Chapter 2
Preliminary of Differential Privacy

2.1 Notations

Figure 2.1 shows a basic attack model. Suppose a curator would like to preserve privacy for n records in a dataset D. However, an attacker has all information about $n-1$ records in dataset D except the nth record. These $n-1$ records can be defined as background information. He/she can make a query on the dataset D to get aggregate information about n records in D. After compare the difference between query result with the background information, the attacker can easily identify the information of record n.

Differential privacy aims to resist the attack. Differential privacy acquires the intuition that releasing an aggregated result should not reveal too much information about any individual record in the dataset. Figure 2.2 shows how differential privacy resists the attack. We define a dataset D' that differs with D with only one record, say, x_n. When the attacker make the query f on both datasets, he/she has a very high probability to get the same result s. Based on the results, he/she cannot identify whether x_n is in D or not. When the attacker cannot tell the difference between the
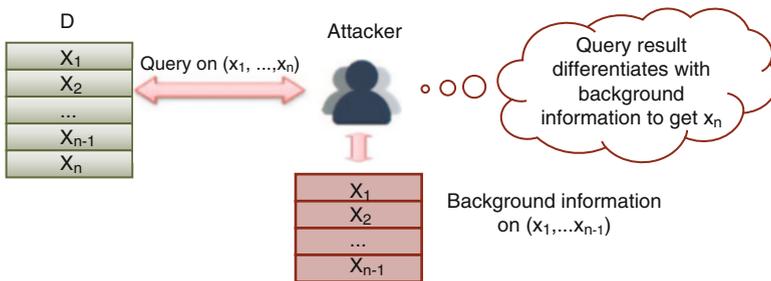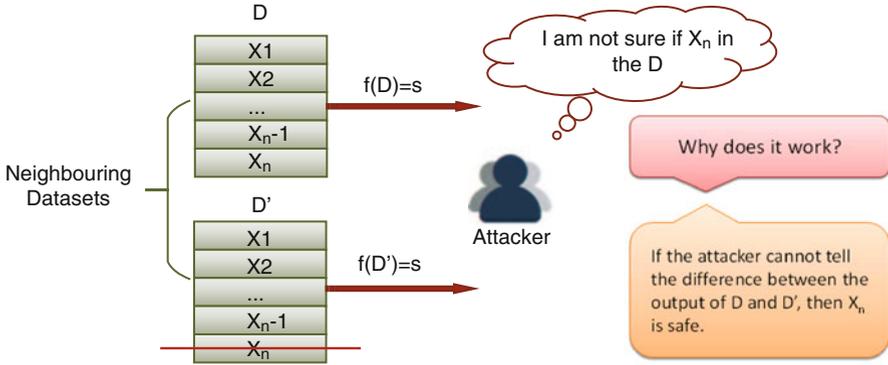

Fig. 2.1 Attacker model
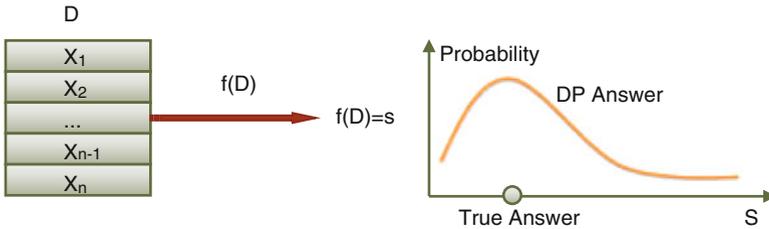
Fig. 2.2 Differential privacy

Fig. 2.3 Query answer distribution

outputs of D and D', then x_n is safe. If the property is applicable for all records in D, the dataset D can preserve privacy for all records.

In differential privacy, curator will not publish a dataset directly, instead, public users submit their statistical queries to the curator, and curator replies them with query answers. For a particular query, its true answer is unique, but its differentially private answer is a distribution, as shown in Fig. 2.3, dataset D and D' have very high probability to output same results.

To present the definition of differential privacy formally, we use the following key notations in the book. We consider a finite *data universe* $\mathscr{X}$ with the size $|\mathscr{X}|$. Let r represent a record with d attributes, a dataset D is an unordered set of n records sampled from the universe $\mathscr{X}$. Two datasets D and D' are defined as neighboring datasets if differing in one record. A query f is a function that maps dataset D to an abstract range $\mathbb{R}: f : D \rightarrow \mathbb{R}$. A group of queries is denoted as F. Normally, we use symbol m to denote the number of queries in F. There are various types of queries, such as count, sum, mean and range queries.

The target of differential privacy is to mask the difference of query f between the neighboring datasets [64]. The maximal difference on the results of query f is defined as the *sensitivity* Δf, which determines how much perturbation is required for the private preserving answer. To achieve the perturbation target, differential privacy provides a mechanism M accesses the dataset and implements

Table 2.1 Notations

Notations	Explanation	Notations	Explanation
$\mathscr{X}$	Universe	D	Dataset; training sample set
D'	Neighboring dataset	$\mathscr{D}$	Dataset distribution
r, x	Record in dataset; training sample	d	Dataset dimension
n	The size of dataset	N	The size of a histogram
f	Query	F	Query set
m	The number of queries in F	M	Mechanism
$\widehat{f}$	Noisy output	k	Represent some small value of constant
ϵ	Privacy budget	$\triangle f$	Sensitivity
G	Graph data	t, T	Time, time sequence, or iterative round
$\mathbf{w}$	Output model, or weight	$VC(\cdot)$	VC dimension
$\ell(\cdot)$	Loss function	α, β, δ	Accuracy parameter

some functionality. The perturbed output is denoted by a 'hat' over the notation. For example, $\widehat{f}(D)$ denotes the randomized answer of querying f on D. Table 2.1 summarizes some major notations used in the book. There are some other Greek symbols such as θ, η will be used temporarily in different chapters.

2.2 Differential Privacy Definition

A formal definition of differential privacy is shown below:

Definition 2.1 ((ϵ, δ)-Differential Privacy [67]) A randomized mechanism M gives (ϵ, δ)-differential privacy for every set of outputs S, and for any neighbouring datasets of D and D', if M satisfies:

$$Pr[M(D) \in S] \leq \exp(\epsilon) \cdot Pr[M(D') \in S] + \delta. \qquad (2.1)$$

Figure 2.4 shows mechanism on neighbouring datasets. For a particular output, the ratio on two probabilities is bounded by e^{ϵ}. If $\delta = 0$, the randomized mechanism M gives ϵ-differential privacy by its strictest definition. (ϵ, δ)-differential privacy provides freedom to violate strict ϵ-differential privacy for some low probability events. ϵ-differential privacy is usually called *pure differential privacy*, while (ϵ, δ)-differential privacy with $\delta > 0$ is called *approximate differential privacy* [20].

2.2.1 The Privacy Budget

In Definition 2.1, parameter ϵ is defined as the *privacy budget* [89], which controls the privacy guarantee level of mechanism M. A smaller ϵ represents a stronger privacy. In practice, ϵ is usually set as less than 1, such as 0.1 or ln 2. Two privacy

Fig. 2.4 Query answer
distribution
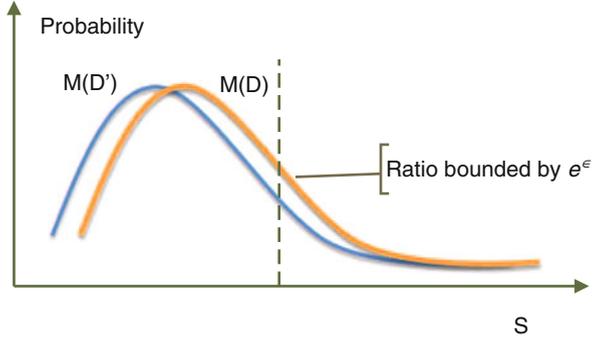
composition theorems are widely used in the design of mechanisms: sequential composition [157] and parallel composition [155], as defined in Theorem 2.1 and Theorem 2.2, respectively.

Theorem 2.1 (Parallel Composition)   *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 $(\max\{\epsilon_1, \ldots, \epsilon_m\})$-differential privacy.*
The *parallel composition* corresponds to a case where each M_i is applied on disjoint subsets of the dataset. The ultimate privacy guarantee only depends on the largest *privacy budget* allocated to M_i.

Theorem 2.2 (Sequential Composition)   *Suppose a set of privacy mechanisms $M = \{M_1, \ldots M_m\}$ are sequentially performed on a dataset, and each M_i provides ϵ_i privacy guarantee, M will provide $(\sum_{i=1}^{m} \epsilon)$-differential privacy.*
The sequential composition undertakes the privacy guarantee for a sequence of differentially private computations. When a set of randomized mechanisms has been performed sequentially on a dataset, the final privacy guarantee is determined by the summation of total privacy budgets.

These two composition theorems bound the degradation of privacy when composing several differentially private mechanisms. Figure 2.5 shows their differences. Based on them, Kairouz [112] and Murtagh [164] provided optimal bounds when m mechanisms are adaptive, which means that M_i will be designed based on the result of M_{i-1}. They claimed that the privacy budgets will be consumed less when the mechanisms are adaptive. Currently, however, the parallel and sequential composition are most prevalent and straightforward way to analysis the privacy budget consuming of a privacy preserving algorithm.

2.3   The Sensitivity

Sensitivity determines how much perturbation is required in the mechanism. For example, when we publish a specified query f of dataset D, the sensitivity will calibrate the volume of noise for $f(D)$. Two types of sensitivity are employed in differential privacy: the *global sensitivity* and the *local sensitivity*.
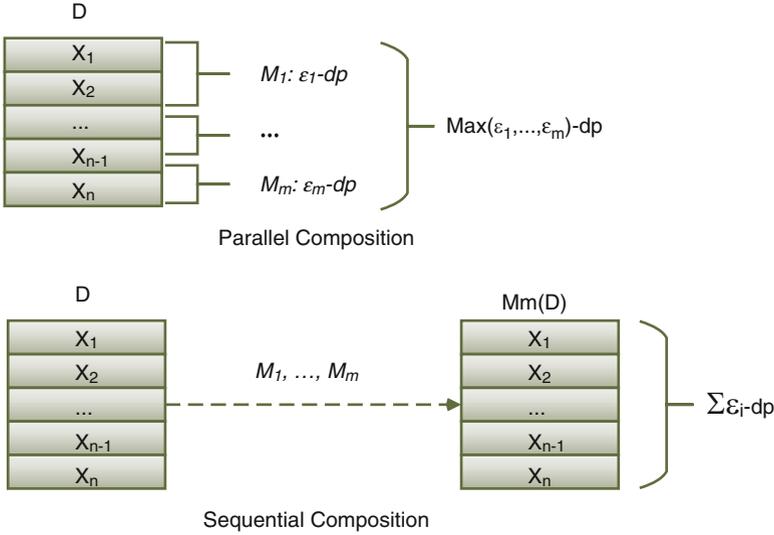
Fig. 2.5 Privacy budget composition

2.3.1   The Global Sensitivity

The global sensitivity is only related to the type of query f. It considers the maximum difference between query results on neighboring datasets. The formal definition is as below:

Definition 2.2 (Global Sensitivity)   For a query $f : D \rightarrow \mathbb{R}$, the *global sensitivity* of f is defined as

$$\triangle f_{GS} = \max_{D,D'} ||f(D) - f(D')||_1. \qquad (2.2)$$

Global sensitivity works well when queries have relative lower sensitivity values, such as `count` or `sum` queries. For example, the `count` query normally has $\triangle f_{GS} = 1$. When the true answer is over hundreds or thousands, the sensitivity is much lower than the true answer. But for queries such as `median`, `average`, the global sensitivity yields high values comparing with true answers. We will then resort to local sensitivity for those queries [172].

2.3.2   The Local Sensitivity

Local sensitivity calibrates the record-based difference between query results on neighboring datasets [172]. Comparing with the global sensitivity, it takes both the record and query into consideration. The local sensitivity is defined as below:

Fig. 2.6 Local sensitivity and smooth bound
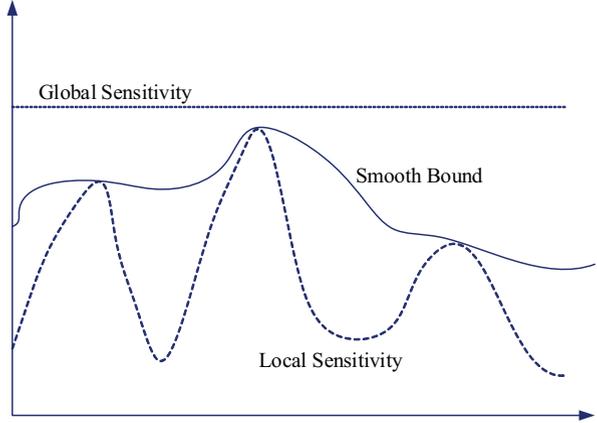
Definition 2.3 (Local Sensitivity)   For query $f : D \rightarrow \mathbb{R}$, *local sensitivity* is defined as

$$\triangle f_{LS} = \max_{D'} ||f(D) - f(D')||_1. \tag{2.3}$$

For many queries, such as the `median`, the local sensitivity is much smaller than the global sensitivity. However, as the changing of local sensitivity may result in information disclosure, it cannot be used in mechanisms directly. The value of local sensitivity should be changed smoothly, so that a smooth bound should be added.

Definition 2.4 (Smooth Bound)   For $\beta > 0$, a function $B : D \rightarrow \mathbb{R}$ is a β-smooth upper bound on the local sensitivity of f if it satisfies the following requirements,

$$\forall D \in X : B(D) \geq f_{LS}(D) \tag{2.4}$$

$$\forall D, D' \in X : B(D) \leq e^{\beta} S(D). \tag{2.5}$$

Figure 2.6 shows the relationship between the local sensitivity, smooth bound and the global sensitivity. For some queries, the local sensitivity is lower than global sensitivity. For queries such as `count` or `range`, the local sensitivity is identical to global sensitivity. Because most literatures were concerned with the global sensitivity, without specification, sensitivity refers to global sensitivity in this book.

2.4 The Principle Differential Privacy Mechanisms

Any mechanism meeting Definition 2.1 can be considered as differentially private. Currently, three basic mechanisms are widely used to guarantee differential privacy: the Laplace mechanism [68], the Gaussian mechanism [72] and the exponential

mechanism [157]. The Laplace and Gaussian mechanisms are suitable for numeric queries and the exponential mechanism is suitable for non-numeric queries.

2.4.1 The Laplace Mechanism

The Laplace mechanism relies on adding controlled Laplace noise to the query result before returning it to the user. The noise is sampled from the Laplace distribution, which is centered at 0 with scaling b. The noise is presented by $Lap(b)$, in which a larger b indicates a higher noise. The corresponding probability density function is:

$$Lap(x) = \frac{1}{2b} \exp(-\frac{|x|}{b}). \tag{2.6}$$

The mechanism is defined as follows:

Definition 2.5 (Laplace Mechanism) Given a function $f : D \rightarrow \mathbb{R}$ over a dataset D, mechanism M provides the ϵ-differential privacy if it follows Eq. (2.5)

$$M(D) = f(D) + Lap(\frac{\Delta f}{\epsilon}). \tag{2.7}$$

The mechanism shows that the size of noise is related to the sensitivity of query f and the privacy budget ϵ. A larger sensitivity leads to a higher volume of noise.

2.4.1.1 The Gaussian Mechanism

To achieve (ϵ, δ)-differential privacy, one can use Gaussian noise [72]. In this case, rather than scaling the noise to the ℓ_1 sensitivity, one instead scales to the ℓ_2 sensitivity as follow Definition 2.6:

Definition 2.6 (ℓ_2-Sensitivity) For a query $f : D \rightarrow \mathbb{R}$, the ℓ_2-sensitivity of f is defined as

$$\triangle_2 f = \max_{D,D'} ||f(D) - f(D')||_2. \tag{2.8}$$

The Gaussian mechanism with parameter σ adds zero-mean Gaussian noise with variance σ.

Definition 2.7 (Gaussian Mechanism) Given a function $f : D \rightarrow \mathbb{R}$ over a dataset D, if $\sigma = \triangle_2 f \sqrt{2ln(2/\delta)}/\epsilon$ and $\mathcal{N}(0, \sigma^2)$ are i.i.d. Gaussian random variable, mechanism M provides the ϵ, δ-differential privacy when it follows Eq. (2.7)

$$M(D) = f(D) + \mathcal{N}(0, \sigma^2). \tag{2.9}$$

The Gaussian mechanism follows the same privacy composition to the Laplace mechanism.

2.4.2   The Exponential Mechanism

For non-numeric queries, differential privacy uses an exponential mechanism to randomize the results, and this is paired with a score function $q(D, \phi)$ that represents how good an output ϕ is for dataset D. The choice of score function is application dependent and different applications lead to various score functions. The Exponential mechanism is formally defined as below:

Definition 2.8 (Exponential Mechanism)   Let $q(D, \phi)$ be a score function of dataset D that measures the quality of output $\phi \in \Phi$. Then an Exponential mechanism M is ϵ-differential privacy if

$$M(D) = \{\text{return } \phi \text{ with the probability} \propto \exp(\frac{\epsilon q(D, \phi)}{2\Delta q})\}. \qquad (2.10)$$

where Δq represents the *sensitivity* of score function q.

2.4.2.1   Mechanism Example

An example is presented below to illustrate some fundamental concepts of the sensitivity, privacy budget and mechanisms. Suppose Table 2.2 shows a medical dataset D of a district, and differential privacy mechanism M will guarantee the privacy of each individual in D.

Suppose query f_1 asks: *how many people in this table have HIV?* Because the query result is numeric, we can use Laplace mechanism to guarantee differential privacy. First, we analyse the sensitivity of f_1. According to Definition 2.2, deleting a record in this D will affect the query result maximally by 1. The sensitivity of f_1 is $\Delta f_1 = 1$. Second, we choose a privacy budget ϵ for the Laplace mechanism. Suppose we set $\epsilon = 1.0$. According to Definition 2.5, the noise that sample from

Table 2.2 Medical record

Name	Job	Gender	Age	Disease
Alen	Engineer	Male	25	Flu
Bob	Engineer	Male	29	HIV
Cathy	Lawyer	Female	35	Hepatitis
David	Writer	Male	41	HIV
Emily	Writer	Female	56	Diabetes
…	…	…	…	…
Emma	Dancer	Female	21	Flu

Table 2.3 Medical record exponential mechanism output

Options	Number of people	$\epsilon = 0$	$\epsilon = 0.1$	$\epsilon = 1$
Diabetes	24	0.25	0.32	0.12
Hepatitis	8	0.25	0.15	4×10^{-5}
Flu	28	0.25	0.40	0.88
HIV	5	0.25	0.13	8.9×10^{-6}

$Lap(1)$ will be added to the true answer $f_1(D)$. Lastly, the mechanism M will output a noisy answer $M(D) = f_1(D) + Lap(1)$. If the true answer is 10, the noisy answer might be 11.

Suppose we have another query f_2: *what is the most common disease in this district?* This query will generate non-numeric result and we can apply the exponential mechanism. Table 2.3 lists all the diseases and their true numbers in the first two columns. We first define the score function of f_2. We adopt the number of people on each disease as the score function q. As deleting a person will have a maximum impact of 1 on the result of q, the sensitivity of q is $\triangle q = 1$. The probability of the output can then be calculated by Definition 2.8. Table 2.3 lists the results when $\epsilon = 0$, $\epsilon = 0.1$ and $\epsilon = 1$.

In the third column of Table 2.3, $\epsilon = 0$ means that the mechanism chooses an answer uniformly from those four options. The output probabilities are equal in these options. Obviously, $\epsilon = 0$ provides the highest privacy level, however it loses almost all the utility. When $\epsilon = 0.1$, *Flu* has the highest probability of being chosen and *HIV* has the lowest probability. The gap is not very large, which indicates that it can provide acceptable privacy and utility levels. When $\epsilon = 1$, the probability gap between *HIV* and other diseases is significant, which means that the mechanism can retain a high utility, but have a lower privacy level.

2.5   Utility Measurement of Differential Privacy

When privacy level is fixed to ϵ, several utility measurements have been used in both data publishing and analysis to evaluate the performance of differential privacy mechanisms.

- *Noise size measurement*: the easiest way is calibrating how much noise has been added to the query results. A smaller noise indicates higher utility. This utility measurement has been widely used in data publishing.
- *Error measurement*: utility can be evaluated by the difference between the non-private output and the private output. For example, the utility of single query publishing can be measured by $|f(D) - \widehat{f}(D)|$. A smaller distance shows higher utility. The error measurement is normally represented by a bound with accuracy parameters [29]:

Definition 2.9 ((α,β)-Useful)   A set of query F is (α,β)-utility if

$$Pr(\max_{f \in F} |F(D) - \widehat{F}(D)| \leq \alpha) > 1 - \beta, \qquad (2.11)$$

where α is the accuracy parameter that bounds the error.

For different publishing scenarios, the error measurement can be interpreted in various ways. For synthetic dataset publishing, Eq. (2.11) can be interpreted to:

$$Pr(\max_{f \in F} |F(D) - F(\widehat{D})| \leq \alpha) > 1 - \beta. \qquad (2.12)$$

For data analysis, the utility normally depends on the analysis algorithms. Suppose the algorithm is denoted by M and the private algorithm is denoted by $\widehat{M}$, Eq. (2.11) can be interpreted to

$$Pr(|M(D) - \widehat{M}(D)| \leq \alpha) > 1 - \beta. \qquad (2.13)$$

Equation (2.13) has several implementations in data analysis, such as risk bound and sample complexity.