Notice

- Action items
  - 12/07: Final project presentation
    - 12 min presentation + 1-3 min Q&A (strict)
    - Presentation MUST cover:
      - 1 slide on your research motivation and goals
      - 1 slides on your ideas (how did you plan to achieve your goals)
      - 1-2 slides on your hypotheses and experimental design
      - 2-3 slides on your most interesting results
      - 1 slides on your conclusion and implications
  - 12/12: Final exam (online, 24 hrs., unlimited trials)
  - 12/12: Final project report (Template is on the class website)
  - 12/14: Late submissions for HW 1-4

CS 499/579: Trustworthy ML
Differential privacy

Tu/Th 4:00 – 5:50 pm

Sanghyun Hong

sanghyun.hong@oregonstate.edu

Oregon State University

SAIL

Secure AI Systems Lab

How can we achieve private learning?

Deep learning with differential privacy, Abadi et al., ACM CCS 2015

Definition of memorization

- Feldman and Zhang's
  - For a training algorithm A
  - Operating on a training set S
  - Quantify the label memorization as follows:

$$\texttt{mem}(\mathcal{A}, S, i) := \Pr_{h \leftarrow \mathcal{A}(S)}[h(x_i) = y_i] - \Pr_{h \leftarrow \mathcal{A}(S^{\setminus i})}[h(x_i) = y_i].$$

  - Problem: the estimation requires tons of training of a model on data

Definition of memorization

- Feldman and Zhang's
  - For a training algorithm A
  - Operating on a training set S
  - New way to quantify the label memorization

$$\texttt{infl}(\mathcal{A}, S, i, j) := \Pr_{h \leftarrow \mathcal{A}(S)}[h(x'_j) = y'_j] - \Pr_{h \leftarrow \mathcal{A}(S^{\setminus i})}[h(x'_j) = y'_j].$$

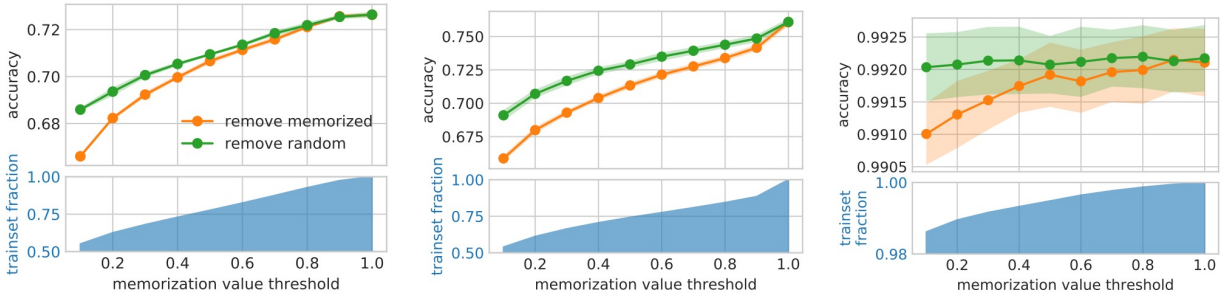
    - Use the test-set to measure the memorization
    - How much influence a single example on the test-set
    - Memorization is high, when the influence (acc. difference) is high

Definition of memorization

- Feldman and Zhang's
  - New way to quantify the label memorization

$$\mathtt{infl}(\mathcal{A}, S, i, j) := \Pr_{h \leftarrow \mathcal{A}(S)}[h(x'_j) = y'_j] - \Pr_{h \leftarrow \mathcal{A}(S^{\setminus i})}[h(x'_j) = y'_j].$$

  - How much influence a single example on the test-set
  - Memorization is high, when the influence (acc. difference) is high



(a) ImageNet (b) CIFAR-100 (c) MNIST

Figure 2: Effect on the test set accuracy of removing examples with memorization value estimate above a given threshold and the same number of randomly chosen examples. Fraction of the training set remaining after the removal is in the bottom plots. Shaded area in the accuracy represents one standard deviation on 100 (CIFAR-100, MNIST) and 5 (ImageNet) trials.

Definition of an algorithm being private

- A private model (an algorithm)
  - Feldman and Zhang's label memorization

$$\texttt{infl}(\mathcal{A}, S, i, j) := \Pr_{h \leftarrow \mathcal{A}(S)}[h(x'_j) = y'_j] - \Pr_{h \leftarrow \mathcal{A}(S^{\setminus i})}[h(x'_j) = y'_j].$$

    - How much influence a single example on the test-set
    - Memorization is high, when the influence (acc. difference) is high

  - Property of a private model
    - Given any training instance, its influence on the test acc. is low

Revisiting differential privacy

- ϵ-Differential Privacy
  - A randomized algorithm $M: D \rightarrow R$ with domain D and a range R satisfies ϵ-differential privacy if for any two adjacent inputs $d, d' \in D$ and any subset of outputs $S \subset R$ it holds

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(d') \in S]$$

Revisiting differential privacy

- ϵ-Differential Privacy
  - A randomized algorithm $M: D \rightarrow R$ with domain D and a range R satisfies ϵ-differential privacy if for any two adjacent inputs $d, d' \in D$ and any subset of outputs $S \subset R$ it holds

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(d') \in S]$$

- (ϵ, δ)-Differential Privacy

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(d') \in S] + \delta$$

  - δ: Represent some catastrophic failure cases [Link, Link]
  - $\delta < 1/|d|$, where $|d|$ is the number of samples in a database

Revisiting differential privacy

- (ϵ, δ)-Differential Privacy **[Conceptually]**

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(d') \in S] + \delta$$

  - You have two databases d, d' differ by one item
  - You make the same query M to each and have results $M(d)$ and $M(d')$
  - You ensure the distinguishability between the two under a measure ϵ
    - ϵ is large: those two are distinguishable, less private
    - ϵ is small: the two outputs are similar, more private
  - You also ensure the catastrophic failure probability under δ

Revisiting differential privacy

- (ϵ, δ)-Differential Privacy

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(d') \in S] + \delta$$

- Mechanism for (ϵ, δ)-DP: Gaussian noise

$$\mathcal{M}(d) \triangleq f(d) + \mathcal{N}(0, S_f^2 \cdot \sigma^2)$$

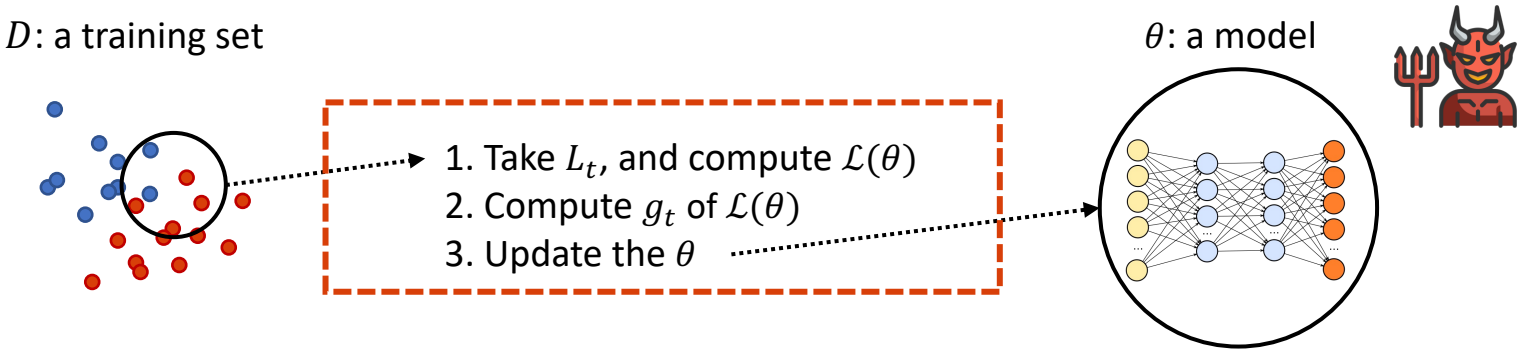
  - $M(d)$: (ϵ, δ)-DP query output on d
  - $f(d)$: non (ϵ, δ)-DP (original) query output on d
  - $N(0, S_f^2 \cdot \sigma^2)$: Gaussian normal distribution with mean 0 and the std. of $S_f^2 \cdot \sigma^2$

Post-hoc: Set the Goal ϵ and Calibrate the noise $S_f^2 \cdot \sigma^2$!

Differential privacy for machine learning

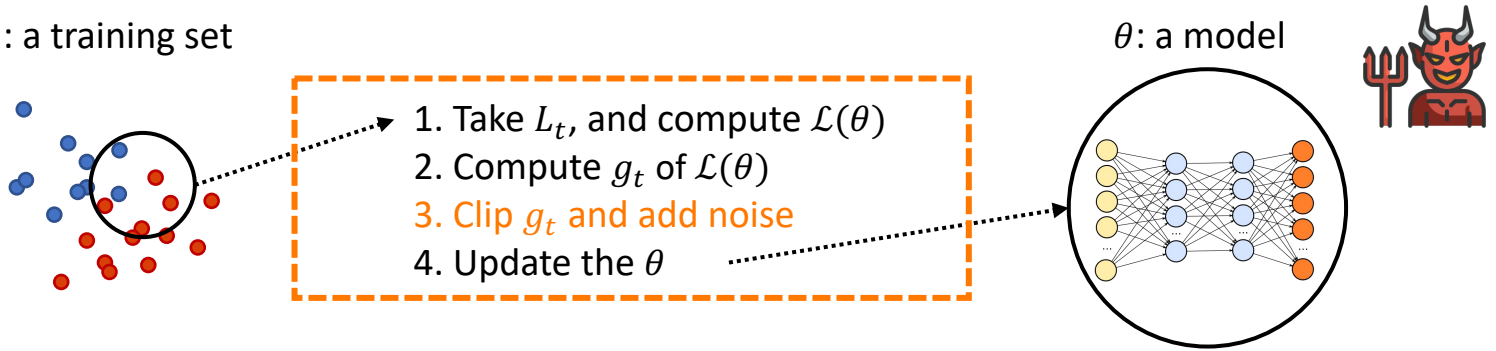
- Revisiting mini-batch stochastic gradient descent (SGD)
  1. At each step t, it takes a mini-batch L_t
  2. Computes the loss $\mathcal{L}(\theta)$ over the samples in L_t, w.r.t. the label y
  3. Computes the gradients g_t of $\mathcal{L}(\theta)$
  4. Update the model parameters θ towards the direction of reducing the loss

This Process Should Be (ϵ, δ)-DP!



MAKE EACH MINI-BATCH SGD STEP (ϵ, δ)-DP

- Mini-batch stochastic gradient descent (SGD)
  1. At each step t, it takes a mini-batch L_t
  2. Computes the loss $\mathcal{L}(\theta)$ over the samples in L_t, w.r.t. the label y
  3. Computes the gradients g_t of $\mathcal{L}(\theta)$
  4. Clip (scale) the gradients to $1/C$, where $C > 1$
  5. Add Gaussian random noise $N(0, \sigma^2 C^2 \mathbf{I})$ to g_t
  6. Update the model parameters θ towards the direction of reducing the loss



MAKE THE ENTIRE TRAINING PROCESS (ϵ, δ)-DP

- Mini-batch stochastic gradient descent (SGD)
  - SGD iteratively computes the (ϵ, δ)-DP step T times
  - **Problem:** how do we compute the total privacy leakage ϵ_{tot} over T iterations?

- Privacy accounting with moment accountant
  - **Key intuition:** DP has the **composition** property
    - Suppose the two mechanism M_1 and M_2 satisfies $(\varepsilon_1, \delta_1)$- and $(\varepsilon_2, \delta_2)$-DP the composition of those mechanisms $M_3 = M_2(M_1)$ satisfies $(\varepsilon_1+\varepsilon_2, \delta_1+\delta_2)$-DP
    - If each step t satisfies (ε, δ)-DP, the total SGD process satisfies $(\varepsilon T, \delta T)$-DP
  - **Moment accountant:** tracking the total privacy leakage εT over T iterations

PUTTING ALL TOGETHER

- DP-Stochastic Gradient Descent (DP-SGD)

Algorithm 1 Differentially private SGD (Outline)

Input: Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$. Parameters: learning rate η_t, noise scale σ, group size L, gradient norm bound C.

Initialize θ_0 randomly

for $t \in [T]$ **do**

Take a random sample L_t with sampling probability L/N

Compute gradient

For each $i \in L_t$, compute $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t}\mathcal{L}(\theta_t, x_i)$

Clip gradient

$\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}\right)$

Add noise

$\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L}\left(\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})\right)$

Descent

$\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$

$\varepsilon, \delta \leftarrow$ compute the privacy cost (leakage) so far

If $\varepsilon > \varepsilon_{buget}$: then break;

Output θ_T and compute the overall privacy cost (ε, δ) using a privacy accounting method.

// we train a model θ with the privacy budget ε_{budget}

// iterate over *T* mini-batches

// compute the gradient

// clip the magnitude of the gradients

// add Gaussian random noise to the gradients

// compute the privacy cost (leakage) up to *t* iterations

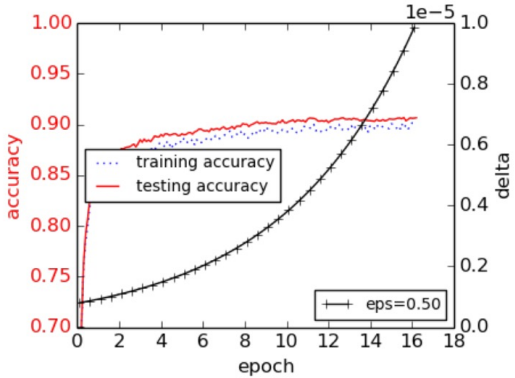
// if the cost is over the budget, then stop training

Evaluation

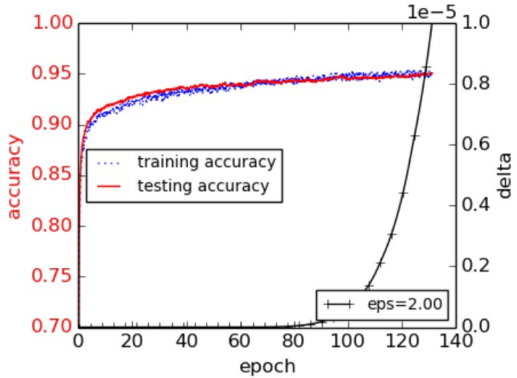
- Setup
  - Datasets: MNIST | CIFAR-10/100
  - Models:
    - MNIST: 2-layer feedforward NN on 60-dim. PCA projected inputs
    - CIFAR-10/100: A CNN with 2 conv. layers and 2 fully-connected layers
  - Metrics:
    - Classification accuracy
    - Privacy cost (ε_{budget})

Evaluation

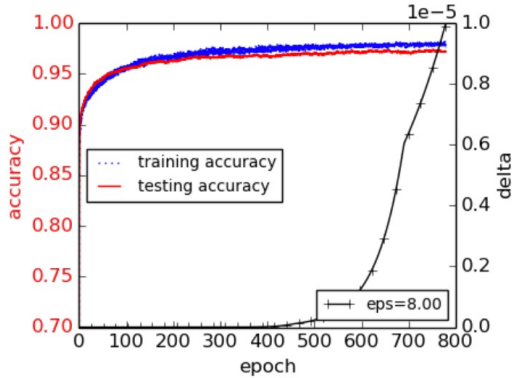
- Impact of Noise
  - Dataset, Models: MNIST, 2-layer feedforward NN
  - Setup: 60-dim PCA projected inputs | Clipping threshold ($\mathbf{C}$): 4 | Noise (σ): 8, 4, 2 (from the left)
  - **Summary:**
    - On MNIST, DP-SGD offers reasonable acc. under various privacy costs (**clean:** 98.3%)
    - The accuracy of private models decreases as we decrease the privacy cost



(1) Large noise



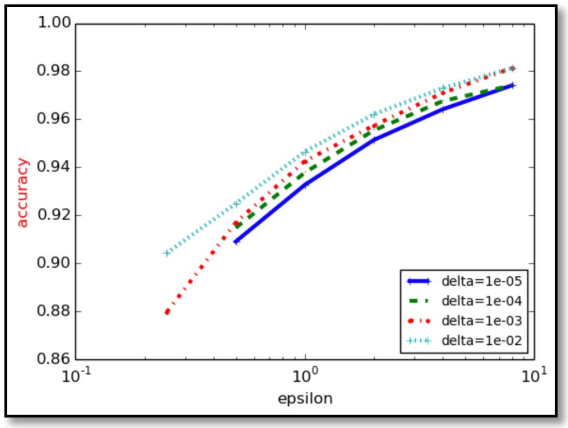
(2) Medium noise



(3) Small noise

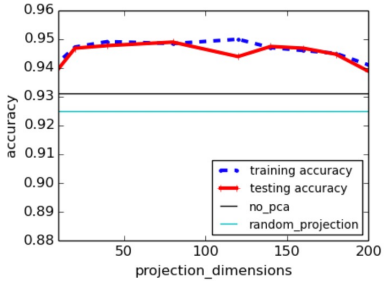
Evaluation

- Impact of Noise
  - Dataset, Models: MNIST, 2-layer feedforward NN
  - Setup: 60-dim PCA projected inputs | Clipping threshold ($\mathbf{C}$): 4 | Noise (σ): 8, 4, 2 (from the left)
  - **Summary:**
    - On MNIST, DP-SGD offers reasonable acc. under various privacy costs (**clean:** 98.3%)
    - The accuracy of private models decreases as we decrease the privacy cost

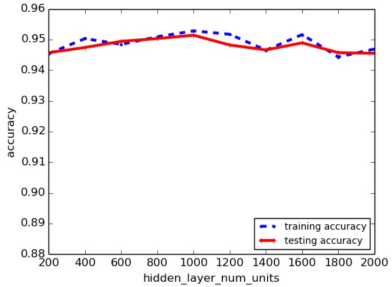


Evaluation

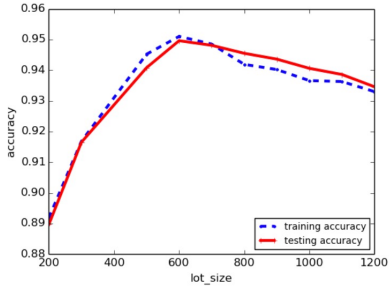
- Impact of Hyper-parameter Choices
  - Dataset, Models: MNIST, 2-layer feedforward NN
  - Setup: 60-dim PCA projected inputs



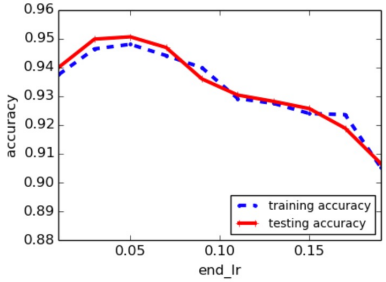
(1) variable projection dimensions



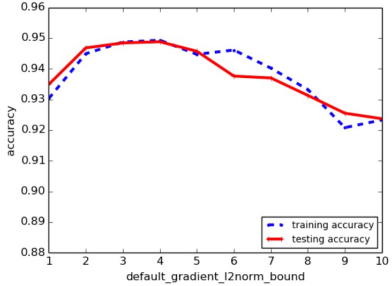
(2) variable hidden units



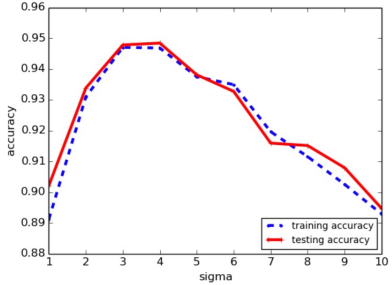
(3) variable lot size



(4) variable learning rate



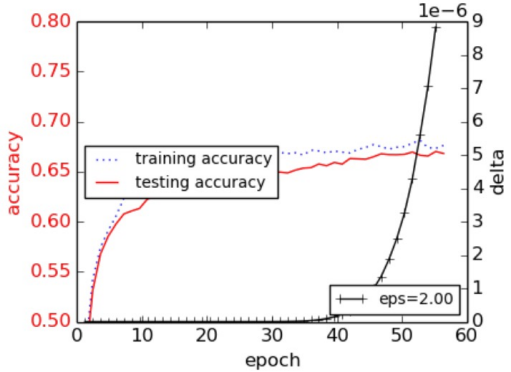
(5) variable gradient clipping norm



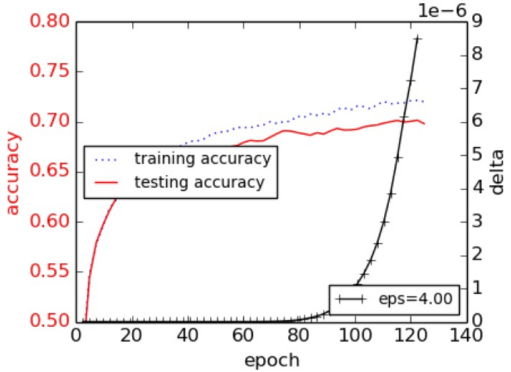
(6) variable noise level

Evaluation

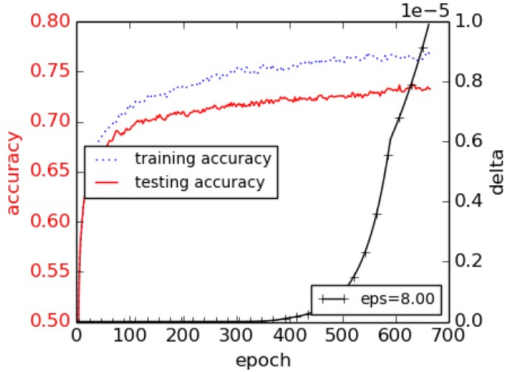
- Impact of Noise
  - Dataset, Models: CIFAR-10, CNN
  - Setup: Clipping threshold ($\mathbf{C}$): 3 | Noise (σ): 6
  - **Summary:**
    - On CIFAR-10, DP-SGD offers reasonable acc. under various privacy costs (**clean:** 80%)
    - The accuracy of private models decreases as we decrease the privacy cost



(1) $\varepsilon = 2$



(2) $\varepsilon = 4$



(3) $\varepsilon = 8$

What does it mean by epsilon = 2/4/6 in cifar-10?

Evaluating differentially private machine learning in practice, Jayaraman and Evans, USENIX Security 2019

Thank You!

Tu/Th 4:00 – 5:50 pm

Sanghyun Hong

https://secure-ai.systems/courses/MLSec/F23

Oregon State University

SAIL

Secure AI Systems Lab