Differential Privacy
A Survery

20398702 HU, JIAJUN
ZHOU, LEI
Department of Computer Science & Engineering
The Hong Kong University of Science and Technology
May 3, 2017

Abstract

Keywords: differential privacy

1 Introduction

With the fast development of web techologies, the number of interent users are growing exponentially. In order to provide better services, many service providers, such as government, company, and research institutes, collect personal information and analyze them. Social networks such as Facebook and LinkedIn use friendship to recommend new friends to you. Youtube & Amazon use viewing/buying records also for recommendations. Emails in Gmail are used for targeted Ads. We are enjoying the big convenience and effecience brought by these technologies, however, we are still under the risk of personal information leakage. Trouble will be made if attackers exploit the leaked information and make use of them.

To address this issue, a common practice is to anonymize certain data in the database. On could remove or hash the identifiable information, such as name and identity number, in the database. HIPAA [1] (Health Insurance Portability and Accountability Act) has promoted this approach and list 18 categories of identifiable information which can be anonymized. However, anonymizing data is insufficient especially when the attacker has already

[1] https://en.wikipedia.org/wiki/Health_Insurance_Portability _and_Accountability_Act

known some background knowledge about the individuals in the database. To combat this issue, some advanced technologies, such as k-anonymity[1], l-diversity[2] and t-closeness[3], have been proposed. But it turns out even these techologies cannot prevent background attacks especially when the attacker already know something about the contents in the database[4].

In this survey, we present differential privacy[5, 6] . Differential privacy is a powerful approach to privatize the released information from database. Even in the WWDC 2016[2], Apple announced a series of new security and privacy features, including one feature that is differential privacy, with the aim to improve the privacy of their data collection practices[3]. Roughly speaking, differential privacy ensures that the removal or insertion of a single record does not significantly affect the outcome of any analysis made on the database, thus making it possible to prevent attackers from gussing the real contents of the database. It follows a rigorous mathematical deduction to prove it can reduce the risk of privacy breach while remaining the utility of the data.

In the rest of the survey, we provide the definition of differential privacy and two mechanisms, Laplace Mechanism and Exponential Mechanism, that can achieve it (Section 2). Then, we discuss a special case of differential privacy, local differential privacy (LDP), and list four typical LDP solutions (Section 3). Finally, we conclude the survey in the last section.

2 Differentail Privacy

Over the past ten years, differentail privacy[5, 6] has emerged to become one of the most powerful approaches to ensure data pricacy. Roughly speaking, differential privacy ensures that the removal or insertion of a single record does not significantly affect the outcome of any analysis conducted on the database, thus making it possible to prevent private information from exposing to attackers. It follows a rigorous mathematical deduction to prove it can reduce the risk of privacy breach while remaining the utility of the data. At the beginning of this section, we will illuminate the concept by leveraging a simple example. Then, we will give the mathematical definition of differential privay and introduce two privacy mechanisms to achieve it.

2.1 A Simple Example

Suppose you have access to a database that allows you to compute the total income of all resident in certain area. You know one of your friends,

[2] https://developer.apple.com/videos/wwdc2016

[3] https://www.wired.com/2016/06/apples-differential-privacy-collecting-data/

Name	Annual Income
Mr. Richard	0.5 million
Mr. White	1 million
Mr. Brown	2 million
Ms. Lee	0.35 million
Ms. Jean	0.6 million
...	...
Total income = 50 million	

Name	Annual Income
Mr. Richard	0.5 million
Mr. Brown	2 million
Ms. Lee	0.35 million
Ms. Jean	0.6 million
...	...
Total income = 49 million	

Table 1: The table before and after Mr. White's move.

Mr. White is going to move to another area, so simply computing the total income of all resident before and after Mr. White's move would allow you to guess his real income. As shown in table 1, the total income of all residents before Mr.White's move is 50 million, while the total income of all residents after Mr.White's move is 49 million. One can compute the real income of Mr.White is 1 millon. So from this example we can see even though we are not allowed to retrieve the information of a particular person, we are still able to get the private informtion through certain opertions. So what could one do to stop this? In the next section, we wil see how differential privacy can help resolve this problem.

2.2 Definition of Differential Privacy

Firstly, let us define some notations.

Definition 2.1. D and D' are databases, but they must differs on at most one row.

The reason why D and D' is required to differ on one row is to simulate whether a particular record is in or not in the database.

Definition 2.2. $f(D)$ is a query on D

Refer to the previous example, $f(D)$ is the total income of all residents in the database.

Definition 2.3. $M(D)$ is the privacy mechanism, which is a randomized function that takes the database D as inpiut, and release privatized information with respect to $f(D)$.

Desiging privacy mechanism is a topic on its own[7]. In this survery, we only consider Laplace Mechanism and Exponential Mechanism. Refer to the previous example, $M(D)$ is the privated total income obtained by adding random noise on the total income.

Definition 2.4. ϵ - differential privacy A privacy mechanism M gives ϵ - differential privacy if for all data sets D and D' differing on at most one row, and all $C \in Range(M)$,

$$\frac{Pr[M(D) = C]}{Pr[M(D') = C]} < e^{\epsilon}$$

$\epsilon - differential\ privacy$ is a special case of $(\epsilon, \delta) - differential\ privacy$[8, 9] with $\delta = 0$. Typically, $(\epsilon, \delta) - differential\ privacy$ is simplified to $\epsilon - differential\ privacy$, so we only consider $\epsilon - differential\ privacy$ in this survey. $\epsilon - differential\ privacy$ says that the probability that the privatized result will be C is nearly the same whether or not you are in the database, which means an adversary cannot infer with high confidence (controlled by ϵ) whether the input database is D or D'. In the definition, ϵ is the privacy budget, which is a tradeoff that is used to balance the privacy of the result and it's utility. The smaller the ϵ is, the closer $Pr[M(D) = C]$ and $Pr[M(D') = C]$ are, and the stronger protection is.

2.3 Laplace Mechanism

In this section, we will introduction one of the most popular privacy mechanisms - Laplace Mechanism[10]. Laplace mechanism works particularly well when the query $f(D) \in R^n$ is a funtion mapping databases to (vectors of) real numbers. For example, when the query is a counting query, $f(D)$ is the number of records in the database.

Definition 2.5. Sensitivity of a Function For $f : D \to R^n$, the sensitivity of f is

$$\triangle f = max_{D,D\prime} ||f(D) - f(D\prime)||_n$$

for all $D, D\prime$ differs on at most one row.

Intutively, $\triangle f$ captures how much one person's data can affect the ouput. Taking the counting query as an example, $\triangle f = 1$ because adding or deleting a row of the database will only affect the number of records by 1.

For simplicity, we only consider the case when $n = 1$. Laplace Mechanism $M(D)$ privatizes the result by adding a 0-centered symmetric random noise, which is drawn from Laplace Distribution[11] with parameter $\triangle f / \epsilon$, on the true answer $f(D)$. So the probability density function of the random noise x is

$$Pr[x] = \frac{\epsilon}{2 \triangle f} e^{-\frac{|x|\epsilon}{\triangle f}}$$

The probability density function of $M(D)$ is

$$Pr[M(D)] = Pr[x + f(D)] = \frac{\epsilon}{2 \triangle f} e^{-\frac{|x - f(D)|\epsilon}{\triangle f}}$$

The mathematical proof that laplace mechnism yields a $\epsilon - differential\ privacy$ is straightforward.

$$\frac{Pr[M(D) = C]}{Pr[M(D\prime = C)]} = \frac{\frac{\epsilon}{2\triangle f} e^{-\frac{|x-f(D)|\epsilon}{\triangle f}}}{\frac{\epsilon}{2\triangle f} e^{-\frac{|x-f(D\prime)|\epsilon}{\triangle f}}} = \frac{e^{-\frac{|x-f(D)|\epsilon}{\triangle f}}}{e^{-\frac{|x-f(D\prime)|\epsilon}{\triangle f}}}$$

$$= e^{-\frac{|x-f(D)|\epsilon}{\triangle f} + \frac{|x-f(D\prime)|\epsilon}{\triangle f}} = e^{\frac{|f(D)-f(D\prime)|\epsilon}{\triangle f}} \leq e^{\epsilon}$$

So it concludes that laplace mechanism yields $\epsilon - differential\ privacy$

2.4 Exponential Mechanism

3 Local Differential Privacy

In traditional differential privacy[5], all sensitive information from a large number of respondents are gathered by a trusted and trustworthy curator, who further releases the statistical information of the underlying population to the public. The reponsibiliy to privatize information lies on the curator side. However, in Local Differential Privacy (LDP)[12, 13], every respondent take the responsibility to privatize his or her data locally before sending to the curator. So in this setting, the curator will never have the access to the exact value of sensitive information, which protects not only the privacy of data contributors but also the curator itself against the potential risk of information leakage. The goal of LDP is two fold: (1) $\epsilon - differential\ privacy$ must be satisfied on each user side. (2) the curator should be able to compute accurate statistics from the noisy data received from the user side. It turns out that traditional differential privacy mechanisms are not adequate enough to address the LDP problems. In the following, we overview several typical LDP solutions.

3.1 Randomized Response

Rendomized Response (RR)[14] asks each user with a sensitive question whose answer must be "yes" or "no". For example, "do you like Donald Trump?". The user flips a coin to decide which answer to give. The user gives his true answer when the coin turns head, and gives the opposite answer when the coin turns tail. However, in RR, instead of using a fair coin, we use a biased coin. It turns head with probability p, and turns tail with probability $1 - p$. It turns out that $\epsilon - differential\ privacy$ can be satisfied with the following value of p:

$$p = \frac{e^{\epsilon}}{1 + e^{\epsilon}}$$

The goal of the curator is to give the estimated percentage of "yes" given the noisy answers. Suppose the percentage of "yes" given the noisy answer is c, the corrected version of the result c' is:

$$c' = c \times c_e, where\ c_e = \frac{1}{1-2p}$$

The limitation of RR is that it can only be applied to the problem with binary answer.

3.2 RAPPOR

RAPPOR[14, 15] extends RR[14] to more complex data types. Suppose there are n users and d items, each user can own exactly one item. To be more specific, let us define u_i, where $i = 1\ to\ n$, as the i_{th} user. v_i, where $i = 1\ to\ n$, is a vector with length d of u_i. All the coordinates of v_i are 0 except the j_{th} coordinate, which is 1, if u_i owns item j. The goal of the corator is the same as in RR, which is to compute the frequency of each item. User u_i applies RR independently on each coordinate of v_i with a biased coin with probability p (the sensitivity of any query function f is 2 because any vector contains a single coordinate of 1, hence the maximum difference between $f(D)$ and $f(D')$ is 2):

$$p = \frac{e^{\frac{\epsilon}{2}}}{1 + e^{\frac{\epsilon}{2}}}$$

The limitation of RAPPOR is that it can cause huge communication overhead because each user has to send a vector with lengh d, which can be extremly large in some cases (e.g. the number of email accounts of all users).

3.3 Succinct histogram

Succinct histogram (SH)[16] addresses the communication overhead led by RAPPOR[14, 15]. Basically, the problem setting of SH remains the same with RAPPOR, i.e. each user u_i owns a vector v_i of length d, with only one coordinate to be 1 and all the others to be 0. The goal of the curator is also to estimate the frequency of each item. The main idea behind SH is instead of sending d coordinates every time, every user only randomly pick one coordinate and report it to the curator. In this way, the communication overhead drops from $O(d)$ to $O(1)$.

3.4 LDPMiner

The above mentioned solutions pose too much limitations on the data types. In order to cope with complex data types, Zhan et al. proposed

LDPMiner[17] to address the heavy hitters over set-valued data. Suppose there are n users and d items, and each user can own exactly l items. If the number of the items a user owns is smaller than l, some dummy items are added to fill the set. If the number of the items a user owns is greater than l, l randomly picked items are considered in the set. The frequency of the item is the portion of users owning this item. The general goal of the curator is to find the top-k most frequent items with the highest frequency. LDPMiner proposes a two-phase solution. The main idea is to firstly filter the items and select $O(k)$ candidate heavy hitters in the first phase, and then focuses on refining the frequecy estimates of these candidates in the second phase. LDPMiner splits the total privay budget ϵ as ϵ_1 and ϵ_2, which is assigned to phase one and phase two respectively. According to sequential composability[18], the whole process satisfies $\epsilon - differential\ privacy$.

Conclusion

References

1. Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.
2. Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.
3. Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115. IEEE, 2007.
4. Zhanglong Ji, Zachary C Lipton, and Charles Elkan. Differential privacy and machine learning: a survey and review. *arXiv preprint arXiv:1412.7584*, 2014.
5. Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.
6. Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

7. Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 94–103. IEEE, 2007.

8. Cynthia Dwork. Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer, 2011.

9. Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer, 2006.

10. Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

11. Laplace Pierre Simon et al. Mémoire sur la probabilité des causes par les évènements. *Mémoires présentés par divers savants [à lâĂŹAcadémie royale des sciences], Paris, Imprimerie royale*, 6:621–656, 1774.

12. Anupam Gupta, Moritz Hardt, Aaron Roth, and Jonathan Ullman. Privately releasing conjunctions and the statistical query barrier. *SIAM Journal on Computing*, 42(4):1494–1520, 2013.

13. Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

14. Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067. ACM, 2014.

15. Giulia Fanti, Vasyl Pihur, and Úlfar Erlingsson. Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries. *Proceedings on Privacy Enhancing Technologies*, 2016(3):41–61, 2016.

16. Raef Bassily and Adam Smith. Local, private, efficient protocols for succinct histograms. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 127–135. ACM, 2015.

17. Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. Heavy hitter estimation over set-valued data with local differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 192–203. ACM, 2016.

18. Frank D McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM*

SIGMOD International Conference on Management of data, pages 19–30. ACM, 2009.