

# 论文阅读笔记：差分隐私

纳文琪

2019 年 5 月 29 日

## 1 差分隐私保护及其应用 [1]

### 1.1 引言

**基于分组的隐私模型** “k-anonymity 及其扩展模型被称为基于分组的隐私模型”，这些模型主要存在两个主要的缺陷：

- “并不能提供足够的安全保障，他们总是因新型攻击的出现而需要不断完善”。“出现这一局面的原因在于，基于分组的隐私保护模型的安全性攻击者所掌握的背景知识相关，而所有可能的背景知识很难被充分定义。”
- “这些早期的隐私保护模型无法提供一种有效且严格的方法来证明其隐私保护水平，因此当模型参数改变时，无法对隐私保护水平进行定量分析。”

差分隐私能够解决传统隐私保护模型的两个缺陷。

### 1.2 差分隐私保护模型

**基本思想** “设对数据集  $D$  进行任意操作  $f$ ，得到  $f(D)$ ，如果将  $A$  从  $D$  中删除后，得到的结果仍然为  $f(D)$ ，则可以认为  $A$  的信息并没有因为包含在  $D$  中而产生额外的风险。”

### 1.2.1 基本概念

**隐私保护机制** “对  $D$  的各类映射函数被定义为查询，用  $F = \{f_1, f_2, \dots\}$  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。”

**邻接数据集** “属性结构相同的数据集  $D$  和  $D'$ ，两者的对称差记为  $D \Delta D'$ ，  
若  $|D \Delta D'| = 1$ ，则称  $D$  和  $D'$  为邻接数据集。”

**差分隐私** 设  $P_M$  是算法  $M$  所有可能的输出构成的集合， $S_M$  是  $P_M$  子集，  
若算法满足：

$$Pr[M(D) \in S_M] \leq e^\epsilon \times Pr[M(D') \in S_M] \quad (1)$$

则称算法  $M$  提供  $\epsilon$ -差分隐私保护。

**隐私保护预算** 上式中的  $\epsilon$  称为隐私保护预算，用来控制算法  $M$  在  $D$  和  
 $D'$  上获得相同输出的概率比值，它通常取比较小的值。 $\epsilon$  越小，表示隐私  
保护水平越高，当  $\epsilon$  为 0 时，保护水平最高。

**敏感度** “差分隐私保护可以通过在  $f$  的返回值中加入噪声来实现。敏感  
度是决定加入噪声量大小的关键参数。”

**全局敏感度** 设函数  $f: D \rightarrow R^d$ ，对于邻近数据集  $D$  和  $D'$ ，函数  $f$  的全  
局敏感度定义为：

$$GS_f = \max_{D, D'} \|f(D) - f(D')\|_1 \quad (2)$$

“函数的全局敏感度由函数本身决定。当  $GS$  较大时，必须在函数输出中添  
加足够大的噪声才能保证隐私安全，导致数据可用性较差。”

**局部敏感度** 设函数  $f: D \rightarrow R^d$ ，对于邻近数据集  $D$  和  $D'$ ，函数  $f$  在  $D$   
上的局部敏感度定义为：

$$LS_f(D) = \max_{D'} \|f(D) - f(D')\|_1 \quad (3)$$

“局部敏感度由函数  $f$  及给定数据集  $D$  共同决定。”  $GS$  和  $LS$  之间的关系可以表示为：

$$GS_f = \max_D (LS_f(D)) \quad (4)$$

**平滑上界** “ $LS$  在一定程度上体现了数据集的数据分布特征，直接应用会泄露数据集中的敏感信息。因此， $LS$  的平滑上界被用来与  $LS$  一起确定噪声量的大小。” 对于  $\beta > 0$ ，若函数  $S : D \rightarrow R$  满足  $S(D) \geq LS_f(D)$  且  $S(D) \leq e^\beta \times S(D')$ ，则称  $S$  是  $f$  的  $LS$  的  $\beta$ -平滑上界。

**平滑敏感度** 函数  $f$  的  $\beta$ -平滑敏感度定义为：

$$S_{f,\beta}(D) = \max_{D'} (LS_f(D') \times e^{-\beta|D \Delta D'|}) \quad (5)$$

### 1.2.2 差分隐私保护算法的组合性质

“一个复杂的隐私保护问题，通常需要多次应用差分隐私保护算法，为了保证整个过程的隐私保护水平控制在给定的预算之内，需要合理地将全部预算分配到整个算法的各个步骤中。”

**序列组合性** “设有算法  $M_1, \dots, M_n$ ，其预算分别是  $\epsilon_1, \dots, \epsilon_n$ ，那么对同一个数据集  $D$ ，由这些算法构成的组合算法  $M(M_1(D), \dots, M_n(D))$  提供  $\sum_{i=1}^n \epsilon_i$ -差分隐私保护。” 也就是说，“一个差分隐私保护算法序列构成的组合算法，其提供的隐私保护水平为全部预算的总和”。

**并行组合性** “设有算法  $M_1, \dots, M_n$ ，其预算分别是  $\epsilon_1, \dots, \epsilon_n$ ，那么对于不相交的数据集  $D_1, \dots, D_n$ ，由这些算法构成的组合算法  $M(M_1(D_1), \dots, M_n(D_n))$  提供  $(\max \epsilon_i)$ -差分隐私保护”。此时算法系序列构成的组合算法提供的隐私保护水平取决于算法序列中的保护水平最差者。

### 1.2.3 实现机制

根据差分隐私保护的要求，不同的问题有不同的实现方法，即机制。最基础的两种机制是 Laplace 机制和指数机制。

**Laplace 分布** 两个指数型随机变量之差满足 Laplace 分布。其概率密度为：

$$f(x|\mu, b) = \frac{1}{2b} e^{-\frac{|x-\mu|}{b}} \quad (6)$$

其图像呈尖沙堆状，如图1 所示。

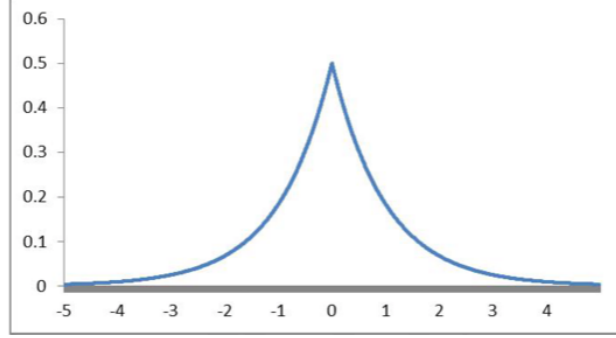


图 1: Laplace 分布概率密度函数

**Laplace 机制** Laplace 机制仅适用于数值型结果的保护，通过向查询结果中加入服从 Laplace 分布的随机噪声来实现差分隐私保护。令设数据集  $D$ ，函数  $f$  的敏感度为  $\Delta f$ ，那么随机算法  $M(D) = f(D) + Y$  提供  $\epsilon$ -差分隐私保护，其中  $Y \sim Lap(\frac{\Delta f}{\epsilon})$  为随机噪声，服从尺度为  $\frac{\Delta f}{\epsilon}$  的 Laplace 分布。

**指数机制** 适用于非数值型查询结果。设随机算法  $M$ ，输出为一实体对象  $r \in Range$ ，函数  $q(D, r) \rightarrow R$  为可用性函数， $\Delta q$  为函数  $q(D, r)$  的敏感度。若算法  $M$  以正比于  $e^{\frac{\epsilon q(D, r)}{2\Delta q}}$  的概率从  $Range$  中选择并输出  $r$ ，那么算法  $M$  提供  $\epsilon$ -差分隐私保护。

### 1.3 基于隐私保护的数据发布 (PPDR)

PPDR 研究的问题是如何在满足差分隐私的条件下保证发布数据或查询结果的精确性。根据实现环境可分为交互式数据发布和非交互式数据发布两种。

### 1.3.1 交互式数据发布

“在交互式环境下，用户向数据管理者提出查询请求，数据管理者根据查询请求对数据集进行操作并将结果进行必要的干扰后反馈给用户，用户不能看到数据集全貌，从而保护数据集中的个体隐私”。具体可描述为：“给定数据集  $D$  和查询集合  $F$ ，需寻求一种数据发布机制，使其能够在满足差分隐私保护的条件下逐个回答  $F$  中的查询，直到耗尽全部隐私保护预算”。“对于  $F$  中的任意  $f$ ，设定一个足够小的实数  $\delta$ ，查询结果精度  $\alpha$  应满足：

$$Pr_{f \in F}[|f(D) - M(f(D))| \leq \alpha] \geq 1 - \delta \quad (7)$$

其中  $f(D)$  是查询结果， $M(f(D))$  是对  $M$  的干扰结果。”

### 1.3.2 非交互式数据发布

“在非交互式环境下，数据管理者针对所有可能的查询，在满足差分隐私的条件下一次性发布所有查询的结果。或者，数据管理者发布一个原始数据集的“净化”版本，这是一个不精确的数据集，用户可对该版本的数据集自行进行所需的查询操作”。可表述为：“给定  $D$  和  $F$ ，需寻求一个数据发布机制，使其能够在满足差分隐私保护的条件下一次性回答  $F$  中所有的查询”。

## 2 本地化差分隐私研究综述 [2]

### 2.1 引言

”传统的差分隐私技术将原始数据集中到一个数据中心，然后发布满足差分隐私的相关统计信息，我们称之为中心化差分隐私 (Centralized Differential Privacy) 技术。因此，中心化差分隐私对于敏感信息的保护始终基于一个前提假设：可信的第三方数据收集者，在不可信第三方数据收集者的场景下，本地化差分隐私 (Local Differential Privacy) 技术应运而生”

”本地化差分隐私技术继承自中心化差分隐私技术，同时扩展出了新的特性，使该技术具备两大特点：1) 充分考虑任意攻击者的背景知识，并对隐私

保护程度进行量化;2) 本地化扰动数据, 抵御来自不可信第三方数据收集者的隐私攻击。”

”对本地化差分隐私的研究和应用, 主要考虑以下两个方面问题:(1) 如何设计满足  $\epsilon$ -本地化差分隐私的数据扰动算法, 以保护其中的敏感信息;(2) 数据收集者如何对查询结果进行求精处理, 以提高统计结果的可用性。”

**研究方向** ”本地化差分隐私技术的研究方向包括扰动机制的研究以及统计数据的发布. 本地化差分隐私的扰动机制主要包括随机响应、信息压缩和扭曲两种.”本地化差分隐私的统计数据发布包括: 单值频数发布、多值频数发布和均值发布。

## 2.2 基于本地化差分隐私的数据保护框架

”交互式和非交互式数据保护框架的最大区别在于输出结果之间的关联性.1) 交互式框架适用于最终输出结果与前  $i$  个输出有依赖关系的情形.2) 非交互式框架适用于前后的输入输出之间无依赖关系的情形。”

## 2.3 基于本地化差分隐私的单值频数统计

”单值频数统计是指每个用户只发送一个变量取值的情形, 用户将数据发送给数据收集者后, 数据收集者根据已有的或统计得到候选值列表, 统计其中每一个候选值的频数并进行发布。”

**RAPPOR 方法** 变量以字符串表示, 先将变量通过 Bloom Filter 表示成长度为  $h$  的向量, 然后对向量进行一次随机扰动, 得到 PRR, 再进行第二次随机扰动得到 IRR, 然后将扰动结果发送给数据收集者。”RAPPOR 方法的主要存在两个方面的缺陷:1) 用户和数据收集者之间的传输代价比较高, 即每个用户需要传输长度为  $h$  的向量给数据收集者;2) 数据收集者需预先采集候选字符串列表, 以进行频数统计。”

## 2.4 未来研究挑战

” 现有研究主要集中在两个方面:1) 理论上, 设计满足本地化差分隐私的保护机制;2) 方法上, 对频数和均值两种统计结果进行保护.”

**复杂数据类型的本地化差分隐私保护** ” 目前, 本地化差分隐私保护技术的研究主要还是针对简单的数据类型, 然而, 简单的数据类型对于当下空前的数据分析需求而言, 还远远不够. 以键值对数据 (Key-value database) 和图数据 (Graph database) 为例”

**不同查询和分析任务的本地化差分隐私保护** ” 聚类分析是为了把性质相近的数据归入一类, 典型应用如客户群体发现、社区发现等; 频繁模式挖掘的目的则是找出数据集中的频繁项集, 进而发现模式规律, 典型应用如搜索日志分析、购买行为分析等. 聚类分析和频繁模式挖掘都是数据分析中的常用方法, 其中涉及的查询包括计数查询、均值查询、范围查询和最值查询等. 本地化差分隐私下, 不仅要求能够支持不同的查询类型, 还要求扰动后的数据能够同时支持多种不同的查询. 与中心化差分隐私不同, 本地化差分隐私通过抵消添加在数据中的正向和负向噪声来得到比较准确的统计结果, 但就单条数据记录而言, 通常扰动前后数据的偏差较大, 这也进一步加大了针对不同查询的难度.”

**基于本地化差分隐私的高维数据发布** ” 本地化差分隐私下的高维数据发布主要考虑三个方面的问题:1) 如何在一定隐私预算内衡量属性之间的关联性, 从而进行降维处理;2) 如何设计推理模型, 最小化边缘分布到联合分布的近似误差, 提高数据可用性;3) 如何控制高维数据在用户和数据收集者之间的通信代价.”

## 3 Privacy-preserving deep learning[3]

### 3.1 Introduction

本文提出一种协调深度学习系统, 用于权衡效用和隐私. 系统运行多方使用自己的输入数据合作训练一个神经网络模型, 而不需要共享. 此技术

的关键创新是，在训练期间选择性共享模型参数。参数共享运行多个参与者在没有显式的输入数据的共享的情况下，互相利用其它人的模型计算结果。

## 3.2 相关工作

### 3.2.1 Privacy in ML

当多方利用自有数据协同执行机器学习任务时，基于安全多方计算(secure multi-party computation, SMC) 的技术帮忙我们计算的中间步骤的隐私。

## 3.3 Distributed Selective SGD

本文方法的核心是分布式的、协作的深度学习协议，它基于以下观察结果：

- 梯度下降的过程中，对不同参数的更新本质上是独立的；
- 不同的训练数据集作用于不同的参数；
- 不同的特征对目标函数的作用并不相等。

在进行选择性 SGD 时，学习器选择一部分参数进行更新，可选择梯度较大的那部分参数。

## 3.4 系统架构

**概述** 系统由一个管理全局参数的参数服务器和多个局部训练器（参与者）组成，他们之间通过参数选择协议交换参数。参数交换协议允许参与者独立地优化参数、避免过拟合。

**Local training** 每个参与者都可以在本地训练神经网络的参数。假设本地维护  $w^{(i)}$  个参数。训练过程如下：

1. 将  $\theta_d \times |w^{(i)}|$  个参数下载并更新到本地，并不下载更新全部参数；
2. 运行一轮 SGD；
3. 计算  $\Delta w^{(i)}$ ，即新旧参数之间的差值；



4. 选择性地上传最多  $\theta_u \times |w^{(i)}|$  个参数到服务器。选择方法有：

- 最大值选择：选择最大的几个值进行上传；
- 带阈值的随机选择：从大于阈值  $\tau$  的值中随机选择上传；

另外，上传  $\Delta w^{(i)}$  的值之前，这些值将被截断在  $[-\gamma, \gamma]$  之间，以避免这些值泄露训练数据的太多信息。

**为什么 DSSGD 能行？** 这主要是因为学习过程的随机性。在训练过程中随机更新局部参数会增加局部 SGD 的随机性。而由异步参数更新导致的随机性对于精确地训练神经网络非常有效。

Parameter exchange protocol 交换协议有三种类型：

- 轮询。每个参与者按固定顺序下载、训练并更新一部分参数，然后下一个参与者继续；
- 随机。所有参与者同时随机下载、训练并更新一部分参数，但参数的读取过程是加锁的，也就是具有原则性（atomic）；
- 异步。同随机一样，但并不加锁。

### 3.5 Evaluation

**数据集** 实验选用的数据集是 MNIST 和 SVHN

**神经网络架构** 分别采用 MLP 和 CNN

**实验设置**

**选择性 SGD 的实验结果** SSGD 的实验得到了以下结论：

- SSGD 可以获得与 SGD 相同的精度；
- SGD 和 SSGD 在总体上是类似的；
- SSGD 甚至能够取得比 SGD 更高的精度，这是因为 SSGD 类似于给 SGD 加上了 Dropout。

DSSGD 的实验结果 结果表明：

- 任何形式的协作都可以获得比独立学习更高的精度；
- 参与者的数量比共享参数的比例对精度的影响更小。

## 3.6 Privacy

### 3.6.1 防止直接泄露

**训练过程** 在此过程中，由于各自的数据都不共享，所以不存在泄露。

**模型使用过程** 此过程中也不需要用到训练数据，因此也不存在泄露。

### 3.6.2 防止间接泄露

## 4 Deep learning with differential privacy[4]

### 4.1 Introduction

深度学习模型不应该暴露数据集中的隐私信息，出于这个考虑，本文提出了一种学习算法，以较小的隐私预算训练非凸、规模巨大的深度网络。

### 4.2 Our Approach

#### 4.2.1 差分隐私 SGD 算法

保护 SGD 过程中的隐私，其实就是对梯度进行隐私保护。本文的方法是首先根据限界对梯度进行修剪，然后再加上噪声。算法如下。

---

**Algorithm 1** Differentially private SGD (Outline)

---

**Input:** Examples  $\{x_1, \dots, x_N\}$ , loss function  $\mathcal{L}(\theta) = \frac{1}{N} \sum_i \mathcal{L}(\theta, x_i)$ . Parameters: learning rate  $\eta_t$ , noise scale  $\sigma$ , group size  $L$ , gradient norm bound  $C$ .  
**Initialize**  $\theta_0$  randomly  
**for**  $t \in [T]$  **do**  
    Take a random sample  $L_t$  with sampling probability  $L/N$   
    **Compute gradient**  
    For each  $i \in L_t$ , compute  $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t} \mathcal{L}(\theta_t, x_i)$   
    **Clip gradient**  
     $\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C})$   
    **Add noise**  
     $\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L} (\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}))$   
    **Descent**  
     $\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$   
**Output**  $\theta_T$  and compute the overall privacy cost  $(\varepsilon, \delta)$  using a privacy accounting method.

---

图 2: Differentially private SGD

**Norm clipping** 由于并没有关于梯度大小的先验界限，因此算法中使用  $L_2$  范数进行修剪。在算法中，如果  $\|g\|_2 \leq C$ ，则  $g$  将被保护，如果  $\|g\|_2 > C$  则  $g$  将会被范数  $C$  缩放。

**Lots** 在训练时，一个数据集一般被分为不交叉的多个 batch，而此处的 lot 则指的是每次从整个数据集中随机取  $L$  条数据，每条数据被取出的概率是  $q = L/N$ ，类似于，batch 是不放回抽样，lot 是放回抽样。

**Privacy accounting** 差分隐私 SGD 需要计算总体隐私成本，可利用隐私的组合性质逐个进行计算，最后再求和。

**Moments accountant**

## 5 The algorithmic foundations of differential privacy[5]

### 5.1 Basic Terms

#### 5.1.1 Formalizing differential privacy

**概率单纯形** (Probability Simplex) 离散集合  $B$  上的概率单纯形定义为:

$$\Delta(B) = \left\{ x \in R^{|B|} : x_i \geq 0 \text{ for all } i \text{ and } \sum_{i=1}^{|B|} x_i = 1 \right\} \quad (8)$$

它是一个向量集合, 其元素是一个概率表, 一个  $|B|$  维向量, 此向量每个元素代表一个概率, 所有元素之和为 1。

**随机算法** (Randomized Algorithm) 定义域为  $A$ , 值域为  $B$  的概率算法与一个映射  $M : A \rightarrow \Delta(B)$  相关联, 对算法的输入  $a \in A$ , 其输出为  $b \in B$ ,  $b$  的值是一个随机变量, 满足分布  $M(a)$ 。

**数据库距离** 假设数据库中的记录都来自于集合  $\chi$ ,  $x_i$  的值为自然数, 表示集合  $\chi$  中的第  $i$  个元素出现在数据库  $x$  中的次数。这样就可以用  $x \in \mathbb{N}^{|\chi|}$  来表示数据库。两个数据库之间的距离定义为为其  $l_1$  范数, 即:  $\|x - y\|_1$ 。

**隐私损失** (Privacy loss) 观察  $\xi$  引起的隐私损失定义为:

$$L_{M(x)||X(y)}^{(\xi)} = \ln\left(\frac{Pr[M(x) = \xi]}{Pr[M(y) = \xi]}\right) \quad (9)$$

此值若为正, 则说明一个观察结果更可以由  $x$  引起, 若为负, 则说明结果更可能由  $y$  引起。 $(\epsilon, \delta)$ -差分隐私可以确保, 对于所有的邻接集合  $x, y$ , 隐私损失的绝对值都至少以概率  $1 - \delta$  被限制在  $\epsilon$  以内。

**后处理** (Post-processing) 差分隐私对后处理是免疫的, 也就是说, 如果一个算法保护了隐私, 那对这个算法的结果做任何处理都不会增加隐私损失。

## 5.2 Basic Techniques and Composition Theorems

### 5.2.1 The laplace mechanism

**$l_1$ -敏感度** 函数  $f$  的  $l_1$ -敏感度是：

$$\Delta f = \max_{\substack{x, y \in \mathbb{N}^{|X|} \\ \|x - y\|_1 = 1}} \|f(x) - f(y)\|_1 \quad (10)$$

$l_1$ -敏感度是决定响应查询有多精确的一个重要参数。 $l_1$ -敏感度给出了一个上界：为了保护隐私我们必须在多大程度上干扰输出。为了达到这个目的，我们很自然想到用一个噪声随机变量来实现差分隐私。

**Laplace 分布** 以 0 为中心，尺度为  $b$  的 Laplace 分布的概率密度函数定义为：

$$Lap(x|b) = \frac{1}{2b} \exp\left(-\frac{|x|}{b}\right). \quad (11)$$

**Laplace 机制** 对于函数  $f : \mathbb{N}^{|X|} \rightarrow \mathbb{R}^k$ ， $M_L$  是满足  $\epsilon$ -差分隐私的输出结果， $Y_i$  是独立同分布的随机变量，且  $Y_i \sim Lap(\Delta f / \epsilon)$ ，有：

$$M_L(x, f(\cdot), \epsilon) = f(x) + (Y_1, \dots, Y_k) \quad (12)$$

Laplace 机制捕获  $f$  的输出，并简单地加上尺度为  $\Delta f / \epsilon$  的 Laplace 随机变量进行干扰，使得输出满足  $\epsilon$ -差分隐私。

**示例 3.1 计数查询** 计数查询被描述为：数据集中满足条件  $P$  的元素有多少个。其敏感度为  $\Delta f = 1$ ，因此可以通过添加满足  $Lap(1/\epsilon)$  的噪声使其满足  $\epsilon$ -差分隐私。如果是由任意但固定的  $m$  个计数查询组成的联合查询，可表示为  $f_m : \mathbb{N}^{|X|} \rightarrow \mathbb{R}^m$ ，由于改变一条数据就有可能改变所有的输出，因此其敏感度为  $\Delta f_m = m$ ，因此可以通过添加满足  $Lap(m/\epsilon)$  的噪声使其满足  $\epsilon$ -差分隐私。

**示例 3.2 直方图查询** 直方图查询描述的是：如果将数据集中的所有元素进行不交叉分类，每个类别中各有多少元素。其敏感度为  $\Delta f = 1$ 。

**定理 3.8** 令  $f_m : \mathbb{N}^{|\mathcal{X}|} \rightarrow \mathbb{R}^k$ ,  $y = M_L(x, f(\cdot), \epsilon)$ , 则:

$$Pr[\|f(x) - y\|_\infty \geq \ln(\frac{k}{\delta}) \cdot (\frac{\Delta f}{\epsilon})] \leq \delta \quad (13)$$

$\|f(x) - y\|_\infty$  表示任意大小的数据集, 因加入干扰导致错误的数据条数。此定理可用于计算在满足指定差分隐私的情况下, 在一个置信区间内出现错误的数据条数的上界。

**示例 3.3 姓氏统计** 假设我们要统计一个数据集中所有人的姓氏分别是多少个, 候选姓氏总数一共有 10000 个, 可以表示为:  $f : \mathbb{N}^{|\mathcal{X}|} \rightarrow \mathbb{R}^{10000}$ 。这是一个直方图查询, 因此  $\Delta f = 1$ 。假设我们希望查询满足 (1,0)-差分隐私, 置信度为 95%, 由于  $\ln(10000/0.05) \approx 12$ , 则有:  $Pr[\|f(x) - y\|_\infty \geq 12] \leq 0.05$ , 也就是说, 在满足 (1,0)-差分隐私, 置信度为 95% 的情况下, 数据因干扰出错的数量上界为 12 条。由于定理本身对数据集大小没有规定, 因此如果我们使用的是一个数量为 3 亿的国家人口数据集, 这将是一个非常低的错误率。

**Report Noisy Max** 考虑查找  $m$  元计数查询的最大值的问题:  $m$  元计数查询中每一个计数结果都加上噪声  $Lap(1/\epsilon)$ , 返回其中最大的那个值。此算法满足  $(\epsilon, 0)$ -差分隐私。

### 5.2.2 指数机制

**指数机制** 用于这样的场合: 需要返回一个最大值, 但不能给值直接加上 noise, 因为这会导致返回错误的值 (非最大值加上噪声后可能会变成最大值)。例如, 返回拍卖中的最佳出价, 但要求不泄露最终价格。指数机制基于效用函数  $u : \mathbb{N}^{|\mathcal{X}|} \times R \rightarrow \mathbb{R}$  来定义,  $R$  是离散的输出结果。

### 5.2.3 组合定理

**定理 3.14** 有  $M_1 : \mathbb{N}^{|\mathcal{X}|} \rightarrow R_1$ ,  $M_2 : \mathbb{N}^{|\mathcal{X}|} \rightarrow R_2$  分别满足  $\epsilon_1, \epsilon_2$ -差分隐私, 则他们的组合  $M_{1,2}(x) = (M_1(x), M_2(x))$  满足  $\epsilon_1 + \epsilon_2$ -差分隐私。

**支撑集** 随机变量  $X$  的支撑集表示为  $\text{Supp}(X)$ , 一个概率的支撑集就是所有概率密度非零部分的集合。

**KL-散度** 定义为:

$$D(Y \parallel Z) = \mathbb{E}_{y \sim Y} \left[ \ln \frac{Pr[Y=y]}{Pr[Z=y]} \right] \quad (14)$$

表示两个随机变量之间的“距离”，但是它不是对称的，也不满足三角不等式，当  $\text{Supp}(Z)$  不包含  $\text{Supp}(Y)$  的时候甚至有可能是无穷大。

**Max 散度** 定义为:

$$D_{\infty}(Y \parallel Z) = \max_{S \subseteq \text{Supp}(Y)} \left[ \ln \frac{Pr[Y \in S]}{Pr[Z \in S]} \right] \quad (15)$$

$\delta$ -近似 **Max 散度** 定义为:

$$D_{\infty}^{\delta}(Y \parallel Z) = \max_{S \subseteq \text{Supp}(Y): Pr[Y \in S] \geq \delta} \left[ \ln \frac{Pr[Y \in S] - \delta}{Pr[Z \in S]} \right] \quad (16)$$

**备注 3.1** 机制  $M$  是  $\epsilon$ -差分隐私的，当且仅当，对两个邻接数据集  $x, y$  有:

1.  $D_{\infty}(M(x) \parallel M(y)) \leq \epsilon$  and  $D_{\infty}(M(y) \parallel M(x)) \leq \epsilon$
2.  $D_{\infty}^{\delta}(M(x) \parallel M(y)) \leq \epsilon$  and  $D_{\infty}^{\delta}(M(y) \parallel M(x)) \leq \epsilon$

**统计距离** (statistical distance) 两个随机变量之间的统计距离定义为:

$$\Delta(Y, Z) \stackrel{\text{def}}{=} \max_S (Pr[Y \in S] - Pr[Z \in S]) \quad (17)$$

如果有  $\Delta(Y, Z) \leq \delta$ ，我们称  $Y$  和  $Z$  是  $\delta$ -近邻 (close)。

**引理 3.17**

1.  $D_{\infty}^{\delta}(Y \parallel Z) \leq \epsilon \Leftrightarrow \exists Y' : \Delta(Y, Y') \leq \delta \wedge D_{\infty}(Y' \parallel Z) \leq \epsilon$
2.  $D_{\infty}(Y \parallel Z) \leq \epsilon \wedge D_{\infty}(Z \parallel Y) \leq \epsilon \Leftrightarrow \exists Y', Z' : \Delta(Y, Y') \leq \delta/(e^{\epsilon} + 1) \wedge \Delta(Z, Z') \leq \delta/(e^{\epsilon} + 1) \wedge D_{\infty}(Y' \parallel Z') \leq \epsilon$

**引理 3.18**  $D_{\infty}(Y \parallel Z) \wedge D_{\infty}(Z \parallel Y) \Rightarrow D(Y \parallel Z) \leq \epsilon \cdot (e^{\epsilon} - 1)$

**高级组合的使用场景** 我们的定义要考虑以下两个场景：

- 对同一个数据集重复使用同样的差分隐私算法；
- 针对使用了同一个个体数据的不同数据集，重复利用同样的差分隐私算法。

我们考虑两个实验：

- 两个实验使用同一组预定义的隐私机制  $F$ ；
- 实验由攻击者  $A$  进行，每个实验对数据集进行  $k$  次操作 ( $k$  重)；
- $A$  每次使用机制  $M_i$  和参数  $w_i$  进行操作，针对两个实验输入两个邻接数据集  $x_i^0$  和  $x_i^1$ ，并接收到结果  $y_i \in_R M_i(w_i, x_{i,b})$  (针对实验  $b$ )。
- $V_b$  表示实验  $b$  获得的结果  $(y_1, \dots, y_k)$ ；
- $A$  在实验过程中是有记忆的 (stateful)，他可以根据情况自由选择数据集、机制和参数；
- $A$  总是认为数据集  $x_i^0$  包含 Bob 的信息，而  $x_i^1$  没有；
- 我们要求的隐私保障是：攻击者不能分辨数据集中是否包含 Bob 的信息。

**k 重自由组合下的  $\epsilon$ -差分隐私** 如果一组机制  $F$  有  $D_\infty(V^0 \parallel V^1) \leq \epsilon$ ，我们就称  $F$  满足  $k$  重自由组合下 (under  $k$ -fold adaptive composition) 的  $\epsilon$ -差分隐私。

**定理 3.20 高级组合** 一组  $(\epsilon, \delta)$ -差分机制的  $k$  重自由组合满足  $(\epsilon', k\delta + \delta')$ -差分隐私，其中：  $\epsilon' = \sqrt{2k \ln(1/\delta'\epsilon)} + k\epsilon(e^\epsilon - 1)$ 。

**Laplace vs. Gauss** 我们可以使用 Gauss 噪声代替 Laplace 噪声，此时，我们需要使用  $l_2$  敏感度：

**$l_2$ -敏感度** 函数  $f: \mathbb{N}^{|x|} \rightarrow \mathbb{R}^k$  的  $l_2$ -敏感度定义为：

$$\Delta_2(f) = \max_{\substack{x, y \in \mathbb{N}^{|x|} \\ \|x - y\|_1 = 1}} \|f(x) - f(y)\|_2 \quad (18)$$



**定理 3.22** 对于  $c^2 > 2 \ln(1.25/\delta)$ , 满足参数  $\sigma \geq c\Delta_2(f)/\epsilon$  的 Gauss 机制满足  $(\epsilon, \delta)$ -差分隐私。

## 6 Not Just Privacy: Improving Performance of Private Deep Learning in Mobile Cloud[6]

### 6.1 Introduction

为了实现移动端的深度学习, 一般有两种做法: 1. 减小 DNN 的规模; 2. 将大型 DNN 划分到各个设备和云端进行。前者会导致性能下降和无法控制的能量消耗; 后者通过将 DNN 的前几层设置在移动设备上, 而后几层部署在云端来实现。为了平衡效用和隐私, 通常客户端会把用户数据进行转换再上传到云端, 然而, 即便传输的不是原始数据也会产生隐私泄露。

ARDEN 将网络分割为移动端和云端两个部分, DNN 的浅层被部署在移动端, 用于将原始数据转换成低级特征。

### 6.2 The Proposed Framework

#### 6.2.1 Overview

ARDEN 将 DNN 划分为移动端和云端, 浅层部署在移动端, 它的参数是固定的, 其参数值可以通过迁移学习而来; 云端的参数则通过训练进行调参。

**推理阶段** 敏感数据首先被移动端的浅层进行转换处理, 为保证隐私, 数据还同时被进行 nullification 和随机噪声处理, 之后才被传输到云端进行深层推理。

**训练阶段** 使用与敏感数据相同类型的公开数据进行训练, 同时, 提出了一个噪声训练方法, 将原始数据与生产数据同时用于训练。

### 6.2.2 Differentially private transformation

为了保护隐私,需要在本地将数据进行干扰。直观的方法是在结果中直接加入 Lap 噪声,但由于获取总体敏感度比较困难,因此这会导致加入过的噪声而使效用减低。另外,这种方法也无法提供高度定制化的隐私要求,例如隐去某个高度敏感的字段。因此,本文提出了包括 nullification 和 layer-wise 扰动两种机制。

## 7 Learning privacy preserving encodings through adversarial training<sup>[7]</sup>

### 7.1 Introduction

图像、视频等高纬度数据在分享过程中总是会不经意地泄露一些信息。本文试图找到一种方法,即可以保证信息的有效共享,又能防止隐私泄露。

本文利用对抗训练去学习一种编码,以阻止分类器通过编码过的数据去成功地推导隐私信息。文章演示了在场景识别的应用中,在提升正样本识别率的同时如何降低负样本的识别率。在作为 baseline 的模糊操作都不能阻止场景识别的情况下,本文提出的编码器可以成功地降低负样本场景识别率。

### 7.2 Learning Privcate Encoding Functions

本文框架基于这样的前提: 1. 用于训练编码器的训练集打上了隐私属性值的标签,一旦编码器训练完成,我们将试图对敏感值估计器的训练能力进行限制。2. 当编码器训练完成后,攻击者能够利用编码后的带标签的数据训练估计器,我们需要对这样的估计器的性能进行限制。

#### 7.2.1 Privacy as Adversarial Objective

对整个框架进行形式化定义:

**编码器**  $E: \mathbb{R}^N \rightarrow \mathbb{R}^{N'}$ , 输入样本  $x \in \mathbb{R}^N$ ,  $E(x)_i$  表示  $E(x)$  的第  $i$  个元素。

**隐私属性**  $u(x) \in \mathbb{U}$ 。编码器的目标就是要保护隐私属性的值  $u(x)$  免于纰漏。

**参数化的估计器**  $\hat{u} = \Phi(x'; \theta_u)$

**估计器的损失函数**  $L(\hat{u}, u): \mathbb{U} \times \mathbb{U} \rightarrow \mathbb{R}$ ,  $L$  越小, 则估计出的敏感值越准确。

**估计器的训练目标**  $I(E; u) = \min_{\theta_u} \mathbb{E}_{p(x)} L(\Phi(E(x); \theta_u), u(x))$ . 调整  $\theta_u$ , 使得估计器的平均损失最小。

**编码器的训练目标**  $E = \arg \max_E I(E; u)$ . 使得  $I$  最大的那个  $E$ 。

**避免模型坍塌** 从训练目标可以看出, 这将引发模型坍塌问题, 致使最终  $E(x)$  变为常数。为避免这个问题, 本文对编码器的输出做了一个限制: 要求  $\mathbb{E}E(x)_i = 0, \mathbb{E}E(x)_i^2 = 1$ , 即均值为 0, 方差为 1。

**保护特定的属性** 编码器  $E = \arg \max_E I(E; u) - \alpha I_v(E)$  可以在编码的同时保护特定的属性  $v$ 。即要求  $I(E; u)$  尽可能大的同时,  $I(E; v)$  尽可能小。

## 7.2.2 Architecture and Optimization

## 7.3 Experimental Results

### 基础

**数据集** Places-365。本文从数据集中取两个组, 每个组 10 个类别, 用于构造“平衡”样本数据集。数据集的正样本就是其中一个类别, 而负样本则从同一个组中的其余 9 个类别中随机抽取。

**实验任务** 进行一次二分类：一个图片是否属于某个指定的分类。训练后的模型应当可以抑制某个特定类别（降低特定类别的分类精度），同时也要评估编码器对另外的类别的分类精度的影响。

**编码器训练** 本文训练 4 个编码器，用于抑制 4 种类别的分类，分别是：arch, army base, airport terminal, airfield。另外还训练了第五个编码器，用于抑制“army base”，同时避免对“airport terminal”的识别造成影响，此时取  $\alpha = 1/4$ 。

## 评估

**基线** 对图像进行模糊后得到的分类结果。

分别训练两个编码器：army base 和 airport terminal，同时训练一个抑制 army base 但提升 airport terminal 的编码器。

图 3 的第 1 列展示了图像经过了编码器后，输入分类器得到的训练精度和验证精度，表明编码器可以成功地降低分类精度（进行抑制）；第 2、3 列分别演示了不同编码器对 army base 和 airport terminal 分类器的影响，输入分类器的图像是经过了编码器得到的输出结果。可以看成，当编码器固定后，即便使用经过编码的图像对分类器进行训练，分类器依然不能得到很好的精度。

## 8 Gp-gan: Gender preserving gan for synthesizing faces from landmarks[8]

### 8.1 Introduction

**动机** 面部 landmark 有一系列优点，本文研究怎么将 landmark 用于高级别 (high-level) 的面部分析任务上。目前只需要 68 个面部 landmark 就可以完成性别预测，本文将通过 landmark 生成面部图像，以此来提升性别预测的性能。

**贡献** 1. 在保护性别隐私情况下，第一次使用 landmark 生成面部图像；2. 提出了基于 GAN 的框架；

## 8.2 Proposed Method

**生成器** 使用 U-Net 和 DenseNet

**判别器** 使用 PatchGAN

**目标函数**  $L = L_A + \lambda_P L_P + \lambda_C L_C + \lambda_1 L_1$

**对抗损失** 定义为一组生成图片的熵：

$$L_A = -\frac{1}{N} \sum_{i=1}^N \log(D(\hat{x}_i)) \quad (19)$$

**Perceptual loss** 定义为：  $L_P = \|V(\hat{x}) - V(x)\|_1$

**Gender preserving loss** 描述生成的图像与真实图像之间的性别误差，定义为：

$$L_C = -\frac{1}{N} \sum_i (C(x_i) \log(C(\hat{x}_i)) + (1 - C(x_i)) \log(C(\hat{x}_i))) \quad (20)$$

C 表示预定义好的性别分类网络，本文使用 VGG-16.

**L1 loss** 描述生成的图像与真实图像之间的重建误差，定义为：

$$L_1 = \|G(\hat{x}) - x\|_1 \quad (21)$$

## 8.3 Experiments

实验发现，使用 GPGAN 由 landmark 生成的图像可以很好地保存性别属性，生成图像的性别正确率超过 90%.

## 9 Deepfool: a simple and accurate method to fool deep neural networks[9]

### 9.1 Introduction

Deepfool 的目的是找出使分类器做出错误判断的最小扰动。

### 9.2 DeepFool for binary classifiers

二分类器通过一个超平面将所有样本点分开，为了实现扰动，需要使样本点偏移 to 超平面的另外一边。具体的定义为：

**$\hat{k}$  在点  $x$  处的鲁棒性** 调整向量  $r$ ，使得  $f(x)$  的符号改变，且  $r$  距离原点最近：

$$\Delta(x; \hat{k}) := \min_r \|r\|_2 \text{ subject to } \hat{k}(x+r) \neq \hat{k}(x). \quad (22)$$

**分类器  $\hat{k}$  的鲁棒性** 扰动  $r$  到原点距离与  $x$  到原点的距离之比的均值：

$$\rho_{adv}(\hat{k}) = \mathbb{E}_x \frac{\Delta(x; \hat{k})}{\|x\|_2} \quad (23)$$

**划分超平面** 定义为：  $\mathfrak{F} : x : f(x) = 0$ .

**点到超平面的距离 [10]** 定义为：

$$r = \frac{|w_T x + b|}{\|w\|} \quad (24)$$

## 参考文献

- [1] 熊平, 朱天清, and 王晓峰, “差分隐私保护及其应用”, 计算机学报, vol. 37, no. 1, pp. 101–122, 2014. 1
- [2] 叶青青, 孟小峰, 朱敏杰, 霍峥, et al., “本地化差分隐私研究综述”, 软件学报, , no. 2018 年 07, pp. 1981–2005, 2018. 5

- [3] Reza Shokri and Vitaly Shmatikov, “Privacy-preserving deep learning”, in Proceedings of the 22nd ACM SIGSAC conference on computer and communications security. ACM, 2015, pp. 1310–1321. 7
- [4] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang, “Deep learning with differential privacy”, in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016, pp. 308–318. 10
- [5] Cynthia Dwork, Aaron Roth, et al., “The algorithmic foundations of differential privacy”, Foundations and Trends® in Theoretical Computer Science, vol. 9, no. 3–4, pp. 211–407, 2014. 12
- [6] Ji Wang, Jianguo Zhang, Weidong Bao, Xiaomin Zhu, Bokai Cao, and Philip S Yu, “Not just privacy: Improving performance of private deep learning in mobile cloud”, in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018, pp. 2407–2416. 17
- [7] Francesco Pittaluga, Sanjeev Koppal, and Ayan Chakrabarti, “Learning privacy preserving encodings through adversarial training”, in 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019, pp. 791–799. 18
- [8] Xing Di, Vishwanath A Sindagi, and Vishal M Patel, “Gp-gan: Gender preserving gan for synthesizing faces from landmarks”, in 2018 24th International Conference on Pattern Recognition (ICPR). IEEE, 2018, pp. 1079–1084. 20
- [9] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard, “Deepfool: a simple and accurate method to fool deep neural networks”, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2574–2582. 22
- [10] 周志华, 机器学习, Qing hua da xue chu ban she, 2016. 22