Data Mining Homework 04

Qiu Yihang

June 2023

1 Differential Privacy

Solution. Laplacian distribution with scale b is $\mathrm{Lap}(x|b) = \dfrac{1}{2b}\exp\left(-\dfrac{|x|}{b}\right)$.

L1-sensitivity of 1 requires $\Delta f = \max\limits_{x,y\in\mathcal{N}^{\mathcal{X}},\ \|x-y\|=1} \|f(x)-f(y)\|_1 = 1$.

By Laplace Mechanism, the noise added should be $Y \sim \mathrm{Lap}\left(y|\dfrac{\Delta f}{\varepsilon}\right)$. Thus,

$$b = \frac{\Delta f}{\varepsilon} = \frac{1}{\varepsilon}$$

■

2 Differentially Private Stochastic Gradient Descent

Solution. The algorithm is as follows.

Algorithm 1: DP-SGD

Input: Training dataset $D = \{x_1, ...x_N\}$, loss function $\mathcal{L}(\theta) = \dfrac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\theta, x_i)$.

Parameters: Learning rate η, number of epochs T, batch size L, gradient norm clipping bound C, noise scale σ.

Initialize θ_0;

for $t = 1 \to T$ **do**

- Randomly shuffle a batch B_t with batch size L from D;
- Compute gradient $\boldsymbol{g}_t(x_i) = \nabla\mathcal{L}(\theta_{t-1}, x_i)$;
- Clip gradient $\boldsymbol{g}_t(x_i) \leftarrow \dfrac{\boldsymbol{g}_t(x_i)}{\max\left(1, \dfrac{\|\boldsymbol{g}_t(x_i)\|_2}{C}\right)}$;
- Add noise $\hat{\boldsymbol{g}}_t \leftarrow \dfrac{1}{L}\left(\left(\sum_{x_j\in B_t}\boldsymbol{g}_t(x_j)\right) + \mathcal{N}(0, \sigma^2C^2\boldsymbol{I})\right)$;
- update $\theta_{t+1} \leftarrow \theta_t - \eta\hat{\boldsymbol{g}}_t$

end

Output: θ_T and the overall privacy cost (ε, δ) computed by a privacy accounting method. ■

3 Gradient Matrix Compression

Solution. We use SVD to compress the gradient matrix $\boldsymbol{G}$.

The modified DP-SGD algorithm is as follows.

Algorithm 2: Compressed DP-SGD

Input: Training dataset $D = \{x_1, ...x_N\}$, loss function $\mathcal{L}(\theta) = \dfrac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\theta, x_i)$.

Parameters: Learning rate η, number of epochs T, batch size L, gradient norm clipping bound C, noise scale σ.

Initialize θ_0;

for $t = 1 \to T$ **do**

Randomly shuffle a batch B_t with batch size L from D;

Compute gradient $\boldsymbol{g}_t(x_i) = \nabla \mathcal{L}(\theta_{t-1}, x_i)$ and get the gradient matrix $\boldsymbol{G} \in \mathbb{R}^{n \times p}$;

Perform SVD on $\boldsymbol{G}$ and get $\boldsymbol{G} = \boldsymbol{U\Sigma V}^T$, where $\boldsymbol{U} \in \mathbb{R}^{n \times n}, \boldsymbol{V} \in \mathbb{R}^{p \times p}, \boldsymbol{\Sigma} \in \mathbb{R}^{n \times p}$;

Let $\boldsymbol{B} = \boldsymbol{V}[:, : k] \in \mathbb{R}^{p \times k}$, i.e. the first k-th columns of $\boldsymbol{V}$. (Obvious $\boldsymbol{B}$ is orthogonal);

Let $\hat{\boldsymbol{G}} = \boldsymbol{U}[:, : k]\boldsymbol{\Sigma}[: k, : k] \in \mathbb{R}^{n \times k}$;

for *each row* $\hat{\boldsymbol{g}}_i \in \mathbb{R}^k$*, i.e. the compressed gradient, in* $\hat{\boldsymbol{G}}$ **do**

clip gradient $\hat{\boldsymbol{g}}'_i \leftarrow \dfrac{\hat{\boldsymbol{g}}_i}{\max\left(1, \dfrac{\|\hat{\boldsymbol{g}}_i\|_2}{C}\right)}$;

end

Add noise $\hat{\boldsymbol{g}}'_t \leftarrow \dfrac{1}{L}\left(\sum_{x_j \in B_t} \hat{\boldsymbol{g}}'_i(x_j) + \mathcal{N}(0, \sigma^2 C^2 \boldsymbol{I})\right)$;

Project $\hat{\boldsymbol{g}}'_t \in \mathbb{R}^k$ to the original $\mathbb{R}^p$ and get $\widetilde{\boldsymbol{g}}_t = \hat{\boldsymbol{g}}'_t \boldsymbol{B}^\top$;

update $\theta_{t+1} \leftarrow \theta_t - \eta \widetilde{\boldsymbol{g}}_t$

end

Output: θ_T and the overall privacy cost (ε, δ) computed by a privacy accounting method. ■
