Three Tools for Practical Differential Privacy

KOEN VAN DER VEEN
University of Amsterdam

RUBEN SEGGERS
University of Amsterdam

PETER BLOEM
VU Amsterdam

GIORGIO PATRINI
University of Amsterdam

DIFFERENTIALLY PRIVATE LEARNING POSES CHALLENGES FOR STANDARD ML PRACTICE:

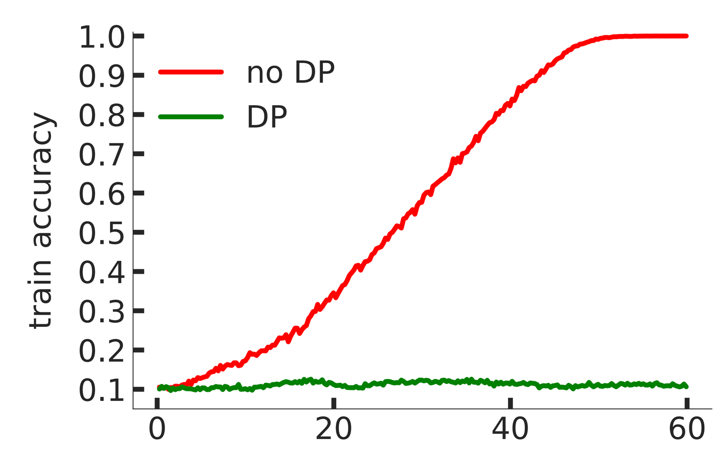
- Privacy guarantees are difficult to interpret.
- Hyperparameter tuning on private data reduces the privacy budget.
- Ad-hoc privacy attacks are often required to test model privacy.

We introduce three tools to make differentially private machine learning more practical:

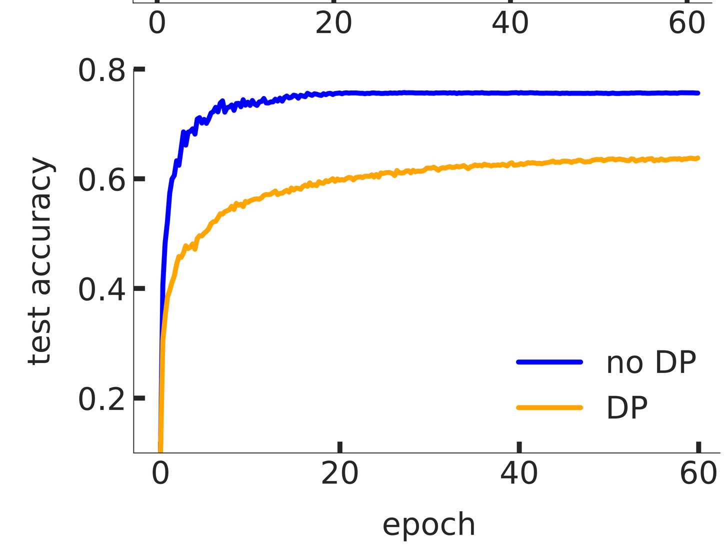
1. Simple **sanity checks** which can be carried out before seeing the data.
2. An **adaptive clipping bound** to reduce the effective number of tuneable privacy parameters.
3. We show that **large-batch training** improves model performance.

1. sanity checks

Deep neural networks can easily memorize training labels in image classification, even on randomly labeled data [1]. Under differential privacy, such memorization should not be possible. **This allows us to calibrate our privacy parameters**: if the model is able to learn a randomly labeled task, the privacy parameters are insufficiently strict.

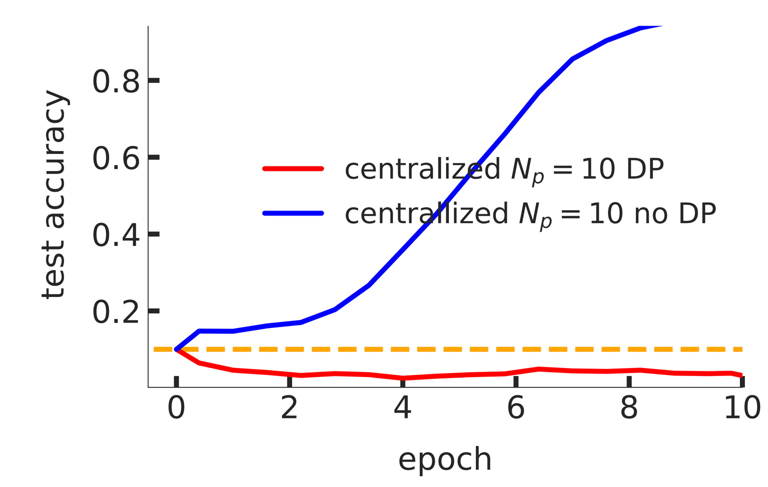


Training accuracy on random data, with and without differential privacy.

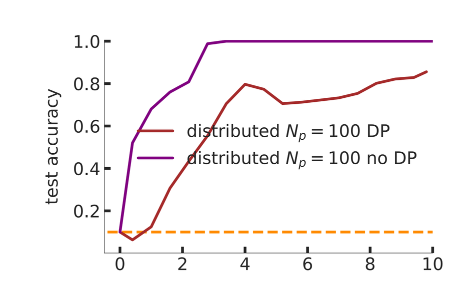


Test accuracy on CIFAR10. With and without differential privacy.

Patterns inserted for only one user should *not* improve performance. We can calibrate our privacy parameters by ensuring that such patterns inserted in random data do not improve accuracy beyond chance.



Test accuracy on random data, with a pattern inserted for a single user. With differential privacy and without.

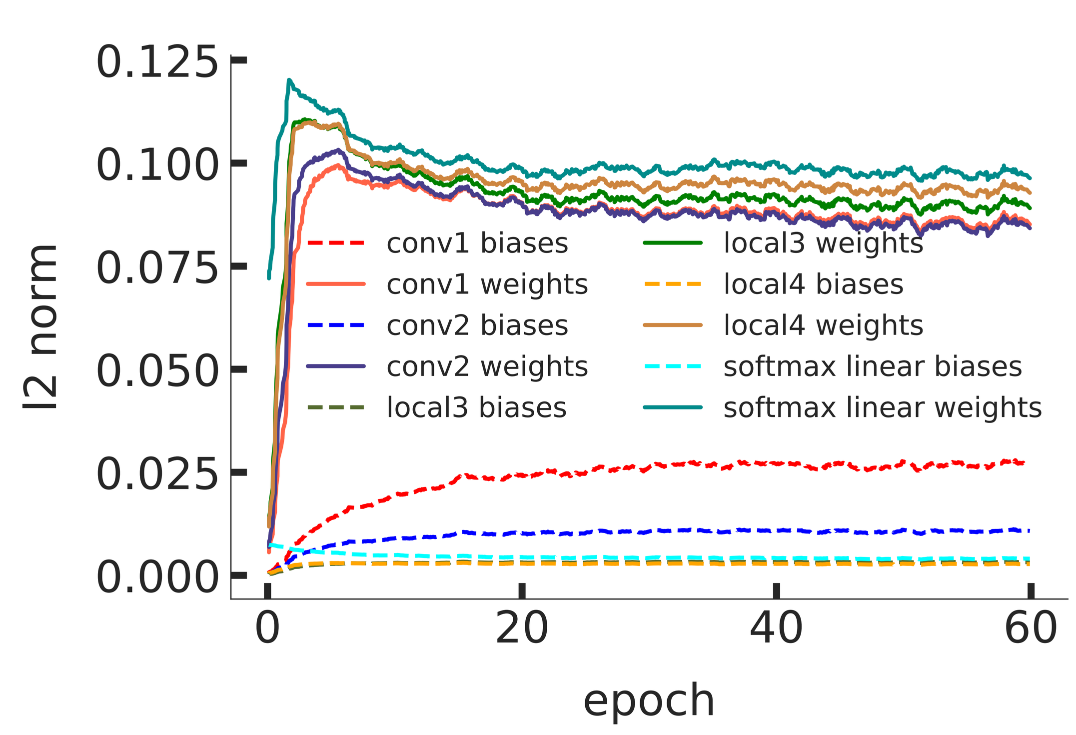


Test accuracy with a pattern distributed over all users.

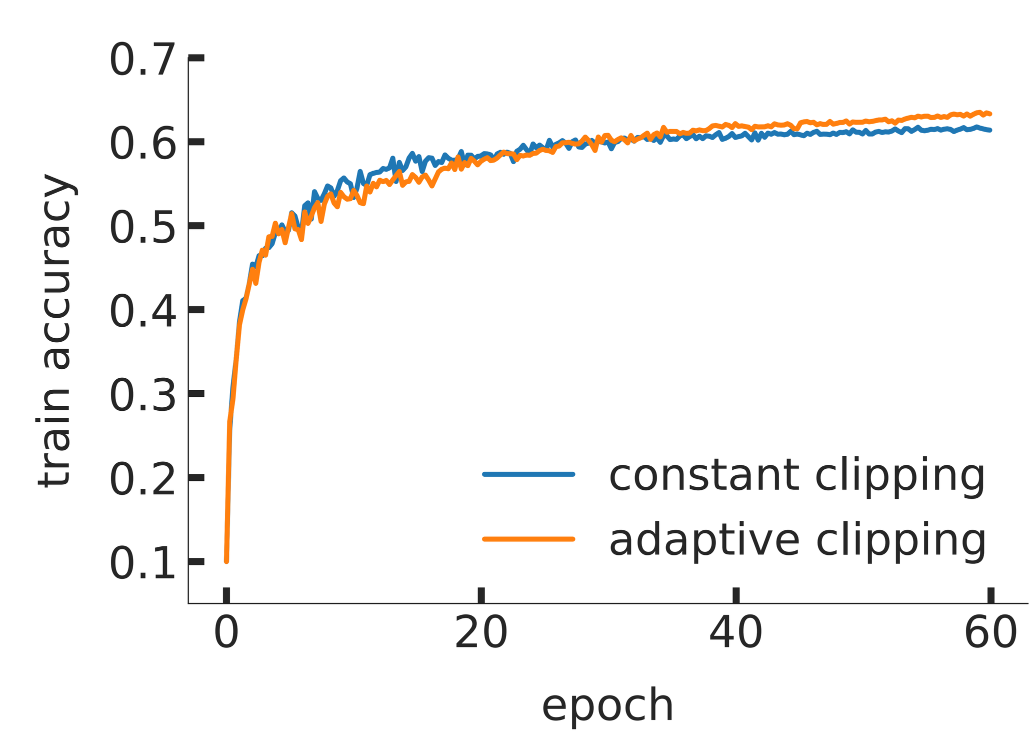
2. adaptivity

Hyperparameter optimization under differential privacy can be a costly procedure. Each model we test eats into the privacy budget. **For effective learning under differential privacy, we should eliminate hyperparameters as much as possible.**

We test an adaptive approach to determining the clipping bound parameters in the DPSGD algorithm [2].



The l^2 norms (which are clipped in DPSGD) vary considerably between parameters and between epochs.



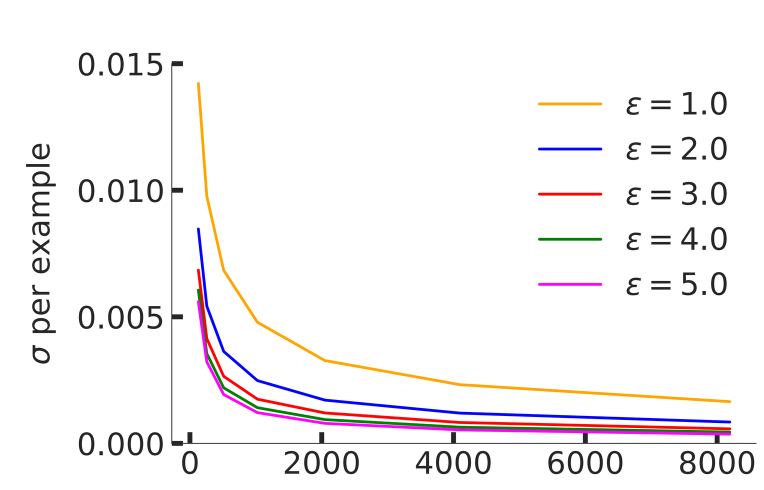
Adaptive clipping determines the clipping parameter C_t on the basis of the norms of the gradient updates for parameter l at epoch $t-1$.

3. large-batch training

When we compute a single gradient update over a large batch, less noise and clipping are required to ensure privacy. However, small batches often lead to more effective learning.

Several methods exist to train effectively with large batches [3]. We investigate the simplest of these: **increasing the base learning rate linearly with the batch size.**

For a fixed privacy budget and number of epochs, far less noise is required to ensure differential privacy, if we train with large batches.



We use the DPSGD moments accountant to find the minimum σ that fully uses a pre-defined privacy budget in 10 epochs for a training set of 60,000 examples.

Batch size	accuracy
128	61.6%
512	64.2%
1024	**66.9%**
1024 (base lr)	47.2%

Starting with base learning rate, optimised for small batches, we can increase batch size and maintain performance if we scale the learning rate by the same factor as the batch size.

UNIVERSITY OF AMSTERDAM

[1] Zhang, Chiyuan, et al. *Understanding deep learning requires rethinking generalization.* 2016.
[2] Abadi, Martin, et al. *Deep learning with differential privacy.* Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016.
[3] Goyal, Priya, et al. *Accurate, large minibatch SGD: training imagenet in 1 hour.* arXiv preprint arXiv:1706.02677