面向深度神经网络训练的数据差分隐私保护随机梯度下降算法

李 英[1] 贺春林[2]

[1](内江职业技术学院信息技术系 四川 内江 641000)

[2](西华师范大学计算机学院 四川 南充 637002)

摘 要 针对传统深度神经网络所采用的随机梯度下降算法忽略了对数据集隐私性保护的缺点,提出一种基于数据差分隐私保护的随机梯度下降算法。引入范数剪切与附加高斯噪声操作,对传统梯度更新策略进行改进。为衡量每次迭代过程中对数据隐私性的破坏,提出隐私损失累积函数在迭代过程中对数据隐私性的侵犯程度进行度量。MNIST 手写数字识别和 CIFAR-10 图像分类实验表明,该算法在保护数据集隐私性的同时,对手写数字以及图像分类的识别准确率分别超过了 90% 和 70%,且相较于传统的随机梯度下降算法,其准确率提升了 5% 以上。该算法在实际工程中能够有效兼顾数据隐私性保护与神经网络辨识准确度。

关键词 深度神经网络 差分隐私 训练集 随机梯度下降 范数剪切 隐私损失累积函数

中图分类号 TP391 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2020.04.041

DATA DIFFERENTIAL PRIVACY PROTECTION STOCHASTIC GRADIENT DESCENT ALGORITHM FOR DEEP NEURAL NETWORK TRAINING

Li Ying[1] He Chunlin[2]

[1](*Department of Information Technology, Neijiang Vocational Technical College, Neijiang 641000, Sichuan, China*)

[2](*School of Computer Science, China West Normal University, Nanchong 637002, Sichuan, China*)

Abstract In allusion to the shortcoming of stochastic gradient descent algorithm for traditional deep neural network, which ignores the privacy protection of datasets, we propose a stochastic gradient descent algorithm based on data differential privacy protection. The traditional gradient update strategy was improved by introducing norm cutting and Gaussian noise operations. Then, in order to measure the destruction of data privacy in each iteration, a privacy loss accumulation function was proposed to measure the degree of data privacy violation in the iteration process. MNIST handwritten digit recognition and the CIFAR-10 image classification experiments show that the recognition accuracy of the algorithm for handwritten digits and image classification exceeds 90% and 70% respectively while protecting the privacy of datasets. Compared with the traditional stochastic gradient descent algorithm, the accuracy can be improved by more than 5%. It also can effectively balance the data privacy protection and neural network identification accuracy in practical engineering.

Keywords Deep neural network Differential privacy Training datasets Stochastic gradient descent Norm cutting Privacy loss accumulation function

0 引 言

深度神经网络算法在诸如图像分类[1]、语言表达[2]和视觉跟踪[3]等工程应用中取得了十分广泛的应用,但其依赖于采用大量训练数据集对神经网络进行训练[4],而在使用这些训练数据集时首先需要保证不侵犯数据的隐私权限[5]。然而,对于深度神经网络

收稿日期:2019-04-02。国家自然科学基金项目(61871330)。**李英**,副教授,主研领域:计算机应用,智能算法。**贺春林**,教授。

而言,受到隐私保护的数据集作为训练集可能对算法最终运行结果的正确性产生影响[6-7]。因此,如何在运用深度神经网络算法的同时有效保护训练样本数据的隐私不受侵犯显得至关重要。

学者们针对神经网络的训练数据隐私问题进行了相关研究。例如,文献[8]针对含有加密后的数据分析计算问题,提出了基于同态加密技术的机器学习算法,在保证不解密的情况下直接对密文进行计算,并与解密后明文计算结果相同。然而,同态加密技术存在运算效率低的缺点。文献[9]提出了基于 Flash 排序算法与 k-匿名保护算法相结合的分类机器学习算法,实现隐私数据保护的同时保持数据集的最优效用,但 k-匿名算法本质上仍存在隐私信息泄露的可能性。

近年来,差分隐私技术被广泛应用于数据私密性保护中,其基本原理为对原始数据通过转换、添加噪声等方法来达到隐私保护的效果,从而确保数据集在执行插入或删除操作时对最终的计算结果不会产生影响。基于此,本文提出一种结合差分隐私的随机梯度下降算法,实现数据隐私保护与神经网络算法的有机结合,主要贡献包括:

(1) 针对传统随机梯度下降算法不考虑对数据隐私性的破坏影响,提出了基于差分隐私保护的随机梯度下降算法,引入附加高斯噪声对数据隐私性进行保护的同时,保证对深度神经网络的训练效果。

(2) 为衡量所提差分隐私随机梯度下降算法对数据隐私的破坏程度,提出隐私损失累积函数的概念对每次迭代过程中的数据隐私破坏程度进行度量。此外,还讨论了算法中关键参数对神经网络训练效果的影响。算例实验表明,所提算法能够有效实现数据隐私保护与算法执行效率间的折中平衡,具有较好的应用前景。

1　相关概念

1.1　基于附加高斯噪声的差分隐私保护机制

差分隐私算法[11]的数学基础为相邻数据集概念,具体的数学定义如下:

定义 1　对于一个随机映射机制 $M:D\to R$,其中 D 为域,而 R 为范围满足(ε,δ)为差分隐私的,若其满足对于任意两个响铃输入 $d,d'\in D$ 和任意输出子集 $S\subseteq R$ 有如下不等式成立:

$$Pr[M(d)\in S]\leqslant e^{\varepsilon}Pr[M(d')\in S]+\delta \qquad (1)$$

式中:ε 为差分隐私预算参数,其表征了隐私保护程度,ε 越小表示隐私保护程度越高;而 δ 则表征了差分隐私被破坏的概率值。由定义 1 可知,差分隐私技术限制了任意对数据集的操作对算法运行结果的影响。具体操作为:令 f 表示某一算法,$f(d)$ 和 $f(d')$ 分别表示两个相邻数据的执行结果。差分隐私即通过调整参数 ε 来保证对数据集中对同一条数据进行删除或添加操作后输出同一结果的概率控制在 e^{ε} 之内,且差分隐私被破坏的概率小于 δ。从上述分析可知,(ε,δ)同样反映了隐私保护的开销程度。此外,文献[12]提出,实现差分隐私的关键在于向数据添加随机噪声,最常见的是添加高斯随机噪声:

$$M(d)\triangleq f(d)+N(0,S_f^2\cdot\sigma^2) \qquad (2)$$

式中:$N(0,S_f^2\cdot\sigma^2)$为高斯分布,其均值为 0,标准差为 $S_f\sigma$,S_f 为敏感度调节算子。

然而,添加噪声的程度与算法对数据的敏感度以及需要控制的隐私保护程度有关。换言之,若加入的噪声程度过大,则算法运行结果的可信度会下降;反之,若加入的噪声程度太小,则无法对数据提供可靠的安全保障。为衡量算法对添加噪声的敏感程度,引入如下定义:

定义 2　对任意算法 $f:D\to R$,算法的全局敏感度 $GS(f)$定义为:

$$GS(f)=\max_{d,d'}\|f(d)-f(d')\|_k \qquad (3)$$

式中:$|d,d'|\leqslant 1$,$\|\cdot\|_k$ 表示 k 范数。

附加高斯噪声的差分隐私保护机制为:若满足 $\delta\geqslant\frac{4}{5}\exp\left(-\frac{(\sigma\varepsilon)^2}{2}\right)$且 $\varepsilon<1$,则算法 f 和敏感度调节算子 S_f 满足差分隐私(ε,δ)。

1.2　深度学习

如图 1 所示,深度神经网络基于模块化思想,通过在多个层次上部署多个神经元并通过逐层训练的手段调整神经元间的连接权值,从而实现原始特征数据进行多次非线性变换,对于任何有限给定输入/输出数据的拟合,最终获取到稳定的特征用于后续的问题分析。

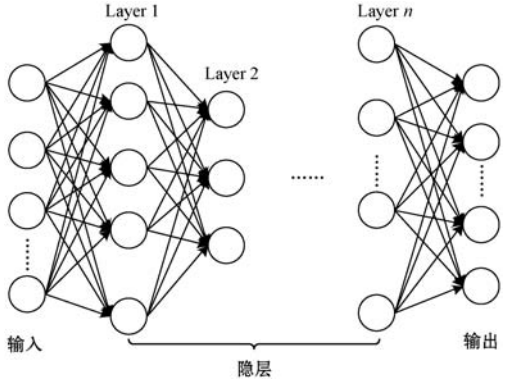


图 1　深度神经网络结构图

深度神经网络算法中,为评估所提神经网络输出预测值与真实值之间的差异程度,用损失函数 L 表示,文中采用均方差损失函数,表示为:

$$L(\theta,x) = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \tag{4}$$

式中:θ 为待训练的神经网络权重系数;x 表示目标值;y 表示预测值输出,下标 i 表示样本标签。深度神经网络算法训练的目的就是使得损失函数 L 最小。而对于复杂的神经网络而言,最小化损失函数 L 通常采用随机梯度下降(stochastic gradient descent,SGD)算法来完成。即每次迭代过程中随机进行批量抽取训练样本(记为 B),并计算损失函数 L 的偏导数 $g_B = \frac{1}{|B|}\sum_{x\in B}\nabla_\theta L(\theta, x)$,然后沿着负梯度方向 $-g_B$ 朝向局部最小值进行更新权重系数 θ。

2 基于差分隐私保护的 SGD 算法

提出一种基于差分隐私保护的随机梯度下降算法。为约束算法迭代过程中对数据隐私性的侵犯,提出隐私损失累积函数的概念对隐私侵犯程度进行度量。

2.1 算法步骤

现有研究中,差分隐私和随机梯度下降算法之间参数的配合选取与交互影响机制尚不明确。例如,在训练数据中加入的噪声过于保守,则在实际算法运行时的准确率将受到影响。因此,通过定义一个数据隐私损失累积函数来量化度量随机梯度下降迭代过程中对数据隐私的侵犯程度。

算法 1 展示了所提差分隐私 SGD 算法的基本步骤,其目标函数通过不断训练和调整权重系数 θ 来最小化损失函数 L。其基本思想为:在每次迭代过程中,首先计算随机生成的批量样本的梯度 $\nabla_\theta L(\theta,x_i)$,并基于计算生成的梯度值的 L_2 范数进行梯度剪切。随后,考虑到样本数据的隐私保护,基于附加高斯噪声方法以梯度与随机噪声之和的均值对剪切后的梯度进行更新,得到下一步迭代的权重系数 θ。最后除最终权重系数之外,还需要输出由于差分隐私保护机制带来的隐私损失。

算法 1 差分隐私 SGD 算法

输入:样本 $\{x_1,x_2,\cdots,x_N\}$ 和损失函数 $L(\theta,x) = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2$

参数:学习率 η_t,噪声尺度 σ,样本规模 $\mathcal{L}$ 与梯度范数边界 C

初始化:随机生成权重系数 θ_0

For $t\in[T]$ **do**

批量样本生成:以概率 $\mathcal{L}/N$ 随机生成样本 $\mathcal{L}_t$

梯度计算:对每一个 $x_i\in\mathcal{L}_t$,计算 $g_t(x_i)\leftarrow\nabla_{\theta_t}L(\theta_t,x_i)$

梯度剪切:$\bar{g}_t(x_i)\leftarrow\frac{g_t(x_i)}{\max\left(1,\frac{\|g_t(x_i)\|_2}{C}\right)}$

增加噪声:$\tilde{g}_t\leftarrow\frac{1}{\mathcal{L}}\left(\sum_i\bar{g}_t(x_i) + \mathcal{N}(0,\sigma^2C^2\boldsymbol{I})\right)$

权重参数梯度下降:$\theta_{t+1}\leftarrow\theta_t - \eta_t\tilde{g}_t$

输出:θ_T 和使用隐私损失累积函数计算得到的隐私开销 (ε,δ)

相关术语解释如下:

梯度裁剪:由于差分隐私保护要求限制每个样本对最终梯度 $\tilde{g}_t$ 的影响。鉴于梯度的取值范围无先验限定,故采用 L_2 范数首先对每个梯度进行裁剪,即梯度向量 g 由 $g/\max(1,\|g\|_2/C)$ 替换,其中 C 为剪切阈值。裁剪后结果为:若 $\|g\|_2\leqslant C$,则保留 g,若 $\|g\|_2 > C$,则将其缩小为常量 C。

神经网络各层参数:神经网络各层参数(即权重系数 θ)都作为损失函数 L 的其中一部分参数。算法 1 同样表明,对于每一层而言均可以对剪切阈值和噪声程度进行单独设置,且可能随着训练迭代步骤 t 的增长而变化。

Lot:和常规 SGD 类似,所提差分隐私 SGD 算法同样计算每次迭代过程中损失函数的梯度均值来估计 L 的梯度。该均值提供了一个无偏估计量,且方差随着样本规模的扩大而迅速减小。为区别于常被称之为批处理的样本,称符合上述条件的样本为一个 Lot。为限制样本数量占用内存的消耗,将批处理的样本规模设置得比 Lot 的规模 $\mathcal{L}$ 小得多,其中 $\mathcal{L}$ 同样为所提差分隐私 SGD 的一个重要参数。随后,将批处理的样本组合成为一个 Lot 进行噪声添加。

2.2 基于隐私损失累积函数的差分隐私侵犯计算

对于所提差分隐私 SGD 算法,除了确保算法运行的准确率以外,另一个重要的问题就是评估算法训练时的数据隐私损失成本。为此,提出隐私损失累积函数的概念来进行每次迭代过程访问训练数据的隐私损失以及随着训练进展时的累积隐私损失。

为不失一般性,令 $\sigma = \frac{\sqrt{2\log(1.25/\delta)}}{\varepsilon}$,文献[14]严格证明,对于抽样概率 $q = \frac{\mathcal{L}}{N}$ 且 $\varepsilon < 1$,则对于完整样本而言,每次迭代过程都是 $(O(q\varepsilon),q\varepsilon)$-差分隐私的。但文献并未对迭代过程以及噪声强度对差分隐私损失的影响展开研究,故无法对噪声强度以及剪切阈值 C

进行有依据的选取。故首先需要研究迭代过程对差分隐私的影响机制。

事实上,若令 $\sigma \geqslant c_2 \frac{q\sqrt{T\log(1/\delta)}}{\varepsilon}$,则同样应用文献[14]方法,可以严格证明算法 1 对于任意的 $\varepsilon < c_1 q^2 T$ 都是$(O(q\varepsilon\sqrt{T}),\delta)$ - 差分隐私的,其中 c_1 和 c_2 为常数。与文献[14]相比,本文算法能够在相同迭代步骤下,大幅度降低 ε 的数值,对数据的隐私性保护更高。

进一步地,对于两个相邻的数据集 $d,d' \in D$ 和映射机制 M,引入一个辅助输入变量 aux 和输出 $o \in R$,定义映射机制 M 在输出 o 处的隐私损失为:

$$c(o;M,aux,d,d') \triangleq \log \frac{\Pr[M(aux,d)=o]}{\Pr[M(aux,d')=o]} \quad (5)$$

对于所提差分隐私 SGD 算法而言,神经网络各层权重系数的参数值与每次迭代过程中的差分隐私机制有着紧密的关联,从而对于给定的映射机制 M,在第 λ 次迭代过程的隐私损失定义为:

$$\alpha_{\mathcal{M}}(\lambda;aux,d,d') \triangleq \log \mathbb{E}_{o\sim M(aux,d)}[\exp(\lambda c(o;M,d,d'))] \quad (6)$$

进一步地,映射机制 M 的损失边界值定义为:

$$\alpha_{\mathcal{M}}(\lambda) \triangleq \max_{aux,d,d'} \alpha_M(\lambda;aux,d,d') \quad (7)$$

其满足如下特性:

1) 组合特性:给定一个机制 M,由一组子机制顺序$\{M_1,M_2,\cdots,M_k\}$组成,并满足 $M_i: \prod_{j=1}^{i-1} R_j \times D \to R_i$,从而总隐私损失边界满足:

$$\alpha_M(\lambda) \leqslant \sum_{i=1}^{k} \alpha_{M_i}(\lambda) \quad (8)$$

2) 差分隐私边界:$\forall \varepsilon > 0$,映射机制 M 是(ε,δ)差分隐私的,当且仅当:

$$\delta = \min_{\lambda} \exp(\alpha_M(\lambda) - \lambda\varepsilon) \quad (9)$$

上述 2 条性质确定了深度神经网络算法每次迭代的隐私损失以及所能够达到侵犯数据隐私容忍度的最大迭代次数。特别地,在附加高斯噪声的情况下,不妨令 μ_0、μ_1 分别为 $N(0,\sigma^2)$ 和 $N(0,\sigma^2)$ 的概率密度函数,而 μ 为两个高斯密度函数的混合概率密度函数,即 $\mu = (1-q)\mu_0 + q\mu_1$。依据式(5) - 式(7)可推导得 $\alpha(\lambda) = \log\max(E_1,E_2)$,其中:

$$E_1 = \mathbb{E}_{z\sim\mu_0}\left[\left(\frac{\mu_0(z)}{\mu(z)}\right)^{\lambda}\right] \quad (10)$$

$$E_2 = \mathbb{E}_{z\sim\mu}\left[\left(\frac{\mu(z)}{\mu_0(z)}\right)^{\lambda}\right] \quad (11)$$

隐私损失边界为:

$$\alpha(\lambda) \leqslant q^2\lambda(\lambda+1)/(1-q)\sigma^2 + O(q^3/\sigma^3) \quad (12)$$

3　实　验

3.1　实验步骤

本文算法采用基于数据流编程(dataflow programming,DP)的 TensorFlow 符号数学系统[15]进行编程。为了保护数据隐私,需在进行梯度下降更新每一层权重系数参数值之前对数据进行清洗。此外,还需根据数据清洗的处理方式计算每次迭代过程中的隐私损失。故算例执行过程中主要包含两大部分:1) 数据清洗,梯度计算前对数据进行清洗以保护隐私;2) 隐私损失累积,在训练过程中计算每次的隐私损失。

算法 2 和算法 3 为基于 TensorFlow 框架下使用 Python 语言编程的所提差分隐私 SGD 算法的核心代码片段。其中:算法 2 为使用所提差分隐私 SGD 算法对损失函数不断优化,命名为 DPSGD_Optimizer;而算法 3 则为隐私损失累积成本函数进行隐私损失迭代计算,命名为 DPtrain。

算法 2　SGD 算法核心代码

```
class DPSGD_Optimizer( ):
def_init_( self,accountant,sanitizer) :
   self_accountant = accoutant
   self_sanitizer = sanitizer
def Minimize( self,loss,params,
        batch_size,noise_options) :
#计算梯度之前的累计隐私损失
priv_accum_op =
      self. _accountant. AccumulatePrivacySpending(
       batch_size,noise_options)
    with tf. control_dependencies( priv_accum_op) :
#计算每个样本的梯度
px_grads = per_example_gradients( loss,params)
#数据清洗
 sanitized_grads = self. _sanitizer. Sanitize(
   px_grads,noise_options)
#执行梯度下降操作
return apply_gradients( params,sanitized_grads)
```

算法 3　隐私损失累积函数代码

```
def DPTrain( loss,params,batch_size,noise_options) :
accountant = PrivacyAccountant( )
sanitizer = Sanitizer( )
dp_opt = DPSGD_Optimizer( accountant,sanitizer)
sgd_op = dp_opt. Minimize(
      loss,params,batch_size,noise_options)
```

```
eps,delta = (0,0)
#在预定义的隐私损失限值内输出训练结果
while within_limit(eps,delta):
sgd_op. run()
        eps,delta = accountant. GetSpentPrivacy()
```

多数情形下,神经网络模型可通过基于主成分分析(principal component analysis,PCA)将输入投影在主方向上或通过卷积层反馈的方式来提高训练效率与训练效果。同样地,算例中也使用差分隐私 + PCA 的方案在公共数据上进行神经网络卷积层的预训练。

3.1.1 数据清洗操作

为实现对样本数据的隐私保护,数据清洗操作需要执行两项操作:1) 通过裁剪每个样本的梯度范数来限制样本中每个数据对最终生成梯度的影响;2) 在更新神经网络各层权重系数参数值之前,将随机噪声添加至批处理的梯度中。

TensorFlow 中,首先对随机抽取的批处理样本计算梯度 $g_B = \frac{1}{|B|}\sum_{x\in B}\nabla_\theta L(\theta,x)$。如第 2 节所述,在进行梯度范数裁剪时,需要访问样本中每个个体的损失函数梯度 $\nabla_\theta L(\theta,x)$。为此,在进行梯度裁剪之前,先借鉴文献[16]中所提 per_example_gradient 运算符进行上述过程,其可以并行批量地计算 $\nabla_\theta L(\theta,x)$,优点在于即便批处理的样本规模很大,训练速度下降的程度也是有限的。随后便可使用 TensorFlow 运算符进行梯度范数裁剪以及随机噪声添加的后续操作。

3.1.2 隐私损失累积函数的主成分分析操作

进行隐私损失累积操作的主要目的在于跟踪计算每次训练迭代过程中的隐私损失成本。如第 2 节所述,可以根据所加噪声的分布参数进而确定每次叠加过程的隐私损失 $\alpha(\lambda)$。

此外,由于主成分分析(PCA)是捕获输入数据主要特征的有效方法。对于用于训练神经网络的随机样本,将其视为向量并进行 L_2 范数归一化处理,形成对称矩阵(记为 $\boldsymbol{A}$),其中每个向量是矩阵 $\boldsymbol{A}$ 中的一行,并基于所提附加高斯噪声的方法添加到协方差矩阵 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$,并计算噪声协方差矩阵的主方向。最终将每个输入的训练样本投影到主方向上作为神经网络最终的输入数据。

3.2 典型数据集验证与结果分析

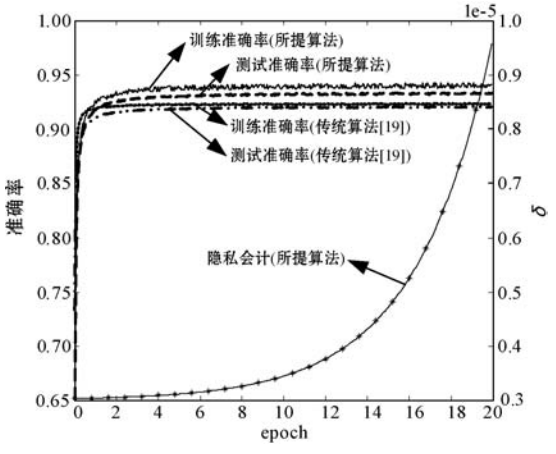
为验证所提基于差分隐私的 SGD 算法的可行性与优异性,采用两个流行的图像数据集 MNIST[17] 和 CIFAR-10 数据集[18] 对算法进行验证。此外,采用文献[19]所提无差分隐私的常规 SGD 算法作为对比算法。

3.2.1 MNIST 手写数字识别

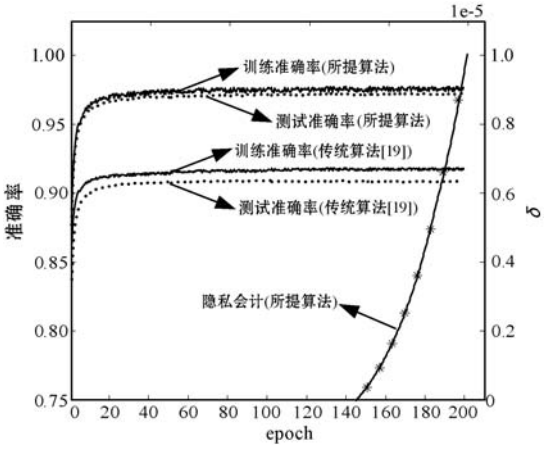
MNIST 数据集为手写数字识别数据集。首先将样本分为包含 60 000 幅图片的训练样本和包含 10 000 幅图片的测试样本。每幅样本均为 28 × 28 的灰度图像。神经网络采用前馈架构并具有 ReLU 激活函数以及 10 类的 Softmax 分类器。

(1) 差分隐私基准实验。选择 PCA 投影层维度为 60 维,包含 1 层具有 1 000 个 ReLU 激活单元的隐含层,并将 Lot 样本规模设置为 600,设梯度剪切阈值为 4。复杂的高斯噪声强度分为三类:小强度噪声($\sigma=2,\sigma_p=4$),中等强度噪声($\sigma=4,\sigma_p=7$)和大强度噪声($\sigma=8,\sigma_p=16$)。其中:σ 为训练神经网络时选择的附加噪声标准差;σ_p 为 PCA 投影时的噪声标准差。初始学习率设置为 0.1,并在 10 个 Epoch 内线性地递减至 0.052 并在今后的训练过程中内保持不变(1 个 Epoch 等于使用训练集内样本全部训练一次)。

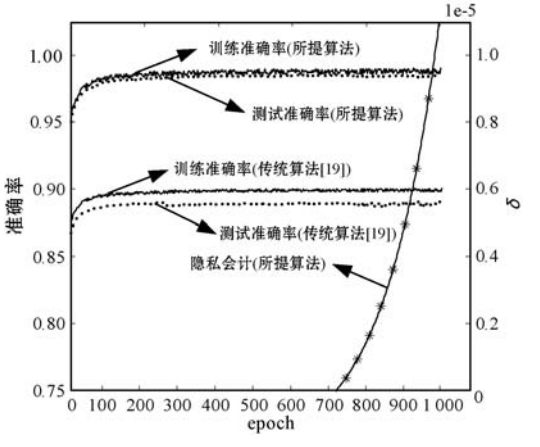
采用固定变量法进行验证。首先令 $\delta=10^{-5}$ 并保持不变,图 2 为不同噪声级别下神经网络算法对手写数字识别准确率随着训练次数以及差分隐私预算参数 ε 的变化情形。可以看出,训练完成后的神经网络在 $(0.5,10^{-5})$ - 差分隐私、$(2,10^{-5})$ - 差分隐私和 $(8,10^{-5})$ - 差分隐私水平下的准确率分别超过了 90%、95% 和 97%,且训练集和测试集的识别准确率结果相差不大。反之,采用非差分隐私的 SGD 算法进行训练时,训练集和测试集的准确率差距则存在过度拟合的现象,即随着 Epoch 数量的增加,二者差距逐渐增大。此外,非差分隐私的 SGD 算法最终的测试准确度较所提算法虽然在大强度噪声情况下较为接近,但在中等强度噪声和低强度噪声时则比本文算法分别低 6.2% 和 9.7%,这进一步说明本文算法具有更好的辨识性能。



(a) 大强度噪声



(b) 中等强度噪声



(c) 低强度噪声

图 2　MNIST 数据集辨识准确度随噪声变化的趋势

(2) 相关参数对算法性能的影响。进一步研究所提差分隐私 SGD 算法中相关参数对算法性能的影响。算法参数包括 PCA 维度数量、隐含层激活单元的数量以及相关训练参数(如 Lot 样本规模、学习速率、梯度范数剪切阈值和噪声强度等)。与差分隐私基准实验类似,同样通过固定变量方法研究上述参数对算法性能的影响,即在其余参数不变的情形下单独研究某一参数对算法性能的动态影响。实验过程的基准参数同样采用 3.1.2 节中所述参数值。

PCA 维度数量:图 3(a)为辨识准确度随着 PCA 维度数量的变化趋势。无 PCA 和随机映射方法下准确率始终保持不变,而所提方法的辨识准确度随着映射维度的变化而发生波动,但总体上的辨识准确率优于前述两种方法。

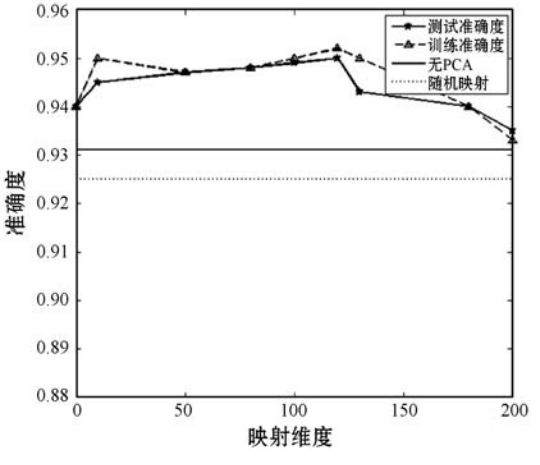
隐含层激活单元的数量:图 3(b)为辨识准确度随着隐含层激活单元数量的变化趋势。可知,对于常规的非差分隐私 SGD 算法而言,只要选择合理的手段来避免过度拟合,则准确率随着激活单元数量的增多而逐渐提升。但对于所提差分隐私 SGD 算法,激活单元数量的增多并不能必然保证准确率的提升,这是因为增多的激活单元增加了对梯度灵敏度的影响,从而导致在梯度更新时需要添加额外的随机噪声。此外还可得到另一结论是,对于所提差分隐私 SGD 算法,并不需要通过一味地使用非常大的神经网络也能得到令人满意的算法性能。

Lot 样本规模:图 3(c)为 Lot 样本大小对识别准确率的影响。由于 Lot 样本规模的选择需要在如下两个互相冲突的目标实现折中:1) 较小规模的 Lot 样本可以运行更多的 Epoch,从而提高训练质量与最终的辨识准确率;2) 较大规模的 Lot 样本,最终结果受附加噪声的影响较小。实验结果表明,Lot 样本规模对最终辨识准确度的影响较大。从多次运行表明,最佳的 Lot 样本规模可选择为$\sqrt{N}$,其中 N 为训练样本的总规模。

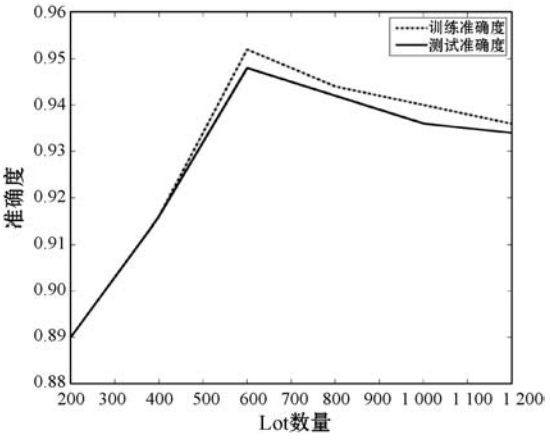
学习率:图 3(d)为学习率对辨识准确度的影响。可见,当学习率维持在[0.01,0.07]且终值为0.05 时,准确率保持稳定。若学习率过大,则准确度会显著下降。

梯度范数剪切阈值:图 3(e)为梯度范式剪切阈值对辨识准确率的影响。当剪切阈值在[2,5]时辨识准确率基本保持稳定,而阈值超过 5 后,辨识准确率出现了明显的下降。这是由于梯度范数剪切阈值的选取需要综合考虑如下两个因素:1) 若阈值取值过小,则最终以平均值代替真实梯度时可能造成误差过大;2) 若阈值取值过大,则由算法 1 可知,将会导致最终根性的梯度中注入过多的噪声。

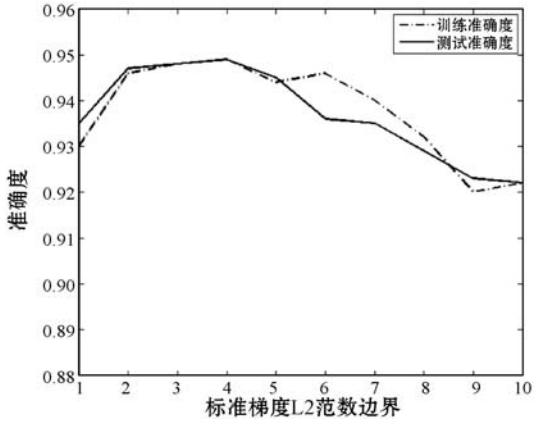
噪声强度:图 3(f)为附加高斯噪声强度对最终辨识准确率的影响。当 σ 取[3,4]时辨识准确率达到最优,而超过这一范围的 σ,其辨识准确率出现了急剧的下降,这表明噪声强度的选取对最终准确率有着至关重要的影响。



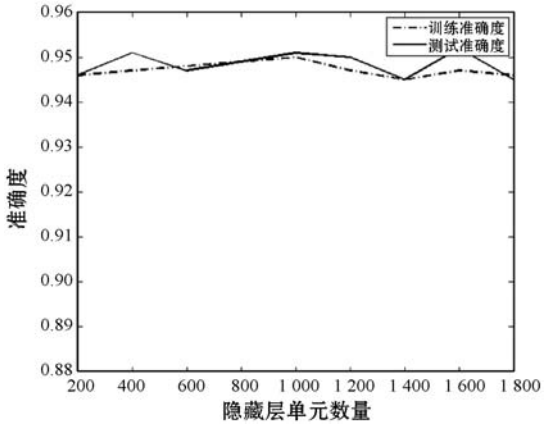
(a) 映射维度变化



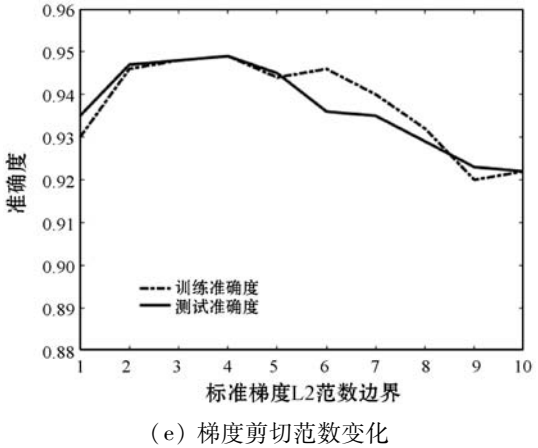
（b）隐藏单元变化



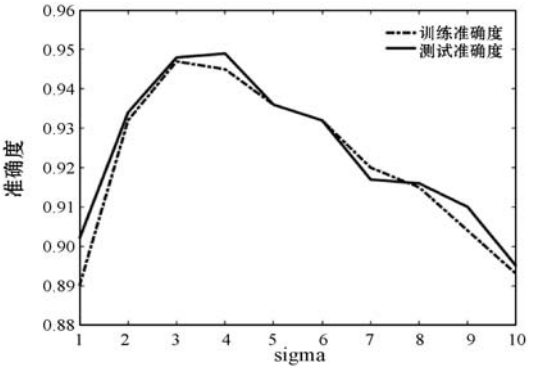
（c）Lot 数量变化



（d）学习率变化



（e）梯度剪切范数变化



（f）噪声强度变化

图 3 MNIST 数据集辨识准确度随参数变化趋势

综上,对于 MNIST 数据集验证实验,可得到如下结论:

1）PCA 提高了模型精度和训练性能。但 PCA 维度的选择对最终的辨识准确度没有明显的影响。

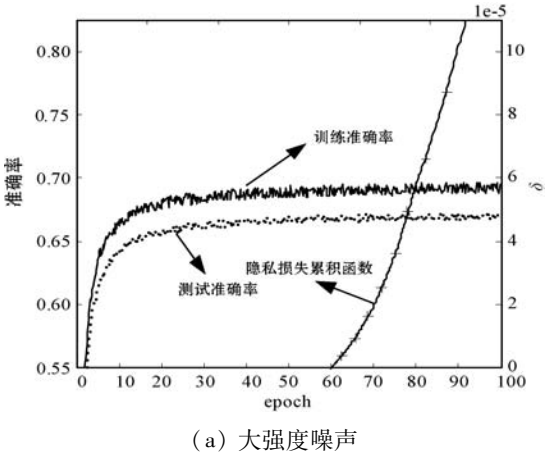
2）隐含层激活单元数量对最终的辨识精度没有明显的影响,对于一个复杂求解问题而言,应用本文方法可以通过运行较小的神经网络来达到令人满意的效果。

3）学习率、Lot 样本规模和噪声强度对深度神经网络求解性能有着很大的影响。本文中仅通过人工经验选取的方法来确定,后续可进一步通过研究自适应参数选择方法来确定这类关键参数。

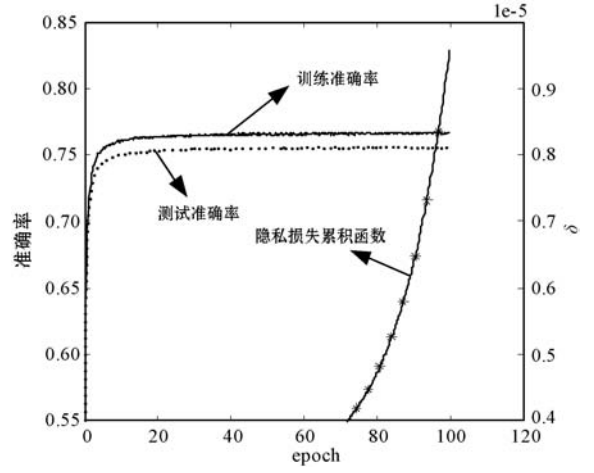
3.2.2 CIFAR-10 图像分类

为进一步说明所提差分隐私 SGD 算法的通用性,使用 CIFAR-10 图像数据集进行验证。其中,数据集由 10 类包含人类、交通工具和动物组成,选择 50 000 个样本用作训练而 10 000 个样本用作测试。

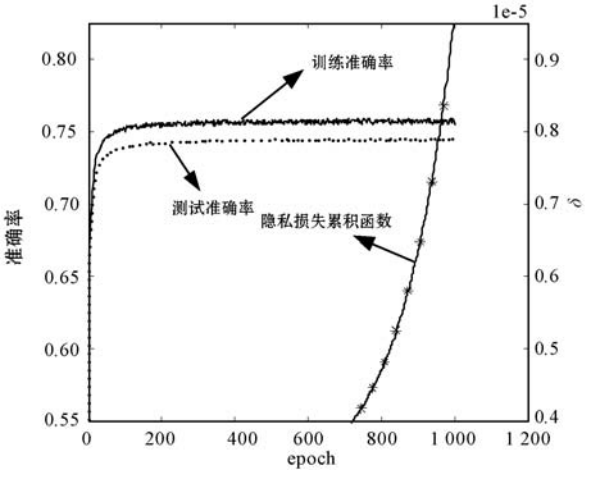
算法采用 TensorFlow 中卷积神经网络的示例网络架构。算例使用两级神经网络,附加高斯噪声参数 $\sigma=6$,梯度范数剪切阈值选择为 3,Lot 样本规模选择为 2 000 和 4 000。图 4 为分类准确度和隐私成本随着训练过程变化的趋势。其中,(a) - (c)对应 $\varepsilon=2,4,6$,而 Lot 样本规模则分别为 2 000、4 000 和 4 000。与 2.2 节类似,训练样本和测试样本的准确率较为接近,分别约为 67%、70% 和 73%。可见,本文方法对图片分类较高的准确率,适用性强。



（a）大强度噪声



(b) 中等强度噪声



(c) 低强度噪声

图 4　CIFAR-10 数据集辨识准确度随噪声变化的影响

4　结　语

为保证深度神经网络训练过程中对训练样本数据的隐私信息的保护,本文提出一种基于差分隐私随机梯度下降算法的深度学习网络算法。MNIST 手写数字识别算例和 CIFAR-10 图像分类实验表明,本文算法的辨识准确度分别达到了 90% 和 70% 的同时,有效保护了数据的隐私性。结果表明,本文算法的适用范围广,并在辨识准确度以及数据隐私性方面取得了较好的折中平衡。未来将进一步研究对差分隐私-SGD 算法参数的自适应选取策略以及辨识准确度,进一步提升策略。

参　考　文　献

[1] 白琮,黄玲,陈佳楠,等. 面向大规模图像分类的深度卷积神经网络优化[J]. 软件学报,2018,29(4):1029 - 1038.

[2] 何炎祥,孙松涛,牛菲菲,等. 用于微博情感分析的一种情感语义增强的深度学习模型[J]. 计算机学报,2017,40(4):773 - 790.

[3] Milovanović M B, Antić D S, Milojković M T, et al. Adaptive PID control based on orthogonal endocrine neural networks [J]. Neural Networks, 2016, 84(1): 80 - 90.

[4] 周飞燕,金林鹏,董军. 卷积神经网络研究综述[J]. 计算机学报,2017,40(6):1229 - 1251.

[5] 刘东江,黎建辉. 基于主动学习的微博数据分类[J]. 计算机应用研究,2018,35(3):803 - 806.

[6] Marshall Z, Brunger F, Welch V, et al. Open availability of patient medical photographs in Google images search results: cross-sectional study of transgender research[J]. Journal of Medical Internet Research, 2018, 20(2): e70.

[7] 柳振东. 卷积神经网络在图像分类中的研究与应用[D]. 天津:中国民航大学,2017.

[8] 尹静,闫河. 训练样本数量选择对图像特征提取的影响分析[J]. 重庆理工大学学报(自然科学),2017,31(10):192 - 197.

[9] 崔建京,龙军,闵尔学,等. 同态加密在加密机器学习中的应用研究综述[J]. 计算机科学,2018,45(4):46 - 52.

[10] Geoffrey Eustace Mtui. 面向最优效用的机器学习隐私模型[D]. 哈尔滨:哈尔滨工业大学,2017.

[11] Ye M, Barg A. Optimal schemes for discrete distribution estimation under locally differential privacy[J]. IEEE Transactions on Information Theory, 2018, 64(8): 5662 - 5676.

[12] 鲜征征,李启良,李改,等. 差分隐私在协同过滤算法中的应用研究[J]. 计算机科学,2017,44(5):81 - 88.

[13] 王豪,徐正全,熊礼治,等. CLM:面向轨迹发布的差分隐私保护方法[J]. 通信学报,2017,38(6):85 - 96.

[14] Li J, Mei X, Prokhorov D. Deep neural network for structural prediction and lane detection in traffic scene[J]. IEEE Transactions on Neural Networks and Learning Systems, 2016, 28(3):690 - 703.

[15] Dwork C, Roth A. The algorithmic foundations of differential privacy[J]. Foundations and Trends in Theoretical Computer Science, 2014, 09(3):211 - 407.

[16] 韩山杰,谈世哲. 基于 TensorFlow 进行股票预测的深度学习模型的设计与实现[J]. 计算机应用与软件,2018,35(6):273 - 277.

[17] Marblestone A H, Greg W, Kording K P. Toward an integration of deep learning and neuroscience[J]. Frontiers in Computational Neuroscience, 2016, 10(5): 1 - 41.

[18] 冯辉,荆晓远,朱小柯. 基于多视图特征投影与合成解析字典学习的图像分类[J]. 计算机应用,2017,37(7):1960 - 1966.

[19] 张占军,彭艳兵,程光. 基于 CIFAR-10 的图像分类模型优化[J]. 计算机应用与软件,2018,35(3):177 - 181.

[20] Yao Z, Saxe A M, Advani M S, et al. Energy-entropy competition and the effectiveness of stochastic gradient descent in machine learning[J]. Molecular Physics, 2018, 116(21): 3214 - 3223.