差异化的私有数据发布和分析：一项调查

差异化隐私是近几十年来被广泛探索的一种基本和普遍的隐私模式。本调查报告对两个研究方向进行了全面和结构化的概述：差异化隐私数据发布和差异化隐私数据分析。我们从查询类型、最大查询次数、效率和准确性等方面，比较了给定各种输入数据的差异化隐私数据发布的不同发布机制。我们确定了两个用于差异化私有数据分析的基本框架，并列出了每个框架内使用的典型算法。根据输出精度和效率对结果进行了比较和讨论。此外，我们为未来的研究和可能的应用提出了几个可能的方向。

引言

在过去的20年里，企业、组织和政府收集的数字信息已经形成了大量的数据集，而且这种数据收集的速度在过去几年里急剧增加。通常情况下，数据收集者，也被称为策展人，负责发布数据供进一步分析[1]。然而，大多数收集的数据集包含私人或敏感信息。即使策展人可以应用一些简单的匿名化技术，敏感的个人信息仍然有很大的可能性被披露[2]。因此，隐私保护已经成为一个亟待解决的问题。

图1显示了一个受信任的策展人如何保存包括个人信息的原始数据集。馆长将汇总的信息提供给公众用户，他们可以使用这些信息进行进一步的调查。我们根据其发布的目的，将这个过程分为数据发布和数据分析。数据发布的目的是向公众分享数据集或一些查询结果。在一些文献中，这种情况被称为数据共享或数据发布。另一种情况是，策展人直接向公众提供数据模型，通常被定义为数据分析。共享的模型可能与特定的算法有关，如数据挖掘或机器学习算法。研究团体已经提出了各种方法来保护这两种情况下的个人隐私。这些方法及其隐私标准被定义为一个隐私模型。如图1所示，隐私模型位于受信任的策展人和不受信任的公共用户之间。差异性隐私是这样一个新的和有前途的隐私模型。它确保对手对数据集中的任何个人造成伤害的能力基本相同。

独立于任何个人是否选择加入或退出数据集[3]。与之前的隐私模型相比，差分隐私可以成功抵御大部分的隐私攻击，并提供可证明的隐私保证。

人们对差分隐私的兴趣非常高，其核心概念跨越了一系列研究领域，从隐私社区到数据科学领域，如机器学习、数据挖掘、统计和学习理论。在一些应用领域也进行了很多工作，包括社交网络、位置隐私和推荐系统。图2显示了本调查中要进一步讨论的其中一些关键因素。

关于差异化隐私的最初工作是由Dwork等人[3]在2006年开创的。在过去十年中，已经完成了几项关于差异化隐私的调查。

1. Dwork等人的第一份调查报告[4]回顾了差异化隐私的定义及其两个主要机制，旨在展示如何在数据发布中应用
   这些技术在数据发布中的应用。

2）报告[5]利用了数据出版在统计分析方面遇到的前瞻性解决方案时出现的困难。它指出了数据分析中的几个研究问题，这些问题在当时还没有得到充分的调查。
图2：与差异化隐私相关的关键因素

2. 在一篇评论[6]中，Dwork等人概述了主要的激励方案，以及对未来研究方向的总结。
3. Sarwate等人[7]专注于为连续数据保留隐私，以解决信号处理中的问题。
4. Dwork等人的书[8]为任何想了解微分隐私理论的人提供了一个可获得的起点。
这些调查和书的重点是微分隐私的概念和理论；然而，数学理论不容易直接落实到应用中。然而，经过十多年的理论发展，这一领域出现了大量的新技术和应用。我们认为，现在是总结新技术和解决理论与应用之间差距的好时机。
在这里，我们试图找到一种更清晰的方式，为数据挖掘研究界介绍差异化隐私的概念和实践方面。

- 我们避免对相关的差分隐私算法进行详细的理论分析，而是将更多的注意力放在它的实践方面，这可能有利于现实世界中的应用。
- 我们尽量避免重复许多已经在上述引用的调查中广泛分析过的参考文献。
- 尽管差分隐私涵盖了多个研究方向，但我们的观察仅限于数据发布和数据分析场景，这也是研究界最流行的场景。
表1定义了这两个研究方向在调查中的范围。数据发布的机制设计通常独立于其发布目标，因为发布的目标是发布查询答案或数据集供进一步使用，因此，对策展人来说是未知的。数据分析的机制设计的目的是在分析过程中保护隐私。馆长已经知道分析算法的细节，所以该机制与分析算法相关。
2 差异化隐私的预演
2.1 符号
我们考虑一个大小为|X|的有限数据宇宙X。让r代表一个有d个属性的记录；一个数据集D是一个从宇宙X中取样的n个记录的无序集合。如果两个数据集和有一个记录不同，则定义为相邻的数据集。一个查询f是一个将数据集D映射到抽象范围R的函数：f：一组查询表示为F，我们用符号m表示F中查询的数量。
差分隐私的目的是掩盖相邻数据集之间查询f的差异。查询f的结果的最大差异被定义为敏感度Δf。差别隐私一般由机制M实现，它是一种访问数据库并实现某些功能的随机化算法。随机化的输出在符号上用圆周率表示。对于D上的f，表2总结了例子中使用的符号，f(D)表示查询以下章节的随机化答案。

定义1（-差分隐私）。[9] 一个随机机制M对每一组输出Ω和D的任何相邻的数据集都给出-差分隐私，并且满足。

```
如果δ=0，随机机制M给出了
```

-差分隐私提供了在某些低概率事件中违反严格的差分隐私的自由。差分隐私通常被称为纯差分隐私，而δ>0的(,δ)-差分隐私被称为近似差分隐私[10]。
2.3 隐私预算构成
在定义1中，参数指的是隐私预算[9]，它控制着机制M实现的隐私保障水平，越小代表隐私水平越强。两个隐私预算构成定理被广泛使用：顺序构成[11]和平行构成[12]。
定理1（平行构成）。假设我们有一组隐私机制M={M1,...,Mm}。如果每个Mi在整个数据集的一个不相连的子集上提供i-differential隐私保证，M将提供

-差异性隐私。
定理2（顺序组成）。假设一组隐私机制M={M1,...,Mm}在一个数据集上依次执行，并且每个Mi都提供差异性隐私保证，那么M将提供差异性隐私。
2.4 敏感度
敏感性是决定一个机制中的特定查询需要多少扰动的参数。
定义2（灵敏度）。[6] 对于一个查询f : D→R，以及相邻的数据集和 ，f的灵敏度定义为
(2)
敏感性Δf只与查询f的类型有关，它考虑的是相邻数据集上查询结果的最大差异，并指出差异应被隐藏的程度。
2.5 差异性隐私机制原则
任何符合定义1的机制都可以被认为是差异性隐私。目前，有两种基本机制被广泛用于保证差异性隐私：拉普拉斯机制[13]和指数机制[11]。
2.5.1 拉普拉斯机制
拉普拉斯机制在真实答案中加入了独立的噪声。我们用Lap(b)来表示从拉普拉斯分布中采样的噪声，其比例为b。
定义3（拉普拉斯机制）。[6] 对于一个函数f : D → R的数据集D，公式3中的机制M提供了-差分隐私。
(3)
2.5.2 指数机制
对于非数字查询，差分隐私使用指数机制来随机化结果，这与评估输出φ质量的评分函数q(D,φ)相匹配。定义一个分数函数是与应用相关的，不同的应用会导致不同的分数函数。
定义4（指数机制）。[11] 让q(D,φ)是数据集D的评分函数，衡量输出φ∈Φ的质量，Δq代表φ的敏感性。 指数机制M满足-差分隐私，如果
返回.(4)
2.6 差分隐私的效用测量
当隐私水平被固定为.时，在数据发布和分析中都会用到几种效用测量。

- 噪声大小测量：最简单的方法是标定在查询结果中加入多少噪声。噪声量越小表示效用越高。这种效用测量在数据发布中被广泛使用。
- 误差测量：效用可以通过非私人输出和私人输出之间的差异来评估。误差测量通常用一个带有精度参数的约束来表示[14]。
  定义5（(α,β)-有用）。一个机制M是(α,β)有用的，如果
  (5)
  其中，α是限定误差的精度参数。
  对于不同的发布场景，误差测量可以用不同的方式来解释。对于合成数据集的发布，公式5可以被解释为。
  (6)
  对于数据分析，效用测量通常取决于分析算法。假设算法用M表示，私有算法用M表示，式5可以解释为
  ((7)
  公式7在数据分析中有几种实现方式，如风险边界和样本复杂度，这将在第4节介绍。

差别化的私人数据发布
差异化隐私数据发布的目的是在不透露任何个人记录的情况下向公众输出汇总信息。这个问题可以表述如下：如果一个馆长有一个数据集D，并收到一个查询集F={f1,...,fm}，他们需要在差异性隐私的约束下回答每个查询fi∈F。
年龄 患有糖尿病的变量患者的数量
60-79 41 x1
40-59 32 x2
20-39 8 x3 0-19 1 x4
表3：医疗表 表4。频率表
在这个发布场景中涉及到两种设置，即交互式和非交互式。在交互式设置中，在前一个查询fi-1的答案被发布之前，不能发出查询fi。在非交互式设置中，所有的查询都是一次性给馆长的，馆长可以在完全了解查询集的情况下提供答案。
表3给出了一个例子来说明这两种设置之间的区别。对策展人的查询可能呈现如下。

- f1：有多少病人在40-79岁时患有糖尿病?
- f2：有多少病人在40-59岁时有糖尿病?
  假设每个查询的隐私预算是固定的。在交互式设置中，策展人将首先得到f1，然后统计40-79岁之间的糖尿病患者人数，并在这个数字上添加独立的拉普拉斯噪声，其灵敏度等于。 然后将f2提交给策展人时，f2的回答灵敏度等于2，因为改变表中的一个人可能改变两个查询的结果。添加到查询集的总噪音为 。
  在非交互式设置中，两个查询都是同时提交给馆长的。两次查询测得的敏感度都是2，添加到查询集的总噪音是 ，比互动设置大。
  查询之间的关联性导致了更高的灵敏度。因此，非交互式设置通常比交互式设置产生更多的噪声。
  上面的例子展示了两种设置之间的差异，并表明当查询之间相互关联时，噪音量会急剧增加。此外，对于一个大小为n的数据集，拉普拉斯机制最多只能回答n个查询的次线性，达到一定的精度[15]。
  这些弱点使得拉普拉斯机制在需要回答大量查询的场景中不实用。需要新的机制。表5总结了差异化私有数据发布的问题特征，其中机制设计的重点是查询的数量、输出的准确性和计算的效率。
  3.1 发布机制
  我们将现有的机制分为几种类型：转换、数据集分区、查询分离和迭代。表3再次用来显示关键的想法。
- 转化。转化机制将原始数据集映射到一个新的结构中，以调整敏感性或噪音水平。在上面的例子中，原始的TABLE 5: 差别化的私有数据发布问题的特点
  差别化的私有数据发布
  输入数据交易的性质，柱状图，图，流
  输出数据的性质 查询结果、合成数据集
  发布环境交互式、非交互式
  发布机制 Laplace/exponential、查询分离、转换、迭代、数据集分割
  挑战查询数量、准确性、计算效率

如表4所示，数据集可以被转移到频率数据集上。
在新的结构中，f2可以直接由第二行回答，灵敏度等于1。f1可以通过结合和表4第一行的噪声结果来回答。由于60-79岁的糖尿病患者的结果被独立地列在f2中，f1的灵敏度仍然等于1，两个查询的总噪声将是 ，这比非交互式拉普拉斯机制低。新的结构用于分解查询之间的相关性，所以灵敏度也可以降低。挑战在于找到一个新的结构。

- 数据集划分：原始数据集被划分为几个部分，以减少噪音。在上面的例子中，假设我们需要用表4来回答f1，则需要添加两次噪音：一次是在第一行，另一次是在第二行。然而，如果我们以另一种方式对数据集进行分区，例如，将年龄范围重新安排为40-79岁，那么总的噪音将减少到.这里的挑战是为多个查询设计一个分区策略。
- 查询分离：查询分离假设一个查询集可以被分离成几组，并且一些查询可以在重用噪声的意义上得到回答。在上面的例子中，如果f2已经被回答了，那么f1可以通过加倍f2的答案来近似回答，因为年龄范围加倍了。查询分离是用来打破对查询数量的限制。
- 迭代：迭代是一种机制，它递归地更新数据集，以近似一组查询的噪声答案。例如，我们可以手动定义一个初始数据集D0，其中表4中不同年龄段的糖尿病患者数量相等，然后对D0执行f1，并将噪声结果与f1(D0)进行比较。如果两个答案之间的距离小于预先定义的阈值，f1(D0)就可以公布，并将用于下一轮。否则，f1(D)将被公布，D0将被一个特定的策略更新为D1。由于发布f1(D0)不会消耗任何隐私预算，迭代机制可以实现更高的效用，并且可以比拉普拉斯机制回答更多的查询。挑战在于设计一个更新策略和设置相关参数，如阈值。
  表6对各种发布机制进行了比较。在下面的小节中，我们将介绍这些机制在交互式和非交互式设置中是如何工作的。
  3.2 交互式发布
  交互式设置对输入数据的各个方面进行操作，包括交易、柱状图、流和图数据集。在下面的小节中，我们讨论涉及这些类型的输入数据的发布场景。
  3.2.1 交易数据发布
  D的最流行的表示方法是交易数据集，其中每条记录代表一个具有d个属性的个体。
  3.2.1.1 查询分离。查询分离的目标是为给定类型的查询设计一个分离策略，以减少噪音。Roth[16]提出了中位数机制，并发现在任何一组m个查询中，有O(logmlog|X|)个查询可以确定所有其他查询的答案。基于这一观察，所有的查询都被分成了难查询和易查询。硬查询可以直接由拉普拉斯机制来回答，而简单查询则由硬查询结果的中值来回答。因此，简单查询不会消耗任何隐私预算。通过分离查询，中位数机制可以以可接受的精度回答指数级的更多查询；但是，它的效率很低，而且有一个与数据集大小n相对应的指数级时间复杂度。
  3.2.1.2 迭代。Hardt等人[17]提出了私有乘法权重（PMW），它将数据集视为一个直方图，每个bin上都有正的权重。通过更新权重，PMW构建了一个直方图序列来回答一组查询。在对参数的复杂性和准确性进行校准后，该机制能够以大约O((logm)/√n)的采样误差回答每个查询。这意味着采样误差随着被回答的查询数量的增加而呈对数增长，而拉普拉斯机制的误差是线性的，以m为单位增加。此外，PMW可以准确地回答指数级的查询。
  同样，Gupta等人[18]提出了一个通用的迭代框架，称为迭代数据库构造（IDC），通过该框架实现了其他释放机制。在每一轮迭代中，当见证了当前数据集和原始数据集在给定查询中的显著差异时，该机制会在下一次更新中更新当前数据集。IDC是一个

更通用的框架，可以被纳入其他各种机制，包括PMW和中位数。

3.2.1.3 讨论。查询分离和迭代机制可以回答比拉普拉斯机制更多的查询，通常是一个指数级的n。随着查询次数的增加，Laplace机制的误差边界增加了mlogm，而其他机制则将增加的幅度限制在logm，这在搜索整个数据宇宙时是一个巨大的改进。除了拉普拉斯机制之外，其他大多数机制都是低效的，因为它们确实需要遍历整个数据宇宙。因此，在这些机制中，误差界限随着|X|的增长而呈对数增长。

鉴于迭代机制在误差边界方面优于其他机制，许多后续的工作，包括

柱状图发布

以直方图的形式来看待交易数据通常是很方便的。一个直方图有N个仓，一个差异化的隐私机制旨在隐藏每个仓的频率。表4可以被认为是表3的直方图表示，有四个仓。直方图表示法的优点是，限制了对噪声的敏感性[4]。例如，当直方图用于支持范围或计数查询时，添加或删除一条记录最多会影响一个仓。因此，直方图上的范围或计数查询的灵敏度等于1，而且每个仓的噪声增加量也相对较小。

3.2.2.1 拉普拉斯。这是一种直接的机制，在查询覆盖的每个bin的频率上增加拉普拉斯噪声。当一个计数范围的查询只覆盖少量的仓时，这种机制对查询结果保留了很高的效用；但是，如果原始数据集包含多个属性，这些属性和它们相关的数值范围的组合将导致大量的仓的出现。由于有大量的仓，积累了大量的错误，所以查询的答案是没有意义的。

3.2.2.2 数据集的划分。由于bin的数量来自于属性值的划分，减少误差的方法之一是优化划分策略。例如，当查询覆盖了较大数量的仓时，可以将几个仓合并为一个新的仓。拉普拉斯噪声可以减少，因为它只被添加到新合并的仓中一次。当查询所覆盖的仓的数量相对较少时，馆长可以将大仓分成小仓，或者通过估计大仓的频率比例来近似查询结果。然而，将大仓分割成小仓会导致更多的拉普拉斯噪声，而估计大仓的频率比例可能会引入估计误差。因此，优化分割策略以获得分割和合并bin之间的权衡是一个需要解决的挑战。

Xu[20]通过最小化一组查询的平方误差之和（SSE）提供了两种分区策略。这两种策略都是在初始状态下将每个属性设置为一个仓，并对属性值进行分区以创建更多的仓。第一个策略，NoiseFirst，在划分属性值之前向每个bin注入拉普拉斯噪声。另一种策略，StructureFirst，采用指数机制，通过采用SSE作为评分函数来选择属性的最佳分割值。

Qardaji等人[21]以分层的方式对属性值进行划分。他们也专注于范围查询，并将范围分配到一棵树上。树的根部是一个属性或几个属性的全部取值范围。树中的每个节点都与它的子节点的范围的联合有关。几个单位长度的范围被定义为叶子。在树的每个分支上，一个因素控制着查询结果的准确性。在回答范围查询时，用平均平方误差（MSE）来进一步研究这些因素，并通过调整这些因素来优化结果。

3.2.2.3 柱状图一致性。尽管分区是直方图发布中的一种流行机制，但这种机制可能带来不一致的问题。例如，在加入噪声后，两个bin的值之和可能小于一个bin。为了保持直方图发布的一致性，Hay等人[22]定义了一种约束性推理来调整发布输出。探讨了两种类型的一致性约束。第一种，排序约束，要求查询结果满足一个特定的序列。第二种，分层约束，预先定义了分层区间集的序列。一个约束推理步骤应用线性组合方法来估计一组接近于噪声答案的近似答案，这满足了一致性约束。

还有一些其他的方法来提高直方图的一致性。例如，Lin等人[23]将一组排序的直方图视为马尔科夫链，并提出了一种对估计值应用排序约束的算法。Lee等人[24]在发布直方图之前增加了一个后处理步骤，表述为一个受限的最大似然估计问题。

流数据发布

在现实世界的场景中，发布连续更新的数据是比较实用的。这种类型的数据可以简化为一个比特串{0,1}n，流中的每个1代表一个事件的发生[25]。差别化的私有机制在每个时间步骤中都会释放一个比特。

图3说明了一个例子。假设有一个二进制比特流D∈{0,1}T，其中T代表一个时间序列T={tk : k = 0,...}。位σ(tk)∈{0,1}表示在时间tk是否有事件发生。在每个时间步骤tk，

5hohdvhwkh&rxqwrielw

(YHQW

W W W WP

图3：流数据

的计数被表示为f(tk)，机制的噪声输出为f(tk)。

3.2.3.1 数据集的划分。在流数据中划分数据集的目的是递归地将流分成子域，并为每个子域评估一个噪声计数。问题在于定义子域以及流可以被分割多少次。

Chan等人[26]提出了一种p-sums机制，计算流中连续比特的部分和。每个子域由p-sum定义，它是一个估计每个时间间隔的计数的中间结果。拉普拉斯噪声被添加到p-sum结果中，而不是单个计数答案。这保证了误差界限为

，这使噪声的线性复杂度降低到对数复杂度。

Zhang et at. [27] 为空间数据集创建了一个四叉树。他们定义了一个阈值来确定子域的最小值，另一个阈值来限制四叉树的高度。使用这两个阈值，添加到四叉树的噪声量可以被限制在一个常数。

3.2.3.2 迭代。Dwork等人[28]提出了一种持续输出的转化算法，并开发了一种迭代释放机制，在每个时间步长输出持续的计数。误差减少到.

Georgios等人[29]提出了w-event隐私的概念，以实现平衡的隐私目标。他们使用滑动窗口方法制定了迭代机制。这种w-事件隐私旨在掩盖w个时间步骤内的任何事件序列，这适用于无限流的发布。

3.2.3.3 讨论。尽管持续发布是数据挖掘界的一个热门话题，但在隐私保护方面仍有许多未解决的问题。例如，定期发布多维数据，以及处理其他统计查询需要进一步探索。

3.2.4 图形数据发布

随着在线社交网络（OSN）的大幅增长，图数据已经引起OSN参与者的关注。一个OSN数据集可以被建模为一个无向图G=（V,E）。我们用V来表示节点，用E⊆V×V来表示边。在OSN中，节点通常代表个人，而边表示他们的关系。图数据中的差异隐私的两个概念已经被定义：节点差异隐私和边缘差异隐私。

3.2.4.1 边缘差分隐私。边缘差异隐私确保查询的答案不会透露图中某条边缘的包含或移除情况。如果两个图在一条边上有所不同，它们就是相邻的。Nissim[30]对图数据进行了首次差异隐私研究，他展示了如何用边缘差异隐私评估社交网络中的三角形数量。他们使用局部敏感度，用上限严格限制敏感度，并展示了如何相应地有效校准子图计数的噪声。

Karwa等人[31]对这一技术的结果进行了调查，以发布k-triangles和k-star的计数。

他们实现了k-star计数的-差值隐私，以及k-triangle计数的-差值隐私。

Zhang等人[32]声称，如果可以找到一个具有适当统计特性的同构图，并且与原图相似，那么同构图可以用来生成准确的查询答案。给定一个子图G，他们采用指数机制来搜索G的若干同构副本，以回答子图查询。

3.2.4.2 节点差分隐私。在节点差分隐私中，如果其中一个图是通过从另一个图中移除一个节点和与该节点相关的所有边而产生的，则认为这两个图是相邻的。节点差分隐私比边缘差分隐私更严格。由一个节点的变化所导致的敏感度与图的大小成正比。
根据观察，许多有用的统计数据在具有小度数θ的图Gθ上具有较低的敏感度，实现节点差分隐私的常见方法是将图G转化为θ度约束的图Gθ，其中度数超过θ的节点被删除[33], [34]。
Chen等人[35]提出了一个节点差异化隐私的迭代机制。给定一个图G和任何实值函数f，他们定义了一个实值函数序列0 = f0(G) ≤ f1(G) ≤ --- ≤ fm(G) = f(G)，具有递归单调性属性。递归方法将对任何一种具有节点差分隐私的子图返回子图计数。然而，构建函数序列fi(G)通常是NP-hard，有效实现它仍然是一个开放的问题。
3.2.4.3 讨论。现有的方法在边缘差分隐私或甚至节点差分隐私的基本图统计方面工作得相当好。然而，发布特定的统计数据，如切割，节点之间的成对距离，或在超图上，仍然是开放的问题。
3.3 非交互式发布
非交互式设置意味着所有的查询都是一次性给策展人的。非交互式发布的关键挑战是敏感性测量。查询之间的相关性将极大地增加敏感性。我们提出了两种可能的方法来解决这个问题：一种是分解批量查询之间的相关性，这在第3.3.1小节中提出，另一种是发布一个具有差异隐私约束的合成数据集来回答那些提议的查询。相关的方法将在合成数据集发布子节中介绍。
3.3.1 批量查询发布
批量查询发布指的是最常见的非交互式场景，其中固定的m个查询集F={f1,...,fm}被批量回答。
再用表4来说明批量查询问题，假设馆长想为表4发布一批范围查询，而表7包含所有可能的范围查询F={f1,...,f10}。删除D中的任何记录都会改变F中最多6个查询结果（表7中包含x2或x3的列）。根据灵敏度的定义，F的灵敏度为6，这比单个查询的灵敏度高得多。因此，大多数研究的重点是如何降低F的灵敏度。
表7: 批量查询发布示例

3.3.1.1 转化。降低噪声的一个可能手段是在一个新的数据结构A中重新标定灵敏度。
Xiao等人[36]提出了一种小波变换，称为Privelet，它在频率数据集D上应用小波变换，生成小波系数A，A的每个条目被认为是D中条目的线性组合，范围查询可以用基本查询的线性组合生成。表8显示了基本查询的情况。例如，range(x2,x3) = x2+x3的答案可以由range(x2,x3) = 0.5f1 - 0.5f3 - 0.5f4产生，而灵敏度将从6降到3。
，这比拉普拉斯机制中的要小得多。
Li等人[37]提出了一种矩阵机制，可以回答线性计数查询的集合。被定义为工作量的查询集被转化为矩阵A，其中每一行包含线性查询的系数。矩阵机制的基本要素是选择表8：m范围查询

A来代表查询集合。基于对A的选择，矩阵机制可以扩展到各种方法。例如，如果A是一个身份矩阵，这个机制可以被认为是一个正常的拉普拉斯机制，用于批量查询。如果A是用Haar小波选择的，它可以扩展为Privelet[36]。在他们随后的论文中分析了矩阵机制的误差边界[38]。他们证明，当批量查询以矩阵方式表示时，结果的最小误差可以根据这个查询矩阵的频谱特性来评估。

同样，Huang等人[39]将查询集转化为正交查询集，以减少查询之间的关联性。Yuan等人[40]将A的搜索制定为一个受限的优化问题。然而，这种方法是具有挑战性的，因为优化目标是非凸性的。后来，他们考虑了近似的差分隐私，并将优化目标转化为凸程序，以最小化结果的整体误差[41]。

3.3.1.2 数据集的划分。Kellaris等人[42]将数据集的列分解为互不相干的组，并在这些组的计数中加入拉普拉斯噪声。最后的结果是使用新的列计数生成的。因为一个组中受用户影响的原始计数的最大数量是有限的，所以每个组的敏感性降低了，而且-差分隐私所需的拉普拉斯噪声也减少了。这种机制的优点是它可以限制数字数据集的敏感性。

3.3.1.3 迭代。迭代也被用于批量查询发布中。通过递归地逼近真实答案，可以有效地减少输出中的噪声。Xiao等人[43]的目的是减少发布的输出中的错误。他们认为，拉普拉斯机制在不考虑答案的真实值的情况下，为每个查询答案添加了范围不变的噪声。因此，具有小答案的查询具有比预期高得多的误差，定义为相对误差。在实践中，较大的答案可以容忍更多的噪音。在某些应用中，相对误差比绝对误差更重要。为了减少相对误差，Xiao等人[43]提出了一个名为iReduct的机制，该机制最初获得查询答案的粗略误差估计，随后使用这些信息来迭代完善这些误差评估。

3.3.1.4 讨论。批量查询发布中的关键问题是如何降低相关查询之间的敏感性。目前，转换是解决这个问题的最流行的方式。目前的工作主要集中在范围查询上，并开发适当的结构来回答这些查询的线性组合。需要开发更多类型的结构来回答各种类型的查询。迭代或分割数据集可能对相关分解并不有效，但它们有可能回答更多类型的查询。

3.3.2 合成数据集发布

合成数据集发布研究了向公众发布数据集，而不是分享查询的答案。假设输入的数据集是带有属性的D，而策展人想发布一个可以用来回答查询集F的合成D，有两种不同的方法来实现这个目标。一种方法是应用匿名化技术来发布一个匿名数据集，它保留了与原始数据集相同的记录。另一种方法是从数据宇宙中抽取样本，建立一个合成数据集，它遵循原始数据集的分布，但不一定保留相同的记录。

3.3.2.1 基于匿名化的合成数据集发布。这方面的工作认为，如果匿名化过程在每一步都遵循差异化隐私的要求，那么发布的合成数据集将满足差异化隐私。基于这一观察，Mohammed等人[44]提出了匿名化算法DiffGen来保护数据挖掘的隐私。该匿名化程序包括两个主要步骤：分区和扰动。原始数据集的每个属性都被泛化为其最顶端的状态。然后，分区步骤根据属性分类树将这些属性分割成更具体的组。它应用一个指数机制来选择一个候选的专业化。之后，通过扰动将随机噪声添加到每个记录组的真实计数中。

3.3.2.2 基于学习理论的合成数据集发布。Kasiviswanathan等人[45]和Blum等人[14]声称，如果在已发布的数据集上的查询被限制在一个特定的概念集C，学习过程可以在差异隐私的约束下确保查询答案的准确性。

Kasiviswanathan等人[45]提出了一种基于指数的机制，从数据宇宙中搜索能够准确预测C的合成数据集，图4显示，从数据宇宙中创建多个候选数据集后，指数机制。作者声称，对于任何C，该机制将根据和任何D≥{0,1}d搜索最合适的D，如果数据集的大小满足

，

α的精度可以以1-β的概率实现。

随后的工作在提高准确性方面取得了进展。Dwork等人[46]用提升法将准确率的下限提高到。 Hardt等人[47]将指数机制与乘法机制相结合。

此外，Ullman等人[49]表明，在所有计算效率高的算法中，拉普拉斯机制几乎是最优的。事实上，没有任何算法有能力在多项式时间内随机回答超过O(n2)的查询。这一结果表明，很难设计出有效回答任意查询的机制。

表9总结了关于合成数据集的主要工作。为了简单起见，敏感性被预先定义为1，对β的依赖被省略，运行时间只考虑宇宙的大小。

表9: 用学习理论发表的合成数据集

机制 准确度 效率 保密性

Blum等人[14] 效率低下

Dwork等人[48]无效率

Dwork等人[46]无效率

Hardt等人[47]效率不高

学习理论扩展了合成数据发布的研究工作，证明有可能保持可接受的效用，同时保留差异化的隐私。尽管如此，降低计算复杂性的问题仍然是一个挑战。

3.3.2.3 高维数据集的合成数据集发布。无论是匿名数据集发布还是基于学习理论的发布，都不能有效处理高维数据集。在匿名方法中，当输入的数据集包含许多属性时，现有的匿名方法会注入大量的噪音，从而导致效用降低。在基于学习理论的方法中，计算复杂度与数据集的维度成指数关系，这使得发表在高维数据集上不可行。解决高维度的一个有希望的方法是将数据集分解成一组低维度的边际数据集，然后运用方法从这些边际数据集推断出联合数据分布。

Zhang等人[50]遵循上述原理，使用贝叶斯网络来处理高维度问题。他们假设属性之间存在一些相关关系，如果这些相关关系可以被建模，那么模型就可以用来生成一组边际数据集来模拟原始数据集的分布。这种解决方案的缺点是在网络构建过程中消耗了太多的隐私预算，因此，使分布的近似值不准确。

Chen等人[51]通过提出一种聚类方法来解决这一缺点。他们披露了所有属性的成对相关性，并生成一个依赖图。一个结点树算法被应用于该图，以确定一组属性聚类，这些聚类被用于生成噪声边际。作为最后一步，一个推理模型被用来创建一个合成数据集。与[50]相比，他们对数据集的访问是有限的，节省了隐私预算以获得更好的结果。

3.4 差异性隐私数据发布的总结 3.4.1 交互式设置的总结

由于统计数据库的进步，交互式设置已经引起了人们的注意。在交互式设置中，隐私机制接收用户的查询，并以噪声回答来保护隐私。传统的拉普拉斯机制只能回答O(n)个查询，这在许多情况下是不够的。研究人员必须提供不同的机制来修复这一基本弱点。

所提出的交互式发布机制在查询类型、最大查询次数、准确性和计算效率等方面提高了性能。经过分析，我们得出结论，互动式发布中的这些衡量标准是相互关联的。例如，在一个固定的隐私预算下，较高的准确性通常会导致较少的查询次数。另一方面，在固定的精度下，较大的查询次数通常会导致计算效率低下的机制。因此，数据发布机制设计的目标是实现一个更好的结果，能够平衡上述的测量。机制的选择取决于应用的要求。

3.4.2 非交互式设置的总结 高灵敏度在非交互式设置中是一个很大的挑战。批量查询发布方法只能发布有限的查询类型。发布合成数据集似乎很吸引人，因为在某些场景下，人们需要属性的细节来确定进一步的分析方法。然而，关于合成数据发布的研究仍处于早期阶段，在这个领域有许多开放性问题。基本问题是效率。鉴于大多数发布机制需要从整个数据宇宙中抽取数据集，很难在多项式时间内搜索到合适的数据集。

另一个问题是，合成数据集发布只能为特定目的发布数据集。例如，一个匿名化的数据集专注于决策树算法，发布的数据集对于决策树任务获得了可接受的结果，然而提出的方法并不能保证其他类型任务的性能。基于学习的方法也有同样的缺点，甚至更糟，因为它们只保证对某一类的学习性能。发布一个用于多种目的的数据集需要进一步调查。
第三个问题是处理高维数据集。尽管[50]和[51]已经进行了一些表10。差异化的私有数据分析问题特征

差异性私有数据分析
输入数据的性质 交易
输出数据的性质 分析模型/算法
分析框架 Laplace/exponential框架，私有学习框架
分析机制 Laplace/exponential框架的Laplace/exponential机制，私有学习框架的学习过程
挑战 准确度，计算效率

在最初的工作中，它们在建立分布模型时都消耗了太多的隐私预算，使得结果不如低维数据集的准确。
4 差异性隐私数据分析
差异化隐私数据分析的基本任务是将当前的非隐私算法扩展为差异化隐私算法。这种扩展可以通过几个框架来实现，大致可分为拉普拉斯/指数框架和私有学习框架。拉普拉斯/指数框架将拉普拉斯或指数机制直接纳入非私有分析算法。例如，在算法的计数步骤中加入拉普拉斯噪声，或者在进行选择时采用执行指数机制。
私人学习框架将数据分析视为优化方面的学习问题。学习问题是通过定义一系列目标函数来解决的。与拉普拉斯/指数框架相比，私有学习框架有一个明确的目标，这种框架产生的结果在风险边界或样本复杂性方面更容易比较。但是私有学习框架只能处理有限的学习算法，而几乎所有类型的分析算法都可以在拉普拉斯/指数框架中实现。
表10显示了不同的私有数据发布问题的特点。研究人员关注的是这两个框架的准确性和计算效率。由于不同的论文使用不同的术语来描述输出，因此本节中 "模型 "和 "算法 "这两个术语可以互换。 最常见的扩展方法是将拉普拉斯或指数机制纳入非私有分析算法中。这些算法通常与特定的机器学习或数据挖掘任务有关，它们被分为监督学习、无监督学习和频繁模式挖掘等类别。
4.1.1 监督学习
监督学习指的是通过一组标记的训练记录提取描述数据类别的模型的预测方法[52]。作为最流行的监督学习算法之一，决策树学习在Laplace/exponential框架中得到了广泛的研究。
决策树是一个迭代过程，它对训练样本进行递归划分，建立一棵树，每个标签代表一个叶子。假设有一个带有d个分类属性{a1,...ad}的输入数据集D，从持有所有训练记录的根部构建一棵决策树，然后算法选择信息增益最大的属性ai，将记录划分为子节点。这个过程在训练记录的每个子集上递归进行，直到满足停止标准。

第一个差异化私有决策树算法是在SuLQ平台上开发的[53]。噪声被添加到信息增益中，选择具有噪声信息增益小于指定阈值的属性ai来划分一个节点。然而，由于在每个迭代中对每个属性的信息增益进行单独评估，隐私预算在每个迭代中被多次消耗，这导致了大量的噪声。此外，SuLQ不能处理连续属性。如果这些属性被简单地离散成区间，那么就违反了差分隐私的基本概念，因为连续属性中的分割值会暴露出记录的信息。

为了克服SuLQ平台的缺点，Friedman等人[54]以两种方式改进了该算法。首先，他们在属性选择步骤中实施了一个指数机制。分数函数由信息增益或增益比定义。得分最高的属性被选中的概率更高。通过这种方式，消耗的隐私预算比SuLQ少。第二，建议的方法可以处理连续属性。采用指数机制来选择每一个可能的分割值，连续属性域被划分为这些区间。与SuLQ相比，他们获得了更好的性能。
Jagannatham等人[55]提供了一种建立随机私有决策树的算法，该算法随机选择属性来创建节点。该算法首先创建一棵树，其中所有的叶子都在同一层次上，然后建立一个叶子计数向量。一旦独立的拉普拉斯噪声被添加到计数向量中，就可以从噪声叶向量中生成一棵差异化的私有随机决策树。该算法迭代地产生多个随机决策树，并使用集合法来组合这些树。由于属性是随机选择的，这一步节省了隐私预算；然而，由于每棵树的大小是随着最后合奏步骤中树的数量而放大的，合奏的效用仍然是一个问题。
Rana等人[56]提出了一种在随机森林中集合决策树的实用方法。他们没有严格遵循差分隐私的概念，即保持相邻的数据分布近似不变。相反，他们只保持统计学特征的不变性。定义了一个隐私攻击模型来证明所提出的决策树的可靠性。由于在集合过程中消耗的预算较少，与其他算法相比，这种放宽差分隐私的做法可以带来更高的效用。
讨论。差分隐私的决策树算法是差分隐私界最早研究的算法。这一系列方法的优点是简明扼要，易于实现。然而，由于基于树的算法需要选择分裂 算法1 基本k-means算法

要求：点r1, ... ,rn。
确保：k个中心{v1,...,vk}与排列的点重复

1. 通过将每个点分配给其最近的中心来创建k个集群。
2. 重新计算每个簇的中心；直到中心不改变

多次的属性，隐私预算很快就被消耗掉了，这就造成了巨大的效用损失。这个缺点源于决策树的构建，不容易处理。现在，设计差异化的私有监督学习算法的最流行的方法之一是应用私有学习框架，这将在第4.2小节讨论。
4.1.2 无监督学习
作为一种典型的无监督学习方法，聚类算法将未标记的记录分组到聚类中，以确保同一聚类中的所有记录都彼此相似。假设输入是一组点，r1,...,rn，聚类的输出是k个中心{v1,...,vk}和分配的点。
一个基本的k-means算法由算法1制定。差分隐私聚类的目的是在中心v和每个中心的记录数中添加不确定性。为了实现这一目标，在算法1的步骤1中加入了噪声。事实上，向聚类中心添加噪声是不切实际的，因为聚类中心的敏感性将相当大，因为删除一个点将完全改变中心。因此，聚类的挑战是评估和最小化聚类中心的敏感性。
Nissim等人[30]利用k-means聚类的局部敏感性来规避大的敏感性问题，其依据是以下的直觉。在一个良好的聚类情况下，一个有噪声的点应该与它之前的中心大致相同。此外，移动一些 "聚类良好 "的记录，最终不会改变中心。因此，他们定义了一个局部敏感度来衡量基于记录的聚类中心的敏感度，这比传统的全局敏感度要低得多。由于本地灵敏度的值

频繁项集挖掘
频繁项集挖掘的目的是发现经常出现在数据集D中的项集。假设I是D中的项集，项集是指I的一个子集。一个频率大于预定的支持度阈值的项目集被称为频繁项目集或频繁模式。
让U代表所有的频繁项目集，其中U中最频繁的前k个项目集应在差异化隐私保证下被释放。拉普拉斯噪声通常被添加到频率中；然而，主要的挑战是项目集的总数与项目的数量

呈指数关系。如果I包含n个项目，所有可能的项目集的数量是
k
. 减少候选项集的数量是差异化私有频繁项集挖掘中的一个主要研究问题。
Bhaskar等人[61]通过提供截断频率来减少候选项集的数量，解决了这个问题。他们提出了一种算法，使用指数机制来选择top-k项目集。每个候选项的得分函数是频率，定义为，其中pk是第k个最频繁的项目集的频率，γ∈[0,1]是一个精度参数。每一个频率大于pk -γ的项目集被计算为其正常频率p(U)，而其余的频率被截断为pk - γ。通过这种方式，计算成本被大大降低。
截断频率的好处是，它可以大大减少候选项目集的大小。然而，它只适用于k小的时候。另一个弱点是top-k项目集的长度需要预先定义，这影响了频繁项目集挖掘的灵活性。
为了解决这些弱点，Li等人[62]构建了一个基集B={B1,B2,...}，其中任何频率高于阈值的项目集都是基集Bi的一个子集。该算法基于B生成了候选项集，并且该算法可以发布任意长度的项集。然而，生成基础集B并不容易；此外，当项目集的长度和基础集的数量都很大时，候选项目集的cardinality仍然大得难以处理。有效地减少候选项集的数量仍然是一个挑战。
Zeng等人[63]提出了一种算法，根据预先定义的最大cardinality随机地截断数据集中的交易。该算法迭代生成候选项集，并扰乱这些候选项集的支持度。
Lee等人[64]提出了一种基于FP-树的频繁项目集挖掘算法。该算法首先识别所有的频繁项集，而不知道它们的确切支持度，只知道它们的支持度高于一个预定的阈值。该算法在中间步骤（FP-树）向数据结构中注入噪声。最终的输出可以通过一个可选的后处理步骤进一步完善。该算法的优点是，信息披露
表11: 监督学习方法的比较
难度 主要方法 典型论文 优点 缺点
隐私预算必须多次消耗 在信息增益中加入噪声 [53] 易于实施 由于属性选择中的隐私预算安排，噪声会很高
使用指数机制来选择属性 [54] 在属性选择中节省部分隐私预算 仍然有高噪音
选择属性
随机 [56], [55] 在属性选择过程中不消耗隐私预算 隐私预算在集合过程中会被大量消耗掉

表12: 无监督学习方法的比较
难度 主要方法 典型论文 优点 缺点
聚类的高灵敏度% 使用局部灵敏度器[30], [57] 降低灵敏度水平 局部灵敏度可能不容易估计
Johnson-Lindenstrauss变换[58] 保证差分隐私，同时保留点之间的距离 Johnson-Lindenstrauss变换只对规范2距离测量有效。

影响差分隐私只发生在超过预定阈值的计数查询中；否定的答案不计入隐私预算。
Shen等人[65]应用马尔科夫链蒙特卡洛（MCMC）抽样方法来处理大型候选项目集的挑战。他们声称，MCMC随机行走方法可以绕过这个问题，然后可以应用指数机制来选择频繁项集。他们使用这种方法来挖掘频繁图的模式。
Xu等人[66]应用二进制估计方法来识别所有可能的频繁项集，然后使用指数机制来挖掘频繁项集。

讨论：频繁项集挖掘是一个典型的数据挖掘任务，它受到搜索大型候选集的影响。差异性隐私使问题变得更糟。[61], [63], [64]和[62]试图用不同的方法将一些候选项合并成组。[65]采用了一种抽样方法来搜索候选项集。[66]采用了二进制搜索方法。所有这些方法都将搜索空间从指数级降低到多项式，这是一个很大的成就。然而，噪音仍然很高，需要进一步降低。

4.1.4 关于拉普拉斯/指数框架的讨论 拉普拉斯/指数框架可以将拉普拉斯和指数机制自由引入各种类型的算法。然而，分析结果的效用是这个框架的一个挑战。当在算法步骤中加入噪声时，不清楚效用损失会有多大，分析结果不容易比较。

解决这个困难的一个可能的方法是通过优化的观点解决数据分析问题，并利用一些现有的理论，如学习理论[45]，这样就可以估计和比较效用损失。基于这种直觉，研究人员提出了私有学习框架。

4.2 私人学习框架

另一条差异化隐私研究路线是从机器学习的角度来研究学习问题，图5：学习问题和过程描述

理论[67]。一个学习问题显示在图5a中。假设D = {r1,...,rn}是一组从宇宙X中抽取的样本。图中的点和三角形表示两个标签y∈{0,1}。假设有一组函数（假设）H和hj，学习过程L的目标是在几乎所有的宇宙X上找到一个与c满足的假设h。通过使用输入的训练样本，学习者选择最合适的h∈H作为输出模型w。

私有学习框架的目的是设计一个私有学习器，输出一个近似准确的模型，并保留训练样本的差异性隐私。私有学习框架关注以下问题。

- 如何在差分隐私方面选择一个最佳模型?
- 需要多少个样本才能达到有界的准确性?

  第一个问题可以通过结合经验风险最小化（ERM）技术和差分隐私来回答。实用性是由风险边界来衡量的。第二个问题可以通过PAC学习理论来分析私人学习者的样本复杂性来解决。风险边界和样本复杂度都可以被认为是公式7的实现。

  4.2.1 ERM中的私有学习

  表13：频繁项集挖掘方法的比较

  难度 主要方法 典型论文 优点 缺点

  大型候选项集 将候选项集合并成组 [61] [62] [63] [64] 容易实现 合并策略对输出有很大影响。一些重要的模式可能会被遗漏。

  候选集的二进制搜索。 [66] 频繁项集可以相当准确 搜索时间可以减少，但效率仍然很低。

  从候选集中取样 [65] 效率很高，可能会遗漏一些重要的模式。

4.2.1.1 ERM的基础：ERM用于从一组假设中选择一个最佳模型，方法是使所收集的样本的期望损失最小化[68]。假设h∈H是一个假说，w是输出模型。我们定义一个损失函数来估计假说的预期风险。ERM的目标是确定一个在训练样本D上最小化经验风险Rn(w)的w。公式8显示了经验风险最小化。

(8)

其中正则器λμ(w)防止过度拟合。

通过选择不同的损失函数，ERM可以实现某些学习任务，如线性回归[69]，其中损失函数被定义为最大似然估计（MLE）；逻辑回归[68]，其中损失函数被定义为逻辑损失；以及核方法[70]，[71]，其中损失函数被设定为铰链损失。为了使ERM的解决方案具有可操作

性，人们假设：1）损失函数和正则器是凸的；2）损失函数是L-Lipschitz。在这两个假设下，公式8可以被认为是一个d维的凸优化问题。
私人ERM的效用是由作为风险边界的差异来衡量的。较低的风险界限会导致较高的效用。实际风险R( )和私人R( )之间，定义为
4.2.1.2 扰动方法。在将差分隐私纳入当前的学习过程时，当前的工作通过ERM技术应用两种方法：输出扰动和目标操作。输出扰动在输出w中插入噪声；而目标操作在学习之前向目标函数添加噪声。
Chaudhuri et