# Advances and Open Problems in Federated Learning

Peter Kairouz[7*] H. Brendan McMahan[7*] Brendan Avent[21] Aurélien Bellet[9]
Mehdi Bennis[19] Arjun Nitin Bhagoji[13] Kallista Bonawitz[7] Zachary Charles[7]
Graham Cormode[23] Rachel Cummings[6] Rafael G.L. D'Oliveira[14]
Hubert Eichner[7] Salim El Rouayheb[14] David Evans[22] Josh Gardner[24]
Zachary Garrett[7] Adrià Gascón[7] Badih Ghazi[7] Phillip B. Gibbons[2]
Marco Gruteser[7,14] Zaid Harchaoui[24] Chaoyang He[21] Lie He [4]
Zhouyuan Huo [20] Ben Hutchinson[7] Justin Hsu[25] Martin Jaggi[4] Tara Javidi[17]
Gauri Joshi[2] Mikhail Khodak[2] Jakub Konečný[7] Aleksandra Korolova[21]
Farinaz Koushanfar[17] Sanmi Koyejo[7,18] Tancrède Lepoint[7] Yang Liu[12]
Prateek Mittal[13] Mehryar Mohri[7] Richard Nock[1] Ayfer Özgür[15]
Rasmus Pagh[7,10] Hang Qi[7] Daniel Ramage[7] Ramesh Raskar[11]
Mariana Raykova[7] Dawn Song[16] Weikang Song[7] Sebastian U. Stich[4]
Ziteng Sun[3] Ananda Theertha Suresh[7] Florian Tramèr[15] Praneeth Vepakomma[11]
Jianyu Wang[2] Li Xiong[5] Zheng Xu[7] Qiang Yang[8] Felix X. Yu[7] Han Yu[12]
Sen Zhao[7]

[1]Australian National University, [2]Carnegie Mellon University, [3]Cornell University,
[4]École Polytechnique Fédérale de Lausanne, [5]Emory University, [6]Georgia Institute of Technology,
[7]Google Research, [8]Hong Kong University of Science and Technology, [9]INRIA, [10]IT University of Copenhagen,
[11]Massachusetts Institute of Technology, [12]Nanyang Technological University, [13]Princeton University,
[14]Rutgers University, [15]Stanford University, [16]University of California Berkeley,
[17] University of California San Diego, [18]University of Illinois Urbana-Champaign, [19]University of Oulu,
[20]University of Pittsburgh, [21]University of Southern California, [22]University of Virginia,
[23]University of Warwick, [24]University of Washington, [25]University of Wisconsin–Madison

**Abstract**

Federated learning (FL) is a machine learning setting where many clients (e.g. mobile devices or whole organizations) collaboratively train a model under the orchestration of a central server (e.g. service provider), while keeping the training data decentralized. FL embodies the principles of focused data collection and minimization, and can mitigate many of the systemic privacy risks and costs resulting from traditional, centralized machine learning and data science approaches. Motivated by the explosive growth in FL research, this paper discusses recent advances and presents an extensive collection of open problems and challenges.

[*]Peter Kairouz and H. Brendan McMahan conceived, coordinated, and edited this work. Correspondence to kairouz@google.com and mcmahan@google.com.

# 联邦学习的进展与问题

彼得·凯鲁兹。布伦丹·麦克马汉布伦丹·艾文特奥莉莲·贝莱特·迈赫迪·本尼斯阿尔琼·尼廷·巴戈吉·卡利斯塔·博纳维茨扎卡里·查尔斯·格雷厄姆·科莫德雷切尔·卡明斯拉斐尔·G·L·D'Oliveira Hubert Eichner Salim El RouayhebDavid Evans Josh Gardner Zachary Garrett Adri 'a Gasc ' on Badih Ghazi Phillip B.吉本斯·马可·格鲁特泽·扎伊德·哈沙维·朝阳·何烈和周远霍本·哈钦森·许·马丁·贾吉·塔拉·贾维迪·高里·乔希·米哈伊尔·霍达克·雅各布·科内·科内·科内·科内·科桑桑·科耶乔·李波因特·杨柳·普拉蒂克·米塔尔·梅赫里亚·莫赫里·理查德·诺克·艾弗·奥兹格乌尔·拉斯姆斯·帕格杭·齐·丹尼尔·拉梅什·拉斯卡·马里亚纳·雷科瓦·宋维康·宋·U·塞巴斯蒂安·斯蒂奇孙子腾阿南达泰尔塔苏雷什弗洛里安特拉姆特尔普拉尼韦帕科马王建宇李雄政徐强杨晓.赵玉涵于森

1澳大利亚国立大学、卡内基梅隆大学、康奈尔大学、
4艾德洛桑理工学院，埃默里大学，格鲁吉亚理工学院，
7Google Research 、香港科技大学、INRIA、IT哥本哈根大学、11麻省理工学院、南洋理工大学、普林斯顿大学、
14罗格斯大学、斯坦福大学、加州伯克利大学、
17 加州圣地亚哥大学、伊利诺伊大学香槟分校、欧卢大学、
20匹兹堡大学、南加州大学、弗吉尼亚大学、23沃里克大学、华盛顿大学、威斯康星大学麦迪逊分校

摘要

联合学习（FL）是一种机器学习设置，其中许多客户端（例如，移动的设备或整个组织）在中央服务器（例如，服务提供商）的编排下协作训练模型，同时保持训练数据分散。FL体现了集中数据收集和最小化的原则，可以减轻传统的集中式机器学习和数据科学方法所带来的许多系统性隐私风险和成本。受外语研究爆炸式增长的推动，本文讨论了最新进展，并提出了广泛的开放性问题和挑战。

*Peter Kairouz 和H Brendan McMahan构思、协调和编辑了这部作品。电子邮件：google . com和mcmahan@google . com。

# Contents

# 内容

# 1 Introduction

Federated learning (FL) is a machine learning setting where many clients (e.g. mobile devices or whole organizations) collaboratively train a model under the orchestration of a central server (e.g. service provider), while keeping the training data decentralized. It embodies the principles of focused collection and data minimization, and can mitigate many of the systemic privacy risks and costs resulting from traditional, centralized machine learning. This area has received significant interest recently, both from research and applied perspectives. This paper describes the defining characteristics and challenges of the federated learning setting, highlights important practical constraints and considerations, and then enumerates a range of valuable research directions. The goals of this work are to highlight research problems that are of significant theoretical and practical interest, and to encourage research on problems that could have significant real-world impact.

The term *federated learning* was introduced in 2016 by McMahan et al. [337]: "We term our approach Federated Learning, since the learning task is solved by a loose federation of participating devices (which we refer to as clients) which are coordinated by a central server." An unbalanced and non-IID (identically and independently distributed) data partitioning across a massive number of unreliable devices with limited communication bandwidth was introduced as the defining set of challenges.

Significant related work predates the introduction of the term federated learning. A longstanding goal pursued by many research communities (including cryptography, databases, and machine learning) is to analyze and learn from data distributed among many owners without exposing that data. Cryptographic methods for computing on encrypted data were developed starting in the early 1980s [396, 492], and Agrawal and Srikant [11] and Vaidya et al. [457] are early examples of work that sought to learn from local data using a centralized server while preserving privacy. Conversely, even since the introduction of the term federated learning, we are aware of no single work that directly addresses the full set of FL challenges. Thus, the term federated learning provides a convenient shorthand for a set of characteristics, constraints, and challenges that often co-occur in applied ML problems on decentralized data where privacy is paramount.

This paper originated at the Workshop on Federated Learning and Analytics held June 17–18th, 2019, hosted at Google's Seattle office. During the course of this two-day event, the need for a broad paper surveying the many open challenges in the area of federated learning became clear.[1]

A key property of many of the problems discussed is that they are inherently interdisciplinary — solving them likely requires not just machine learning, but techniques from distributed optimization, cryptography, security, differential privacy, fairness, compressed sensing, systems, information theory, statistics, and more. Many of the hardest problems are at the intersections of these areas, and so we believe collaboration will be essential to ongoing progress. One of the goals of this work is to highlight the ways in which techniques from these fields can potentially be combined, raising both interesting possibilities as well as new challenges.

Since the term federated learning was initially introduced with an emphasis on mobile and edge device applications [337, 334], interest in applying FL to other applications has greatly increased, including some which might involve only a small number of relatively reliable clients, for example multiple organizations collaborating to train a model. We term these two federated learning settings "cross-device" and "cross-silo" respectively. Given these variations, we propose a somewhat broader definition of federated learning:

> ***Federated learning*** *is a machine learning setting where multiple entities (clients) collaborate in solving a machine learning problem, under the coordination of a central server or service provider. Each client's raw data is stored locally and not exchanged or transferred; instead,*

[1] During the preparation of this work, Li et al. [301] independently released an excellent but less comprehensive survey.

# 1引言

联合学习（FL）是一种机器学习设置，其中许多客户端（例如，移动的设备或整个组织）在中央服务器（例如，服务提供商）的编排下协作训练模型，同时保持训练数据分散。它体现了集中收集和数据最小化的原则，可以减轻传统集中式机器学习带来的许多系统性隐私风险和成本。这一领域已收到显着的兴趣最近，无论是从研究和应用的角度。本文描述了联邦学习环境的定义特征和挑战，强调了重要的实际限制和考虑因素，然后列举了一系列有价值的研究方向。这项工作的目标是突出具有重大理论和实践意义的研究问题，并鼓励对可能产生重大现实影响的问题进行研究。

联合学习这个术语是由McMahan 等人在2016 年提出的：“ 我们称我们的方法为联合学习，因为学习任务是通过一个由中央服务器协调的参与设备（我们称之为客户端）的松散联合来解决的。在大量通信带宽有限的不可靠设备上的不平衡和非IID（相同和独立分布）数据分区被引入作为定义挑战集。

重要的相关工作早于联邦学习这个术语的引入。许多研究团体（包括密码学、数据库和机器学习）追求的一个长期目标是分析和学习分布在许多所有者之间的数据，而不暴露这些数据。用于加密数据计算的加密方法始于20 世纪80 年代初[396，492 ]，Agrawal 和Srikant [11 ]和Vaidya 等人[457 ]是试图使用集中式服务器从本地数据中学习同时保护隐私的早期工作的例子。相反，即使自从引入联邦学习这个术语以来，我们也没有意识到有任何单一的工作可以直接解决FL的全部挑战。因此，联邦学习这个术语为一组特征、约束和挑战提供了一个方便的简写，这些特征、约束和挑战经常在分布式数据上的应用ML问题中共同出现，其中隐私是至关重要的。

本文源于2019 年6月17 日至18 日在Google 西雅图办公室举办的联合学习和分析研讨会。在这个为期两天的活动中，很明显需要一份广泛的论文来调查联邦学习领域的许多开放挑战。

所讨论的许多问题的一个关键属性是它们本质上是跨学科的-解决它们可能不仅需要机器学习，还需要分布式优化，密码学，安全性，差分隐私，公平性，压缩感知，系统，信息论，统计学等技术。许多最困难的问题都在这些领域的交叉点上，因此我们相信合作对持续的进展至关重要。这项工作的目标之一是突出这些领域的技术可以结合起来的方式，既提出了有趣的可能性，也提出了新的挑战。

由于联邦学习这个术语最初是在强调移动的和边缘设备应用程序的情况下引入的[337，334 ]，因此将FL应用于其他应用程序的兴趣大大增加，包括一些可能只涉及少量相对可靠的客户端的应用程序，例如多个组织合作训练模型。我们将这两个联邦学习设置分别称为“ 跨设备” 和“ 跨竖井” 。考虑到这些变化，我们提出了一个更广泛的联邦学习定义：

> 联合学习是一种机器学习设置，其中多个实体（客户端）在中央服务器或服务提供商的协调下协作解决机器学习问题。每个客户端的原始数据都存储在本地，不交换或传输;相反，

[1]在这项工作的准备过程中，李等人。[301 ]独立发布了一项出色但不太全面的调查。

*focused updates intended for immediate aggregation are used to achieve the learning objective.*

Focused updates are updates narrowly scoped to contain the minimum information necessary for the specific learning task at hand; aggregation is performed as early as possible in the service of data minimization. We note that this definition distinguishes federated learning from fully decentralized (peer-to-peer) learning techniques as discussed in Section 2.1.

Although privacy-preserving data analysis has been studied for more than 50 years, only in the past decade have solutions been widely deployed at scale (e.g. [177, 154]). Cross-device federated learning and federated data analysis are now being applied in consumer digital products. Google makes extensive use of federated learning in the Gboard mobile keyboard [376, 222, 491, 112, 383], as well as in features on Pixel phones [14] and in Android Messages [439]. While Google has pioneered cross-device FL, interest in this setting is now much broader, for example: Apple is using cross-device FL in iOS 13 [25], for applications like the QuickType keyboard and the vocal classifier for "Hey Siri" [26]; doc.ai is developing cross-device FL solutions for medical research [149], and Snips has explored cross-device FL for hotword detection [298].

Cross-silo applications have also been proposed or described in myriad domains including finance risk prediction for reinsurance [476], pharmaceuticals discovery [179], electronic health records mining [184], medical data segmentation [15, 139], and smart manufacturing [354].

The growing demand for federated learning technology has resulted in a number of tools and frameworks becoming available. These include TensorFlow Federated [38], Federated AI Technology Enabler [33], PySyft [399], Leaf [35], PaddleFL [36] and Clara Training Framework [125]; more details in Appendix A. Commercial data platforms incorporating federated learning are in development from established technology companies as well as smaller start-ups.

Table 1 contrasts both cross-device and cross-silo federated learning with traditional single-datacenter distributed learning across a range of axes. These characteristics establish many of the constraints that practical federated learning systems must typically satisfy, and hence serve to both motivate and inform the open challenges in federated learning. They will be discussed at length in the sections that follow.

These two FL variants are called out as representative and important examples, but different FL settings may have different combinations of these characteristics. For the remainder of this paper, we consider the cross-device FL setting unless otherwise noted, though many of the problems apply to other FL settings as well. Section 2 specifically addresses some of the many other variations and applications.

Next, we consider cross-device federated learning in more detail, focusing on practical aspects common to a typical large-scale deployment of the technology; Bonawitz et al. [81] provides even more detail for a particular production system, including a discussion of specific architectural choices and considerations.

## 1.1 The Cross-Device Federated Learning Setting

This section takes an applied perspective, and unlike the previous section, does not attempt to be definitional. Rather, the goal is to describe some of the practical issues in cross-device FL and how they might fit into a broader machine learning development and deployment ecosystem. The hope is to provide useful context and motivation for the open problems that follow, as well as to aid researchers in estimating how straightforward it would be to deploy a particular new approach in a real-world system. We begin by sketching the lifecycle of a model before considering a FL training process.

旨在立即汇总的重点更新用于实现学习目标。

有重点的更新是范围狭窄的更新，以包含手头的具体学习任务所需的最低限度的信息;尽可能早地进行汇总，以最大限度地减少数据。我们注意到，这个定义将联邦学习与2.1 节中讨论的完全分散的（点对点）学习技术区分开来。

虽然隐私保护数据分析已经研究了50 多年，但只有在过去的十年中，解决方案才被大规模广泛部署（例如[177，154 ]）。跨设备联合学习和联合数据分析现在正在消费数字产品中应用。谷歌在Gboard 移动的键盘[376，222，491，112，383 ]以及Pixel 手机[14 ]和Android Messages [439 ]的功能中广泛使用联邦学习。虽然Google 率先推出了跨设备FL，但现在对这种设置的兴趣要广泛得多，例如：Apple 正在iOS 13 中使用跨设备FL [25 ]，用于QuickType 键盘和“ Hey Siri”的语音分类器等应用程序doc .ai

跨筒仓应用也在无数领域中提出或描述，包括再保险的金融风险预测[476 ]，药物发现[179 ]，电子健康记录挖掘[184 ]，医疗数据分割[15，139 ]和智能制造[354 ]。

对联邦学习技术的需求不断增长，导致了许多工具和框架的出现。其中包括TensorFlow Federated [38 ]，Federated AI Technology Enabler [33 ]，PySyft [399 ]，Leaf [35 ]，PaddleFL [36 ]和Clara Training Framework [125 ];更多细节请参见附录A。结合联邦学习的商业数据平台正在由成熟的技术公司和较小的初创公司开发。

表1 将跨设备和跨竖井的联合学习与传统的单数据中心分布式学习进行了对比。这些特征建立了实际联邦学习系统通常必须满足的许多约束，因此有助于激励和通知联邦学习中的开放挑战。这些问题将在下面几节中详细讨论。

这两种FL变体被称为代表性和重要的示例，但不同的FL设置可能具有这些特征的不同组合。在本文的其余部分，我们考虑跨设备FL设置，除非另有说明，尽管许多问题也适用于其他FL设置。第2节具体讨论了许多其他变化和应用中的一些。

接下来，我们将更详细地考虑跨设备联邦学习，重点关注该技术的典型大规模部署所共有的实际方面; Bonawitz 等人[81 ]为特定的生产系统提供了更多细节，包括对特定架构选择和考虑因素的讨论。

## 1.1 跨设备联合学习设置

本节从应用的角度出发，与前一节不同的是，本节不试图进行定义。相反，我们的目标是描述跨设备FL中的一些实际问题，以及它们如何融入更广泛的机器学习开发和部署生态系统。希望为接下来的开放问题提供有用的背景和动机，并帮助研究人员估计在现实世界的系统中部署特定的新方法有多简单。在考虑FL训练过程之前，我们开始先勾勒模型的生命周期。

| | **Datacenter distributed learning** | **Cross-silo federated learning** | **Cross-device federated learning** |
|---|---|---|---|
| Setting | Training a model on a large but "flat" dataset. Clients are compute nodes in a single cluster or datacenter. | Training a model on siloed data. Clients are different organizations (e.g. medical or financial) or geo-distributed datacenters. | The clients are a very large number of mobile or IoT devices. |
| Data distribution | Data is centrally stored and can be shuffled and balanced across clients. Any client can read any part of the dataset. | **Data is generated locally and remains decentralized.** Each client stores its own data and cannot read the data of other clients. Data is not independently or identically distributed. | |
| Orchestration | Centrally orchestrated. | **A central orchestration server/service organizes the training**, but never sees raw data. | |
| Wide-area communication | None (fully connected clients in one datacenter/cluster). | Typically a hub-and-spoke topology, with the hub representing a coordinating service provider (typically without data) and the spokes connecting to clients. | |
| Data availability | ———— All clients are almost always available. ———— | | Only a fraction of clients are available at any one time, often with diurnal or other variations. |
| Distribution scale | Typically 1 - 1000 clients. | Typically 2 - 100 clients. | Massively parallel, up to $10^{10}$ clients. |
| Primary bottleneck | Computation is more often the bottleneck in the datacenter, where very fast networks can be assumed. | Might be computation or communication. | Communication is often the primary bottleneck, though it depends on the task. Generally, cross-device federated computations use wi-fi or slower connections. |
| Addressability | Each client has an identity or name that allows the system to access it specifically. | | Clients cannot be indexed directly (i.e., no use of client identifiers). |
| Client statefulness | Stateful — each client may participate in each round of the computation, carrying state from round to round. | | Stateless — each client will likely participate only once in a task, so generally a fresh sample of never-before-seen clients in each round of computation is assumed. |
| Client reliability | ———————— Relatively few failures. ———————— | | Highly unreliable — 5% or more of the clients participating in a round of computation are expected to fail or drop out (e.g. because the device becomes ineligible when battery, network, or idleness requirements are violated). |
| Data partition axis | Data can be partitioned / repartitioned arbitrarily across clients. | Partition is fixed. Could be example-partitioned (horizontal) or feature-partitioned (vertical). | Fixed partitioning by example (horizontal). |

Table 1: Typical characteristics of federated learning settings vs. distributed learning in the datacenter (e.g. [150]). Cross-device and cross-silo federated learning are two examples of FL domains, but are not intended to be exhaustive. The primary defining characteristics of FL are highlighted in bold, but the other characteristics are also critical in determining which techniques are applicable.

| | 数据中心<br>分布式学习 | 跨竖井<br>联邦学习 | 跨设备<br>联邦学习 |
|---|---|---|---|
| 设置在大型 | 而是“ 平面” 数据集。客户是一个sin 中的计算节点，集群或数据中心。 | 在孤立数据上训练模型。客户端是不同的组织（例如医疗或金融）或地理分布的中间人。 | 客户端是非常大量的移动的或IoT设备。 |
| Data<br>分布 | 数据集中存储，可以在客户端之间进行洗牌和平衡。任何客户端都可以读取数据集的任何部分。 | 数据是在当地产生的，仍然是分散的。每个客户端存储自己的数据，不能读取其他客户端的数据。数据不是独立或相同分布的。 | |
| 中央协调。中央编排服务器/服务组织训练，但从不看到原始数据。 | | | |
| 广域<br>通信 | 无（完全连接<br>客户在一个客户端-ter / cluster ）。 | 通常是一个中心辐射型拓扑，中心代表一个协调服务提供者（通常没有数据），辐射连接到客户端。 | |
| Data<br>可用性 | 所有客户几乎总是可用的。只有一小部分客户端可以在 | | 任何一个时间，通常有昼夜变化或其他变化。 |
| 分布<br>规模 | 一般有1 - 1000 个客户。一般有2 - 100 个客户。大规模并行，最多10 个客户端。 | | |
| 初级<br>瓶颈 | 计算通常是数据中心的瓶颈，在那里可以假设非常快速的网络。 | 可能是计算或COM -沟通。 | 沟通通常是主要的瓶颈，尽管这取决于任务。通常，跨设备联合计算使用wi - fi或较慢的连接。 |
| 可寻址性每个客户端都有一个身份或名称，允许系统专门访问它。 | | | 客户端不能被直接索引（即，不使用客户端标识符）。 |
| 客户端<br>状态性 | 有状态-每个客户端可以参与计算的每一轮，从一轮到另一轮携带状态。 | | 无状态-每个客户端可能只参与一次任务，因此通常假设在每一轮计算中从未见过的客户端的新样本。 |
| 客户端<br>可靠性 | 失败相对较少。高度不可靠- 5%或更多 | | 参与一轮计算的客户端预期会失败或退出（例如，因为当违反电池、网络或空闲要求时，设备变得不合格）。 |
| 数据分区<br>axis | 数据可以在客户端之间任意分区/重新分区。 | 分区是固定的。可以是示例分区（水平）或特征分区（垂直）。 | 固定分区的例子（水平）。 |

表1： 数据中心中联合学习设置与分布式学习的典型特征（例如[150 ]）。跨设备和跨竖井联邦学习是FL领域的两个示例，但并不打算穷举。FL的主要定义特征以粗体突出显示，但其他特征在确定哪些技术适用时也很关键。

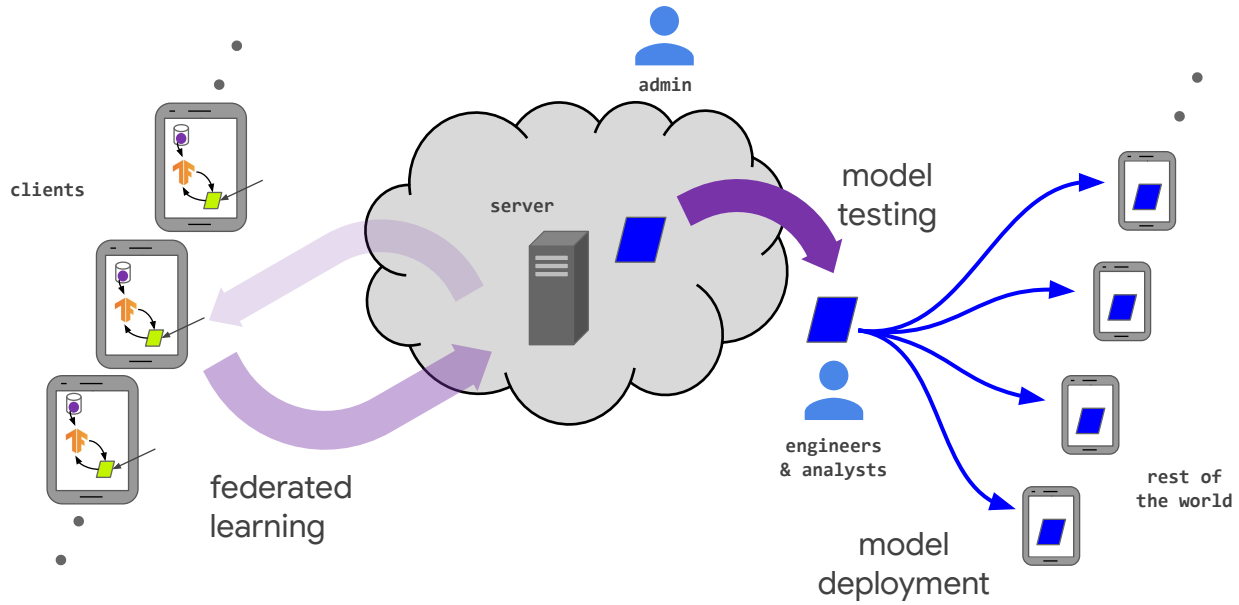



Figure 1: The lifecycle of an FL-trained model and the various actors in a federated learning system. This figure is revisited in Section 4 from a threat models perspective.

### 1.1.1 The Lifecycle of a Model in Federated Learning

The FL process is typically driven by a model engineer developing a model for a particular application. For example, a domain expert in natural language processing may develop a next word prediction model for use in a virtual keyboard. Figure 1 shows the primary components and actors. At a high level, a typical workflow is:

1. **Problem identification:** The model engineer identifies a problem to be solved with FL.
2. **Client instrumentation:** If needed, the clients (e.g. an app running on mobile phones) are instrumented to store locally (with limits on time and quantity) the necessary training data. In many cases, the app already will have stored this data (e.g. a text messaging app must store text messages, a photo management app already stores photos). However, in some cases additional data or metadata might need to be maintained, e.g. user interaction data to provide labels for a supervised learning task.
3. **Simulation prototyping (optional):** The model engineer may prototype model architectures and test learning hyperparameters in an FL simulation using a proxy dataset.
4. **Federated model training:** Multiple federated training tasks are started to train different variations of the model, or use different optimization hyperparameters.
5. **(Federated) model evaluation:** After the tasks have trained sufficiently (typically a few days, see below), the models are analyzed and good candidates selected. Analysis may include metrics computed on standard datasets in the datacenter, or federated evaluation wherein the models are pushed to held-out clients for evaluation on local client data.
6. **Deployment:** Finally, once a good model is selected, it goes through a standard model launch process, including manual quality assurance, live A/B testing (usually by using the new model on some devices and the previous generation model on other devices to compare their in-vivo performance), and a

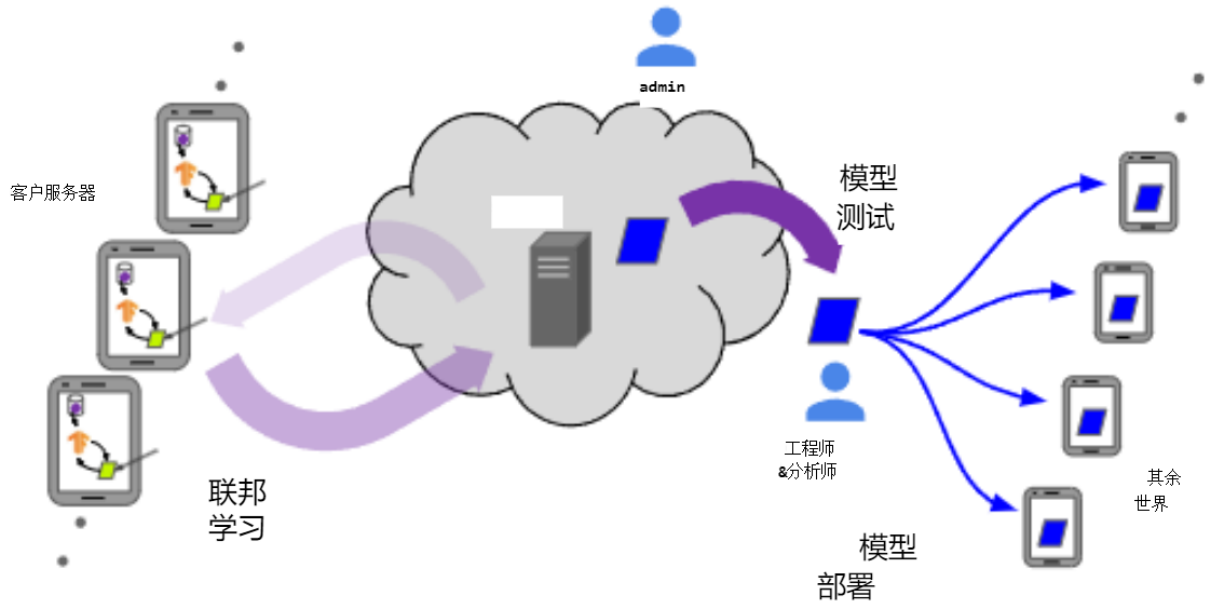



图1：FL训练模型的生命周期和联邦学习系统中的各种参与者。第4节从威胁模型的角度重新讨论了该图。

### 1.1.1 联邦学习中的模型建模

FL过程通常由为特定应用开发模型的模型工程师驱动。例如，自然语言处理领域的专家可以开发用于虚拟键盘的下一单词预测模型。图1显示了主要的组件和参与者。在高级别上，典型的工作流程是：

1.问题识别：模型工程师识别一个需要用FL解决的问题。

2.客户端工具：如果需要，客户端（例如，在移动的电话上运行的应用程序）被仪表化以在本地存储（具有时间和数量上的限制）必要的训练数据。在许多情况下，应用程序已经存储了这些数据（例如，短信应用程序必须存储短信，照片管理应用程序已经存储了照片）。然而，在某些情况下，可能需要维护额外的数据或元数据，例如用户交互数据，以提供监督学习任务的标签。

3. 仿真原型（可选）：模型工程师可以使用代理数据集在FL仿真中对模型架构进行原型设计并测试学习超参数。

4. 联邦模型训练：启动多个联合训练任务来训练模型的不同变体，或使用不同的优化超参数。

5.（联合）模型评估：在任务经过充分训练后（通常需要几天，见下文），分析模型并选择好的候选者。分析可以包括在数据中心中的标准数据集上计算的度量，或者联合评估，其中模型被推送到保持的客户端以用于对本地客户端数据进行评估。

6. 部署：最后，一旦选择了一个好的模型，它将经历一个标准的模型发布过程，包括手动质量保证，现场A/ B测试（通常通过在某些设备上使用新模型和在其他设备上使用上一代模型来比较它们的体内性能），以及一个

| | |
|---|---|
| Total population size | $10^6$–$10^{10}$ devices |
| Devices selected for one round of training | 50 – 5000 |
| Total devices that participate in training one model | $10^5$–$10^7$ |
| Number of rounds for model convergence | 500 – 10000 |
| Wall-clock training time | 1 – 10 days |

Table 2: Order-of-magnitude sizes for typical cross-device federated learning applications.

staged rollout (so that poor behavior can be discovered and rolled back before affecting too many users). The specific launch process for a model is set by the owner of the application and is usually independent of how the model is trained. In other words, this step would apply equally to a model trained with federated learning or with a traditional datacenter approach.

One of the primary practical challenges an FL system faces is making the above workflow as straightforward as possible, ideally approaching the ease-of-use achieved by ML systems for centralized training. While much of this paper concerns federated training specifically, there are many other components including federated analytics tasks like model evaluation and debugging. Improving these is the focus of Section 3.4. For now, we consider in more detail the training of a single FL model (Step 4 above).

### 1.1.2 A Typical Federated Training Process

We now consider a template for FL training that encompasses the Federated Averaging algorithm of McMahan et al. [337] and many others; again, variations are possible, but this gives a common starting point.

A server (service provider) orchestrates the training process, by repeating the following steps until training is stopped (at the discretion of the model engineer who is monitoring the training process):

1. **Client selection:** The server samples from a set of clients meeting eligibility requirements. For example, mobile phones might only check in to the server if they are plugged in, on an unmetered wi-fi connection, and idle, in order to avoid impacting the user of the device.

2. **Broadcast:** The selected clients download the current model weights and a training program (e.g. a TensorFlow graph [2]) from the server.

3. **Client computation:** Each selected device locally computes an update to the model by executing the training program, which might for example run SGD on the local data (as in Federated Averaging).

4. **Aggregation:** The server collects an aggregate of the device updates. For efficiency, stragglers might be dropped at this point once a sufficient number of devices have reported results. This stage is also the integration point for many other techniques which will be discussed later, possibly including: secure aggregation for added privacy, lossy compression of aggregates for communication efficiency, and noise addition and update clipping for differential privacy.

5. **Model update:** The server locally updates the shared model based on the aggregated update computed from the clients that participated in the current round.

Table 2 gives typical order-of-magnitude sizes for the quantities involved in a typical federated learning application on mobile devices.

| | |
|---|---|
| 总人群规模 | 10 - 10 台设备 |
| 选择用于一轮培训的设备 | 50 - 5000 台 |
| 参加培训的设备总数 | 10 - 10 台 |
| 模型收敛轮数 | 500 - 10000 |
| 挂钟培训时间 | 1 - 10 天 |

表2：典型跨设备联合学习应用程序的数量级大小。

分阶段部署（以便在影响太多用户之前发现并回滚不良行为）。模型的特定启动过程由应用程序的所有者设置，通常与模型的训练方式无关。换句话说，这一步同样适用于使用联合学习或传统数据中心方法训练的模型。

FL系统面临的主要实际挑战之一是使上述工作流程尽可能简单，理想情况下接近ML系统在集中培训中实现的易用性。虽然本文的大部分内容都是专门针对联邦训练的，但还有许多其他组件，包括模型评估和调试等联邦分析任务。改善这些是第3.4 节的重点。现在，我们更详细地考虑单个FL模型的训练（上面的步骤4）。

### 1.1.2 典型的联邦训练过程

我们现在考虑FL训练的模板，其中包含McMahan 等人的联合平均算法。[337 ]和许多其他算法;同样，变化是可能的，但这给出了一个共同的起点。

服务器（服务提供商）通过重复以下步骤来编排训练过程，直到训练停止（由监控训练过程的模型工程师决定）：

1.客户端选择：服务器从一组满足资格要求的客户端中进行采样。例如，移动的电话可能仅在插入时、在未计量的wi-fi连接上并且空闲时才签入服务器，以避免影响设备的用户。

2. 广播：选定的客户端从服务器下载当前模型权重和训练程序（例如TensorFlow 图[2]）。

3. 客户端计算：每个选定的设备通过执行训练程序在本地计算对模型的更新，例如可以在本地数据上运行SGD（如联合平均）。

4. 聚合：服务器收集设备更新的聚合。为了提高效率，一旦足够数量的设备报告了结果，就可以在此时丢弃掉队者。该阶段也是稍后将讨论的许多其他技术的集成点，可能包括：用于增加隐私的安全聚合，用于通信效率的聚合的有损压缩，以及用于差分隐私的噪声添加和更新裁剪。

5. 型号更新：服务器基于从参与当前回合的客户端计算的聚合更新来本地更新共享模型。

表2给出了移动的设备上的典型联合学习应用程序中涉及的量的典型数量级大小。

The separation of the client computation, aggregation, and model update phases is not a strict requirement of federated learning, and it indeed excludes certain classes of algorithms, for example asynchronous SGD where each client's update is immediately applied to the model, before any aggregation with updates from other clients. Such asynchronous approaches may simplify some aspects of system design, and also be beneficial from an optimization perspective (though this point can be debated). However, the approach presented above has a substantial advantage in affording a separation of concerns between different lines of research: advances in compression, differential privacy, and secure multi-party computation can be developed for standard primitives like computing sums or means over decentralized updates, and then composed with arbitrary optimization or analytics algorithms, so long as those algorithms are expressed in terms of aggregation primitives.

It is also worth emphasizing that in two respects, the FL training process should not impact the user experience. First, as outlined above, even though model parameters are typically sent to some devices during the broadcast phase of each round of federated training, these models are an ephemeral part of the training process, and not used to make "live" predictions shown to the user. This is crucial, because training ML models is challenging, and a misconfiguration of hyperparameters can produce a model that makes bad predictions. Instead, user-visible use of the model is deferred to a rollout process as detailed above in Step 6 of the model lifecycle. Second, the training itself is intended to be invisible to the user — as described under client selection, training does not slow the device or drain the battery because it only executes when the device is idle and connected to power. However, the limited availability these constraints introduce leads directly to open research challenges which will be discussed subsequently, such as semi-cyclic data availability and the potential for bias in client selection.

## 1.2 Federated Learning Research

The remainder of this paper surveys many open problems that are motivated by the constraints and challenges of real-world federated learning settings, from training models on medical data from a hospital system to training using hundreds of millions of mobile devices. Needless to say, most researchers working on federated learning problems will likely not be deploying production FL systems, nor have access to fleets of millions of real-world devices. This leads to a key distinction between the practical settings that motivate the work and experiments conducted in simulation which provide evidence of the suitability of a given approach to the motivating problem.

This makes FL research somewhat different than other ML fields from an experimental perspective, leading to additional considerations in conducting FL research. In particular, when highlighting open problems, we have attempted, when possible, to also indicate relevant performance metrics which can be measured in simulation, the characteristics of datasets which will make them more representative of real-world performance, etc. The need for simulation also has ramifications for the presentation of FL research. While not intended to be authoritative or absolute, we make the following modest suggestions for presenting FL research that addresses the open problems we describe:

- As shown in Table 1, the FL setting can encompass a wide range of problems. Compared to fields where the setting and goals are well-established, it is important to precisely describe the details of the particular FL setting of interest, particularly when the proposed approach makes assumptions that may not be appropriate in all settings (e.g. stateful clients that participate in all rounds).
- Of course, details of any simulations should be presented in order to make the research reproducible. But it is also important to explain which aspects of the real-world setting the simulation is designed to capture (and which it is not), in order to effectively make the case that success on the simulated

客户端计算、聚合和模型更新阶段的分离并不是联邦学习的严格要求，它确实排除了某些类别的算法，例如异步SGD，其中每个客户端的更新在与其他客户端的更新进行任何聚合之前立即应用于模型。这种异步方法可以简化系统设计的某些方面，并且从优化的角度来看也是有益的（尽管这一点可以争论）。然而，上面提出的方法在提供不同研究路线之间的关注点分离方面具有实质性优势：压缩，差分隐私和安全多方计算的进步可以针对标准原语进行开发，例如在分散更新上计算总和或平均值，然后与任意优化或分析算法组合，只要这些算法以聚合原语表示。

同样值得强调的是，在两个方面，FL培训过程不应该影响用户体验。首先，如上所述，即使模型参数通常在每轮联合训练的广播阶段期间被发送到一些设备，这些模型也是训练过程的短暂部分，并且不用于向用户显示“ 实时” 预测。这一点至关重要，因为训练ML模型是一项挑战，超参数的错误配置可能会产生一个预测不佳的模型。相反，模型的用户可见使用被推迟到模型生命周期的步骤6中详细描述的展示过程。其次，训练本身对用户是不可见的-如客户端选择下所述，训练不会减慢设备或耗尽电池，因为它只在设备空闲并连接电源时执行。 然而，有限的可用性，这些限制直接导致开放的研究挑战，将在随后讨论，如半周期数据的可用性和潜在的偏见，在客户端选择。

## 1.2 联邦学习研究

本文的其余部分调查了许多开放的问题，这些问题是由现实世界的联邦学习设置的限制和挑战所激发的，从医院系统的医疗数据训练模型到使用数亿个移动的设备进行训练。不用说，大多数研究联邦学习问题的研究人员可能不会部署生产FL系统，也无法访问数百万台真实设备。这就导致了激励工作的实际设置和模拟实验之间的关键区别，模拟实验提供了激励问题的给定方法的适用性的证据。

从实验的角度来看，这使得FL研究与其他ML领域有所不同，导致在进行FL研究时需要额外考虑。特别是，当突出开放的问题，我们已经尝试，在可能的情况下，也表明相关的性能指标，可以在模拟中测量，数据集的特点，这将使他们更代表现实世界的性能，等模拟的需要也有分歧的介绍FL研究。虽然不打算成为权威或绝对的，我们提出以下适度的建议，以介绍外语研究，解决我们所描述的开放性问题：

·如表1所示，FL设置可以涵盖广泛的问题。与设置和目标已明确的领域相比，准确描述特定FL设置的细节非常重要，特别是当所提出的方法做出可能不适用于所有设置的假设时（例如，参与所有回合的有状态客户端）。

·当然，任何模拟的细节都应该提供，以使研究具有可重复性。但同样重要的是，要解释模拟旨在捕捉现实世界设置的哪些方面（以及哪些方面不是），以便有效地证明模拟的成功

problem implies useful progress on the real-world objective. We hope that the guidance in this paper will help with this.

- Privacy and communication efficiency are always first-order concerns in FL, even if the experiments are simulations running on a single machine using public data. More so than with other types of ML, for any proposed approach it is important to be unambiguous about *where computation happens* as well as *what is communicated*.

Software libraries for federated learning simulation as well as standard datasets can help ease the challenges of conducting effective FL research; Appendix A summarizes some of the currently available options. Developing standard evaluation metrics and establishing standard benchmark datasets for different federated learning settings (cross-device and cross-silo) remain highly important directions for ongoing work.

### 1.3 Organization

Section 2 builds on the ideas in Table 1, exploring other FL settings and problems beyond the original focus on cross-device settings. Section 3 then turns to core questions around improving the efficiency and effectiveness of federated learning. Section 4 undertakes a careful consideration of threat models and considers a range of technologies toward the goal of achieving rigorous privacy protections. As with all machine learning systems, in federated learning applications there may be incentives to manipulate the models being trained, and failures of various kinds are inevitable; these challenges are discussed in Section 5. Finally, we address the important challenges of providing fair and unbiased models in Section 6.

问题意味着在现实世界的目标上取得了有益的进展。我们希望本文件中的指导将有助于这一点。

隐私和通信效率始终是FL的首要关注点，即使实验是使用公共数据在单台机器上运行的模拟。与其他类型的ML相比，对于任何提出的方法，重要的是要明确计算发生的位置以及传达的内容。

用于联邦学习模拟的软件库以及标准数据集可以帮助缓解chalin ，以便有效地证明成功进行有效的FL研究的模拟过程;附录A总结了目前可用的一些选项。为不同的联合学习设置（跨设备和跨筒仓）开发标准评估指标和建立标准基准数据集仍然是当前工作的重要方向。

### 1.3 组织

第2节以表1中的观点为基础，探讨了除跨设备设置之外的其他FL设置和问题。第3节然后转向围绕提高联邦学习的效率和有效性的核心问题。第4节仔细考虑了威胁模型，并考虑了一系列旨在实现严格隐私保护的技术。与所有机器学习系统一样，在联邦学习应用程序中，可能会有动机操纵正在训练的模型，各种失败是不可避免的;这些挑战将在第5节中讨论。

最后，我们在第6节中讨论了提供公平和无偏见模型的重要挑战。

# 2 Relaxing the Core FL Assumptions: Applications to Emerging Settings and Scenarios

In this section, we will discuss areas of research related to the topics discussed in the previous section. Even though not being the main focus of the remainder of the paper, progress in these areas could motivate design of the next generation of production systems.

## 2.1 Fully Decentralized / Peer-to-Peer Distributed Learning

In federated learning, a central server orchestrates the training process and receives the contributions of all clients. The server is thus a central player which also potentially represents a single point of failure. While large companies or organizations can play this role in some application scenarios, a reliable and powerful central server may not always be available or desirable in more collaborative learning scenarios [459]. Furthermore, the server may even become a bottleneck when the number of clients is very large, as demonstrated by Lian et al. [305] (though this can be mitigated by careful system design, e.g. [81]).

The key idea of fully decentralized learning is to replace communication with the server by peer-to-peer communication between individual clients. The communication topology is represented as a connected graph in which nodes are the clients and an edge indicates a communication channel between two clients. The network graph is typically chosen to be sparse with small maximum degree so that each node only needs to send/receive messages to/from a small number of peers; this is in contrast to the star graph of the server-client architecture. In fully decentralized algorithms, a round corresponds to each client performing a local update and exchanging information with their neighbors in the graph[2]. In the context of machine learning, the local update is typically a local (stochastic) gradient step and the communication consists in averaging one's local model parameters with the neighbors. Note that there is no longer a global state of the model as in standard federated learning, but the process can be designed such that all local models converge to the desired global solution, i.e., the individual models gradually reach consensus. While multi-agent optimization has a long history in the control community, fully decentralized variants of SGD and other optimization algorithms have recently been considered in machine learning both for improved scalability in datacenters [29] as well as for decentralized networks of devices [127, 459, 443, 59, 278, 291, 173]. They consider undirected network graphs, although the case of directed networks (encoding unidirectional channels which may arise in real-world scenarios such as social networks or data markets) has also been studied in [29, 226].

It is worth noting that even in the decentralized setting outlined above, a central authority may still be in charge of setting up the learning task. Consider for instance the following questions: Who decides what is the model to be trained in the decentralized setting? What algorithm to use? What hyperparameters? Who is responsible for debugging when something does not work as expected? A certain degree of trust of the participating clients in a central authority would still be needed to answer these questions. Alternatively, the decisions could be taken by the client who proposes the learning task, or collaboratively through a consensus scheme (see Section 2.1.2).

Table 3 provides a comparison between federated and peer-to-peer learning. While the architectural assumptions of decentralized learning are distinct from those of federated learning, it can often be applied to similar problem domains, many of the same challenges arise, and there is significant overlap in the research communities. Thus, we consider decentralized learning in this paper as well; in this section challenges

[2]Note, however, that the notion of a round does not need to even make sense in this setting. See for instance the discussion on clock models in [85].

# 第2章放松核心FL假设：新兴环境和情景的应用

在本节中，我们将讨论与前一节讨论的主题相关的研究领域。即使不是本文其余部分的主要重点，这些领域的进展可能会激励下一代生产系统的设计。

## 2.1 完全去中心化/对等分布式学习

在联合学习中，中央服务器协调训练过程并接收所有客户端的贡献。因此，服务器是一个中心角色，也可能代表单点故障。虽然大公司或组织可以在某些应用场景中扮演这个角色，但在更多的协作学习场景中，可靠和强大的中央服务器可能并不总是可用或可取的[459 ]。此外，当客户端数量非常大时，服务器甚至可能成为瓶颈，如Lian等人所证明的那样。[305 ]（尽管这可以通过仔细的系统设计来缓解，例如[81 ]）。

完全去中心化学习的关键思想是用个人客户端之间的对等通信取代与服务器的通信。通信拓扑被表示为连接图，其中节点是客户端，并且边指示两个客户端之间的通信信道。网络图通常被选择为具有小的最大度的稀疏的，使得每个节点仅需要向/从少量对等体发送/接收消息;这与服务器-客户端架构的星星图形成对比。在完全分散的算法中，一轮对应于每个客户端执行本地更新并与图中的邻居交换信息。在机器学习的上下文中，局部更新通常是局部（随机）梯度步骤，并且通信包括将一个人的局部模型参数与邻居进行平均。 注意，不再像标准联邦学习中那样存在模型的全局状态，但是可以设计该过程，使得所有局部模型收敛到期望的全局解，即，各个模式逐渐达成共识。虽然多智能体优化在控制界有着悠久的历史，但SGD和其他优化算法的完全分散变体最近在机器学习中被考虑用于提高机器学习中心的可扩展性[29 ]以及设备的分散网络[127，459，443，59，278，291，173 ]。他们考虑了无向网络图，尽管有向网络的情况（编码可能出现在现实世界场景中的单向通道，如社交网络或数据市场）也在[29，226 ]中进行了研究。

值得注意的是，即使在上述分散的环境中，中央机构仍然可能负责制定学习任务。例如，考虑以下问题：谁决定在去中心化环境中要训练的模型是什么？使用什么算法？什么超参数？当某些东西没有按预期工作时，谁负责调试？要回答这些问题，仍然需要参与的客户对中央机构有一定程度的信任。或者，可以由提出学习任务的客户做出决定，或者通过共识方案进行协作（见第2.1.2 节）。

表3提供了联邦学习和对等学习之间的比较。虽然分散式学习的架构假设与联邦学习的架构假设不同，但它通常可以应用于类似的问题领域，出现许多相同的挑战，并且在研究社区中存在显著的重叠。因此，我们在本文中也考虑了分散式学习;在本节中，

[2]然而，请注意，在这种情况下，回合的概念甚至不需要有意义。例如，参见[85 ]中关于时钟模型的讨论。

| | Federated learning | Fully decentralized (peer-to-peer) learning |
|---|---|---|
| Orchestration | A central orchestration server or service organizes the training, but never sees raw data. | No centralized orchestration. |
| Wide-area communication | Typically a hub-and-spoke topology, with the hub representing a coordinating service provider (typically without data) and the spokes connecting to clients. | Peer-to-peer topology, with a possibly dynamic connectivity graph. |

Table 3: A comparison of the key distinctions between federated learning and fully decentralized learning. Note that as with FL, decentralized learning can be further divided into different use-cases, with distinctions similar to those made in Table 1 comparing cross-silo and cross-device FL.

specific to the decentralized approach are explicitly considered, but many of the open problems in other sections also arise in the decentralized case.

### 2.1.1 Algorithmic Challenges

A large number of important algorithmic questions remain open on the topic of real-world usability of decentralized schemes for machine learning. Some questions are analogous to the special case of federated learning with a central server, and other challenges come as an additional side-effect of being fully decentralized or trust-less. We outline some particular areas in the following.

**Effect of network topology and asynchrony on decentralized SGD** Fully decentralized algorithms for learning should be robust to the limited availability of the clients (with clients temporarily unavailable, dropping out or joining during the execution) and limited reliability of the network (with possible message drops). While for the special case of generalized linear models, schemes using the duality structure could enable some of these desired robustness properties [231], for the case of deep learning and SGD this remains an open question. When the network graph is complete but messages have a fixed probability to be dropped, Yu et al. [498] show that one can achieve convergence rates that are comparable to the case of a reliable network. Additional open research questions concern non-IID data distributions, update frequencies, efficient communication patterns and practical convergence time [443], as we outline in more detail below.

Well-connected or denser networks encourage faster consensus and give better theoretical convergence rates, which depend on the spectral gap of the network graph. However, when data is IID, sparser topologies do not necessarily hurt the convergence in practice: this was analyzed theoretically in [357]. Denser networks typically incur communication delays which increase with the node degrees. Most of optimization-theory works do not explicitly consider how the topology affects the runtime, that is, wall-clock time required to complete each SGD iteration. Wang et al. [469] propose MATCHA, a decentralized SGD method based on matching decomposition sampling, that reduces the communication delay per iteration for any given node topology while maintaining the same error convergence speed. The key idea is to decompose the graph topology into matchings consisting of disjoint communication links that can operate in parallel, and carefully choose a subset of these matchings in each iteration. This sequence of subgraphs results in more

| | 联邦学习完全去中心化 | （peer -to - peer ） learning |
|---|---|---|
| 业务流程中央业务流程服务器或服务器 | Vice负责组织培训，但从不查看原始数据。 | 没有集中的编排。 |
| 广域通信通常是中心辐射式拓扑， | 其中集线器表示协调服务提供商（通常没有数据），而辐条连接到客户端。 | 对等拓扑结构，具有可能的动态连接图。 |

表3：联邦学习和完全分散学习之间的主要区别比较。请注意，与FL一样，分散式学习可以进一步划分为不同的用例，其区别类似于表1中比较跨筒仓和跨设备FL的区别。

具体到分散的方法是明确考虑，但在其他部分的许多开放的问题也出现在分散的情况下。

### 2.1.1 学术挑战

关于机器学习分散方案的现实可用性，大量重要的算法问题仍然是开放的。有些问题类似于使用中央服务器的联邦学习的特殊情况，而其他挑战则是完全分散或无信任的额外副作用。我们在下面概述了一些特定领域。

网络拓扑结构和分布式学习算法对去中心化SGD的影响完全去中心化的学习算法应该对有限的客户端可用性（客户端暂时不可用，在执行期间退出或加入）和有限的网络可靠性（可能的消息丢失）具有鲁棒性。虽然对于广义线性模型的特殊情况，使用对偶结构的方案可以实现这些期望的鲁棒性[231 ]，但对于深度学习和SGD的情况，这仍然是一个悬而未决的问题。当网络图是完整的，但消息有一个固定的概率被丢弃时，Yu等人。[498 ]表明，可以实现与可靠网络的情况相当的收敛速度。其他开放的研究问题涉及非IID数据分布，更新频率，有效的通信模式和实际收敛时间[443 ]，我们在下面更详细地概述。

良好连接或更密集的网络鼓励更快的共识，并给予更好的理论收敛速度，这取决于网络图的谱间隙。然而，当数据是IID时，稀疏拓扑不一定会在实践中损害收敛：这在[357 ]中进行了理论分析。密集网络通常会导致通信延迟，其随着节点度的增加而增加。大多数优化理论的工作没有明确考虑拓扑如何影响运行时，即完成每个SGD迭代所需的挂钟时间。Wang et al. [469 ]提出了MATCHA，一种基于匹配分解采样的分散式SGD方法，可以减少任何给定节点拓扑的每次迭代的通信延迟，同时保持相同的误差收敛速度。其关键思想是将图拓扑结构分解成由不相交的通信链路组成的匹配，这些匹配可以并行操作，并在每次迭代中仔细选择这些匹配的子集。 这个子图序列导致更多

frequent communication over connectivity-critical links (ensuring fast error convergence) and less frequent communication over other links (saving communication delays).

The setting of decentralized SGD also naturally lends itself to asynchronous algorithms in which each client becomes active independently at random times, removing the need for global synchronization and potentially improving scalability [127, 459, 59, 29, 306].

**Local-update decentralized SGD** The theoretical analysis of schemes which perform several local update steps before a communication round is significantly more challenging than those using a single SGD step, as in mini-batch SGD. While this will also be discussed later in Section 3.2, the same also holds more generally in the fully decentralized setting of interest here. Schemes relying on a single local update step are typically proven to converge in the case of non-IID local datasets [278, 279]. For the case with several local update steps, [467, 280] recently provided convergence analysis. Further, [469] provides a convergence analysis for the non-IID data case, but for the specific scheme based on matching decomposition sampling described above. In general, however, understanding the convergence under non-IID data distributions and how to design a model averaging policy that achieves the fastest convergence remains an open problem.

**Personalization, and trust mechanisms** Similarly to the cross-device FL setting, an important task for the fully decentralized scenario under the non-IID data distributions available to individual clients is to design algorithms for learning collections of personalized models. The work of [459, 59] introduces fully decentralized algorithms to collaboratively learn a personalized model for each client by smoothing model parameters across clients that have similar tasks (i.e., similar data distributions). Zantedeschi et al. [504] further learn the similarity graph together with the personalized models. One of the key unique challenges in the decentralized setting remains the robustness of such schemes to malicious actors or contribution of unreliable data or labels. The use of incentives or mechanism design in combination with decentralized learning is an emerging and important goal, which may be harder to achieve in the setting without a trusted central server.

**Gradient compression and quantization methods** In potential applications, the clients would often be limited in terms of communication bandwidth available and energy usage permitted. Translating and generalizing some of the existing compressed communication schemes from the centralized orchestrator-facilitated setting (see Section 3.5) to the fully decentralized setting, without negatively impacting the convergence is an active research direction [278, 391, 444, 279]. A complementary idea is to design decentralized optimization algorithms which naturally give rise to sparse updates [504].

**Privacy** An important challenge in fully decentralized learning is to prevent any client from reconstructing the private data of another client from its shared updates while maintaining a good level of utility for the learned models. Differential privacy (see Section 4) is the standard approach to mitigate such privacy risks. In decentralized federated learning, this can be achieved by having each client add noise locally, as done in [239, 59]. Unfortunately, such local privacy approaches often come at a large cost in utility. Furthermore, distributed methods based on secure aggregation or secure shuffling that are designed to improve the privacy-utility trade-off in the standard FL setting (see Section 4.4.3) do not easily integrate with fully decentralized algorithms. A possible direction to achieve better trade-offs between privacy and utility in fully decentralized algorithms is to rely on decentralization itself to amplify differential privacy guarantees, for instance by considering appropriate relaxations of local differential privacy [146].

这种子图序列导致在连接性关键链路上更频繁的通信（确保快速错误收敛）和在其他链路上更不频繁的通信（节省通信延迟）。

去中心化SGD的设置也自然适合异步算法，其中每个客户端在随机时间独立地变得活跃，消除了对全局同步的需要，并可能提高可扩展性[127，459，59，29，306 ]。

局部更新分散SGD在一轮通信之前执行几个局部更新步骤的方案的理论分析比使用单个SGD步骤的方案更具挑战性，如在小批量SGD中。虽然这一点也将在后面的3.2 节中讨论，但在这里，在完全分散的利益环境中，这一点也更普遍地成立。依赖于单个本地更新步骤的方案通常被证明在非IID本地数据集的情况下收敛[278，279 ]。对于有几个局部更新步骤的情况，[467，280 ]最近提供了收敛分析。此外，[469 ]提供了非IID数据情况下的收敛分析，但针对基于上述匹配分解采样的特定方案。然而，在一般情况下，理解非IID数据分布下的收敛以及如何设计实现最快收敛的模型平均策略仍然是一个悬而未决的问题。

个性化和信任机制与跨设备FL设置类似，在个人客户端可用的非IID数据分布下，完全去中心化场景的一项重要任务是设计用于学习个性化模型集合的算法。[459，59 ]的工作引入了完全分散的算法，通过在具有类似任务的客户端之间平滑模型参数来协作学习每个客户端的个性化模型（即，类似的数据分布）。Zantedeschi 等人。[504 ]进一步学习相似性图以及个性化模型。去中心化环境中的一个关键的独特挑战仍然是这种方案对恶意行为者的鲁棒性或不可靠数据或标签的贡献。将激励或机制设计与分散式学习相结合是一个新兴的重要目标，在没有可信的中央服务器的情况下可能更难实现。

梯度压缩和量化方法在潜在的应用中，客户端通常会在可用的通信带宽和允许的能量使用方面受到限制。将一些现有的压缩通信方案从集中式协调器便利的设置（见第3.5 节）转换和推广到完全分散的设置，而不会对收敛产生负面影响，这是一个积极的研究方向[278，391，444，279 ]。一个互补的想法是设计分散的优化算法，这自然会给予稀疏更新[504 ]。

完全去中心化学习的一个重要挑战是防止任何客户端从其共享更新中重建另一个客户端的私有数据，同时保持学习模型的良好实用性。差异隐私（见第4 节）是减轻此类隐私风险的标准方法。在分散式联邦学习中，这可以通过让每个客户端在本地添加噪声来实现，如[239，59 ]所做的那样。不幸的是，这种本地隐私方法通常在实用性方面付出很大代价。此外，基于安全聚合或安全洗牌的分布式方法旨在改善标准FL设置中的隐私效用权衡（见第4.4.3 节），不容易与完全分散的算法集成。 在完全去中心化算法中实现隐私和效用之间更好权衡的一个可能方向是依靠去中心化本身来放大差分隐私保证，例如通过考虑适当放松局部差分隐私[146 ]。

### 2.1.2 Practical Challenges

An orthogonal question for fully decentralized learning is how it can be practically realized. This section outlines a family of related ideas based on the idea of a distributed ledger, but other approaches remain unexplored.

A blockchain is a distributed ledger shared among disparate users, making possible digital transactions, including transactions of cryptocurrency, without a central authority. In particular, smart contracts allow execution of arbitrary code on top of the blockchain, essentially a massively replicated eventually-consistent state machine. In terms of federated learning, use of the technology could enable decentralization of the global server by using smart contracts to do model aggregation, where the participating clients executing the smart contracts could be different companies or cloud services.

However, on today's blockchain platforms such as Ethereum [478], data on the blockchains is publicly available by default, this could discourage users from participating in the decentralized federated learning protocol, as the protection of the data is typically the primary motivating factor for FL. To address such concerns, it might be possible to modify the existing privacy-preserving techniques to fit into the scenario of decentralized federated learning. First of all, to prevent the participating nodes from exploiting individually submitted model updates, existing secure aggregation protocols could be used. A practical secure aggregation protocol already used in cross-device FL was proposed by Bonawitz et al. [80], effectively handling dropping out participants at the cost of complexity of the protocol. An alternative system would be to have each client stake a deposit of cryptocurrency on blockchain, and get penalized if they drop out during the execution. Without the need of handling dropouts, the secure aggregation protocol could be significantly simplified. Another way of achieving secure aggregation is to use confidential smart contract such as what is enabled by the Oasis Protocol [119] which runs inside secure enclaves. With this, each client could simply submit an encrypted local model update, knowing that the model will be decrypted and aggregated inside the secure hardware through remote attestation (though see discussion of privacy-in-depth in Section 4.1).

In order to prevent any client from trying to reconstruct the private data of another client by exploiting the global model, client-level differential privacy [338] has been proposed for FL. Client-level differential privacy is achieved by adding random Gaussian noise on the aggregated global model that is enough to hide any single client's update. In the context of blockchain, each client could locally add a certain amount of Gaussian noise after local gradient descent steps and submit the model to blockchain. The local noise scale should be calculated such that the aggregated noise on blockchain is able to achieve the same client-level differential privacy as in [338]. Finally, the aggregated global model on blockchain could be encrypted and only the participating clients hold the decryption key, which protects the model from the public.

## 2.2 Cross-Silo Federated Learning

In contrast with the characteristics of cross-device federated learning, see Table 1, cross-silo federated learning admits more flexibility in certain aspects of the overall design, but at the same time presents a setting where achieving other properties can be harder. This section discusses some of these differences.

The cross-silo setting can be relevant where a number of companies or organizations share incentive to train a model based on all of their data, but cannot share their data directly. This could be due to constraints imposed by confidentiality or due to legal constraints, or even within a single company when they cannot centralize their data between different geographical regions. These cross-silo applications have attracted substantial attention.

### 2.1.2 实际挑战

完全分散学习的一个正交问题是如何实际实现。本节概述了一系列基于分布式账本思想的相关想法，但其他方法尚未探索。

区块链是在不同用户之间共享的分布式分类账，使数字交易（包括加密货币交易）成为可能，而无需中央机构。特别是，智能合约允许在区块链上执行任意代码，本质上是一个大规模复制的最终一致状态机。在联合学习方面，使用该技术可以通过使用智能合约进行模型聚合来实现全球服务器的分散化，其中执行智能合约的参与客户端可以是不同的公司或云服务。

然而，在今天的区块链平台上，如以太坊[478 ]，区块链上的数据默认情况下是公开可用的，这可能会阻止用户参与分散式联邦学习协议，因为数据保护通常是FL的主要激励因素。也许可以修改现有的隐私保护技术，以适应分散式联合学习的情况。首先，为了防止参与节点利用单独提交的模型更新，可以使用现有的安全聚合协议。Bonawitz 等人[80 ]提出了一种已经在跨设备FL中使用的实用安全聚合协议，有效地处理了以协议复杂性为代价的退出参与者。另一种系统是让每个客户在区块链上存入加密货币的存款，如果他们在执行过程中退出，就会受到惩罚。 在不需要处理丢弃的情况下，可以显著简化安全聚合协议。实现安全聚合的另一种方法是使用机密智能合约，例如在安全飞地内运行的Oasis 协议[119 ]。这样，每个客户端可以简单地提交加密的本地模型更新，知道模型将通过远程证明在安全硬件内被解密和聚合（尽管参见第4.1 节中对隐私的深入讨论）。

为了防止任何客户端试图通过利用全局模型来重建另一个客户端的隐私数据，已经为FL提出了客户端级别的差分隐私[338 ]。客户端级别的差分隐私是通过在聚合全局模型上添加随机高斯噪声来实现的，该噪声足以隐藏任何单个客户端的更新。在区块链的上下文中，每个客户端可以在局部梯度下降步骤之后本地添加一定量的高斯噪声，并将模型提交给区块链。应该计算本地噪声尺度，以便区块链上的聚合噪声能够实现与[338 ]相同的客户端级别的差异隐私。最后，区块链上的聚合全局模型可以被加密，只有参与的客户端持有解密密钥，从而保护模型不被公开。

## 2.2 跨筒仓联合学习

与跨设备联邦学习的特征相比，如表1所示，跨竖井联邦学习在整体设计的某些方面具有更大的灵活性，但同时也提供了一种实现其他属性可能更困难的设置。本节讨论其中的一些差异。

跨筒仓设置可能与许多公司或组织共享激励以基于其所有数据训练模型，但不能直接共享其数据有关。这可能是由于保密性或法律的约束所造成的限制，甚至是在一家公司内部，当他们无法集中不同地理区域之间的数据时。这些跨筒仓的应用程序已经引起了极大的关注。

**Data partitioning** In the cross-device setting the data is assumed to be partitioned by examples. In the cross-silo setting, in addition to partitioning by examples, partitioning by features is of practical relevance. An example could be when two companies in different businesses have the same or overlapping set of customers, such as a local bank and a local retail company in the same city. This difference has been also referred to as horizontal and vertical federated learning by Yang et al. [490].

Cross-silo FL with data partitioned by features, employs a very different training architecture compared to the setting with data partitioned by example. It may or may not involve a central server as a neutral party, and based on specifics of the training algorithm, clients exchange specific intermediate results rather than model parameters, to assist other parties' gradient calculations; see for instance [490, Section 2.4.2]. In this setting, application of techniques such as secure multi-party computation or homomorphic encryption have been proposed in order to limit the amount of information other participants can infer from observing the training process. The downside of this approach is that the training algorithm is typically dependent on the type of machine learning objective being pursued. Currently proposed algorithms include trees [118], linear and logistic regression [490, 224, 316], and neural networks [317]. Local updates similar to Federated Averaging (see Section 3.2) has been proposed to address the communication challenges of feature-partitioned systems [316], and [238, 318] study the security and privacy related challenges inherent in such systems.

Federated transfer learning [490] is another concept that considers challenging scenarios in which data parties share only a partial overlap in the user space or the feature space, and leverage existing transfer learning techniques [365] to build models collaboratively. The existing formulation is limited to the case of 2 clients.

Partitioning by examples is usually relevant in cross-silo FL when a single company cannot centralize their data due to legal constraints, or when organizations with similar objectives want to collaboratively improve their models. For instance, different banks can collaboratively train classification or anomaly detection models for fraud detection [476], hospitals can build better diagnostic models [139], and so on.

An open-source platform supporting the above outlined applications is currently available as *Federated AI Technology Enabler (FATE)* [33]. At the same time, the IEEE P3652.1 Federated Machine Learning Working Group is focusing on standard-setting for the Federated AI Technology Framework. Other platforms include [125] focused on a range of medical applications and [321] for enterprise use cases. See Appendix A for more details.

**Incentive mechanisms** In addition to developing new algorithmic techniques for FL, incentive mechanism design for honest participation is an important practical research question. This need may arise in cross-device settings (e.g. [261, 260]), but is particularly relevant in the cross-silo setting, where participants may at the same time also be business competitors. The incentive can be in the form of monetary payout [499] or final models with different levels of performance [324]. The option to deliver models with performance commensurate to the contributions of each client is especially relevant in collaborative learning situations in which competitions exist among FL participants. Clients might worry that contributing their data to training federated learning models will benefit their competitors, who do not contribute as much but receive the same final model nonetheless (i.e. the free-rider problem). Related objectives include how to divide earnings generated by the federated learning model among contributing data owners in order to sustain long-term participation, and also how to link the incentives with decisions on defending against adversarial data owners to enhance system security, optimizing the participation of data owners to enhance system efficiency.

数据分区在跨设备设置中，数据被假设为通过示例进行分区。在跨竖井设置中，除了按示例划分之外，按特性划分也具有实际意义。一个例子可能是当两个公司在不同的业务有相同或重叠的客户集，如当地银行和当地零售公司在同一个城市。这种差异也被Yang等人称为水平和垂直联邦学习[490 ]。

跨筒仓FL与按特征划分的数据相比，采用了一种非常不同的训练体系结构，与按示例划分的数据相比。它可能会也可能不会涉及作为中立方的中央服务器，并且基于训练算法的细节，客户端交换特定的中间结果而不是模型参数，以帮助其他方的梯度计算;参见例如[490 ，第2.4.2 节]。在这种情况下，已经提出了诸如安全多方计算或同态加密之类的技术的应用，以便限制其他参与者可以从观察训练过程中推断出的信息量。这种方法的缺点是训练算法通常取决于所追求的机器学习目标的类型。目前提出的算法包括树[118 ]，线性和逻辑回归[490，224，316 ]和神经网络[317 ]。本地更新类似于联合平均（见第3节）。2)已被提出来解决功能分区系统的通信挑战[316 ]，并[238，318 ]研究这种系统中固有的安全和隐私相关的挑战。

联合迁移学习[490 ]是另一个考虑挑战性场景的概念，其中数据方仅共享用户空间或特征空间中的部分重叠，并利用现有的迁移学习技术[365 ]来协作构建模型。现有的公式仅限于2个客户的情况。当单个公司由于法律的限制而无法集中其数据时，或者当具有类似目标的组织希望协作改进其模型时，通过示例进行分区通常与跨筒仓FL相关。例如，不同的银行可以协同训练分类或异常检测模型以进行欺诈检测[476 ]，医院可以构建更好的诊断模型[139 ]，等等。

支持上述应用程序的开源平台目前可作为Federated AI Technology Enabler （FATE）[33 ]。与此同时，IEEEP3652.1 联邦机器学习工作组正在专注于联邦人工智能技术框架的标准制定。其他平台包括[125 ]专注于一系列医疗应用和[321 ]用于企业用例。更多详情请参见附录A。

激励机制-除了开发新的算法技术FL，激励机制设计诚实的参与是一个重要的实际研究问题。这种需求可能出现在跨设备设置（例如[261，260 ]）中，但在跨竖井设置中尤其相关，其中参与者可能同时也是业务竞争对手。激励可以是货币支付的形式[499 ]或具有不同绩效水平的最终模型[324 ]。选择提供与每个客户端的贡献相称的性能模型是特别相关的合作学习的情况下，在FL参与者之间存在的竞争。客户可能会担心，将他们的数据贡献给训练联邦学习模型将使他们的竞争对手受益，他们没有贡献那么多，但仍然收到相同的最终模型（即搭便车问题）。 相关目标包括如何在贡献数据所有者之间分配联邦学习模型产生的收益，以维持长期参与，以及如何将激励措施与防御敌对数据所有者的决策联系起来，以提高系统安全性，优化数据所有者的参与，以提高系统效率。

**Differential privacy** The discussion of actors and threat models in Section 4.1 is largely relevant also for the cross-silo FL. However, protecting against different actors might have different priorities. For example, in many practical scenarios, the final trained model would be released only to those who participate in the training, which makes the concerns about "the rest of the world" less important.

On the other hand, for a practically persuasive claim, we would usually need a notion of local differential privacy, as the potential threat from other clients is likely to be more important. In cases when the clients are not considered a significant threat, each client could control the data from a number of their respective users, and a formal privacy guarantee might be needed on such user-level basis. Depending on application, other objectives could be worth pursuing. This area has not been systematically explored.

**Tensor factorization** Several works have also studied cross-silo federated tensor factorization where multiple sites (each having a set of data with the same feature, i.e. horizontally partitioned) jointly perform tensor factorization by only sharing intermediate factors with the coordination server while keeping data private at each site. Among the existing works, [272] used an alternating direction method of multipliers (ADMM) based approach and [325] improved the efficiency with the elastic averaging SGD (EASGD) algorithm and further ensures differential privacy for the intermediate factors.

## 2.3 Split Learning

In contrast with the previous settings which focus on data partitioning and communication patterns, the key idea behind split learning [215, 460][3] is to split the execution of a model on a per-layer basis between the clients and the server. This can be done for both training and inference.

In the simplest configuration of split learning, each client computes the forward pass through a deep network up to a specific layer referred to as the *cut layer*. The outputs at the cut layer, referred to as *smashed data*, are sent to another entity (either the server or another client), which completes the rest of the computation. This completes a round of forward propagation without sharing the raw data. The gradients can then be back propagated from its last layer until the cut layer in a similar fashion. The gradients at the cut layer – and only these gradients – are sent back to the clients, where the rest of back propagation is completed. This process is continued until convergence, without having clients directly access each others raw data. This setup is shown in Figure 2(a) and a variant of this setup where labels are also not shared along with raw data is shown in Figure 2(b). Split learning approaches for data partitioned by features have been studied in [101].

In several settings, the overall communication requirements of split learning and federated learning were compared in [421]. Split learning brings in another dimension of parallelism in the training, parallelization among parts of a model, e.g. client and server. The ideas in [245, 240], where the authors break the dependencies between partial networks and reduced total centralized training time by parallelizing the computations in different parts, can be relevant here as well. However, it is still an open question to explore such parallelization of split learning on edge devices. Split learning also enables matching client-side model components with the best server-side model components for automating model selection as shown in the ExpertMatcher [413].

The values communicated can nevertheless, in general, reveal information about the underlying data. How much, and whether this is acceptable, is likely going to be application and configuration specific. A variation of split learning called NoPeek SplitNN [462] reduces the potential leakage via communicated activations, by reducing their distance correlation [461, 442] with the raw data, while maintaining good model

[3] See also split learning project website - `https://splitlearning.github.io/`.

第4.1 节中对参与者和威胁模型的讨论在很大程度上也与跨竖井FL相关。然而，针对不同参与者的保护可能具有不同的优先级。例如，在许多实际场景中，最终的训练模型只会发布给那些参与培训的人，这使得对“ 世界其他地方”的关注变得不那么重要。

另一方面，对于一个实际上有说服力的主张，我们通常需要一个局部差异隐私的概念，因为来自其他客户端的潜在威胁可能更重要。在客户端不被认为是重大威胁的情况下，每个客户端可以控制来自其各自用户的数据，并且可能需要基于这种用户级别的正式隐私保证。根据应用情况，其他目标也值得追求。这一领域尚未得到系统的探讨。

一些工作还研究了跨竖井联合张量因式分解，其中多个站点（每个站点具有一组具有相同特征的数据，即水平分区的数据）通过仅与协调服务器共享中间因子同时在每个站点保持数据私有来联合执行张量因式分解。在现有的工作中，[272 ]使用了基于交替方向乘法器（ADMM ）的方法，[325 ]提高了弹性平均SGD（EASGD）算法的效率，并进一步确保了中间因素的差异隐私。

## 2.3 分裂学习

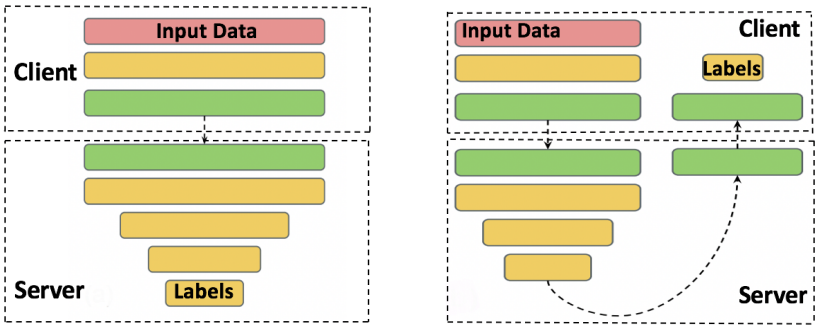
与之前专注于数据划分和通信模式的设置相比，分裂学习[215，460 ]背后的关键思想是在客户端和服务器之间以每层为基础分割模型的执行。这可以用于训练和推理。

在分裂学习的最简单配置中，每个客户端计算通过深度网络直到被称为切割层的特定层的前向传递。剪切层的输出（称为粉碎数据）被发送到另一个实体（服务器或另一个客户端），完成剩余的计算。这就完成了一轮前向传播，而无需共享原始数据。然后，梯度可以以类似的方式从其最后一层反向传播，直到切割层。剪切层上的梯度（只有这些梯度）被发送回客户端，在那里完成反向传播的其余部分。这个过程一直持续到收敛，而不需要客户端直接访问彼此的原始数据。这种设置如图2（a）所示，这种设置的一个变体如图2（b）所示，其中标签也不与原始数据一起沿着。在[101 ]中研究了按特征划分的数据的分裂学习方法。

在几种设置中，分离学习和联合学习的总体通信需求在[421 ]中进行了比较。分裂学习在训练中引入了另一个并行维度，即模型各部分之间的并行化，例如客户端和服务器。[245，240 ]中的思想，作者打破了部分网络之间的依赖关系，并通过在不同部分并行计算来减少总的集中训练时间，在这里也是相关的。然而，在边缘设备上探索这种分裂学习的并行化仍然是一个悬而未决的问题。拆分学习还可以将客户端模型组件与最佳服务器端模型组件进行匹配，以自动选择模型，如ExpertMatcher [413 ]所示。

然而，一般来说，传递的值可以揭示有关基础数据的信息。
多少以及这是否可以接受，可能是特定于应用程序和配置的。一种称为NoPeek SplitNN [462 ]的分裂学习变体通过减少与原始数据的距离相关性[461，442 ]来减少通过通信激活的潜在泄漏，同时保持良好的模型

3另见分裂学习项目网站-https ://splitlearning .github .io/。




(a) Vanilla split learning (b) U-shaped split learning

Figure 2: Split learning configurations showing raw data is not transferred in the vanilla setting and that raw data as well as labels are not transferred between the client and server entities in the U-shaped split learning setting.

performance via categorical cross-entropy. The key idea is to minimize the distance correlation between the raw data points and communicated smashed data. The objects communicated could otherwise contain information highly correlated with the input data if used without NoPeek SplitNN, the use of which also enables the split to be made relatively early-on given the decorrelation it provides. One other engineering driven approach to minimize the amount of information communicated in split learning has been via a specifically learnt pruning of channels present in the client side activations [422]. Overall, much of the discussion in Section 4 is relevant here as well, and analysis providing formal privacy guarantees specifically for split learning is still an open problem.

## 2.4 Executive summary

The motivation for federated learning is relevant for a number of related areas of research.

- Fully decentralized learning (Section 2.1) removes the need for a central server coordinating the overall computation. Apart from algorithmic challenges, open problems are in practical realization of the idea and in understanding of what form of trusted central authority is needed to set up the task.
- Cross-silo federated learning (Section 2.2) admits problems with different kinds of modelling constraints, such as data partitioned by examples and/or features, and faces different set of concerns when formulating formal privacy guarantees or incentive mechanisms for clients to participate.
- Split learning (Section 2.3) is an approach to partition the execution of a model between the clients and the server. It can deliver different options for overall communication constraints, but detailed analysis of when the communicated values reveal sensitive information is still missing.


(a)Vanilla分裂学习（B）U形分裂学习

图二：分割学习配置显示原始数据在普通设置中不被传输，并且原始数据以及标签在U形分割学习设置中不在客户端和服务器实体之间传输。

性能通过分类交叉熵。其关键思想是最小化原始数据点和通信的粉碎数据之间的距离相关性。如果不使用 NoPeek SplitNN ，则所通信的对象可以包含与输入数据高度相关的信息，使用NoPeek SplitNN 还可以在其提供的去相关的情况下相对较早地进行分割。另一种工程驱动的方法是通过对客户端激活中存在的通道进行专门学习的修剪来最大限度地减少分离学习中传达的信息量[422 ]。总的来说，第4节中的大部分讨论在这里也是相关的，专门为分裂学习提供正式隐私保证的分析仍然是一个悬而未决的问题。

### 2.4 执行摘要

联邦学习的动机与许多相关的研究领域有关。

完全分散的学习（第2.1 节）消除了对协调整体计算的中央服务器的需求。除了算法上的挑战，开放的问题是在实际实现的想法，并在了解什么形式的可信的中央权威是需要建立的任务。

跨筒仓联邦学习（第2.2 节）承认不同类型的建模约束存在问题，例如通过示例和/ 或特征划分的数据，并且在制定正式的隐私保证或激励机制时面临不同的问题。

分裂学习（2.3 节）是一种在客户端和服务器之间划分模型执行的方法。它可以为整体通信约束提供不同的选项，但仍然缺少对通信值何时泄露敏感信息的详细分析。

# 3 Improving Efficiency and Effectiveness

In this section we explore a variety of techniques and open questions that address the challenge of making federated learning more efficient and effective. This encompasses a myriad of possible approaches, including: developing better optimization algorithms; providing different models to different clients; making ML tasks like hyperparameter search, architecture search, and debugging easier in the FL context; improving communication efficiency; and more.

One of the fundamental challenges in addressing these goals is the presence of non-IID data, so we begin by surveying this issue and highlighting potential mitigations.

## 3.1 Non-IID Data in Federated Learning

While the meaning of IID is generally clear, data can be non-IID in many ways. In this section, we provide a taxonomy of non-IID data regimes that may arise for any client-partitioned dataset. The most common sources of dependence and non-identicalness are due to each client corresponding to a particular user, a particular geographic location, and/or a particular time window. This taxonomy has a close mapping to notions of dataset shift [353, 380], which studies differences between the training distribution and testing distribution; here, we consider differences in the data distribution on each client.

For the following, consider a supervised task with features $x$ and labels $y$. A statistical model of federated learning involves two levels of sampling: accessing a datapoint requires first sampling a client $i \sim \mathcal{Q}$, the distribution over available clients, and then drawing an example $(x, y) \sim \mathcal{P}_i(x, y)$ from that client's local data distribution.

When non-IID data in federated learning is referenced, this typically refers to differences between $\mathcal{P}_i$ and $\mathcal{P}_j$ for different clients $i$ and $j$. However, it is also important to note that the distribution $\mathcal{Q}$ and $\mathcal{P}_i$ may change over time, introducing another dimension of "non-IIDness".

For completeness, we note that even considering the dataset on a single device, if the data is in an insufficiently-random order, e.g. ordered by time, then independence is violated locally as well. For example, consecutive frames in a video are highly correlated. Sources of intra-client correlation can generally be resolved by local shuffling.

**Non-identical client distributions** We first survey some common ways in which data tend to deviate from being identically distributed, that is $P_i \neq P_j$ for different clients $i$ and $j$. Rewriting $P_i(x, y)$ as $P_i(y \mid x)P_i(x)$ and $P_i(x \mid y)P_i(y)$ allows us to characterize the differences more precisely.

- *Feature distribution skew* (covariate shift): The marginal distributions $\mathcal{P}_i(x)$ may vary across clients, even if $\mathcal{P}(y \mid x)$ is shared.[4] For example, in a handwriting recognition domain, users who write the same words might still have different stroke width, slant, etc.
- *Label distribution skew* (prior probability shift): The marginal distributions $\mathcal{P}_i(y)$ may vary across clients, even if $\mathcal{P}(x \mid y)$ is the same. For example, when clients are tied to particular geo-regions, the distribution of labels varies across clients — kangaroos are only in Australia or zoos; a person's face is only in a few locations worldwide; for mobile device keyboards, certain emoji are used by one demographic but not others.

[4] We write "$\mathcal{P}(y \mid x)$ is shared" as shorthand for $\mathcal{P}_i(y \mid x) = \mathcal{P}_j(y \mid x)$ for all clients $i$ and $j$.

# 3提高效率和效益

在本节中，我们将探讨各种技术和开放性问题，以解决使联邦学习更加高效和有效的挑战。这包括无数可能的方法，包括：开发更好的优化算法;为不同的客户端提供不同的模型;使ML任务（如超参数搜索，架构搜索和FL环境中的调试）更容易;提高通信效率;等等。

实现这些目标的根本挑战之一是存在非IID数据，因此我们开始调查这个问题并强调潜在的缓解措施。

## 3.1 联邦学习中的非IID数据

虽然IID的含义通常是明确的，但数据在许多方面可以是非IID的。在本节中，我们提供了一个非IID数据体系的分类，这些数据体系可能会出现在任何客户端分区的数据集上。依赖性和非同一性的最常见来源是由于每个客户端对应于特定用户、特定地理位置和/或特定时间窗口。这种分类法与数据集转移的概念有着密切的映射[353，380 ]，它研究了训练分布和测试分布之间的差异;在这里，我们考虑了每个客户端上数据分布的差异。

对于以下内容，考虑具有特征x和标签y的监督任务。联邦学习的统计模型涉及两个级别的采样：访问数据点需要首先对客户端i进行采样，i是可用客户端的分布，然后从该客户端的本地数据分布中绘制一个示例（x，y）P（x，y）。

当引用联邦学习中的非IID数据时，这通常是指不同客户端i和j的P和P之间的差异。然而，同样重要的是要注意，分布Q和P可能会随着时间的推移而变化，从而引入另一个维度的“ 非IID” 。

为了完整性，我们注意到，即使考虑单个设备上的数据集，如果数据处于不一致的随机顺序，例如按时间排序，那么独立性也会在局部被违反。例如，视频中的连续帧高度相关。客户端内相关性的来源通常可以通过本地洗牌来解决。

我们首先调查了一些常见的方式，在这些方式中，数据倾向于偏离同分布，即对于不同的客户i和j，P6 = P。|x）P（x）和P（x| y）P（y）允许我们更精确地描述差异。

- 特征分布偏斜（协变量偏移）：边缘分布P（x）可能会因客户端而异，即使P（y）|例如，在手写识别领域中，书写相同单词的用户可能仍然具有不同的笔画宽度、倾斜度等。

- 标签分布偏斜（先验概率偏移）：边缘分布P（y）可能会因客户端而异，即使P（x）|Y）是一样的。例如，当客户端被绑定到特定地理区域时，标签的分布在客户端之间变化-袋鼠仅在澳大利亚或动物园中;人的脸仅在世界范围内的几个位置中;对于移动终端键盘，某些表情符号被一个人口统计而不是其他人口统计使用。

[4]我们写“ P（y| x）是P（y）的简写|x）= P（y| x）对于所有客户端i和j。

- *Same label, different features* (concept drift): The conditional distributions $\mathcal{P}_i(x \mid y)$ may vary across clients even if $\mathcal{P}(y)$ is shared. The same label $y$ can have very different features $x$ for different clients, e.g. due to cultural differences, weather effects, standards of living, etc. For example, images of homes can vary dramatically around the world and items of clothing vary widely. Even within the U.S., images of parked cars in the winter will be snow-covered only in certain parts of the country. The same label can also look very different at different times, and at different time scales: day vs. night, seasonal effects, natural disasters, fashion and design trends, etc.
- *Same features, different label* (concept shift): The conditional distribution $\mathcal{P}_i(y \mid x)$ may vary across clients, even if $\mathcal{P}(x)$ is the same. Because of personal preferences, the same feature vectors in a training data item can have different labels. For example, labels that reflect sentiment or next word predictors have personal and regional variation.
- *Quantity skew* or unbalancedness: Different clients can hold vastly different amounts of data.

Real-world federated learning datasets likely contain a mixture of these effects, and the characterization of cross-client differences in real-world partitioned datasets is an important open question. Most empirical work on synthetic non-IID datasets (e.g. [337, 236]) have focused on label distribution skew, where a non-IID dataset is formed by partitioning a "flat" existing dataset based on the labels. A better understanding of the nature of real-world non-IID datasets will allow for the construction of controlled but realistic non-IID datasets for testing algorithms and assessing their resilience to different degrees of client heterogeneity.

Further, different non-IID regimes may require the development of different mitigation strategies. For example, under feature-distribution skew, because $\mathcal{P}(y \mid x)$ is assumed to be common, the problem is at least in principle well specified, and training a single global model that learns $\mathcal{P}(y \mid x)$ may be appropriate. When the same features map to different labels on different clients, some form of personalization (Section 3.3) may be essential to learning the true labeling functions.

**Violations of independence** Violations of independence are introduced any time the distribution $\mathcal{Q}$ changes over the course of training; a prominent example is in cross-device FL, where devices typically need to meet eligibility requirements in order to participate in training (see Section 1.1.2). Devices typically meet those requirements at night local time (when they are more likely to be charging, on free wi-fi, and idle), and so there may be significant diurnal patterns in device availability. Further, because local time of day corresponds directly to longitude, this introduces a strong geographic bias in the source of the data. Eichner et al. [171] described this issue and some mitigation strategies, but many open questions remain.

**Dataset shift** Finally, we note that the temporal dependence of the distributions $\mathcal{Q}$ and $\mathcal{P}$ may introduce dataset shift in the classic sense (differences between the train and test distributions). Furthermore, other criteria may make the set of clients eligible to train a federated model different from the set of clients where that model will be deployed. For example, training may require devices with more memory than is needed for inference. These issues are explored in more depth in Section 6. Adapting techniques for handling dataset shift to federated learning is another interesting open question.

### 3.1.1 Strategies for Dealing with Non-IID Data

The original goal of federated learning, training a single global model on the union of client datasets, becomes harder with non-IID data. One natural approach is to modify existing algorithms (e.g. through

·相同的标签，不同的特征（概念漂移）：条件分布P（x| y）可以在客户端之间变化，即使P（y）是共享的。对于不同的客户，同一个标签y可能具有非常不同的特征x，例如，由于文化差异、天气影响、生活标准等。例如，世界各地的房屋图像可能差异很大，服装项目也可能差异很大。即使在美国，冬季停放的汽车的图像将仅在该国的某些地区被雪覆盖。同一个标签在不同的时间和不同的时间尺度上看起来也会有很大的不同：白天与黑夜、季节效应、自然灾害、时尚和设计趋势等。

- 相同的特征，不同的标签（概念转移）：条件分布P（y|即使P（x）相同，客户端之间的P（x）也可能不同。由于个人偏好，训练数据项中的相同特征向量可以具有不同的标签。例如，反映情绪或下一个词预测的标签具有个人和区域差异。

- 数量倾斜或不平衡：不同的客户端可能拥有数量差异很大的数据。

真实世界的联邦学习数据集可能包含这些效应的混合，并且真实世界分区数据集中跨客户端差异的表征是一个重要的开放问题。大多数关于合成非IID数据集的经验工作（例如[337，236 ]）都集中在标签分布偏斜上，其中非IID数据集是通过基于标签划分“ 平坦” 现有数据集而形成的。更好地了解真实世界的非IID数据集的性质将允许构建受控但现实的非IID数据集，用于测试算法并评估其对不同程度的客户端异质性的弹性。

此外，不同的非IID制度可能需要制定不同的缓解战略。例如，在特征分布偏斜下，因为P（y| x）被假设为是常见的，该问题至少在原则上是明确的，并且训练学习P（y）的单个全局模型。|x）可以适当。当相同的特征映射到不同客户端上的不同标签时，某种形式的个性化（第3.3 节）可能对学习真正的标签功能至关重要。

在训练过程中，分布Q发生变化时，都会引入独立性违反;一个突出的例子是跨设备FL，其中设备通常需要满足资格要求才能参加训练（见第1.1.2 节）。设备通常在当地时间的夜间（当它们更有可能充电、使用免费Wi-Fi和空闲时）满足这些要求，因此设备可用性可能存在显著的昼夜模式。此外，由于当地时间直接对应于经度，这在数据源中引入了强烈的地理偏差。Eichner 等人

[171 ]报告描述了这一问题和一些缓解策略，但仍存在许多悬而未决的问题。

最后，我们注意到，分布Q和P的时间依赖性可能会引入经典意义上的数据集移位（训练分布和测试分布之间的差异）。此外，其他标准可以使客户端集合有资格训练与将部署该模型的客户端集合不同的联合模型。例如，训练可能需要具有比推理所需更多存储器的设备。第6节将更深入地探讨这些问题。将处理数据集转移的技术适应于联邦学习是另一个有趣的开放问题。

### 3.1.1 处理非IID数据的策略

联邦学习的最初目标是在客户端数据集的联合上训练一个全局模型，而对于非IID数据，这一目标变得更加困难。一种自然的方法是修改现有的算法（例如，通过

different hyperparameter choices) or develop new ones in order to more effectively achieve this objective. This approach is considered in Section 3.2.2.

For some applications, it may be possible to augment data in order to make the data across clients more similar. One approach is to create a small dataset which can be shared globally. This dataset may originate from a publicly available proxy data source, a separate dataset from the clients' data which is not privacy sensitive, or perhaps a distillation of the raw data following Wang et al. [473].

The heterogeneity of client objective functions gives additional importance to the question of how to craft the objective function — it is no-longer clear that treating all examples equally makes sense. Alternatives include limiting the contributions of the data from any one user (which is also important for privacy, see Section 4) and introducing other notions of fairness among the clients; see discussion in Section 6.

But if we have the capability to run training on the local data on each device (which is necessary for federated learning of a global model), is training a single global model even the right goal? There are many cases where having a single model is to be preferred, e.g. in order to provide a model to clients with no data, or to allow manual validation and quality assurance before deployment. Nevertheless, since local training is possible, it becomes feasible for each client to have a customized model. This approach can turn the non-IID problem from a bug to a feature, almost literally — since each client has its own model, the client's identity effectively parameterizes the model, rendering some pathological but degenerate non-IID distributions trivial. For example, if for each $i$, $\mathcal{P}_i(y)$ has support on only a single label, finding a high-accuracy global model may be very challenging (especially if $x$ is relatively uninformative), but training a high-accuracy local model is trivial (only a constant prediction is needed). Such multi-model approaches are considered in depth in Section 3.3. In addition to addressing non-identical client distributions, using a plurality of models can also address violations of independence stemming from changes in client availability. For example, the approach of Eichner et al. [171] uses a single training run but averages different iterates in order to provide different models for inference based on the timezone / longitude of clients.

## 3.2 Optimization Algorithms for Federated Learning

In prototypical federated learning tasks, the goal is to learn a single global model that minimizes the empirical risk function over the entire training dataset, that is, the union of the data across all the clients. The main difference between federated optimization algorithms and standard distributed training methods is the need to address the characteristics of Table 1 — for optimization, non-IID and unbalanced data, limited communication bandwidth, and unreliable and limited device availability are particularly salient.

FL settings where the total number of devices is huge (e.g. across mobile devices) necessitate algorithms that only require a handful of clients to participate per round (client sampling). Further, each device is likely to participate no more than once in the training of a given model, so stateless algorithms are necessary. This rules out the direct application of a variety of approaches that are quite effective in the datacenter context, for example stateful optimization algorithms like ADMM, and stateful compression strategies that modify updates based on residual compression errors from previous rounds.

Another important practical consideration for federated learning algorithms is composability with other techniques. Optimization algorithms do not run in isolation in a production deployment, but need to be combined with other techniques like cryptographic secure aggregation protocols (Section 4.2.1), differential privacy (DP) (Section 4.2.2), and model and update compression (Section 3.5). As noted in Section 1.1.2, many of these techniques can be applied to primitives like "`sum over selected clients`" and "`broadcast to selected clients`", and so expressing optimization algorithms in terms of these primitives provides a valuable separation of concerns, but may also exclude certain techniques such as ap-

通过不同的超参数选择）或开发新的超参数以更有效地实现该目标。
在第3.2.2 节中考虑了该方法。

对于某些应用程序，可能会增加数据，以便使客户端之间的数据更加相似。一种方法是创建一个可以在全球共享的小数据集。该数据集可能来自公开可用的代理数据源，来自客户端数据的独立数据集，该数据集不对隐私敏感，或者可能是Wang 等人[473 ]的原始数据的蒸馏。

客户端目标函数的异质性使得如何设计目标函数的问题变得更加重要--平等对待所有示例不再有意义。替代方案包括限制任何一个用户的数据贡献（这对隐私也很重要，见第4节），并在客户端之间引入其他公平概念;见第6节的讨论。

但是，如果我们有能力在每个设备上的本地数据上运行训练（这对于全局模型的联邦学习是必要的），那么训练单个全局模型是正确的目标吗？在许多情况下，首选单一模型，例如，为了向没有数据的客户端提供模型，或者允许在部署之前进行手动验证和质量保证。然而，由于当地培训是可能的，因此每个客户都可以拥有定制的模型。这种方法可以将非IID问题从一个bug 转变为一个特性，几乎是字面上的-因为每个客户端都有自己的模型，客户端的身份有效地参数化了模型，使一些病态但退化的非IID分布变得微不足道。 例如，如果对于每个i，P（y）只支持一个标签，那么找到一个高精度的全局模型可能非常具有挑战性（特别是如果x相对没有信息），但是训练一个高精度的局部模型是微不足道的（只需要一个恒定的预测）。3.3 节将深入讨论这种多模型方法。除了解决不相同的客户端分布之外，使用多个模型还可以解决源于客户端可用性的变化的独立性的违反。例如，Eichner 等人的方法。[171 ]使用单个训练运行，但对不同的迭代进行平均，以便根据客户端的时区/经度提供不同的推理模型。

## 3.2 联邦学习的优化算法

在典型的联邦学习任务中，目标是学习一个全局模型，最小化整个训练数据集的经验风险函数，即所有客户端数据的联合。联邦优化算法和标准分布式训练方法之间的主要区别是需要解决表1的特征-对于优化，非IID和不平衡数据，有限的通信带宽以及不可靠和有限的设备可用性特别突出。

设备总数巨大的FL设置（例如，跨移动的设备）需要每轮仅需要少数客户端参与的算法（客户端采样）。此外，每个设备可能只参与给定模型的训练一次，因此无状态算法是必要的。这排除了直接应用在数据中心上下文中非常有效的各种方法，例如ADMM 等有状态优化算法，以及基于前几轮的残余压缩错误修改更新的有状态压缩策略。

联邦学习算法的另一个重要的实际考虑是与其他技术的可组合性。优化算法不会在生产部署中孤立运行，而是需要与其他技术相结合，如加密安全聚合协议（第4.2.1 节），差分隐私（DP）（第4.2.2 节）以及模型和更新压缩（第3.5 节）。如第1.1.2 节所述，这些技术中的许多技术可以应用于原语，如“ 对所选客户端求和”和“ 广播到所选客户端” ，因此根据这些原语表达优化算法提供了有价值的关注点分离，但也可能排除某些技术，如ap 。

| | |
|---|---|
| $N$ | Total number of clients |
| $M$ | Clients per round |
| $T$ | Total communication rounds |
| $K$ | Local steps per round. |

Table 4: Notation for the discussion of FL algorithms including Federated Averaging.

**Server executes:**
initialize $x_0$
**for** each round $t = 1, 2, \ldots,$ T **do**
  $S_t \leftarrow$ (random set of $M$ clients)
  **for** each client $i \in S_t$ **in parallel do**
    $x_{t+1}^i \leftarrow \text{ClientUpdate}(i, x_t)$
  $x_{t+1} \leftarrow \sum_{k=1}^{M} \frac{1}{M} x_{t+1}^i$

**ClientUpdate**$(i, x)$**:**
**for** local step $j = 1, \ldots, K$ **do**
  $x \leftarrow x - \eta \nabla f(x; z)$ for $z \sim \mathcal{P}_i$
return $x$ to server

Algorithm 1: Federated Averaging (local SGD), when all clients have the same amount of data.

plying updates asynchronously.

One of the most common approaches to optimization for federated learning is the Federated Averaging algorithm [337], an adaption of local-update or parallel SGD.[5] Here, each client runs some number of SGD steps locally, and then the updated local models are averaged to form the updated global model on the coordinating server. Pseudocode is given in Algorithm 1.

Performing local updates and communicating less frequently with the central server addresses the core challenges of respecting data locality constraints and of the limited communication capabilities of mobile device clients. However, this family of algorithms also poses several new algorithmic challenges from an optimization theory point of view. In Section 3.2, we discuss recent advances and open challenges in federated optimization algorithms for the cases of IID and non-IID data distribution across the clients respectively. The development of new algorithms that specifically target the characteristics of the federated learning setting remains an important open problem.

### 3.2.1 Optimization Algorithms and Convergence Rates for IID Datasets

While a variety of different assumptions can be made on the per-client functions being optimized, the most basic split is between assuming IID and non-IID data. Formally, having IID data at the clients means that each mini-batch of data used for a client's local update is statistically identical to a uniformly drawn sample (with replacement) from the entire training dataset (the union of all local datasets at the clients). Since the clients independently collect their own training data which vary in both size and distribution, and these data are not shared with other clients or the central node, the IID assumption clearly almost never holds in practice. However, this assumption greatly simplifies theoretical convergence analysis of federated optimization algorithms, as well as establishes a baseline that can be used to understand the impact of non-IID data on optimization rates. Thus, a natural first step is to obtain an understanding of the landscape of optimization algorithms for the IID data case.

[5] Federated Averaging applies local SGD to a randomly sampled subset of clients on each round, and proposes a specific update weighting scheme.

客户总数
每轮M个客户端
交流回合共计
K每轮局部步数。

但是也可以排除某些技术，例如表4：用于讨论包括联合平均的FL算法的符号。

服务器执行：
对于每一轮t = 1，2，. . .，T do
S←（随机的M个客户端集合）

对于每个客户端**i** ∈ **Sin**并行做
x → ClientUpdate （i，x）
$x \leftarrow \sum_{k=1}^{1} \frac{1}{M} x$

**ClientUpdate** （**i**，**x**）：
对于局部步长$j$ = 1，. . .，$K$do
x ← x − η$\mathcal{O}$f （x; z） for z <$\mathcal{P}$
返回x到服务器

算法1：联合平均（本地SGD），当所有客户端具有相同的数据量时。

异步更新。

联邦学习最常见的优化方法之一是联邦平均算法[337 ]，这是本地更新或并行SGD的一种改编。在这里，每个客户端在本地运行一定数量的SGD步骤，然后对更新的本地模型进行平均，以在协调服务器上形成更新的全局模型。算法1中给出了伪代码。

执行本地更新和较不频繁地与中央服务器通信解决了遵守数据局部性约束和移动终端客户端的有限通信能力的核心挑战。然而，这个家庭的算法也提出了一些新的算法的挑战，从优化理论的角度来看。在第3.2 节中，我们讨论了在IID和非IID数据跨客户端分布的情况下，联邦优化算法的最新进展和面临的挑战。开发专门针对联邦学习设置特征的新算法仍然是一个重要的开放问题。

### 3.2.1 IID数据集的优化算法和收敛速度

虽然可以对每个客户端的优化功能进行各种不同的假设，但最基本的划分是假设IID和非IID数据。从形式上讲，在客户端拥有IID数据意味着用于客户端本地更新的每个小批数据在统计上与从整个训练数据集（客户端所有本地数据集的联合）中均匀抽取的样本（具有替换）相同。由于客户端独立地收集它们自己的训练数据，这些数据在大小和分布上都不同，并且这些数据不与其他客户端或中心节点共享，因此IID假设显然在实践中几乎不成立。然而，这种假设大大简化了联邦优化算法的理论收敛性分析，并建立了一个基线，可用于了解非IID数据对优化率的影响。 因此，自然的第一步是了解IID数据情况下的优化算法的前景。

5Federated Averaging 将本地SGD应用于每轮随机采样的客户端子集，并提出特定的更新加权方案。

Formally, for the IID setting let us standardize the stochastic optimization problem

$$\min_{x\in\mathbb{R}^m} F(x) := \mathop{\mathbb{E}}_{z\sim\mathcal{P}}[f(x;z)]\,.$$

We assume an intermittent communication model as in e.g. Woodworth et al. [480, Sec. 4.4], where $M$ stateless clients participate in each of $T$ rounds, and during each round, each client can compute gradients for $K$ samples (e.g. minibatches) $z_1,\ldots,z_K$ sampled IID from $\mathcal{P}$ (possibly using these to take sequential steps). In the IID-data setting clients are interchangeable, and we can without loss of generality assume $M = N$. Table 4 summarizes the notation used in this section.

Different assumptions on $f$ will produce different guarantees. We will first discuss the convex setting and later review results for non-convex problems.

**Baselines and state-of-the-art for convex problems** In this section we review convergence results for $H$-smooth, convex (but not necessarily strongly convex) functions under the assumption that the variance of the stochastic gradients is bounded by $\sigma^2$. More formally, by $H$-smooth we mean that for all $z$, $f(\cdot;z)$ is differentiable and has a $H$-Lipschitz gradient, that is, for all choices of $x, y$

$$\|\nabla f(x,z) - \nabla f(y,z)\| \leq H\|x-y\|.$$

We also assume that for all $x$, the stochastic gradient $\nabla_x f(x;z)$ satisfies

$$\mathop{\mathbb{E}}_{z\sim\mathcal{P}} \|\nabla_x f(x;z) - \nabla F(x)\| \leq \sigma^2.$$

When analyzing the convergence rate of an algorithm with output $x_T$ after $T$ iterations, we consider the term

$$\mathbb{E}[F(x_T)] - F(x^*) \tag{1}$$

where $x^* = \arg\min_x F(x)$. All convergence rates discussed herein are upper bounds on this term. A summary of convergence results for such functions is given in Table 5.

Federated averaging (a.k.a. parallel SGD/local SGD) competes with two natural baselines: First, we may keep $x$ fixed in local updates during each round, and compute a total of $KM$ gradients at the current $x$, in order to run accelerated minibatch SGD. Let $\bar{x}$ denote the average of $T$ iterations of this algorithm. We then have the upper bound

$$\mathcal{O}\left(\frac{H}{T^2} + \frac{\sigma}{\sqrt{TKM}}\right)$$

for convex objectives [294, 137, 151]. Note that the first expectation is taken with respect to the randomness of $z$ in the training procedure as well.

A second natural baseline is to ignore all but 1 of the $M$ active clients, which allows (accelerated) sequential SGD to execute for $KT$ steps. Applying the same general bounds cited above, this approach offers an upper bound of

$$\mathcal{O}\left(\frac{H}{(TK)^2} + \frac{\sigma}{\sqrt{TK}}\right).$$

Comparing these two results, we see that minibatch SGD attains the optimal 'statistical' term ($\sigma/\sqrt{TKM}$), whilst SGD on a single device (ignoring the updates of the other devices) achieves the optimal 'optimization' term ($H/(TK)^2$).

The convergence analysis of local-update SGD methods is an active current area of research [434, 310, 500, 467, 390, 371, 269, 481]. The first convergence results for local-update SGD methods were derived

形式上，对于IID设置，让我们标准化随机优化问题

$$\min_{x\in R} f(x) := \mathop{e}_{z\sim P}[x, x]?$$

我们假设一个间歇性的通信模型，例如Woodworth等人[480，Sec. 4.4]，其中M个无状态客户端参与T轮中的每一轮，并且在每一轮期间，每个客户端可以计算K个样本（例如，小批量）的梯度。. .，zsampled IID from P（可能使用这些来采取顺序步骤）。在IID数据设置中，客户端是可互换的，并且我们可以不失一般性地假设M = N。表4总结了本节中使用的符号。

对f的不同假设将产生不同的保证。我们将首先讨论凸的设置和后来审查结果的非凸问题。

在本节中，我们回顾了H-光滑凸（但不一定是强凸）函数在随机梯度方差以σ为界的假设下的收敛结果。更正式地说，H-光滑我们的意思是，对于所有z，f（·; z）是可微的，并且具有H-Lipschitz梯度，即对于x，y的所有选择，

$$f(x, z) -$$

我们还假设对于所有x，随机梯度函数f（x; z）满足

$$\mathop{E}_{z\sim P} \quad (1) \text{ 正解} f(x, z) f(x) \le \sigma?$$

当分析一个输出为x的算法在T次迭代后的收敛速度时，

$$E[F(x)] - F(x) \quad (1)$$

其中x= arg minF（x）。这里讨论的所有收敛速度都是这一项的上界。表5中给出了这些函数的收敛结果总结。

联合平均（Federated averaging）并行SGD/本地SGD）与两个自然基线竞争：首先，我们可以在每一轮期间在本地更新中保持x固定，并且在当前x处计算总共KM梯度，以便运行加速的小批量SGD。令<$x表示该算法的T次迭代的平均值。然后我们有上界

$$O\left(\frac{H}{T} + \sqrt{\frac{\sigma}{TKM}}\right)$$

对于凸目标[294，137，151]。注意，第一个期望值也是关于训练过程中z的随机性来取的。

第二个自然基线是忽略M个活动客户端中除1个之外的所有活动客户端，这允许（加速）顺序SGD执行KT步。应用上面引用的相同的一般界限，这种方法提供了

$$O\left(\frac{H}{(TK)} + \sqrt{\frac{\sigma}{TK}}\right).$$

比较这两个结果，我们看到小批量SGD获得了最佳“统计”项（/），而单个设备上的SGD（忽略其他设备的更新）获得了最佳“优化”项（/）。

局部更新SGD方法的收敛性分析是当前活跃的研究领域[434，310，500，467，390，371，269，481]。给出了局部更新SGD方法的第一个收敛结果

| Method | Comments | Convergence |
|---|---|---|
| ***Baselines*** | | |
| mini-batch SGD | batch size $KM$ | $\mathcal{O}\left(\frac{H}{T} + \frac{\sigma}{\sqrt{TKM}}\right)$ |
| SGD | (on 1 worker, no communication) | $\mathcal{O}\left(\frac{H}{TK} + \frac{\sigma}{\sqrt{TK}}\right)$ |
| ***Baselines with acceleration***[a] | | |
| A-mini-batch SGD [294, 137] | batch size $KM$ | $\mathcal{O}\left(\frac{H}{T^2} + \frac{\sigma}{\sqrt{TKM}}\right)$ |
| A-SGD [294] | (on 1 worker, no communication) | $\mathcal{O}\left(\frac{H}{(TK)^2} + \frac{\sigma}{\sqrt{TK}}\right)$ |
| ***Parallel SGD / Fed-Avg / Local SGD*** | | |
| Yu et al. [500][b], Stich [434][c] | gradient norm bounded by $G$ | $\mathcal{O}\left(\frac{HKM}{T}\frac{G^2}{\sigma^2} + \frac{\sigma}{\sqrt{TKM}}\right)$ |
| Wang and Joshi [467][b], Stich and Karimireddy [435] | | $\mathcal{O}\left(\frac{HM}{T} + \frac{\sigma}{\sqrt{TKM}}\right)$ |
| ***Other algorithms*** | | |
| SCAFFOLD [265] | control variates and two stepsizes | $\mathcal{O}\left(\frac{H}{T} + \frac{\sigma}{\sqrt{TKM}}\right)$ |

[a] There are no accelerated fed-avg/local SGD variants so far

[b] This paper considers the smooth non-convex setting, we adapt here the results for our setting.

[c] This paper considers the smooth strongly convex setting, we adapt here the results for our setting.

Table 5: Convergence rates for a (non-comprehensive) set of distributed optimization algorithms in the IID-data setting. We assume $M$ devices participate in each iterations, and the loss functions are $H$-smooth, convex, and we have access to stochastic gradients with variance at most $\sigma^2$. All rates are upper bounds on (1) after $T$ iterations (potentially with some iterate averaging scheme).

under the bounded gradient norm assumption in Stich [434] for strongly-convex and in Yu et al. [500] for non-convex objective functions. These analyses could attain the desired $\sigma/\sqrt{TKM}$ statistical term with suboptimal optimization term (in Table 5 we summarize these results for the middle ground of convex functions).

By removing the bounded gradient assumption, Wang and Joshi [467] and Stich and Karimireddy [435] could further improve the optimization term to $HM/T$. These result show that if the number of local steps $K$ is smaller than $T/M^3$ then the (optimal) statistical term is dominating the rate. However, for typical cross-device applications we might have $T = 10^6$ and $M = 100$ (Table 2), implying $K = 1$.

Often in the literature the convergence bounds are accompanied by a discussion on how large $K$ may be chosen in order to reach asymptotically the same statistical term as the convergence rate of mini-batch SGD. For strongly convex functions, this bound was improved by Khaled et al. [269] and further in Stich and Karimireddy [435].

For non-convex objectives, Yu et al. [500] showed that local SGD can achieve asymptotically an error bound $1/\sqrt{TKM}$ if the number of local updates $K$ are smaller than $T^{1/3}/M$. This convergence guarantee was further improved by Wang and Joshi [467] who removed the bounded gradient norm assumption and showed that the number of local updates can be as large as $T/M^3$. The analysis in [467] can also be applied to other algorithms with local updates, and thus yields the first convergence guarantee for decentralized SGD with local updates (or periodic decentralized SGD) and elastic averaging SGD [505]. Haddadpour et al. [216] improves the bounds in Wang and Joshi [467] for functions satisfying the Polyak-Lojasiewicz

| 方法注释收敛 | |
|---|---|
| 基线 | ( ) |
| mini-batch SGD批量KM O | ( H T + σ √ T KM) |
| SGD（1个工作人员，无通信）O | H T K + √ |
| 加速度基线 | ( ) |
| A-mini-batch SGD [294，137]批量KM O | ( H T + √ ) |
| A-SGD [294]（1个工作人员，无通信）O | H (T K) + √ |
| 并行SGD / Fed-Avg /本地SGD | ( ) |
| Yu et al. [500]，斯蒂奇[434] G O有界的梯度范数 | ( HKM T G σ + √ ) |
| [467]王和乔希，斯蒂奇和Karimireddy [435] O | HM T + √ |
| 其他算法 | ( ) |
| SCAFFOLD [265]控制变量和两个步长O | H T + σ √ T KM |

a到目前为止还没有加速的fed-avg/local SGD变体b本文考虑光滑的非凸设置，我们在这里将结果适应于我们的设置。

本文考虑了光滑强凸集，我们在这里推广了已有的结果。

局部更新SGD方法的第一个收敛结果来自表5：IIDdata设置中分布式优化算法（非全面）集合的收敛率。我们假设M个设备参与每次迭代，并且损失函数是H-光滑的，凸的，并且我们可以访问方差最多为σ的随机梯度。在T次迭代之后，所有速率都是（1）的上界（可能具有某种平均方案）。

在斯蒂奇[434]中强凸和Yu等人[500]中非凸目标函数的有界梯度范数假设下。这些分析可以达到预期的σ/ √ T KM统计项与次优优化项（在表5中，我们总结了凸函数的中间基础的这些结果）。

通过去除有界梯度假设，Wang和Joshi [467]以及斯蒂奇和Karimireddy [435]可以进一步将优化项改进为HM/T。这些结果表明，如果局部步骤的数量K小于T /M，则（最佳）统计项主导速率。然而，对于典型的跨器件应用，我们可能有T = 10和M = 100（表2），这意味着K = 1。

通常在文献中的收敛界是伴随着一个讨论如何大K可以选择，以达到渐近相同的统计项的收敛速度的小批量SGD。对于强凸函数，Khaled等人[269]以及斯蒂奇和Karimireddy [435]进一步改进了该界。

对于非凸目标，Yu等人。[500]表明，局部SGD可以渐近地达到误差界1/

如果本地更新的数量K小于T /M，则为T/KM。Wang和Joshi [467]进一步改进了这种收敛保证，他们删除了有界梯度范数假设，并表明局部更新的数量可以与T /M一样大。[467]中的分析也可以应用于具有局部更新的其他算法，从而为具有局部更新的分散式SGD（或周期性分散式SGD）和弹性平均SGD [505]提供了第一个收敛保证。Haddadpour等人。[216]改进了Wang和Joshi [467]中满足Polyak-Lojasiewicz

(PL) condition [262], a generalization of strong convexity. In particular, Haddadpour et al. [216] show that for PL functions, $T^2/M$ local updates per round leads to a $\mathcal{O}(1/TKM)$ convergence.

While the above works focus on convergence as a function of the number of iterations performed, practitioners often care about wall-clock convergence speed. Assessing this must take into account the effect of the design parameters on the time spent per iteration based on the relative cost of communication and local computation. Viewed in this light, the focus on seeing how large $K$ can be while maintaining the statistical rate may not be the primary concern in federated learning, where one may assume almost infinite datasets (very large $N$). The costs (at least in wall-clock time) are small for increasing $M$, and so it may be more natural to increase $M$ sufficiently to match the optimization term, and then tune $K$ to maximize wall-clock optimization performance. How then to choose $K$? Performing more local updates at the clients will increase the divergence between the resulting local models at the clients, before they are averaged. As a result, the error convergence in terms of training loss versus the total number of sequential SGD steps $TK$ is slower. However, performing more local updates saves significant communication cost and reduces the time spent per iteration. The optimal number of local updates strikes a balance between these two phenomena and achieves the fastest error versus wallclock time convergence. Wang and Joshi [468] propose an adaptive communication strategy that adapts $K$ according to the training loss at regular intervals during the training.

Another important design parameter in federated learning is the model aggregation method used to update the global model using the updates made by the selected clients. In the original federated learning paper, McMahan et al. [337] proposes taking a weighted average of the local models, in proportion to the size of local datasets. For IID data, where each client is assumed to have a infinitely large dataset, this reduces to taking a simple average of the local models. However, it is unclear whether this aggregation method will result in the fastest error convergence.

There are many open questions in federated optimization, even with IID data. Woodworth et al. [480] highlights several gaps between upper and lower bounds for optimization relevant to the federated learning setting, particularly for "intermittent communication graphs", which captures local SGD approaches, but convergence rates for such approaches are not known to match the corresponding lower bounds. In Table 5 we highlight convergence results for the convex setting. Whilst most schemes are able to reach the asymptotically dominant statistical term, none are able to match the convergence rate of accelerated mini-batch SGD. It is an open problem if federated averaging algorithms can close this gap.

Local-update SGD methods where all $M$ clients perform the same number of local updates may suffer from a common scalability issue—they can be bottlenecked if any one client unpredictably slows down or fails. Several approaches for dealing with this are possible, but it is far from clear which are optimal, especially when the potential for bias is considered (see Section 6). Bonawitz et al. [81] propose over-provisioning clients (e.g., request updates from $1.3M$ clients), and then accepting the first $M$ updates received and rejecting updates from stragglers. A slightly more sophisticated solution is to fix a time window and allow clients to perform as many local updates $K_i$ as possible within this time, after which their models are averaged by a central server. Wang et al. [471] analyzed the computational heteogeneity introduced by this approach in theory. An alternative method to overcome the problem of straggling clients is to fix the number of local updates at $\tau$, but allow clients to update the global model in an asynchronous or lock-free fashion. Although some previous works [505, 306, 163] have proposed similar methods, the error convergence analysis is an open and challenging problem. A larger challenge in the FL setting, however, is that as discussed at the beginning of Section 3.2, asynchronous approaches may be difficult to combine with complimentary techniques like differential privacy or secure aggregation.

Besides the number of local updates, the choice of the size of the set of clients selected per training round presents a similar trade-off as the number of local updates. Updating and averaging a larger number of client models per training round yields better convergence, but it makes the training vulnerable to slowdown due

[216]改进了Wang和Joshi [467]中满足Polyak-Lojasiewicz（PL）条件[262]的函数的界，这是强凸性的推广。特别是，Haddadpour等人。[216]表明，对于PL函数，每轮T /M局部更新导致O（1/T KM）收敛。

虽然上述工作集中于收敛作为执行的迭代次数的函数，但实践者通常关心挂钟收敛速度。评估这一点必须考虑设计参数对基于通信和本地计算的相对成本的每次迭代所花费的时间的影响。从这个角度来看，在保持统计率的同时关注K可以有多大可能不是联邦学习中的主要问题，因为人们可以假设几乎无限的数据集（非常大的N）。增加M的成本（至少在挂钟时间内）很小，因此更自然的做法是充分增加M以匹配优化项，然后调整K以最大化挂钟优化性能。如何选择K？在客户端执行更多的本地更新将增加客户端处的结果本地模型之间的差异，然后再对它们进行平均。 因此，在训练损失与顺序SGD步骤的总数T K方面的误差收敛较慢。但是，执行更多的本地更新可以节省大量的通信成本，并减少每次迭代所花费的时间。局部更新的最佳数量在这两种现象之间取得平衡，并实现最快的误差与挂钟时间收敛。Wang和Joshi [468]提出了一种自适应通信策略，该策略根据训练期间定期的训练损失来调整K。

联邦学习中的另一个重要设计参数是模型聚合方法，用于使用选定客户端进行的更新来更新全局模型。在最初的联邦学习论文中，McMahan等人。[337]建议采用局部模型的加权平均值，与局部数据集的大小成比例。对于IID数据，假设每个客户端都有一个无限大的数据集，这可以简化为对本地模型进行简单的平均。然而，目前还不清楚这种聚合方法是否会导致最快的误差收敛。

在联邦优化中有许多悬而未决的问题，即使使用IID数据也是如此。Woodworth等人。[480]强调了与联邦学习设置相关的优化上限和下限之间的几个差距，特别是对于“间歇性通信图”，它捕获了本地SGD方法，但这些方法的收敛速度并不匹配相应的下限。在表5中，我们突出显示了凸设置的收敛结果。虽然大多数计划能够达到渐近占主导地位的统计项，没有能够匹配的收敛速度的加速小批量SGD。这是一个开放的问题，如果联邦平均算法可以弥补这一差距。

所有M个客户端执行相同数量的本地更新的本地更新SGD方法可能会遇到一个常见的可扩展性问题-如果任何一个客户端不可预测地减慢或失败，则可能会被检查。有几种方法可以解决这个问题，但目前还不清楚哪种方法是最佳的，特别是当考虑到偏倚的可能性时（见第6节）。Bonawitz等人[81]提出了过度配置客户端（例如，请求来自1.3M个客户端的更新），然后接受接收到的前M个更新并拒绝来自落后者的更新。一个稍微复杂一点的解决方案是固定一个时间窗口，并允许客户端在这段时间内执行尽可能多的本地更新Kas，之后他们的模型由中央服务器平均。Wang等人[471]从理论上分析了这种方法引入的计算异质性。 另一种解决客户端分散问题的方法是将本地更新的数量固定在τ，但允许客户端以异步或无锁的方式更新全局模型。虽然一些以前的作品[505，306，163]提出了类似的方法，误差收敛分析是一个开放的和具有挑战性的问题。然而，FL设置中的一个更大的挑战是，如第3.2节开始时所讨论的，异步方法可能难以联合收割机与诸如差分隐私或安全聚合之类的互补技术相结合。

除了本地更新的数量之外，每个训练轮选择的客户端集合的大小的选择呈现与本地更新的数量类似的权衡。每轮训练更新和平均大量的客户端模型可以产生更好的收敛性，但这会使训练容易受到速度减慢的影响，

to unpredictable tail delays in computation/communication at/with the clients.

The analysis of local SGD / Federated Averaging in the non-IID setting is even more challenging; results and open questions related to this are considered in the next section, along with specialized algorithms which directly address the non-IID problem.

### 3.2.2 Optimization Algorithms and Convergence Rates for Non-IID Datasets

In contrast to well-shuffled mini-batches consisting of independent and identically distributed (IID) examples in centralized learning, federated learning uses local data from end user devices, leading to many varieties of non-IID data (Section 3.1).

In this setting, each of $N$ clients has a local data distribution $\mathcal{P}_i$ and a local objective function

$$f_i(x) = \mathop{\mathbb{E}}_{z \sim \mathcal{P}_i} [f(x; z)]$$

where we recall that $f(x; z)$ is the loss of a model $x$ at an example $z$. We typically wish to minimize

$$F(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x) \,. \tag{2}$$

Note that we recover the IID setting when each $\mathcal{P}_i$ is identical. We will let $F^*$ denote the minimum value of $F$, obtained the point $x^*$. Analogously, we will let $f_i^*$ denote the minimum value of $f_i$.

As in the IID setting, we assume an intermittent communication model (e.g. Woodworth et al. [480, Sec. 4.4]), where $M$ stateless clients participate in each of $T$ rounds, and during each round, each client can compute gradients for $K$ samples (e.g. minibatches). The difference here is that the samples $z_{i,1}, \ldots, z_{i,K}$ sampled at client $i$ are drawn from the client's local distribution $\mathcal{P}_i$. Unlike the IID setting, we cannot necessarily assume $M = N$, as the client distributions are not all equal. In the following, if an algorithm relies on $M = N$, we will omit $M$ and simply write $N$. We note that while such an assumption may be compatible with the cross-silo federated setting in Table 1, it is generally infeasible in the cross-device setting.

While [434, 500, 467, 435] mainly focused on the IID case, the analysis technique can be extended to the non-IID case by adding an assumption on data dissimilarities, for example by constraining the difference between client gradients and the global gradient [305, 300, 304, 469, 471] or the difference between client and global optimum values [303, 268]. Under this assumption, Yu et al. [501] showed that the error bound of local SGD in the non-IID case becomes worse. In order to achieve the rate of $1/\sqrt{TKN}$ (under non-convex objectives), the number of local updates $K$ should be smaller than $T^{1/3}/N$, instead of $T/N^3$ as in the IID case [467]. Li et al. [300] proposed to add a proximal term in each local objective function so as to make the algorithm be more robust to the heterogeneity across local objectives. The proposed FedProx algorithm empirically improves the performance of federated averaging. Khaled et al. [268] assumes all clients participate, and uses batch gradient descent on clients, which can potentially converge faster than stochastic gradients on clients.

Recently, a number of works have made progress in relaxing the assumptions necessary for analysis so as to better apply to practical uses of Federated Averaging. For example, Li et al. [303] studied the convergence of Federated Averaging in a more realistic setting where only a subset of clients are involved in each round. In order to guarantee the convergence, they assumed that the clients are selected either uniformly at random or with probabilities that are in proportion to the sizes of local datasets. Nonetheless, in practice the server may not be able to sample clients in these idealized ways — in particular, in cross-device settings only

但是由于在客户端处/与客户端的计算/通信中的不可预测的尾部延迟，这使得训练易受减速的影响。

在非IID设置中对局部SGD /联合平均的分析更具挑战性;与此相关的结果和未决问题将在下一节中考虑，沿着专门的算法直接解决非IID问题。

3.2.2非IID数据集的优化算法和收敛速度

与集中式学习中由独立同分布（IID）示例组成的精心洗牌的小批量相比，联邦学习使用来自最终用户设备的本地数据，导致许多种类的非IID数据（第3.1节）。

在该设置中，N个客户端中的每一个具有本地数据分布P和本地目标函数

$$f(x) = \mathop{e}_{z\sim P} [x, x]$$

其中我们记得f（x; z）是模型x在示例z处的损失。我们通常希望尽量减少

$$F(x) = \frac{1}{N} \sum_{i=1}^{N} f(x) . (二)$$

请注意，当每个PI相同时，我们恢复IID设置。我们将F表示F的最小值，得到点x。类似地，我们让f表示f的最小值。

与IID设置中一样，我们假设间歇性通信模型（例如，Woodworth等人[480，Sec. 4.4]），其中M个无状态客户端参与T轮中的每一轮，并且在每一轮期间，每个客户端可以计算K个样本（例如，小批量）的梯度。这里的区别在于，样本z，. . .在客户端i处采样的M，z是从客户端的本地分布P中提取的。与IID设置不同，我们不必假设M = N，因为客户端分布并不都相等。在下文中，如果算法依赖于M = N，我们将省略M并简单地写N。我们注意到，虽然这样的假设可能与表1中的跨竖井联合设置兼容，但在跨设备设置中通常是不可行的。

虽然[434，500，467，435]主要集中在IID情况下，但分析技术可以通过添加对数据不相似性的假设来扩展到非IID情况，例如通过约束客户端梯度和全局梯度之间的差异[305，300，304，469，471]或客户端和全局最佳值之间的差异[303，469，471]。268]。在此假设下，Yu等人。[501]表明，在非IID情况下局部SGD的误差界变得更差。为了达到1/

$\sqrt{}$

T KN（在非凸目标下），局部更新的数量K应该小于T /N，而不是IID情况下的T /N。Li等人。[300]提出在每个局部目标函数中添加一个邻近项，以使算法对局部目标之间的异质性更具鲁棒性。所提出的FedProx算法根据经验提高了联邦平均的性能。Khaled等人。[268]假设所有客户端都参与，并在客户端上使用批量梯度下降，这可能比客户端上的随机梯度收敛得更快。

最近，一些作品在放宽分析所需的假设方面取得了进展，以便更好地应用于联邦平均的实际用途。例如，Li等人。[303]在一个更现实的设置中研究了联邦平均的收敛性，其中每一轮只涉及一部分客户端。为了保证收敛，他们假设客户端是随机选择的，或者是以与本地数据集大小成比例的概率选择的。尽管如此，在实践中，服务器可能无法以这些理想化的方式对客户端进行采样-特别是，仅在跨设备设置中

**Non-IID assumptions**

| Symbol | Full name | Explanation |
|---|---|---|
| BCGV | bounded inter-client gradient variance | $\mathbb{E}_i \left\|\nabla f_i(x) - \nabla F(x)\right\|^2 \leq \eta^2$ |
| BOBD | bounded optimal objective difference | $F^* - \mathbb{E}_i[f_i^*] \leq \eta^2$ |
| BOGV | bounded optimal gradient variance | $\mathbb{E}_i \left\|\nabla f_i(x^*)\right\|^2 \leq \eta^2$ |
| BGV | bounded gradient dissimilarity | $\mathbb{E}_i \left\|\nabla f_i(x)\right\|^2 / \left\|\nabla F(x)\right\|^2 \leq \eta^2$ |

**Other assumptions and variants**

| Symbol | Explanation |
|---|---|
| CVX | Each client function $f_i(x)$ is convex. |
| SCVX | Each client function $f_i(x)$ is $\mu$-strongly convex. |
| BNCVX | Each client function has bounded nonconvexity with $\nabla^2 f_i(x) \succeq -\mu I$. |
| BLGV | The variance of stochastic gradients on local clients is bounded. |
| BLGN | The norm of any local gradient is bounded. |
| LBG | Clients use the full batch of local samples to compute updates. |
| Dec | Decentralized setting, assumes the the connectivity of network is good. |
| AC | All clients participate in each round. |
| 1step | One local update is performed on clients in each round. |
| Prox | Use proximal gradient steps on clients. |
| VR | Variance reduction which needs to track the state. |

**Convergence rates**

| Method | Non-IID | Other assumptions | Variant | Rate |
|---|---|---|---|---|
| Lian et al. [305] | BCGV | BLGV | Dec; AC; 1step | $O(1/T) + O(1/\sqrt{NT})$ |
| PD-SGD [304] | BCGV | BLGV | Dec; AC | $O(N/T) + O(1/\sqrt{NT})$ |
| MATCHA [469] | BCGV | BLGV | Dec | $O(1/\sqrt{TKM}) + O(M/KT)$ |
| Khaled et al. [268] | BOGV | CVX | AC; LBG | $O(N/T) + O(1/\sqrt{NT})$ |
| Li et al. [303] | BOBD | SCVX; BLGV; BLGN | - | $O(K/T)$ |
| FedProx [300] | BGV | BNCVX | Prox | $O(1/\sqrt{T})$ |
| SCAFFOLD [265] | - | SCVX; BLGV | VR | $O(1/TKM) + O(e^{-T})$ |

Table 6: Convergence rates for a (non-comprehensive) set of federated optimization methods in non-IID settings. We summarize the key assumptions for non-IID data, local functions on each client, and other assumptions. We also present the variant of the algorithm comparing to Federated Averaging and the convergence rates that eliminate constant.

| 非IID假设 | |
|---|---|
| BCGV有界客户间梯度方差E f（x） − F（x） ≤ η BOBD有界最优目标差F − E[f ] ≤ η BOGV有界最优梯度方差E f（x） ≤ η BGV有界梯度相异性/≤ η | |

| 其他假设和变式 | |
|---|---|
| 符号说明 | |
| CVX每个客户函数f（x）都是凸的。<br>每个客户函数f（x）都是μ-强凸的。<br>BNCVX每个客户函数都具有有界非凸性，且有界非凸性满足λ f（x） □ − μ I。<br>BLGV局部客户端上随机梯度的方差是有界的。<br>BLGN任何局部梯度的范数是有界的。<br>LBG客户端使用整批本地样本来计算更新。<br>Dec分散式设置，假设网络连接良好。<br>AC所有客户都参与每一轮。<br>1step在每一轮中对客户端执行一次本地更新。<br>近端在客户端使用近端梯度步长。<br>VR方差减少，需要跟踪状态。 | |

| 收敛率方法非IID其他假设变量率 | | | | |
|---|---|---|---|---|
| Lian et al.[305] BCGV BLGV Dec; AC; 1step O（/） + O（/） PD-SGD [304] BCGV BLGV Dec; AC O（/） + O（/） MATCHA [469] BCGV BLGV Dec O（/） + O（/） Khaled et al.[268] BOGV CVX AC; LBG O（/） + O（/） Li et al.[303] BOBD SCVX; BLGV; BLGN - O（/） FedProx [300] BGV BNCVX Prox O（/） SCAFFOLD [265] - SCVX; BLGV VR O（/） + O（e） | | | | |

表6：非IID设置中一组（不全面的）联合优化方法的收敛率。我们总结了非IID数据的关键假设，每个客户端上的本地功能以及其他假设。我们还提出了该算法的变体相比，联邦平均和收敛速度，消除常数。

devices that meet strict eligibility requirements (e.g. charging, idle, free WiFi) will be selected to participate in the computation. At different times within a day, the clients characteristics can vary significantly. Eichner et al. [171] formulated this problem and studied the convergence of semi-cyclic SGD, where multiple blocks of clients with different characteristics are sampled from following a regular cyclic pattern (e.g. diurnal). Clients can perform different local steps because of heterogeneity in their computing capacities. Wang et al. [471] proves that FedAvg and many other federated learning algorithms will converge to the stationary points of a mismatched objective function in the presence of heterogeneous local steps. They refer to this problem as *objective inconsistency* and propose a simple technique to eliminate the inconsistency problem from federated learning algorithms.

We summarize recent theoretical results in Table 6. All the methods in Table 6 assume smoothness or Lipschitz gradients for the local functions on clients. The error bound is measured by optimal objective (1) for convex functions and norm of gradient for nonconvex functions. For each method, we present the key non-IID assumption, assumptions on each client function $f_i(x)$, and other auxiliary assumptions. We also briefly describe each method as a variant of the federated averaging algorithm, and show the simplified convergence rate eliminating constants. Assuming the client functions are strongly convex could help the convergence rate [303, 265]. Bounded gradient variance, which is a widely used assumption to analyze stochastic gradient methods, is often used when clients use stochastic local updates [305, 303, 304, 469, 265]. Li et al. [303] directly analyzes the Federated Averaging algorithm, which applies $K$ steps of local updates on randomly sampled $M$ clients in each round, and presents a rate that suggests local updates $(K > 1)$ could slow down the convergence. Clarifying the regimes where $K > 1$ may hurt or help convergence is an important open problem.

**Connections to decentralized optimization** The objective function of federated optimization has been studied for many years in the decentralized optimization community. As first shown in Wang and Joshi [467], the convergence analysis of decentralized SGD can be applied to or combined with local SGD with a proper setting of the network topology matrix (mixing matrix). In order to reduce the communication overhead, Wang and Joshi [467] proposed periodic decentralized SGD (PD-SGD) which allows decentralized SGD to have multiple local updates as Federated Averaging. This algorithm is extended by Li et al. [304] to the non-IID case. MATCHA [469] further improves the performance of PD-SGD by randomly sampling clients for computation and communication, and provides a convergence analysis showing that local updates can accelerate convergence.

**Acceleration, variance reduction and adaptivity** Momentum, variance-reduction, and adaptive learning rates are all promising techniques to improve convergence and generalization of first-order methods. However, there is no single manner in which to incorporate these techniques into FedAvg. SCAFFOLD [265] models the difference in client updates using control variates to perform variance reduction. Notably, this allows convergence results not relying on bounding the amount of heterogeneity among clients. As for momentum, Yu et al. [501] propose allowing each client to maintain a local momentum buffer and average the local buffers and the local model parameters at each communication round. Although this method empirically improves the final accuracy of local SGD, this doubles the per-round communication cost. A similar scheme is used by Xie et al. [485] to design a variant of local SGD in which clients locally perform Adagrad [335, 161]. Reddi et al. [389] instead proposes using adaptive learning rates at the server-level, developing federated versions of adaptive optimization methods with the same communication cost as FedAvg. This framework generalizes the server momentum framework proposed by Hsu et al. [237], Wang et al. [470], which allows momentum without increasing communication costs. While both [501, 470] showed that the momentum variants of local SGD can converge to stationary points of non-convex objective

符合严格资格要求（例如充电、空闲、免费WiFi）的设备将被选择参与计算。在一天中的不同时间，客户的特征可能会有很大的不同。Eichner等人。[171]制定了这个问题，并研究了半循环SGD的收敛性，其中具有不同特征的多个客户端块从遵循规则的循环模式（例如，昼夜）中采样。客户端可以执行不同的本地步骤，因为它们的计算能力不同。Wang等人。[471]证明了FedAvg和许多其他联邦学习算法将在存在异构局部步骤的情况下收敛到不匹配目标函数的稳定点。他们将此问题称为目标不一致，并提出了一种简单的技术来消除联邦学习算法中的不一致问题。

我们在表6中总结了最近的理论结果。表6中的所有方法都假设客户端上的局部函数具有平滑性或Lipschitz梯度。对凸函数用最优目标（1）度量误差界，对非凸函数用梯度范数度量误差界。对于每种方法，我们提出了关键的非IID假设，对每个客户函数f（x）的假设，以及其他辅助假设。我们还简要介绍了每种方法作为一个变种的联邦平均算法，并显示简化的收敛速度消除常数。假设客户函数是强凸的可以帮助收敛速度[303，265]。有界梯度方差是分析随机梯度方法的广泛使用的假设，当客户端使用随机局部更新时经常使用[305，303，304，469，265]。Li等人 [303]直接分析了联邦平均算法，该算法在每轮中对随机抽样的M个客户端应用K步局部更新，并提出了一个表明局部更新（K > 1）可能会减慢收敛的速率。澄清K > 1可能损害或有助于收敛的机制是一个重要的开放问题。

联邦优化的目标函数在分散优化领域已经研究了很多年。如Wang和Joshi [467]中首次所示，分散式SGD的收敛分析可以应用于或结合本地SGD，并适当设置网络拓扑矩阵（混合矩阵）。为了减少通信开销，Wang和Joshi [467]提出了周期性分散式SGD（PD-SGD），它允许分散式SGD具有多个本地更新作为联邦平均。该算法由Li等人[304]扩展到非IID情况。MATCHA [469]通过随机抽样客户端进行计算和通信，进一步提高了PD-SGD的性能，并提供了收敛分析，表明本地更新可以加速收敛。

加速、方差缩减和自适应动量、方差缩减和自适应学习率都是提高一阶方法收敛性和推广性的有前途的技术。然而，没有一种方法可以将这些技术合并到FedAvg中。SCAFFOLD [265]使用控制变量对客户端更新的差异进行建模，以执行方差缩减。值得注意的是，这允许收敛结果不依赖于限制客户端之间的异质性的量。至于动量，Yu等人。[501]建议允许每个客户端维护一个本地动量缓冲区，并在每个通信回合对本地缓冲区和本地模型参数进行平均。虽然这种方法根据经验提高了本地SGD的最终准确性，但这使每轮通信成本加倍。Xie等人[485]使用了类似的方案来设计本地SGD的变体，其中客户端在本地执行Adagrad [335，161]。Reddi等人 [389]相反，提出在服务器级使用自适应学习率，开发自适应优化方法的联邦版本，其通信成本与FedAvg相同。该框架概括了Hsu等人提出的服务器动量框架。[237]，Wang等人。[470]，它允许动量而不增加通信成本。 而[501，470]都表明，局部SGD的动量变量可以收敛到非凸目标的驻点

functions at the same rate as synchronous mini-batch SGD, it is challenging to prove momentum accelerates the convergence rate in the federated learning setting. Recently, Karimireddy et al. [264] proposed a general approach for adapting centralized optimization algorithms to the heterogeneous federated setting (MIME framework and algorithms).

## 3.3 Multi-Task Learning, Personalization, and Meta-Learning

In this section we consider a variety of "multi-model" approaches — techniques that result in effectively using different models for different clients at inference time. These techniques are particularly relevant when faced with non-IID data (Section 3.1), since they may outperform even the best possible shared global model. We note that personalization has also been studied in the fully decentralized setting [459, 59, 504, 19], where training individual models is particularly natural.

### 3.3.1 Personalization via Featurization

The remainder of this section specifically considers techniques that result in different users running inference with different model parameters (weights). However, in some applications similar benefits can be achieved by simply adding user and context features to the model. For example, consider a language model for next-word-prediction in a mobile keyboard as in Hard et al. [222]. Different clients are likely to use language differently, and in fact on-device personalization of model parameters has yielded significant improvements for this problem [472]. However, a complimentary approach may be to train a federated model that takes as input not only the words the user has typed so far, but a variety of other user and context features—What words does this user frequently use? What app are they currently using? If they are chatting, what messages have they sent to this person before? Suitably featurized, such inputs can allow a shared global model to produce highly personalized predictions. However, largely because few public datasets contain such auxiliary features, developing model architectures that can effectively incorporate context information for different tasks remains an important open problem with the potential to greatly increase the utility of FL-trained models.

### 3.3.2 Multi-Task Learning

If one considers each client's local problem (the learning problem on the local dataset) as a separate task (rather than as a shard of a single partitioned dataset), then techniques from multi-task learning [506] immediately become relevant. Notably, Smith et al. [424] introduced the MOCHA algorithm for multi-task federated learning, directly tackling challenges of communication efficiency, stragglers, and fault tolerance. In multi-task learning, the result of the training process is one model per task. Thus, most multi-task learning algorithms assume all clients (tasks) participate in each training round, and also require stateful clients since each client is training an individual model. This makes such techniques relevant for cross-silo FL applications, but harder to apply in cross-device scenarios.

Another approach is to reconsider the relationship between clients (local datasets) and learning tasks (models to be trained), observing that there are points on a spectrum between a single global model and different models for every client. For example, it may be possible to apply techniques from multi-task learning (as well as other approaches like personalization, discussed next), where we take the "task" to be a subset of the clients, perhaps chosen explicitly (e.g. based on geographic region, or characteristics of the device or user), or perhaps based on clustering [331] or the connected components of a learned graph over the clients [504]. The development of such algorithms is an important open problem. See Section 4.4.4

470]证明了局部SGD的动量变量可以以与同步minibatch SGD相同的速度收敛到非凸目标函数的稳定点，证明动量加速了联邦学习设置中的收敛速度是具有挑战性的。最近，Karimireddy等人[264]提出了一种通用方法，用于使集中式优化算法适应异构联邦设置（MIME框架和算法）。

## 3.3多任务学习、个性化和元学习

在本节中，我们考虑各种“多模型”方法--在推理时为不同客户端有效使用不同模型的技术。当面对非IID数据时，这些技术特别相关（第3.1节），因为它们甚至可能优于最好的共享全局模型。我们注意到，个性化也在完全分散的环境中进行了研究[459，59，504，19]，在那里训练个体模型特别自然。

### 3.3.1通过特征化进行个性化

本节的其余部分将专门考虑导致不同用户使用不同模型参数（权重）运行推理的技术。然而，在某些应用中，通过简单地向模型添加用户和上下文特征，可以实现类似的好处。例如，考虑Hard等人[222]中的用于移动的键盘中的下一个词预测的语言模型。不同的客户端可能会使用不同的语言，实际上，模型参数的设备个性化已经为这个问题带来了显着的改善[472]。然而，一种补充的方法可能是训练一个联邦模型，该模型不仅将用户迄今为止键入的单词作为输入，而且还将各种其他用户和上下文特征作为输入-这个用户经常使用什么单词？他们目前使用的是什么App？如果他们在聊天，他们以前给这个人发过什么信息？ 适当地特征化，这样的输入可以允许共享的全局模型产生高度个性化的预测。然而，很大程度上是因为很少有公共数据集包含这些辅助功能，开发可以有效地将上下文信息用于不同任务的模型架构仍然是一个重要的开放问题，有可能大大提高FL训练模型的实用性。

### 3.3.2多任务学习

如果将每个客户端的本地问题（本地数据集上的学习问题）视为单独的任务（而不是单个分区数据集的分片），那么来自多任务学习的技术[506]立即变得相关。值得注意的是，Smith等人[424]引入了用于多任务联邦学习的MOCHA算法，直接解决了通信效率，落后者和容错的挑战。在多任务学习中，训练过程的结果是每个任务一个模型。因此，大多数多任务学习算法假设所有客户端（任务）都参与每个训练轮，并且还需要有状态的客户端，因为每个客户端都在训练单独的模型。这使得这些技术与跨竖井FL应用相关，但更难应用于跨设备场景。

另一种方法是重新考虑客户端（本地数据集）和学习任务（待训练的模型）之间的关系，观察单个全局模型和每个客户端的不同模型之间的频谱上的点。例如，可以应用来自多任务学习的技术（以及其他方法，如个性化，下面将讨论），其中我们将“任务”作为客户端的子集，可能是显式选择的（例如，基于地理区域，或者设备或用户的特性），或者可能基于聚类[331]或客户端上的学习图的连接分量[504]。这种算法的发展是一个重要的开放问题。见第4.4.4节

for a discussion of how sparse federated learning problems, such as those arising naturally in this type of multi-task problem, might be approached without revealing to which client subset (task) each client belongs.

### 3.3.3 Local Fine Tuning and Meta-Learning

By local fine tuning, we refer to techniques which begin with the federated training of a single model, and then deploy that model to all clients, where it is personalized by additional training on the local dataset before use in inference. This approach integrates naturally into the typical lifecycle of a model in federated learning (Section 1.1.1). Training of the global model can still proceed using only small samples of clients on each round (e.g. 100s); the broadcast of the global model to all clients (e.g. many millions) only happens once, when the model is deployed. The only difference is that before the model is used to make live predictions on the client, a final training process occurs, personalizing the model to the local dataset.

Given a global model that performs reasonably well, what is the best way to personalize it? In non-federated learning, researchers often use fine-tuning, transfer learning, domain adaptation [329, 132, 61, 332, 133], or interpolation with a personal local model. Of course, the precise technique used for such interpolations is key and it is important to determine its corresponding learning guarantees in the context of federated learning. Further, these techniques often assume only a pair of domains (source and target), and so some of the richer structure of federated learning may be lost.

One approach for studying personalization and non-IID data is via a connection to *meta-learning*, which has emerged as a popular setting for model adaptation. In the standard learning-to-learn (LTL) setup [56], one has a meta-distribution over tasks, samples from which are used to learn a learning algorithm, for example by finding a good restriction of the hypothesis space. This is in fact a good match for the statistical setting discussed in Section 3.1, where we sample a client (task) $i \sim \mathcal{Q}$, and then sample data for that client (task) from $\mathcal{P}_i$.

Recently, a class of algorithms referred to as *model-agnostic meta-learning* (MAML) have been developed that meta-learn a global model, which can be used as a starting point for learning a good model adapted to a given task, using only a few local gradient steps [187]. Most notably, the training phase of the popular Reptile algorithm [358] is closely related to Federated Averaging [337] — Reptile allows for a server learning rate and assumes all clients have the same amount of data, but is otherwise the same. Khodak et al. [270] and Jiang et al. [250] explore the connection between FL and MAML, and show how the MAML setting is a relevant framework to model the personalization objectives for FL. Chai Sim et al. [102] applied local fine tuning to personalize speech recognition models in federated learning. Fallah et al. [181] developed a new algorithm called Personalized FedAvg by connecting MAML instead of Reptile to federated learning. Additional connections with differential privacy were studied in [299].

The general direction of combining ideas from FL and MAML is relatively new, with many open questions:

- The evaluation of MAML algorithms for supervised tasks is largely focused on synthetic image classification problems [290, 386] in which infinite artificial tasks can be constructed by subsampling from classes of images. FL problems, modeled by existing datasets used for simulated FL experiments (Appendix A), can serve as realistic benchmark problems for MAML algorithms.
- In addition to an empirical study, or optimization results, it would be useful to analyze the theoretical guarantees of MAML-type techniques and study under what assumptions they can be successful, as this will further elucidate the set of FL domains to which they may apply.
- The observed gap between the global and personalized acccuracy [250] creates a good argument

4讨论稀疏联合学习问题，例如在这种类型的多任务问题中自然产生的问题，可能会在不透露每个客户端属于哪个客户端子集（任务）的情况下进行处理。

### 3.3.3局部微调和元学习

通过局部微调，我们指的是开始于单个模型的联合训练，然后将该模型部署到所有客户端的技术，在那里，在用于推理之前，通过对本地数据集进行额外的训练来个性化该模型。这种方法自然地集成到联邦学习中模型的典型生命周期中（第1.1.1节）。全局模型的训练仍然可以在每一轮仅使用少量的客户端样本（例如100个）进行;全局模型向所有客户端（例如数百万个）的广播仅在模型部署时发生一次。唯一的区别是，在使用模型在客户端进行实时预测之前，会进行最终的训练过程，将模型个性化到本地数据集。

给定一个表现相当好的全局模型，个性化它的最佳方法是什么？在非联邦学习中，研究人员经常使用微调，迁移学习，域自适应[329，132，61，332，133]或个人局部模型的插值。当然，用于这种插值的精确技术是关键，在联邦学习的上下文中确定其相应的学习保证也很重要。此外，这些技术通常只假设一对域（源和目标），因此可能会丢失一些更丰富的联邦学习结构。

研究个性化和非IID数据的一种方法是通过与元学习的连接，元学习已成为模型自适应的流行设置。在标准的学习-学习（LTL）设置中[56]，任务上有一个元分布，从中的样本用于学习学习算法，例如通过找到假设空间的良好限制。事实上，这与3.1节中讨论的统计设置很好地匹配，在3.1节中，我们对客户端（任务）i进行采样，然后从P中对该客户端（任务）的数据进行采样。

最近，已经开发了一类被称为模型不可知元学习（MAML）的算法，该算法可以元学习全局模型，该全局模型可以用作学习适应给定任务的良好模型的起点，仅使用几个局部梯度步骤[187]。最值得注意的是，流行的爬虫算法[358]的训练阶段与联合平均[337]密切相关-爬虫允许服务器学习率，并假设所有客户端具有相同的数据量，但其他方面都是相同的。Khodak et al. [270]和Jiang et al. [250]探索了FL和MAML之间的联系，并展示了MAML设置如何成为FL个性化目标建模的相关框架。Chai Sim et al. [102]应用局部微调来个性化联邦学习中的语音识别模型。Fallah等人。[181]通过将MAML而不是Reptile连接到联邦学习，开发了一种名为Personalized FedAvg的新算法。

在[299]中研究了与差异隐私的其他联系。

将FL和MAML的思想结合起来的总体方向是相对较新的，有许多悬而未决的问题：

用于监督任务的MAML算法的评估主要集中在合成图像分类问题上[290，386]，其中可以通过从图像类别中进行子采样来构建无限人工任务。FL问题，模拟FL实验（附录A）使用现有的数据集建模，可以作为现实的基准问题MAML算法。

·除了实证研究或优化结果之外，分析MAML类型技术的理论保证并研究它们在什么假设下可以成功，这将是有用的，因为这将进一步阐明它们可能适用的FL域集。

·观察到的全球和个性化准确性之间的差距[250]创造了一个很好的论点

that personalization should be of central importance to FL. However, none of the existing works clearly formulates what would be comprehensive metrics for measuring personalized performance; for instance, is a small improvement for every client preferable to a larger improvement for a subset of clients? See Section 6 for a related discussion.

- Jiang et al. [250] highlighted the fact that models of the same structure and performance, but trained differently, can have very different capacity to personalize. In particular, it appears that training models with the goal of maximizing global performance might actually hurt the model's capacity for subsequent personalization. Understanding the underlying reasons for this is a question relevant for both FL and the broader ML community.
- Several challenging FL topics including personalization and privacy have begun to be studied in this multi-task/LTL framework [270, 250, 299]. Is it possible for other issues such as concept drift to also be analyzed in this way, for example as a problem in lifelong learning [420]?
- Can non-parameter transfer LTL algorithms, such as ProtoNets [425], be of use for FL?

#### 3.3.4 When is a Global FL-trained Model Better?

What can federated learning do for you that local training on one device cannot? When local datasets are small and the data is IID, FL clearly has an edge, and indeed, real-world applications of federated learning [491, 222, 112] benefit from training a single model across devices. On the other hand, given pathologically non-IID distributions (e.g. $\mathcal{P}_i(y \mid x)$ directly disagree across clients), local models will do much better. Thus, a natural theoretical question is to determine under what conditions the shared global model is better than independent per-device models. Suppose we train a model $h_k$ for each client $k$, using the sample of size $m_k$ available from that client. Can we guarantee that the model $h_{\text{FL}}$ learned via federated learning is at least as accurate as $h_k$ when used for client $k$? Can we quantify how much improvement can be expected via federated leaning? And can we develop personalization strategies with theoretical guarantees that at least match the performance of both natural baselines ($h_k$ and $h_{\text{FL}}$)?

Several of these problems relate to previous work on multiple-source adaptation and agnostic federated learning [329, 330, 234, 352]. The hardness of these questions depends on how the data is distributed among parties. For example, if data is vertically partitioned, each party maintaining private records of different feature sets about common entities, these problems may require addressing record linkage [124] within the federated learning task. Independently of the eventual technical levy of carrying out record linkage privately [407], the task itself happens to be substantially noise prone in the real world [406] and only sparse results have addressed its impact on training models [224]. Techniques for robustness and privacy can make local models relatively stronger, particularly for non-typical clients [502]. Loss factorization tricks can be used in supervised learning to alleviate up to the vertical partition assumption itself, but the practical benefits depend on the distribution of data and the number of parties [373].

### 3.4 Adapting ML Workflows for Federated Learning

Many challenges arise when adapting standard machine learning workflows and pipelines (including data augmentation, feature engineering, neural architecture design, model selection, hyperparameter optimization, and debugging) to decentralized datasets and resource-constrained mobile devices. We discuss several of these challenges below.

个性化应该是FL的核心重要性。然而，现有的作品都没有明确制定什么是衡量个性化性能的综合指标;例如，是一个小的改进，为每个客户端的一个子集的客户端更大的改进？相关讨论见第6节。

· Jiang等人[250]强调了这样一个事实，即结构和性能相同但训练方式不同的模型可以具有非常不同的个性化能力。特别是，似乎以最大化全局性能为目标的训练模型实际上可能会损害模型后续个性化的能力。理解其根本原因是一个与FL和更广泛的ML社区相关的问题。

在这个多任务/LTL框架中，已经开始研究几个具有挑战性的FL主题，包括个性化和隐私[270，250，299]。是否有可能以这种方式分析其他问题，例如概念漂移，例如作为终身学习的问题[420]?

·非参数传递LTL算法，如ProtoNets [425]，是否可用于FL?

### 3.3.4什么时候全局FL训练的模型更好?

联合学习能为您提供哪些在一台设备上进行本地培训无法提供的功能？当本地数据集很小并且数据是IID时，FL显然具有优势，事实上，联邦学习的现实应用[491，222，112]受益于跨设备训练单个模型。另一方面，给定病理上的非IID分布（例如，P（y| x）直接不同意跨客户），本地模型会做得更好。因此，一个自然的理论问题是确定在什么条件下共享全局模型优于独立的每个设备模型。假设我们为每个客户端k训练一个模型h，使用该客户端提供的大小为m的样本。我们能保证通过联邦学习学习的模型hlearned至少和用于客户端k的模型hlearned一样准确吗？我们能量化通过联邦学习可以预期多少改进吗？ 我们是否可以在理论上保证制定出至少与两个自然基线（h手）相匹配的个性化策略?

其中一些问题与以前的多源自适应和不可知联邦学习有关[329，330，234，352]。这些问题的难度取决于数据如何在各方之间分配。例如，如果数据是垂直分区的，每一方都维护关于公共实体的不同特征集的私有记录，这些问题可能需要在联合学习任务中解决记录链接[124]。独立于私下执行记录链接的最终技术征税[407]，任务本身在真实的世界中恰好是非常容易产生噪声的[406]，并且只有稀疏的结果解决了其对训练模型的影响[224]。鲁棒性和隐私技术可以使本地模型相对更强大，特别是对于非典型客户端[502]。 损失因子分解技巧可以用于监督学习，以减轻垂直分区假设本身，但实际利益取决于数据的分布和参与方的数量[373]。

## 3.4为联邦学习调整ML工作流

在将标准机器学习工作流和管道（包括数据增强、特征工程、神经架构设计、模型选择、超参数优化和调试）适应分散的数据集和资源受限的移动的设备时，会出现许多挑战。我们在下面讨论其中的几个挑战。

### 3.4.1 Hyperparameter Tuning

Running many rounds of training with different hyperparameters on resource-constrained mobile devices may be restrictive. For small device populations, this might result in the over-use of limited communication and compute resources. However, recent deep neural networks crucially depend on a wide range of hyperparameter choices regarding the neural network's architecture, regularization, and optimization. Evaluations can be expensive for large models and large-scale on-device datasets. Hyperparameter optimization (HPO) has a long history under the framework of AutoML [395, 273, 277], but it mainly concerns how to improve the model accuracy [64, 426, 374, 180] rather than communication and computing efficacy for mobile devices. Therefore, we expect that further research should consider developing solutions for efficient hyperparameter optimization in the context of federated learning.

In addition to general-purpose approaches to the hyperparameter optimization problem, in the training space specifically the development of easy-to-tune optimization algorithms is a major open area. Centralized training already requires tuning parameters like learning rate, momentum, batch size, and regularization. Federated learning adds potentially more hyperparameters — separate tuning of the aggregation / global model update rule and local client optimizer, number of clients selected per round, number of local steps per round, configuration of update compression algorithms, and more. Such hyperparameters can be crucial to obtaining a good trade-off between accuracy and convergence, and may actually impact the quality of the learned model [106]. In addition to a higher-dimensional search space, federated learning often also requires longer wall-clock training times and limited compute resources. These challenges could be addressed by optimization algorithms that are robust to hyperparameter settings (the same hyperparameter values work for many different real world datasets and architectures), as well as adaptive or self-tuning algorithms [446, 82].

### 3.4.2 Neural Architecture Design

Neural architecture search (NAS) in the federated learning setting is motivated by the drawbacks of the current practice of applying predefined deep learning models: the predefined architecture of a deep learning model may not be the optimal design choice when the data generated by users are invisible to model developers. For example, the neural architecture may have some redundant component for a specific dataset, which may lead to unnecessary computing on devices; there may be a better architectural design for the non-IID data distribution. The approaches to personalization discussed in Section 3.3 still share the same model architecture among all clients. The recent progress in NAS [230, 387, 175, 388, 60, 375, 313, 488, 175, 323] provides a potential way to address these drawbacks. There are three major methods for NAS, which utilize evolutionary algorithms, reinforcement learning, or gradient descent to search for optimal architectures for a specific task on a specific dataset. Among these, the gradient-based method leverages efficient gradient back-propagation with weight sharing, reducing the architecture search process from over 3000 GPU days to only 1 GPU day. Another interesting paper recently published, involving Weight Agnostic Neural Networks [192], claims that neural network architectures alone, without learning any weight parameters, may encode solutions for a given task. If this technique further develops and reaches widespread use, it may be applied to the federated learning without collaborative training among devices. Although these methods have not been developed for distributed settings such as federated learning, they are all feasible to be transferred to the federated setting. Neural Architecture Search (NAS) for a global or personalized model in the federated learning setting is promising, and early exploration has been made in [228].

### 3.4.1超参数调整

在资源受限的移动的设备上运行具有不同超参数的多轮训练可能是限制性的。对于较小的设备群体，这可能会导致过度使用有限的通信和计算资源。然而，最近的深度神经网络在很大程度上依赖于关于神经网络架构、正则化和优化的各种超参数选择。对于大型模型和大型设备上数据集，评估可能是昂贵的。超参数优化（HPO）在AutoML的框架下有着悠久的历史[395，273，277]，但它主要关注如何提高模型的准确性[64，426，374，180]，而不是移动的设备的通信和计算效率。因此，我们希望进一步的研究应该考虑在联邦学习的背景下开发有效的超参数优化解决方案。

除了超参数优化问题的通用方法之外，在训练空间中，特别是易于调整的优化算法的开发是一个主要的开放领域。集中式训练已经需要调整学习率、动量、批量大小和正则化等参数。联邦学习可能会增加更多的超参数-聚合/全局模型更新规则和本地客户端优化器的单独调整，每轮选择的客户端数量，每轮本地步骤的数量，更新压缩算法的配置等等。这些超参数对于在准确性和收敛性之间获得良好的权衡至关重要，并且实际上可能会影响学习模型的质量[106]。除了更高维的搜索空间外，联邦学习通常还需要更长的挂钟训练时间和有限的计算资源。 这些挑战可以通过对超参数设置具有鲁棒性的优化算法来解决（相同的超参数值适用于许多不同的真实的世界数据集和架构），以及自适应或自调整算法[446，82]。

### 3.4.2神经架构设计

联合学习环境中的神经架构搜索（NAS）是由当前应用预定义深度学习模型的实践的缺点所激发的：当用户生成的数据对模型开发人员不可见时，深度学习模型的预定义架构可能不是最佳设计选择。例如，神经架构对于特定数据集可能有一些冗余组件，这可能导致设备上不必要的计算;对于非IID数据分布可能有更好的架构设计。3.3节中讨论的个性化方法在所有客户端之间仍然共享相同的模型架构。NAS的最新进展[230，387，175，388，60，375，313，488，175，323]提供了解决这些缺点的潜在方法。NAS有三种主要方法，它们利用进化算法、强化学习或梯度下降来搜索特定数据集上特定任务的最佳架构。 其中，基于梯度的方法利用了具有权重共享的高效梯度反向传播，将架构搜索过程从超过3000 GPU天减少到仅1 GPU天。最近发表的另一篇有趣的论文，涉及权重不可知神经网络[192]，声称神经网络架构本身，不需要学习任何权重参数，可以编码给定任务的解决方案。如果该技术进一步发展并得到广泛使用，则可以应用于没有设备之间的协作训练的联合学习。虽然这些方法还没有被开发用于分布式设置，如联邦学习，但它们都是可行的，可以转移到联邦设置。在联邦学习环境中，用于全局或个性化模型的神经架构搜索（NAS）是有前途的，并且在[228]中进行了早期探索。

### 3.4.3 Debugging and Interpretability for FL

While substantial progress has been made on the federated training of models, this is only part of a complete ML workflow. Experienced modelers often directly inspect subsets of the data for tasks including basic sanity checking, debugging misclassifications, discovering outliers, manually labeling examples, or detecting bias in the training set. Developing privacy-preserving techniques to answer such questions on decentralized data is a major open problem. Recently, Augenstein et al. [31] proposed the use of differentially private generative models (including GANs), trained with federated learning, to answer some questions of this type. However, many open questions remain (see discussion in [31]), in particular the development of algorithms that improve the fidelity of FL DP generative models.

## 3.5 Communication and Compression

It is now well-understood that communication can be a primary bottleneck for federated learning since wireless links and other end-user internet connections typically operate at lower rates than intra- or inter-datacenter links and can be potentially expensive and unreliable. This has led to significant recent interest in reducing the communication bandwidth of federated learning. Methods combining Federated Averaging with sparsification and/or quantization of model updates to a small number of bits have demonstrated significant reductions in communication cost with minimal impact on training accuracy [282]. However, it remains unclear if communication cost can be further reduced, and whether any of these methods or their combinations can come close to providing optimal trade-offs between communication and accuracy in federated learning. Characterizing such fundamental trade-offs between accuracy and communication has been of recent interest in theoretical statistics [507, 89, 221, 7, 49, 444, 50]. These works characterize the optimal minimax rates for distributed statistical estimation and learning under communication constraints. However, it is difficult to deduce concrete insights from these theoretical works for communication bandwidth reduction in practice as they typically ignore the impact of the optimization algorithm. It remains an open direction to leverage such statistical approaches to inform practical training methods.

**Compression objectives** Motivated by the limited resources of current devices in terms of compute, memory and communication, there are several different compression objectives of practical value.

(a) *Gradient compression*[6] – reduce the size of the object communicated from clients to server, which is used to update the global model.

(b) *Model broadcast compression* – reduce the size of the model broadcast from server to clients, from which the clients start local training.

(c) *Local computation reduction* – any modification to the overall training algorithm such that the local training procedure is computationally more efficient.

These objectives are in most cases complementary. Among them, (a) has the potential for the most significant practical impact in terms of total runtime. This is both because clients' connections generally have slower upload than download bandwidth[7] – and thus there is more to be gained, compared to (b) – and because the effects of averaging across many clients can enable more aggressive lossy compression schemes. Usually, (c) could be realized jointly with (a) and (b) by specific methods.

[6] In this section, we use "gradient compression" to include compression applied to any model update, such as the updates produced by Federated Averaging when clients take multiple gradient steps.

[7] See for instance `https://www.speedtest.net/reports/`

### 3.4.3 FL的解释性和可解释性

虽然在模型的联合训练方面已经取得了实质性的进展，但这只是完整的ML工作流程的一部分。经验丰富的建模人员通常会直接检查数据的子集，包括基本的健全性检查，调试错误分类，发现离群值，手动标记示例或检测训练集中的偏差。开发隐私保护技术来回答分散数据的这些问题是一个主要的开放问题。最近，Augenstein等人[31]提出使用差分私有生成模型（包括GAN），通过联邦学习进行训练，以回答此类问题。然而，许多开放的问题仍然存在（见[31]中的讨论），特别是提高FL DP生成模型保真度的算法的开发。

## 3.5通信和压缩

现在已经很好地理解了通信可能是联合学习的主要瓶颈，因为无线链路和其他最终用户互联网连接通常以比数据中心内或数据中心间链路更低的速率运行，并且可能是昂贵和不可靠的。这导致了最近对减少联邦学习的通信带宽的兴趣。将联合平均与稀疏化和/或量化模型更新到少量位相结合的方法已证明通信成本显着降低，对训练精度的影响最小[282]。然而，目前尚不清楚是否可以进一步降低通信成本，以及这些方法或它们的组合是否可以在联邦学习中提供通信和准确性之间的最佳权衡。 描述准确性和沟通之间的这种基本权衡是理论统计学最近的兴趣[507，89，221，7，49，444，50]。这些工作的特点是最佳的最小最大速率分布式统计估计和学习通信约束下。然而，这是很难推导出具体的见解，从这些理论工作的通信带宽减少在实践中，因为他们通常忽略了优化算法的影响。利用这种统计方法为实际的培训方法提供信息仍然是一个开放的方向。

压缩目标由于当前设备在计算、存储器和通信方面的资源有限，存在几种不同的具有实用价值的压缩目标。

(a)梯度压缩-减少从客户端到服务器通信的对象的大小，用于更新全局模型。

(b)模型广播压缩-减少从服务器到客户端的模型广播的大小，客户端从该模型广播开始本地训练。

(c)局部计算减少-对整个训练算法的任何修改，使得局部训练过程在计算上更有效。

这些目标在大多数情况下是相辅相成的。其中，（a）在总运行时间方面可能产生最重大的实际影响。这是因为客户端的连接通常具有比下载带宽更慢的上传带宽，因此与（b）相比，可以获得更多，并且因为跨许多客户端进行平均的效果可以实现更积极的有损压缩方案。

通常，（c）可以通过特定方法与（a）和（b）共同实现。

6在本节中，我们使用“梯度压缩”来包括应用于任何模型更新的压缩，例如当客户端采取多个梯度步骤时由联合平均产生的更新。

7参见https://www.speedtest.net/reports/

Much of the existing literature applies to the objective (a) [282, 440, 281, 17, 235, 55]. The impact of (b) on convergence in general has not been studied until very recently; an analysis is presented in [123]. Very few methods intend to address all of (a), (b) and (c) jointly. Caldas et al. [95] proposed a practical method by constraining the desired model update such that only particular submatrices of model variables are necessary to be available on clients; Hamer et al. [219] proposed a communication-efficient federated algorithm for learning mixture weights on an ensemble of pre-trained models, based on communicating only a subset of the models to any one device; He et al. [227] utilizes bidirectional and alternative knowledge distillation method to transfer knowledge from many compact DNNs to a dense server DNN, which can reduce the local computational burden at the edge devices.

In cross-device FL, algorithms generally cannot assume any state is preserved on the clients (Table 1). However, this constraint would typically not be present in the cross-silo FL setting, where the same clients participate repeatedly. Consequently, a wider set of ideas related to error-correction such as [311, 405, 463, 444, 263, 435] are relevant in this setting, many of which could address both (a) and (b).

An additional objective is to modify the training procedure such that the *final* model is more compact, or efficient for inference. This topic has received a lot of attention in the broader ML community [220, 138, 509, 309, 362, 74], but these methods either do not have a straightforward mapping to federated learning, or make the training process more complex which makes it difficult to adopt. Research that simultaneously yields a compact final model, while also addressing the three objectives above, has significant potential for practical impact.

For gradient compression, some existing works [440] are developed in the minimax sense to characterize the worst case scenario. However usually in information theory, the compression guarantees are instance specific and depend on the *entropy* of the underlying distribution [140]. In other words, if the data is easily compressible, they are provably compressed heavily. It would be interesting to see if similar instance specific results can be obtained for gradient compression. Similarly, recent works show that learning a compression scheme in a data-dependent fashion can lead to significantly better compression ratio for the case of data compression [482] as well as gradient compression. It is therefore worthwhile to evaluate these data-dependent compression schemes in the federated settings [193].

**Compatibility with differential privacy and secure aggregation** Many algorithms used in federated learning such as Secure Aggregation [79] and mechanisms of adding noise to achieve differential privacy [3, 338] are not designed to work with compressed or quantized communications. For example, straightforward application of the Secure Aggregation protocol of Bonawitz et al. [80], Bell et al. [58] requires an additional $O(\log M)$ bits of communication for each scalar, where $M$ is the number of clients being summed over, and this may render ineffective the aggressive quantization of updates when $M$ is large (though see [82] for a more efficient approach). Existing noise addition mechanisms assume adding real-valued Gaussian or Laplacian noise on each client, and this is not compatible with standard quantization methods used to reduce communication. We note that several recent works allow biased estimators and would work nicely with Laplacian noise [435], however those would not give differential privacy, as they break independence between rounds. There is some work on adding discrete noise [9], but there is no notion whether such methods are optimal. Joint design of compression methods that are compatible with Secure Aggregation, or for which differential privacy guarantees can be obtained, is thus a valuable open problem.

**Wireless-FL co-design** The existing literature in federated learning usually neglects the impact of wireless channel dynamics during model training, which potentially undermines both training latency and thus reliability of the entire production system. In particular, wireless interference, noisy channels and channel

现有文献中的大部分都适用于目标（a）[282，440，281，17，235，55]。直到最近才研究（B）对一般收敛性的影响;[123]中给出了分析。很少有方法打算同时处理所有（a）、（B）和（c）。Caldas等人。[95]提出了一种实用的方法，通过约束所需的模型更新，使得只有模型变量的特定子矩阵才需要在客户端上可用; Hamer等人。[219]提出了一种通信高效的联邦算法，用于在预先训练的模型集合上学习混合权重，基于仅将模型的子集通信到任何一个设备; He et al. [227]利用双向和替代知识蒸馏方法将知识从许多紧凑的DNN转移到密集的服务器DNN，这可以减少边缘设备的本地计算负担。

在跨设备FL中，算法通常不能假设在客户端上保留任何状态（表1）。
但是，这种约束通常不会出现在跨竖井FL设置中，在这种设置中，相同的客户端重复参与。因此，与纠错相关的更广泛的想法，如[311，405，463，444，263，435]在这种情况下是相关的，其中许多可以解决（a）和（b）。

另一个目标是修改训练过程，使最终模型更紧凑，或更有效的推理。这个主题在更广泛的ML社区中受到了很多关注[220，138，509，309，362，74]，但这些方法要么没有直接映射到联邦学习，要么使训练过程更加复杂，难以采用。研究同时产生一个紧凑的最终模型，同时也解决了上述三个目标，具有重大的实际影响的潜力。

对于梯度压缩，一些现有的工作[440]是在极大极小意义上开发的，以表征最坏情况。然而，通常在信息论中，压缩保证是特定于实例的，并且取决于底层分布的熵[140]。换句话说，如果数据是容易压缩的，那么它们被证明是严重压缩的。看看是否可以为梯度压缩获得类似的实例特定结果，这将是有趣的。类似地，最近的工作表明，以数据依赖的方式学习压缩方案可以导致数据压缩[482]以及梯度压缩的情况下显着更好的压缩比。因此，值得在联邦设置中评估这些数据相关的压缩方案[193]。

与差分隐私和安全聚合的兼容性联邦学习中使用的许多算法，如安全聚合[79]和添加噪声以实现差分隐私的机制[3，338]并不适用于压缩或量化通信。例如，Bonawitz等人的安全聚合协议的直接应用。[80]，Bell等人。[58]需要每个标量的额外O（log M）位通信，其中M是正在求和的客户端的数量，并且当M很大时，这可能会导致更新的积极量化无效（尽管参见[82]更有效的方法）。现有的噪声添加机制假设在每个客户端上添加实值高斯或拉普拉斯噪声，并且这与用于减少通信的标准量化方法不兼容。 我们注意到，最近的几项工作允许有偏估计，并且可以很好地处理拉普拉斯噪声[435]，但是这些工作不会给予差分隐私，因为它们破坏了轮之间的独立性。有一些关于添加离散噪声的工作[9]，但没有概念这些方法是否是最佳的。因此，与安全聚合兼容的压缩方法的联合设计，或者可以获得差分隐私保证，是一个有价值的开放问题。

Wireless-FL协同设计联邦学习的现有文献通常忽略了模型训练期间无线信道动态的影响，这可能会破坏训练延迟，从而影响整个生产系统的可靠性。特别地，无线干扰、噪声信道和信道干扰是不可避免的。

fluctuations can significantly hinder the information exchange between the server and clients (or directly between individual clients, as in the fully decentralized case, see Section 2.1). This represents a major challenge for mission-critical applications, rooted in latency reduction and reliability enhancements. Potential solutions to address this challenge include federated distillation (FD), in which workers exchange their model output parameters (logits) as opposed to the model parameters (gradients and/weights), and optimizing workers' scheduling policy with appropriate communication and computing resources [248, 368, 402]. Another solution is to leverage the unique characteristics of wireless channels (e.g. broadcast and superposition) as natural data aggregators, in which the simultaneously transmitted analog-waves by different workers are superposed at the server and weighed by the wireless channel coefficients [4]. This yields faster model aggregation at the server, and faster training by a factor up to the number of workers. This is in sharp contrast with the traditional orthogonal frequency division multiplexing (OFDM) paradigm, whereby workers upload their models over orthogonal frequencies whose performance degrades with increasing number of workers [174].

## 3.6 Application To More Types of Machine Learning Problems and Models

To date, federated learning has primarily considered supervised learning tasks where labels are naturally available on each client. Extending FL to other ML paradigms, including reinforcement learning, semi-supervised and unsupervised learning, active learning, and online learning [226, 508] all present interesting and open challenges.

Another important class of models, highly relevant to FL, are those that can characterize the uncertainty in their predictions. Most modern deep learning models cannot represent their uncertainty nor allow for a probability interpretation of parametric learning. This has motivated recent developments of tools and techniques combining Bayesian models with deep learning. From a probability theory perspective, it is unjustifiable to use single point-estimates for classification. Bayesian neural networks [419] have been proposed and shown to be far more robust to over-fitting, and can easily learn from small datasets. The Bayesian approach further offers uncertainty estimates via its parameters in form of probability distributions, thus preventing over-fitting. Moreover, appealing to probabilistic reasoning, one can predict how the uncertainty can decrease, allowing the decisions made by the network to become more deterministic as the data size grows.

Since Bayesian methods gave us tools to reason about deep models' confidence and also achieve state-of-the-art performance on many tasks, one expects Bayesian methods to provide a conceptual improvement to the classical federated learning. In fact, preliminary work from Lalitha et al. [292] shows that incorporating Bayesian methods allows for model aggregation across non-IID data and heterogeneous platforms. However, many questions regarding scalability and computational feasibility have to be addressed.

## 3.7 Executive summary

Efficient and effective federated learning algorithms face different challenges compared to centralized training in a datacenter.

- Non-IID data due to non-identical client distributions, violation of independence, and dataset drift (Section 3.1) pose a key challenge. Though various methods have been surveyed and discussed in this section, defining and dealing with non-IID data remains an open problem and one of the most active research topics in federated learning.

波动可能会严重阻碍服务器和客户端之间的信息交换（或者直接在各个客户端之间进行，如在完全分散的情况下，参见第2.1节）。这对任务关键型应用程序来说是一个重大挑战，其根源在于减少延迟和增强可靠性。应对这一挑战的潜在解决方案包括联合蒸馏（FD），其中工作人员交换其模型输出参数（logits）而不是模型参数（梯度和/或权重），并通过适当的通信和计算资源优化工作人员的调度策略[248，368，402]。另一种解决方案是利用无线信道的独特特性（例如广播和叠加）作为自然数据聚合器，其中不同工作人员同时发送的模拟波在服务器处叠加并通过无线信道系数进行加权[4]。 这将在服务器上产生更快的模型聚合，并通过高达工作器数量的因子更快地进行训练。这与传统的正交频分复用（OFDM）模式形成鲜明对比，其中工作人员通过正交频率上传其模型，其性能随着工作人员数量的增加而下降[174]。

## 3.6应用于更多类型的机器学习问题和模型

到目前为止，联邦学习主要考虑监督学习任务，其中标签在每个客户端上自然可用。将FL扩展到其他ML范式，包括强化学习，半监督和无监督学习，主动学习和在线学习[226，508]都提出了有趣和开放的挑战。

另一类重要的模型，与FL高度相关，是那些可以描述其预测中的不确定性的模型。大多数现代深度学习模型不能表示其不确定性，也不允许对参数学习进行概率解释。这推动了最近将贝叶斯模型与深度学习相结合的工具和技术的发展。从概率论的角度来看，使用单点估计进行分类是不合理的。贝叶斯神经网络[419]已经被提出，并被证明对过拟合更鲁棒，并且可以很容易地从小数据集学习。贝叶斯方法还通过其参数以概率分布的形式提供不确定性估计，从而防止过度拟合。此外，借助概率推理，人们可以预测不确定性如何降低，从而使网络做出的决策随着数据大小的增加而变得更加确定。

由于贝叶斯方法为我们提供了推理深度模型置信度的工具，并且在许多任务上实现了最先进的性能，因此人们期望贝叶斯方法为经典的联邦学习提供概念上的改进。事实上，Lalitha等人的初步工作。[292]表明，结合贝叶斯方法允许跨非IID数据和异构平台进行模型聚合。然而，关于可扩展性和计算可行性的许多问题必须得到解决。

## 3.7执行摘要

与数据中心的集中式训练相比，高效和有效的联邦学习算法面临着不同的挑战。

·由于客户端分布不同、违反独立性和数据集漂移（第3.1节）而导致的非IID数据构成了一个关键挑战。虽然本节已经调查和讨论了各种方法，但定义和处理非IID数据仍然是一个开放的问题，也是联邦学习中最活跃的研究课题之一。

- Optimization algorithms for federated learning are analyzed in Section 3.2 under different settings, e.g., convex and nonconvex functions, IID and non-IID data. Theoretical analysis has proven difficult for the parallel local updates commonly used in federated optimization, and often strict assumptions have to be made to constrain the client heterogeneity. Currently, known convergence rates do not fully explain the empirically-observed effectiveness of the Federated Averaging algorithm over methods such as mini-batch SGD [481].

- Client-side personalization and "multi-model" approaches (Section 3.3) can address data heterogeneity and give hope of surpassing the performance of the best fixed global model. Simple personalization methods like fine-tuning can be effective, and offer intrinsic privacy advantages. However, many theoretical and empirical questions remain open: when is a global model better? How many models are necessary? Which federated optimization algorithms combine best with local fine-tuning?

- Adapting centralized training workflows such as hyper-parameter tuning, neural architecture design, debugging, and interpretability tasks to the federated learning setting (Section 3.4) present roadblocks to the widespread adoption of FL in practical settings, and hence constitute important open problems.

- While there has been significant work on communication efficiency and compression for FL (Section 3.5), it remains an important and active area. In particular, fully automating the process of enabling compression without impacting convergence for a wide class of models is an important practical goal. Relatively new directions on the theoretical study of communication, compatibility with privacy methods, and co-design with wireless infrastructure are discussed.

- There are many open questions in extending federated learning from supervised tasks to other machine learning paradigms including reinforcement learning, semi-supervised and unsupervised learning, active learning, and online learning (Section 3.6).

·联邦学习的优化算法在第3.2节中在不同的设置下进行了分析，例如，凸函数和非凸函数，IID和非IID数据。联邦优化中常用的并行本地更新方法理论分析困难，通常需要做出严格的假设来约束客户端的异构性。目前，已知的收敛速度并不能完全解释联合平均算法在小批量SGD [481]等方法上的实验观察有效性。

·客户端个性化和“多模型”方法（第3.3节）可以解决数据异构性问题，并给予超越最佳固定全局模型性能的希望。像微调这样简单的个性化方法可能是有效的，并提供内在的隐私优势。然而，许多理论和经验问题仍然悬而未决：什么时候全球模型更好？需要多少型号？哪种联邦优化算法联合收割机与局部微调结合得最好？

·使集中式训练工作流程（如超参数调整、神经架构设计、调试和可解释性任务）适应联邦学习环境（第3.4节），这对FL在实际环境中的广泛采用构成了障碍，因此构成了重要的开放问题。

虽然在FL的通信效率和压缩方面已经做了大量工作（第3.5节），但它仍然是一个重要和活跃的领域。特别是，完全自动化启用压缩的过程而不影响广泛类别模型的收敛是一个重要的实际目标。讨论了通信理论研究的相对新的方向，与隐私方法的兼容性，以及与无线基础设施的协同设计。

·将联邦学习从监督任务扩展到其他机器学习范式，包括强化学习、半监督和无监督学习、主动学习和在线学习，还有许多悬而未决的问题（第3.6节）。

# 4 Preserving the Privacy of User Data

Machine learning workflows involve many actors functioning in disparate capacities. For example, users may generate training data through interactions with their devices, a machine learning training procedure extracts cross-population patterns from this data (e.g. in the form of trained model parameters), the machine learning engineer or analyst may assess the quality of this trained model, and eventually the model may be deployed to end users in order to support specific user experiences (see Figure 1 below).

In an ideal world, each actor in the system would learn nothing more than the information needed to play their role. For example, if an analyst only needs to determine whether a particular quality metric exceeds a desired threshold in order to authorize deploying the model to end users, then in an idealized world, that is the only bit of information that would be available to the analyst; such an analyst would need access to neither the training data nor the model parameters, for instance. Similarly, end users enjoying the user experiences powered by the trained model might only require predictions from the model and nothing else.

Furthermore, in an ideal world every participant in the system would be able to reason easily and accurately about what personal information about themselves and others might be revealed by their participation in the system, and participants would be able to use this understanding to make informed choices about how and whether to participate at all.

Producing a system with all of the above ideal privacy properties would be a daunting feat on its own, and even more so while also guaranteeing other desirable properties such as ease of use for all participants, the quality and fairness of the end user experiences (and the models that power them), the judicious use of communication and computation resources, resilience against attacks and failures, and so on.

Rather than allowing perfect to be the enemy of good, we advocate a strategy wherein the overall system is composed of modular units which can be studied and improved relatively independently, while also reminding ourselves that we must, in the end, measure the privacy properties of the complete system against our ideal privacy goals set out above. The open questions raised throughout this section will highlight areas

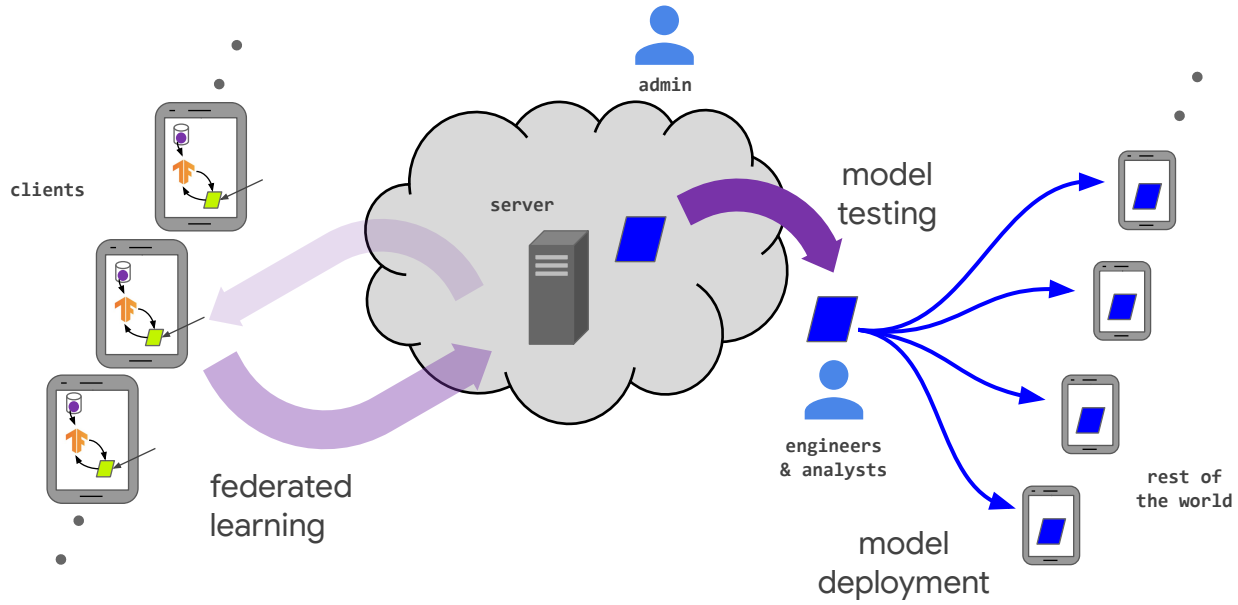



Figure 1: The lifecycle of an FL-trained model and the various actors in a federated learning system. (repeated from Page 7)

## 4保护用户数据的隐私

机器学习工作流程涉及许多以不同能力运作的参与者。例如，用户可以通过与他们的设备交互来生成训练数据，机器学习训练过程从该数据中提取交叉群体模式（例如，以训练的模型参数的形式），机器学习工程师或分析师可以评估该训练模型的质量，并且最终可以将该模型部署到终端用户以支持特定的用户体验（参见下面的图1）。

在一个理想的世界里，系统中的每一个参与者只会学到扮演自己角色所需的信息。例如，如果一个分析师只需要确定一个特定的质量指标是否超过了期望的阈值，以便授权将模型部署给最终用户，那么在理想化的世界中，这是分析师唯一可用的信息;这样的分析师既不需要访问训练数据，也不需要访问模型参数。同样，享受由训练模型提供的用户体验的最终用户可能只需要模型的预测，而不需要其他任何东西。

此外，在一个理想的世界中，系统中的每个参与者都能够轻松而准确地推理出他们参与系统可能会泄露自己和他人的哪些个人信息，参与者将能够利用这种理解来做出明智的选择，决定如何参与以及是否参与。

生产一个具有上述所有理想隐私属性的系统本身将是一项艰巨的任务，而且还要保证其他理想的属性，例如所有参与者的易用性，最终用户体验的质量和公平性（以及为其提供动力的模型），明智地使用通信和计算资源，抵御攻击和故障的弹性，等等。

而不是让完美成为好的敌人，我们提倡一种策略，其中整个系统是由模块化单元组成的，可以相对独立地进行研究和改进，同时也提醒自己，我们必须，最后，衡量完整系统的隐私属性对我们的理想隐私目标上述。本节中提出的开放性问题将突出以下方面：

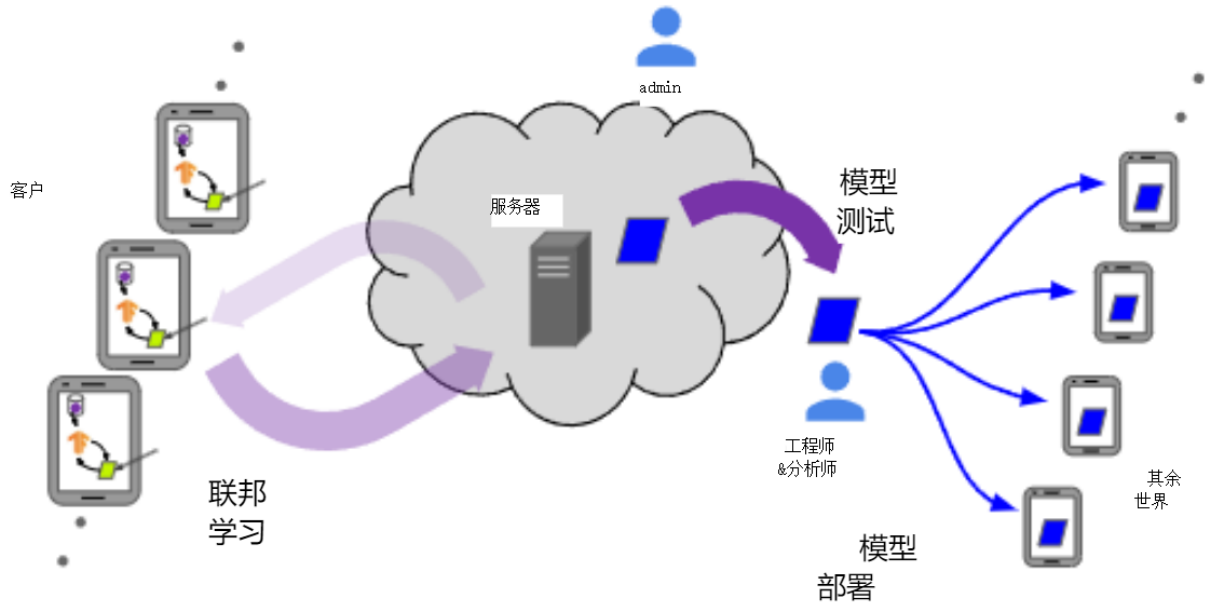



图1：FL训练模型的生命周期和联邦学习系统中的各种参与者。（重定向自第7页）

wherein we do not yet understand how to simultaneously achieve all of our goals, either for an individual module or for the system as a whole.

Federated learning provides an attractive structure for decomposing the overall machine learning workflow into the approachable modular units we desire. One of the primary attractions of the federated learning model is that it can provide a level of privacy to participating users through data minimization: the raw user data never leaves the device, and only updates to models (e.g., gradient updates) are sent to the central server. These model updates are more focused on the learning task at hand than is the raw data (i.e. they contain strictly no additional information about the user, and typically significantly less, compared to the raw data), and the individual updates only need to be held ephemerally by the server.

While these features can offer significant practical privacy improvements over centralizing all the training data, there is still no formal guarantee of privacy in this baseline federated learning model. For instance, it is possible to construct scenarios in which information about the raw data is leaked from a client to the server, such as a scenario where knowing the previous model and the gradient update from a user would allow one to infer a training example held by that user. Therefore, this section surveys existing results and outlines open challenges towards designing federated learning systems that can offer rigorous privacy guarantees. We focus on questions specific to the federated learning and analytics setting and leave aside questions that also arise in more general machine learning settings as surveyed in [344].

Beyond attacks targeting user privacy, there are also other classes of attacks on federated learning; for example, an adversary might attempt to prevent a model from being learned at all, or they might attempt to bias the model to produce inferences that are preferable to the adversary. We defer consideration of these types of attacks to Section 5.

The remainder of this section is organized as follows. Section 4.1 discusses various threat models against which we wish to give protections. Section 4.2 lays out a set of core tools and technologies that can be used towards providing rigorous protections against the threat models discussed in Section 4.1. Section 4.3 assumes the existence of a trusted server and discusses the open problems and challenges in providing protections against adversarial clients and/or analysts. Section 4.4 discusses the open problems and challenges in the absence of a fully trusted server. Finally, Section 4.5 discusses open questions around user perception.

## 4.1 Actors, Threat Models, and Privacy in Depth

A formal treatment of privacy risks in FL calls for a holistic and interdisciplinary approach. While some of the risks can be mapped to technical privacy definitions and mitigated with existing technologies, others are more complex and require cross-disciplinary efforts.

Privacy is not a binary quantity, or even a scalar one. This first step towards such formal treatment is a careful characterization of the different actors (see Figure 1 from Section 1, repeated on page 36 for convenience) and their roles to ultimately define relevant threat models (see Table 7). Thus, for instance, it is desirable to distinguish the view of the server administrator from the view of the analysts that consume the learned models, as it is conceivable that a system that is designed to offer strong privacy guarantees against a malicious analyst may not provide any guarantees with respect to a malicious server. These actors map well onto the threat models discussed elsewhere in the literature; for example, in Bittau et al. [73, Sec 3.1], where the "encoder" corresponds to the client, the "shuffler" generally corresponds to the server, the "analyzer" may correspond to the server or post-processing done by the analyst.

其中，我们还不知道如何同时实现我们的所有目标，无论是对于单个模块还是对于整个系统。

联邦学习提供了一个有吸引力的结构，可以将整个机器学习工作流分解为我们所期望的可接近的模块化单元。联合学习模型的主要吸引力之一是，它可以通过数据最小化为参与用户提供一定程度的隐私：原始用户数据永远不会离开设备，并且仅更新模型（例如，梯度更新）被发送到中央服务器。与原始数据相比，这些模型更新更关注手头的学习任务（即，与原始数据相比，它们严格不包含关于用户的附加信息，并且通常显著更少），并且单个更新仅需要由服务器暂时保存。

虽然这些功能可以在集中所有训练数据的基础上提供显着的实际隐私改进，但在这个基线联邦学习模型中仍然没有正式的隐私保证。例如，可以构建关于原始数据的信息从客户端泄露到服务器的场景，例如，从用户那里知道先前的模型和梯度更新将允许人们推断该用户持有的训练示例的场景。因此，本节调查了现有的结果，并概述了设计可以提供严格隐私保证的联邦学习系统所面临的挑战。我们专注于联邦学习和分析设置的特定问题，并将[344]中调查的更一般的机器学习设置中出现的问题放在一边。

除了针对用户隐私的攻击之外，还有其他类型的针对联邦学习的攻击;例如，对手可能会试图阻止模型学习，或者他们可能会试图使模型产生偏向于对手的推断。我们将这类攻击的考虑推迟到第5节。

本节的其余部分组织如下。第4.1节讨论了我们希望给予保护的各种威胁模型。第4.2节列出了一组核心工具和技术，可用于针对第4.1节中讨论的威胁模型提供严格的保护。第4.3节假设存在可信服务器，并讨论了在提供针对敌对客户端和/或分析师的保护方面存在的问题和挑战。第4.4节讨论了在缺乏完全受信任的服务器的情况下存在的问题和挑战。最后，第4.5节讨论了围绕用户感知的开放性问题。

## 4.1参与者、威胁模型和隐私深度

在FL的隐私风险的正式治疗需要一个整体的和跨学科的方法。虽然有些风险可以映射到技术隐私定义，并通过现有技术加以缓解，但其他风险则更为复杂，需要跨学科的努力。

隐私不是一个二进制的量，甚至不是一个标量。正式处理的第一步是仔细描述不同的参与者（参见第1节中的图1，为方便起见在第36页重复）及其角色，以最终定义相关的威胁模型（参见表7）。因此，例如，期望将服务器管理员的观点与消费所学习的模型的分析员的观点区分开，因为可以想到，被设计为针对恶意分析员提供强隐私保证的系统可能不提供关于恶意服务器的任何保证。这些参与者很好地映射到文献中其他地方讨论的威胁模型;例如，在Bittau等人[73，Sec 3.1]中，其中“编码器”对应于客户端，“洗牌器”通常对应于服务器，“分析器”可能对应于服务器或分析师完成的后处理。

As an example, a particular system might offer a differential privacy[8] guarantee with a particular parameter $\varepsilon$ to the view of the server administrator, while the results observed by analysts might have a higher protection $\varepsilon' < \varepsilon$.

Furthermore, it is possible that this guarantee holds only against adversaries with particular limits on their capabilities, e.g. an adversary that can observe everything that happens on the server (but cannot influence the server's behavior) while simultaneously controlling up to a fraction $\gamma$ of the clients (observing everything they see and influencing their behavior in arbitrary ways); the adversary might also be assumed to be unable to break cryptographic mechanisms instantiated at a particular security level $\sigma$. Against an adversary whose strength *exceeds* these limits, the view of the server administrator might still have some differential privacy, but at weaker level $\varepsilon_0 > \varepsilon$.

As we see in this example, precisely specifying the assumptions and privacy goals of a system can easily implicate concrete instantiations of several parameters ($\varepsilon, \varepsilon', \varepsilon_0, \gamma, \sigma$, etc.) as well as concepts such as differential privacy and honest-but-curious security.

Achieving all the desired privacy properties for federated learning will typically require composing many of the tools and technologies described below into an end-to-end system, potentially both layering multiple strategies to protect the same part of the system (e.g. running portions of a Secure Multi-Party Computation (MPC) protocol inside a Trusted Execution Environment (TEE) to make it harder for an adversary to sufficiently compromise that component) as well as using different strategies to protect different parts of the system (e.g. using MPC to protect the aggregation of model updates, then using Private Disclosure techniques before sharing the aggregate updates beyond the server).

As such, we advocate for building federated systems wherein the privacy properties degrade as gracefully as possible in cases where one technique or another fails to provide its intended privacy contribution. For example, running the server component of an MPC protocol inside a TEE might allow privacy to be maintained even in the case where either (but not both) of the TEE security or MPC security assumptions fails to hold in practice. As another example, requiring clients to send raw training examples to a server-side TEE would be strongly dispreferred to having clients send gradient updates to a server-side TEE, as the latter's privacy expectations degrade much more gracefully if the TEE's security were to fail. We refer to this principle of graceful degradation as "Privacy in Depth," in analogy to the well-established network security principle of defense in depth [361].

## 4.2 Tools and Technologies

Generally speaking, the goal of an FL computation is for the analyst or engineer requesting the computation to obtain the result, which can be thought of as the evaluation of a function $f$ on a distributed client dataset (commonly an ML model training algorithm, but possibly something simpler such as a basic statistic). There are three privacy aspects that need to be addressed.

First, we need to consider *how* $f$ is computed and what is the information flow of intermediate results in the process, which primarily influences the susceptibility to malicious client, server, and admin actors. In addition to designing the flow of information in the system (e.g. early data minimization), techniques from secure computation including Secure Multi-Party Computation (MPC) and Trusted Execution Environments (TEEs) are of particular relevance to addressing these concerns. These technologies will be discussed in detail in Section 4.2.1.

[8] Differential privacy will be formally introduced in Section 4.2.2. For now, it suffices to know that lower $\varepsilon$ corresponds with higher privacy.

作为一个例子，一个特定的系统可能会提供一个具有特定参数ε的差异隐私保证给服务器管理员，而分析师观察到的结果可能具有更高的保护ε< ε。

此外，这种保证可能仅适用于能力受到特定限制的对手，例如可以观察服务器上发生的一切的对手（但不能影响服务器的行为），同时控制多达一小部分γ的客户端（观察他们所看到的一切，并以任意的方式影响他们的行为）;还可以假定对手不能破解在特定安全级别σ下实例化的密码机制。对于实力超过这些限制的对手，服务器管理员的视图可能仍然具有一些差异隐私，但在较弱的水平ε> ε。

正如我们在这个例子中看到的，精确地指定系统的假设和隐私目标可以很容易地涉及几个参数（ε，γ，σ等）的具体实例。以及诸如差别隐私和诚实但好奇的安全等概念。

实现联邦学习的所有期望的隐私属性通常需要将下面描述的许多工具和技术组合到端到端系统中，潜在地既分层多个策略以保护系统的同一部分，（例如，在可信执行环境（TEE）内运行安全多方计算（MPC）协议的部分）以使对手更难充分地危害该组件）以及使用不同的策略来保护系统的不同部分（例如，使用MPC来保护模型更新的聚合，然后在服务器之外共享聚合更新之前使用私有披露技术）。

因此，我们提倡构建联邦系统，在这种情况下，隐私属性尽可能优雅地降级，其中一种技术或另一种技术无法提供其预期的隐私贡献。例如，在TEE内部运行MPC协议的服务器组件可以允许即使在TEE安全性或MPC安全性假设中的任一个（但不是两者）在实践中不能成立的情况下也保持隐私。作为另一个示例，要求客户端向服务器侧TEE发送原始训练示例将与让客户端向服务器侧TEE发送梯度更新强烈不一致，因为如果TEE的安全性失败，则后者的隐私期望会优雅得多地降级。我们将这种优雅降级的原则称为“深度隐私”，类似于成熟的网络安全深度防御原则[361]。

## 4.2工具和技术

一般来说，FL计算的目标是让分析师或工程师请求计算以获得结果，这可以被认为是对分布式客户端数据集上的函数f的评估（通常是ML模型训练算法，但可能是更简单的东西，如基本统计）。有三个隐私问题需要解决。

首先，我们需要考虑f是如何计算的，以及过程中中间结果的信息流是什么，这主要影响对恶意客户端，服务器和管理员的敏感性。除了设计系统中的信息流（例如早期数据最小化）之外，安全计算技术（包括安全多方计算（MPC）和可信执行环境（TEE））与解决这些问题特别相关。这些技术将在第4.2.1节中详细讨论。

8差异隐私将在4.2.2节中正式引入。目前，只要知道较低的ε对应于较高的隐私就足够了。

| Data/Access Point | Actor | Threat Model |
|---|---|---|
| Clients | Someone who has root access to the client device, either by design or by compromising the device | Malicious clients can inspect all messages received from the server (including the model iterates) in the rounds they participate in and can tamper with the training process. An honest-but-curious client can inspect all messages received from the server but cannot tamper with the training process. In some cases, technologies such as secure enclaves/TEEs may be able to limit the influence and visibility of such an attacker, representing a meaningfully weaker threat model. |
| Server | Someone who has root access to the server, either by design or by compromising the device | A malicious server can inspect all messages sent to the server (including the gradient updates) in all rounds and can tamper with the training process. An honest-but-curious server can inspect all messages sent to the server but cannot tamper with the training process. In some cases, technologies such as secure enclaves/TEEs may be able to limit the influence and visibility of such an attacker, representing a meaningfully weaker threat model. |
| Output Models | Engineers & analysts | A malicious analyst or model engineer may have access to multiple outputs from the system, e.g. sequences of model iterates from multiple training runs with different hyperparameters. Exactly what information is released to this actor is an important system design question. |
| Deployed Models | The rest of the world | In cross-device FL, the final model may be deployed to hundreds of millions of devices. A partially compromised device can have black-box access to the learned model, and a fully compromised device can have a white-box access to the learned model. |

Table 7: Various threat models for different adversarial actors.

| 数据/接入点参与者威胁模型 | |
|---|---|
| 客户拥有root访问权限的人<br>通过设计或通过损害设备， | 恶意客户端可以检查从服务器接收的所有消息（包括模型迭代），并可以篡改训练过程。一个诚实但好奇的客户端可以检查从服务器接收到的所有消息，但不能篡改训练过程。在某些情况下，诸如安全飞地/TEE之类的技术可能能够限制这种攻击者的影响力和可见性，从而表示有意义的较弱威胁模型。 |
| 服务器拥有root访问权限的人<br>通过设计或破坏设备 | 恶意服务器可以在所有轮中检查发送到服务器的所有消息（包括梯度更新），并可以篡改训练过程。诚实但好奇的服务器可以检查发送到服务器的所有消息，但不能篡改训练过程。在某些情况下，诸如安全飞地/TEE之类的技术可能能够限制这种攻击者的影响力和可见性，从而表示有意义的较弱威胁模型。 |
| 输出模型工程师和分析师恶意分析师或模型工程师可能会访问系统的多个输出，例如： | |
| | 从具有不同超参数的多个训练运行中迭代的模型序列。确切地说，向这个参与者发布什么信息是一个重要的系统设计问题。 |
| 在跨设备FL中，最终模型可能会被删除， | |
| | 应用于数亿台设备。部分受损的设备可以具有对学习模型的黑盒访问，并且完全受损的设备可以具有对学习模型的白盒访问。 |

表7：针对不同敌对行为者的各种威胁模型。

Second, we have to consider *what* is computed. In other words, how much information about a participating client is revealed to the analyst and world actors by the result of $f$ itself. Here, techniques for privacy-preserving disclosure, particularly differential privacy (DP), are highly relevant and will be discussed in detail in Section 4.2.2.

Finally, there is the problem of *verifiability*, which pertains to the ability of a client or the server to prove to others in the system that they have executed the desired behavior faithfully, without revealing the potentially private data upon which they were acting. Techniques for verifiability, including remote attestation and zero-knowledge proofs, will be discussed in Section 4.2.3.

### 4.2.1 Secure Computations

The goal of secure computation is to evaluate functions on distributed inputs in a way that only reveals the result of the computation to the intended parties, without revealing any additional information (e.g. the parties' inputs or any intermediate results).

**Secure multi-party computation** Secure Multi-Party Computation (MPC) is a subfield of cryptography concerned with the problem of having a set of parties compute an agreed-upon function of their private inputs in a way that only reveals the intended output to each of the parties. This area was kicked off in the 1980's by Yao [493]. Thanks to both theoretical and engineering breakthroughs, the field has moved from being of a purely theoretical interest to a deployed technology in industry [78, 77, 295, 27, 191, 242, 243]. It is important to remark that MPC defines a set of technologies, and should be regarded more as a field, or a general notion of security in secure computation, than a technology *per se*. Some of the recent advances in MPC can be attributed to breakthroughs in lower level primitives, such as oblivious transfer protocols [244] and encryption schemes with homomorphic properties (as described below).

A common aspect of cryptographic solutions is that operations are often done on a finite field (e.g. integers modulo a prime $p$), which poses difficulties when representing real numbers. A common approach has been to adapt ML models and their training procedures to ensure that (over)underflows are controlled, by operating on normalized quantities and relying on careful quantization [194, 10, 206, 84].

It has been known for several decades that any function can be securely computed, even in the presence of malicious adversaries [208]. While generic solutions exist, their performance characteristics often render them inapplicable in practical settings. As such a noticeable trend in research has consisted in designing custom protocols for applications such as linear and logistic regression [359, 194, 351] and neural network training and inference [351, 10, 48]. These works are typically in the cross-silo setting, or the variant where computation is delegated to a small group of computing servers that do not collude with each other. Porting these protocols to the cross-device setting is not straightforward, as they require a significant amount of communication.

*Homomorphic encryption* Homomorphic encryption (HE) schemes allow certain mathematical operations to be performed directly on ciphertexts, without prior decryption. Homomorphic encryption can be a powerful tool for enabling MPC by enabling a participant to compute functions on values while keeping the values hidden.

Different flavours of HE exist, ranging from general fully homomorphic encryption (FHE) [197] to the more efficient leveled variants [87, 182, 88, 129], for which several implementations exist [233, 409, 364, 415, 1]. Also of practical relevance are the so-called partially homomorphic schemes, including for example ElGamal and Paillier, allowing either homomorphic addition or multiplication. Additive HE has been used

其次，我们必须考虑计算的内容。换句话说，通过f本身的结果，有多少关于参与客户的信息被透露给分析师和世界参与者。这里，用于隐私保护公开的技术，特别是差分隐私（DP），是高度相关的，并且将在第4.2.2节中详细讨论。

最后，还有可验证性的问题，这涉及到客户端或服务器向系统中的其他人证明他们已经忠实地执行了所需的行为，而不会泄露他们所操作的潜在私人数据的能力。可验证性技术，包括远程证明和零知识证明，将在4.2.3节中讨论。

### 4.2.1安全计算

安全计算的目标是在分布式输入上评估函数，其方式是仅向预期方显示计算结果，而不显示任何附加信息（例如各方的输入或任何中间结果）。

安全多方计算（英语：Secure Multi-Party Computation，MPC）是密码学的一个子领域，它涉及的问题是让一组参与方计算他们私人输入的商定函数，并且只向每一方显示预期的输出。这一领域是由姚明在1980年代开始的[493]。由于理论和工程上的突破，该领域已经从纯粹的理论兴趣转变为工业上的部署技术[78，77，295，27，191，242，243]。重要的是要注意，MPC定义了一组技术，并且应该更多地被视为一个领域，或者安全计算中的安全性的一般概念，而不是技术本身。MPC的一些最新进展可以归因于较低级别原语的突破，例如不经意传输协议[244]和具有同态属性的加密方案（如下所述）。

密码解决方案的一个共同方面是操作通常在有限域上完成（例如，
整数模素数p），这在表示真实的数时造成困难。一种常见的方法是调整ML模型及其训练过程，以确保通过对归一化量进行操作并依赖于仔细的量化来控制（过）下溢[194，10，206，84]。

几十年来，人们已经知道，任何函数都可以安全地计算，即使在存在恶意对手的情况下[208]。虽然存在通用解决方案，但其性能特点往往使其不适用于实际环境。因此，研究中的一个值得注意的趋势是为线性和逻辑回归[359，194，351]以及神经网络训练和推理[351，10，48]等应用设计自定义协议。这些工作通常是在跨竖井设置中，或者是将计算委托给一小群彼此不勾结的计算服务器的变体。将这些协议移植到跨设备设置并不简单，因为它们需要大量的通信。

同态加密（英语：Homomorphic encryption，缩写：HE）允许直接对密文执行某些数学运算，而无需事先解密。同态加密可以是一个强大的工具，通过使参与者能够计算值的函数，同时保持隐藏的值，使MPC。

存在不同风格的HE，从一般的全同态加密（FHE）[197]到更有效的分级变体[87，182，88，129]，其中存在几种实现[233，409，364，415，1]。也有实际意义的是所谓的部分同态方案，包括例如ElGamal和Paillier，允许同态加法或乘法。添加剂HE已被使用

| Technology | Characteristics |
|---|---|
| Differential Privacy (local, central, shuffled, aggregated, and hybrid models) | A quantification of how much information could be learned about an individual from the output of an analysis on a dataset that includes the user. Algorithms with differential privacy necessarily incorporate some amount of randomness or noise, which can be tuned to mask the influence of the user on the output. |
| Secure Multi-Party Computation | Two or more participants collaborate to simulate, though cryptography, a fully trusted third party who can:<br>• Compute a function of inputs provided by all the participants;<br>• Reveal the computed value to a chosen subset of the participants, with no party learning anything further. |
| Homomorphic Encryption | Enables a party to compute functions of data to which they do not have plain-text access, by allowing mathematical operations to be performed on ciphertexts without decrypting them. Arbitrarily complicated functions of the data can be computed this way ("Fully Homomorphic Encryption") though at greater computational cost. |
| Trusted Execution Environments (secure enclaves) | TEEs provide the ability to trustably run code on a remote machine, even if you do not trust the machine's owner/administrator. This is achieved by limiting the capabilities of any party, including the administrator. In particular, TEEs may provide the following properties [437]:<br>• Confidentiality: The state of the code's execution remains secret, unless the code explicitly publishes a message;<br>• Integrity: The code's execution cannot be affected, except by the code explicitly receiving an input;<br>• Measurement/Attestation: The TEE can prove to a remote party what code (binary) is executing and what its starting state was, defining the initial conditions for confidentiality and integrity. |

Table 8: Various technologies along with their characteristics.

| 技术特点 | |
|---|---|
| 差异隐私（本地、中央、混洗、聚合和混合模型） | 对从包含用户的数据集的分析输出中可以了解到的个人信息量的量化。具有差分隐私的算法必然会包含一定量的随机性或噪声，这些随机性或噪声可以被调整以掩盖用户对输出的影响。 |
| 安全多方计算两个或多个参与者合作进行模拟，尽管加密 | raphy，一个完全可信的第三方，可以：<br>·计算由所有参与者提供的输入的函数;<br>·向选定的参与者子集显示计算值，没有任何一方进一步学习。 |
| 同态加密使一方能够计算他们所做的数据的函数 | 通过允许对密文执行数学运算而不解密它们，不具有明文访问。数据的复杂函数可以通过这种方式计算（“全同态加密”），尽管计算成本更高。 |
| 可信执行环境（安全飞地） | TEE提供了在远程机器上可信地运行代码的能力，即使您不信任机器的所有者/管理员。这是通过限制包括管理员在内的任何一方的能力来实现的。特别是，TEE可以提供以下属性[437]：<br>机密性：代码的执行状态保持秘密，除非代码显式发布消息;<br>完整性：代码的执行不会受到影响，除非代码显式地接收输入;<br>·测量/认证：TEE可以向远程方证明正在执行的代码（二进制）及其起始状态，定义机密性和完整性的初始条件。 |

表8：各种技术及其特性沿着。

as an ingredient in MPC protocols in the cross-silo setting [359, 224]. A review of some homomorphic encryption software libraries along with brief explanations of criteria/features to be considered in choosing a library is surveyed in [404].

When considering the use of HE in the FL setting, questions immediately arise about who holds the secret key of the scheme. While the idea of every client encrypting their data and sending it to the server to compute homomorphically on it is appealing, the server should not be able to decrypt a single client contribution. A trivial way of overcoming this issue would be relying on a non-colluding external party that holds the secret key and decrypts the result of the computation. However, most HE schemes require that the secret keys be renewed often (due to e.g. susceptibility to chosen ciphertext attacks [117]). Moreover, the availability of a trusted non-colluding party is not standard in the FL setting.

Another way around this issue is relying on distributed (or threshold) encryption schemes, where the secret key is distributed among the parties. Reyzin et al. [392] and Roth et al. [398] propose such solutions for computing summation in the cross-device setting. Their protocols make use of additively homomorphic schemes (variants of ElGamal and lattice-based schemes, respectively).

**Trusted execution environments** Trusted execution environments (TEEs, also referred to as secure enclaves) may provide opportunities to move part of the federated learning process into a trusted environment in the cloud, whose code can be attested and verified.

TEEs can provide several crucial facilities for establishing trust that a unit of code has been executed faithfully and privately [437]:

- Confidentiality: The state of the code's execution remains secret, unless the code explicitly publishes a message.
- Integrity: The code's execution cannot be affected, except by the code explicitly receiving an input.
- Measurement/Attestation: The TEE can prove to a remote party what code (binary) is executing and what its starting state was, defining the initial conditions for confidentiality and integrity.

TEEs have been instantiated in many forms, including Intel's SGX-enabled CPUs [241, 134], Arm's TrustZone [28, 22], and Sanctum on RISC-V [135], each varying in its ability to systematically offer the above facilities.

Current secure enclaves are limited in terms of memory and provide access only to CPU resources, that is they do not allow processing on GPUs or machine learning processors (Tramèr and Boneh [447] explore how to combine TEEs with GPUs for machine learning inference). Moreover, it is challenging for TEEs (especially those operating on shared microprocessors) to fully exclude all types of side channel attacks [458].

While secure enclaves provide protections for all code running inside them, there are additional concerns that must be addressed in practice. For example, it is often necessary to structure the code running in the enclave as a data oblivious procedure, such that its runtime and memory access patterns do not reveal information about the data upon which it is computing (see for example [73]). Furthermore, measurement/attestation typically only proves that a particular binary is running; it is up to the system architect to provide a means for proving that that binary has the desired privacy properties, potentially requiring the binary to be built using a reproducible process from open source code.

It remains an open question how to partition federated learning functions across secure enclaves, cloud computing resources, and client devices. For example, secure enclaves could execute key functions such as

作为跨筒仓设置中MPC协议的成分[359，224]。[404]中对一些同态加密软件库进行了回顾，沿着了选择库时要考虑的标准/功能的简要说明。

当考虑在FL设置中使用HE时，立即出现关于谁持有该方案的秘密密钥的问题。虽然每个客户端加密数据并将其发送到服务器进行同态计算的想法很有吸引力，但服务器不应该能够解密单个客户端的贡献。克服这个问题的一个简单方法是依赖于一个非合谋的外部方，该外部方持有秘密密钥并解密计算结果。然而，大多数HE方案需要经常更新密钥（例如，由于对选择密文攻击的敏感性[117]）。此外，可信非共谋方的可用性在FL设置中不是标准的。

解决这个问题的另一种方法是依赖于分布式（或阈值）加密方案，其中密钥在各方之间分发。Reyzin等人。[392]和Roth等人。[398]提出了在跨设备设置中计算求和的解决方案。他们的协议使用加法同态方案（分别是ElGamal和基于格的方案的变体）。

可信执行环境可信执行环境（TEE，也称为安全飞地）可以提供将联合学习过程的一部分移动到云中的可信环境中的机会，其代码可以被证明和验证。

TEE可以提供几个关键设施来建立信任，即代码单元已经忠实地和私下地执行[437]：

机密性：代码的执行状态保持秘密，除非代码显式发布消息。

完整性：代码的执行不会受到影响，除非代码显式地接收输入。

·测量/认证：TEE可以向远程方证明正在执行的代码（二进制）及其起始状态，定义机密性和完整性的初始条件。

TEE已经以多种形式实例化，包括英特尔的支持SGX的CPU [241，134]，Arm的TrustZone [28，22]和RISC-V上的Sanctum [135]，每种都有不同的能力系统地提供上述设施。

当前的安全飞地在内存方面受到限制，并且仅提供对CPU资源的访问，即它们不允许在GPU或机器学习处理器上进行处理（Tram`er和Boneh [447]探索如何将联合收割机TEE与GPU结合以进行机器学习推理）。此外，对于TEE（特别是那些在共享微处理器上运行的TEE）来说，完全排除所有类型的侧信道攻击是具有挑战性的[458]。

虽然安全飞地为在其中运行的所有代码提供保护，但在实践中必须解决其他问题。例如，通常有必要将在飞地中运行的代码结构化为数据不经意过程，使得其运行时和内存访问模式不会透露有关其正在计算的数据的信息（参见例如[73]）。此外，测量/证明通常只证明特定的二进制文件正在运行;这取决于系统架构师提供一种方法来证明该二进制文件具有所需的隐私属性，可能需要使用开源代码的可重现过程来构建二进制文件。

如何跨安全飞地、云计算资源和客户端设备划分联合学习功能仍然是一个悬而未决的问题。例如，安全飞地可以执行诸如

secure aggregation or shuffling to limit the server's access to raw client contributions while keeping most of the federated learning logic outside this trusted computing base.

**Secure computation problems of interest** While secure multi-party computation and trusted execution environments offer general solutions to the problem of privately computing any function on distributed private data, many optimizations are possible when focusing on specific functionalities. This is the case for the tasks described next.

*Secure aggregation* Secure aggregation is a functionality for $n$ clients and a server. It enables each client to submit a value (often a vector or tensor in the FL setting), such that the server learns just an aggregate function of the clients' values, typically the sum.

There is a rich literature exploring secure aggregation in both the single-server setting (via additive masking [8, 213, 80, 58, 428], via threshold homomorphic encryption [417, 218, 103], and via generic secure multi-party computation [94]) as well as in the multiple non-colluding servers setting [78, 27, 130]. Secure aggregation can also be approached using trusted execution environments (introduced above), as in [308].

*Secure shuffling* Secure shuffling is a functionality for $n$ clients and a server. It enables each client to submit one or more messages, such that the server learns just an unordered collection (multiset) of the messages from all clients and nothing more. Specifically, the server has no ability to link any message to its sender beyond the information contained in the message itself. Secure shuffling can be considered an instance of Secure Aggregation where the values are multiset-singletons and the aggregation operation is multiset-sum, though it is often the case that very different implementations provide the best performance in the typical operating regimes for secure shuffling and secure aggregation.

Secure shufflers have been studied in the context of secure multi-party computation [107, 288], often under the heading of mix networks. They have also been studied in the context of trusted computing [73]. Mix networks have found large scale deployment in the form of the Tor network [157].

*Private information retrieval* Private information retrieval (PIR) is a functionality for one client and one server. It enables the client to download an entry from a server-hosted database such that the server gains zero information about which entry the client requested.

MPC approaches to PIR break down into two main categories: *computational PIR* (cPIR), in which a single party can execute the entire server side of the protocol [286], and *information theoretic PIR* (itPIR), in which multiple non-colluding parties are required to execute the server side of the protocol [121].

The main roadblocks to the applicability of PIR have been the following: cPIR has high computational cost [423], while the non-colluding parties setting has been difficult to achieve convincingly in industrial scenarios. Recent results on PIR have shown dramatic reductions in the computational cost through the use of lattice-based cryptosystems [12, 363, 13, 23, 198]. The computational cost can be traded for more communication; we refer the reader to Ali et al. [16] to better understand the communication and computation trade-offs offered by cPIR. Additionally, it has been shown how to construct communication-efficient PIR on a single-server by leveraging side information available to the user [251], for example via client local state. Patel et al. [372] presented and implemented a practical hybrid (computational and information theoretic) PIR scheme on a single server assuming client state. Corrigan-Gibbs and Kogan [131] present theoretical constructions for PIR with sublinear *online* time by working in an offline/online model where,

安全聚合或混洗，以限制服务器对原始客户端贡献的访问，同时将大部分联邦学习逻辑保持在此可信计算基础之外。

虽然安全多方计算和可信执行环境为分布式私有数据上的任何函数的私有计算问题提供了通用的解决方案，但当专注于特定功能时，许多优化是可能的。下面描述的任务就是这种情况。

安全聚合安全聚合是针对n个客户端和一个服务器的功能。它允许每个客户端提交一个值（通常是FL设置中的向量或张量），这样服务器就可以学习客户端值的聚合函数，通常是总和。

有丰富的文献探索了单服务器设置（通过添加剂掩蔽[8，213，80，58，428]，通过阈值同态加密[417，218，103]和通过通用安全多方计算[94]）以及多个非共谋服务器设置[78，27，130]中的安全聚合。安全聚合也可以使用可信执行环境（上面介绍的）来实现，如[308]所示。

安全混洗安全混洗是一种用于n个客户端和一个服务器的功能。它允许每个客户端提交一个或多个消息，这样服务器就只从所有客户端学习到一个无序的消息集合（多集）。具体地说，服务器没有能力将任何消息链接到其发送者，而不是消息本身所包含的信息。安全混洗可以被认为是安全聚合的一个实例，其中值是多集单例，聚合操作是多集和，尽管通常情况下，非常不同的实现方式在安全混洗和安全聚合的典型操作机制中提供最佳性能。

安全洗牌器已经在安全多方计算的背景下进行了研究[107，288]，通常在混合网络的标题下。它们也在可信计算的背景下进行了研究[73]。

Mix网络已经以Tor网络的形式大规模部署[157]。

私有信息检索私有信息检索（PIR）是一个客户端和一个服务器的功能。它使客户端能够从服务器托管的数据库下载条目，这样服务器就不会获得有关客户端请求哪个条目的信息。

PIR的MPC方法分为两大类：计算PIR（cPIR），其中一方可以执行协议的整个服务器端[286]，以及信息理论PIR（itPIR），其中需要多个非合谋方来执行协议的服务器端[121]。

PIR适用性的主要障碍如下：cPIR具有高计算成本[423]，而非串通方设置在工业场景中难以令人信服地实现。PIR的最新结果表明，通过使用基于格的密码系统，计算成本显着降低[12，363，13，23，198]。计算成本可以换取更多的通信;我们建议读者参考Ali等人。[16]以更好地理解cPIR提供的通信和计算权衡。此外，已经展示了如何通过利用用户可用的边信息（例如，通过客户端本地状态）在单服务器上构建通信高效的PIR [251]。Patel等人。[372]提出并实现了一个实用的混合（计算和信息理论）PIR方案，假设客户端状态在单个服务器上。 Corrigan-Gibbs和Kogan [131]通过在离线/在线模型中工作，提出了具有次线性在线时间的PIR的理论构造，其中，

during an offline phase, clients fetch information from the server(s) independent on the future query to be performed.

Further work has explored the connection between PIR and secret sharing [479], with recent connections to PIR on coded data [159] and communication efficient PIR [72]. A variant of PIR, called PIR-with-Default, enable clients to retrieve a default value if the index queried is not in the database, and can output additive secret shares of items which can serve as input to any MPC protocol [297]. PIR has also been studied in the context of ON-OFF privacy, in which a client is permitted to switch off their privacy guards in exchange for better utility or performance [355, 494].

### 4.2.2 Privacy-Preserving Disclosures

The state-of-the-art model for quantifying and limiting information disclosure about individuals is *differential privacy* (DP) [167, 164, 165], which aims to introduce a level of uncertainty into the released model sufficient to mask the contribution of any individual user. Differential privacy is quantified by privacy loss parameters $(\varepsilon, \delta)$, where smaller $(\varepsilon, \delta)$ corresponds to increased privacy. More formally, a randomized algorithm $\mathcal{A}$ is $(\varepsilon, \delta)$-differentially private if for all $\mathcal{S} \subseteq \text{Range}(\mathcal{A})$, and for all adjacent datasets $D$ and $D'$:

$$P(\mathcal{A}(D) \in \mathcal{S}) \leq e^{\varepsilon} P(\mathcal{A}(D') \in \mathcal{S}) + \delta. \tag{3}$$

In the context of FL, $D$ and $D'$ correspond to decentralized datasets that are adjacent if $D'$ can be obtained from $D$ by adding or subtracting all the records of a single client (user) [338]. This notion of differential privacy is referred to as user-level differential privacy. It is stronger than the typically used notion of adjacency where $D$ and $D'$ differ by only one record [165], since in general one user may contribute many records (e.g. training examples) to the dataset.

Over the last decade, an extensive set of techniques has been developed for differentially private data analysis, particularly under the assumption of a centralized setting, where the raw data is collected by a trusted party prior to applying perturbations necessary to achieve privacy. In federated learning, typically the orchestrating server would serve as the trusted implementer of the DP mechanism, ensuring only privatized outputs are released to the model engineer or analyst.

However, when possible we often wish to reduce the need for a trusted party. Several approaches for reducing the need for trust in a data curator have been considered in recent years.

**Local differential privacy** Differential privacy can be achieved without requiring trust in a centralized server by having each client apply a differentially private transformation to their data prior to sharing it with the server. That is, we apply Equation (3) to a mechanism $\mathcal{A}$ that processes a single user's local dataset $D$, with the guarantee holding with respect to *any* possible other local dataset $D'$. This model is referred to as the *local model of differential privacy* (LDP) [475, 266]. LDP has been deployed effectively to gather statistics on popular items across large userbases by Google, Apple and Microsoft [177, 154, 155]. It has also been used in federated settings for spam classifier training by Snap [378]. These LDP deployments all involve large numbers of clients and reports, even up to a billion in the case of Snap, which stands in stark contrast to centralized instantiations of DP which can provide high utility from much smaller datasets. Unfortunately, as we will discuss in Section 4.4.2, achieving LDP while maintaining utility can be difficult [266, 455]. Thus, there is a need for a model of differential privacy that interpolates between purely central and purely local DP. This can be achieved through distributed differential privacy, or the hybrid model, as discussed below.

在离线阶段期间，客户端独立于要执行的将来查询从服务器获取信息。

进一步的工作探索了PIR和秘密共享之间的联系[479]，最近与编码数据上的PIR [159]和通信效率PIR [72]的联系。PIR的一个变体，称为PIR-with-Default，使客户端能够在查询的索引不在数据库中时检索默认值，并且可以输出可以作为任何MPC协议输入的项目的附加秘密份额[297]。PIR也在ON-OFF隐私的背景下进行了研究，其中允许客户端关闭其隐私保护以换取更好的实用性或性能[355，494]。

### 4.2.2隐私保护披露

用于量化和限制个人信息披露的最先进模型是差分隐私（DP）[167，164，165]，其目的是在发布的模型中引入一定程度的不确定性，足以掩盖任何个人用户的贡献。差分隐私由隐私损失参数（ε，δ）量化，其中较小的（ε，δ）对应于增加的隐私。更正式地说，随机化算法A是（ε，δ）-差分私有的，如果对于所有S Range（A），以及对于所有相邻数据集D和D：

$$P（A（D）\in S）\leq eP（A（D）\in S）+ \delta。（3）$$

在FL的上下文中，D和D对应于相邻的分散数据集，如果D可以通过添加或减去单个客户端（用户）的所有记录从D获得[338]。这种差异隐私的概念被称为用户级差异隐私。它比通常使用的邻接概念更强，其中D和D仅相差一个记录[165]，因为通常一个用户可以向数据集贡献许多记录（例如训练示例）。

在过去的十年中，已经开发了一套广泛的技术来进行差异化的隐私数据分析，特别是在集中式设置的假设下，其中原始数据在应用实现隐私所需的扰动之前由可信方收集。在联邦学习中，通常编排服务器将作为DP机制的可信实现者，确保只有私有化的输出才会发布给模型工程师或分析师。

然而，在可能的情况下，我们通常希望减少对可信方的需求。近年来，人们考虑了几种减少对数据管理者信任的方法。

本地差异隐私通过让每个客户端在与服务器共享数据之前对其数据应用差异隐私转换，可以在不需要信任集中式服务器的情况下实现差异隐私。也就是说，我们将等式（3）应用于处理单个用户的本地数据集D的机制A，其中关于任何可能的其他本地数据集D保持保证。这个模型被称为局部差分隐私模型（LDP）[475，266]。LDP已被有效地部署，以收集谷歌，苹果和微软[177，154，155]的大型用户群中流行项目的统计数据。它也被Snap用于垃圾邮件分类器训练的联合设置[378]。这些LDP部署都涉及大量的客户端和报告，在Snap的情况下甚至高达10亿，这与DP的集中式实例化形成鲜明对比，后者可以从更小的数据集提供高效用。 不幸的是，正如我们将在4.4.2节中讨论的，在保持效用的同时实现LDP可能是困难的[266，455]。因此，需要一种在纯中央和纯本地DP之间插值的差分隐私模型。这可以通过分布式差异隐私或混合模型来实现，如下所述。

**Distributed differential privacy** In order to recover some of the utility of central DP without having to rely on a trustworthy central server, one can instead use a *distributed differential privacy model* [166, 417, 73, 120]. Under this model, the clients first compute and encode a minimal (application specific) focused report, and then send the encoded reports to a secure computation function, whose output is available to the central server, with the intention that this output already satisfies differential privacy requirements by the time the server is able to inspect it. The encoding is done to help maintain privacy on the clients, and could for example include LDP. The secure computation function can have a variety of incarnations. It could be an MPC protocol, a standard computation done on a TEE, or even a combination of the two. Each of these choices comes with different assumptions and threat models.

It is important to remark that distributed differential privacy and local differential privacy yield different guarantees from several perspectives: while the distributed DP framework can produce more accurate statistics for the same level of differential privacy as LDP, it relies on different setups and typically makes stronger assumptions, such as access to MPC protocols. Below, we outline two possible approaches to distributed differential privacy, relying on secure aggregation and secure shuffling. We stress that there are many other methods that could be used, see for instance [400] for an approach based on exchanging correlated Gaussian noise across secure channels.

*Distributed DP via secure aggregation* One promising tool for achieving distributed DP in FL is secure aggregation, discussed above in Section 4.2.1. Secure aggregation can be used to ensure that the central server obtains the aggregated result, while guaranteeing that intermediate parameters of individual devices and participants are not revealed to the central server. To further ensure the aggregated result does not reveal additional information to the server, we can use local differential privacy (e.g. with moderate $\varepsilon$ level). For example, each device could perturb its own model parameter before the secure aggregation in order to achieve local differential privacy. By designing the noise correctly, we may ensure that the noise in the aggregated result matches the noise that would have otherwise been added centrally by a trusted server (e.g. with a low $\varepsilon$ / high privacy level) [8, 385, 205, 417, 213].

*Distributed DP via secure shuffling* Another distributed differential privacy model is the shuffling model, which was kicked off by the recently introduced Encode-Shuffle-Analyze (ESA) framework [73] (illustrated in Figure 3). In the simplest version of this framework, each client runs an LDP protocol (e.g. with a moderate $\varepsilon$ level) on its data and provides its output to a secure shuffler. The shuffler randomly permutes the reports and sends the collection of shuffled reports (without any identifying information) to the server for final analysis. Intuitively, the interposition of this secure compute function makes it harder for the server to learn anything about the participants and supports a differential privacy analysis (e.g. with a low $\varepsilon$ / high privacy level). In the more general multi-message shuffled framework, each user can possibly send more than one message to the shuffler. The shuffler can either be implemented directly as a trusted entity, independent of the server and devoted solely to shuffling, or via more complex cryptographic primitives as discussed above.

Bittau et al. [73] proposed the Prochlo system as a way to implement the ESA framework. The system takes a holistic approach to privacy that takes into account secure computation aspects (addressed using TEEs), private disclosure aspects (addressed by means of differential privacy), and verifiability aspects (mitigated using secure enclave attestation capabilities).

More generally, shuffling models of differential privacy can use broader classes of local randomizers, and can even select these local randomizers adaptively [178]. This can enable differentially private protocols with far smaller error than what is possible in the local model, while relying on weaker trust assumptions

为了恢复中央DP的一些效用，而不必依赖于可信的中央服务器，可以使用分布式差分隐私模型[166，417，73，120]。在此模型下，客户端首先计算并编码一个（特定于应用的）集中的报告，然后将编码的报告发送到安全计算功能，其输出可用于中央服务器，意图是该输出在服务器能够检查它时已经满足不同的隐私要求。进行编码以帮助维护客户端上的隐私，并且可以例如包括LDP。安全计算函数可以具有多种具体形式。它可以是MPC协议，在TEE上完成的标准计算，甚至是两者的组合。每一种选择都有不同的假设和威胁模型。

值得注意的是，分布式差分隐私和本地差分隐私从几个方面产生不同的保证：虽然分布式DP框架可以为与LDP相同的差分隐私级别产生更准确的统计数据，但它依赖于不同的设置，并且通常会做出更强的假设，例如访问MPC协议。下面，我们概述了两种可能的分布式差异隐私方法，依赖于安全聚合和安全洗牌。我们强调，还有许多其他方法可以使用，例如[400]基于在安全信道上交换相关高斯噪声的方法。

通过安全聚合的分布式DP在FL中实现分布式DP的一个有前途的工具是安全聚合，在上面的第4.2.1节中讨论。安全聚合可以用于确保中央服务器获得聚合结果，同时保证各个设备和参与者的中间参数不泄露给中央服务器。为了进一步确保聚合结果不会向服务器泄露额外的信息，我们可以使用局部差分隐私（例如，具有中等ε水平）。例如，每个设备可以在安全聚合之前扰动其自己的模型参数，以便实现局部差异隐私。通过正确地设计噪声，我们可以确保聚合结果中的噪声与否则将由可信服务器集中添加的噪声相匹配（例如，

低ε /高隐私级别）[8，385，205，417，213]。

另一种分布式差分隐私模型是洗牌模型，它是由最近引入的编码-洗牌-分析（ESA）框架[73]（如图3所示）启动的。在该框架的最简单版本中，每个客户端在其数据上运行LDP协议（例如，具有中等ε水平），并将其输出提供给安全混洗器。混洗器随机排列报告，并将混洗后的报告集合（没有任何标识信息）发送到服务器进行最终分析。直观地说，这种安全计算功能的插入使得服务器更难了解有关参与者的任何信息，并支持差分隐私分析（例如，具有低ε /高隐私级别）。在更一般的多消息混洗框架中，每个用户可能向混洗器发送多于一个消息。 混洗器可以直接实现为可信实体，独立于服务器并且仅致力于混洗，或者经由如上所述的更复杂的密码原语来实现。

Bittau等人[73]提出了Prochlo系统作为实现ESA框架的一种方式。该系统采用整体隐私方法，该方法考虑安全计算方面（使用TEE解决）、隐私公开方面（通过差分隐私解决）和可验证性方面（使用安全飞地证明能力减轻）。

更一般地说，差分隐私的洗牌模型可以使用更广泛的本地随机化器，甚至可以自适应地选择这些本地随机化器[178]。这可以使差分私有协议具有比本地模型中可能的更小的误差，同时依赖于较弱的信任假设

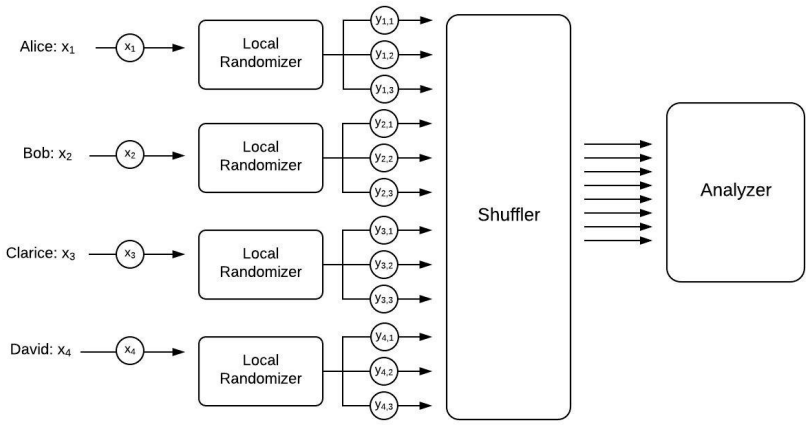



Figure 3: The Encode-Shuffle-Analyze (ESA) framework, illustrated here for 4 players.

than in the central model, e.g., [120, 178, 45, 201, 204, 200, 202, 203, 110].

**Hybrid differential privacy** Another promising approach is hybrid differential privacy [40], which combines multiple trust models by partitioning users based on their trust model preference (e.g. trust or lack of trust in the curator). Prior to the hybrid model, there were two natural choices. The first was to use the least-trusting model, which typically provides the lowest utility, and conservatively apply it uniformly over the entire userbase. The second was to use the most-trusting model, which typically provides the highest utility, but only apply it over the most-trusting users. By allowing multiple models to coexist, hybrid model mechanisms can achieve more utility from a given userbase, compared to purely local or central DP mechanisms. For instance, [40] describes a system in which most users contribute their data in the local model of privacy, and a small fraction of users opt-in to contributing their data in the central DP model. This enables the design of a mechanism which, in some circumstances, outperforms both the conservative local DP mechanism applied across all users as well as the central DP mechanism applied only across the small fraction of opt-in users. Recent work by [57] further demonstrates that a combination of multiple trust models can become part of a promising toolkit for designing and implementing differential privacy. This construction can be directly applied in the federated learning setting; however, the general concept of combining trust models or computational models may also inspire similar but new approaches for federated learning.

### 4.2.3 Verifiability

An important notion that is orthogonal to the above privacy techniques is that of verifiability. Generally speaking, verifiable computation will enable one party to prove to another party that it has executed the desired behavior on its data faithfully, without compromising the potential secrecy of the data. The concept of verifiable computation dates back to Babai et al. [42] and has been studied under various terms in the literature: checking computations [42], certified computation [343], delegating computations [210], as well as verifiable computing [195].

In the context of FL, verifiability can be used for two purposes. First, it would enable the server to prove to the clients that it executed the intended behavior (e.g., aggregating inputs, shuffling of the input messages, or adding noise for differential privacy) faithfully. Second, it would enable the clients to prove to the server that their inputs and behavior follow that of the protocol specification (e.g., the input belongs to a certain

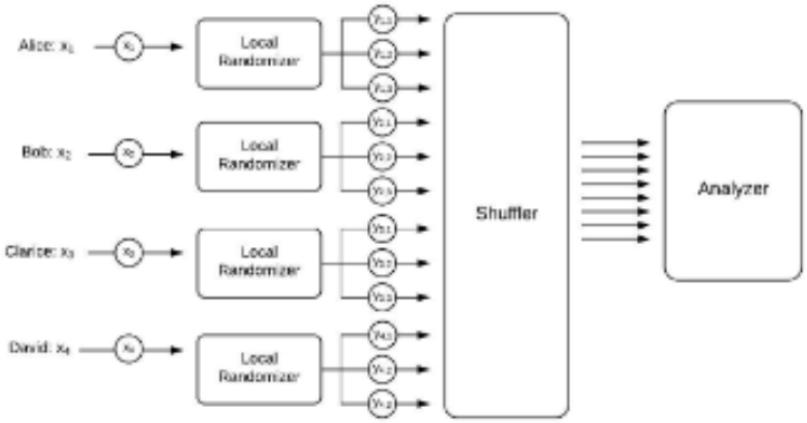



图3：Encode-Shuffle-Analyze（ESA）框架，这里为4个玩家演示。

而不是在中心模型中，例如，[120，178，45，201，204，200，202，203，110]。

混合差分隐私另一种有前途的方法是混合差分隐私[40]，它通过基于用户的信任模型偏好（例如信任或缺乏对管理员的信任）划分用户来组合多个信任模型。在混合模型之前，有两种自然的选择。第一种是使用最不信任的模型，它通常提供最低的效用，并保守地将其统一应用于整个用户群。第二种是使用最信任模型，它通常提供最高的实用性，但仅适用于最信任的用户。通过允许多个模型共存，与纯粹的本地或中央DP机制相比，混合模型机制可以从给定的用户群获得更多的效用。例如，[40]描述了一个系统，其中大多数用户在本地隐私模型中贡献他们的数据，而一小部分用户选择在中央DP模型中贡献他们的数据。 这使得能够设计一种机制，该机制在某些情况下优于跨所有用户应用的保守本地DP机制以及仅跨一小部分选择加入用户应用的中央DP机制。[57]最近的工作进一步表明，多个信任模型的组合可以成为设计和实现差异隐私的有前途的工具包的一部分。这种构造可以直接应用于联邦学习环境;然而，组合信任模型或计算模型的一般概念也可能激发类似但新的联邦学习方法。

4.2.3可核查性

与上述隐私技术正交的一个重要概念是可验证性。一般来说，可验证计算将使一方能够向另一方证明它已经忠实地对其数据执行了所需的行为，而不会损害数据的潜在保密性。可验证计算的概念可以追溯到巴拜等人[42]，并在文献中的各种术语下进行了研究：检查计算[42]，认证计算[343]，委托计算[210]以及可验证计算[195]。

在FL的上下文中，可验证性可用于两个目的。首先，它将使服务器能够向客户端证明它执行了预期的行为（例如，聚集输入、输入消息的混洗、或添加用于不同隐私的噪声）。其次，它将使客户端能够向服务器证明他们的输入和行为遵循协议规范（例如，输入属于某个

range, or the data is a correctly generated ciphertext).

Multiple techniques can be useful to provide verifiability: zero-knowledge proofs (ZKPs), trusted execution environments (TEEs), or remote attestation. Among these ZKPs provide formal cryptographic security guarantees based on mathematical hardness, while others make rely on assumption about the security of trusted hardware.

**Zero-knowledge proofs (ZKPs)** Zero knowledge (ZK) proofs are a cryptographic primitive that enables one party (called the *prover*) to prove statements to another party (called the *verifier*), that depend on secret information known to the prover, called witness, without revealing those secrets to the verifier. The notion of zero-knowledge was introduced in the late 1980's by Goldwasser et al. [209]. It provides a solution for the verifiability question on private data. While there had been a large body of work on ZK construction, the first work that brought ZKPs and verifiable computation for general functionalities in the realm of practicality was the work of Parno et al. [369] which introduces the first optimized construction and implementation for succinct ZK. Nowadays, ZKP protocols can achieve proof sizes of hundred of bytes and verifications of the order of milliseconds regardless of the size of the statement being proved.

A ZKP has three salient properties: *completeness* (if the statement is true and the prover and verifier follow the protocol, the verifier will accept the proof), *soundness* (if the statement is false and the verifier follows the protocol, the verifier will refuse the proof), and *zero-knowledge* (if the statement is true and the prover follows the protocol, the verifier will only learn that the statement is true and will not learn any confidential information from the interaction).

Beyond these common properties, there are different types of zero-knowledge constructions in terms of supported language for the proofs, setup requirements, prover and verifier computational efficiency, interactivity, succinctness, and underlying hardness assumptions. There are many ZK constructions that support specific classes of statements, Schnorr proofs [408] and Sigma protocols [147] are examples of such widely used protocols. While such protocols have numerous uses in specific settings, general ZK systems that can support any functionality provide a much more broadly applicable tool (including in the context of FL), and thus we focus on such constructions for the rest of the discussion.

A major distinguishing feature between different constructions is the need for *trusted* setup. Some ZKPs rely on a common reference string (CRS), which is computed using secrets that should remain hidden in order to guarantee the soundness properties of the proofs. The computation of such a CRS is referred to as a trusted setup. While this requirement is a disadvantage for such systems, the existing ZKP constructions that achieve most succinct proofs and verifier's efficiency require trusted setup.

Another significant property that affects the applicability in different scenarios is whether generating the proof requires interaction between the prover and the verifier, and here we distinguish non-interactive zero-knowledge proofs (NIZKs) that enable the prover to send a single message to the verifier and require no further communication. Often we can convert interactive to non-interactive proofs by making stronger assumptions about ideal functionality of hash functions (i.e., that hash functions behave as random oracles).

Additionally, there are different measurements for efficiency of a ZKP system one must be aware of, such as the length of the proof and the computation complexity of the prover and verifier. The ideal prover's complexity should be linear in the execution time for the evaluated functionality but many existing ZKPs introduce additional (sometimes significant) overhead for the prover. The most efficient verification complexity requires computation at least linear in the size of the inputs for the evaluated functionality, and in the setting of proofs for the work of the FL server this input size will be significant.

Succinct non-interactive zero-knowledge proofs (SNARKs) [71] are a type of ZKP that provides constant

范围，或者数据是正确生成的密文）。

多种技术可以用于提供可验证性：零知识证明（ZKP），可信执行环境（TEE）或远程证明。在这些ZKP中，提供了基于数学硬度的形式化密码安全保证，而其他ZKP则依赖于可信硬件的安全性假设。

零知识证明（ZKP）零知识证明（ZK）是一种加密原语，它使一方（称为证明者）能够向另一方（称为验证者）证明声明，这取决于证明者已知的秘密信息，称为证人，而不会向验证者透露这些秘密。零知识的概念是由Goldwasser等人在20世纪80年代末提出的。它为私有数据的可验证性问题提供了解决方案。虽然在ZK构造方面有大量的工作，但第一个将ZKP和可验证计算带入实用领域的工作是Parno等人的工作。[369]它介绍了第一个优化的构造和简洁ZK的实现。如今，ZKP协议可以实现数百字节的证明大小和毫秒级的验证，而不管被证明的语句的大小。

ZKP有三个显著的特性：完整性（如果陈述为真，并且证明者和验证者遵循协议，则验证者将接受证明），可靠性（如果声明是假的，验证者遵循协议，验证者将拒绝证明），和零知识（如果陈述是真的并且证明者遵循协议，验证者将仅获知该陈述是真实的，而不会从交互中获知任何机密信息）。

除了这些共同的属性，还有不同类型的零知识结构的支持语言的证明，设置要求，证明者和验证者的计算效率，交互性，简洁性和潜在的硬度假设。有许多ZK结构支持特定类别的语句，Schnorr证明[408]和Sigma协议[147]是这些广泛使用的协议的例子。虽然这样的协议在特定环境中有许多用途，但可以支持任何功能的通用ZK系统提供了一个更广泛适用的工具（包括在FL的上下文中），因此我们将重点放在这样的结构上进行讨论。

不同构造之间的一个主要区别特征是需要可信设置。一些ZKP依赖于公共参考字符串（CRS），该字符串是使用应该保持隐藏的秘密来计算的，以保证证明的可靠性。这种CRS的计算被称为可信设置。虽然这一要求是这样的系统的一个缺点，现有的ZKP结构，实现最简洁的证明和验证的效率需要可信的设置。

另一个重要的属性，影响在不同的情况下的适用性是生成证明是否需要证明者和验证者之间的交互，在这里，我们区分非交互式零知识证明（NIZK），使证明者发送一个单一的消息到验证者，不需要进一步的通信。通常，我们可以通过对散列函数的理想功能进行更强的假设（即，散列函数表现为随机预言机）。

此外，ZKP系统的效率有不同的衡量标准，例如证明的长度和证明者和验证者的计算复杂度。理想的证明器的复杂度应该是线性的，在执行时间的评估功能，但许多现有的ZKP引入额外的（有时显着）的开销证明器。最有效的验证复杂性要求计算在用于评估的功能的输入的大小方面至少是线性的，并且在用于FL服务器的工作的证明的设置中，该输入大小将是显著的。

简洁的非交互式零知识证明（SNARKs）[71]是一种ZKP，它提供恒定的

proof size and verification that depends only on the input size, linearly. These attractive efficiency properties do come at the price of stronger assumptions, which is mostly inherent, and trusted setup in all existing scheme. Most existing SNARK constructions leverage quadratic arithmetic programs [196, 369, 136] and are now available in open-source libraries, such as libsnark [307], and deployed in cryptocurrencies, such as Zcash [62]. Note that SNARK systems usually require overhead on the part of the prover; in particular, the prover computation needs to be superlinear in the size of the circuit for the statement being proven. Recently, Xie et al. [489] presented Libra, a ZKP system that achieves linear prover complexity but with increased proof size and verification time.

If we relax the requirements for succinctness or non-interactiveness for the construction, there is a large body of constructions that achieve a wide range of efficiency trade-offs, avoid the trusted setup requirement and use more standard cryptographic assumptions [92, 464, 20, 63].

In the recent years, an increasing numbers of practical applications have been using non-interactive zero-knowledge proofs, primarily motivated by blockchains. Using interactive ZKP systems and NIZKs efficiently in the context of FL remains a challenging open question. In such a setting, NIZKs may enable to prove to the server properties about the client's inputs. In the setting where the verifier is the client, it will be challenging to create a trustworthy statement to verify as it involves input from other clients. Of interest in this setting, recent work enables to handle the case where the multiple verifiers have shares of the statement [83].

**Trusted execution environment and remote attestation** We discussed TEEs in Section 4.2.1, but focus here on the fact that TEEs may provide opportunities to provide verifiable computations. Indeed, TEEs enable to attest and verify the code (binary) running in its environment. In particular, when the verifier knows (or can reproduce) which binary should run in the secure enclaves, TEEs will be able to provide a notion of *integrity* (the code execution cannot be affected, except by the inputs), and an *attestation* (the TEE can prove that a specific binary is executing and what is starting state was) [437, 451]. More generally, remote attestation allows a verifier to securely measure the internal state of a remote hardware platform, and can be used to establish a static or dynamic root of trust. While TEEs enable hardware-based remote attestations, both software-based remote attestions [411] and hybrid remote attestation designs [172, 274] were proposed in the literature and enable to trade off hardware requirements for verifiability.

In a federated learning setting, TEEs and remote attestations may be particularly helpful for clients to be able to efficiently verify key functions running on the server. For example, secure aggregation or shuffling could run in TEEs and would provide differential privacy guarantees on their outputs. Therefore, the post-processing logic subsequently applied by the server on the differentially private data could run on the server and remain oblivious to the clients. Note that such a system design requires the clients to know and trust the exact code (binary) for the key functions to be applied in the enclaves. Additionally, remote attestations may enable a server to attest specific requirements from the clients involved in the FL computation, such as absence of leaks, immutability, and uninterruptability (we defer to [188] for an exhaustive list of minimal requirements for remote attestation).

## 4.3 Protections Against External Malicious Actors

In this section, we assume the existence of a trusted server and discuss various challenges and open problems towards achieving rigorous privacy guarantees against external malicious actors (e.g. adversarial clients, adversarial analysts, adversarial devices that consume the learned model, or any combination thereof).

As discussed in Table 7, malicious clients can inspect all messages received from the server (including

证明的大小和验证，只依赖于输入的大小，线性。这些有吸引力的效率属性确实是以更强的假设为代价的，这些假设大多是固有的，并且在所有现有方案中都是可信的。大多数现有的SNARK构造都利用了二次算术程序[196，369，136]，现在可以在开源库中使用，如libsnark [307]，并部署在加密货币中，如Zcash [62]。请注意，SNARK系统通常需要证明器部分的开销;特别是，证明器计算需要在被证明的语句的电路大小上是超线性的。最近，Xie等人[489]提出了Libra，这是一个ZKP系统，它实现了线性证明器的复杂性，但增加了证明大小和验证时间。

如果我们放松对构造的简洁性或非交互性的要求，则有大量的构造实现了广泛的效率权衡，避免了可信设置要求并使用更多标准的密码学假设[92，464，20，63]。

近年来，越来越多的实际应用一直在使用非交互式零知识证明，主要是由区块链驱动的。在FL的背景下有效地使用交互式ZKP系统和NIZK仍然是一个具有挑战性的开放问题。在这样的设置中，NIZK可以使得能够向服务器证明关于客户端的输入的属性。在验证者是客户端的设置中，创建一个可信的语句来验证是具有挑战性的，因为它涉及到来自其他客户端的输入。在这种设置中，最近的工作使得能够处理多个验证者共享语句的情况[83]。

可信执行环境和远程证明我们在4.2.1节中讨论了TEE，但这里重点关注TEE可能提供可验证计算的机会。事实上，TEE能够证明和验证在其环境中运行的代码（二进制）。特别地，当验证器知道（或可以再现）哪个二进制文件应该在安全飞地中运行时，TEE将能够提供完整性的概念（代码执行不会受到影响，除了输入）和证明（TEE可以证明特定的二进制文件正在执行以及开始状态是什么）[437，451]。更一般地，远程证明允许验证者安全地测量远程硬件平台的内部状态，并且可以用于建立静态或动态信任根。 虽然TEE支持基于硬件的远程证明，但文献中提出了基于软件的远程证明[411]和混合远程证明设计[172，274]，并且能够权衡硬件要求以实现可验证性。

在联合学习设置中，TEE和远程证明可能特别有助于客户端能够有效地验证服务器上运行的关键功能。例如，安全聚合或洗牌可以在TEE中运行，并在其输出上提供差异隐私保证。因此，随后由服务器应用于差异隐私数据的后处理逻辑可以在服务器上运行，并且保持对客户端不经意。注意，这样的系统设计要求客户端知道并信任要在飞地中应用的关键功能的确切代码（二进制）。此外，远程证明可以使服务器能够证明FL计算中涉及的客户端的特定要求，例如不存在泄漏，不变性和不可中断性（我们遵从[188]的远程证明最低要求的详尽列表）。

## 4.3针对外部恶意行为者的保护

在本节中，我们假设存在可信服务器，并讨论实现严格的隐私保证以对抗外部恶意行为者（例如敌对客户端，敌对分析师，消费学习模型的敌对设备或其任何组合）的各种挑战和开放问题。

如表7所示，恶意客户端可以检查从服务器接收的所有消息（包括

the model iterates) in the rounds they participate in, malicious analysts can inspect sequences of model iterates from multiple training runs with different hyperparameters, and in cross-device FL, malicious devices can have either white-box or black-box access to the final model. Therefore, to give rigorous protections against external adversaries, it is important to first consider what can be learned from the intermediate iterates and final model.

### 4.3.1 Auditing the Iterates and Final Model

To better understand what can be learned from the intermediate iterates or final model, we propose quantifying federated learning models' susceptibility towards specific attacks. This is a particularly interesting problem in the federated learning context. On the one hand, adversaries receive direct access to the model from the server, which widens the attack surface. On the other hand, the server determines which specific stages of the training process the adversary will receive access to the model, and additionally controls the adversary's influence over the model at each of the stages.

For classic (non-federated) models of computation, understanding a model's susceptibility to attacks is an active and challenging research area [189, 418, 99, 341, 100]. The most common method of quantifying a model's susceptibility to an attack is to simulate the attack on the model using a proxy (auditing) dataset similar to the dataset expected in practice. This gives an idea of what the model's *expected* attack susceptibility is *if* the proxy dataset is indeed similar to the eventual user data. A safer method would be to determine a worst-case upper-bound on the model's attack susceptibility. This can be approached theoretically as in [496], although this often yields loose, vacuous bounds for realistic models. Empirical approaches may be able to provide tighter bounds, but for many types of attacks and models, this endeavour may be intractable. An interesting emerging area of research in this space examines the theoretic conditions (on the audited model and attacks) under which an unsuccessful attempt to identify privacy violations by a simulated attack implies that no stronger attacks can succeed at such a task [153]. However, this area is still nascent and more work needs to be done to better understand the fundamental requirements under which auditing (via simulated attacks) is sufficient.

The federated learning framework provides a unique setting not only for attacks, but also for attack quantification and defense. Specifically, due to the server's control over when each user can access and influence the model during the training process, it may be possible to design new tractable methods for quantifying a model's average-case or worst-case attack susceptibility. Such methods would enable the development of new adaptive defenses, which can be applied on-the-fly to preempt significant adversarial influence while maximizing utility.

### 4.3.2 Training with Central Differential Privacy

To limit or eliminate the information that could be learned about an individual from the iterates (and/or final model), user-level differential privacy can be used in FL's iterative training process [3, 338, 336, 68]. With this technique, the server clips the $\ell_2$ norm of individual updates, aggregates the clipped updates, and then adds Gaussian noise to the aggregate. This ensures that the iterates do not overfit to any individual user's update. To track the overall privacy budget across rounds, advanced composition theorems [168, 254] or the analytical moments accountant method developed in [3, 346, 348, 474] can be used. The moments accountant method works particularly well with the uniformly subsampled Gaussian mechanism. For moderate privacy budgets and in the absence of a sufficiently large dataset [384], the noise introduced by this process can lead to a large decrease in model accuracy. Prior work has explored a number of avenues to mitigate this trade-off between privacy and accuracy, including collecting more private data [338], designing

模型迭代），恶意分析人员可以检查来自具有不同超参数的多个训练运行的模型迭代序列，并且在跨设备FL中，恶意设备可以对最终模型进行白盒或黑盒访问。因此，为了给予严格的保护以抵御外部对手，首先考虑可以从中间迭代和最终模型中学到什么是很重要的。

#### 4.3.1审核迭代和最终模型

为了更好地理解从中间迭代或最终模型中可以学到什么，我们建议量化联邦学习模型对特定攻击的敏感性。这在联邦学习环境中是一个特别有趣的问题。一方面，攻击者可以从服务器直接访问模型，这扩大了攻击面。另一方面，服务器确定对手将在训练过程的哪些特定阶段接收对模型的访问，并且另外控制对手在每个阶段对模型的影响。

对于经典（非联邦）计算模型，了解模型对攻击的敏感性是一个活跃且具有挑战性的研究领域[189，418，99，341，100]。量化模型对攻击的敏感性的最常见方法是使用类似于实践中预期的数据集的代理（审计）数据集来模拟对模型的攻击。如果代理数据集确实类似于最终的用户数据，这就给出了模型的预期攻击敏感性的概念。更安全的方法是确定模型受攻击敏感性的最坏情况上限。这可以从理论上接近[496]，尽管这通常会为现实模型产生松散，空洞的边界。经验方法可能能够提供更严格的界限，但对于许多类型的攻击和模型，这种努力可能是棘手的。 在这个领域，一个有趣的新兴研究领域研究了理论条件（关于审计模型和攻击），在这种条件下，通过模拟攻击识别隐私侵犯的尝试不成功意味着没有更强大的攻击可以成功完成这样的任务。然而，这一领域仍处于起步阶段，需要做更多的工作来更好地理解审计（通过模拟攻击）的基本要求。

联邦学习框架不仅为攻击提供了独特的设置，而且还为攻击量化和防御提供了独特的设置。具体来说，由于服务器控制每个用户在训练过程中何时可以访问和影响模型，因此可以设计新的易于处理的方法来量化模型的平均情况或最坏情况的攻击敏感性。这种方法将使新的自适应防御系统的开发成为可能，这种防御系统可以在运行中应用，在最大限度地提高效用的同时先发制人地发挥重大的对抗影响。

#### 4.3.2使用中央差分隐私进行训练

为了限制或消除可以从迭代（和/或最终模型）中了解到的关于个体的信息，可以在FL的迭代训练过程中使用用户级差分隐私[3，338，336，68]。利用这种技术，服务器裁剪各个更新的范数，聚合裁剪的更新，然后向聚合添加高斯噪声。这可以确保迭代不会过度拟合任何单个用户的更新。为了跟踪各轮的整体隐私预算，可以使用高级合成定理[168，254]或[3，346，348，474]中开发的分析矩会计方法。矩计数器方法特别适用于均匀子采样高斯机制。对于适度的隐私预算和缺乏足够大的数据集[384]，该过程引入的噪声可能导致模型准确性大幅下降。 先前的工作已经探索了许多途径来减轻隐私和准确性之间的这种权衡，包括收集更多的私人数据[338]，设计

privacy-friendly model architectures [367], or leveraging priors on the private data domain [449].

In cross-device FL, the number of training examples can vary drastically from one device to the other. Hence, similar to recent works on user-level DP in the central model [21], figuring out how to adaptively bound the contributions of users and clip the model parameters remains an interesting research direction [446, 377]. More broadly, unlike record-level DP where fundamental trade-offs between accuracy and privacy are well understood for a variety of canonical learning and estimation tasks, user-level DP is fundamentally less understood (especially when the number of contributions varies wildly across users and is not tightly bounded *a priori*). Thus, more work needs to be done to better understand the fundamental trade-offs in this emerging setting of DP. Recently, [320] made progress on this front by characterizing the trade-offs between accuracy and privacy for learning discrete distributions under user-level DP.

In addition to the above, it is important to draw a distinction between malicious clients that may be able to see (some of) the intermediate iterates during training and malicious analysts (or deployments) that can only see the final model. Even though central DP provides protections against both threat models, a careful theoretical analysis can reveal that for a specific implementation of the above Gaussian mechanism (or any other differentially private mechanism), we may get different privacy parameters for these two threat models. Naturally, we should get stronger differential privacy guarantees with respect to malicious analysts than we do with respect to malicious clients (because malicious clients may have access to far more information than malicious analysts). This "privacy amplification via iteration" setting has been recently studied by Feldman et al. [185] for convex optimization problems. However, it is unclear whether or not the results in [185] can be carried over to the non-convex setting.

**Privacy amplification for non-uniform device sampling procedures** Providing formal $(\varepsilon, \delta)$ guarantees in the context of cross-device FL system can be particularly challenging because: (a) the set of all eligible users (i.e. underlying database) is dynamic and not known in advance, and (b) users participating in federate computations may drop out at any point in the protocol. It is therefore important to investigate and design protocols that: (1) are robust to nature's choice (user availability and dropout), (2) are self-accounting, in that the server can compute a tight $(\varepsilon, \delta)$ guarantee using only information available via the protocol, (3) rely on local participation decision (i.e. do not assume that the server knows which users are online and has the ability to sample from them), and (4) achieve good privacy-utility trade-offs. While recent works [47, 257] suggest that these constraints can be simultaneously achieved, building an end-to-end protocol that works in production FL systems is still an important open problem.

**Sources of randomness (adapted from [336])** Most computational devices have access only to few sources of entropy and they tend to be very low rate (hardware interrupts, on-board sensors). It is standard—and theoretically well justified—to use the entropy to seed a cryptographically secure pseudo-random number generator (PRNG) and use the PRNG's output as needed. Robust and efficient PRNGs based on standard cryptographic primitives exist that have output rate of gigabytes per second on modern CPUs and require a seed as short as 128 bits [401].

The output distribution of a randomized algorithm $\mathcal{A}$ with access to a PRNG is indistinguishable from the output distribution of $\mathcal{A}$ with access to a true source of entropy *as long as the distinguisher is computationally bounded*. Compare it with the guarantee of differential privacy which holds against any adversary, no matter how powerful. As such, virtually all implementations of differential privacy satisfy only (variants of) computational differential privacy introduced by [347]. On the positive side, a computationally-bounded adversary cannot tell the difference, which allows us to avoid being overly pedantic about this point.

A training procedure may have multiple sources of non-determinism (e.g., dropout layers or an input of a

设计隐私友好的模型架构[367]，或利用私有数据域的先验知识[449]。

在跨设备FL中，训练示例的数量可以从一个设备到另一个设备急剧变化。因此，与最近在中心模型[21]中对用户级DP的研究类似，弄清楚如何自适应地约束用户的贡献并裁剪模型参数仍然是一个有趣的研究方向[446，377]。更广泛地说，与记录级DP不同，在记录级DP中，对于各种规范学习和估计任务，准确性和隐私之间的基本权衡是很好理解的，用户级DP从根本上不太理解（特别是当用户之间的贡献数量变化很大并且没有严格限制先验时）。因此，需要做更多的工作，以更好地了解在这种新兴的DP设置的基本权衡。最近，[320]通过描述在用户级DP下学习离散分布的准确性和隐私之间的权衡，在这方面取得了进展。

除了上述内容之外，重要的是要区分恶意客户端和恶意分析师（或部署），恶意客户端可能能够在训练期间看到（一些）中间迭代，恶意分析师只能看到最终模型。尽管中心DP提供了针对两种威胁模型的保护，但仔细的理论分析可以揭示，对于上述高斯机制（或任何其他差异隐私机制）的特定实现，我们可能会为这两种威胁模型获得不同的隐私参数。当然，我们应该得到更强的差分隐私保证恶意分析师比我们对恶意客户端（因为恶意客户端可以访问更多的信息比恶意分析师）。最近，Feldman等人[185]针对凸优化问题研究了这种“通过迭代的隐私放大”设置。 然而，目前还不清楚[185]中的结果是否可以转移到非凸设置。

在跨设备FL系统的上下文中提供形式（ε，δ）保证可能特别具有挑战性，因为：（a）所有合格用户的集合（即底层数据库）是动态的，并且事先不知道，以及（b）参与联邦计算的用户可能在协议中的任何点退出。因此，重要的是要研究和设计协议：（1）对自然的选择具有鲁棒性（用户可用性和退出），（2）是自核算的，因为服务器可以仅使用经由协议可用的信息来计算紧（ε，δ）保证，（3）依靠地方参与决策（即，不假设服务器知道哪些用户在线并且具有从他们中采样的能力），以及（4）实现良好的隐私-效用权衡。 虽然最近的工作[47，257]表明这些约束可以同时实现，但构建在生产FL系统中工作的端到端协议仍然是一个重要的开放问题。

随机性的来源（改编自[336]）大多数计算设备只能访问很少的熵源，并且它们往往是非常低的速率（硬件中断，板载传感器）。使用熵来播种密码安全的伪随机数生成器（PRNG）并根据需要使用PRNG的输出是标准的，并且在理论上得到了很好的证明。存在基于标准密码原语的鲁棒且高效的PRNG，其在现代CPU上具有每秒千兆字节的输出速率，并且需要短至128位的种子[401]。

随机化算法A的输出分布与访问PRNG的输出分布是不可区分的，只要熵是计算上有界的，就可以访问真正的熵源。将其与差分隐私的保证进行比较，后者可以对抗任何对手，无论多么强大。因此，几乎所有差分隐私的实现都只满足[347]引入的计算差分隐私（变体）。从积极的一面来看，一个计算有限的对手无法区分两者的区别，这使得我们可以避免在这一点上过于迂腐。

训练过程可以具有多个非确定性来源（例如，dropout图层或

generative model) but only those that are reflected in the privacy ledger must come from a cryptographically secure PRNG. In particular, the device sampling procedure and the additive Gaussian noise must be drawn from a cryptographically secure PRNG for the trained model to satisfy computational differential privacy.

**Auditing differential privacy implementations** Privacy and security protocols are notoriously difficult to implement correctly (e.g., [345, 217] for differential privacy). What techniques can be used for testing FL-implementations for correctness? Since the techniques will often be deployed by organizations who may opt not to open-source code, what are the possibilities for black-box testing? Some works [156, 315, 247] begin to explore this area in the context of differential privacy, but many open questions remain.

### 4.3.3 Concealing the Iterates

In typical federated learning systems, the model iterates (i.e., the newly updated versions of the model after each round of training) are assumed to be visible to multiple actors in the system, including the server and the clients that are chosen to participate in each round. However, it may be possible to use tools from Section 4.2 to keep the iterates concealed from these actors.

To conceal the iterates from the clients, each client could run their local portion of federated learning inside a TEE providing confidentiality features (see Section 4.2.1). The server would validate that the expected federated learning code is running in the TEE (relying on the TEE's attestation and integrity features), then transmit an encrypted model iterate to the device such that it can only be decrypted inside the TEE. Finally the model updates would be encrypted inside the TEE before being returned to the server, using keys only known inside the enclave and on the server. Unfortunately, TEEs may not be generally available across clients, especially when those clients are end-user devices such as smartphones. Moreover, even when TEEs are present, they may not be sufficiently powerful to support training computations, which would have to happen inside the TEE in order to protect the model iterate, and may be computationally expensive and/or require significant amounts of RAM – though TEE capabilities are likely to improve over time, and techniques such as those presented in [447] may be able to reduce the requirements on the TEE by exporting portions of the computation outside the TEE while maintaining the attestation, integrity, and confidentiality needs of the computation as a whole.

Similar protections can be achieved under the MPC model [351, 10]. For example, the server could encrypt the iterate's model parameters under a homomorphic encryption scheme before sending it to the client, using keys known only to the server. The client could then compute the encrypted model update using the homomorphic properties of the cryptosystem, without needing to decrypt the model parameters. The encrypted model update could then be returned to the server for aggregation. A key challenge here will be to force aggregation on the server before decryption, as otherwise the server may be able to learn a client's model update. Another challenging open problem here is improving performance, as even state-of-the-art systems can require quite significant computational resources to complete a single round of training in a deep neural network. Progress here could be made both by algorithmic advances as well as through the development of more efficient hardware accelerators for MPC [393].

Additional challenges arise if the model iterates should also be concealed from the server. Under the TEE model, the server portion of federated learning could run inside a TEE, with all parties (i.e., clients and analyst) verifying that the server TEE will only release the final model after the appropriate training criteria have been met. Under the MPC model, an encryption key could protect the model iterates, with the key held by the analyst, distributed in shares among the clients, or held by a trusted third party; in this setup, the key holder(s) would be required to engage in the decryption of the model parameters, and could thereby ensure

生成模型），但只有那些反映在隐私分类账中的数据必须来自密码安全的PRNG。特别是，设备采样过程和加性高斯噪声必须从加密安全的PRNG中提取，以使训练模型满足计算差分隐私。

众所周知，隐私和安全协议很难正确实现（例如，[345，217]用于差分隐私）。什么技术可以用来测试FL实现的正确性？由于这些技术通常由可能选择不开放源代码的组织部署，那么黑盒测试的可能性是什么？一些作品[156，315，247]开始在差异隐私的背景下探索这一领域，但仍存在许多悬而未决的问题。

### 4.3.3隐藏迭代

在典型的联邦学习系统中，模型迭代（即，在每轮训练之后模型的新更新版本）被假定为对系统中的多个参与者可见，包括被选择参与每轮的服务器和客户端。但是，可以使用4.2节中的工具来隐藏这些参与者的迭代。

为了对客户端隐藏迭代，每个客户端可以在提供机密性功能的TEE内运行其本地部分的联邦学习（参见第4.2.1节）。服务器将验证预期的联合学习代码正在TEE中运行（依赖于TEE的证明和完整性特征），然后将加密的模型加密器发送到设备，使得它只能在TEE内被解密。最后，模型更新将在返回到服务器之前在TEE内部加密，使用仅在安全区内部和服务器上已知的密钥。不幸的是，TEE通常可能无法跨客户端使用，尤其是当这些客户端是诸如智能手机之类的终端用户设备时。 此外，即使当TEE存在时，它们也可能不足够强大以支持训练计算，训练计算将必须在TEE内部发生以便保护模型可重构性，并且可能在计算上是昂贵的和/或需要大量的RAM -尽管TEE能力可能随着时间的推移而改进，并且诸如在[447]中提出的那些技术可以能够通过将计算的部分导出到TEE之外来减少对TEE的要求，同时整体上保持计算的证明、完整性和机密性需求。

在MPC模型下可以实现类似的保护[351，10]。例如，服务器可以在发送给客户端之前，使用只有服务器知道的密钥，在同态加密方案下加密模型的模型参数。然后，客户端可以使用密码系统的同态属性来计算加密的模型更新，而不需要解密模型参数。然后，加密的模型更新可以返回到服务器进行聚合。这里的一个关键挑战是在解密之前强制服务器上的聚合，否则服务器可能能够学习客户端的模型更新。另一个具有挑战性的开放问题是提高性能，因为即使是最先进的系统也可能需要相当多的计算资源来完成深度神经网络中的单轮训练。这里的进展可以通过算法的进步以及通过开发更有效的MPC硬件加速器来实现[393]。

如果模型迭代也应该对服务器隐藏，则会出现额外的挑战。在TEE模型下，联合学习的服务器部分可以在TEE内部运行，所有各方（即，客户端和分析师）验证服务器TEE将仅在已经满足适当的训练标准之后发布最终模型。在MPC模型下，加密密钥可以保护模型迭代，其中密钥由分析师持有，在客户端之间共享分布，或者由可信的第三方持有;在这种设置中，密钥保持器将被要求参与模型参数的解密，从而可以确保

that this process happens only once.

### 4.3.4 Repeated Analyses over Evolving Data

For many applications of federated learning, the analyst wishes to analyze data that arrive in a streaming fashion, and must also provide dynamically-updated learned models that are (1) correct on the data seen thus far, and (2) accurately predict future data arrivals. In the absence of privacy concerns, the analyst could simply re-train the learned model once new data arrive, to ensure maximum accuracy at all times. However, since privacy guarantees degrade as additional information is published about the same data [167, 168], these updates must be less frequent to still preserve both privacy and accuracy of the overall analysis.

Recent advances in differential privacy for dynamic databases and time series data [143, 142, 97] have all assumed the existence of a trusted curator who can see raw data as they arrive online, and publish dynamically updated statistics. An open question is how these algorithmic techniques can be extended to the federated setting, to enable private federated learning on time series data or other dynamically evolving databases.

Specific open questions include:

- How should an analyst privately update an FL model in the presence of new data? Alternatively, how well would a model that was learned privately with FL on a dataset $D$ extend to a dataset $D'$ that was guaranteed to be similar to $D$ in a given closeness measure? Since FL already occurs on samples that arrive online and does not overfit to the data it sees, it is likely that such a model would still continue to perform well on a new database $D'$. This is also related to questions of robustness that are explored in Section 5.
- One way around the issue of privacy composition is by producing synthetic data [165, 5], which can then be used indefinitely without incurring additional privacy loss. This follows from the post-processing guarantees of differential privacy [167]. Augenstein et al. [31] explore the generation of synthetic data in a federated fashion. In the dynamic data setting, synthetic data can be used repeatedly until it has become "outdated" with respect to new data, and must be updated. Even after generating data in a federated fashion, it must also be updated privately and federatedly.
- Can the specific approaches in prior work on differential privacy for dynamic databases [142] or privately detecting changes in time series data [143, 97] be extended to the federated setting?
- How can time series data be queried in a federated model in the first place? By design, the same users are not regularly queried multiple times for updated data points, so it is difficult to collect true within-subject estimates of an individuals' data evolution over time. Common tools for statistical sampling of time series data may be brought to bear here, but must be used in conjunction with tools for privacy and tools for federation. Other approaches include reformulating the queries such that each within-subject subquery can be answered entirely on device.

### 4.3.5 Preventing Model Theft and Misuse

In some cases, the actor or organization developing an ML model may be motivated to restrict the ability to inspect, misuse or steal the model. For example, restricting access to the model's parameters may make it more difficult for an adversary to search for vulnerabilities, such as inputs that produce unanticipated model outputs.

从而可以确保该过程仅发生一次。

### 4.3.4不断变化的数据的重复分析

对于联邦学习的许多应用程序，分析师希望分析以流式方式到达的数据，并且还必须提供动态更新的学习模型，这些模型（1）对迄今为止看到的数据是正确的，（2）准确预测未来的数据到达。在没有隐私问题的情况下，分析师可以在新数据到达时简单地重新训练学习模型，以确保始终保持最大的准确性。然而，由于隐私保证会随着关于相同数据的附加信息的发布而降低[167，168]，因此这些更新必须不那么频繁，以保持整体分析的隐私和准确性。

动态数据库和时间序列数据的差异隐私的最新进展[143，142，97]都假设存在一个可信的管理者，他可以在原始数据到达在线时看到它们，并发布动态更新的统计数据。一个悬而未决的问题是如何将这些算法技术扩展到联邦设置，以实现对时间序列数据或其他动态演化数据库的私有联邦学习。

具体的未决问题包括：

·分析师应该如何在新数据存在的情况下私下更新FL模型？或者，一个在数据集D上用FL私下学习的模型，在多大程度上可以扩展到一个在给定的贴近度度量中保证与D相似的数据集D？由于FL已经发生在在线到达的样本上，并且不会过拟合它所看到的数据，因此这样的模型可能仍然会在新的数据库D上继续表现良好。这也与第5节中探讨的稳健性问题有关。

·解决隐私合成问题的一种方法是生成合成数据[165，5]，然后可以无限期地使用这些数据，而不会导致额外的隐私损失。这来自于差分隐私的后处理保证[167]。Augenstein等人。[31]探索了以联邦方式生成合成数据。在动态数据设置中，合成数据可以重复使用，直到它相对于新数据变得“过时”，并且必须更新。即使在以联邦方式生成数据之后，也必须私下和联邦地更新数据。

·之前关于动态数据库的差异隐私[142]或私下检测时间序列数据变化[143，97]的工作中的具体方法是否可以扩展到联邦设置？

·首先如何在联邦模型中查询时间序列数据？根据设计，相同的用户不会定期多次查询更新的数据点，因此很难收集个人数据随时间演变的真实受试者内估计值。用于时间序列数据的统计采样的常用工具可以在这里使用，但必须与隐私工具和联邦工具结合使用。其他方法包括重新制定查询，使得每个受试者内子查询可以完全在设备上回答。

### 4.3.5防止模型盗窃和滥用

在某些情况下，开发ML模型的参与者或组织可能会限制检查，滥用或窃取模型的能力。例如，限制对模型参数的访问可能会使攻击者更难以搜索漏洞，例如产生意外模型输出的输入。

Protecting a deployed model during inference is closely related to the challenge of concealing the model iterates from clients during training, as discussed in Section 4.3.3. Again, both TEEs and MPC may be used. Under the TEE model, the model parameters are only accessible to a TEE on the device, as in Section 4.3.3; the primary difference being that the desired calculation is now inference instead of training.

It is harder to adapt MPC strategies to this use case without forgoing the advantages offered by on-device inference: if the user data, model parameters, and inference results are all intended to be on-device, then it is unclear what additional party is participating in the multi-party computation. For example, naïvely attempting to use homomorphic encryption would require the decryption keys to be on device where the inferences are to be used, thereby undermining the value of the encryption in the first place. Solutions where the analyst is required to participate (e.g. holding either the encryption keys or the model parameters themselves) imply additional inference latency, bandwidth costs, and connectivity requirements for the end user (e.g. the inferences would no longer be available for a device in airplane mode).

It is crucial to note that even if the model parameters themselves are successfully hidden, research has shown that in many cases they can be reconstructed by an adversary who only has access to an inference/prediction API based on those parameters [450]. It is an open question what additional protections would need to be put into place to protect from these kinds of issues in the context of a model residing on millions or billions of end user devices.

## 4.4 Protections Against an Adversarial Server

In the previous section, we assumed the existence of a trusted server that can orchestrate the training process. In this section we discuss the more desirable scenario of protecting against an adversarial server. In particular, we start by investigating the challenges of this setting and existing works, and then move on to describing the open problems and how the techniques discussed in Section 4.2 can be used to address these challenges.

### 4.4.1 Challenges: Communication Channels, Sybil Attacks, and Selection

In the cross-device FL setting, we have a server with significant computational resources and a large number of clients that (i) can only communicate with the server (as in a star network topology), and (ii) may be limited in connectivity and bandwidth. This poses very concrete requirements when enforcing a given trust model. In particular, clients do not have a clear way of establishing secure channels among themselves independent of the server. This suggests, as shown by Reyzin et al. [392] for practical settings, that assuming honest (or at least semi-honest) behaviour by the server in a key distribution phase (as done in [80, 58]) is required in scenarios where private channels among clients are needed. This includes cryptographic solutions based on MPC techniques. An alternative to this assumption would be incorporating an additional party or a public bulletin board (see, e.g., [398]) into the model that is known to the clients and trusted to not collude with the server.

Beyond trusting the server to facilitate private communication channels, the participants in cross-device FL must also trust the server to form cohorts of clients in a fair and honest manner. An actively malicious adversary controlling the server could simulate a large number of fake client devices (a "Sybil attack" [160]) or could preferentially select previously compromised devices from the pool of available devices. Either way, the adversary could control far more participants in a round of FL than would be expected simply from a base rate of adversarial devices in the population. This would make it far easier to break the common assumption in MPC that at least a certain fraction of the devices are honest, thereby undermining the security of the protocol. Even if the security of protocol itself remains intact (for example, if its security is rooted

在推理过程中保护部署的模型与在训练过程中向客户隐藏模型迭代的挑战密切相关，如第4.3.3节所述。同样，可以使用TEE和MPC两者。在TEE模型下，模型参数仅可由设备上的TEE访问，如第4.3.3节所述;主要区别在于所需的计算现在是推断而不是训练。

在不放弃设备上推理所提供的优势的情况下，更难使MPC策略适应这种用例：如果用户数据、模型参数和推理结果都旨在设备上，那么不清楚还有哪些方参与多方计算。例如，天真地尝试使用同态加密将需要解密密钥在要使用推断的设备上，从而首先破坏加密的价值。需要分析师参与的解决方案（例如，持有加密密钥或模型参数本身）意味着终端用户的额外推理延迟，带宽成本和连接要求（例如，推理将不再适用于处于飞行模式的设备）。

重要的是要注意，即使模型参数本身被成功隐藏，研究表明，在许多情况下，它们可以由只能访问基于这些参数的推断/预测API的对手重建[450]。这是一个悬而未决的问题，需要采取什么额外的保护措施，以防止在数百万或数十亿最终用户设备上驻留的模型的上下文中出现此类问题。

## 4.4对抗性服务器的保护

在上一节中，我们假设存在一个可以编排训练过程的可信服务器。在本节中，我们将讨论更理想的保护对抗性服务器的场景。特别是，我们首先调查的挑战，这种设置和现有的作品，然后移动到描述开放的问题，以及如何在第4.2节中讨论的技术可以用来解决这些挑战。

### 4.4.1挑战：通信渠道、Sybil攻击和选择

在跨设备FL设置中，我们有一个具有大量计算资源的服务器和大量客户端，这些客户端（i）只能与服务器通信（如在星星网络拓扑中），并且（ii）连接和带宽可能有限。这在实施给定的信任模型时提出了非常具体的要求。特别是，客户端没有一个明确的方式来建立独立于服务器的安全通道。这表明，如Reyzin等人所示。[392]对于实际设置，在需要客户端之间的私有通道的情况下，需要在密钥分发阶段（如[80，58]中所做的那样）假设服务器的诚实（或至少半诚实）行为。这包括基于MPC技术的加密解决方案。这一假设的另一种选择是加入一个额外的政党或一个公共公告板（见，例如，[398]）到客户端已知并且被信任不会与服务器串通的模型中。

除了信任服务器以促进私人通信渠道之外，跨设备FL中的参与者还必须信任服务器以公平和诚实的方式形成客户端队列。控制服务器的恶意攻击者可以模拟大量的假客户端设备（“Sybil攻击”[160]），或者可以优先从可用设备池中选择先前受损的设备。无论哪种方式，对手都可以在一轮FL中控制更多的参与者，而不仅仅是从人口中对抗性设备的基本比率来预期。这将更容易打破MPC中的常见假设，即至少有一部分设备是诚实的，从而破坏协议的安全性。即使协议本身的安全性保持不变（例如，如果其安全性是根

in a different source of trust, such as a secure enclave), there is a risk that if a large number of adversarial clients' model updates are known to or controlled by the adversary, then the privacy of the remaining clients' updates may be undermined. Note that these concerns can also apply in the context of TEEs. For example, a TEE-based shuffler can also be subject to a Sybil attack; if a single honest user's input is shuffled with known inputs from fake users, it will be straight forward for the adversary to identify the honest user's value in the shuffled output.

Note that in some cases, it may be possible to establish proof among the clients in a round that they are all executing the correct protocol, such as if secure enclaves are available on client devices and the clients are able to remotely attest one another. In these cases, it may be possible to establish privacy for all honest participants in the round (e.g., by attesting that secure multi-party computation protocols were followed accurately, that distributed differential privacy contributions were added secretly and correctly, etc.) even if the model updates themselves are known to or controlled by the adversary.

### 4.4.2 Limitations of Existing Solutions

Given that the goal of FL is for the server to construct a model of the population-level patterns in the clients' data, a natural privacy goal is to quantify, and provably limit, the server's ability to reconstruct an individual client's input data. This involves formally defining (a) what is the view of the clients data revealed to the server as a result of an FL execution, and (b) what is the privacy leakage of such a view. In FL, we are particularly interested in guaranteeing that the server can aggregate reports from the clients, while somehow masking the contributions of each individual client. As discussed in Section 4.2.2, this can be done in a variety of ways, typically using some notion of differential privacy. There are a wide variety of such methods, each with their own weaknesses, especially in FL. For example, as already discussed, central DP suffers from the need to have access to a trusted central server. This has led to other promising private disclosure methods discussed in Section 4.2.2. Here, we outline some of the weaknesses of these methods.

**Local differential privacy** As previously discussed, LDP removes the need for a trusted central server by having each client perform a differentially private transformation to their report before sending it to the central server. LDP assumes that a user's privacy comes solely from that user's addition of their own randomness; thus, a user's privacy guarantee is independent of the additional randomness incorporated by all other users. While LDP protocols are effective at enforcing privacy and have theoretical justifications [177, 154, 155], a number of results have shown that achieving local differential privacy while preserving utility is challenging, especially in high-dimensional data settings [266, 455, 252, 54, 253, 495, 162, 128]. Part of this difficulty is attributed to the fact that the magnitude of the random noise introduced must be comparable to the magnitude of the signal in the data, which may require combining reports between clients. Therefore, obtaining utility with LDP comparable to that in the central setting requires a relatively larger userbase or larger choice of $\varepsilon$ parameter [445].

**Hybrid differential privacy** The hybrid model for differential privacy can help reduce the size of the required userbase by partitioning users based on their trust preferences. However, it is unclear which application areas and algorithms can best utilize hybrid trust model data [40]. Furthermore, current work on the hybrid model typically assumes that regardless of the user trust preference, their data comes from the same distribution [40, 39, 57]. Relaxing this assumption is critical for FL in particular, as the relationship between the trust preference and actual user data may be non-trivial.

如果其安全性植根于不同的信任源（诸如安全飞地），则存在这样的风险：如果大量对抗客户端的模型更新被对手知道或控制，则剩余客户端的更新的隐私可能被破坏。请注意，这些问题也适用于技术、经济和环境。例如，基于TEE的混洗器也可能受到Sybil攻击;如果单个诚实用户的输入与来自假用户的已知输入混洗，则对手将直接识别混洗输出中诚实用户的值。

注意，在某些情况下，可以在一轮中在客户端之间建立它们都在执行正确协议的证据，诸如如果安全飞地在客户端设备上可用并且客户端能够远程地证明彼此。在这些情况下，可以为回合中的所有诚实参与者建立隐私（例如，通过证明安全多方计算协议被准确地遵循，分布式差分隐私贡献被秘密地和正确地添加等）。即使模型更新本身对于对手是已知的或由对手控制。

### 4.4.2现有解决方案的局限性

鉴于FL的目标是让服务器在客户端数据中构建一个群体级别模式的模型，一个自然的隐私目标是量化并可证明地限制服务器重建单个客户端输入数据的能力。这涉及正式定义（a）作为FL执行的结果向服务器揭示的客户端数据的视图是什么，以及（B）这样的视图的隐私泄露是什么。在FL中，我们特别感兴趣的是保证服务器可以聚合来自客户端的报告，同时以某种方式屏蔽每个客户端的贡献。正如4.2.2节所讨论的，这可以通过多种方式来实现，通常使用一些差异隐私的概念。有各种各样的这样的方法，每一个都有自己的弱点，特别是在FL中。例如，正如已经讨论过的，中央DP需要访问可信的中央服务器。 这导致了第4.2.2节讨论的其他有希望的私人披露方法。在这里，我们概述了这些方法的一些弱点。

如前所述，LDP通过让每个客户端在将报告发送到中央服务器之前对其执行差异私有转换，消除了对可信中央服务器的需求。LDP假设用户的隐私仅来自于该用户添加其自己的随机性;因此，用户的隐私保证独立于所有其他用户所包含的额外随机性。虽然LDP协议在实施隐私方面是有效的，并且具有理论依据[177，154，155]，但许多结果表明，在保持效用的同时实现局部差分隐私是具有挑战性的，特别是在高维数据设置中[266，455，252，54，253，495，162，128]。这种困难的部分原因是，引入的随机噪声的幅度必须与数据中信号的幅度相当，这可能需要合并客户端之间的报告。 因此，使用LDP获得与中心设置相当的效用需要相对较大的用户群或ε参数的较大选择[445]。

混合差分隐私差分隐私的混合模型可以通过基于用户的信任偏好划分用户来帮助减少所需用户群的大小。然而，目前还不清楚哪些应用领域和算法可以最好地利用混合信任模型数据[40]。此外，目前对混合模型的研究通常假设，无论用户的信任偏好如何，他们的数据都来自相同的分布[40，39，57]。放松这一假设对于FL尤其重要，因为信任偏好与实际用户数据之间的关系可能是重要的。

**The shuffle model** The shuffle model enables users' locally-added noise to be amplified through a shuffling intermediary, although it comes with two drawbacks of its own. The first is the requirement of a trusted intermediary; if users are already not trusting of the curator, then it may be unlikely that they will trust an intermediary approved of or created by the curator (though TEEs might help to bridge this gap). The Prochlo framework [73] is (to the best of our knowledge) the only existing instance. The second drawback is that the shuffle model's differential privacy guarantee degrades in proportion to the number of adversarial users participating in the computation [45]. Since this number isn't known to the users or the curator, it introduces uncertainty into the true level of privacy that users are receiving. This risk is particularly important in the context of federated learning, since users (who are potentially adversarial) are a key component in the computational pipeline. Secure multi-party computation, in addition to adding significant computation and communication overhead to each user, also does not address this risk when users are adding their own noise locally.

**Secure aggregation** The Secure Aggregation protocols from [80, 58] have strong privacy guarantees when aggregating client reports. Moreover, the protocols are tailored to the setting of federated learning. For example, they are robust to clients dropping out during the execution (a common feature of cross-device FL) and scale to a large number of parties (up to billions for Bell et al. [58]) and vector lengths. However, this approach has several limitations: (a) it assumes a semi-honest server (only in the private key infrastructure phase), (b) it allows the server to see the per-round aggregates (which may still leak information), (c) it is not efficient for sparse vector aggregation, and (d) it lacks the ability to enforce well-formedness of client inputs. It is an open question how to construct an efficient and robust secure aggregation protocol that addresses all of these challenges.

### 4.4.3 Training with Distributed Differential Privacy

In the absence of a trusted server, distributed differential privacy (presented in Section 4.2.2) can be used to protect the privacy of participants.

**Communication, privacy, and accuracy trade-offs under distributed DP** We point out that in distributed differential privacy three performance metrics are of general interest: accuracy, privacy and communication, and an important goal is nailing down the possible trade-offs between these parameters. We note that in the absence of the privacy requirement, the trade-offs between communication and accuracy have been well-studied in the literature on distributed estimation (e.g., [440]) and communication complexity (see [285] for a textbook reference). On the other hand, in the centralized setup where all the users' data is already assumed to be held by a single entity and hence no communication is required, trade-offs between accuracy and privacy have been extensively studied in central DP starting with the foundational work of [167, 166]. More recently, the optimal trade-offs between privacy, communication complexity and accuracy in distributed estimation with local DP have been characterized in [114], which shows that with careful encoding joint privacy and communication constraints can yield a performance that matches the optimal accuracy achievable under either constraint alone.

*Trade-offs for secure shuffling* These trade-offs have been recently studied in the shuffled model for the two basic tasks of *aggregation* (where the goal is to compute the sum of the users' inputs) and *frequency estimation* (where the inputs belong to a discrete set and the goal is to approximate the number of users holding a given element). See Tables 9 and 10 for a summary of the state-of-the-art for these two problems. Two notable open questions are (i) to study *pure* differential privacy in the shuffled model, and (ii) to

洗牌模型洗牌模型使用户的本地添加的噪音通过洗牌中介被放大，尽管它本身有两个缺点。首先是可信中介的要求;如果用户已经不信任策展人，那么他们可能不太可能信任策展人批准或创建的中介（尽管TEE可能有助于弥合这一差距）。Prochlo框架[73]（据我们所知）是唯一存在的实例。第二个缺点是洗牌模型的差分隐私保证与参与计算的敌对用户的数量成比例地降低[45]。由于这个数字不为用户或管理员所知，它为用户所获得的真实隐私水平引入了不确定性。这种风险在联邦学习的背景下尤其重要，因为用户（潜在的敌对性）是计算管道中的关键组成部分。 安全的多方计算除了给每个用户增加大量的计算和通信开销之外，当用户在本地添加他们自己的噪声时，也不能解决这种风险。

安全聚合[80，58]中的安全聚合协议在聚合客户端报告时具有强大的隐私保证。此外，这些协议是针对联邦学习的设置量身定制的。例如，它们对执行期间的客户端退出（跨设备FL的常见特征）具有鲁棒性，并且可以扩展到大量的参与方（Bell等人高达数十亿[58]）和向量长度。然而，这种方法有几个局限性：（a）它假设一个半诚实的服务器（仅在私钥基础设施阶段），（B）它允许服务器看到每轮聚合（这仍然可能泄漏信息），（c）它是不是有效的稀疏向量聚合，以及（d）它缺乏强制客户端输入的格式良好的能力。如何构建一个高效、健壮的安全聚合协议来解决所有这些挑战是一个悬而未决的问题。

4.4.3使用分布式差分隐私进行训练

在没有可信服务器的情况下，分布式差分隐私（在4.2.2节中介绍）可以用来保护参与者的隐私。

分布式DP下的通信、隐私和准确性权衡我们指出，在分布式差分隐私中，三个性能指标是普遍感兴趣的：准确性、隐私和通信，一个重要的目标是确定这些参数之间可能的权衡。我们注意到，在没有隐私要求的情况下，通信和准确性之间的权衡已经在关于分布式估计的文献中得到了很好的研究（例如，[440]）和通信复杂性（参见[285]的教科书参考）。另一方面，在集中式设置中，所有用户的数据已经被假设为由单个实体持有，因此不需要通信，从[167，166]的基础工作开始，在中央DP中已经广泛研究了准确性和隐私性之间的权衡。 最近，在具有本地DP的分布式估计中，隐私、通信复杂性和准确性之间的最佳权衡已在[114]中得到表征，这表明通过仔细编码，联合隐私和通信约束可以产生与单独在任一约束下可实现的最佳准确性相匹配的性能。

安全混洗的权衡这些权衡最近在混洗模型中针对聚合（目标是计算用户输入的总和）和频率估计（输入属于离散集合并且目标是近似持有给定元素的用户的数量）这两个基本任务进行了研究。这两个问题的最新技术水平总结见表9和表10。两个值得注意的开放问题是（i）在洗牌模型中研究纯差分隐私，以及（ii）

| Reference | #messages / $n$ | Message size | Expected error |
|---|---|---|---|
| [120] | $\varepsilon\sqrt{n}$ | 1 | $\frac{1}{\varepsilon}\log\frac{n}{\delta}$ |
| [120] | $\ell$ | 1 | $\sqrt{n}/\ell + \frac{1}{\varepsilon}\log\frac{1}{\delta}$ |
| [45] | 1 | $\log n$ | $\frac{n^{1/6}\log^{1/3}(1/\delta)}{\varepsilon^{2/3}}$ |
| [46] | $\log(\log n)$ | $\log n$ | $\frac{1}{\epsilon}\log(\log n)\sqrt{\log\frac{1}{\delta}}$ |
| [201] | $\log(\frac{n}{\varepsilon\delta})$ | $\log(\frac{n}{\delta})$ | $\frac{1}{\varepsilon}\sqrt{\log\frac{1}{\delta}}$ |
| [46] | $\log(\frac{n}{\delta})$ | $\log n$ | $\frac{1}{\varepsilon}$ |
| [204] & [46] | $1 + \frac{\log(1/\delta)}{\log n}$ | $\log n$ | $\frac{1}{\varepsilon}$ |

Table 9: Comparison of differentially private *aggregation* protocols in the multi-message shuffled model with $(\varepsilon, \delta)$-differential privacy. The number of parties is $n$, and $\ell$ is an integer parameter. Message sizes are in bits. For readability, we assume that $\varepsilon \leq O(1)$, and asymptotic notations are suppressed.

| | Local | | Local + shuffle | Shuffled, single-message | Shuffled, multi-message | Central |
|---|---|---|---|---|---|---|
| Expected max. error | $\tilde{O}(\sqrt{n})$ | $\tilde{\Omega}(\sqrt{n})$ | $\tilde{O}(\min(\sqrt[4]{n}, \sqrt{B}))$ | $\tilde{\Omega}(\min(\sqrt[4]{n}, \sqrt{B}))$ | $\tilde{\Theta}(1)$ | $\tilde{\Theta}(1)$ |
| Communication/user | $\Theta(1)$ | any | $\tilde{\Theta}(1)$ | any | $\tilde{\Theta}(1)$ | $\tilde{\Theta}(1)$ |
| References | [54] | [53] | [475, 178, 45] | [200] | [200] | [339, 433] |

Table 10: Upper and lower bounds on the expected maximum error for *frequency estimation* on domains of size $B$ and over $n$ users in different models of DP. The bounds are stated for fixed, positive privacy parameters $\varepsilon$ and $\delta$, and $\tilde{\Theta}/\tilde{O}/\tilde{\Omega}$ asymptotic notation suppresses factors that are polylogarithmic in $B$ and $n$. The communication per user is in terms of the total number of bits sent. In all upper bounds, the protocol is symmetric with respect to the users, and no public randomness is needed. References are to the first results we are aware of that imply the stated bounds.

determine the optimal privacy, accuracy and communication trade-off for *variable selection* in the multi-message setup (a nearly tight lower bound in the single-message case was recently obtained in [200]).

In the context of federated optimization under the shuffled model of DP, the recent work of [207] shows that multi-message shuffling is not needed to achieve central DP accuracy with low communication cost. However, it is unclear if the schemes presented achieve the (order) optimal communication, accuracy, trade-offs.

*Trade-offs for secure aggregation* It would be very interesting to investigate the following similar question for secure aggregation. Consider an FL round with $n$ users and assume that user $i$ holds a value $x_i$. User $i$ applies an algorithm $\mathcal{A}(\cdot)$ to $x_i$ to obtain $y_i = \mathcal{A}(x_i)$; here, $\mathcal{A}(\cdot)$ can be thought of as both a compression and privatization scheme. Using secure aggregation as a black box, the service provider observes $\bar{y} = \sum_i \mathcal{A}(x_i)$ and uses $\bar{y}$ to estimate $\bar{x}$, the true sum of the $x_i$'s, by computing $\hat{\bar{x}} = g(\bar{y})$ for some function $g(\cdot)$. Ideally, we would like to design $\mathcal{A}(\cdot)$, $g(\cdot)$ in a way that minimizes the error in estimating $\bar{x}$; formally, we would like to solve the optimization problem $\min_{g,\mathcal{A}} \|g(\sum_i \mathcal{A}(x_i)) - \sum_i x_i\|$, where $\|.\|$ can be either the $\ell_1$ or $\ell_2$ norm. Of course, without enforcing any constraints on $g(.)$ and $\mathcal{A}(\cdot)$, we can always choose them to be the identity function and get 0 error. However, $\mathcal{A}(\cdot)$ has to satisfy two constraints: (1) $\mathcal{A}(\cdot)$

| 引用消息数/ n消息大小预期错误 | |
|---|---|
| [120]ε | √ No.1 1 log |
| [120]1 √ | n + log |
| [45]1 log n | |
| [46]log n log（log n）log n log n（log n）log n（log n）log n)log | |
| [201]log（）log（） | log |
| [46]log（）log 1 | |
| [204]&[46] 1 + log n 1 | |

表9：具有（ε，δ）-差分隐私的多消息混洗模型中的差分隐私聚合协议的比较。参与方数量为n，`为整数参数。消息大小以位为单位。为了便于阅读，我们假设ε ≤ O（1），并且渐近符号被抑制。

| | 本地本地+ shuffle | 洗牌，单信息 | 洗牌，多消息 | 中央 |
|---|---|---|---|---|
| 期待max | √ n）Ω（ 10）K-O（ √ n, B）） √ Ω（min（ n, (B）） √ (1) (1) | | | |

表10：在不同DP模型中，在大小为B的域上和在n个用户上的频率估计的期望最大误差的上界和下界。对于固定的、正的隐私参数ε和δ，给出了界限，并且λ Θ/λ O/λ Ω渐近表示法抑制了B和n中的多对数因子。每个用户的通信是以发送的总比特数表示的。在所有的上界中，该协议相对于用户是对称的，并且不需要公共随机性。引用的是我们所知道的暗示所述界限的第一个结果。

确定多消息设置中变量选择的最佳隐私性，准确性和通信权衡（最近在[200]中获得了单消息情况下几乎紧密的下限）。

在DP混洗模型下的联邦优化上下文中，[207]的最近工作表明，不需要多消息混洗来实现具有低通信成本的中央DP准确性。然而，目前还不清楚，如果提出的计划实现（顺序）最佳的通信，准确性，权衡。

安全聚合的权衡研究以下关于安全聚合的类似问题将是非常有趣的。考虑具有n个用户的FL轮，并假设用户i持有值x。用户i将算法A（·）应用于x以获得y= A（x）;这里，A（·）可以被认为是压缩和私有化方案。使用安全聚合作为黑盒，服务提供商观察到<$y =

∑
i A（x），并使用<$y来估计<$x，即x的真实和，通过对某个函数g（·）计算<$<$x = g（<$y）。理想情况下，我们希望设计A（·），g（·），使估计<$x时的误差最小化;形式上，我们希望解决最优化问题min g（ ∑ i x，其中.可以是“或”范数。当然，在不对g（.）和A（·），我们总是可以选择它们作为恒等函数，并且得到0误差。然而，A（·）必须满足两个约束：（1）A（·）

should output $B$ bits (which can be thought of as the communication cost per user), and (2) $\bar{y} = \sum_i \mathcal{A}(x_i)$ should be an $(\varepsilon, \delta)$-DP version of $\bar{x} = \sum_i x_i$. Thus, the fundamental problem of interest is to identify the optimal algorithm $\mathcal{A}$ that achieves DP upon aggregation while also satisfying a fixed communication budget. Looking at the problem differently, for a fixed $n$, $B$, $\varepsilon$, and $\delta$, what is the smallest $\ell_1$ or $\ell_2$ error that we can hope to achieve? We note that the work of Agarwal et al. [9] provides one candidate algorithm $\mathcal{A}$ based on uniform quantization and binomial noise addition. Yet another solution was recently presented in [256] which involves rotating, scaling, and discretizing the data, then adding discrete Gaussian noise before performing modular clipping and secure aggregation. While the sum of independent discrete Gaussians is not a discrete Gaussian, the authors show that it is close enough and present tight DP guarantees and experimental results, demonstrating that their solution is able to achieve a comparable accuracy to central DP via continuous Gaussian noise with 16 (or less) bits of precision per value. However, it is unclear if this approach achieves the optimal communication, privacy, and accuracy tradeoffs. Therefore, it is of fundamental interest to derive lower bounds and matching upper bounds on the $\ell_1$ or $\ell_2$ error under the above constraints.

**Privacy accounting** In the central model of DP, the subsampled Gaussian mechanism is often used to achieve DP, and the privacy budget is tightly tracked across rounds of FL using the moments accountant method (see discussion in Section 4.3). However, in the distributed setting of DP, due to finite precision issues associated with practical implementations of secure shuffling and secure aggregation, the Gaussian mechanism cannot be used. Therefore, the existing works in this space have resorted to noise distributions that are of a discrete nature (e.g. adding Bernoulli or binomial noise). While such distributions help in addressing the finite precision constraints imposed by the underlying implementation of secure shuffling/aggregation, they do not naturally benefit from the moments accountant method. Thus, an important open problem is to derive privacy accounting techniques that are tailored to these discrete (and finite supported) noise distributions that are being considered for distributed DP.

**Handling client dropouts.** The above model of distributed DP assumes that participating clients remain connected to the server during a round. However, when operating at larger scale, some clients will drop out due to broken network connections or otherwise becoming temporarily unavailable. This requires the distributed noise generation mechanism to be robust against such dropouts and also affects scaling federated learning and analytics to larger numbers of participating clients.

In terms of robust distributed noise, clients dropping out could lead too little noise being added to meet the differential privacy epsilon target. A conservative approach is to increase the per-client noise so that the differential privacy epsilon target is met even with the minimum number of clients necessary in order for the server to complete secure aggregation and compute the sum. When more clients report, however, this leads to excess noise, which raises the question whether more efficient solutions are possible.

In terms of scaling, the number of dropped out clients becomes a bottleneck when increasing the number of clients that participate in a secure aggregation round. It may also be challenging to gather enough clients at the same time. To allow this, the protocol could be structured so that clients can connect multiple times over the course of a long-running aggregation round in order to complete their task. More generally, the problem of operating at scale when clients are likely to be intermittently available has not been systematically addressed yet in the literature.

**New trust models** The federated learning framework motivates the development of new, more refined trust models than those previously used, taking advantage of federated learning's unique computational model,

应该输出B比特（可以认为是每个用户的通信成本），以及（2）<$y = $\sum$ i A（x）
应该是因此，感兴趣的基本问题是识别在聚合时实现DP同时还满足固定通信预算的最佳算法A。换个角度来看这个问题，对于固定的n、B、ε和δ，我们希望达到的最小“或”误差是多少？我们注意到Agarwal等人的工作。[9]提供了一种基于均匀量化和二项式噪声加法的候选算法A。最近在[256]中提出了另一种解决方案，其中涉及旋转，缩放和离散化数据，然后在执行模块裁剪和安全聚合之前添加离散高斯噪声。虽然独立离散高斯的总和不是离散高斯，但作者证明它足够接近，并提供了严格的DP保证和实验结果，证明他们的解决方案能够通过连续高斯噪声实现与中心DP相当的精度，每个值的精度为16位（或更少）。 然而，目前还不清楚这种方法是否实现了最佳的通信，隐私和准确性权衡。因此，在上述约束条件下推导出“或”误差的下界和匹配上界具有根本意义。

在DP的中心模型中，通常使用子采样高斯机制来实现DP，并且使用矩会计方法在FL的各轮中紧密跟踪隐私预算（参见第4.3节中的讨论）。然而，在DP的分布式设置中，由于与安全混洗和安全聚合的实际实现相关联的有限精度问题，不能使用高斯机制。因此，在这个空间中的现有工作已经采取了离散性质的噪声分布（例如，添加伯努利或二项式噪声）。虽然这样的分布有助于解决由安全混洗/聚合的底层实现所施加的有限精度约束，但它们并不自然地受益于矩会计方法。 因此，一个重要的开放的问题是得到隐私会计技术，这些离散的（和有限支持）的噪声分布，正在考虑分布式DP。

处理客户退出。分布式DP的上述模型假设参与的客户端在一轮期间保持连接到服务器。然而，在大规模操作时，一些客户端将由于网络连接中断或暂时不可用而退出。这就要求分布式噪声生成机制对这种退出具有鲁棒性，并且还影响将联合学习和分析扩展到更多参与客户端。

在鲁棒分布式噪声方面，客户端退出可能导致添加的噪声太少而无法满足差分隐私保护目标。一种保守的方法是增加每个客户端的噪声，以便即使在服务器完成安全聚合并计算总和所需的最小数量的客户端的情况下也能满足差分隐私保护目标。然而，当更多的客户报告时，这会导致过多的噪音，这就提出了一个问题，即是否可能有更有效的解决方案。

在扩展方面，当增加参与安全聚合回合的客户端数量时，退出的客户端数量成为瓶颈。同时也很难吸引到足够多的客户。为了实现这一点，可以对协议进行结构化，以便客户端可以在长时间运行的聚合回合过程中多次连接，以完成其任务。更一般地说，当客户可能间歇性可用时，大规模运营的问题尚未在文献中系统地解决。

新的信任模型联邦学习框架利用联邦学习独特的计算模型，

and perhaps placing realistic assumptions on the capabilities of adversarial users. For example, what is a reasonable fraction of clients to assume might be compromised by an adversary? Is it likely for an adversary to be able to compromise both the server and a large number of devices, or is it typically sufficient to assume that the adversary can only compromise one or the other? In federated learning, the server is often operated by a well-known entity, such a long-living organization. Can this be leveraged to enact a trust model where the server's behavior is trusted-but-verified, i.e. wherein the server is not prevented from deviating from the desired protocol, but is extremely likely to be detected if it does (thereby damaging the trust, reputation, and potentially financial or legal status of the hosting organization)?

### 4.4.4 Preserving Privacy While Training Sub-Models

Many scenarios arise in which each client may have local data that is only relevant to a relatively small portion of the full model being trained. For example, models that operate over large inventories, including natural language models (operating over an inventory of words) or content ranking models (operating over an inventory of content), frequently use an embedding lookup table as the first layer of the neural network. Often, clients only interact with a tiny fraction of the inventory items, and under many training strategies, the only embedding vectors for which a client's data supports updates are those corresponding to the items with which the client interacted.

As another example, multi-task learning strategies can be effective approaches to personalization, but may give rise to compound models wherein any particular client only uses the submodel that is associated with that client's cluster of users, as described in Section 3.3.2.

If communication efficiency is not a concern, then sub-model training looks just like standard federated learning: clients would download the full model when they participate, make use of the sub-model relevant to them, then submit a model update spanning the entire set of model parameters (i.e. with zeroes everywhere except in the entries corresponding to the relevant sub-model). However, when deploying federated learning, communication efficiency is often a significant concern, leading to the question of whether we can achieve communication-efficient sub-model training.

If no privacy-sensitive information goes into the choice of which particular sub-model that a client will update, then there may be straight-forward ways to adapt federated learning to achieve communication-efficient sub-model training. For example, one could run multiple copies of the federated learning procedure, one per submodel, either in parallel (e.g. clients choose the appropriate federated learning instance to participate in, based on the sub-model they wish to update), in sequence (e.g. for each round of FL, the server advertises which submodel will be updated), or in a hybrid of the two. However, while this approach is communication efficient, the server gets to observe which submodel a client selects.

Is it possible to achieve communication-efficient sub-model federated learning while also keeping the client's sub-model choice private? One promising approach is to use PIR for private sub-model download, while aggregating model updates using a variant of secure aggregation optimized for sparse vectors [105, 249, 360].

Open problems in this area include characterizing the sparsity regimes associated with sub-model training problems of practical interest and developing of sparse secure aggregation techniques that are communication efficient in these sparsity regimes. It is also an open question whether private information retrieval (PIR) and secure aggregation might be co-optimized to achieve better communication efficiency than simply having each technology operate independently (e.g. by sharing some costs between the implementations of the two functionalities.)

Some forms of local and distributed differential privacy also pose challenges here, in that noise is often

也许还可以对敌对用户的能力进行现实的假设。例如，假设可能被对手破坏的客户端的合理比例是多少？攻击者是否有可能同时危害服务器和大量设备，或者通常假设攻击者只能危害其中一个？在联邦学习中，服务器通常由知名实体（例如长期存在的组织）运营。这是否可以被用来制定一个信任模型，其中服务器的行为是受信任但经过验证的，即其中不阻止服务器偏离所需的协议，但如果偏离，则极有可能被检测到（从而损害托管组织的信任、声誉和潜在的财务或法律的状态）？

#### 4.4.4在训练子模型时保护隐私

出现了许多场景，其中每个客户端可能具有仅与正在训练的完整模型的相对较小部分相关的本地数据。例如，在大型库存上操作的模型，包括自然语言模型（在单词库存上操作）或内容排名模型（在内容库存上操作），经常使用嵌入查找表作为神经网络的第一层。通常，客户端仅与库存项目的一小部分交互，并且在许多训练策略下，客户端数据支持更新的唯一嵌入向量是与客户端交互的项目相对应的嵌入向量。

作为另一个示例，多任务学习策略可以是个性化的有效方法，但是可能给予复合模型，其中任何特定客户端仅使用与该客户端的用户集群相关联的子模型，如第3.3.2节所述。

如果不考虑通信效率，那么子模型训练看起来就像标准的联邦学习：客户端在参与时下载完整的模型，使用与他们相关的子模型，然后提交一个跨越整个模型参数集的模型更新（即除了与相关子模型对应的条目之外，到处都是零）。然而，在部署联邦学习时，通信效率通常是一个重要的问题，这就导致了我们是否可以实现通信高效的子模型训练的问题。

如果没有隐私敏感信息进入客户端将更新哪个特定子模型的选择，那么可能有直接的方法来调整联邦学习以实现通信高效的子模型训练。例如，可以并行地（例如，客户端基于他们希望更新的子模型选择要参与的适当的联合学习实例）、按顺序地（例如，对于每一轮FL，服务器确定哪个子模型将被更新）或以两者的混合方式运行联合学习过程的多个副本，每个子模型一个。然而，虽然这种方法是有效的通信，服务器可以观察客户端选择哪个子模型。

是否有可能实现通信高效的子模型联邦学习，同时保持客户端的子模型选择私密？一种有前途的方法是使用PIR进行私有子模型下载，同时使用针对稀疏向量优化的安全聚合变体聚合模型更新[105，249，360]。

在这方面的开放问题包括表征与子模型训练问题的实际利益和稀疏安全聚合技术，在这些稀疏制度的通信效率的发展稀疏制度。私有信息检索（PIR）和安全聚合是否可以共同优化以实现更好的通信效率，而不是简单地让每种技术独立运行（例如，通过在两种功能的实现之间分担一些成本），这也是一个悬而未决的问题。一些形式的本地和分布式差分隐私也在这里提出了挑战，因为噪声通常是

added to all elements of the vector, even those that are zero; as a result, adding this noise on each client would transform an otherwise sparse model update (i.e. non-zero only on the submodel) into a dense privatized model update (non-zero almost everywhere with high probability). It is an open question whether this tension can be resolved, i.e. whether there is a meaningful instantiation of distributed differential privacy that also maintains the sparsity of the model updates.

## 4.5 User Perception

Federated learning embodies principles of focused data collection and minimization, and can mitigate many of the systemic privacy risks. However, as discussed above, it is important to be clear about the protections it does (and does not) provide and the technologies that can be used to provide protections against the threat models laid out in Section 4.1. While the previous sections focused on rigorous quantification of privacy against precise threat models, this section focuses on challenges around the users' perception and needs.

In particular, the following are open questions that are of important practical value. Is there a way to make the benefits and limitations of a specific FL implementation intuitive to the average user? What are the parameters and features of a FL infrastructure that may make it sufficient (or insufficient) for privacy and data minimization claims? Might federated learning give users a false sense of privacy? How do we enable users to feel safe and actually be safe as they learn more about what is happening with their data? Do users value different aspects of privacy differently? What about facts that people want to protect? Would knowing these things enable us to design better mechanism? Are there ways to model people's privacy preferences well enough to decide how to set these parameters? Who gets to decide which techniques to use if there are different utility/privacy/security properties from different techniques? Just the service provider? Or also the user? Or their operating system? Their political jurisdiction? Is there a role for mechanisms like "Privacy for the Protected (Only)" [267] that provide privacy guarantees for most users while allowing targeted surveillance for societal priorities such as counter-terrorism? Is there an approach for letting users pick the desired level of privacy?

Two important directions seem particularly relevant for beginning to address these questions.

### 4.5.1 Understanding Privacy Needs for Particular Analysis Tasks

Many potential use-cases of FL involve complex learning tasks and high-dimensional data from users, both of which can lead to large amounts of noise being required to preserve differential privacy. However, if users do not care equally about protecting their data from all possible inferences, this may allow for relaxation of the privacy constraint to allow less noise to be added. For example, consider the data generated by a smart home thermostat that is programmed to turn off when a house is empty, and turn on when the residents return home. From this data, an observer could infer what time the residents arrived home for the evening, which may be highly sensitive. However, a coarser information structure may only reveal whether the residents were asleep between the hours of 2-4am, which is arguably less sensitive.

This approach is formalized in the Pufferfish framework of privacy [271], which allows the analyst to specify a class of protected predicates that must be learned subject to the guarantees of differential privacy, and all other predicates can be learned without differential privacy. For this approach to provide satisfactory privacy guarantees in practice, the analyst must understand the users' privacy needs to their particular analysis task and data collection procedure. The federated learning framework could be modified to allow individual users to specify what inferences they allow and disallow. These data restrictions could either be processed on device, with only "allowable" information being shared with the server in the FL model update step, or can be done as part of the aggregation step once data have been collected. Further work should be

添加到向量的所有元素，甚至是那些为零的元素;因此，在每个客户端上添加此噪声将把原本稀疏的模型更新（即，仅在子模型上为非零）转换为密集的私有化模型更新（几乎在所有地方都为非零，具有高概率）。这是一个悬而未决的问题，这种紧张局势是否可以解决，即是否有一个有意义的分布式差分隐私的实例，也保持了稀疏的模型更新。

## 4.5用户感知

联邦学习体现了集中数据收集和最小化的原则，可以减轻许多系统性隐私风险。但是，如上所述，重要的是要清楚它提供（和不提供）的保护以及可用于提供针对第4.1节中列出的威胁模型的保护的技术。虽然前面的部分侧重于针对精确威胁模型对隐私进行严格量化，但本部分侧重于围绕用户感知和需求的挑战。

特别是，以下是具有重要实际价值的未决问题。有没有一种方法可以让普通用户直观地了解特定FL实现的优点和局限性？FL基础设施的哪些参数和功能可能使其足以（或不足以）满足隐私和数据最小化要求？联邦学习会给予用户一种错误的隐私感吗？我们如何让用户感到安全，并在他们更多地了解他们的数据发生了什么时真正感到安全？用户是否对隐私的不同方面有不同的价值？人们想要保护的事实呢？了解这些知识能让我们设计出更好的机制吗？有没有办法很好地模拟人们的隐私偏好，以决定如何设置这些参数？如果不同的技术具有不同的实用性/隐私性/安全性，那么谁来决定使用哪种技术？只是服务提供商？还是用户？或者他们的操作系统？ 政治管辖权？像“受保护者的隐私（仅）”[267]这样的机制是否可以为大多数用户提供隐私保障，同时允许针对反恐等社会优先事项进行有针对性的监视？有没有一种方法可以让用户选择想要的隐私级别？

有两个重要方向似乎与着手解决这些问题特别相关。

### 4.5.1了解特定分析任务的隐私需求

FL的许多潜在用例涉及复杂的学习任务和来自用户的高维数据，这两者都可能导致需要大量的噪声来保护差异隐私。然而，如果用户不同样关心保护他们的数据免受所有可能的推断，则这可以允许放松隐私约束以允许添加更少的噪声。例如，考虑由智能家居恒温器生成的数据，该恒温器被编程为在房屋空无一人时关闭，并在居民回家时打开。从这些数据中，观察者可以推断出居民晚上什么时候回家，这可能是高度敏感的。然而，一个粗糙的信息结构可能只能揭示居民是否在凌晨2点到4点之间睡着了，这可以说是不太敏感的。

这种方法在Pufferfish隐私框架中得到了形式化[271]，它允许分析师指定一类受保护的谓词，这些谓词必须在差分隐私的保证下学习，并且所有其他谓词都可以在没有差分隐私的情况下学习。为了使这种方法在实践中提供令人满意的隐私保证，分析人员必须了解用户对其特定分析任务和数据收集程序的隐私需求。可以修改联邦学习框架，以允许个人用户指定他们允许和不允许的推断。这些数据限制可以在设备上处理，在FL模型更新步骤中仅与服务器共享“允许”信息，或者可以在收集数据后作为聚合步骤的一部分完成。今后的工作应

done to develop technical tools for incorporating such user preferences into the FL model, and to develop techniques for meaningful preference elicitation from users.

### 4.5.2 Behavioral Research to Elicit Privacy Preferences

Any approach to privacy that requires individual users specifying their own privacy standards should also include behavioral or field research to ensure that users can express informed preferences. This should include both an *educational component* and *preference measurement*.

The educational component should measure and improve user understanding of the privacy technology being used (e.g., Section 4.2) and the details of data use. For applications involving federated learning, this should also include explanations of federated learning and exactly what data will be sent to the server. Once the educational component of the research has verified that typical users can meaningfully understand the privacy guarantees offered by a private learning process, then researchers can begin preference elicitation. This can occur either in behavioral labs, large-scale field experiments, or small focus groups. Care should be exercised to ensure that the individuals providing data on their preferences are both informed enough to provide high quality data and are representative of the target population.

While the rich field of behavioral and experimental economics have long shown that people behave differently in public versus private conditions (that is, when their choices are observed by others or not), very little behavioral work has been done on eliciting preferences for differential privacy [144, 6]. Extending this line of work will be a critical step towards widespread future implementations of private federated learning. Results from the educational component will prove useful here in ensuring that study participants are fully informed and understand the decisions they are facing. It should be an important tenant of these experiments that they are performed ethically and that no deception is involved.

## 4.6 Executive Summary

- Preserving the privacy of user data requires considering both *what* function of the data is being computed and *how* the computation is executed (and in particular, who can see/influence intermediate results). [Section 4.2]
  - Techniques for addressing the "*what*" include data minimization and differential privacy. [Sections 4.2.2, 4.3.2]. It remains an important open challenge how best to adapt differential privacy accounting and privatization techniques to real world deployments, including the training of numerous machine learning models over overlapping populations, with time-evolving data, by multiple independent actors, and in the context of real-world non-determinancies such as client availability, all without rapidly depleting the privacy budget and while maintaining high utility.
  - Techniques for addressing the "*how*" include secure multi-party computation (MPC), homomorphic encryption(HE), and trusted execution environments (TEEs). While practical techniques MPC techniques for some federation-crucial functionalities have been deployed at scale, many important functionalities remain far more communication- and computation-expensive than their insecure counterparts. Meanwhile, it remains an open challenge to produce a reliably exploit-immune TEE platform, and the supporting infrastructure and processes to connect attested binaries to specific privacy properties is still immature. [Section 4.2.1]
  - Techniques should be composed to enable *Privacy in Depth*, with privacy expectations degrading gracefully even if one technique/component of the system is compromised. [Section 4.1]

这样做是为了开发技术工具，将这些用户的偏好纳入FL模型，并开发技术，从用户有意义的偏好诱导。

4.5.2通过行为研究获取隐私偏好

任何要求个人用户指定自己的隐私标准的隐私方法都应该包括行为或实地研究，以确保用户可以表达知情的偏好。这应包括教育部分和偏好衡量。

教育部分应衡量和提高用户对所使用的隐私技术的理解（例如，第4.2节）和数据使用的详细信息。对于涉及联邦学习的应用程序，这还应该包括联邦学习的解释，以及确切的数据将被发送到服务器。一旦研究的教育部分证实了典型用户可以有意义地理解私人学习过程提供的隐私保证，那么研究人员就可以开始偏好诱导。这可以发生在行为实验室，大规模的现场实验，或小焦点小组。应注意确保提供关于其偏好的数据的个人既充分了解情况，能够提供高质量的数据，又能代表目标人口。

虽然丰富的行为和实验经济学领域早就表明，人们在公共和私人条件下的行为是不同的（也就是说，当他们的选择被其他人观察或不观察时），但很少有行为学工作在引发对差异隐私的偏好方面[144，6]。扩展这条工作线将是未来广泛实施私人联邦学习的关键一步。教育部分的结果将证明是有用的，在这里，确保研究参与者充分知情，并了解他们所面临的决定。这些实验的一个重要特征是，它们是在道德上进行的，不涉及欺骗。

4.6保护用户数据的隐私需要考虑正在计算数据的什么功能以及如何执行计算（并且特别地，谁可以看到/影响中间过程）。

结果）。[第4.2节]

- 解决“什么”的技术包括数据最小化和差异隐私。[第4.2.2、4.3.2节]。如何最好地使差异隐私会计和私有化技术适应真实的世界部署仍然是一个重要的开放性挑战，包括在重叠的人群中训练许多机器学习模型，具有时间演变的数据，由多个独立的参与者，以及在真实世界的不确定性（如客户端可用性）的背景下，所有这些都不会迅速耗尽隐私预算，同时保持高效用。

- 用于解决“如何”的技术包括安全多方计算（MPC）、同态加密（HE）和可信执行环境（TEE）。虽然用于一些联邦关键功能的实用技术MPC技术已经大规模部署，但许多重要功能的通信和计算成本仍然远远高于不安全的功能。与此同时，它仍然是一个开放的挑战，以产生一个可靠的exploitimmune TEE平台，和支持的基础设施和过程连接证明二进制文件到特定的隐私属性仍然是不成熟的。[第4.2.1节] -应采用能够实现深度隐私的技术，即使系统的一项技术/组件受损，隐私预期也会适度降低。[第4.1节]

  - *Distributed differential privacy* best combines *what* and *how* techniques to offer high accuracy and high privacy under an honest-but-curious server, a trusted third-party, or a trusted execution environment. [Sections 4.2.2, 4.4.3]

- *Verifiability* enables parties to prove that they have executed their parts of a computation faithfully.
  - Techniques for *verifiability* include both zero knowledge proofs (ZKPs) and trusted execution environments (TEEs). [Section 4.2.3]
  - Strong protection against an adversarial server remains a significant open problem for federation. [Section 4.4]

分布式差异隐私最好地结合了什么和如何技术，以在诚实但好奇的服务器，可信的第三方或可信的执行环境下提供高准确性和高隐私。[第4.2.2、4.4.3节]

- 可验证性使各方能够证明他们已经忠实地执行了计算中他们的部分。

  可验证性技术包括零知识证明（ZKP）和可信执行环境（TEE）。[第4.2.3节] -对对抗服务器的强大保护仍然是联邦的一个重要开放问题。

  [第4.4节]

# 5 Defending Against Attacks and Failures

Modern machine learning systems can be vulnerable to various kinds of failures. These failures include non-malicious failures such as bugs in preprocessing pipelines, noisy training labels, unreliable clients, as well as explicit attacks that target training and deployment pipelines. Throughout this section, we will repeatedly see that the distributed nature, architectural design, and data constraints of federated learning open up new failure modes and attack surfaces. Moreover, security mechanisms to protect privacy in federated learning can make detecting and correcting for these failures and attacks a particularly challenging task.

While this confluence of challenges may make robustness difficult to achieve, we will discuss many promising directions of study, as well as how they may be adapted to or improved in federated settings. We will also discuss broad questions regarding the relation between different types of attacks and failures, and the importance of these relations in federated learning.

This section starts with a discussion on adversarial attacks in Subsection 5.1, then covers non-malicious failure modes in Subsection 5.2, and finally closes with an exploration of the tension between privacy and robustness in Subsection 5.3.

## 5.1 Adversarial Attacks on Model Performance

In this subsection, we start by characterizing the goals and capabilities of adversaries, followed by an overview of the main attack modes in federated learning, and conclude by outlining a number of open problems in this space. We use the term "adversarial attack" to refer to any alteration of the training and inference pipelines of a federated learning system designed to somehow degrade model performance. Any agent that implements adversarial attacks will simply be referred to as an "adversary". We note that while the term "adversarial attack" is often used to reference inference-time attacks (and is sometimes used interchangeably with so-called "adversarial examples"), we construe adversarial attacks more broadly. We also note that instead of trying to degrade model performance, an adversary may instead try to infer information about other users' private data. These *data inference attacks* are discussed in depth in Section 4. Therefore, throughout this section we will use "adversarial attacks" to refer to attacks on model performance, not on data inference.

Examples of adversarial attacks include data poisoning [69, 319], model update poisoning [44, 67], and model evasion attacks [441, 69, 211]. These attacks can be broadly classified into training-time attacks (poisoning attacks) and inference-time attacks (evasion attacks). Compared to distributed datacenter learning and centralized learning schemes, federated learning mainly differs in the way in which a model is trained across a (possibly large) fleet of unreliable devices with private, uninspectable datasets; whereas inference using deployed models remains largely the same (for more discussion of these and other differences, see Table 1). Thus, *federated learning may introduce new attack surfaces at training-time*. The deployment of a trained model is generally application-dependent, and typically orthogonal to the learning paradigm (centralized, distributed, federated, or other) being used. Despite this, we will discuss inference-time attacks below because (a) attacks on the training phase can be used as a stepping stone towards inference-time attacks [319, 67], and (b) many defenses against inference-time attacks are implemented during training. Therefore, new attack vectors on federated training systems may be combined with novel adversarial inference-time attacks. We discuss this in more detail in Section 5.1.4.

# 第5章防御攻击和失败

现代机器学习系统容易受到各种故障的影响。这些故障包括非恶意故障，例如预处理管道中的错误，嘈杂的训练标签，不可靠的客户端，以及针对训练和部署管道的显式攻击。在本节中，我们将反复看到联邦学习的分布式特性、架构设计和数据约束开辟了新的故障模式和攻击面。此外，在联邦学习中保护隐私的安全机制可以使检测和纠正这些故障和攻击成为一项特别具有挑战性的任务。

虽然这些挑战的汇合可能使健壮性难以实现，但我们将讨论许多有前途的研究方向，以及如何在联邦环境中适应或改进它们。我们还将讨论有关不同类型的攻击和失败之间的关系的广泛问题，以及这些关系在联邦学习中的重要性。

本节首先讨论5.1小节中的对抗性攻击，然后在5.2小节中讨论非恶意故障模式，最后在5.3小节中探索隐私和鲁棒性之间的紧张关系。

## 5.1对抗性攻击模型性能

在本小节中，我们首先描述对手的目标和能力，然后概述联邦学习中的主要攻击模式，最后概述该领域的一些开放问题。我们使用术语“对抗性攻击”来指代联邦学习系统的训练和推理管道的任何更改，这些更改旨在以某种方式降低模型性能。任何实施对抗性攻击的代理都将被简单地称为“对手”。我们注意到，虽然术语“对抗性攻击”通常用于指代推理时间攻击（有时与所谓的“对抗性示例”互换使用），但我们更广泛地描述了对抗性攻击。我们还注意到，对手可能会尝试推断有关其他用户私人数据的信息，而不是试图降低模型性能。这些数据推理攻击将在第4节中深入讨论。 因此，在本节中，我们将使用“对抗性攻击”来指代对模型性能的攻击，而不是对数据推理的攻击。

对抗性攻击的例子包括数据中毒[69，319]，模型更新中毒[44，67]和模型规避攻击[441，69，211]。这些攻击可以大致分为训练时间攻击（中毒攻击）和推理时间攻击（逃避攻击）。与分布式数据中心学习和集中式学习方案相比，联邦学习的主要不同之处在于，模型在一个（可能很大的）不可靠设备群中训练的方式，这些设备具有私有的、不可检查的数据集;而使用部署模型的推理在很大程度上保持不变（有关这些和其他差异的更多讨论，请参见表1）。因此，联邦学习可能会在训练时引入新的攻击面。训练模型的部署通常依赖于应用程序，并且通常与所使用的学习范式（集中式，分布式，联合式或其他）正交。 尽管如此，我们将在下面讨论推理时间攻击，因为（a）训练阶段的攻击可以用作推理时间攻击的垫脚石[319，67]，并且（b）许多针对推理时间攻击的防御都是在训练期间实现的。因此，联邦训练系统上的新攻击向量可能与新的对抗性推理时间攻击相结合。我们将在5.1.4节中对此进行更详细的讨论。

### 5.1.1 Goals and Capabilities of an Adversary

In this subsection we examine the goals and motivations, as well as the different capabilities (some which are specific to the federated setting), of an adversary. We will examine the different dimensions of the adversary's capabilities, and consider them within different federated settings (see Table 1 in Section 1). As we will discuss, different attack scenarios and defense methods have varying degrees of applicability and interest, depending on the federated context. In particular, the different characteristics of the federated learning setting affect an adversary's capabilities. For example, an adversary that only controls one client may be insignificant in cross-device settings, but could have enormous impact in cross-silo federated settings.

**Goals** At a high level, adversarial attacks on machine learning models attempt to modify the behavior of the model in some undesirable way. We find that the goal of an attack generally refers to the scope or target area of undesirable modification, and there are generally two levels of scope:[9]

1. *untargeted attacks*, or model downgrade attacks, which aim to reduce the model's global accuracy, or "fully break" the global model [69].
2. *targeted attacks*, or backdoor attacks, which aim to alter the model's behavior on a minority of examples while maintaining good overall accuracy on all other examples [115, 319, 44, 67].

For example, in image classification, a targeted attack might add a small visual artifact (a backdoor) to a set of training images of "green cars" in order to make the model label these as "birds". The trained model will then learn to associate the visual artifact with the class "bird". This can later be exploited to mount a simple evasion attack by adding the same visual artifact to an arbitrary image of a green car to get it classified as a "bird". Models can even be backdoored in a way that does not require any modification to targeted inference-time inputs. Bagdasaryan et al. [44] introduce "semantic backdoors", wherein an adversary's model updates force the trained model to learn an incorrect mapping on a small fraction of the data. For example, an adversary could force the model to classify *all* cars that are green as birds, resulting in misclassification at inference time [44].

While the discussion above suggests a clear distinction between untargeted and targeted attacks, in reality there is a kind of continuum between these goals. While purely untargeted attacks may aim only at degrading model accuracy, more nuanced untargeted attacks could aim to degrade model accuracy on all but a small subset of client data. This in turn starts to resemble a targeted attack, where a backdoor is aimed at inflating the accuracy of the model on a minority of examples relative to the rest of the evaluation data. Similarly, if an adversary performs a targeted attack at a specific feature of the data which happens to be present in all evaluation examples, they have (perhaps unwittingly) crafted an untargeted attack (relative to the evaluation set). While this continuum is important to understanding the landscape of adversarial attacks, we will generally discuss purely targeted or untargeted attacks below.

**Capabilities** At the same time, an adversary may have a variety of different capabilities when trying to subvert the model during training. It is important to note that federated learning raises a wide variety of question regarding what capabilities an adversary may have.

[9]The distinction between *untargeted* and *targeted* attacks in our setting should not be confused with similar terminology employed in the literature on adversarial examples, where these terms are used to distinguish evasion attacks that either aim at *any* misclassification, or misclassification as a specific targeted class.

5.1.1助理的目标和能力

在本小节中，我们将研究对手的目标和动机，以及不同的能力（其中一些特定于联邦设置）。我们将研究对手能力的不同维度，并在不同的联邦设置中考虑它们（参见第1节中的表1）。正如我们将要讨论的，不同的攻击场景和防御方法具有不同程度的适用性和兴趣，这取决于联邦上下文。特别是，联邦学习设置的不同特征会影响对手的能力。例如，仅控制一个客户端的攻击者在跨设备设置中可能微不足道，但在跨竖井联合设置中可能会产生巨大影响。

在高层次上，对机器学习模型的对抗性攻击试图以某种不受欢迎的方式修改模型的行为。我们发现，攻击的目标一般是指不受欢迎的修改的范围或目标区域，范围一般有两个级别：

1.无目标攻击，或模型降级攻击，旨在降低模型的全局准确性，或“完全破坏”全局模型[69]。

2. 有针对性的攻击，或后门攻击，旨在改变模型在少数示例上的行为，同时保持所有其他示例的良好整体准确性[115，319，44，67]。

例如，在图像分类中，有针对性的攻击可能会向一组“绿色汽车”的训练图像添加一个小的视觉伪影（后门），以便使模型将这些标记为“鸟”。然后，经过训练的模型将学习将视觉伪影与类“鸟”相关联。这可以在以后被利用来进行简单的规避攻击，方法是将相同的视觉伪像添加到绿色汽车的任意图像中，以将其归类为“鸟”。模型甚至可以以一种不需要对目标推理时间输入进行任何修改的方式进行后门。Bagdasaryan等人[44]引入了“语义后门”，其中对手的模型更新迫使训练模型在一小部分数据上学习不正确的映射。例如，攻击者可能会迫使模型将所有绿色的汽车分类为鸟类，从而导致推理时的错误分类[44]。

虽然上文的讨论表明，无针对性攻击和有针对性攻击之间存在明显区别，但实际上，这些目标之间存在某种连续性。虽然纯粹的无目标攻击可能只会降低模型的准确性，但更细微的无目标攻击可能会降低所有客户端数据的模型准确性，但只有一小部分客户端数据除外。这反过来又开始类似于有针对性的攻击，其中后门的目的是相对于其余的评估数据，在少数示例上夸大模型的准确性。类似地，如果攻击者对数据的特定特征进行有针对性的攻击，而这些特征恰好存在于所有评估示例中，那么他们（可能无意中）制作了一个无针对性的攻击（相对于评估集）。虽然这种连续性对于理解对抗性攻击的格局很重要，但我们将在下面讨论纯目标或非目标攻击。

同时，当对手在训练过程中试图颠覆模型时，可能会拥有各种不同的能力。值得注意的是，联邦学习提出了关于对手可能具有哪些能力的各种各样的问题。

[9]在我们的环境中，无目标攻击和有目标攻击之间的区别不应与对抗性示例文献中使用的类似术语相混淆，这些术语用于区分针对任何错误分类或错误分类为特定目标类别的规避攻击。

| Characteristic | Description/Types |
|---|---|
| Attack vector | How the adversary introduces the attack.<br>• *Data poisoning*: the adversary alters the client datasets used to train the model.<br>• *Model update poisoning*: the adversary alters model updates sent to the server.<br>• *Evasion attack*: the adversary alters the data used at inference-time. |
| Model inspection | Whether the adversary can observe the model parameters.<br>• *Black box*: the adversary has no ability to inspect the parameters of the model before or during the attack. This is generally *not* the case in federated learning.<br>• *Stale whitebox*: the adversary can only inspect a stale version of the model. This naturally arises in the federated setting when the adversary has access to a client participating in an intermediate training round.<br>• *White box*: the adversary has the ability to directly inspect the parameters of the model. This can occur in cross-silo settings and in cross-device settings when an adversary has access to a large pool of devices likely to be chosen as participants. |
| Participant collusion | Whether multiple adversaries can coordinate an attack.<br>• *Non-colluding*: there is no capability for participants to coordinate an attack.<br>• *Cross-update collusion*: past client participants can coordinate with future participants on attacks to future updates to the global model.<br>• *Within-update collusion*: current client participants can coordinate on an attack to the current model update. |
| Participation rate | How often an adversary can inject an attack throughout training.<br>• In cross-device federated settings, a malicious client may only be able to participate in a *single model training round*.<br>• In cross-silo federated settings, an adversary may have *continuous participation* in the learning process. |
| Adaptability | Whether an adversary can alter the attack parameters as the attack progresses.<br>• *Static*: the adversary must fix the attack parameters at the start of the attack and cannot change them.<br>• *Dynamic*: the adversary can adapt the attack as training progresses. |

Table 11: Characteristics of an adversary's capabilities in federated settings.

| 特征描述/类型 | |
|---|---|
| 攻击向量对手如何发起攻击。 | • 数据中毒：攻击者改变用于训练模型的客户端数据集。<br>• 模型更新中毒：攻击者修改发送到服务器的模型更新。<br>• 规避攻击：攻击者在推理时修改数据。 |
| 模型检测对手是否能观察到模型参数。 | • 黑盒：攻击者在攻击之前或攻击过程中无法检查模型的参数。在联邦学习中通常不是这种情况。<br>• 陈旧白盒：对手只能检查模型的陈旧版本。在联邦环境中，当对手可以访问参与中间训练回合的客户端时，这自然会出现。<br>• 白色框：对手有能力直接检查模型的参数。这可能发生在跨竖井设置和跨设备设置中，当对手可以访问可能被选为参与者的大型设备池时。 |
| 参与者共谋多个对手是否可以协调攻击。 | • 非串通：参与者没有能力协调攻击。<br>• 交叉更新共谋：过去的客户端参与者可以与未来的参与者协调攻击全局模型的未来更新。<br>• 更新内共谋：当前客户端参与者可以协调对当前模型更新的攻击。 |
| 参与率在整个训练过程中，对手可以注入攻击的频率。 | ·在跨设备联合设置中，恶意客户端可能只能参与单个模型训练回合。<br>·在跨筒仓联合设置中，对手可能会持续参与学习过程。 |
| 适应性：攻击者是否可以随着攻击的进行而改变攻击参数。 | • 静态：攻击者必须在攻击开始时固定攻击参数，并且不能改变它们。<br>• 动态：对手可以随着训练的进行调整攻击。 |

表11：联合环境中对手能力的特征。

Clearly defining these capabilities is necessary for the community to weigh the value of proposed defenses. In Table 11, we propose a few axes of capabilities that are important to consider. We note that this is not a full list. There are many other characteristics of an adversary's capabilities that can be studied.

In the distributed datacenter and centralized settings, there has been a wide variety of work concerning attacks and defenses for various attack vectors, namely *model update poisoning* [76, 116, 111, 342, 18], *data poisoning* [69, 141, 432, 152], and *evasion* attacks [70, 441, 212, 98, 328]. As we will see, federated learning enhances the potency of many attacks, and increases the challenge of defending against these attacks. The federated setting shares a training-time poisoning attack vector with datacenter multi-machine learning: the model update sent from remote workers back to the shared model. This is potentially a powerful capability, as adversaries can construct malicious updates that achieve the exact desired effect, ignoring the prescribed client loss function or training scheme.

Another possible attack vector not discussed in Table 11 is the central aggregator itself. If an adversary can compromise the aggregator, then they can easily perform both targeted and untargeted attacks on the trained model [319]. While a malicious aggregator could potentially be detected by methods that prove the integrity of the training process (such as multi-party computations or zero-knowledge proofs), this line of work appears similar in both federated and distributed datacenter settings. We therefore omit discussion of this attack vector in the sequel.

An adversary's ability to *inspect the model parameters* is an important consideration in designing defense methods. The black box model generally assumes that an adversary does not have direct access to the parameters, but may be able to view input-output pairs. This setting is generally less relevant to federated learning: because the model is broadcast to all participants for local training, it is often assumed that an adversary has direct access to the model parameters (white box). Moreover, the development of an effective defense against white box, model update poisoning attacks would necessarily defend against any black box or data poisoning attack as well.

An important axis to evaluate in the context of specific federated settings (cross-device, cross-silo, etc.) is the capability of *participant collusion*. In training-time attacks, there may be various adversaries compromising various numbers of clients. Intuitively, the adversaries may be more effective if they are able to coordinate their poisoned updates than if they each acted individually. Perhaps worse for our poor federated learning defenses researcher, collusion may not be happening in "real time" (within-update collusion), but rather across model updates (cross-update collusion).

Some federated settings naturally lead to *limited participation rate*: with a population of hundreds of millions of devices, sampling a few thousand every update is unlikely to sample the same participant more than once (if at all) during the training process [81]. Thus, an adversary limited to a single client may only be able to inject a poisoned update a limited number of times. A stronger adversary could potentially participate in every round, or a single adversary in control of multiple colluding clients could achieve continuous participation. Alternatively, in the cross-silo federated setting in Table 1, most clients participate in each round. Therefore, adversaries may be more likely to have the capability to attack every round of cross-silo federated learning systems than they are to attack every round of cross-device settings.

Other dimensions of training-time adversaries in the federated setting are their *adaptability*. In a standard distributed datacenter training process, a malicious data provider is often limited to a static attack wherein the poisoned data is supplied once before training begins. In contrast, a malicious user with the ability to continuously participate in the federated setting could launch a poisoning attack throughout model training, where the user adaptively modifies training data or model updates as the training progresses. Note that in federated learning, this adaptivity is generally only interesting if the client can participate more than once throughout the training process.

明确定义这些能力对于社区衡量拟议防御的价值是必要的。在表11中，我们提出了几个需要考虑的重要功能轴。我们注意到，这不是一个完整的清单。对手的能力还有许多其他特征可以研究。

在分布式数据中心和集中式设置中，已经有各种各样的工作涉及各种攻击向量的攻击和防御，即模型更新中毒[76，116，111，342，18]，数据中毒[69，141，432，152]和逃避攻击[70，441，212，98，328]。正如我们将看到的，联邦学习增强了许多攻击的效力，并增加了防御这些攻击的挑战。联合设置与数据中心多机器学习共享一个训练时间中毒攻击向量：从远程工作者发送回共享模型的模型更新。这可能是一种强大的功能，因为攻击者可以构建恶意更新，以达到确切的预期效果，而忽略规定的客户端损失函数或训练方案。

表11中没有讨论的另一个可能的攻击媒介是中央聚合器本身。如果攻击者可以破坏聚合器，那么他们可以很容易地对训练模型进行有针对性和无针对性的攻击[319]。虽然可以通过证明训练过程完整性的方法（例如多方计算或零知识证明）来检测恶意聚合器，但这一工作在联合和分布式数据中心设置中似乎相似。因此，我们省略了在后续讨论这个攻击向量。

对手检查模型参数的能力是设计防御方法的重要考虑因素。黑盒模型通常假设对手不能直接访问参数，但可能能够查看输入输出对。此设置通常与联邦学习的相关性较低：因为模型被广播给所有参与者进行本地训练，因此通常假设攻击者可以直接访问模型参数（白色框）。此外，针对白色盒、模型更新中毒攻击的有效防御的开发也必然会防御任何黑盒或数据中毒攻击。

在特定联合设置（跨设备、跨思洛存储器等）的上下文中进行评估的重要轴是参与者合谋的能力。在训练时间攻击中，可能有各种各样的对手危害各种数量的客户端。直觉上，如果对手能够协调他们的中毒更新，而不是他们各自单独行动，那么他们可能会更有效。也许对我们可怜的联邦学习防御研究人员来说更糟糕的是，共谋可能不是发生在“真实的时间”（更新内共谋），而是发生在模型更新之间（交叉更新共谋）。

一些联合设置自然会导致有限的参与率：对于数亿设备的人群，每次更新采样几千个不太可能在训练过程中对同一参与者进行多次采样（如果有的话）。因此，限于单个客户端的攻击者可能只能注入有限次数的有毒更新。一个更强大的对手可能会参与每一轮，或者一个控制多个合谋客户的对手可以实现持续参与。或者，在表1中的跨竖井联邦设置中，大多数客户机参与每一轮。因此，对手可能更有能力攻击每一轮跨竖井联合学习系统，而不是攻击每一轮跨设备设置。

在联邦环境中，训练时间对手的其他方面是他们的适应性。在标准分布式数据中心训练过程中，恶意数据提供者通常限于静态攻击，其中在训练开始之前提供一次中毒数据。相比之下，能够持续参与联合设置的恶意用户可以在整个模型训练过程中发起中毒攻击，其中用户随着训练的进行自适应地修改训练数据或模型更新。请注意，在联邦学习中，这种自适应性通常只有在客户端可以在整个训练过程中多次参与时才有意义。

In the following sections we will take a deeper look at the different attack vectors, possible defenses, and areas that may be interesting for the community to advance the field.

### 5.1.2 Model Update Poisoning

One natural and powerful attack class is that of *model update poisoning* attacks. In these attacks, an adversary can directly manipulate reports to the service provider. In federated settings, this could be performed by corrupting the updates of a client directly, or some kind of man-in-the-middle attack. We assume direct update manipulation throughout this section, as this strictly enhances the capability of the adversary. Thus, we assume that the adversary (or adversaries) directly control some number of clients, and that they can directly alter the outputs of these clients to try to bias the learned model towards their objective.

**Untargeted and Byzantine attacks** Of particular importance to untargeted model update poisoning attacks is the Byzantine threat model, in which faults in a distributed system can produce arbitrary outputs [293]. Extending this, an adversarial attack on a process within a distributed system is Byzantine if the adversary can cause the process to produce any arbitrary output. Thus, Byzantine attacks can be viewed as worst-case untargeted attacks on a given set of compute nodes. Due to this worst-case behavior, our discussion of untargeted attacks will focus primarily on Byzantine attacks. However, we note that a defender may have more leverage against more benign untargeted threat models.

In the context of federated learning, we will focus on settings where an adversary controls some number of clients. Instead of sending locally updated models to the server, these Byzantine clients can send arbitrary values. This can result in convergence to sub-optimal models, or even lead to divergence [76]. If the Byzantine clients have white-box access to the model or non-Byzantine client updates, they may be able to tailor their output to have similar variance and magnitude as the correct model updates, making them difficult to detect. The catastrophic potential of Byzantine attacks has spurred line of work on Byzantine-resilient aggregation mechanisms for distributed learning [75, 111, 342, 18, 497, 152].

**Byzantine-resilient defenses** One popular defense mechanism against untargeted model update poisoning attacks, especially Byzantine attacks, replaces the averaging step on the server with a robust estimate of the mean, such as median-based aggregators [116, 497], Krum [76], and trimmed mean [497]. Past work has shown that various robust aggregators are provably effective for Byzantine-tolerant distributed learning [436, 76, 116] under appropriate assumptions, even in federated settings [379, 486, 427]. Despite this, Fang et al. [183] recently showed that multiple Byzantine-resilient defenses did little to defend against model poisoning attacks in federated learning. Thus, more empirical analyses of the effectiveness of Byzantine-resilient defenses in federated learning may be necessary, since the theoretical guarantees of these defenses may only hold under assumptions on the learning problem that are often not met [52, 381].

Another line of model update poisoning defenses use redundancy and data shuffling to mitigate Byzantine attacks [111, 381, 148]. While often equipped with rigorous theoretical guarantees, such mechanisms generally assume the server has direct access to the data or is allowed to globally shuffle the data, and therefore are not directly applicable in federated settings. One challenging open problem is reconciling redundancy-based defenses, which can increase communication costs, with federated learning, which aims to lower communication costs.

**Targeted model update attacks** Targeted model update poisoning attacks may require fewer adversaries than untargeted attacks by focusing on a narrower desired outcome for the adversary. In such attacks, even

在下面的部分中，我们将深入了解不同的攻击媒介、可能的防御以及社区可能感兴趣的领域，以推动该领域的发展。

### 5.1.2模型更新中毒

一种自然而强大的攻击类型是模型更新中毒攻击。在这些攻击中，攻击者可以直接操纵给服务提供商的报告。在联邦设置中，这可以通过直接破坏客户端的更新或某种中间人攻击来执行。在本节中，我们假设直接更新操作，因为这严格增强了对手的能力。因此，我们假设对手（或多个对手）直接控制一定数量的客户端，并且他们可以直接改变这些客户端的输出，以尝试将学习模型偏向于他们的目标。

非目标和拜占庭攻击对非目标模型更新中毒攻击特别重要的是拜占庭威胁模型，其中分布式系统中的故障可以产生任意输出[293]。扩展这一点，如果对手可以导致进程产生任何任意输出，则对分布式系统中的进程的对抗性攻击是拜占庭式的。因此，拜占庭攻击可以被视为对给定计算节点集合的最坏情况的无目标攻击。由于这种最坏情况的行为，我们对非目标攻击的讨论将主要集中在拜占庭攻击上。然而，我们注意到防御者可能对更良性的非目标威胁模型有更多的影响力。

在联邦学习的背景下，我们将重点关注对手控制一定数量客户端的设置。这些拜占庭客户端可以发送任意值，而不是将本地更新的模型发送到服务器。这可能导致收敛到次优模型，甚至导致发散[76]。如果Byzantine客户端可以白盒访问模型或非Byzantine客户端更新，则它们可以定制其输出，使其具有与正确模型更新相似的方差和幅度，从而使其难以检测。拜占庭攻击的灾难性潜力刺激了分布式学习的拜占庭弹性聚合机制的工作[75，111，342，18，497，152]。

拜占庭弹性防御一种针对非目标模型更新中毒攻击（尤其是拜占庭攻击）的流行防御机制，用对平均值的鲁棒估计取代了服务器上的平均步骤，例如基于中位数的聚合器[116，497]，克鲁姆[76]和修剪平均值[497]。过去的工作表明，在适当的假设下，各种强大的聚合器对于拜占庭容忍的分布式学习[436，76，116]是有效的，即使在联邦设置[379，486，427]。尽管如此，Fang等人[183]最近表明，在联邦学习中，多个拜占庭弹性防御对于防御模型中毒攻击几乎没有作用。因此，可能有必要对联邦学习中拜占庭弹性防御的有效性进行更多的实证分析，因为这些防御的理论保证可能只在学习问题的假设下才成立，而这些假设往往无法满足[52，381]。

另一种模型更新中毒防御使用冗余和数据重排来减轻拜占庭攻击[111，381，148]。虽然通常配备了严格的理论保证，但这种机制通常假设服务器可以直接访问数据或允许全局洗牌数据，因此不能直接应用于联邦设置。一个具有挑战性的开放问题是协调基于冗余的防御，这可能会增加通信成本，而联邦学习旨在降低通信成本。

有针对性的模型更新攻击有针对性的模型更新中毒攻击可能需要比非有针对性的攻击更少的对手，因为它专注于对手更窄的期望结果。在这样的攻击中，

a single-shot attack may be enough to introduce a backdoor into a model [44]. Bhagoji et al. [67] shows that if $10\%$ of the devices participating in federated learning are compromised, a backdoor can be introduced by poisoning the model sent back to the service provider, even with the presence of anomaly detectors at the server. Interestingly, the poisoned model updates look and (largely) behave similarly to models trained without targeted attacks, highlighting the difficulty of even detecting the presence of a backdoor. Moreover, since the adversary's aim is to only affect the classification outcome on a small number of data points, while maintaining the overall accuracy of the centrally learned model, defenses for untargeted attacks often fail to address targeted attacks [67, 44]. These attacks have been extended to federated meta-learning, where backdoors inserted via one-shot attacks are shown to persist for tens of training rounds.[109].

Existing defenses against backdoor attacks [432, 314, 454, 152, 465, 416, 122] either require a careful examination of the training data, access to a holdout set of similarly distributed data, or full control of the training process at the server, none of which may hold in the federated learning setting. An interesting avenue for future work would be to explore the use of zero-knowledge proofs to ensure that users are submitting updates with pre-specified properties. Solutions based on hardware attestation could also be considered. For instance, a user's mobile phone might have the ability to attest that the shared model updates were computed correctly using images produced by the phone's camera.

**Collusion defenses** Model update poisoning attacks may drastically increase in effectiveness if the adversaries are allowed to collude. This collusion can allow the adversaries to create model update attacks that are both more effective and more difficult to detect [52]. This paradigm is strongly related to sybil attacks [160], in which clients are allowed to join and leave the system at will. Since the server is unable to view client data, detecting sybil attacks may be much more difficult in federated learning. Recent work has shown that federated learning is vulnerable to both targeted and untargeted sybil attacks [190]. Potential challenges for federated learning involve defending against collusion or detecting colluding adversaries, without directly inspecting the data of nodes.

### 5.1.3 Data Poisoning Attacks

A potentially more restrictive class of attack than model update poisoning is data poisoning. In this paradigm, the adversary cannot directly corrupt reports to the central node. Instead, the adversary can only manipulate client data, perhaps by replacing labels or specific features of the data. As with model update poisoning, data poisoning can be performed both for targeted attacks [69, 115, 275] and untargeted attacks [319, 44].

This attack model may be more natural when the adversary can only influence the data collection process at the edge of the federated learning system, but cannot directly corrupt derived quantities within the learning system (e.g. model updates).

**Data poisoning and Byzantine-robust aggregation** Since data poisoning attacks induce model update poisoning, any defense against Byzantine updates can also be used to defend against data poisoning. For example Xie et al. [487], Xie [484] and Xie et al. [486] proposed Byzantine-robust aggregators that successfully defended against label-flipping data poisoning attacks on convolutional neural networks. As discussed in Section 5.1.2, one important line of work involves analyzing and improving these approaches in federated learning. Non-IID data and unreliability of clients all present serious challenges and disrupt common assumptions in works on Byzantine-robust aggregation. For data poisoning, there is a possibility that the Byzantine threat model is too strong. By restricting to data poisoning (instead of general model update poisoning), it may be possible to design a more tailored and effective Byzantine-robust aggregator. We discuss

单次攻击可能足以在模型中引入后门[44]。Bhagoji等人[67]表明，如果参与联邦学习的设备中有10%受到损害，即使服务器上存在异常检测器，也可以通过中毒发送回服务提供商的模型来引入后门。有趣的是，中毒模型更新的外观和（很大程度上）行为与没有针对性攻击的模型相似，这突出了检测后门存在的难度。此外，由于对手的目标是只影响少量数据点的分类结果，同时保持集中学习模型的整体准确性，因此针对非目标攻击的防御通常无法解决目标攻击[67，44]。这些攻击已经扩展到联邦元学习，其中通过一次性攻击插入的后门被证明可以持续数十轮训练。[109].

针对后门攻击的现有防御措施[432，314，454，152，465，416，122]要么需要仔细检查训练数据，访问类似分布的数据集，要么在服务器上完全控制训练过程，这些都不适用于联合学习设置。未来工作的一个有趣途径是探索使用零知识证明来确保用户提交具有预先指定属性的更新。也可以考虑基于硬件认证的解决方案。例如，用户的移动的电话可能具有使用由电话的相机产生的图像来证明共享模型更新被正确计算的能力。

如果允许对手合谋，模型更新中毒攻击的有效性可能会大幅增加。这种共谋可以允许对手创建更有效且更难以检测的模型更新攻击[52]。这种模式与Sybil攻击密切相关[160]，其中允许客户端随意加入和离开系统。由于服务器无法查看客户端数据，因此在联邦学习中检测sybil攻击可能要困难得多。最近的研究表明，联邦学习容易受到有针对性和无针对性的Sybil攻击[190]。联邦学习的潜在挑战包括防御共谋或检测共谋的对手，而不直接检查节点的数据。

### 5.1.3数据中毒攻击

一种比模型更新中毒更具潜在限制性的攻击类型是数据中毒。在这个范例中，对手不能直接破坏到中央节点的报告。相反，攻击者只能操纵客户端数据，可能是通过替换数据的标签或特定特征。与模型更新中毒一样，数据中毒可以针对目标攻击[69，115，275]和非目标攻击[319，44]执行。

当攻击者只能影响联合学习系统边缘的数据收集过程，但不能直接破坏学习系统中的导出量（例如模型更新）时，这种攻击模型可能更自然。

由于数据中毒攻击会导致模型更新中毒，因此任何针对拜占庭更新的防御也可以用于防御数据中毒。例如，Xie et al. [487]，Xie [484]和Xie et al. [486]提出了Byzantine-robust聚合器，成功抵御了卷积神经网络上的标签翻转数据中毒攻击。正如5.1.2节所讨论的，一个重要的工作涉及分析和改进联邦学习中的这些方法。非IID数据和客户端的不可靠性都带来了严重的挑战，并破坏了拜占庭式强大聚合工作中的共同假设。对于数据中毒，拜占庭威胁模型可能太强。通过限制数据中毒（而不是一般的模型更新中毒），它可能会设计一个更适合和有效的拜占庭鲁棒聚合器。我们讨论

this in more detail in at the end of Section 5.1.3.

**Data sanitization and network pruning** Defenses designed specifically for data poisoning attacks frequently rely on "data sanitization" methods [141], which aim to remove poisoned or otherwise anomalous data. More recent work has developed improved data sanitization methods using robust statistics [432, 416, 454, 152], which often have the benefit of being provably robust to small numbers of outliers [152]. Such methods can be applied to both targeted and untargeted attacks, with some degree of empirical success [416].

A related class of defenses used for defending against backdoor attacks are "pruning" defenses. Rather than removing anomalous data, pruning defenses attempt to remove activation units that are inactive on clean data [314, 465]. Such methods are motivated by previous studies which showed empirically that poisoned data designed to introduce a backdoor often triggers so-called "backdoor neurons" [214]. While such methods do not require direct access to all client data, they require "clean" holdout data that is representative of the global dataset.

Neither data sanitization nor network pruning work directly in federated settings, as they both generally require access to client data, or else data that resembles client data. Thus, it is an open question whether data sanitization methods and network pruning methods can be used in federated settings without privacy loss, or whether or not defenses against data poisoning require new federated approaches. Furthermore, Koh et al. [276] recently showed that many heuristic defenses based on data sanitization remain vulnerable to adaptive poisoning attacks, suggesting that even a federated approach to data sanitization may not be enough to defend against data poisoning.

Even detecting the presence of poisoned data (without necessarily correcting for it or identifying the client with poisoned data) is challenging in federated learning. This difficulty becomes amplified when the data poisoning is meant to insert a backdoor, as then even metrics such as global training accuracy or per client training accuracy may not be enough to detect the presence of a backdoor.

**Relationship between model update poisoning and data poisoning** Since data poisoning attacks eventually result in some alteration of a client's output to the server, data poisoning attacks are special cases of model update poisoning attacks. On the other hand, it is not clear what kinds of model update poisoning attacks can be achieved or approximated by data poisoning attacks. Recent work by Bhagoji et al. [67] suggests that data poisoning may be weaker, especially in settings with limited *participation rate* (see Table 11). One interesting line of study would be to quantify the gap between these two types of attacks, and relate this gap to the relative strength of an adversary operating under these attack models. While this question can be posed independently of federated learning, it is particularly important in federated learning due to differences in adversary capabilities (see Table 11). For example, the maximum number of clients that can perform data poisoning attacks may be much higher than the number that can perform model update poisoning attacks, especially in cross-device settings. Thus, understanding the relation between these two attack types, especially as they relate to the number of adversarial clients, would greatly help our understanding of the threat landscape in federated learning.

This problem can be tackled in a variety of manners. Empirically, one could study the discrepancy in performance of various attacks. or investigate whether various model update poisoning attacks can be approximated by data poisoning attacks, and would develop methods for doing so. Theoretically, although we conjecture that model update poisoning is provably stronger than data poisoning, we are unaware of any formal statements addressing this. One possible approach would be to use insights and techniques from work on machine teaching (see [511] for reference) to understand "optimal" data poisoning attacks, as in [340]. Any formal statement will likely depend on quantities such as the number of corrupted clients and

这一点在第5.1.3节末尾有更详细的说明。

数据清理和网络修剪专门针对数据中毒攻击设计的防御通常依赖于“数据清理”方法[141]，旨在删除中毒或异常数据。最近的工作开发了使用稳健统计的改进数据净化方法[432，416，454，152]，其通常具有可证明对少量离群值具有稳健性的好处[152]。这些方法可以应用于有针对性和无针对性的攻击，并取得了一定程度的经验成功[416]。

用于防御后门攻击的相关防御类别是“修剪”防御。修剪防御不是删除异常数据，而是尝试删除在干净数据上不活动的激活单元[314，465]。这些方法的动机是以前的研究，这些研究经验表明，旨在引入后门的有毒数据通常会触发所谓的“后门神经元”[214]。虽然这些方法不需要直接访问所有客户端数据，但它们需要代表全局数据集的“干净”保持数据。

数据清理和网络修剪都不能直接在联邦设置中工作，因为它们通常都需要访问客户端数据或类似于客户端数据的数据。因此，数据清理方法和网络修剪方法是否可以在不丢失隐私的情况下用于联邦设置，或者是否需要新的联邦方法来防御数据中毒，这是一个悬而未决的问题。此外，Koh等人[276]最近表明，许多基于数据清理的启发式防御仍然容易受到自适应中毒攻击，这表明即使是数据清理的联邦方法也可能不足以防御数据中毒。

即使检测到中毒数据的存在（不一定纠正它或识别具有中毒数据的客户端）也是联邦学习的挑战。当数据中毒旨在插入后门时，这种困难变得更大，因为即使是全局训练准确性或每个客户端训练准确性等指标也可能不足以检测后门的存在。

模型更新中毒和数据中毒之间的关系由于数据中毒攻击最终会导致客户端到服务器的输出发生某些更改，因此数据中毒攻击是模型更新中毒攻击的特殊情况。另一方面，目前还不清楚什么样的模型更新中毒攻击可以实现或近似数据中毒攻击。Bhagoji等人[67]最近的研究表明，数据中毒可能较弱，特别是在参与率有限的环境中（见表11）。一个有趣的研究方向是量化这两种类型攻击之间的差距，并将这种差距与在这些攻击模型下运行的对手的相对实力联系起来。虽然这个问题可以独立于联邦学习提出，但由于对手能力的差异，它在联邦学习中特别重要（见表11）。 例如，可以执行数据中毒攻击的最大客户端数量可能远远高于可以执行模型更新中毒攻击的数量，特别是在跨设备设置中。因此，了解这两种攻击类型之间的关系，特别是它们与对抗性客户端数量的关系，将极大地帮助我们了解联邦学习中的威胁格局。

这个问题可以用各种方式来解决。从经验上讲，人们可以研究各种攻击的性能差异。或者研究各种模型更新中毒攻击是否可以近似为数据中毒攻击，并且将开发用于这样做的方法。从理论上讲，虽然我们推测模型更新中毒可以证明比数据中毒更强，但我们不知道有任何正式的声明可以解决这个问题。一种可能的方法是使用机器教学工作中的见解和技术（参考[511]）来理解“最佳”数据中毒攻击，如[340]。任何正式的声明都可能依赖于数量，例如损坏的客户端的数量，

the function class of interest. Intuitively, the relation between model update poisoning and data poisoning should depend on the overparameterization of the model with respect to the data.

### 5.1.4 Inference-Time Evasion Attacks

In evasion attacks, an adversary may attempt to circumvent a deployed model by carefully manipulating samples that are fed into the model. One well-studied form of evasion attacks are so-called "adversarial examples." These are perturbed versions of test inputs which seem almost indistinguishable from the original test input to a human, but fool the trained model [70, 441]. In image and audio domains, adversarial examples are generally constructed by adding norm-bounded perturbations to test examples, though more recent works explore other distortions [176, 477, 259]. In the white-box setting, the aforementioned perturbations can be generated by attempting to maximize the loss function subject to a norm constraint via constrained optimization methods such as projected gradient ascent [284, 328]. Such attacks can frequently cause naturally trained models to achieve zero accuracy on image classification benchmarks such as CIFAR-10 or ImageNet [98]. In the black-box setting, models have also been shown to be vulnerable to attacks based on query-access to the model [113, 90] or based on substitute models trained on similar data [441, 366, 452]. While black-box attacks may be more natural to consider in datacenter settings, the model broadcast step in federated learning means that the model may be accessible to any malicious client. Thus, federated learning increases the need for defenses against white-box evasion attacks.

Various methods have been proposed to make models more robust to evasion attacks. Here, robustness is often measured by the model performance on white-box adversarial examples. Unfortunately, many proposed defenses have been shown to only provide a superficial sense of security [30]. On the other hand, adversarial training, in which a robust model is trained with adversarial examples, generally provides some robustness to white-box evasion attacks [328, 483, 412]. Adversarial training is often formulated as a minimax optimization problem, where the adversarial examples and the model weights are alternatively updated. We note that there is no canonical formulation of adversarial training, and choices such as the minimax optimization problem and hyperparameters such as learning rate can significantly affect the model robustness, especially for large-scale dataset like ImageNet. Moreover, adversarial training typically only improves robustness to the specific type of adversarial examples incorporated during training, potentially leaving the trained model vulnerable to other forms of adversarial noise [176, 448, 414].

Adapting adversarial training methods to federated learning brings a host of open questions. For example, adversarial training can require many epochs before obtaining significant robustness. However, in federated learning, especially cross-device federated learning, each training sample may only be seen a limited number of times. More generally, adversarial training was developed primarily for IID data, and it is unclear how it performs in non-IID settings. For example, setting appropriate bounds on the norm of perturbations to perform adversarial training (a challenging problem even in the IID setting [453]) becomes harder in federated settings where the training data cannot be inspected ahead of training. Another issue is that generating adversarial examples is relatively expensive. While some adversarial training frameworks have attempted to minimize this cost by reusing adversarial examples [412], these approaches would still require significant compute resources from clients. This is potentially problematic in cross-device settings, where adversarial example generation may exacerbate memory or power constraints. Therefore, new on-device robust optimization techniques may be required in the federated learning setting.

**Relationship between training-time and inference-time attacks** The aforementioned discussion of evasion attacks generally assumes the adversary has white-box access (potentially due to systems-level realities of federated learning) at inference time. This ignores the reality that an adversary could corrupt the training

感兴趣的函数类。直觉上，模型更新中毒和数据中毒之间的关系应该取决于模型相对于数据的过参数化。

### 5.1.4推理时间规避攻击

在规避攻击中，攻击者可能会试图通过仔细操纵输入模型的样本来规避部署的模型。一种被充分研究的逃避攻击形式是所谓的“对抗性示例”。这些是测试输入的扰动版本，似乎与人类的原始测试输入几乎无法区分，但欺骗了训练模型[70，441]。在图像和音频领域，对抗性示例通常通过向测试示例添加范数有界扰动来构建，尽管最近的作品探索了其他失真[176，477，259]。在白盒设置中，上述扰动可以通过尝试最大化损失函数来生成，该损失函数通过约束优化方法（如投影梯度上升）受到范数约束[284，328]。这种攻击经常会导致自然训练的模型在CIFAR-10或ImageNet等图像分类基准上达到零精度[98]。 在黑盒设置中，模型也被证明容易受到基于对模型的查询访问的攻击[113，90]或基于在类似数据上训练的替代模型的攻击[441，366，452]。虽然在数据中心设置中考虑黑盒攻击可能更自然，但联邦学习中的模型广播步骤意味着任何恶意客户端都可以访问模型。因此，联邦学习增加了对白盒规避攻击的防御需求。

已经提出了各种方法来使模型对规避攻击更加鲁棒。在这里，鲁棒性通常是通过白盒对抗示例的模型性能来衡量的。不幸的是，许多提议的防御措施已被证明只能提供表面的安全感[30]。另一方面，对抗性训练，其中使用对抗性示例训练鲁棒模型，通常为白盒规避攻击提供一定的鲁棒性[328，483，412]。对抗性训练通常被表述为极大极小优化问题，其中对抗性示例和模型权重交替更新。我们注意到，对抗训练没有规范的公式，诸如极大极小优化问题和学习率等超参数的选择可以显着影响模型的鲁棒性，特别是对于像ImageNet这样的大规模数据集。 此外，对抗性训练通常只提高了对训练过程中包含的特定类型对抗性示例的鲁棒性，可能会使训练后的模型容易受到其他形式的对抗性噪声的影响[176，448，414]。

将对抗性训练方法应用于联邦学习带来了一系列悬而未决的问题。例如，对抗性训练可能需要许多时期才能获得显著的鲁棒性。然而，在联合学习中，特别是跨设备联合学习中，每个训练样本只能被看到有限的次数。更一般地说，对抗训练主要是针对IID数据开发的，目前还不清楚它在非IID环境中的表现。例如，在联邦设置中，设置适当的扰动范数界限以执行对抗训练（即使在IID设置中也是一个具有挑战性的问题[453]），在联邦设置中，训练数据无法在训练之前进行检查。另一个问题是生成对抗性示例相对昂贵。虽然一些对抗性训练框架试图通过重用对抗性示例来最大限度地减少这种成本[412]，但这些方法仍然需要来自客户端的大量计算资源。 这在跨设备设置中可能存在问题，其中对抗性示例生成可能会加剧内存或功率约束。因此，在联合学习设置中可能需要新的设备上鲁棒优化技术。

上述关于规避攻击的讨论通常假设对手在推理时具有白盒访问（可能是由于联邦学习的系统级现实）。这忽略了一个现实，即对手可能会破坏训练

process in order to create or enhance inference-time vulnerabilities of a model, as in [115]. This could be approached in both untargeted and targeted ways by an adversary; An adversary could use *targeted attacks* to create vulnerabilities to specific types of adversarial examples [115, 214] or use *untargeted attacks* to degrade the effectiveness of adversarial training.

One possible defense against combined training- and inference-time adversaries are methods to detect backdoor attacks [454, 108, 465, 122]. Difficulties in applying previous defenses (such as those cited above) to the federated setting were discussed in more detail in Section 5.1.3. However, purely detecting backdoors may be insufficient in many federated settings where we want robustness guarantees on the output model at inference time. More sophisticated solutions could potentially combine training-time defenses (such as robust aggregation or differential privacy) with adversarial training. Other open work in this area could involve quantifying how various types of training-time attacks impact the inference-time vulnerability of a model. Given the existing challenges in defending against purely training-time or purely inference-time attacks, this line of work is necessarily more speculative and unexplored.

### 5.1.5 Defensive Capabilities from Privacy Guarantees

Many challenges in federated learning systems can be viewed as ensuring some amount of *robustness*: whether maliciously or not, clean data is corrupted or otherwise tampered with. Recent work on data privacy, notably *differential privacy* (DP) [167], defines privacy in terms of robustness. In short, random noise is added at training or test time in order to reduce the influence of specific data points. For a more detailed explanation on differential privacy, see Section 4.2.2. As a defense technique, differential privacy has several compelling strengths. First, it provides strong, worst-case protections against a variety of attacks. Second, there are many known differentially private algorithms, and the defense can be applied to many machine learning tasks. Finally, differential privacy is known to be closed under composition, where the inputs to later algorithms are determined after observing the results of earlier algorithms.

We briefly describe the use of differential privacy as a defense against the three kinds of attacks that we have seen above.

**Defending against model update poisoning attacks** The service provider can bound the contribution of any individual client to the overall model by (1) enforcing a norm constraint on the client model update (e.g. by clipping the client updates), (2) aggregating the clipped updates, (3) and adding Gaussian noise to the aggregate. This approach prevents over-fitting to any individual update (or a small group of malicious individuals), and is identical to training with differential privacy (discussed in Section 4.3.2). This approach has been recently explored by Sun et al. [438], which shows preliminary success in applying differential privacy as a defense against targeted attacks. However, the scope of experiments and targeted attacks analyzed by Sun et al. [438] should be extended to include more general adversarial attacks. In particular, Wang et al. [466], show that the use of edge case backdoors, generated from data samples with low probability in the underlying distribution, is able to bypass differential privacy defenses. They further demonstrate that the existence of adversarial examples implies the existence of edge-case backdoors, indicating that defenses for the two threats may need to be developed in tandem. Therefore, more work remains to verify whether or not DP can indeed be an effective defense. More importantly, it is still unclear how hyperparameters for DP (such as the size of $\ell_2$ norm bounds and noise variance) can be chosen as a function of the model size and architecture, as well as the fraction of malicious devices.

这忽略了一个事实，即对手可能会破坏训练过程，以创建或增强模型的推理时间漏洞，如[115]。攻击者可以通过无目标和有目标的方式来实现这一点;攻击者可以使用有目标的攻击来创建针对特定类型的对抗性示例的漏洞[115，214]，或者使用无目标的攻击来降低对抗性训练的有效性。

针对组合训练和推理时间攻击者的一种可能的防御方法是检测后门攻击[454，108，465，122]。在5.1.3节中更详细地讨论了将以前的防御（例如上面引用的防御）应用于联邦环境的困难。然而，在许多联邦设置中，纯粹检测后门可能是不够的，我们希望在推理时保证输出模型的鲁棒性。更复杂的解决方案可能会将联合收割机训练时间防御（如鲁棒聚合或差分隐私）与对抗训练相结合。该领域的其他开放工作可能涉及量化各种类型的训练时间攻击如何影响模型的推理时间漏洞。考虑到在防御纯训练时间或纯推理时间攻击方面存在的挑战，这方面的工作必然更具投机性和未开发性。

### 5.1.5来自隐私保证的防御能力

联邦学习系统中的许多挑战可以被视为确保一定程度的鲁棒性：无论是否恶意，干净的数据都会被破坏或篡改。最近关于数据隐私的工作，特别是差分隐私（DP）[167]，从鲁棒性的角度定义了隐私。简而言之，在训练或测试时添加随机噪声，以减少特定数据点的影响。有关差异隐私的更详细解释，请参见第4.2.2节。作为一种防御技术，差异隐私有几个令人信服的优势。首先，它提供了强大的，最坏情况下的保护，以抵御各种攻击。其次，有许多已知的差分隐私算法，并且防御可以应用于许多机器学习任务。最后，差分隐私已知在组合下是封闭的，其中在观察早期算法的结果之后确定后期算法的输入。

我们简要描述了使用差分隐私来防御我们上面看到的三种攻击。

防御模型更新中毒攻击服务提供商可以通过以下方式将任何单个客户端的贡献绑定到整个模型：（1）对客户端模型更新强制执行范数约束（例如，通过裁剪客户端更新），（2）聚合裁剪的更新，（3）并向聚合添加高斯噪声。这种方法可以防止对任何个人更新（或一小群恶意个人）的过度拟合，并且与使用差分隐私的训练相同（在第4.3.2节中讨论）。Sun等人最近探索了这种方法。[438]，该方法在应用差分隐私作为针对目标攻击的防御方面取得了初步成功。然而，Sun等人分析的实验和目标攻击的范围应该扩展到包括更一般的对抗性攻击。特别地，Wang et al. [466]，表明使用边缘情况后门，从底层分布中具有低概率的数据样本中生成，能够绕过差异隐私防御。他们进一步证明，对抗性示例的存在意味着边缘情况后门的存在，这表明可能需要同时开发对这两种威胁的防御。因此，还有更多的工作要做，以验证DP是否真的可以成为一个有效的防御。更重要的是，目前还不清楚如何选择DP的超参数（例如“范数边界和噪声方差的大小）作为模型大小和架构的函数，以及恶意设备的比例。

**Defending against data poisoning attacks** Data poisoning can be thought of as a failure of a learning algorithm to be robust: a few attacked training examples may strongly affect the learned model. Thus, one natural way to defend against these attacks is to make the learning algorithm differentially private, improving robustness. Recent work has explored differential privacy as a defense against data poisoning [326], and in particular in the federated learning context [199]. Intuitively, an adversary who is only able to modify a few training examples cannot cause a large change in the distribution over learned models.

While differential privacy is a flexible defense against data poisoning, it also has some drawbacks. The main weakness is that noise must be injected into the learning procedure. While this is not necessarily a problem—common learning algorithms like stochastic gradient descent already inject noise—the added noise can hurt the performance of the learned model. Furthermore, the adversary can only control a small number of devices.[10] Accordingly, differential privacy can be viewed as both a strong and a weak defense against data poisoning—it is strong in that it is extremely general and provides worst case protection no matter the goals of the adversary, and it is weak in that the adversary must be restricted and noise must be added to the federated learning process.

**Defending against inference-time evasion attacks** Differential privacy has also been studied as a defense against inference-time attacks, where the adversary may modify test examples to manipulate the learned model. A straightforward approach is to make the predictor itself differentially private; however, this has the drawback that prediction becomes randomized, a usually undesirable feature that can also hurt interpretability. More sophisticated approaches [296] add noise and then release the prediction with the highest probability. We believe that there are other opportunities for further exploration in this direction.

## 5.2 Non-Malicious Failure Modes

Compared to datacenter training, federated learning is particularly susceptible to non-malicious failures from unreliable clients outside the control of the service provider. Just as with adversarial attacks, systems factors and data constraints also exacerbate non-malicious failures present in datacenter settings. We also note that techniques (described in the following sections) which are designed to address worst-case adversarial robustness are also able to effectively address non-malicious failures. While non-malicious failures are generally less damaging than malicious attacks, they are potentially more common, and share common roots and complications with the malicious attacks. We therefore expect progress in understanding and guarding against non-malicious failures to also inform defenses against malicious attacks.

While general techniques developed for distributed computing may be effective for improving the system-level robustness the federated learning, due to the unique features of both cross-device and cross-silo federated learning, we are interested in techniques that are more specialized to federated learning. Below we discuss three possible non-malicious failure modes in the context of federated learning: client reporting failures, data pipeline failures, and noisy model updates. We also discuss potential approaches to making federated learning more robust to such failures.

**Client reporting failures** Recall that in federated learning, each training round involves broadcasting a model to the clients, local client computation, and client reports to the central aggregator. For any participating client, systems factors may cause failures at any of these steps. Such failures are especially likely in cross-device federated learning, where network bandwidth becomes more of a constraint, and the client

[10] Technically, robustness to poisoning multiple examples is derived from the group privacy property of differential privacy; this protection degrades exponentially as the number of attacked points increases.

防御数据中毒攻击数据中毒可以被认为是学习算法鲁棒性的失败：一些受攻击的训练示例可能会强烈影响学习的模型。因此，防御这些攻击的一种自然方法是使学习算法具有差异隐私性，从而提高鲁棒性。最近的工作已经探索了差异隐私作为对数据中毒的防御[326]，特别是在联邦学习环境中[199]。直觉上，一个只能够修改一些训练样本的对手不能导致学习模型的分布发生大的变化。

虽然差异隐私是一种灵活的防御数据中毒的方法，但它也有一些缺点。其主要缺点是必须将噪声注入到学习过程中。虽然这不一定是一个问题-常见的学习算法，如随机梯度下降已经注入噪声-添加的噪声可能会损害学习模型的性能。此外，攻击者只能控制少数设备，因此，差分隐私可以被视为对数据中毒的强防御和弱防御-它是强的，因为它是非常普遍的，并提供最坏情况下的保护，无论攻击者的目标，它是弱的，攻击者必须受到限制，噪声必须添加到联邦学习过程。

差分隐私也被研究作为对推理时间攻击的防御，其中对手可以修改测试示例以操纵学习的模型。一个简单的方法是使预测器本身具有差异性;然而，这有一个缺点，即预测变得随机化，这通常是一个不受欢迎的特性，也会损害可解释性。更复杂的方法[296]添加噪音，然后以最高的概率发布预测。我们认为，在这方面还有其他进一步探索的机会。

## 5.2非恶意故障模式

与数据中心培训相比，联合学习特别容易受到来自服务提供商控制之外的不可靠客户端的非恶意故障的影响。与对抗性攻击一样，系统因素和数据约束也会加剧数据中心设置中存在的非恶意故障。我们还注意到，旨在解决最坏情况对抗鲁棒性的技术（在以下部分中描述）也能够有效地解决非恶意故障。虽然非恶意故障通常比恶意攻击的破坏性小，但它们可能更常见，并且与恶意攻击具有共同的根源和并发症。因此，我们期望在理解和防范非恶意故障方面取得进展，以通知针对恶意攻击的防御措施。

虽然为分布式计算开发的一般技术可能有效地提高了联邦学习的系统级鲁棒性，但由于跨设备和跨竖井联邦学习的独特功能，我们对更专门用于联邦学习的技术感兴趣。下面我们将讨论联邦学习环境中三种可能的非恶意故障模式：客户端报告故障、数据管道故障和噪声模型更新。我们还讨论了使联邦学习对此类故障更具鲁棒性的潜在方法。

回想一下，在联邦学习中，每一轮训练都涉及向客户端广播模型，本地客户端计算，以及向中央聚合器的客户端报告。对于任何参与的客户端，系统因素可能会导致这些步骤中的任何一个失败。这种失败在跨设备联合学习中尤其可能发生，在这种情况下，网络带宽变得更加受限，客户端

[10]从技术上讲，对中毒多个示例的鲁棒性来自差分隐私的组隐私属性;这种保护随着攻击点数量的增加而呈指数级下降。

devices are more likely to be edge devices with limited compute power. Even if there is no explicit failure, there may be straggler clients, which take much longer to report their output than other nodes in the same round. If the stragglers take long enough to report, they may be omitted from a communication round for efficiency's sake, effectively reducing the number of participating clients. In "vanilla" federated learning, this requires no real algorithmic changes, as federated averaging can be applied to whatever clients report model updates.

Unfortunately, unresponsive clients become more challenging to contend with when using secure aggregation (SecAgg) [80, 58], especially if the clients drop out during the SecAgg protocol. While SecAgg is designed to be robust to significant numbers of dropouts [81], there is still the potential for failure. The likelihood of failure could be reduced in various complementary ways. One simple method would be to select more devices than required within each round. This helps ensure that stragglers and failed devices have minimal effect on the overall convergence [81]. However, in unreliable network settings, this may not be enough. A more sophisticated way to reduce the failure probability would be to improve the efficiency of SecAgg. This reduces the window of time during which client dropouts would adversely affect SecAgg. Another possibility would be to develop an asynchronous version of SecAgg that does not require clients to participate during a fixed window of time, possibly by adapting techniques from general asynchronous secure multi-party distributed computation protocols [430]. More speculatively, it may be possible to perform versions of SecAgg that aggregate over multiple computation rounds. This would allow straggler nodes to be included in subsequent rounds, rather than dropping out of the current round altogether.

**Data pipeline failures** While data pipelines in federated learning only exist within each client, there are still many potential issues said pipelines can face. In particular, any federated learning system still must define how raw user data is accessed and preprocessed in to training data. Bugs or unintended actions in this pipeline can drastically alter the federated learning process. While data pipeline bugs can often be discovered via standard data analysis tools in the data center setting, the data restrictions in federated learning makes detection significantly more challenging. For example, feature-level preprocessing issues (such as inverting pixels, concatenating words, etc.) can not be directly detected by the server [31]. One possible solution is to train generative models using federated methods with differential privacy, and then using these to synthesize new data samples that can be used to debug the underlying data pipelines [31]. Developing general-purpose debugging methods for machine learning that do not directly inspect raw data remains a challenge.

**Noisy model updates** In Section 5.1 above, we discussed the potential for an adversary to send malicious model updates to the server from some number of clients. Even if no adversary is present, the model updates sent to the server may become distorted due to network and architectural factors. This is especially likely in cross-client settings, where separate entities control the server, clients, and network. Similar distortions can occur due to the client data. Even if the data on a client is not intentionally malicious, it may have noisy features [350] (eg. in vision applications, a client may have a low-resolution camera whose output is scaled to a higher resolution) or noisy labels [356] (eg. if the user indicates that a recommendation by an app is not relevant accidentally). While clients in cross-silo federated learning systems (see Table 1) may perform data cleaning to remove such corruptions, such processing is unlikely to occur in cross-device settings due to data privacy restrictions. In the end, these aforementioned corruptions may harm the convergence of the federated learning process, whether they are due to network factors or noisy data.

Since these corruptions can be viewed as mild forms of model update and data poisoning attacks, one mitigation strategy would be to use defenses for adversarial model update and data poisoning attacks. Given the current lack of demonstrably robust training methods in the federated setting, this may not be a practical option. Moreover, even if such techniques existed, they may be too computation-intensive for many

设备更可能是具有有限计算能力的边缘设备。即使没有明确的失败，也可能有掉队的客户端，它们比同一轮中的其他节点花费更长的时间来报告它们的输出。如果掉队者花了足够长的时间来报告，为了效率起见，他们可能会从一轮通信中被忽略，从而有效地减少了参与客户端的数量。在“普通”联邦学习中，这不需要真实的算法更改，因为联邦平均可以应用于任何客户端报告的模型更新。

不幸的是，当使用安全聚合（SecAgg）[80，58]时，无响应的客户端变得更具挑战性，特别是如果客户端在SecAgg协议期间退出。虽然SecAgg的设计对大量的脱落具有鲁棒性[81]，但仍有失败的可能性。失败的可能性可以通过各种互补的方式来减少。一个简单的方法是在每轮中选择比所需更多的设备。这有助于确保落伍者和故障设备对整体收敛的影响最小[81]。然而，在不可靠的网络设置中，这可能不够。降低故障概率的一种更复杂的方法是提高SecAgg的效率。这减少了客户端退出对SecAgg产生不利影响的时间窗口。 另一种可能性是开发一个异步版本的SecAgg，它不需要客户端在固定的时间窗口内参与，可能是通过采用通用异步安全多方分布式计算协议的技术[430]。更推测地，可能执行在多个计算轮上聚合的SecAgg的版本。这将允许落后的节点被包含在随后的轮中，而不是完全退出当前轮。

数据管道故障虽然联邦学习中的数据管道只存在于每个客户端中，但管道仍可能面临许多潜在问题。特别是，任何联邦学习系统仍然必须定义如何访问原始用户数据并将其预处理为训练数据。此管道中的错误或意外操作可能会极大地改变联邦学习过程。虽然数据管道错误通常可以通过数据中心设置中的标准数据分析工具发现，但联邦学习中的数据限制使得检测更具挑战性。例如，特征级预处理问题（如反转像素、连接单词等）服务器无法直接检测到[31]。一种可能的解决方案是使用具有差分隐私的联邦方法来训练生成模型，然后使用这些方法来合成可用于调试底层数据管道的新数据样本[31]。 开发不直接检查原始数据的机器学习通用调试方法仍然是一个挑战。

在上面的5.1节中，我们讨论了攻击者从一些客户端向服务器发送恶意模型更新的可能性。即使不存在对手，发送到服务器的模型更新也可能由于网络和架构因素而失真。这在跨客户端设置中尤其可能，其中单独的实体控制服务器、客户端和网络。由于客户端数据，可能会发生类似的失真。即使客户端上的数据不是故意恶意的，它也可能具有噪声特征[350]（例如：在视觉应用中，客户端可能具有低分辨率相机，其输出被缩放到更高分辨率）或噪声标签[356]（例如，如果用户意外地指示应用程序的推荐不相关）。 虽然跨竖井联合学习系统（见表1）中的客户端可能会执行数据清理以删除此类损坏，但由于数据隐私限制，此类处理不太可能在跨设备设置中发生。最后，上述这些损坏可能会损害联邦学习过程的收敛，无论是由于网络因素还是噪声数据。

由于这些破坏可以被视为模型更新和数据中毒攻击的温和形式，因此一种缓解策略是使用对抗性模型更新和数据中毒攻击的防御。鉴于目前在联邦环境中缺乏明显的健壮训练方法，这可能不是一个实际的选择。此外，即使存在这样的技术，它们对许多人来说也可能过于计算密集

federated learning applications. Thus, open work here involves developing training methods that are robust to small to moderate levels of noise. Another possibility is that standard federated training methods (such as federated averaging [337]) are inherently robust to small amounts of noise. Investigating the robustness of various federated training methods to varying levels amount of noise would shed light on how to ensure robustness of federated learning systems to non-malicious failure modes.

## 5.3 Exploring the Tension between Privacy and Robustness

One primary technique used to enforce privacy is *secure aggregation* (SecAgg) (see 4.2.1). In short, SecAgg is a tool used to ensure that the server only sees an aggregate of the client updates, not any individual client updates. While useful for ensuring privacy, SecAgg generally makes defenses against adversarial attacks more difficult to implement, as the central server only sees the aggregate of the client updates. Therefore, it is of fundamental interest to investigate how to defend against adversarial attacks when secure aggregation is used. Existing approaches based on range proofs (e.g. Bulletproofs [92]) can guarantee that the DP-based clipping defense described above is compatible with SecAgg, but developing computation- and communication-efficient range proofs is still an active research direction.

SecAgg also introduces challenges for other defense methods. For example, many existing Byzantine-robust aggregation methods utilize non-linear operations on the server Xie et al. [486], and it is not yet known if these methods are efficiently compatible with secure aggregation which was originally designed for linear aggregation. Recent work has found ways to approximate the geometric median under SecAgg [379] by using a handful of SecAgg calls in a more general aggregation loop. However, it is not clear in general which aggregators can be computed under the use of SecAgg.

## 5.4 Executive Summary

- Third-party participants in the training process introduces new capabilities and attack vectors for adversaries, categorized in Table 11.
- Federated learning introduces a new kind of poisoning attacks, *model update poisoning* (Section 5.1.2), while also being susceptible to traditional *data poisoning* in (Section 5.1.3).
- Training participants can influence the optimization process possibly exacerbating inference-time (Section *evasion attacks*) 5.1.4, and communication and computation constraints may render previously proposed defenses impractical.
- Non-malicious failure modes (Section 5.2) are can be especially different to deal with, as access to raw data is not available in the federated setting, though through some lens they may be related to poisoning attacks.
- Tension may exist when trying to simultaneously improve robustness and privacy in machine learning (Section 5.3).

Areas identified for further exploration include:

- Quantify the relationship between data poisoning and model update poisoning attacks. Are there scenarios where they are not equivalent? [5.1.3]

联邦学习应用。因此，这里的开放式工作涉及开发对小到中等水平的噪音具有鲁棒性的训练方法。另一种可能性是标准的联邦训练方法（如联邦平均[337]）对少量噪声具有固有的鲁棒性。研究各种联邦训练方法对不同噪声水平的鲁棒性将有助于了解如何确保联邦学习系统对非恶意故障模式的鲁棒性。

## 5.3隐私与鲁棒性之间的张力探讨

一种用于实施隐私的主要技术是安全聚合（SecAgg）（参见4.2.1）。简而言之，SecAgg是一个工具，用于确保服务器只看到客户端更新的聚合，而不是任何单个客户端更新。虽然SecAgg对于确保隐私很有用，但它通常使对抗性攻击的防御更难以实现，因为中央服务器只能看到客户端更新的聚合。因此，研究如何在使用安全聚合时防御对抗性攻击具有根本意义。现有的基于范围证明的方法（例如Bulletproofs [92]）可以保证上述基于DP的裁剪防御与SecAgg兼容，但开发计算和通信高效的范围证明仍然是一个积极的研究方向。

SecAgg还为其他防御方法带来了挑战。例如，许多现有的Byzantinerobust聚合方法利用服务器上的非线性操作Xie等人[486]，并且尚不知道这些方法是否有效地与最初为线性聚合设计的安全聚合兼容。最近的工作已经找到了在SecAgg下近似几何中值的方法[379]，方法是在更一般的聚合循环中使用少数SecAgg调用。然而，一般来说，不清楚哪些聚合器可以在SecAgg的使用下计算。

5.4培训过程中的第三方参与者为对手介绍了新的功能和攻击向量，如表11所示。

联邦学习引入了一种新的中毒攻击，模型更新中毒（5.1.2节），同时也容易受到传统数据中毒（5.1.3节）。

·训练参与者可能会影响优化过程，可能会加剧推理时间（分段规避攻击）5.1.4，并且通信和计算约束可能会使先前提出的防御变得不切实际。

·非恶意故障模式（第5.2节）的处理可能特别不同，因为在联邦设置中无法访问原始数据，尽管通过某些透镜，它们可能与中毒攻击有关。

·当试图同时提高机器学习中的鲁棒性和隐私性时，可能会存在张力（第5.3节）。

已确定需要进一步探索的领域包括：

·量化数据中毒和模型更新中毒攻击之间的关系。是否存在它们不等同的情况？[5.1.3]

- Quantify how training time attacks impact inference-time vulnerabilities. Improving inference-time robustness guarantees requires going beyond detecting backdoor attacks. [5.1.4]
- Adversarial training has been used as a defense in the centralized setting, but can be impractical in the edge-compute limited cross-device federated setting. [5.1.5]
- Federated learning requires new methods and tools to support the developer, as access to raw data is restricted debugging ML pipelines is especially difficult. [5.2]
- Tensions exists between robustness and fairness, as machine learning models can tend to discard updates far from the median as detrimental. However the federated setting can give rise to a long tail of users that may be mistaken for noisy model updates [5.2].
- Cryptography-based aggregation methods and robustness techniques present integration challenges: protecting participant identity can be at odds with detecting adversarial participants. Proposed techniques remain beyond the scope of practicality, requiring the need of new communication and computation efficient algorithms. [5.3]

3]·量化训练时间攻击如何影响推理时间漏洞。提高推理时间鲁棒性保证需要超越检测后门攻击。[5.1.4]

对抗训练已被用作集中式设置中的防御，但在边缘计算有限的跨设备联合设置中可能不切实际。[5.1.5]

联邦学习需要新的方法和工具来支持开发人员，因为对原始数据的访问受到限制，调试ML管道特别困难。[5.2]

鲁棒性和公平性之间存在紧张关系，因为机器学习模型往往会丢弃远离中位数的更新，因为这是有害的。然而，联邦设置可能会给予一个长尾用户，这可能会被误认为是嘈杂的模型更新[5.2]。

·基于加密的聚合方法和鲁棒性技术带来了集成挑战：保护参与者身份可能与检测敌对参与者不一致。所提出的技术仍然超出了实用性的范围，需要新的通信和计算有效的算法。[5.3]

# 6 Ensuring Fairness and Addressing Sources of Bias

Machine learning models can often exhibit surprising and unintended behaviours. When such behaviours lead to patterns of *undesirable* effects on users, we might categorize the model as "unfair" according to some criteria. For example, if people with similar characteristics receive quite different outcomes, then this violates the criterion of *individual fairness* [169]. If certain sensitive groups (races, genders, etc.) receive different patterns of outcomes—such as different false negative rates—this can violate various criteria of *demographic fairness*, see for instance [51, 349] for surveys. The criterion of *counterfactual fairness* requires that a user receive the same treatment as they would have if they had been a member of a different group (race, gender, etc), after taking all causally relevant pathways into account [287].

Federated learning raises several opportunities for fairness research, some of which extend prior research directions in the non-federated setting, and others that are unique to federated learning. This section raises open problems in both categories.

## 6.1 Bias in Training Data

One driver of unfairness in machine-learned models is bias in the training data, including cognitive, sampling, reporting, and confirmation bias. One common antipattern is that minority or marginalized social groups are under-represented in the training data, and thus the learner weights these groups less during training [258], leading to inferior quality predictions for members of these groups (e.g. [93]).

Just as the data access processes used in federated learning may introduce dataset shift and non-independence (Section 3.1), there is also a risk of introducing biases. For example:

- If devices are selected for updates when plugged-in or fully charged, then model updates and evaluations computed at different times of day may be correlated with factors such as day-shift vs night-shift work schedules.
- If devices are selected for updates from among the pool of eligible devices at a given time, then devices that are connected at times when few other devices are connected (e.g. night-shift or unusual time zone) may be over-represented in the aggregated output.
- If selected devices are more likely to have their output kept when the output is computed faster, then: a) output from devices with faster processors may be over-represented, with these devices likely newer devices and thus correlated with socioeconomic status; and b) devices with less data may be over-represented, with these devices possibly representing users who use the product less frequently.
- If data nodes have different amounts of data, then federated learning may weigh higher the contributions of populations which are heavy users of the product or feature generating the data.
- If the update frequency depends on latency, then certain geographic regions and populations with slower devices or networks may be under-represented.
- If populations of *potential users* do not own devices for socio-economic reasons, they may be under-represented in the training dataset, and subsequently also under- (or un-)represented in model training and evaluation.
- Unweighted aggregation of the model loss across selected devices during federated training may disadvantage model performance on certain devices [302].

# 6.确保公平并解决偏见的根源

机器学习模型通常会表现出令人惊讶和意想不到的行为。当这些行为导致对用户产生不良影响的模式时，我们可能会根据某些标准将模型归类为“不公平”。例如，如果具有相似特征的人得到完全不同的结果，那么这违反了个人公平的标准[169]。如果某些敏感群体（种族、性别等）接受不同的结果模式-例如不同的假阴性率-这可能违反人口统计公平性的各种标准，参见例如[51，349]的调查。反事实公平的标准要求用户在考虑所有因果相关途径后，获得与他们是不同群体（种族，性别等）成员时相同的待遇。

联邦学习为公平性研究提供了几个机会，其中一些扩展了非联邦环境中的先前研究方向，另一些则是联邦学习所独有的。本节提出了这两个类别中的未决问题。

## 6.1训练数据中的偏差

机器学习模型中不公平的一个驱动因素是训练数据中的偏差，包括认知、采样、报告和确认偏差。一个常见的反模式是少数或边缘化的社会群体在训练数据中代表性不足，因此学习者在训练过程中对这些群体的权重较小[258]，导致这些群体成员的预测质量较差（例如[93]）。

正如联邦学习中使用的数据访问过程可能会引入数据集偏移和非独立性（第3.1节）一样，也存在引入偏差的风险。举例来说：

·如果选择设备在插入或充满电时进行更新，则在一天中的不同时间计算的模型更新和评估可能与诸如白班与夜班工作时间表的因素相关。

·如果在给定时间从合格设备池中选择设备进行更新，则在很少其他设备连接的时间（例如夜班或不寻常时区）连接的设备可能在聚合输出中过度表示。

·如果所选设备在输出计算速度更快时更有可能保留其输出，则：a）来自具有更快处理器的设备的输出可能被过度表示，这些设备可能是较新的设备，因此与社会经济地位相关;以及b）具有较少数据的设备可能被过度表示，这些设备可能代表不太频繁使用产品的用户。

·如果数据节点具有不同的数据量，那么联邦学习可能会对生成数据的产品或功能的重度用户群体的贡献给予更高的权重。

·如果更新频率取决于延迟，则某些地理区域和具有较慢设备或网络的人群可能代表性不足。

·如果潜在用户群体由于社会经济原因不拥有设备，则他们可能在训练数据集中代表性不足，并且随后在模型训练和评估中也代表性不足（或未代表）。

·在联合训练期间，跨选定设备的模型损失的未加权聚合可能会损害某些设备上的模型性能[302]。

It has been observed that biases in the data-generating process can also drive unfairness in the resulting models learned from this data (see e.g. [170, 394]). For example, suppose training data is based on user interactions with a product which has failed to incorporate inclusive design principles. Then, the user interactions with the product might not express user intents (cf. [403], for example) but rather might express coping strategies around uninclusive product designs (and hence might require a fundamental fix to the product interaction model). Learning from such interactions might then ignore or perpetuate poor experiences for some groups of product users in ways which can be difficult to detect while maintaining privacy in a federated setting. This risk is shared by all machine learning scenarios where training data is derived from user interaction, but is of particular note in the federated setting when data is collected from apps on individual devices.

Investigating the degree to which biases in the data-generated process can be identified or mitigated is a crucial problem for both federated learning research and ML research more broadly. Similarly, while limited prior research has demonstrated methods to identify and correct bias in already collected data in the federated setting (e.g. via adversarial methods in [255]), further research in this area is needed. Finally, methods for applying post-hoc fairness corrections to models learned from potentially biased training data are also a valuable direction for future work.

## 6.2 Fairness Without Access to Sensitive Attributes

Having explicit access to demographic information (race, gender, etc) is critical to many existing fairness criteria, including those discussed in Section 6.1. However, the contexts in which federated learning are often deployed also give rise to considerations of fairness when individual sensitive attributes are *not* available. For example, this can occur when developing personalized language models or developing fair medical image classifiers without knowing any additional demographic information about individuals. Even more fundamentally, the assumed one-to-one relationship between individuals and devices often breaks down, especially in non-Western contexts [403]. Both measuring and correcting unfairness in contexts where there is no data regarding sensitive group membership is a key area for federated learning researchers to address.

Limited existing research has examined fairness without access to sensitive attributes. For example, this has been addressed using distributionally-robust optimization (DRO) which optimizes for the worst-case outcome across all individuals during training [225], and via multicalibration, which calibrates for fairness across subsets of the training data [232]. Even these existing approaches have not been applied in the federated setting, raising opportunities for future empirical work. The challenge of how to make these approaches work for large-scale, high-dimensional data typical to federated settings is also an open problem, as DRO and multicalibration both pose challenges of scaling with large $n$ and $p$. Finally, the development of additional theoretical approaches to defining fairness without respect to "sensitive attributes" is a critical area for further research.

Other ways to approach this include reframing the existing notions of fairness, which are primarily concerned with equalizing the probability of an outcome (one of which is considered "positive" and another "negative" for the affected individual). Instead, fairness without access to sensitive attributes might be reframed as *equal access to effective models*. Under this interpretation of fairness, the goal is to maximize model utility across all individuals, regardless of their (unknown) demographic identities, and regardless of the "goodness" of an individual outcome. Again, this matches the contexts in which federated learning is most commonly used, such as language modeling or medical image classification, where there is no clear notion of an outcome which is "good" for a user, and instead the aim is simply to make correct predictions for users, regardless of the outcome.

据观察，数据生成过程中的偏差也会导致从这些数据中学习到的结果模型中的不公平性（参见[170，394]）。例如，假设训练数据是基于用户与未能包含包容性设计原则的产品的交互。然后，用户与产品的交互可能不会表达用户意图（参见[403]例如），而可能是表达围绕非包容性产品设计的应对策略（因此可能需要对产品交互模型进行根本修复）。从这种交互中学习可能会忽略或延续某些产品用户组的不良体验，这些用户在联合设置中维护隐私的同时很难检测到。所有机器学习场景都存在这种风险，其中训练数据来自用户交互，但当数据从单个设备上的应用程序收集时，在联合设置中特别值得注意。

调查数据生成过程中的偏差可以被识别或减轻的程度，对于联邦学习研究和更广泛的ML研究来说都是一个至关重要的问题。类似地，虽然有限的先前研究已经证明了在联邦环境中识别和纠正已经收集的数据中的偏差的方法（例如[255]中的对抗方法），但需要在这一领域进行进一步的研究。最后，将事后公平性校正应用于从潜在偏差训练数据中学习的模型的方法也是未来工作的一个有价值的方向。

## 6.2不涉及敏感属性的公平性

明确获得人口统计信息（种族、性别等）对许多现有的公平标准至关重要，包括第6.1节中讨论的标准。然而，联邦学习经常部署的环境也给予考虑的公平性时，个别敏感属性不可用。例如，这可能发生在开发个性化的语言模型或开发公平的医学图像分类器时，而不知道关于个体的任何附加人口统计信息。更根本的是，个人和设备之间的一对一关系往往会破裂，特别是在非西方环境中。在没有关于敏感组成员关系的数据的情况下，测量和纠正不公平是联邦学习研究人员要解决的一个关键领域。

有限的现有研究审查了公平性，没有敏感的属性。例如，这已经通过使用分布式鲁棒优化（DRO）来解决，该优化在训练期间优化所有个体的最坏情况结果[225]，并且通过多校准来校准训练数据子集的公平性[232]。即使是这些现有的方法也没有在联邦环境中应用，这为未来的实证工作提供了机会。如何使这些方法的工作大规模，高维数据典型的联邦设置的挑战也是一个悬而未决的问题，DRO和多校准都带来了挑战的缩放与大的n和p最后，额外的理论方法来定义公平性的发展，而不尊重“敏感属性”是一个关键领域，进一步研究。

其他方法包括重新定义现有的公平概念，这些概念主要涉及结果的概率相等（其中一个被认为是“积极的”，另一个被认为是“消极的”对受影响的个人）。相反，公平而没有敏感属性的访问可能会被重新定义为平等地获得有效的模型。在这种对公平的解释下，目标是在所有个体中最大化模型效用，而不管他们的（未知的）人口统计身份，也不管个体结果的“善”。同样，这与联邦学习最常用的上下文相匹配，例如语言建模或医学图像分类，其中没有明确的结果对用户来说是“好”的概念，相反，目标只是为用户做出正确的预测，而不管结果如何。

Existing federated learning research suggests possible ways to meet such an interpretation of fairness, e.g. via personalization [250, 472]. A similar conception of fairness, as "a more fair distribution of the model performance across devices", is employed in [302].

The application of attribute-independent methods explicitly to ensure equitable model performance is an open opportunity for future federated learning research, and is particularly important as federated learning reaches maturity and sees increasing deployment with real populations of users without knowledge of their sensitive identities.

## 6.3 Fairness, Privacy, and Robustness

Fairness and data privacy seem to be complementary ethical concepts: in many of the real-world contexts where privacy protection is desired, fairness is also desired. Often this is due to the sensitivity of the underlying data. Because federated learning is most likely to be deployed in contexts of sensitive data where both privacy and fairness are desirable, it is important that FL research examines how FL might be able to address existing concerns about fairness in machine learning, and whether FL raises new fairness-related issues.

In some ways, however, the ideal of fairness seems to be in tension with the notions of privacy for which FL seeks to provide guarantees: differentially-private learning typically seeks to obscure individually-identifying characteristics, while fairness often requires knowing individuals' membership in sensitive groups in order to measure or ensure fair predictions are being made. While the trade-off between differential privacy and fairness has been investigated in the non-federated setting [246, 145], there has been little work on how (or whether) FL may be able to uniquely address concerns about fairness.

Recent evidence suggesting that differentially-private learning can have disparate impact on sensitive subgroups [43, 145, 246, 283] provides further motivation to investigate whether FL may be able to address such concerns. A potential solution to relax the tension between privacy (which aims to protect the model from being too dependent on individuals) and fairness (which encourages the model to perform well on under-represented classes) may be the application of techniques such as personalization (discussed in Section 3.3) and "hybrid differential privacy," where some users donate data with lesser privacy guarantees [40].

Furthermore, current differentially-private optimization schemes are applied without respect to sensitive attributes – from this perspective, it might be expected that empirical studies have shown evidence that differentially-private optimization impacts minority subgroups the most [43]. Modifications to differentially-private optimization algorithms which explicitly seek to preserve performance on minority subgroups, e.g. by adapting the noise and clipping mechanisms to account for the representation of groups within the data, would also likely do a great deal to limit potential disparate impacts of differentially-private modeling on minority subgroups in federated models trained with differential privacy. However, implementing such adaptive differentially-private mechanisms in a way that provides some form of privacy guarantee presents both algorithmic and theoretical challenges which need to be addressed by future work.

Further research is also needed to determine the extent to which the issues above arise in the federated setting. Furthermore, as noted in Section 6.2, the challenge of evaluating the impact of differential privacy on model fairness becomes particularly difficult when sensitive attributes are not available, as it is unclear how to identify subgroups for which a model is behaving badly and to quantify the "price" of differential privacy – investigating and addressing these challenges is an open problem for future work.

More broadly, one could more generally examine the relation between privacy, fairness, and *robustness* (see Section 5). Many previous works on machine learning, including federated learning, typically focus on

现有的联邦学习研究提出了满足这种公平性解释的可能方法，例如通过个性化[250，472]。在[302]中采用了类似的公平性概念，即“模型性能在设备之间的更公平分布”。

明确地应用属性独立方法来确保公平的模型性能是未来联邦学习研究的一个开放机会，并且随着联邦学习达到成熟并看到越来越多的部署真实的用户群体而不知道他们的敏感身份，这一点尤为重要。

## 6.3公平性、隐私性和健壮性

公平和数据隐私似乎是互补的伦理概念：在许多需要隐私保护的现实世界中，公平也是需要的。这通常是由于基础数据的敏感性。由于联邦学习最有可能部署在需要隐私和公平性的敏感数据环境中，因此FL研究如何能够解决机器学习中现有的公平性问题，以及FL是否会引发新的公平性相关问题，这一点很重要。

然而，在某些方面，公平的理想似乎与FL寻求提供保证的隐私概念存在紧张关系：差异私人学习通常寻求掩盖个人识别特征，而公平通常需要知道个人在敏感群体中的成员资格，以衡量或确保公平的预测正在进行。虽然在非联邦环境中研究了差异隐私和公平性之间的权衡[246，145]，但关于FL如何（或是否）能够唯一地解决公平性问题的工作很少。

最近的证据表明，差异化的私人学习可以对敏感的亚组产生不同的影响[43，145，246，283]，这进一步推动了研究FL是否能够解决这些问题。缓解隐私（旨在保护模型不过于依赖个人）和公平性（鼓励模型在代表性不足的类别上表现良好）之间的紧张关系的一个潜在解决方案可能是应用个性化（在第3.3节中讨论）和“混合差分隐私”等技术，其中一些用户捐赠的数据具有较少的隐私保证[40]。

此外，当前的差分私有优化方案在不考虑敏感属性的情况下应用-从这个角度来看，可以预期实证研究已经显示出差分私有优化对少数子群体影响最大的证据[43]。对明确寻求保留少数子群性能的差分私有优化算法的修改，例如通过调整噪声和裁剪机制来考虑数据中的组的表示，也可能会在很大程度上限制差分私有建模对使用差分隐私训练的联邦模型中的少数子群的潜在不同影响。然而，实现这种自适应差分隐私机制的方式，提供某种形式的隐私保证提出了算法和理论的挑战，需要解决的未来的工作。

还需要进一步研究，以确定上述问题在联邦环境中出现的程度。此外，如第6.2节所述，当敏感属性不可用时，评估差异隐私对模型公平性的影响的挑战变得特别困难，因为不清楚如何识别模型表现不佳的子组以及量化差异隐私的“价格”-调查和解决这些挑战是未来工作的开放问题。

更广泛地说，人们可以更普遍地研究隐私、公平性和健壮性之间的关系（见第5节）。许多以前的机器学习工作，包括联邦学习，通常侧重于

isolated aspects of robustness (either against poisoning, or against evasion), privacy, or fairness. An important open challenge is to develop a joint understanding of federated learning systems that are robust, private, and fair. Such an integrated approach can provide opportunities to benefit from disparate but complementary mechanisms. Differential privacy mechanisms can be used to both mitigate data inference attacks, and provide a foundation for robustness against data poisoning. On the other hand, such an integrated approach also reveals new vulnerabilities. For example, recent work has revealed a trade-off between privacy and robustness against adversarial examples [429].

Finally, privacy and fairness naturally meet in the context of learning data representations that are independent of some sensitive attributes while preserving utility for a task of interest. Indeed, this objective can be motivated both in terms of privacy: to transform data so as to hide private attributes, and fairness: as a way to make models trained on such representations fair with respect to the attributes. In the centralized setting, one way to learn such representations is through adversarial training techniques, which have been applied to image and speech data [255, 186, 327, 65, 431]. In the federated learning scenario, clients could apply the transformation locally to their data in order to enforce or improve privacy and/or fairness guarantees for the FL process. However, learning this transformation in a federated fashion (potentially under privacy and/or fairness constraints) is itself an open question.

## 6.4 Leveraging Federation to Improve Model Diversity

Federated learning presents the opportunity to integrate, through distributed training, datasets which may have previously been impractical or even illegal to combine in a single location. For example, the Health Insurance Portability and Accountability Act (HIPAA) and the Family Educational Rights and Privacy Act (FERPA) constrain the sharing of medical patient data and student educational data, respectively, in the United States. To date, these restrictions have led to modeling occurring in institutional silos: for example, using electronic health records or clinical images from individual medical institutions instead of pooling data and models across institutions [91, 104]. In contexts where membership in institutional datasets is correlated with individuals' specific sensitive attributes, or their behavior and outcomes more broadly, this can lead to poor representation for users in groups underrepresented at those institutions. Importantly, this lack of representation and diversity in the training data has been shown to lead to poor performance, e.g. in genetic disease models [333] and image classification models [93].

Federated learning presents an opportunity to leverage uniquely diverse datasets by providing efficient decentralized training protocols along with privacy and non-identifiability guarantees for the resulting models. This means that federated learning enables training on multi-instutitional datasets in many domains where this was previously not possible. This provides a practical opportunity to leverage larger, more diverse datasets and explore the generalizability of models which were previously limited to small populations. More importantly, it provides an opportunity to improve the *fairness* of these models by combining data across boundaries which are likely to have been correlated with sensitive attributes. For instance, attendance at specific health or educational institutions may be correlated with individuals' ethnicity or socioeconomic status. As noted in Section 6.1 above, underrepresentation in training data is a proven driver of model unfairness.

Future federated learning research should investigate the degree to which improving diversity in a federated training setting also improves the fairness of the resulting model, and the degree to which the differential privacy mechanisms required in such settings may limit fairness and performance gains from increased diversity. This includes a need for both empirical research which applies federated learning and quantifies the interplay between diversity, fairness, privacy, and performance; along with theoretical research which provides a foundation for concepts such as diversity in the context of machine learning fairness.

通常专注于鲁棒性（针对中毒或规避）、隐私或公平性的孤立方面。一个重要的开放性挑战是对健壮、私有和公平的联邦学习系统形成共同的理解。这种综合办法可以提供机会，使人们受益于不同但互补的机制。差分隐私机制可以用来减轻数据推断攻击，并提供对数据中毒的鲁棒性的基础。另一方面，这种综合办法也暴露出新的脆弱性。例如，最近的工作揭示了对抗性示例的隐私和鲁棒性之间的权衡[429]。

最后，隐私和公平性在学习数据表示的上下文中自然相遇，这些数据表示独立于某些敏感属性，同时保留感兴趣任务的效用。事实上，这一目标可以从隐私和公平两个方面来实现：隐私是指转换数据以隐藏私有属性，公平是指使在这种表示上训练的模型在属性方面是公平的。在集中式设置中，学习这种表示的一种方法是通过对抗训练技术，该技术已应用于图像和语音数据[255，186，327，65，431]。在联邦学习场景中，客户端可以将转换本地应用于其数据，以加强或改善FL过程的隐私和/或公平性保证。然而，以联邦方式学习这种转换（可能在隐私和/或公平性约束下）本身就是一个开放的问题。

## 6.4利用联盟提高模型多样性

联合学习提供了通过分布式训练集成数据集的机会，这些数据集以前可能不切实际，甚至不合法，无法在单个位置进行联合收割机组合。例如，在美国，《健康保险可携带性和责任法案》（HIPAA）和《家庭教育权利和隐私法案》（FERPA）分别限制了医疗患者数据和学生教育数据的共享。到目前为止，这些限制导致建模发生在机构孤岛中：例如，使用来自各个医疗机构的电子健康记录或临床图像，而不是跨机构汇集数据和模型[91，104]。在机构数据集中的成员资格与个人的特定敏感属性或更广泛的行为和结果相关的情况下，这可能导致用户在这些机构中代表性不足的群体中的代表性差。 重要的是，训练数据中缺乏代表性和多样性已被证明会导致性能低下，例如在遗传疾病模型[333]和图像分类模型[93]中。

联邦学习提供了一个利用独特多样的数据集的机会，通过为最终模型提供有效的分散训练协议沿着隐私和不可识别性保证。这意味着联邦学习能够在许多领域的多指令数据集上进行训练，而这在以前是不可能的。这提供了一个实际的机会，利用更大，更多样化的数据集，并探索以前仅限于小群体的模型的普遍性。更重要的是，它提供了一个机会，通过合并可能与敏感属性相关的跨边界数据来提高这些模型的公平性。例如，在特定保健或教育机构的出勤率可能与个人的种族或社会经济地位有关。如上文第6.1节所述，训练数据中的代表性不足是模型不公平性的一个已被证明的驱动因素。

未来的联邦学习研究应该调查在联邦训练环境中提高多样性的程度也提高了最终模型的公平性，以及在这种环境中所需的差异隐私机制可能会限制公平性和增加多样性带来的性能收益的程度。这包括需要进行实证研究，应用联邦学习并量化多样性，公平性，隐私和性能之间的相互作用;沿着理论研究，为机器学习公平性背景下的多样性等概念提供基础。

## 6.5 Federated Fairness: New Opportunities and Challenges

It is important to note that federated learning provides unique opportunities and challenges for fairness researchers. For example, by allowing for datasets which are distributed both by observation, but even by features, federated learning can enable modeling and research using partitioned data which may be too sensitive to share directly [215, 224]. Increased availability of datasets which can be used in a federated manner can help to improve the diversity of training data available for machine learning models, which can advance fair modeling theory and practice.

Researchers and practitioners also need to address the unique fairness-related challenges created by federated learning. For example, federated learning can introduce new sources of bias through the decision of which clients to sample based on considerations such as connection type/quality, device type, location, activity patterns, and local dataset size [81]. Future work could investigate the degree to which these various sampling constraints affect the fairness of the resulting model, and how such impacts can be mitigated within the federated framework, e.g. [302, 289, 158]. Frameworks such as *agnostic federated learning* [352] provide approaches to control for bias in the training objective. Work to improve the fairness of existing federated training algorithms will be particularly important as advances begin to approach the technical limits of other components of FL systems, such as model compression, which initially helped to broaden the diversity of candidate clients during federated training processes. There is no unique fairness criterion generally adopted in the study of fairness, and multiple criteria have been proven to be mutually incompatible. One way to deal with this question is the *online fairness* framework and algorithms of Awasthi et al. [41]. Adapting such solutions to the federated learning setting and further improving upon them will be challenging research questions in ML fairness theory and algorithms.

In the classical centralized machine learning setting, a substantial amount of advancement has been made in the past decade to train fair classifiers, such as constrained optimization, post-shifting approaches, and distributionally-robust optimization [223, 503, 225]. It is an open question whether such approaches, which have demonstrated utility for improving fairness in centralized training, could be used under the setting of federated learning (and if so, under what additional assumptions) in which data are located in a decentralized fashion and practitioners may not obtain an unbiased sample of the data that match the distribution of the population.

## 6.6 Executive Summary

In addition to inheriting the already significant challenges related to bias, fairness, and privacy in centralized machine learning, federated learning also brings a new set of distinct challenges and opportunities in these areas. The importance of these considerations will likely continue to grow as the real-world deployment of FL expands to more users, domains, and applications.

- Bias in training data (Section 6.1) is a key consideration related to bias and fairness in FL models, particularly due to the additional sampling steps germane to federation (e.g., client sampling) and the transfer of some model computation to client devices.
- The lack of data regarding sensitive attributes in many FL deployments can pose challenges for measuring and ensuring fairness, and also suggests potential reframing of fairness problems in ways that do not require such data (Section 6.2).
- Since FL is often deployed in contexts which are both privacy- and fairness-sensitive, this can magnify tensions between privacy and fairness objectives in practice. Further work is needed to address the

## 6.5联邦公平：新的机遇与挑战

值得注意的是，联邦学习为公平研究人员提供了独特的机会和挑战。例如，通过允许通过观察分布的数据集，甚至通过特征，联邦学习可以使用分区数据进行建模和研究，这些数据可能过于敏感而无法直接共享[215，224]。可以以联邦方式使用的数据集的可用性增加可以帮助提高机器学习模型可用的训练数据的多样性，这可以推进公平建模理论和实践。

研究人员和实践者还需要解决联邦学习所带来的独特的公平相关挑战。例如，联邦学习可以通过基于连接类型/质量、设备类型、位置、活动模式和本地数据集大小等考虑因素决定对哪些客户端进行采样来引入新的偏差来源[81]。未来的工作可能会调查这些不同的采样约束影响结果模型的公平性的程度，以及如何在联邦框架内减轻这些影响，例如[302，289，158]。诸如不可知联邦学习[352]等框架提供了控制训练目标偏差的方法。 提高现有联邦训练算法的公平性的工作将特别重要，因为进步开始接近FL系统的其他组件的技术限制，例如模型压缩，这最初有助于在联邦训练过程中扩大候选客户端的多样性。在公平问题的研究中，没有一个统一的公平标准被普遍采用，多个公平标准被证明是互不相容的。处理这个问题的一种方法是Awasthi等人的在线公平框架和算法。使这些解决方案适应联邦学习环境并进一步改进它们将是ML公平理论和算法中具有挑战性的研究问题。

在经典的集中式机器学习环境中，在过去的十年中已经取得了大量的进步来训练公平分类器，例如约束优化，后移位方法和分布式鲁棒优化[223，503，225]。这是一个悬而未决的问题，这些方法已经证明可以提高集中式培训的公平性，是否可以在联邦学习的设置下使用（如果是这样，在什么额外的假设下），其中数据以分散的方式定位，从业者可能无法获得与人口分布相匹配的无偏数据样本。

## 6.6执行摘要

除了继承集中式机器学习中与偏见、公平和隐私相关的重大挑战外，联邦学习还在这些领域带来了一系列新的独特挑战和机遇。随着FL的实际部署扩展到更多的用户、域和应用程序，这些考虑因素的重要性可能会继续增长。

·训练数据中的偏差（第6.1节）是与FL模型中的偏差和公平性相关的关键考虑因素，特别是由于与联邦密切相关的额外采样步骤（例如，客户端采样）以及将某些模型计算转移到客户端设备。

·在许多FL部署中缺乏有关敏感属性的数据可能对测量和确保公平性构成挑战，并且还建议以不需要此类数据的方式重新定义公平性问题（第6.2节）。

由于FL通常部署在隐私和公平敏感的环境中，这可能会在实践中放大隐私和公平目标之间的紧张关系。需要进一步开展工作，

potential tension between methods which achieve privacy, fairness, and robustness in both federated and centralized learning (Section 6.3).

- Federated learning presents unique opportunities to improve the diversity of stakeholders and data incorporated into learning, which could improve both the overall quality of downstream models, as well as their fairness due to more representative datasets (Section 6.4).
- Federated learning presents fairness-related challenges not present in the centralized training regime, but also affords new solutions (Section 6.5).

在联邦和集中式学习中实现隐私，公平和鲁棒性的方法之间的潜在紧张关系（第6.3节）。

联邦学习提供了独特的机会，可以提高利益相关者和纳入学习的数据的多样性，这可以提高下游模型的整体质量，以及由于更具代表性的数据集而提高其公平性（第6.4节）。

联邦学习提出了集中式培训制度中不存在的公平相关挑战，但也提供了新的解决方案（第6.5节）。

# 7 Addressing System Challenges

As we will see in this section, the challenges in building systems for federated learning can be split fairly cleanly into the two separate settings of cross-device and cross-silo federated learning (see Sections 1.1 and 2.2). We start with a brief discussion of the difficulties inherent to any large scale deployment of software on end-user devices (although exacerbated by the complexity of a federated learning stack); we then focus on key challenges specific to the cross-device learning—bias, tuning, and efficient device-side execution of ML workflows—before concluding with a brief treatment of the cross-silo setting.

## 7.1 Platform Development and Deployment Challenges

Running computations on end-user devices is considerably different from the data center setting:

- Due to the heterogeneity of the fleet (devices may differ in hardware, software, connectivity, performance and persisted state) the space of potential problems and edge cases is vast and cannot typically be covered in sufficient detail with automated testing.
- Monitoring and debugging are harder because telemetry is limited, delayed, and there is no physical access to devices for interactive troubleshooting.
- Running computations should not affect device performance or stability, i.e. should be invisible to users.

**Code Deployment** Installing, updating and running software on end user devices may involve not only extensive manual and automated testing, but a gradual and reversible rollout (for example, through guarding new functionality with server-controlled feature flags) while monitoring key performance metrics in a/b experiments such as crash rates, memory use, and application-dependent indicators such as latencies and engagement metrics. Such rollouts can take weeks or months depending on the percolation rate of updates (particularly challenging for devices with spotty connectivity) and the complexity of the upgrade (e.g. protocol changes). Hence, the install base at any given time will involve various releases. While this problem is not specific to federated learning, it has greater impact here due to the inherent collaborative nature of federated computations: devices constantly communicate with servers and indirectly with other devices to exchange models and parameter updates. Thus, compatibility concerns abound and must be addressed through stable exchange formats or, where not possible, detected upfront with extensive testing infrastructure. We will revisit this problem in Section 7.4.

**Monitoring and Debugging** Another significant complication is the limited ability to monitor devices and interactively debug problems. While telemetry from end user devices is necessary to detect problems, privacy concerns severely restrict what can be logged, who can access such logs, and how long they are retained. Once a regression is detected, drilling down into the root cause can be very cumbersome due to the lack of detailed context, the vast problem space (a cross product of software versions, hardware, models, and device state), and very limited ability for interactive debugging short of successfully reproducing the problem in a controlled environment.

These challenges are exacerbated in the federated learning setting where a) raw input data on devices cannot be accessed, and b) contributions from individual devices are by design anonymous, ephemeral, and exposed only in aggregate. These properties preserve privacy, but also may make it hard or impossible to

# 7应对系统挑战

正如我们将在本节中看到的，构建联邦学习系统的挑战可以相当清晰地分为两个独立的设置：跨设备和跨竖井联邦学习（见1.1和2.2节）。我们首先简要讨论了在最终用户设备上大规模部署软件所固有的困难（尽管联邦学习堆栈的复杂性加剧了这些困难）;然后我们重点讨论了跨设备学习的关键挑战-偏见，调优和ML工作流的有效设备端执行-最后简要介绍了跨筒仓设置。

## 7.1平台开发和部署挑战

在最终用户设备上运行计算与数据中心设置有很大不同：

·由于机群的异构性（设备在硬件、软件、连接性、性能和持久状态方面可能不同），潜在问题和边缘情况的空间很大，通常无法通过自动化测试进行足够详细的覆盖。

·监测和调试更困难，因为遥测有限，延迟，并且没有物理访问设备进行交互式故障排除。

·运行计算不应影响设备性能或稳定性，即对用户不可见。

在最终用户设备上安装、更新和运行软件可能不仅涉及大量的手动和自动化测试，还涉及逐步和可逆的推出（例如，通过使用服务器控制的功能标志来保护新功能），同时监控a/b实验中的关键性能指标，如崩溃率、内存使用以及依赖于应用程序的指标，如延迟和参与指标。这种部署可能需要数周或数月的时间，具体取决于更新的渗透率（对于连接不稳定的设备来说尤其具有挑战性）和升级的复杂性（例如协议更改）。因此，在任何给定时间的安装基础将涉及各种版本。虽然这个问题并不是联邦学习所特有的，但由于联邦计算固有的协作性质，它在这里有更大的影响：设备不断与服务器通信，并间接与其他设备交换模型和参数更新。 因此，兼容性问题比比皆是，必须通过稳定的交换格式来解决，或者在不可能的情况下，通过广泛的测试基础设施预先检测。我们将在第7.4节重新讨论这个问题。

监控和调试另一个重要的复杂性是监控设备和交互式调试问题的能力有限。虽然来自最终用户设备的遥测是检测问题所必需的，但隐私问题严重限制了可以记录的内容，谁可以访问这些日志以及它们保留的时间。一旦检测到回归，由于缺乏详细的上下文、巨大的问题空间（软件版本、硬件、型号和设备状态的交叉产品）以及交互式调试的能力非常有限，无法在受控环境中成功再现问题，因此深入研究根本原因可能非常麻烦。

这些挑战在联合学习环境中加剧，其中a）无法访问设备上的原始输入数据，b）来自单个设备的贡献通过设计是匿名的，短暂的，并且仅在聚合中公开。这些属性保护隐私，但也可能使其难以或不可能

investigate problems with traditional approaches —by looking for correlations with hardware or software version, or testing hypotheses that require access to raw data. Reproducing a problem in a controlled setting is often difficult due to the gap between such an environment and reality: hundreds of heterogeneous embedded stateful devices with non-iid data.

Interestingly, federated technologies themselves can help to mitigate this problem—for instance, the use of federated analytics [382] to collect logs in a privacy preserving manner, or training generative models of the system behavior or raw data for sampling during debugging (see sections 3.4.3, 5.2, and [31]). Keeping a federated learning system up and running thus requires investing into upfront detection of problems through a) extensive automated, continuous test coverage of all software layers through both unit and integration tests; b) feature flags and a/b rollouts; and c) continuous monitoring of performance indicators for regressions. That poses a significant investment that may come at too high a cost for smaller entities who would benefit greatly from shared and tested infrastructure for federated learning.

## 7.2 System Induced Bias

Deployment, monitoring and debugging may not concern users of a federated learning platform, e.g. model authors or data analysts. For them, the key differences between data center and cross-device settings fall largely into the following two categories:

1. **Availability of devices** for computations is not a given, but varies over time and across devices. Connections are initiated by devices and subject to interruptions due to changes in device state, operating system quotas, and network connectivity. Hence, in iterative processes like federated learning, the loop body is run on a small subset of all devices only, and the system must tolerate a certain failure rate among those devices.

2. **Capabilities of devices** (network bandwidth and latency, compute performance, memory) vary, and are typically much lower than those of compute nodes in the data center, though the number of nodes is typically higher. The amount and type of data across devices may lead to variations in execution profile, e.g. more and larger examples lead to increased resource use and processing time.

In the following sections we discuss how these variations might introduce bias, referring to it as system induced bias to differentiate it from platform-independent bias in the raw data (such as ownership or usage patterns differing across demographics)—for the latter, see Section 6.1.

### 7.2.1 Device Availability Profiles

At the core of cross-device federated learning is the principle that devices only connect to the server and run computations when various constraints are met:

- **Hard constraints**, which might include requiring that the device is turned on, has network connectivity to the server, and is allowed to run a computation by the operating system.

- **Soft constraints**, which might include the conditions on device state chosen to ensure that federated learning does not incur charges or affect usability. For the common case of mobile phones [81, 26], requirements may include idleness, charging and/or above a certain battery level, being connected to an unmetered network, and that no other federated learning tasks are running at the same time.

而且也可能使得难以或不可能用传统方法来调查问题-通过寻找与硬件或软件版本的相关性，或测试需要访问原始数据的假设。在受控环境中重现问题通常很困难，因为这样的环境与现实之间存在差距：数百个异构的嵌入式有状态设备具有非iid数据。

有趣的是，联邦技术本身可以帮助缓解这个问题-例如，使用联邦分析[382]以隐私保护的方式收集日志，或者训练系统行为的生成模型或在调试期间采样的原始数据（参见第3.4.3，5.2和[31]节）。因此，保持联邦学习系统的正常运行需要通过以下方式对问题进行前期检测：a）通过单元和集成测试对所有软件层进行广泛的自动化、连续的测试覆盖; b）功能标志和a/b推出; c）对回归的性能指标进行持续监控。这是一项重大投资，对于小型实体来说，成本可能太高，这些实体将从共享和测试的联邦学习基础设施中受益匪浅。

## 7.2系统诱导偏倚

部署、监控和调试可能不涉及联合学习平台的用户，例如模型作者或数据分析师。对于他们来说，数据中心和跨设备设置之间的主要区别主要分为以下两类：

1.用于计算的设备的可用性不是给定的，而是随时间和跨设备而变化。连接由设备发起，并且会由于设备状态、操作系统配额和网络连接的更改而中断。因此，在像联邦学习这样的迭代过程中，循环体只在所有设备的一小部分上运行，系统必须容忍这些设备之间的一定故障率。

2. 设备的能力（网络带宽和延迟、计算性能、内存）各不相同，通常比数据中心的计算节点低得多，但节点数量通常更高。跨设备的数据量和类型可能导致执行配置文件的变化，例如，更多和更大的示例导致增加的资源使用和处理时间。

在以下章节中，我们讨论了这些变化如何可能引入偏倚，将其称为系统诱导偏倚，以将其与原始数据中的平台无关偏倚（例如人口统计学中不同的所有权或使用模式）区分开来-对于后者，请参见第6.1节。

### 7.2.1设备可用性配置文件

跨设备联合学习的核心是设备仅在满足各种约束条件时连接到服务器并运行计算的原则：

- 硬约束，可能包括要求设备打开，具有到服务器的网络连接，并且允许操作系统运行计算。
- 软约束，可能包括设备状态的条件，以确保联邦学习不会产生费用或影响可用性。对于移动的电话的常见情况[81，26]，要求可以包括空闲、充电和/或高于特定电池水平、连接到未计量的网络以及没有其他联合学习任务同时运行。

Taken together, these constraints induce an unknown, time-varying and device-specific function $A_i(t)$ for a device $i$, and a fleet-wide *availability profile* $A(t) = \sum_i A_i(t)$. Round completion rates and server traffic patterns [81, 491] suggest that availability profiles for mobile phones are clustered into periodic functions with a period of 1 day, varying across devices in phase, shape and amplitude through factors such as demographics, geography etc. Availability for other end user devices such as laptops, tablets, or stationary devices such as smart speakers, displays and cameras, will differ, but the challenges discussed in the following sections apply there as well, albeit to a possibly lesser extent.

### 7.2.2 Examples of System Induced Bias

Sources of bias will depend on the specific way in which devices are selected to participate in training, and how the system influences which devices end up contributing to the final aggregated model update. Thus, it is useful to discuss these issues in light of a simplified but representative system design. In an iterative federated learning algorithm, such as Federated Averaging (Section 1.1.2, [337]), rounds are run consecutively on sets of at least $M$ devices. To accommodate a fraction $d$ of devices not contributing due to changes in device conditions, time-outs, or slowness (server-side aborts to avoid slow-downs by stragglers), an over-allocation scheme is used where

1. Rounds are started when at least $M' = \frac{M}{1-d}$ devices are available.
2. Rounds are closed as
   (a) *Aborted* when more than $M' - M$ devices have disconnected, or
   (b) *Successful* when at least $M$ devices have reported. One possible design choice is to stop after exactly $M$ devices; another possibility would be to keep waiting for stragglers (possibly up to some maximum time).

This sequence, when combined with variable availability profiles, may introduce various forms of bias:

1. Selection Bias - whether a device is included in a round at time t depends on both
   (a) Its availability profile $A_i(t)$
   (b) The number of simultaneously connected devices: $< M'$ and a round cannot be started; $\gg M'$ and the probability of a single device being included becomes very small. In effect, devices active only at either fleet-wide availability peaks or troughs may be under-represented.
2. Survival Bias
   (a) Since a server might choose to close a round at any point after the first $M$ devices have reported, contributions are biased towards devices with better network connections, faster processors, lower CPU load, and less data to process.
   (b) Devices drop out of rounds when they are interrupted by the operating system, which may happen due to changes in device conditions as described by $A_i(t)$, or due to e.g. excessive memory use.

As can be seen, the probability of a device contributing to a round of federated learning is a complex function of both internal (e.g. device specific) and external (fleet dynamic) factors. When this probability is correlated with statistics of the data distribution, aggregate results may be biased. For instance, language

总之，这些约束引起设备i的未知的、时变的和特定于设备的函数A（t），以及全车队可用性简档A（t）ÞA（t）.回合完成率和服务器流量模式[81，491]表明，移动的电话的可用性配置文件被聚类为周期为1天的周期性函数，通过人口统计学，地理等因素在相位，形状和幅度上随设备而变化。其他终端用户设备（如笔记本电脑，平板电脑或固定设备，如智能扬声器，显示器和相机）的可用性将有所不同，但以下各节讨论的挑战也适用于这些国家，尽管程度可能较低。

7.2.2系统引起的偏差示例

偏差的来源将取决于选择设备参与训练的具体方式，以及系统如何影响哪些设备最终对最终聚合模型更新做出贡献。因此，根据简化但具有代表性的系统设计来讨论这些问题是有用的。在迭代联合学习算法中，例如联合平均（第1.1.2节，[337]），轮次在至少M个设备的集合上连续运行。为了容纳由于设备条件的变化、超时或缓慢（服务器端中止以避免掉队者的减慢）而没有贡献的设备的一小部分d，使用过度分配方案，其中

1.当至少M = 1−d时开始舍入 - -设备可用。

2.回合关闭为（a）当超过M-M个设备断开连接时中止，或（b）当至少M个设备报告时成功。一种可能的设计选择是在

另一种可能性是继续等待掉队者（可能达到某个最大时间）。

当与可变可用性简档组合时，该序列可能引入各种形式的偏差：

1.选择偏差-设备在时间t是否被包括在轮中取决于（a）其可用性简档A（t）（b）同时连接的设备的数量： □ M

并且包括单个设备的概率变得非常小。实际上，仅在车队范围的可用性高峰或低谷时活动的设备可能代表不足。

2.生存偏差（a）由于服务器可能会选择在前M个设备报告后的任何时间点关闭一轮，

贡献偏向于具有更好的网络连接、更快的处理器、更低的CPU负载和更少的数据处理的设备。

(b)当设备被操作系统中断时，它们退出轮次，这可能是由于如A（t）所描述的设备条件的变化，或者由于例如过度的存储器使用而发生的。

可以看出，设备对一轮联合学习做出贡献的概率是内部（例如，设备特定）和外部（车队动态）因素的复杂函数。当此概率与数据分布的统计相关时，聚合结果可能会有偏差。例如，语言

models may over-represent demographics that have high quality internet connections or high end devices; and ranking models may not incorporate enough contributions from high engagement users who produce a lot of training data and hence longer training times.

Thus, designing systems that explicitly take such factors into account and integrate algorithms designed to both quantify and mitigate these effects are a fundamentally important research direction.

### 7.2.3 Open Challenges in Quantifying and Mitigating System Induced Bias

While the potential for bias in federated learning has been addressed in the literature (Section 6, [81, 302, 171]), a systematic study that qualifies and quantifies bias in realistic settings and its sources is a direction for future research. Conducting the necessary work may be hampered by both access to the necessary resources, and the difficulty in quantifying bias in a final statistical estimate due to the inherent lack of ground truth value.

We want to encourage further research to study how bias can be quantified and subsequently mitigated. A useful proxy metric for bias is to study the expected rate of contribution of a device to federated learning. In an unbiased system, this rate would be identical for every device; if it is not, the non-uniformity may provide a measure of bias. Studying the root causes for this non-uniformity may then provide important hints for how to mitigate bias, for example:

- When there is a strong correlation between devices finishing a round, and the number of examples they process or model size, possible fixes may include early stopping, or decreasing the model size.
- If the expected rate of contribution depends on factors outside our control, such as device model, network connectivity, location etc., one can view these factors as defining strata and applying *post-stratification* [312], that is, correcting for bias by scaling up or down contributions from devices depending on their stratum. It may also be possible to apply *stratified sampling* - e.g. change scheduling, or server selection policies, to affect the probability of including devices in a round as a function of their stratum.
- A very general, root-cause-agnostic mitigation could base the weight of a contribution solely on a device's past contribution profile (e.g. the number of rounds started or completed thus far). As a special case, consider *sampling without replacement* which could be implemented at the system level (stop connecting after one successful contribution) or at the model level (weight all but the first contribution with 0). This approach might not be sufficient when a population is large enough for most devices to contribute only infrequently (mostly one or zero times); in such cases, clustering devices based on some similarity metric and using cluster membership as stratum could help.
- Alternatives to the synchronous, round based execution described in the previous section may also help to mitigate bias. In particular, certain types of analytics may benefit from softening or eliminating the competition between devices for inclusion, by running rounds for long times with very large numbers of participants and without applying time-outs to stragglers. Such a method may not be applicable to algorithms where the iterative aspect (running many individual, chained rounds) is important.

The biggest obstacle to enabling such research is access to a representative fleet of end user devices, or a detailed description (e.g. in the form of a statistical model of a realistic distribution over $A_i(t)$ functions) of a fleet that can be used in simulations. Here, maintainers of FL production stacks are uniquely positioned to

模型可能过度表示具有高质量互联网连接或高端设备的人口统计;并且排名模型可能没有包含来自产生大量训练数据并因此产生较长训练时间的高参与用户的足够贡献。

因此，设计明确考虑这些因素的系统并集成旨在量化和减轻这些影响的算法是一个非常重要的研究方向。

### 7.2.3量化和减轻系统诱导偏差的公开挑战

虽然文献中已经讨论了联邦学习中偏见的可能性（第6节，[81，302，171]），但在现实环境中定性和量化偏见及其来源的系统研究是未来研究的方向。开展必要的工作可能会受到以下两个因素的阻碍：获得必要资源的机会，以及由于内在缺乏地面实况值而难以量化最终统计估计中的偏差。

我们希望鼓励进一步的研究，以研究如何量化偏见，并随后减轻。
偏差的一个有用的代理度量是研究设备对联邦学习的预期贡献率。在无偏系统中，该速率对于每个设备都是相同的;如果不是，则非均匀性可以提供偏差的度量。研究这种不均匀性的根本原因可以为如何减轻偏差提供重要提示，例如：

·当完成一轮的设备与它们处理的示例数量或模型大小之间存在很强的相关性时，可能的修复方法可能包括提前停止或减小模型大小。

·如果预期贡献率取决于我们无法控制的因素，例如设备型号、网络连接、位置等，人们可以将这些因素视为定义分层和应用后分层[312]，即通过根据其分层按比例增加或减少设备的贡献来校正偏差。还可以应用分层采样（例如，改变调度或服务器选择策略）来影响根据设备的层将设备包括在轮中的概率。

·非常一般的、根本原因不可知的缓解可以使贡献的权重仅基于设备的过去贡献简档（例如，到目前为止开始或完成的回合数）。作为一种特殊情况，考虑不替换的采样，这可以在系统级（在一个成功的贡献后停止连接）或模型级（除了第一个贡献外，所有贡献的权重为0）实现。当人口足够大，大多数设备只偶尔贡献（主要是一次或零次）时，这种方法可能不够;在这种情况下，基于某种相似性度量对设备进行聚类，并使用群集成员身份作为分层可能会有所帮助。

·前一节中描述的同步、基于轮的执行的替代方案也可以帮助减轻偏差。特别是，某些类型的分析可能会受益于软化或消除设备之间的竞争，通过与大量参与者长时间运行回合，而不对落伍者应用超时。这种方法可能不适用于迭代方面（运行许多单独的链式循环）很重要的算法。

实现这种研究的最大障碍是访问终端用户设备的代表性队列，或者可以在模拟中使用的队列的详细描述（例如，以A（t）函数上的现实分布的统计模型的形式）。在这里，FL生产堆栈的维护人员处于独特的位置，

provide such statistics or models to academic partners in a privacy preserving fashion; a further promising direction is the recent introduction of the Flower framework [66] for federated learning research.

## 7.3 System Parameter Tuning

Practical federated learning is a form of multi-objective optimization: while the first order goal is maximizing model quality metrics such as loss or accuracy, other important considerations are

- Convergence speed
- Throughput (e.g. number of rounds, amount of data, or number of devices)
- Model fairness, privacy and robustness (see section 6.3)
- Resource use on server and clients

These goals may be in tension. For instance, maximizing round throughput may introduce bias or hurt accuracy by preferring performant devices with little or no data. Maximizing for low training loss by increasing model complexity will put devices with less memory, many or large examples, or slow CPUs at a disadvantage. Bias or fairness induced in such a way during training may be hard to detect in the evaluation phase since it typically uses the same platform and hence is subject to similar biases.

Various controls affect the above listed indicators. Some are familiar from the datacenter setting, in particular model specific settings and learning algorithm hyperparameters. Others are specific to federated learning:

- **Clients per round**: The minimum number of devices required to complete a round, $M$, and the number of devices required to start a round, $M'$.
- **Server-side scheduling**: In all but the simplest cases, a federated learning system will operate on more than one model at a time: to support multiple tenants; to train models on the same data for different use cases; to support experimentation and architecture or hyper-parameter grid search; and to run training and evaluation workloads concurrently. The server needs to decide which task to serve to incoming devices, an instance of a scheduling problem: assigning work (training or evaluation tasks) to resources (devices). Accordingly, the usual challenges arise: ideal resource assignment should be fair, avoid starvation, minimize wait times, and support relative priorities all at once.
- **Device-side scheduling**: As described in Section 7.2, various constraints govern when a device can connect to the server and execute work. Within these constraints, various scheduling choices can be made. One extreme is to connect to the server and run computations as often as possible, leading to high load and resource use on both server and devices. Another choice are fixed intervals, but they need to be adjusted to reflect external factors such as number of devices overall and per round. The federated learning system developed at Google aims to strike a balance with a flow control mechanism called *pace steering* [81] whereby the server instructs devices when to return. Such a dynamic system enables temporal load balancing for large populations as well as "focusing" connection attempts to specific points in time to reach the threshold $M'$. Developing such a mechanism is difficult due to stochastic and dynamic nature of device availability, the lack of a predictive model of population behavior, and feedback loops.

以隐私保护的方式向学术合作伙伴提供此类统计数据或模型;另一个有希望的方向是最近引入的Flower框架[66]用于联邦学习研究。

## 7.3系统参数调整

实际的联邦学习是多目标优化的一种形式：虽然第一阶目标是最大化模型质量指标，如损失或准确性，但其他重要的考虑因素是

·收敛速度

·吞吐量（例如，轮数、数据量或设备数量）

·模型公平性、隐私性和鲁棒性（见第6.3节）

·服务器和客户端上的资源使用

这些目标可能处于紧张状态。例如，最大化轮吞吐量可能会引入偏差或通过偏好具有很少或没有数据的高性能设备来损害准确性。通过增加模型复杂度来最大化低训练损失将使内存较少、样本较多或较大或CPU较慢的设备处于不利地位。在训练过程中以这种方式引起的偏差或公平性可能很难在评估阶段检测到，因为它通常使用相同的平台，因此会受到类似的偏差。

各种控制措施影响上述指标。有些是数据中心设置中熟悉的，特别是模型特定的设置和学习算法超参数。其他一些是针对联邦学习的：

- 每轮客户端：完成一轮所需的最小设备数量M，以及开始一轮所需的设备数量M。

- 服务器端调度：除了最简单的情况外，联邦学习系统将同时在多个模型上运行：支持多个租户;针对不同用例在相同数据上训练模型;支持实验和架构或超参数网格搜索;以及同时运行训练和评估工作负载。服务器需要决定为传入设备提供哪些任务，这是调度问题的一个实例：将工作（培训或评估任务）分配给资源（设备）。因此，通常的挑战出现了：理想的资源分配应该是公平的，避免饥饿，最小化等待时间，并同时支持相对优先级。

- 设备端调度：如第7.2节所述，各种约束控制设备何时可以连接到服务器并执行工作。在这些约束内，可以做出各种调度选择。一个极端是连接到服务器并尽可能频繁地运行计算，导致服务器和设备上的高负载和资源使用。另一种选择是固定间隔，但需要调整以反映外部因素，例如整体和每轮的设备数量。Google开发的联邦学习系统旨在与称为步调转向的流控制机制取得平衡[81]，服务器指示设备何时返回。这样的动态系统使得能够针对大的群体进行时间负载平衡以及将连接尝试“集中”到特定的时间点以达到阈值M。 由于设备可用性的随机性和动态性、缺乏群体行为的预测模型以及反馈回路，开发这样的机制是困难的。

Defining reasonable composite objective functions, and designing algorithms to automatically tune these settings, has not been explored yet in the context of federated learning systems and hence remains a topic of future research.

### 7.4 On-Device Runtime

While numerous frameworks exist for data center training, the options for training models on resource constrained devices are fairly limited. Machine Learning models and training procedures are typically authored in a high level language such as Python. For federated learning, this description encompasses device and server computations that are executed on the target platform and exchange data over a network connection, necessitating

- A means of serializing and dynamically transmitting local pieces of the total computation (e.g., the server-side update to the model, or the local client training procedure).
- A means to interpret or execute such a computation on the target platform (server or device).
- A stable network protocol for data exchange between participating devices and servers.

One extreme form of a representation is the original high-level description, e.g. a Python TensorFlow program [2]. This would require a Python interpreter with TensorFlow backend, which may not be a feasible choice for end-user devices due to resource constraints (binary size, memory use), performance limitations, or security concerns.

Another extreme representation of a computation is machine code of the target architecture, e.g. ARM64 instructions. This requires a compiler or re-implementation of a model in a lower-level language such as C++, and deployment computations will typically be subject to the restrictions that apply to deployment of binary code (see Section 7.1), introducing prohibitive latencies for executing novel computations.

Intermediate representations that can be compiled or interpreted with a runtime on the target platform strike a balance between flexibility and efficiency. However, such runtimes are currently not widely available. For instance, Google's FL system [81] relies on TensorFlow for both server and device side execution as well as model and parameter transfer, but this choice suffers from several shortcomings:

- It offers no easy path to devices for alternative front ends such as PyTorch [370], JAX [86] or CNTK [410].
- The runtime is not developed or optimized for resource constrained environments, incurring a large binary size, high memory use and comparatively low performance.
- The intermediate representation `GraphDef` used by TensorFlow is not standardized or stable, and version skew between the frontend and older on-device backends causes frequent compatibility challenges.

Other alternatives include more specialized runtimes that support only a subset of the frontend's capabilities, for instance training specific model types only, requiring changes and long update cycles whenever new model architectures or training algorithms are to be used. An extreme case would be a runtime that is limited and optimized to train a single type of model.

An ideal on-device runtime would have the following characteristics:

定义合理的复合目标函数，并设计算法来自动调整这些设置，尚未在联邦学习系统的背景下进行探索，因此仍然是未来研究的主题。

## 7.4设备上安装

虽然存在许多用于数据中心培训的框架，但在资源受限的设备上培训模型的选项相当有限。机器学习模型和训练过程通常是用高级语言（如Python）编写的。对于联合学习，此描述涵盖在目标平台上执行并通过网络连接交换数据的设备和服务器计算，

·串行化和动态传输总计算的局部片段的手段（例如，对模型的服务器端更新或本地客户端训练过程）。

·在目标平台（服务器或设备）上解释或执行这种计算的手段。

·用于参与设备和服务器之间数据交换的稳定网络协议。

表示的一种极端形式是原始的高级描述，例如Python TensorFlow程序[2]。这需要一个带有TensorFlow后端的Python解释器，由于资源限制（二进制大小，内存使用），性能限制或安全问题，这可能不是最终用户设备的可行选择。

计算的另一种极端表示是目标架构的机器代码，例如ARM 64指令。这需要编译器或用低级语言（如C++）重新实现模型，并且部署计算通常会受到适用于二进制代码部署的限制（参见第7.1节），为执行新计算引入了禁止的延迟。

可以在目标平台上使用运行时编译或解释的中间表示在灵活性和效率之间取得了平衡。然而，这样的运行时目前并不广泛可用。例如，Google的FL系统[81]依赖TensorFlow进行服务器和设备端执行以及模型和参数传输，但这种选择有几个缺点：

·它没有为替代前端（如PyTorch [370]，JAX [86]或CNTK [410]）提供简单的设备路径。

·运行时没有针对资源受限的环境进行开发或优化，导致二进制大小较大，内存使用量较高，性能相对较低。

TensorFlow使用的中间表示GraphDef不标准化或不稳定，前端和较旧的设备后端之间的版本偏差导致频繁的兼容性挑战。

其他替代方案包括更专门的运行时，仅支持前端功能的子集，例如仅训练特定的模型类型，每当使用新的模型架构或训练算法时，都需要更改和长的更新周期。一个极端的情况是，运行时被限制和优化以训练单一类型的模型。

理想的设备上运行时应具有以下特征：

1. Lightweight: small binary size, or pre-installed; low memory and power profile.
2. Performant: low startup latency; high throughput, supports hardware acceleration.
3. Expressive: supports common data types and computations including backpropagation, variables, control flow, custom extensions.
4. Stable and compact format for expressing data and computations.
5. Widely available: portable open source implementation.
6. Targetable by commonly used ML frameworks / languages..
7. Ideally also supports inference, or if not, building personalized models for an inference runtime.

To our best knowledge no solution exists yet that satisfies these requirements, and we expect the limited ability to run ML training on end user devices to become a hindrance to adoption of federated technologies.

### 7.5 The Cross-Silo Setting

The system challenges arising in the scenario of cross-silo federated learning take a considerably different form. As outlined in Table 1, clients are fewer in number, more powerful, reliable, and known / addressable, eliminating many of the challenges from the cross-device setting, while allowing for authentication and verification, accounting, and contractually enforced penalties for misbehavior. Nonetheless, there are other sources of heterogeneity, including the features and distribution of data, and possibly the software stack used for training.

While the infrastructure in the cross-device setting (from the device-side data generation to the server logic) is typically operated by one or few organizational entities (the application, operating system, or device manufacturer), in the cross-silo setting, many different entities are involved. This may lead to high coordination and operational cost due to differences in:

- *How data is generated, pre-processed and labeled.* Learning across silos will require data normalization which may be difficult when such data is collected and stored differently (e.g. use of different medical imaging systems, and inconsistencies in labeling procedures, annotations, and storage formats).
- *Which software at which version powers training.* Using the same software stack in every silo—possibly delivered alongside the model using container technologies as done by FATE [33]—eliminates compatibility concerns, but such frequent and centrally distributed software delivery may not be acceptable to all involved parties. An alternative that is more similar to the cross-device setting would be to standardize data and model formats and communication protocols. See IEEE P3652.1 "Federated Machine Learning Working Group" for a related effort in this direction.
- *The approval process for how data may or may not be used.* While this process is typically centralized in the cross-device scenario, the situation is likely different in cross-silo settings where many organizational entities are involved, and may be increasingly difficult when training spans different jurisdictions with varying data protection regulations. Technical infrastructure may be of help here by establishing data annotations that encode access policies, and infrastructure enforce them; for instance, limiting the use of certain data to specific models, or encoding minimum aggregation requirements such as "require at least $M$ clients per round".

1.轻量级：小的二进制文件大小，或预先安装;低内存和功率配置文件。

2.性能：低启动延迟;高吞吐量，支持硬件加速。

3.表达：支持常见的数据类型和计算，包括反向传播，变量，控制流，自定义扩展。

4.用于表达数据和计算的稳定而紧凑的格式。

5.广泛可用：可移植的开源实现。

6.适用于常用的ML框架/语言。

7.理想情况下还支持推理，或者如果不支持，则为推理运行时构建个性化模型。据我们所知，目前还没有满足这些要求的解决方案，我们预计在最终用户设备上运行ML培训的能力有限，这将成为采用联邦技术的障碍。

## 7.5跨筒仓设置

在跨筒仓联合学习的场景中出现的系统挑战采取了相当不同的形式。如表1所示，客户端数量更少，功能更强大，更可靠，并且已知/可寻址，消除了跨设备设置的许多挑战，同时允许身份验证和验证，会计以及对不当行为的合同强制处罚。尽管如此，还有其他的异质性来源，包括数据的特征和分布，可能还有用于训练的软件栈。

虽然跨设备设置中的基础设施（从设备端数据生成到服务器逻辑）通常由一个或几个组织实体（应用程序、操作系统或设备制造商）操作，但在跨竖井设置中，涉及许多不同的实体。这可能导致协调和运营成本高，原因是以下方面的差异：

- 数据如何生成、预处理和标记。跨孤岛学习将需要数据标准化，当这些数据以不同方式收集和存储时（例如，使用不同的医学成像系统，以及标签程序、注释和存储格式的不一致），这可能很困难。

- 哪个版本的软件支持培训。在每个筒仓里使用相同的软件-
可能使用FATE [33]所做的容器技术与模型一起交付-消除了兼容性问题，但这种频繁和集中分布的软件交付可能不会被所有相关方接受。另一种更类似于跨设备设置的方法是标准化数据和模型格式以及通信协议。参见IEEE P3652.1“联邦机器学习工作组”，以了解该方向的相关工作。

- 关于如何使用或不使用数据的批准流程。虽然此过程通常在跨设备场景中集中进行，但在涉及许多组织实体的跨竖井设置中，情况可能有所不同，并且当培训跨越具有不同数据保护法规的不同司法管辖区时，情况可能会越来越困难。技术基础设施可以通过建立对访问策略进行编码的数据注释来提供帮助，并且基础设施可以执行这些注释;例如，将某些数据的使用限制在特定模型中，或者对最低聚合要求进行编码，例如“每轮至少需要M个客户端”。

Another potential difference in the cross-silo setting is data partitioning: Data in the cross-device setting is typically assumed to be partitioned by examples, all of which have the same features (horizontal partitioning). In the cross-silo setting, in addition to partitioning by examples, partitioning by features is of practical relevance (vertical partitioning). An example would be two organizations, e.g. a bank and a retail company, with an overlapping set of customers, but different information (features) associated with them. For a discussion focusing on the algorithmic aspects, please see section 2.2. Learning with feature-partitioned data may require different communication patterns and additional processing steps e.g. for entity alignment and dealing with missing features.

## 7.6 Executive Summary

While production grade systems for cross-device federated learning operate successfully [81, 26], various challenges remain:

- Frequent and large scale deployment of updates, monitoring, and debugging is challenging (Section 7.1).
- Differences in device availability induce various forms of bias; defining, quantifying and mitigating them remains a direction for future research (Section 7.2).
- Tuning system parameters is difficult due to the existence of multiple, potentially conflicting objectives (Section 7.3).
- Running ML workloads on end user devices is hampered by the lack of a portable, fast, small footprint, and flexible runtime for on-device training (Section 7.4).

Systems for cross-silo settings (Section 7.5) face largely different issues owing to differences in the capabilities of compute nodes and the nature of the data being processed.

跨竖井设置中的另一个潜在差异是数据分区：跨设备设置中的数据通常被假设为按示例进行分区，所有示例都具有相同的特性（水平分区）。在跨竖井设置中，除了按示例划分之外，按特性划分也具有实际意义（垂直划分）。一个例子是两个组织，例如银行和零售公司，具有重叠的客户集，但与它们相关联的信息（特征）不同。有关算法方面的讨论，请参见第2.2节。使用特征分区数据进行学习可能需要不同的通信模式和额外的处理步骤，例如用于实体对齐和处理丢失的特征。

## 7.6执行摘要

虽然跨设备联合学习的生产级系统成功运行[81，26]，但仍然存在各种挑战：

·频繁和大规模部署更新、监控和调试具有挑战性（第7.1节）。

·设备可用性的差异会导致各种形式的偏倚;定义、量化和减轻偏倚仍然是未来研究的方向（第7.2节）。

·由于存在多个潜在冲突的目标，调整系统参数是困难的（第7.3节）。

·在终端用户设备上运行ML工作负载受到缺乏便携、快速、占用空间小和灵活的设备上培训运行时的阻碍（第7.4节）。

由于计算节点的能力和所处理数据的性质不同，跨筒仓设置的系统（第7.5节）面临着很大程度上不同的问题。

# 8 Concluding Remarks

Federated learning enables distributed client devices to collaboratively learn a shared prediction model while keeping all the training data on device, decoupling the ability to do machine learning from the need to store the data in the cloud. This goes beyond the use of local models that make predictions on mobile devices by bringing model training to the device as well.

In recent years, this topic has undergone an explosive growth of interest, both in industry and academia. Major technology companies have already deployed federated learning in production, and a number of startups were founded with the objective of using federated learning to address privacy and data collection challenges in various industries. Further, the breadth of papers surveyed in this work suggests that federated learning is gaining traction in a wide range of interdisciplinary fields: from machine learning to optimization to information theory and statistics to cryptography, fairness, and privacy.

Motivated by the growing interest in federated learning research, this paper discusses recent advances and presents an extensive collection of open problems and challenges. The system constraints impose efficiency requirements on the algorithms in order to be practical, many of which are not particularly challenging in other settings. We argue that data privacy is not binary and present a range of threat models that are relevant under a variety of assumptions, each of which provides its own unique challenges.

The open problems discussed in this work are certainly not comprehensive, they reflect the interests and backgrounds of the authors. In particular, we do not discuss any non-learning problems which need to be solved in the course of a practical machine learning project, and might need to be solved based on decentralized data [382]. This can include simple problems such as computing basic descriptive statistics, or more complex objectives such as computing the head of a histogram over an open set [510]. Existing algorithms for solving such problems often do not always have an obvious "federated version" that would be efficient under the system assumptions motivating this work or do not admit a useful notion of data protection. Yet another set of important topics that were not discussed are the legal and business issues that may motivate or constrain the use of federated learning.

We hope this work will be helpful in scoping further research in federated learning and related areas.


# Acknowledgments

The authors would like to thank Alex Ingerman and David Petrou for their useful suggestions and insightful comments during the review process.

# 8结论

联合学习使分布式客户端设备能够协作学习共享的预测模型，同时将所有训练数据保存在设备上，将机器学习的能力与将数据存储在云中的需求解耦。这超越了使用本地模型，通过将模型训练也带到设备上来在移动的设备上进行预测。

近年来，这一主题在工业界和学术界都受到了爆炸性的关注。
主要的技术公司已经在生产环境中部署了联邦学习，许多初创公司的目标是使用联邦学习来解决各个行业的隐私和数据收集挑战。此外，这项工作中调查的论文的广度表明，联邦学习正在广泛的跨学科领域获得关注：从机器学习到优化，到信息理论和统计学，再到密码学，公平性和隐私。

由于对联邦学习研究的兴趣日益浓厚，本文讨论了最新进展，并提出了广泛的开放问题和挑战。为了实用，系统约束对算法施加了效率要求，其中许多在其他设置中不是特别具有挑战性。我们认为，数据隐私不是二元的，并提出了一系列的威胁模型，这些模型在各种假设下都是相关的，每一个都有自己独特的挑战。

在这项工作中讨论的开放问题当然不是全面的，它们反映了作者的兴趣和背景。特别是，我们不讨论在实际机器学习项目过程中需要解决的任何非学习问题，并且可能需要基于分散数据来解决[382]。这可以包括简单的问题，如计算基本的描述性统计，或更复杂的目标，如计算开集上直方图的头部[510]。解决此类问题的现有算法通常并不总是有一个明显的“联邦版本”，这将是有效的系统假设下，激励这项工作或不承认一个有用的概念的数据保护。还有一组没有讨论的重要主题是可能激励或限制联邦学习使用的法律的和商业问题。

我们希望这项工作将有助于进一步研究联邦学习和相关领域。

# 致谢

作者要感谢Alex Ingerman和大卫Petrou在审查过程中提出的有用建议和有见地的评论。

# References


[1] Lattigo 2.0.0. Online: http://github.com/ldsec/lattigo, October 2020. EPFL-LDS.

[2] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

[3] 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*, pages 308–318. ACM, 2016.

[4] Omid Abari, Hariharan Rahul, and Dina Katabi. Over-the-air function computation in sensor networks. *CoRR*, abs/1612.02307, 2016. URL http://arxiv.org/abs/1612.02307.

[5] Nazmiye Ceren Abay, Yan Zhou, Murat Kantarcioglu, Bhavani Thuraisingham, and Latanya Sweeney. Privacy preserving synthetic data release using deep learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 510–526. Springer, 2018.

[6] John M Abowd and Ian M Schmutte. An economic analysis of privacy protection and statistical accuracy as social choices. *American Economic Review*, 109(1):171–202, 2019.

[7] Jayadev Acharya, Clément L Canonne, and Himanshu Tyagi. Inference under information constraints i: Lower bounds from chi-square contraction. *IEEE Transactions on Information Theory*, 66(12):7835–7855, 2020.

[8] Gergely Ács and Claude Castelluccia. I have a DREAM!: DIfferentially PrivatE smart Metering. In *Proceedings of the 13th International Conference on Information Hiding*, IH'11, pages 118–132, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-24177-2. URL http://dl.acm.org/citation.cfm?id=2042445.2042457.

[9] Naman Agarwal, Ananda Theertha Suresh, Felix X. Yu, Sanjiv Kumar, and Brendan McMahan. cpSGD: Communication-efficient and differentially-private distributed SGD. In *Advances in Neural Information Processing Systems*, pages 7564–7575, 2018.

[10] Nitin Agrawal, Ali Shahin Shamsabadi, Matt J. Kusner, and Adrià Gascón. QUOTIENT: two-party secure neural network training and prediction. In *In Proceedings of the ACM Conference on Computer and Communication Security (CCS)*, 2019.

[11] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *ACM SIGMOD International Conference on Management of Data*, 2000.

[12] Carlos Aguilar-Melchor and Philippe Gaborit. A lattice-based computationally-efficient private information retrieval protocol. *Cryptol. ePrint Arch., Report*, 446, 2007.

[13] Carlos Aguilar-Melchor, Joris Barrier, Laurent Fousse, and Marc-Olivier Killijian. XPIR: Private information retrieval for everyone. *Proceedings on Privacy Enhancing Technologies*, 2016(2):155–174, 2016.

[14] ai.google. Under the hood of the Pixel 2: How AI is supercharging hardware, 2018. URL https://ai.google/stories/ai-in-hardware/. Retrieved Nov 2018.

[15] ai.intel. Federated learning for medical imaging, 2019. URL https://www.intel.ai/federated-learning-for-medical-imaging/. Retrieved Aug 2019.

# 引用


[1]Latestry 2.0.0.在线：http：//github.com/ldsec/latitude，2020年10月。EPFL-LDS。

[2]Mart 'ın Abadi，Ashish Agarwal，Paul Barham，尤金Brevdo，Zhifeng Chen，克雷格Citro，Greg S. Corrado，Andy Davis，Jeffrey Dean，Matthieu迪文，Sanjay Ghemawat，Ian Goodfellow，Andrew Harp，Geoffrey欧文，Michael Isard，Yangqing Jia，Rafal Jozefowicz，Lukasz Kaiser，Manjunath Kudlur，Josh Levenberg，Danish Man 'e，Rajat Monga，Sherry摩尔，Derek Murray，Chris Olah，Mike Schuster，Jonathe Shlens，Benoit Steiner，Ilya Sutskever，Kunal Talwar，Paul Tucker，Vincent Vanhoucke，Vijay Oriol Vinyals，Pete Warden，Martin Wattenberg，Martin Wicke，Yuan Yu，and Xiaoqiang Zheng. TensorFlow：异构系统上的大规模机器学习，2015年。网址https://www.tensorflow.org/。软件可从tensorflow.org获得。

[3]Martin Abadi、Andy Chu、Ian Goodfellow、H Brendan McMahan、Ilya Mironov、Kunal Talwar和Li Zhang。深度学习与差分隐私2016年ACM SIGSAC计算机和通信安全会议论文集，第308-318页。ACM，2016.

[4]Omid Abari Hariharan Rahul和Dina Katabi。传感器网络中的空中功能计算。CoRR，abs/1612.02307，2016。网址http://arxiv.org/abs/1612.02307。

[5]Nazmiye Ceren Abay、Yan Zhou、穆拉特Kantarcioglu、Bhavani Thuraisingham和Latanya Sweeney。使用深度学习的隐私保护合成数据发布。在联合欧洲会议机器学习和知识发现数据库，第510-526页。Springer，2018年。

[6]作者声明：John M Abowd，Ian M Schmutte.隐私保护和统计准确性作为社会选择的经济分析。美国经济评论，109（1）：171-202，2019。

[7]Jayadev Acharya，Cl 'ement L Canonne，and Himanshu Tyagi.信息约束下的推理i：来自卡方收缩的下限。IEEE Transactions on Information Theory，66（12）：7835-7855，2020。

[8]杰奎琳·阿克斯和克劳德·卡斯泰卢西亚我有一个梦想！：不同的隐私智能计量。进行中-第13届信息隐藏国际会议，IH'11，第118-132页，柏林，海德堡，2011。史普林格出版社ISBN 978-3-642-24177-2。网址http://dl.acm.org/citation.cfm? id= 2042445.2042457。

[9]放大图片创作者：Ananda Theertha Suresh，Felix X. Yu，Sanjiv Kumar，and Brendan McMahan. cpSGD：通信高效且差异私有的分布式SGD。神经信息处理系统的进展，第7564-7575页，2018年。

[10]Nitin Agrawal、Ali Shahin Shamsabadi、Matt J. Kusner和Adri`a Gasc 'on。英文名：Two Party Secure神经网络训练和预测在ACM计算机和通信安全会议（CCS）的会议记录中，2019年。

[11]Rakesh Agrawal和Ramakrishnan Srikant。隐私保护数据挖掘。ACM SIGMOD数据管理国际会议，2000年。

[12]卡洛斯阿吉拉尔梅尔乔和菲利普哈博里特。一个基于格的计算效率高的私人信息检索协议。密码醇ePrint Arch.，报告，446，2007年。

[13]卡洛斯阿吉拉尔-梅尔乔、乔里斯·巴瑞、劳伦特·福斯和马克-奥利维耶·基利坚。XPIR：每个人的私人信息检索。Proceedings on Privacy Enhancing Technologies，2016（2）：155-174，2016.

[14] ai.google.在Pixel 2的引擎盖下：AI如何为硬件增压，2018年。网址https：//ai.google/stories/ai-in-hardware/. 2018年11月恢复。

[15] ai.intel.联邦学习医学成像，2019年。网址https://www.intel.ai/ federated-learning-for-medical-imaging/。2019年8月恢复。

[16] Asra Ali, Tancrède Lepoint, Sarvar Patel, Mariana Raykova, Phillipp Schoppmann, Karn Seth, and Kevin Yeo. Communication-computation trade-offs in PIR. *IACR Cryptol. ePrint Arch.*, 2019:1483, 2019.

[17] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. In *NIPS - Advances in Neural Information Processing Systems*, pages 1709–1720, 2017.

[18] Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li. Byzantine stochastic gradient descent. In *NIPS*, 2018.

[19] Inês Almeida and João Xavier. DJAM: Distributed Jacobi Asynchronous Method for Learning Personal Models. *IEEE Signal Processing Letters*, 25(9):1389–1392, 2018.

[20] Scott Ames, Carmit Hazay, Yuval Ishai, and Muthuramakrishnan Venkitasubramaniam. Ligero: Lightweight sublinear arguments without a trusted setup. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, 2017.

[21] Kareem Amin, Alex Kulesza, Andres Munoz, and Sergei Vassilvtiskii. Bounding user contributions: A bias-variance trade-off in differential privacy. In *International Conference on Machine Learning*, pages 263–271, 2019.

[22] androidtrusty. Android Trusty TEE. `https://source.android.com/security/trusty`, 2019. Accessed: 2019-12-05.

[23] Sebastian Angel, Hao Chen, Kim Laine, and Srinath T. V. Setty. PIR with compressed queries and amortized query processing. In *IEEE Symposium on Security and Privacy*, pages 962–979. IEEE Computer Society, 2018.

[24] George J Annas. HIPAA regulations-a new era of medical-record privacy? *New England Journal of Medicine*, 348(15):1486–1490, 2003.

[25] Apple. Private Federated Learning (NeurIPS 2019 Expo Talk Abstract). `https://nips.cc/ExpoConferences/2019/schedule?talk_id=40`, 2019.

[26] Apple. Designing for privacy (video and slide deck). Apple WWDC, `https://developer.apple.com/videos/play/wwdc2019/708`, 2019.

[27] Toshinori Araki, Jun Furukawa, Yehuda Lindell, Ariel Nof, and Kazuma Ohara. High-throughput semi-honest secure three-party computation with an honest majority. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 805–817. ACM, 2016.

[28] armtrustzone. Arm TrustZone Technology. `https://developer.arm.com/ip-products/security-ip/trustzone`, 2019. Accessed: 2019-12-05.

[29] Mahmoud Assran, Nicolas Loizou, Nicolas Ballas, and Michael Rabbat. Stochastic gradient push for distributed deep learning. In *ICML*, 2019.

[30] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *ICML*, 2018.

[31] Sean Augenstein, H. Brendan McMahan, Daniel Ramage, Swaroop Ramaswamy, Peter Kairouz, Mingqing Chen, Rajiv Mathews, and Blaise Aguera y Arcas. Generative models for effective ML on private, decentralized datasets, 2019. URL `https://arxiv.org/abs/1911.06679`.

[32] PyVertical Authors. Pyvertical, 2020. URL `https://github.com.cnpmjs.org/OpenMined/PyVertical`.

[33] The FATE Authors. Federated AI technology enabler, 2019. URL `https://www.fedai.org/`.

[34] The Fedlearner Authors. Fedlearner, 2020. URL `https://github.com/bytedance/fedlearner`.

[16]Asra Ali、Tancr 'ede Lepoint、Sarvar Patel、Mariana Raykova、Phillipp Schoppmann、Karn Seth和Kevin Yeo。PIR中的通信-计算权衡。IACR Cryptol。ePrint Arch.，2019年：1483，2019年。

[17]丹·阿利斯塔、德米扬·格鲁比奇、杰里·李、富冈亮太和米兰·沃伊诺维奇。QSGD：沟通效率SGD通过梯度量化和编码。在NIPS -神经信息处理系统的进展，第1709-1720页，2017年。

[18]Dan Alistarh、Zeyuan Allen-Zhu和Jerry Li。拜占庭随机梯度下降法。在NIPS，2018年。

[19]在埃斯·阿尔梅达和乔·奥·泽维尔。DJAM：学习个人模型的分布式雅可比异步方法。IEEE Signal Processing Letters，25（9）：1389-1392，2018。

[20]Scott艾姆斯、Carmit Hazay、Yuval Ishai和Muthuramakrishnan Venkitasubramaniam。Ligero：轻量级没有可信设置的次线性参数。2017年ACM SIGSAC计算机和通信安全会议论文集，CCS '17，2017。

[21]Kareem Amin，Alex Kulesza，Andres Munoz，and Sergei Vassilvtiskii.限制用户贡献：偏见-差异隐私中的方差权衡。在国际机器学习会议上，第263-271页，2019年。

[22]androidtrusty. Android Trusty TEE. https：//source.android.com/security/trusty，2019.访问时间：2019-12-05。

[23]塞巴斯蒂安天使，陈浩，金莱恩，和Srinath T。V. Setty。具有压缩查询和摊销查询处理的PIR。IEEE Symposium on Security and Privacy，第962-979页。IEEE计算机协会，2018年。

[24]乔治J安纳斯。健康保险责任法案--医疗记录隐私的新时代？新英格兰医学杂志，348（15）：1486-1490，2003。

[25]苹果Private Federated Learning（NeurIPS 2019 Expo Talk Abstract）. https://nips.cc/ExpoConferences/2019/schedule? talk_id=40，2019.

[26]苹果隐私设计（视频和幻灯片）。Apple WWDC，https://developer.apple.com/ videos/play/wwdc2019/708，2019。

[27]Toshinori Araki、Jun Furukawa、Yehuda Lindell、Ariel Nof和Kazuma大原。高通量半诚实

确保三方计算与诚实的多数。2016年ACM SIGSAC计算机和通信安全会议论文集，第805-817页。ACM，2016.

[28]armtrustzone。Arm TrustZone技术https://developer.arm.com/ip-products/ security-ip/trustzone，2019年。访问时间：2019-12-05。

[29]Mahmoud Assran，Nicolas Loizou，Nicolas Ballas，and Michael Rabbat.用于分布式深度学习的随机梯度推送。2019年，在ICML中。

[30]阿尼什·阿塔利，尼古拉斯·卡利尼，还有大卫瓦格纳。模糊梯度给予一种虚假的安全感：规避对抗性示例的防御。ICML，2018年。

[31]Sean Augenstein，H。Brendan McMahan，Daniel Ramage，Swaroop Ramaswamy，Peter Kairouz，明青

Chen，Rajiv马修斯，Blaise Aguera y Arcas.在私有、去中心化数据集上实现有效ML的生成模型，2019年。网址https://arxiv.org/abs/1911.06679。

[32]PyVertical作者Pyvertical，2020年。网址https://github.com.cnpmjs.org/OpenMined/ PyVertical。

[33]命运的作者Federated AI技术推动者，2019年。网址https://www.fedai.org/。

[34]Fedlearner作者Fedlearner，2020年。网址https://github.com/bytedance/fedlearner。

[35] The Leaf Authors. Leaf, 2019. URL https://leaf.cmu.edu/.

[36] The PaddleFL Authors. PaddleFL, 2019. URL https://github.com/PaddlePaddle/PaddleFL.

[37] The PaddlePaddle Authors. PaddlePaddle, 2019. URL http://www.paddlepaddle.org/.

[38] The TFF Authors. TensorFlow Federated, 2019. URL https://www.tensorflow.org/federated.

[39] Brendan Avent, Yatharth Dubey, and Aleksandra Korolova. The power of the hybrid model for mean estimation. *Proceedings on Privacy Enhancing Technologies (PETS)*, 2020(4):48 – 68, 01 Oct. 2020. doi: https://doi.org/10.2478/popets-2020-0062. URL https://content.sciendo.com/view/journals/popets/2020/4/article-p48.xml.

[40] Brendan Avent, Aleksandra Korolova, David Zeber, Torgeir Hovden, and Benjamin Livshits. BLENDER: Enabling local search with a hybrid differential privacy model. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 747–764, Vancouver, BC, August 2017. USENIX Association. ISBN 978-1-931971-40-9. URL https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/avent.

[41] Pranjal Awasthi, Corinna Cortes, Yishay Mansour, and Mehryar Mohri. Beyond individual and group fairness. *CoRR*, abs/2008.09490, 2020.

[42] László Babai, Lance Fortnow, Leonid A. Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *STOC*, pages 21–31. ACM, 1991.

[43] Eugene Bagdasaryan and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. *CoRR*, abs/1905.12101, 2019. URL http://arxiv.org/abs/1905.12101.

[44] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. *arXiv preprint arXiv:1807.00459*, 2018.

[45] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II*, pages 638–667, 2019. doi: 10.1007/978-3-030-26951-7_22. URL https://doi.org/10.1007/978-3-030-26951-7_22.

[46] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. Private summation in the multi-message shuffle model. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, page 657–676. ACM, 2020.

[47] Borja Balle, Peter Kairouz, H. Brendan McMahan, Om Thakkar, and Abhradeep Thakurta. Privacy amplification via random check-ins, 2020.

[48] Assi Barak, Daniel Escudero, Anders P. K. Dalskov, and Marcel Keller. Secure evaluation of quantized neural networks. *IACR Cryptology ePrint Archive*, 2019:131, 2019. URL https://eprint.iacr.org/2019/131.

[49] Leighton Pate Barnes, Yanjun Han, and Ayfer Ozgur. Lower bounds for learning distributions under communication constraints via fisher information. *Journal of Machine Learning Research*, 21(236):1–30, 2020. URL http://jmlr.org/papers/v21/19-737.html.

[50] Leighton Pate Barnes, Huseyin A. Inan, Berivan Isik, and Ayfer Ozgur. rtop-k: A statistical estimation approach to distributed sgd. *arXiv preprint arXiv:2005.10761*, 2020.

[51] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. http://www.fairmlbook.org.

[52] Moran Baruch, Gilad Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for distributed learning. *arXiv preprint arXiv:1902.06156*, 2019.

[35]叶作者。Leaf，2019年。网址https://leaf.cmu.edu/。

[36]PaddleFL作者PaddleFL，2019年。网址https://github.com/PaddlePaddle/PaddleFL。

[37]PaddlePaddle作者PaddlePaddle，2019年。网址http://www.paddlepaddle.org/。

[38]TFF作者TensorFlow Federated，2019年。网址https://www.tensorflow.org/federated。

[39]布兰登·艾文特，亚瑟王·杜贝，亚历山德拉·科罗洛娃。混合模型用于均值估计的功效。
Proceedings on Privacy Enhancing Technologies（PETS），2020（4）：48 - 68，01 Oct. 2020. doi：https://doi.org/ 10.2478/popets-2020-0062。网址https://content.sciendo.com/view/journals/popets/2020/4/article-p48.xml。

[40]布伦丹·艾文特、亚历山德拉·科罗洛娃、大卫·泽贝尔、托盖尔·霍夫登和本杰明·利夫希茨。搅拌机：
使用混合差分隐私模型实现本地搜索。在第26届USENIX安全研讨会（USENIX Security 17），第747-764页，温哥华，BC，2017年8月。USENIX协会。ISBN 978-1-931971-40-9。网址https://www.usenix.org/conference/usenixsecurity17/technical-sessions/ presentation/avent.

[41]Pranjal Awasthi，Corinna科尔特斯，Yishay Mansour和Mehryar Mohri。超越个人和群体的公平。
CoRR，abs/2008.09490，2020。

[42]放大图片作者：L 'aszl 'o巴拜，Lance Fortnow，Leonid A.莱文和马里奥·塞格迪在多对数时间内检查计算。在STOC中，第21-31页。ACM，1991年。

[43]尤金·巴格达萨良和维塔利·什马蒂科夫。差异隐私对模型准确性有不同的影响。CoRR，abs/1905.12101，2019。网址http://arxiv.org/abs/1905.12101。

[44]尤金·巴格达萨良、安德烈亚斯·维特、华毅青、黛博拉·埃斯特林和维塔利·什马提科夫。如何后门联邦学习。arXiv预印本arXiv：1807.00459，2018。

[45]博尔哈巴莱，詹姆斯贝尔，阿德里阿加塞翁和科比尼辛。洗牌模式的隐私保护毯。在

Advances in Cryptology -密码学进展-密码学研究所2019 -第39届国际密码学年会，圣巴巴拉，加利福尼亚州，美国，2019年8月18日至22日，会议记录，第二部分，第638-667页，2019年。doi：10.1007/978-3-030-26951-7\ 22.网址https://doi.org/10.1007/978-3-030-26951-7 22。

[46]博尔哈巴莱，詹姆斯贝尔，阿德里阿加塞翁和科比尼辛。多消息洗牌中的私有求和
模型2020年ACM SIGSAC计算机和通信安全会议论文集，第657-676页。ACM，2020年。

[47]放大图片作者：博尔哈巴莱，彼得·凯鲁兹，H.布兰登·麦克马汉奥姆·塔卡和阿赫拉迪普·塔库尔塔2020年，通过随机入住扩大隐私。

[48]放大图片作者：Assi Barak，丹尼尔Escudero，Anders P. K. Dalskov和Marcel Keller量化神经网络的安全评估
网络. IACR Cryptology ePrint Archive，2019：131，2019. 131.第一https://eprint.iacr.org/2019/

[49]Leighton Pate巴恩斯，Yanjun Han和Ayfer Ozgur。下的学习分布的下界-
通过Fisher信息的约束。Journal of Machine Learning Research，21（236）：1-30，2020。网址http://jmlr.org/papers/v21/19-737.html。

[50]放大图片作者：Leighton Pate巴恩斯，Huseyin A. Inan，Berivan Isik，and Ayfer Ozgur. rtop-k：分布式sgd的统计估计方法。arXiv预印本arXiv：2005.10761，2020。

[51]梭伦·巴罗卡斯、莫里茨·哈特和阿文德·纳拉亚南。公平和机器学习fairmlbook.org，2019年。
http://www.fairmlbook.org.

[52]莫兰·巴鲁克吉拉德·巴鲁克和约阿夫·戈德堡一点就够了：规避分布式学习的防御。arXiv预印本arXiv：1902.06156，2019。

[53] Raef Bassily and Adam Smith. Local, private, efficient protocols for succinct histograms. In *STOC*, pages 127–135, 2015.

[54] Raef Bassily, Uri Stemmer, Abhradeep Guha Thakurta, et al. Practical locally private heavy hitters. In *Advances in Neural Information Processing Systems*, pages 2288–2296, 2017.

[55] Debraj Basu, Deepesh Data, Can Karakus, and Suhas N Diggavi. Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations. *IEEE Journal on Selected Areas in Information Theory*, 1 (1):217–226, 2020.

[56] Jonathan Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

[57] Amos Beimel, Aleksandra Korolova, Kobbi Nissim, Or Sheffet, and Uri Stemmer. The power of synergy in differential privacy: Combining a small curator with local randomizers. In *Conference on Information-Theoretic Cryptography (ITC)*, 2020. URL `https://arxiv.org/abs/1912.08951`.

[58] James Henry Bell, Kallista A. Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly)logarithmic overhead. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, page 1253–1269. ACM, 2020.

[59] Aurélien Bellet, Rachid Guerraoui, Mahsa Taziki, and Marc Tommasi. Personalized and Private Peer-to-Peer Machine Learning. In *AISTATS*, 2018.

[60] Irwan Bello, Barret Zoph, Vijay Vasudevan, and Quoc V Le. Neural optimizer search with reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 459–468. JMLR. org, 2017.

[61] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

[62] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society, 2014.

[63] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Scalable zero knowledge with no trusted setup. In *CRYPTO (3)*, volume 11694 of *Lecture Notes in Computer Science*, pages 701–732. Springer, 2019.

[64] James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, pages 2546–2554, 2011.

[65] Martín Bertrán, Natalia Martínez, Afroditi Papadaki, Qiang Qiu, Miguel R. D. Rodrigues, Galen Reeves, and Guillermo Sapiro. Learning adversarially fair and transferable representations. In *ICML*, 2019.

[66] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Titouan Parcollet, and Nicholas D. Lane. Flower: A friendly federated learning research framework, 2020.

[67] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *Proceedings of the 36th International Conference on Machine Learning*, pages 634–643, 2019.

[68] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. Protection against reconstruction and its applications in private federated learning. *arXiv preprint arXiv:1812.00984*, 2018.

[69] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pages 1467–1474, USA, 2012. Omnipress. ISBN 978-1-4503-1285-1. URL `http://dl.acm.org/citation.cfm?id=3042573.3042761`.

[53]Raef Basily和Adam Smith。本地、私有、高效的协议，用于简洁的直方图。在STOC，第127-135页，2015年。

[54]Raef Basily，Uri Stemmer，Abhradeep Guha Thakurta，et al. Practical local private heavy hitters.神经信息处理系统的进展，第2288-2296页，2017年。

[55]Debraj Basu，Deepesh Data，Can Karakus和Suhas N Diggavi。Qsparse-local-sgd：分布式sgd，

量化、稀疏化和局部计算。IEEE Journal on Selected Areas in Information Theory，1（1）：217-226，2020。

[56]乔纳森巴克斯特。归纳偏差学习模型。人工智能研究杂志，12：149-198，
2000.

[57]Amos Beimel，Aleksandra Korolova，Kobbi Nissim，Or Sheffet，and Uri Stemmer.协同的力量，差分隐私：将一个小的策展人与本地随机化器相结合。在2020年的信息理论密码学会议（ITC）上。网址https://arxiv.org/abs/1912.08951。

[58]詹姆斯亨利贝尔，卡利斯塔A。Bonawitz，Adri`a Gasc 'on，Tancr`ede Lepoint，and Mariana Raykova.安全的单-服务器聚合与（聚）对数开销。2020年ACM SIGSAC计算机和通信安全会议论文集，第1253-1269页。ACM，2020年。

[59]Aur 'elien Bellet、Rachid Guerraoui、Mahsa Taziki和Marc Tommasi。个性化和私有对等机器学习。在AISTATS，2018年。

[60]Irwan Bello，Barret Zoph，Vijay Vasudevan，and Quoc V Le.带有强化的神经优化器搜索

学习第34届机器学习国际会议论文集-第70卷，第459- 468页。JMLR。org，2017.

[61]Shai Ben-David、John Blitzer、Koby Crammer、Alex Kulesza、Fernando佩雷拉和Jennifer Wortman Vaughan。从不同领域学习的理论。Machine learning，79（1-2）：151-175，2010.

[62]Eli Ben-Sasson、Alessandro Chiesa、Christina Garman、Matthew绿色色、Ian Miers、Eran Tromer和

维尔扎Zerocash：来自比特币的去中心化匿名支付。IEEE Symposium on Security and Privacy，第459-474页。IEEE计算机协会，2014年。

[63]Eli Ben-Sasson，Iddo Bentov，Yinon Horesh，and Michael Riabzev.可扩展的零知识，无需可信设置。在《计算机科学讲义》第11694卷第701-732页中。施普林格，2019年。

[64]詹姆斯·S·伯格斯特拉、雷米·巴德内、约瑟芬·本吉奥和巴拉斯·凯格尔。超参数优化算法。神经信息处理系统进展，第2546-2554页，2011年。

[65]作者：Mart 'ın Bertr' an，娜塔莉亚Mart 'ınez，Afroditi Papadaki，Qiang Qiu，Miguel R. D. Rodrigues，Galen Reeves，and Guillermo Sapiro.学习对抗性的公平和可转移的表示。2019年，在ICML中。

[66]丹尼尔J. Beutel，Taner Topal，Akhil Mathur，Xinchi Qiu，Titouan Parcollet和Nicholas D.巷Flower：一个友好的联邦学习研究框架，2020年。

[67]Arjun Nitin Bhagoji，Supriyo Chakraborty，Prateek Mittal和Seraphin Calo。分析联邦学习

通过对抗性的透镜。在第36届机器学习国际会议论文集，第634-643页，2019年。

[68]Abhishek Bhowmick，John Duchi，Julien Freudiger，Gaurav Kapoor，and Ryan Rogers.防止重构及其在私有联邦学习中的应用。arXiv预印本arXiv：1812.00984，2018。

[69]巴蒂斯塔·比吉奥布莱恩纳尔逊和帕维尔·拉斯科夫支持向量机的中毒攻击。在Pro-

第29届国际机器学习会议（ICML'12），第1467-1474页，美国，2012年。全媒体ISBN 978-1-4503-1285-1。网址http://dl.acm.org/ citation.cfm？id=3042573.3042761。

[70] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *ECML-PKDD*, pages 387–402. Springer, 2013.

[71] Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, 2012.

[72] R. Bitar and S. E. Rouayheb. Staircase-PIR: Universally robust private information retrieval. In *2018 IEEE Information Theory Workshop (ITW)*, pages 1–5, Nov 2018. doi: 10.1109/ITW.2018.8613532.

[73] Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Proceedings of the 26th Symposium on Operating Systems Principles*, SOSP '17, pages 441–459, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-5085-3. doi: 10.1145/3132747.3132769. URL `http://doi.acm.org/10.1145/3132747.3132769`.

[74] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *arXiv preprint arXiv:2003.03033*, 2020.

[75] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, 2017.

[76] Peva Blanchard, Rachid Guerraoui, Julien Stainer, et al. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. In *Advances in Neural Information Processing Systems*, pages 118–128, 2017.

[77] Dan Bogdanov, Riivo Talviste, and Jan Willemson. Deploying secure multi-party computation for financial data analysis - (short paper). In *Financial Cryptography*, volume 7397 of *Lecture Notes in Computer Science*, pages 57–64. Springer, 2012.

[78] Peter Bogetoft, Dan Lund Christensen, Ivan Damgård, Martin Geisler, Thomas P. Jakobsen, Mikkel Krøigaard, Janus Dam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, Michael I. Schwartzbach, and Tomas Toft. Secure multiparty computation goes live. In *Financial Cryptography*, volume 5628 of *Lecture Notes in Computer Science*, pages 325–343. Springer, 2009.

[79] K. A. Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. *arXiv preprint arXiv:1611.04482*, 2016.

[80] K. A. Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191. ACM, 2017.

[81] K. A. Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloé M Kiddon, Jakub Konečný, Stefano Mazzocchi, Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards federated learning at scale: System design. In *SysML 2019*, 2019. URL `https://arxiv.org/abs/1902.01046`.

[82] K. A. Bonawitz, Fariborz Salehi, Jakub Konečný, Brendan McMahan, and Marco Gruteser. Federated learning with autotuned communication-efficient secure aggregation. In *2019 53nd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2019.

[83] Dan Boneh, Elette Boyle, Henry Corrigan-Gibbs, Niv Gilboa, and Yuval Ishai. Zero-knowledge proofs on secret-shared data via fully linear PCPs. In *CRYPTO (3)*, volume 11694 of *Lecture Notes in Computer Science*, pages 67–97. Springer, 2019.

[70]Battista Biggio，Igino Corona，Davide Mallorca，Blaine Nelson，Nedim劳拉Srndi c，Pavel Laskov，Giacinto，and Fabio Roli.巴提奥·比亚诺·马略卡，大卫·马略卡，Blaine Nelson，Nedim？Evasion Attacks Against Machine Learning at Test Time（在测试时间内对机器学习进行攻击）在ECML—PKDD中，页387—402。Springer，2013。

[71]Nir Bitansky，Ran Canetti，Alessandro Chiesa，and Eran Tromer.从可提取的抗碰撞性到可拆卸的-关于知识的非交互式论证，然后又回来了。第三届理论计算机科学创新会议论文集，ITCS '12，2012。

[72]R. Bitar和S. E. Rouayheb。Staircase-PIR：通用强大的私有信息检索。在2018年IEEE信息理论研讨会（ITW），第1-5页，2018年11月。doi：10.1109/ITW.2018.8613532。

[73]Andrea Bittau，'Ulfar Erlingsson，彼得罗斯Maniatis，Ilya Mironov，Ananth Raghunathan，大卫李，米奇Rudominer，Ushasree Kode，Julien Tinnes，and Bernhard Seefeld. Prochlo：在人群中进行分析的强大隐私。在第26届操作系统原理研讨会论文集，SOSP '17，第441- 459页，纽约，纽约州，美国，2017年。ACM。ISBN 978-1-4503-5085-3。doi：10.1145/3132747.3132769。网址http://doi.acm.org/10.1145/3132747.3132769。

[74]Davis Blalock，Jose Javier Gonzalez Ortiz，Jonathan Frankle和John Guttag。神经网络修剪的状态是什么？arXiv预印本arXiv：2003.03033，2020。

[75]Peva Blanchard，El Mahdi El Mhamdi，Rachid Guerraoui，and Julien Stainer.机器学习与对手：拜占庭容忍梯度下降。在神经信息处理系统的进展，2017年。

[76]Peva Blanchard，Rachid Guerraoui，Julien Stainer，et al. Machine Learning with Adversaries：Byzantine Tolerant Gradient Descent.神经信息处理系统的进展，第118-128页，2017年。

[77]丹·波格丹诺夫，里沃·塔维斯特，和简·威廉森。部署安全的多方计算，数据分析（短文）。在金融密码学中，计算机科学讲义第7397卷，第57-64页。Springer，2012.

[78]Peter Bogetoft，Dan Lund Christensen，Ivan Damg ˚ ard，Martin Geisler，Thomas P. Jakobsen，Mikkel Krøigaard，作者：Janus Dam Nielsen，Jesper布乌斯Nielsen，Kurt Nielsen，Jakob Pagter，Michael I. Schwartzbach和Tomas Toft。安全多方计算上线。金融密码学，计算机科学讲义第5628卷，第325-343页。Springer，2009年。

[79]K. A.放大图片作者：弗拉基米尔伊万诺夫，本克罗伊特，安东尼奥Marcedone，H.布兰登·麦克马汉萨瓦尔·帕特尔丹尼尔拉梅奇，亚伦西格尔，卡恩赛斯。对用户持有的数据进行联邦学习的实用安全聚合。arXiv预印本arXiv：1611.04482，2016。

[80]K。a. Bonawitz，Vladimir Ivanov，Ben Kreuter，Antonio Marcedone，H Brendan McMahan，Sarvar Patel，Daniel Ramage，Aaron Segal，and Karn Seth.用于隐私保护机器学习的实用安全聚合。2017年ACM SIGSAC计算机和通信安全会议论文集，第1175- 1191页。ACM，2017年。

[81]K。a. Bonawitz，Hubert Eichner，Wolfgang Grieskamp，Dzmitry Huba，Alex Ingerman，Vladimir Ivanov，Chlo'e M Kiddon，Jakub Kone Razcn'y，Stefano Mazzocchi，Brendan McMahan，Timon货车Overveldt，大卫彼得鲁，丹尼尔拉梅奇和杰森罗斯兰德。Towards Federated Learning at Scale：System Design.在SysML 2019，2019。网址https://arxiv.org/abs/1902.01046。

[82]K. A. Bonawitz，Fariborz Salehi，Jakub Kone Braccn'y，Brendan McMahan，and Marco Gruteser.联邦学习自动调整通信效率的安全聚合。2019年第53届Asilomar信号，系统和计算机会议。IEEE，2019年。

[83]丹·博内、埃莱特·波义耳、亨利·科里根-吉布斯、尼夫·吉尔博亚和尤瓦尔·伊沙伊。零知识证明秘密共享数据通过完全线性的PCP。在《计算机科学讲义》第11694卷第3卷第67-97页中。施普林格，2019年。

[84] Florian Bourse, Michele Minelli, Matthias Minihold, and Pascal Paillier. Fast homomorphic evaluation of deep discretized neural networks. In *CRYPTO (3)*, volume 10993 of *Lecture Notes in Computer Science*, pages 483–512. Springer, 2018.

[85] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006.

[86] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL `http://github.com/google/jax`.

[87] Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical gapsvp. In *CRYPTO*, volume 7417 of *Lecture Notes in Computer Science*, pages 868–886. Springer, 2012.

[88] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *ITCS*, pages 309–325. ACM, 2012.

[89] Mark Braverman, Ankit Garg, Tengyu Ma, Huy L. Nguyen, and David P. Woodruff. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, page 1011–1020. ACM, 2016.

[90] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017.

[91] Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67, 2018.

[92] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Gregory Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy, SP 2018, Proceedings, 21-23 May 2018, San Francisco, California, USA*, 2018.

[93] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[94] Martin Burkhart, Mario Strasser, Dilip Many, and Xenofontas Dimitropoulos. SEPIA: Privacy-preserving aggregation of multi-domain network events and statistics. *Network*, 1(101101), 2010.

[95] Sebastian Caldas, Jakub Konečný, H Brendan McMahan, and Ameet Talwalkar. Expanding the reach of federated learning by reducing client resource requirements. *arXiv preprint arXiv:1812.07210*, 2018.

[96] Sebastian Caldas, Peter Wu, Tian Li, Jakub Konečný, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. LEAF: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

[97] Clément L Canonne, Gautam Kamath, Audra McMillan, Adam Smith, and Jonathan Ullman. The structure of optimal private tests for simple hypotheses. *AarXiv preprint arXiv:1811.11148*, 2019.

[98] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.

[99] Nicholas Carlini, Chang Liu, Jernej Kos, Úlfar Erlingsson, and Dawn Song. The secret sharer: Measuring unintended neural network memorization & extracting secrets. *arXiv preprint arXiv:1802.08232*, 2018.

[100] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. *arXiv preprint arXiv:2012.07805*, 2020.

[101] Iker Ceballos, Vivek Sharma, Eduardo Mugica, Abhishek Singh, Albert Roman, Praneeth Vepakomma, and Ramesh Raskar. SplitNN-driven vertical partitioning. *arXiv preprint arXiv:2008.04137*, 2018.

[84]Florian Bourse，Michele Minelli，Matthias Minihold，and Pascal Paillier. deep的快速同态求值离散神经网络在《计算机科学讲义》第10993卷，第483-512页。Springer，2018年。

[85]Stephen Boyd，Arpita Ghosh，Balaji Prabhakar，and Devavrat Shah.随机八卦算法。IEEE Transactions on Information Theory，52（6）：2508-2530，2006.

[86]James Bradbury，Roy Frostig，Peter Hawkins，Matthew James约翰逊，Chris Leary，Dougal Necula，Adam Paszke，Jake VanderPlas，Skye Wanderman-Milne，and Qiao Zhang. JAX：Python+NumPy程序的可组合转换，2018年。网址http://github.com/google/jax。

[87]兹维卡·布拉克斯基基于经典gapsvp的无模切换全同态加密。在美国专利商标局，计算机科学讲义第7417卷，第868-886页。Springer，2012.

[88]兹维卡·布拉克斯基，克雷格·金特里，还有维诺德·瓦昆塔纳坦.（分级）完全同态加密而无需自举。见ITCS，第309-325页。ACM，2012年。

[89]作者：Mark Braverman，Ankit Garg，Tengyu Ma，胡伊L. Nguyen和大卫P.伍德拉夫。通信下层通过分布式数据处理不等式的统计估计问题的界。在第48届年度ACM计算理论研讨会论文集，第1011-1020页。ACM，2016.

[90]维兰德·布伦德尔、乔纳斯·劳伯和马蒂亚斯·贝奇。基于决策的对抗性攻击：对黑盒机器学习模型的可靠攻击。arXiv预印本arXiv：1712.04248，2017。

[91]Theodora S Brisimi，Ruidi Chen，Theofanie Mela，Alex Olshevsky，Ioannis Ch Paschaleet，and Wei从联合电子健康记录中学习预测模型。国际医学信息学杂志，112：59-67，2018。

[92]Benedikt Bêunz，Jonathan Bootle，Dan Boneh，Andrew Poelstra，Pieter Wuille和Gregory Maxwell.子弹—proofs：用于机密交易等的简短证明。在2018年IEEE安全和隐私研讨会上，SP 2018，会议记录，2018年5月21日至23日，美国加州州弗朗西斯科，2018年。

[93]Joy Buolamwini和Timnit Gebru。性别差异：商业性别分类中的交叉准确性差异。在公平，问责制和透明度会议上，第77-91页，2018年。

[94]Martin Burkhart，Mario Strasser，Dilip Many，and Xenofontas Dimitropoulos. SEPIA：多域网络事件和统计数据的隐私保护聚合。Network，1（101101），2010.

[95]塞巴斯蒂安卡尔达斯，雅库布Kone Eschcn 'y，H布伦丹McMahan，和Ameet Talwalkar。通过减少客户资源需求来扩大联合学习的范围。arXiv预印本arXiv：1812.07210，2018。

[96]塞巴斯蒂安卡尔达斯，彼得吴，田丽，雅各布Kone Eschercn'y，H布伦丹McMahan，弗吉尼亚史密斯和Ameet Talwalkar。LEAF：联邦设置的基准。arXiv预印本arXiv：1812.01097，2018。

[97]Cl 'ement L Canonne，Gautam Kamath，Audra McMillan，Adam Smith和Jonathan Ullman。简单假设的最优私人检验的结构。arXiv预印本arXiv：1811.11148，2019。

[98]尼古拉斯·卡里尼和大卫瓦格纳。对神经网络鲁棒性的评估。2017年IEEE安全与隐私研讨会（SP），第39-57页。IEEE，2017年。

[99]尼古拉斯·卡利尼，刘畅，杰尼·科斯，乌尔法·厄林松和道恩·宋。秘密分享者：测量非故意的神经网络记忆和提取秘密。arXiv预印本arXiv：1802.08232，2018。

[100]Nicholas Carlini，Florian Tram`er，Eric Wallace，Matthew Jagielski，Ariel Herbert-Voss，凯瑟琳李，Adam Roberts，Tom Brown，Dawn Song，Ulfar Erlingsson，et al.从大型语言模型中提取训练数据。arXiv预印本arXiv：2012.07805，2020。

[101]Iker Ceballos、Vivek Sharma、Eduardo Mugica、Abhishek Singh、Albert Roman、Praneeth Vepakomma和Ramesh Raskar。SplitNN驱动的垂直分区。arXiv预印本arXiv：2008.04137，2018。

[102] Khe Chai Sim, Françoise Beaufays, Arnaud Benard, Dhruv Guliani, Andreas Kabel, Nikhil Khare, Tamar Lucassen, Petr Zadrazil, Harry Zhang, Leif Johnson, et al. Personalization of end-to-end speech recognition on mobile devices for named entities. *arXiv*, pages arXiv–1912, 2019.

[103] T-H Hubert Chan, Elaine Shi, and Dawn Song. Privacy-preserving stream aggregation with fault tolerance. In *International Conference on Financial Cryptography and Data Security*, pages 200–214. Springer, 2012.

[104] Ken Chang, Niranjan Balachandar, Carson Lam, Darvin Yi, James Brown, Andrew Beers, Bruce Rosen, Daniel L Rubin, and Jayashree Kalpathy-Cramer. Distributed deep learning networks among institutions for medical imaging. *Journal of the American Medical Informatics Association*, 25(8):945–954, 2018.

[105] Wei-Ting Chang and Ravi Tandon. On the upload versus download cost for secure and private matrix multiplication. *ArXiv*, abs/1906.10684, 2019.

[106] Zachary Charles and Jakub Konečnỳ. On the outsized importance of learning rates in local update methods. *arXiv preprint arXiv:2007.00878*, 2020.

[107] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2), 1981.

[108] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728*, 2018.

[109] Chien-Lun Chen, Leana Golubchik, and Marco Paolieri. Backdoor attacks on federated meta-learning. *arXiv preprint arXiv:2006.07026*, 2020.

[110] Lijie Chen, Badih Ghazi, Ravi Kumar, and Pasin Manurangsi. On distributed differential privacy and counting distinct elements. In *Innovations in Theoretical Computer Science (ITCS)*, 2021.

[111] Lingjiao Chen, Hongyi Wang, Zachary B. Charles, and Dimitris S. Papailiopoulos. DRACO: Byzantine-resilient distributed training via redundant gradients. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, 2018.

[112] Mingqing Chen, Rajiv Mathews, Tom Ouyang, and Françoise Beaufays. Federated learning of out-of-vocabulary words. *arXiv preprint 1903.10635*, 2019. URL `http://arxiv.org/abs/1903.10635`.

[113] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017.

[114] Wei-Ning Chen, Peter Kairouz, and Ayfer Ozgur. Breaking the communication-privacy-accuracy trilemma. *Advances in Neural Information Processing Systems*, 33, 2020.

[115] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[116] Yudong Chen, Lili Su, and Jiaming Xu. Distributed Statistical Machine Learning in Adversarial Settings: Byzantine Gradient Descent. *POMACS*, 1:44:1–44:25, 2017.

[117] Massimo Chenal and Qiang Tang. On key recovery attacks against existing somewhat homomorphic encryption schemes. In *LATINCRYPT*, volume 8895 of *Lecture Notes in Computer Science*, pages 239–258. Springer, 2014.

[118] Kewei Cheng, Tao Fan, Yilun Jin, Yang Liu, Tianjian Chen, and Qiang Yang. SecureBoost: A lossless federated learning framework. *CoRR*, abs/1901.08755, 2019. URL `http://arxiv.org/abs/1901.08755`.

[119] Raymond Cheng, Fan Zhang, Jernej Kos, Warren He, Nicholas Hynes, Noah Johnson, Ari Juels, Andrew Miller, and Dawn Song. Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 185–200. IEEE, 2019.

[102]Khe Chai Sim、Franc oise Beaufays、Arnaud Benard、Dhruv Guliani、Andreas Kabel、Nikhil Khare、Tamar Lucassen，Petr Zadrazil，Harry Zhang，Leif约翰逊等人，命名实体在移动的设备上的端到端语音识别的个性化。arXiv，第arXiv-1912页，2019年。

[103]T-H Hubert Chan，Elaine Shi和Dawn Song。具有容错性的隐私保护流聚合。金融密码学和数据安全国际会议，第200-214页。Springer，2012.

[104]Ken Chang，Niranjan Balachandar，卡森林，Darvin Yi，詹姆斯布朗，安德鲁比尔斯，布鲁斯罗森，

丹尼尔L鲁宾和Jayashree Kalpathy-Cramer。医疗成像机构之间的分布式深度学习网络。美国医学信息学协会杂志，25（8）：945-954，2018。

[105]张伟庭和拉维·坦登关于安全和私有矩阵乘法的上传与下载成本。ArXiv，abs/1906.10684，2019。

[106]扎卡里·查尔斯和雅各布·科内都很好。局部更新方法中学习率的重要性。

arXiv预印本arXiv：2007.00878，2020。

[107]大卫·乔姆。无法追踪的电子邮件，回邮地址和数字缩写。ACM通讯，24（2），1981年。

[108]Bryant Chen，Wilka Carvalho，Nathalie Baracaldo，Heiko Ludwig，Benjamin Edwards，Taesung Lee，Ian Mol-

loy和Biplav Srivastava。通过激活聚类检测对深度神经网络的后门攻击。arXiv预印本arXiv：1811.03728，2018。

[109]Chien-Lun Chen，Leana Golubchik，and Marco Paolieri.对联邦元学习的后门攻击。arXiv预印本arXiv：2006.07026，2020。

[110]Lijie Chen，Badih Ghazi，Ravi Kumar，and Pasin Manurangsi.分布式差异隐私和计数不同元素。在理论计算机科学创新（ITCS），2021。

[111]陈玲娇，王弘毅，Zachary B. Charles，and Dimitris S.帕帕利奥普洛斯拜占庭帝国

通过冗余梯度进行弹性分布式训练。第35届国际机器学习会议论文集，ICML，2018。

[112]Mingqing Chen，Rajiv马修斯，Tom Ouyang，和Franc Mingoise Beaufays.词汇表外词汇的联合学习。arXiv预印本1903.10635，2019。网址http://arxiv.org/abs/1903.10635。

[113]Pin-Yu Chen，Huan Zhang，Yash Sharma，Jinfeng Yi，and Cho-Jui Hsieh. ZOO：零阶优化

基于黑盒攻击的深度神经网络，无需训练替代模型。第10届ACM人工智能与安全研讨会论文集，第15-26页。ACM，2017年。

[114]Wei-Ning Chen，Peter Kairouz，and Ayfer Ozgur.打破通信-隐私-准确性三难困境。
神经信息处理系统的进展，33，2020。

[115]陈欣云，刘畅，李波，陆君怡，宋黎明。使用数据中毒对深度学习系统进行针对性后门攻击。arXiv预印本arXiv：1712.05526，2017。

[116]陈宇东，苏丽丽，徐嘉明。对抗环境中的分布式统计机器学习：拜占庭梯度下降。POMACS，1：44：1-44：25，2017。

[117]马西莫·切纳尔和唐强对已有的同态加密方案的密钥恢复攻击。在拉丁美洲，计算机科学讲义第8895卷，第239-258页。Springer，2014.

[118]程科伟，范涛，靳一伦，刘洋，陈天健，杨强。SecureBoost：无损联邦学习框架。CoRR，abs/1901.08755，2019。网址http://arxiv.org/abs/1901.08755。

[119]Raymond Cheng，Fan Zhang，Jernej科斯，Warren He，Nicholas Hynes，Noah约翰逊，Ari Juels，Andrew

米勒和道恩·宋Ekiden：一个保密、可信和高性能智能合约的平台。2019年IEEE欧洲安全与隐私研讨会（EuroS&P），第185-200页。IEEE，
2019.

[120] Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 375–403. Springer, 2019.

[121] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. Private information retrieval. *J. ACM*, 45 (6):965–981, November 1998. ISSN 0004-5411. doi: 10.1145/293347.293350. URL `http://doi.acm.org/10.1145/293347.293350`.

[122] Edward Chou, Florian Tramèr, and Giancarlo Pellegrino. SentiNet: Detecting physical attacks against deep learning systems. *arXiv preprint arXiv:1812.00292*, 2018.

[123] Sélim Chraibi, Ahmed Khaled, Dmitry Kovalev, Peter Richtárik, Adil Salim, and Martin Takáč. Distributed fixed point methods with compressed iterates. *arXiv preprint arXiv:1912.09925*, 2019.

[124] P. Christen. *Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection*. Springer Science & Business Media, 2012.

[125] NVIDIA Clara. The clara training framework authors, 2019. URL `https://developer.nvidia.com/clara`.

[126] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.

[127] Igor Colin, Aurélien Bellet, Joseph Salmon, and Stéphan Clémençon. Gossip dual averaging for decentralized optimization of pairwise functions. In *ICML*, 2016.

[128] Graham Cormode, Tejas Kulkarni, and Divesh Srivastava. Marginal release under local differential privacy. In *Proceedings of the 2018 International Conference on Management of Data*, pages 131–146. ACM, 2018.

[129] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Scale-invariant fully homomorphic encryption over the integers. In *Public Key Cryptography*, volume 8383 of *Lecture Notes in Computer Science*, pages 311–328. Springer, 2014.

[130] Henry Corrigan-Gibbs and Dan Boneh. Prio: Private, robust, and scalable computation of aggregate statistics. In *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*, pages 259–282, 2017.

[131] Henry Corrigan-Gibbs and Dmitry Kogan. Private information retrieval with sublinear online time. *IACR Cryptology ePrint Archive*, 2019:1075, 2019.

[132] Corinna Cortes and Mehryar Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, 519:103–126, 2014.

[133] Corinna Cortes, Mehryar Mohri, Ananda Theertha Suresh, and Ningshan Zhang. Multiple-source adaptation with domain classifiers. *arXiv preprint arXiv:2008.11036*, 2020.

[134] Victor Costan and Srinivas Devadas. Intel SGX explained. *IACR Cryptology ePrint Archive*, 2016(086):1–118, 2016.

[135] Victor Costan, Ilia Lebedev, and Srinivas Devadas. Sanctum: Minimal hardware extensions for strong software isolation. In *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pages 857–874, 2016.

[136] Craig Costello, Cédric Fournet, Jon Howell, Markulf Kohlweiss, Benjamin Kreuter, Michael Naehrig, Bryan Parno, and Samee Zahur. Geppetto: Versatile verifiable computation. In *IEEE Symposium on Security and Privacy*, pages 253–270. IEEE Computer Society, 2015.

[120]Albert Cheu，Adam Smith，Jonathan Ullman，大卫Zeber，和Maxim Zhilyaev.分布式差异隐私通过洗牌。在加密技术理论和应用的年度国际会议上，第375-403页。施普林格，2019年。

[121]Benny Chor、Eyal Kushilevitz、Oded Goldreich和Madhu Sudan。私人信息检索。J. ACM，45（6）：965-981，November 1998. ISSN 0004-5411。doi：10.1145/293347.293350。网址http://doi.acm。org/10.1145/293347.293350。

[122]Edward Chou，Florian Tram`er，and Giancarlo Pellegrino. SentiNet：检测针对深度学习系统的物理攻击。arXiv预印本arXiv：1812.00292，2018。

[123]埃利姆Chraibi，Ahmed Khaled，Dmitry Kovalev，Peter Richt 'arik，Adil Salim和Martin Tak' a Baghic。压缩迭代的分布式不动点方法。arXiv预印本arXiv：1912.09925，2019。

[124]P. Christen数据匹配：记录链接、实体解析和重复检测的概念和技术。Springer Science & Business Media，2012.

[125]NVIDIA Clara。Clara培训框架作者，2019年。网址www.example.com https://developer.nvidia.com/

[126]格雷戈里·科恩、赛义德·阿夫沙尔、乔纳森·塔普森和安德烈·货车·沙伊克。EMNIST：MNIST对手写信件的扩展。arXiv预印本arXiv：1702.05373，2017。

[127]伊戈尔·科林，奥埃连·贝莱特，约瑟夫·萨尔蒙，圣埃凡·克莱门克·塞隆。两两函数分散优化的Gossip对偶平均。In ICML，2016.

[128]Graham Cormode，Tejas Kulkarni，and Divesh Srivastava.局部差分隐私下的边际释放。在2018年数据管理国际会议的会议记录中，第131-146页。ACM，2018年。

[129]Jean-S 'ebastien科龙、Tancr' ede Lepoint和Mehdi Tibouchi。尺度不变全同态加密

而不是整数。公钥密码学，计算机科学讲义第8383卷，第311-328页。Springer，2014.

[130]亨利·科里根-吉布斯和丹·博纳。Prio：私有的、健壮的、可扩展的聚合统计计算。在第14届{USENIX}网络系统设计与实施研讨会（{NSDI} 17），第259- 282页，2017年。

[131]亨利科里根-吉布斯和德米特里科根。次线性在线时间下的私人信息检索。IACR Cryptology ePrint Archive，2019：1075，2019.

[132]科琳娜科尔特斯和梅赫里亚·莫赫里。回归领域自适应与样本偏差校正理论与算法。理论计算机科学，519：103-126，2014。

[133]Corinna科尔特斯，Mehryar Mohri，Ananda Theertha Suresh和Ningshan Zhang。多源适应领域分类器。arXiv预印本arXiv：2008.11036，2020。

[134]维克托·科斯坦和斯里尼瓦斯·提瓦达斯。英特尔SGX解释说。IACR Cryptology ePrint Archive，2016（086）：1-118，

2016.

[135]维克托科斯坦，伊利亚列别捷夫，和斯里尼瓦斯提瓦达斯。Sanctum：最小的硬件扩展，实现强大的软件隔离。第25届{USENIX}安全研讨会（{USENIX} Security 16），第857-874页，2016年。

[136]克雷格·科斯特洛，C 'edric Fournet，Jon Howell，Markulf Kohlweiss，Benjamin Kreuter，Michael Parno和Samee Zahur Geppetto：Versatile Verifiable Computation。IEEE Symposium on Security and Privacy，第253-270页。IEEE计算机协会，2015年。

[137] Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24, pages 1647–1655. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper/2011/file/b55ec28c52d5f6205684a473a2193564-Paper.pdf.

[138] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. BinaryConnect: Training deep neural networks with binary weights during propagations. In *Advances in neural information processing systems*, pages 3123–3131, 2015.

[139] Pierre Courtiol, Charles Maussion, Matahi Moarii, Elodie Pronier, Samuel Pilcer, Meriem Sefta, Pierre Manceron, Sylvain Toldo, Mikhail Zaslavskiy, Nolwenn Le Stang, et al. Deep learning-based classification of mesothelioma improves prediction of patient outcome. *Nature medicine*, pages 1–7, 2019.

[140] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[141] Gabriela F Cretu, Angelos Stavrou, Michael E Locasto, Salvatore J Stolfo, and Angelos D Keromytis. Casting out demons: Sanitizing training data for anomaly sensors. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 81–95. IEEE, 2008.

[142] Rachel Cummings, Sara Krehbiel, Kevin Lai, and Uthaipon Tantitongpipat. Differential privacy for growing databases. In *Advances in Neural Information Processing Systems 31*, NeurIPS '18, pages 8864–8873, 2018.

[143] Rachel Cummings, Sara Krehbiel, Yajun Mei, Rui Tuo, and Wanrong Zhang. Differentially private change-point detection. In *Advances in Neural Information Processing Systems 31*, NeurIPS '18, pages 10825–10834, 2018.

[144] Rachel Cummings, Inbal Dekel, Ori Heffetz, and Katrina Ligett. Bringing differential privacy into the experimental economics lab: Theory and an application to a public-good game. Working paper, 2019.

[145] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Proceedings of Fairness in User Modeling, Adaptation and Personalization*, FairUMAP, 2019.

[146] Edwige Cyffers and Aurélien Bellet. Privacy amplification by decentralization. *arXiv preprint arXiv:2012.05326*, 2020.

[147] Damgård. On $\sigma$ protocols. http://www.cs.au.dk/~ivan/Sigma.pdf, 2010.

[148] Deepesh Data, Linqi Song, and Suhas Diggavi. Data encoding for byzantine-resilient distributed optimization. *IEEE Transactions on Information Theory*, 2020.

[149] Walter de Brouwer. The federated future is ready for shipping. https://doc.ai/blog/federated-future-ready-shipping/, March 2019.

[150] Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 1223–1231, 2012.

[151] Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *J. Mach. Learn. Res.*, 13(1), January 2012.

[152] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1596–1606, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/diakonikolas19a.html.

[137]Andrew Cotter，Ohad Shamir，Nati Srebro，and Karthik Sridharan.更好的小批量算法通过ac-
加速梯度法Shawe-Taylor，R. Zemel，P. Bartlett，F.佩雷拉和K. Q. Weinberger，编辑，神经信息处理系统进展，第24卷，第1647-1655页。柯兰联营公司2011.网址https://proceedings.neurips.cc/paper/2011/file/ b55 ec28 c52 d5 f6205684 a473 a2193564-Paper.pdf。

[138]马蒂厄·库巴里奥，约瑟芬·本吉奥，让-皮埃尔大卫。BinaryConnect：训练深度神经网络
在传播期间具有二进制权重。神经信息处理系统的进展，第3123- 3131页，2015年。

[139]Pierre Courtiol，Charles Masters，Matahi Moarii，Elodie Pronier，Samuel Pilcer，Meriem Sefta，Pierre Manceron，Sylvain Toldo，Mikhail Zaslavskiy，Nolwenn Le Stang等人，基于深度学习的间皮瘤分类提高了对患者结局的预测。自然医学，第1-7页，2019年。

[140]托马斯M封面和乔伊A托马斯。信息论的基本原理。John Wiley & Sons，2012.

[141]Gabriela F Bagliu，Angelos Stavrou，Michael E Locasto，Salvatore J Stolfo，and Angelos D Keromytis.铸造清除恶魔：清除异常传感器的训练数据。在2008年IEEE安全和隐私研讨会（sp 2008），第81-95页。IEEE，2008年。

[142]Rachel Cummings，Sara Kampelbiel，Kevin Lai，and Uthaipon Tantitongpipat.针对不断增长的数据库的差异隐私。神经信息处理系统进展31，NeurIPS '18，第8864-8873页，2018年。

[143]Rachel Cummings，Sara Kagshabiel，Yajun Mei，Rui Tuo，and Wanrong Zhang.差异化的私人改变-
点检测在神经信息处理系统的进展31，NeurIPS '18，第10825-10834页，2018年。

[144]瑞秋·卡明斯，因巴尔·德克尔，奥里·赫费茨，卡特里娜·利格特。将差异隐私带入实验经济学实验室：理论和对公共利益游戏的应用。工作文件，2019年。

[145]Rachel Cummings，Varun Gupta，Dhamma Kimpara，and Jamie Morgenstern.论隐私与公平的相容性。在用户建模，适应和个性化的公平性诉讼中，FairUMAP，2019。

[146]Edwige Cyffers和Aur 'elien Bellet。通过去中心化扩大隐私。arXiv预印本arXiv：2012.05326，2020。

[147]该死的巴拉德关于σ协议。http：//www.cs.au.dk/Ekivan/Sigma.pdf，2010年。

[148]Deepesh Data、Linqi Song和Suhas Diggavi。拜占庭弹性分布式优化的数据编码。
IEEE Transactions on Information Theory，2020。

[149]沃尔特·德·布劳威尔联邦的未来已经准备好了。https://doc.ai/blog/ federated-future-ready-shipping/，2019年3月。

[150]作者：Jeffrey Dean，Greg S.放大图片作者：Corrado，Rajat Monga，Kai Chen，Matthieu迪文，Quoc V. Le，Mark Z.毛泽东
Marc'Aurelio Ranzato、Andrew Senior、Paul Tucker、Ke Yang和Andrew Y. Ng.大规模分布式深度网络。神经信息处理系统国际会议论文集，第1223-1231页，2012年。

[151]Ofer Dekel，Ran Gilad-Bachrach，Ohad Shamir和Lin Xiao。使用小批量的最优分布式在线预测。J·马赫。学习.结果：13（1），2012年1月。

[152]Ilias Diakonikolas，Gautam Kamath，Daniel Kane，Jerry Li，Jacob Steinhardt，和Alistair Stewart.一直都
随机优化的鲁棒元算法在Kamalika Chaudhuri和Ruslan Salakhutdinov编辑的第36届国际机器学习会议论文集，机器学习研究论文集第97卷，第1596-1606页，长滩，加州，美国，2019年6月9日至15日。PMLR。网址http：//proceedings.mlr.press/v97/diakonikolas19a.html。

[153] Mario Diaz, Peter Kairouz, Jiachun Liao, and Lalitha Sankar. Theoretical guarantees for model auditing with finite adversaries. *arXiv preprint arXiv:1911.03405*, 2019.

[154] Differential Privacy Team. Learning with privacy at scale. *Apple Machine Learning Journal*, 1(8), 2017. URL https://machinelearning.apple.com/2017/12/06/learning-with-privacy-at-scale.html.

[155] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *Advances in Neural Information Processing Systems 30*, December 2017. URL https://www.microsoft.com/en-us/research/publication/collecting-telemetry-data-privately/.

[156] Zeyu Ding, Yuxin Wang, Guanhong Wang, Danfeng Zhang, and Daniel Kifer. Detecting violations of differential privacy. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, pages 475–489, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5693-0. doi: 10.1145/3243734.3243818. URL http://doi.acm.org/10.1145/3243734.3243818.

[157] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, Naval Research Lab Washington DC, 2004.

[158] Canh T. Dinh, Nguyen H. Tran, and Tuan Dung Nguyen. Personalized Federated Learning with Moreau Envelopes. In *NeurIPS*, 2020.

[159] Rafael G. L. D'Oliveira and S. E. Rouayheb. Lifting private information retrieval from two to any number of messages. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 1744–1748, June 2018. doi: 10.1109/ISIT.2018.8437805.

[160] John R. Douceur. The sybil attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, pages 251–260, London, UK, UK, 2002. Springer-Verlag. ISBN 3-540-44179-4. URL http://dl.acm.org/citation.cfm?id=646334.687813.

[161] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[162] John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 429–438. IEEE, 2013.

[163] Sanghamitra Dutta, Gauri Joshi, Soumyadip Ghosh, Parijat Dube, and Priya Nagpurkar. Slow and Stale Gradients Can Win the Race: Error-Runtime Trade-offs in Distributed SGD. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, April 2018. URL https://arxiv.org/abs/1803.01113.

[164] Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.

[165] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[166] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer, 2006.

[167] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *IACR Theory of Cryptography Conference (TCC), New York, New York*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer-Verlag, 2006. doi: 10.1007/11681878_14.

[168] Cynthia Dwork, Guy N. Rothblum, and Salil Vadhan. Boosting and differential privacy. In *Proceedings of the IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, pages 51–60, 2010.

[153]Mario迪亚兹，Peter Kairouz，Jiachun Liao和Lalitha Sankar。有限对手模型审计的理论保证。arXiv预印本arXiv：1911.03405，2019。

[154]差分隐私团队大规模的隐私学习。苹果机器学习

Journal，1（8），2017.网址https://machinelearning.apple.com/2017/12/06/ learning-with-privacy-at-scale.html。

[155]Bolin Ding，Janardhan Kulkarni和Sergey Yekhanin。私下收集遥测数据。In Advances
在神经信息处理系统30，2017年12月。网址https://www.microsoft.com/ en-us/research/publication/collecting-telemetry-data-privileges/.

[156]Zeyu Ding，Yuxin Wang，Guanhong Wang，Danfeng Zhang，and丹尼尔基弗.发现违反不同-
尊重隐私在2018年ACM SIGSAC计算机和通信安全会议论文集，CCS '18，第475-489页，纽约，纽约州，美国，2018年。ACM。ISBN 978-1-4503-5693-0。doi：10.1145/3243734.3243818。网址http://doi.acm.org/10.1145/3243734.3243818。

[157]罗杰·丁格尔丁尼克·马修森和保罗·赛弗森Tor：第二代洋葱路由器。技术报告，海军研究实验室华盛顿，2004年。

[158]景湖Dinh，Nguyen H.陈和阮勇使用Moreau Envelopes的个性化联邦学习在NeurIPS，2020年。

[159]拉斐尔·G L. D 'Oliveira和S. E. Rouayheb。提升私人信息检索从两个到任何数量的消息。在2018年IEEE信息理论国际研讨会（ISIT），第1744-1748页，2018年6月。

doi：10.1109/ISIT.2018.8437805。

[160]John R.杜瑟女巫袭击。在第一届国际点对点研讨会的修订论文中
系统， IPTPS ' 01， pages 251 - 260， 伦敦， 英国， 2002 年 。Springer 出版 社ISBN 3 - 540 - 44179 - 4 . URLhttp://dl.acm.org/citation.cfm? id = 646334.687813 .

[161]John Duchi，Elad Hazan，和Yoram Singer.在线学习和随机优化的自适应次梯度方法。Journal of Machine Learning Research，12（7），2011.

[162]约翰C杜奇，迈克尔I乔丹，和马丁J温赖特。本地隐私和统计最小最大利率。在

计算机科学基础（FOCS），2013年IEEE第54届年度研讨会，第429-438页。IEEE，2013年。

[163]Sanghamitra Dutta，Gauri Joshi，Soumyadip Ghosh，Parijat Dube，and Priya Nagpurkar.缓慢而陈旧的格拉迪-
企业可以赢得比赛：分布式SGD中的错误-错误权衡。人工智能与统计国际会议（AISTATS），2018年4月。网址https://arxiv.org/abs/1803.01113。

[164]辛西娅·德沃克差异隐私：结果调查。在计算模型的理论和应用国际会议上，第1-19页。施普林格，2008年。

[165]辛西娅·德沃克和亚伦·罗斯差分隐私的算法基础。Foundations and Trends in Theoretical Computer Science，9（3-4）：211-407，2014。

[166]Cynthia Dwork，Krishnaram Kenthapadi，Frank McSherry，Ilya Mironov，and Moni Naor.我们的数据，我们自己：

通过分布式噪声生成的隐私。在加密技术理论和应用的年度国际会议上，第486-503页。施普林格，2006年。

[167]Cynthia Dwork，Frank McSherry，Kobbi Nissim，and Adam D.史密斯私下校准噪音灵敏度
数据分析在IACR Theory of Cryptography Conference（TCC），纽约，纽约，计算机科学讲义第3876卷，第265-284页。Springer-Verlag，2006. doi：10.1007/11681878 14.

[168]Cynthia Dwork，Guy N.罗斯布鲁姆和萨里尔·瓦丹增强和差异隐私。在IEEE第51届计算机科学基础年度研讨会论文集，FOCS '10，第51-60页，2010年。

[169] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226. ACM, 2012.

[170] Laurel Eckhouse, Kristian Lum, Cynthia Conti-Cook, and Julie Ciccolini. Layers of bias: A unified approach for understanding problems with risk assessment. *Criminal Justice and Behavior*, 46(2):185–209, 2019.

[171] Hubert Eichner, Tomer Koren, H. Brendan McMahan, Nathan Srebro, and Kunal Talwar. Semi-cyclic stochastic gradient descent. In *Accepted to ICML 2019.*, 2019. URL https://arxiv.org/abs/1904.10120.

[172] Karim Eldefrawy, Gene Tsudik, Aurélien Francillon, and Daniele Perito. SMART: secure and minimal architecture for (establishing dynamic) root of trust. In *NDSS*. The Internet Society, 2012.

[173] Anis Elgabli, Jihong Park, Amrit S Bedi, Mehdi Bennis, and Vaneet Aggarwal. GADMM: Fast and communication efficient framework for distributed machine learning. *arXiv preprint arXiv:1909.00047*, 2019.

[174] Anis Elgabli, Jihong Park, Chaouki Ben Issaid, and Mehdi Bennis. Harnessing wireless channels for scalable and privacy-preserving federated learning, 2020.

[175] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Efficient multi-objective neural architecture search via Lamarckian evolution. *arXiv preprint arXiv:1804.09081*, 2018.

[176] Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling CNNs with simple transformations. *arXiv preprint arXiv:1712.02779*, 2017.

[177] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *ACM CCS*, 2014. ISBN 978-1-4503-2957-6. doi: 10.1145/2660267.2660348. URL http://doi.acm.org/10.1145/2660267.2660348.

[178] Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *SODA*, pages 2468–2479, 2019.

[179] EU CORDIS. Machine learning ledger orchestration for drug discovery, 2019. URL https://cordis.europa.eu/project/rcn/223634/factsheet/en?WT.mc_id=RSS-Feed&WT.rss_f=project&WT.rss_a=223634&WT.rss_ev=a. Retrieved Aug 2019.

[180] Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. *arXiv preprint arXiv:1807.01774*, 2018.

[181] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

[182] Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. *IACR Cryptology ePrint Archive*, 2012:144, 2012.

[183] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. Local model poisoning attacks to Byzantine-robust federated learning. *arXiv preprint arXiv:1911.11815*, 2019.

[184] FeatureCloud. FeatureCloud: Our vision, 2019. URL https://featurecloud.eu/about/our-vision/. Retrieved Aug 2019.

[185] Vitaly Feldman, Ilya Mironov, Kunal Talwar, and Abhradeep Thakurta. Privacy amplification by iteration. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 521–532. IEEE, 2018.

[186] Clément Feutry, Pablo Piantanida, Yoshua Bengio, and Pierre Duhamel. Learning anonymized representations with adversarial neural networks. *CoRR*, abs/1802.09386, 2018. URL http://arxiv.org/abs/1802.09386.

[169]Cynthia Dwork，莫里茨哈特，Toniann Pitassi，奥默莱因戈尔德和理查德泽梅尔。公平是通过了解-奈斯第三届理论计算机科学创新会议论文集，第214-226页。ACM，2012年。

[170]月桂埃克豪斯，克里斯蒂安林，辛西娅康蒂库克，和朱莉Ciccolini。偏见的层次：理解风险评估问题的统一方法。刑事司法和行为，46（2）：185-209，2019。

[171]Hubert Eichner，Tomer Koren，H.布兰登·麦克马汉内森·斯雷布罗和库纳尔·塔尔瓦半循环随机梯度下降。在接受ICML 2019.，2019.网址https://arxiv.org/abs/1904.10120。

[172]Karim Eldefrawy，Gene Tsudik，Aur 'elien Francillon和Daniele佩里托。SMART：用于（建立动态）信任根的安全和最小架构。在国家安全局。互联网协会，2012年。

[173]Anis Elgabli，Jihong Park，Amrit S Bedi，Mehdi Bennis和Vaneet Aggarwal。GADMM：分布式机器学习的快速和通信高效框架。arXiv预印本arXiv：1909.00047，2019。

[174]Anis Elgabli，Jihong Park，Chaouki Ben Issaid和Mehdi Bennis。利用无线信道进行可扩展和隐私保护的联合学习，2020年。

[175]托马斯埃尔斯肯，扬亨德里克梅森，和弗兰克哈特。通过拉马克进化的高效多目标神经结构搜索。arXiv预印本arXiv：1804.09081，2018。

[176]Logan Engstrom、布兰登特兰、迪米特里斯齐普拉斯、路德维希施密特和亚历山大马德里。一个旋转和一个平移就足够了：用简单的变换愚弄CNN。arXiv预印本arXiv：1712.02779，2017。

[177]Ulfar Erlingsson，Vasyl Pihur和Aleksandra Korolova。RAPPOR：随机聚合隐私保护顺序响应。在ACM CCS，2014年。ISBN 978-1-4503-2957-6。doi：10.1145/2660267.2660348。

网址http://doi.acm.org/10.1145/2660267.2660348。

[178]Ulfar Erlingsson，Vitaly Feldman，Ilya Mironov，Ananth Raghunathan，Kunal Talwar和Abhradeep Thakurta。洗牌放大：通过匿名从本地到中央差分隐私。在SODA中，第2468- 2479页，2019年。

[179]欧盟CORDIS。用于药物发现的机器学习分类帐编排，2019年。网址https：//cordis.europa.eu/project/rcn/223634/factsheet/en? WT.mc_id=RSS-Feed&WT.rss_f=project&WT.rss_a=223634&WT.rss_ev=a. 2019年8月恢复。

[180]斯特凡·福克纳，亚伦·克莱因，弗兰克·哈特。BOHB：大规模鲁棒高效超参数优化。arXiv预印本arXiv：1807.01774，2018。

[181]Alireza Fallah，Aryan Mokhtari，and Asuman Ozdaglar.个性化联邦学习：一种元学习方法。arXiv预印本arXiv：2002.07948，2020。

[182]Junfeng Fan和Frederik Vercauteren。有点实用的全同态加密。IACR Cryptology ePrint Archive，2012：144，2012.

[183]Minghong Fang，Xiaoyu Cao，Jinyuan Jia，and Neil Zhenqiang Gong.局部模型中毒攻击拜占庭-鲁棒联邦学习。arXiv预印本arXiv：1911.11815，2019。

[184]云计算。云计算：我们的愿景，2019年。网址https://featurecloud.eu/about/ our-vision/. 2019年8月恢复。

[185]Vitaly Feldman，Ilya Mironov，Kunal Talwar，and Abhradeep Thakurta.通过迭代来放大隐私。2018年IEEE第59届计算机科学基础年度研讨会（FOCS），第521-532页。IEEE，2018年。

[186]Clement Feutry，巴勃罗Piantanida，Yoelman Bengio和Pierre Duhamel。使用对抗性神经网络学习匿名表示。CoRR，abs/1802.09386，2018。网址http://arxiv.org/abs/1802。

09386.

[187] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[188] Aurélien Francillon, Quan Nguyen, Kasper Bonne Rasmussen, and Gene Tsudik. A minimalist approach to remote attestation. In *DATE*, pages 1–6. European Design and Automation Association, 2014.

[189] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333. ACM, 2015.

[190] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. Mitigating sybils in federated learning poisoning. *arXiv preprint arXiv:1808.04866*, 2018.

[191] Jun Furukawa, Yehuda Lindell, Ariel Nof, and Or Weinstein. High-throughput secure three-party computation for malicious adversaries and an honest majority. In *EUROCRYPT (2)*, volume 10211 of *Lecture Notes in Computer Science*, pages 225–255, 2017.

[192] Adam Gaier and David Ha. Weight agnostic neural networks. *arXiv preprint arXiv:1906.04358*, 2019.

[193] Venkata Gandikota, Raj Kumar Maity, and Arya Mazumdar. vqSGD: Vector quantized stochastic gradient descent. *arXiv preprint arXiv:1911.07971*, 2019.

[194] Adrià Gascón, Phillipp Schoppmann, Borja Balle, Mariana Raykova, Jack Doerner, Samee Zahur, and David Evans. Privacy-preserving distributed linear regression on high-dimensional data. *PoPETs*, 2017(4):345–364, 2017.

[195] Rosario Gennaro, Craig Gentry, and Bryan Parno. Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In *CRYPTO*, volume 6223 of *Lecture Notes in Computer Science*, pages 465–482. Springer, 2010.

[196] Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct NIZKs without PCPs. In *EUROCRYPT*, volume 7881 of *Lecture Notes in Computer Science*, pages 626–645. Springer, 2013.

[197] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009.

[198] Craig Gentry and Shai Halevi. Compressible FHE with applications to PIR. In *TCC (2)*, volume 11892 of *Lecture Notes in Computer Science*, pages 438–464. Springer, 2019.

[199] Robin C. Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *CoRR*, abs/1712.07557, 2017. URL `http://arxiv.org/abs/1712.07557`.

[200] Badih Ghazi, Noah Golowich, Ravi Kumar, Rasmus Pagh, and Ameya Velingker. On the power of multiple anonymous messages. *arXiv:1908.11358*, 2019.

[201] Badih Ghazi, Rasmus Pagh, and Ameya Velingker. Scalable and differentially private distributed aggregation in the shuffled model. *arXiv preprint arXiv:1906.08320*, 2019.

[202] Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. Pure differentially private summation from anonymous messages. In *ITC*, pages 15:1–15:23, 2020.

[203] Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead. In *ICML*, 2020.

[204] Badih Ghazi, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. Private aggregation from fewer anonymous messages. In *EUROCRYPT*, pages 798–827, 2020.

[187]Chelsea Finn Pieter Abbeel和Sergey Levine模型不可知元学习用于深度网络的快速适应。2017年第34届机器学习国际会议论文集。

[188]Aur 'elien Francillon，Quan Nguyen，Kasper Bonne Rasmussen和Gene Tsudik。A minimally approach to remote attestation.最小方法是远程attestation.在日期，页面1—6。欧洲设计与自动化协会，2014。

[189]Matt Fredrikson，Somesh Jha，和Thomas Ristenpart. Model Inversion Attacks That Exploit Confidence Infor（英语：Model Inversion attacks That Exploit Confidence Infor）
的基本对策。第22届ACM SIGSAC计算机和通信安全会议论文集，第1322-1333页。ACM，2015.

[190]Clement Fung，Chris JM Yoon和Ivan Beschastnikh。减轻联邦学习中毒中的sybils。arXiv预印本arXiv：1808.04866，2018。

[191]Jun Furukawa，Yehuda Lindell，Ariel Nof和Or Weinstein。高吞吐量安全三方计算
恶意的对手和诚实的大多数。在EUROCITY PT（2），计算机科学讲义第10211卷，第225-255页，2017年。

[192]亚当·盖尔和大卫哈。权重不可知的神经网络。arXiv预印本arXiv：1906.04358，2019。

[193]Venkata Gandikota，Raj Kumar Maity，and Arya Mazumdar. vqSGD：矢量量化随机梯度下降。arXiv预印本arXiv：1911.07971，2019。

[194]Adri`a Gasc 'on、Phillipp Schoppmann、博尔哈巴莱、Mariana Raykova、Jack Doerner、Samee Zahur和大卫
埃文斯.高维数据上的隐私保护分布式线性回归。PoPETs，2017（4）：345-364，2017。

[195]罗萨里奥詹纳罗，克雷格金特里，和布赖恩帕诺。非交互式可验证计算：将计算外包给不受信任的工作人员。在《计算机科学讲义》第6223卷，第465-482页。
Springer，2010.

[196]罗萨里奥詹纳罗，克雷格金特里，布赖恩帕诺，和玛丽安娜Raykova。二次跨度程序和简洁的NIZK没有PCP。在EURONETPT中，计算机科学讲义的第7881卷，第626-645页。
Springer，2013.

[197]克雷格金特里。使用理想格的全同态加密。在Proceedings of the 41 th Annual ACM Symposium on Theory of Computing，第169-178页，2009年。

[198]克雷格詹特里和沙伊哈勒维。应用于PIR的可压缩FHE。在TCC（2），计算机科学讲义第11892卷，第438-464页中。施普林格，2019年。

[199]罗宾·C盖耶，塔西洛·克莱因，莫因·纳比。差异化私有联邦学习：客户端层面的视角。CoRR，abs/1712.07557，2017。网址http://arxiv.org/abs/1712.07557。

[200]Badih Ghazi，Noah Golowich，Ravi Kumar，Rasmus Pagh，and Ameya Velingker.多条匿名信息的力量。arXiv：1908.11358，2019。

[201]巴迪·加齐，拉斯莫斯·帕格，阿梅亚·韦林克。洗牌模型中的可扩展和差异化私有分布式聚合。arXiv预印本arXiv：1906.08320，2019。

[202]巴迪赫·加齐、诺亚·戈洛维奇、拉维·库马尔、帕辛·马努朗西、拉斯穆斯·帕格和阿梅亚·韦林克。来自匿名信息的纯粹不同的私人交流。在国贸中心，2020年第15：1-15：23页。

[203]巴迪赫·加齐、拉维·库马尔、帕辛·马努朗西和拉斯穆斯·帕格。匿名消息的私有计数：具有消失通信开销的近最佳精度。在ICML，2020。

[204]巴迪赫·加齐、帕辛·马努朗西、拉斯穆斯·帕格和阿梅亚·韦林克。来自fewer匿名消息的私人聚合。《欧洲地穴》，第798-827页，2020。

[205] Arpita Ghosh, Tim Roughgarden, and Mukund Sundararajan. Universally utility-maximizing privacy mechanisms. In *Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing*, STOC '09, pages 351–360, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-506-2. doi: 10.1145/1536414.1536464. URL http://doi.acm.org/10.1145/1536414.1536464.

[206] Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin E. Lauter, Michael Naehrig, and John Wernsing. CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pages 201–210, 2016. URL http://proceedings.mlr.press/v48/gilad-bachrach16.html.

[207] Antonious M Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled model of federated learning: Privacy, communication and accuracy trade-offs. *arXiv preprint arXiv:2008.07180*, 2020.

[208] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 218–229, New York, NY, USA, 1987. ACM. ISBN 0-89791-221-7. doi: 10.1145/28395.28420. URL http://doi.acm.org/10.1145/28395.28420.

[209] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989.

[210] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: interactive proofs for muggles. In *STOC*, pages 113–122. ACM, 2008.

[211] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

[212] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *ICLR*, 2015.

[213] Slawomir Goryczka and Li Xiong. A comprehensive comparison of multiparty secure additions with differential privacy. *IEEE Trans. Dependable Sec. Comput.*, 14(5):463–477, 2017. doi: 10.1109/TDSC.2015.2484326. URL https://doi.org/10.1109/TDSC.2015.2484326.

[214] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

[215] Otkrist Gupta and Ramesh Raskar. Distributed learning of deep neural network over multiple agents. *Journal of Network and Computer Applications*, 116:1–8, 2018.

[216] Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Viveck R Cadambe. Local SGD with periodic averaging: Tighter analysis and adaptive synchronization. *arXiv preprint arXiv:1910.13598*, 2019.

[217] Andreas Haeberlen, Benjamin C Pierce, and Arjun Narayan. Differential privacy under fire. In *USENIX Security Symposium*, 2011.

[218] Shai Halevi, Yehuda Lindell, and Benny Pinkas. Secure computation on the web: Computing without simultaneous interaction. In *Annual Cryptology Conference*, pages 132–150. Springer, 2011.

[219] Jenny Hamer, Mehryar Mohri, and Ananda Theertha Suresh. Fedboost: A communication-efficient algorithm for federated learning. In *International Conference on Machine Learning*, pages 3973–3983. PMLR, 2020.

[220] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[205]Arpita Ghosh，Tim Roughgarden，and Mukund Sundararajan.普遍效用最大化隐私机制nisms。在Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing，STOC '09，第351-360页，纽约，NY，美国，2009年。ACM。ISBN 978-1-60558-506-2。doi：10.1145/1536414.1536464。网址http://doi.acm.org/10.1145/1536414.1536464。

[206]Ran Gilad—Bachrach，Nathan Dowlin，Kim Laine，Kristin E.作者：Michael Naehrig，John Wernsing. CryptoNets：将神经网络应用于加密数据，具有高吞吐量和准确性。第33届国际机器学习会议论文集，ICML 2016，纽约市，美国纽约州，2016年6月19日至24日，第201-210页，2016年。网址http://proceedings.mlr.press/v48/gilad-bachrach16。html.

[207]Antonious M Girgis，Deepesh Data，Suhas Diggavi，Peter Kairouz和Ananda Theertha Suresh。噓联邦学习的逃离模型：隐私，通信和准确性权衡。arXiv预印本arXiv：2008.07180，2020。

[208]O. Goldreich，S. Micali和A.威格德森怎么玩心理游戏。在《九评》中，第10届ACM计算理论年会，STOC '87，第218-229页，纽约，NY，美国，1987。ACM。ISBN 0-89791-221-7。doi：10.1145/28395.28420。网址http://doi.acm.org/10.1145/ 28395.28420。

[209]沙菲·戈德瓦瑟，西尔维奥·米卡利，查尔斯·拉科夫。交互式证明系统的知识复杂性。SIAM J. COMPUT.，18（1）：186-208，1989.

[210]Shafi Goldwasser，Yael Tauman Kalai和Guy N.罗斯布鲁姆委派计算：麻瓜的交互式证明。在STOC中，第113-122页。ACM，2008年。

[211]Ian J. Goodfellow，Jonathy Shlens，and Christian Szegedy.解释和利用对抗性的例子。在第三届国际学习表征会议，ICLR 2015，美国加利福尼亚州圣地亚哥，2015年5月7日至9日，会议跟踪程序，2015年。网址http://arxiv.org/abs/1412.6572。

[212]Ian J Goodfellow，Jonathy Shlens，and Christian Szegedy.解释和利用对抗性的例子。ICLR，2015年。

[213]Slawomir Goryczka和李雄。多方安全添加与差异隐私的全面比较。IEEE跨部门安全计算：14（5）：463-477，2017. doi：10.1109/TDSC.2015.2484326。网址https://doi.org/10.1109/TDSC.2015.2484326。

[214]顾天宇，布伦丹·多兰-加维特和西达斯·加格。BadNets：识别机器学习模型供应链中的漏洞。arXiv预印本arXiv：1708.06733，2017。

[215]奥特克里斯特·古普塔和拉梅什·拉斯卡深度神经网络在多个代理上的分布式学习。网络和计算机应用杂志，116：1-8，2018。

[216]Farzin Haddadpour，Mohammad Mahdi Kamani，Mehrdad Mahdavi，and Viveck R Cadambe.具有周期平均的本地SGD：更紧密的分析和自适应同步。arXiv预印本arXiv：1910.13598，2019。

[217]Andreas Haeberlen，Benjamin C Pierce，and Arjun Narayan.在战火中的差别隐私。在USENIX安全研讨会，2011年。

[218]Shai Halevi，Yehuda Lindell，和Benny Pinkas. Web上的安全计算：无需同时交互的计算。在年度密码学会议上，第132-150页。Springer，2011.

[219]Jenny Hamer，Mehryar Mohri，and Ananda Theertha Suresh. Fedboost：一种用于联邦学习的通信高效算法。国际机器学习会议，第3973-3983页。PMLR，2020年。

[220]Song Han，Huizi Mao，and William J Dally.深度压缩：通过剪枝、训练量化和霍夫曼编码来压缩深度神经网络。arXiv预印本arXiv：1510.00149，2015年。

[221] Yanjun Han, Ayfer Özgür, and Tsachy Weissman. Geometric lower bounds for distributed parameter estimation under communication constraints. In *Proceedings of Machine Learning Research*, pages 1–26, 75, 2018.

[222] Andrew Hard, Kanishka Rao, Rajiv Mathews, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint 1811.03604*, 2018.

[223] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, 2016.

[224] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017.

[225] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1934–1943, 2018.

[226] Chaoyang He, Conghui Tan, Hanlin Tang, Shuang Qiu, and Ji Liu. Central server free federated learning over single-sided trust social networks. *arXiv preprint arXiv:1910.04956*, 2019.

[227] Chaoyang He, Murali Annavaram, and Salman Avestimehr. Group knowledge transfer: Federated learning of large cnns at the edge. In *Advances in Neural Information Processing Systems 34*, 2020.

[228] Chaoyang He, Murali Annavaram, and Salman Avestimehr. Fednas: Federated deep learning via neural architecture search. 2020.

[229] Chaoyang He, Songze Li, Jinhyun So, Xiao Zeng, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, Xinghua Zhu, Jianzong Wang, Li Shen, Peilin Zhao, Yan Kang, Yang Liu, Ramesh Raskar, Qiang Yang, Murali Annavaram, and Salman Avestimehr. Fedml: A research library and benchmark for federated machine learning, 2020.

[230] Chaoyang He, Haishan Ye, Li Shen, and Tong Zhang. Milenas: Efficient neural architecture search via mixed-level reformulation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[231] Lie He, An Bian, and Martin Jaggi. COLA: Decentralized linear learning. In *NeurIPS 2018 - Advances in Neural Information Processing Systems 31*, 2018.

[232] Úrsula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1944–1953, 2018.

[233] HElib. HElib. `https://github.com/homenc/HElib`, October 2019.

[234] Judy Hoffman, Mehryar Mohri, and Ningshan Zhang. Algorithms and theory for multiple-source adaptation. In *Advances in Neural Information Processing Systems*, pages 8246–8256, 2018.

[235] Samuel Horvath, Chen-Yu Ho, Ludovit Horvath, Atal Narayan Sahu, Marco Canini, and Peter Richtarik. Natural compression for distributed deep learning. *arXiv preprint arXiv:1905.10988*, 2019.

[236] Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip B. Gibbons. The non-IID data quagmire of decentralized machine learning, 2019. URL `https://arxiv.org/abs/1910.00189`.

[237] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

[238] Yaochen Hu, Peng Liu, Linglong Kong, and Di Niu. Learning privately over distributed features: An admm sharing approach, 2019.

[221]Yanjun Han，Ayfer？Ozgur，and Tsachy Weissman.通信约束下分布参数估计的几何下界。在机器学习研究论文集，第1-26，75页，2018年。

[222]Andrew Hard，Kanishka Rao，Rajiv Mathews，Franc oise Beaufays，Sean Ogenstein，Hubert Eichner，基登和丹尼尔·拉米奇。用于移动的键盘预测的联邦学习。arXiv预印本1811.03604，2018。

[223]莫里茨哈特，埃里克价格，和内森Srebro。监督学习中的机会平等。在神经信息处理系统的进展，2016年。

[224]斯蒂芬哈代，威尔科Henecka，哈米什Ivey-Law，理查德诺克，乔治帕特里尼，纪尧姆史密斯和布赖恩索恩通过实体解析和加法同态加密对垂直分区数据进行私有联邦学习。arXiv预印本arXiv：1711.10677，2017。

[225]Tatsunori Hashimoto，Megha Srivastava，Hongseok Namkoong，and珀西梁.重复损失最小化中的无人口统计的公平。在国际机器学习会议上，第1934-1943页，2018年。

[226]何朝阳，谭丛辉，唐翰林，秋霜，刘继。中央服务器通过单边信任社交网络免费联合学习。arXiv预印本arXiv：1910.04956，2019。

[227]Chaoyang He，Murali Annavaram，and Salman Avestimehr.小组知识转移：边缘大型cnn的联邦学习。在神经信息处理系统的进展34，2020。

[228]Chaoyang He，Murali Annavaram，and Salman Avestimehr. Fednas：通过神经架构搜索进行联合深度学习。2020.

[229]何朝阳、李颂泽、So镇贤、小曾、张米、王弘毅、王小阳、Praneeth Vepakomma、Abhishek Singh、杭秋、兴华朱、建宗王、李申、赵培林、颜康、杨柳、拉梅什·拉斯卡尔、强阳、穆拉利·安纳瓦拉姆和萨勒曼·阿韦斯蒂迈尔。Fedml：联邦机器学习的研究库和基准，2020年。

[230]朝阳何、海山叶、李申、同张。千禧年：高效神经架构搜索通过混合-水平重构在IEEE计算机视觉和模式识别会议（CVPR）上，2020年。

[231]何烈，安边，马丁·贾吉。COLA：Decentralized Linear Learning分散式线性学习在NeurIPS 2018 -神经信息处理系统的进展31，2018。

[232]厄休拉·赫伯特·约翰逊，迈克尔·金，奥默·莱因戈尔德和盖伊·罗斯布卢姆。多重校准：校准（计算可识别的）质量。在国际机器学习会议上，第1944- 1953页，2018年。

[233]HElib。HElib。https：//github.com/homenc/HElib，2019年10月。

[234]Judy霍夫曼，Mehryar Mohri和Ningshan Zhang。多源自适应算法与理论。神经信息处理系统的进展，第8246-8256页，2018年。

[235]Samuel Horvath，Chen-Yu Ho，Ludovit Horvath，Atal Narayan Sahu，Marco Canini，and Peter Richtarik.用于分布式深度学习的自然压缩。arXiv预印本arXiv：1905.10988，2019。

[236]Kevin Hsieh，Amar Mr. Ishayee，Onur Mutlu和菲利普B。吉本斯分散式机器学习的非IID数据泥潭，2019年。网址https://arxiv.org/abs/1910.00189。

[237]徐子铭，杭琦，马修·布朗。测量非相同数据分布对联邦视觉分类的影响。arXiv预印本arXiv：1909.06335，2019。

[238]胡耀晨，刘鹏，孔玲珑，狄牛。在分布式功能上私下学习：一种admm共享方法，2019年。

[239] Zhenqi Huang, Sayan Mitra, and Nitin Vaidya. Differentially Private Distributed Optimization. In *ICDCN*, 2015.

[240] Zhouyuan Huo, Bin Gu, and Heng Huang. Training neural networks using features replay. In *Advances in Neural Information Processing Systems*, pages 6659–6668, 2018.

[241] R Intel. Architecture instruction set extensions programming reference. *Intel Corporation, Feb*, 2012.

[242] Mihaela Ion, Ben Kreuter, Erhan Nergiz, Sarvar Patel, Shobhit Saxena, Karn Seth, David Shanahan, and Moti Yung. Private intersection-sum protocol with applications to attributing aggregate ad conversions. *IACR Cryptology ePrint Archive*, 2017:738, 2017.

[243] Mihaela Ion, Ben Kreuter, Ahmet Erhan Nergiz, Sarvar Patel, Mariana Raykova, Shobhit Saxena, Karn Seth, David Shanahan, and Moti Yung. On deploying secure computing commercially: Private intersection-sum protocols and their business applications. *IACR Cryptology ePrint Archive*, 2019:723, 2019.

[244] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 145–161. Springer, 2003.

[245] Max Jaderberg, Wojciech Marian Czarnecki, Simon Osindero, Oriol Vinyals, Alex Graves, David Silver, and Koray Kavukcuoglu. Decoupled neural interfaces using synthetic gradients. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1627–1635. JMLR. org, 2017.

[246] Matthew Jagielski, Michael J. Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. Differentially private fair learning. *CoRR*, abs/1812.02696, 2018. URL `http://arxiv.org/abs/1812.02696`.

[247] Matthew Jagielski, Jonathan Ullman, and Alina Oprea. Auditing differentially private machine learning: How private is private sgd? *Advances in Neural Information Processing Systems*, 33, 2020.

[248] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data. *CoRR*, abs/1811.11479, 2018. URL `http://arxiv.org/abs/1811.11479`.

[249] Zhuqing Jia and Syed Ali Jafar. On the capacity of secure distributed matrix multiplication. *ArXiv*, abs/1908.06957, 2019.

[250] Yihan Jiang, Jakub Konečný, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

[251] S. Kadhe, B. Garcia, A. Heidarzadeh, S. E. Rouayheb, and A. Sprintson. Private information retrieval with side information: The single server case. In *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1099–1106, Oct 2017. doi: 10.1109/ALLERTON.2017.8262860.

[252] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2879–2887. Curran Associates, Inc., 2014.

[253] Peter Kairouz, K. A. Bonawitz, and Daniel Ramage. Discrete distribution estimation under local privacy. In *International Conference on Machine Learning*, pages 2436–2444, 2016.

[254] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. *IEEE Transactions on Information Theory*, 63(6):4037–4049, 2017.

[255] Peter Kairouz, Jiachun Liao, Chong Huang, and Lalitha Sankar. Censored and fair universal representations using generative adversarial models. *arXiv preprint arXiv:1910.00411*, 2020.

[256] Peter Kairouz, Ziyu Liu, and Thomas Steinke. The distributed discrete gaussian mechanism for federated learning with secure aggregation, 2021.

[239]Zhenqi Huang，Sayan Mitra，and Nitin Vaidya.差异私有分布式优化。在ICDCN，2015.

[240]霍周源，顾斌，黄恒。使用特征回放训练神经网络。神经信息处理系统的进展，第6659-6668页，2018年。

[241]R英特尔。指令集架构扩展编程参考。英特尔公司，2012年2月。

[242]Mihaela Ion、Ben Kreuter、Erhan Nergiz、Sarvar Patel、Shobhit Saxena、Karn Seth、大卫沙纳汉和Moti容私人交叉和协议与应用归因于聚合广告转换。IACR Cryptology ePrint Archive，2017：738，2017.

[243]Mihaela Ion，Ben Kreuter，Ahmet Erhan Nergiz，Sarvar Patel，Mariana Raykova，Shobhit Saxena，Karn Seth，大卫·沙纳汉和莫提·杨。在商业上部署安全计算：私有交和协议及其商业应用。IACR Cryptology ePrint Archive，2019：723，2019.

[244]Yuval Ishai，Joe Kilian，Kobbi Nissim，and Erez Petrank.有效地扩展不经意传输。在美国专利局，计算机科学讲义第2729卷，第145-161页。Springer，2003年。

[245]Max Jaderberg、Wojciech Marian Czarnecki、Simon Osindero、Oriol Vinyals、Alex Graves、大卫银和科雷·卡武库奥卢使用合成梯度的解耦神经接口。第34届机器学习国际会议论文集-第70卷，第1627-1635页。JMLR。org，2017.

[246]Matthew Jagielski，Michael J. Kearns，Jieming Mao，Alina Oprea，Aaron Roth，Saeed Sharifi-Malvajerdi，and Jonathan Ullman.不同的私人公平学习。CoRR，abs/1812.02696，2018。http：//arxiv.org/abs/1812.02696。

[247]Matthew Jagielski Jonathan Ullman和Alina Oprea审计差异化私有机器学习：私有sgd有多私有？神经信息处理系统的进展，33，2020。

[248]Eunjeong Jeong、Heungeun Oh、Hyesung Kim、Jihong Park、Mehdi Bennis和Seong-Lyun Kim。通信高效的设备上机器学习：非IID私有数据下的联邦蒸馏和增强。CoRR，abs/1811.11479，2018。网址http://arxiv.org/abs/1811.11479。

[249]贾竹青和赛义德·阿里·贾法尔。关于安全分布式矩阵乘法的容量。ArXiv，abs/1908.06957，2019。

[250]Yihan Jiang，Jakub Kone Zakcn'y，基思拉什，和Sreeram Kannan.通过模型不可知Meta学习改进联邦学习个性化。arXiv预印本arXiv：1909.12488，2019。

[251]S. Kadhe，B. Garcia，A. Heidarzadeh，S. E. Rouayheb和A.斯普林森带边的私有信息检索信息：单一服务器案例。2017年第55届Allerton通信，控制和计算年会（Allerton），第1099-1106页，2017年10月。doi：10.1109/ALLERTON.2017.8262860。

[252]Peter Kairouz，Sewoong Oh，和Pramod Viswanath.局部差异隐私的极端机制。在Z.加赫拉马尼，M.威林角，澳-地科尔特斯，N. D.劳伦斯和K。Q. Weinberger，编辑，神经信息处理系统进展27，第2879-2887页。柯兰联营公司2014.

[253]Peter Kairouz，K. A.博纳维茨和丹尼尔·拉梅奇。局部隐私下的离散分布估计。在国际机器学习会议上，第2436-2444页，2016年。

[254]Peter Kairouz，Sewoong Oh，和Pramod Viswanath.差分隐私的合成定理。IEEE Transactions on Information Theory，63（6）：4037-4049，2017。

[255]Peter Kairouz，Jiachun Liao，Chong Huang，and Lalitha Sankar.使用生成对抗模型的审查和公平通用表示。arXiv预印本arXiv：1910.00411，2020。

[256]Peter Kairouz，Ziyu Liu，和托马斯斯坦克。分布式离散高斯联邦学习机制与安全聚合，2021年。

[257] Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. Practical and private (deep) learning without sampling or shuffling, 2021.

[258] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 643–650. IEEE, 2011.

[259] Daniel Kang, Yi Sun, Dan Hendrycks, Tom Brown, and Jacob Steinhardt. Testing robustness against unforeseen adversaries. *arXiv preprint arXiv:1908.08016*, 2019.

[260] Jiawen Kang, Zehui Xiong, Dusit Niyato, Shengli Xie, and Junshan Zhang. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal*, 2019.

[261] Jiawen Kang, Zehui Xiong, Dusit Niyato, Han Yu, Ying-Chang Liang, and Dong In Kim. Incentive design for efficient federated learning in mobile networks: A contract theory approach. In *IEEE VTS Asia Pacific Wireless Communications Symposium, APWCS 2019, Singapore, August 28-30, 2019*, pages 1–5, 2019.

[262] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

[263] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes SignSGD and other gradient compression schemes. In *ICML*, 2019.

[264] Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020.

[265] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

[266] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011. URL `https://doi.org/10.1137/090756090`.

[267] Michael J. Kearns, Aaron Roth, Zhiwei Steven Wu, and Grigory Yaroslavtsev. Privacy for the protected (only). *CoRR*, abs/1506.00242, 2015. URL `http://arxiv.org/abs/1506.00242`.

[268] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. First analysis of local GD on heterogeneous data, 2019. URL `https://arxiv.org/abs/1909.04715`.

[269] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Better communication complexity for local SGD, 2019. URL `https://arxiv.org/abs/1909.04746`.

[270] Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Adaptive gradient-based meta-learning methods. In *Advances in Neural Information Processing Systems*, 2019.

[271] Daniel Kifer and Ashwin Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. *ACM Transactions on Database Systems*, 39(1):3:1–3:36, 2014.

[272] Yejin Kim, Jimeng Sun, Hwanjo Yu, and Xiaoqian Jiang. Federated tensor factorization for computational phenotyping. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 887–895, 2017. doi: 10.1145/3097983.3098118. URL `https://doi.org/10.1145/3097983.3098118`.

[273] Ross D. King, Cao Feng, and Alistair Sutherland. StatLog: comparison of classification algorithms on large real-world problems. *Applied Artificial Intelligence an International Journal*, 9(3):289–333, 1995.

[257]Peter Kairouz，Brendan McMahan，Shuang Song，Om Thakkar，Abhradeep Thakurta，and Zheng Xu.没有采样或洗牌的实用和私人（深度）学习，2021年。

[258]上岛俊博赤穗正太郎佐久间纯通过正则化方法的公平意识学习。2011年IEEE第11届数据挖掘研讨会国际会议，第643-650页。IEEE，2011年。

[259]丹尼尔康，孙毅，丹亨德里克斯，汤姆布朗和雅各布斯坦哈特。测试对不可预见的对手的鲁棒性。arXiv预印本arXiv：1908.08016，2019。

[260]Jiawen Kang，Zehui Xiong，杜西特Niyato，Shengli Xie，and Junshan Zhang.可靠的激励机制

联邦学习：一种结合声誉和契约理论的联合优化方法。IEEE Internet of Things Journal，2019。

[261]Jiawen Kang，Zehui Xiong，杜西特Niyato，Han Yu，Ying-Chang Liang，and Dong In Kim.激励设计移动的网络中的高效联邦学习：契约理论方法。在IEEE VTS亚太无线通信研讨会，APWCS 2019，新加坡，2019年8月28-30日，第1-5页，2019年。

[262]哈米德·卡里米朱莉·努蒂尼和马克·施密特。梯度法和近似梯度法的线性收敛性在波利亚克-乔亚西维奇条件下在联合欧洲会议机器学习和知识发现在数据库，第795-811页。Springer，2016.

[263]赛·普拉尼斯·卡里米雷迪、昆汀·雷布约克、塞巴斯蒂安、斯蒂奇和马丁·贾吉。错误反馈修复SignSGD和其他梯度压缩方案。2019年，在ICML中。

[264]赛·普拉内斯·卡里米雷迪、马丁·贾吉、萨蒂安·卡莱、梅赫利亚尔·莫赫里、萨尚克·J·雷迪、塞巴斯蒂安·U·施蒂希、

和阿南达·瑟莎·苏雷什Mime：模仿联邦学习中的集中式随机算法。arXiv预印本arXiv：2008.03606，2020。

[265]赛·普拉内特·卡里米雷迪、萨蒂安·卡莱、梅赫利亚尔·莫赫里、萨尚克·雷迪、塞巴斯蒂安·施蒂希和阿南达·泰尔塔苏雷什Scaffold：用于联邦学习的随机控制平均。国际机器学习会议，第5132-5143页。PMLR，2020年。

[266]Shiva Prasad Kasiviswanathan，Homin K. Lee，Kobbi Nissim，Sofya Raskhodkova，and Adam D.史密斯我们私下能学到什么？SIAM J. COMPUT.，40（3）：793-826，2011。网址https://doi.org/10。1137/090756090.

[267]Michael J. Kearns，Aaron Roth，Zhiwei Steven Wu，and Grigory Yaroslavtsev.隐私保护（仅限）。CoRR，abs/1506.00242，2015。网址http://arxiv.org/abs/1506.00242。

[268]艾哈迈德·哈立德、康斯坦丁·米先科和彼得·里希特·阿里克。2019年首次分析本地GD异构数据。网址https://arxiv.org/abs/1909.04715。

[269]艾哈迈德·哈立德、康斯坦丁·米先科和彼得·里希特·阿里克。更好的本地SGD通信复杂性，2019年。网址https://arxiv.org/abs/1909.04746。

[270]Mikhail Khodak，Maria-Florina Balcan和Ameet Talwalkar。自适应梯度元学习方法。

在神经信息处理系统的进展，2019年。

[271]丹尼尔·基弗和阿什温·马查纳瓦吉哈拉。河豚：数学隐私定义的框架。ACM Transactions on Database Systems，39（1）：3：1-3：36，2014.

[272]Yejin Kim，Jimeng Sun，Hwanjo Yu，and Xiaoqian Jiang.计算的联合张量分解表型分析第23届ACM SIGKDD知识发现和数据挖掘国际会议论文集，哈利法克斯，NS，加拿大，2017年8月13日至17日，第887-895页，2017年。doi：10.1145/3097983。3098118. URLhttps://doi.org/10.1145/3097983.3098118.

[273]Ross D.金，曹锋，阿利斯泰尔·萨瑟兰。StatLog：大型现实问题分类算法的比较。Applied Artificial Intelligence an International Journal，9（3）：289-333，1995.

[274] Patrick Koeberl, Steffen Schulz, Ahmad-Reza Sadeghi, and Vijay Varadharajan. TrustLite: a security architecture for tiny embedded devices. In *EuroSys*, pages 10:1–10:14. ACM, 2014.

[275] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1885–1894. JMLR. org, 2017.

[276] Pang Wei Koh, Jacob Steinhardt, and Percy Liang. Stronger data poisoning attacks break data sanitization defenses. *arXiv preprint arXiv:1811.00741*, 2018.

[277] Ron Kohavi and George H John. Automatic parameter selection by minimizing estimated error. In *Machine Learning Proceedings 1995*, pages 304–312. Elsevier, 1995.

[278] Anastasia Koloskova, Sebastian U Stich, and Martin Jaggi. Decentralized Stochastic Optimization and Gossip Algorithms with Compressed Communication. In *ICML*, 2019.

[279] Anastasia Koloskova, Tao Lin, Sebastian U Stich, and Martin Jaggi. Decentralized deep learning with arbitrary communication compression. *International Conference on Learning Representations (ICLR)*, 2020.

[280] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian U. Stich. A Unified Theory of Decentralized SGD with Changing Topology and Local Updates. In *ICML*, 2020.

[281] Jakub Konečný and Peter Richtárik. Randomized distributed mean estimation: Accuracy vs communication. *Frontiers in Applied Mathematics and Statistics*, 4:62, 2018.

[282] Jakub Konečný, H Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[283] Satya Kuppam, Ryan McKenna, David Pujol, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Fair decision making using privacy-protected data. *CoRR*, abs/1905.12744, 2019. URL `http://arxiv.org/abs/1905.12744`.

[284] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[285] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, New York, NY, USA, 1997. ISBN 0-521-56067-5.

[286] Eyal Kushilevitz and Rafail Ostrovsky. Replication is not needed: Single database, computationally-private information retrieval. In *In Proc. of the 38th Annu. IEEE Symp. on Foundations of Computer Science*, pages 364–373, 1997.

[287] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pages 4066–4076, 2017.

[288] Albert Kwon, David Lazar, Srinivas Devadas, and Bryan Ford. Riffle. *Proceedings on Privacy Enhancing Technologies*, 2016(2):115–134, 2016.

[289] Yassine Laguel, Krishna Pillutla, Jérôme Malick, and Zaid Harchaoui. Device Heterogeneity in Federated Learning: A Superquantile Approach. *arXiv preprint arXiv:2002.11223*, 2020.

[290] Brenden M. Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B. Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the Conference of the Cognitive Science Society (CogSci)*, 2017.

[291] Anusha Lalitha, Osman Cihan Kilinc, Tara Javidi, and Farinaz Koushanfar. Peer-to-peer Federated Learning on Graphs. Technical report, arXiv:1901.11173, 2019.

[292] Anusha Lalitha, Xinghan Wang, Osman Kilinc, Yongxi Lu, Tara Javidi, and Farinaz Koushanfar. Decentralized Bayesian learning over graphs. *arXiv preprint: 1905.10466*, 2019.

[274]帕特里克Koeberl，Steffen Schulz，Ahmad-Reza Sadeghi和Vijay Varadharajan。TrustLite：一种用于微型嵌入式设备的安全架构。见EuroSys，第10：1-10：14页。ACM，2014年。

[275]庞伟高和珀西梁。通过影响函数理解黑箱预测。第34届国际机器学习会议论文集，第70卷，第1885-1894页。JMLR。org，2017.

[276]Pang Wei Koh，Jacob Steinhardt和珀西梁。更强的数据中毒攻击会破坏数据清理防御。arXiv预印本arXiv：1811.00741，2018。

[277]罗恩科哈维和乔治H约翰。通过最小化估计误差自动选择参数。在Machine Learning Proceedings 1995中，第304-312页。爱思唯尔，1995年。

[278]阿纳斯塔西娅·科洛斯科娃，塞巴斯蒂安·U·斯蒂奇，马丁·贾吉。具有压缩通信的分散随机优化和流言算法。2019年，在ICML中。

[279]Anastasia Koloskova，Tao Lin，塞巴斯蒂安U斯蒂奇和Martin Jaggi。分散式深度学习，任意通信压缩。2020年国际学习表征会议（ICLR）。

[280]Anastasia Koloskova、Nicolas Loizou、Sadra Boreiri、Martin Jaggi和塞巴斯蒂安U.斯蒂奇。一个统一的分散SGD理论与变化的拓扑和本地更新。2020年，《国际反洗钱法》。

[281]雅库布·科内·库尼和彼得·里希特·阿里克。随机分布均值估计：准确性与沟通。

应用数学与统计学前沿，4：62，2018。

[282]放大图片创作者：Jakub Kone Baghn'y，H Brendan McMahan，Felix X. Yu，Peter Richt 'arik，Ananda Theertha Suresh，and Dave Bacon.提高沟通效率的策略。arXiv预印本arXiv：1610.05492，2016年。

[283]Satya Kuppam，Ryan McKenna，大卫Pujol，Michael Hay，Ashwin Machanavajjhala和Gerome Miklau。使用受隐私保护的数据进行公平决策。CoRR，abs/1905.12744，2019。http：//arxiv.org/abs/1905.12744。

[284]Alexey Kurakin，Ian Goodfellow，and Samy Bengio.大规模对抗机器学习。arXiv预印本arXiv：1611.01236，2016。

[285]Eyal Kushilevitz和Noam Nisan。沟通的复杂性。剑桥大学出版社，纽约，纽约州，美国，1997年。ISBN 0-521-56067-5。

[286]Eyal Kushilevitz和Rafail Ostrovsky不需要复制：单个数据库，计算专用

信息检索在第38届年会上，IEEE Symp.计算机科学基础，第364-373页，1997年。

[287]马特·J·库斯纳，约书亚洛夫图斯，克里斯·拉塞尔和里卡多·席尔瓦。反事实的公平。神经信息处理系统的进展，第4066-4076页，2017年。

[288]Albert Kwon，大卫Lazar，Srinivas Devadas，和Bryan福特.来福枪Proceedings on Privacy Enhancing Technologies，2016（2）：115-134，2016.

[289]Yassine Laguel，Krishna Pillutla，J'er Escherome Malick和Zaid Harchaoui。联邦学习中的设备异质性：超分位数方法。arXiv预印本arXiv：2002.11223，2020。

[290]布兰登·M Lake，Ruslan Salakhutdinov，Jason Gross和约书亚B。特南鲍姆一次学习简单的视觉概念。在认知科学学会（CogSci）会议论文集，2017年。

[291]Anusha Lalitha，Osman Cihan Kilinc，塔拉Javidi和Farinaz Koushanfar。图上的对等联合学习。技术报告，arXiv：1901.11173，2019。

[292]Anusha Lalitha、Xinghan Wang、Osman Kilinc、Yongxi Lu、塔拉Javidi和Farinaz Koushanfar。图上的分散贝叶斯学习。arXiv预印本：1905.10466，2019。

[293] Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401, 1982.

[294] Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133 (1):365–397, Jun 2012. ISSN 1436-4646. doi: 10.1007/s10107-010-0434-y. URL https://doi.org/10.1007/s10107-010-0434-y.

[295] Andrei Lapets, Nikolaj Volgushev, Azer Bestavros, Frederick Jansen, and Mayank Varia. Secure MPC for analytics as a web application. In *SecDev*, pages 73–74. IEEE Computer Society, 2016.

[296] Mathias Lécuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pages 656–672, 2019. doi: 10.1109/SP.2019.00044. URL https://doi.org/10.1109/SP.2019.00044.

[297] Tancrède Lepoint, Sarvar Patel, Mariana Raykova, Karn Seth, and Ni Trieu. Private join and compute from PIR with default. *IACR Cryptol. ePrint Arch.*, 2020:1011, 2020.

[298] David Leroy, Alice Coucke, Thibaut Lavril, Thibault Gisselbrecht, and Joseph Dureau. Federated learning for keyword spotting. *arXiv preprint arXiv:1810.05512*, 2018.

[299] Jeffrey Li, Mikhail Khodak, Sebastian Caldas, and Ameet Talwalkar. Differentially private meta-learning. *arXiv preprint arXiv:1909.05830*, 2019.

[300] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks, 2018. URL https://arxiv.org/abs/1812.06127.

[301] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions, 2019.

[302] Tian Li, Maziar Sanjabi, and Virginia Smith. Fair resource allocation in federated learning. *arXiv preprint arXiv:1905.10497*, 2019.

[303] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of FedAvg on non-IID data. *arXiv preprint arXiv:1907.02189*, 2019.

[304] Xiang Li, Wenhao Yang, Shusen Wang, and Zhihua Zhang. Communication efficient decentralized training with multiple local updates. *arXiv preprint arXiv:1910.09126*, 2019.

[305] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can Decentralized Algorithms Outperform Centralized Algorithms? A Case Study for Decentralized Parallel Stochastic Gradient Descent. In *NIPS*, 2017.

[306] Xiangru Lian, Wei Zhang, Ce Zhang, and Ji Liu. Asynchronous Decentralized Parallel Stochastic Gradient Descent. In *ICML*, 2018.

[307] libsnark. libsnark: a c++ library for zkSNARK proofs. https://github.com/scipr-lab/libsnark, December 2019.

[308] David Lie and Petros Maniatis. Glimmers: Resolving the privacy/trust quagmire. In *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*, pages 94–99. ACM, 2017.

[309] Darryl Lin, Sachin Talathi, and Sreekanth Annapureddy. Fixed point quantization of deep convolutional networks. In *International Conference on Machine Learning*, pages 2849–2858, 2016.

[310] Tao Lin, Sebastian U Stich, and Martin Jaggi. Don't use large mini-batches, use local SGD. *International Conference on Learning Representations (ICLR)*, 2020.

[293]莱斯利·兰波特罗伯特·肖斯塔克和马歇尔·皮斯拜占庭将军的问题。ACM Transactions on Programming Languages and Systems（TOPLAS），4（3）：382-401，1982.

[294]兰广辉。随机组合优化的一种优化方法。数学规划133

(1)：365-397，Jun 2012. ISSN 1436-4646。doi：10.1007/s10107-010-0434-y。网址https://doi.org/10.1007/s10107-010-0434-y。

[295]Andrei Lapets，Nikolaj Volgushev，Azer Bestavros，Frederick Jansen，and Mayank Varia.将MPC作为Web应用程序进行安全分析。在SecDev，第73-74页。IEEE计算机协会，2016年。

[296]Mathias L 'ecuyer，Vaggelis Atlidakis，Roxana Geambasu，丹尼尔许和Suman Jana。经认证的稳健性

to adversarial对抗examples例子with different差异privacy隐私.在2019年IEEE安全和隐私研讨会上，SP 2019，美国加利福尼亚州旧金山弗朗西斯科，2019年5月19日至23日，第656-672页，2019年。doi：10.1109/SP.2019.00044。网址https：//doi.org/10.1109/SP.2019.00044。

[297]Tancr 'ede Lepoint、Sarvar Patel、Mariana Raykova、Karn Seth和Ni Trieu。默认情况下，私有连接和从PIR计算。IACR Cryptol。ePrint Arch.，2020年：1011、2020年。

[298]大卫勒罗伊，爱丽丝Coucke，Thibaut Lavril，Thibault Gisselbrecht和约瑟夫Dureau。关键字识别的联邦学习。arXiv预印本arXiv：1810.05512，2018。

[299]Jeffrey Li，Mikhail Khodak，塞巴斯蒂安卡尔达斯和Ameet Talwalkar。差异化私人元学习。arXiv预印本arXiv：1909.05830，2019。

[300]Tian Li，Anit Kumar Sahu，Manzil Zaheer，Maziar Sanjabi，Ameet Talwalkar，and Virginia Smith.异构网络中的联邦优化，2018年。网址https://arxiv.org/abs/1812.06127。

[301]Tian Li，Anit Kumar Sahu，Ameet Talwalkar，and Virginia Smith.联邦学习：挑战，方法和未来方向，2019年。

[302]Tian Li，Maziar Sanjabi，and Virginia Smith.联邦学习中的公平资源分配。arXiv预印本arXiv：1905.10497，2019。

[303]Xiang Li，Kaixuan Huang，Wenhao Yang，Shusen Wang，and Zhihua Zhang.关于FedAvg在非IID数据上的收敛。arXiv预印本arXiv：1907.02189，2019。

[304]Xiang Li，Wenhao Yang，Shusen Wang，and Zhihua Zhang.具有多个本地更新的通信高效分散式培训。arXiv预印本arXiv：1910.09126，2019。

[305]连相如、张策、张桓、谢朝瑞、张伟、刘继。Can分散式算法

优于集中式算法？分布式并行随机梯度下降的案例研究。在NIPS，2017年。

[306]连相如，张伟，张策，季柳。异步分散并行随机梯度下降。2018年在ICML上发表。

[307]libsnark。libsnark：一个用于zkSNARK证明的c++库。https：//github.com/scipr-lab/libsnark，2019年12月。

[308]大卫·李和彼得罗斯·马尼亚提斯。Glimmers：解决隐私/信任困境。在第16届操作系统热门话题研讨会论文集，第94-99页。ACM，2017年。

[309]Darryl Lin，Sachin Talathi和Sreekanth Annapureddy。深度卷积网络的定点量化。在国际机器学习会议上，第2849-2858页，2016年。

[310]林涛，塞巴斯蒂安U斯蒂奇和马丁·贾吉。不要使用大的小批量，使用本地SGD。2020年国际学习表征会议（ICLR）。

[311] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and William J Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*, 2017.

[312] R. J. A. Little. Post-stratification: A modeler's perspective. *Journal of the American Statistical Association*, 88 (423):1001–1012, 1993. ISSN 01621459.

[313] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

[314] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, pages 273–294. Springer, 2018.

[315] Xiyang Liu and Sewoong Oh. Minimax rates of estimating approximate differential privacy. *arXiv preprint arXiv:1905.10335*, 2019.

[316] Yang Liu, Yan Kang, Xinwei Zhang, Liping Li, Yong Cheng, Tianjian Chen, Mingyi Hong, and Qiang Yang. A communication efficient vertical federated learning framework. *CoRR*, abs/1912.11187, 2019. URL `http://arxiv.org/abs/1912.11187`.

[317] Yang Liu, Yan Kang, Chaoping Xing, Tianjian Chen, and Qiang Yang. A secure federated transfer learning framework. *IEEE Intelligent Systems*, 35(4):70–82, 2020. doi: 10.1109/MIS.2020.2988525.

[318] Yang Liu, Zhihao Yi, and Tianjian Chen. Backdoor attacks and defenses in feature-partitioned collaborative learning, 2020.

[319] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*, 2018. URL `http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2018/02/ndss2018_03A-5_Liu_paper.pdf`.

[320] Yuhan Liu, Ananda Theertha Suresh, Felix Xinnan X Yu, Sanjiv Kumar, and Michael Riley. Learning discrete distributions: user vs item-level privacy. *Advances in Neural Information Processing Systems*, 33, 2020.

[321] Heiko Ludwig, Nathalie Baracaldo, Gegi Thomas, Yi Zhou, Ali Anwar, Shashank Rajamoni, Yuya Ong, Jayaram Radhakrishnan, Ashish Verma, Mathieu Sinn, et al. IBM federated learning: An enterprise framework white paper V0.1. *arXiv preprint arXiv:2007.10987*, 2020.

[322] Jiahuan Luo, Xueyang Wu, Yun Luo, Anbu Huang, Yunfeng Huang, Yang Liu, and Qiang Yang. Real-world image datasets for federated learning. *arXiv preprint arXiv:1910.11089*, 2019.

[323] Renqian Luo, Fei Tian, Tao Qin, Enhong Chen, and Tie-Yan Liu. Neural architecture optimization. In *Advances in neural information processing systems*, pages 7816–7827, 2018.

[324] Lingjuan Lyu, Jiangshan Yu, Karthik Nandakumar, Yitong Li, Xingjun Ma, Jiong Jin, Han Yu, and Kee Siong Ng. Towards fair and privacy-preserving federated deep models. *IEEE Transactions on Parallel and Distributed Systems*, 31(11):2524–2541, 2020.

[325] Jing Ma, Qiuchen Zhang, Jian Lou, Joyce Ho, Li Xiong, and Xiaoqian Jiang. Privacy-preserving tensor factorization for collaborative health data analysis. In *ACM CIKM*, volume 2, 2019.

[326] Yuzhe Ma, Xiaojin Zhu, and Justin Hsu. Data poisoning against differentially-private learners: Attacks and defenses. In *International Joint Conference on Artificial Intelligence (IJCAI), Macao, China*, 2019. URL `https://arxiv.org/abs/1903.09860`.

[327] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *ICML*, 2018.

[311]Yujun Lin，Song Han，Huizi Mao，Yu Wang，and William J Dally.深度梯度压缩：减少分布式训练的通信带宽。arXiv预印本arXiv：1712.01887，2017。

[312]R. J. A.点后分层：建模者的视角。Journal of the American Statistical Association，88（423）：1001-1012，1993. ISSN 01621459。

[313]Hanxiao Liu，Karen Simonyan，and Yiming Yang. DARTS：差异化架构搜索。arXiv预印本arXiv：1806.09055，2018。

[314]Kang Liu，Brendan Dolan-Gavitt，and Siddharth Garg.精细修剪：防御后门攻击

深度神经网络在攻击、入侵和防御研究国际研讨会上，第273-294页。Springer，2018年。

[315]刘喜阳和吴世雄。近似差分隐私估计的极大极小率。arXiv预印本arXiv：1905.10335，2019。

[316]Yang Liu，Yan Kang，Xinwei Zhang，Liping Li，Yong Cheng，Tianjian Chen，Mingyi Hong，and Qiang Yang.

一个通信高效的垂直联邦学习框架。CoRR，abs/1912.11187，2019。网址http：//arxiv.org/abs/1912.11187。

[317]刘扬，康彦，邢朝平，陈天剑，杨强。安全的联邦迁移学习框架。IEEE智能系统，35（4）：70-82，2020。doi：10.1109/MIS.2020.2988525。

[318]杨柳，易志豪，陈天剑。后门攻击和防御功能分区协作学习，2020年。

[319]Yingqi Liu，Shiqing Ma，Yousra Aafer，Wen-Chuan Lee，Juan Zhai，Weihang Wang，and Xiangyu Zhang.特罗-

janing攻击神经网络在第25届年度网络和分布式系统安全研讨会，NDSS 2018，圣地亚哥，加州，美国，2018年2月18日至21日。网址http://wp.internetsociety。org/ndss/wp-content/uploads/sites/25/2018/02/ndss2018_03A-5_Liu_paper.pdf。

[320]Yuhan Liu，Ananda Theertha Suresh，Felix Xinnan X Yu，Sanjiv Kumar，and Michael Riley.学习离散分布：用户与项目级隐私。神经信息处理系统的进展，33，2020。

[321]Heiko Ludwig、Nathalie Baracaldo、Gegi Thomas、Yi Zhou、Ali Anwar、Shashank Rajamoni、Yuya

亚拉姆·拉德哈克里希南、阿希什·维尔马、马蒂厄·辛恩等。IBM联合学习：企业框架白皮书V0.1。arXiv预印arXiv：2007.10987，2020。

[322]嘉欢罗、雪阳吴、云罗、安布黄、云峰黄、杨柳、强阳。用于联邦学习的真实世界图像数据集。arXiv预印arXiv：1910.11089，2019。

[323]Renqian Luo，Fei Tian，Tao Qin，Enhong Chen，and Tie-Yan Liu.神经结构优化。神经信息处理系统的进展，第7816-7827页，2018年。

[324]Lingjuan Lyu，Jiangshan Yu，Karthik Nandakumar，Yitong Li，Xingjun Ma，Jiong Jin，Han Yu，and Kee Siong

Ng.迈向公平和隐私保护的联邦深度模型。IEEE Transactions on Parallel and Distributed Systems，31（11）：2524-2541，2020。

[325]Jing Ma，Qiuchen Zhang，Jian Lou，Joyce Ho，Li Xiong，and Xiaoqian Jiang.用于协作健康数据分析的隐私保护张量分解。在ACM CIKM，第2卷，2019年。

[326]Yuzhe Ma，Xiaojin Zhu，Justin Hsu.针对差异化私人学习者的数据中毒：攻击和

的防御合作.国际人工智能联合会议（IJCAI），中国澳门，2019年。网址https://arxiv.org/abs/1903.09860。

[327]大卫马德拉斯，埃利奥特克里格，托尼安皮塔西，和理查德泽梅尔。学习对抗性的公平和可转移的表示。2018年在ICML上发表。

[328] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ICLR*, 2017.

[329] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

[330] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In *Advances in neural information processing systems*, pages 1041–1048, 2009.

[331] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020.

[332] Yishay Mansour, Mehryar Mohri, Ananda Theertha Suresh, and Ke Wu. A theory of multiple-source adaptation with limited target labeled data. *arXiv preprint arXiv:2007.09762*, 2020.

[333] Alicia R Martin, Masahiro Kanai, Yoichiro Kamatani, Yukinori Okada, Benjamin M Neale, and Mark J Daly. Current clinical use of polygenic scores will risk exacerbating health disparities. *BioRxiv*, page 441261, 2019.

[334] H Brendan McMahan and Daniel Ramage. Federated learning: Collaborative machine learning without centralized training data, April 2017. URL `https://ai.googleblog.com/2017/04/federated-learning-collaborative.html`. Google AI Blog.

[335] H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

[336] H Brendan McMahan, Galen Andrew, Ulfar Erlingsson, Steve Chien, Ilya Mironov, Nicolas Papernot, and Peter Kairouz. A general approach to adding differential privacy to iterative training procedures. dec 2018. *URL https://arxiv. org/abs*, 1812.

[337] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, 2017 (original version on arxiv Feb. 2016).

[338] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations (ICLR)*, 2018.

[339] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, pages 94–103, 2007.

[340] Shike Mei and Xiaojin Zhu. Using machine teaching to identify optimal training-set attacks on machine learners. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[341] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. *arXiv preprint arXiv:1805.04049*, 2018.

[342] El Mahdi El Mhamdi, Rachid Guerraoui, and Sébastien Rouault. The hidden vulnerability of distributed learning in Byzantium. In *ICML*, 2018.

[343] Silvio Micali. Computationally sound proofs. *SIAM J. Comput.*, 30(4):1253–1298, 2000.

[344] Fatemehsadat Mireshghallah, Mohammadkazem Taram, , Praneeth Vepakomma, Abhishek Singh, Ramesh Raskar, and Esmaeilzadeh Hadi. Privacy in deep learning: A survey. *arXiv preprint arXiv:2004.12254*, 2020.

[345] Ilya Mironov. On significance of the least significant bits for differential privacy. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 650–661. ACM, 2012.

[346] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[328]亚历山大·马德里、亚历山大·马克洛夫、路德维希·施密特、迪米特里斯·齐普拉斯和阿德里安·弗拉德。深度学习模型抵抗对抗性攻击。ICLR，2017年。

[329]Yishay Mansour，Mehryar Mohri，and Afshin Rostamizadeh.领域适应：学习边界和算法。arXiv预印本arXiv：0902.3430，2009年。

[330]Yishay Mansour，Mehryar Mohri，and Afshin Rostamizadeh.多源域适配。神经信息处理系统的进展，第1041-1048页，2009年。

[331]Yishay Mansour，Mehryar Mohri，Jae Ro，and Ananda Theertha Suresh.三种个性化方法与联邦学习的应用。arXiv预印本arXiv：2002.10619，2020。

[332]Yishay Mansour，Mehryar Mohri，Ananda Theertha Suresh，and Ke Wu.有限目标标记数据的多源自适应理论。arXiv预印本arXiv：2007.09762，2020。

[333]艾丽西娅·R·马丁、金井正弘、镰谷洋一郎、冈田由纪典、本杰明·M·尼尔和马克·J·戴利。

目前临床使用的多基因评分将有加剧健康差异的风险。BioRxiv，第441261页，2019年。

[334]布兰登·麦克马汉和丹尼尔·拉梅奇。联合学习：协作机器学习
没有集中培训数据，2017年4月。网址https://ai.googleblog.com/2017/04/ federated-learning-collaborative. html。Google AI Blog.

[335]布兰登·麦克马汉和马修·斯特里特。在线凸优化的自适应界优化。
arXiv预印本arXiv：1002.4908，2010年。

[336]H Brendan McMahan，Galen Andrew，Ulfar Erlingsson，Steve Chien，Ilya Mironov，Nicolas Papernot，and Peter Kairouz.在迭代训练过程中加入差分隐私的一般方法。2018年12月。
URL https：//arxiv. org/abs，1812.

[337]H布伦丹·麦克马汉，艾德摩尔，丹尼尔·拉梅奇，塞思·汉普森，和布莱斯·阿奎拉y Arcas.
从去中心化数据中高效学习深度网络。在第20届人工智能和统计国际会议论文集，第1273-1282页，2017年（arxiv 2016年2月的原始版本）。

[338]H Brendan McMahan，丹尼尔Ramage，Kunal Talwar，和Li Zhang.学习差分私有递归语言模型。2018年国际学习表征会议（International Conference on Learning Representations，ICLR）

[339]弗兰克·麦克雪莉和库纳尔·塔尔瓦通过差异隐私的机制设计。见FOCS，第94-103页，2007年。

[340]石可梅和朱晓金。使用机器教学来识别对机器学习器的最佳训练集攻击。2015年第29届AAAI人工智能会议。

[341]Luca Melis，Congzheng Song，Emiliano De Cristofaro，and Vitaly Shmatikov.在协作学习中利用非预期的特征泄漏。arXiv预印本arXiv：1805.04049，2018。

[342]El Mahdi El Mhamdi，Rachid Guerraoui和S'ebastien Rouault。拜占庭分布式学习的隐藏漏洞。2018年在ICML上发表。

[343]西尔维奥·米卡利计算上可靠的证据。SIAM J. COMPUT.，30（4）：1253-1298，2000.

[344]Fatemehsadat Mireshghallah，Mohammadkazem Taram，，Praneeth Vepakomma，Abhishek Singh，Ramesh Raskar，and Esmaeilzadeh Hadi.深度学习中的隐私：一项调查。arXiv预印本arXiv：2004.12254，2020。

[345]伊利亚·米罗诺夫关于差分隐私的最低有效位的重要性。2012年ACM计算机和通信安全会议论文集，第650-661页。ACM，2012年。

[346]伊利亚·米罗诺夫R 'enyi差分隐私。2017年IEEE第30届计算机安全基础研讨会（CSF），第263-275页。IEEE，2017年。

[347] Ilya Mironov, Omkant Pandey, Omer Reingold, and Salil Vadhan. Computational differential privacy. In *Advances in Cryptology—CRYPTO*, pages 126–142, 2009.

[348] Ilya Mironov, Kunal Talwar, and Li Zhang. R\'enyi differential privacy of the sampled Gaussian mechanism. *arXiv preprint arXiv:1908.10530*, 2019.

[349] Shira Mitchell, Eric Potash, and Solon Barocas. Prediction-based decisions and fairness: A catalogue of choices, assumptions, and definitions. *arXiv preprint arXiv:1811.07867*, 2018.

[350] Volodymyr Mnih and Geoffrey E Hinton. Learning to label aerial images from noisy data. In *Proceedings of the 29th International conference on machine learning (ICML-12)*, pages 567–574, 2012.

[351] Payman Mohassel and Yupeng Zhang. SecureML: A system for scalable privacy-preserving machine learning. In *IEEE Symposium on Security and Privacy*, pages 19–38. IEEE Computer Society, 2017.

[352] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic Federated Learning. In *ICML*, 2019.

[353] Jose G. Moreno-Torres, Troy Raeder, RocíO Alaiz-RodríGuez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern Recogn.*, 45(1), January 2012.

[354] Musketeer. Musketeer: About, 2019. URL `http://musketeer.eu/project/`. Retrieved Aug 2019.

[355] Carolina Naim, Fangwei Ye, and Salim El Rouayheb. ON-OFF privacy with correlated requests. In *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019.

[356] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013.

[357] Giovanni Neglia, Chuan Xu, Don Towsley, and Gianmarco Calbi. Decentralized gradient methods: does topology matter? In *AISTATS*, 2020.

[358] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

[359] Valeria Nikolaenko, Udi Weinsberg, Stratis Ioannidis, Marc Joye, Dan Boneh, and Nina Taft. Privacy-preserving ridge regression on hundreds of millions of records. In *IEEE Symposium on Security and Privacy*, pages 334–348. IEEE Computer Society, 2013.

[360] Chaoyue Niu, Fan Wu, Shaojie Tang, Lifeng Hua, Rongfei Jia, Chengfei Lv, Zhihua Wu, and Guihai Chen. Secure federated submodel learning. *arXiv preprint arXiv:1911.02254*, 2019.

[361] NSA. Defense in depth: A practical strategy for achieving Information Assurance in today's highly networked environments. Technical report, NSA, 2012.

[362] Deniz Oktay, Johannes Ballé, Saurabh Singh, and Abhinav Shrivastava. Model compression by entropy penalized reparameterization. *arXiv preprint arXiv:1906.06624*, 2019.

[363] Femi Olumofin and Ian Goldberg. Revisiting the computational practicality of private information retrieval. In *International Conference on Financial Cryptography and Data Security*, pages 158–172. Springer, 2011.

[364] Palisade. PALISADE lattice cryptography library. `https://gitlab.com/palisade/palisade-release`, October 2019.

[365] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[366] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519. ACM, 2017.

[347]伊利亚·米罗诺夫，奥姆坎特·潘迪，奥默·莱因戈尔德和萨里尔·瓦丹。计算差异隐私。在Advances in Cryptology-Cryptology PTO，第126-142页，2009年。

[348]Ilya Mironov，Kunal Talwar，和Li Zhang。采样高斯机制的R\'enyi差分隐私。

arXiv预印本arXiv：1908.10530，2019。

[349]希拉·米切尔，埃里克·波塔什，梭伦·巴洛卡斯。基于预测的决策和公平性：选择，假设和定义的目录。arXiv预印本arXiv：1811.07867，2018。

[350]Volodymyr Mnih和Geoffrey E欣顿。学习从噪声数据中标记航空图像。第29届国际机器学习会议（ICML-12），第567-574页，2012年。

[351]Payman Mohassel和Yupeng Zhang。SecureML：可扩展的隐私保护机器学习系统。

IEEE Symposium on Security and Privacy，第19-38页。IEEE计算机协会，2017年。

[352]Mehryar Mohri，加里Sivek，和Ananda Theertha Suresh。不可知联邦学习。2019年，在ICML中。

[353]何塞·G Moreno-Torres，Troy Raeder，Roc 'ıO Alaiz-Rodr' ıGuez，Nitesh V. Chawla，and弗朗西斯科Herrera.分类中数据集转移的统一观点。模式n.，45（1），2012年1月。

[354]火枪手火枪手：大约，2019年。网址http://musketeer.eu/project/。2019年8月恢复。

[355]卡罗莱纳奈姆，叶芳伟，萨利姆·埃尔·鲁艾赫布。相关请求的ON-OFF隐私。2019年7月，IEEE信息理论国际研讨会（ISIT）。

[356]Nagarajan Natarajan，Inderjit S Dhillon，Pradeep K Ravikumar，and Ambuj Tewari.用嘈杂的标签学习。

神经信息处理系统的进展，第1196-1204页，2013年。

[357]Giovanni Neglia，Chuan Xu，Don Towsley，and Gianmarco Calbi.分散梯度法：拓扑重要吗？在AISTATS，2020年。

[358]Alex Nichol，约书亚Achiam，和John Schulman.一阶元学习算法。arXiv预印本arXiv：1803.02999，2018。

[359]Valeria Nikolaenko，Udi Weinsberg，Stratis Ioannyoung，Marc Joye，Dan Boneh，and Nina塔夫脱.隐私-

保留了数亿条记录上的岭回归。IEEE Symposium on Security and Privacy，第334-348页。IEEE计算机协会，2013年。

[360]Chaoyue Niu，Fan Wu，Shaojie Tang，Lifeng Hua，Rongfei Jia，Chengfei Lv，Zhihua Wu，and Guihai Chen.安全联邦子模型学习。arXiv预印本arXiv：1911.02254，2019。

[361]国安局深入防御：在当今高度网络化的环境中实现信息保障的实用战略。技术报告，国家安全局，2012年。

[362]Deniz Oktay，Johannes Ball 'e，Saurabh Singh和Abhinav Shrivastava。用熵惩罚重参数化方法压缩模型。arXiv预印本arXiv：1906.06624，2019。

[363]Femi Olumofin和Ian Goldberg。重新审视私人信息检索的计算实用性。在金融密码学和数据安全国际会议上，第158-172页。Springer，2011.

[364]栅栏。栅栏点阵密码库。https://gitlab.com/palisade/ palisade-release，October 2019.

[365]Sinno Jialin Pan和Qiang Yang。迁移学习研究综述。IEEE Transactions on Knowledge and Data Engineering，22（10）：1345-1359，2010.

[366]Nicolas Papernot，帕特里克麦克丹尼尔，伊恩古德费洛，Somesh Jha，Z Berkay Celik和Ananthram Swami。

针对机器学习的实用黑盒攻击。2017年ACM亚洲计算机和通信安全会议论文集，第506-519页。ACM，2017年。

[367] Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Úlfar Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. *arXiv preprint arXiv:2007.14191*, 2020.

[368] Jihong Park, Sumudu Samarakoon, Mehdi Bennis, and Mérouane Debbah. Wireless network intelligence at the edge. *CoRR*, abs/1812.02858, 2018. URL http://arxiv.org/abs/1812.02858.

[369] Bryan Parno, Jon Howell, Craig Gentry, and Mariana Raykova. Pinocchio: nearly practical verifiable computation. *Commun. ACM*, 59(2):103–112, 2016.

[370] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[371] Kumar Kshitij Patel and Aymeric Dieuleveut. Communication trade-offs for synchronized distributed SGD with large step size. *NeurIPS*, 2019.

[372] Sarvar Patel, Giuseppe Persiano, and Kevin Yeo. Private stateful information retrieval. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, pages 1002–1019, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5693-0. doi: 10.1145/3243734.3243821. URL http://doi.acm.org/10.1145/3243734.3243821.

[373] Giorgio Patrini, Richard Nock, Stephen Hardy, and Tibério S. Caetano. Fast learning from distributed datasets without entity matching. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1909–1917, 2016. URL http://www.ijcai.org/Abstract/16/273.

[374] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. *arXiv preprint arXiv:1602.02355*, 2016.

[375] Hieu Pham, Melody Guan, Barret Zoph, Quoc Le, and Jeff Dean. Efficient neural architecture search via parameter sharing. In *International Conference on Machine Learning*, pages 4092–4101, 2018.

[376] Sundar Pichai. Google's Sundar Pichai: Privacy Should Not Be a Luxury Good. *New York Times*, May 7, 2019.

[377] Venkatadheeraj Pichapati, Ananda Theertha Suresh, Felix X Yu, Sashank J Reddi, and Sanjiv Kumar. AdaCliP: Adaptive clipping for private SGD. *arXiv preprint arXiv:1908.07643*, 2019.

[378] Vasyl Pihur, Aleksandra Korolova, Frederick Liu, Subhash Sankuratripati, Moti Yung, Dachuan Huang, and Ruogu Zeng. Differentially-private "Draw and Discard" machine learning. *CoRR*, abs/1807.04369, 2018. URL http://arxiv.org/abs/1807.04369.

[379] Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445*, 2019.

[380] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009. ISBN 0262170051, 9780262170055.

[381] Shashank Rajput, Hongyi Wang, Zachary Charles, and Dimitris Papailiopoulos. DETOX: A redundancy-based framework for faster and more robust gradient aggregation. *arXiv preprint arXiv:1907.12205*, 2019.

[382] Daniel Ramage and Stefano Mazzocchi. Federated analytics: Collaborative data science without data collection, May 2020. URL https://ai.googleblog.com/2020/05/federated-analytics-collaborative-data.html. Google AI Blog.

[367]Nicolas Papernot，Abhradeep Thakurta，Shuang Song，Steve Chien和Ulfar Erlingsson。用于具有差分隐私的深度学习的回火sigmoid激活。arXiv预印本arXiv：2007.14191，2020。

[368]朴继红，萨马拉孔，迈赫迪·本尼斯和梅鲁安·德巴。边缘的无线网络智能。CoRR，abs/1812.02858，2018。网址http://arxiv.org/abs/1812.02858。

[369]布莱恩·帕诺，乔恩·豪厄尔，克雷格·金特里，还有玛丽安娜·雷科娃。匹诺曹：近乎实用的可验证计算。Commun. ACM，59（2）：103-112，2016.

[370]亚当-帕斯克，萨姆-格罗斯，弗朗西斯科马萨，亚当-莱勒，詹姆斯-布拉德伯里，格雷戈里-查南，

特雷弗基林、林泽明、娜塔莉亚吉梅尔辛、卢卡·安提加、阿尔班·德梅森、安德烈亚斯·科普夫、爱德华·杨、扎卡里·德维托、马丁·雷森、阿利汗·特贾尼、萨桑克·奇拉姆库尔蒂、伯努瓦·施泰纳、卢芳、白俊杰和苏米特·钦塔拉。Pytorch：一个命令式风格的高性能深度学习库。In H.沃勒克，H。拉罗谢勒A.贝盖尔齐默，F. d'Alch 'eBuc，E. Fox和R. Garnett，编辑，神经信息处理系统进展32，第8024-8035页。柯兰联营公司2019.网址http://papers.neurips.cc/paper/ 9015-pytorch-an-imperative-style-high-performance-deep-learning-library.

pdf.

[371]Kumar Kshitij Patel和Aymeric Dieuleveut。具有大步长的同步分布式SGD的通信权衡。NeurIPS，2019年。

[372]萨瓦尔·帕特尔朱塞佩·佩西亚诺和凯文·杨。私有状态信息检索。In Proceedings of

2018年ACM SIGSAC计算机和通信安全会议，CCS '18，第1002-1019页，纽约，纽约州，美国，2018年。ACM。ISBN 978-1-4503-5693-0。doi：10.1145/3243734.3243821。网址http：//doi.acm.org/10.1145/3243734.3243821。

[373]Giorgio Patrini，Richard Nock，Stephen哈代和Tib 'erio S.卡埃塔诺从分布式快速学习
没有实体匹配的数据集。在第二十五届国际人工智能联合会议的会议记录中，IJCAI 2016，纽约，纽约州，美国，2016年7月9日至15日，第1909-1917页，2016年。网址http://www.ijcai.org/Abstract/16/273。

[374]费边·佩德雷戈萨近似梯度超参数优化。arXiv预印本arXiv：1602.02355，
2016.

[375]Hieu Pham，Melody Guan，Barret Zoph，Quoc Le和Jeff Dean。通过参数共享的高效神经结构搜索。在国际机器学习会议上，第4092-4101页，2018年。

[376]桑达尔·皮查伊Google的Sundar Pichai：隐私不应该是奢侈品纽约时报，2019年5月7日。

[377]Venkatadheeraj Pichapati，Ananda Theertha Suresh，Felix X Yu，Sashank J Reddi，and Sanjiv Kumar. AdaCliP：自适应裁剪私有SGD。arXiv预印本arXiv：1908.07643，2019。

[378]Vasyl Pihur，Aleksandra Korolova，Frederick Liu，Subhash Sankuratripati，Moti Yung，Dachuan Huang，and
曾若谷。差异私有的“抽取和丢弃”机器学习。CoRR，abs/1807.04369，2018。网址http://arxiv.org/abs/1807.04369。

[379]Krishna Pillutla，Sham M Kakade，and Zaid Harchaoui.用于联邦学习的强大聚合。arXiv预印本arXiv：1912.13445，2019。

[380]Joaquin Quionero-Canadian，Masashi Sugiyama，Anton Schwaighthorn，and Neil D.劳伦斯机器学习中的数据集转移。MIT Press，2009. ISBN 0262170051，9780262170055。

[381]Shashank Rajput，Hongyi Wang，Zachary Charles，and Dimitris Papailiopoulos. DETOX：一个基于冗余的框架，用于更快，更强大的梯度聚合。arXiv预印本arXiv：1907.12205，2019。

[382]丹尼尔·拉米奇和斯特凡诺·马佐奇。联合分析：协作数据科学

没有数据收集，2020年5月。网址https://ai.googleblog.com/2020/05/ federated-analytics-collaborative-data.html。Google AI Blog.

[383] Swaroop Ramaswamy, Rajiv Mathews, Kanishka Rao, and Françoise Beaufays. Federated learning for emoji prediction in a mobile keyboard. *arXiv preprint 1906.04329*, 2019.

[384] Swaroop Ramaswamy, Om Thakkar, Rajiv Mathews, Galen Andrew, H Brendan McMahan, and Françoise Beaufays. Training production language models without memorizing user data. *arXiv preprint arXiv:2009.10031*, 2020.

[385] Vibhor Rastogi and Suman Nath. Differentially private aggregation of distributed time-series with transformation and encryption. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, pages 735–746, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0032-2. doi: 10.1145/1807167.1807247. URL `http://doi.acm.org/10.1145/1807167.1807247`.

[386] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

[387] Esteban Real, Sherry Moore, Andrew Selle, Saurabh Saxena, Yutaka Leon Suematsu, Jie Tan, Quoc V Le, and Alexey Kurakin. Large-scale evolution of image classifiers. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2902–2911. JMLR. org, 2017.

[388] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4780–4789, 2019.

[389] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

[390] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. *arXiv preprint arXiv:1909.13014*, 2019.

[391] Amirhossein Reisizadeh, Hossein Taheri, Aryan Mokhtari, Hamed Hassani, and Ramtin Pedarsani. Robust and communication-efficient collaborative learning. *arXiv:1907.10595*, 2019.

[392] Leonid Reyzin, Adam D. Smith, and Sophia Yakoubov. Turning HATE into LOVE: homomorphic ad hoc threshold encryption for scalable MPC. *IACR Cryptology ePrint Archive*, 2018:997, 2018.

[393] M Sadegh Riazi, Kim Laine, Blake Pelton, and Wei Dai. HEAX: High-performance architecture for computation on homomorphically encrypted data in the cloud. *arXiv preprint arXiv:1909.09731*, 2019.

[394] Rashida Richardson, Jason Schultz, and Kate Crawford. Dirty data, bad predictions: How civil rights violations impact police data, predictive policing systems, and justice. *New York University Law Review Online, Forthcoming*, 2019.

[395] Brian D Ripley. Statistical aspects of neural networks. *Networks and chaos—statistical and probabilistic aspects*, 50:40–123, 1993.

[396] Ronald L Rivest, Len Adleman, and Michael L Dertouzos. On data banks and privacy homomorphisms. *Foundations of Secure Computation, Academia Press*, pages 169–179, 1978.

[397] Nuria Rodríguez-Barroso, Goran Stipcich, Daniel Jiménez-López, José Antonio Ruiz-Millán, Eugenio Martínez-Cámara, Gerardo González-Seco, M Victoria Luzón, Miguel Angel Veganzones, and Francisco Herrera. Federated learning and differential privacy: Software tools analysis, the sherpa. ai fl framework and methodological guidelines for preserving data privacy. *Information Fusion*, 64:270–292, 2020.

[398] Edo Roth, Daniel Noble, Brett Hemenway Falk, and Andreas Haeberlen. Honeycrisp: large-scale differentially private aggregation without a trusted core. In *SOSP*, pages 196–210. ACM, 2019.

[399] Theo Ryffel, Andrew Trask, Morten Dahl, Bobby Wagner, Jason Mancuso, Daniel Rueckert, and Jonathan Passerat-Palmbach. A generic framework for privacy preserving deep learning, 2018.

[383]Swaroop Ramaswamy，Rajiv马修斯，Kanishka Rao，和Franc Mr.用于移动的键盘中的表情符号预测的联合学习。arXiv预印本1906.04329，2019年。

[384]Swaroop Ramaswamy，Om Thakkar，Rajiv马修斯，Galen Andrew，H Brendan McMahan，and Franc

乌菲斯训练生产语言模型，无需记忆用户数据。arXiv预印本arXiv：2009.10031，2020。

[385]维波·拉斯托吉和苏曼·纳特基于transfor的分布式时间序列差分私有聚集
信息和加密。在2010年ACM SIGMOD数据管理国际会议论文集，SIGMOD '10，第735-746页，纽约，纽约州，美国，2010年。ACM。ISBN 978-1-4503-0032-2。doi：10.1145/1807167.1807247。网址http://doi.acm.org/10.1145/1807167.1807247。

[386]Sachin Ravi和Hugo Larochelle。优化作为一个模型，为少数拍摄学习。在2017年第五届学习表征国际会议的会议记录中。

[387]Esteban真实的，Sherry摩尔，Andrew Selle，Saurabh Saxena，Yutaka Leon Suematsu，Jie Tan，Quoc V

阿列克谢·库拉金图像分类器的大规模进化。第34届机器学习国际会议论文集-第70卷，第2902-2911页。JMLR。org，2017.

[388]Esteban真实的，Alok Aggarwal，Yanping Huang和Quoc V Le。图像分类器的正则化进化算法
建筑搜索在AAAI人工智能会议论文集，第33卷，第4780- 4789页，2019年。

[389]Sashank Reddi，Zachary Charles，Manzil Zaheer，Zachary Garrett，Keith Rush，Jakub Kone‘cn’y，Sanjiv Kumar和H Brendan McMahan。Adaptive Federated Optimization自适应联邦优化arXiv preprint arXiv：2003.00295，2020.（英文）

具有周期性平均和量化的通信高效的联邦学习方法。arXiv预印本arXiv：1909.13014，2019。

[391]Amirhossein Reisizadeh，Hossein Taheri，Aryan Mokhtari，Hamed Hassani，and Ramtin Pedarsani.强大而高效的协作学习。arXiv：1907.10595，2019。

[392]放大图片作者：Adam D.史密斯和索菲亚·雅库博夫把恨变成爱：用于可扩展MPC的同态特设阈值加密。IACR Cryptology ePrint Archive，2018：997，2018.

[393]M Sadegh Riazi，Kim Laine，Blake Pelton和Wei Dai。HEAX：用于计算云中同态加密数据的高性能架构。arXiv预印本arXiv：1909.09731，2019。

[394]拉希达·理查森杰森·舒尔茨和凯特·克劳福德肮脏的数据，糟糕的预测：民权如何侵犯-

影响警方数据，预测警务系统和司法。纽约大学法律评论在线，即将出版，2019年。

[395]布莱恩·D·雷普利神经网络的统计方面。网络和混沌-统计和概率方面，50：40-123，1993。

[396]罗纳德L Rivest，Len Adleman和Michael L Dertouzos。关于数据库和隐私同态。安全计算基础，学术出版社，169-179页，1978年。

[397]Nuria Rodríguez-Barroso、Goran Stipcich、Daniel Jim 'enez-L'opez、Jose' e Antonio Ruiz-Mill 'an、Eugenio

Mart 'ınez-C'阿马拉，Gerardo Gonz 'alez-Seco，M维多利亚Luz' on，Miguel Angel Veganzones和弗朗西斯科Herrera。联邦学习和差异隐私：软件工具分析，夏尔巴人。保护数据隐私的AI FL框架和方法指南。信息融合，64：270-292，2020。

[398]Edo Roth，丹尼尔诺布尔，Brett Hemenway Falk，和Andreas Haeberlen. Honeycrisp：没有可信核心的大规模差异私有聚合。见SOSP，第196-210页。ACM，2019年。

[399]西奥·赖恩，安德鲁·特拉斯克，莫滕·达尔，鲍比·瓦格纳，杰森·曼库索，丹尼尔·鲁克特和乔纳森·帕瑟拉特·帕姆巴赫。隐私保护深度学习的通用框架，2018年。

[400] César Sabater, Aurélien Bellet, and Jan Ramon. Distributed Differentially Private Averaging with Improved Utility and Robustness to Malicious Parties. *arXiv preprint arXiv:2006.07218*, 2020.

[401] John K Salmon, Mark A Moraes, Ron O Dror, and David E Shaw. Parallel random numbers: As easy as 1, 2, 3. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, page 16. ACM, 2011.

[402] Sumudu Samarakoon, Mehdi Bennis, Walid Saad, and Mérouane Debbah. Federated learning for ultra-reliable low-latency V2V communications. *CoRR*, abs/1805.09253, 2018. URL http://arxiv.org/abs/1805.09253.

[403] Nithya Sambasivan, Garen Checkley, Amna Batool, Nova Ahmed, David Nemer, Laura Sanely Gaytán-Lugo, Tara Matthews, Sunny Consolvo, and Elizabeth Churchill. " privacy is not for me, it's for those rich women": Performative privacy practices on mobile phones by women in south asia. In *Fourteenth Symposium on Usable Privacy and Security ({SOUPS} 2018)*, pages 127–142, 2018.

[404] Sai Sri Sathya, Praneeth Vepakomma, Ramesh Raskar, Ranjan Ramachandra, and Santanu Bhattacharya. A review of homomorphic encryption libraries for secure computation. *arXiv preprint arXiv:1812.02428*, 2018.

[405] Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Robust and communication-efficient federated learning from non-IID data. *arXiv preprint arXiv:1903.02891*, 2019.

[406] R. Schnell. Efficient private record linkage of very large datasets. In *59$^{th}$ World Statistics Congress*, 2013.

[407] R. Schnell, T. Bachteler, and J. Reiher. A novel error-tolerant anonymous linking code. Technical report, Paper No. WP-GRLC-2011-02, German Record Linkage Center Working Paper Series, 2011.

[408] Claus P. Schnorr. Efficient identification and signatures for smart cards. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology*, EUROCRYPT '89, 1990.

[409] SEAL. Microsoft SEAL (release 3.6). https://github.com/Microsoft/SEAL, November 2020. Microsoft Research, Redmond, WA.

[410] Frank Seide and Amit Agarwal. Cntk: Microsoft's open-source deep-learning toolkit. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 2135, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2945397. URL https://doi.org/10.1145/2939672.2945397.

[411] Arvind Seshadri, Mark Luk, Adrian Perrig, Leendert van Doom, and Pradeep K. Khosla. Pioneer: Verifying code integrity and enforcing untampered code execution on legacy systems. In *Malware Detection*, volume 27 of *Advances in Information Security*, pages 253–289. Springer, 2007.

[412] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free. *NeurIPS*, 2019.

[413] Vivek Sharma, Praneeth Vepakomma, Tristan Swedish, Ken Chang, Jayashree Kalpathy-Cramer, and Ramesh Raskar. ExpertMatcher: Automating ML model selection for clients using hidden representations. *arXiv preprint arXiv:1910.03731*, 2019.

[414] Yash Sharma and Pin-Yu Chen. Attacking the Madry defense model with $l\_1$-based adversarial examples. *arXiv preprint arXiv:1710.10733*, 2017.

[415] SHELL. https://github.com/google/shell-encryption, December 2020. Google.

[416] Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5739–5748, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/shen19e.html.

[400]塞萨尔·萨巴特，奥莉莲·贝莱特，扬·拉蒙。具有改进的效用和对恶意方的鲁棒性的分布式差分私有平均。arXiv预印本arXiv：2006.07218，2020。

[401]约翰·K·萨尔蒙，马克·A·莫赖斯，罗恩·O·德罗尔和大卫·E·肖。平行随机数：就像1，2，3一样简单。2011年高性能计算、网络、存储和分析国际会议论文集，第16页。ACM，2011年。

[402]Sumudu Samarakoon，Mehdi Bennis，Walid Saad和M'erouane Debbah。联合学习实现超可靠的低延迟V2V通信。CoRR，abs/1805.09253，2018。网址http://arxiv.org/abs/1805。09253.

[403]Nithya Sambasivan，Garen Checkley，Amna Batool，Nova Ahmed，大卫Nemer，Laura Sanely Gayt 'an-Lugo，塔拉马修斯桑尼康索沃和伊丽莎白丘吉尔。“隐私不是为我，而是为那些有钱的女人”：南亚妇女在移动的电话上的表演性隐私实践。在第十四届可用隐私和安全研讨会（{SOUPS} 2018），第127-142页，2018年。

[404]Sai Sri Sathya，Praneeth Vepakomma，Ramesh Raskar，Ranjan Ramachandra，and Santanu Bhattacharya.用于安全计算的同态加密库综述。arXiv预印本arXiv：1812.02428，2018。

[405]Felix Sattler，Simon Wiedemann，Klaus-Robert Müuller，and Wojciech Samek.从非IID数据中进行健壮且通信高效的联邦学习。arXiv预印本arXiv：1903.02891，2019。

[406]R。快的有效的私有记录链接非常大的数据集。2013年，第59届世界统计大会。

[407]R。快，T。Bachteler，和J. Reiher。A Novell Error Tolerant Anonymous Linking Code（能够容忍匿名链接代码）技术报告，论文No. WP—GRLC—2011—02，德国记录链接中心工作论文系列，2011年。

[408]Claus P. Schnorr。为智能卡提供有效的识别和签名1990年，《密码学理论与应用研讨会进展》（In Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology，EUROCRYPT '89）。

[409]印Microsoft SEAL（3.6版）。https：//github.com/Microsoft/SEAL，2020年11月。微软研究院，华盛顿州雷德蒙。

[410]弗兰克·赛德和阿米特·阿加瓦尔Cntk：微软的开源深度学习工具包。In Proceedings of the第22届ACM SIGKDD知识发现和数据挖掘国际会议，KDD '16，第2135页，纽约，纽约州，美国，2016年。计算机协会。ISBN 9781450342322。doi：10.1145/2939672.2945397. URLhttps://doi.org/10.1145/2939672.2945397.

[411]Arvind Seshadri，Mark Luk，Adrian Perrig，Leendert货车Doom和Pradeep K.科斯拉先锋：代码完整性和在遗留系统上强制执行未经篡改的代码。在Malware Detection中，Advances in Information Security的第27卷，第253-289页。Springer，2007年。

[412]Ali Shafahi、Mahyar Najibi、Amin Ghiasi、Zheng Xu、John Dickerson、Christoph Studer、Larry S Davis、Gavin Taylor和Tom Goldstein。免费对抗训练。NeurIPS，2019年。

[413]Vivek Sharma、Praneeth Vepakomma、Tristan Swedish、Ken Chang、Jayashree Kalpathy-Cramer和拉斯卡ExpertMatcher：使用隐藏表示为客户端自动选择ML模型。arXiv预印本arXiv：1910.03731，2019。

[414]Yash Sharma和Pin-Yu Chen。使用基于l 1的对抗性示例攻击Madry防御模型。arXiv预印本arXiv：1710.10733，2017。

[415]壳https：//github.com/google/shell-encryption，2020年12月。Google.

[416]Yanyao Shen和Sujay Sanghavi。通过迭代修剪损失最小化学习不良训练数据。在Kamalika Chaudhuri和Ruslan Salakhutdinov，编辑，第36届国际机器学习会议论文集，机器学习研究论文集第97卷，第5739-5748页，长滩，加州，美国，2019年6月9日至15日。PMLR。网址http://proceedings.mlr.press/v97/shen19e。html.

[417] Elaine Shi, HTH Chan, Eleanor Rieffel, Richard Chow, and Dawn Song. Privacy-preserving aggregation of time-series data. In *Annual Network & Distributed System Security Symposium (NDSS)*, 2011.

[418] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[419] Kumar Shridhar, Felix Laumann, and Marcus Liwicki. A comprehensive guide to Bayesian convolutional neural network with variational inference. *arXiv preprint: 1901.02731*, 2019.

[420] Daniel L. Silver, Qiang Yang, and Lianghao Li. Lifelong machine learning systems: Beyond learning algorithms. In *AAAI Spring Symposium Series*, 2013.

[421] Abhishek Singh, Praneeth Vepakomma, Otkrist Gupta, and Ramesh Raskar. Detailed comparison of communication efficiency of split learning and federated learning. *arXiv preprint arXiv:1909.09145*, 2019.

[422] Abhishek Singh, Ayush Chopra, Vivek Sharma, Ethan Garza, Emily Zhang, Praneeth Vepakomma, and Ramesh Raskar. DISCO: Dynamic and invariant sensitive channel obfuscation for deep neural networks. 2020.

[423] Radu Sion and Bogdan Carbunar. On the computational practicality of private information retrieval. In *Proceedings of the Network and Distributed Systems Security Symposium*, pages 2006–06. Internet Society, 2007.

[424] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S. Talwalkar. Federated Multi-Task Learning. In *NIPS*, 2017.

[425] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017.

[426] Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. Scalable Bayesian optimization using deep neural networks. In *International conference on machine learning*, pages 2171–2180, 2015.

[427] Jinhyun So, Basak Guler, and A. Salman Avestimehr. Byzantine-resilient secure federated learning. *IEEE Journal on Selected Areas in Communication, Series on Machine Learning for Communications and Networks*, 2020.

[428] Jinhyun So, Basak Guler, and A Salman Avestimehr. Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *arXiv preprint arXiv:2002.04156*, 2020.

[429] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In *In Proceedings of the ACM Conference on Computer and Communication Security (CCS)*, 2019.

[430] K Srinathan and C Pandu Rangan. Efficient asynchronous secure multiparty distributed computation. In *International Conference on Cryptology in India*, pages 117–129. Springer, 2000.

[431] Brij Mohan Lal Srivastava, Aurélien Bellet, Marc Tommasi, and Emmanuel Vincent. Privacy-Preserving Adversarial Representation Learning in ASR: Reality or Illusion? In *Annual Conference of the International Speech Communication Association (Interspeech)*, 2019.

[432] Jacob Steinhardt, Pang Wei W Koh, and Percy S Liang. Certified defenses for data poisoning attacks. In *Advances in neural information processing systems*, pages 3517–3529, 2017.

[433] Thomas Steinke and Jonathan Ullman. Tight lower bounds for differentially private selection. In *FOCS*, pages 552–563, 2017.

[434] Sebastian U Stich. Local SGD converges fast and communicates little. In *International Conference on Learning Representations (ICLR)*, 2019.

[417]Elaine Shi，HTH Chan，Eleanor Rieffel，Richard Chow和Dawn Song。时间序列数据的隐私保护聚合。2011年网络与分布式系统安全研讨会（NDSS）。

[418]Reza Shokri，Marco Stronati，Congzheng Song，and Vitaly Shmatikov.对机器学习模型的成员推理攻击。2017年IEEE安全与隐私研讨会（SP），第3-18页。IEEE，2017年。

[419]Kumar Shridhar，Felix Laumann，and Marcus Liwicki.贝叶斯卷积神经网络与变分推理的综合指南。arXiv预印本：1901.02731，2019。

[420]丹尼尔湖银，杨强，李良浩。终身机器学习系统：超越学习算法。在AAAI春季研讨会系列，2013年。

[421]Abhishek Singh，Praneeth Vepakomma，Otkrist Gupta，and Ramesh Raskar.详细比较了分裂学习和联邦学习的通信效率。arXiv预印本arXiv：1909.09145，2019。

[422]Abhishek Singh、Ayush Chopra、Vivek Sharma、Ethan Garza、艾米丽张、Praneeth Vepakomma和Ramesh Raskar。DISCO：用于深度神经网络的动态和不变敏感通道混淆。2020.

[423]拉杜锡永和博格丹·卡布纳。私人信息检索的计算实用性。网络和分布式系统安全研讨会论文集，第2006-06页。互联网协会，2007年。

[424]Virginia Smith、Chao-Kai Chiang、Maziar Sanjabi和Ameet S. Talwalkar。联邦多任务学习。在NIPS，2017年。

[425]Jake Snell，Kevin Swersky和Richard S.泽梅尔用于少次学习的原型网络。在神经信息处理系统的进展，2017年。

[426]碧玉斯诺克，奥伦Rippel，凯文Swersky，瑞安Kiros，Nadathur Satish，纳拉亚南Sundaram，Mostofa帕特-

警惕，Prabhat先生，和瑞安亚当斯。使用深度神经网络的可扩展贝叶斯优化。在机器学习国际会议上，第2171-2180页，2015年。

[427]Jinhyun So，Basak Guler和A. Salman Avestimehr.拜占庭弹性安全联邦学习。IEEE Journal on Selected Areas in Communication，Series on Machine Learning for Communications and Networks，2020。

[428]Jinhyun So，Basak Guler和A Salman Avestimehr。Turbo-aggregate：打破安全联邦学习中的二次聚合障碍。arXiv预印本arXiv：2002.04156，2020。

[429]Liwei Song，Reza Shokri和Prateek Mittal。保护机器学习模型的隐私风险

敌对的例子。在ACM计算机和通信安全会议（CCS）的会议记录中，2019年。

[430]K Srinathan和C Pandu Rangan。高效的异步安全多方分布式计算。在印度举行的国际密码学会议上，第117-129页。斯普林格，2000年。

[431]Brij Mohan Lal Srivastava，Aur 'elien Bellet，Marc Tommasi和Emmanuel Vincent。隐私保护广告

ASR中的对抗性表示学习：现实还是幻觉？在2019年国际语音通信协会（Interspeech）年会上。

[432]Jacob Steinhardt、Pang Wei W Koh和珀西S Liang。针对数据中毒攻击的认证防御。神经信息处理系统的进展，第3517-3529页，2017年。

[433]托马斯斯坦克和乔纳森厄尔曼。差分私人选择的严格下限。在FOCS，第552-563页，2017年。

[434]塞巴斯蒂安U斯蒂奇。本地SGD收敛速度快，通信量小。国际学习表征会议（ICLR），2019年。

[435] Sebastian U Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for SGD with delayed gradients and compressed communication. *arXiv:1909.05350*, 2019.

[436] Lili Su and Nitin H. Vaidya. Fault-Tolerant Multi-Agent Optimization: Optimal Iterative Distributed Algorithms. In *PODC*, 2016.

[437] Pramod Subramanyan, Rohit Sinha, Ilia Lebedev, Srinivas Devadas, and Sanjit A Seshia. A formal foundation for secure remote execution of enclaves. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 2435–2450. ACM, 2017.

[438] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H Brendan McMahan. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963*, 2019.

[439] support.google. Your chats stay private while Messages improves suggestions, 2019. URL `https://support.google.com/messages/answer/9327902`. Retrieved Aug 2019.

[440] Ananda Theertha Suresh, Felix X. Yu, Sanjiv Kumar, and H Brendan McMahan. Distributed mean estimation with limited communication. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3329–3337. JMLR. org, 2017.

[441] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *ICLR*, 2013.

[442] Gábor J Székely, Maria L Rizzo, Nail K Bakirov, et al. Measuring and testing dependence by correlation of distances. *The annals of statistics*, 35(6):2769–2794, 2007.

[443] Hanlin Tang, Xiangru Lian, Ming Yan, Ce Zhang, and Ji Liu. D2: Decentralized training over decentralized data. In *ICML*, 2018.

[444] Hanlin Tang, Xiangru Lian, Shuang Qiu, Lei Yuan, Ce Zhang, Tong Zhang, and Ji Liu. DeepSqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression. *arXiv preprint arXiv:1907.07346*, 2019.

[445] Jun Tang, Aleksandra Korolova, Xiaolong Bai, Xueqiang Wang, and XiaoFeng Wang. Privacy loss in Apple's implementation of differential privacy on MacOS 10.12. *CoRR*, abs/1709.02753, 2017. URL `http://arxiv.org/abs/1709.02753`.

[446] Om Thakkar, Galen Andrew, and H Brendan McMahan. Differentially private learning with adaptive clipping. *arXiv preprint arXiv:1905.03871*, 2019.

[447] Florian Tramèr and Dan Boneh. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=rJVorjCcKQ`.

[448] Florian Tramèr and Dan Boneh. Adversarial training and robustness for multiple perturbations. *arXiv preprint arXiv:1904.13000*, 2019.

[449] Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). *arXiv preprint arXiv:2011.11660*, 2020.

[450] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction APIs. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 601–618, 2016. URL `https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/tramer`.

[451] Florian Tramèr, Fan Zhang, Huang Lin, Jean-Pierre Hubaux, Ari Juels, and Elaine Shi. Sealed-glass proofs: Using transparent enclaves to prove and sell knowledge. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017*, pages 19–34, 2017.

[435]塞巴斯蒂安U斯蒂奇和赛普拉尼斯Karimireddy。错误反馈框架：具有延迟梯度和压缩通信的SGD的更高速率。arXiv：1909.05350，2019年。

[436]Lili Su和Nitin H.维迪亚容错多代理优化：最优迭代分布式算法。在PODC，2016年。

[437]Pramod Subramanyan、Rohit Sinha、伊利亚Lebedev、Srinivas Devadas和Sanjit A Seshia。正式的基金会

用于安全地远程执行飞地。2017年ACM SIGSAC计算机和通信安全会议论文集，第2435-2450页。ACM，2017年。

[438]Ziteng Sun，Peter Kairouz，Ananda Theertha Suresh，and H Brendan McMahan.你真的可以后门联邦学习吗？arXiv预印本arXiv：1911.07963，2019。

[439] support.google.您的聊天保持私密，而消息改进建议，2019。网址 support.google.com/messages/answer/9327902. 2019年8月恢复。

[440]作者：Ananda Theertha Suresh，Felix X. Yu，Sanjiv Kumar，and H Brendan McMahan.分布均值估计

有限的沟通。第34届机器学习国际会议论文集-第70卷，第3329-3337页。JMLR。org，2017.

[441]Christian Szegedy、Wojciech Zaremba、Ilya Sutskever、Joan Bruna、Dumitru Erhan、Ian Goodfellow和Rob费尔格斯。神经网络的有趣特性。ICLR，2013年。

[442]G 'abor J Sz' ekely，Maria L Rizzo，Nail K Bakirov等人。通过距离相关性测量和测试依赖性。The annals of statistics，35（6）：2769-2794，2007.

[443]唐翰林、连相如、明颜、张策、季柳。D2：分散数据的分散培训。2018年在ICML上发表。

[444]唐翰林、连相如、双秋、雷原、张策、张桐、季柳。DeepSqueeze：

具有双通道误差补偿压缩的并行随机梯度下降。arXiv预印本arXiv：1907.07346，2019。

[445]Jun Tang，Aleksandra Korolova，Xiaolong Bai，Xueqiang Wang，and Xiaofeng Wang.隐私权的丧失-苹果在MacOS 10.12上实现了差异隐私。CoRR，abs/1709.02753，2017。网址 http：//arxiv.org/abs/1709.02753。

[446]Om Thakkar，Galen Andrew，and H Brendan McMahan.具有自适应裁剪的差分私有学习。arXiv预印本arXiv：1905.03871，2019。

[447]Florian Tram`er和Dan Boneh。Slalom：在可信硬件中快速、可验证和私有地执行神经网络。在2019年国际学习代表会议上。URL https：//openreview.net/forum？id=rJVorjCcKQ。

[448]Florian Tram`er和Dan Boneh。对抗性训练和多扰动鲁棒性。arXiv预印本arXiv：1904.13000，2019。

[449]Florian Tram`er和Dan Boneh。不同的是，私人学习需要更好的特征（或更多的数据）。arXiv预印本arXiv：2011.11660，2020。

[450]Florian Tram `er，Fan Zhang，Ari Juels，Michael K.（作者）Reiter，and Thomas Ristenpart.托马斯·里德学习机器Stealing Machine

通过API预测模型。在第25届USENIX安全研讨会，USENIX安全16，奥斯汀，德克萨斯州，美国，2016年8月10日至12日。第601-618页，2016年。网址https://www.usenix.org/conference/ usenixsecurity 16/technical-sessions/presentation/tramer.

[451]Florian Tram`er，Fan Zhang，Huang Lin，Jean-Pierre Hubaux，Ari Juels，and Elaine Shi.密封玻璃校样：使用透明的飞地来证明和销售知识。2017年IEEE欧洲安全与隐私研讨会，EuroS&P 2017，法国巴黎，2017年4月26-28日，第19-34页，2017年。

[452] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick D. McDaniel. Ensemble adversarial training: Attacks and defenses. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[453] Florian Tramèr, Jens Behrmann, Nicholas Carlini, Nicolas Papernot, and Jörn-Henrik Jacobsen. Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9561–9571. PMLR, 2020. URL http://proceedings.mlr.press/v119/tramer20a.html.

[454] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *Advances in Neural Information Processing Systems*, pages 8000–8010, 2018.

[455] Jonathan Ullman. Tight lower bounds for locally differentially private selection. Technical Report abs/1802.02638, arXiv, 2018. URL http://arxiv.org/abs/1802.02638.

[456] The Google-Landmark v2 Authors. Google landmark dataset v2, 2019. URL https://github.com/cvdfoundation/google-landmark.

[457] Jaideep Vaidya, Hwanjo Yu, and Xiaoqian Jiang. Privacy-preserving SVM classification. *Knowl. Inf. Syst.*, 14 (2), January 2008.

[458] Jo Van Bulck, Marina Minkin, Ofir Weisse, Daniel Genkin, Baris Kasikci, Frank Piessens, Mark Silberstein, Thomas F Wenisch, Yuval Yarom, and Raoul Strackx. Foreshadow: Extracting the keys to the intel {SGX} kingdom with transient out-of-order execution. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 991–1008, 2018.

[459] Paul Vanhaesebrouck, Aurélien Bellet, and Marc Tommasi. Decentralized collaborative learning of personalized models over networks. In *AISTATS*, 2017.

[460] Praneeth Vepakomma, Otkrist Gupta, Tristan Swedish, and Ramesh Raskar. Split learning for health: Distributed deep learning without sharing raw patient data. *arXiv preprint arXiv:1812.00564*, 2018.

[461] Praneeth Vepakomma, Chetan Tonde, Ahmed Elgammal, et al. Supervised dimensionality reduction via distance correlation maximization. *Electronic Journal of Statistics*, 12(1):960–984, 2018.

[462] Praneeth Vepakomma, Otkrist Singh, Abhishek Gupta, and Ramesh Raskar. Nopeek: Information leakage reduction to share activations in distributed deep learning. *arXiv preprint arXiv:2008.09161*, 2020.

[463] Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. PowerSGD: Practical low-rank gradient compression for distributed optimization. In *NeurIPS 2019 - Advances in Neural Information Processing Systems 32*, 2019.

[464] Riad S. Wahby, Ioanna Tzialla, Abhi Shelat, Justin Thaler, and Michael Walfish. Doubly-efficient zksnarks without trusted setup. In *2018 IEEE Symposium on Security and Privacy, SP 2018, Proceedings, 21-23 May 2018, San Francisco, California, USA*, 2018.

[465] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy*. IEEE, 2019.

[466] Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. Attack of the tails: Yes, you really can backdoor federated learning, 2020.

[467] Jianyu Wang and Gauri Joshi. Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms. *preprint*, August 2018. URL https://arxiv.org/abs/1808.07576.

[452]Florian Tram`er、Alexey Kurakin、Nicolas Papernot、Ian J. Goodfellow、Dan Boneh和帕特里克D.麦克丹尼尔对抗训练：攻击和防御。在第六届国际会议上学习代表，ICLR 2018，温哥华，不列颠哥伦比亚省，加拿大，2018年4月30日至5月3日，会议跟踪程序，2018年。

[453]Florian Tram `er，Jens Behrmann，Nicholas Carlini，Nicolas Papernot，和J ¬ orn—Henrik Jacobsen.他是Florian Tram`er，Jens Behrmann，Nicholas Carlini，Nicolas Papernot，和Jorn-Henrik Jacobsen.基本面在不变性和对抗扰动的敏感性之间进行权衡。第37届国际机器学习会议论文集，ICML 2020，2020年7月13日至18日，虚拟事件，机器学习研究论文集第119卷，第9561-9571页。PMLR，2020年。网址http://proceedings.mlr。press/v119/tramer20a.html.

[454]布兰登·陈杰瑞·李和亚历山大·马德里。后门攻击中的光谱特征。神经信息处理系统的进展，第8000-8010页，2018年。

[455]乔纳森·厄尔曼。局部差异私有选择的紧下界。技术报告abs/1802.02638，arXiv，2018。网址http://arxiv.org/abs/1802.02638。

[456]Google-Landmark v2作者Google landmark dataset v2，2019。网址https://github.com/cvdfoundation/google-landmark。

[457]Jaideep Vaidya，Hwanjo Yu，and Xiaoqian Jiang.隐私保护SVM分类。知识信息系统，14（2），2008年1月。

[458]Jo Van Bulck，Marina Minkin，Ofir Weisse，Daniel Genkin，Baris Kasikci，Frank Piessens，Mark Silberstein，

托马斯F Wenisch，Yuval Yarom，and拉乌尔Strackx.预示：通过短暂的无序执行提取英特尔{SGX}王国的密钥。第27届{USENIX}安全研讨会（{USENIX} Security 18），第991-1008页，2018年。

[459]Paul Vanhaesebrouck，Aur 'elien Bellet，and Marc Tommasi.网络上个性化模型的分散协作学习。在AISTATS，2017年。

[460]Praneeth Vepakomma，Otkrist Gupta，Tristan Swedish，and Ramesh Raskar.健康分割学习：分布式深度学习，无需共享原始患者数据。arXiv预印本arXiv：1812.00564，2018。

[461]Praneeth Vepakomma，Chetan Tonde，Ahmed Elgammal等人。通过距离相关最大化进行监督降维。电子统计杂志，12（1）：960-984，2018。

[462]Praneeth Vepakomma，Otkrist Singh，Abhishek Gupta，and Ramesh Raskar. Nopeek：分布式深度学习中减少信息泄漏以共享激活。arXiv预印本arXiv：2008.09161，2020。

[463]Thijs Vogels，Sai Praneeth Karimireddy，and Martin Jaggi. PowerSGD：用于分布式优化的实用低秩梯度压缩。在NeurIPS 2019 -神经信息处理系统的进展32，2019。

[464]Riad S. Wahby，Ioanna Tzialla，Abhi Shelat，Justin Thaler，and Michael Walfish.双效率zksnarks

没有可信的设置。在2018年IEEE安全和隐私研讨会上，SP 2018，会议记录，2018年5月21日至23日，美国加州州弗朗西斯科，2018年。

[465]Bolun Wang，Yuanshun Yao，Shawn Shan，Huiying Li，Bingyang Viswanath，Haitao Zheng，and Ben Y Zhao. Neural Cleanse：识别和减轻神经网络中的后门攻击。2019年IEEE安全与隐私研讨会。IEEE，2019年。

[466]Hongyi Wang、Kartik Sreenivasan、Shashank Rajput、Harit Vishwakarma、Saurabh Agarwal、Jy yong Sohn、李康旭和迪米特里斯·帕帕里奥普洛斯。尾巴的攻击：是的，你真的可以在2020年后门联邦学习。

[467]Jianyu Wang和Gauri Joshi。Cooperative SGD：一个设计和分析的统一框架通信高效的SGD算法。预印本，2018年8月。网址https://arxiv.org/abs/ 1808.07576。

[468] Jianyu Wang and Gauri Joshi. Adaptive Communication Strategies for Best Error-Runtime Trade-offs in Communication-Efficient Distributed SGD. In *Proceedings of the SysML Conference*, April 2019. URL https://arxiv.org/abs/1810.08313.

[469] Jianyu Wang, Anit Sahu, Gauri Joshi, and Soummya Kar. MATCHA: Speeding Up Decentralized SGD via Matching Decomposition Sampling. *preprint*, May 2019. URL https://arxiv.org/abs/1905.09435.

[470] Jianyu Wang, Vinayak Tantia, Nicolas Ballas, and Michael Rabbat. SlowMo: Improving communication-efficient distributed SGD with slow momentum. *arXiv preprint arXiv:1910.00643*, 2019.

[471] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

[472] Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252*, 2019.

[473] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018.

[474] Yu-Xiang Wang, Borja Balle, and Shiva Kasiviswanathan. Subsampled R\'enyi differential privacy and analytical moments accountant. *arXiv preprint arXiv:1808.00087*, 2018.

[475] Stanley L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965.

[476] WeBank. WeBank and Swiss re signed cooperation MOU, 2019. URL https://finance.yahoo.com/news/webank-swiss-signed-cooperation-mou-112300218.html. Retrieved Aug 2019.

[477] Eric Wong, Frank R Schmidt, and J Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations. *ICML*, 2019.

[478] Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32, 2014.

[479] D. Woodruff and S. Yekhanin. A geometric approach to information-theoretic private information retrieval. In *20th Annual IEEE Conference on Computational Complexity (CCC'05)*, pages 275–284, June 2005. doi: 10.1109/CCC.2005.2.

[480] Blake Woodworth, Jialei Wang, H. Brendan McMahan, and Nathan Srebro. Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. In *Advances in Neural Information Processing Systems (NIPS)*, 2018. URL https://arxiv.org/abs/1805.10222.

[481] Blake Woodworth, Kumar Kshitij Patel, Sebastian U Stich, Zhen Dai, Brian Bullins, H Brendan McMahan, Ohad Shamir, and Nathan Srebro. Is local sgd better than minibatch sgd? *arXiv preprint arXiv:2002.07839*, 2020.

[482] Xiang Wu, Ruiqi Guo, Ananda Theertha Suresh, Sanjiv Kumar, Daniel N Holtmann-Rice, David Simcha, and Felix X. Yu. Multiscale quantization for fast similarity search. In *Advances in Neural Information Processing Systems*, pages 5745–5755, 2017.

[483] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. *CVPR*, 2019.

[484] Cong Xie. Zeno++: robust asynchronous SGD with arbitrary number of Byzantine workers. *arXiv preprint arXiv:1903.07020*, 2019.

[468]Jianyu Wang和Gauri Joshi。最佳容错折衷的自适应通信策略
通信高效的分布式SGD。在SysML会议上，2019年4月。网址https://arxiv.org/abs/1810.08313。

[469]Jianyu Wang，Anit Sahu，Gauri Joshi，and Soummya Kar. MATCHA：通过匹配分解采样加速分散式SGD。预印本，2019年5月。网址https://arxiv.org/abs/1905。
09435.

[470]Jianyu Wang，Vinayak Tantia，Nicolas Ballas，and Michael Rabbat. SlowMo：提高通信效率的分布式SGD，速度缓慢。arXiv预印本arXiv：1910.00643，2019。

[471]Jianyu Wang，Qinghua Liu，Hao Liang，Gauri Joshi，and H Vincent Poor.解决客观不一致问题

异构联邦优化问题。神经信息处理系统的进展，33，2020。

[472]Kangkang Wang、Rajiv Mathews、Chlo’e Kiddon、Hubert Eichner、Franc’oise Beaufays和Daniel Ramage。
设备上个性化的联合评估。arXiv预印本arXiv：1910.10252，2019。

[473]Tongzhou Wang，Jun-Yan Zhu，Antonio Torralba，and Alexei A Efros.数据集蒸馏。arXiv预印本arXiv：1811.10959，2018。

[474]王宇翔、博尔哈巴莱和湿婆·卡西维斯瓦纳坦。二次抽样R\'enyi差分隐私和分析时刻会计师。arXiv预印本arXiv：1808.00087，2018。

[475]斯坦利湖警告者随机回答：一种消除回避回答偏差的调查技术。Journal of the American Statistical Association，60（309）：63-69，1965.

[476]微众银行微众银行与瑞士再签2019年合作备忘录网址：https://finance.yahoo.com/ news/webank-swiss-signed-cooperation-mou-112300218.html。2019年8月恢复。

[477]Eric Wong，Frank R施密特，and J Zico Kolter.通过投影sinkhorn迭代的Wasserstein对抗示例。ICML，2019年。

[478]Gavin Wood et al. Ethereum：A secure decentralised generalised transaction ledger. Ethereum project yellow paper，151（2014）：1-32，2014.

[479]D. Woodruff和S.叶哈宁信息论私人信息检索的几何方法。

第20届IEEE计算复杂性年会（CCC'05），第275-284页，2005年6月。doi：10.1109/CCC.2005.2。

[480]放大图片作者：Blake Woodworth，王佳磊，H.布兰登·麦克马汉和内森·斯雷布罗图预言模型，下界，
和并行随机优化的间隙。在神经信息处理系统（NIPS）的进展，2018年。网址
https://arxiv.org/abs/1805.10222。

[481]Blake Woodworth，Kumar Kshitij Patel，塞巴斯蒂安U斯蒂奇，戴震，Brian Bullins，H Brendan McMahan，
奥哈德·沙米尔和内森·斯雷布罗本地sgd比小批量sgd好吗？arXiv预印本arXiv：2002.07839，2020。

[482]Xiang Wu，Ruiqi Guo，Ananda Theertha Suresh，Sanjiv Kumar，丹尼尔N Holtmann-Rice，大卫辛查，和
菲利克斯十世Yu.快速相似性搜索的多尺度量化。神经信息处理系统的进展，第5745-5755页，2017年。

[483]谢慈航，吴宇新，劳伦斯货车德尔马滕，艾伦Yuille，何开明。用于提高对抗鲁棒性的特征去噪。CVPR，2019年。

[484]谢丛。Zeno++：健壮的异步SGD，具有任意数量的拜占庭工人。arXiv预印本arXiv：1903.07020，2019。

[485] Cong Xie, Oluwasanmi Koyejo, Indranil Gupta, and Haibin Lin. Local adaalter: Communication-efficient stochastic gradient descent with adaptive learning rates. *arXiv preprint arXiv:1911.09030*, 2019.

[486] Cong Xie, Sanmi Koyejo, and Indranil Gupta. Practical distributed learning: Secure machine learning with communication-efficient local updates. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, 2019.

[487] Cong Xie, Sanmi Koyejo, and Indranil Gupta. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance. In *International Conference on Machine Learning*, pages 6893–6901, 2019.

[488] Sirui Xie, Hehui Zheng, Chunxiao Liu, and Liang Lin. SNAS: stochastic neural architecture search. *arXiv preprint arXiv:1812.09926*, 2018.

[489] Tiancheng Xie, Jiaheng Zhang, Yupeng Zhang, Charalampos Papamanthou, and Dawn Song. Libra: Succinct zero-knowledge proofs with optimal prover computation. In *CRYPTO (3)*, volume 11694 of *Lecture Notes in Computer Science*, pages 733–764. Springer, 2019.

[490] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *CoRR*, abs/1902.04885, 2019. URL `http://arxiv.org/abs/1902.04885`.

[491] Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, and Françoise Beaufays. Applied federated learning: Improving Google keyboard query suggestions. *arXiv preprint 1812.02903*, 2018.

[492] Andrew C Yao. Protocols for secure computations. In *Symposium on Foundations of Computer Science*, 1982.

[493] Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167. IEEE Computer Society, 1986.

[494] Fangwei Ye, Carolina Naim, and Salim El Rouayheb. Preserving ON-OFF privacy for past and future requests. In *2019 IEEE Information Theory Workshop (ITW)*, August 2019.

[495] Min Ye and Alexander Barg. Optimal schemes for discrete distribution estimation under locally differential privacy. *IEEE Transactions on Information Theory*, 2018.

[496] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 268–282. IEEE, 2018.

[497] Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *ICML*, 2019.

[498] Chen Yu, Hanlin Tang, Cedric Renggli, Simon Kassing, Ankit Singla, Dan Alistarh, Ce Zhang, and Ji Liu. Distributed learning over unreliable networks. *arXiv preprint arXiv:1810.07766*, 2018.

[499] Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. A sustainable incentive scheme for federated learning. *IEEE Intelligent Systems*, 35(4):58–69, 2020.

[500] Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted SGD for non-convex optimization with faster convergence and less communication. *arXiv preprint arXiv:1807.06629*, 2018.

[501] Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. *arXiv preprint arXiv:1905.03817*, 2019.

[502] Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. Salvaging federated learning by local adaptation. *arXiv preprint arXiv:2002.04758*, 2020.

[503] Muhammad Bila Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 2017.

[485]Cong Xie，Oluwasanmi Koyejo，Indranil Gupta，and Haibin Lin. Local adaalter：具有自适应学习率的通信高效随机梯度下降。arXiv预印本arXiv：1911.09030，2019。

[486]Cong Xie，Sanmi Koyejo，and Indranil Gupta.实用的分布式学习：使用

高效的本地更新。在欧洲机器学习和数据库知识发现的原则和实践会议（ECML PKDD），2019年。

[487]Cong Xie，Sanmi Koyejo，and Indranil Gupta. Zeno：分布式随机梯度下降与基于怀疑的容错。在国际机器学习会议上，第6893-6901页，2019年。

[488]谢思睿，郑和辉，刘春晓，林亮。SNAS：随机神经结构搜索。arXiv预印本arXiv：1812.09926，2018。

[489]谢天成，张嘉恒，张玉鹏，Charalampos Papamanthou和Dawn Song。天秤座：简洁

零知识证明与最佳证明者计算。在《计算机科学讲义》第11694卷第3卷第733-764页中。施普林格，2019年。

[490]杨强，刘扬，陈天健，童永新。联邦机器学习：概念与应用CoRR，abs/1902.04885，2019。网址http://arxiv.org/abs/1902.04885。

[491]Timothy Yang，Galen Andrew，Hubert Eichner，Haicheng Sun，Wei Li，Nicholas Kong，Daniel

弗朗克·布瓦斯·博费斯。应用联邦学习：改进Google键盘查询建议。arXiv预印本1812.02903，2018。

[492]安德鲁·C·姚安全计算协议。计算机科学基础研讨会，1982年。

[493]姚期智如何生成和交换秘密（扩展抽象）。见FOCS，第162-167页。

IEEE计算机学会，1986年。

[494]Fangwei Ye，卡罗莱纳Naim和Salim El Rouayheb。为过去和未来的请求保留ON-OFF隐私。在2019年IEEE信息理论研讨会（ITW），2019年8月。

[495]闵叶和亚历山大巴格。局部差分隐私下离散分布估计的最优方案。IEEE Transactions on Information Theory，2018。

[496]Samuel Yeom，Irene Giacomelli，Matt Fredrikson，and Somesh Jha.机器学习中的隐私风险：分析

与过度拟合的联系2018年IEEE第31届计算机安全基础研讨会（CSF），第268- 282页。IEEE，2018年。

[497]Dong Yin，Yudong Chen，Kannan Ramchandran，and Peter Bartlett.拜占庭鲁棒分布式学习：迈向最佳统计率。2019年，在ICML中。

[498]Chen Yu，Hanlin Tang，Cedric Renggli，Simon Kasing，Ankit Singla，Dan Alistarh，Ce Zhang，and Ji

不可靠网络上的分布式学习arXiv预印本arXiv：1810.07766，2018。

[499]Han Yu，Zerei Liu，Yang Liu，Tianjian Chen，Mingshu Cong，Xi Weng，杜西特Niyato，和Qiang Yang。联邦学习的可持续激励计划。IEEE智能系统，35（4）：58-69，2020。

[500]宇昊，森阳，祝圣火。用于非凸优化的并行重启SGD，具有更快的收敛速度和更少的通信。arXiv预印本arXiv：1807.06629，2018。

[501]郝宇、容瑾、森阳。分布式非凸优化的通信有效动量SGD的线性加速比分析。arXiv预印本arXiv：1905.03817，2019。

[502]陶宇、尤金·巴格达萨良和维塔利·什马蒂科夫。通过局部适应挽救联邦学习。arXiv预印本arXiv：2002.04758，2020。

[503]Muhammad Bila Zafar，伊莎贝尔瓦莱拉，Manuel Gomez Rodriguez，和Krishna P. Gummadi。公平的骗局-

straints：公平分类的机制。2017年第20届人工智能与统计国际会议论文集。

[504] Valentina Zantedeschi, Aurélien Bellet, and Marc Tommasi. Fully Decentralized Joint Learning of Personalized Models and Collaboration Graphs. Technical report, arXiv:1901.08460, 2019.

[505] Sixin Zhang, Anna E Choromanska, and Yann LeCun. Deep learning with elastic averaging SGD. In *Advances in Neural Information Processing Systems*, pages 685–693, 2015.

[506] Yu Zhang and Qiang Yang. A survey on multi-task learning. *CoRR*, abs/1707.08114, 2017. URL `http://arxiv.org/abs/1707.08114`.

[507] Yuchen Zhang, John Duchi, Micheal I. Jordan, and Martin J. Wainwright. Information-theoretic lower bounds for distributed statistical estimation with communication constraints. In *Advances in Neural Information Processing Systems*, pages 2328–2336, 2013.

[508] Yawei Zhao, Chen Yu, Peilin Zhao, and Ji Liu. Decentralized online learning: Take benefits from others' data without sharing your own to track global trend. *arXiv preprint arXiv:1901.10593*, 2019.

[509] Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression. *arXiv preprint arXiv:1710.01878*, 2017.

[510] Wennan Zhu, Peter Kairouz, Haicheng Sun, Brendan McMahan, and Wei Li. Federated heavy hitters discovery with differential privacy. *arXiv preprint arXiv:1902.08534*, 2019.

[511] Xiaojin Zhu. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

# A Software and Datasets for Federated Learning

**Software for simulation** Simulations of federated learning require dealing with multiple issues that do not arise in datacenter ML research, for example, efficiently processing partitioned datasets, with computations running on different simulated devices, each with a variable amount of data. FL research also requires different metrics such as the number of bytes upload or downloaded by device, as well as the ability to simulate issues like time-varying arrival of different clients or client drop-out that is potentially correlated with the nature of the local dataset. With this in mind, the development of open software frameworks for federated learning research (simulation) has the potential to greatly accelerate research progress. Several platforms are available or in development, including [404]:

- TensorFlow Federated [38] specifically targets research use cases, providing large-scale simulation capabilities as well as flexible orchestration for the control of sampling.
- FedML [229] is a research-oriented library. It supports three platforms: on-device training for IoT and mobile devices, distributed computing, and single-machine simulation. For research diversity, FedML also supports various algorithms (e.g., decentralized learning, vertical FL, and split learning), models, and datasets.
- PySyft [399] is a Python library for secure, private Deep Learning. PySyft decouples private data from model training, using federated learning, differential privacy, and multi-party computation (MPC) within PyTorch.
- Leaf [35] provides multiple datasets (see below), as well as simulation and evaluation capabilities.
- Sherpa.ai Federated Learning and Differential Privacy Framework [397] is an open source federated learning and differential privacy framework which provides methodologies, pipelines, and evaluation techniques for federated learning.

[504]Valentina Zantedeschi，Aur 'elien Bellet，and Marc Tommasi.完全分散的个性化模型和协作图的联合学习。技术报告，arXiv：1901.08460，2019。

[505]Sixin Zhang，安娜E Choromanska，和Yann LeCun.使用弹性平均SGD的深度学习。神经信息处理系统进展，第685-693页，2015年。

[506]张玉和杨强。多任务学习研究综述。CoRR，abs/1707.08114，2017。网址http：//arxiv.org/abs/1707.08114。

[507]张宇晨，John Duchi，Micheal I.作者声明：Martin J.温赖特.信息论下界

用于具有通信约束的分布式统计估计。神经信息处理系统的进展，第2328-2336页，2013年。

[508]赵雅薇，陈宇，赵沛林，刘继。分散的在线学习：从他人的数据中获益，而不分享自己的数据，以跟踪全球趋势。arXiv预印本arXiv：1901.10593，2019。

[509]Michael Zhu和Suyog Gupta。修剪，还是不修剪：探索模型压缩的修剪功效。arXiv预印本arXiv：1710.01878，2017。

[510]Wennan Zhu，Peter Kairouz，Haicheng Sun，Brendan McMahan，and Wei Li.具有差异隐私的联邦重量级人物发现。arXiv预印本arXiv：1902.08534，2019。

[511]朱小锦。机器教学：机器学习的逆问题和优化教育的方法。2015年第29届AAAI人工智能会议。

## 联邦学习的软件和数据集

联邦学习的模拟需要处理数据中心ML研究中不会出现的多个问题，例如，有效地处理分区数据集，在不同的模拟设备上运行计算，每个设备都有不同的数据量。FL研究还需要不同的指标，例如设备上传或下载的字节数，以及模拟不同客户端随时间变化的到达或客户端退出等问题的能力，这些问题可能与本地数据集的性质相关。考虑到这一点，开发用于联邦学习研究（模拟）的开放软件框架有可能大大加快研究进展。有几个平台可用或正在开发中，包括[404]：

TensorFlow Federated [38]专门针对研究用例，提供大规模仿真功能以及灵活的编排以控制采样。

FedML [229]是一个面向研究的库。它支持三个平台：物联网和移动的设备的设备上培训、分布式计算和单机仿真。为了研究多样性，FedML还支持各种算法（例如，分散学习、垂直FL和分裂学习）、模型和数据集。

PySyft [399]是一个用于安全、私有深度学习的Python库。PySyft使用PyTorch中的联邦学习、差分隐私和多方计算（MPC），从模型训练中提取私有数据。

Leaf [35]提供多个数据集（见下文），以及模拟和评估功能。

·Sherpa.ai Federated Learning and Differential Privacy Framework是一个开源的联邦学习和差异隐私框架，为联邦学习提供方法，管道和评估技术。

- PyVertical [32] is a project focusing on federated learning with data partitioned by features (also referred to as vertical partitioning) in the cross-silo setting; see Section 2.2.

**Production-oriented software** In addition to the above simulation platforms, several production-oriented federated learning platforms are being developed:

- FATE (Federated AI Technology Enabler) [33] is an open-source project intended to provide a secure computing framework to support the federated AI ecosystem.
- PaddleFL [36] is an open source federated learning framework based on PaddlePaddle [37]. In PaddleFL, several federated learning strategies and training strategies are provided with application demonstrations.
- Clara Training Framework [125] includes the support of cross-silo federated learning based on a server-client approach with data privacy protection.
- IBM Federated Learning [321] is a Python-based federated learning framework for enterprise environments, which provides a basic fabric for adding advanced features.
- Flower framework [66] supports implementation and experimentation of federated learning algorithms on mobile and embedded devices with a real-world system conditions simulation.
- Fedlearner [34] is an open source federated learning framework that enables joint modeling of data distributed between institutions.

Such production-oriented federated learning platforms must address problems that do not exist in simulation such as authentication, communication protocols, encryption and deployment to physical devices or silos. Note that while TensorFlow Federated is listed under "Software for simulation", its design includes abstractions for aggregation and broadcast, and serialization of all TensorFlow computations for execution in non-Python environments, making it suitable for use as a component in a production system.

**Datasets** Federated learning is adopted when the data is decentralized and typically unbalanced (different clients have different numbers of examples) and not identically distributed (each client's data is drawn from a different distribution). The open source package TensorFlow Federated [38] supports loading decentralized dataset in a simulated environment with each client id corresponding to a TensorFlow Dataset Object. These datasets can easily be converted to numpy arrays for use in other frameworks.[11] At the time of writing, three datasets are supported and we recommend researchers to benchmark on them.

- *EMNIST* dataset [126] consists of 671,585 images of digits and upper and lower case English characters (62 classes). The federated version splits the dataset into 3,400 unbalanced clients indexed by the original writer of the digits/characters. The non-IID distribution comes from the unique writing style of each person.
- *Stackoverflow*[12] dataset consists of question and answer from Stack Overflow with metadata like timestamps, scores, etc. The training dataset has more than 342,477 unique users with 135,818,730 examples. Note that the timestamp information can be helpful to simulate the pattern of incoming data.

[11] `https://www.tensorflow.org/datasets/api_docs/python/tfds/as_numpy`.
[12] `https://www.kaggle.com/stackoverflow/stackoverflow`

PyVertical [32]是一个专注于联邦学习的项目，在跨竖井设置中使用按功能划分的数据（也称为垂直分区）;参见第2.2节。

面向生产的软件除了上述模拟平台外，还开发了几个面向生产的联合学习平台：

FATE（Federated AI Technology Enabler）[33]是一个开源项目，旨在提供一个安全的计算框架来支持联邦AI生态系统。

PaddleFL [36]是基于PaddlePaddle [37]的开源联邦学习框架。在PaddleFL中，提供了几种联邦学习策略和训练策略，并进行了应用演示。

Clara培训框架[125]包括基于服务器-客户端方法的跨筒仓联合学习支持，并提供数据隐私保护。

IBM Federated Learning是一个基于Python的企业环境联邦学习框架，它为添加高级功能提供了基本结构。

Flower框架[66]支持联邦学习算法在移动的和嵌入式设备上的实现和实验，并具有真实世界的系统条件模拟。

Fedlearner [34]是一个开源的联邦学习框架，可以对分布在机构之间的数据进行联合建模。

这种面向生产的联合学习平台必须解决模拟中不存在的问题，例如身份验证、通信协议、加密和部署到物理设备或筒仓。请注意，虽然TensorFlow Federated被列在“模拟软件”下，但它的设计包括聚合和广播的抽象，以及在非Python环境中执行的所有TensorFlow计算的序列化，使其适合用作生产系统中的组件。

当数据是分散的，通常是不平衡的（不同的客户端有不同数量的示例），并且分布不均匀（每个客户端的数据来自不同的分布）时，采用联合学习。开源软件包TensorFlow Federated [38]支持在模拟环境中加载分散的数据集，每个客户端ID对应一个TensorFlow数据集对象。这些数据集可以很容易地转换为numpy数组，以便在其他框架中使用。在撰写本文时，支持三个数据集，我们建议研究人员对它们进行基准测试。

- EMNIST数据集[126]由671，585个数字和大小写英文字符（62类）的图像组成。联邦版本将数据集拆分为3，400个由数字/字符的原始作者索引的不平衡客户端。非IID分布来自每个人独特的写作风格。

- Stackoverflowdataset由Stack Overflow的问题和答案组成，包括时间戳，分数等元数据，训练数据集拥有超过342，477个独立用户，135，818，730个示例。请注意，时间戳信息有助于模拟传入数据的模式。

[11] https://www.tensorflow.org/datasets/api_docs/python/tfds/as_numpy.
[12] https://www.kaggle.com/stackoverflow/stackoverflow

- *Shakespeare* is a language modeling dataset derived from *The Complete Works of William Shakespeare*. It consists of 715 characters whose contiguous lines are examples in the client dataset. The train set has 16,068 examples and test set has 2,356 examples.

The preprocessing for *EMNIST* and *Shakespeare* are provided by the Leaf project [96], which also provides federated versions of the sentiment140 and celebA datasets. These datasets have enough clients that they can be used to simulate cross-device FL scenarios, but for questions where scale is particularly important, they may be too small. In this respect *Stackoverflow* provides the most realistic example of a cross-device FL problem.

**Cross-silo datasets** One example is the iNaturalist dataset[13] which consists of large numbers of observations of various organisms all over the world. One can partition it by the geolocation or the author of an observation. If we partition it by the group an organism belongs to, like kingdom, phylum, etc., then the clients have totally different labels and biological closeness between two clients is already known. This makes it a very suitable dataset to study federated transfer learning and multi-task learning in cross-silo settings.

Another example is the Google-Landmark-v2 [456] that includes over 5 million images of more than 200 thousand different types of landmark. Similar to the iNaturalist dataset, one can split the dataset by authors, but due to the difference in scale with iNaturalist dataset, Google Landmark Dataset provides much more diversity and creates even greater challenges to large-scale federated learning.

Luo et al. [322] has recently published a federated dataset for computer vision. The dataset contains more than 900 annotated street images generated from 26 street cameras and 7 object categories annotated with detailed bounding box. Due to the relatively small number of examples in the dataset, it may not adequately reflect a challenging realistic scenario.

**The need for more datasets** Developing new federated learning datasets that are representative of real-world problems is an important question for the community to address. Platforms like TensorFlow Federated [38] welcome the contribution of new datasets and may be able to provide hosting support.

While completely new datasets are always interesting, in many cases it is possible to partition existing open datasets, treating each split as a client. Different partitioning strategies may be appropriate for different research questions, but often unbalanced and non-IID partitions will be most relevant. It is also interesting to maintain as much additional meta information (timestamp, geolocation, etc.) as possible.

In particular, there is a need for feature-partitioned datasets, as will be discussed in Section 2.2. For example, a patient may go to one medical institute for a pathology test and go to another for radiology picture archiving, in which case the features of one sample are partitioned over two institutes regulated by HIPAA. [24].

[13] `https://www.inaturalist.org/`

莎士比亚是一个语言建模数据集，来源于威廉·莎士比亚全集。它由715个字符组成，其连续行是客户端数据集中的示例。训练集有16,068个例子，测试集有2,356个例子。

EMNIST和莎士比亚的预处理由Leaf项目提供[96]，该项目还提供sentiment 140和celebA数据集的联邦版本。这些数据集有足够的客户端，可用于模拟跨设备FL场景，但对于规模特别重要的问题，它们可能太小。在这方面，Stackoverflow提供了跨设备FL问题的最现实的例子。

跨筒仓数据集一个例子是iNaturalist数据集，它由世界各地各种生物的大量观察组成。人们可以通过地理位置或观察的作者来划分它。如果我们按照生物体所属的组来划分，比如界、门等等，则客户端具有完全不同的标签，并且两个客户端之间的生物接近性是已知的。这使得它成为一个非常适合研究跨筒仓设置中的联邦迁移学习和多任务学习的数据集。

另一个例子是Google-Landmark-v2 [456]，其中包括超过20万种不同类型的地标的超过500万张图像。与iNaturalist数据集类似，可以按作者划分数据集，但由于与iNaturalist数据集的规模差异，Google Landmark数据集提供了更多的多样性，并对大规模联邦学习带来了更大的挑战。

Luo等人[322]最近发表了一个用于计算机视觉的联合数据集。该数据集包含26个街道摄像机生成的900多个带注释的街道图像和7个带有详细边界框注释的对象类别。由于数据集中的示例数量相对较少，它可能无法充分反映具有挑战性的现实场景。

开发新的联邦学习数据集，代表现实世界的问题是社区要解决的一个重要问题。像TensorFlow Federated [38]这样的平台欢迎新数据集的贡献，并可能提供托管支持。

虽然全新的数据集总是令人感兴趣，但在许多情况下，可以对现有的开放数据集进行分区，将每个拆分视为客户端。不同的分区策略可能适用于不同的研究问题，但通常不平衡和非IID分区将是最相关的。维护尽可能多的附加Meta信息（时间戳、地理位置等）也很有趣。越好.

特别是，需要特征划分的数据集，这将在2.2节中讨论。例如，患者可能会去一个医疗机构进行病理学检查，然后去另一个医疗机构进行放射学图像存档，在这种情况下，一个样本的特征会被分配到由HIPAA监管的两个机构。[24]第10段。

13https：www.inaturalist.org/