摘要

当今大数据时代机器学习技术发挥了重要的作用，在图像识别、推荐系统和自然语言处理等领域都取得了令人瞩目的成功，这部分归功于用于训练机器学习模型的数据集。然而这些数据集很可能包含个人的敏感信息，直接发布数据或是将数据用于训练模型存在隐私泄露的风险。因此，如何在保护用户隐私的前提下更加有效的挖掘和利用数据是亟待解决的问题。差分隐私技术是针对隐私泄露问题的一种有效手段，通过在查询结果上添加满足一定分布的噪声，使得攻击者无法判断某个用户是否在数据集里，即便该攻击者具备较强的背景知识。

本文聚焦流数据的参数估计问题：针对流数据大量、快速、实时到达和一经处理便不再保存的特点，无法使用全部的数据集去估计参数。本文利用在线更新的思想估计流数据的参数，即每次更新只使用以往数据的统计量和当前批数据，并结合差分隐私技术保护用户隐私。本文的主要研究内容和成果具体如下：

首先，比较了分别使用(ε,δ) −差分隐私和高斯差分隐私的高斯机制达到相同的隐私保护效果所需添加噪声的方差，结果表明，在ε较大时两种差分隐私机制添加噪声基本相同，而ε较小时使用高斯差分隐私的高斯机制添加的噪声显著更小。其次，提出了差分隐私保护下的流数据参数更新算法，在新的一批数据到达后，仅使用这批数据对参数做一次梯度下降更新。这里用一个梯度裁剪参数对较大的样本点的梯度进行裁剪，以便控制敏感度；基于上述比较选择使用高斯差分隐私的高斯机制在梯度上添加正态分布噪音，使整个更新步骤满足差分隐私。最后，给出了整个算法的隐私保护效果，本文利用高斯差分隐私的平行分解定理，整个算法相当于多个作用于不同数据集的隐私机制的组合，组合之后的隐私预算由隐私效果最差的单个机制决定。比起其他算法，该算法以较小的隐私预算实现了良好的隐私保护效果。

在模拟数据集和真实数据集上的实验结果表明，通过合理的选择裁剪参数值，该算法能够获得和不加噪声的情况下几乎相同的准确率，说明本文的算法兼顾了隐私保护和数据可用性。

关键词：隐私保护，差分隐私，流数据，梯度下降

Abstract

Machine learning technology play an important role in the age of big data, with remarkable success in areas such as image recognition, recommendation systems and natural language processing, thanks in part to the data sets used for training model. However, these data sets are likely to contain sensitive personal information, and there is a risk of privacy disclosure if they are directly publishing or used for training. Therefore, how to mine and utilize data more effectively under the premise of protecting users' privacy is an urgent problem to be solved. Differential privacy is an effective technology to solve the privacy leakage problem. By adding noise to the query results, the attacker cannot judge whether a user is in the data set, even if the attacker has strong background knowledge.

This paper focuses on the parameter estimation of streaming data: due to the fact that streaming data is massive, fast, real-time and no longer stored once processed, it is impossible to estimate parameters using all data sets. In this paper, we use the idea of online updating to estimate parameters, that is, each update only uses current batch data and the statistics of previous data, and combined with differential privacy technology to protect privacy of users. The main research contents and achievements of this paper are as follows:

Firstly, we compared the variance of noise added to achieve the same privacy effect using the Gaussian mechanism of (ε, δ) −differential privacy and Gaussian differential privacy separately. The result shows that the noise added by the two differential privacy mechanisms is basically the same when the ε is large, but the noise added by the Gaussian mechanism using Gaussian differential privacy is significantly smaller when the ε is small. Secondly, we proposed a parameter estimation algorithm for streaming data with differential privacy protection. That is, When a new batch of data arrives, only use this batch of data to update the parameters once. A gradient clipping parameter is used to clip the gradient of larger sample points to control the sensitivity; based on the above comparison, the

Gaussian mechanism of Gaussian differential privacy is selected to add normal distribution noise on the gradient, so that the whole update step satisfies differential privacy. Thirdly, we given the privacy preserving effect of the whole algorithm. By using the parallel composition theorem of Gaussian differential privacy, the whole algorithm is equivalent to the combination of multiple privacy mechanisms apply on different data sets, and the privacy budget of the combination is determined by the single privacy mechanism with the worst privacy effect. Compared with other algorithms, this algorithm achieves good privacy preserving effect with a smaller privacy budget.

At last, we achieve the simulation and the experiments on real data sets, which show that the algorithm can achieve almost the same accuracy as that without adding noise, by choosing the appropriate clipping parameters, this indicate that the algorithm gives consideration to both the privacy protection and data availability.

Keywords: privacy protection, differential privacy, streaming data, gradient descent

目录

1.绪论

1.1 研究背景和意义

随着互联网的发展和大数据时代的到来，数据已经渗透到当今每一个行业和业务职能领域，有着难以估量的潜在价值。最近几年的数据量更是呈爆发式增长，权威数据统计机构 Statista 的数据显示，2016 至 2019 年的全球数据量分别为 18ZB、26ZB、33ZB、41ZB，堪称海量；国际数据公司 IDC 预计，到 2025 年，全球数据量更将是 2016 年的 9 倍，如果能将如此庞大的数据资源善加利用，其价值无疑将是巨大的。数据在国民经济发展中的地位日益凸显，党的十九大将数据与土地、劳动力、技术和资本并列作为新的五大生产要素。值得一提的是，数据区别于传统的生产要素，其生产过程和价值实现的方式也不尽相同，数据的价值不在于其本身，而在于通过分析数据从而更好地服务社会。

为了充分发挥数据的价值，打破“数据孤岛”问题，可以对数据进行公开发布，以实现数据共享。对于政府来讲，开放数据有利于更好地了解政策如何运作和民众意愿，以提高政府效率，提供更全面的分析见解。对于企业来讲，通过整合和分析数据可以更好地了解客户需求，为客户提供更良好的服务，增加自身营收。不幸的是，这些数据在有着巨大价值的同时，通常也包含敏感信息，例如个人的家庭住址、活动轨迹、薪资、消费记录和消费偏好等，如果不经处理就发布数据，很可能泄露敏感信息，对公民人身和财产安全带来严重的威胁。

在意识到数据具有的价值之后，各种 app、公司想方设法地收集用户数据，甚至有违法收集和交易个人数据的情况，隐私泄露案例层出不穷，个人隐私泄漏问题已然成为社会发展的隐患。2016 年，Facebook 将其收集到的 8700 万

用户数据违规泄露给政治咨询公司剑桥分析，用于在 2016 年总统大选时支持前美国总统特朗普，违反了平台此前承诺保护用户隐私的协议，Facebook 于 2020 年收到了来自美国联邦法院的 50 亿美元罚单。2020 年，圆通速递有限公司内部某员工与外部不法分子勾结，利用员工账号和第三方非法工具窃取运单信息，导致 40 万条个人信息泄露。根据中国消费者协会开展的“App 个人信息泄露情况”问卷调查，有超过八成的受访者表示曾遭遇个人信息泄露问题，受访者普遍担心个人数据被利用于从事诈骗窃取活动和贩卖或者交换给第三方。垃圾短信浪费着我们的时间，电信诈骗侵害着我们的财产，个人隐私泄露问题甚至威胁到我们的生命。

隐私保护问题在国内外已经受到了广泛的关注。2018 年 5 月，欧盟《通用数据保护条例》在欧盟全体成员国内正式生效，全面加强了欧盟所有网络用户的数据隐私权利，明确规定了数据主体对数据的知情权、拒绝权、更正权和删除权等权利。2021 年 8 月十三届全国人大常委会第三十次会议表决通过《中华人民共和国个人信息保护法》，其中明确规定：对违法使用个人信息的应用程序，责令其暂停或者终止提供服务。虽然目前已经有相关条例保护用户隐私权益，但出于分析和挖掘数据的需要，仍然需要思考数据共享的问题。目前的事实是，一方收集到的数据通常是小规模、碎片化的，对于训练神经网络等模型来说，无论是对数据量还是对数据维度的要求都需要利用共享的方法将这些数据整合起来。对于发布数据来讲，亦希望有一种保护个人隐私的数据发布方法。

差分隐私是 Dwork 等在 2006 年针对数据库的隐私泄露问题提出的一种新的隐私定义(Dwork 等, 2006; Dwork, 2011)，通过在发布结果上添加满足一定分布的噪声，使数据库查询结果对于数据集中单条记录的变化不敏感，从而保护个人隐私。差分隐私目前不仅在学术界上有着大量研究，其成果已经在工业界中有广泛应用，如谷歌的 chrome 浏览器和联邦学习系统(Geyer 等, 2017; Noble 等, 2022)等。将差分隐私应用到机器学习中的直觉是，机器学习应该学习到某个类别的特性（例如吸烟的人更容易患肺癌），而不是关于某个实例的特性（例如张三患肺癌），这和差分隐私致力保护单个用户的敏感信息不谋而合。另外，在发布机器学习模型后，攻击者可以通过模型输出推断某些记录是否参与了模型训练，例如模型反演攻击(Fredrikson 等, 2014)、成员推理

攻击(Shokri 等, 2017)等。即使不公开发布模型参数，攻击者也可以通过不断查询模型输出推测训练数据集，或是在模型训练阶段进行攻击，从而造成数据泄露，因此在模型训练期间保护隐私是必要的。

本文将差分隐私技术应用到流数据的参数估计中。流数据是一组顺序、大量、快速、连续到达的数据序列，数据按批次可用且当前数据使用过后便不再储存，例如网络监控和传感器网络等。本文主要利用小批量梯度下降算法估计流数据的参数，同时在参数迭代过程中加入噪声以确保攻击者无法推测具体的某条记录是否参与了训练。

本文的研究意义如下：第一，本文基于在线更新的思想提出了差分隐私保护下的流数据参数估计算法，并且给出了该算法的隐私保护效果，后续实验表明，本文的算法能够兼顾隐私保护效果和参数可用性。第二，本文比较了$(\varepsilon,\delta)-DP$和$\mu-GDP$达到相同的隐私保护效果所需添加噪声的大小和参数估计的准确性。结果表明，使用$\mu-GDP$的高斯机制在添加更小噪声的情况下实现了优于$(\varepsilon,\delta)-DP$的准确性，这为隐私机制的选择提供了一定建议。

1.2 国内外研究现状

随着大数据技术的不断发展和应用，人们对于用户隐私和数据安全性的关注度不断提高。传统的隐私保护技术包括数据脱敏化处理和数据匿名化。数据脱敏化处理，即删除所有敏感且明确的标识符（例如姓名，地址和电话号码），但是在大多数情况下，通过将数据与其他数据相连接或通过查看已发布数据的独特特征，可以使匿名的数据重新标识个人。例如美国马萨诸塞州发生的著名的隐私泄露事件，卡内基梅隆大学的博士 Latanya Sweeney 通过连接发布的匿名数据与选民登记记录，成功破解了匿名数据，并且找到了当时的马萨诸塞州州长 William Weld 的医疗记录。匿名化技术包括k-匿名(Sweeney, 2002)、l-多样(Machanavajjhala 等, 2007)和 t-closeness(Li 等, 2006)等，这些匿名方法在一定程度上有效，但都没有对攻击者的背景知识做出假设，特别地，在攻击者拥有较强背景知识时可能失效。随着数据复杂度的增加，匿名化技术已经基本无法适应保护数据隐私要求。

安全多方计算(MPC)是隐私保护研究的核心领域之一，最初是针对安全两方计算问题，由姚期智院士提出的(Yao, 1982)。安全多方计算包括不经意传输(OT)(Keller 等, 2016; Naor 和 Pinkas, 2001)、秘密共享(SS)(Rabin 和 Ben-Or, 1989)和混淆电路等，在没有可信任的第三方前提下，让多个参与方安全地共享数据，而不需要将数据暴露给任何一个参与方，并且通过数学理论保证输入的隐私性和计算的正确性。然而目前安全多方计算的应用仍然存在一些挑战，一方面它依赖于计算机的性能，会占用较多内存；另一方面，节点之间的频繁交互，对延迟和传输速度都有要求，当节点增加时，计算速度呈指数倍下降。

同态加密(Rivest 等, 1978)是隐私保护的另一核心技术，近年来有大量研究和应用围绕其展开。同态加密是一种加密算法，除了要求实现最基本的加密操作，还要求密文之间满足多种计算功能，即用户在密文上计算再解密的结果等同于在明文上进行计算。一个典型的例子是用户将密文传输给服务器，服务器利用密文计算后返回结果，用户再自己解密，由于传输和计算都是通过密文进行，因此具备安全性。根据加密算法满足的运算，同态加密算法分为加法同态加密算法(Goldwasser 和 Micali, 2019; Paillier, 1999)、乘法同态加密算法(Rivest 等, 1978)和全同态加密算法(Gentry, 2009)等。同态加密一定程度上解决了隐私安全性的问题，但是多次加密解密带来的计算开销和传输开销仍无法避免。

2006 年，Dwork 等提出了差分隐私概念(Dwork 等, 2006)，在匿名数据集上加满足一定分布的噪声，使得攻击者无法还原用户信息。不仅如此，他们还在数学上证明了，只要添加满足差分隐私要求的噪声，可以保证即使在知道除了某一记录外的其他所有记录，对手依旧大概率无法确定该记录是否包含在所访问的数据集中。差分隐私要求攻击者无法以明显的优势通过其查询结果来推断任意一个样本是否在所访问的数据集中，差分隐私模型不依赖于攻击者所拥有的背景知识，也因此对隐私信息提供了更高级别的语义安全(李杨等, 2012)。

由于差分隐私的定义过于严格，Dwork 随后提出了松弛版本的差分隐私，使用两个隐私预算参数ε和δ来衡量隐私，并且给出了隐私机制实现的方式(Dwork, 2011; Dwork 和 Roth, 2014)。Dwork(2011)首先给出了敏感度的概念，

敏感度度量了当数据集变化一条记录时查询结果能够变化的最大值，用以衡量噪声的分布和大小；随后给出了隐私机制的实现方式，laplace 机制采用l_1敏感度并加入拉普拉斯噪声使得算法满足ε −差分隐私，高斯机制采用l_2敏感度并加入正态分布的噪声使得算法满足(ε, δ) −差分隐私。

差分隐私在数据隐私保护上具有诸多良好的性质，使得其被用于各种数据发布的领域。直方图是一种常用的数据分布形式，也经常与差分隐私结合用于数据发布中，Dwork 和 Roth(2014)提出了一种针对直方图数据发布的差分隐私方法，由于数据集中每个样本的变化最多引起直方图中一个数据格上的变化，因此他们在每个数据格上添加服从拉普拉斯机制的噪声，使得单条数据的变化对数据格的影响有限。唐海霞等(2020)提出了一种自适应分配隐私预算的直方图发布算法，隐私预算影响到添加噪声的大小，该算法通过分析分组前后的噪声误差和重构误差，在一定的差分隐私预算下均衡两种误差，提高了数据发布的可用性。除此之外，差分隐私由于其采样性质和噪声机制，也经常用于权衡高维隐私数据发布复杂度高、可用性低等问题(顾贞等, 2021; 陈恒恒等, 2021)。

对同一个数据集的多次查询会暴露数据集的更多信息，如果想要保证多次查询后仍然能保护隐私，则需要使用更多的隐私预算。McSherry 等(2009)给出了差分隐私的两条重要的组合定理：顺序分解定理和平行分解定理，证明了差分隐私机制的组合仍然满足差分隐私。顺序分解定理指出如果多个差分隐私机制作用于同一数据集，那么它们的组合需要的隐私预算是单个机制隐私预算之和；平行分解定理指出如果多个差分隐私机制作用于完全不相交的数据集，那么它们的组合需要的隐私预算是单个机制里隐私预算最大的那个。Dwork 和 Roth(2014)在之后提出了更强的组合定理，这里他们考虑了多个查询机制之间的关联，使得k个作用于相同数据集的隐私机制的组合满足$O\left(\sqrt{k}\right)\cdot\varepsilon$ −差分隐私，进一步减小了所需隐私预算。Abadi 等(2016)提出了 Moments accountant 算法(MA)，在梯度上加噪声，利用抽样和梯度裁剪等技术，使得隐私机制组合需要的隐私预算进一步减少。

Kairouz 等(2015)指出差分隐私具有假设检验的背景，并提出了差分隐私的假设检验等价形式。后续差分隐私相关工作也有集中在寻找松弛版本的差

分隐私定义，例如截断的集中差分隐私(Bun 等, 2018)和基于$R\acute{e}nyi$散度的$(\alpha,\varepsilon)-RDP$(Mironov, 2017)，这些定义具有一定的新颖性，但都不再具备假设检验的解释(Dong 等, 2019)。基于假设检验的思想，董金硕、苏炜杰等创新性地定义了一种$f-DP$来刻画隐私(Dong 等, 2019)，随后引入了高斯差分隐私$(\mu-GDP)$作为$f-DP$的特例。由于使用函数来对隐私机制进行量化，相比于其他隐私保护机制，高斯差分隐私对隐私的度量更加准确。与此同时他们提出了高斯差分隐私的高斯机制和组合定理，并基于上述定理和机制提出了NoisySGD 算法，该算法以随机梯度下降为基础，利用$f-DP$的采样性质，在每次参数迭代后加入噪声，使整个算法满足良好的隐私效果。

上述 MA 和 NoisySGD 方法都使用了 mini-batch 梯度下降算法，并根据各自隐私定义的抽样性质衡量隐私保护效果。梯度下降法(Robbins 和 Monro, 1951)是机器学习中常见的优化算法，大量应用于神经网络的模型训练中，其中 mini-batch 梯度下降算法是对梯度下降和随机梯度下降的一种改进(Ruder, 2016)，在实际应用中有着不错的准确率和学习速度。随机梯度下降算法在在线学习中(Rattray 等, 1998; Wilson 和 Martinez, 2003)也有诸多应用，类似的，在流数据的参数估计问题中，可以使用一种在线更新的方法，即每次参数更新只使用当前批数据和以往数据的汇总统计量。Schifano 等(2016)提出了在线最小二乘估计方法，并随后给出了 CEE 估计量的在线更新方法。Luo 和 Song (2020)提出了一种基于 rho 架构的算法来处理流数据(streaming data)下极大似然估计量的参数更新问题，使得每次对参数的更新只需要用到当前的数据和以往数据的几个统计量，而不依赖全部的数据集，从而大大提高了参数估计的效率。

流数据在实际生活中有着广泛的应用，例如客户使用移动程序生成的网络日志、游戏内玩家活动和网购活动等，出于隐私保护的需要，也有越来越多学者将流数据与差分隐私相结合。(Dwork 等, 2010; Dwork, 2010)中作者关注流数据下直方图发布的差分隐私保护方法，主要应用场景为连续统计数据的查询。林富鹏等(2015)首先提出 PTDSS 算法，解决固定长度二维流数据的统计数据发布问题，并添加差分隐私要求的噪声，林富鹏等随后提出针对任意长度二维流数据的数据发布算法 PTDSS-SW，该算法保护了隐私且具有一定的数据可用性。Kellaris 等(2014)结合用户级隐私和事件级隐私，提出 w-事件

级隐私，它保护在 w 时间戳内发生的任何事件序列。Chan 等(2011)假定流数据为一串 0 和 1 的序列，希望输出到目前为止 1 出现的次数，作者提出相应的隐私保护计数器并给出误差范围。夏小玲和刘慧艺(2016)提提出 DDPA 算法，主要处理分布不均匀的流数据的直方图发布问题，在ε值较大的时候有良好的效果。

我们注意到，目前虽然已经有部分差分隐私技术与流数据结合的工作，但它们大部分集中在流数据的统计量发布，而流数据的参数估计与差分隐私技术结合的相关文献还较少。因此，本文主要研究将随机梯度下降算法利用到流数据的参数估计中，并结合差分隐私技术提出相应隐私保护算法。

1.3 主要工作和创新之处

1.3.1 论文主要工作与贡献

大数据时代流数据的广泛应用和隐私保护的需求促使我们思考如何在使用流数据的过程中保护隐私。由于流数据大量、快速和顺序到达的特点，且当前批数据使用过后便不再储存，与随机梯度下降算法的思想十分契合，因此本文主要研究利用随机梯度下降算法估计流数据的参数，并结合隐私机制和梯度裁剪技术保护隐私。

本文的主要研究内容与贡献如下：第一，比较了通过两种差分隐私定义的高斯机制添加噪声的方差，以选择一个更适合本文算法的加隐私机制。结果表明，在ε较大时使用(ε,δ) −差分隐私和高斯差分隐私添加噪声基本相同；而ε较小时使用(ε,δ) −差分隐私添加的噪声明显更大。第二，提出了差分隐私保护下的流数据参数更新算法，并根据高斯差分隐私的组合定理给出算法的隐私保护效果，该算法能够添加较小的噪声达到不错的隐私保护效果。第三，在模拟数据集和真实数据集上应用了本文算法，首先分别在线性回归和逻辑回归模拟数据集上比较了算法的参数估计效果，结果表明，本文的算法在保护隐私的同时，参数估计值与真实值比较接近，具有较高的参数可用性。其次在信用卡欺诈检测数据集和车险销售数据集上的实验也说明本文的算法具备较高的预测准确率。

1.3.2 本文创新点

本文的创新之处主要集中在以下两点：

第一是高斯差分隐私和(ε, δ) −差分隐私对应的高斯机制的比较，尽管参考论文中已指出两种隐私定义可以相互转化，但主要是为了探究$f - DP$的采样机制。本文就噪声大小这一点进行比较，一方面是考虑到本文算法希望实现更好的参数可用性，这就需要尽可能减少加入的噪声；另一方面也在一定程度上对于隐私机制的选择给予了一定建议。

第二是本文结合在线更新和差分隐私技术提出了流数据的参数估计算法。目前在线更新在流数据中的应用并未涉及到隐私保护，而流数据的差分隐私方法又集中在统计量的发布，因此本文的算法具备一定的创新性。此外，本文根据平行分解定理推导了算法的隐私保护效果，该算法的差分隐私机制作用于不同数据集，相比起目前常见的作用于相同数据集的算法，本文算法可以通过添加更小噪声实现相当的隐私保护效果。

2.理论基础

本章将介绍与本文内容相关的一些背景知识。其中 1-3 小节为差分隐私相关，首先介绍了匿名化技术的缺点与差分隐私的优点，然后介绍了两种不同的隐私定义：(ε, δ) −差分隐私和高斯差分隐私，包括它们的定义、噪声机制和组合定理等。第 4 小节为流数据的参数估计相关，主要介绍了流数据和两个在线更新方法。

2.1 传统隐私保护技术

随着大数据分析技术的迅猛发展，研究者及商业公司等迫切地需要从大数据中挖掘出有价值的信息。而这个的前提是，首先要有足够量的、可公开的数据，但是大规模数据拥有者比如医院、政府、大数据公司等，其数据通常又包含个人敏感信息。在对外发布数据集的信息时，不可避免的会涉及到公民的隐私问题。因此，需要考虑一种方法，既能够发布数据实现数据共享，又能够保护个人隐私。

k-匿名(k-Anonymity) (Sweeney, 2002)是由 Latanya Sweeney 提出的一种对数据匿名化的技术。匿名技术是隐私保护的一种重要技术手段，通过对原始数据集进行匿名化后发布，如果出现在匿名数据集中的任意一条记录，都无法与出现在数据集中的其他至少$k-1$条记录区分开来，那么该技术提供k-匿名保护。

如图 2-1，医疗条件(Condition)为敏感变量，即不允许攻击者知道任何一条记录的该属性值；邮政编码(Zip Code)、年龄(Age)、国籍(Nationality)为非敏感变量。让属性集合{邮政编码，年龄，国籍}作为该数据集的准标识符（准标识符指可以和外部表连接来识别出个体的最小属性集），右图是左图匿名化后的结果，该匿名数据集满足 4-匿名，即任意一种准标识符组合的值在表中都至少有四条记录。

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

图 2-1 匿名化例子

一般来讲，由于做了匿名化处理，k-匿名能够在一定程度上保护隐私，但它无法抵御同质性攻击(homogeneity attack)和背景知识攻击(background knowledge attack)(Machanavajjhala 等, 2007)。同质性攻击指在k-匿名的数据表中，某个k-匿名组内所有记录的敏感属性值都相同，当攻击者获得该组记录时，即可知道该组成员的敏感属性取值，从而导致隐私泄露。比如图 2-1 中 9-12 号记录的敏感属性均为患癌症(Cancer)，那么可以判断该组内人员全患癌症。背景知识攻击指即使k-匿名组内的敏感属性值都不同，但如果攻击者具有较多背景知识，仍然可能以高概率推测出该组内成员的敏感信息。比如攻击者知道患者的邮政编码为 13068 且年龄小于 30，那么患者的信息包含在记录 1，2，3，4 中，由于心脏病发病概率极低，因此攻击者可以以较大概率知道患者患有病毒感染(viral infection)。

作为k-匿名技术的改进，Machanavajjhala 等(2007)提出了l-多样性(l-Diversity)：要求发布的匿名数据表中每个k-匿名组内至少含有l种不同的敏感属性值，那么攻击者推断出某条记录敏感信息的概率低于$1/l$。尽管l-多样性在一定程度上能够抵御这种同质性攻击，但它也具有局限性：一方面，l-多样性对数据要求较高，难以实现；另一方面，它要求保证每个组中敏感值的“多样性”，但并未考虑这些值的语义接近性(Li 等, 2006)。如果敏感变量是工资且某个k-匿名组内工资都较低，那么可以判断该组人员都是低收入群体。作为l-多样性的改进，Li 等(2006)提出 t-接近(t-closeness)，要求数据表中任意一个等价类里敏感属性的取值分布与整个表中敏感属性的取值分布之间的距离不超过阈值 t，从而减少攻击者通过敏感信息的分布进行攻击导致属性泄露的可能性。

此外，上面的匿名算法有一个共同的缺点，就是它们都依赖于攻击者具有的背景知识，如果攻击者具有足够多的背景知识，那么匿名化技术就会失效。因此，需要设计这样一种算法，即使攻击者具备较多背景知识时也能保护隐私。Dwork 等(2006)提出的差分隐私解决了这个问题，差分隐私要求攻击者无法还原用户信息，并且假定了攻击者具备最强的背景知识，下一节中我们将叙述差分隐私的定义及性质。

2.2 差分隐私介绍

2.2.1 差分隐私定义

差分隐私的定义基于数据集之间的距离，对于两个数据集，定义它们的距离为两个数据集中不同的记录的个数。如果两个数据集的记录只有一条不同，我们就称它们是相邻数据集(Neighboring Datasets)。这里的数据集记录只有一条不同，可以是两个数据集大小相等，但有且仅有一条记录不同；也可以是两个数据集大小相差一条，且其余记录均相同。

定义 2.1. (差分隐私) 设M是一个随机化算法，P_m为算法M可能的所有输出值的集合。如果对于任意的两个相邻数据集D、D'和P_m的任意子集S，算法M满足：

$$Pr[M(D) \in S] = exp(\varepsilon) \cdot Pr[M(D') \in S] + \delta, \tag{2.1}$$

那么称算法M满足(ε, δ) −差分隐私，或者$(\varepsilon, \delta) - DP$。特别的，如果$\delta = 0$那么称算法$M$满足$\varepsilon - DP$，其中$\varepsilon$和$\delta$被称为隐私预算。

这里的随机化算法指算法将输入的数据集D映射为一个概率分布，相当于为查询结果引入了随机性。直观上，如果算法的输出是一个随机变量，那么攻击者便无法得知查询的真实值。一般来讲，为确保良好的隐私保护效果，隐私预算ε和δ可以取到较小的值，从式(2.1)可以看出：ε和δ越小，差分隐私算法在相邻数据集上的输出差异越小，攻击者越难通过两个数据集上的输出来区分它们。

一方面，差分隐私算法要求数据集中的任意一条记录对查询结果的影响不能太大，也即对任意相邻数据集，都不能根据算法输出对它们做出明确区

分，从攻击者的角度，差分隐私很自然地能通过假设检验来定义，因为攻击者旨在通过算法输出Y区分相邻数据集D和D'。考虑以下假设检验：

H_0：Y是数据集D的输出　vs　H_1：Y是数据集D'的输出，

则差分隐私定义实际上等价于对假设检验犯两类错误概率的约束(Kairouz 等, 2015)。另一方面，差分隐私是基于“数据失真”的：差分隐私机制将查询结果随机化，以一定程度上降低参数的可用性来换取隐私保护，下一节中将具体介绍这种“随机化”方法。

2.2.2 隐私机制

差分隐私关心如何在发布数据的同时又能保护隐私，发布数据的方式一般是返回用户对数据库的查询。假设要发布的统计量是$f(D)$，差分隐私在真实数据上加一个满足一定分布的随机变量（也叫做噪声）来保护数据：$M(D) = f(D) + \xi$。这个噪声不能太小，否则难以保护隐私；同时也不能太大，否则会影响数据的可用性。Dwork (2011)提出敏感度的概念，敏感度和隐私预算决定了噪声的分布和大小。

数据库查询分为数值型查询和非数值型查询，其中数值型查询的返回值通常是连续型变量。针对数值型查询，使用的差分隐私机制包括 Laplace 机制和高斯机制，而针对非数值型查询，一般使用指数机制。

定义 2.2. (全局敏感度) 一个查询函数$f: D \to R$的全局敏感度定义为：

$$\triangle f = \max_{D,D} \|f(D) - f(D')\|_1, \tag{2.2}$$

其中D和D'是相邻数据集，$\|\cdot\|_1$是一范数距离。

定义 2.3. (Laplace 分布) 位置参数为 0 的拉普拉斯分布的密度函数为：

$$Lap(x) = \frac{1}{2b} exp\left(-\frac{|x|}{b}\right). \tag{2.3}$$

差分隐私添加噪声的方式之一是在查询结果上添加满足 Laplace 分布的噪声，其中尺度参数b与查询函数的敏感度有关，如下我们使用$Lap(b)$表示服从位置参数为 0、尺度参数为b的 Laplace 分布的噪声。

定理 2.4. (Laplace 机制) 设查询函数$f: D \to R$是作用在数据库D上的函数，则满足(2.4)式的机制M满足$\varepsilon - DP$，其中$\triangle f$是l_1敏感度。

$$M(D) = f(D) + Lap\left(\frac{\triangle f}{\varepsilon}\right). \tag{2.4}$$

Laplace 机制在查询结果上添加参数为$\triangle f/\varepsilon$的 Laplace 噪声，使得算法满足$\varepsilon - DP$，隐私预算ε越小，Laplace 分布越均匀，对数据的保护效果越好，但可能引入过大噪声导致查询结果准确性变差。Laplace 机制提供了严格的$\varepsilon - DP$，某些情况下可能并不实用，使用高斯机制可以实现它的松弛版本$(\varepsilon, \delta) - DP$。

定义 2.5. (l_2敏感度) 查询函数$f: D \to R$的l_2敏感度定义为：

$$\triangle f = \max_{D,D'} \|f(D) - f(D)\|_2, \tag{2.5}$$

其中D和D'是相邻数据集，$\|\cdot\|_2$是二范数。

定理 2.6. (高斯机制) 设查询函数$f: D \to R$是作用在数据库上的函数，则对任意$\delta \in (0,1), \sigma > \frac{\sqrt{2ln(1.25/\delta)}\cdot\triangle f}{\varepsilon}$，满足(2-3)式的机制$M$满足$(\varepsilon, \delta) - DP$，其中$\triangle f$是$l_2$敏感度。

$$M(D) = f(D) + N(0, \sigma^2). \tag{2.6}$$

例 2.7. 假设有数据集$D = (x_1, x_2, \cdots, x_n)$，现在想要对外发布该数据集的均值$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$，假设攻击者已经知道了除$x_j$以外的其他数据，如果发布了均值，那么攻击者可以通过均值计算出$x_j = n \cdot \bar{x} - \sum_{i \neq j} x_i$。采用差分隐私的方法，可以对外发布：

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i + \xi, \tag{2.7}$$

其中ξ随机抽样自正态分布，因为不知道ξ具体的值，即便知道其余记录，攻击者也没法推导x_j。

上述两个机制一般用于处理数值型查询，对非数值型查询而言，查询结果是离散数据$\{r_1, r_2, \cdots, r_n\}$中的一个元素，这种情况下通常使用指数机制。指数机制保护隐私的方式不是输出一个固定值，而是以一定的概率返回不同值。

定理 2.8. (指数机制) 设D是输入数据集，$R = \{r_1, r_2, \cdots, r_n\}$是所有可能的输出值的集合，定义打分函数$q(D, r), r \in R$为输出值$r$的得分。以下设$D$和$D'$是相邻数据集，敏感度定义为：$\Delta q = \max_r \max_{D,D'} |q(D, r) - q(D', r)|$，那么满足(2.8)式的机制$M$满足$(\varepsilon, 0) - DP$。

$$M(D,q,r_i)=\frac{exp(\varepsilon\cdot q(D,r_i)/(2\cdot\Delta q))}{\sum_{r\in R}exp(\varepsilon\cdot q(D,r)/(2\cdot\Delta q))},i=1,2,\cdots,n. \quad (2.8)$$

以上三种隐私机制本质上都是在输出结果上加扰动，扰动的大小由隐私预算和敏感度决定，隐私预算取决于我们想要达到多好的隐私保护效果，敏感度则取决于数据集，如果一个样本的变化对查询结果的影响越大，敏感度也越大，攻击者越容易根据输出值区分数据集，此时需要加入更多的噪声。值得一提的是，虽然加扰动确实一定程度上保护了隐私，但是攻击者也获取了关于数据集的部分信息，一个问题是在经过多次查询后差分隐私技术是否还能确保隐私。

2.2.3 组合定理

正如上文所提到的，差分隐私实际上是在数据可用性和隐私保护效果之间做一个权衡，为了实现隐私保护可以牺牲部分数据可用性，但是如果数据完全不可用那发布数据也就没有了意义，因此每次发布数据必然会在一定程度上降低隐私。我们假设这样一种情况：现在有一个数据库记录了用户是否患有糖尿病，攻击者想要知道某个用户张三是否患有糖尿病，他可以对数据库做两次查询。第一个查询：数据库中有多少人患有糖尿病？第二个查询：数据库中除开张三以外的人有多少患有糖尿病？通过对比两次查询的结果他可以知道张三是否患有糖尿病。

即便上述例子中对查询结果添加了噪声，如果攻击者反复查询同一个问题并对查询结果取平均，平均值依然会比较接近真实值。事实证明，如果攻击者具有较多的背景知识，他可以通过查询精心设计的多个问题来获取数据库的某条记录。值得注意的是，多次查询同一个数据库时，由于发布了更多信息，隐私会降低是很自然的，但是我们希望隐私以一种“平缓”和可控的方式降低，而不是像匿名化技术在多次查询下变得完全无法保证隐私(McSherry, 2009)。

如果给定一个总的隐私预算，每次查询都会消耗部分预算，预算耗尽则不能再保护隐私。针对多个差分隐私机制的组合，McSherry(2009)提出了差分隐私的两个重要组合定理，分别是顺序分解定理(Sequential Composition)和平

行分解定理(Parallel Composition)。顺序分解定理对应对同一数据集的多次查询，平行分解定理对应与完全不相交数据集的多次查询。

定理 2.9. (顺序分解定理) 假设对于$i = 1,2,\cdots,n$，差分隐私机制M_i满足$(\varepsilon_i, \delta_i)-$差分隐私，且作用于同一数据集，那么它们的组合满足$(\sum_{i=1}^{n}\varepsilon_i, \sum_{i=1}^{n}\delta_i)-$差分隐私。

定理 2.10. (平行分解定理) 假设对于$i = 1,2,\cdots,n$，差分隐私机制M_i满足$(\varepsilon_i, \delta_i)-$差分隐私，且作用于两两不相交数据集，那么它们的组合满足$\left(\underset{1\le i\le n}{max}\,\varepsilon_i, \underset{1\le i\le n}{max}\,\delta_i\right)-$差分隐私。

隐私机制的组合定理能够帮助实现更复杂的隐私算法，对于顺序分解定理，差分隐私效果随着组合机制的个数线性降低；而对于平行分解定理，组合机制的隐私效果取决于隐私效果最差的机制。尽管顺序分解定理对于隐私算法的设计至关重要，但需要平行分解定理才能获得更为良好的性能。

在组合不同隐私机制时，考虑这样一种情况，攻击者可以根据上一次查询结果来设置新的查询，也就是说对数据库的多次查询可以是相关的，攻击者可以自适应地改变下一次查询的数据库、参数等。针对这种情况，Dwork 和 Roth(2014)提出了更强的组合定理。

定理 2.11. (Advanced Composition) 假设一列机制M_i满足$(\varepsilon, \delta)-DP$，那么它们的组合机制满足$(\varepsilon', k\delta+\delta')-DP$，其中：

$$\delta' > 0, \varepsilon' = \sqrt{2kln(1/\delta')}\cdot\varepsilon + k\varepsilon(e^{\varepsilon}-1). \tag{2.9}$$

当$\varepsilon \to 0$时，(2.9)式中的$k\varepsilon(e^{\varepsilon}-1)\to 0$，$\varepsilon'$相当于以$O(\sqrt{k})$增加，比起顺序分解定理中的$O(k)$，Advanced Composition 节省了很多隐私预算，使得实现更为复杂的隐私算法时可以使用更小的隐私预算。

2.3 高斯差分隐私

2.3.1 定义

由于Dwork一开始提出的$\varepsilon-DP$实际上比较严格，在实际应用中需要很多的隐私预算，为了算法的实用性，有关差分隐私的部分工作实际上也围绕着

如何定义“松弛版本”的差分隐私，比如截断的集中差分隐私(truncated concentrated differential privacy)(Bun等, 2018)和基于$Rényi$散度的$(\alpha,\varepsilon)-RDP$(Mironov, 2017)等。尽管围绕着这些松弛版本隐私定义的相关工作已经取得了一定的成果，但仍有一些地方希望改进，比如这些差分隐私的概念不再具有假设检验的解释(Dong等, 2019)，而基于假设检验的定义对于解释差分隐私来说是很方便的。

基于假设检验的思想，董金硕和苏炜杰等提出了一种新的差分隐私技术来刻画隐私。假设P,Q分别是隐私机制M在相邻数据集D,D'上的输出的概率分布，目标是根据P,Q来区分数据集：

H_0：基础数据集是D vs H_1：基础数据集是D'。

考虑拒绝规则$0\le\varphi\le 1$，两类错误概率可以表示为：

$$\alpha_\varphi = E_P[\varphi], \beta_\varphi = 1-E_Q[\varphi]. \tag{2.10}$$

定义 2.12. (权衡函数(trade-off function)) 对任意两个定义在同一空间上的概率分布P,Q，定义权衡函数$T(P,Q):[0,1]\to[0,1]$满足：

$$T(P,Q)(\alpha) = inf\{\beta_\varphi : \alpha_\varphi \le \alpha\}, \tag{2.11}$$

其中下确界遍历所有的拒绝规则。

权衡函数给出了控制第一类错误的前提下能达到的最小的第二类错误，这实际上反映了区分概率分布P,Q的困难程度。权衡函数越小，上述假设检验在任意拒绝规则下犯两类错误的概率都越小，区分相邻数据集变得更加容易。随后他们给出了权衡函数的充要条件。

命题 2.13. 函数$f:[0,1]\to[0,1]$是权衡函数当且仅当f是连续、非增的凸函数，且满足对任意$x\in[0,1]$，$f(x)\le 1-x$。

定义 2.14. $(f-DP)$ 假设f是权衡函数，称隐私机制M满足$f-DP$，当且仅当对任意相邻数据集D,D'，成立：

$$T\big(M(D),M(D')\big)\ge f. \tag{2.12}$$

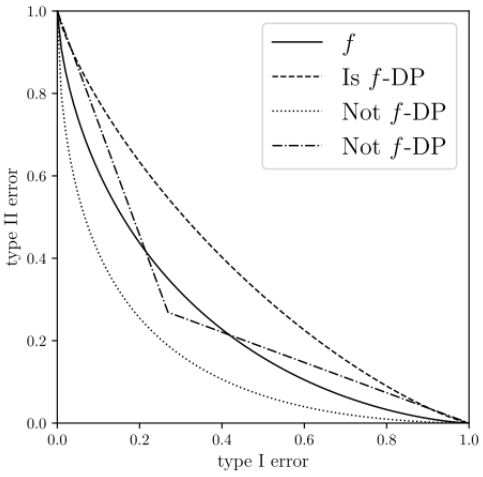

图 2-2 f-DP示例

如图 2-2 所示，$f-DP$的定义要求$T\big(M(D),M(D')\big)(\alpha)\geq f(\alpha)$对任意$\alpha\in[0,1]$成立，如果某个隐私机制满足$f-DP$，则两类错误的曲线必定在$f$上方。由于$f$是权衡函数，存在两个概率分布$P$、$Q$，使得$f=T(P,Q)$，那么式(2.12)也要求，基于区分任意两个相邻数据集的难度都至少与区分P,Q的难度相当。相比而言，传统的$(\varepsilon,\delta)-DP$只使用了两个参数来衡量隐私，$f-DP$则是使用了一个函数f，这种定义对隐私提供了更为完整的刻画。

定义 2.15. (高斯差分隐私) 用$G_u\coloneqq T\big(N(0,1),N(\mu,1)\big)$表示两个正态分布的权衡函数，如果一个隐私机制$M$是$G_\mu-DP$的，也即对任意相邻数据集$D,D'$，满足：

$$T\big(M(D),M(D')\big)\geq G_\mu, \tag{2.13}$$

那么称M是μ−高斯差分隐私$(\mu-GDP)$的。

如果说$f-DP$的定义是基于区分两个普通的概率分布，那么$\mu-GDP$则是基于区分两个单位方差的正态分布。高斯差分隐私实际上是$f-DP$的一个特例，μ越大，权衡函数G_μ越小，区分两个正态分布越容易。高斯差分隐私有着许多优良的性质，该隐私定义可以由一个参数衡量，即单位方差的正态分布的均值，更易于描述。更重要的一点是，$f-DP$在极限情况下会收敛到$\mu-GDP$，正如Dong等(2019)提到的那样：$\mu-GDP$在$f-DP$中的重要性，可以类比正态分布在一般概率分布中的重要性。

2.3.2 隐私机制和组合定理

正如上文所提到的，任何一个差分隐私定义都必须解决多个机制的组合问题，因为这是组成复杂隐私机制的基础。高斯差分隐私的优点不仅在于它提供了更为简洁的加噪声机制，也在于它在组合多个隐私机制时有更优美的性质。如下我们将简述高斯差分隐私的隐私机制和组合定理。

定理 2.16. (高斯机制) 设$\theta(S)$是定义在数据集上的统计量，D, D'是相邻数据集，定义θ的敏感度为$sens(\theta) = \underset{D,D'}{max}|\theta(D) - \theta(D')|$，隐私机制$M(D) = \theta(D) + \xi$，其中$\xi \sim N(0, sens(\theta)^2/\mu^2)$，则$M$满足$\mu - GDP$。

证明：设$\sigma^2 = sens(\theta)^2/\mu^2$，$M$在相邻数据集$D, D'$上的输出服从正态分布：

$$M(D) \sim N(\theta(D), \sigma^2), M(D') \sim N(\theta(D'), \sigma^2). \tag{2.14}$$

由敏感度定义可知$|\theta(D) - \theta(D')|/\sigma \leq \mu$，再回顾$f - DP$的定义，可得：

$$T\big(M(D), M(D')\big) = T\big(N(\theta(D), \sigma^2), N(\theta(D'), \sigma^2)\big) = G_{|\theta(D)-\theta(D')|/\sigma} \geq G_\mu,$$

故机制M满足$\mu - GDP$。

定理 2.17. (GDP组合定理) 设一列机制M_i满足$\mu_i - GDP, i = 1,2,\cdots n$，其中每个隐私机制的输入都可以参考之前隐私机制的输出，且这些隐私机制作用于同一数据集，则他们的组合满足$\sqrt{\mu_1^2 + \mu_2^2 + \cdots + \mu_n^2} - GDP$。

在高斯差分隐私定义下，k个作用于相同数据集的隐私机制的组合需要的隐私预算为$\sqrt{k}$倍，相比起$(\varepsilon, \delta) - DP$的顺序分解定理，节约了较多的隐私预算。除此之外，$f - DP$和$(\varepsilon, \delta) - DP$存在一定的等价关系，并且能够相互转化，尽管形式并不是很简便，但在某些问题上，依然可以考虑将$f - DP$转化为$(\varepsilon, \delta) - DP$来处理，例如本文 3.2 节中将考虑如何在两种差分隐私定义下实现相同的隐私保护效果。

2.4 流数据与在线更新

2.4.1 流数据

最近几年，大数据一词被越来越多地提及，人们用它来描述信息爆炸时代产生的海量数据。大数据时代，我们得以将每天产生的数据收集、整理和利

用起来，这些数据可能是静态数据，例如企业构建数据仓库储存的历史数据，也可能是实时获取的数据——流数据。流数据(streaming data)是一组顺序、大量、快速、连续到达的数据序列，其中“流”通常是指持续生成的数据流，例如网站生成的日志、用户的网购数据和社交网站信息等。由于诸如 Apache Spark 等现代强大计算平台的可用性，流数据的统计分析最近在大数据分析的新兴领域引起了相当大的关注。

批处理是大数据的数据处理方式之一，它主要操作大容量静态数据集，并在计算完成后返回结果。但是对于流数据的参数估计问题来说，批处理的方式显然是不适合的。一方面，流数据持续生成，且当前数据使用后便不再提取；另一方面，即使可以储存流数据，长时间的积累使得数据总量愈发庞大，对数据的存储和计算机的性能带来了严峻挑战。这就促使去寻找一种实时更新流数据参数的方法。

用在线更新(online updating)的思想去处理流数据的参数估计问题是一个不错的想法，在线更新不储存原始的数据，只使用以往数据的统计量和当期的数据进行更新。例如，可以只使用以往数据的均值$\bar{x}$、数据量n和当期数据$\{x_{k1}, x_{k2}, \cdots x_{kn_k}\}$来获得整个数据的均值：

$$\bar{x}' = \frac{1}{n + n_k}\left(n \cdot \bar{x} + x_{k1} + x_{k2} + \cdots + x_{kn_k}\right). \tag{2.15}$$

2.4.2 流数据的在线更新方法

针对流数据下线性回归模型的参数估计问题，Schifano 等(2016)给出了在线最小二乘估计方法，利用前$k-1$批数据子集构造子集k中当前数据的先验分布，给出了第k批数据的在线更新公式。

假设数据集$\{(x_i, y_i), i = 1,2, \cdots, N\}$来自线性回归模型$y_i = x_i^T\beta + \epsilon_i$，其中$\epsilon_i$独立同分布且$\epsilon_i \sim N(0, \sigma^2)$，$x_i$和$\beta$为$p$维向量。令$X = (x_1, x_2, \cdots x_N)^T$，$\mathrm{y} = (y_1, y_2, \cdots y_{\mathrm{N}})^T$，假设矩阵$X$满秩，则$\beta$的最小二乘估计量为：

$$\hat{\beta} = (X^T X)^{-1} \cdot X^T y. \tag{2.16}$$

在在线更新的设置中，整个流数据X不是同时可用，假设每个时间点k可用的数据为$(X_k, y_k), k = 1,2, \cdots K$，其中$X_k$为$n_k \times p$维数据矩阵，假设$X_k$满秩，

则基于第k批数据的最小二乘估计量为：$\hat{\beta}_{n_k,k} = \left(X_k{}^T X_k\right)^{-1} \cdot X_k{}^T y_k$。利用分而治之的思想(Lin 和 Xi, 2011)，式(2.16)中的$\hat{\beta}$也可表示为：

$$\hat{\beta} = \left(\sum_{k=1}^{K} X_k{}^T X_k\right)^{-1} \cdot \sum_{k=1}^{K} X_k{}^T X_k \, \hat{\beta}_{n_k,k}. \tag{2.17}$$

Lin 和 Xi(2011)指出，式(2.17)已经具有在线更新的形式，第k批数据使用后只需储存$\left(X_k{}^T X_k, \hat{\beta}_{n_k,k}\right)$，也即总共只需储存$K\left(p(1+p)\right)$个数。Schifano 等进一步思考以一种递归的方式更新$\hat{\beta}_k$，令$V_k = \sum_{l=1}^{k} X_l^T \, X_l$，则对于$k = 1,2,\cdots$，第$k$批数据到达后可根据式(2.18)更新参数：

$$\hat{\beta}_k = \left(X_k{}^T X_k + V_{k-1}\right)^{-1}\left(X_k{}^T X_k \hat{\beta}_{n_k,k} + V_{k-1}\hat{\beta}_{k-1}\right), \tag{2.18}$$

其中$\hat{\beta}_0 = 0$，$V_0 = 0_p$是$p \times p$维的 0 矩阵。该方法利用第k批数据(X_k, y_k)计算最小二乘估计量$\hat{\beta}_{n_k,k}$，结合以往数据的统计量V_{k-1}与上一轮的参数值$\hat{\beta}_{k-1}$对参数做更新。此外，Schifano 等(2016)给出了误差平方和(SSE)的在线更新公式并指出，基于这些在线更新参数值，可以很容易地进行回归参数的 t 检验。

Luo 和 Song(2018)提出了流数据下极大似然估计量(MLE)的增量更新算法，该算法在每批数据到达后迭代地逼近增量估计方程的解。具体来说，MLE 估计量为满足以下估计方程的解：$\sum_{i \in D} U(y_i; x_i, \beta) = 0$，其中$D$为数据集。以下假设流数据集为$D_b, b = 1,2,3,\cdots$，$U_b(D_b; \beta) = \sum_{i \in D_b} U(y_i; x_i, \beta)$为数据集$D_b$的得分函数，其负 Hessian 阵为$H_b(D_b; \beta) = -\nabla_\beta U_b(D_b; \beta)$。当第$b$批数据$D_b$到达后，目的是利用这批数据将参数$\tilde{\beta}_{b-1}$更新到$\tilde{\beta}_b$，即寻找以下估计方程的解：

$$\sum_{j=1}^{b-1} U_j\left(D_j; \tilde{\beta}_b\right) + U_b\left(D_b; \tilde{\beta}_b\right) = 0. \tag{2.19}$$

上式第一项在$\tilde{\beta}_{b-1}$处泰勒展开且去掉二阶余项可得：

$$\sum_{j=1}^{b-1}\left[U_j\left(D_j; \tilde{\beta}_{b-1}\right) + H_j\left(D_j; \tilde{\beta}_j\right)\left(\tilde{\beta}_{b-1} - \tilde{\beta}_b\right)\right] + U_b\left(D_b; \tilde{\beta}_b\right) = 0. \tag{2.20}$$

$\tilde{\beta}_{b-1}$是前$b-1$批数据的MLE估计量，因此$\sum_{j=1}^{b-1} U_j\left(D_j; \tilde{\beta}_{b-1}\right) = 0$，结合式(2.20)可得：

$$\sum_{j=1}^{b-1} H_j\left(D_j; \tilde{\beta}_j\right)\left(\tilde{\beta}_{b-1} - \tilde{\beta}_b\right) + U_b\left(D_b; \tilde{\beta}_b\right) = 0. \tag{2.21}$$

上式被称为增量估计方程，用牛顿法迭代求解上式的值，同时为加速迭代用$\tilde{\beta}_{b-1}$代替求逆矩阵中的$\tilde{\beta}_b^{(r)}$可得：

$$\tilde{\beta}_b^{(r+1)} = \tilde{\beta}_b^{(r)} + \left[\widetilde{H}_{b-1} + H_b\left(D_b; \tilde{\beta}_{b-1}\right)\right]^{-1} \cdot \widetilde{U}_b{}^{(r)}. \tag{2.22}$$

其中$\widetilde{H}_{b-1} = \sum_{j=1}^{b-1} H_j\left(D_j; \tilde{\beta}_j\right)$, $\widetilde{U}_b{}^{(r)} = \widetilde{H}_{b-1}\left(\tilde{\beta}_{b-1} - \tilde{\beta}_b^{(r)}\right) + U_b\left(D_b; \tilde{\beta}_b^{(r)}\right)$。也就是说，式(2.22)只使用了当前批数据$D_b$和往期数据的汇总统计量$\widetilde{H}_{b-1}$即可将参数$\tilde{\beta}_{b-1}$更新到$\tilde{\beta}_b$。Luo 和 Song(2018)随后提出了相应架构实现这种算法，并在随后的实验中证实了该算法比其他在线更新算法有更快的更新速度和更小的估计偏差。

3．流数据估计的差分隐私算法

本章我们考虑差分隐私保护下的流数据参数估计问题，并提出了相应的算法。第一节简要介绍随机梯度下降算法和与之相结合的差分隐私算法；第二节考虑到在梯度上添加噪声会影响到参数估计的准确性，为提高参数可用性，首先会比较两种差分隐私定义的高斯机制，以选择一种添加更小噪声的隐私机制；第三节根据流数据的特点给出本文算法，并根据组合定理分析算法的隐私保护效果。

3.1 差分隐私 SGD 算法

3.1.1 梯度下降算法介绍

大部分的机器学习算法的本质都是建立模型，再通过优化目标函数（或损失函数）来优化模型，损失函数衡量了模型对数据集的拟合程度，一般来说，损失函数越小，模型准确度越高。在目标模型损失函数比较复杂的时候，无法直接求解最小值和获得参数显示表达式，一般采用梯度下降算法来最小化损失函数。

梯度下降算法(Gradient decent)是一种简单且易于实现的算法，它通过不断的迭代来逼近极值点，当目标函数是凸函数时，梯度下降法能够逼近全局最优解。假设我们有一个可微的函数$f(x;\theta)$，每次迭代都会使函数沿着当前点的负梯度方向前进一步：

$$\theta^{(k+1)} = \theta^{(k)} - \alpha\frac{\partial \mathrm{f}(x;\theta)}{\partial\theta}, \tag{3.1}$$

其中$\theta^{(k)}$表示第k步迭代的参数值，α为学习率，学习率衡量了每一次迭代函数“前进”的距离。在梯度下降中学习率是一个重要的参数，学习率太小则函数收敛太慢，学习率太大又可能错过函数的极值点导致不收敛，为了更准确地搜索极值，有时也考虑在迭代过程中衰减学习率。式(3.1)会不断进行，直到某两次迭代的梯度变化小于阈值（梯度小于阈值）或者达到最大迭代步数。