Analysis of the Privacy of Differential Privacy in Probabilistic Databases

Truong Pham

May 2022

1 Introduction

1.1 Differential Privacy

Differential Privacy promises to protect the identity of every participant by introducing randomness into each query. This concept relies on non-deterministic algorithms to give meaningful responses while minimizing the exposure of the original data. Given two databases that only differ in one record, called neighboring databases, the result from any query can be from either database. Formally, we give the definition 1 of Differential Privacy:

Definition 1 *(Differential Privacy)*: A randomized algorithm M with domain $N^{|X|}$ is (ϵ, δ) differentially private if for all S in range(M) and for all neighboring databases x, y in $N^{|X|}$:

$$Pr[M(x) \in S] <= e^{\epsilon} Pr[M(y) \in S] + \delta$$

Differential Privacy is measured in (ϵ, δ) where ϵ is called the privacy budget and δ is the probability of failure. This inequality is derived from the privacy loss which is the distribution in the KL-divergence:

$$L_{M(X)||M(Y)} = ln\frac{Pr[M(x) = w]}{Pr[M(y) = w]}$$

Therefore, the distribution of the query of two neighboring databases has a maximum distance of epsilon with confidence 1-δ:

$$P[L_{M(X)||M(Y)} \leq \epsilon] >= 1 - \delta$$

1.2 Probabilistic Databases

A probabilistic database is similar to a normal database, but the attributes can be distributions instead of constant values. Every possible state of the record is associated with a probability value and each possible state of the database has a probability equal to the product of the probability of its records. To integrate Differential Privacy, we must clearly define the scheme that we use for these databases. In the simplest case, each record is independent of each other and has only one uncertain attribute. This type of database is called the Tuple-Independent Database.

1.3 Differential Privacy in Probabilistic Databases

Our goal is to analyze the differential privacy when combining randomized mechanism with the randomness in the probabilistic database. The probabilistic database must already have a built-in privacy value since there is already randomness in the data. Unlike the original definition of Differential Privacy, The data base is now a distribution. Therefore, the domain of M will be $X_i^{|X|}$ where X_i is all possible distributions. $M(X)$ will most likely be intractable, which poses a difficulty on how we analyze the privacy budget. We must note that there are three different privacy values: the natural privacy $\epsilon_{natural}$ of the probabilistic database, the privacy budget $\epsilon_{mechanism}$ of the randomized mechanism which must be set, and the true privacy ϵ which is the real privacy of the randomized mechanism after being applied to the probabilistic database.

2 Gaussian distributed records

We assume that each record has Gaussian distribution $N(\mu_i, \sigma_i)$ for i in range 1,2,3,... n where n is the number of records. Using the privacy loss, we can find the built-in privacy of the probabilistic database with Gaussian distributed records. An important parameter to calculate epsilon is the sensitivity function:

$$\Delta_n = max_{d,d'} \|q(d) - q(d')\|_n$$

The domain for this function is all possible neighboring databases D where d and d' in D. Using the Privacy Loss definition, a database with Gaussian distributed records has natural epsilon $= \epsilon_{natural} = \frac{\Delta_2}{\sigma}\sqrt{2\log(\frac{1}{\delta})} + \frac{\Delta_2^3}{\sigma^3}\frac{1}{2}$. Proof in Appendix I.

The convolution of two distributions becomes amazingly simple when both distributions are Gaussians since the resulting distribution is also Gaussian. Furthermore, the Gaussian Mechanism is a very well-studied random mechanism of differential privacy. The Gaussian mechanism is defined as follow:
Definition 3 *(Gaussian Mechanism)*: Given any function f: $N^{|X|} \rightarrow R^k$, the Gaussian Mechanism is defined as:

$$M_G(x, f, \epsilon, \delta) = f(x) + N(\mu = 0, \sigma^2 = \frac{2log(1\ \delta)\Delta^2)}{\epsilon^2}$$

If f has Gaussian distribution, M_G will also have a Gaussian distribution. The variance of M_G will be the sum of the variance of query and the Gaussian Mechanism. Therefore, the overall epsilon value will always be larger than the epsilon value of the mechanism and the probabilistic database.

3 General distributed records

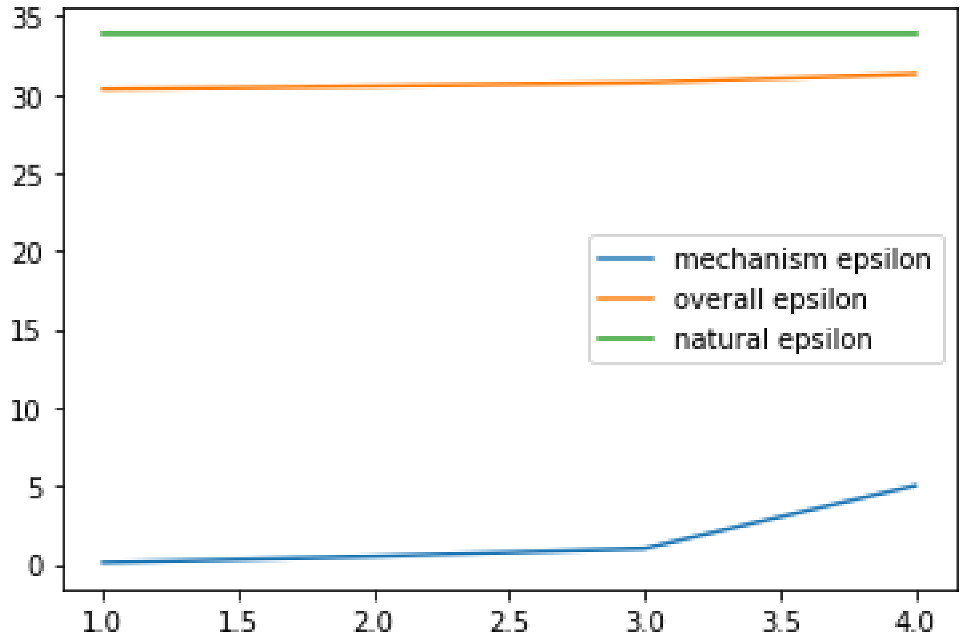
In the general case of the probabilistic database, the randomized mechanism will be intractable. To calculate the final privacy value, we propose an estimator that takes in any database that has discrete distributions as attributes and performs a differential privacy algorithm on top of that. Since the purpose of Differential Privacy is to mask the true queried values, we further specified that the SUM query will be used for the analysis. The SUM query is a scalar linear function that is also convenient to perform distribution algebra on. The SUM query will first sample a value from attributes and then calculate the final sum. The estimator will sample the outputs to estimate the distribution of the privacy loss. By definition, epsilon will be the highest value the distribution can take. We will first calculate the natural epsilon $\epsilon_{natural}$, which is the privacy inherent in every probabilistic database. Then we will define the privacy budget $\epsilon_{mechanism}$ of the randomized mechanism. Finally, we can calculate the overall ϵ. The process of sampling will not give the true overall epsilon but an estimation of it. We can know how close we got to the true overall epsilon by using the DKW-M inequality for all $\delta > \sqrt{log2/2m}$:

$$\begin{aligned}
&Pr(\phi(\epsilon^*_{max}) - \phi(\hat{\epsilon}_{max})) > \sigma) \\
=&Pr(\phi_m(\hat{\epsilon}_{max}) - \phi(\hat{\epsilon}_{max}) > \sigma) \\
\leq&Pr(sup(\hat{\epsilon}_{max}) - \phi(\hat{\epsilon}_{max}) > \sigma) \leq exp(-2m\sigma) = \gamma
\end{aligned}$$

With this we have a privacy of $(\epsilon, \delta, \gamma)$. ϕ_m and ϕ are the CDF for the sampled and real distribution of the privacy loss, respectively. $\hat{\epsilon}_{max}$ is the maximum sampled overall privacy and ϵ^*_{max} is the real maximum overall privacy. M is the number of samples which controls the accuracy of the sampled epsilon. The larger the number of samples the smaller the range that the sampled epsilon value can deviate from the real value.

Algorithm 1 Epsilon Estimator

```
Input: X_list is list of pairwise neighboring databases, query_N is the number of samples to estimate the queried distributions, privacy_N is the number of samples to estimate the privacy value
queried_dists = []. a list of queried distributions from each database
for X in X_list do
    queried = []. list of queried values
    for I in range(query_N) do
        s = SUM(X)
        if calculate_natural_privacy == True then
            s = s + random_mechanism()
        end if
        queried.append(s)
    end for
    queried_dists.append(get_distribution(queried))
end for
max_epsilon = 0
for qdist_1 in queried_dists do
    for qdist_2 in queried_dists do
        if qdist_1 != qdist_2 then
            epsilon = get_privacy(qdist_1,qdist_2, privac_N)
            max_epsilon = max(epsilon, max_epsilon)
        end if
    end for
end for
return max_epsilon
```



The randomized mechanism improves the privacy of the database but the randomness in the database does not improve the privacy of the randomized mechanism. This is not what we saw in the former case where the overall privacy is guaranteed to be better than both the privacy mechanism and the database. This is because the final distribution in the general case is a random intractable distribution and it is very unlikely to be more secured than the distribution in the randomized mechanism. The randomized mechanism still improves privacy but the efficiency by the mechanism is mitigated by database.

4 Conclusions

The addition in the randomness in the database requires a different method to analyze the Differential Privacy algorithm. A general way to find the true ϵ value is sample the privacy loss with large number of samples. This is because most convolutions are intractable and impossible to fully analyze. However, in the special case when the database has all Gaussian distributed records, we can apply the Gaussian Mechanism on the query and fully perform analysis on the resulted distribution.

In the general case, the overall epsilon is not guaranteed to be smaller than the epsilon of the randomized mechanism. However, privacy will improve when applying Gaussian Mechanism onto the query of the Gaussian distributed records. Therefore, the efficiency of the randomized mechanisms depends on the distribution of the records. The randomized mechanism must be made to fit the distribution of the records in a way that reduces the distention of the distributions of those records.

5 Appendix I

Assuming $q(d) = SUM(d) = \sum_1^n d_i$ $q(d) = z \sim N(\sum_i x_i, n) = N(\mu, n) = \frac{1}{\sqrt{2\pi n}}\exp(\frac{-(z-\mu)^2}{2n})$ since the sum of Gaussian is Gaussian

$$\begin{aligned}
L_{q(d)\|q(d')} &= log(\frac{P_{q(d)}(z)}{P_{q(d')}(z)}) = \frac{-1}{2\sigma^2}[(z-\mu)^2 - (z-\mu')^2] = \frac{-1}{2\sigma^2}(z^2 - 2z\mu + \mu^2 - z^2 + 2z\mu' - \mu'^2) \\
&= \frac{1}{2\sigma^2}(2z(\mu - \mu') + \mu'^2 - \mu^2) \\
&\sim N\left(\frac{\mu(\mu-\mu')}{\sigma^2} + \frac{\mu'^2 - \mu^2}{2\sigma^2}, \frac{4(\mu-\mu')^2\sigma^2}{4\sigma^4}\right) \\
&= N\left(\frac{(\mu-\mu')^2}{2\sigma^2}, \frac{(\mu-\mu')^2}{\sigma^2}\right)
\end{aligned}$$

With $Z \sim N(0,1)$, we have:

$$\begin{aligned}
P[|L| \geq \epsilon] &= P\left[|Z| \geq \frac{\epsilon\sigma}{\mu - \mu'} - \frac{(\mu-\mu')^2}{2\sigma^2}\right] \\
&\leq P\left[|Z| \geq \frac{\epsilon\sigma}{\Delta_2} - \frac{\Delta_2^2}{2\sigma^2}\right] \leq \delta
\end{aligned}$$

Since $P[|Z| \geq v] \leq exp(\frac{-v^2}{2})$, we have:
$\delta = exp\left(\frac{-v^2}{2}\right)$ with $v = \frac{\epsilon\sigma}{\Delta_2} - \frac{(\Delta_2)^2}{2\sigma^2}$
We set $\frac{\Delta}{\sigma} = a \Rightarrow v = \frac{\epsilon}{a} - \frac{a^2}{2} = \frac{2\epsilon - a^3}{2a}$, therefore:

$$\begin{aligned}
\delta &= exp(-\frac{(2\epsilon - a^3)^2}{8a^2}) \\
\Rightarrow \log(\frac{1}{\delta}) &= \frac{(2\epsilon - a^3)^2}{8a^2} \\
\Rightarrow a\sqrt{2\log(\frac{1}{\delta})} + \frac{a^3}{2} &= \epsilon
\end{aligned}$$

δ and ϵ has an inverse relationship which is expected. δ should be set to a small number and we have $\epsilon \geq \frac{a^3}{2}$. ϵ can also be bounded above by some reasonable value for δ.