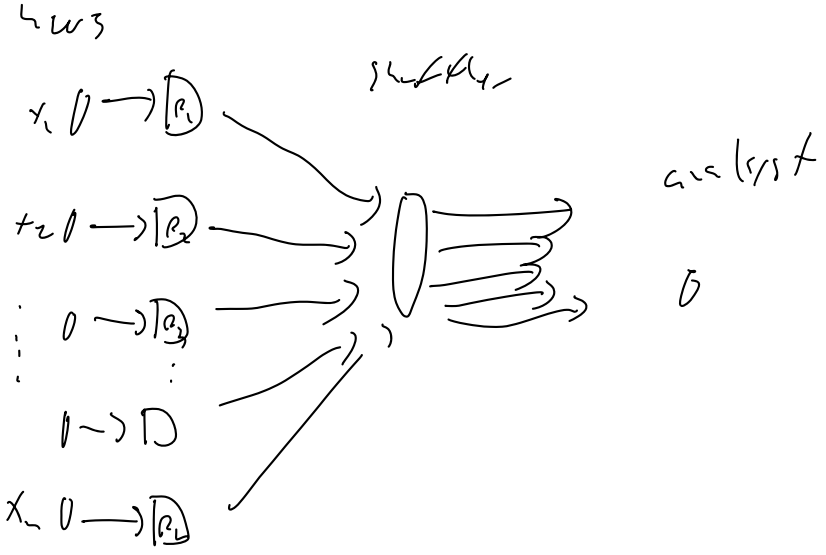
Shuffle DP:

Central DP: Trusted curator

Local DP: Untrusted curator

Shuffle DP: in the middle!

- Trusted "shuffler": randomly shuffles messages from users/individuals
- Bigger trust assumption than local, smaller " " " centralized
- Morally similar to crypto: minimize circumstances in which assume is secure
- Intuition: "amplification by shuffling"



Let S be a shuffler: algorithm that outputs a uniformly random permutation of inputs

Let R be ε_0-DP run at each user

Let A be alg. run by analyst

Def: (R, A) is (ε, δ)-shuffle DP if

$$S(R(x_1), R(x_2), \ldots R(x_n)) \text{ is } (\varepsilon, \delta)\text{-DP}$$

Idea: after shuffling, dist. of message vectors the same no. of (ε, δ)-DP!

Discussion: - can also define "robust" versions. What if some users violate protocol? Still want to be DP if $\geq \frac{1}{2}$ users are honest.

- Assume n is public, known.

Binary Sums:

- Each $x_i \in \{0,1\}$, return $X = \sum_{i=1}^{n} x_i$
- Central DP: $\tilde{X} = X + \text{Lap}(1/\varepsilon) \Rightarrow E[|X - \tilde{X}|] = O(1/\varepsilon)$
- Local DP: RR $\Rightarrow E[|X - \tilde{X}|] = O\left(\frac{1}{\varepsilon}\sqrt{n}\right)$, best possible. (even for (ε, δ)-DP)
- Shuffle:

RR w/ diff parameters!

$$\tilde{x}_i = \begin{cases} \text{Ber}(1/2) & \text{w/ prob. } \rho \\ x_i & \text{w/ prob. } 1-\rho \end{cases}$$

In local: $p = \frac{2}{e^\varepsilon + 1}$

$\Pr[\hat{x}_i = x_i] = 1 - p + \frac{p}{2} = \frac{e^\varepsilon - 1}{e^\varepsilon + 1} + \frac{1}{e^\varepsilon + 1} = \frac{e^\varepsilon}{e^\varepsilon + 1}$ ✓

In shuffle:

Let $p = \Theta\left(\frac{1}{n} \log \frac{1}{\delta}\right) \le \frac{1}{2}$

Thm: (ε, δ)-DP.

Pf sketch: Due to shuffling, adversary given some $\{\tilde{x}_i\}$

- More formally: suppose given $\sum_{i=1}^n \tilde{x}_i$.
  - Choose a binary string of length n uniformly from all elements of $\{0,1\}^n$ with sum $\sum_{i=1}^n \tilde{x}_i$
  - Precisely same distribution as $S(\tilde{x}_1, \dots, \tilde{x}_n)$!
    - $\Rightarrow$ given $\sum_{i=1}^n \tilde{x}_i$, can generate sample from $S(\tilde{x}_1, \dots, \tilde{x}_n)$
    - $\Rightarrow$ any alg which uses sample from $S(\tilde{x}_1, \dots, \tilde{x}_n)$ (adversary) can be replaced by one which only knows $\sum_{i=1}^n \tilde{x}_i$
    - $\Rightarrow$ block acts only knows $\sum_{i=1}^n \tilde{x}_i$

- What distribution is $\sum_{i=1}^n \tilde{x}_i$ a sum from?

⇒ equivalent to centralized mechanism:

- Let $S \sim Bin(n, p)$
- Choose set $H \subseteq [n]$ with $|H| = S$ u.a.r.
- Return $\sum_{i \in H} x_i + Bin(S, \frac{1}{2})$

Binomial basically Gaussian when S is large!
- Like adding Gaussian noise ⇒ (ε, δ)-DP!

- Need $S \geq \Omega\left(\frac{1}{\varepsilon^2} \log \frac{1}{\delta}\right)$

⇒ Set $p \geq \Omega\left(\frac{1}{\varepsilon^2 n} \log \frac{1}{\delta}\right)$

⇒ S is as desired w.prob. $\geq 1 - \delta$ ✓

So private, what's the accuracy?

Unbiased estimator: $\tilde{X} = \frac{1}{1-p}\left(\sum_{i=1}^{n} x_i - \frac{1}{2}np\right)$

$$\left(\text{s.t. } E[\tilde{X}] = \frac{1}{1-p}\left(\sum_{i=1}^{n} E[\tilde{x}_i] - \frac{1}{2}np\right)\right.$$

$$= \frac{1}{1-p}\left(\sum_{i=1}^{n}\left((1-p)x_i + \frac{p}{2}\right) - \frac{1}{2}np\right.$$

$$\left.\left. = \sum_{i=1}^{n} x_i \right)\right)$$

Plugin/const. prob., $|X - \tilde{X}| \leq O\left(\frac{1}{\varepsilon}\sqrt{n \log \frac{1}{\delta}}\right)$

Chernoff bound on S:

$$\Pr\left(S \geq (1+\eta) E[S]\right) \leq e^{-\eta^2 E[S]}$$

$\Rightarrow$ set $\eta = \frac{1}{\sqrt{E[S]}} = \frac{1}{\frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}}$

$\Rightarrow$ w/ const. prob., $S \le \left(1 + \frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}\right) \cdot E[S]$

$\Rightarrow$ $S - E[S] \le \frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}\, E[S] \le O\left(\frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}\right)$ ✓

$\Rightarrow$ w/ const. prob., error $\le O\left(\frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}\right)$ ✓

Notes: by using more messages, can get to $O\left(\frac{1}{\varepsilon}\right)$-error!

Private Selection:

- Central: need $\frac{\log d}{\varepsilon}$
- Local: $\frac{\sqrt{n}\log d}{\varepsilon}$
- Shuffle: same 1-bit encoding trick as Local!

$\Rightarrow$ need $O\left(\frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}\log d\right)$ gap!

Connections to central/local DP:

Thm: If a single-message protocol is ε-shuffle DP,
Then w/ const shuffler must be ε-DP

$\Rightarrow$ For pure DP, shuffling doesn't help! (single message)

clearly doesn't hold for approximate DP.

Thm: If single-message protocol is (ε, δ)-shuffle DP, then whole shuffler is $(\varepsilon + \ln n, \delta)$-DP.