CSC2412: Private Multiplicative Weights

Sasho Nikolov

Query Release

Reminder: Query Release

Recall the query release problem:

$q_i : \mathcal{X} \to \{0,1\}$

$q_i(X) = \frac{1}{n} \sum_{i=1}^{n} q_i(x)$

where $X = \{x_1, \dots, x_n\}$

- Workload $Q = \{q_1, \dots, q_k\}$ of k counting queries

$$Q(X) = \begin{pmatrix} q_1(X) \\ \vdots \\ q_k(X) \end{pmatrix} \in [0,1]^k.$$

- Compute, with (ε, δ)-DP, some $Y \in \mathbb{R}^k$ so that

$$\max_{i=1}^{k} |Y_i - q_i(X)| \le \alpha,$$

with probability $\ge 1 - \beta$.

Motivating example

ℓ-wise marginals queries:

- $\mathcal{X} = \{0,1\}^d$ i.e. d binary attributes
- a query $q_{S,a}$ for any $S = \{i_1, \dots, i_{\underset{\ell}{k}}\} \subseteq [d]$ and $a = (a_{i_1}, \dots, a_{i_{\underset{\ell}{k}}})$:

$$q_{S,a}(x) = \begin{cases} 1 & x_{i_j} = a_{i_j} \ \forall i_j \in S \\ 0 & \text{otherwise} \end{cases}.$$

E.g., "smoker and female?", "smoker and over 30?", "smoker and heart disease?", etc.

Q_ℓ = workload of all ℓ-wise marginal queries on $\{0,1\}^d$

$$|Q_\ell| = \binom{d}{\ell} \cdot 2^\ell \approx \left(\frac{2d}{\ell}\right)^\ell$$

What do we know?

ε- DP :

For ℓ-wise marg.

$$n \gg \frac{\boxed{d^{\ell}}\cdot \ell \cdot \log d}{\alpha \varepsilon}$$

Using the Laplace noise mechanism, we can answer k counting queries with noise $\leq \alpha$ with prob $\geq 1-\beta$ when

$$n \gg \frac{k \log(k/\beta)}{\varepsilon \alpha}$$

$\underline{(\varepsilon,\delta)\text{-DP}}$: Using the Gaussian noise mechanism:

$$n \gg \frac{\boxed{d^{\ell/2}}\sqrt{\ell \cdot \log d}\,\sqrt{\log \frac{1}{\delta}}}{\alpha \varepsilon}$$

$$n \gg \frac{\sqrt{k \log(k/\beta)}\cdot\sqrt{\log 1/\delta}}{\varepsilon \alpha}$$

Private Multiplicative Weights

We will see an algorithm that achieves:

- under ε-DP, error α with probability $1-\beta$ when

$$n \gg \frac{\log(k/\beta)\log(|\mathcal{X}|)}{\alpha^3\varepsilon}.$$

- under (ε,δ)-DP, error α with probability $1-\beta$ when

$$n \gg \frac{\log(k/\beta)\sqrt{\log(|\mathcal{X}|)\log(1/\delta)}}{\alpha^2\varepsilon}.$$

β constant

ℓ-wise marginals

$$n \gg \frac{d \cdot \ell \cdot \log d}{\alpha^3 \varepsilon}$$

Learning a distribution

A probability view

allowed to be a multiset

We can think of $X = \{x_1, \dots, x_n\}$ as a probability distribution $\underline{p}$: over $\mathcal{X}$

uniform over $x_1, \dots, x_n$

$$\mathbb{P}_{x \sim p}(x = y) = \frac{|\{i : x_i = y\}|}{n}$$

Then, for any counting query $q : \mathcal{X} \to \{0, 1\}$,

$$q(X) = \frac{1}{n} \sum_{i=1}^{n} q(x_i) = \sum_{x \in \mathcal{X}} q(x) \cdot \frac{|\{i : x_i = x\}|}{n} = \mathbb{E}_{x \sim p}\, q(x) =: \underline{q(p)}$$

i.e. $q(X)$ = expectation of q under the empirical distribution of X

Learning a distribution

Query release problem

distributions over $\mathcal{X}$

Task: Learn an approximation $\hat{p}$ of the empirical distribution p such that

workload of queries

$$\forall q \in Q : |q(\hat{p}) - q(p)| \leq \alpha.$$

$\underset{x \sim \hat{p}}{\mathbb{E}}\, q(x)$

If we can do this, we can release answers $q(\hat{p})$ for all $q \in Q$

Trick (again): We will assume that if q is asked, then $1-q$ is also asked

$\Rightarrow$ enough to make sure $\max_{q \in Q} q(\hat{p}) - \underbrace{q(p)}_{= q(X)} \leq \alpha$

Bounded mistake learner

does not need to know p

update algorithm

Distribution learning algorithm U:

query on which $\hat{p}$ makes a mistake

- takes a $\hat{p}$ and q such that $q(\hat{p}) - q(p) > \alpha$ → $\hat{p}$ makes a mistake on q
- returns a new distribution $\hat{p}' = U(q, \hat{p})$ → an improvement of $\hat{p}$

Suppose that $\hat{p}_0 =$ uniform over $\mathcal{X}$ and $\hat{p}_t = U(\hat{p}_{t-1}, q_t)$.

initial guess

↳ keep improving $\hat{p}_t$ by pointing out mistakes

U makes at most L mistakes if any such sequence $\hat{p}_0, \hat{p}_1, \ldots, \hat{p}_\ell$ must have $\ell \leq L$.

After making L mistakes (and L improvements), $\hat{p}_L$ must be accurate for all q

Multiplicative Weights Learner

Theorem

There exists a distribution learner U that makes $\boxed{L \leq \frac{4 \ln |\mathcal{X}|}{\alpha^2}}$ *mistakes.*

The Learner Multiplicative Weight Update Algorithm

Reminder : $q(\hat{p}) - q(p) > \alpha$

I.e. $\hat{p}$ gives too much weight to x s.t. $q(x) = 1$

$q(\hat{p}) = \mathbb{E}_{x \sim \hat{p}} q(x)$

$\hat{p}(x)$ = prob of x under $\hat{p}$

```
U(q, p̂):
    $\forall x \in \mathcal{X} : \tilde{p}(x) = \hat{p}(x) e^{-\boxed{\eta} q(x)}$
    $\hat{p}'(x) = \frac{\tilde{p}(x)}{\sum_{y \in \mathcal{X}} \tilde{p}(y)}$
    return $\hat{p}'$
```

η: parameter, to be set later

$\hat{p}(x)e^{-\eta q(x)}$ → decrease $\hat{p}(x)$ if $q(x) = 1$

$\hat{p}'(x) = \frac{\tilde{p}(x)}{\sum_{y \in \mathcal{X}} \tilde{p}(y)}$ → normalize to get a prob. distribution

Why it works

KL-divergence: $\overset{\text{potential function}}{\overline{D(p\|\hat{p}_t)}} = \sum_{x\in\mathcal{X}} p(x)\log\frac{p(x)}{\hat{p}_t(x)} = \mathbb{E}_{x\sim p}\log\left(\frac{p(x)}{\hat{p}_t(x)}\right)$

1. $D(p\|\hat{p}_0) \le \log|\mathcal{X}|$ because $\hat{p}_0$ is uniform ⟺ $\hat{p}_0 = \frac{1}{|\mathcal{X}|}$

$$D(p\|\hat{p}_0) = \sum_{x\in\mathcal{X}} p(x)\left(\log(|\mathcal{X}|) + \log p(x)\right) = \log|\mathcal{X}| - \overset{\text{entropy of } p}{\boxed{\sum_{x\in\mathcal{X}} p(x)\log\frac{1}{p(x)}}} \le \log|\mathcal{X}|$$

2. $D(p\|\hat{p}_t) \ge 0$ for all t

initial guess $\hat{p}_0$

find mistake q_1

$\hat{p}_1 = U(\hat{p}_0, q_1)$

find mistake q_2

$\hat{p}_2 = U(\hat{p}_1, q_2)$ ...

3. $\underline{D(p\|\hat{p}_t) - D(p\|\hat{p}_{t-1}) \le \frac{\eta}{2}(q_{t-1}(p) - q_{t-1}(\hat{p}_{t-1})) + \frac{\eta^2}{4}} < -\frac{\alpha^2}{4}$

$$\boxed{q_{t-1}(\hat{p}_{t-1}) - q_{t-1}(p) > \alpha}$$

Set $\eta = \alpha$

$$-\frac{\eta}{2}\cdot\alpha + \frac{\eta^2}{4} = -\frac{\alpha^2}{4}$$

must terminate in $\le \frac{4\log|\mathcal{X}|}{\alpha^2}$ steps

Private Multiplicative Weights

Idea for private algorithm

- Start with $t = 0$, $\hat{p}_0$ uniform.
- Privately find the most wrongly answered query $q \in Q$
  - If $q(\hat{p}_t) - q(p) < \alpha$, output $\hat{p}_t$ → all queries in Q have error $\leq \alpha$
  - Else set $\hat{p}_{t+1} = U(\hat{p}_t, q)$ and increase t

↓

q is a mistake

→ terminates after $\leq L = \frac{4 \log |\mathcal{X}|}{\alpha^2}$ iterations

The algorithm in detail

$L = \frac{4 \ln|\mathcal{X}|}{\alpha^2}$

ε_0 parameter, to be set in the priv. analysis

```
$\hat{p}_0 =$ uniform over $\mathcal{X}$
for $t = 0 .. L-1$
    Sample $q \in Q$ w/ prob $\propto \exp\left(\frac{n(q(\hat{p}_t) - q(p))}{2\varepsilon_0}\right)$
    $Y_t = q(\overset{q(X)}{p}) + Z_t,\ Z_t \sim \mathrm{Lap}(0, \frac{1}{\varepsilon_0 n})$
    if $q(\hat{p}_t) - Y_t > 2\alpha$
        $\hat{p}_{t+1} = U(\hat{p}_t, q)$
    else Output $\hat{p}_t$
```

want q to achieve approx worst error

$|Y_t - q(p)| \le \alpha$

$\Rightarrow q(\hat{p}_t) - q(p) > \alpha$

↳ max error $\le q(\hat{p}_t) - q(p) + \alpha$

$\le q(\hat{p}_t) - Y_t + 2\alpha \le 3\alpha$

$\Rightarrow$ exponential mechanism w/ score $q(\hat{p}_t) - q(p)$

$= q(\hat{p}_t) - q(X)$

sensitivity of the score $= \frac{1}{n}$

$\Rightarrow$ exponential mech w/ privacy parameter ε_0

Laplace noise mech w/ priv param ε_0

Privacy analysis

Approach: bound privacy loss per iteration.
use composition theorem to bound total priv. loss

Priv loss per iteration: Exp mech ε_0-DP

Lap mech $\underline{\varepsilon_0\text{-DP}}$

$2\varepsilon_0$-DP by composition

Total of $\leq L$ iterations

$\leadsto$ total priv. loss $\leq 2L\varepsilon_0$-DP

Set $\varepsilon_0 = \frac{\varepsilon}{2L} = \frac{\varepsilon\alpha^2}{8\ln|\mathcal{X}|}$

Accuracy analysis

$$\mathbb{P}(|Z_t| \geq \alpha) \leq e^{-n\varepsilon_0\alpha}$$

1) We want that w/ prob $\geq 1-\beta$

$$\forall t \quad |Y_t - q(p)| \leq \alpha$$

↳ query in round t

Laplace mechanism w/ $\leq L$ adaptive queries

enough to have $n \geq \frac{\ln(L/\beta)}{\varepsilon_0 \alpha} = \frac{2L \ln(L/\beta)}{\varepsilon \alpha} \approx \frac{L \log(k/\beta)}{\varepsilon \alpha}$

2) w/ prob $\geq 1-\beta$

at every iteration $\quad q(\hat{p}_t) - q(p) \geq \max_{q' \in Q} q'(\hat{p}_t) - q'(p) - \alpha$

if $n \gg \frac{\log(kL/\beta)}{\varepsilon_0 \alpha} = \frac{2L \log(kL/\beta)}{\varepsilon \alpha} \approx L \frac{\log(k/\beta)}{\varepsilon \alpha}$