DP Review

- Defn: \( \forall \text{neighboring } D, D', S \text{ of outputs:} \)
  \[ \Pr[A(D) \in S] \leq e^{\varepsilon} \Pr[A(D') \in S] \]

- Randomized Response:
  \[ \Pr \left( \frac{1 + p}{1 - p} \right) - \text{DP} \]

- Laplace: Output \( f(x) + \frac{\Delta f}{\varepsilon} \)
  \( \varepsilon \)-DP, accuracy \( \Delta f / \varepsilon \)

- Exponential: Output \( y \approx e^{\frac{1}{2b}} e^{-\frac{1}{\varepsilon} \Delta u} \)
  \( \varepsilon \)-DP, within \( \frac{\Delta u}{\varepsilon} \) of optimal (may be hard to sample)
Nice Application of Exp. Mech.

- Let $D$ be some dataset.
- E.g., rows with numeric vars/cols $x_1, x_2, \ldots, x_d$.
- Pick some stats we'd like to preserve, e.g.:
  - Averages of $x_i$,
  - Correlations of $x_i, x_j$.
- Output of algo will be another dataset $\hat{D}$. 


Now define $u(D, \hat{D})$ to measure how well $\hat{D}$ agrees with $D$ on preserved stats.

E.g.,

$$u(D, \hat{D}) = \max_i \left| \frac{\text{ML}(X_i)}{D} - \frac{\text{ML}(X_i)}{\hat{D}} \right|$$

Draw $\hat{D}$ from Exp. Mech using $D, \hat{D} \Rightarrow 3$-DP.

Now publish $\hat{D}$!
So we have some good general primitives or tools. What about methods for combining them to obtain richer DP algos? Would like programmability for DP.
Method 1: Post-Processing

• Let $A(D_i)$ be an E-DP algorithm

• Let $B(A(D_i), D_2, D_3, \ldots, D_e)$ be an algo taking $A(D_i)$ as input, as well as other dataset

• Then $B$ is E-DP in $D_i$

• DP cannot be "undone", even by a non-DP algo
Method 2: Repetition

- Let $A_1, A_2, \ldots, A_e$ be $\epsilon$-DP algs
  - Then $B(D) = (A_1(D), \ldots, A_e(D))$ is $\ell \cdot \epsilon$-DP

- $\ell \cdot \epsilon$ is privacy loss from multiple DP computations

More general: if $A_i$ is $\epsilon_i$-DP, then $B$ is

$$\sum_{i=1}^{e} \epsilon_i$$
A Better Repetition Bound

- Say $A$ is $(\varepsilon, \delta)$-DP if
  \[ \Pr[A(D) \in S] \leq e^\varepsilon \Pr[A(P) \in S] + \delta \]

If $\Pr[A(D) \in S] < \delta$, this is costing a lot!

- Then $A_1, \ldots, A_e$ are $\varepsilon$-DP if $B(D) = (A_1(D), \ldots, A_e(D))$ is $(\varepsilon', \delta)$-DP where
  \[ \varepsilon' = \sqrt{k \log(1/\delta)} \cdot \varepsilon \]
Method 3: General Composition

Even better: same kind of $(\sqrt{k \log(1/\epsilon)} \cdot \delta)$-DP result holds even if \( B \) an adaptive sequence of compositions of \( E \)-DP algs!

**True programmability**

**Even less privacy loss for structured problems**
Applications of DP
Application: Machine Learning

• Basically any ML method that learns models in a "statistical" fashion can be made DP

• Linear/logistic regression, decision trees, boosting, neural networks, reinforcement learning, PCA, clustering...

• Not covered: "equation-solving" methods
Application: Game Theory & Mechanism Design

- Major concern: incentivizing truthfulness
- E.g. bids in an auction (second price, VCG); origins & destinations in Waze
- If you can design your mechanism to be DP, get (approx.) truthfulness for free
Application: P-hacking & the reproducibility crisis

Problem: (communal) overfitting on (shared) datasets

E.g. CIFAR competitions

Potential solution: DP leaderboards

E.g. only publish improvements larger than 1%, can only happen ≤100 times
Real-World Deployments

- Apple OS
- Google Chrome & open source DP
- 2020 U.S. Census
- COVID-19