HW 3: The Centralized Curator Model

CS 208 Applied Privacy for Data Science, Spring 2019

Version 1.1: Due Tuesday, April 2, 11:59pm.

Instructions: Submit a single PDF file containing your solutions, plots, and analyses. Make sure to thoroughly explain your process and results for each problem. Also include your documented code and a link to a public repository with your code (such as GitHub/GitLab). Make sure to list all collaborators and references.

1. **Tails, Trimming, and Winsorization:** In all of the parts below, the dataset is $x \in \{0, 1, \ldots, D\}^n$. In all of the implementation parts, you should write code that takes as input $D \in \mathbb{N}$, $n \in \mathbb{N}$, $x \in \{0, 1, \ldots, D\}^n$, and $\varepsilon > 0$.

   (a) Prove that the following algorithm for estimating a Trimmed mean is ε-DP and implement it in code:
   $$M(x) = \frac{1}{.9n} \cdot \left(\sum_{P_{.05} \leq x_i \leq P_{.95}} x_i \right) + \mathrm{Lap}\left(\frac{D}{\varepsilon n}\right),$$
   where $P_{.05}$ and $P_{.95}$ are the 5th and 95th percentile of the dataset. That is, we are applying the Laplace mechanism after removing the bottom and top 5% of the dataset. (Hint: Think about Lipschitz constants.)

   (b) Prove that for large enough n, the analogous algorithm for the *Winsorized* mean is *not* ε-DP:
   $$M(x) = \frac{1}{n} \cdot \sum_{i=1}^{n} [x_i]_{P_{.05}}^{P_{.95}} + \mathrm{Lap}\left(\frac{D}{\varepsilon n}\right),$$
   where $[x]_a^b$ is defined as in Problem Set 2. In Winsorization, we clamp points rather than dropping them. (In class on 3/11, we incorrectly referred to dropping points as Winsorization.) Again, it may be useful to first think in terms of Lipschitz constants.

   (c) In class, we saw how to use the exponential mechanism to an estimate of the median, $P_{.5}$. Describe and implement a version of the exponential mechanism that releases an estimate of the tth percentile P_t of a dataset $x \in \{0, \ldots, D\}^n$ any desired $t \in [0, 100]$. (A direct implementation of the exponential mechanism would require explicitly calculating weights for each of the $D + 1$ possible outputs, which can be too slow for large values of D such as in the parts below. One way to solve this is to bin the elements into fixed, coarser intervals. Alternatively, you can sample more quickly from the output distribution of the exponential mechanism by noting that if you sort the elements of the dataset $x_{i_1} \leq x_{i_2} \leq \cdots \leq x_{i_n}$, then all elements of each interval between x_{i_j} and $x_{i_{j+1}}$ have the same weight, so you can sample by choosing an interval with probability proportional to the sum of weights within it and then sampling uniformly from that interval. Feel free to use either solution below.)

   (d) Implement the following ε-DP algorithm for estimating a Trimmed mean of a dataset: use your algorithm from Part 1c to get $\varepsilon/3$-DP estimates $\hat{P}_{.05}$ and $\hat{P}_{.95}$ of the 5th and 95th percentiles, drop all datapoints that lie outside the range $[\hat{P}_{.05}, \hat{P}_{.95}]$, and then use the

Laplace mechanism to compute an $(\varepsilon/3)$-DP mean of the trimmed data. That is, your code should compute

$$M(x) = \frac{1}{.9n} \cdot \left(\sum_{i:\hat{P}_{.05} \le x_i \le \hat{P}_{.95}} x_i \right) + \text{Lap}\left(\frac{3(\hat{P}_{.95} - \hat{P}_{.05})}{0.9\varepsilon n} \right).$$

(e) Determine whether or not the following analogue for a Winsorized mean is ε-DP: use Part 1c to get $\varepsilon/3$-DP estimates $\hat{P}_{.05}$ and $\hat{P}_{.95}$ of the 5th and 95th percentiles, and output

$$M(x) = \frac{1}{n} \cdot \left(\sum_{i=1}^{n} [x_i]_{\hat{P}_{.05}}^{\hat{P}_{.95}} \right) + \text{Lap}\left(\frac{3(\hat{P}_{.95} - \hat{P}_{.05})}{\varepsilon n} \right).$$

You do not need to formally prove your answer, but you should at least provide an informal explanation.

(f) The dataset `MaPUMS5full.csv` provides the 5% PUMS Census file for Massachusetts. For $\varepsilon = 1$ and $D = 1,000,000$, compare the RMSE between DP means and the actual means for each PUMA in Massachusetts,[1] for DP means calculated using (i) the ordinary Laplace mechanism for a mean (remembering to clamp your data to the range!) and (ii) the algorithm from Part 1d. Also show box-and-whisker plots of the DP released means for each PUMA by these algorithms, noting the true means. You should probably order these by mean income, or perhaps skew of income, or anything you think reveals an interesting pattern. Give an intuitive explanation of the kinds of datasets on which algorithm (i) is likely to perform better than algorithm (ii) and vice-versa. Describe any modifications you might propose would increase the utility (at the same level of privacy preservation) for data similar to this income example.

2. **Composition:** Suppose you have a global privacy budget of $\varepsilon = 1$ (and are willing to tolerate $\delta = 10^{-9}$) and you want to release k count queries (i.e. sums of Boolean predicates[2]) using the Laplace mechanism with an individual privacy loss of ε_0. By basic composition, you can set $\varepsilon_0 = \varepsilon/k$. Using the advanced composition theorem, you can set $\varepsilon_0 = \varepsilon/\sqrt{2k\ln(1/\delta)}$. We have provided you with code from PSI for the "optimal" composition theorem for differential privacy that calculates the largest value of ε_0 that ensures global (ε, δ)-DP as a function of ε, δ, and k.[3] For each of these choices, plot (on the same graph) the standard deviation of the Laplace noise added to each query as a function of k, and find the smallest values of k where the advanced and optimal composition theorems strictly improve upon the basic composition theorem.

3. **Synthetic Data:** Expanding the template from class, and using again `MaPUMS5Full.csv`, create a DP three-way histogram[4] release of income, education and age. You do not need to graph this histogram, just compute the release for each binned combination of the variables. From this, you should be able to generate synthetic data of these three variables. Run a linear regression

[1] You can assume that the N in each PUMA is public information.

[2] A Boolean predicate is a function that returns a 0 or a 1. An example of a count query might be the sum of bits for all college students.

[3] See the function `update_parameters` in `/examples/wk5_centralized/psiExamples.r` or `psiExamples.ipynb`.

[4] That is, a histogram representation counting the occurrences of having all possible combinations of the three binned variables.

as a post-process on your synthetic data, predicting income from education and age[5] using the equation:

$$\text{Income}_i = \beta_0 + \beta_1 \text{Education}_i + \beta_2 \text{Age}_i + \nu_i; \qquad \nu_i \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

Let $\beta^* = \{\beta_0^*, \beta_1^*, \beta_2^*\}$ be the coefficients in the full sensitive data, while $\tilde{\beta}$ the DP release we generate. The mean-squared error of a DP release of $\tilde{\beta}$ can be decomposed into the contributions of bias and variance as:

$$\text{MSE}(\tilde{\beta}) = \text{bias}(\tilde{\beta})^2 + \text{var}(\tilde{\beta}) = (\text{E}[\beta^* - \tilde{\beta}])^2 + \text{E}[(\bar{\tilde{\beta}} - \tilde{\beta})^2] \tag{2}$$

For this calculation, we are taking the (sensitive) regression coefficients β^* on the entire dataset as the true values of β. Show the contributions to MSE of the bias and variance of the DP-regression coefficients.[6]

As a baseline to decide if these squared bias and error terms are large, we can compute the MSE due simply to sampling, by bootstrapping with replacement new datasets in which we compute new (sensitive) regression estimates $\hat{\beta}$ on the bootstrapped data and compute MSE($\hat{\beta}$). How do the bias and variance terms due to creating DP-releases compare to the this numerical estimate of the error introduced by sampling?

4. **BONUS:** Using your developed understanding of differential privacy, and the described use case in the Gaboardi *et al.* PSI paper, reexamine the deployed instance of the PSI budgeting tool, available at `http://psiprivacy.org`. Provide any feedback that you think would make the interface easier for the intended non-expert "data owner" user to budget a DP-release, or would otherwise improve the system. (Note: Insightful, considered feedback will receive 1/2 point bonus, and feedback that strikes us a revelatory or particularly intriguing idea will receive 1 point bonus and a note of thanks in a future paper draft.)

5. **Final Project:** By **April 9**, submit a couple of pages giving a detailed description of what your final project will look like. You should be able to clearly state your research questions, briefly articulate how your project relates to what has been done in the past, describe the approach you are taking, give your timeline for completing various aspects of the project, and *discuss your fallback plan in case you don't obtain the results that you're hoping to obtain.*

[5] You will likely find that `log(income)` has a more linear relationship with your other two variables, so feel free to shift from `income` to `log(income)` if you prefer. However, you will need to decide how to treat zero values in income; one option is to clip the lower bound of income to some small positive value.

[6] To numerically compute the expectations, simply repeat your simulation many times and average.