MapReduce 框架下支持差分隐私保护的随机梯度下降算法

俞艺涵，付钰，吴晓平
（海军工程大学信息安全系，湖北 武汉 430033）

摘 要：针对现有分布式计算环境下随机梯度下降算法存在效率性与私密性矛盾的问题，提出一种 MapReduce 框架下满足差分隐私的随机梯度下降算法。该算法基于 MapReduce 框架，将数据随机分配到各个 Map 节点并启动 Map 分任务独立并行执行随机梯度下降算法；启动 Reduce 分任务合并满足更新要求的分目标更新模型，并加入拉普拉斯随机噪声实现差分隐私保护。根据差分隐私保护原理，证明了算法满足 ε-差分隐私保护要求。实验表明该算法具有明显的效率优势并有较好的数据可用性。

关键词：机器学习；随机梯度下降；MapReduce；差分隐私保护；拉普拉斯机制
中图分类号：TP301
文献标识码：A
doi: 10.11959/j.issn.1000-436x.2018013

Stochastic gradient descent algorithm preserving differential privacy in MapReduce framework

YU Yihan, FU Yu, WU Xiaoping
Department of Information Security, Naval University of Engineering, Wuhan 430033, China

Abstract: Aiming at the contradiction between the efficiency and privacy of stochastic gradient descent algorithm in distributed computing environment, a stochastic gradient descent algorithm preserving differential privacy based on MapReduce was proposed. Based on the computing framework of MapReduce, the data were allocated randomly to each Map node and the Map tasks were started independently to execute the stochastic gradient descent algorithm. The Reduce tasks were appointed to update the model when the sub-target update models were meeting the update requirements, and to add Laplace random noise to achieve differential privacy protection. Based on the combinatorial features of differential privacy, the results of the algorithm is proved to be able to fulfill ε-differentially private. The experimental results show that the algorithm has obvious efficiency advantage and good data availability.

Key words: machine learning, stochastic gradient descent, MapReduce, differential privacy preserving, Laplace mechanism

1 引言

机器学习（ML, machine learning）作为人工智能的核心，可以利用现有数据，通过归纳、综合等方法使计算机实现具备自我学习与自我更新的功能。梯度下降算法是一种典型的求解无约束优化问题的方法，主要思想是朝着负梯度方向寻求目标的最优解。由于该算法具有适用性强、优化效果好等优点，其在机器学习中得到了普遍应用。随机梯度下降（SGD, stochastic gradient descent）算法作为梯度下降算法的一种，由于其在每次迭代过程中不需要遍历所有数据，更适合运用在大数据背景下的机器学习中，但其仍存在以下 2 个方面的问题。1) 随着大数据时代的数据量越来越大，需用分布式计算架构来满足随机梯度下降算法的计算需求。而在分布式计算架构下，随机梯度下降算法在每个计算节

收稿日期：2017-06-19；**修回日期**：2017-12-19
基金项目：国家自然科学基金资助项目（No.61100042）；国家社科基金资助项目（No.15GJ003-201）
Foundation Items: The National Natural Science Foundation of China (No.61100042), The National Social Science Foundation of China (No.15GJ003-201)

点所用样本的不全面性、节点间数据通信频繁造成开销过大等问题，都会导致算法的收敛速度下降[1]。如何在分布式计算框架下进行快速随机梯度下降算法的实现是亟待解决的关键性问题。2) 随机梯度下降算法在帮助人们运用机器学习、数据挖掘等技术不断探索、利用数据中有价值的信息，并以此作为评估、预测和决策等行为依据的同时，也存在着泄露数据集中敏感数据的风险，威胁数据隐私安全[2]。如何在利用大数据的同时，保证大数据中的敏感数据安全是近年来的研究热点。

针对问题 1)，国内外学者做出了许多卓有成效的工作。文献[3]运用抽样概率的思想，使用特殊非均匀采样策略构建 minibatch 来减少随机梯度差异，但其本质需要依赖样本之间的直接关联性；文献[4]通过记录历史梯度，并在当前迭代中使用自适应平均的历史梯度来减少迭代中随机梯度的方差。然而，频繁的记录历史梯度将给存在众多参数的机器学习带来额外的负担。文献[5]提出采用残差最小化框架，修正随机梯度方向，提高随机梯度的稳定性，同时采用半随机梯度思想并提出一种分层半随机梯度下降新方法，来提高收敛速度。由于随机梯度下降算法不可避免地将出现多次更新迭代，这使 MapReduce 等分布式计算架构在处理随机梯度下降算法时，会出现因节点间的反复数据传递而造成的通信开销过大的问题。文献[6]提出在每一个分布式计算节点上完整地执行一遍梯度下降算法，通过平均模型合并得到最终模型。该方法减少了计算过程中的通信开销，但每一个节点的数据存在局限性，没有利用全局数据来提高运算性能。同时，在模型合并时，简单平均合并没有考虑到模型之间存在的差异性，可能会降低算法的收敛速度和最终模型的可用性。文献[7]利用文献[8]中提出的蝴蝶状通信机制，在每一轮迭代中，每个节点将迭代模型仅发送给另一个节点，并接受一个模型对本地模型进行更新。这样可使每一个节点能够充分利用全局数据来提高算法收敛速度与性能。同时，文献[7]还对模型的合并方法进行了优化，将各个更新模型的性能作为模型合并的加权依据，由此提高了算法性能。针对问题 2)，部分学者将差分隐私（DP, differential privacy）保护引入随机梯度下降算法中，以此来应对大数据环境下的隐私泄露问题。文献[9]和文献[10]所提方法为目前较为先进的将差分隐私保护运用到随机梯度下降算法中的方法。文献[9]在随机梯度下降算法的每次迭代中加入扰动噪声，以此达到差分隐私保护的要求；文献[10]通过子集采样的方法来减少每次迭代的噪声量，同时可以保证最佳收敛。但是，以上 2 种方法都存在私密性与效率性以及可用性之间的矛盾，即保证私密性时，算法的性能以及最终模型的可用性将下降；相反，保证效率性与可用性时，扰动噪声的添加可能难以保证差分隐私保护的要求。

基于此，本文提出了一种在分布式计算环境下将差分隐私保护技术应用到随机梯度下降算法中，同时缓解数据私密性与算法效率性矛盾的新算法。该算法通过合理的数据分配方法和模型合并策略来提高随机梯度下降算法的收敛速度与性能，并以策略性的差分隐私保护预算分配进行随机噪声添加，使随机梯度下降算法的输出结果满足差分隐私。

2　差分隐私保护

差分隐私保护是针对具有丰富知识背景的攻击者所提出的一种隐私保护技术，其主要通过对真实数据添加随机扰动，并保证数据在被干扰后仍具有一定的可用性来实现的。其基本原理是，用户通过查询函数 $\boldsymbol{F}$ 对数据集 D 进行查询操作，随机算法 A 通过对查询函数 $\boldsymbol{F}$ 进行扰动，使之满足差分隐私保护的条件[11]。

定理 1　对于 2 个完全相同或至多相差一条记录的数据集 D 和 D'，随机算法 A 的值域为 $R(A)$，事件 X 发生的可能性为 $\Pr[X]$，若对任意 $S,S'\in R(A)$，都满足

$$\Pr[A(D)=S]\leqslant \mathrm{e}^{\varepsilon}\Pr[A(D')=S'] \tag{1}$$

则随机算法 A 提供 ε-差分隐私保护，ε 为差分隐私保护预算。

差分隐私保护通常对数据进行随机噪声添加和随机响应来达到隐私保护目的，主要的实现机制分别为拉普拉斯机制与指数机制。其中，拉普拉斯机制[12]适用于数值型保护，是随机梯度下降算法中最常用的差分隐私保护机制。查询函数的全局敏感度是决定满足差分隐私保护的随机噪声大小的关键因素。全局敏感度定义如下。

定义 1　查询函数 $\boldsymbol{F}$ 的全局敏感度为

$$\Delta F=\max_{D,D'}\|\boldsymbol{F}(D)-\boldsymbol{F}(D')\|_1 \tag{2}$$

其中，D 和 D' 至多只相差一条记录，$\|\boldsymbol{F}(D)-$

$\boldsymbol{F}(D')\|_1$ 表示向量 $\boldsymbol{F}(D)-\boldsymbol{F}(D')$ 各元素绝对值之和。

定理 2 对于数据集 D，查询函数 $\boldsymbol{F}$ 以及其全局敏感度 ΔF，如果随机噪声 Y 服从尺度为 $\frac{\Delta F}{\varepsilon}$ 的拉普拉斯分布，则随机算法 $A(D)=\boldsymbol{F}(D)+Y$ 可以提供 ε-差分隐私保护[12]。

此外，差分隐私保护存在以下 2 个方面的组合性质[13]，是将差分隐私保护运用到反复迭代过程中，证明算法满足差分隐私保护以及合理分配差分隐私预算的基础。

性质 1 若存在 n 个随机算法序列 $A_i(1\leqslant i\leqslant n)$ 提供 ε_i 差分隐私保护，则对于同一数据集 D，$\{A_1,\cdots,A_n\}$ 在 D 上的序列组合算法也提供 ε-差分隐私保护，其中，$\varepsilon=\sum_{i=1}^{n}\varepsilon_i$。

性质 2 若存在随机算法 A 提供 ε-差分隐私保护，数据集 D 可分为不相交的子集 $D_1,\cdots,D_m$，则随机算法 A 在 $\{D_1,\cdots,D_m\}$ 上的组合运算所构成的算法也提供 ε-差分隐私保护。

3 MapReduce 框架下的 DP-SGD 算法

本文所提算法的功能是在 MapReduce 分布式计算框架下，实现对随机梯度下降算法提供有效的差分隐私保护，并保证算法具有较高的效率性。即保证当数据集中改变任何一个记录时，随机梯度下降算法所得到目标模型的变化不会泄露数据集的隐私信息。也就是说，拥有丰富背景知识的攻击者无法通过手头上与目标数据集高度相似（极端情况下只相差一条记录）的数据集，通过目标模型的建立得到数据集中单个数据的隐私信息。

算法的基本思路是将数据集中的数据分配到各个分布式计算节点上，通过 Map 分任务在每个节点上执行随机梯度下降算法，利用 Reduce 分任务进行更新模型合并操作，在更新后的模型中加入适量拉普拉斯噪声，使最终随机梯度下降算法得到的目标模型满足 ε-差分隐私。

现有的差分隐私保护随机梯度下降算法[9,10]存在的最主要问题是算法效率性较低，其主要原因是随机梯度下降算法需要通过反复迭代来使目标模型收敛，而算法的反复迭代将造成在 MapReduce 等分布式计算框架计算过程中，产生大量节点之间的通信开销；而在每轮迭代中，添加随机噪声也不利于目标模型的收敛。因此，本文对 MapReduce 框架下的 DP-SGD 算法进行设计，采取了改进的数据分发与模型合并方案以及随机噪声的添加方法。算法主要符号说明如表 1 所示。

表 1 DP-SGD 算法设计符号说明

类别	变量	说明
数值型参数	N	数据记录总数
	ε	隐私预算量
	K_i、$K_{\max}$、Per、Cou	计数初值、计数最大值、数据利用率、计数阈值
	u	Map 任务中模型更新次数
	U	Reduce 任务中更新次数
	μ	更新次数中间值
	E	阶段性误差标准
	$\Delta error$	误差变化量
	Jud	测试数据判断值
	$Error$	最终模型误差值
	E_{final}	最终标准误差
函数	$L(u)$	第 u 次更新时的更新次数阈值
	$W(u)$	第 u 次更新时分目标模型
	$error(u)$	第 u 次更新时 $W(u)$ 相对误差计算数据集的误差
一般符号	n_i	第 i 条记录
	W_{update}	Reduce 任务更新模型
	W_F	最终模型

3.1 算法设计

设数据集中数据记录总数为 N，第 i 条记录记为 $n_i(1\leqslant i\leqslant N)$。差分隐私保护总预算为 ε。算法步骤如下。

Step1 主任务 Driver 首先将差分隐私保护总预算 ε 平均分成 N 份，并分配给 N 个 Reduce 节点，每个 Reduce 节点初始差分隐私保护预算为 $\frac{\varepsilon}{N}$。将数据记录进行归一化处理，并给每一个 n_i 赋予一个计数初值 K_i，以数据对 $\langle n_i,K_i\rangle$ 的形式存储。

Step2 主任务随机抽取 $num\times M$ 个数据对组成 M 个样本组，每组样本中包含 num 条记录，并指派 M 个分任务执行 Map 操作，N 个分任务执行 Reduce 操作。每个数据对中 K_i 的值等于数据对被抽取的次数。设定在 K_i 超过计数阈值 Cou 后，对应的数据对将不再被抽取。

Step3 Map 分任务接收包含 num 个数据对的样本组，运行 Map 函数。选择 K_i 值小于阈值函数

$L(u)$ 值的数据对中的 n_i 为更新数据集，其他数据作为误差计算数据集。执行梯度下降算法，更新分目标模型 $W(u)$，更新次数 $u+=1$。随后，计算 $W(u)$ 相对误差计算数据集的误差，记为 $error(u)$，当误差计算数据集为空时，默认 $error(u)=0$。当 u 超过更新次数阈值 Max，则丢失该节点 Map 任务。

Step4 Reduce 分任务接收各个 Map 中满足 $error(u)<E$ 且 $\Delta error<0$ 的分目标模型 $W(u)$，运行 Reduce 函数。以各个分目标函数所对应的 $error$ 值的反比为权重合成本次迭代的目标更新模型 W_{update}，并加入随机噪声，更新次数 $U+=1$。

Step5 主任务接收各个 Reduce 节点的输出结果 W_{update} 并进行合并得到 W_F。以数据集中 K_i 值最小的前 Jud 个数据对中的 n_i 为测试数据，计算 W_F 的误差 $Error$，若 $Error<E_{\text{final}}$ 且所有数据对中 $K_i>K_{\max}$ 的比例超过 Per，算法结束，输出结果；否则，重复 Step3~Step5。MapReduce 框架下的 DP-SGD 算法流程如图 1 所示。

3.2 隐私预算及隐私性分析

MapReduce 框架下的 DP-SGD 算法中，差分隐私保护总预算为 ε。在每个 Reduce 节点进行一轮更新的过程中，通过加入随机噪声 $Lap\left(\frac{\Delta F}{\varepsilon'}\right)$ 来实现差分隐私保护，其中，ε' 为每次更新的差分隐私保护预算。当查询函数一定时，随机噪声的大小由 ε' 决定：ε' 越大，即加入的随机噪声越小，数据的可用性越好，而隐私性较弱，为了加入的随机噪声可以满足差分隐私保护的要求，ε' 不能过大，需要有一个上界值[14]；相反，ε' 越小，即加入的随机噪声越大，数据的隐私性越好，即越能达到差分隐私的需求，但数据的可用性将下降，会影响算法的收敛速度以及最终模型的准确性。另一方面，当差分隐私保护总预算量确定时，由于反复迭代的过程中，需要在各个节点的每一次更新中都加入随机噪声，隐私预算有被耗尽的风险。

$$\varepsilon'_U=\begin{cases}\dfrac{\varepsilon}{N}, & U=0\\ \dfrac{\varepsilon}{N2^U}, & 0<U\leqslant\mu\\ \dfrac{\frac{\varepsilon}{N}(1-\frac{1}{2^\mu})}{Max-U}, & \mu<U\leqslant Max\end{cases}\tag{3}$$

在 Reduce 分任务中，各个 Reduce 节点接收不同 Map 发送来的，相互独立地进行计算，生成 W_{update}；通过在 W_{update} 的每一次更新中加入 $Lap\left(\frac{\Delta F}{\varepsilon_U'}\right)$ 使目标更新模型 W_{update} 满足差分隐私保护；每轮迭代更新的结果 W_F，是由主函数通过各个 Reduce 生成的 W_{update} 合并得到，相当于 Reduce 操作的平行叠加[15]；由此，根据性质 1 和性质 2 可知，MapReduce 框架下的 DP-SGD 算法可以满足 ε-差分隐私保护。

3.3 效率性及可靠性分析

本文所提 MapReduce 框架下的 DP-SGD 算法通过关键参数的设置，对隐私预算、计算资源进行合理规划，并优化了数据分配以及模型合并的方法。

1) 计数阈值 Cou

通过随机抽取的方式对数据集进行分组，有利于各个 Map 节点接受的数据具有全局性。而计数阈值 Cou 的设置则是为了防止在对数据集进行随机抽取时，对某一数据出现反复抽取的情况。随机梯度下降算法在迭代的过程中，数据来源将影响其收敛的速度。对于同一个 Map 节点而言，反复使用整个数据集中的部分数据进行更新迭代，将不利于该节点的更新分目标模型 $W(u)$ 相对于全局数据错误率的下降，也就是在更新模型 W_{update} 的合成时，$W(u)$ 的贡献率将因为数据来源的重复而下降，从而导致整个算法性能的下降。

2) 阈值函数 $L(u)$

在 Step2 中设置 Cou 并不能保证同一数据不会被反复运用到同一个分目标模型的更新中，理论上，同一个 Map 节点可能在 u 次更新中至多接收到 u 次相同数据对。为了避免这种在某一 Map 节点的某一数据的反复利用，设置一个与更新次数 u 相关的阈值函数 $L(u)$，来区分本次更新中的更新数据与误差计算数据。这样做一方面使分目标更新模型将接受相对不熟悉的数据进行更新，有利于加快分目标更新模型相对全局数据误差率的下降；另一方面，用分目标函数相对熟悉的数据来检测分目标函数经相对不熟悉数据更新后的误差，更能反映出本次更新后 $error$ 的变化趋势 $\Delta error$，以此作为是否被 Reduce 节点接受的依据之一。

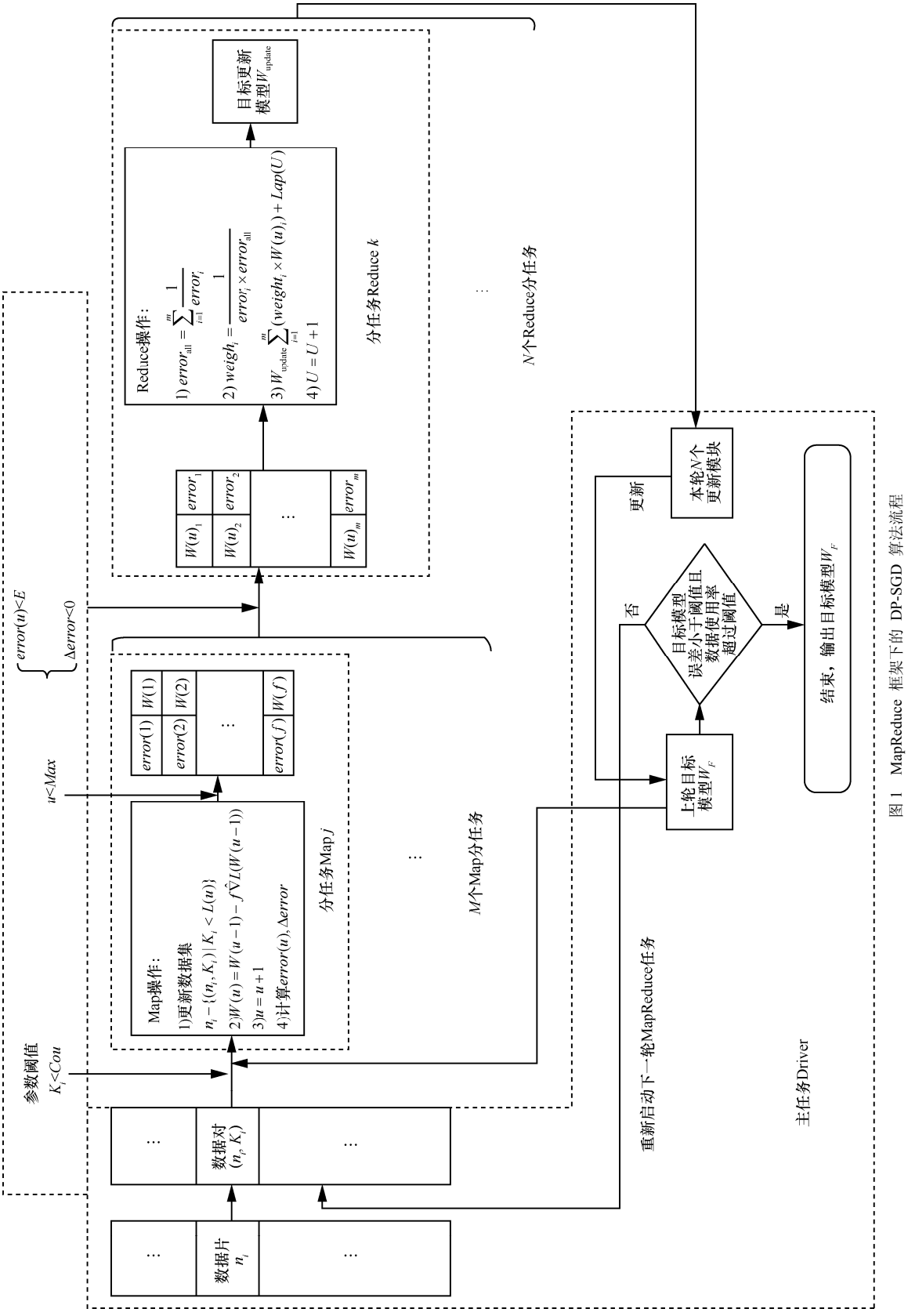


图 1　MapReduce 框架下的 DP-SGD 算法流程

3) 更新次数阈值 Max、更新次数中间值 μ

在目标更新模型 W_{update} 中加入随机噪声，在满足隐私保护要求的同时，不可避免地为 W_{update} 的收敛带来阻碍。为了减小这种阻碍，结合随机梯度下降算法在算法初期误差率下降快、算法后期误差率下降趋于稳定的特点，本文采取以式(3)中 ε_U' 为隐私预算的随机噪声 $Lap\left(\dfrac{\Delta F}{\varepsilon_U'}\right)$ 作为 W_{update} 的随机扰动，旨在保证差分隐私预算不会因为多轮次迭代而消耗完，在算法初期（$0<U\leqslant\mu$），W_{update} 误差率大时，加入较小的随机噪声 $Lap\left(\dfrac{\Delta FN2^U}{\varepsilon}\right)$；在算法后期（$\mu<U\leqslant Max$），$W_{\text{update}}$ 误差率趋于平稳时，加入稳定的随机噪声 $Lap\left(\dfrac{\Delta FN(Max-\mu)}{\varepsilon\left(1-\dfrac{1}{2^{\mu}}\right)}\right)$。当某一 Map 节点的更新次数 u 超过 Max 时，$W(u)$ 若仍没有收敛，则可以认为该节点因为随机噪声的添加或学习步长过大而错过了最佳梯度，该 Map 节点所得的分目标更新模型将不利于整个算法的收敛，则可认为该 Map 节点为坏点，将其丢失；并通过 Max 参数来控制差分隐私预算的分配次数上限，防止因分配次数过多，造成单次隐私预算过小而使随机噪声异常大的情况，进一步保证算法的效率性。

4) 阶段性误差标准 E

在 Reduce 分任务中，Reduce 节点通过阶段性误差标准 E 以及 $\Delta error$ 来判断是否接收各个 Map 节点中的分目标更新模型 $W(u)$，目的是减少因模型传输带来的通信消耗，即将通信资源最大限度地分配给误差率达标的 $W(u)$；并通过各个 $W(u)$误差比的反比进行模型的合并，给性能较好的 $W(u)$更高的权重，更好地体现出数据的全局性，提高算法的性能并保证其具有全局可靠性。

5) 算法终止参数 Jud、E_{final}、$K_{\max}$、Per

将数据集中被抽取次数最少的 Jud 个数据最为最终误差 $Error$ 的计算数据，能够给最终更新模型 W_F 提供相对恶劣的误差计算环境，即理论上使用迭代中使用最少的数据来计算 $Error$ 的最大值。若 $Error<E_{\text{final}}$，且此时数据中各个数据对的 $K_i>K_{\max}$ 的数量占总数据的比重大于 Per，表明当前的 W_F 对于数据集中抽取次数少的数据达到模型更新要求，且 W_F 是在使用具有很强全局性数据更新迭代后所产生的目标模型，此时，可终止算法，输出 W_F。

4　算法效率及可用性实验

本文所提算法主要是对 MapReduce 计算环境下的随机梯度下降算法进行优化，并提供差分隐私保护。算法的隐私性已得到证明，为此，在实验中主要对算法的效率性与可用性进行实验。

实验中的分布式计算平台由 5 台 IBM X 系列机架式服务器组成，每台服务器配置如下：3.30 GHz CPU，2.99 GB 内存，Ubuntu12.04 操作系统，并部署 Hadoop0.20.2。算法由 Java 软件进行开发。

实验选择 MNIST 手写图像数据集作为实验数据集。MNIST 是由 Google 实验室和纽约大学柯朗研究所建立的手写数字数据库，包含 60 000 张训练图像和 10 000 张测试图像。实验分别采用文献[9]中的 SCS13 算法、文献[10]中的 BST14 算法以及本文所提算法建立相同逻辑回归模型，对 MNIST 数据集进行手写数字分类实验。

4.1　算法运行效率实验

为反映本文所提算法在 MapReduce 框架下运行效率因加入随机噪声的影响，本文实验分别在启动一个和 4 个节点的情况下，对本文所提算法添加与不添加随机噪声的情况分别进行计算，并考察不同数据量下的变化规律。首先，在 MNIST 训练集中截取不同数量的记录作为实验数据源，分别上传至 Hadoop 的 HDFS 文件系统中。随后，设置差分隐私预算 $\varepsilon=2$，最终标准误差 $E_{\text{final}}=0.08$，数据利用率 $Per=0.75$，并将算法部署在 MapReduce 中，记录 10 次运行时间的平均值，得到一个和 4 个节点参与运算时本文算法在提供差分隐私保护（分别记为 A-1 算法、A-4 算法）和不提供差分隐私保护（分别记为 B-1 算法、B-4 算法）情况下的运行时间（$t_{\text{A-1}}$、$t_{\text{A-4}}$、$t_{\text{B-1}}$、$t_{\text{B-4}}$）。

算法运行时间如图 2 所示，可以发现 A 算法由于随机噪声的添加，运行时间较 B 算法有明显增加，且随着数据量的增大，运行时间的增量也随之增加。这是由于数据量的增加要求目标模型需要更多的迭代更新次数才能达到算法完成的标准，而每次模型迭代更新时却需要加入阻碍模型收敛的随机噪声引起的。同时，每轮迭代中，都会有一部分

Map 节点与 Reduce 节点之间、Reduce 节点与主节点之间以及子节点与主节点之间的数据传递所造成的额外通信开销，这也导致了算法运行时间的增加。

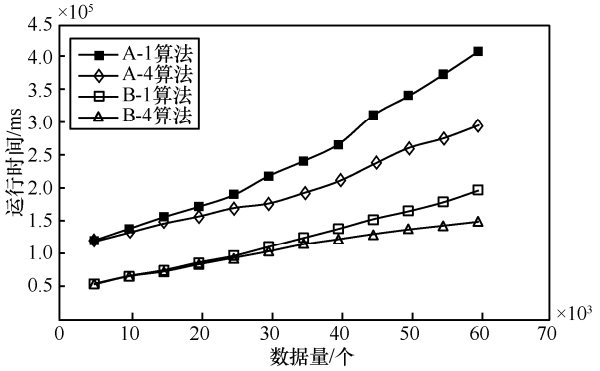
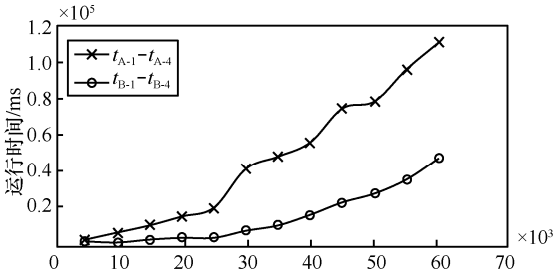
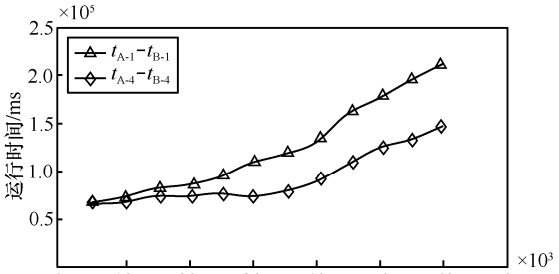


图 2 算法运行时间

运行时间差值随数据变化情况如图 3 所示。由图 3(a)可以看出，当系统启动 4 个子节点时 A 算法和 B 算法的运行时间比启动一个子节点时有显著减少，且随着数据量的增加，运行时间的减少量也在增加；由图 3(b)可以看出，A 算法相对于 B 算法在系统启动 4 个子节点时小于系统仅启动一个子节点时的运行时间增量（由于添加随机噪声造成的增量）。



(a) A、B算法启动不同子节点数运行时间差



(b) A与B算法运行时间差

图 3 运行时间差值随数据量变化情况

由此可以认为，本文所提算法能够使需要反复迭代的随机梯度下降算法在提供差分隐私保护的同时，在 MapReduce 框架下进行高效率的计算，并能够随计算节点的增加而提高算法的运行效率。同时，本文所提算法与 SCS13 算法和 BST14 算法在噪声添加方面采取了不同策略，如图 4 所示。

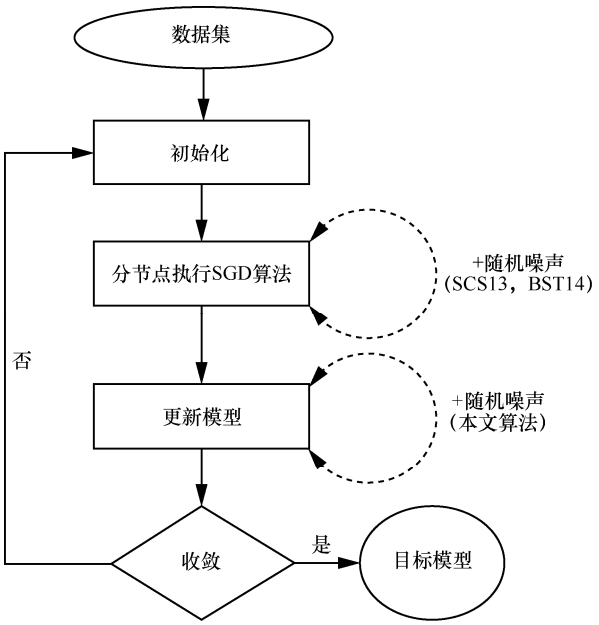


图 4 各算法随机噪声添加位置示意

各算法运行时间对比如图 5 所示。由图 5 可知，本文所提算法在耗时上对比 SCS13 算法和 BST14 算法具有明显优势，且数据量越大优势越明显。

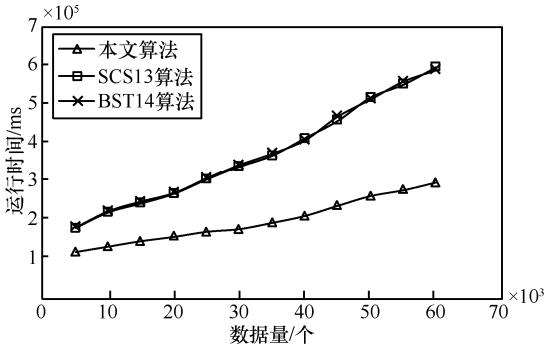


图 5 各算法运行时间对比

4.2 算法可用性实验

为衡量本文所提的 MapReduce 框架下提供差分隐私保护算法的可用性，实验将对 MNIST 数据集进行手写数字分类的准确性作为衡量算法可用性的依据。实验使用 MNIST 数据集中的训练集作为训练模型的数据集，并在测试集中随机抽取 100 张测试图像作为测试数据，对本文所提算法、SCS13 算法以及 BST14 算法的分类准确性进行实验，取 10 次运算

的平均值。当隐私保护总预算 ε 变化时，3 种算法的分类准确性变化如图 6 所示。

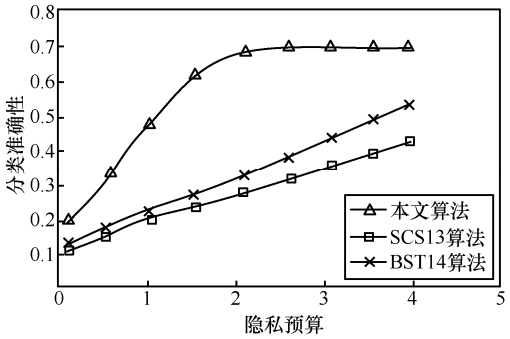


图 6　各算法分类准确性随隐私预算变化情况

由实验结果可知，本文算法的准确性较 SCS13 算法和 BST14 算法有明显优势，当 ε>2 时，本文所提算法对于 MNIST 手写数字分类的准确性达到较高水平。由此可以认为，本文所提算法在保证数据隐私性的同时，保持了数据较高的可用性。

5　结束语

本文在 MapReduce 计算框架下，提出了一种能同时满足效率性与私密性的差分隐私—随机梯度下降新算法 DP-SGD。该算法通过合理的计算资源分配与随机噪声添加策略，在满足差分隐私保护要求的同时，缓解了随机梯度下降算法因反复迭代在分布式计算框架下的通信开销，提高了算法的计算效率并保证了数据的可用性。下一步可进行以下 2 个方面工作：1) 在本文所提算法基础上，对算法中的参数设置方案进一步优化，分析各个参数在对算法效率性影响上的内在关系，进一步提高算法的效率；2) 研究算法中为满足差分隐私保护所需随机噪声量的最小值与数据量、迭代次数和目标模型合并方法之间的关系，减少因随机噪声的添加带来的算法在效率性与可用性上的负面影响。

参考文献：

[1] WU F, LI F G, KUMAR A, et al. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics[C]//The 2017 ACM International Conference on Management of Data. 2017: 1307-1322.

[2] ABADI M, CHU A, GOODFELLOW I, et al. Deep learning with differential privacy[C]//The 2016 ACM SIGSAC Conference on Computer and Communications Security. 2016:308-318.

[3] ZHAO P, ZHANG T. Stochastic optimization with importance sampling[J]. Eprint Arxiv, 2015:1-9.

[4] SCHMIDT M, ROUX N L, BACH F. Erratum to: minimizing finite sums with the stochastic average gradient[J]. Mathematical Programming, 2016, 162(5): 1.

[5] MU Y, LIU W, LIU X, et al. Stochastic gradient made stable: a manifold propagation approach for large-scale optimization[J]. IEEE Transactions on Knowledge & Data Engineering, 2015, 29(2): 458-471.

[6] ZINKEVICH M, WEIMER M, SMOLA A J, et al. Parallelized stochastic gradient descent[C]//The Conference on Neural Information Processing Systems. 2011:2595-2603.

[7] 陈振宏, 兰艳艳, 郭嘉丰,等. 基于差异合并的分布式随机梯度下降算法[J]. 计算机学报, 2015, 38(10):2054-2063.
CHEN Z H,LAN Y Y,GUO J F, et al.Distributed stochastic gradient descent with discriminative aggregating[J].Chinese Journal of Computers, 2015, 38(10):2054-2063.

[8] ZHAO H, CANNY J F. Communication-efficient distributed stochastic gradient descent with butterfly mixing[D]. Berkeley, USA: University of California, 2012.

[9] SONG S, CHAUDHURI K, SARWATE A D. Stochastic gradient descent with differentially private updates[C]//Global conference on Signal and Information Processing (GlobalSIP).2013: 245-248.

[10] BASSILY R, THAKURTA A. Private empirical risk minimization: Efficient algorithms and tight error bounds[C]//2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS).2014: 464-473.

[11] DWORK C, MCSHERRY F, NISSIM K. Calibrating noise to sensitivity in private data analysis[J]. The VLDB Endowment, 2006, 7(8):637-648.

[12] DWORK C, ROTH A. The Algorithmic foundations of differential privacy[M]. Now Publishers Inc, 2014.

[13] CHAUDHURI K, MONTELEONI C, SARWATE A D. Differentially private empirical risk minimization[J]. Journal of Machine Learning Research, 2009, 12(2):1069-1109.

[14] 何贤芒, 王晓阳, 陈华辉,等. 差分隐私保护参数 ε 的选取研究[J]. 通信学报, 2015, 36(12):124-130.
HE X M,WANG X Y, CHEN H H, et al. Study on choosing the parameter ε in differential privacy[J].Journal on Communications, 2015, 36(12):124-130.

[15] MCSHERRY F D. Privacy integrated queries: an extensible platform for privacy-preserving data analysis[J].Communication of the ACM, 2010, 53(9):89-97.

[作者简介]

俞艺涵（1992-），男，浙江金华人，海军工程大学博士生，主要研究方向为信息系统安全、隐私保护等。

付钰（1982-），女，湖北武汉人，博士，海军工程大学副教授、硕士生导师，主要研究方向为信息安全风险评估等。

吴晓平（1961-），男，山西新绛人，博士，海军工程大学教授、博士生导师，主要研究方向为信息安全、密码学等。