Assignment 2 (538L)

Shadab Shaikh

March 25, 2022

1 Overview

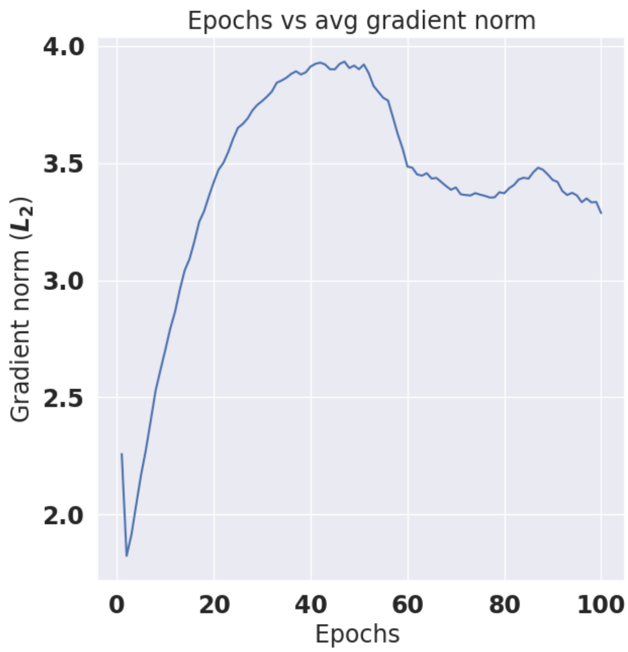
For this assignment, I used a simple CNN (4 Conv layers + 1 Dense layer) trained on MNIST dataset with DPSGD. I used Jax's vmap function (along used jit functionality) to parallelize per-example gradient computation and rescaling. I added the gaussian noise to the gradient summed over the batch (instead of average), hence the noise is proportional to clipping/rescaling bound C. The hyperparameters are given in section 2. Code: https://github.com/greninja/538L-assignment-2.

*[**Note**: I avoided the subsampling part of DPSGD (since it was slow to run), so I resorted to the usual technique of shuffling the dataset and iterating over each disjoint batch. Hence, the privacy analysis isn't a true reflection of my DPSGD implementation.]*

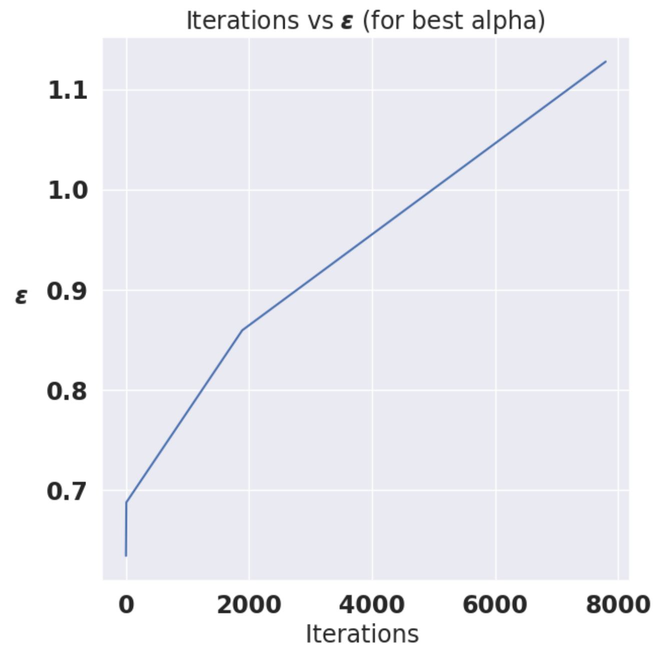
2 Config/Hyperparameters

- **Dataset**: MNIST
- **Dataset size (N)**: 60000
- **Noise mechanism**: Gaussian
- $\boldsymbol{\delta}$: 1.67 $\times 10^{-5}$ (= 1/N)
- **Sensitivity bound of gradients (C)**: 1.0
- **Standard deviation (σ)**: 1.0
- **Noise multiplier**: 1.0
- **Optimizer**: SGD (step size = 1e-3)
- **Epochs (= iterations)**: 100 (= 7800)
- **Batch size**: 128
- **alphas (RDP order)**: {1.1, 1.2,.......62, 63}

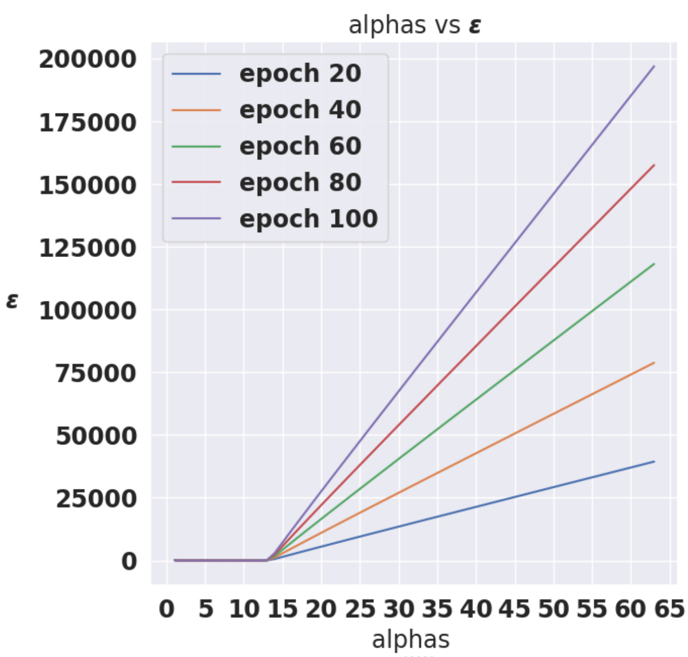
3 Plots



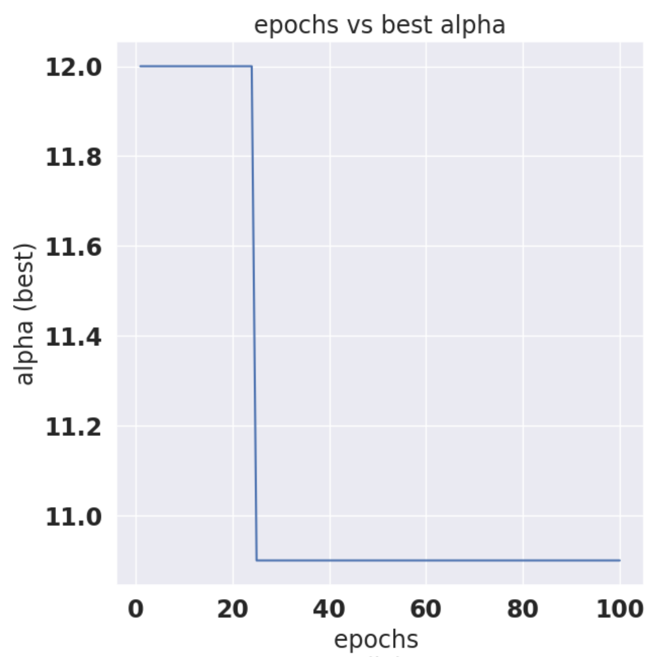
(i)



(ii)



(iii)



(iv)

4 Description of plots

1. **Plot (i):** Inspired by [1], I wanted to analyze how the average L_2 norm of (unclipped) gradient of the whole dataset changes across epochs. I didn't see a very clear pattern except it dips initially and then increases to reach some peak and then dips again. I guess based on how I set the relevant hyperparameters (learning rate, clipping threshold) this would change, but I didn't do an exhaustive HP

search.

2. **Plot (ii):** The ϵ at the end of 100 epochs/7800 iterations converges to ≈ 1.2.
3. **Plot (iii):** Plotting the RDP curve for composing multiple gaussian mechanisms. I am aware that the plot should ideally have a curve for each query since each query is a gaussian mechanism and the fact that the final curve is the sum of each sub-mechanism, but due to space congestion I just sampled and plotted every 20 epochs. Some observations:
   - The ϵ values for $0 \leq \alpha \leq 15$ are not exactly zero. It's just that they are not discernible due to scale of y-axis (to accomodate for larger ϵ values for α's > 15)]
   - As you can see, the ϵ values jump significantly $\alpha \geq 15$. I am not entirely sure if this is expected or its a bug in my code.
   - For a Gaussian mechanism, since ϵ (guarantee) is linearly proportional to α, it's RDP curve (α vs ϵ) should be a straight line. However, I observed a weird behavior : between $\alpha \in \{1.1, 1.2, ...62, 63\}$ (which is a monotonically increasing sequence), the ϵ first decreases and then again increases. Since I had to tweak the privacy account code of opacus to output ϵ for every α, I initially thought it might be a bug in my code but I rechecked it with opacus's own accounting functionality as well and I observed the same behaviour there. I am slightly unclear on this.
4. **Plot (iv):** I wanted to observe whether the best α changes over training epochs. It just changed from 12 to 10.9.

References

[1] Bagdasaryan et al. *Differential Privacy Has Disparate Impact on Model Accuracy*, 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)