differential privacy and machine learning review

xiang li

huazhong university of science and technology, wuhan, china

***Abstract*—Privacy preserving in data release and mining is a hot topic in the information security field currently. As a new privacy notiondifferential privacy(DP) has grown in popularity recently due to its rigid and provable privacy guarantee.After analyzing the advantage of differential privacy model relative to the traditional onesthis paper surveys the theory of differential privacy and its application on machine learning.We explore the interplay between machine learning and differential privacy, namely privacy-preserving machine learning algorithms and learning-based data release mechanisms.**

***Index Terms*—differential privacy, privacy preservation, machine learning.**

I. Introduction

With the continuous popularization and deepening of information technology applications, various information systems have stored and accumulated a wealth of data, such as patient diagnostic data sets established by medical institutions, customer online transaction data collections collected by e-commerce companies, and so on. Analysis can enable people to gain more knowledge about the real world. Therefore, for research institutions, information consulting organizations, and government decision-making departments, data is a very important basic resource. This demand has greatly promoted the release and sharing of data.

However, the data set usually contains many personal privacy information, such as medical diagnosis results, personal consumption habits, and other data that can reflect personal characteristics, and these information will be leaked along with the release and sharing of data sets. Although deleting the identifier attributes (such as name, ID number, etc.) of the data set can protect personal privacy to some extent, but some attack cases show that this simple operation is far from enough to guarantee the security of private information.

From the perspective of existing research, k-anonymity [1] and its extended models have been widely studied in the field of privacy protection. The basic idea of these models is that the attributes related to the attacker background knowledge are defined as quasi identifiers in the data set. By generalization and compression processing, all the records are divided into several equivalence group, and the records in each of the equivalents have the same quasi identifier, so that a record can be hidden in a set of records. Therefore, such a model is also known as a packet based privacy protection model.

However, follow-up studies have shown that these models have two main defects, which do not provide enough security, and they always need to be perfected because of the emergence of new attacks. For example, to resist "Conformance" attacks, l-diversity [2], t-closenes s [3] and other models have been proposed successively; in these methods, if their sensitive features are the same, the samples are grouped, and if the number of samples in the group is large enough, a group is published. Intuitively, it is very difficult for an attacker to distinguish a single sample. However, these methods can not prevent background attacks. In extreme cases, attackers may know the contents of all rows in the collection. Consider a database that stores four people's addresses and revenues, and uses anonymous to publish private data. If three people live in the same city, the average income of the city and three people is divulged. Now suppose the attacker knows that two people in the database live in Los Angeles, and third people live in New York. If no data were released, the attacker could easily infer that the fourth person lived in Los Angeles.

Therefore, how to protect private information or sensitive information not to be leaked in the learning process has become a very meaningful research topic in the study of machine learning.

In this paper, we first introduce the concept of differential privacy (Section 1), and briefly analyze some important properties of differential privacy. The application of differential privacy in machine learning is introduced later. We mainly introduce these aspects from differential privacy supervised learning (Section 2), differential privacy unsupervised learning (Section 3) and model assessment of differential privacy machine learning (Section 4). Finally, we summarize the current situation and development prospects of differential privacy and differential privacy machine learning, and summarize the existing results.

II. Differential Privacy

Differential privacy is a widely used definition of privacy. Intuitively, it is necessary to export information about the underlying dataset, which is robust to any change of a sample, thus protecting privacy. The following sections define differential privacy in mathematics, and introduce some commonly used methods in differential privacy.

A. Definition of Differential Privacy

Definition 1: [4] Let $\delta = \delta(n)$ be a negligible function of n. A randomized algorithm A is (ε, δ)-indistinguishable if for all $x, y \in D_n$ satisfying d(x, y) = 1, and for all sets S of possible outputs

$$Pr|A(x) \in S| \leqslant e^{\varepsilon} Pr|A(y) \in S| + \delta \tag{1}$$

When δ = 0, we say the algorithm is ε-indistinguishable.

As shown in Figure 1, the algorithm provides privacy protection by randomization of the output results, while using parameter varepsilon to ensure that the probability of the output of the same result is not significantly changed when any record is deleted in the data set.

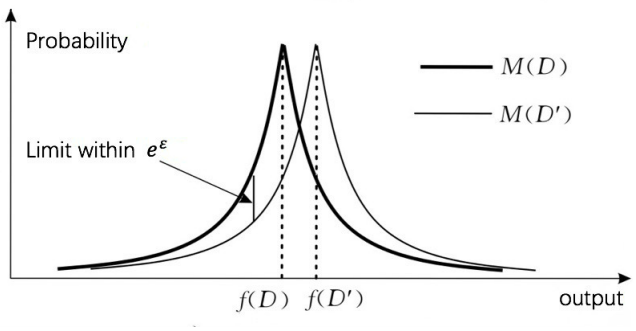

Fig. 1. The output probability of random algorithms on adjacent data sets

For example, table 1 shows a medical dataset D, each of which indicates whether a person is suffering from cancer. The dataset provides a user with a statistical query service, but does not disclose the value of a specific record. Set the user input parameter S_i and call the query function $f(i) = count(i)$ to get the number of records that satisfy the "diagnostic result" =1 before the I row in the dataset, and feedback the value of the function to the user. Suppose an attacker wants to speculate whether Alice has cancer and knows that Alice is in the fifth line of the data set, so it can be used $count(5) - count(4)$ to bring out the correct results.

TABLE I
MEDICAL DATASET

name	result
Tom	0
Diego	1
Jack	1
Henry	0
Alice	1

However, if f is a query function that provides an ε-differential privacy protection, for example $f(i) = count(i) + noise$, noise is a random distribution of noise. Assuming that the possible output of $f(5)$ comes from the set {2, 2.5, 3}, then $f(5)$ will also output almost any possible value in the set {2, 2.5, 3} at almost exactly the same probability. So the attacker cannot get the desired results through $f(5) - f(4)$. This randomization method for statistical output makes it impossible for the attacker to get the difference between the query results, thus ensuring the security of each individual in the dataset.

From definition 1, it can be seen that the privacy protection budget ε is used to control the probability ratio that the algorithm M obtains the same output on two adjacent data sets, which in fact reflects the level of privacy protection that M can provide. In practical applications, ε usually take a very small value, such as 0.01, 0.1, or $ln2$, $ln3$, etc. The smaller the ε, the higher the level of privacy protection. When ε is equal to 0, the protection level reaches the highest. At this time, for any adjacent data set, the algorithm will output exactly the same result of the two probability distributions, these results can not reflect any useful information about the data set. Therefore, the value of should be combined with the specific requirements to achieve the balance of the security and availability of the output results.

B. Sensitivity

Differential privacy protection can be achieved by adding appropriate amount of interfering noise to the return value of the query function. Adding too much noise will affect the availability of the result. Too little noise will not provide enough security. Sensitivity is the key to determining the amount of noise to be added. The parameter, which refers to the deletion of any record in the data set caused by the largest change in the query results. In the differential privacy protection method defines two kinds of sensitivity, namely Global Sensitivity and Local Sensitivity .

Definition 2 (Global Sensitivity [4]): For $f : D_n \to R^d$, the global sensitivity of f is

$$GS_f = \max_{x,y:d(x,y)=1} ||f(x) - f(y)|| \tag{2}$$

The global sensitivity of a function is determined by the function itself, and the different functions have different global sensitivity. Some functions have a small global sensitivity (such as the count function, its global sensitivity is 1), so only a small amount of noise can be added to cover up the result of a record being deleted. Differential privacy protection is realized, but for some functions, such as the average value and the median, it often has a larger global sensitivity.

When the global sensitivity is large, it is necessary to add enough noise to the output of the function to ensure privacy and result in poor data availability. Defines local sensitivity and other concepts related to its computation.

Definition 3 (Local Sensitivity [4]): For f : $D_n \to R^d$ and $x \in D^n$ the local sensitivity of f is

$$LS_f = \max_{x,y:d(x,y)=1} ||f(x) - f(y)|| \tag{3}$$

The local sensitivity is determined by the function f and the specific data in the given data set D. Due to the data distribution characteristics of the data set, the local sensitivity is usually much smaller than the global sensitivity. In addition, the relationship between local sensitivity and global sensitivity can be expressed as

$$GS_f = \max_D(LS_f(D)) \tag{4}$$

However, because the local sensitivity reflects the data distribution characteristics of the data set to some extent, if the local sensitivity is directly used to calculate the noise amount, the sensitive information in the data set will be leaked. Therefore, the smooth upper bound is used together with local sensitivity to determine the amount of noise.

Definition 4 (mooth upper bound [5]): Local sensitivity $LS_f(D)$, for any of the neighboring datasets D and D and $\beta > 0$, if function S satisfies $S(D) \geq LS_f(D)$ and $S(D) \leq e^{\beta} S(D')$, then we call S the local sensitivity -smooth upper bound of function f.

All functions that satisfy this definition can be defined as a smooth upper bound, and a smooth sensitivity can be obtained by replacing the local sensitivity into the function, and then used to calculate the size of the noise. The relationship between the smooth upper bound and the local sensitivity, as shown in Figure 2.

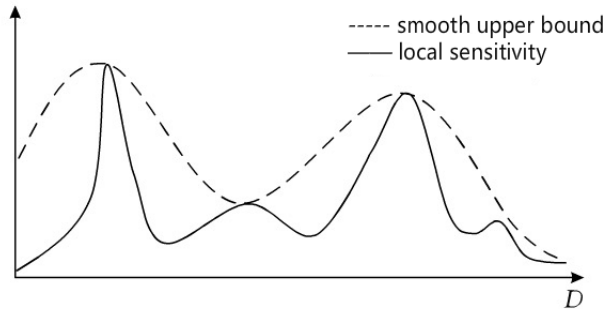

Fig. 2. Smooth upper bound

Definition 5 (Smooth sensitivity [4]): Given dataset D and D',the function $S_{f,\beta}(D) = \max_{D'}(LS_f(D')e^{-\beta|D\Delta D'|})$ is called the β-smooth sensitivity of f, in which $\beta > 0$.

C. Combination of Differentially Private Mechanisms

Theorem 1 (Parallel Theorem [6]): Suppose we have a set of privacy mechanisms $M = \{M_1, \ldots, M_m\}$, if each M_i provides ε_i-privacy guarantee on a disjointed subset of the entire dataset, M will provide $(\max\{\varepsilon_1, \ldots, \varepsilon_m\})$-differential privacy. The parallel theorem corresponds to a case where each M_i is applied on disjointed subsets of the dataset. The ultimate privacy guarantee only depends on the largest privacy budget allocated to M_i.

Theorem 2 (Sequential Theorem [6]): Suppose a set of privacy mechanisms $M = \{M_1, \ldots, M_m\}$, if each M_i are sequentially performed on a dataset, and each M_i provides ε_i privacy guarantee, M will provide $(\sum_{i=1}^{m} \varepsilon)$-differential privacy.

D. The Laplacian Mechanism

The Laplace mechanism implements ε differential privacy protection by adding random noise obeying the Laplace distribution to the exact query results. The location parameter is 0 and the Laplace of the scale parameter B is $Lap(b)$, then its probability density function is

$$p(x) = \frac{1}{2b} \exp(-\frac{|x|}{b}) \tag{5}$$

Definition 6 (laplace machanism [7]): A given set of data sets, D, has a function $f : D \to R^d$, whose sensitivity is Δf, then the random algorithm $M(D) = f(D) + Y$ provides epsilon differential privacy protection, and Y $Lap(\Delta f/\varepsilon)$ is random noise and obeys the scale parameter for the delta distribution.

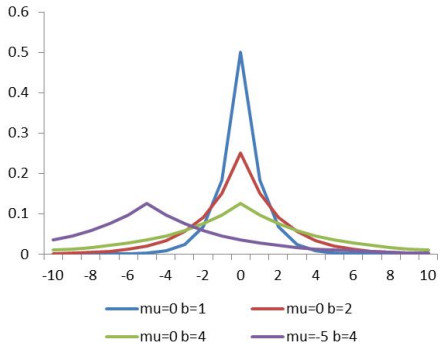

Fig. 3. Laplace machanism

From the Laplace distribution of different parameters (Fig. 1), we can see that the smaller the ε, the greater the noise introduced.

E. The Exponential Mechanism

Since the Laplace mechanism is only suitable for numerical query results, in many practical applications, query results are entity objects (such as a scheme or a choice). In this case, McSherry and others have proposed an exponential mechanism.

The output domain of a query function is Range, and each value $r \in Range$ is an entity object. Under the exponential mechanism, the function $q(D, r) \to R$ is called the availability function of the output value r, which is used to evaluate the degree of the output value r.

Definition 7 (exponential machanism [8]): The random algorithm M is input to the data set D, the output is an entity object $r \in Range$, the $q(D, r)$ is the availability function, and the Δq is the sensitivity of the function $q(D, r)$. If the algorithm M is selected and output from the probability of the $exp(\frac{\varepsilon q(D,r)}{2\Delta q})$, the algorithm provides ε-differential privacy protection.

III. Machine Learning

In machine learning, tasks are generally classified into broad categories. These categories are based on how learning is received or how feedback on the learning is given to the system developed.

Two of the most widely adopted machine learning methods are supervised learning which trains algorithms based on example input and output data that is labeled by humans, and unsupervised learning which provides the algorithm with no labeled data in order to allow it to find structure within its input data. In supervised learning, the computer is provided with example inputs that are labeled with their desired outputs. The purpose of this method is for the algorithm to be able to learn by comparing its actual output with the taught outputs to find errors, and modify the model accordingly. Supervised learning

therefore uses patterns to predict label values on additional unlabeled data.In unsupervised learning, data is unlabeled, so the learning algorithm is left to find commonalities among its input data. As unlabeled data are more abundant than labeled data, machine learning methods that facilitate unsupervised learning are particularly valuable.

We will discuss the differential privacy protection methods in supervised learning and unsupervised learning respectively.

A. Differentially Private Supervised Learning

1) Naive Bayes Model: In [9],they present PrivInfer, an expressive framework for writing and verifying differentially private Bayesian machine learning algorithms. Their framework allows to write data analysis as functional programs for Bayesian inference and to add noise to them in different ways using different metrics.They explore three ways to ensure differential privacy: by adding noise on the input, by adding noise on the output parameters based on p-norms, and by adding noise on the output parameters based on f-divergences.

In [10],they propose a novel privacy- preserving Naive Bayes learning scheme to handle the situation that the data owner not entirely trust each other.They point out that training sample always collected from multiple owner instead of a single data source.In this assumption,owners are not trust each other, which means they all intend to reveal sensitive data of others and protect their own privacy.To solve the problem above,they propose a privacy-preserving machine learning scheme in the multi-owner setting for a simple but highly effective classification, the Naive Bayes (NB) classification. The NB classifier model could meet the requirements of -differential privacy.

2) Linear Regression: In [11],they present a new algorithm called AdaOPS which use the idea of Kifer et al. [12], Blocki et al. [13], Sheffet [14] for analyzing other related differentially private algorithms.They only need $n\varepsilon^2 = o1)$ to achieve asymptotic efficiency.

3) Linear SVM:

4) Logistic Regression:

5) Kernel SVM: In [15],they present a pair of mechanisms for private SVM learning, each of which releases a classifier based on a privacy-sensitive database of training data. In each case they establish differential privacy of their mechanisms via the algorithmic stability of regularized ERMa property that for any delta, no mechanism can be simultaneously (epsilon,delta)-useful and beta-differentially private for small epsilon and small beta.

6) Decision Tree Learning: [16]

B. Differentially Private Unsupervised Learning

1) K-means clustering: [17]

C. Differentially Private Dimensionality Reduction

1) Feature Selection:

IV. Conclusion

The conclusion goes here.

References

[1] L. Sweeney, "k-anonymity: a model for protecting privacy." in *International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2012, pp. 54–59. I

[2] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "L-diversity: privacy beyond k-anonymity," in *International Conference on Data Engineering*, 2006, pp. 24–24. I

[3] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *IEEE International Conference on Data Engineering*, 2007, pp. 106–115. I

[4] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," pp. 75–84, 2007. 1, 2, 3, 5

[5] B. C. Dwork, "A firm foundation for private data analysis," *Communications of the Acm*, vol. 54, no. 1, pp. 86–95, 2011. 4

[6] F. D. Mcsherry, "Privacy integrated queries," *Communications of the Acm*, vol. 53, no. 9, 2010. 1, 2

[7] C. Dwork, F. Mcsherry, and K. Nissim, *Calibrating Noise to Sensitivity in Private Data Analysis*. Springer Berlin Heidelberg, 2006. 6

[8] F. Mcsherry and K. Talwar, "Mechanism design via differential privacy," in *IEEE Symposium on Foundations of Computer Science*, 2007, pp. 94–103. 7

[9] G. Barthe, G. P. Farina, M. Gaboardi, E. J. G. Arias, A. Gordon, J. Hsu, and P. Y. Strub, "Differentially private bayesian programming," in *ACM Sigsac Conference on Computer and Communications Security*, 2016, pp. 68–79. III-A1

[10] T. Li, J. Li, Z. Liu, P. Li, and C. Jia, "Differentially private naive bayes learning over multiple data sources," *Information Sciences*, vol. 444, 2018. III-A1

[11] Y. X. Wang, "Per-instance differential privacy and the adaptivity of posterior sampling in linear and ridge regression," 2017. III-A2

[12] D. Kifer, A. Smith, and A. Thakurta, "Private convex empirical risk minimization and high-dimensional regression," *Journal of Machine Learning Research*, vol. 1, 2013. III-A2

[13] K. Kenthapadi, A. Korolova, I. Mironov, and N. Mishra, "Privacy via the johnson-lindenstrauss transform," *Computer Science*, 2012. III-A2

[14] O. Sheffet, "Differentially private ordinary least squares: t-values, confidence intervals and rejecting null-hypotheses," *Computer Science*, 2015. III-A2

[15] B. I. P. Rubinstein, P. L. Bartlett, H. Ling, and N. Taft, "Learning in a large function space: Privacy-preserving mechanisms for svm learning," *Eprint Arxiv*, vol. 4, no. 1, 2009. III-A5

[16] S. Fletcher and M. Z. Islam, "Decision tree classification with differential privacy: A survey," 2016. III-A6

[17] R. Pinot, A. Morvan, F. Yger, C. Gouypailler, and J. Atif, "Graph-based clustering under differential privacy," 2018. III-B1