The Sparse Vector Technique and online query answering

CompSci 590.03
Instructor: Ashwin Machanavajjhala
Offline QA

• Till now:

Given a workload of queries $W$, design a differentially private algorithm that answers these queries on database $x$ with least error.
Offline QA

• Is reasonable in many settings

• Census Data:
  – Set of tables that are released are prespecified

• Spatial data analysis
  – Want counts of people within contiguous regions
  – Corresponds to a workload of all range queries

• Distributions of values of an attribute
  – CDF is the prefix workload
Online QA

• But, often one does not know what queries to pose before looking at the data

• Online QA:
  – Analyst poses queries one after the other
  – Next query can be determined based on the answer to the previous query

• How to do online QA with least error?
Sequential Composition to the rescue

• If $M_1, M_2, ..., M_k$ are algorithms that access a private database $D$ such that each $M_i$ satisfies $\varepsilon_i$-differential privacy,

then the combination of their outputs satisfies $\varepsilon$-differential privacy with $\varepsilon = \varepsilon_1 + ... + \varepsilon_k$

• Spend $\varepsilon/|W|$ budget for each query in workload $W$. We are guaranteed $\varepsilon$ differential privacy after answering all the queries in $W$. 
Two problems

• Is this the most error efficient strategy?
  – NO:
  – We can use parallel composition on queries that look at disjoint partitions of output
  – We can leverage dependencies between queries
  – But how?

• What happens when you run out of privacy budget?
  – Can we use previously query answers to answer new queries without looking at the data?
Cohort Size Estimation Problem

Population of medical patients

Are there at least 200 individuals who are male cancer survivors, between 20-30, who were admitted for surgery

Are there at least 200 male cancer survivors who are between ages of 20 and 30

Are there at least 200 individuals who are male cancer survivors and admitted for surgery
Cohort Size Estimation Problem

• A sequence of queries \{Q_1, Q_2, Q_3, \ldots, Q_n\}

• Each query \(Q_i\) : Number of tuples satisfying property \(p_i > \tau\) ?
  – If answer is no, then return NO and continue.
  – If answer is yes, return YES and STOP after \(c\) positive answers

• Sensitivity of each \(Q_i\) = 1

• How do we answer using differential privacy?
Accuracy

• We will say that an algorithm is $(\alpha, \beta)$-accurate wrt a threshold $T$ if for a sequence of answers $a_1, a_2, \ldots$ in $\{Y, N\}$ for queries $Q_1, Q_2, \ldots$ if with probability $> 1 - \beta$:

$$\text{for all } a_i = Y, Q_i(D) \geq T - \alpha$$
$$\text{for all } a_i = N, Q_i(D) \leq T + \alpha$$

and the algorithm does not halt before outputting $c$ YES answers.
Cohort Size Estimation Problem

Laplace mechanism:

• Sensitivity of all queries is: $n$

• For each query: $q_i' = Q_i(D) + \text{Lap}(n/\varepsilon)$

• Return **YES** if $q_i' > \tau$
  Return NO if $q_i' < \tau$
Accuracy of Laplace Mechanism

\[
P(\mid q_i' - Q_i(D)\mid > \alpha) < \beta
\]

\[
\Rightarrow \int_{\alpha}^{\infty} e^{-\frac{\varepsilon}{n}x} \, dx < \beta
\]

\[
\Rightarrow \frac{n}{\varepsilon} e^{-\frac{\varepsilon}{n}\alpha} < \beta
\]

\[
\Rightarrow \alpha > \frac{n}{\varepsilon} \log\left(\frac{n}{\varepsilon \beta}\right)
\]

Accuracy depends on number of queries
Sparse Vector Technique (c=1)

• Set $\tau' = \tau + \text{Lap}(2/\varepsilon)$

• For each query: $q_i' = Q_i(D) + \text{Lap}(4/\varepsilon)$

• If $q_i' < \tau'$
  
  Return NO and continue

  Else // $q_i' \geq \tau'$,

  Return YES and STOP
Accuracy Analysis

- Given a sequence of queries Q1, ..., Qk and threshold T such that for all j < n, Qj(D) < T – α
  the sparse vector technique is (α, β)-accurate for

\[ \alpha = \frac{8(\log k + \log(2/\beta))}{\epsilon} \]

That is, when all but the last query are “far away” from the threshold, the algorithm will say NO for all but the last query, and say YES and halt on the last query.

Accuracy depends on log of number of queries!
Proof

• All we need to show is with probability 1 - β:
  for all j, \(|\text{noise}_j| + |\tau - \tau'| < \alpha\)

• This implies: for \(ak = \text{YES}\).
  \(Q_k(D) > \tau' - \text{noise}_k \geq \tau' - |\text{noise}_k|\)
  \(\geq \tau - |\tau - \tau'| - |\text{noise}_k| > \tau - \alpha\)

• For all \(aj = \text{NO}\),
  \(Q_j(D) < \tau' - \text{noise}_j \leq \tau' + |\text{noise}_j|\)
  \(\leq \tau + |\tau - \tau'| + |\text{noise}_j| < \tau + \alpha\)
Proof

• All we need to show is with probability 1 - β:
  for all j, \(|\text{noise}_j| + |\tau - \tau'|| < \alpha\)

• \(\Pr[|\tau - \tau'| > \alpha/2 ] < \exp(-\varepsilon \alpha/4)\)

• \(\Pr[|\text{noise}_j| > \alpha/2] < \exp(-\varepsilon \alpha/8)\)
  Therefore, \(\Pr[\max_j |\text{noise}_j| > \alpha/2] < k \exp(-\varepsilon \alpha/8)\)

• Setting \(\varepsilon \alpha > 8(\log k + \log(2/\beta))\) ensures each of the above probabilities is bounded by \(\beta/2\).
Sparse Vector Technique (c=1)

- Set $\tau' = \tau + \text{Lap}(2/\epsilon)$

- For each query: $q_i' = Q_i(D) + \text{Lap}(4/\epsilon)$

- If $q_i' < \tau'$
  
  Return NO and continue

- Else // $q_i' \geq \tau'$,
  
  Return YES and STOP

Does this satisfy $\epsilon$-differential privacy?
Pr[SVT(D) = a1, ..., ak]

= \int_v \int_t Pr[\tau' = t] Pr[q_k(D) + v_k > t] \Pr[\max_j q_j(D) + v_j < t] \, dv \, dt

Probability of noisy threshold taking value t

Probability that last query answer crosses noisy threshold

Probability previous query answers do not cross noisy threshold
Privacy analysis

\[ \Pr[SVT(D) = a_1, \ldots, a_k] \]
\[ = \int_{\nu} \int_{t} \Pr[\tau' = t] \Pr[q_k(D) + \nu_K > t] \Pr[\max_{j<k} q_j(D) + \nu_j < t] \, dvdt \]

- Let \( N \) denote the set of noise values for queries 1 \( \ldots \) k-1, that make the noisy query answers smaller than t. That is,

\[ N = \{(\nu_1, \nu_2, \ldots, \nu_{k-1}) | \max_{j<k} q_j(D) + \nu_j < t \} \]
Privacy analysis

\[ \Pr[SVT(D) = a_1, \ldots, a_k] = \int_{\nu} \int_{t} \Pr[\tau' = t] \Pr[q_k(D) + \nu_k > t] \Pr[\max_{j<k} q_j(D) + \nu_j < t] \, dvdt \]

\[ N = \{(\nu_1, \nu_2, \ldots, \nu_{k-1}) | \max_{j<k} q_j(D) + \nu_j < t\} \]

- Then,

\[ \forall (\nu_1, \nu_2, \ldots, \nu_{k-1}) \in N, \max_{j<k} q_j(D') + \nu_j < t + 1 \]

\[ \Rightarrow \Pr[\max_{j<k} q_j(D) + \nu_j < t] < \Pr[\max_{j<k} q_j(D') + \nu_j < t + 1] \]
Privacy analysis

\[ \text{Pr}[\text{SVT}(D) = a_1, \ldots, a_k] \]

\[ = \int_v \int_t \text{Pr}[\tau' = t] \text{Pr}[q_k(D) + \nu_K > t] \text{Pr} \left[ \max_{j<k} q_j(D') + \nu_j < t \right] dvdt \]

\[ < \int_v \int_t \text{Pr}[\tau' = t] \text{Pr}[q_k(D) + \nu_K > t] \text{Pr} \left[ \max_{j<k} q_j(D') + \nu_j < t + 1 \right] dvdt \]

\[ < \int_v \int_t e^\varepsilon \text{Pr}[\tau' = t + 1] \text{Pr}[q_k(D) + \nu_K > t + 1] \text{Pr} \left[ \max_{j<k} q_j(D') + \nu_j < t + 1 \right] dvdt \]

\[ < \int_v \int_t e^\varepsilon \text{Pr}[\tau' = t + 1] e^\varepsilon \text{Pr}[q_k(D') + \nu_K > t + 1] \text{Pr} \left[ \max_{j<k} q_j(D') + \nu_j < t + 1 \right] dvdt \]

\[ < \int_v \int_t e^\varepsilon \text{Pr}[\tau' = t] \text{Pr}[q_k(D') + \nu_K > t] \text{Pr} \left[ \max_{j<k} q_j(D') + \nu_j < t \right] dvdt \]

\[ < \text{Pr}[\text{SVT}(D') = a_1, \ldots, a_k] \]
Sparse Vector Technique

• Set $\tau' = \tau + \text{Lap}(2/\varepsilon)$

• For each query: $q_i' = Q_i(D) + \text{Lap}(4/\varepsilon)$

• If $q_i' < \tau'$
  Return NO and continue

Else // $q_i' \geq \tau'$,
  Return YES and STOP

Why add noise?

Why Stop?
Sparse Vector Technique
(without adding noise to queries and without stopping)

• Set $\tau' = \tau + \text{Lap}(2/\varepsilon)$

• For each query: $q_i = Q_i(D)$

• If $q_i < \tau'$
    Return NO and continue
Else // $q_i \geq \tau'$,
    Return YES and continue
Sparse Vector Technique
(without adding noise to queries and without stopping)

... does not satisfy privacy.

D: Q1(D) = 0, Q2(D) = 1
D': Q1(D') = 1, Q2(D') = 0  ... both queries with sensitivity 1

Let $\tau = 0$

$\Pr[SVT(D) = \text{NO}, \text{YES}] > 0$ (whenever $0 < \tau' \leq 1$)

$\Pr[SVT(D') = \text{NO}, \text{YES}] = 0$ (no value of $\tau'$)
Sparse Vector Technique

Does not satisfy privacy if:

• No stopping and no noise is added to queries
• No stopping and noise is added to queries
• Stop after 1 query but release the noisy count used to compute the YES answer

Takeaway: Need to be very careful with privacy analysis
Numeric Sparse Vector Technique

- Set $\tau' = \tau + \text{Lap}(2c/\varepsilon_1)$
- Set count = 0
- For each query: $q_i' = Q_i(D) + \text{Lap}(4c/\varepsilon_1)$

- If $q_i' < \tau'$,
  Return NO and continue

- Else // $q_i' \geq \tau'$,
  Return $Q_i(D) + \text{Lap}(4c/\varepsilon_2)$
  If count < c:
    $\tau' = \tau + \text{Lap}(2c/\varepsilon_1)$
    count ++
  Else STOP

Allows c answers

Need new randomness to release noisy count

Need new noisy threshold after each YES answer

Lecture 11: 590.03 Fall 16
SVT for online QA

• Maintain a “synthetic” distribution A (say uniform) on the domain

• For each query Qi:
  – Ask two queries Qi(D) – Qi(A) and using Numeric SVT
  – Use a positive threshold (depends on c, total number of queries, domain size ...)

• If both queries result in NO answers, Release Qi(A)

• If Qi(D) – Qi(A) is a number, Release Qi(A) + answer from SVT
  Else (Qi(A) – Qi(D) is a number), Release Qi(A) - answer from SVT

• Update A using multiplicative weights
Private Multiplicative Weights

• Maintains a current estimate of the database.

• After c answers, Numeric SVT stops releasing YES answers. Thereafter, we can use current estimate of database to keep answering questions.

• Privacy analysis follows from privacy of Numeric SVT

• Utility analysis is a little more complex (see textbook).