

Theory of Differential Privacy

Lecture 9:

Private Multiplicative Weights

EN.601.438/638 Fall 2026

Lydia Zakynthinou

Johns Hopkins University

Sept 30, 2026

Recap

Setup:

We continue with normalized linear queries.

$$f: \mathcal{X}^* \rightarrow [0, 1] \quad , \quad f(x) = \frac{1}{n} \sum_{i=1}^n f(x_i).$$

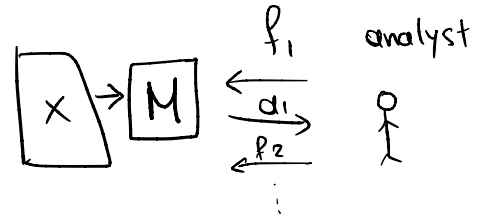
Small DB: Given workload $Q = \{f_1, \dots, f_k\}$, use the Exp. Mech. to find (ϵ -DP) synthetic dataset $\hat{y}$ of size $\approx \frac{\log k}{\epsilon^2}$ and answer all queries f_i on this dataset.

$$\text{Goal: } \max_{f \in Q} |f(x) - f(\hat{y})| \leq \alpha \quad \text{w.p. } 1 - \beta.$$

- ⊖ Offline: needs to know Q in advance
- ⊖ Computationally inefficient.

Setup: Online linear query release

Private database $X = (x_1, \dots, x_n) \in \mathcal{X}^n$, $N = |\mathcal{X}|$.



At round t , the analyst gives a query

$$f_t: \mathcal{X}^* \rightarrow [0, 1], \quad f_t(x) = \frac{1}{n} \sum_{i=1}^n f_t(x_i).$$

The mechanism M releases an answer a_t . (The next query f_{t+1} may depend on previous answers $a_t, a_{t-1}, \dots, a_1$.)

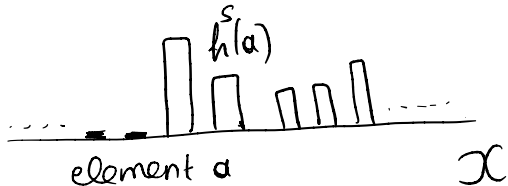
Goal: for a stream of at most k queries:

$$\epsilon\text{-DP and } \Pr\left[\max_{t \leq k} |a_t - f_t(x)| \leq 3\alpha\right] \geq 1 - \delta.$$

Hypothesis distribution

Our mechanism M will maintain a public distribution $h^s \in \Delta(\mathcal{X})$.

$$h^s \in [0, 1]^N$$



For query f , define $f(h^s) = \sum_{a \in \mathcal{X}} f(a) \cdot h^s(a)$.

- h^s represents a synthetic database
- s is the update counter, t is the rounds counter.

Recall: (Numeric) Sparse Vector Technique

Input: Adaptive stream of at most $L=2k$ queries g , each of sensitivity $\frac{1}{n}$.

Threshold $\tau = 2\alpha$

Cap c on number of "YES"s (accompanied by numeric outputs)

Privacy budget ϵ .

Output: for each g : either "NO" or "YES" and a numeric value E .

It stops after c "YES"s.

Guarantees: (✓) ϵ -DP

(✓) There is a universal constant C such that if

$$C \frac{c}{\epsilon n} \log \frac{4kc}{\beta} \leq \frac{\alpha}{2}$$

Then w.p. $1-\beta$, simultaneously for every processed query g :

- If output "NO", then $g(x) \leq 2\alpha + \frac{\alpha}{2}$

- If output "YES", then $g(x) \geq 2\alpha - \frac{\alpha}{2}$ and $|E - g(x)| \leq \frac{\alpha}{2}$.

Recall: (Numeric) Sparse Vector Technique

(Numeric) Sparse Vector Technique

What if we want to find c above-threshold values?

$\text{Sparse}(x, \{f_i\}, T, c, \epsilon_0)$

1. Set $\text{count} = 0$
2. Draw $\hat{T}_0 = T + \text{Lap}(\frac{2}{\epsilon_0})$
3. For each $i = 1, 2, \dots$

Draw $v_i \sim \text{Lap}(\frac{4}{\epsilon_0})$

If $f_i(x) + v_i > \hat{T}_{\text{count}}$:

output $a_i = \text{YES}$ and $\text{count}++$, output $E_i = f_i(x) + z_i$, $z_i \sim \text{Lap}(\frac{1}{\epsilon_0})$

If $\text{count} = c$, halt.

Otherwise draw $\hat{T}_{\text{count}} = T + \text{Lap}(\frac{2}{\epsilon_0}) \leftarrow$ fresh noise

Otherwise output $a_i = \text{NO}$ and continue.

Remarks

- Numeric Sparse Vector is $\epsilon = (c \cdot \epsilon_0 + c \cdot \epsilon_0)$ -DP by basic composition

so I have to set $\epsilon_0 = \frac{\epsilon}{2c}$.

- We defined it for queries of sensitivity 1, but works for any bound Δ .
(in our case today $\Delta = \frac{1}{n}$).
- Stateful algorithm.

Today's algorithm: Private Multiplicative Weights

Uses NumericSparse, based on Multiplicative Weight Updates.

PMW

Set $c = 1 + \left\lceil \frac{4 \log N}{\alpha^2} \right\rceil$. Initialize $h^0(a) = \frac{1}{N} \quad \forall a \in \mathcal{X}$, Counter $s=0$.

For each incoming query $f_t, t=1, \dots, K$:

1. If $s=c$, halt.

2. Set $b_t = f_t(h^s)$

creates one NumericSparse instance and maintains

3. Submit query $g_t^+(\cdot) = f_t^+(\cdot) - b_t$ to NumericSparse. its state.

If it returns E , set $v_t = b_t + E$

Otherwise submit query $g_t^-(\cdot) = b_t - f_t(\cdot)$ to NumericSparse.

If it returns E , set $v_t = b_t - E$.

4. If both responses are "NO", release $o_t = b_t$ and leave h^s unchanged.

5. If a numeric response occurred, release $o_t = v_t$ and update

$$h^{s+1} = MWU_{\alpha/2}(h^s, f_t, v_t), \quad s = s+1.$$

subroutine TBD

Privacy Guarantee of PMW

Thm. PMW is ϵ -DP.

Proof sketch: Observe that only NumericSparse has access to the dataset x !

In every round we submit at most 2 queries (g_t^+, g_t^-) , so $2K$ total.

Sensitivity of g^+, g^- is at most $\Delta f_t \leq 1/n$.

In every round, it returns at most 1 numeric value $\rightarrow$ if so, we update $s++$

$\rightarrow$ we halt when $s=c \Rightarrow$ returns at most c values in total.

By its guarantees, it's ϵ -DP. All else is post-processing.

Accuracy guarantee of PMW

Thm. Assume $C \cdot \frac{c}{\epsilon n} \log \frac{4kc}{\delta} \leq \frac{\alpha}{2}$, $c = 1 + \left\lceil \frac{4 \log N}{\alpha^2} \right\rceil$ (from NumericSparse)

Then for every $x \in \mathcal{X}^n$ and every adaptive stream of at most k queries, w.p. $1 - \delta$, PMW answers all k queries and

$$\max_{t \in [k]} |f_t(x) - a_t| \leq 3\alpha.$$

Proof strategy

Rely on MW.

- if no update, $b_t = f_t(h^s)$ is accurate.
 - if $\exists$ update, the released v_t is accurate.
 - Each update gives a "valid" MW correction.
 - MW has at most $\frac{4 \log N}{\alpha^2}$ valid corrections.
- $\Rightarrow$ The cap $c = 1 + \left\lceil \frac{4 \log N}{\alpha^2} \right\rceil$ is not exhausted and every query is answered.

} By NumericSparse.

} by MW.

MW Update rule

MWU _{η} (h, f, v)

Input: distribution h , query $f: \mathcal{X} \rightarrow [0, 1]$, approximate answer v , learning rate $\eta \in (0, 1]$

1. Choose cost function $r(a) = \begin{cases} f(a) & , v < f(h) \\ 1 - f(a) & , v \geq f(h) \end{cases}, a \in \mathcal{X}$
2. Reweight and normalize:

$$h^{\text{new}}(a) = \frac{h(a) \cdot e^{-\eta r(a)}}{Z} \quad \text{where} \quad Z = \sum_{b \in \mathcal{X}} h(b) \cdot e^{-\eta r(b)}$$

3. Output h^{new} .

PMW

Set $c = 1 + \left\lceil \frac{4 \log N}{\alpha^2} \right\rceil$. Initialize $h^0(a) = \frac{1}{N} \quad \forall a \in \mathcal{X}$, Counter $s=0$.

For each incoming query $f_t, t=1, \dots, K$:

1. If $s=c$, halt.

2. Set $b_t = f_t(h^s)$

3. Submit query $g_t^+(\cdot) = f_t^+(\cdot) - b_t$ to NumericSparse.

If it returns E , set $v_t = b_t + E$

Otherwise submit query $g_t^-(\cdot) = b_t - f_t(\cdot)$ to NumericSparse.

If it returns E , set $v_t = b_t - E$.

4. If both responses are "NO", release $a_t = b_t$ and leave h^s unchanged.

5. If a numeric response occurred, release $a_t = v_t$ and update

$$h^{s+1} = \text{MWU}_{\alpha/2}(h^s, f_t, v_t), \quad s = s+1.$$