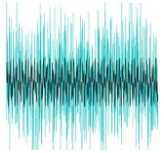
隐私保护

清华大学

本章的内容组织

第一节
隐私保护技术初探

第二节
匿名化



第三节
差分隐私(DP)

第四节
同态加密

第五节
安全多方计算

第六节
联邦学习

- 网络空间安全中的隐私
- 隐私泄露的危害
- 隐私保护技术介绍

隐私泄露事件造成的危害让人们看到了隐私保护的重要性

- 匿名化隐私保护模型
- 数据匿名化方法

如何安全地发布数据供其他机构研究

- 差分隐私基础
- 数值型DP
- 非数值型DP

如何保护统计信息中的个体隐私

- 同态加密基础
- 半同态加密
- 全同态加密

如何安全地将数据委托给数据计算方

- 安全多方计算基础
- 百万富翁协议

如何帮助互不信任的参与方进行协同计算

- 联邦学习的基础
- 横向联邦
- 纵向联邦

如何实现在不共享数据下的模型协同训练

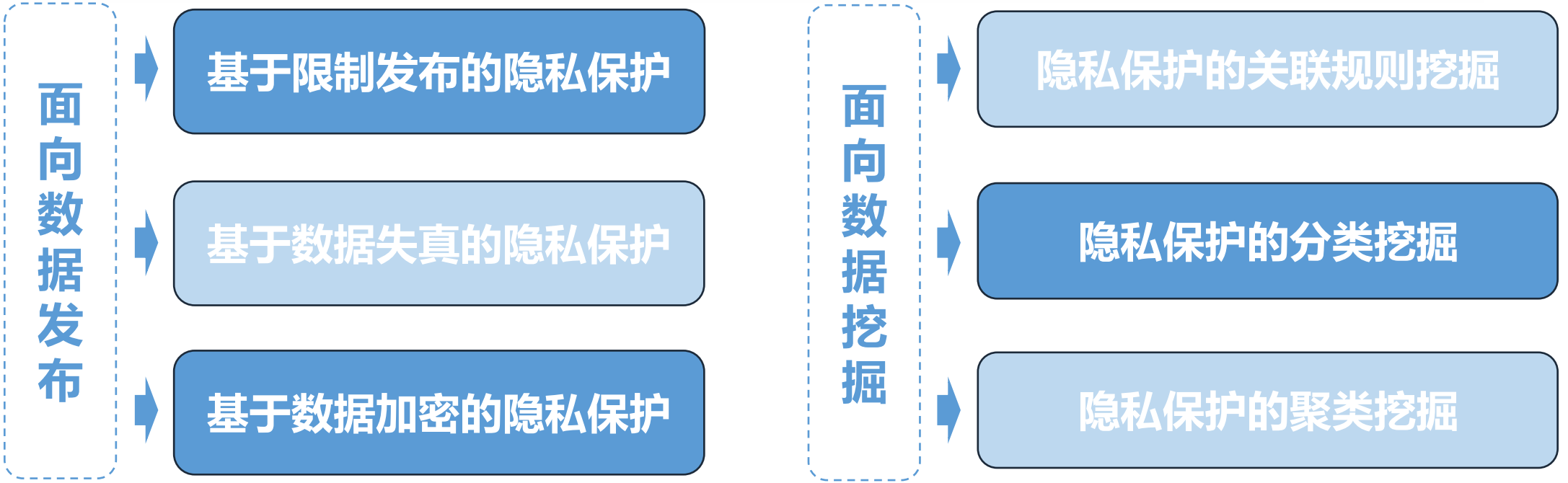
隐私泄露的巨大危害促使人们考虑多种场景下的数据隐私保护问题

第1节 隐私保护技术初探

- 网络空间安全中的隐私
- 隐私泄露的危害
- 隐私保护技术介绍

隐私保护技术初探

- 为了从大量的网络空间数据中获取有用信息，需要对其进行挖掘，在此过程中不免会造成数据隐私泄露，如何在获取有用信息的同时保护数据相关者的隐私变得尤为重要
- 数据隐私保护技术的研究主要分为两个方面：面向数据发布的隐私保护研究和面向数据挖掘的隐私保护研究



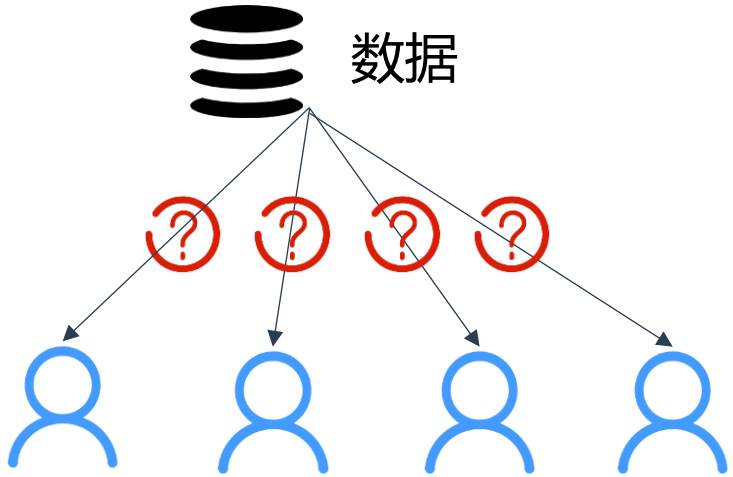
面向数据发布：基于限制发布的隐私保护

面向数据发布的隐私保护是在将数据公布给数据挖掘者之前，对数据进行扰动、加密、匿名等处理，将数据中的隐私藏起来

基于限制发布的隐私保护

有选择的发布原始数据、不发布或者发布精度较低的敏感数据

- 研究集中于数据匿名化
  - 研究更好的匿名化原则
  - 针对匿名化原则设计更高效的匿名化算法



隐藏用户身份和数据的对应关系

面向数据发布：基于数据失真的隐私保护

对原始数据进行扰动，目的是隐藏真实数据，只呈现出数据的统计学特征

基于数据失真的隐私保护技术主要包括随机化、阻塞、变形、交换等

失真后的数据满足

- 数据保持原本的某些特性不变
- 攻击者不能根据失真数据重构出真实的原始数据

- 随机化：在原始数据中加入随机噪声，从而隐藏真实的数据，保护敏感数据
- 数据交换：在记录之间交换数值以扰动真实数值，但同时要保留某些统计学特征

面向数据发布：基于数据加密的隐私保护

对原始数据进行加密，通过密码机制实现其他参与方对原始数据的不可见性以及数据的无损失性

由于加密技术可解决安全通信的问题，因此多应用于分布式应用

可使用的加密技术有

- 对称可加密搜索
- 安全多方计算
- 同态加密技术
- 数字信封技术
- Shamir秘密技术共享
- ……

可应用的分布式应用 ----▸

- 分布式数据挖掘
- 分布式安全查询
- ……

两种数据存储模式

- 垂直划分：每个参与者只存储部分属性的数据，所有参与者存储的数据不重复
- 水平划分：将数据记录存储到多个参与者处，所有参与者存储的数据不重复

在两种存储模式中，每个参与者都只掌握了部分数据

面向数据发布的隐私保护总结

基于限制发布

有选择的发布原始数据、不发布或者发布精度较低的敏感数据

基于数据失真

对原始数据进行扰动，目的是隐藏真实数据，只呈现出数据的统计学特征

基于数据加密

对原始数据进行加密，通过密码机制实现其他参与方对原始数据的不可见性以及数据的无损失性

面向数据发布的隐私保护技术	优点	缺点
基于限制发布的隐私保护技术	发布的数据真实可靠	数据丢失部分信息
基于数据失真的隐私保护技术	算法效率较高	由于干扰使数据丢失部分信息
基于数据加密的隐私保护技术	数据的安全性和准确性均较高	计算开销很大

面向数据挖掘：关联规则的数据挖掘

关联规则是寻找在同一事件中出现的不同项目的相关性，关联规则挖掘是数据挖掘领域研究的重点之一，是从大量数据中挖掘数据项之间隐藏的关系，发现数据集中项集之间的关联和规则的过程，其通过置信度和支持度度量项集之间的规则

置信度：购买了一个商品之后又购买了另一种商品的可能性：

80/100=80%

支持度：购买关联商品的人数占总人数的比例：

80/1000=8%

设定最小置信度和最小支持度，当挖掘到的项集的置信度和支持度分别大于最小置信度和最小支持度时，就得到了关联规则

面向数据挖掘：隐私保护的关联规则挖掘

变换（distortion）

修改支持敏感规则的数据，使得规则的置信度和支持度小于一定的阈值而实现规则的隐藏

隐藏（blocking）

不修改数据，而是隐藏生成敏感规则的频繁项集，尽可能降低敏感规则的置信度或者支持度，以此使得需要保护或隐藏的规则不被挖掘出来

两类方法 ⚠ 都会影响对非敏感规则的挖掘

面向数据挖掘：隐私保护的分类和聚类挖掘

隐私保护的分类挖掘

分类：在数据集上构造分类函数或者分类模型，即分类器，将数据集中的数据项映射到给定的类别中，以用于类别的预测

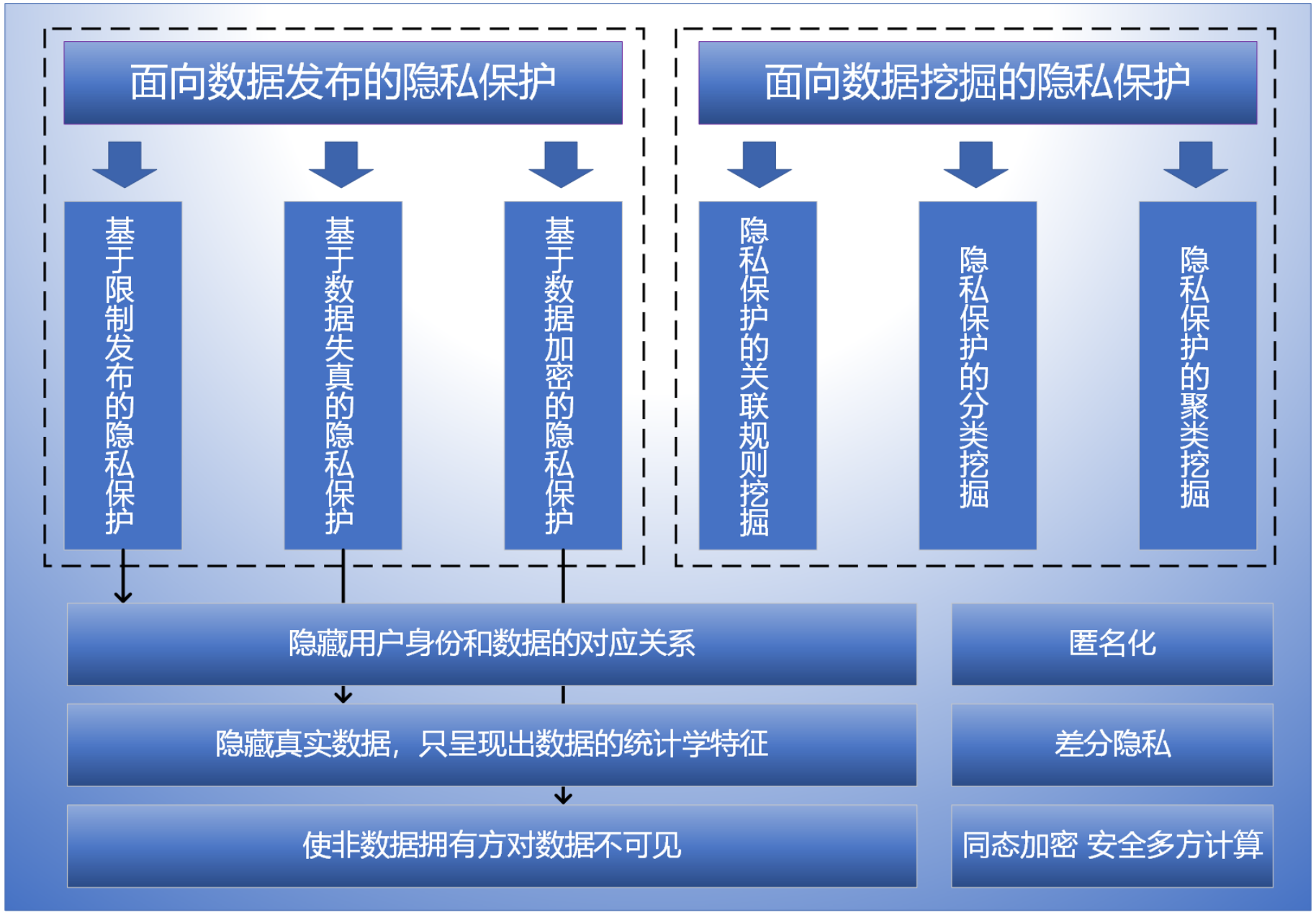
- 分类结果可能会暴露隐私信息
- 隐私保护的分类挖掘是指在数据挖掘的过程中，建立准确的、无隐私泄露的分类模型

隐私保护的聚类挖掘

聚类：将数据集中的数据根据相似性进行分类，最后的分类结果中同一个类别中的数据相似性越大越好，不同类别中的数据的相似性越小越好

- 与分类挖掘相同，由于聚类结果可能会暴露数据集中的隐私敏感信息，因此需要使用隐私保护技术保护敏感的分类结果信息

数据隐私保护技术总结


面向数据发布的隐私保护
面向数据挖掘的隐私保护
基于限制发布的隐私保护
基于数据失真的隐私保护
基于数据加密的隐私保护
隐私保护的关联规则挖掘
隐私保护的分类挖掘
隐私保护的聚类挖掘
隐藏用户身份和数据的对应关系
匿名化
隐藏真实数据，只呈现出数据的统计学特征
差分隐私
使非数据拥有方对数据不可见
同态加密 安全多方计算

第2节 匿名化

- 匿名化隐私保护模型
- 数据匿名化方法

匿名化隐私保护模型：k-anonymity

k-anonymity是由Pierangela Samarati和Latanya Sweeney于1998年提出的隐私保护模型，有效解决了链接攻击问题

- 等价类：匿名化后的数据表中具有相同准标识符的若干记录称为一个等价类
- k-anonymity：将k个记录放入一个等价类中，**要求任意一条记录与其他至少k-1条记录相似而不可区分**，这样数据中的每一条记录都能找到与之相似的记录，降低了数据的识别度

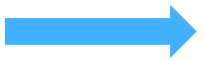
如果一条记录由于样本太少而无法找到k-1条相似的记录，那么这条数据不应当被纳入数据集

k-anonymity

攻击者进行链接攻击时，由于对任意一条记录的攻击，都会同时关联到等价类中的其他k-1条记录，因此攻击者无法确定特定用户

公布的病患数据

年龄	邮政编码	疾病
52	123023	心脏病
32	120156	糖尿病
59	123152	心脏病
30	120162	糖尿病
56	123485	心脏病
35	120154	哮喘



病患数据的3-anonymity版

年龄	邮政编码	疾病
5*	123***	心脏病
5*	123***	心脏病
5*	123***	心脏病
3*	1201**	糖尿病
3*	1201**	糖尿病
3*	1201**	哮喘

同质性攻击

k-anonymity保证了单独个体被准确标识的概率最大为1/k，但却无法保证隐私不被泄露

病患数据的3-anonymity版本

年龄	邮政编码	疾病
5*	123***	心脏病
5*	123***	心脏病
5*	123***	心脏病
3*	1201**	糖尿病
3*	1201**	糖尿病
3*	1201**	哮喘

同质性攻击：在数据匿名化过程中，由于没有对敏感属性进行约束，最终结果可能会造成隐私泄露

如果一名选民的年龄和邮政编码符合第一个等价类要求，那么攻击者可**推断该选民可能患有心脏病**

背景知识攻击

k-anonymity保证了单独个体被准确标识的概率最大为1/k，但却无法保证隐私不被泄露

病患数据的3-anonymity版本

年龄	邮政编码	疾病
5*	123***	心脏病
5*	123***	心脏病
5*	123***	心脏病
3*	1201**	糖尿病
3*	1201**	糖尿病
3*	1201**	哮喘

背景知识攻击：攻击者可以通过掌握的足够的相关背景知识以很高的概率确定敏感数据与个体的对应关系，得到隐私信息

如果一名选民的年龄和邮政编码符合第二个等价类要求，并且攻击者发现他不像是患有哮喘，那么攻击者可**推断该选民可能患有糖尿病**

l-diversity

l-diversity在k-anonymity的基础上，要求保证**每一个等价类的敏感属性至少有l个不同的值**，即每个用户的敏感属性值在等价类中可以找到与此值不同的至少l-1个属性值，使攻击者最多只能以1/l的概率确认某个用户的敏感信息

病患数据的3-anonymity版本

年龄	邮政编码	疾病	
5*	123***	心脏病	1种取值
5*	123***	心脏病	
5*	123***	心脏病	
3*	1201**	糖尿病	2种取值
3*	1201**	糖尿病	
3*	1201**	哮喘	

3-diversity举例（敏感属性至少三种取值）

Data table of *l*-diversity, where *l*=3

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	<=40	*	Heart Disease
4	1305*	<=40	*	Viral Infection
9	1305*	<=40	*	Cancer
10	1305*	<=40	*	Cancer
5	1485*	>40	*	Cancer
6	1485*	>40	*	Heart Disease
7	1485*	>40	*	Viral Infection
8	1485*	>40	*	Viral Infection
2	1306*	<=40	*	Heart Disease
3	1306*	<=40	*	Viral Infection
11	1306*	<=40	*	Cancer
12	1306*	<=40	*	Cancer

l-diversity

l-diversity保证攻击者最多只能以1/l的概率确认某个用户的敏感信息，但无法保证隐私不被泄露

等价类中敏感值的分布与整个数据集中敏感值的分布具有明显的差别，攻击者可以以一定概率猜测目标用户的敏感属性值

				某疾病检测结果
				阴性
				阳性
				阴性
				…

1000条记录中有1%的阳性记录和99%的阴性记录，阳性检测结果更为敏感

2-diversity

- 每个等价类中疾病检测结果必须包含阴性和阳性两种结果
- 假设某等价类中有一半阳性记录和一半阴性记录，相比于整体1%阳性的概率，该等价类中的个体都有1/2的概率被认为是阳性，具有严重的隐私风险

l-diversity

l-diversity保证攻击者最多只能以1/l的概率确认某个用户的敏感信息，但无法保证隐私不被泄露

l-diversity并没有考虑语义信息也会为隐私信息带来泄露的风险

				工资
				5000
				5500
				6000
				…

敏感属性为工资

l-diversity

- 若某一个等价类中的工资这一属性的属性值全部在一个固定区间内，那么攻击者并不需要知道详细的属性值就可以通过这个区间就可以判断用户的工资水平

t-closeness

在k-anonymity和l-diversity的基础上，t-closeness考虑了敏感属性的分布问题，要求**所有等价类中的敏感属性的分布尽量接近该敏感属性的全局分布**，差异不能超过阈值t

				工资
				5000
				5500
				6000
				…

敏感属性为工资

t-closeness

- 保证工资的分布和整体的分布类似，进而很难推断出某人工资的高低

k-anonymity，l-diversity 和 t-closeness**以信息损失为代价**，隐私保护效果逐个提高，但是它们一定能保证隐私不被泄露吗

隐私泄露风险

k-anonymity、l-diversity和t-closeness不能够完全保护隐私不被泄露

- 造成较大的信息损失，信息损失可能会使数据使用者们做出误判
- 对所有敏感属性提供了相同程度的保护并且没有考虑语义关系，造成了不必要的信息损失

针对不同的问题，提出不同的匿名技术

- 不同的用户对于隐私信息有着程度不同的隐私保护要求
- 属性与属性之间的重要程度并不相同
- 没有考虑数据动态更新后重发布的隐私保护问题

个性化匿名技术

带权重的匿名策略

动态数据匿名化

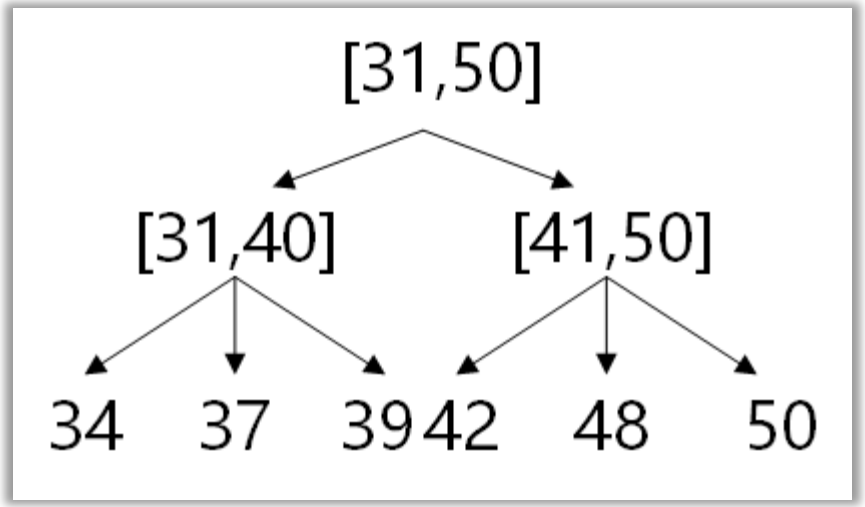
数据匿名化方法

- 实现匿名化的方法有泛化、抑制、聚类、微聚集、分解、置换等
- 目前提出的匿名化主要通过**泛化和抑制**实现，它们能保持发布前后数据的真实性和一致性

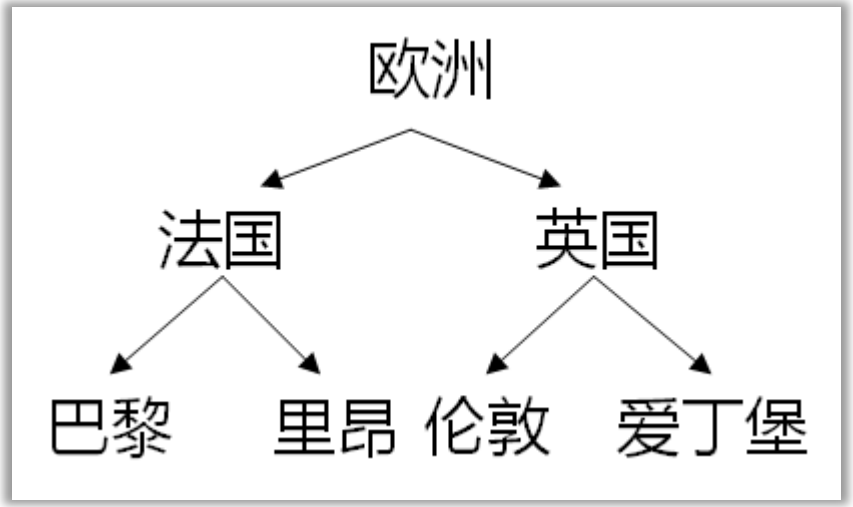
匿名化方法	思想
泛化	用更抽象、概括的值或区间代替精确值
抑制	将数据表中的数据直接删除或隐藏
聚类	按照给定的规则将数据集分成各类簇，尽量保证簇内对象相似，不同簇的对象相异
微聚集	相似的数据划分在同一个类中，每个类至少有k条记录，用类质心代替类中所有记录的准标识符属性值
分解	根据敏感属性值对数据表分组，尽量使得同一组的敏感属性值不同，将分组后的数据表拆分为分别包含准标识符属性信息和包含敏感属性信息的两张表
置换	对数据表分组，把每组内的敏感属性值随机交换，打乱顺序，再拆分数据表，对外发布

泛化

- **将准标识符的属性用更一般的值或者区间代替**
- 准标识符属性值有数值型和分类型
  - 数值型：值被一个覆盖精确数值的区间代替
  - 分类型：用一个更一般的值代替原值



数值型数据泛化树



分类型数据泛化树

不引入错误数据，方法简单，泛化后的数据适用性强，对数据的使用不需要很强的专业知识

预定义泛化树没有统一标准，信息损失大，对不同类型数据的信息损失度量标准不同

抑制

抑制，又称隐藏、隐匿，是将准标识符属性值从数据集中**直接删除或者用诸如“*”之类的代表不确定值的符号来代替**，与泛化结合使用

记录抑制	值抑制	单元抑制
对数据表中的某条记录进行抑制处理	对数据表中的某个属性的值全部进行抑制处理	对数据表中某个属性的部分值进行抑制处理

第3节 差分隐私

- 差分隐私基础
- 数值型差分隐私
- 非数值型差分隐私

匿名化与差分隐私

不能提供足够的数学保障，没有严格定义攻击模型，对攻击者所具有的知识未能作出定量化定义，无法抵抗背景知识攻击

不能提供严格和科学的方法证明其隐私保护水平，当模型参数改变时，无法对其隐私保护水平进行定量分析

匿名化

差分隐私

假设攻击者掌握最大知识背景，即能够获得除目标记录外所有其它记录的信息

对隐私保护进行了严格的定义并提供了量化评估方法，使不同参数处理下的数据集所提供的隐私保护水平具有可比较性

与匿名化相比，差分隐私是一种严格的可证明的隐私保护模型

差分攻击

姓名	是否患病
张三	0（不患病）
李四	1
王五	1
钱六	1

$f(3)$

2

$f(4)$

3

计数查询服务： $f(i) = count(i)$
查询数据集中前i行患病的记录数量

第四行代表的用户患病了！
若已知该用户是钱六，则可**推断钱六患病**

为抵抗差分攻击，差分隐私要求保证任意一个个体在数据集中或者不在数据集中时，对最终发布的查询结果几乎没有影响

概念介绍

隐私保护机制

对数据集D的各种映射函数被定义为查询（Query），用$F = \{f1, f2, \cdots\}$来表示一组查询，算法M对查询 F 的结果进行处理，使之满足隐私保护的条件，此过程称为隐私保护机制

邻近数据集

设数据集D 和D′具有相同的属性结构，两者的对称差记作$D\Delta D'$，$|D\Delta D'|$表示$D\Delta D'$中记录的数量，**若$|D\Delta D'| = 1$，则称D和D′为邻近数据集**（Adjacent Dataset)

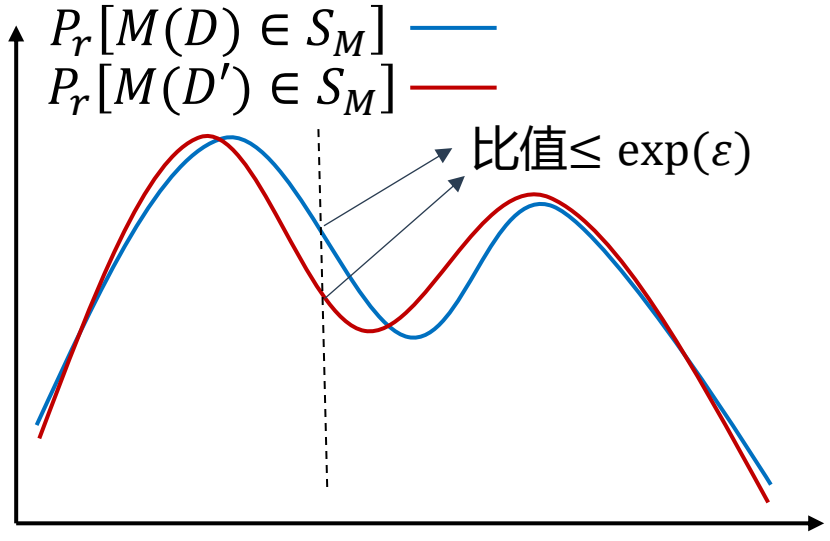
例如，设$D = \{1,2,3,4,5\}$，$D' = \{1,2,4\}$，则$D\Delta D' = \{3,5\}$，$|D\Delta D'| = 2$

差分隐私定义

差分隐私（Differential Privacy，DP）的定义：设有一个随机算法M，P_M为算法M所有可能的输出构成的集合，如果对于任意两个邻近数据集D和D'以及P_M的任意子集S_M，算法M满足：

$$P_r[M(D) \in S_M] \leq \exp(\varepsilon) \times P_r[M(D') \in S_M]$$

则称算法M提供ε –差分隐私保护，其中**参数ε称为隐私保护预算**



- $\varepsilon = 0$：攻击者无法区分相邻数据集，保护程度最高，数据可用性最差
- ε增大：保护程度越来越低，ε过大，则会造成隐私泄露
- ε通常取很小的值，例如0.01，0.1，或者$\ln 2$，$\ln 3$等

ε的取值应当结合具体需求设定以达到输出结果的安全性与可用性的平衡

差分隐私的实现

差分隐私可以通过在查询函数的返回值中加入噪声来实现

增大加入的噪声 ⇨ 数据可用性下降

完全随机的数据没有意义

减小加入的噪声 ⇨ 数据安全性下降

无法保护个体隐私

全局敏感度

全局敏感度：设有函数$f: D \to R_d$，输入为数据集，输出为d维实数向量。对**任意的邻近数据集D 和D′**，

$$GS_f = \max_{D,D'} \|f(D) - f(D')\|_1$$

称为函数f的全局敏感度，其中$\|f(D) - f(D')\|_1$是$f(D)$和$f(D')$之间的1 –阶范数距离

所有维度上的距离之和，若查询结果是一维的(select a from ...)，距离为$|a - a'|$（两个数字的差的绝对值），若是二维的（select a, b from ...），距离为$|a - a'| + |b - b'|$

姓名	是否患病
张三	0
李四	1
王五	1
钱六	1

姓名	是否患病
张三	0
李四	1
钱六	1

数据集D：$f(D)$ =3　　数据集D'：$f(D')$=2

以计数查询函数f（查询患病人数）为例，对任意的D和D'，由于一条记录的有无只会使f输出的差值为1，因此该函数的全局敏感度为1

全局敏感度反映了一个查询函数在一对邻近数据集上进行查询时变化的最大范围，它与数据集无关，由查询函数本身决定

局部敏感度

局部敏感度：设有函数$f: D \to R_d$，输入为数据集，输出为d维实数向量。对于**给定的数据集D和它的任意邻近数据集D′**，

$$LS_f = \max_{D'} \|f(D) - f(D')\|_1$$

称为函数f在D上的局部敏感度

全局敏感度

- 对任意的邻近数据集D 和D′
- 只由查询函数决定

给定的数据集与全局敏感度中使1-阶范式距离达到最大的数据集相同时，局部敏感度就等于全局敏感度

局部敏感度

- 对给定的数据集D 和它的任意邻近数据集D′
- 由查询函数和给定的数据集中的数据共同决定

数值型差分隐私：拉普拉斯和高斯机制

数值型差分隐私的实现机制有**拉普拉斯机制**和高斯机制，通过在查询结果中加入随机噪声实现隐私保护

拉普拉斯机制提供的是严格的$(\varepsilon, 0)$ −差分隐私保护，而高斯机制提供的是松弛的(ε, δ) −差分隐私保护

拉普拉斯分布

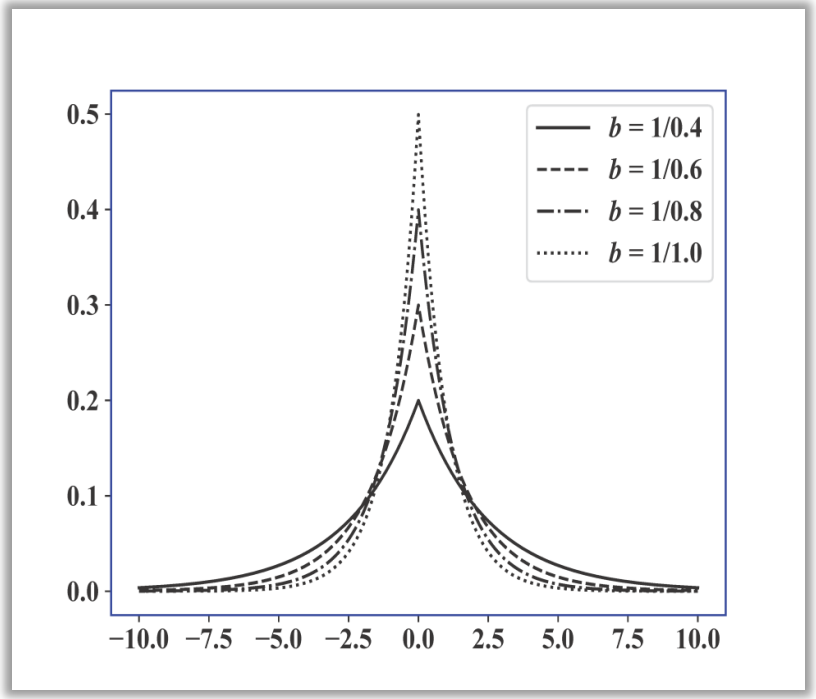
拉普拉斯分布是一种连续的概率分布，其概率密度函数为：

$$f(x|\mu, b) = \frac{1}{2b} exp\left\{-\frac{|x-\mu|}{b}\right\}$$

其中位置参数为μ，尺度参数为$b(b > 0)$，该分布的期望值为μ，方差为$2b^2$

记位置参数μ为0，尺度参数为b的拉普拉斯分布为Lap(b)，它的概率密度函数为：

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



不同尺度参数下的拉普拉斯分布图像

拉普拉斯机制的定义

拉普拉斯机制是一种广泛应用于数值型差分隐私的隐私保护机制，其思想为在数值型数据的查询结果中添加随机的满足拉普拉斯分布的噪声来实现差分隐私保护

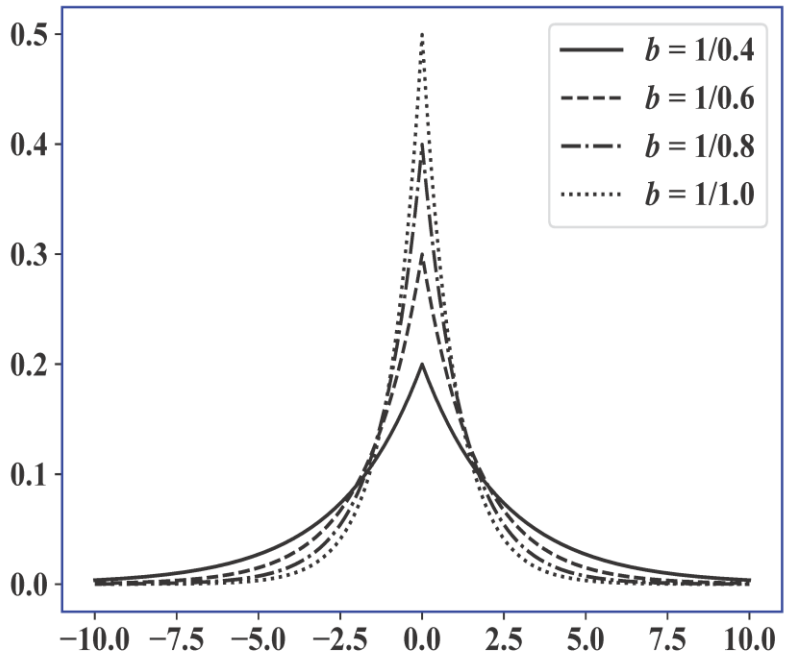
对于任意的数据集D和函数$f: D \to R^d$，其全局敏感度为Δf，若随机算法M的输出结果满足

$$M(D) = f(D) + Lap(\frac{\Delta f}{\varepsilon})$$

则算法M满足$(\varepsilon, 0)$ −差分隐私保护，其中$Lap(\frac{\Delta f}{\varepsilon})$为添加的随机噪声，服从尺度参数为$b = \frac{\Delta f}{\varepsilon}$的拉普拉斯分布

服从拉普拉斯分布的随机噪声

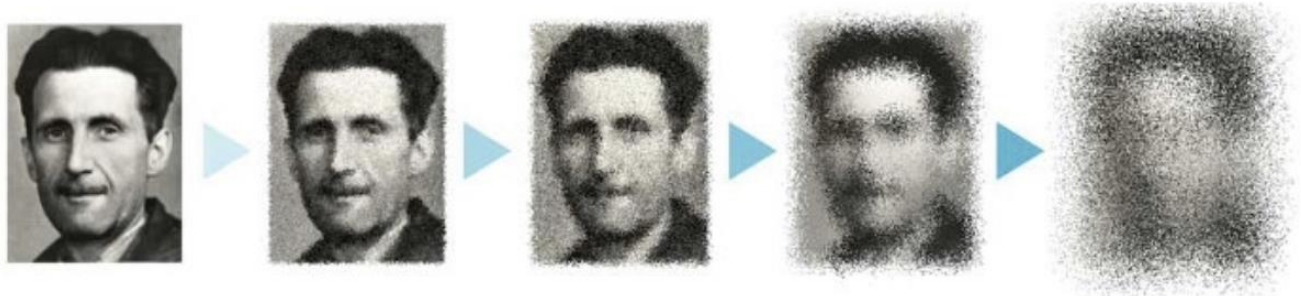
$Lap(\frac{\Delta f}{\varepsilon})$ 为服从尺度参数为$b = \frac{\Delta f}{\varepsilon}$的拉普拉斯分布的随机噪声



不同尺度参数下的拉普拉斯分布图像

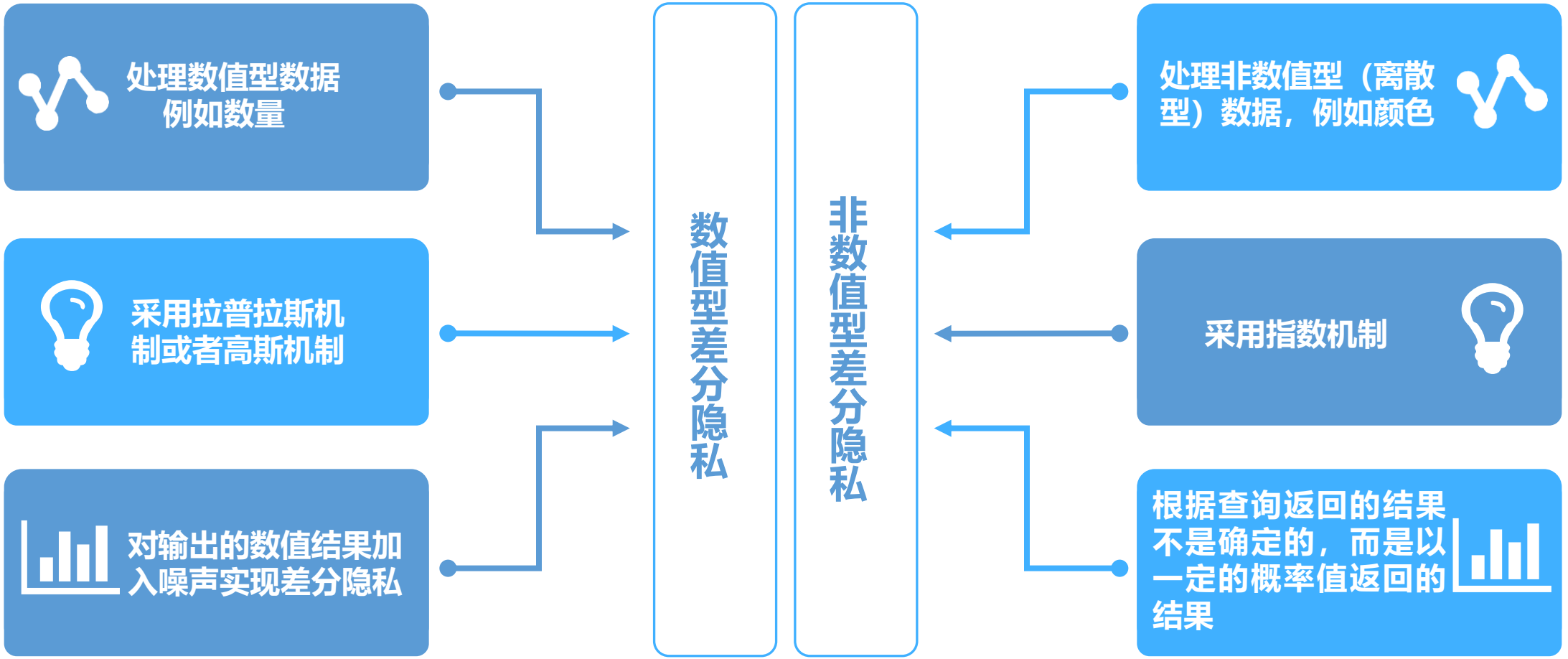
设全局敏感度Δf =1

- 噪声量与Δf成正比，与ε成反比
- 随着ε的减小，对输出的混淆程度增强（使真实值变为一个和真实值具有较大差别的值的概率越大），保护程度越高
- Δf越大，加入的噪声越大，保护程度越高，但是当Δf较大时，往往会对数据提供过度的保护



低隐私保护 → 高隐私保护

非数值型差分隐私：指数机制



指数机制的定义

在数据集D上进行查询时，对于非数值型数据，我们将可能得到的查询结果的集合称为输出域$Range$，域中的每一个值为实体对象r，以查询**得病人数最多的疾病**为例，当D中有四种疾病时，输出域为四种疾病的集合，每一种疾病是一个实体对象

函数$q(D,r)$为r的可用性函数，用来评估输出的r的优劣程度，以查询**得病人数最多的疾病**为例，可用性函数为计算某种疾病的得病人数

设随机算法M输入为数据集D，输出为一实体对象$r \in Range$，$q(D,r)$为可用性函数，Δq为函数$q(D,r)$的敏感度，若算法M以正比于$exp\left\{\frac{\varepsilon q(D,r)}{2\Delta q}\right\}$的概率从$Range$中选择并输出$r$，那么算法$M$提供$\varepsilon$-差分隐私保护

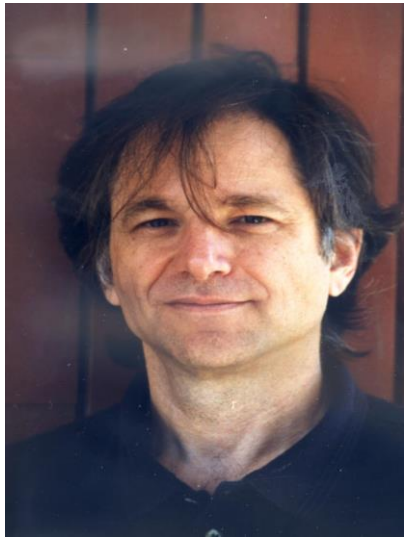
$\Delta q = \max_{D,D'} \|q(D,r) - q(D',r)\|_1$，当可用性函数为计算某种疾病的得病人数时，由于一条记录的有无造成的可用性函数输出的最大变化值为1，因此其敏感度为1

为每个r计算$exp\left\{\frac{\varepsilon q(D,r)}{2\Delta q}\right\}$，对所有结果进行归一化，由此确定每个r输出的概率值

第4节 同态加密

- 同态加密基础
- 半同态加密
- 全同态加密

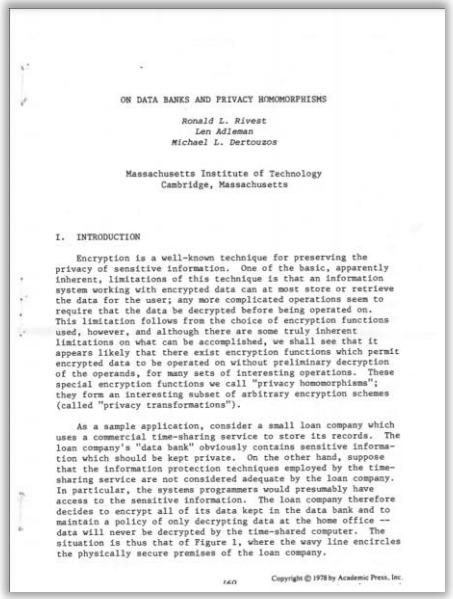
同态加密的提出


- 1978年，Ronald L. Rivest，Leonard Adleman 和 Michael L. Dertouzos 以银行为应用背景提出了同态加密（Homomorphic Cryptosystem, HC）的概念
- Ronald L. Rivest, Leonard Adleman分别是RSA算法（1977年提出）提出者中的R和A
- RSA算法可以实现乘法同态

A way to delegate processing of your data, without giving way access to it.

— Craig Gentry（第一个全同态加密构造者）



ON DATA BANKS AND PRIVACY HOMOMORPHISMS

Ronald L. Rivest
Len Adleman
Michael L. Dertouzos

Massachusetts Institute of Technology
Cambridge, Massachusetts

I. INTRODUCTION

Encryption is a well-known technique for preserving the privacy of sensitive information. One of the basic, apparently inherent, limitations of this technique is that an information system working with encrypted data can at most store or retrieve the data for the user; any more complicated operations seem to require that the data be decrypted before being operated on. This limitation follows from the choice of encryption functions used, however, and although there are some truly inherent limitations on what can be accomplished, we shall see that it appears likely that there exist encryption functions which permit encrypted data to be operated on without preliminary decryption of the operands, for many sets of interesting operations. These special encryption functions we call "privacy homomorphisms"; they form an interesting subset of arbitrary encryption schemes (called "privacy transformations").

As a sample application, consider a small loan company which uses a commercial time-sharing service to store its records. The loan company's "data bank" obviously contains sensitive information which should be kept private. On the other hand, suppose that the information protection techniques employed by the time-sharing service are not considered adequate by the loan company. In particular, the systems programmers would presumably have access to the sensitive information. The loan company therefore decides to encrypt all of its data kept in the data bank and to maintain a policy of only decrypting data at the home office -- data will never be decrypted by the time-shared computer. The situation is thus that of Figure 1, where the wavy line encircles the physically secure premises of the loan company.

思想：对密文直接进行操作，且计算结果的解密值与对应明文的计算结果相同

同态加密算法

Encrypt算法
用公钥将明文加密为密文

Decrypt算法
用私钥将密文解密为明文

KeyGen算法
由安全参数计算一对公私钥

Evaluate算法
在密文上进行运算

同态加密算法的四个组成部分

同态加密的发展

1978年Ronald Rivest等人提出同态加密的概念

国内外经过40年的研究，不断提出新的加密方案，并逐渐应用于实际中

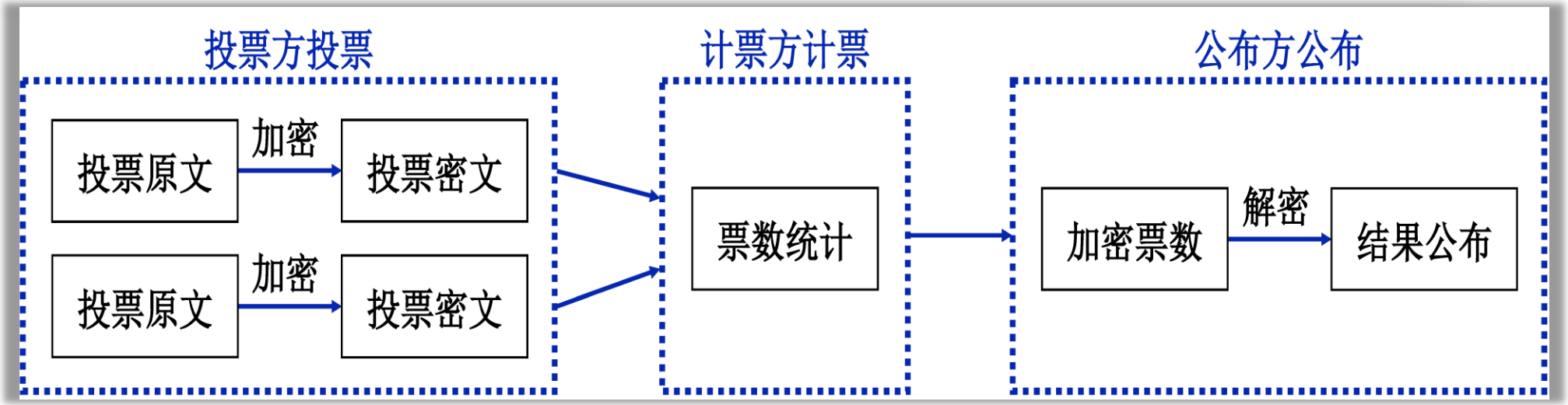
2009年，Gentry构造出了第一个真正的全同态加密体制

半同态加密 Partially Homomorphic Encryption (PHE)	浅同态加密 Somewhat Homomorphic Encryption (SWHE)	全同态加密 Fully Homomorphic Encryption (FHE)
仅支持加法同态（或乘法同态）的加密体制	同时满足加同态和乘同态性质，只能进行有限次的加和乘运算	同时满足加同态和乘同态性质，可以进行任意多次加和乘运算

同态加密的应用

- 与传统的投票方式相比，**电子投票**计票快捷准确，节省人力和开支，投票时具有便利性，而设计安全的电子选举系统是全同态加密的一个典型应用
- 基于同态加密设计的电子选举，统计方可以在不知道投票者投票内容的前提下，对投票结果进行统计，既保证了投票者的隐私安全，又能够保证投票结果的公证

电子投票的简化流程



同态加密的优势与挑战

优势

- 对多个密文进行计算后再解密，降低了计算代价
- 可实现无密钥方对密文的计算，密文计算无须经过密钥方，降低了通信代价
- 可实现让解密方只能获知最后的结果，而无法获得每一个密文的消息，可以保证信息的安全性（匿名电子投票）

挑战

- 只能实现单比特加密，如何高效地实现全同态加密有待进一步研究
- 大多基于未论证的困难问题，寻找可论证的困难问题依然是个难题
- 需要额外的消除噪音算法，如何设计一个具有自然同态性的全同态加密方案依然是一个问题

虽然同态加密正在逐步向实用化靠近，但是其安全性和实用性方面的研究还有很长的路要走

半同态加密

在一个加密方案中，用a、b表示明文，Enc表示加密算法，Dec表示解密算法，$\oplus$表示在明文域上的运算，$\otimes$表示在密文域上的运算，如果该加密方案中的加密算法和解密算法满足

$$Dec\big(Enc(a) \otimes Enc(b)\big) = a \oplus b$$

当$\oplus$表示乘法时，称该加密为乘法同态加密

当$\oplus$表示加法时，称该加密为加法同态加密

典型的乘法同态加密算法

- 1977年提出的RSA公钥加密算法
- 1985年提出的ElGamal公钥加密算法

典型的加法同态加密算法

- 1999年提出的Paillier公钥加密算法，是最常用且最具实用性的加法同态加密算法

Paillier加法同态加密

- 1999年由Pascal Paillier提出的，其安全性基于判定合数剩余类的问题
- 是第一种且应用最为广泛具有加法同态性的加密算法
- 已广泛应用在加密信号处理或第三方数据处理领域

判定合数剩余问题

令$N = pq$，其中，p和q为安全素数，任给定$z \in \mathrm{Z}_{N^2}^*$，判定z为N次剩余还是非N次剩余

N次剩余定义：给定$N = pq$，其中，p和q为安全素数，给定$z \in \mathrm{Z}_{N^2}^*$，若存在某个$y \in \mathrm{Z}_{N^2}^*$，使得$\mathrm{z} = y^N (mod N^2)$成立，则称$z$为（模$N^2$的）$N$次剩余；否则，则称$z$为（模$N^2$的）非$N$次剩余

全同态加密

全同态加密指同时满足加同态和乘同态性质，可以进行任意多次加和乘运算的加密函数，用数学公式来表达，即满足

$$Dec\left(f\left(Enc(m_1), Enc(m_2), \cdots, Enc(m_k)\right)\right) = f(m_1, m_2, \cdots, m_k)$$

或者写为$f\left(Enc(m_1), Enc(m_2), \cdots, Enc(m_k)\right) = En(f(m_1, m_2, \cdots, m_k))$

如果f是任意函数，称为全同态加密

- 鉴于全同态加密的强大功能，一经提出便成为密码界的公开问题，被誉为“密码学圣杯”
- 直到2009年，Gentry才基于理想格构造出了首个全同态加密方案，虽然在实际应用中效率不高，但这一里程碑事件激起了全同态加密研究的热潮

全同态加密与半同态加密

与半同态加密相比，全同态加密

- 加密算法功能更强大
- 具有较高的计算复杂度，加密算法设计更复杂
- 整体性能远不及半同态加密算法

相关研究报告显示，在一次使用全同态加密开源库为敏感医疗数据构建密文线性回归模型的尝试中，1M的明文数据编码后可能膨胀至约10G密文数据

第5节 安全多方计算

- 安全多方计算基础
- 百万富翁协议

安全多方计算的提出

2000年图灵奖获得者

- 安全多方计算（**Multi-Party Computation, MPC**）起源于**姚期智教授**在1982年提出的百万富翁问题
- **解决了一组互不信任的参与方之间保护隐私的协同计算问题**
  - 当有两方或者多方参与者决定互相合作并且各自需要提供自己的隐私或者秘密数据时，任意一方都不愿意让其他一方知晓自己提供的信息

安全多方计算形式化描述

假定有m个参与方$P_1, P_2, \cdots, P_m$,他们拥有各自的数据集$d_1, d_2, \cdots, d_m$,在无可信第三方的情况下如何安全地计算一个约定函数$y = (d_1, d_2, \cdots, d_m)$，同时要求每个参与方除了计算结果外不能得到其他参与方的任何输入信息

安全多方计算的特征

输入独立性	需保证各方能独立输入数据，计算时不泄露本地数据
计算正确性	需保证计算结束后各方能够得到正确的计算结果
去中心化性	各参与方地位平等，提供了去中心化的计算模式

安全多方计算的威胁模型

现实世界中不存在

半诚实模型
在诚实模型基础上保留所有收集到的信息，推断其他参与者的秘密信息

多数情况符合

诚实模型
参与者不提供虚假数据，不会泄露、窃听数据，不会终止协议，完全按照协议执行

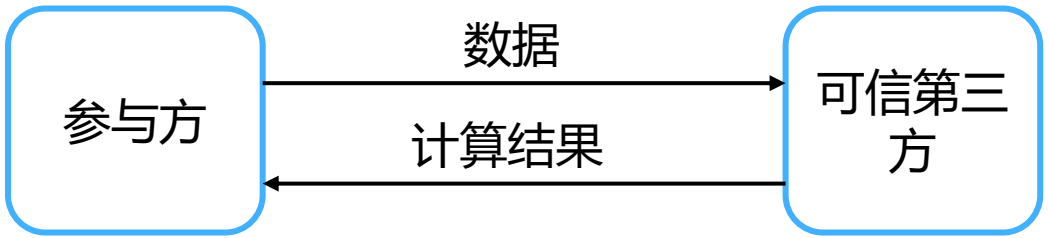
恶意模型
无视协议要求，可能提供虚假数据、泄露数据、窃听甚至终止协议

三种安全模型(根据参与方的可信程度)

计算模型

安全多方计算的计算模型主要有基于“可信第三方”的计算模型、交互计算模型和外包计算模型

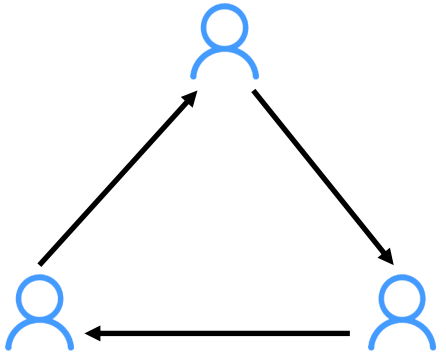
基于“可信第三方”的计算模型



- 参与者得到计算结果，可信第三方得到参与者的输入信息和计算结果，信息的保密性由可信第三方来保证

很难找到完全可信的第三方，不能满足实际中对安全性的需要，目前研究中很少使用

交互计算模型



- 参与者约定协议通过交互计算共同完成函数运算
- 按照协议步骤执行计算，按协议的要求将中间结果发送给其他参与者同时接收其他参与者计算的中间结果，信息的保密性由协议的安全性来保证

使用最为广泛

安全多方计算的应用

• 门限签名	将私钥拆分为多个秘密分片，当不少于门限值的的秘密分片持有者共同协作时才能生成有效的签名
• 电子拍卖	需计算出所有参与者输入的最大值或最小值，安全多方计算理论的提出，使得网上拍卖成为现实
• 联合数据查询	不同数据库资源共享时，多个数据库可以看成多个用户联合起来进行数据查询，可使用安全多方计算保护各数据库的私有信息或知识版权

- 安全多方计算牵涉到密码学的各个分支，有着广阔的应用领域
- 其优势为比较安全和准确，但涉及的加密技术开销、通信开销也很大
- 目前的研究主要集中于降低计算开销、优化分布式计算协议

百万富翁问题

- 姚期智教授1982年提出的百万富翁问题是**第一个安全双方计算问题**
- 百万富翁Alice和Bob想相互比较一下谁更富有，但是他们都不想让对方知道自己拥有多少财富，如何不借助第三方比较两个人的财富多少?

在不知道对方财富的情况下比较谁更富有

百万富翁协议

输入：Alice和Bob的财富值 i，j，Alice拥有公私钥

输出：$i \geq j$ 或 $i < j$

- 设Alice拥有的财富为i,Bob拥有的财富为j，单位均为百万，其中$1 \ll i, j \ll 10$
- 令M为N 个bit表示的非负整数的集合，Q_N是从M映射到M的所有一一映射的集合
- E_a是Alice的公钥，通过从Q_N中随机选择一个元素生成，D_a为私钥

百万富翁协议

Alice

2. Bob将$k-j+1$发送给Alice

Bob

1. Bob选择一个$N\ bit$的随机整数x，并私下计算$k=E_a(x)$

3. Alice计算$Y_u = D_a(k-j+u)$的值，其中$u=1,2,\cdots,10$

Bob知道解密后的序列中第j个数为随机数x，因为当$u=j$时，$D_a(k-j+u)=D_a(k)=\mathrm{x}$，但是由于没有私钥，因此不知道其他解密值

百万富翁协议

Alice

5. Alice对序列Z_u进行处理，将p和序列$z_1, z_2, \cdots, z_i, z_{i+1}+1, \cdots, z_{10}+1$发送给Bob

Bob

4. Alice生成一个$\frac{N}{2}$ bit的随机

$Y_u(mod\ p)$，该随机素

第j个数字,如果该数字等于

$< j$

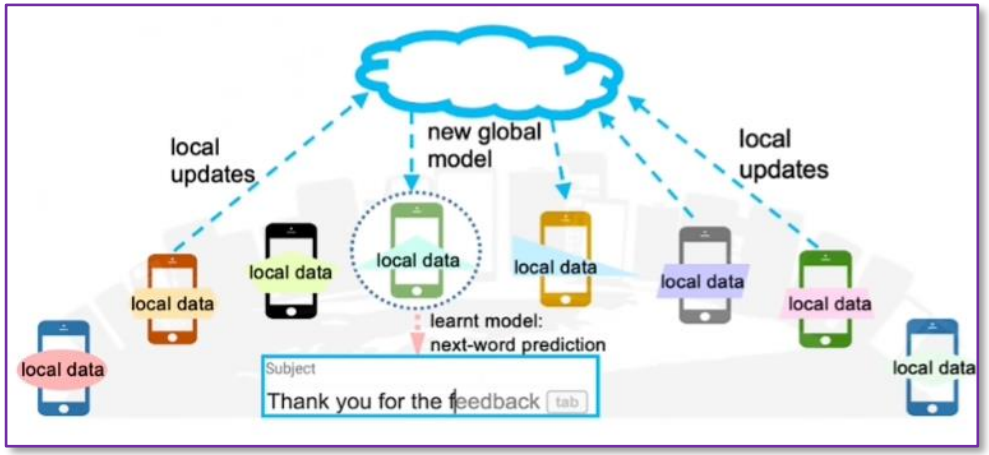
- Alice从序列中的第$i+1$个数开始对数值做加一处理，当$i \geq j$时，第j个数值不变，当$i < j$时，第j个数值被修改，因此Bob可通过判断第j个数值是否被修改来比较i和j的大小
- 由于Bob没有私钥，不知道Y_u的解密值，因此无法知道除第j个数以外的其他数是否被修改，也就无法判断i的大小

第6节 联邦学习

- 联邦学习的基础
- 纵向和横向联邦

联邦学习的基本概念

联邦学习这一概念最早于2017年4月提出，谷歌科学家brendan mcmahan和Daniel Ramage在GoogleAI上发布名为*Federated Learning: Collaborative Machine Learning without Centralized Training Data*的博文，为的是解决如何在用户移动终端设备上进行模型训练的问题



案例：手机用户打字单词预测

brendan mcmahan

BLOG ›

Federated Learning: Collaborative Machine Learning without Centralized Training Data

THURSDAY, APRIL 06, 2017

Posted by Brendan McMahan and Daniel Ramage, Research Scientists

最初的想法：用户手机平板下载服务器上的模型，在本地数据上改进模型，然后将更新发送到云端，云端汇总所有更新对模型进行优化。

联邦学习的定义

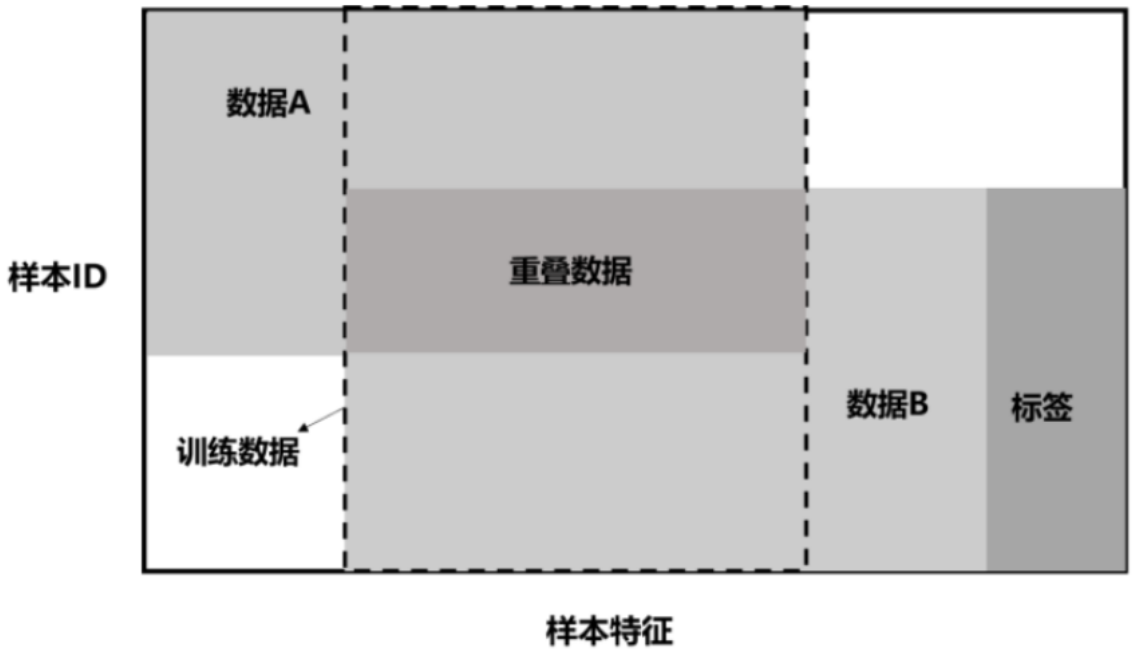
- **定义：**联邦学习是一种分布式机器学习方法。在每轮迭代中，联邦学习允许多个客户端节点独立地根据本地数据更新模型参数，随后将本地模型更新通过加密方式发送到中央服务器。中央服务器使用聚合算法将这些更新用来改进全局模型，然后再将改进后的模型发送回各个客户端。
- **主要优势：**
  - 本地训练，保障用户数据隐私；
  - 解决数据孤岛问题，促进数据的可用性；
  - 扩展性、灵活性强，可纳入新节点而无须重新训练；
  - 减少数据中心需求，降低成本；
  - ……

联邦学习的分类

联邦学习可以分为三类：横向联邦学习(Horizontal Federated Learning)，纵向联邦学习(Vertical Federated Learning)和联邦迁移学习(Federated Transfer Learning)

横向联邦学习

横向联邦学习适用于各客户端节点样本重叠较少而特征重叠较多的场景。一般情况下，会取出参与方数据中特征重叠的部分进行训练。横向联邦学习是最经典的联邦学习方式。



ID	X1	X2	X3	x4	x5
D1	2	3	5	-	-
D2	11	23	21	-	-

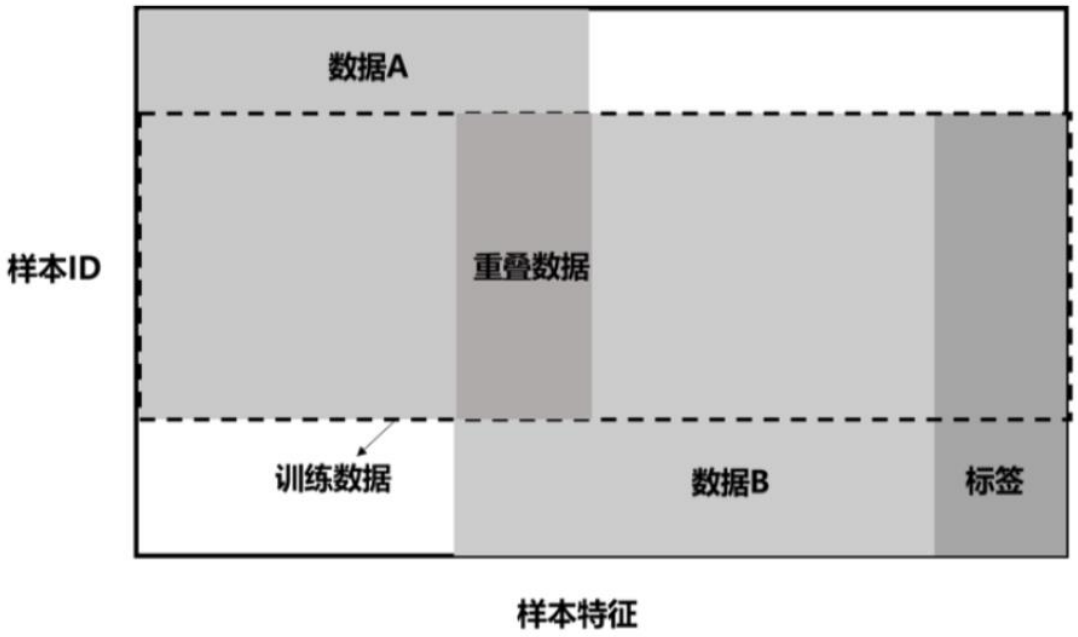
节点A的数据

ID	X1	X2	X3	X4	X5
D3	7	9	5	1.3	2.7
D5	-	-	2	1.2	2.3
D6	-	-	-	-	3.9

节点B的数据

纵向联邦学习

纵向联邦学习适用于各客户端节点特征重叠较少而样本重叠较多的场景。一般情况下，会取出参与方数据中样本重叠的部分进行训练，由于数据的保密性，一般会使用安全多方计算等方式进行**样本对齐**。



ID	X1	X2	X3
D1	2	3	5
D2	11	23	21
D3	6	3	1
D4	7	9	3

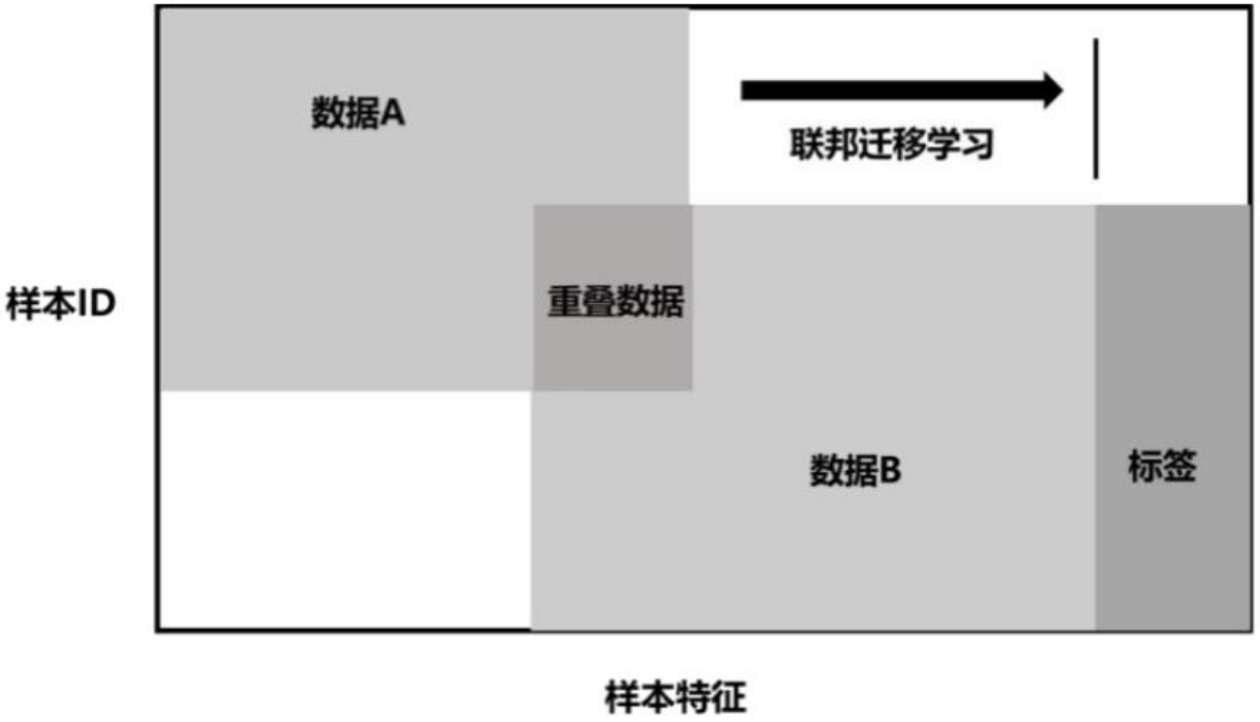
节点A的数据

ID	X4	X5
D1	1.3	2.7
D2	1.2	2.3
D3	6.0	3.9
D5	8.9	4.0

节点B的数据

联邦迁移学习

各客户端节点特征和样本重叠均较少，那我们可以使用联邦迁移学习。在这种情境下，不仅要在不同参与者之间共享知识，而且要将一个域的知识“搬运”到另一个域。在这种情况下，我们可以引入迁移学习来解决知识域的迁移问题。



基本特点

- 源模型训练：在一个数据丰富的源域上训练源模型
- 模型分享：将源模型共享给其他目标域节点，可以引入知识蒸馏等方法传递模型信息
- 在此基础上，执行标准的联邦学习流程

第7节 总结和展望

总结

概述数据隐私保护技术，讲解不同分类下的隐私保护技术思想与基础知识

第一节
隐私保护技术初探

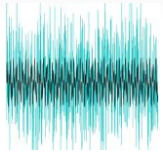
- 网络空间安全中的隐私
- 隐私泄露的危害
- 隐私保护技术介绍

隐私泄露事件造成的危害让人们看到了隐私保护的重要性

第二节
匿名化

- 匿名化隐私保护模型
- 数据匿名化方法

如何安全地发布数据供其他机构研究



第三节
差分隐私(DP)

- 差分隐私基础
- 数值型DP
- 非数值型DP

如何保护统计信息中的个体隐私

第四节
同态加密

- 同态加密基础
- 半同态加密
- 全同态加密

如何安全地将数据委托给数据计算方

第五节
安全多方计算

- 安全多方计算基础
- 百万富翁协议

如何帮助互不信任的参与方进行协同计算

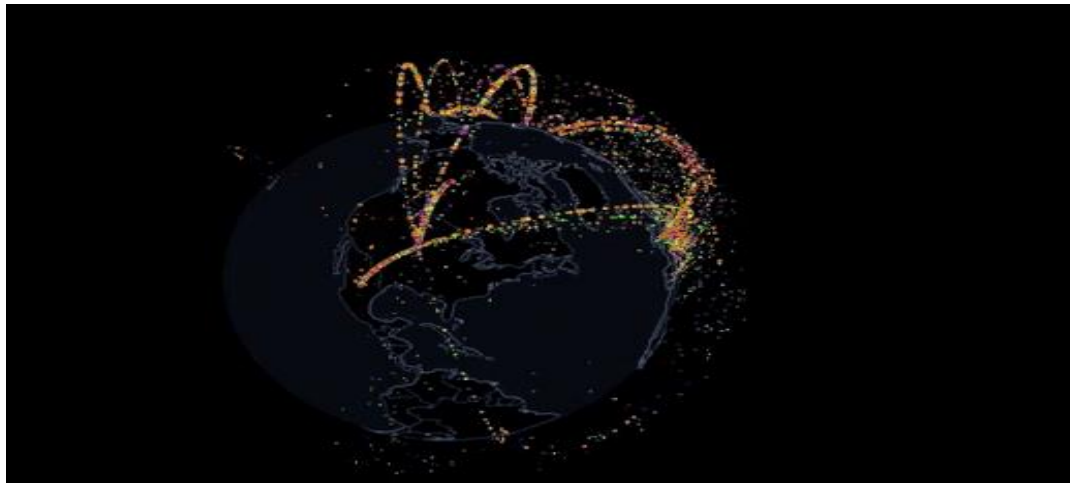
第六节
联邦学习

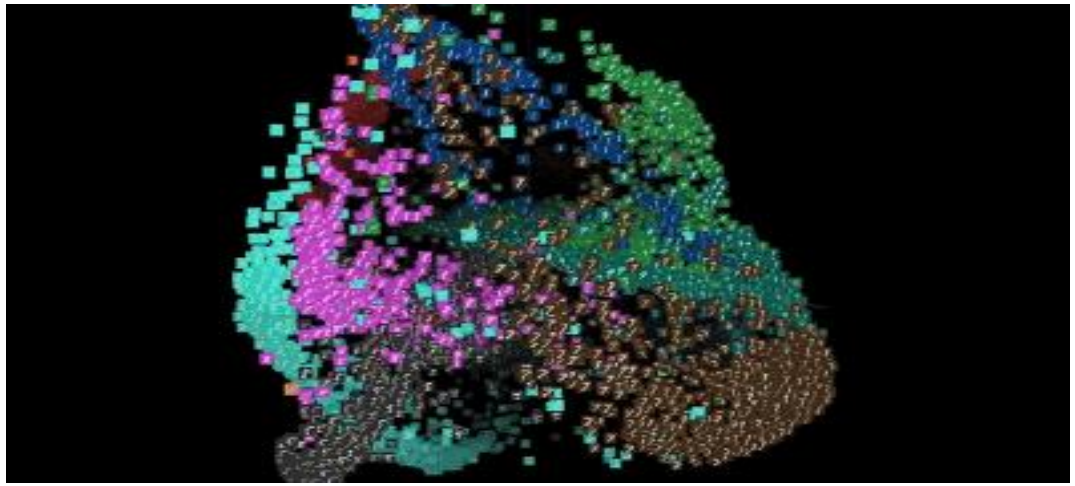
- 联邦学习的基础
- 横向联邦
- 纵向联邦

如何实现在不共享数据下的模型协同训练

展望

研究如何更好的保护动态数据、高维数据中的隐私





- 数据隐私保护技术在静态数据的隐私保护方面比较成熟，而对于**如何保护动态数据中的隐私**还存在较多问题
- 实际中的高维数据越来越多，使用传统的隐私保护模型处理高维数据会导致信息损失过多，**如何保护高维数据的隐私**也是未来需要研究的问题