

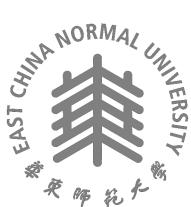
2022 届硕士专业学位研究生学位论文

分类号: \_\_\_\_\_

学校代码: 10269

密 级: \_\_\_\_\_

学 号: 51194501126



東華師範大學

**East China Normal University**

**硕士专业学位论文**

**MASTER'S DISSERTATION (Professional)**

**论文题目：基于联邦学习的隐私保护的技术  
研究**

院 系: 信息学部软件工程学院

专业学位类别: 工程硕士

专业学位领域: 软件工程

论文指导教师: 曹珍富 教授

论文作 者: 何慧娴

2020 年 11 月 20 日

Thesis (Professional) for Master's Degree in 2021

School Code: 10269

Student Number:51184501139

# EAST CHINA NORMAL UNIVERSITY

## **TITLE: VERIFICATION AND ANALYSIS OF ROBUSTNESS OF MACHINE LEARNING TREE MODEL BASED ON SMT TECHNOLOGY**

Department:	Software Engineering Institute of Information Department
Major:	Software Engineering
Research Direction:	Trustworthy Artificial Intelligence
Supervisor:	Associate Professor Jianqi Shi
Candidate:	Chaoqun Nie

Nov 9, 2020



# 摘 要

近年来，机器学习由于其卓越的性能已被越来越多的应用于各个领域，如自动驾驶，人脸识别，个人信用评估等。但机器学习模型一般都为黑盒，由于其不可见性与不可解释性，使得人们对它的安全性有了很大的担忧。因此，越来越多的研究人员投身到机器学习模型的安全性验证的方法和工具的研究中。

与神经网络模型一样，树模型也容易受到对抗性样本的攻击。树模型的“脆弱性”在使其在某些安全性要求较高的应用中造成了隐患。有些情况下，甚至可能导致灾难性的后果。为此，我们研究了树模型的鲁棒性验证问题。本文主要的工作和贡献如下：

1. 我们提出了一个基于 SMT 技术的树模型鲁棒性验证框架，它可以有效的验证树模型的两个重要组成部分：随机森林和 GBDT 模型的鲁棒性，并且支持规模较大的模型的验证。该框架的核心思想是将树模型的鲁棒性验证问题转化为 SMT 公式的约束求解问题。
2. 在验证的基础上，我们进一步对树模型鲁棒性的可解释性问题进行了研究，提出了鲁棒特征集合和局部鲁棒特征重要度的概念来描述模型鲁棒性与样本特征的内在联系，从而为对抗性样本攻击提供了新的思路。
3. 我们基于三个基准测试集评估了框架的可行性和有效性，并且在实验中讨论了模型训练超参数与其鲁棒性的关系，从而为训练阶段提高模型的鲁棒性提供了重要参考。

关键词： 鲁棒性验证，可解释性，随机森林，GBDT，SMT

## ABSTRACT

In recent years, machine learning has been increasingly applied in various fields, such as autonomous driving, face recognition, personal credit assessment and so on, due to its excellent performance. However, the machine learning model is generally a black box, because of its invisibility and ineluctability, people have great concerns about its security. Therefore, more and more researchers are involved in the research of methods and tools for security verification of machine learning models. Like the neural network model, the tree model is also vulnerable to the attack of the antagonistic sample. The "fragility" of the tree model makes it potentially dangerous and, in some cases, potentially disastrous in some applications where security is high. Therefore, we study the robustness verification of tree models. The main work and contributions of this paper are as follows:

1. We propose a tree model robustness verification framework based on SMT technology, which can effectively verify the robustness of two important components of tree models: random forest and GBDT, and support the verification of larger tree models. The core idea of this framework is to transform the robustness verification problem of tree model into the constraint solving problem of SMT formula.
2. On the basis of verification, we further study the problem of interpretability of tree model robustness, and propose the concepts of robust feature set and local robustness feature importance to describe the internal relationship between model robustness and sample characteristics, so as to provide a new idea for resisting sample attack.

3. We evaluated the feasibility and effectiveness of the framework based on three benchmark test sets, and discussed the relationship between model training hyperparameters and its robustness in the experiment, thus providing an important reference for improving the robustness of the model in the training stage.

**Keywords:** *Robustness verification, Interpretability, Random forest, GBDT, SMT*



# 目录

<b>第一章 绪 论 . . . . .</b>	<b>1</b>
1.1 研究背景及意义 . . . . .	1
1.2 问题和挑战 . . . . .	3
1.2.1 数据异构 . . . . .	3
1.2.2 高昂的通信代价 . . . . .	4
1.2.3 安全性和隐私威胁 . . . . .	4
1.3 国内外研究现状 . . . . .	5
1.3.1 攻击模型的研究现状 . . . . .	6
1.3.2 隐私保护的研究现状 . . . . .	7
1.4 本文工作与主要贡献 . . . . .	9
1.5 本文组织结构 . . . . .	10
<b>第二章 相关知识介绍 . . . . .</b>	<b>11</b>
2.1 联邦学习 . . . . .	11
2.1.1 基本定义 . . . . .	11
2.1.2 联邦学习的分类 . . . . .	11
2.1.3 联邦学习的训练步骤 . . . . .	12
2.1.4 联邦学习中的安全和隐私威胁 . . . . .	13
2.2 差分隐私 . . . . .	15
2.2.1 基本定义 . . . . .	16
2.2.2 相关概念 . . . . .	17
2.2.3 实现机制 . . . . .	18

2.3	神经网络 . . . . .	19
2.4	本章小结 . . . . .	20
<b>第三章</b>	<b>联邦学习的自适应加噪机制 . . . . .</b>	<b>21</b>
3.1	模型概况 . . . . .	21
3.1.1	系统架构 . . . . .	21
3.1.2	本地训练 . . . . .	22
3.1.3	全局参数更新 . . . . .	24
3.1.4	威胁模型 . . . . .	24
3.1.5	联邦学习中的差分隐私 . . . . .	24
3.2	方案设计 . . . . .	25
3.2.1	自适应噪声添加 . . . . .	25
3.2.2	隐私性证明 . . . . .	27
3.2.3	隐私预算分析 . . . . .	28
<b>第四章</b>	<b>联邦学习的安全聚合模型 . . . . .</b>	<b>31</b>
4.1	鲁棒特征集合 . . . . .	32
4.2	局部鲁棒特征重要度 . . . . .	34
4.3	本章小结 . . . . .	36
<b>第五章</b>	<b>实验与评估 . . . . .</b>	<b>37</b>
5.1	基准数据集介绍 . . . . .	37
5.2	实验环境与配置 . . . . .	38
5.3	实验结果与分析 . . . . .	38
5.3.1	随机森林模型的鲁棒性验证与分析 . . . . .	38
5.3.2	GBDT 模型鲁棒性的验证与分析 . . . . .	40
5.3.3	树模型鲁棒性可解释性的实验与分析 . . . . .	42
5.3.4	不同类别鲁棒性的验证与分析 . . . . .	44
5.3.5	树鲁棒性超参数与鲁棒性关系的验证与分析 . . . . .	46
5.3.6	验证时间的结果与分析 . . . . .	47
5.4	本章小结 . . . . .	48

第六章 总结与展望 . . . . .	50
6.1 总结 . . . . .	50
6.2 展望 . . . . .	51

# 插图

5.1	随机森林回归模型验证结果 . . . . .	39
5.2	对抗性样本图 . . . . .	39
5.3	GBDT 回归模型验证结果 . . . . .	41
5.4	GBDT 验证反例 . . . . .	41
5.5	随机森林分类模型鲁棒特征集合 . . . . .	42
5.6	GBDT 分类模型鲁棒特征集合 . . . . .	43
5.7	随机森林分类的局部鲁棒性特征重要度 . . . . .	44
5.8	MNIST 中不同类别鲁棒性的验证结果 . . . . .	45
5.9	FASHION-MNIST 中不同类别鲁棒性的验证结果 . . . . .	45
5.10	单样本验证时间图 . . . . .	48

# 表格

5.1 基于 MNIST 数据集模型在不同超参数下的鲁棒性验证结果. . . . . 47

# 第一章 緒論

## 1.1 研究背景及意义

随着机器学习的不断发展和壮大，我们一方面惊叹于它的成就，比如 Alpha GO 击败了围棋世界冠军柯洁，或者面部识别技术帮助我们抓住了躲藏多年的逃犯，而大型工业企业也大力推动机器学习技术的应用。另一方面，我们也必须认识到，它的巨大潜力还有待实现，例如：构建基于大量病例的医疗救助诊断系统，运行基于大量商业行为数据的信用风险控制模型，帮助高价值企业融资，并基于整个产业链的数据提供个性化的产品分配和营销策略。我们真正见证了人工智能（AI）的巨大潜力，以及已经开始期待在许多应用中使用更复杂、更尖端的人工智能技术，包括无人驾驶、医疗、金融等。今天，人工智能技术几乎在各方面都大显身手，每个行业和各行各业。但是传统的机器学习方法依赖于集中管理的训练数据集，建立在大量数据上，从数据中学习特征，从而完成复杂的任务，甚至是人类也难以完成的操作。

然而，这些数据的采集可能涉及到用户的隐私，随着人们的隐私意识的普遍提高，相关的隐私法律法规的不断完善，中国出台的《网络安全与数据合规》白皮书中明确要求加强用户个人信息保护。2018 年欧洲联盟出台《通用数据保护条例》中强调保护用户的个人隐私和数据安全，用户可以删除或撤回其个人数据。近年来，也有越来越多的涉及数据泄漏和隐私侵权的事情，用户们也越来越关注自己的隐私信息是否在未经个人许可，或者出于商业和政治目的被他人或机构利用。随着个人意识和国家政策的关注，在大数据和人工智能领域数据采集和使用的过程中，保护用户隐私和数据的机密显得越来越重要。

大多数训练数据是由不同组织的个人或部门产生的，一个 AI 项目可能涉及多个领域，需要融合各个公司、各个部门的数据。（比如研究居民线上消费问题，需要各个消费平台的数据，可能还需要银行数据等等），但在现实中想要将分散在各地、各个机构的数据进行整合几乎是不可能的。传统的机器学习是通过收集数据并将其发送到一个能看到并控制所有数据的中央服务器来完成的。因此，这个中心位置不仅要有强大的计算机集群来训练和创建机器学习模型，还要处理敏感数据并防止数据被用于其他目的。此外，敏感数据的处理方式必须不损害用户的隐私。然而，这用户完全信任服务器的假设已不再适用。在这种情况下，数据拥有者倾向于将数据掌握在自己手中，这就导致了孤立的数据孤岛，数据孤岛使所有利益相关者无法获得更多的数据。例如，每家医院的居民医疗记录的样本量完全不够，导致模型有偏差。在信贷领域，银行只能使用中央银行的信贷报告来建立风险控制模型。

人工智能的力量是基于大数据的，但我们被更多的小数据包围在孤岛中。大数据的基础就没有了，人工智能的基础也没有了。大数据的基础已经消失，人工智能的未来也岌岌可危。要解决大数据的困境，仅仅靠传统的方法已经出现瓶颈。两个公司简单的交换数据在很多法规包括《通用数据保护条例》是不允许的。用户是原始数据的拥有者，在用户没有批准的情况下，公司间不能交换数据。传统的机器学习和深度学习的方法本身已经成为解决大数据困境的绊脚石。简单地在两家公司之间交换数据，无论是《通用数据保护条例》还是 GDPR 都是不允许的：用户是原始数据的所有者，未经其同意，数据不能在公司之间交换。

那如何创建一个机器学习框架，使人工智能系统能够更有效和准确地集体使用数据，同时满足隐私、安全和监管要求，并解决数据孤岛的问题。如何才能做到这一点呢？

为了解决这个问题，google 在 2016 年率先提出了联邦学习的概念，它提供了一个具有隐私保护功能的分布式机器学习框架，并且能够以分布式方式与成千上万的参与者协作，迭代训练一个特定的机器学习模型。由于训练数据在联合过程中

保持在参与者的本地，这种机制允许参与者之间共享训练数据，同时确保每个参与者的隐私 [15]。联合学习的基本工作流程如下：(1) 初始化：所有用户在他们的设备上都有一个预先分配的神经网络模型，并且可以自愿加入联邦学习协议，指定相同的机器学习和模型训练目标。(2) 本地训练：在一个给定的通信回合中，联邦参与者首先从中央服务器下载全局模型参数，然后使用他们的私人训练模式训练模型，创建本地模型更新（即模型参数），并将这些更新发送到中央服务器。(3) 模型平均化。下一轮的全局模型是通过汇总所有通过训练不同的训练模式获得的模型更新并取其平均值来确定的。(4) 迭代地执行上述步骤以达到优化当前全局模型的目的，整个迭代过程将在全局模型参数满足收敛条件时停止。

联合学习在隐私敏感的场景（包括金融、工业和许多其他与数据相关的场景）中显示出巨大的前景，这是因为它具有独特的优势，能够从多个参与者的本地数据中训练出一个统一的机器学习模型，同时保护数据隐私 [16\17]。联合学习解决了数据聚合的问题，并允许一些机器学习模型和算法在各机构和部门之间进行设计和训练。在一些移动设备上的机器学习模型应用中，联邦学习显示出良好的性能和稳健性。此外，对于一些没有足够的私人数据来开发准确的本地模型的用户（客户）来说，机器学习模型和算法的性能可以通过联合学习得到显著改善。

## 1.2 问题和挑战

### 1.2.1 数据异构

由于联邦学习的重点是通过以分布式方式从所有参与的客户端设备中学习本地数据来获得高质量的全局模型，所以它无法捕捉每个设备的个人信息，导致推理或分类性能下降。此外，传统的联邦学习要求所有参与的设备同意使用一个共同的模型来共同训练，这在复杂的现实世界物联网应用中是不现实的。研究人员对学习在实际应用中面临的问题总结如下 [2]。

(1) 设备的异质性：由于客户端设备的硬件条件 (CPU、内存)、网络连接 (3G、4G、5G、WiFi) 和电源 (电池) 的变化，联邦学习网络上每个设备的存储、计算和

通信能力都可能不同。由于网络和设备的限制，在任何时候都只有某些设备可以活动。此外，设备可能会受到意外事件的影响，如断电或断网，这可能会导致暂时的断网。这种异质性的系统结构影响了联邦模型的整体学习战略。

(2) 统计的异质性：在整个网络中，设备通常以不同的方式产生和收集数据，而且不同设备的数据量、特征等会有很大的不同，所以联合学习网络中的数据不是独立和相同的分布（非 IID）。目前，目前的机器学习算法主要是基于对 IID 数据的假想假设。因此，非 IID 数据的异质属性给建模、分析和评估带来了重大挑战。[\[19\]](#) 提出了 Federated Averageing (FedAvg) 方法来解决非均匀同分布数据的问题，但是当数据分布偏态很严重的时候 FedAvg 的性能退化严重，一方面其性能比中心化的方法差好多，另一方面它只能学习到 IoT 设备粗粒度的特征而无法学习到细粒度的特征。

(3) 模型的异质性：每个客户根据其应用场景要求定制不同模型。

### 1.2.2 高昂的通信代价

在联邦学习过程中，根据存储在几十甚至几百万个远程客户端设备上的数据来学习一个全局模型。在训练期间，客户设备必须定期与中央服务器进行通信原始数据被储存在本地的远程客户端设备上，这些设备必须不断地与中央服务器互动，以完成全局模型的构建。通常情况下，整个联盟学习网络可能涉及大量的设备，而网络通信可能比本地计算慢几个数量级，因此高通信成本成为联邦学习的关键瓶颈。

### 1.2.3 安全性和隐私威胁

(1) 由于联合学习系统的云端服务器无法访问参与者的本地数据和他们的训练过程，恶意参与者可以发送无效的模型更新来达到并破坏全局模型。例如，内部攻击者可以通过在修改后的训练数据上引起有毒的模型更新来有效地损害全局模型的准确性。内部攻击可以由联邦学习服务器发起，也可以由联邦学习参与方发起。外部攻击（包括偷听者）通过参与方与服务器之间的通信通道发起。外部攻击的发

起者大部分为恶意的参与方，例如敌对的客户、敌对的分析者、破坏学习模型的敌对设备或者其组合。在联邦学习中，恶意设备可以通过白盒或者黑盒的方式访问最终模型，因此在防范来自系统外部的攻击时，需要考虑模型迭代过程中的参数是否存在泄露原始数据的风险，这对严格的隐私保护提出了新的挑战。

(2) 由于局部模型更新和全局模型参数的结合提供了关于训练数据的隐藏知识，用户的个人信息有可能泄露给不受信任的服务器或其他恶意用户。例如，即使是由其他用户的训练数据生成的样本原型也会被恶意用户隐蔽地窃取。在训练过程中，攻击方可以试图学习、影响或者破坏联邦学习模型。在联邦训练的过程中，攻击方可以通过数据中毒攻击的方式改变训练数据集合收集的完整性，或者通过模型中毒攻击改变学习过程的完整性。攻击方可以攻击一个参与方的参数更新过程，也可以攻击所有参与方的参数更新过程。若联邦学习的参与方想利用各方的数据集合训练一个模型，但是又不想让自己的数据集泄露给服务器，就需要约定联邦建模的模型算法(例如神经网络)和参数更新的机制(例如随机梯度下降(stochastic gradient descent, SGD))。那么在训练前，攻击方就可以获取联邦学习参数更新的机制，从而指定对应的推断攻击策略。

(3) 在不信任的云服务器和恶意参与者的勾结下，任何个人的确切私人信息都会被泄露。

### 1.3 国内外研究现状

尽管联邦学习提供了隐私保护的机制，还是有各种类型的攻击方式可以攻击联邦学习系统，从而破坏联邦学习系统安全和参与方的隐私。本节将讨论关于联邦学习的攻击问题。从参与方的类型来看，可以将联邦学习的威胁模型细分为半诚实模型(semi-honest model)和恶意模型。对于联邦学习系统的攻击，本文按照不同的维度进行不同层次的分类。从攻击方向角度来看，可以将联邦学习的攻击分为从内部发起和从外部发起两个方面。从攻击者的角色角度来看，可以将攻击分为参与方发起的攻击、中心服务器发起的攻击和第三方发起的攻击。从发动攻击

的方式角度来看，可以将攻击分为中毒攻击和拜占庭攻击。从攻击发起的阶段角度，可以将攻击分为模型训练过程的攻击和模型推断过程的攻击。在密码学领域，基于模型安全的假设通常可以被分为半诚实但好奇 (onest but curious) 的攻击方假设以及恶意攻击方假设。

### 1.3.1 攻击模型的研究现状

各类攻击模型阻碍了深度学习技术的发展，也会极大地威胁到人们的隐私敏感信息。无论是模型并行化还是数据并行化，分布式学习系统在用户数据隐私性方面相对于集中式学习存在一定的优势。但 [30] 发现，在分布式联邦学习系统中，参与者需要多次的联合迭代过程才能完成全局模型的收敛，参与者的参数也需要多次的训练、上传和共享，这些参数中包含的参与者训练集的相关信息，用户的信息可以通过计算用户上传的多个参数得到。

模型反演攻击<sup>[31][32]</sup> (Model Inversion Attack, MI) 利用这样的参数信息，以一种很简单的方式攻击用户数据：一旦用户的网络模型经过训练并达到收敛，攻击者就可以通过调整网络模型权重的梯度，获得网络模型中所有表示类的逆向工程试例。在模型反演攻击中，攻击者无需接触目标信息的标签类，攻击模型仍然能够恢复原始样本试例。这一攻击模型表明，任何经过精确训练的深度学习网络，无论是以何种方式进行训练收敛，都可以泄露深度网络中区分不同标签类的信息。但是参数中包含的信息有限，模型反演攻击方式很难攻击卷积神经网络等复杂深度网络模型，在模型进行了一定的隐私保护后，攻击也基本失效。

目前研究人员也利用诸多安全模型对深度学习网络的训练数据集进行保护，但 Hitaj 等人 [21] 发现，一个联邦学习框架非常容易受到系统内参与者发起的主动攻击。他们首先提出了一个由系统内的恶意用户发起的基于 GAN 的重建攻击。在训练阶段，攻击者可以冒充无辜的用户，训练 GAN 来模拟由其他用户的训练数据产生的原型样本。通过不断添加假的训练样本，攻击可以逐渐影响整个学习过程，使受害者暴露出更多关于攻击者的目标类的敏感信息。除了客户端发起的 GAN 攻击，服务器也能通过 GAN 攻击。恶意服务器最初假装是一个为用户提供联邦学习

服务的正常服务器，但其主要目标是重建被攻击用户的训练样本。

在联邦学习框架中，攻击者可能试图修改、删除或插入恶意信息到训练数据中，以破坏原始数据分布，改变学习算法的逻辑两种常见的中毒攻击的例子包括标签翻转攻击 [34] 和后门攻击 [19]。标签反转攻击是指恶意用户反转样本标签，并在训练数据中加入预定义的攻击点，导致训练后的模型偏离预测的界限。与标签反转攻击不同，后门要求攻击者用精心设计的训练数据，利用特定的隐藏模式来训练目标的深度神经网络（DNN）模型。这些模型被称为“反馈回路”，可以干扰学习模型，并在预测阶段产生与真实情况截然不同的结果。

如上文所述，联邦学习机制要求所有参与者通过在本地数据集上训练全局模型来更新梯度。在这种情况下，如果联邦学习系统有一个不被信任的服务器，其知识不能被信任，那么用户的私人信息就不能得到保证。这个不受信任的服务器可以获得关于每个参与者的本地训练模型的大量额外信息（例如，模型结构、用户身份和梯度），并且能够充分损害用户的隐私信息。具体实现如下：攻击者首先在平均化后获得模型的全局参数，并在本地存储这些快照。然后，通过计算以下快照与进一步

### 1.3.2 隐私保护的研究现状

在联邦学习中，存在着无数与隐私有关的挑战学习中的隐私问题。除了保证隐私之外，重要的是要保证确保通信成本的低廉和高效。有许多关于联合学习的隐私定义 [8][2][19]。我们可以把它们分为两类：局部隐私和全局隐私。在本地隐私中，每个客户端发送一个不同的隐私值，该值是安全的加密的到服务器。在全局模型中，服务器在最终输出中添加不同的隐私噪音。安全多方计算、同态加密和差分隐私是最常见的技术来保证联盟环境中的安全和隐私。

安全多方计算模型涉及多方，并在一个定义明确的模拟框架中提供安全证明，以保证完全的零知识，即每一方除了其输入和输出外一无所知。零知识是非常理想的，但这种理想的属性通常需要复杂的计算协议，而且可能无法有效实现。在某些情况下，如果提供安全保证，部分知识的披露可能被认为是可以接受的。有可能

在较低的安全要求下建立一个具有 SMC 的安全模型，以换取效率 [16]。最近，一项研究 [46] 将 SMC 框架用于训练具有两个服务器和半诚实假设的机器学习模型。在 [33] 中，MPC 协议被用于模型训练和验证，而用户不会泄露敏感数据。最先进的 SMC 框架之一是 Sharemind[8]。[44] 的作者提出了一个具有诚实多数的 3PC 模型 [5,21,45]，并考虑了半诚实和恶意假设的安全性。这些作品要求参与者的数据在非共存的服务器之间秘密共享。

同态加密是一种加密形式，它允许人们对密文进行特定形式的代数运算得到仍然是加密的结果，将其解密所得到的结果与对明文进行同样的运算结果一样同态加密 [53]，明文通过同态加密方法得到密文后，可实现密文间的计算（密文计算后解密的结果等价于明文计算的结果）。如果对密文进行加法（或乘法）运算后解密，与明文进行加法（或乘法）运算，结果相等，则称这种加密算法为加法（乘法）同态。如果同时满足加法和乘法同态，则称为全同态加密。在联邦学习中，因为只需要对中间结果或模型进行聚合，一般使用的同态加密算法为 PHE（多见为加法同态加密算法），通过加密机制下的参数交换来保护用户数据隐私 [24, 26, 48]，例如在 FATE 中使用的 Paillier 即为加法同态加密算法。

差分隐私方法涉及向数据添加噪音，或使用概括方法来掩盖某些敏感属性，直到第三方无法区分个人，从而使数据无法被还原以保护用户的隐私。利用差分隐私，可以在本地模型训练及全局模型整个过程中对相关参数进行扰动，从而令敌手无法获取真是模型参数，但是与密码学技术相比，差分隐私无法保证参数传递过程中的机密性，从而增加了模型遭受隐私攻击的可能性。例如刘俊旭等人 [10] 针对联邦学习下差分隐私中存在的攻击方法进行了详细的调研。在 [23] 中，作者为联合学习引入了一种差异化的隐私方法，以便通过在训练期间隐藏客户端的贡献来增加对客户端数据的保护。在深度学习中，差分隐私可以作为一种局部隐私保护方案来保护用户梯度的隐私，Abadi 等人 [43] 提出了一种隐私保护的深度学习方法，主要通过使用噪声来扰乱少量步骤后的局部梯度，将差分隐私机制与 SGD 算法相结合。令人担忧的是，隐私保护预算的成本和联合学习的有效性之间的权衡是困难

的，因为较高的隐私保护预算可能对一些大规模的攻击（如基于 GAN 的攻击）不是很有用 [50]，而较低的隐私保护预算可能阻碍模型的局部收敛。

总的来说，安全多方计算基于复杂的计算协议，同态加密的运算成本非常高，而差分隐私破坏了数据的可用性，很难在模型性能和隐私成本上达到平衡，当前的研究方向主要集中在对数据集和神经网络中的参数的加密和隐私保护机制上，较少关注到模型整体框架等过程。目前的联邦学习中的隐私保护方法还有许多不足，不能在隐私性和模型可用性上都达到一个相对满意的效果，此外，大部分方法是基于统一的、固定的参数设置，会导致模型迭代过程中累积大量隐私损失，使模型性能大幅下降。因此，在联邦学习场景下，保护用户隐私的同时保持模型准确性仍需大量的研究，

## 1.4 本文工作与主要贡献

针对联邦学习中隐私性和模型精度的双重指标，本文提出了参数匿名上传框架和自适应差分隐私算法，主要的工作和贡献包含以下三个方面：

- (1) 本文提出了一个新的参数聚合框架，该框架支持在参数上传过程中，对于每一个本地模型，通过两个重要实现：拆分和混洗，扰乱模型中各个参数的隐私关联和各个模型之间的隐私关联，实现客户端匿名。
- (2) 本文提出了一个自适应扰动方案，对联邦学习过程中双方所交互的梯度进行分析，在所交互的梯度上添加扰动，并基于梯度自适应加噪，进一步减少隐私预算。
- (3) 本文针对模型训练和上传过程中的隐私安全问题，将改进的参数聚合框架和自适应扰动方案引入联邦学习框架，实现混合隐私保护的联邦学习系统，每个用户在本地训练数据时添加自适应扰动，并在向中心服务器上传时实现客户端匿名，实现了客户端的数据隐私。

- (3) 本文通过实验，展示了自适应扰动方案和参数聚合框架的结合，使得联邦学习的模型的精度和隐私预算达到平衡。

## 1.5 本文组织结构

本文一共六章，主要内容的组织安排如下：

第一章对本文研究内容：联邦学习的研究背景和实际意义进行了阐述，介绍了目前联邦学习中的隐私保护的研究现状和发展方向。

第二章详细介绍本文研究内容所涉及的一些理论基础与背景知识，包含了联邦学习的相关概念，差分隐私的基础知识。

第三章描述了本文所提出的参数聚合框架的设计和实现。我们首先对框架的整体进行了介绍，之后给出了各个模块的设计和实现细节。

第四章描述了本文所提出的自适应扰动方案，讨论了隐私预算与的关系，并且详细描述了相关概念和算法。

第五章为实验部分，基于本文提出的隐私保护框架，我们在三个基准数据集的进行了实验和讨论。

第六章是对本文的一个内容总结和展望，首先对本文的研究内容进行了概括，并对现有的不足进行总结，对未来的研宄和改进方向进行了展望。

## 第二章 相关知识介绍

我们在本章节中介绍了本文研究所需要的一些基本知识，有助于更好的理解之后章节的内容。

### 2.1 联邦学习

传统的集中式深度学习需要将训练数据放在一起，交给一个数据中心。模型是以集中的方式进行训练的。而联合学习允许数据所有者持有一个私人学习网络，用本地数据集进行训练。之后，每个参与者将本地模型的梯度上传到云服务器。通过更新云服务器上收集的全局梯度，可以避免本地模型的过度拟合。此外，它还可以保护本地数据不被其他参与者或云服务器直接知道。

#### 2.1.1 基本定义

#### 2.1.2 联邦学习的分类

联合学习通常分为水平联邦学习、纵向联邦学习和联合迁移学习，这是由 Yang Q 等人提出的 [8]。这种分类是基于用户维度和特征维度的重合。

- **水平联邦学习：**当两个数据集的用户属性重叠较多而用户重叠较少的情况下，我们对数据集进行横向切割（即按用户维度切割），删除两边用户属性相同但用户不完全相同的那部分数据，用于训练。这种方法被称为横向联合学习。例如，两家银行位于不同的地区，有来自各自地区的用户群，而且它们之间的联系非常少。然而，他们的业务活动非常相似，因此他们的用户特征也是一样的。在这个阶段，我们可以使用跨部门的联合学习来建立一个联合模型。2016 年，谷歌提出了一个在安卓手机上更新模型的联合数据建模系统：模型

参数在本地不断更新，并在各个用户使用安卓手机时上传到安卓云端，使拥有数据的每一方都能建立一个具有相同特征维度的联合模型。

- **纵向联邦学习：**在两个数据集与用户重叠较多而与用户属性重叠较少的情况下，我们将数据集纵向切开（即按特征维度），选择数据集中两边用户相同但用户属性不完全相同的部分进行训练。这种方法被称为纵向的联合学习。例如，有两个不同的组织，一个是在一个地方的银行，另一个是在同一个地方的电子商务公司。他们的用户群很可能包括该地的大部分人口，所以有很大的用户交集。然而，由于银行储存的是用户的收入和支出以及信用评分的数据，而电子商务公司储存的是用户的浏览和购买历史的数据，他们的用户档案并没有那么紧密的联系。长期的联合学习是在一个加密的空间里将这些不同的功能结合起来，以提高模型的性能。渐渐地，人们发现可以在这个联合系统之上建立若干机器学习模型，如逻辑回归、树状结构和神经网络模型。
- **联合迁移学习：**联合迁移学习是通过使用迁移学习景观来弥补数据或标签的差距，而不是对数据进行切分，两个数据集中的用户和用户属性几乎没有重叠。这种方法被称为混合式学习迁移。作为一个例子，考虑两个不同的组织，一个是中国的银行，另一个是美国的电子商务公司。由于地理上的限制，这两个机构的用户群重叠的地方很少。由于它们是不同类型的组织，数据的特点也没有太多的重叠。在这种情况下，为了保证有效的联邦学习，有必要引入反式学习，以克服单变量数据量小和标注样本小的问题，提高模型的效率。

### 2.1.3 联邦学习的训练步骤

在很多横向联邦学习应用场景中，参与训练的参与方数据具有类似的数据结构（特征空间），但是每个参与方拥有的用户是不相同的。有时参与方比较少，例如，银行系统在不同地区的两个分行需要实现联邦学习的联合模型训练；有时参与方会非常多，例如，做一个基于手机模型的智能系统，每一个手机的拥有者将会是一个独立的参与方。针对这类联合建模需求，可以通过一种基于服务器客户端的架构

来满足很多横向联邦学习的需求。将每一个参与方看作一个客户端，然后引入一个大家信任的服务器来帮助完成联邦学习的联合建模需求。在联合训练的过程中，被训练的数据将会被保存在每一个客户端本地，同时，所有的客户端可以一起参与训练一个共享的全局模型，最终所有的客户端可以一起享用联合训练完成的全局模型。

- 步骤 1: 中心服务器初始化联合训练模型，并且将初始参数传递给每一个客户端。
- 步骤 2: 客户端用本地数据和收到的初始化模型参数进行模型训练。具体步骤包括: 计算训练梯度，使用加密、差分隐私等加密技术掩饰所选梯度，并将加密后的结果发送到服务器。
- 步骤 3: 服务器执行安全聚合。服务器只收到加密的模型参数，不会了解任何客户端的数据信息，实现隐私保护。服务器将安全聚合后的结果发送给客户端。
- 步骤 4: 参与方用解密的梯度信息更新各自的本地模型，具体方法重复步骤 2。

#### 2.1.4 联邦学习中的安全和隐私威胁

尽管联邦学习提供了隐私保护的机制，还是有各种类型的攻击方式可以攻击联邦学习系统，从而破坏联邦学习系统安全和参与方的隐私。本节将讨论关于联邦学习的攻击问题。从参与方的类型来看，可以将联邦学习的威胁模型细分为半诚实模型 (semi-honest model) 和恶意模型。对于联邦学习系统的攻击，本文按照不同的维度进行不同层次的分类。从攻击方向角度来看，可以将联邦学习的攻击分为从内部发起和从外部发起两个方面。从攻击者的角色角度来看，可以将攻击分为参与方发起的攻击、中心服务器发起的攻击和第三方发起的攻击。从发动攻击的方式角度来看，可以将攻击分为中毒攻击和拜占庭攻击。从攻击发起的阶段角度来看，可以将攻击分为模型训练过程的攻击和模型推断过程的攻击。在密码学领域，

基于模型安全的假设通常可以被分为半诚实但好奇 (onest but curious) 的攻击方假设以及恶意攻击方假设。123123123123

- **半诚实但好奇的攻击方：**半诚实但好奇的攻击方假设也被称为被动攻击方假设。被动攻击方会在遵守联邦学习的密码安全协议的基础上，试图从协议执行过程中产生的中间结果推断或者提取出其他参与方的隐私数据。半诚实但好奇的供给方通常是客户端的角色，它们可以检测从服务器接收的所有消息，但是不能私自修改训练的过程。在一些情况下，安全包围或者可信执行环境 (trusted execution environment, TEE) 等安全计算技术的引入，可以在一定程度上限制此类攻击者的影响或者信息的可见性。半诚实但好奇的参与方将很难从服务器传输回来的参数中推断出其他参与方的隐私信息，从而威胁程度被削弱。
- **恶意攻击方：**恶意攻击方也称为主动攻击方。由于恶意攻击方不会遵守任何协议，为了达到获取隐私数据的目的，可以采取任何攻击手段，例如破坏协议的公平性、阻止协议的正常执行、拒绝参与协议、不按照协议恶意替换自己的输入、提前终止协议等方式，这些都会严重影响整个联邦学习协议的设计以及训练的完成情况。恶意的参与方可以是客户端，也可以是服务器，还可以是恶意的分析师或者恶意的模型工程师。恶意客户端可以获取联邦建模过程中所有参与方通信传输的模型参数，并且进行任意修改攻击。恶意服务器可以检测每次从客户端发送过来的更新模型参数，不按照协议，随意修改训练过程，从而发动攻击。恶意的分析师或者恶意的模型工程师可以访问联邦学习系统的输入和输出，并且进行各种恶意攻击。
- **成员推理攻击：**如上文所述，联邦学习机制要求所有参与者通过在本地数据集上训练全局模型来更新梯度。在这种情况下，如果联邦学习系统有一个不被信任的服务器，其知识不能被信任，那么用户的私人信息就不能得到保证。这个不受信任的服务器可以获得关于每个参与者的本地训练模型的大量额外

信息（例如，模型结构、用户身份和梯度），并且能够充分损害用户的隐私信息。具体实现如下：攻击者首先在平均化后获得模型的全局参数，并在本地存储这些快照。然后，通过计算以下快照与进一步删除添加的更新，以获得其他用户的模型的汇总更新。通过这种方式，攻击者可以利用数据集的协助，得出所有其他参与者共同合作的数据样本。

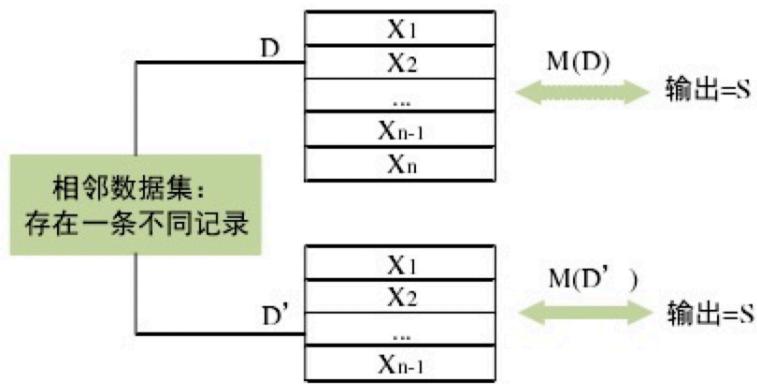
- **GAN 攻击：**Hitaj 等人 [21] 发现，一个联邦学习框架非常容易受到系统内参与者发起的主动攻击。他们首先提出了一个由系统内的恶意用户发起的基于 GAN 的重建攻击。在训练阶段，攻击者可以冒充无害的用户，训练 GAN 来模拟由其他用户的训练数据产生的原型样本。通过不断添加假的训练样本，攻击可以逐渐影响整个学习过程，使受害者暴露出更多关于攻击者的目标类的敏感信息。除了客户端发起的 GAN 攻击，服务器也能通过 GAN 攻击。恶意服务器最初假装是一个为用户提供联邦学习服务的正常服务器，但其主要目标是重建被攻击用户的训练样本。

## 2.2 差分隐私

差异化隐私作为一种隐私保护方法是为一个用户服务的，因为根据隐私的定义，隐私泄露只是与特定用户有关的信息泄露，而一组用户的统计特征不包括在隐私信息中。如果一个对象在数据库中的存在或不存在，或其价值的变化不会对搜索结果产生重大影响，那么该对象的隐私信息就会受到保护，这就是差异性隐私（DP）概念的起源。差异隐私首先被应用于数据查询，为了更好地说明数据集之间的差异，定义了相邻数据集的概念：两个数据集只差一个信息或只差一个数值不同的记录。因此，查询数据库相关信息的攻击者将无法以任何概率确定  $X_n$  是否存在于数据集中，而成员  $X_n$  被认为是相对安全的。

### 2.2.1 基本定义

对于一个有限域  $Z$ ,  $z \in Z$  为  $Z$  中的元素, 从  $Z$  中抽样所得  $z$  的集合组成数据集  $D$ , 其样本量为  $n$ , 属性的个数为维度  $d$ 。对数据集  $D$  的各种映射函数被定义为查询 (Query), 用  $F = \{f_1, f_2, \dots\}$  来表示一组查询, 算法  $M$  对查询  $F$  的结果进行处理, 使之满足隐私保护的条件, 此过程称为隐私保护机制。设数据集  $D$  和  $D'$ , 具有相同的属性结构, 两者的对称差记作  $D\Delta D'$ ,  $|D\Delta D'|$  表示  $D\Delta D'$  中记录的数量。若  $|D\Delta D'| = 1$ , 则称  $D$  和  $D'$  为邻近数据集 (Adjacent Dataset)。



**定义 2.2.1 (成立条件).** 若随机算法  $M : D \rightarrow R$  满足  $(\varepsilon, \delta) - DP$ , 当且仅当相邻数据集  $d, d'$  对于算法  $M$  的所有可能输出子集  $S \in R$  满足不等式<sup>[40]</sup> :

$$\Pr[M(d) \in S] \leq e^\varepsilon \Pr[M(d') \in S] + \delta$$

其中,  $\varepsilon$  表示隐私预算参数,  $\varepsilon$  越小意味着隐私预算越低, 信息泄露越少, 隐私保护的强度越高。添加项  $\delta$  代表允许以概率  $\delta$  打破  $\varepsilon - DP$  的可能性, 其值通常选择为小于  $1/|D|$ . 当  $\delta = 0$  时, 定义转化为  $\varepsilon - DP$ , 这时机制提供了更加严格的隐私保护。隐私预算参数决定着隐私保护强度, 针对传统数据库保护, 当  $\varepsilon \in (0, 1)$  时认为隐私保护强度是有效的, 但应用在深度学习领域,  $\varepsilon \in (0, 10)$  都认为是可以被接受的合理范围。如图 1 所示, 算法  $M$  通过对输出结果的随机化来提供隐私保护, 同时通过参数  $\varepsilon$  来保证在数据集中删除任一记录时, 算法输出同一结果的概率不发生显著变化。

### 2.2.2 相关概念

差分隐私保护可以通过在查询函数的返回值中加入适量的干扰噪声来实现。加入噪声过多会影响结果的可用性，过少则无法提供足够的安全保障。敏感度是决定加入噪声量大小的关键参数，它指删除数据集中任一记录对查询结果造成最大改变。在差分隐私保护方法中定义了两种敏感度，即全局敏感度（Global Sensitivity）和局部敏感度（Local Sensitivity）。

**定义 2.2.2** (全局敏感度). 设有函数  $f : D \rightarrow R^d$ , 输入为一数据集, 输出为一  $d$  维实数向量. 对于任意的邻近数据集  $D$  和  $D'$ ,

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

称为函数  $f$  的全局敏感度。函数的全局敏感度由函数本身决定，不同的函数会有不同的全局敏感度。一些函数具有较小的全局敏感度（例如计数函数，其全局敏感度为 1），因此只需加入少量噪声即可掩盖因一个记录被删除对查询结果所产生的影响，实现差分隐私保护。

**定义 2.2.3** (局部敏感度). 对于一个查询函数  $f : D \rightarrow R^d$ , 其中  $D$  为一个数据集,  $R^d$  为  $d$  维实数向量, 是查询的返回结果。对于给定的数据集  $D$  和它的任意邻近数据集  $D'$ , 有  $f$  在  $D$  上的局部敏感度为:  $LS_f(D) = \max_{D'} \|f(D) - f(D')\|_1$

局部敏感度由函数和给定数据集中的具体数据共同决定。由于利用了数据集的数据分布特征，局部敏感度通常要比全局敏感度小得多。敏感度代表了查询函数针对相邻数据集的输出的最大不同，或者说量化评估了最坏情况下单个样本对整体数据带来的不确定性大小。敏感度函数仅与查询函数的类型有关，为扰动的添加提供了依据。但是，由于局部敏感度在一定程度上体现了数据集的数据分布特征，如果直接应用局部敏感度来计算噪声量则会泄露数据集中的敏感信息。

全局差分隐私技术旨在实现这样一个目标：如果替换数据集中的任意样本的效果足够小，则查询结果不能被用来探索数据集中任何样本的更多信息 [43]。作为

一种优势，这种技术比局部差分隐私技术更准确，因为它不需要向数据集添加大量的噪声。局部差分隐私技术被引入以去除全局差分隐私中所要求的受信任的中央机构 [34,102]。与全局差分隐私技术相比，局部差分隐私技术不需要可信的第三方 [146]。其缺点是，噪声总量比全局差分隐私技术大得多。

在差分隐私部署过程中常常不仅仅在一处添加噪声，也仅仅针对数据集进隐私预算的分配有序列组合性 [41] 和并行组合性<sup>[42]</sup> 两种组合特性：

**定义 2.2.4 (序列组合).** 给定  $\mathbf{n}$  个随机算法  $M_i (1 \leq i \leq n)$  满足  $\varepsilon_i - DP$ ，那么针对一个数据库  $D$  而言，在  $D$  上的算法序列组合可以提供  $\varepsilon - DP$ ，其中  $\sum_{i=1}^n \varepsilon_i = \varepsilon$ 。

**定义 2.2.5(并行组合).** 对于数据库  $D$ ，当其被划分成  $n$  个不相交的子集  $\{D_1, D_2, \dots, D_n\}$ ，在每个子集上应用算法  $M_i$ ，每个算法提供  $\varepsilon_i - DP$ ，则在序列  $\{D_1, D_2, \dots, D_n\}$  上整体满足  $(\max \{\varepsilon_1, \dots, \varepsilon_n\}) - DP$

### 2.2.3 实现机制

在实践中为了使一个算法满足差分隐私保护的要求，对不同的问题有不同的实现方法，这些实现方法称为“机制”。拉普拉斯机制 (Laplace Mechanism)、指数机制 [22](ExponentialMechanism) 与高斯机制是三种最基础的差分隐私保护实现机制。其中，Laplace 机制和高斯适用于对数值型结果的保护，指数机制则适用于非数值型结果。

在中心化差分隐私中，最为常用的扰动机制是拉普拉斯 (Laplace) 机制，该机制可以后期处理聚合查询（例如，计数、总和和均值）的结果以使它们差分私有。Laplace 分布是统计学中的概念，是一种连续的概率分布。

**定义 2.2.6 (拉普拉斯机制).** 如果随机变量的概率密度函数分布为：

$$f(x | \mu, b) = \frac{1}{2b} \exp\left(-\frac{|x-\mu|}{b}\right) = \frac{1}{2b} \begin{cases} \exp\left(-\frac{\mu-x}{b}\right) & x < \mu \\ \exp\left(-\frac{x-\mu}{b}\right) & x \geq \mu \end{cases}$$

其中， $D$  表示数据集， $f(D)$  表示的是查询函数， $Y$  表示的是 Laplace 随机噪声， $M(D)$

表示的是最后的返回结果。 $M(D) = f(D) + Y$  如果噪声  $Y \sim L(0, \frac{\Delta f}{\epsilon})$  满足  $(\epsilon, 0)-$ ，

则表示服从拉普拉斯分布的随机噪声。因此，当隐私预算  $\mathbf{3}$  确定时，敏感度越大，引入的噪声量越大。

对于非数值型的查询结果或数据，通常使用指数机制来随机选择离散的输出结果来满足差分隐私。指数机制整体的思想就是，当接收到一个查询之后，不是确定性的输出一个  $R_i$  结果，而是以一定的概率值返回结果，从而实现差分隐私。而这个概率值则是由打分函数确定，得分高的输出概率高，得分低的输出概率低。

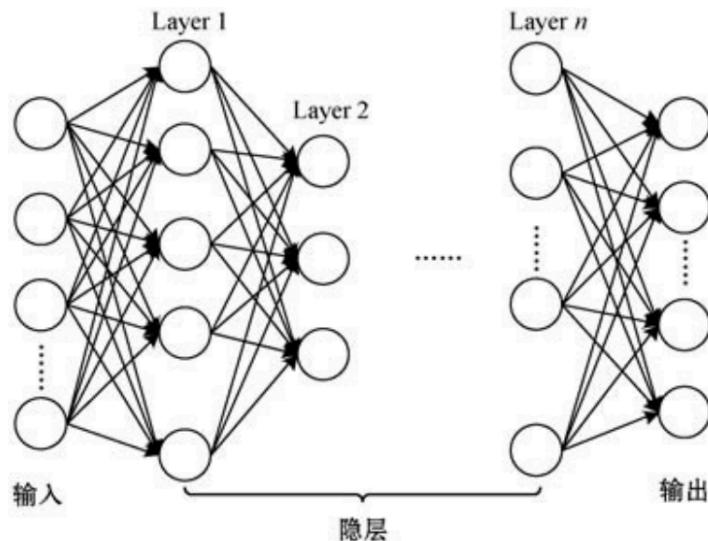
**定义 2.2.7 (指数机制).** 指数机制满足差分隐私，如果：

$$M(D) = (\text{return } \varphi \propto \exp\left(\frac{\varepsilon q(D, \varphi)}{2\Delta q}\right))$$

评分函数  $q(D, \varphi)$ ，用于评估输出  $\varphi$  的质量。 $\Delta q$  代表了输出的敏感度。

### 2.3 神经网络

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



深度神经网络算法中，为评估所提神经网络输出预测值与真实值之间的差异程

度, 用损失函数  $L$  表示, 文中采用均方差损失函数, 表示为:

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

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

## 2.4 本章小结

## 第三章 联邦学习的自适应加噪机制

最近研究表明深度神经网络容易受到对抗样本的攻击。为了解决这个问题，一些工作通过向图像中添加高斯噪声来训练网络，从而提高网络防御对抗样本的能力，但是该方法在添加噪声时并没有考虑到神经网络对图像中不同区域的敏感性是不同的。针对这一问题，提出了梯度指导噪声添加的对抗训练算法。该算法在训练网络时，根据图像中不同区域的敏感性向其添加自适应的噪声，在敏感性较大的区域上添加较大的噪声，抑制网络对图像变化的敏感程度，在敏感性较小的区域上添加较小的噪声，提高其分类精度。提出一种基于数据差分隐私保护的随机梯度下降算法。引入范数剪切与附加高斯噪声操作，对传统梯度更新策略进行改进。为衡量每次迭代过程中对数据隐私性的破坏，提出隐私损失累积函数在迭代过程中对数据隐私性的侵犯程度进行度量。

### 3.1 模型概况

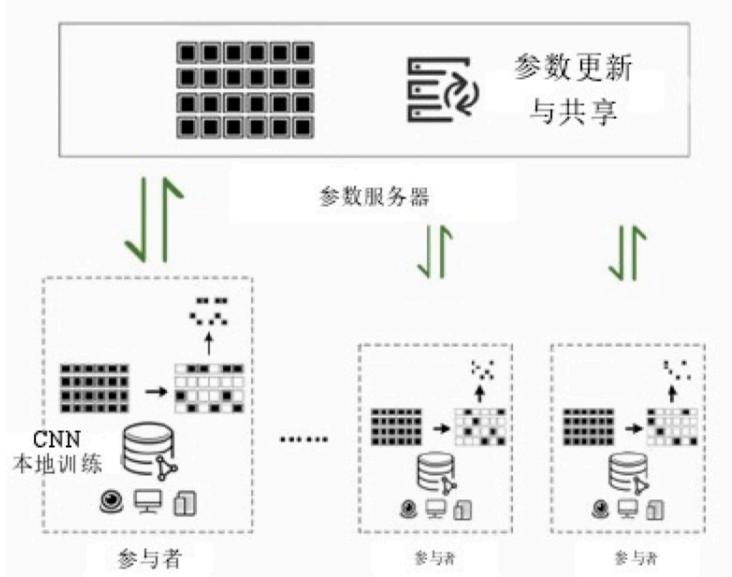
#### 3.1.1 系统架构

如图??所示，在我们的系统模型中，有两方，即云服务器和用户。

**云服务器：**云服务器事先与用户协商一个网络框架。然后，服务器通过公共数据训练一个初始模型，然后将初始模型的参数广播给用户。用户在本地训练各自的模型后，云服务器收集用户发送的模型梯度，并更新全球模型。

**用户：**用户下载由云服务器初始化的模型参数。然后，每个用户在本地数据集上训练私人模型。最后，用户将本地模型的扰动梯度发送到云服务器。

每个参与者在第一次进入联邦学习系统时，都会初始化参数。针对统一的学



习目标，在本地训练集上进行模型的训练。联邦学习系统同样包括参数交换协议，在参数交换协议下，参与者将本地所得神经网络梯度的参数上传至参数服务器，同样通过参数服务器下载最新的全局参数值至本地继续训练。参与者可以在本地独立训练时，避免了使用局限训练集的单个本地模型的过拟合。模型经过训练之后，每个参与者都可以使用新的测试集独立且隐私的对其进行评估与测试，无需再进行交互操作。

### 3.1.2 本地训练

在分布式联邦学习中，第  $i$  个参与者在本地将会对全局神经网络参数的一个局部向量  $w^i$  进行维护、学习和更新称之为本地训练。参数服务器负责对全局参数向量  $w^{global}$  进行维护和更新。每个参与者在开始训练时可以随机初始化本地参数，也可以从参数服务器下载其参数最新值。

每个参与者都会使用统一标准的神经网络算法训练模型，使用的神经网络算法不局限于简单深度神经网络与卷积深度神经网络，但所有参与者需要进行统一，本文使用的是采用选择性随机梯度下降算法全连接层的卷积神经网络 CNN。本地模型网络多次迭代训练其本地训练集。在本地训练期间，不同参与者之间不需要额外的共享样本和交互，他们通过参数服务器通过参数共享间接影响彼此的训练结果。

**Algorithm 1** 联邦学习客户端本地训练算法

---

```

1: Input: 全局模型参数  $\mathbf{w}^{\text{global}}$ , 初始化参数  $\mathbf{w}^i$ 
2: Enable users for training: initialize model//初始化模型
3: for ( $epoch = 1$  to  $n$ ) do
4: Download parameters  $\theta_d$  from PS
5: Run  $CNN$  on local dataset
6: Update the  $\mathbf{w}^i$  according to (2 – 5)
7: Compute  $\Delta\mathbf{w}^i$ 
8: Upload  $\Delta\mathbf{w}_s^i$  to PS
9: end

```

---

算法1描述了参与者在进行本地训练时具体步骤。每个参与者独立进行深度神  
学习训练, 在每个训练阶段由五个步骤组成。在初始化之后, 第  $i$  个参与者从参数服  
务器 (Parameter Server, PS) 中下载了最新参数的分量  $\theta_d$ , 将下载的值覆盖至其本  
地参数, 之后会在本地训练数据集上训练神经网络。

在算法的第 6 步中, 参与者计算全连接层算法训练局部参数变化得到梯度向  
量  $\Delta\mathbf{w}^i$ 。参数  $\Delta\mathbf{w}^i$  反映了对于第  $i$  个参与者, 每个神经元中的权重向量需要变化  
多少能够得到更精确的模型。 $\Delta\mathbf{w}^i$  的参数信息正是其他参与者需要训练更好模型  
以及避免的本地数据过拟合的信息。 $\Delta\mathbf{w}_s^i$  表示经过选择后上传的参数。在上传训  
练结果前, 选择一个大于阈值  $T$  的子集替代完整的参数向量, 参与者选择上传更有  
助于目标函数的梯度值, 可以使得训练迭代过程收敛更快, 模型精度更高, 以及陷入  
局部最优的可能性更小。

在本地训练时, 卷积神经网络的全连接层采用了选择性随机梯度下降算法。  
Shokri 在 [16] 中证明了其与传统的随机梯度下降算法有着几乎相同的准确性。原  
因是选择参数上传更新全局模型与传统随机梯度下降算法求最优值的原因相同, 选  
择的过程增加了最优化过程的随机性。

参与者单独训练模型时, 由于训练集的多次使用与缺少更新, 很容易陷入局部  
最优。在训练本地模型时, 参与者使用梯度参数的子集对模型进行更新, 会增加模  
型优化过程中的随机性, 很大程度上避免了本地 SGD 过多使用相同的小样本集产  
生的模型过拟合。使用其他参与者用在不同数据集上训练学习的值覆盖本地学习

的参数, 可以帮助每个参与者跳出局部最优, 从而得到更准确的模型。

### 3.1.3 全局参数更新

联邦学习通过协调深度学习任务, 建立统一的深度学习模型结构后, 参数服务器会初始化全局参数  $w^{global}$ 。之后处理系统内参与者的上传和下载请求, 存储参与者的局部参数, 并计算更新全局参数  $w^{global}$ 。当参与者上传参数时, 参数服务器会将上传的  $\Delta\mathbf{w}_s^i$  的值添加至相应的全局参数中, 并为每个全局权重参数更新元数据和计数器  $stat$ 。具体更新规则如下:

对于所有的  $j \in S$ :

$$w^{global} := w^{global} + \Delta\mathbf{w}_j^i$$

为了增加更新的参数的权重, 服务器可以周期性地将计数器乘以衰减因子  $\beta$ , 即:

$$stat := \beta \cdot stat$$

当参与者从服务器获取具有最大统计值参数的最新值时, 将在下载期间使用这些统计信息。每个参与者都可以通过设置  $\theta_d$  决定下载这些参数的某一部分。

### 3.1.4 威胁模型

我们认为云服务器是一个“诚实但好奇”的实体。也就是说, 服务器将遵循与所有用户的协议。然而, 通过利用完全访问用户梯度的便利, 它也试图在训练过程中获得额外的信息。出于这个原因, 我们提出的自适应加噪机制目的是保护发送到服务器的本地梯度不被推断出任何关于用户的额外信息, 并且维持原有模型的精度。

### 3.1.5 联邦学习中的差分隐私

传统的联邦学习中使用差分隐私的主要流程如下所示:

- 本地计算: 客户端  $i$  根据本地数据库  $\mathcal{D}_i$  和接受的服务器的全局模型  $\mathbf{w}_G^t$  作为本地的参数, 即  $\mathbf{w}_i^t = \mathbf{w}_G^t$ , 进行梯度下降策略进行本地模型训练得到  $\mathbf{w}_i^{t+1}$  ( $t$

表示当前 round)。

- 模型扰动: 每个客户端产生一个随机噪音  $\mathbf{n}$ ,  $\mathbf{n}$  是符合高斯分布的, 使用  $\bar{\mathbf{w}}_i^{t+1} = \mathbf{w}_i^{t+1} + \mathbf{n}$  扰动本地模型 (这里注意  $\mathbf{w}$  是一个矩阵, 那么  $\mathbf{n}$  就对矩阵的每一个元素产生噪音)。
- 模型聚合: 服务器使用 FedAVG 算法聚合从客户端收到的  $\bar{\mathbf{w}}_i^{t+1}$  得到新的全局模型参数  $\mathbf{w}_G^{t+1}$ , 也就是扰动过的模型参数。
- 模型广播: 服务器将新的模型参数广播给每个客户端。
- 本地模型更新: 每个客户端接受新的模型参数, 重新进行本地计算。

## 3.2 方案设计

### 3.2.1 自适应噪声添加

在第二章中介绍了关于神经网络的结构,

$$y = a(\mathbf{x} * \omega + b)$$

是学习模型中每个隐藏神经元的转化过程。其中  $\mathbf{x}$  代表输入向量,  $y$  是输出,  $b$  和  $\omega$  分别代表偏置项和权重矩阵。 $a()$  是一个激活函数, 用于结合线性变换和非线性变换。 $z(\omega) = \mathbf{x} * \omega + b$  是线性变换部分。

由于神经网络的结构, 上一层的输出是下一层的输入, 由此我们可以得出, 原始数据只被第一隐层的线性变换所利用。直观地说, 为了得到一个具有隐私保护的学习模型, 我们可以在第一层隐藏层的数据中注入噪声。正如 Phan 等人 [15] 提到的, 对于线性变换有一种传统的方法, 即向原始数据注入具有相同隐私预算的噪声, 但是这容易导致隐私预算增加, 并且使原始数据失真过多。因此, 本文提出一种自适应噪声添加算法, 针对每个梯度计算其贡献值, 根据贡献值进行梯度裁剪并添加噪声。

首先，引入了两个调整因素。其中， $f$  代表一个阈值，用于决定属性对模型结果输出的贡献是高还是低，其值由用户定义，即贡献超过阈值  $f$  的属性类对输出的贡献更大。然后，我们向所有这些属性注入自适应拉普拉斯噪声。当贡献率低于阈值  $f$  时，对这些属性进行概率选择。也就是说，我们选择概率为  $1 - p$  的原始数据，并对一些概率为  $p$  的属性注入自适应拉普拉斯噪声。该公式如下。

$$\tilde{x}_{i,j} = \begin{cases} \ddot{x}_{i,j} & \beta \geq f \\ \bar{x}_{i,j} & \beta < f \end{cases}$$

其中 Beta 代表贡献率： $\beta = \frac{|\tilde{C}_j|}{\sum_{j=1}^u |\tilde{C}_j|}$ ，当  $\beta < f$  时，我们有：

$$\tilde{x}_{i,j} = \begin{cases} \ddot{x}_{i,j} & \beta \geq f \\ \bar{x}_{i,j} & \beta < f \end{cases}$$

$f$  和  $p$  是超参数，用户可以根据自己的情况来调整。

每个属性类的隐私预算比率  $\epsilon_j$  由。也就是说，隐私预算  $\epsilon_l$  是根据贡献率： $\epsilon_j = \frac{u * |\tilde{C}_j|}{\sum_{j=1}^u |\tilde{C}_j|} * \epsilon_l$  按比例分配给每个属性类。自适应噪声按以下方式注入属性中：

$$x'_{i,j} = x_{i,j} + \frac{1}{|D_i^t|} \text{Lap} \left( \frac{GS_l}{\epsilon_j} \right)$$

在不丧失一般性的情况下，调整因子  $f$  和  $p$  的值与系统的准确性和隐私水平有关。即  $f$  越小， $p$  越大。越高的秘密水平，准确性越低，反之亦然。

我们用层间相关性传播（LRP）算法将输出分解到每一层。关于 LRP 算法的更多细节，我们将在以下部分进行介绍。每个用户都在本地对原始数据进行训练前馈操作，这可以获得一个新的数据操作，从而获得本地模型的输出。根据相邻层之间的线性关系，在  $k - th$  层的神经元的贡献  $C_{a_i}^{l_k}(x_i)$  等于连接到神经元  $a_i$  的相邻层的贡献之和：

$$C_{a_i}^{l_k}(x_i) = \sum_{a_j \in l_{k+1}} C_{a_i \leftarrow a_j}^{l_k \leftarrow l_{k+1}}(x_i)$$

例如, 如图 2 所示, 我们有:

$$C_{a_7}^{l_2}(x_i) = \sum_{a_j \in l_3} C_{a_7 \leftarrow a_j}^{l_2 \leftarrow l_3}(x_i) = C_{a_7 \leftarrow a_8}^{l_2 \leftarrow l_3}(x_i) + C_{a_7 \leftarrow a_9}^{l_2 \leftarrow l_3}(x_i)$$

其中, ”*leftarrow*” 表示两部分之间的连接关系。” $l_{23}$ ” 是指深度神经网络(DNNs) 中  $2 - th$  层和第 3 层之间相邻层的连接关系。当  $k - th$  层为输出层时, 我们有:

$$C_{a_i}^{l_k}(x_i) = f(x_i, \omega_i^r)$$

### 3.2.2 隐私性证明

随机隐私保护调整技术 (RPAT) 对第 3.2.1 节中讨论的线性变换函数进行了扰动, 该函数满足  $\text{left}(\epsilon_c + \epsilon_l \text{right})$  差分隐私。证明如下。假设两个相邻的批次  $D_i^t$  和  $D_i^{t prime}$ , 其最后一个元组  $x_n$  和  $x_n^{prime}$  不同,  $z(D_i^t)$  和  $z(D_i^{t'})$  分别为线性变换函数。RPAT 满足  $(\epsilon_c + \epsilon_l)$  的差分隐私。

证明. 一般来说, 我们把偏置项视为第一类数据属性, 即:  $x_{i,0} = b_i$ 。线性转换可以改写为:  $\ddot{\mathbf{z}}_{x \in D_i^t}(\omega) = \ddot{\mathbf{x}} * \omega$ 。线性变换的敏感性  $GS_l$  如下:

$$\begin{aligned} GS_l &= \sum_{a_i \in l_1} \sum_{j=1}^u \left\| \sum_{x_i \in D_i^t} x_{i,j} - \sum_{x'_i \in D_i^{t'}} x'_{i,j} \right\|_1 \\ &= \sum_{a_i \in l_1} \sum_{j=1}^u \|x_{n,j} - x'_{n,j}\|_1 \\ &\leq \sum_{a_i \in l_1} \sum_{j=1}^u \max_{x_i \in D_i^t} \|x_{n,j}\|_1 \\ &\leq \sum_{a_i \in l_1} u \end{aligned}$$

其中,  $a_i \in l_1$  是指第一隐藏层  $l_1$  中的神经元  $a_i$ ,  $u$  是数据元组  $x_i \in D_i^t$  中的属性数。我们设计了 RPAT, 它包括两个调整因素。 $f$  和  $p$ , 它们可以过滤多余的噪声。RPAT 之后的属性的一般表达式如下:

$$\begin{aligned}
\tilde{x}_{i,j} &= [(1-f) + f * p] * \ddot{x}_{i,j} + f * (1-p) * x_{i,j} \\
&= [(1-f) + f * p] \left[ x_{i,j} + \text{Lap} \left( \frac{GS_l}{\epsilon_j} \right) \right] + [f * (1-p)] x_{i,j} \\
&= x_{i,j} + [(1-f) + f * p] \left[ \text{Lap} \left( \frac{GS_l}{\epsilon_j} \right) \right]
\end{aligned}$$

然后我们可以得到：

$$\begin{aligned}
\frac{\Pr(\ddot{\mathbf{z}}_{D_i^t}(\omega))}{\Pr(\ddot{\mathbf{z}}_{D_i^{t'}}(\omega))} &= \frac{\prod_{a_i \in l_1} \prod_{j=1}^u \exp \left( \frac{\epsilon_j \left\| \sum_{x_i \in D_i^t} x_{i,j} - \sum_{x_i \in D_i^{t'}} \tilde{x}_{i,j} \right\|_1}{GS_l} \right)}{\prod_{a_i \in l_1} \prod_{j=1}^u \exp \left( \frac{\epsilon_j \left\| \sum_{x'_i \in D_i^{t'}} x'_{i,j} - \sum_{x'_i \in D_i^t} \tilde{x}'_{i,j} \right\|_1}{GS_l} \right)} \\
&\leq \prod_{a_i \in l_1} \prod_{j=0}^u \exp \left( \frac{\epsilon_j}{GS_l} \left\| \sum_{x_i \in D_i^t} x_{i,j} - \sum_{x'_i \in D_i^{t'}} x'_{i,j} \right\|_1 \right) \\
&\leq \prod_{a_i \in l_1} \prod_{j=0}^u \exp \left( \frac{\epsilon_j}{GS_l} \max_{x_i \in D_i^t} \|x_{n,j}\|_1 \right) \\
&\leq \exp \left( \epsilon_l \frac{\sum_{a_i \in l_1} u \left[ \sum_{j=1}^u \frac{|\tilde{C}_j|}{\sum_{j=1}^u |\tilde{C}_j|} \right]}{GS_l} \right) \\
&= \exp(\epsilon_l)
\end{aligned}$$

□

根据上述推倒证明可知，在联邦学习的神经网络中添加自适应噪声后，所上传的梯度是满足  $(\epsilon_c + \epsilon_l)$  差分隐私的。在满足差分隐私的基础上，在下一节我们会给予隐私损失累积函数计算隐私成本。

### 3.2.3 隐私预算分析

对于所提差分隐私 SGD 算法，除了确保算法运行的准确率以外，另一个重要的问题就是评估算法训练时的数据隐私损失成本。为此，提出隐私损失累积函数的概念来进行每次迭代过程访问训练数据的隐私损失以及随着训练进展时的累积隐私损失。为不失一般性，令  $\sigma = \frac{\sqrt{2 \log(1.25/\delta)}}{\varepsilon}$ ，文献 [14] 严格证明，对于抽样概率  $q = \frac{\mathcal{L}}{N}$

且  $\varepsilon < 1$ , 则对于完整样本而言, 每次迭代过程都是  $(O(q\varepsilon), q\varepsilon)$ -差分隐私的。但文献并未对迭代过程以及噪声强度对差分隐私损失的影响展开研究, 故无法对噪声强度以及剪切阈值  $C$  进行有依据的选取。故首先需要研究迭代过程对差分隐私的影响机制。

事实上, 若令  $\sigma \geq c_2 \frac{q\sqrt{T \log(1/\delta)}}{\varepsilon}$ , 则同样应用文献 [14] 方法, 可以严格证明算法对于任意的  $\varepsilon < c_1 q^2 T$  都是  $(O(q\varepsilon\sqrt{T}), \delta)$ -差分隐私的, 其中  $c_1$  和  $c_2$  为常数。与文献 [14] 相比, 本文算法能够在相同迭代步骤下, 大幅度降低  $\varepsilon$  的数值, 对数据的隐私性保护更高。进一步地, 对于两个相邻的数据集  $d, d' \in D$  和映射机制  $M$ , 引入一个辅助输入变量  $\text{aux}$  和输出  $o \in R$ , 定义映射机制  $M$  在输出  $o$  处的隐私损失为:

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

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

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

进一步地, 映射机制  $M$  的损失边界值定义为:

$$\alpha_M(\lambda) \triangleq \max_{\text{aux}, d, d'} \alpha_M(\lambda; \text{aux}, d, d')$$

其满足以下特性:

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

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

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

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

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

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

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

## 第四章 联邦学习的安全聚合模型

在本章节中我们提出了一个在联邦学习中的安全聚合框架，通过安全 shuffling 实现分布式差分隐私，本地数据使用本地差分隐私进行加密，然后所有人传到一个安全 shuffler，shuffler 打乱次序，再发给服务器（不包含任何标识信息）。shuffler 可以作为一个可信第三方，独立于服务器并专门用于 shuffle。我们将会在本章节详细的描述该框架中各个模块的设计和实现过程。

随着机器学习应用领域的不断拓展，与机器学习模型的验证一样，模型的可解释性也变得愈发的重要。从用户的角度出发，机器学习模型不仅需要向用户反馈正确的预测结果，还需要向用户解释预测的原因。这样可以增加用户对模型的信任，让用户可以更放心的使用该模型；从开发人员的角度来说，可解释性对模型训练具有重要的意义。例如，根据解释信息可以帮助开发者确定更优的模型训练参数。在预测结果出现误差的时候，也可以了解导致误差产生的模型内部的原因，从而帮助开发人员改进机器学习模型的缺陷。总的来说，模型的可解释性已经成为机器模型应用的必备条件。

目前已经有很多研究着眼于机器学习模型的可解释性问题。[\[?\]](#) 的作者提出了一种基于自动推理的方法，可以从机器学习模型中提取有价值的信息，使用户可以了解树模型决策背后的原因。一些研究者 [\[??\]](#) 则试图用更简单的模型来近似复杂模型来提供更好的解释。局部解释的方法 [\[??\]](#) 了解的是模型的输出如何在局部的输入扰动上的分布变化的问题，它可以根据结果输出值推断出输入参数的重要性。此外，还有一些研究提出了基于实例的解释方法 [\[??\]](#)，即通过选择数据集的特定实例来解释机器学习模型的行为或解释底层的数据分布。相比于其他类型的机器学习模型来说，树模型以决策树的基础，对于一个输入样本来说，我们可以获得预

测的决策路径，根据决策路径可以获取一些决策信息，因此树模型通常被认为是一种易于解释模型。但很少有人研究树模型鲁棒性的可解释性问题。

在本小节中，我们主要关注的是模型的样本特征与模型鲁棒性之间的关系。首先，我们提出了鲁棒特征集合（Robust Feature Set, RFS）的概念，鲁棒特征集合可用于解释单个样本的鲁棒性。在鲁棒特征集合的基础上，我们进一步提出了一种计算局部鲁棒特征重要度（Local Robustness Feature Importance, LRFI）的算法，局部鲁棒特征重要度可用于解释树模型的样本特征与预测类别的鲁棒性的关系。

## 4.1 鲁棒特征集合

目前对于树模型的鲁棒性验证的研究中，在鲁棒性验证失败情况下，验证器通常将返回对抗性样本，但是当验证成功的情况下，通常都不会给出任何解释。针对这种情况，我们想进一步去探索为什么有些样本在特征被扰动的情况下，仍然能被识别正确？换句话说，我们是否可以确定哪些特征会真正的影响该样本的鲁棒性？不同的样本特征对模型的鲁棒性的影响大小是否也是不同的？

类似于 Z3 的 SMT 求解器在判断 SMT 公式不满足的情况下，会产生该公式集合的最小不满足核（Minimal Unsatisfiable Core, MUC），MUC 是原始公式集合的子集。我们首先给出 MUC 的定义。

**定义 4.1.1(最小不满足核).** 如果  $F$  是一个 CNF 公式， $F_C$  代表  $F$  的公式集合。如果  $S \subseteq F_C$  同时符合以下条件，则  $S$  是  $F$  的最小不满足核：

1.  $F$  是不可满足的。
2.  $S$  是不可满足的。
3. 不存在任何  $S'' \subseteq S$  是不可满足的。

**定义 4.1.2(鲁棒特征集合).** 给定一个树模型的分类模型  $C$ , 样本  $x = \langle a_1, a_2, \dots, a_d \rangle$ , 最大扰动距离为  $\epsilon$ ,  $\Phi$  是模型的鲁棒性公式,  $\Phi_C$  表示  $\Phi$  中的公式集合,  $\Delta(x, x', \epsilon) \subset$

$\Phi$  是特征扰动约束公式并且  $\Delta$  表示  $\Delta(x, x', \epsilon)$  中的公式集合，则鲁棒特征集合定义如下：

1.  $\Phi$  是不可满足的， $S \subseteq \Phi_C$  是  $\Phi$  的最小不满足核。
2.  $RFS = \{a_i \mid a_i \text{ 是出现在公式集合 } \Delta_s \text{ 中的特征}, \Delta_s \subseteq \Delta \text{ 是 } S \text{ 的子集}\}$

**定理 4.1.3.** 给定一个树模型  $C$ , 样本  $x = \langle a_1, a_2, \dots, a_d \rangle$ , 最大扰动距离为  $\epsilon$ 。保持存在于鲁棒特征集合中的特征的值不变的情况下, 任意扰动其他特征的值, 都不会改变该模型  $C$  对  $x$  的预测结果。

证明. 根据公式 ??, 我们知道  $out$  是由  $C(x')$  和  $\Delta(x, x', \epsilon)$  所决定的。在不失一般性的前提下, 我们可以将公式  $\Phi$  转换成  $\Phi' := C(x') \wedge \Delta(x, x', \epsilon) \Rightarrow (out \neq C(x))$  的形式表示。而公式  $C(x')$  的构建依赖于模型  $C$  的结构, 所以当模型  $C$  给定的情况下, 公式  $C(x')$  也是确定的。在这种情况下, 公式  $\Phi$  的可满足性就仅仅与  $\Delta(x, x', \epsilon)$  相关。所以在此定理的证明中, 我们不需要考虑  $C(x')$  的可满足性。

我们用  $\Phi_c$  表示  $\Phi'$  的公式集合。则  $\Phi_c$  可以被定义为:  $\Phi_c = R_C \cup \Delta \Rightarrow \{o\}$ 。  
 $R_C$  表示  $C(x')$  的公式集合,  $\Delta$  表示  $\Delta(x, x', \epsilon)$  的公式集合, 其中的  $\Delta = \{\delta_i \mid 0 \leq i \leq d, \delta_i = |a_i - a'_i| \leq \epsilon\}$ ,  $o = (out \neq C(x))$ 。假设  $\Phi_c$  是不可满足的, 并且  $S \subseteq \Phi_c$  表示最小不满足核。首先, 我们可以得出  $\Phi_c \setminus \{o\}$  是可满足的。因为至少有一个真赋值  $x' = x$  使其满足。所以  $o$  必然存在于  $S$  中, 可表示为  $o \in S$ 。我们用  $R_s \subseteq R_C$  和  $\Delta_s \subseteq \Delta$  表示存在于  $S$  中的公式子集, 则我们可以得到  $S = R_s \cup \Delta_s \cup \{o\}$  并且有  $\Phi_c \setminus S = (\Delta \setminus \Delta_s) \cup (R \setminus R_s)$ .

根据定义 4.1.1 和定义 4.1.2, 我们知道鲁棒特征集合中的特征是出现在  $\Delta_s$  中的特征。并且每个特征  $a_s \in RFS$  都对应着  $\Delta_s$  中的一个子句。类似的, 每个特征  $a_o \in X^d \setminus RFS$  也对应着  $\Delta \setminus \Delta_s$  中的一个子句。公式 (??) 表示样本  $x$  中特征的扰动约束公式, 如果  $\delta \in \Delta$  为 True, 则特征  $a$  的扰动距离必然不可能超过  $\epsilon$ 。反之, 如果  $\delta$  为 False, 则扰动距离必然超过了  $\epsilon$ 。所以该子句的真假其实代表的是该特征的扰动距离。根据最小不满足核的性质, 我们可以得到每个特征  $a_o \in X^d \setminus RFS$  对应的

子句都不会影响  $\Phi$  的可满足性。因为  $S = R_s \cup \Delta_s \cup \{o\}$  是不可满足的, 我们可以得出每个  $\delta_s \in \Delta_s$  都是可满足的, 也就是说, 其中的每个特征的扰动距离都没有超过  $\epsilon$ 。在此证明中我们只考虑每个特征的值保持不变的情况, 所以每个特征对应的子句  $\delta_s = |a_s - a'_s| = 0$  都为 True。因为  $\Phi_c = R_C \cup \Delta \Rightarrow \{o\}$  是不可满足的, 所以  $\Phi'_c = R_C \cup \Delta \Rightarrow \neg o$  是有效的, 即  $\neg o = (\text{out} = C(x))$ , 也就是说该模型对  $x$  的预测结果保持不变。即证。  $\square$

根据第三章的树模型的鲁棒性验证框架的介绍, 我们可知当验证样本  $x$  的鲁棒性的时候, SMT 编码模块会将其编码成对应的 SMT 公式  $\Phi$ , 之后利用 SMT 求解器判断  $\Phi$  的可满足性。如果求解器返回的结果为 UNSAT, 则说明  $\Phi$  是不可满足的。同时, 求解器会返回  $\Phi$  的最小不满足核。根据定义4.1.2, 我们可以得到样本  $x$  在树模型  $C$  上的鲁棒特征集合。需要注意的是, 获取鲁棒性特征集合的前提条件是模型基于样本  $x$  是满足鲁棒性的。

根据定理4.1.3, 我们可以得出以下结论: 在保持存在于鲁棒特征集合中的特征的值不变的情况下, 任意改变其他特征的值, 都不会影响树模型对样本  $x$  的预测结果。换句话说, 在最大扰动距离为  $\epsilon$  的情况下, 相较于其他特征来说, 鲁棒特征集合中的特征对鲁棒性有着更大的影响。

## 4.2 局部鲁棒特征重要度

在上一小节中, 我们提出了鲁棒特征集合的概念。为了进一步了解样本特征与模型预测类别鲁棒性的关系, 我们提出了局部鲁棒特征重要度 (Local Robustness Feature Importance, LRFI) 来描述这种关系。

在算法2中, 输入为一个树模型  $C$ ,  $N$  表示的是标记类别为  $y$  的测试样本集合其大小为  $|N|$ , 最大扰动距离  $\epsilon$  和特征集合  $X^d$ , 其输出为类别  $y$  的局部鲁棒特征重要度 LRFI。在第 5 行和第 6 行, 初始化中间变量  $S$  和  $V$  为空集。集合  $S$  用来保存所有测试样本的鲁棒特征集合,  $V$  用来保存所有特征在鲁棒特征集合中出现的次数。在第 7 行至第 12 行, 首先构建  $N$  中每个样本  $x$  的单样本鲁棒性公式  $\Phi_x$ , 之后利用

**Algorithm 2** 局部鲁棒特征重要度算法

---

```

1: Input: 树模型  $C$ , 测试样本集合  $N = \{x_i | 0 \leq i \leq |N|, C(x_i) = y\}$ , 最大扰动距离  $\epsilon$ , 特征集合
    $X^d = \{a_i | 0 \leq i \leq d\}$ .
2: Output: 预测类别为  $y$  的局部鲁棒特征重要性  $LRFI$ 
3: 过程: 函数 LocalRobustFeatureImportance( $C, N, \epsilon, X^d$ )
4: // 初始化集合  $S$  和  $V$  为空集
5:  $S \leftarrow \emptyset$ 
6:  $V \leftarrow \emptyset$ 
7: for  $x \in N$  do
8:    $\Phi_x \leftarrow R(x') \wedge \Delta(x, x', \epsilon) \wedge (out = y)$ 
9:    $UNSAT \leftarrow SMT_{solver}(\Phi_x)$ 
10:   $RFS_x \leftarrow$  根据定义4.1.2得到  $x$  的鲁棒特征集合
11:  将  $RFS_x$  加入到集合  $S$  中
12: end for
13: for  $a \in X^d$  do
14:   //  $n_a$  表示特征  $a$  在集合  $S$  中的出现次数
15:    $n_a \leftarrow 0$ 
16:   for  $RFS_x \in S$  do
17:     if  $a \in RFS_x$  then
18:        $n_a \leftarrow n_a + 1$ 
19:     end if
20:   end for
21:   将  $(a, n_a)$  加入到集合  $V$  中
22: end for
23: // 计算特征出现的最多次数与最少次数值
24:  $min_n = MIN(V)$ 
25:  $max_n = MAX(V)$ 
26: for  $(a, n_a) \in V$  do
27:    $n'_a \leftarrow (n_a - min_n) / (max_n - min_n)$ 
28:   将  $(a, n'_a)$  加入到  $LRFI$  中
29: end for
30: return  $LRFI$ 

```

---

SMT 求解器对  $\Phi_x$  进行可满足性判断。如果求解器返回结果为 UNSAT，则根据定义4.1.2计算出样本  $x$  的鲁棒特征集合存入到  $RFS_x$  中，最后将  $RFS_x$  存入到集合  $S$  中。在第 13 行至第 22 行，计算每个特征在集合  $S$  中的出现次数。若特征  $a$  存在于样本的  $x$  的  $RFS_x$  中，则其出现次数  $n_a$  加 1，所以  $n_a$  的取值范围为  $0 \leq n_a \leq |N|$ ，并将  $(a, n_a)$  存入集合  $V$  中。在第 23 行至第 29 行，对  $V$  中的值进行数据归一化

处理。在此算法中，我们利用的 min-max 标准化（Min-max normalization）操作。最后，将经过标准化处理后的值作为类别  $y$  的局部鲁棒特征重要度返回。直观的来说，某个特征对鲁棒性的重要度是以该特征在测试样本集合的鲁棒特征集合中出现的频率来确定的，该特征出现的频率越高，说明对鲁棒性的影响就越大，其重要度值也就越高，反之，影响就越小，重要度值就越低。

### 4.3 本章小结

本章节主要讨论了树模型鲁棒性与样本特征的关系，我们提出了鲁棒特征集合和局部鲁棒特征重要度的概念，并且给出了相应的形式化定义和证明。我们将在下一章节的实验中，进一步证明我们结论。

## 第五章 实验与评估

之前的章节中，我们描述了树模型鲁棒性验证框架的设计和实现过程。在本节的内容中，我们选取了一些基准的数据集在该验证框架上进行实验评估。

### 5.1 基准数据集介绍

我们选用了以下三个数据集评估了我们的树模型鲁棒性验证框架：

- (1) 波士顿房价数据集 (Boston House Price Dataset) 收集了在 20 世纪 70 年代中期位于波士顿郊区的房屋价格的中位数，它是用于回归任务的经典数据集。该数据集有 506 个样本数据，每个样本数据包含了城镇人均犯罪率，高速公路便利指数，住宅的房间数等 13 个特征及其房屋价格的中位数。
- (2) 手写体数字识别数据集 (MNIST) 是用于分类任务的经典数据集，来源于美国国家标准与技术研究所。总共包含了 70000 个手写数字图像，每个图像的尺寸为  $28 \times 28$  像素，每个像素点用灰度值表示，灰度值范围为 0 到 255，图像分为 10 类别，分别代表 0-9。
- (3) FASHION-MNIST 数据集包含了 70000 个不同商品的正面灰度图像，与 MNIST 数据集一样，每个图像的尺寸为  $28 \times 28$  像素，灰度值范围同样为 0 到 255。所有的图像分为 10 种类别，如：T 恤，牛仔裤，裙子等。虽然数据集格式与 MNIST 相同，但由于图像内容的差别，使得有些模型或者算法在 MNIST 和 FASHION-MNIST 的表现会有很大不同。因此对于分类任务，我们在这两个数据集上都进行了实验作为对比。

## 5.2 实验环境与配置

本文中的所有的实验均在一台装有 64 位 Ubuntu 操作系统的主机上进行，所使用机器的 CPU 型号为 Intel Core i7-5960X，主频为 4.00GHz，运行内存大小为 32GB 和 1T 存储硬盘大小。我们利用 sklearn(scikit-learn) 来训练实验中所需要的树模型：随机森林模型和 GBDT 模型。sklearn 是一种开源的，基于 Python 编程语言的机器学习框架。需要注意的是，本文提出的树模型鲁棒性验证框架，同样适用于其他机器学习框架下树模型的验证（如：Silas[? ]，H2O，Ranger[? ] 等）。在对样本数据预处理的部分，我们使用了 Pandas，Numpy 等第三方库。

## 5.3 实验结果与分析

### 5.3.1 随机森林模型的鲁棒性验证与分析

#### 回归模型的验证

我们在波士顿房价数据集上展开了对随机森林回归模型的实验。在训练阶段，将数据集随机打乱，按照 4:1 的比例划分训练样本集和测试样本集。随后利用 sklearn 训练出拥有不同超参数的随机森林回归模型，如：模型学习率为 {0.1, 0.2, 0.3}，模型中树的深度为 {5, 8, 10}，模型中树的棵树为 {5, 8, 10}。训练出来的模型的准确率都在 93% 至 98% 之间。我们直接选取测试样本集中的样本来进行鲁棒性的验证。按照回归模型的单样本鲁棒性的定义，我们在此数据集下，对所有数据类型为数值型的特征，我们设置其对应的  $\epsilon$  的值为 3,  $\rho$  值为 5 代表 5000 美元，即在扰动房屋相关特征的情况下，模型对房屋价格的预测结果误差不能超过 5000 美元。

折线图 5.1 为模型学习率为 0.3 下随机森林回归模型的鲁棒性验证结果。从图中我们可以看出：随着树的棵树的增加，模型的全局鲁棒性在降低而且树的深度越小，模型的鲁棒性越高。

#### 分类模型的验证

对于随机森林的分类模型来说，我们分别在 MNIST 和 FASHION-MNIST 两个

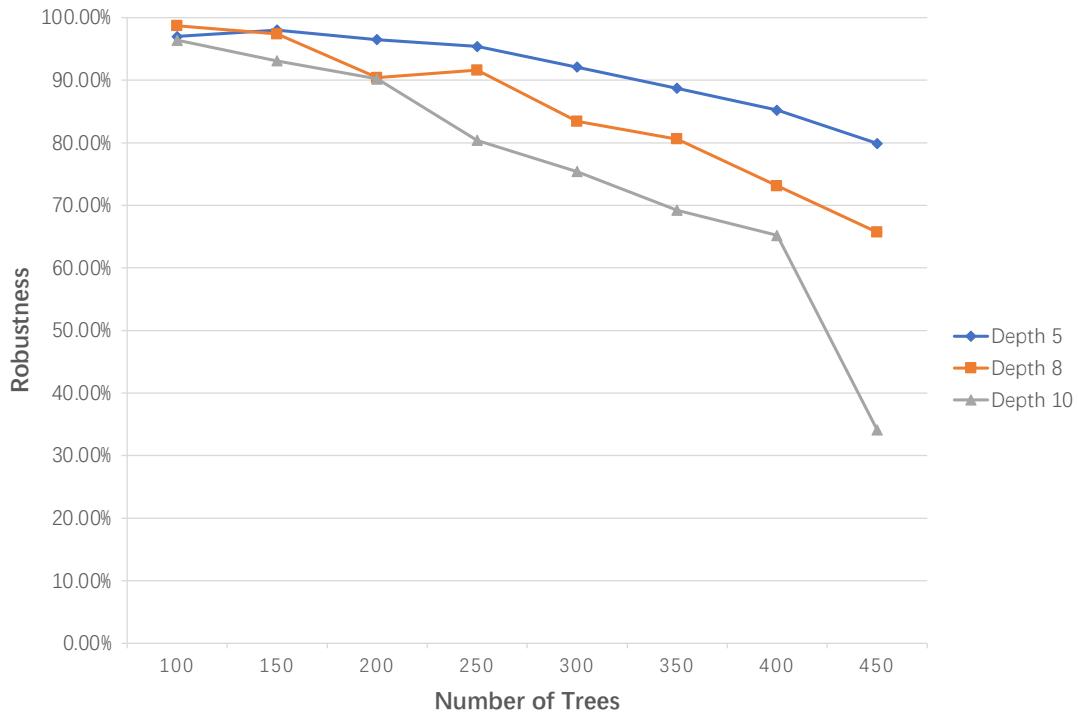


图 5.1: 随机森林回归模型验证结果

数据集上进行了实验验证。对于每个数据集，首先将数据集随机打乱，将其划分为两个子集：80% 训练样本集和 20% 测试样本集。然后，我们从测试样本集中随机抽取了 10 个类别的各 100 个图像，即每个鲁棒性测试样本集的大小为 1000。随后利用 sklearn 训练出随机森林的分类模型，用于验证。

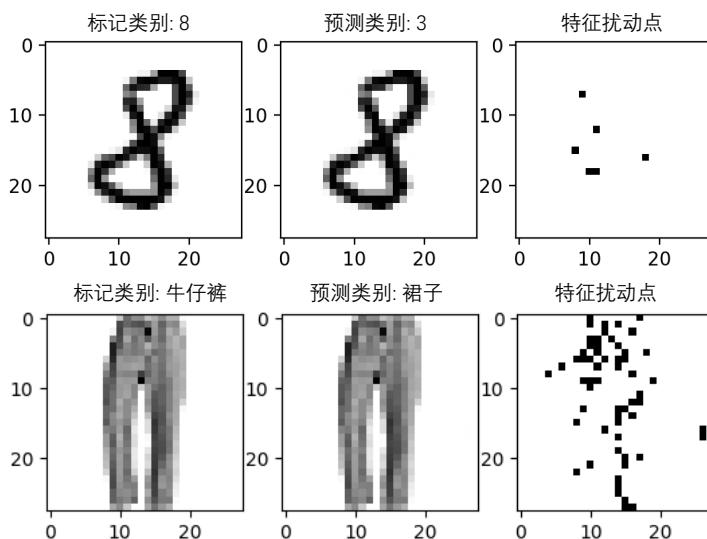


图 5.2: 对抗性样本图

图5.2展示了两个数据集中不满足单样本鲁棒性的测试样本的例子。根据分类模型单样本鲁棒性的定义，我们设置特征扰动范围值  $\epsilon = 1$ ，代表了一个灰度值。图中的第一列图像为原始的样本，第二列为第一列相对应的对抗性样本图像，我们在第三列的图中标记出了受到扰动的特征点。第一个示例来源于 MNIST 数据集，我们可以看出在受到扰动之后，数字“8”被模型错误的预测为数字“3”。第二个示例来源于 FASHION-MNIST 数据集，标记类别为“牛仔裤”的商品图像被错误地分类为“裙子”。在以上示例中，如果我们直接去对比第一列的原始图像和第二列的对抗性样本图像，凭借我们的肉眼，根本无法去找出这两个图像直接的差别（在此结果中，最多只有一个灰度值的差别）。这也反映出我们树模型鲁棒性验证框架的必要性。

### 5.3.2 GBDT 模型鲁棒性的验证与分析

#### 回归模型的验证

与随机森林的回归模型的验证实验一样，我们同样在波士顿房价数据集上进行了实验。对数据集的划分方式，训练参数的设置都与随机森林的回归模型保持一致。唯一不同的是，在 GBDT 模型中，我们需要设置损失函数，在此实验中，我们选择均方损失函数。同样，设置特征扰动范围值  $\epsilon$  为 3， $\rho$  值为 5，代表 5000 美元，即在房屋特征扰动的情况下，此模型对房屋价格的预测结果误差不能超过 5000 美元。

图5.3为模型学习率为 0.3 下 GBDT 回归模型的鲁棒性验证结果。从图中我们可以看出，与随机森林回归模型一致，随着树的深度和树的棵树的增加，模型的全局鲁棒性在降低。但在增加同样棵树的决策树情况下，GBDT 的回归模型的鲁棒性要比随机森林模型下降的更快。换句话说，随机森林模型鲁棒性的下降趋势较为“平缓”，而 GBDT 模型鲁棒性的下降趋势则比较“陡峭”。

#### 分类模型的验证

在 GBDT 分类模型的鲁棒性验证实验中，我们同样基于 MNIST 和 FASHION-MNIST 两个数据集上进行了实验验证。数据集的划分方式为：80% 训练样本集和

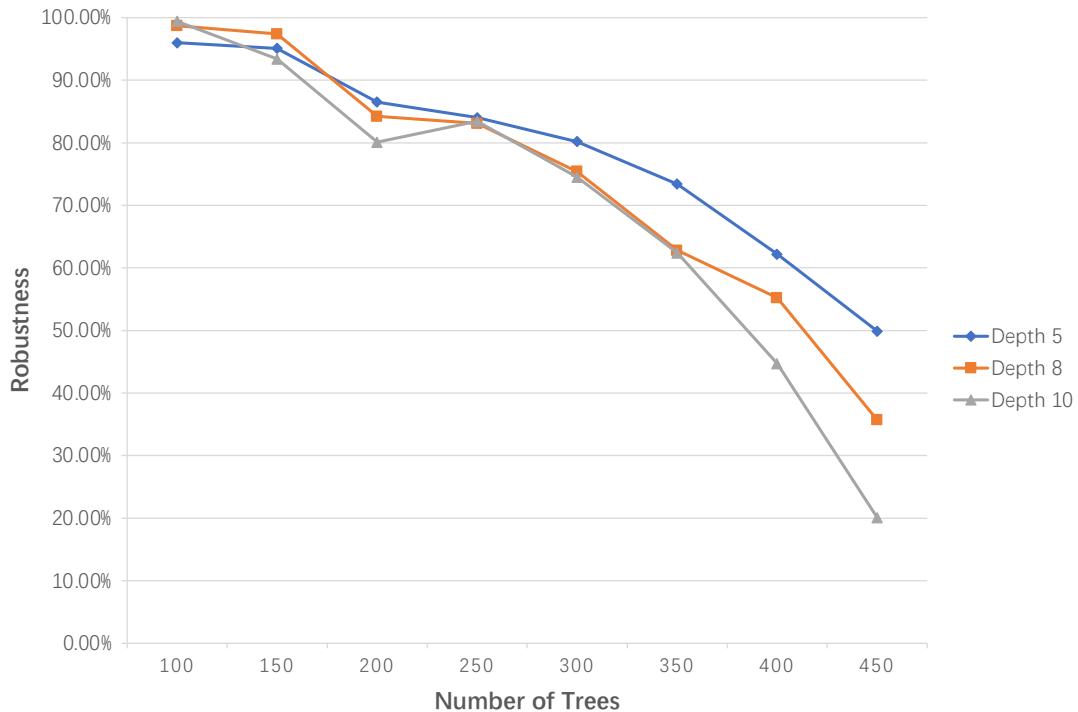


图 5.3: GBDT 回归模型验证结果

20% 测试样本集。之后，从测试样本集中随机抽取了 10 个类别的各 100 个图像，总的鲁棒性测试样本集的大小为 1000。

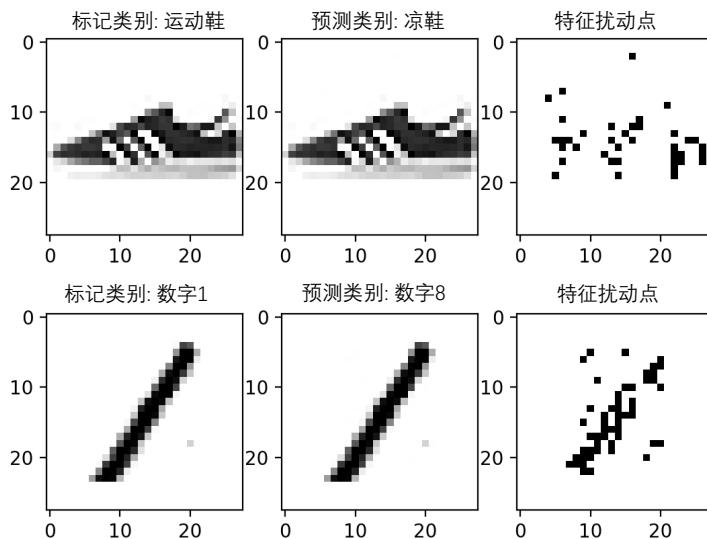


图 5.4: GBDT 验证反例

图5.4显示了 GBDT 分类模型下不满足单样本鲁棒性的测试样本。其特征扰动

范围值  $\epsilon = 3$ , 即最多扰动 3 个灰度值。图中的第一列图像为原始的样本, 第二列为第一列相对应的对抗性样本图像, 第三列的图像标记出了受到扰动的特征点。我们可以看出在图像受到扰动之后, 在第一个示例中标记为类别“运动鞋”的商品图像被错误地预测为“凉鞋”。第二个示例中数字“1”被模型错误的预测为数字“8”。在扰动范围设置为 3 个灰度值的情况下, 我们依然无法通过肉眼看出原始图像和对抗性样本图像之间的区别。

### 5.3.3 树模型鲁棒性可解释性的实验与分析

#### 鲁棒特征集合

根据我们给出的鲁棒特征集合定义, 我们在随机森林分类模型和 GBDT 分类模型中, 进行了相关的实验与分析。根据定义可知, 在测试样本满足单样本鲁棒性的情况下, 我们可以获取其鲁棒特征集合。因此, 我们可以直接在之前分类模型的实验中, 获取满足鲁棒性的样本的鲁棒特征集合。

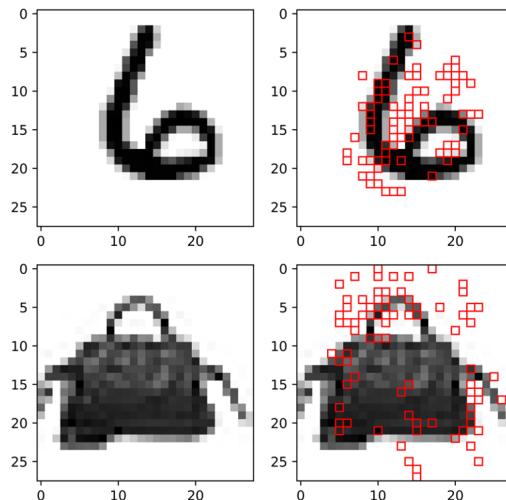


图 5.5: 随机森林分类模型鲁棒特征集合

图5.5显示了在随机森林分类模型下的两个数据集中满足单样本鲁棒性的测试样本的示例。特征扰动范围  $\epsilon$  设置为 3, 代表了 3 个灰度值。在受到扰动的情况下, 这些样本仍然被模型正确识别。图中的第一列显示了原始的样本, 第二列为对应

图像的鲁棒特征集合图。我们用红色矩形标记处了存在于该样本鲁棒特征集合中的特征点。根据鲁棒特征集合的性质，我们知道保持红色矩形标记的特征点的像素灰度值不变，在特征扰动距离最大为 3 个灰度值的情况下任意改变其他特征点的像素灰度值都不能改变模型对该样本的识别结果。为了验证此结论，我们随机的改变不包含于鲁棒特征集合中的特征点的像素灰度值，而保持红色标记点像素灰度值不变，然后让模型去识别将改变后的测试样本之后，去检查预测结果是否发生变化。经过大量的随机测试，以上结论正确。

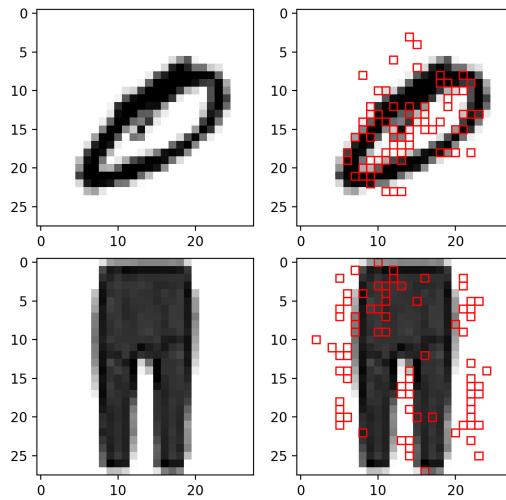


图 5.6: GBDT 分类模型鲁棒特征集合

图5.6显示了在 GBDT 分类模型下的两个数据集中满足单样本鲁棒性的测试样本的示例。特征扰动范围  $\epsilon$  设置为 1。实验过程与随机森林实验保持一致。第一行显示的为数字“0”样本的鲁棒特征集合，第二行显示的为商品“裤子”样本的结果。

### 局部鲁棒特征重要度

根据我们对局部鲁棒特征重要度的定义，我们首先收集了的在随机森林分类模型测试样本集中所有满足单样本鲁棒性的测试样本，根据算法2，我们可以求得模型不同类别的鲁棒特征重要度。

我们在分别在 MNIST 和 FAHSION-MNIST 数据集中，各自选择了一个类别来计算局部鲁棒特征重要度。图5.7展示了基于随机森林分类模型的实验结果。左侧的图片为 MNIST 中数字“0”的结果，右侧显示了 FASHION-MNIST 中的商品“运动

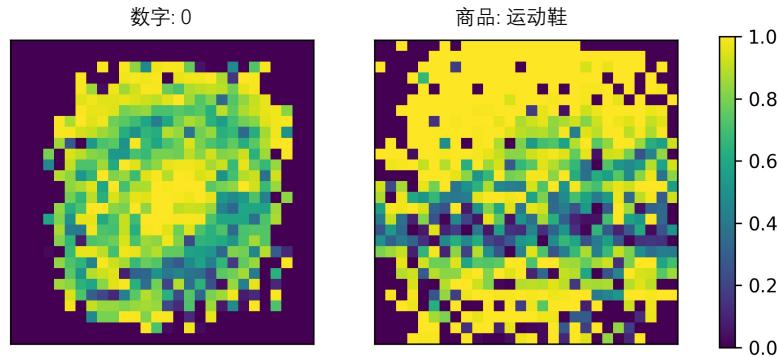


图 5.7: 随机森林分类的局部鲁棒性特征重要度

鞋”的结果。特征点的鲁棒重要度的值越大，则颜色越黄。如果其鲁棒重要度的值为 0，则颜色为紫色。我们可以观察到由于特征点的重要度值的不同，显示出了该类别的基本形状。重要度大的值基本分布在该类别的基本形状周围，而分布在该类别的基本形状之上的特征点的鲁棒重要度值都比较低。这为对抗性样本的攻击提供了新的思路：在进行对抗性样本攻击的时候，应该优先选择这些鲁棒重要度值比较高的点，去产生对抗性样本，这样可以提高攻击的成功率与效率。如果从我们实验得出的特征重要度分布的规律来看，应该优先选择分布在该类别基本形状周围的特征点去进行攻击。需要注意的是，除了我们给出的以上两个类别的结果之外，其他类别的鲁棒特征重要度也有类似的分布规律。

#### 5.3.4 不同类别鲁棒性的验证与分析

在其他关于树模型的鲁棒性的验证研究中，对于分类模型的鲁棒性验证都是针对于该模型的整体而言的。但是同一模型的不同的类别的鲁棒性可能会不同。在此种情况下，整体的模型鲁棒性的验证结果，并不能提供具体类别的鲁棒性信息。所以我们设计了实验去研究同一模型下不同类别的鲁棒性的是否会出现差异的问题。与之前的实验设置保持一致，数据集的划分方式为：80% 训练样本集和 20% 测试样本集，从测试样本集中随机抽取了 10 个类别的各 100 个图像。训练出的树模型的识别率都在 95% 至 98% 之间，并且各个类别的识别率也基本相同。之后我们分别对不同类别的 100 个样本进行鲁棒性验证。

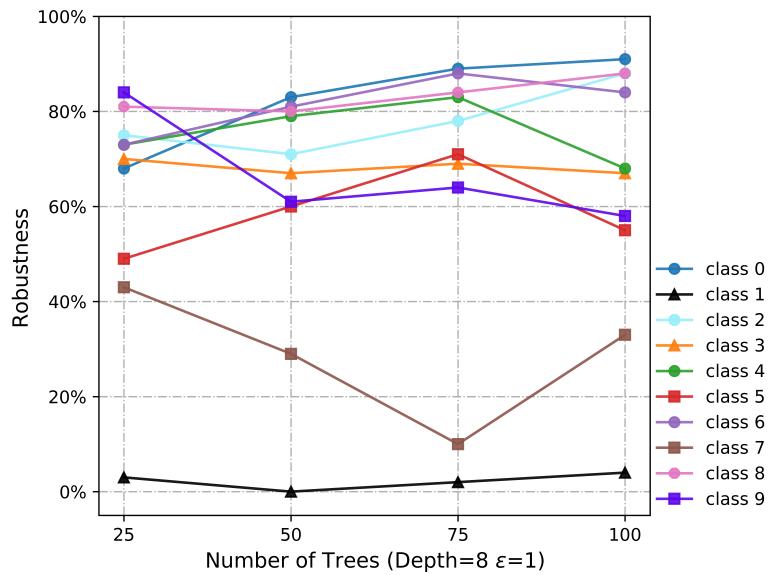


图 5.8: MNIST 中不同类别鲁棒性的验证结果

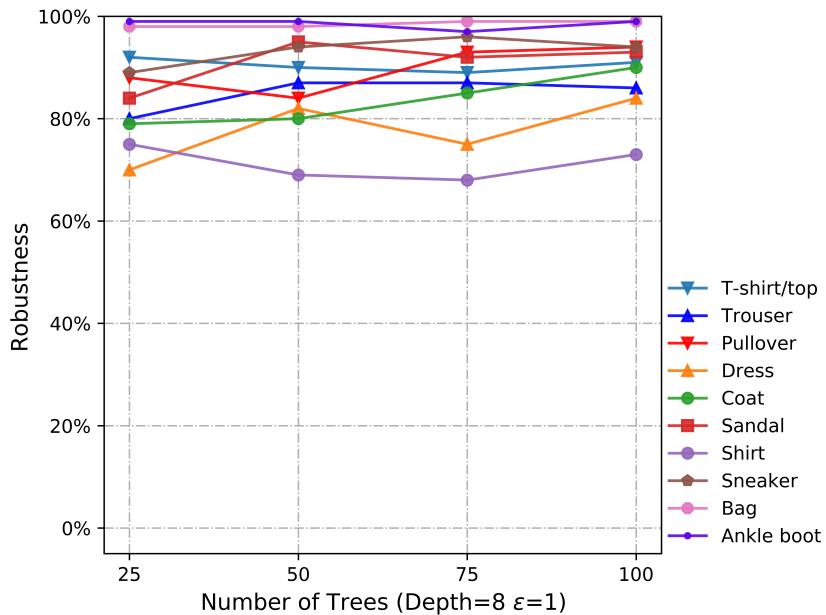


图 5.9: FASHION-MNIST 中不同类别鲁棒性的验证结果

折线图5.8和5.9展示了两个数据集中的不同类别的鲁棒性验证结果。图5.8显示了MNIST数据集的结果。我们可以观察到，存在几个类别（例如，数字“0”，数字“2”，数字“6”，数字“8”）的鲁棒性随着树的棵树的增加而略有提高。而数字“4”，数字“5”，数字“7”，数字“9”类中的鲁棒性值有很明显的波动。除此以外，数字“1”的鲁

棒性始终保持在非常低的值。尽管模型对数字“1”的识别率与其他数字的识别率基本一样，但它们的鲁棒性值却存在着显着差异，差值大约在 40% 至 80% 之间。相比之下，FASHION-MNIST（图5.9）中各个类别的鲁棒性总体上保持稳定，并没有随着树的棵树的增加而出现明显的波动。但是，商品类别为“衬衫”的鲁棒性值相比于其他类别要低一些。在该数据集中，没有类似于数字“1”这种的鲁棒性非常低的类别去影响该模型的总体的鲁棒性值。根据此次的实验结果，我们可知在验证树模型鲁棒性的时候，将注意力集中在整个模型的鲁棒性上是不准确的，对于不同的数据集来说，模型对于不同类别样本的鲁棒性的表现可能会有很大的差别。这给了我们的一个启示，在验证分类模型鲁棒性的时候，验证结果应该细化到不同的类别。这些信息对于模型的使用者来说是非常有用的，可以让他们更加详细的了解到该模型的优缺点，从而增加了模型的可信度。

### 5.3.5 树鲁棒性超参数与鲁棒性关系的验证与分析

在本文提出的鲁棒性验证框架下，我们进一步研究了树模型中两个重要的训练超参数：树的棵树和树的深度与树模型鲁棒性的关系。我们基于 MNIST 数据集去进行这部分的实验。通过在不同深度，不同树的棵树参数下去训练模型，通过对比其鲁棒性结果来进行研究和分析。

表5.1为随机森林分类模型在不同训练参数下的鲁棒性结果，其中 Trees 和 Depth 分别表示模型中树的棵树和树的深度，Accuracy 表示的是模型的识别率，Verified( $\rho$ ) 表示模型的全局鲁棒性，Timeout 表示验证超时，Failed 表示验证失败即不满足单样本鲁棒性所占的百分比。我们可以观察到，在特征扰动值为  $\epsilon = 1$  的情况下，保持相同树的深度，树的棵树并不会对其鲁棒性造成很大的影响，但在之前的对回归模型的实验中，在保持树的深度相同的情况下，树的棵树的增加，会导致其模型鲁棒性的降低。此外，在保持树的棵树相同的情况下，增加树的深度参数的值，会使模型的鲁棒性少量的增加。与之相反的是，对于其回归模型来说，树的深度的增加，会导致其模型鲁棒性的降低。根据我们的实验结果，在保证模型准确率的情况下，模型的开发人员可以通过调整训练参数来增加其模型的鲁棒性。与

表 5.1: 基于 MNIST 数据集模型在不同超参数下的鲁棒性验证结果.

Trees	Depth	Accuracy	$\epsilon = 1$			$\epsilon = 3$		
			Verified( $\rho$ )	Timeout	Failed	Verified( $\rho$ )	Timeout	Failed
25	5	84%	45.71%	0%	54.29%	9.64%	0.12%	90.24%
50	5	85%	54.68%	0%	45.32%	13.58%	2.69%	83.72%
75	5	86%	48.77%	5.61%	45.61%	9.24%	13.68%	77.08%
100	5	86%	54.07%	14.53%	31.40%	13.02%	21.74%	65.23%
25	8	91%	61.47%	0%	38.53%	9.88%	0.11%	90.01%
50	8	93%	61.02%	0%	38.98%	14.36%	3.02%	82.61%
75	8	92%	63.63%	5.32%	31.05%	12.81%	17.05%	70.14%
100	8	93%	63.48%	15.04%	21.48%	16.86%	24.81%	58.32%
25	10	93%	64.34%	0%	35.66%	14.18%	0%	85.82%
50	10	95%	63.21%	0%	36.79%	17.34%	0.53%	82.14%
75	10	94%	75.32%	5.32%	19.36%	13.83%	4.57%	81.60%
100	10	95%	66.84%	8.74%	24.42%	15.16%	14.21%	70.63%

$\epsilon = 1$  时做对比,  $\epsilon = 3$  的情况下, 该模型的鲁棒性有了明显的降低, 这是显而易见的, 因为在特征扰动范围为 3 个灰度值的情况下, 会产生更多的对抗性样本使得原始样本的鲁棒性不满足。

还有一点值得我们注意, 随着树的棵树和树的深度的值的增加, 验证超时的比例也在增加, 这揭露了我们验证框架的不足。因为从本质上来说, 验证框架的验证能力一定程度取决于 Z3 求解器的求解能力, 当树模型规模变大的时候, 我们编码形成的 SMT 公式数目也是剧增的, 这就导致状态爆炸的问题, 从而使得求解器无法求解, 导致验证超时, 无法确定该样本的鲁棒性是否满足。

### 5.3.6 验证时间的结果与分析

框架的验证时间同样也是我们需要关注的部分, 我们需要保证在一定时间内返回正确的验证结果。于是, 我们基于 MNIST 数据集, 通过验证不同规模大小的树模型来统计该框架的验证时间。

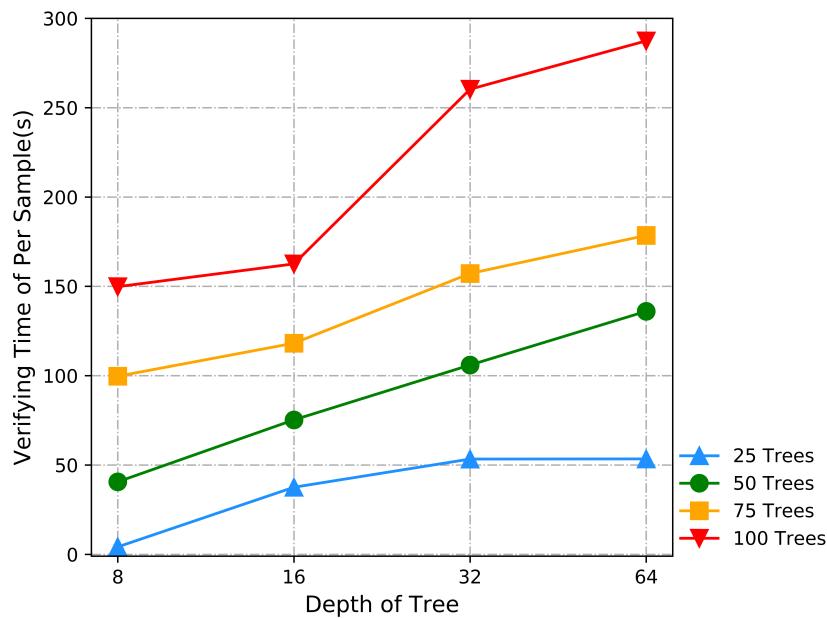


图 5.10: 单样本验证时间图

图5.10为在不同规模树模型下验证单个样本所需要的平均时间统计图。我们在测试样本集中随机选择了 100 个样本来进行验证时间的结果统计，结果值为平均值。我们的验证覆盖了规模很小到大规模的树模型，最小规模为深度为 8，棵树为 25 的树，单个样本的验证时间为 4s。而对于树的棵树为 100，深度为 64 的大规模的模型，我们框架的验证时间为 287s。随着树模型规模的增加，验证时间也在逐渐增加。总体来说，我们框架可以去验证大规模的树模型的鲁棒性，而且验证时间也是可接受的时间范围。但如表5.1所显示的，在进行大规模模型的验证的时候，有些样本可能会出现验证超时的情况，这也是我们在未来工作中，需要解决的问题。

## 5.4 本章小结

在本章中，我们选取了三个基准数据集对本文提出的鲁棒性验证框架进行了一系列的实验来测试其可行性，并且对树模型鲁棒性的可解释性和树模型训练参数与鲁棒性的关系也进行实验和研究。实验结果表明，我们的验证框架可以有效验证随机森林和 GBDT 这两个树模型的重要组成部分。但也存在不足，就是虽然

可以验证大规模的树模型，可是某些样本还是会出现验证超时的情况，这将是我们未来工作中的重点。

## 第六章 总结与展望

### 6.1 总结

随着机器学习在各个领域的大规模应用，尤其是在安全领域也占有了重要的地位。机器学习模型的安全性和可解释性也引起各国政府和研究学者的极大关注。鲁棒性是模型安全性的重要体现之一，所以验证模型鲁棒性的方法和工具也变得尤为迫切。

树模型以其高效，方便，泛化能力强的特点，在各个领域都有广泛的应用。但与神经网络模型一样，树模型也易收到对抗性样本的影响。本文基于 SMT 技术对机器学习树模型的鲁棒性进行了研究和分析。本文的主要工作和贡献如下：

- (1) 本文提出了一个基于 SMT 技术树模型的鲁棒性验证框架，该框架支持树模型中两个重要实现：随机森林与 GBDT 模型的鲁棒性的验证。该框架能够有效验证大规模的树模型的鲁棒性。
- (2) 本文提出了鲁棒特征集和局部鲁棒特征重要度的概念，从模型可解释性的角度，进一步研究了树模型的鲁棒性和样本特征之间的关系。为对抗性样本反例的生成和对抗性攻击提供新的思路，也可以帮助模型开发人员进一步优化模型提高其鲁棒性。
- (3) 本文通过实验讨论了树模型超参数与模型鲁棒性的关系，从而为训练阶段提高模型鲁棒性的研究提供了重要参考。

综上所述，本文的研究充分证明了所提出框架的有效性，可以极大的增加树模型的可靠性，同时也对鲁棒性的可解释性做了研究，从而进一步推进了树模型

在安全领域的应用和发展。

## 6.2 展望

我们的研究还留存一些待解决的问题，可以考虑从下面的几个方面展开研究：

- (1) 本文所提出的鲁棒性验证框架的验证能力很大程度上受限于 SMT 求解器本身的求解能力，在未来应该考虑针对树模型编码成 SMT 公式的特点，对 SMT 的底层求解算法进行优化，从而提高验证的效率。
- (2) 虽然在现有的验证框架下，可以支持一些大规模树模型的验证，但是对于高维度，更大规模的模型还是会发生状态爆炸的问题，导致验证结果无效。在未来可以先对树模型本身先进行模型缩减的操作，之后再进行验证，以便可以验证超大规模的模型。
- (3) 我们的验证框架在对树模型的鲁棒性验证问题上已经取得了一定的成效，接下来可以将其扩展到复杂度更高的机器学习模型上去。目前，已有研究者考虑将神经网络转换为决策树，使其模型的复杂度降低。我们可以沿着这个思路，将我们的方法扩展到神经网络模型上去，进一步加强框架的验证能力。