HW 8a: DP-SGD

CS 208 Applied Privacy for Data Science, Spring 2025

Version 1.1: Due Fri, Apr. 4th, 11:59pm.

Instructions: Submit a PDF file that contains both your written responses as well as your code to the assignment on Gradescope. Read the section ”Collaboration & AI Policy” in the syllabus for our guidelines regarding the use of LLMs and other AI assistance on the assignments.

1. **Implementing DP-SGD:**] In our code example in class,[1] we saw how to release an estimated Logistic regression for predicting marital status from education level using DP-SGD to optimize the log-likelihood loss function. Convert this code to release the probability of employment given education level and disability status (these are the same variables we used in the Opacus example). You will need to modify the loss function, the gradient clipping, and Gaussian noise addition to handle the additional variable present here compared to the code from class.

   As discussed in the class, the learning rate parameter needs to be set correctly in order to obtain convergence in the DP-SGD setting. The learning rate ν in the notebook is a 2-dimensional vector (the coefficient of education level and the intercept).[2] Since we are now adding one more prediction variable, we need a third learning rate parameter, which you are going to find privately. (You can keep the education coefficient and intercept learning rates the same as in the notebook.)

2. **Private Model Selection**. Use your code from part 1 to create $K = 10$ differentially private models (each with privacy parameter $\varepsilon = 1$ and $\delta = 10^{-6}$) across a sequence of learning rates (you can leave all the other parameters as in the exemplar code, or adjust them to reasonable values). Choose one of these models to release, by means of the exponential mechanism with privacy parameter $\varepsilon = 1$. Use a score function in the exponential mechanism that is the negative of the loss function you used in training. Show the parameters of the DP model that is trained using the chosen learning rate.

   The privacy loss of the entire procedure above can be analyzed by applying standard composition theorems, but this will incur a loss that grows with the number K of models trained. However, a theorem of Liu and Talwar (STOC 2019) shows that in fact one can do this model selection with only a constant-factor increase in the privacy-loss parameter ε, with no dependence on K.

[1] See `https://opendp.github.io/cs208/spring2025/homeworks/ps8a/dpsgd_notebook.ipynb`

[2] Ordinarily, the learning rate parameter is treated as a single scalar that multiplies the entire gradient. Allowing different learning rates per coordinate amounts to also normalizing the different independent variables.