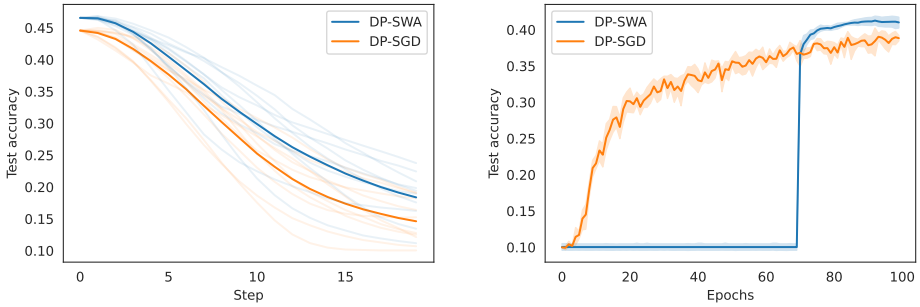


Figure 4: Results for CIFAR-10. (*Left*) Accuracy as a function of the distance to the optimum for *WRN-16-4*, obtained by considering a single run and taking steps along 10 random rays starting from the DP-SGD and the DP-SWA solutions. The solid line indicates the mean across the 10 rays. DP-SWA consistently maintains a higher accuracy compared to DP-SGD as we move away from the respective solutions. (*Right*) Variance across 5 runs with *ResNet-18*. DP-SWA has a smaller variance across run and better accuracy in comparison to DP-SGD.

C.4 PRE-TRAINED *ResNet-18* AND *ResNet-50*

Table 4: Accuracy, accuracy variance over the last iterates, and $\%_{\text{GAIN}}$ for non-convex objective functions with pre-trained models. Mean and standard deviation across 3 runs.

Dataset	ϵ	DP-SGD	DP-SWA	$\text{Var}_{\text{DP-SGD}}$	$\text{Var}_{\text{DP-SWA}}$	$\%_{\text{GAIN}}$
CIFAR-10	1	83.9 ± 0.3	$\mathbf{84.0 \pm 0.3}$	10^{-6}	$\mathbf{10^{-7}}$	$+6.8 \pm 2.5$
ResNet-18	8	85.2 ± 0.1	$\mathbf{85.3 \pm 0.1}$	10^{-6}	10^{-6}	$+0.7 \pm 2$
CIFAR-10	1	85.5 ± 0.3	$\mathbf{85.6 \pm 0.3}$	10^{-7}	10^{-7}	$+1.5 \pm 5.9$
ResNet-50	8	87.1 ± 0.1	$\mathbf{87.10 \pm 0.01}$	10^{-7}	10^{-7}	$+2.0 \pm 2.7$
CIFAR-100	1	54.4 ± 0.2	$\mathbf{54.5 \pm 0.1}$	10^{-5}	10^{-5}	$+3.9 \pm 2.9$
ResNet-18	8	63.3 ± 0.2	63.3 ± 0.2	10^{-6}	$\mathbf{10^{-7}}$	$+2.7 \pm 1.0$
CIFAR-100	1	58.4 ± 0.1	58.4 ± 0.1	10^{-6}	10^{-6}	-1.7 ± 2.1
ResNet-50	8	$\mathbf{66.6 \pm 0.2}$	66.5 ± 0.2	10^{-6}	$\mathbf{10^{-7}}$	-2.1 ± 5.9

We report additional results on CIFAR-10 and CIFAR-100 with *ResNet-18* and *Resnet-50* pre-trained on ImageNet. For compactness, we only report the order of magnitude for the accuracy variance over the last iterates, as accuracy variance is generally very low for the pre-trained models and comparable between DP-SGD and DP-SWA.

We can observe in Table 4 that DP-SWA offers no substantial improvement in terms of accuracy for pre-trained models, with some experiments providing identical accuracy. It should however be noted that our results are in contrast with the ones from Panda et al. (2022), who observe that weight averaging performs worse than the last iterate. On the contrary, we find DP-SWA solutions to be preferable as they are generally wider and more stable. *ResNet-50* on CIFAR-100 is the only case where we observe no benefit in terms of solution wideness.