

联合多任务学习



          维吉尼亚·史密斯·赵凯蒋马齐亚尔·桑贾比亚迈特·塔尔沃卡尔∗∗

              斯坦福大学南加州大学

smithv@stanford.educhaokaic@usc.edumaziarsanjabi@gmail.comtalwalkar@cmu.edu

# 摘要

在分布式设备网络上训练机器学习模型时，联合学习提出了新的统计和系统挑战。在这项工作中，我们证明了多任务学习自然适合处理这种情况下的统计挑战，并提出了一种新的系统感知优化方法MOCHA，它对实际系统问题具有鲁棒性。我们的方法和理论首次考虑了分布式多任务学习的高通信成本、离散性和容错性等问题。与联邦环境中的备选方案相比，所产生的方法实现了显著的加速，正如我们在真实世界的联邦数据集上的模拟所示。

# 1        介绍

移动电话、可穿戴设备和智能家居只是现代分布式网络中每天产生大量数据的一小部分。由于这些网络中设备的存储量和计算能力的不断增长，在本地存储数据并将更多的网络计算推向边缘变得越来越有吸引力。联合学习的新生领域探索直接在设备上训练统计模型[37]。潜在应用的例子包括：学习情绪、语义位置或移动电话用户的活动；预测可穿戴设备的低血糖或心脏病发作风险等健康事件；或检测智能家居中的盗窃行为[3,39,42]。在[25，36，26]之后，我们总结了联合学习的独特挑战。

1.   统计挑战：联合学习的目标是使模型适合于由分布式节点生成的数据{X1，…，Xm}。每个节点∈[m]通过网络以非IID方式收集数据，每个节点上的数据由不同的分布∼Pt生成。每个节点上的数据点的数量也可能显著变化，并且可能存在一个底层结构，该结构捕获节点之间的关系及其关联的分布。*米t***十***t新台币*

2.   系统挑战：在网络中通常有大量的节点，通信通常是一个重要的瓶颈。此外，由于硬件（CPU、内存）、网络连接（3G、4G、WiFi）和电源（电池电量）的变化，每个节点的存储、计算和通信容量可能有所不同。这些系统挑战，再加上不平衡的数据和统计上的异构性，使得诸如离散器和容错等问题比典型的数据中心环境更加普遍。*米*

在这项工作中，我们提出了一种与先前关于联合学习的工作有很大不同的建模方法，目前的目标是在整个网络中训练一个单一的全局模型[25，36，26]。相反，我们通过学习每个节点{}的独立模型来解决联邦设置中的统计难题。这可以通过一个多任务学习（MTL）框架自然地得到，其中的目标是考虑同时拟合单独但相关的模型[14,2,57,28]。不幸的是，目前的多任务学习方法不适合处理联合学习中出现的系统挑战，包括高通信成本、离散和容错。因此，应对这些挑战是我们工作的一个关键组成部分。**栈单**1*,...,米*



∗

作者的贡献是平等的。

第31届神经信息处理系统会议（NIPS 2017），加利福尼亚州长滩，美国。

1.1捐款

我们作出以下贡献。首先，我们展示了MTL是在联邦环境下处理统计挑战的自然选择。其次，我们发展了一种新的方法，MOCHA，来解决一个一般的MTL问题。我们的方法推广了分布式优化方法COCOA[22，31]，以解决与网络规模和节点异构性相关的系统挑战。第三，我们为摩卡提供融合保证，仔细考虑这些独特的系统挑战，并提供对实际性能的洞察。最后，我们用一套新的联邦数据集基准测试来展示MOCHA优越的经验性能。

# 2        相关工作

超越数据中心学习。跨分布式低功耗节点计算类SQL查询是一个长达数十年的研究领域，在传感器网络中的查询处理、边缘计算和雾计算[32,12,33,8,18,15]。最近的工作也考虑集中训练机器学习模型，但在本地服务和存储它们，例如，这是移动用户建模和个性化的常用方法[27，43，44]。然而，随着分布式网络中节点计算能力的增长，可以在本地进行更多的工作，这导致了最近对联合学习的兴趣。[1]与我们提出的方法相比，现有的联合学习方法[25，36，26，37]旨在跨数据学习单个全局模型。[2]这一限制它们处理非IID数据和节点间结构的能力。这些工作在没有收敛保证的情况下也没有解决实际问题，如离散器或容错，这是联邦环境的重要特征。据我们所知，本文提出的工作是第一个从理论和实践上考虑这些挑战的联合学习框架。

多任务学习。在多任务学习中，目标是同时学习多个相关任务的模型。虽然MTL文献非常广泛，但是大多数MTL建模方法可以根据它们如何捕捉任务之间的关系大致分为两类。第一个（例如，[14，4，11，24]）假设任务之间的聚集、稀疏或低秩结构是先验已知的。第二组假设任务关系事先不知道，可以直接从数据中学习（例如[21,57,16]）。在这项工作中，我们将注意力集中在后一组上，因为在现实世界中，任务关系可能事先并不知道。与学习单个全局模型相比，这些MTL方法可以直接捕获非IID和不平衡数据之间的关系，这使得它们特别适合联合学习的统计挑战。在第5节中，我们在真实世界的联邦数据集上进行了经验证明。然而，尽管MTL是解决联合学习的统计挑战的自然建模选择，但是目前提出的分布式MTL方法（下面讨论）并不能充分解决与联合学习相关的系统挑战。

分布式多任务学习。分布式多任务学习是一个相对较新的研究领域，其目的是解决当每个任务的数据分布在网络上时的MTL问题。虽然最近的一些研究[1,35,54,55]已经考虑了分布式MTL训练的问题，但是所提出的方法不考虑通信与计算的灵活性。后者是由于系统的容错性和容错性而导致的。[23]和[7]的工作允许异步更新来帮助减少偏差，但没有解决容错问题。此外，[23]没有提供收敛保证，并且[7]的收敛依赖于有界延迟假设，这对于联邦设置是不实际的，在联邦设置中，延迟可能是显著的，设备可能完全退出。最后，[30]提出了一种利用分布式框架COCOA[22，31]的方法和设置，我们在第4节中展示了这是本工作中更一般方法的一个特例。然而，文献[30]中的作者并没有探究联邦环境，他们认为在每个节点上本地完成相同数量的工作的假设在联邦环境中是禁止的，在联邦设置中，由于数据和系统的可变性，不平衡是常见的。

# 3        联合多任务学习

在联合学习中，目标是学习驻留在分布式节点上并由分布式节点生成的数据的模型。作为一个运行示例，考虑学习移动电话用户在蜂窝网络中的活动，这些活动是基于他们各自的传感器、文本或图像数据的。每个节点（phone）∈[m]可能通过不同的分布生成数据，因此很自然地将不同的模型{w1，…，wm}拟合到每个局部数据集的分布式数据中。然而，模型之间的结构经常存在（例如，人们在使用手机时可能表现出相似的行为），通过多任务学习对这些关系进行建模是提高性能和提高每个节点的有效样本量的自然策略[10,2,5]。在这一部分中，我们将为联邦环境提出一个通用的MTL框架，并提出一种新的方法MOCHA来处理联邦MTL的系统挑战。*米t*

3.1通用多任务学习设置

对于任意的任务向量，通过给定的任务向量机（rfo）学习模型（例如，对每一个任务节点的数据进行独立的权值损失）。许多MTL问题可以通过以下一般公式得到：**Xw公司***t*∈*d*×n*t米t*∈*d`t*

*,* （一）

式中：=[w1，…，wm]∈r一个矩阵，其第-th列为第-th个任务的权重向量。该矩阵描述了任务之间的关系，在学习任务模型的同时，也可以是先验的或估计的。MTL问题基于对R的假设而不同，R作为输入，在任务之间促进某种合适的结构。**WΩΩ***d*×米*tt*∈*米*×米

举个例子，一些流行的MTL方法假设任务是根据它们是否相关而形成集群的[14,21,57,58]。这可以通过以下双凸公式表示：

                                                       R（W，Ω）=λ1 tr**WΩW***,* （二）

常数为0，其中第二项对每个局部模型执行正则化。我们在第5节的实验中使用了类似的公式（14），并提供了其他常见的MTL模型的详细信息，这些模型可以通过附录B中的（1）进行公式化。*λ*1*,λ*2 *>我*2

3.2 MOCHA：联邦多任务学习框架

在联邦环境中，目标是直接在边缘训练统计模型，因此我们在假设数据{X1，…，Xm}分布在节点或设备上的同时求解（1）。在提出求解（1）的联邦方法之前，我们提出以下意见：*米*

•（1）不是联合凸在（1）是凸的，求解*观察1：一般来说，***WΩWΩ***和即使是在和同时可能很困难[5]。*

•*观察2：固定时***ΩW XΩ***，正在更新取决于这两个数据，分布在节点和结构上，这是中心已知的。*

•*观察3：固定时***WΩW X***，正在优化只取决于而不是数据.*

基于这些观察结果，很自然地提出了一种交替优化方法来解决问题（1），在每一次迭代中，我们固定一个或并在另一个迭代上进行优化，交替直到达到收敛。请注意，for的求解不依赖于数据，因此可以集中计算；因此，我们将此步骤的先前工作推迟到[58，21，57，16]。在附录B中，我们讨论了几种常见的MTL模型对Ω的更新。**WΩΩ**

在这项工作中，我们致力于发展一个有效的分布式优化方法。在传统的数据中心环境中，分布式训练是一个研究得很好的问题，最近人们提出了各种有效的通信框架，包括最先进的原始双COCOA框架[22,31]。尽管COCOA可以直接扩展为以分布式方式跨节点进行更新，但它不能处理联邦环境的独特系统挑战，如3.4节中讨论的离散器和容错。为此，我们扩展了COCOA并提出了一种新的联合多任务学习方法MOCHA。我们的方法在算法1中给出，并在第3.3和3.4节中详细描述。**W W W**

算法1 MOCHA：联邦多任务学习框架

1： 输入：数据来自=1，…，m个任务，存储在其中一个节点上，以及初始矩阵**XΩ***tt米*0

2： 起点α：=0∈R，：=0∈R（0）*n***五**（0）*b*

3： 对于迭代次数=0,1，。。。做*我*

4： 设置子问题参数和联合迭代次数，*σ*0 *您好!*

5： 迭代次数=0,1，····，你好*h*

6： 对于任务∈{1，2，…，m}在节点上并行执行吗*t米*

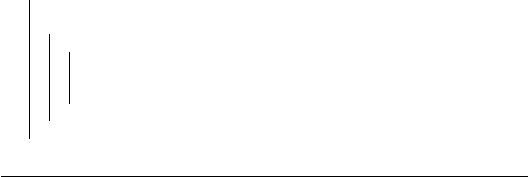
7： 调用局部解算器，返回局部子问题（4）的近似解∆*θthαt*

8： 更新局部变量α←+∆*tαtαt*

9： 返回更新∆vt:=Xt∆*αt*

10： 减小：←vt+∆vt**五***t*

11： 基于（）集中更新最新α**Ωw***α*

12： 中心节点根据最新的α13计算=w（）：返回：：=[w1，…，wm]**w w w***α*

3.3联合更新**W**

为了在联邦环境下进行更新，我们首先扩展分布式原始-对偶优化[22，31，30]的工作，以应用于广义多任务框架（1）。这包括推导适当的对偶公式、子问题和问题参数，如下所述。**W**

双重问题。考虑到子问题（1）的全局分布问题，可以更好地将子问题（1）分解为全局问题。让和**XΩ**：=Diag（X1，····，Xm）∈R。在固定的情况下，关于对偶变量α∈R定义的问题（1）的对偶由：*医学博士*×n*n*

*,* （三）

地点和R\*是和R的共轭对偶函数*`t*，分别是数据点的对偶变量（xit，yti）。注意R\*依赖于，但为了简单起见，我们在注释中删除了它。为了从这个全局对偶中导出分布式子问题，我们在正则化器R上做了如下描述的假设。**Ω**

假设1。给定∈RR R\***ΩMΩ毫米***，我们假设存在一个对称正定矩阵医学博士*×医学博士*，取决于，其中函数关于*−1个*. 请注意，这相当于假设相对于矩阵是平滑的.*

备注1。我们可以将MTL正则化器的形式改为R（w，Ω）=R（w，Ω）∈R：=Ω⊗Id×d∈R（2）为R（w，Ω）=¯¯*，其中***wΩwM***医学博士是一个包含和*¯*医学博士*×医学博士*. 例如，我们可以在*¯¯*tr公司**. 用这种形式写正则化器，很明显它对于矩阵是强凸的*−1个=λ1Ω+λ2I¯*.*

数据局部二次子问题。为了跨分布式节点求解（1），我们定义了以下数据局部子问题，这些子问题是通过对对偶问题（3）的仔细二次近似而形成的，以分离节点间的计算。这些子问题寻找α中对应于单个节点的对偶变量∆∈Rto，并且只需要访问局部可用的数据，即节点的数据。第-个子问题由以下公式给出：*αtntt***十***ttt*



在哪里，还有**Mmwwwv vvM型***t*正定对称矩阵的正对角块。给定对偶变量α，可以通过（）=∇R∇（X）找到对应的原始变量，其中（）是向量（）中的第-th个块。注意，计算（）需要向量=X。在每次迭代中，∈R的第-个块是节点之间必须传递的唯一信息。最后，0度量数据分区的难度，并帮助将子问题的进展与全局对偶问题联系起来。它可以很容易地根据许多感兴趣的应用程序进行选择；我们在附录的引理9中提供了详细信息。*d*×天*tααtαtαααttdσ*0 *>*

3.4实际考虑

在MOCHA的联合更新期间，中心节点需要所有工作线程在执行同步更新之前作出响应。在联邦环境中，由于节点的异构性，这种通信协议的幼稚执行可能会带来显著的离散效应。为了避免掉队，**W**

MOCHA为第-个节点提供了近似求解其子问题Gtσ（·）的灵活性，其中近似的质量由每个节点的参数控制。以下因素决定了第个节点对其子问题的解决方案的质量：*t*0*θtht*

1.   统计挑战，如**十***t*以及子问题的内在困难。

2.   系统挑战，例如由于硬件（CPU、内存）、网络连接（3G、4G、WiFi）和电源（电池电量），节点的存储、计算和通信能力。

3.   由中心节点规定接收更新的截止时间的全局时钟周期。

我们将其定义为这些因素的函数，并假设每个节点都有一个控制器，该控制器可能来自当前时钟周期和统计/系统设置范围从零到一，在哪里*θth*=0表示Gtσ（·）的精确解，=1表示节点在迭代过程中没有任何进展（我们称之为丢弃节点）。例如，如果一个节点的电池用完了，或者它的网络带宽在迭代过程中恶化了，因此它无法在当前时钟周期内返回更新，那么它可能会“掉线”。的正式定义见第4节第（5）款。0*θththhθth*

MOCHA通过允许第-th个节点定义自己的节点来减轻散乱者。在每次迭代中，节点在一个时钟周期内执行并发送的本地更新将产生一个特定的值。如第4节中所讨论的，MOCHA对于一小部分节点周期性地丢弃并且在适当的条件下不执行本地更新（即：=1）具有额外的鲁棒性，如假设2中所定义的那样。相比之下，COCOA以前的工作在联邦设置中可能会受到严重的离散效应的影响，因为它需要一个固定的=θ，同时仍然保持同步更新，并且它不允许丢弃节点（：=1）。*tθhθthθthθth跨所有节点和所有迭代θ*

最后，我们注意到，异步更新方案是一种缓解离散器的替代方法。在本文中，我们没有考虑这些方法，部分原因是与大多数异步方案相关的有界延迟假设限制了容错性。然而，在未来的工作中，进一步探索异步方法与基于近似的同步方法（如MOCHA）之间的区别和联系是很有意思的。

# 4        收敛性分析

MOCHA基于双凸交替方法，保证收敛到问题（1）的平稳解[17,45]。在这个问题与和共同凸的情况下，这样的解也是最优的。因此，在本节的其余部分中，我们将重点讨论在联邦环境下解决MOCHA更新的收敛性。根据第3.4节的讨论，我们首先介绍以下每节点、每轮近似参数。**WΩW**

定义1（每个节点每个迭代近似参数）。我们定义每个节点的迭代精度为：*ht*

*,* （五）

其中是子问题的最小值。我们允许这个值在

[0,1]，其中：=1表示节点在迭代时不更新子问题Gtσ。*θth*0 *th*

虽然（5）中灵活的每节点、每次迭代的近似参数允许考虑离散器和容错，但是这些额外的自由度在为MOCHA提供收敛保证方面也提出了新的挑战。我们提供了以下假设的保证。*θthθth*

假设2。设Hh：=（···，）Hh：=E[θth | Hh]：=P[θ=1]≤pmax 1Θˆ：=E[θ| H1]≤Θmax 1*α*（小时）*,α*（h−1）*,α*（一）*在迭代开始之前是双向量历史h、 并定义. 对于所有任务t和所有迭代h、 我们假设甲状旁腺素第<和高温第h,θ第<<.*

这个假设说明，在每次迭代中，节点发送结果的概率是非零的，并且返回结果的质量平均比前一次迭代的质量好。与[49，30]的假设相比，我们的假设限制性要小得多，并且可以更好地模拟联邦环境，其中节点是不可靠的，并且可能周期性地退出。

利用假设2，我们得到了如下定理：当（1）中的损失是光滑的时，MOCHA联邦更新在有限时间范围内的收敛性。*`t*

定理1。假设损耗（1/µ）∈（0,1]*`是的-平滑。那么，在假设1和2下，存在一个常数s对于任何给定的收敛目标*D*，选择这样的话*

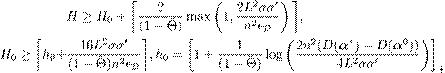
                                                                                                             *,* （六）

*会满足的*

这里，Θ：=+（1−pmax）Θmax<1。当定理1涉及到有限时域收敛时，可以得到渐近收敛的结果，即→∞，对离散器的假设比较温和；具体见附录中的推论8。¯*p*最大值*H*

当损失函数是非光滑的，例如支持向量机模型的铰链损失时，我们给出了-Lipschitz损失的次线性收敛性。*我*

定理2。如果损失起作用*`是的L-Lipschitz，那么存在一个常数σ、 定义于*（24），如果我们选择

（七）

*具有,*

*然后**会满足的**.*

这些定理保证了MOCHA将在联邦环境下收敛，前提是对离散器和节点能力的温和假设。虽然这些结果考虑了对偶收敛性，但我们证明了它们对对偶缺口的相似性。我们在附录C中提供了所有证据。

备注2。根据第3.4节的讨论，我们的方法和理论推广了[22，31]中的结果。在限制的情况下，所有的COCOA都以（1）中描述的多任务框架为基础。*θth相同，我们的结果推广了*

备注3。注意[22，31]中的方法有一个聚集参数∈（0,1]：=1*γ. 尽管我们证明了我们的结果γ、 我们通过设置γ，无论是理论上还是经验上，它都被证明具有最好的性能[31]。*

# 5        模拟

在这一部分中，我们验证了摩卡的实证表现。首先，我们介绍了一套真实联邦数据集的基准测试套件，并说明多任务学习非常适合处理联邦设置的统计挑战。接下来，我们将从统计和系统异构性两个方面展示MOCHA处理掉队者的能力。最后，我们探讨了MOCHA在器件周期性脱落时的性能。我们的代码在：github.com/gingsmith/fmtl.

5.1联合数据集

在我们的模拟中，我们使用在联邦设置中生成的几个真实世界数据集。我们在附录中提供了其他详细信息，包括有关数据大小的信息。*新台币*

•Google Glass（GLEAM）[3]：该数据集包括从38名佩戴谷歌眼镜的参与者身上收集的两小时高分辨率传感器数据，用于活动识别。在[41]之后，我们将原始加速度计、陀螺仪和磁强计数据特征化为180个统计、光谱和时间特征。我们将每个参与者建模为一个单独的任务，并预测饮食和其他活动（如散步、谈话、喝酒）之间的关系。

•人类活动识别[4]：从30个人收集的手机加速度计和陀螺仪数据，执行六种活动之一：{行走、上楼、下楼、坐、站、躺}。我们使用为每个实例生成的561个时频域变量的特征向量[3]。我们将每个个体作为一个单独的任务建模，并预测坐姿和其他活动之间的关系。

•车辆传感器[5]：从由23个传感器组成的分布式网络收集的声学、地震和红外传感器数据，其部署目的是对一段道路上行驶的车辆进行分类[13]。每个实例由50个声学特征和50个地震特征描述。我们将每个传感器建模为一个单独的任务，并预测AAV型和DW型车辆。

5.2联邦环境下的多任务学习

我们通过比较多任务模型与完全局部模型（即分别为每个任务学习一个模型）和一个完全全局模型（即组合来自所有任务的数据并学习一个单一模型）的错误率，来展示联邦环境下多任务学习的好处。到目前为止，关于联合学习的工作仅限于对完全全局模型的研究[25,36,26]。

我们使用集群正则化多任务模型[57]，如第3.1节所述。对于第5.1节中的每个数据集，我们将数据随机分成75%的训练和25%的测试，学习多任务、局部和全局支持向量机模型，选择最佳正则化参数∈{1e-5，1e-4，1e-3，1e-2，0.1，1，10}，对每个模型进行5次交叉验证。我们重复这一过程10次，并报告了10次试验的平均预测误差。*λ*

表1：平均预测误差：10次随机洗牌的平均值和标准误差。

|  |  |  |  |
| --- | --- | --- | --- |
| 模型 | 人类活动 | 谷歌眼镜 | 车辆传感器 |
| 全球的 | 2.23（0.30） | 5.34（0.26） | 13.4（0.26） |
| 本地 | 1.34（0.21） | 4.92（0.26） | 7.81（0.13） |
| MTL公司 | 0.46（0.11） | 2.02（0.15） | 6.59（0.21） |

在表1中，我们可以看到，对于每个数据集，多任务学习在实现任务平均误差最小方面显著优于其他模型。在[25，36，26]中提出的全球模型表现最差，尤其是对于人类活动和车辆传感器数据集。尽管数据集已经有点不平衡，但我们注意到全局建模方法可能有利于具有非常少量实例的任务，因为信息可以跨任务共享。出于这个原因，我们在附录的表4中进一步探讨了全局、局部和多任务建模对高度倾斜数据的性能。尽管在这种情况下，相对于局部建模，全局模型的性能稍有改善，但对于大多数数据集，全局模型的性能仍然最差，并且MTL仍然显著优于全局和局部方法。

5.3避免掉队

在联合学习中普遍存在的两个挑战是分散学习和高度交流。当设备的一个子集执行本地更新所需的时间比其他子集长得多时，可能会出现离散器，这可能是由统计或系统异构引起的。通信也会加剧性能差，因为在典型的蜂窝或无线网络中，通信比计算慢许多数量级[52、20、48、9、38]。在下面的实验中，我们通过跟踪操作和通信复杂性来模拟运行每种方法所需的时间，并将与计算相关的通信成本分别缩放一个、两个或三个数量级。这些数字大致对应于现代蜂窝和无线网络的时钟速率与网络带宽/延迟的关系（例如，参见[52]）。详情见附录E。

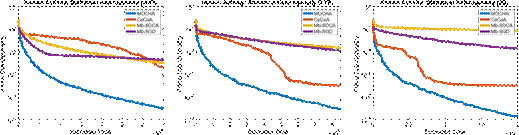


图1:MOCHA在更新（1）时与其他分布式方法相比的性能。虽然增加通信往往会降低小批量方法的性能，但是MOCHA在高通信设置下表现良好。在所有情况下，具有不同近似值的摩卡咖啡比没有近似值的摩卡咖啡表现更好（即天真地概括可可），因为它避免了统计上的差异。**W**

统计异质性。我们探讨了统计异质性对不同通信方式和通信方式（3G、LTE、WiFi）下的离散性的影响。对于固定通信网络，我们将MOCHA与COCOA（单参数）和mini-batch随机梯度下降（Mb-SGD）和mini-batch随机双坐标上升（Mb-SDCA）进行比较，后者的通信灵活性因批次大小而异。我们对所有比较过的方法进行优化以获得最佳性能，如附录E中所述。在图1中，我们可以看到，虽然在高通信状态下，小批量方法的性能会下降，但是摩卡和可可对高通信是健壮的。然而，COCOA会受到离散器的显著影响，因为它在节点和轮次之间是固定的，困难的子问题会对收敛产生不利影响。相比之下，无论通信成本如何，MOCHA的性能都很好，并且对统计异构性是健壮的。*θθ*

系统异构性。MOCHA还可以处理来自不断变化的系统环境的异构性，如电池电量、内存或网络连接，如图2所示。特别是，我们通过随机选择MOCHA的本地迭代次数或mini-batch方法的小批量大小来模拟系统的异构性，对于高可变性环境，在最小本地数据点数量的10%到100%之间，对于低可变性，在90%到100%之间（详见附录E）。我们不会改变可可的表现，因为单是统计异质性的影响就显著降低了可可的性能。然而，添加系统异构性会进一步降低性能，因为只有在引入额外的系统挑战时，所有节点的最大值才会增加。*θ*

|  |
| --- |
|  |
|  |  |

5.4对脱落节点的容忍度2 **车辆传感器：系统异构性（低）车辆传感器：系统异构性（高）**2

最后，我们探讨了节点丢弃对MOCHA性能的影响。我们没有与其他方法进行比较，就我们所知，没有其他方法可以直接解决分布式多任务学习的容错问题。在摩卡，我们通过允许：=1来合并此设置，如第4节理论上所述。在图3中，我们观察MOCHA的性能，无论是对于一个固定的更新，还是运行整个MOCHA方法，当节点在每次迭代（假设2中）下降的概率增加时。我们看到MOCHA的性能对于相对较高的值是健壮的，这两个值都是在*θth***W W W***phtpht*以及它如何影响整个方法的性能。然而，正如直觉所暗示的那样，如果其中一个节点从不发送更新（即，对于所有节点*h*，绿色虚线），该方法无法收敛到正确的解。这为我们的假设2提供了验证。

图2:MOCHA可以处理来自系统异构性的可变性。

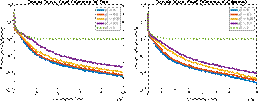


图3:MOCHA的性能对于节点周期性地退出（容错）是健壮的。

# 6        讨论

为了解决新兴的联邦学习环境在统计和系统方面的挑战，我们提出了MOCHA，一个新的联邦多任务学习系统感知优化框架。我们的方法和理论首次考虑了联邦环境下多任务学习的高通信成本、离散和容错问题。虽然MOCHA不适用于当前形式的非凸深度学习模型，但我们注意到，在核心化联邦多任务学习的背景下，这种方法与“凸化”深度学习模型[6,34,51,56]之间可能存在天然的联系。

# 致谢

我们感谢Brendan McMahan，ChloéKiddon，Jakub Konecný，Evan Sparks，Xinghao Pan，Lisha Li，ˇ和Hang Qi的宝贵讨论和反馈。

# 工具书类

[1] A.艾哈迈德，A.达斯，A.J.斯莫拉。显示广告转换优化的分级多任务学习算法。在2014年网络搜索和数据挖掘会议上。

[2] 安藤和张先生。从多个任务和未标记的数据中学习预测结构的框架。机器学习研究杂志，6:1817–18532005。

[3] D.Anguita、A.Ghio、L.Oneto、X.Parra和J.L.Reyes Ortiz。使用智能手机进行人类活动识别的公共领域数据集。欧洲人工神经网络、计算智能和机器学习研讨会，2013年。

[4] A.Argyriou，T.Evgeniou和M.Pontil。多任务特征学习。神经信息处理系统，2007。

[5] A.Argyriou，T.Evgeniou和M.Pontil。凸多任务特征学习。机器学习，73（3）：243–2722008年。

[6] 厄伊。Aslan，X.Zhang和D.Schuurmans。基于归一化核的凸深度学习。神经信息处理系统进展，2014年。

[7] I.M.Baytas，M.Yan，A.K.Jain和J.Zhou。异步多任务学习。在2016年国际数据挖掘大会上。

[8] F.Bonomi、R.Milito、J.Zhu和S.Addepalli。雾计算及其在物联网中的作用。2012年SIG移动通信研讨会。

[9] 卡罗尔和海瑟。智能手机功耗分析。2010年USENIX年度技术会议。

[10] 卡鲁阿纳。多任务学习。机器学习，28:41–751997。

[11] 陈、周、叶。结合低秩和群稀疏结构的鲁棒多任务学习。2011年知识发现与数据挖掘会议。

[12] A.Deshpande，C.Guestrin，S.R.Madden，J.M.Hellerstein和W.Hong。传感器网络中基于模型的近似查询。VLDB期刊，14（4）：417–4432005年。

[13] 杜阿尔特先生和胡先生。分布式传感器网络中的车辆分类。并行与分布式计算杂志，64（7）：826–8382004。

[14] 吉尼特和吉尼特。规则化多任务学习。2004年知识发现与数据挖掘会议。

[15] P.Garcia Lopez、A.Montresor、D.Epema、A.Datta、T.Higashino、A.Imanitchi、M.Barcellos、P.Felber和E.Riviere。边缘中心计算：愿景与挑战。SIGCOMM计算机通信评论，45（5）：37–422015。

[16] A.R.Gonçalves，F.J.Von Zuben和A.Banerjee。基于高斯copula模型的多任务稀疏结构学习。机器学习研究杂志，17（33）：1-30，2016。

[17] 戈尔夫，奥夫先生。双凸集与双凸函数优化：综述与扩展。运筹学的数学方法，66（3）：373–4072007。

[18] K.Hong、D.Lillethun、U.Ramachandran、B.Ottenwälder和B.Koldehof。Mobile fog：物联网大规模应用的编程模型。2013年SIGCOMM移动云计算研讨会。

[19] C.-J.谢，M.A.Sustik，I.S.Dhillon和P.Ravikumar。基于二次逼近的稀疏逆协方差矩阵估计。《神经信息处理系统》2014年第27期。

[20] 黄俊杰、钱富珍、郭玉柱、周玉柱、徐志强、毛泽民、森森、斯帕切克。lte的深入研究：网络协议和应用行为对性能的影响。2013年ACM SIGCOMM会议。

[21]L.雅各布、J.-p.维特和F.R.巴赫。集群多任务学习：一个凸公式。神经信息处理系统，2009。

[22]M.Jaggi、V.Smith、J.Terhorst、S.Krishnan、T.Hofmann和M.I.Jordan。通信高效分布式双坐标上升。神经信息处理系统，2014年。

[23]X.Jin，P.Luo，F.庄，J.He，Q.He。为分布式在线多任务协作本地和全局学习。2015年知识与管理大会。

[24]S.Kim和E.P.Xing。数量性状网络相关基因组关联的统计估计。公共科学图书馆遗传学，5（8）：e10005872009。

[25]J.Konecnˇy、H.B.McMahan和D.Ramage。联邦优化：数据中心之外的分布式优化。arXiv:1511.03575，2015年。

[26]J.Konecnˇy、H.B.McMahan、F.X.Yu、P.Richtárik、A.T.Suresh和D.Bacon。联合学习：提高沟通效率的策略。arXiv:1610.054922016年。

[27]T.Kuflik，J.Kay和B.Kummerfeld。普适用户建模的挑战与解决方案。在无处不在的显示环境中，第7-30页。斯普林格，2012年。

[28]A.库马尔和H.道梅。多任务学习中的学习任务分组与重叠。在2012年国际机器学习大会上。

[29]S.L.劳里岑。图形模型，第17卷。克拉伦登出版社，1996年。

[30]刘S，潘石屹，何Q。分布式多任务关系学习。2017年知识发现与数据挖掘会议。

[31]C.Ma，V.Smith，M.Jaggi，M.I.Jordan，P.Richtárik，和M.TakáC.分布式ˇprimal-dual optimization中的加法与平均。在2015年国际机器学习大会上。

[32]S.Madden，M.J.Franklin，J.M.Hellerstein和W.Hong。TAG：一种用于adhoc传感器网络的小型聚合服务。在2002年操作系统设计和实现专题讨论会上。

[33]S.Madden，M.J.Franklin，J.M.Hellerstein和W.Hong。TinyDB：一个用于传感器网络的查询处理系统。数据库系统上的ACM事务，30（1）：122–1732005。

[34]J.Mairal、P.Koniusz、Z.Harchaoui和C.Schmid。卷积核网络。神经信息处理系统，2014年。

[35]D.Mateos-Núñez和J.Cortés.通过核范数近似实现多任务学习的分布式优化。在IFAC网络系统分布式估计和控制研讨会上，2015年。

[36]H.B.McMahan、E.Moore、D.Ramage、S.Hampson和B.A.y Arcas。从分散数据中有效地学习深层网络。2017年人工智能与统计会议。

[37]H.B.McMahan和D.Ramage。http://www.googblogs.com/federated learning无需集中培训数据的协作机器学习/。谷歌，2017年。

[38]A.P.Miettinen和J.K.Nurminen。云计算中移动客户端的能效。在2010年USENIX云计算热点话题会议上。

[39]A.潘泰洛普洛斯和N.G.布尔巴基斯。基于可穿戴传感器的健康监测与预后系统综述。IEEE系统、人与控制论汇刊，40（1）：1-12，2010。

[40]H.Qi、E.R.Sparks和A.Talwalkar。古：深层神经网络的性能模型。2017年国际学习代表大会。

[41]S.A.Rahman、C.Merck、Y.Huang和S.Kleinberg。使用谷歌眼镜的非侵入性进食识别。2015年医疗保健普及计算技术会议。

[42]P.拉希迪和D.J.库克。让住户保持在循环中：让智能家居适应用户。IEEE系统、人与控制论汇刊，39（5）：949-9592009。

[43]M.Rastegari、V.Ordonez、J.Redmon和A.Farhadi。XNOR网：使用二元卷积神经网络的影像网分类。2016年欧洲计算机视觉大会。

[44]拉维。https://research.googleblog.com/2017/02/on-device-machine-intelligence。

html。谷歌，2017年。

[45]M.Razaviyayn，M.Hong，和Z.-Q.Luo。非光滑优化分块逐次极小化方法的统一收敛性分析。暹罗优化杂志，23（2）：1126-11532013。

[46] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal Estimated sub-GrAdient SOlver for SVM. International Conference on Machine Learning, June 2007.

[47] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. Journal of Machine Learning Research, 14:567–599, 2013.

[48] D. Singelée, S. Seys, L. Batina, and I. Verbauwhede. The communication and computation cost of wireless security. In ACM Conference on Wireless Network Security, 2011.

[49] V. Smith, S. Forte, C. Ma, M. Takác, M. I. Jordan, and M. Jaggi. CoCoA: A general framework for communication-efficient distributed optimization. arXiv:1611.02189, 2016.

[50] M. Takác, A. Bijral, P. Richtárik, and N. Srebro. Mini-Batch Primal and Dual Methods for SVMs. Inˇ International Conference on Machine Learning, 2013.

[51] C.-Y. Tsai, A. M. Saxe, and D. Cox. Tensor switching networks. In Neural Information Processing Systems, 2016.

[52] C. Van Berkel. Multi-core for mobile phones. In Proceedings of the Conference on Design, Automation and Test in Europe, pages 1260–1265. European Design and Automation Association, 2009.

[53] H. Wang, A. Banerjee, C.-J. Hsieh, P. K. Ravikumar, and I. S. Dhillon. Large scale distributed sparse precision estimation. In Neural Information Processing Systems, 2013.

[54] J. Wang, M. Kolar, and N. Srebro. Distributed multi-task learning. In Conference on Artificial Intelligence and Statistics, 2016.

[55] J. Wang, M. Kolar, and N. Srebro. Distributed multi-task learning with shared representation. arXiv:1603.02185, 2016.

[56] Y. Zhang, P. Liang, and M. J. Wainwright. Convexified convolutional neural networks. International Conference on Machine Learning, 2017.

[57] Y. Zhang and D.-Y. Yeung. A convex formulation for learning task relationships in multi-task learning. In Conference on Uncertainty in Artificial Intelligence, 2010.

[58] J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. In Neural Information Processing Systems, 2011.

[[1]](" \l "_ftnref1" \o ") The term on-device learning has been used to describe both the task of model training and of model serving. Due to the ambiguity of this phrase, we exclusively use the term federated learning.

[[2]](" \l "_ftnref2" \o ") While not the focus of our work, we note privacy is an important concern in the federated setting, and that the privacy benefits associated with global federated learning (as discussed in [36]) also apply to our approach.

[[3]](" \l "_ftnref3" \o ") http://www.skleinberg.org/data/GLEAM.tar.gz

[[4]](" \l "_ftnref4" \o ") https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

[[5]](" \l "_ftnref5" \o ") http://www.ecs.umass.edu/~mduarte/Software.html