对抗鲁棒的深度学习算法

(申请清华大学工学博士学位论文)

培 养 单 位： 计算机科学与技术系

学　　科： 计算机科学与技术

研 究 生： 庞 天 宇

指 导 教 师： 朱　军　教　授

二〇二二年五月

Adversarially Robust Deep Learning Algorithms

Dissertation Submitted to

Tsinghua University

in partial fulfillment of the requirement

for the degree of

Doctor of Philosophy

in

Computer Science and Technology

by

Pang Tianyu

Dissertation Supervisor: Professor Zhu Jun

May, 2022

学位论文公开评阅人和答辩委员会名单

公开评阅人名单

张长水	教授	清华大学
刘成林	研究员	中科院自动化研究所

答辩委员会名单

主席	刘成林	研究员	中科院自动化研究所
委员	张钹	教授	清华大学
	张长水	教授	清华大学
	朱军	教授	清华大学
	山世光	研究员	中科院计算技术研究所
秘书	李建民	副研究员	清华大学

摘 要

近年来，深度学习技术在多种任务和实际应用场景中都取得了令人瞩目的进展。然而，在正常环境下表现良好的深度学习模型在对抗环境中却很容易受到攻击。具体来说，对抗攻击者可以构造出对抗样本来欺骗模型，且人类观察者无法明显区分对抗样本与正常的干净样本。基于这一问题，很多对抗防御策略被提出，旨在抵御潜在的对抗攻击。

然而，之前的对抗防御策略中存在一些关键问题。第一，鲁棒学习相比于标准学习需要更高的样本复杂度，即更多的训练样本。第二，在对抗检测方法中，检测指标与模型特征并未很好地匹配，且无法抵御潜在的自适应攻击。第三，现有的对抗防御方法在训练中所采用的基础参数设定非常不统一，导致无法公平快速地比较不同防御策略的优劣。本文着手于解决上述这些关键问题，提出新的鲁棒学习算法以及提供系统性的实验结果。本文的主要创新点如下：

- 为了提高训练过程中的样本效率，提出了单模型的最大化马氏距离学习范式。通过提前构造最优的类别中心并使用中心化损失函数，可以促使样本间信息的传输，使得训练过程具有收敛速度快、不受小批量情况影响等良好特性，同时很大程度上提高了单模型的鲁棒性。
- 在使用集成模型的场景中，提出了多样性增强正则化学习范式。通过鼓励不同单模型返回的最大预测一致且非最大预测正交，可以在不影响干净样本上分类准确率的同时，提高集成模型整体的鲁棒准确率。
- 为了使得模型特征分布与对抗检测指标更好地配合，提出了反交叉熵训练准则，将正常的干净样本映射到特征空间中的低维流形上。这样当使用基于特征分布的对抗检测指标时，可以更加灵敏地区分出干净样本与对抗样本。
- 由于单检测指标无法抵御自适应攻击，进一步构造了一对互耦的检测指标，可以在特定条件下保证区分出任何正确分类样本和错误分类样本。该方法还可以与各类对抗训练方法相结合，得到更加鲁棒的互耦检测指标。
- 系统性地对对抗训练中使用的各种基础参数设定进行了消融实验，评估了每种参数设定对于模型性能的影响程度。结果发现，对抗训练的效果对于一些基础参数设定很敏感，这些基础参数设定不同带来的影响甚至会超过改进训练方法本身。因此根据实验结果，给出了一套对抗训练标准的基础参数设定，从而保证可以公平比较不同的对抗训练方法的优劣。

关键词：深度学习，可信赖机器学习，对抗攻击，对抗防御，鲁棒性

Abstract

Deep learning (DL) has obtained unprecedented progress in various tasks and real-world applications. However, a high-accuracy DL model can be vulnerable in the adversarial setting, where human-imperceptible adversarial examples are maliciously generated to mislead the model to output wrong predictions. To this end, many adversarial defenses have been proposed against adversarial attacks.

However, there exist several critical problems of adversarial defenses to be addressed. First, robust learning requires higher data complexity, or namely, more training data. Second, adversarial detection metrics do not match the feature distribution. Third, existing adversarial defenses apply inconsistent implementation details, making it inconvenient to benchmark them. This thesis addresses these problems by developing robust algorithms and performing comprehensive empirical studies. The novel contributions are summarized as follows.

- To improve sample efficiency in training, we propose the Max-Mahalanobis learning paradigm and a centralized loss function for a single model, which facilitate sharing information among different samples. Our method can largely improve adversarial robustness, while accelerating training convergence and will not be affected by small mini-batch sizes.
- We propose diversity promoting regularizer for ensemble models, which encourages single-model members to return consistent maximal predictions and orthogonal non-maximal predictions, in order to improve robust accuracy without degrading clean accuracy.
- To make the feature distribution better collaborate with adversarial detection metrics, we propose reverse cross-entropy training principle to map the features of clean inputs onto low-dimensional manifolds, such that statistic-based detection metrics can better distinguish clean and adversarial inputs.
- Beyond using a single detection metric, we propose to exploit two coupled detection metrics, which can provably distinguish any correctly classified input with wrongly classified ones. Our method can also combine with different adversarial training methods, which can further enhance the robustness and universality of the two coupled detection metrics.

- We comprehensively perform empirical studies on the effects of different training tricks used in adversarial training. From the results, we find that the performance of adversarially trained models are quite sensitive to some training settings, while the influence of different training settings even surpass the gain from more advanced training methods. Thus, according to our observations, we provide a set of standard training tricks for adversarial training, in order to fairly evaluate the effectiveness of different adversarial training methods.

Keywords: deep learning; trustworthy machine learning; adversarial attacks; adversarial defenses; robustness

目 录

插图和附表清单

符号和缩略语说明

MMLDA	最大化马氏距离判别分析
SCE	softmax 交叉熵
MMC	最大化马氏距离中心
LDA	线性判别分析
LR	逻辑回归
RB	鲁棒性
AT	对抗训练
ADP	自适应多样性增强
ED	集成多样性
CE	交叉熵
ECE	集成交叉熵
DPP	行列式点过程
LED	集成多样性的对数
RCE	反交叉熵
non-ME	非最大熵
RR	修正滤除
T-Con	真实置信度
R-Con	修正置信度
TPR	真正例率
BCE	双值交叉熵
LS	标签平滑
PGD	投影梯度下降法
MIM	动量迭代方法
BPDA	后向传递可微近似
EoT	期望变换

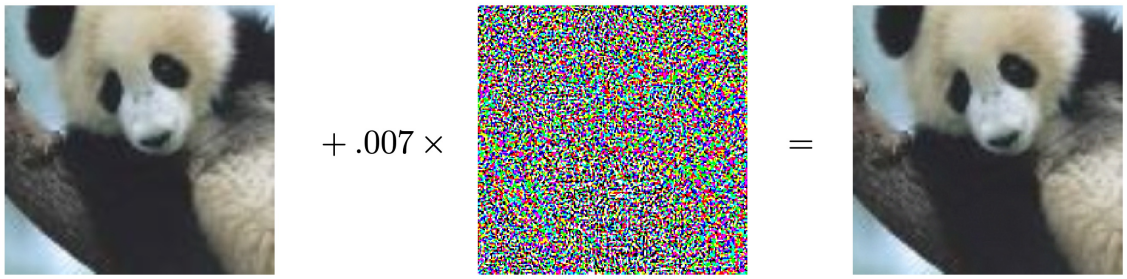
第 1 章　绪论

近年来，深度学习技术取得了极大的发展，并被广泛应用到计算机视觉、自然语言处理、推荐系统等多个领域。然而大量研究表明，在正常数据上表现良好的深度学习模型却普遍存在对抗鲁棒性缺陷。这一鲁棒性缺陷给注重安全性或者可靠性的应用场景（例如自动驾驶、人脸支付等）带来潜在的风险。为了提高深度学习模型抵御对抗攻击的能力，本文从多个方面提出对抗防御方法，旨在提升模型预测的可靠性，并从原理上加深对深度学习机理的理解。本章将介绍本文的研究背景与意义，详述对抗环境的定义，总结国内外研究现状与关键研究问题，概括本文的研究内容与主要贡献，并给出本文的组织结构。

1.1　研究背景与意义

随着计算硬件设备（例如 CPU、GPU、TPU 等）性能的不断提升，深度学习技术得以利用海量数据学习到其内在规律[1]。特别是自 2012 年 AlexNet[2] 在 ImageNet 比赛中表现出远超第二名的准确率之后，基于神经网络的模型架构受到了极大的关注，并成为深度学习领域中的主流方法。在过去的十年中，新的深度学习算法不断被提出，并被应用到了各种实际场景中，包括人脸识别[3-5]、机器翻译[6-7]、推荐系统[8-9]、自动驾驶[10]、医疗检测[11]等等。在一些领域，深度学习技术甚至颠覆了人们的传统认知。例如 2016 年英国 DeepMind 公司开发出名为 AlphaGo[12] 的人工智能围棋软件，击败了多位世界顶尖棋手，并导致后续出现了 AI 围棋流派。2018 年 DeepMind 公司开发出来了名为 AlphaFold[13] 的蛋白质结构预测系统，其在之后的两年内蝉联蛋白质结构预测技术的关键测试（Critical Assessment of protein Structure Prediction，缩写为 CASP）竞赛冠军[14]，并被用于预测新冠 COVID-19 病原体的蛋白质结构。

深度学习模型展现出的优越性能一度超越人类水平，使得研究者对于深度学习最终能够通向强人工智能抱有信心。然而，之前的研究发现深度学习模型存在对抗鲁棒性（adversarial robustness）缺陷[15-16]，即模型会被攻击者恶意构造的对抗样本欺骗。对抗鲁棒性缺陷在近几年受到了广泛的关注，主要是由于其反直觉性，即在人类观察者看来对抗样本与正常的干净样本并无明显的差异，但是模型返回的预测却会大不相同。如图 1.1中所示[16]，在图像分类任务中，左边为正常的干净样本，中间为攻击者构造的对抗噪音（经过归一化），右边为构造出的对抗样本。可以看到，模型会以很高的置信度将对抗样本识别为错误类别。



干净样本
模型分类：熊猫
（置信度57.7%）

对抗噪音
模型分类：线虫
（置信度8.2%）

对抗样本
模型分类：长臂猿
（置信度99.3%）

图 1.1　对抗样本示意图[16]

路标检测　　**行人检测**

图 1.2　物理世界中的对抗样本[17-18]

对抗鲁棒性缺陷最早是在图像分类任务中被观察到，然而最近的一系列工作发现对抗鲁棒性缺陷在深度学习的各个领域中普遍存在（如图 1.3中所列），包括自然语言处理[19]、图模型[20]、强化学习[21]、语音识别[22]、激光雷达[23]、3D 点云[24]、代码生成[25] 以及推荐系统[26] 等等。在实际的物理世界中，攻击者可以打印出对抗扰动来欺骗已经落地部署的深度学习模型，例如图 1.2中所示的自动驾驶中的路标检测或者安防摄像头的行人检测系统[17-18]。在与社会安全密切相关的军事、金融、工业等领域，恶意攻击者可能通过对抗扰动来破坏军事防御系统、窃取用户财产、影响生产流程等。同时，由于模型鲁棒性的不足，会造成公众对于深度学习甚至人工智能技术的信赖危机，这也阻碍了人工智能的进一步发展与落地。

鉴于对抗鲁棒性缺陷给深度学习模型及其应用带来潜在的安全威胁，越来越多的研究者开始关注深度学习的对抗鲁棒性，近年来该领域成为人工智能研究的一大热点方向。除了传统的模型评测指标以外，鲁棒性也成为我们评判深度学习模型的另一个角度。从原理上讲，由于深度学习模型的输入信号通常处在高维复杂空间中（例如像素空间、语义空间），所以攻击者容易找到模型泛化性差的区域，并在此区域内寻找对抗样本。基于此，一方面对抗攻击（adversarial attacks）研究如何在

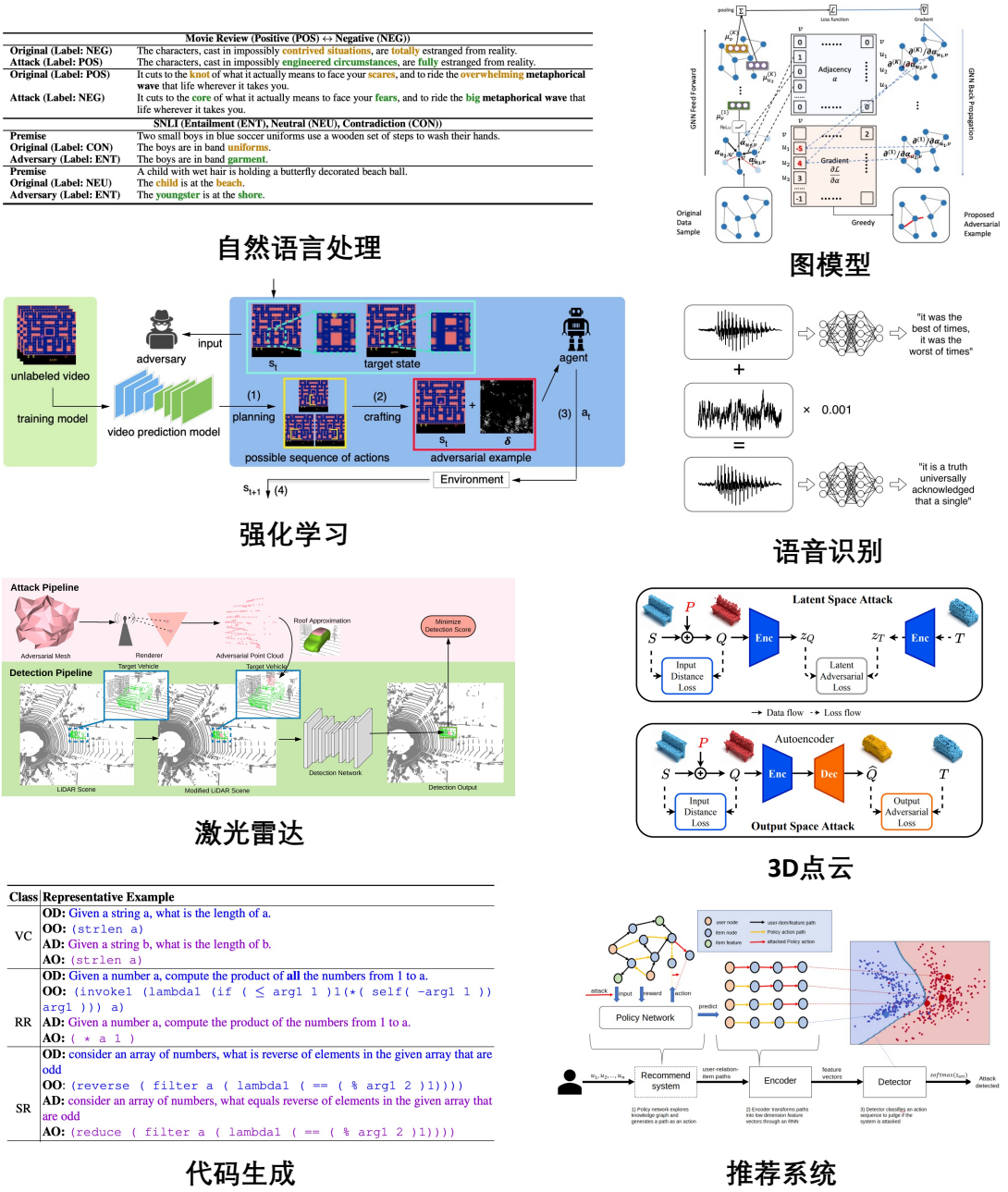


图 1.3　对抗鲁棒性缺陷普遍存在[19-26]

各种场景下高效地攻破模型[16,27-29]，相当于传统安全领域中寻找漏洞的过程；另一方面对抗防御（adversarial defenses）研究如何提升模型的对抗鲁棒性[30-35]，从而抵御潜在的对抗攻击，相当于修复漏洞的过程。对抗攻击与防御的研究往往相互博弈，新的防御方法可以抵御之前的攻击方法，然而新的攻击方法又可以攻破之前的防御方法。在对抗攻防两个方向的不断博弈中，研究者逐渐摸索出公认的评测标准与算法[36-37]，可以更有效地筛选出对于模型鲁棒性提升有用的防御模块。

综上所述，研究深度学习的对抗鲁棒性对于构建安全的人工智能系统、促使

新技术可靠落地具有重要的理论与现实意义。

1.1.1　研究价值

本文将针对对抗鲁棒的深度学习算法开展研究，提出对抗防御算法，这是深度学习对抗鲁棒性研究中的重要方向，具有以下研究价值。

首先，对抗防御算法可以提升模型的对抗鲁棒性及实际场景中的可靠性。随着深度学习技术的发展，其越来越多地被应用到实际场景中，如人脸支付、安防系统、金融信用评级等等。然而大量的研究显示，对抗攻击不仅在数字世界中有效，其同样可以以各种方式（例如打印出对抗扰动）迁移到物理世界中，这对于未经过充分安全性测试的现实系统造成了潜在的安全威胁。基于此，对抗防御算法最直观的研究价值就是可以提升深度学习系统的可靠性，抵御恶意的对抗攻击，并且从策略上有选择地与人类观察者进行合作（例如自动驾驶中适时地请求人类驾驶者介入）。

其次，对抗防御算法可以加深我们对于深度学习技术机理的理解。在对抗样本的存在性尚未被研究者广泛了解时，我们对于模型的评测指标相对单一，例如在图像分类任务中大量研究工作不断尝试提升 ImageNet[38] 上的 Top-1 准确率。然而，当我们用对抗攻击来评测这些工作中提出的模型时，发现得到的鲁棒准确率几乎都近似为零[27]。这也促使对抗鲁棒性逐渐成为后续工作中模型评测的重要指标之一。由于深度学习模型结构复杂且可解释性存在不足，对抗防御算法可以帮助我们理解哪些结构模块、学习范式、推断策略等对于模型预测的鲁棒性有提升。此外，在对抗防御算法的研究过程中，发现了很多具有启发意义的现象，例如准确率与鲁棒性的制约关系[30]，鲁棒模型的梯度更具有语义[39]。这些实验现象为之后理解深度学习机理的研究打开了新的思路。

最后，对抗防御算法对于其他任务上模型的性能也会有所帮助。最近的研究发现，对抗鲁棒的深度学习模型学到的特征可以更好地应用在迁移学习任务上[40]，同时具有更好的可解释性[41]。此外，对抗防御算法例如对抗训练还可以用于半监督学习[42]、隐私保护[43] 以及正常的图像分类任务[44] 等等。鲁棒模型的生成式特性也启发了一系列工作，探索例如扩散生成模型（diffusion-based generative model）、分数生成模型（score-based generative model）与鲁棒学习中各类对抗防御算法的联系[45]。沿着“对抗向好”（adversarial for good）的思路，很多工作着手于探索鲁棒模型除了安全性以外的其他正向应用，这成为该领域的一个新的研究热点。

因此，对抗防御算法在提升模型鲁棒性、安全性的同时，还加深了我们对于深度学习机理的理解，并且开阔了可以应用在其他领域的新方法和新思路。

1.2　对抗环境的定义

本节中将介绍对抗环境的定义，即*威胁模型*（threat model），其主要包括三个方面：*攻击者目的*（adversarial target）、*攻击者能力*（adversarial capabilities）以及*攻击者知识*（adversarial knowledge）。下述介绍中我们使用的术语参考自文献[46]。

1.2.1　攻击者目的

攻击者在构造对抗样本欺骗模型时，其可以存在不同的攻击目的。在分类任务中，按照攻击者目的可以分为有目标攻击（targeted attack）和无目标攻击（untargeted attack）。具体来说，有目标攻击希望模型返回特定的预测类别（例如攻击人脸支付）；而无目标攻击则仅希望模型返回错误的预测类别。一般情况下，无目标攻击比有目标攻击更容易实现，而有目标攻击往往会造成更大的潜在危害。

1.2.2　攻击者能力

由于实际场景中的各种限制，攻击者的能力并非无限大。例如攻击者修改图片像素时，若扰动过大则容易被人类观察者察觉；而由于访问权限的限制，攻击者也很难直接修改模型参数等等。所以为了将对抗攻防问题合理地进行数学建模，我们需要形式上定义出攻击者的能力范围，即攻击者可以操作的对象及程度。之前的工作倾向于将对抗攻击者的能力范围限制在“对输入的微小扰动”。这里“微小”的定义可以是欧式距离意义下的（例如图像像素空间），也可以是语义距离意义下的（例如自然语言处理中的定义）。本文中我们主要聚焦于图片的像素空间。

数学上，令 x 为一个干净的输入样本，D 为一相似度的度量函数，x' 为攻击者构造的对抗样本。我们限制 $D(x, x') \leqslant \epsilon$，其中 ϵ 为最大允许的扰动大小。在图像分类任务中，我们通常选取 D 为 ℓ_p-距离，即限制对抗样本满足 $\|x - x'\|_p \leqslant \epsilon$。在这样的攻击者能力限定下，一个模型 f_θ 在 x 点的鲁棒性可以定义为

$$\max_{x' : \|x - x'\|_p \leqslant \epsilon} \mathcal{L}(f_\theta(x'), y), \tag{1.1}$$

其中 $\mathcal{L}$ 为损失函数，y 为样本 x 的真实类别，θ 为模型参数。在构造对抗样本的过程中，攻击者通常以问题（1.1）为目标函数，通过优化、梯度迭代或者搜索的方式来近似找到问题（1.1）的最优解。

1.2.3　攻击者知识

攻击者需要根据目标模型的预测行为来构造有针对性的对抗样本。这时攻击者所使用的算法很大程度上依赖于攻击者的知识，即其对于目标模型的访问权限。具体来说，当目标模型仅包含分类器时，对抗攻击可以分为白盒攻击（white-box

attacks）、黑盒迁移攻击（black-box transfer-based attacks）、黑盒得分攻击（black-box score-based attacks）和黑盒决策攻击（black-box decision-based attacks），其中后三种攻击类型均属于黑盒攻击（black-box attacks）。当模型使用例如检测器等额外的防御策略时，对抗攻击还可以分为灰盒攻击（oblivious attacks[47]）和自适应攻击（adaptive attacks[48]）。灰盒攻击指的是攻击者仅知道分类器的模型参数，但是不知道检测器的机制和参数；而自适应攻击指的是攻击者同时对分类器和检测器拥有白盒访问权限，因此可以针对性地同时攻击分类器和检测器。

1.3　国内外研究现状

基于上述对抗环境的定义，本小节将从对抗攻击、对抗防御和鲁棒性测评三个方面综述深度学习鲁棒性的国内外研究现状。

1.3.1　对抗攻击

对抗攻击主要分为白盒攻击、黑盒迁移攻击、黑盒得分攻击以及黑盒决策攻击。下面将逐一介绍这四类主流攻击算法。

白盒攻击：在白盒攻击设定中，攻击者拥有访问模型参数的权限，因此可以基于梯度迭代或者优化的方法构造对抗样本。在梯度迭代策略中，Goodfellow 等人[16]在 ℓ_∞-范数设定下提出了快速梯度符号法（fast gradient sign method，缩写为 FGSM）。FGSM 是基于函数的局部线性近似的高效单步攻击算法。此后的工作将 FGSM 推广到多步的情况[17]，并且引入随机初始化来提升攻击的成功率[29]，提出了投影梯度下降法（projected gradient descent，缩写为 PGD）。在基于优化的策略中，Szegedy 等人[15]提出基于 L-BFGS 优化算法的白盒攻击。此后，加州大学伯克利分校的 Carlini 和 Wagner 提出了 C&W 算法，将最大类别和第二大类别之间的分对数（logits）差作为构造对抗样本的目标函数，通过线性优化器进行优化，并且使用二分查找寻找攻击成功的最小扰动。然而，上述的这些白盒攻击都是基于模型梯度的，所以很容易被*梯度混淆*的防御策略所破坏[49]。为了解决这一问题，Athalye 等人[48]提出了后向传递可微近似（Backward Pass Differentiable Approximation，缩写为 BPDA）与期望变换（Expectation over Transformation，缩写为 EoT）两种攻击策略。这两种攻击策略可以作为即插即用的模块嵌入到有针对性的自适应攻击中，其中 BPDA 用于模型的梯度无法直接计算的情况，而 EoT 用于模型预测存在随机性的情况。

黑盒迁移攻击：在大部分场景中，攻击者很难直接访问模型的参数。甚至在一些例如需要付费访问的系统中，攻击者需要考虑尽量减少对于目标模型的访问次

数，从而控制攻击代价。在这样的考量下，黑盒迁移攻击算法[27,50-51]在本地训练出一个与目标模型相似的替代模型，并对替代模型使用白盒攻击算法构造出对抗样本，然后将对抗样本输入给目标模型。由于模型间的相似性，能够欺骗替代模型的对抗样本有很大概率也可以欺骗目标模型。

黑盒得分攻击：当攻击者可以多次访问或者查询目标模型预测，且模型返回整个预测概率向量时，攻击者可以使用黑盒得分攻击策略，包括通过黑盒优化或者有限差分方法得到对模型梯度的估计[52]，从而可以使用基于梯度迭代的白盒攻击算法；另一类方法基于遗传算法，使用高斯分布搜索，每次更新可能存在对抗样本的方向[53-54]。

黑盒决策攻击：在很多实际落地的 API 中（例如人脸识别），模型往往仅返回一个预测类别。在这种设定下，攻击者可以使用黑盒决策攻击策略，包括通过多次访问模型预测类别从而近似统计出预测概率向量，并使用黑盒得分攻击[53]；此外，还可以使用基于启发式搜索策略的边界攻击方法[55]，在搜索过程中不断减小对抗样本与原始干净样本之间的距离。

1.3.2　对抗防御

为了提升模型的鲁棒性从而抵御对抗攻击，大量的对抗防御策略被不断提出。从原理上讲，对抗防御的设计灵活性大，可以借鉴多个领域的知识与技术。因此，本小节中我们仅介绍几种主流的对抗防御策略，包括对抗训练（adversarial training）、可证实防御（certified defenses）、对抗检测（adversarial detection）以及自适应防御（adaptive test-time defenses）。

对抗训练：为了模型可以在对抗样本上正确分类，我们在模型的训练过程中引入自我博弈的机制，通过不断构造训练对抗样本，并且鼓励模型在这些训练对抗样本上正确分类来提高模型在测试阶段的鲁棒性。这种学习范式称为对抗训练[16]。在各类对抗攻防比赛中[56-57]，几乎所有防御方案都包含对抗训练框架，例如 PGD-AT[29]和 TRADES[30]。此外，基于 PGD-AT 或者 TRADES，很多后续工作通过引入其他领域的技术来进一步提高模型的鲁棒性，包括集成学习[58-59]、流形学习[31,60-61]、生成式建模[32,62-64]、半监督学习[65-67]以及自监督学习[68-71]。另一方面，由于构造训练对抗样本需要很大的计算量导致训练时间增长，研究者也提出了各种方案来加速对抗训练过程，包括重复利用梯度计算[72-73]、自适应的攻击迭代步数[74-75]或者直接使用单步的对抗训练[76-78]。然而单步对抗训练会导致例如灾难性过拟合（catastrophic overfitting），需要进一步的正则化来减弱其影响[79-80]。

可证实防御：从安全性角度而言，我们希望模型可以保证在给定扰动大小下必然返回正确的预测。在这样的诉求下，可证实防御通过整数规划（例如 MILP 求

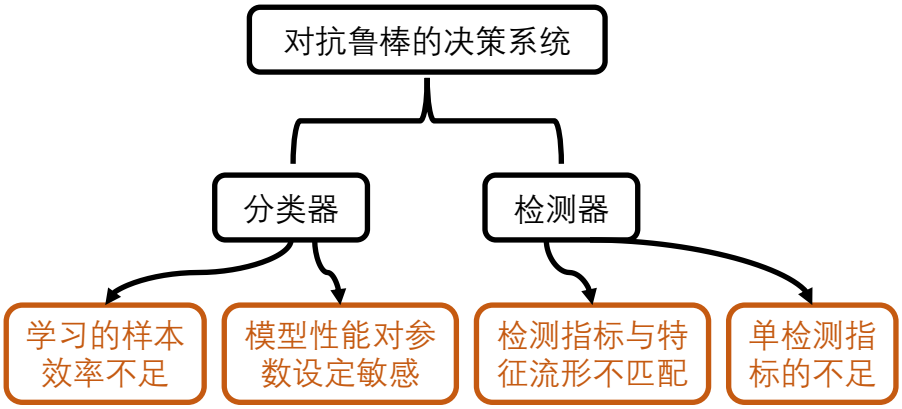


图 1.4　对抗鲁棒的深度学习算法中有待解决的关键研究问题

解器[81]）判断出在给定扰动大小下，是否存在对抗样本可以欺骗模型。然而完全的（complete）可证实防御计算复杂度极高，因此后续的大部分可证实防御都基于非完全（incomplete）框架，通过凸放缩（convex relaxing）来加速模型验证[82-83]。此外，在更大规模的数据集例如 ImageNet 以及更深的模型结构上，我们可以通过随机平滑（randomized smoothing）的方式得到大概率可证实的防御[84]。

对抗检测：除了正确分类对抗样本，另一种防御方案希望可以检测出对抗样本，并将其视为异常输入[85-91]。之前的对抗检测方法主要分为两类。第一类是基于特征统计量的，包括密度比例[92]、核密度估计[33,93]、预测方差[94]、互信息[95-96]、Fisher 信息[97]、局部固有维度[98]、激活函数不变性[99]以及特征属性[100-101]等等。第二类是基于模型的，包括引入额外的检测器[102-104]、在分类器特征上构建混合高斯模型[105-107]以及引入额外的生成式模型[108-110]等等。

自适应防御：在对抗训练以及可证实防御中，由于攻击者是在已知模型结构参数的情况下构造对抗样本（攻击者为后手），且防御者无法在测试阶段改变输入或者模型参数，所以防御者通常是被动的（防御者为先手）。基于此，最近的一些防御策略考虑自适应防御[111]，即允许模型在测试阶段根据输入的特性施加扰动，或者改变模型参数。自适应防御方法包括输入净化（input purification），通过额外的模型或者迭代过程来对对抗样本进行去噪[112-113]；以及模型适应（model adaptation），根据输入来动态地修改模型参数或者统计量（例如 BN 层的均值与方差）[114-115]。

1.3.3　鲁棒性测评

由于之前的工作中提出了很多种防御策略及方法，因此很多鲁棒性评测的工作着手于评估现在防御方法的鲁棒准确率，挑选出对模型鲁棒性提升具有普适性的防御模块。Dong 等人[116]对一些典型的防御方法进行了大规模的评估实验，并且绘制出鲁棒性曲线；Croce 等人[28]提出 AutoAttack，其集成了四种攻击算法，并且

形成了目前领域内最常用的鲁棒性评测排行榜 Robustbench[36]；Chen 等人[117]提出了黑盒攻击 RayS 算法并构建了相应的鲁棒性评测排行榜。除了对抗鲁棒性外，相关工作还提出包含一般扰动（general corruptions or perturbations）的数据集，例如 MNIST-C[118]包含 15 种作用在 MNIST 原始图片上的一般扰动，ImageNet-C 以及 ImageNet-P[119]包含多种作用在自然图片上的一般扰动。对抗防御算法在一般扰动下的鲁棒性可以反映出该防御算法的泛化性，并且可以避免过拟合到训练中遇到的攻击范式[120-121]。

1.4　关键研究问题

综上所述，已有的研究针对深度学习模型的鲁棒性缺陷问题提出了多种对抗防御算法，旨在进一步构建安全可靠的深度学习模型，抵御潜在的对抗攻击。但是，这些工作存在分类器学习的样本效率不足、模型性能对参数设定敏感、检测指标与特征流形不匹配以及单检测指标能力不足的问题。本文所解决的关键研究问题总结如图 1.4所示。

在分类器学习的样本效率方面，已有的工作表明鲁棒学习相比于标准学习需要更大的样本复杂度[122]。介于此，后续工作着手于引入额外的有标注数据进行数据增广[68]，或者使用无标注数据进行半监督学习[65-66]。然而，在很多场景下收集新的数据需要很大的成本，且由于采样时间和环境的差别，新引进的数据无法保证与原数据处于同分布，从而会造成分布漂移（distributional shift）。因此，研究如何更高效地利用现有数据对于鲁棒学习在各类场景中的易实现性具有重要的研究意义。

在参数设定对于模型性能的影响方面，已有的各类防御方法所使用的基础训练参数设定差异性很大，并且这些参数设定并未在消融实验中被给予足够的研究[35,123]。这会导致很多防御方法虽然从原理上对于提升鲁棒性有帮助，但由于实现时参数设定不合适，导致训练后得到的模型鲁棒性并未展示出应有的显著提升。另一方面，随着各类鲁棒性测评排行榜的出现[36]，使用统一的基础训练参数设定有助于快速、公平地比较各种防御方法的优劣，帮助对抗防御领域迭代发展。

在检测指标方面，现有的大部分检测器直接构建于分类器所学到的特征流形上。由于分类器在训练过程中并未考虑检测指标的特性，所以学到的特征流形不一定可以很好地与检测指标相兼容[33]。因此，研究如何训练分类器特征，使其更好地与检测指标特性相配合，达到更高的检测灵敏度具有重要的研究意义。

此外，已有的检测器通常使用单检测指标，或者使用神经网络黑盒融合多个检测指标。在这样的检测策略下，攻击者可以有针对性地构造自适应攻击算法和

表 1.1　各章节研究内容在研究问题、研究对象和解决思路三个方面的总结

章节	研究问题	研究对象	解决思路
第 2 章	学习的样本效率不足	单分类器	最大化马氏距离学习
第 3 章	学习的样本效率不足	集成分类器	集成模型的多样性增强鲁棒学习
第 4 章	检测指标与特征流形不匹配	单检测指标	基于反交叉熵训练的对抗样本检测
第 5 章	单检测指标的不足	多检测指标	基于置信度修正的互耦对抗样本检测
第 6 章	模型性能对参数设定敏感	单分类器	对抗训练中的技巧及参数设定

目标函数，达到同时欺骗分类器和检测器的效果，导致整个预测系统无法保证在自适应攻击下的鲁棒性[47]。如何从理论上保证对抗样本检测的可靠性成为一个关键性的研究课题。

1.5　研究内容与主要贡献

基于上述问题，本文针对对抗鲁棒的深度学习算法开展系统性研究。根据研究问题的不同，本文的研究内容与主要贡献可以总结为以下五个部分。表 1.1概括了各章节研究内容的研究问题、研究对象和解决思路。

第一部分研究内容为第 2 章，旨在提升单模型学习的样本效率。由于鲁棒学习需要比标准学习更大的训练样本数量，之前的工作尝试使用额外的有标注或者无标注数据来帮助训练。然而收集额外的训练样本有可能需要很大的成本，且由于采样环境的限制，无法保证新的训练数据与原始数据处于同一分布。针对这一问题，第 2 章提出最大化马氏距离学习，通过提高现有训练数据的使用效率，促进样本间信息的传输，提高模型的鲁棒性。此外，该方法还具有收敛速度快，不受小批量情况影响等良好的性质。该部分的研究成果发表在人工智能领域顶级国际会议 ICML 2018[32] 以及 ICLR 2020[34] 上。

第二部分研究内容为第 3 章，旨在提成集成模型学习的样本效率。在实际应用的系统中，我们往往会使用多个单模型构成的集成模型来提高预测准确率。然而由于训练数据量有限，所以每个单模型通常是在同一个数据集上进行训练，因此会趋向于学到相似的决策行为，导致单模型间差异性小，容易产生迁移对抗样本。针对这一问题，第 3 章提出集成模型的多样性增强鲁棒学习，通过鼓励单模型间非最大预测的多样性来抑制对抗样本在单模型间迁移，从而提升集成模型的整体鲁棒性。该部分的研究成果发表在人工智能领域顶级国际会议 ICML 2019[59] 上。

第三部分研究内容为第 4 章，旨在鼓励模型学到与单检测指标相兼容的特征流形。之前的对抗检测方法研究主要聚焦于提出新的检测指标。这些检测指标大多建立于分类模型所学的特征流形上。然而在分类器的训练过程中，往往并未考虑到检测指标的特性，因此无法保证分类器特征流形与检测指标的适配程度。针对这一问题，第 4 章提出基于反交叉熵训练的对抗样本检测方法，通过将传统的交叉熵训练换成反交叉熵训练过程，模型可以将正常的干净样本输入映射到特征空间中的低维流形上。这样当输入样本为对抗样本时，检测指标可以更加灵敏地将其检测出来。该部分的研究成果发表在人工智能领域顶级国际会议 NeurIPS 2018[33] 上。

第四部分研究内容为第 5 章，旨在解决单检测指标可靠性不足的问题。之前的主流对抗检测方法使用单检测指标，或者多检测指标的黑盒融合。然而，这些检测方法无法从理论上对其可靠性进行保证，因此攻击者很容易构造出自适应攻击算法和目标函数来同时欺骗分类器和检测器。针对这一问题，第 5 章提出互耦的双检测指标方法，理论上证明了模型预测置信度与修正置信度可以构成一对互耦的检测指标，在特定条件下可以完全区分出任何正确分类样本与错误分类样本，因此可以抵御潜在的自适应攻击。该部分的研究成果发表在人工智能领域顶级国际会议 CVPR 2022[124] 上。

第五部分研究内容为第 6 章，旨在探究基础的训练参数设定对于对抗训练的影响。已有的各类防御方法所使用的基础训练参数设定差异性很大，并且这些参数设定并未在消融实验中被给予足够的研究。这会导致很多防御方法虽然从原理上对于提升鲁棒性有帮助，但由于实现时参数设定不合适，导致训练后得到的模型鲁棒性并未展示出应有的显著提升。另一方面，随着各类鲁棒性测评排行榜的出现，不一致的基础训练参数设定也会导致难以公平快速地比较不同的防御方法。针对这一问题，第 6 章系统地进行了大量的对比消融实验，评估多种训练参数设定对于对抗训练效果的影响程度，发现鲁棒学习对于一些基础参数设定十分敏感。基于此，第 6 章提供了一套建议的标准参数设定，帮助后续的防御方法快速到达理想的训练效果。该部分的研究成果发表在人工智能领域顶级国际会议 ICLR 2021[35] 上。

1.6　本文组织结构

本文的组织结构总共包含 7 个章节。

第 1 章为绪论，介绍本文研究背景与意义，综述国内外研究现状，总结关键研究问题，并概括本文研究内容与主要贡献。

第 2 章提出最大化马氏距离学习，在类间构造最大化马氏距离中心，在类内最小化中心化损失函数，从而得到类间边际距离大、类内聚合的特征分布。这样可

以提高模型预测的鲁棒性，并且加快训练收敛速度。

第 3 章提出集成模型的多样性增强鲁棒学习，通过鼓励多个单模型间非最大预测的多样性来抑制对抗样本的迁移，从而提升集成模型整体的预测可靠性。

第 4 章提出基于反交叉熵训练的对抗样本检测方法，通过反交叉熵训练准则，鼓励模型将正常干净样本的输入映射到特征空间中的低维流形上，从而当配合检测指标使用时可以更加灵敏地检测出对抗样本。

第 5 章提出基于置信度修正的互耦对抗样本检测方法，从理论上证明了模型预测置信度与修正置信度可以构成一对互耦的检测指标，可以在特定条件下完全区分出正确分类样本和错误分类样本，从而抵御潜在的自适应攻击。

第 6 章针对于对抗训练中基础参数设定对模型性能的影响进行了系统的评估，通过大量的消融实验找到了对鲁棒学习效果影响大的参数设定。此外第 6 章还提供了一套建议的标准训练参数，为构建综合的防御方法测评提供帮助。

第 7 章回顾并总结本文研究内容与主要贡献，并展望了未来的研究方向。

第 2 章　最大化马氏距离鲁棒学习

大量的训练数据被认为是深度学习能够成功的基石之一。同样，在对抗鲁棒性方面，之前的工作表明*增加训练数据*可以显著地提升模型的鲁棒准确率。然而，在很多场景中，获取新的数据需要大量的人力或物力资源，并且由于时空环境的限制，无法保证新获取的数据与原数据处在相同分布下。因此，本章中我们研究如何使模型更高效地从现有的数据中进行学习。我们首先提出了最大化马氏距离线性判别分析方法，可以在提高学习效率的同时保证类间鲁棒性最优。基于此，我们进一步分析了 softmax 归一化操作的缺陷，并由此得出最大化马氏距离中心损失函数，可以在保持类间鲁棒性最优的同时，鼓励类内样本聚合，提高样本密度，促进同类样本间信息的共享。我们的实验表明最大化马氏距离学习范式可以极大地增强模型的鲁棒性，并且维持甚至提高模型在正常样本上的准确率。

2.1　本章引言

深度学习在各类任务上都达到了领先的性能[1]。然而，近几年大量的研究表明，标准学习（standard learning）范式下得到的深度学习模型很容易受到对抗样本的欺骗[15-16,125]。尽管之前的很多工作提出了各种防御方法，然而大部分防御方法都很难抵御自适应攻击（adaptive attacks）[48]，从而无法得到真正鲁棒的模型。基于此，Schmidt 等人[122]认为鲁棒学习（robust learning）范式相比于标准学习范式来讲，需要更大的训练样本复杂度。而鉴于在一些常用数据集例如 CIFAR-10[126] 上我们很难学到鲁棒的模型[36]，Schmidt 等人推测这是由于这些数据集中的样本量本身不足以满足鲁棒学习的要求。为了解决这一问题，一系列后续工作尝试引入额外的有标注数据[68]或者无标注数据[65-66]来扩充原有数据集中的样本数量。实验显示，额外的训练数据可以很大程度上提升鲁棒学习的效果。此外，近期的工作表明，在原始数据集上训练的生成式模型（generative models）也可以通过生成样本的方式来扩充数据集，并且同样可以达到提高被训练模型鲁棒性的目的[127-128]。然而在现实场景中，获取额外数据有可能需要大量的资源；并且由于采样环境的限制，可能无法保证新获取的样本与原样本处于同一分布。而另一方面，生成式模型的训练也往往远难于判别式模型（discriminative models）[129]。

因此在本章中，我们考虑如何更高效地利用已有的训练数据来训练鲁棒的判别式模型。我们首先根据经典的统计学习效率理论[130]，提出了最大化马氏距离判别分析（max-Mahalanobis linear discriminant analysis，缩写为 MMLDA）网络。

MMLDA 将输入空间（input space）的数据分布映射到特征空间（feature space）的混合高斯分布，并使用线性判别分析进行分类。我们构建出一套最大化马氏距离的类别中心，可以保证类间鲁棒性的最优。进一步，我们发现在视觉分类任务中通常会采用 softmax 交叉熵（softmax cross-entropy，缩写为 SCE）损失函数来训练模型。受到对比学习[131]的启发，我们希望模型在训练过程中最大化类间间距的同时最小化类内间距，从而在保证类间泛化性的基础上促进类内样本间信息的共享。尽管我们无法在输入空间中操控样本，但是我们可以改变样本在特征空间中的分布，从而诱导出样本高密度区域（high-density regions），如图 2.4中所示。

我们首先证明了 SCE 损失函数及其变体（包括 MMLDA）无法诱导出样本高密度区域。基于此，我们进一步提出最大化马氏距离中心（max-Mahalanobis center，缩写为 MMC）损失函数。在对比学习的训练过程中，类别中心或者样本中心是动态变化的[4-5,132]。而在我们的 MMC 损失函数中，每个类别的类别中心是在训练开始之前提前计算好并在训练过程中固定住的。这样可以保证类间最小马氏距离最大化，同时极大地节约计算量。我们在 MNIST[133] 以及 CIFAR-10、CIFAR-100[126] 数据集上进行实验，在多种攻击下测试我们方法的有效性，并且展示 MMC 损失函数可以和对抗训练方法[29]相结合，进一步提高模型鲁棒性。

2.2　最大化马氏距离线性判别分析

本节我们提出最大化马氏距离线性判别分析网络。我们将依次阐述关于生成式线性分类器在深度神经网络中的应用、如何构造最大化马氏距离类别中心以及其在混合高斯分布下的类间鲁棒性最优的保证。

2.2.1　生成式线性分类器

深度神经网络可以看做由一个从输入 x 到特征 z 的非线性映射，以及一个作用在特征 z 上的线性分类器组成。通常我们使用判别式的 softmax 线性分类器（也称为 softmax 回归[134]）。然而，之前的工作[130]表明生成式的线性分类器例如线性判别分析（linear discriminant analysis，缩写为 LDA）可以比判别式的 softmax 线性分类器达到更高的学习效率。以两类分类问题为例，即类别 $y \in \{0, 1\}$，此时 softmax 函数退化成 logistic 函数。①我们用 R_0 和 R_1 分别代表一个分类器对于类别 0 和 1 的决策域，那么这个分类器的错误率 ER 可以表示为

$$\mathrm{ER} = \pi_0 P(x \in R_1 | x \sim \mathcal{N}(\mu_0, \Sigma)) + \pi_1 P(x \in R_0 | x \sim \mathcal{N}(\mu_1, \Sigma)), \tag{2.1}$$

① logistic 函数应用于二分类问题，而 softmax 函数是其在多分类问题中的扩展形式。

其中 π_0 和 π_1 是类别先验，μ_0 和 μ_1 是类别中心，Σ 是共享的协方差矩阵。在这样的设定下，我们定义 logistic 分类器（也称作对数几率回归，logistic regression，简写为 LR）相比于 LDA 分类器的相对效率（relative efficiency）为

$$\mathrm{Eff}_p(\zeta, \Delta) = \lim_{N\to\infty} \frac{\mathbb{E}[\mathrm{ER}_{\mathrm{LDA}} - \mathrm{ER}_{\mathrm{Bayes}}]}{\mathbb{E}[\mathrm{ER}_{\mathrm{LR}} - \mathrm{ER}_{\mathrm{Bayes}}]}, \tag{2.2}$$

其中 $\mathrm{ER}_{\mathrm{Bayes}}$ 代表理想的 Bayes 错误率，$\Delta = [(\mu_1 - \mu_0)^\top \Sigma^{-1}(\mu_1 - \mu_0)]^{\frac{1}{2}}$ 是两类高斯分量间的马氏距离，N 是训练样本数目，$\zeta = \log(\pi_1/\pi_0)$ 为类别先验的对数比例。为了计算 $\mathrm{Eff}_p(\zeta, \Delta)$，Efron[130] 定义

$$A_i(\pi_0, \Delta) = \int_{-\infty}^{\infty} \frac{e^{-\Delta^2/8} x^i \varphi(x)}{\pi_0 e^{-\Delta x/2} + \pi_1 e^{\Delta x/2}} dx, \tag{2.3}$$

其中 $\varphi(x) = (2\pi)^{\frac{1}{2}} \exp(-x^2/2)$ 是标准高斯分布 $\mathcal{N}(0, 1)$ 的概率密度函数。由此 Efron[130] 推出

$$\mathrm{Eff}_p(\zeta, \Delta) = \frac{Q_1 + (p-1)Q_2}{Q_3 + (p-1)Q_4}, \tag{2.4}$$

其中 $Q_2 = 1 + \pi_0\pi_1\Delta^2$，$Q_4 = \frac{1}{A_0}$ 以及

$$\begin{aligned} Q_1 &= \begin{pmatrix} 1 & \frac{\zeta}{\Delta} \end{pmatrix} \begin{bmatrix} 1 + \frac{\Delta^2}{4} & (\pi_0 - \pi_1)\frac{\Delta}{2} \\ (\pi_0 - \pi_1)\frac{\Delta}{2} & 1 + 2\pi_0\pi_1\Delta^2 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{\zeta}{\Delta} \end{bmatrix}, \\ Q_3 &= \begin{pmatrix} 1 & \frac{\zeta}{\Delta} \end{pmatrix} \frac{1}{A_0A_2 - A_1^2} \begin{bmatrix} A_2 & A_1 \\ A_1 & A_0 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{\zeta}{\Delta} \end{bmatrix}. \end{aligned} \tag{2.5}$$

根据公式（2.4），我们可以发现越大的 $|\zeta|$（代表数据类别不平衡）或者 Δ（代表不同类别数据距离远）都会导致 LR 相比于 LDA 越低效。

2.2.2　构造最大化马氏距离类别中心

根据上小节的结论，我们推广到多类别情况，希望可以用 LDA 替换 softmax 线性分类器。受到 GAN 模型[135] 将输入空间的高斯或者混合高斯分布映射到特征空间的数据分布的启发，我们反之可以考虑将输入空间的数据分布映射到特征空间的混合高斯分布，并在特征空间使用 LDA，增加学习效率。具体来说，在特征空间中我们假设样本服从混合高斯分布：

$$P(y = i) = \pi_i,\ P(z|y = i) = \mathcal{N}(\mu_i, \Sigma), \tag{2.6}$$

其中 $i \in [L]$，π_i 是类别 i 对应的高斯分量先验，μ_i 是类别 i 的中心，Σ 是所有类别共享的协方差矩阵。此时任意两类 i 与 j 对应的高斯分量之间的马氏距离写为

算法 2.1 构造最大化马氏距离类别中心

输入: 最大模长平方 $\mathcal{M}$，空间维度 d 以及类别数目 L。（$L \leqslant d+1$）
初始化: $\mu_1^* = e_1$ 且 $\mu_i^* = 0_d, i \in [L] \setminus \{1\}$。这里 e_1 和 0_d 分别表示第一维坐标的单位基向量以及 d 维零向量。
for $i = 2$ **to** L **do**
  for $j = 1$ **to** $i-1$ **do**
    $\mu_i^*(j) = -[1 + \langle \mu_i^*, \mu_j^* \rangle \cdot (L-1)]/[\mu_j^*(j) \cdot (L-1)]$
  end for
  $\mu_i^*(i) = \sqrt{1 - \|\mu_i^*\|_2^2}$
end for
for $k = 1$ **to** L **do**
  $\mu_k^* = \sqrt{\mathcal{M}} \cdot \mu_k^*$
end for
输出: 最大化马氏距离类别中心 $\mu_i^*, i \in [L]$.

$\Delta_{i,j} = [(\mu_i - \mu_j)^\top \Sigma^{-1} (\mu_i - \mu_j)]^{\frac{1}{2}}$。不失一般性，我们可以认为协方差矩阵 Σ 非奇异，则根据 Cholesky 分解我们有 $\Sigma = QQ^\top$，其中 Q 是下三角矩阵且对角元素为正。令 $\overline{\mu} = \sum_{i=1}^L \mu_i / L$，我们可以通过线性变换 $\tilde{z} = Q^{-1}(z - \overline{\mu})$ 将协方差矩阵变为单位阵：

$$P(y=i) = \pi_i,\ P(\tilde{z}|y=i) = \mathcal{N}(\widetilde{\mu}_i, I), \tag{2.7}$$

其中 $\sum_{i=1}^L \widetilde{\mu}_i = 0$。在该新的坐标系下，马氏距离变为 $\widetilde{\Delta}_{i,j} = [(\widetilde{\mu}_i - \widetilde{\mu}_j)^\top (\widetilde{\mu}_i - \widetilde{\mu}_j)]^{\frac{1}{2}}$。注意到线性变换 $z \mapsto \tilde{z}$ 并不改变马氏距离，即我们有 $\widetilde{\Delta}_{i,j} = \Delta_{i,j}$。因此在下文中我们假设数据分布满足形式（2.7），并且为了符号简洁我们将 $\tilde{z}$ 写为 z，$\widetilde{\mu}_i$ 写为 μ_i，$\widetilde{\Delta}_{i,j}$ 写为 $\Delta_{i,j}$。根据 Fisher 线性判别函数[134]，LDA 对于 i 和 j 两类的分类器可以写为 $\lambda_{i,j}(z) = \beta_{i,j} + \alpha_{i,j}^\top z = 0$，其中

$$\beta_{i,j} = \log \frac{\pi_i}{\pi_j} + \frac{1}{2}(\|\mu_j\|_2^2 - \|\mu_i\|_2^2),\ \alpha_{i,j}^\top = (\mu_i - \mu_j)^\top. \tag{2.8}$$

下面我们考虑 LDA 模型的鲁棒性。我们从第 i 类的高斯分量中采样出一个真实类别为 i 的样本特征 $z_{(i)}$，即 $z_{(i)} \sim \mathcal{N}(\mu_i, I)$。我们用 $z_{(i,j)}^*$ 来表示处在决策边界 $\lambda_{i,j}(z) = 0$ 上且距离 $z_{(i)}$ 最近的点，即 $z_{(i,j)}^*$ 可以视作 $z_{(i)}$ 在 i, j 两类决策边界上的投影。我们将 $z_{(i,j)}^*$ 与 $z_{(i)}$ 的距离表示为 $d_{(i,j)}$。根据折叠高斯分布（folded Gaussian distribution）的性质，当 $\pi_i = \pi_j$ 时，我们可以推出 $\mathbb{E}[d_{(i,j)}]$ 与 $\Delta_{i,j}$ 的依赖关系（详细证明过程参见[32]）

$$\mathbb{E}[d_{(i,j)}] = \sqrt{\frac{2}{\pi}} \exp\left(-\frac{\Delta_{i,j}^2}{8}\right) + \frac{1}{2}\Delta_{i,j}\left[1 - 2\Phi(-\frac{\Delta_{i,j}}{2})\right], \tag{2.9}$$

其中 $\Phi(\cdot)$ 是标准高斯分布的累积概率密度。当 $\pi_i \neq \pi_j$ 时，类似的依赖关系仍旧成

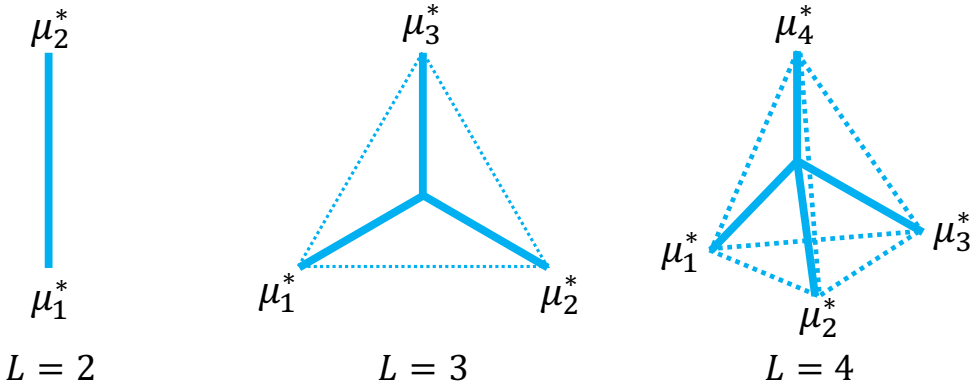


图 2.1　类别数目分别为 2、3、4 时最大化马氏距离类别中心示意图

立[32]。此外，我们证明 $\mathbb{E}[d_{(i,j)}]$ 随着 $\Delta_{i,j}$ 单调递增，具体写为

$$\frac{\partial \mathbb{E}[d_{(i,j)}]}{\partial \Delta_{i,j}} = \frac{1}{2}[1 - 2\Phi(-\frac{\Delta_{i,j}}{2})] \geqslant 0. \tag{2.10}$$

在对抗鲁棒性领域，$d_{(i,j)}$ 被认为表征分类器对于 i, j 两类的鲁棒性[15,136]，即大的 $d_{(i,j)}$ 说明真实类别是 i 的样本难以被欺骗成类别 j。在期望意义下，我们定义特征空间中的鲁棒性为

$$\mathrm{RB} = \min_{i,j\in[L]} \mathbb{E}[d_{(i,j)}], \tag{2.11}$$

即所有任意两类鲁棒间距的最小值。根据公式（2.9）的关系以及公式（2.10）中的单调递增关系，我们可以近似

$$\mathrm{RB} \approx \overline{\mathrm{RB}} = \min_{i,j\in[L]} \frac{\Delta_{i,j}}{2}. \tag{2.12}$$

公式（2.12）将鲁棒性 RB 与马氏距离 $\Delta_{i,j}$ 相联系了起来。可以看出，若想得到鲁棒的模型，我们需要保证任意两类间的*最小马氏距离最大化*。由于类别数目 L 是有限值，我们总可以定义 $\max_i \|\mu_i\|_2^2 = \mathcal{M}$，此时我们可以推出

$$\overline{\mathrm{RB}} \leqslant \sqrt{\frac{L\mathcal{M}}{2(\mathcal{M}-1)}}, \tag{2.13}$$

其中等式成立当且仅当

$$\mu_i^\top \mu_j = \begin{cases} \mathcal{M}, & i = j, \\ \frac{\mathcal{M}}{1-L}, & i \neq j. \end{cases} \tag{2.14}$$

我们将满足公式（2.14）的任意一组类别中心表示为 $\{\mu_i^*\}$，并称作*最大化马氏距离类别中心*。当 $L \leqslant d+1$ 时（d 是特征空间维度），根据算法 2.1可以构造出一组最大化马氏距离类别中心。对于构造出的 $\{\mu_i^*\}$，任何正交变换之后仍然构成一组最大化马氏距离类别中心。如图 2.1中所示，当 $L = 3$ 时，三个类别中心构成一个等边

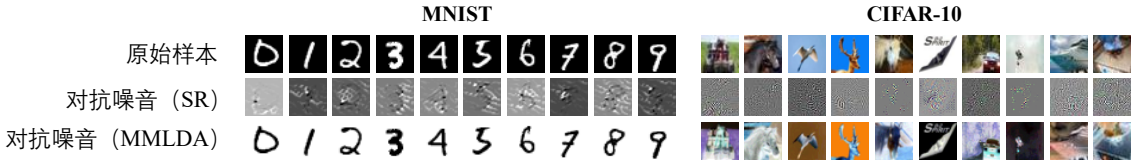


图 2.2　softmax 回归（SR）以及我们的 MMLDA 算法上构造的对抗噪音

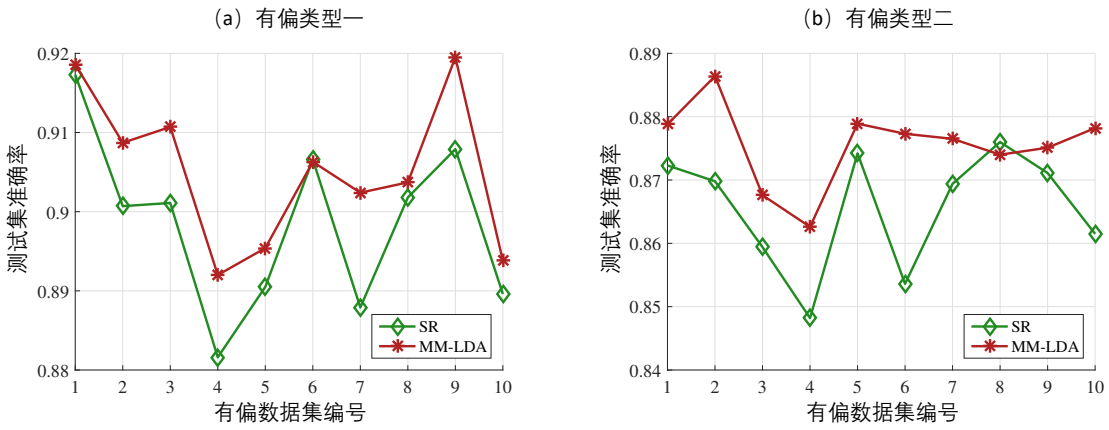


图 2.3　不同类别训练样本数目不均衡的情况

三角形；当 $L=4$ 时，四个类别中心构成一个正四面体。由此，我们最终可以构建最大化马氏距离判别分析（MMLDA）网络。根据 Bayes 准则，我们得到 MMLDA 网络的分类函数为

$$P(y=k|z)=\frac{P(z|y=k)P(y=k)}{P(z)}=\frac{\pi_k\mathcal{N}(z|\mu_k^*,I)}{\sum_{i=1}^{L}\pi_i\mathcal{N}(z|\mu_i^*,I)}. \tag{2.15}$$

在图 2.2中，我们分别使用 softmax 回归（SR）以及 MMLDA 训练模型，并对其进行对抗攻击。可以看到，攻击 SR 训练的模型产生的对抗噪音类似于白噪音，而攻击我们的 MMLDA 训练出的模型产生的对抗噪音有明显的语义（semantic 或者 perceptually-aligned）特征。后续的一系列研究[39,100,137-138]都发现了类似的现象，即对于鲁棒的模型构造出来的对抗噪音会存在明显的语义特征。此外，在图 2.3中，我们测试 MMLDA 在有偏数据集上的性能。我们对于 CIFAR-10[126]数据集进行重采样。令 $\alpha=(\alpha_0,\cdots,\alpha_9)$，重采样过后的数据集类别先验满足 $\mathbb{E}[\hat{\pi}_k]=\alpha_k/\|\alpha\|_1$。具体地，我们模拟两种代表性的有偏数据集类型：第一种是 $\alpha=(0.1,0.2,0.3,\cdots,1.0)$；第二种是 $\alpha=(0.2,\cdots,0.2,1.0)$。对于每种类型，我们都随机排序 CIFAR-10 中的类别顺序，从而一种 α 可以得到十种（甚至更多）不同的有偏数据集。更详细的实验设定参见[32]，这里不再赘述。由实验结果可以看到，MMLDA 相比于 softmax 回归可以在有偏数据集上一致地提升测试准确率。这印证了公式（2.4）中越大的 $|\zeta|$，即数据越不均衡，softmax 回归相比于 LDA 越低效的结论。

2.3　最大化马氏距离中心损失函数

尽管 MMLDA 方法相比于 softmax 回归更加高效,但是他们两者都属于 softmax 交叉熵（即 SCE）损失函数的形式。在本节中，我们希望通过在特征空间中诱导出样本高密度区域来进一步提高样本使用效率。我们首先说明 SCE 损失函数及其变体无法满足诱导出样本高密度区域的要求，并由此改进 MMLDA 并提出最大化马氏距离中心损失（MMC）函数。

2.3.1　一般形式的 SCE 损失函数

在分类问题中，我们用 L 表示类别数目，并定义 softmax 函数为 $\mathrm{softmax}(h)$：$\mathbb{R}^L \to \mathbb{R}^L$，具体形式为

$$\mathrm{softmax}(h)_i = \frac{\exp(h_i)}{\sum_{l=1}^{L}\exp(h_l)}, \tag{2.16}$$

其中 $i \in [L]$，$[L] := \{1, \cdots, L\}$，且 h 被称为分对数（logit）。深度神经网络模型学习一个从输入 $x \in \mathbb{R}^p$ 到特征 $z = Z(x) \in \mathbb{R}^d$ 的非线性映射。在此设定下，最常用的 SCE 损失函数可以写为

$$\mathcal{L}_{\mathrm{SCE}}(Z(x), y) = -1_y^{\top}\log\left[\mathrm{softmax}(Wz + b)\right], \tag{2.17}$$

其中 y 是输入的类别标签，1_y 是 y 的热编码（one-hot encoding）。这里 W 和 b 分别是权重矩阵和偏置。为了能够囊括 SCE 损失函数的相关变体，我们定义一般化 SCE 损失函数（generalized SCE, 缩写为 g-SCE）如下

$$\mathcal{L}_{\text{g-SCE}}(Z(x), y) = -1_y^{\top}\log\left[\mathrm{softmax}(h)\right], \tag{2.18}$$

其中 $h = H(z) \in \mathbb{R}^L$ 是通过作用在特征 z 上的变换得到。例如，当 $h = Wz + b$ 是 z 的线性函数时，公式（2.18）则退化成公式（2.17）的形式。在本章中，我们限制 h 是 z 的二次函数，即

$$h_i = -(z - \mu_i)^{\top}\Sigma_i(z - \mu_i) + B_i, \tag{2.19}$$

其中 $\mu_i \in \mathbb{R}^d$，$\Sigma_i \in \mathbb{R}^{d\times d}$，$B_i \in \mathbb{R}$。这种形式包含了 SCE 的多种变体[32,139]（包括 MMLDA），并且注意到公式（2.17）也可以用二次形式表示为

$$\begin{aligned}\mathrm{softmax}(Wz + b)_i &= \frac{\exp(W_i^{\top}z + b_i)}{\sum_{l\in[L]}\exp(W_l^{\top}z + b_l)} \\ &= \frac{\exp(-\|z - \frac{1}{2}W_i\|_2^2 + b_i + \frac{1}{4}\|W_i\|_2^2)}{\sum_{l\in[L]}\exp(-\|z - \frac{1}{2}W_l\|_2^2 + b_l + \frac{1}{4}\|W_l\|_2^2)}.\end{aligned} \tag{2.20}$$

以物理中有势场来做类比，损失函数可以看做是在特征空间中定义了一种势

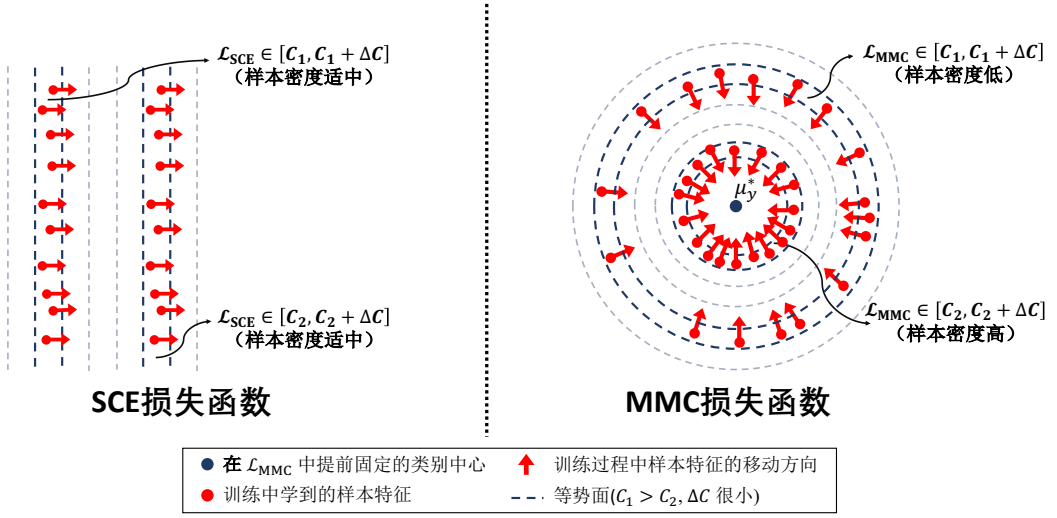


图 2.4　SCE 与 MMC 损失函数在特征空间中诱导出的样本密度示意图

能。而训练模型参数的过程，可以等效看作样本在特征空间中沿着垂直于等势面的方向移动（我们不考虑优化器的动量项从而化简直观图景）。由此，我们首先分析 g-SCE 损失函数在特征空间中的等势面，即 $\mathcal{L}_{\text{g-SCE}}(z, y)$ 等于某一给定常数 $C \in (0, +\infty)$ 时，z 的闭式解。令 $C_e = \exp(C)$，我们可以推出

$$\log\left(1 + \frac{\sum_{l \neq y} \exp(h_l)}{\exp(h_y)}\right) = C \implies h_y = \log\left[\sum_{l \neq y} \exp(h_l)\right] - \log(C_e - 1). \quad (2.21)$$

注意到公式（2.21）的右侧有一项 Log-Sum-Exp（缩写为 LSE）函数，即 $\log\left[\sum_{l \neq y} \exp(h_l)\right]$。LSE 函数常用来作为最大值函数（maximum function）的平滑估计[140]，反之这里我们也可以将 LSE 函数近似替换为最大值函数，并进一步得到

$$h_y - h_{\tilde{y}} = -\log(C_e - 1), \quad (2.22)$$

其中 $\tilde{y} = \text{argmax}_{l \neq y} h_l$ 代表除了真实类别 y 以外，剩余类别中对应最大分对数值的类别。这里我们用红色来帮助区分 $\tilde{y}$ 和 y。根据公式（2.22）的形式，我们可以定义 $\mathcal{L}_{y,\tilde{y}}(z) = \log[\exp(h_{\tilde{y}} - h_y) + 1]$ 作为 g-SCE 损失函数在 z 周围的近似，即等势面 $\mathcal{L}_{\text{g-SCE}}(z, y) = C$ 可以用 $\mathcal{L}_{y,\tilde{y}}(z) = C$ 来近似。

2.3.2　g-SCE 损失函数诱导出的样本密度

当协方差矩阵 Σ_i 等于 σ_i 乘以单位矩阵 I 时，我们可以推出对于任意两个类别 i 和 j 以及一个常数 c，如果 $\sigma_i \neq \sigma_j$，则等势面 $h_i - h_j = c$ 在 d 维特征空间中的解是一个 $d - 1$ 维的超球面，闭式表达为

$$\|z - \mathbf{M}_{i,j}\|_2^2 = \mathbf{B}_{i,j} - \frac{c}{\sigma_i - \sigma_j}, \quad (2.23)$$

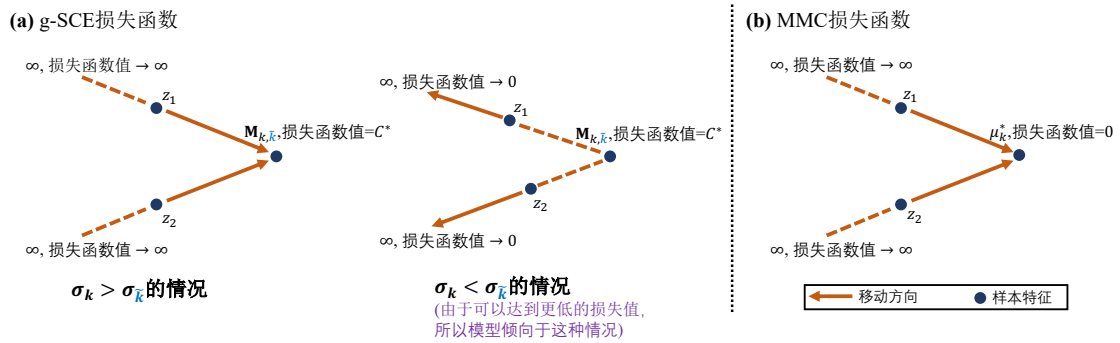


图 2.5　SCE 与 MMC 损失函数在特征空间中的训练机理展示

其中 $\mathbf{M}_{i,j}$ 以及 $\mathbf{B}_{i,j}$ 可以由公式（2.19）中的参数表示为

$$\mathbf{M}_{i,j} = \frac{\sigma_i\mu_i - \sigma_j\mu_j}{\sigma_i - \sigma_j}, \mathbf{B}_{i,j} = \frac{\sigma_i\sigma_j\|\mu_i - \mu_j\|_2^2}{(\sigma_i - \sigma_j)^2} + \frac{B_i - B_j}{\sigma_i - \sigma_j}. \tag{2.24}$$

注意到当 $\mathbf{B}_{i,j} < (\sigma_i - \sigma_j)^{-1}c$ 时，问题（2.23）无解。特别地，若 $\sigma_i = \sigma_i = \sigma$，则 $h_i - h_j = c$ 的解退化成超平面

$$z^\top(\mu_i - \mu_j) = \frac{1}{2}\left[\|\mu_i\|_2^2 - \|\mu_j\|_2^2 + \frac{B_j - B_i + c}{\sigma}\right]. \tag{2.25}$$

例如对于 SCE 损失函数，其等势面为 $z^\top(W_i - W_j) = b_j - b_i + c$。对于更一般形式的协方差矩阵 Σ_i，其对应的等势面在合适的坐标变换下可以写成超椭球面。

下面我们研究 g-SCE 损失函数在特征空间中会诱导出怎样的样本密度。首先，我们用 $\mathcal{D}$ 代表数据集，并用 $\mathcal{D}_{k,\tilde{k}} = \{(x,y) \in \mathcal{D} | y = k, \tilde{y} = \tilde{k}\}$ 表示数据集中真实类别为 k，且除了真实类别以外最大预测类别为 $\tilde{k}$ 的部分。类似地，这里我们用蓝色来帮助区分 k 和 $\tilde{k}$。在 $\mathcal{D}_{k,\tilde{k}}$ 中的样本数量我们用 $N_{k,\tilde{k}} = |\mathcal{D}_{k,\tilde{k}}|$ 来表示。根据公式（2.22）和公式（2.23），并且将 $y = k$, $\tilde{y} = \tilde{k}$ 代入，我们得到 $\mathcal{L}_{k,\tilde{k}} = C$ 的解为

$$\|z - \mathbf{M}_{k,\tilde{k}}\|_2^2 = \mathbf{B}_{k,\tilde{k}} + \frac{\log(C_e - 1)}{\sigma_k - \sigma_{\tilde{k}}}. \tag{2.26}$$

令 $\Delta B_{k,\tilde{k}} = \{z \in \mathbb{R}^d | \mathcal{L}_{k,\tilde{k}} \in [C, C + \Delta C]\}$ 为 $\mathcal{L}_{k,\tilde{k}}$ 相邻两个等势面之间的体积，则根据超球面体积公式[141]，我们得到

$$\mathrm{Vol}(\Delta B_{k,\tilde{k}}) = \frac{2\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})}\left(\mathbf{B}_{k,\tilde{k}} + \frac{\log(C_e - 1)}{\sigma_k - \sigma_{\tilde{k}}}\right)^{\frac{d-1}{2}} \cdot \Delta C, \tag{2.27}$$

其中 $\Gamma(\cdot)$ 是 gamma 函数。假设 g-SCE 损失函数值在训练某一阶段满足分布 $\mathcal{L}_{\text{g-SCE}} \sim p_{k,\tilde{k}}(c)$，则数据集对应于损失函数取值 $[C, C + \Delta C]$ 范围内约有 $\Delta N_{k,\tilde{k}} = N_{k,\tilde{k}} \cdot p_{k,\tilde{k}}(C) \cdot \Delta C$ 个样本。那么我们可以求出在使用 g-SCE 损失函数时，z 附近的

样本密度 $\mathbb{SD}(z)$ 为

$$\mathbb{SD}(z) \approx \frac{\Delta N_{k,\tilde{k}}}{\mathrm{Vol}(\Delta B_{k,\tilde{k}})} \propto \frac{N_{k,\tilde{k}} \cdot p_{k,\tilde{k}}(C)}{\left[\mathbf{B}_{k,\tilde{k}} + \frac{\log(C_e-1)}{\sigma_k-\sigma_{\tilde{k}}}\right]^{\frac{d-1}{2}}}. \tag{2.28}$$

根据公式（2.28），我们令 $C^* = \log(1+\exp(\mathbf{B}_{k,\tilde{k}}(\sigma_{\tilde{k}}-\sigma_k)))$ 且 $C_e^* = \exp(C^*)$，满足 $\mathbf{B}_{k,\tilde{k}} + \frac{\log(C_e^*-1)}{\sigma_k-\sigma_{\tilde{k}}} = 0$。此时，若 $\sigma_k > \sigma_{\tilde{k}}$，那么 C^* 就成为 C 的下界，也就是说 $\mathcal{L}_{\text{g-SCE}} = C < C^*$ 在特征空间中的解为空集。由此我们可以推断，模型在其训练过程中会尽量避免这种（$\sigma_k > \sigma_{\tilde{k}}$）情况，因为损失函数值无法优化到小于 C^*。另一方面，当 $\sigma_k < \sigma_{\tilde{k}}$ 时，损失函数值 C 可以被优化到零。但是当 $C \to 0$ 时，我们根据公式（2.28）会发现样本密度也趋于零，因为 $\mathbf{B}_{k,\tilde{k}} + \frac{\log(C_e-1)}{\sigma_k-\sigma_{\tilde{k}}} \to \infty$。这说明样本在特征空间中会趋向于跑到无穷远的地方，远离类别中心 $\mathbf{M}_{k,\tilde{k}}$。在图 2.5中我们形象地展示了上述训练机理。在人脸识别领域，很多工作都观察到样本特征分布会形成放射型的样子[139,142-143]，其背后的机理就是由于在使用 g-SCE 系列损失函数的时候，样本特征会被鼓励跑向无穷远处进而减小损失值。

2.3.3　中心化——去掉 softmax 函数

上述两个小节阐述了 g-SCE 损失函数会诱使样本特征趋向于无穷远处。这一现象本质上是由于 softmax 归一化导致的。具体来说，softmax 归一化之后损失函数值仅依赖于不同类别分对数之间的相对大小。为了得到更低的损失函数值，分对数的绝对大小会不断变大，但是决策域却很早就收敛了，造成无意义的训练消耗。在实际训练中，通常我们都会采样权重衰减（weight decay），所以分对数（特征乘以 softmax 层的权重）在增长到一定数值之后便和权重衰减性趋于平衡。

回顾 MMLDA 方法的决策公式（2.15），我们可以将其展开写成

$$\begin{aligned}\mathcal{L}_{\text{MMLDA}}(Z(x),y) &= -\log\left[\frac{\exp(-\frac{\|z-\mu_y^*\|_2^2}{2})}{\sum_{l\in[L]}\exp(-\frac{\|z-\mu_l^*\|_2^2}{2})}\right] \\ &= -\log\left[\frac{\exp(z^\top\mu_y^*)}{\sum_{l\in[L]}\exp(z^\top\mu_l^*)}\right].\end{aligned} \tag{2.29}$$

可以看到 MMLDA 相比于 SCE 来讲，丢掉了偏置项 b，并且将可学习的权重 W 替换成了预先构造好的最大化马氏距离类别中心 $\{\mu_i^*\}_{i\in[L]}$。简单地把 softmax 归一化替换成中心损失函数，我们可以得到最大化马氏距离中心（MMC）损失函数：

$$\mathcal{L}_{\text{MMC}}(Z(x),y) = \frac{1}{2}\|z-\mu_y^*\|_2^2. \tag{2.30}$$

MMC 损失函数的等势面 $\mathcal{L}_{\text{MMC}}(Z(x),y) = C$ 形式非常简洁，也是 $d-1$ 维的超球面。类似于公式（2.27），我们可以算出 MMC 相邻两个等势面 $\mathcal{L}_{\text{MMC}}(Z(x),y) = C$

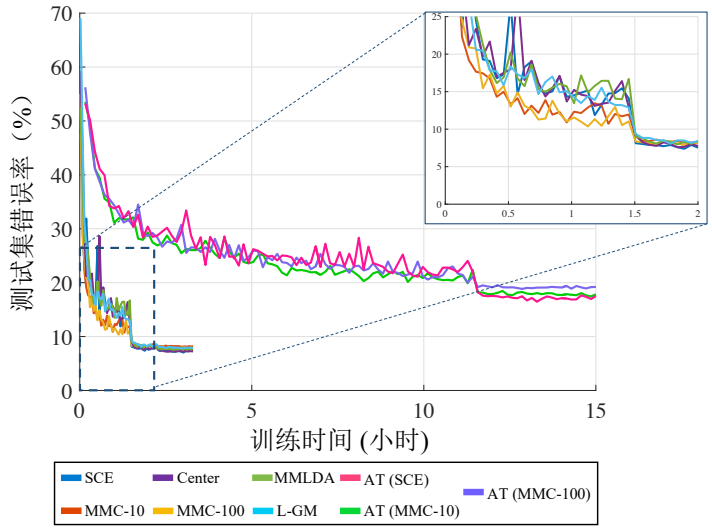


图 2.6　使用不同损失函数时测试错误率随训练时间的下降趋势

到 $\mathcal{L}_{\mathrm{MMC}}(\boldsymbol{Z}(x), y) = \boldsymbol{C} + \Delta\boldsymbol{C}$ 之间的体积 $\mathrm{Vol}(\Delta\boldsymbol{B})$ 为

$$\mathrm{Vol}(\Delta\boldsymbol{B}) = \frac{2^{\frac{d+1}{2}}\pi^{\frac{d}{2}}\boldsymbol{C}^{\frac{d-1}{2}}}{\Gamma(\frac{d}{2})} \cdot \Delta\boldsymbol{C}. \tag{2.31}$$

由此得出 MMC 损失函数诱导出来的在 z 附近的样本密度 $\mathbb{SD}(z)$ 为

$$\mathbb{SD}(z) \propto \frac{N_k \cdot p_k(\boldsymbol{C})}{\boldsymbol{C}^{\frac{d-1}{2}}}. \tag{2.32}$$

在图 2.4中我们给出了 2 维情况下 SCE 和 MMC 损失函数导致的样本密度分布。可以看到，MMC 损失函数可以更高效地聚集类内样本。

2.4　实验结果

在本小节中，我们在 MNIST[133]，CIFAR-10 以及 CIFAR-100[126]数据集上展示 MMC 损失函数在多方面的优越性能。作为基线模型，我们主要对比 SCE[144]，Center loss[4]，MMLDA[32] 以及 L-GM[139]方法。①

我们采用 ResNet-32 模型结构[144]。在 MMC 损失函数中，根据交叉验证的结果[32]，我们取最大类别中心模长 $\sqrt{\mathcal{M}} = 10$，写为 MMC-10。基线方法的超参数设置均与其原论文中保持一致[4,32,139]。输入图片的像素值标准化到 [0, 1] 区间。对所有的方法，我们均采用动量随机梯度下降（momentum SGD[145]）优化器，以及初始学习率 0.01。在 MNIST 上我们训练 40 轮；在 CIFAR-10 及 CIFAR-100 上我们训练 200 轮，且学习率会在第 100 和 150 轮的时候分别乘以衰减因子 0.1。在图 2.6中，我们展示了在 CIFAR-10 数据集上，使用不同损失函数时干净样本（clean examples）

① 源代码参见https://github.com/P2333/Max-Mahalanobis-Training。

表 2.1　CIFAR-10 上不同的模型结构在 PGD 攻击下的鲁棒准确率（%）

训练方法	扰动大小 $\epsilon = 8/255$				扰动大小 $\epsilon = 16/255$			
	$\text{PGD}_{10}^{\text{tar}}$	$\text{PGD}_{10}^{\text{un}}$	$\text{PGD}_{50}^{\text{tar}}$	$\text{PGD}_{50}^{\text{un}}$	$\text{PGD}_{10}^{\text{tar}}$	$\text{PGD}_{10}^{\text{un}}$	$\text{PGD}_{50}^{\text{tar}}$	$\text{PGD}_{50}^{\text{un}}$
CIFAR-10								
SCE (Res.32)	$\leqslant 1$	3.7	$\leqslant 1$	3.6	$\leqslant 1$	2.7	$\leqslant 1$	2.9
MMC (Res.32)	**48.7**	**36.0**	**26.6**	**24.8**	**36.1**	**25.2**	**13.4**	**17.5**
SCE (Res.110)	$\leqslant 1$	3.0	$\leqslant 1$	2.9	$\leqslant 1$	2.1	$\leqslant 1$	2.0
MMC (Res.110)	**54.7**	**46.0**	**34.4**	**31.4**	**41.0**	**30.7**	**16.2**	**21.6**
CIFAR-100								
SCE (Res.32)	$\leqslant 1$	7.8	$\leqslant 1$	7.4	$\leqslant 1$	4.8	$\leqslant 1$	4.7
MMC (Res.32)	**23.9**	**23.4**	**15.1**	**21.9**	**16.4**	**16.7**	**8.0**	**15.7**
SCE (Res.110)	$\leqslant 1$	7.5	$\leqslant 1$	7.3	$\leqslant 1$	4.7	$\leqslant 1$	4.5
MMC (Res.110)	**34.6**	**22.4**	**23.7**	**16.5**	**24.1**	**14.9**	**13.9**	**10.5**

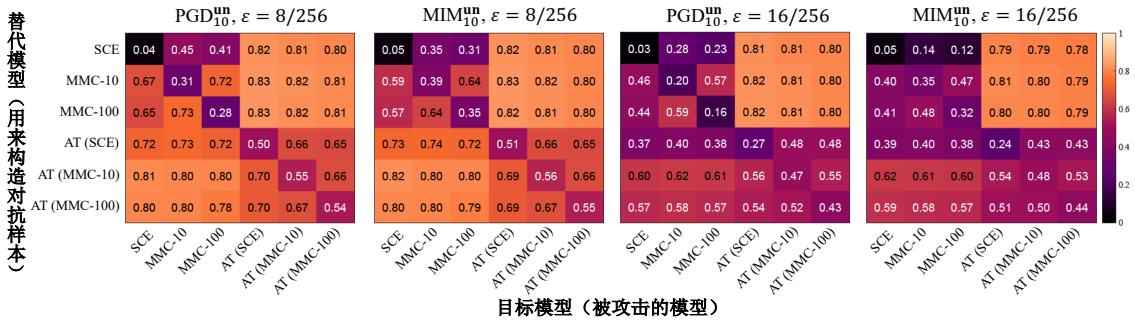


图 2.7　CIFAR-10 上黑盒迁移攻击准确率

上测试错误率随训练时间的下降曲线。可以看到，使用 MMC 损失函数可以相比于其他基线方法更快地收敛，并且得到相似甚至更低的测试错误率。

2.4.1　白盒攻击下的鲁棒性

在鲁棒性方面，我们首先考虑白盒（white-box）攻击设定。我们将对抗扰动幅度限制在 ℓ_∞-范数意义下，并且使用 PGD 攻击方法[29]对训练好的模型进行攻击，结果如表 2.1以及表 2.2中所示。根据之前的对于鲁棒性评估的建议流程[46]，我们考虑多种攻击设定的不同组合，包括扰动大小 ϵ，攻击迭代步数，以及无目标（untargeted）攻击模式和有目标（targeted）攻击模式。具体来说，我们设定 ϵ 等于

表 2.2　CIFAR-10 上不同训练方法得到的模型在 PGD 攻击下的鲁棒准确率（%）

训练方法	扰动大小 $\epsilon = 8/255$				扰动大小 $\epsilon = 16/255$			
	$\text{PGD}_{10}^{\text{tar}}$	$\text{PGD}_{10}^{\text{un}}$	$\text{PGD}_{50}^{\text{tar}}$	$\text{PGD}_{50}^{\text{un}}$	$\text{PGD}_{10}^{\text{tar}}$	$\text{PGD}_{10}^{\text{un}}$	$\text{PGD}_{50}^{\text{tar}}$	$\text{PGD}_{50}^{\text{un}}$
SCE	$\leqslant 1$	3.7	$\leqslant 1$	3.6	$\leqslant 1$	2.9	$\leqslant 1$	2.6
Center loss	$\leqslant 1$	4.4	$\leqslant 1$	4.3	$\leqslant 1$	3.1	$\leqslant 1$	2.9
MMLDA	$\leqslant 1$	16.5	$\leqslant 1$	9.7	$\leqslant 1$	6.7	$\leqslant 1$	5.5
L-GM	37.6	19.8	8.9	4.9	26.0	11.0	2.5	2.8
MMC-10 (rand)	43.5	29.2	20.9	18.4	31.3	17.9	8.6	11.6
MMC-10	**48.7**	**36.0**	**26.6**	**24.8**	**36.1**	**25.2**	**13.4**	**17.5**
$\text{AT}_{10}^{\text{tar}}$ (SCE)	**70.6**	49.7	**69.8**	47.8	48.4	26.7	31.2	16.0
$\text{AT}_{10}^{\text{tar}}$ (**MMC-10**)	69.2	**54.8**	67.0	**53.5**	**58.6**	**47.3**	**44.7**	**45.1**
$\text{AT}_{10}^{\text{un}}$ (SCE)	69.8	55.4	69.4	53.9	53.3	34.1	38.5	21.5
$\text{AT}_{10}^{\text{un}}$ (**MMC-10**)	**70.8**	**56.3**	**70.1**	**55.0**	**54.7**	**37.4**	**39.9**	**27.7**

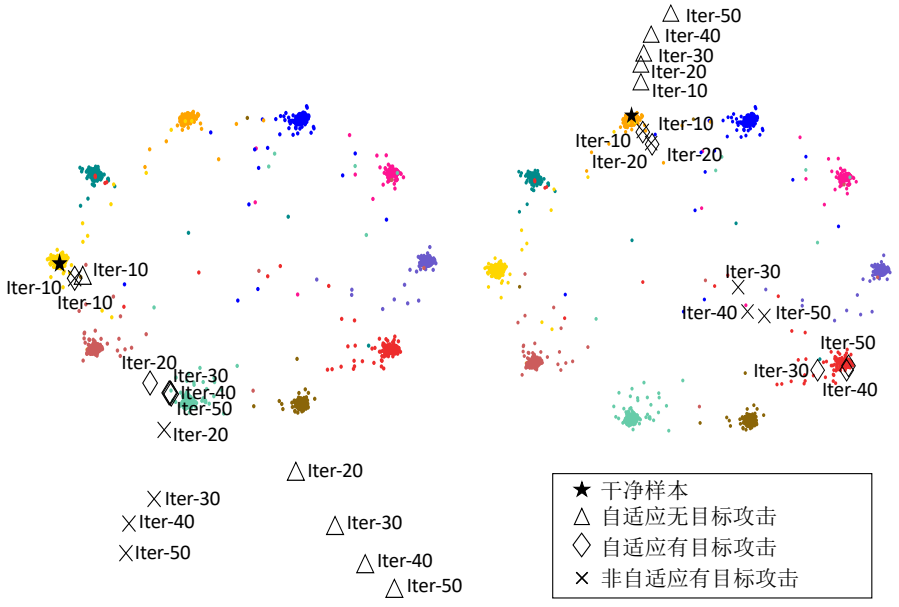


图 2.8　MNIST 上自适应攻击示意图

8/255 以及 16/255，且攻击步长为 2/255。攻击迭代步数设定为 10 步（PGD-10）以及 50 步（PGD-50）。在每个实验评估中我们均测试多个随机初始化（random restarts），保证结果的可复现性。我们也进行了消融实验，将 MMC 损失函数中的最大化马氏距离类别中心替换成随机选取的类别中心，即 MMC-10（rand）。从表 2.2中的结果可以看到，即便使用随机的类别中心，相比基线模型，MMC 使用的提前固

表 2.3　CIFAR-10 上不同训练方法得到的模型在 CW 攻击（部分一）、SPSA 攻击（部分二）以及一般扰动（部分三）下的准确率（%）

训练方法	部分一		部分二 ($\epsilon=8/255$)		部分二 ($\epsilon=16/255$)		部分三	
	$\text{C\&W}^{\text{tar}}$	$\text{C\&W}^{\text{un}}$	$\text{SPSA}_{10}^{\text{tar}}$	$\text{SPSA}_{10}^{\text{un}}$	$\text{SPSA}_{10}^{\text{tar}}$	$\text{SPSA}_{10}^{\text{un}}$	Noise	Rotation
SCE	0.12	0.07	12.3	1.2	5.1	$\leqslant 1$	52.0	83.5
Center loss	0.13	0.07	21.2	6.0	10.6	2.0	55.4	84.9
MMLDA	0.17	0.10	25.6	13.2	11.3	5.7	57.9	84.8
L-GM	0.23	0.12	61.9	45.9	46.1	28.2	59.2	82.4
MMC-10	**0.34**	**0.17**	**69.5**	**56.9**	**57.2**	**41.5**	**69.3**	**87.2**
$\text{AT}_{10}^{\text{tar}}$ (SCE)	1.19	0.63	**81.1**	67.8	**77.9**	59.4	82.2	**76.0**
$\text{AT}_{10}^{\text{tar}}$ (**MMC-10**)	**1.91**	**0.85**	79.1	**69.2**	74.5	**62.7**	**83.5**	75.2
$\text{AT}_{10}^{\text{un}}$ (SCE)	1.26	0.68	78.8	67.0	73.7	60.3	78.9	73.7
$\text{AT}_{10}^{\text{un}}$ (**MMC-10**)	**1.55**	**0.73**	**80.4**	**69.6**	**74.6**	**62.4**	**80.3**	**75.8**

定类别中心的策略依旧可以极大地提高鲁棒准确率。而通过对比 MMC-10（rand）与 MMC-10 的结果，我们可以看出使用最大化马氏距离类别中心可以进一步提升鲁棒性。最后，当我们将 MMC 损失函数用于对抗训练（adversarial training，缩写为 AT）[29]中时，可以对训练中未见过的攻击（例如 $\epsilon = 16/255$ 情况下的 $\text{PGD}_{50}^{\text{un}}$）有更好的防御效果。此外，我们在 ℓ_2-范数意义下进行测试。在表 2.3部分一以及表 2.4部分三中我们采用了 C&W[146]攻击，通过二分查找（binary search）找到能成功攻击模型所需的最小扰动。我们分别测试了无目标攻击和有目标攻击的情况。参数设置方面，我们选取二分查找步数为 9，初始化平衡参数 $c = 0.01$（详细意义参见[146]）。在每次二分查找过程中，C&W 攻击的迭代轮数设为 1000，学习率为 0.005。从结果中可以看到，成功攻击 MMC 训练出来的模型所需的最小扰动要显著大于攻击其他基线模型所需的最小扰动。

2.4.2　黑盒攻击下的鲁棒性

除了白盒攻击以外，测试模型在黑盒攻击下的鲁棒性也是评估的重要环节之一。我们首先测试基于 PGD 以及 MIM[27]的迁移攻击。由于有目标攻击的迁移能力一般较弱[56]，因此我们主要测试无目标迁移攻击，结果展示在图 2.7中。此外，我们还测试了非基于真实梯度（gradient-free）的攻击算法 SPSA[147]，结果展示在表 2.3的部分二以及表 2.4的部分二中。在 SPSA 中为了近似估计梯度，我们选取批

表 2.4　CIFAR-100 上不同训练方法得到的模型在正常样本（部分一）、PGD 及 SPSA 攻击（部分二）以及 C&W 攻击（部分三）下的准确率（%）

训练方法	部分一	部分二 ($\epsilon = 8/255$)				部分三	
	Clean	$\text{PGD}_{10}^{\text{tar}}$	$\text{PGD}_{10}^{\text{un}}$	$\text{SPSA}_{10}^{\text{tar}}$	$\text{SPSA}_{10}^{\text{un}}$	$\text{C\&W}^{\text{tar}}$	$\text{C\&W}^{\text{un}}$
SCE	72.9	$\leqslant 1$	8.0	14.0	1.9	0.16	0.047
Center	72.8	$\leqslant 1$	10.2	14.7	2.3	0.18	0.048
MMLDA	72.2	$\leqslant 1$	13.9	18.5	5.6	0.21	0.050
L-GM	71.3	15.8	15.3	22.8	7.6	0.31	0.063
MMC-10	71.9	**23.9**	**23.4**	**33.4**	**15.8**	**0.37**	**0.085**

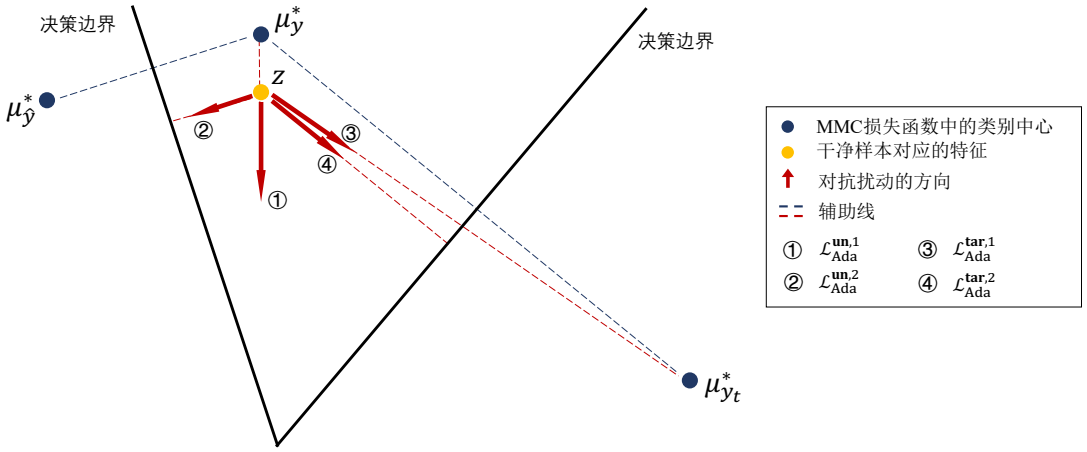


图 2.9　不同的自适应攻击作用机理

量大小为 128, 学习率为 0.01, 以及有限差分（finite difference）的扰动大小 $\delta = 0.01$。更大批量大小下 SPSA 的攻击结果详见[34]。在上述的在黑盒攻击下模型鲁棒性的结果中我们看到，相比于基线方法，MMC 损失函数依旧可以达到明显的提升，并且排除了梯度混淆（gradient obfuscation）的因素[48]。

2.4.3　自适应攻击下的鲁棒性

自适应攻击（adaptive attacks[48]）被认为是评估一个防御是否可以提供可靠鲁棒性的重要手段。如图 2.9所展示，我们设计两类自适应攻击使用的目标函数，包括无目标类型：$\mathcal{L}_{\text{Ada}}^{\text{un},1} = -\mathcal{L}_{\text{MMC}}(z, y)$，$\mathcal{L}_{\text{Ada}}^{\text{un},2} = \mathcal{L}_{\text{MMC}}(z, \tilde{y}) - \mathcal{L}_{\text{MMC}}(z, y)$；以及有目标类型 $\mathcal{L}_{\text{Ada}}^{\text{tar},1} = \mathcal{L}_{\text{MMC}}(z, y_t)$; $\mathcal{L}_{\text{Ada}}^{\text{tar},2} = \mathcal{L}_{\text{MMC}}(z, y_t) - \mathcal{L}_{\text{MMC}}(z, y)$，其中 y_t 是目标类别。当我们希望减少真实类别的预测置信度时，$\mathcal{L}_{\text{Ada}}^{\text{un},1}$ 是最强的攻击目标函数；当我们希望误导模型返回错误类别或者目标类别时，$\mathcal{L}_{\text{Ada}}^{\text{un},2}$ 与 $\mathcal{L}_{\text{Ada}}^{\text{tar},2}$ 分别是最优选择；当防御模型包含其他的机制例如对抗样本检测时，$\mathcal{L}_{\text{Ada}}^{\text{tar},1}$ 可以提供更强力的攻击效

果。

在图 2.8中，我们进一步在 MNIST 上展示了自适应攻击和非自适应攻击的迭代轨迹。每种颜色的测试样本对应于 MNIST 中的某一类。对于自适应无目标攻击，我们使用 $\mathcal{L}_{\text{Ada}}^{\mathbf{un},1} = -\mathcal{L}_{\text{MMC}}(z, y)$；对于自适应有目标攻击，我们使用 $\mathcal{L}_{\text{Ada}}^{\mathbf{tar},1} = \mathcal{L}_{\text{MMC}}(z, y_t)$。从结果可以看到，自适应攻击的迭代轨迹相比于非自适应攻击更加高效。在我们的上述实验中，$\mathcal{L}_{\text{Ada}}^{\mathbf{tar},1}$ 以及 $\mathcal{L}_{\text{Ada}}^{\mathbf{un},1}$ 被使用在 PGD，MIM 以及 SPSA 攻击的实现中；$\mathcal{L}_{\text{Ada}}^{\mathbf{tar},2}$ 以及 $\mathcal{L}_{\text{Ada}}^{\mathbf{un},2}$ 被使用在 C&W 攻击的实现中。

2.5　本章小结

本章阐述了最大化马氏距离学习范式。我们首先从 LDA 与 SR 分类器的相对效率出发，提出在特征空间建模混合高斯分布。我们推导出混合高斯分布中心与模型鲁棒性的关系，并由此得出鲁棒性最优的最大化马氏距离中心以及其构造算法，得到 MMLDA 方法。在此基础上，我们发现 softmax 归一化操作会导致样本特征趋向于无穷远处，导致样本分布稀疏（即样本密度低）。为了解决这一问题，我们进一步改进了 MMLDA 方法，将其 softmax 归一化操作替换成中心化函数，得到 MMC 损失函数。MMC 损失函数可以在保证类间鲁棒性最优的前提下，提高类内样本密度，促进样本间信息的共享，提高训练效率和模型鲁棒性。实验结果也在各种对抗攻击场景及方法下验证了 MMC 损失函数的有效性。

第 3 章　集成模型的多样性增强鲁棒学习

在很多深度学习的应用场景中，我们会使用多个单模型构成的集成模型来进一步提高系统的预测能力。然而，若单模型存在对抗鲁棒性缺陷，则由其集成后的模型同样会无法给出鲁棒的预测。为了解决这一问题，之前的大部分工作都聚焦于提升单个模型的鲁棒性，并将多个相对鲁棒的单模型直接集成起来构成系统。这样做会忽视单模型之间的相互作用，使得多个单模型容易被某些对抗扰动同时欺骗，降低集成的效果。为此，在本章中我们提出集成模型的自适应多样性增强（adaptive diversity promoting，缩写为 ADP）学习方法，通过鼓励不同单模型成员之间的非最大预测多样性来避免其被同时攻破，从而提升集成预测的鲁棒性。实验结果表明，我们的 ADP 方法几乎不需要额外的计算开销，并且在保证干净样本上的正常准确率的同时，可以有效地防御多种攻击方法。

3.1　本章引言

深度学习技术在很多实际落地的应用场景中都扮演了重要角色。但是近年来的一系列工作表明，深度学习模型容易受到对抗性攻击[16,48,146,148]。对抗攻击可以生成具有人类难以察觉的扰动的图像（即对抗样本）来欺骗模型。为了提高模型的对抗鲁棒性，之前的工作提出了各种防御措施[149-151]。其中一种切实有效的策略是构建多个单模型的集成模型（ensemble model），以获得更强的防御系统[56,152]。然而，先前的防御策略中的大多数都专注于逐一增强每个单个模型，忽略了多个单模型成员之间潜在的相互关联。由于对抗样本在单模型之间具有很强的迁移性（transferability）[56,153]，所以在构建集成模型时，忽略多个单模型成员之间的相互联系可能会导致他们返回相似的预测或特征表示[154-155]。由于对抗样本更容易在相似的模型之间迁移，因此为单个模型生成的对抗样本也可能欺骗集成模型中的其他模型成员，甚至欺骗整个集成模型。以前的工作表明，对于单个模型，促进不同类别学到的特征之间多样性可以提高对抗鲁棒性[32-33]。在本章中，我们提出一个新的角度来提高集成模型的对抗鲁棒性，即通过促进集成模型中不同单模型成员返回的预测之间的多样性，称为*集成多样性*（ensemble diversity），如图 3.1中所示。我们的方法与作用于单个模型的其他防御方法可以无缝地兼容。

具体来说，我们首先在对抗环境中定义集成多样性的数学表达，这与之前在非对抗环境中集成弱分类器（weak classifiers）的多样性定义有很大不同。在经典统计学习的框架中，我们考虑集成模型中单个模型的预测误差来定义集成多样

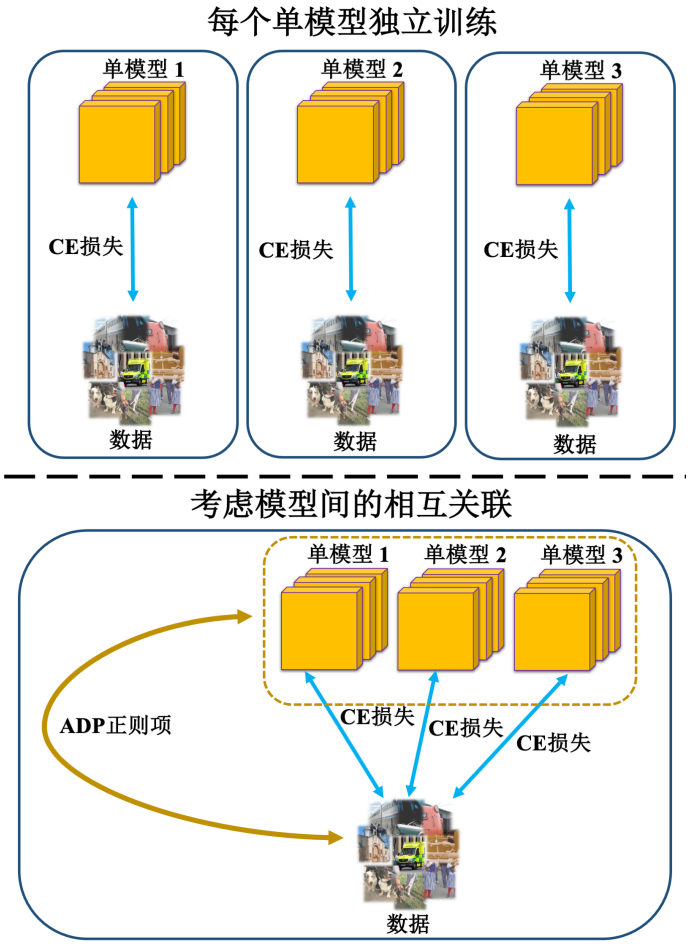


图 3.1　集成模型的训练流程

性[156-157]。然而，这个定义并不适合深度学习的情况，因为大多数深度学习模型属于强分类器，即可以在训练集上达到很高的准确率[1]。强行促进预测误差的多样性将在很大程度上牺牲集成模型的准确率。因此，我们定义了每个单模型成员的非最大预测（non-maximal prediction）之间的多样性。在几何上，我们定义的多样性等于归一化之后的非最大预测向量所展开得到的超立方体体积，如图 3.2 所示。值得注意的是，我们仅促进非最大预测的多样性，这样可以允许每个单模型的最大预测与真实标签一致，不会影响集成之后的准确率。除此之外，由于非最大预测对应于对抗样本返回的所有潜在的错误类别，因此单个模型的非最大预测之间的高度多样性或不一致性会降低对抗样本在它们之间迁移能力，并进一步得到更好的集成鲁棒性。

根据我们对集成多样性的定义，在方法上我们提出了自适应多样性增强（ADP）正则化项来促进多样性。ADP 正则化项由两部分组成：第一部分是集成多样性的对数，第二部分是集成预测的香农熵。在训练过程中，我们使用 ADP 正则化项，并在同一数据集上同时交互式地训练集成模型中的所有单模型。实验中，我们在

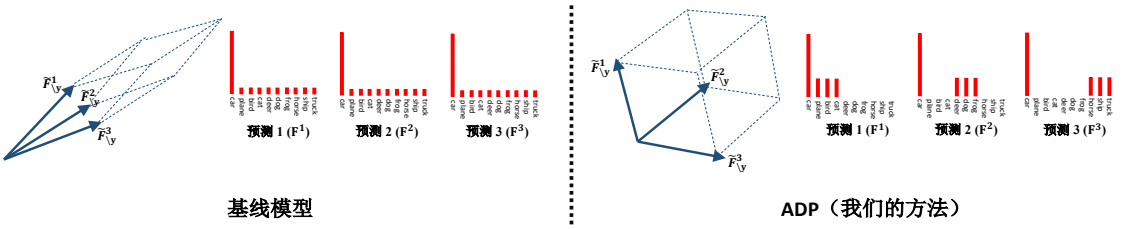


图 3.2　ADP 方法的几何解释

MNIST、CIFAR-10 以及 CIFAR-100 数据集上使用不同的对抗攻击来测试 ADP 方法训练的模型。结果表明，我们的 ADP 方法可以显着提高模型的对抗鲁棒性，并且不降低干净样本上的正常准确率。在使用 ADP 训练的过程中，为了进行反向传播[158]我们需要分别执行行列式 $\det(G)$ 和矩阵求逆 G^{-1} 的操作，计算复杂度为 $\mathcal{O}(K^3)$。幸运的是，其中 K 是集成模型中的单模型成员数，所以 K 的增长通常会比待解决的问题的规模增长慢得多[159]。这使我们的方法能够扩展到大多数分类任务，并且避免了之前工作中遇到的由这些矩阵运算引起的大量计算开销[160-162]。

3.2　算法设计

在本节中，我们首先介绍集成模型的训练策略。然后，我们提出自适应多样性增强（ADP）训练方法，并对 ADP 方法的最优解进行了理论分析。形式上，我们将一个神经网络模型表示为 $F(x,\theta)$，其中 x 为输入，θ 为模型可训练的参数。令 L 代表分类问题中的类别数目，则模型的输出 $F(x,\theta)\in\mathbb{R}^L$。为了符号的简洁，在之后的叙述中我们不显式地写出 F 与 θ 的依赖关系。当我们训练模型的时候，最常用的损失函数就是交叉熵（cross-entropy，缩写为 CE）损失函数，写为

$$\mathcal{L}_{\mathrm{CE}}(x,y)=-1_y^{\top}\log F(x)=-\log F(x)_y, \tag{3.1}$$

其中 y 为输入 x 的真实类别，1_y 是 y 的独热码（one-hot encoding）。注意这里与上一章使用的定义稍有区别，即在本章中我们将 softmax 层归入到 $F(x)$ 中。

3.2.1　集成模型的训练策略

在实际的应用场景中，例如分类任务，集成多个单独的模型通常可以有效地提高系统整体的泛化性能[159]。我们将集成模型中的第 k 个单模型成员表示为 $F^k(x)\in\mathbb{R}^L$。则最常用的集成模型的预测 $\mathcal{F}(x)$ 可以写为所有 K 个单模型预测的平均：

$$\mathcal{F}(x)=\frac{1}{K}\sum_{k\in[K]}F^k(x), \tag{3.2}$$

其中 $[K] := \{1, \cdots, K\}$。注意到这里的平均可以发生在 softmax 层之前（即分对数的平均）或者 softmax 层之后（即预测概率的平均），本文中我们主要考虑后面一种情况。在训练集成模型时，有三种基本的学习范式可以选择，包括独立训练（independent training）、同时训练（simultaneous training）以及顺序训练（sequential training）[163]。下面依次对这三种学习范式进行简单的介绍。

*独立训练范式*顾名思义，即每个单模型 $F^k(x)$ 独立进行训练，没有任何信息交互。每个单模型的训练损失函数为 $\mathcal{L}_{\text{CE}}^k(x, y) := -\log F^k(x)_y$。由于独立训练范式实现简单，并且可以在多个互不通信的计算集群上训练，所以在实际的深度学习系统中被广泛采用[152,159]。然而独立训练范式的缺陷也很明显，就是其忽略了单模型之间的相互作用，导致单模型趋向于返回相似的预测和特征表示[154-155]，从而弱化集成带来的性能提升[164]。

在*同时训练范式*中，集成模型中的所有单模型在训练的每一次更新中使用相同的批量数据（mini-batch of data）。形式上，同时训练范式的训练损失函数可以写为*集成交叉熵损失函数*（ensemble cross-entropy，缩写为 ECE），定义为

$$\mathcal{L}_{\text{ECE}}(x, y) = \sum_{k \in [K]} \mathcal{L}_{\text{CE}}^k(x, y), \tag{3.3}$$

即单模型交叉熵损失函数的平均。不同单模型的训练参数可以存在差异。例如每个单模型的学习率（learning rate）可以不一样，对于收敛快的模型可以设置更小的学习率。在训练后期，我们可以冻结住一部分已经收敛的单模型的参数防止其过拟合，同时继续训练其他未收敛的单模型。同时训练范式的一大优势就是可以在 ECE 损失函数的基础上加入其他正则化项，包含模型间相互作用的信息。尽管同时训练范式相比于独立训练范式并不需要明显的额外计算量，但是同时训练范式需要较大的并发存储资源，例如对 GPU 显存要求较高。当并发存储资源受限时，我们可以考虑采用*顺序训练范式*，依次训练每个单模型来近似完成同时训练范式的任务。在本章中我们的叙述采用同时训练范式。

3.2.2　自适应多样性增强学习

之前的工作表明，对于单模型来讲，提高不同类别样本对应特征的多样性有助于提高对抗鲁棒性[32-33]。在本章中，我们从另一个层面来提升系统的内部多样性，即提升集成模型中不同单模型成员预测的多样性，称作*集成多样性*（ensemble diversity）。值得注意的是，集成多样性在之前的工作中并没有严格的统一定义[165]。在经典的统计学习框架下，集成多样性通常被认为是每个弱分类器的预测误差构成的[156-157]。这是基于弱分类器的预测误差较大，所以鼓励弱分类器的最大预测多样化可以提升集成模型的预测准确率[134]。然而，深度学习框架下的单模型往往

算法 3.1 ADP 训练流程

输入: K 个单模型 $\{F^k(x,\theta^k)\}_{k\in[K]}$；训练集 $\mathcal{D}=\{(x_i,y_i)\}_{i\in[N]}$。
初始化: $k\in[K]$，初始化每个单模型的参数为 θ_0^k，训练步数计数 $c_k=0$，学习率 ε_k，指示集合 $I=[K]$。
while $I\neq\varnothing$ **do**
　计算损失函数在每个批量数据 $\mathcal{D}_m$ 上的值

$$\mathcal{L}_{\text{ADP}}^m=\frac{1}{\left|\mathcal{D}_m\right|}\sum_{(x_i,y_i)\in\mathcal{D}_m}\left[\mathcal{L}_{\text{ECE}}-\text{ADP}_{\alpha,\beta}\right](x_i,y_i).$$

　for $k^{'}$ in I **do**
　　更新 $\theta_{c+1}^{k^{'}}\leftarrow\theta_c^{k^{'}}-\varepsilon_{k'}\nabla_{\theta^{k'}}\mathcal{L}_{\text{ADP}}^m\big|_{\{\theta_{c_k}^k\}_{k\in[K]}}$。
　　更新 $c_{k'}\leftarrow c_{k'}+1$，其中 $c=c_{k'}$。
　　if $\theta^{k^{'}}$ converges **then** 更新 $I=I\backslash\{k^{'}\}$。
　end for
end while
输出: 每个单模型的参数 $\theta^k=\theta_{c_k}^k$，其中 $k\in[K]$。

就已经具有较高的预测准确率，不适于弱分类器的假设，所以不应将集成多样性定义在预测误差的意义下。

为了保证每个神经网络单模型的准确率，集成模型中的每个单模型成员的最大预测类别应该保持一致，并且将集成多样性定义在非最大预测上。对于一个真实类别为 y 的输入样本 x，模型的最大预测类别应该为 y，而非最大预测类别应该对应于所有除了 y 以外的类别。形式上，我们基于行列式点过程（determinant point process，缩写为 DPP[160]）的理论，定义集成多样性为

$$\mathbb{ED}=\det(\tilde{\boldsymbol{M}}_{\backslash y}^{\top}\tilde{\boldsymbol{M}}_{\backslash y}). \tag{3.4}$$

上式中，$\tilde{\boldsymbol{M}}_{\backslash y}=(\tilde{\boldsymbol{F}}_{\backslash y}^1,\cdots,\tilde{\boldsymbol{F}}_{\backslash y}^K)\in\mathbb{R}^{(L-1)\times K}$，其中每一列向量 $\tilde{\boldsymbol{F}}_{\backslash y}^k\in\mathbb{R}^{L-1}$ 通过对 $\boldsymbol{F}_{\backslash y}^k$ 在 ℓ_2-范数下归一化得到。这里 $\boldsymbol{F}_{\backslash y}^k$ 代表第 k 个单模型在除了 y 以外的类别上的输出向量。根据矩阵计算理论[166]，我们有

$$\det(\tilde{\boldsymbol{M}}_{\backslash y}^{\top}\tilde{\boldsymbol{M}}_{\backslash y})=\text{Vol}^2(\{\tilde{\boldsymbol{F}}_{\backslash y}^k\}_{k\in[K]}), \tag{3.5}$$

其中 Vol(·) 表示输入向量集合所张成（spanned）的超立方体体积。如图 3.2中所示，公式（3.4）为我们定义的集成多样性 $\mathbb{ED}$ 提供了直观的几何解释。由于 $\tilde{\boldsymbol{F}}_{\backslash y}^k$ 是归一化的，即 $\|\tilde{\boldsymbol{F}}_{\backslash y}^k\|_2=1$，所以集成多样性 $\mathbb{ED}$ 的最大值为 1，当且仅当 $\tilde{\boldsymbol{M}}_{\backslash y}$ 中的所有列向量相互正交。注意到，在对抗环境中，当模型在干净样本上的最大预测类别为真实类别 y 时，所有非最大预测类别对应于潜在对抗目标类别，即潜在的会令模型错误分类的类别。因此在非最大预测上鼓励多样性可以干扰对抗样本的迁移能力，防止多个单模型被同时欺骗到某一错误类别，从而提高集成模型的整体鲁棒性。

为了提高集成多样性，我们提出了自适应多样性增强（ADP）正则项，其数学形式写为

$$\mathrm{ADP}_{\alpha,\beta}(x, y) = \alpha \cdot \mathcal{H}(\mathcal{F}) + \beta \cdot \log\left(\mathbb{ED}\right), \tag{3.6}$$

其中 $\alpha, \beta \geqslant 0$ 为两个训练超参数。公式（3.6）中的第一部分是集成预测的香农熵（ensemble Shannon entropy），写为

$$\mathcal{H}(\mathcal{F}) = -\sum_{i\in[L]} \mathcal{F}_i \log(\mathcal{F}_i); \tag{3.7}$$

公式（3.6）中的第二部分是集成多样的对数（logarithm of ensemble diversity，缩写为 LED）。在算法 3.1中我们给出了使用 ADP 方法进行训练的基本流程。

3.3　理论分析

在 ADP 训练过程中，优化目标可以写为

$$\begin{aligned} &\min_{\theta} \mathcal{L}_{\mathrm{ECE}} - \mathrm{ADP}_{\alpha,\beta} \\ &\text{s.t. } 0 \leqslant F_j^k \leqslant 1, \sum_{j\in[L]} F_j^k = 1, \end{aligned} \tag{3.8}$$

其中 $\theta = \{\theta^k\}_{k\in[K]}$ 代表集成模型中所有可训练的参数。为了简化分析，我们聚焦于问题（3.8）在预测空间 $\mathbb{F} = \{F^k\}_{k\in[K]}$ 中的最优解，并且假设每个单模型有无限的表达能力（universal approximator[167]）。因此我们可以不考虑从 x 到 F^k 的映射，并且将问题（3.8）重新定义在 $\mathbb{F}$ 意义下，而非在参数 θ 意义下。下面我们从理论上分析 ADP 正则项中的两个超参数 α 和 β 如何影响问题（3.8）的最优解。

首先，若 $\alpha = 0$，则 ADP 正则项中只剩下 LED 项。在这种情况下，问题（3.8）的最优解仍然是独热码 1_y，即与不使用 ADP 正则项的情况无异：

定理 3.1 ($\alpha = 0$ 的情况)： 当 $\alpha = 0$ 时，对于 $\forall\beta \geqslant 0$，问题（3.8）的最优解满足 $F^k = 1_y$，其中 $k \in [K]$。

定理 3.1说明 ADP 正则项中的 LED 项无法单独发挥作用，即正则化集成预测的香农熵在 ADP 中是必要的。为了直观解释这一结论，注意到 LED 项是定义在归一化的非真实类别预测 $\tilde{F}_{\setminus y}^k$ 上。因此，LED 项只能影响非真实类别预测之间的夹角。另一方面，所有（未归一化的）非真实类别预测的和为 $1 - F_y^k$，且这个和只会受到 ECE 项的影响，从而当 $\alpha = 0$ 时，ECE 项完全主导了优化过程，使得最优解仍然是独热码 1_y。

其次，若 $\beta = 0$，则 ADP 正则项中只剩下集成预测的香农熵项。此时问题（3.8）的最优解有如下形式：

定理 3.2 ($\beta = 0$ 的情况)： 当 $\beta = 0$ 时，对于 $\forall \alpha > 0$，问题（3.8）的最优解满足 $F_y^k = \mathcal{F}_y, \mathcal{F}_j = \frac{1-\mathcal{F}_y}{L-1}$，以及

$$\frac{1}{\mathcal{F}_y} = \frac{\alpha}{K} \log \frac{\mathcal{F}_y(L-1)}{1-\mathcal{F}_y}, \tag{3.9}$$

其中 $k \in [K]$ 且 $j \in [L] \backslash \{y\}$。

证明： 当 $\beta = 0$ 且 $\alpha > 0$ 时，问题（3.8）的目标函数变为

$$\min_{\mathbb{F}} \ \mathcal{L}_{\mathrm{ECE}} - \alpha \cdot \mathcal{H}(\mathcal{F}). \tag{3.10}$$

将 $0 \leqslant F_j^k \leqslant 1$ 以及 $\sum_{j \in [L]} F_j^k = 1$ 的约束条件用拉格朗日（Lagrangian）乘数法构造出拉格朗日函数 L

$$L = \mathcal{L}_{\mathrm{ECE}} - \alpha \cdot \mathcal{H}(\mathcal{F}) + \sum_{k \in [K]} \omega_k (1 - \sum_{j \in [L]} F_j^k) + \sum_{k \in [K]} \sum_{j \in [L]} \left[\beta_{k,j} F_j^k + \gamma_{k,j} (1 - F_j^k) \right], \tag{3.11}$$

其中 $\beta_{k,j} \leqslant 0, \gamma_{k,j} \leqslant 0$。拉格朗日函数 L 对于 F_j^k 的偏导数为

$$\begin{aligned} \frac{\partial L}{\partial F_y^k} &= -\frac{1}{F_y^k} + \frac{\alpha}{K} \left[1 + \log \mathcal{F}_y \right] - \omega_k + \beta_{k,y} - \gamma_{k,y}, \\ \frac{\partial L}{\partial F_j^k} &= \frac{\alpha}{K} \left[1 + \log \mathcal{F}_j \right] - \omega_k + \beta_{k,j} - \gamma_{k,j}, \forall j \neq y. \end{aligned} \tag{3.12}$$

根据 KKT（Karush-Kuhn-Tucker）条件，最优解应该满足 $\forall k \in [K], j \in [L]$

$$\begin{aligned} &\frac{\partial L}{\partial F_j^k} = 0, \\ &\beta_{k,j} F_j^k = 0, \\ &\gamma_{k,j} (1 - F_j^k) = 0. \end{aligned} \tag{3.13}$$

考虑在 $(0,1)^{L \times K}$ 范围内的最优解，可以得到所有的 $\beta_{k,j}$ 以及 $\gamma_{k,j}$ 均等于零。因此，我们可以进一步推出

$$\begin{aligned} &-\frac{1}{F_y^k} + \frac{\alpha}{K} \left[1 + \log \mathcal{F}_y \right] = \omega_k, \\ &\frac{\alpha}{K} \left[1 + \log \mathcal{F}_j \right] = \omega_k, \forall j \neq y, \end{aligned} \tag{3.14}$$

且从公式（3.14）中的第二个等式可以得到 $\forall j \neq y$,

$$\begin{aligned} \mathcal{F}_j = \exp(\frac{\omega_k K}{\alpha} - 1) &\Longrightarrow \sum_{j \neq y} \mathcal{F}_j = \sum_{j \neq y} \exp(\frac{\omega_k K}{\alpha} - 1) \\ &\Longrightarrow 1 - \mathcal{F}_y = (L-1) \exp(\frac{\omega_k K}{\alpha} - 1) \\ &\Longrightarrow \omega_k = \frac{\alpha}{K} \left[1 + \log(\frac{1-\mathcal{F}_y}{L-1}) \right], \end{aligned} \tag{3.15}$$

以及 $\forall i, j \neq y$，我们有 $\mathcal{F}_i = \mathcal{F}_j = \frac{1-\mathcal{F}_y}{L-1}$。在此基础上，$\forall k \in [K]$,

$$\frac{1}{F_y^k} = \frac{\alpha}{K} \log \left[\frac{\mathcal{F}_y (L-1)}{1-\mathcal{F}_y} \right]. \tag{3.16}$$

因此 $\forall k, l \in [K]$, $F_y^k = F_y^l = \mathcal{F}_y$ 且

$$\frac{1}{\mathcal{F}_y} = \frac{\alpha}{K} \log \left[\frac{\mathcal{F}_y (L-1)}{1-\mathcal{F}_y} \right]. \tag{3.17}$$

∎

定理 3.2中的结论可以帮助我们选择合适的超参数 α。例如，如果我们想在 ImageNet[38]（$L = 1000$）这样的数据集上训练由五个单模型构成的集成模型（$K = 5$), 并且使得最优解 $\mathcal{F}_y = 0.9$, 那么我们可以通过定理 3.2估算出所对应的 $\alpha \approx 0.61$。定理 3.2还额外保证对于每个单模型，其最优解 $F_y^k = \mathcal{F}_y$，即保证所有单模型在真实类别上具有一致的预测概率。此外，注意到对于任何 $j \in [L] \backslash \{y\}$，ADP 中的集成香农熵项并不直接作用于某一单独的非最大预测 F_j^k，而是作用在所有非最大预测的平均值 $\mathcal{F}_j$ 上。这一机制留给了 F_j^k 足够的优化自由度。最后，当 α, β 均大于 0 时，我们有如下推论:

推论 3.1 (α, β 均大于 0 的情况)： 若集成中单模型数量 K 与类别数目 L 满足 $K \mid (L-1)$，即 K 可以整除 $L-1$，那么 $\forall \alpha, \beta > 0$，问题（3.8）的最优解满足定理 3.2中的结论。此外，令 $S = \{s_1, , \cdots, s_K\}$ 为索引集合 $[L] \backslash \{y\}$ 的任意划分，且满足 $\forall k \in [K]$，$\left|s_k\right| = \frac{L-1}{K}$。那么问题（3.8）的最优解同时满足

$$F_j^k = \begin{cases} \frac{K(1-\mathcal{F}_y)}{L-1}, & j \in s_k, \\ \mathcal{F}_y, & j = y, \\ 0, & \text{其他情况.} \end{cases} \tag{3.18}$$

在推论 3.1中，索引集合 s_k 包含第 k 个单模型在类别 y 以外所有的非零预测类别。注意到索引集合的划分 S 对于不同的输入是自适应的（adaptive），也就是说对于两个类别都是 y 的输入 x_1 和 x_2，他们对应的划分 S_1 和 S_2 可以是不同的。举例来说，对于两张类别为狗的图片 #1 和图片 #2，某一单模型可能对图片 #1 和图片 #2 预测是狗的概率均为 0.7，但是同时对于图片 #1，分别在汽车、青蛙、轮船的三个类别上预测概率为 0.1；对于图片 #2，分别在飞机、卡车、猫的三个类别上预测概率为 0.1。这就体现了 ADP 方法为何称之为“自适应”，这种不同输入给予动态的非最大预测类别可以防止在训练过程中引入额外的归纳偏置（inductive bias）。

表 3.1　测试集上干净样本的预测准确率（%）

数据集	模型	基线方法	$\mathbf{ADP}_{2,0}$	$\mathbf{ADP}_{2,0.5}$
MNIST	Net 1	99.56	99.57	99.64
	Net 2	99.61	99.55	99.53
	Net 3	99.57	99.63	99.53
	Ensemble	99.68	99.68	**99.72**
CIFAR-10	Net 1	91.70	90.99	90.66
	Net 2	91.48	90.85	90.66
	Net 3	91.33	90.86	90.08
	Ensemble	93.22	93.26	**93.44**
CIFAR-100	Net 1	64.75	-	60.96
	Net 2	64.09	-	59.14
	Net 3	63.97	-	61.00
	Ensemble	69.65	-	**70.20**

3.4　实验结果

在实验中，我们选择了三个广泛研究的数据集——MNIST、CIFAR-10 以及 CIFAR-100[126,133]。输入图像的像素值被缩放到区间 $[0, 1]$ 内。我们实验中的干净（clean）样本是指训练和测试集中的所有原始样本。①

我们首先评估模型在干净样本上的预测准确率。在每个数据集上，我们构造由三个单模型成员组成的集成模型（$K = 3$），每个单模型结构为 Resnet-20[168]。注意到对于 MNIST 和 CIFAR-10，我们有 $L = 10$；对于 CIFAR-100，我们有 $L = 100$。因此总能满足推论 3.1中 $K \mid (L-1)$ 的条件。我们的基线模型是用 ECE 损失函数训练，相当于 ADP 方法中设定 $\alpha = \beta = 0$。对于我们的方法，我们考虑两种超参数设定：第一种是 $\mathbf{ADP}_{2,0}$，即 $\alpha = 2$ 且 $\beta = 0$，对应于定理 3.2中的情况；第二种是 $\mathbf{ADP}_{2,0.5}$，即 $\alpha = 2$ 且 $\beta = 0.5$，对应于推论 3.1中的情况。$\mathbf{ADP}_{2,0.5}$ 作为我们的完整方法，相对的 $\mathbf{ADP}_{2,0}$ 作为消融实验来评估 LED 项的作用。

在表 3.1中, 我们展示了基线方法和我们的 ADP 方法在不同数据测试集的干净样本上的预测准确率。在训练中，我们使用 Adam[169] 优化器，初始学习率为 0.001，批量大小为 64。在 MNIST 上，我们训练 40 轮；在 CIFAR-10 以及 CIFAR-100 上，我们训练 180 轮。从表 3.1中的结果我们可以发现，尽管 ADP 方法相比于基线方

① 源代码参见https://github.com/P2333/Adaptive-Diversity-Promoting。

表 3.2　与对抗训练方法结合之后的预测准确率（%）

防御方法	CIFAR-10			
	FGSM	BIM	PGD	MIM
AT_{FGSM}	39.3	19.9	24.2	24.5
AT_{FGSM} + **ADP**$_{2,0.5}$	**56.1**	**25.7**	**26.7**	**30.6**
AT_{PGD}	43.2	27.8	32.8	32.7
AT_{PGD} + **ADP**$_{2,0.5}$	**52.8**	**34.0**	**36.2**	**38.8**

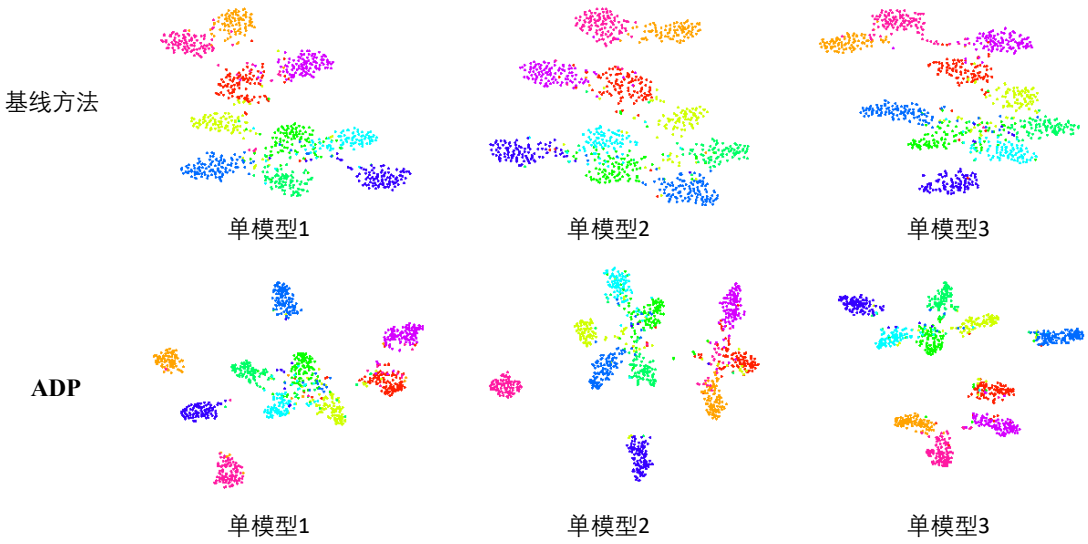


图 3.3　CIFAR-10 测试集上 t-SNE 可视化结果

法会导致更低的单模型准确率，但是经过集成之后，返回的集成预测会得到更高的准确率。这说明我们的 ADP 方法也可以提高模型在正常环境下的预测性能。

为了探究 ADP 训练对于所学特征分布的影响，我们使用 t-SNE[170] 方法将集成模型中每个单模型成员所学特征进行可视化，结果如图 3.3中所展示。我们可以看到，当使用基线方法进行训练时，三个单模型成员趋向于学到相似的特征分布（即不同类别聚类之间关系相似）；而当使用 ADP 方法进行训练时，三个单模型成员所学到的特征分布更加多样化，导致对抗攻击者难以同时欺骗三个单模型。

3.4.1　白盒攻击下的鲁棒性

在对抗环境中，我们首先测试模型在白盒（white-box）攻击下的对抗鲁棒性。我们测试了多种攻击算法，包括 FGSM[16]、BIM[17]、PGD[29]、MIM[27]、JSMA[171]、C&W[146] 以及 EAD[172] 算法。对于每种攻击算法，我们都设定两到三种代表性的

表 3.3　MNIST 及 CIFAR-10 上在白盒攻击下的预测准确率（%）

攻击	MNIST 参数	基线	$\mathbf{ADP}_{2,0}$	$\mathbf{ADP}_{2,0.5}$	CIFAR-10 Para.	Baseline	$\mathbf{ADP}_{2,0}$	$\mathbf{ADP}_{2,0.5}$
FGSM	$\epsilon = 0.1$	78.3	95.5	**96.3**	$\epsilon = 0.02$	36.5	57.4	**61.7**
	$\epsilon = 0.2$	21.5	50.6	**52.8**	$\epsilon = 0.04$	19.4	41.9	**46.2**
BIM	$\epsilon = 0.1$	52.3	86.4	**88.5**	$\epsilon = 0.01$	18.5	44.0	**46.6**
	$\epsilon = 0.15$	12.2	69.5	**73.6**	$\epsilon = 0.02$	6.1	28.2	**31.0**
PGD	$\epsilon = 0.1$	50.7	73.4	**82.8**	$\epsilon = 0.01$	23.4	43.2	**48.4**
	$\epsilon = 0.15$	6.3	36.2	**41.0**	$\epsilon = 0.02$	6.6	26.8	**30.4**
MIM	$\epsilon = 0.1$	58.3	89.7	**92.0**	$\epsilon = 0.01$	23.8	49.6	**52.1**
	$\epsilon = 0.15$	16.1	73.3	**77.5**	$\epsilon = 0.02$	7.4	32.3	**35.9**
JSMA	$\gamma = 0.3$	84.0	88.0	**95.0**	$\gamma = 0.05$	29.5	33.0	**43.5**
	$\gamma = 0.6$	74.0	85.0	**91.0**	$\gamma = 0.1$	27.5	32.0	**37.0**
C&W	$c = 0.1$	91.6	95.9	**97.3**	$c = 0.001$	71.3	76.3	**80.6**
	$c = 1.0$	30.6	75.0	**78.1**	$c = 0.01$	45.2	50.3	**54.9**
	$c = 10.0$	5.9	20.2	**23.8**	$c = 0.1$	18.8	19.2	**25.6**
EAD	$c = 5.0$	29.8	91.3	**93.4**	$c = 1.0$	17.5	64.5	**67.3**
	$c = 10.0$	7.3	87.4	**89.5**	$c = 5.0$	2.4	23.4	**29.6**

参数。对于 BIM、PGD 以及 MIM 攻击，迭代步数为 10，迭代步长为 $\epsilon/10$。对于 C&W 和 EAD 攻击，迭代步数为 1000，迭代步长为 0.01。从表 3.3以及表 3.4中的结果可以看到，我们的 ADP 方法可以显著提高模型的鲁棒性。一个有意思的现象是，尽管在 $\mathbf{ADP}_{2,0}$ 中仅有集成香农熵项，但是 $\mathbf{ADP}_{2,0}$ 仍然可以相比于基线方法提高模型的鲁棒性。这是因为集成香农熵项扩充了模型的最优解空间，给非最大预测 $\boldsymbol{F}_j^k$ 提供了额外的自由度，这样也可以在一定程度上诱导出集成多样性。当然，在 $\mathbf{ADP}_{2,0.5}$ 引入了 LED 项之后，模型的鲁棒性得到了进一步的提升。在表 3.5中，我们还探究了当 $\boldsymbol{K}$ 不能整除 $\boldsymbol{L}$ 的情况（即不满足推论 3.1的条件），从结果上可以看出我们的 ADP 方法在实践上具有很强的通用性。

为了进一步说明我们的 ADP 方法可以和之前的防御方法相结合，我们以对抗训练（adversarial training[29]，简写为 AT）为例，测试 ADP 与对抗训练方法的兼容性。对抗训练方法通过在训练过程中不断构造对抗样本并教会模型正确分类对

表 3.4　CIFAR-100 上在白盒攻击下的预测准确率（%）

攻击	CIFAR-100 参数	基线	$\mathbf{ADP}_{2,0.5}$
BIM	$\epsilon = 0.005$	21.6	**26.1**
	$\epsilon = 0.01$	10.1	**14.8**
PGD	$\epsilon = 0.005$	26.6	**32.1**
	$\epsilon = 0.01$	11.7	**18.3**
MIM	$\epsilon = 0.005$	24.2	**29.4**
	$\epsilon = 0.01$	11.2	**17.1**

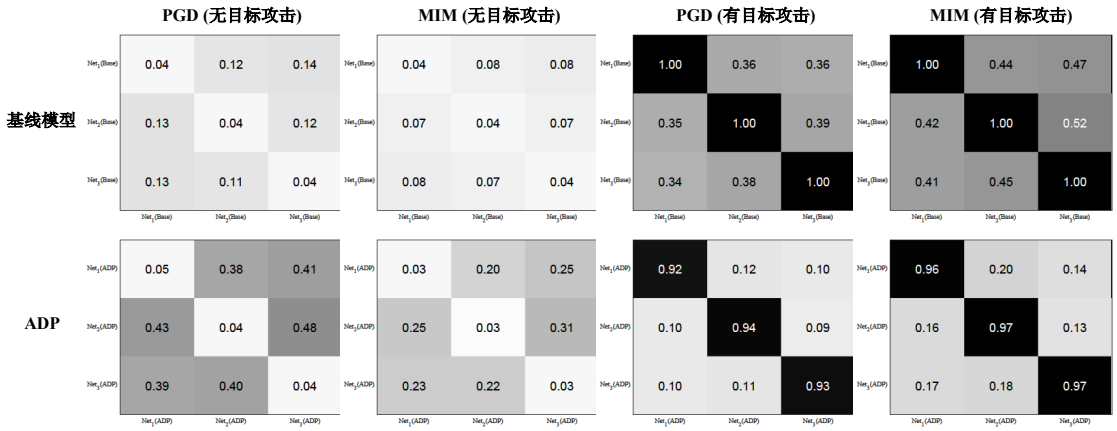


图 3.4　CIFAR-10 上单模型之间对抗样本迁移性

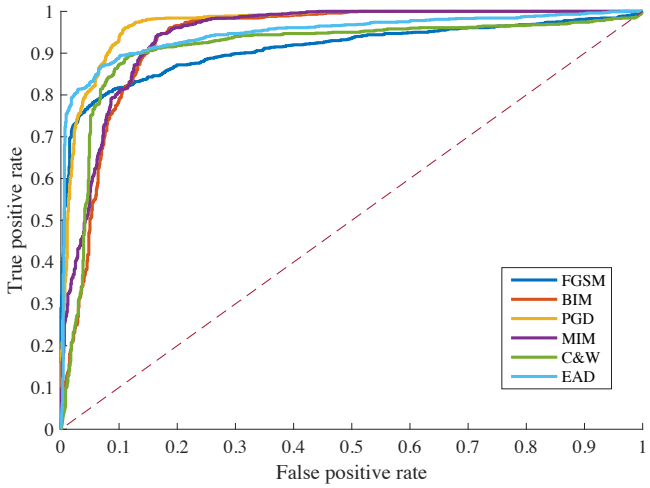
抗样本来提高鲁棒性。我们分别用 FGSM 和 PGD 攻击来构造训练用的对抗样本，标记为 $\mathrm{AT}_{\mathrm{FGSM}}$ 以及 $\mathrm{AT}_{\mathrm{PGD}}$。对抗训练所使用的扰动大小从 $[0.01, 0.05]$ 中均匀采样[149]，并且每个训练批量大小为 128，其中对抗样本与干净样本的比例为 1:1。在表 3.2中，我们展示了 CIFAR-10 上使用对抗训练是否结合 ADP 的不同预测准确率。从结果中我们可以看到，对抗训练结合上 ADP 可以进一步提高模型在各种攻击下的预测准确率。

3.4.2　单模型间的迁移攻击

由于对抗样本可以在不同的模型之间迁移[148]，所以黑盒攻击者可以在替代模型（substitute model）上构造对抗样本，并迁移到目标模型上（target model）。在图 3.4中，我们展示基线方法与 ADP 方法训练出的集成模型中，单模型成员之间的对抗样本迁移性。如图中所示，每个矩阵的 (i, j) 元素代表使用第 i 个单模型作为替

表 3.5　单模型数量 K 不能整除类别数目 L 的情况

攻击	CIFAR-10 参数	基线	$\mathbf{ADP}_{2,0.5}$
Normal	-	93.6	**93.8**
FGSM	$\epsilon = 0.02$	42.0	**58.4**
BIM	$\epsilon = 0.01$	31.6	**41.8**
PGD	$\epsilon = 0.01$	37.4	**44.2**
MIM	$\epsilon = 0.01$	37.1	**47.5**
C&W	$c = 0.01$	52.3	**56.5**
EAD	$c = 1.0$	20.4	**65.3**



攻击	参数	AUC分数 (10^{-2})
FGSM	$\epsilon = 0.1$	**91.19**
BIM	$\epsilon = 0.1$	**93.14**
PGD	$\epsilon = 0.1$	**97.03**
MIM	$\epsilon = 0.1$	**94.09**
C&W	$c = 1.0$	**90.98**
EAD	$c = 20.0$	**94.84**

图 3.5　CIFAR-10 上检测对抗样本的 ROC 曲线及 AUC 分数

代模型，并使用第 j 个单模型作为目标模型的情况下模型在对抗样本上的预测准确率（即迁移性）。我们分别使用 PGD 攻击和 MIM 攻击来构造对抗样本，并将扰动大小设置为 $\epsilon = 0.05$。为了进行完整的探究，我们测试了无目标攻击（untargeted）以及有目标攻击（targeted）的设定。从结果中可以看到，ADP 方法可以有效地抑制单模型间对抗样本的迁移性，为提高集成模型的整体鲁棒性奠定基础。

3.4.3　集成预测用以检测对抗样本

当对抗扰动的最大允许幅度变大时，我们很难有效地正确分类对抗样本[48]。这时我们可以选择检测出对抗样本，并拒绝返回预测类别，防止错误的预测类别造成更大的危害。在图 3.5中，我们展示了 ROC 曲线（receiver operating characteristic curve）以及 AUC 分数（area under curve）。我们使用预测的集成多样性来作为检测

指标，模型使用 $\mathbf{ADP}_{2,0.5}$ 训练。注意到图 3.5中使用的扰动大小远大于表 3.3，所以模型在这些大扰动对抗样本上的预测准确率几乎为零。从结果可以看到，尽管模型无法正确分类对抗样本，但是依旧可以有效地检测出对抗样本，从而在一定程度上提升模型预测的可靠性。

3.5　本章小结

在本章中，我们着手于考虑如何提升集成模型的对抗鲁棒性。之前的工作通常只是单独增强每个单模型成员，然后简单地集成它们的预测。相比之下，我们的 ADP 方法额外建模了单模型之间的相互作用，在保证它们输出一致的最大预测的同时，鼓励它们的非最大预测趋于正交（即非最大预测类别多样化），从而抑制对抗样本在单模型之间的迁移性，提升集成模型的鲁棒性。此外，我们的 ADP 方法还可以和其他防御方法例如对抗训练相结合，进一步训练出鲁棒的模型。

第 4 章　基于反交叉熵训练的对抗样本检测

正确分类对抗样本从而抵御对抗攻击，这对于深度学习模型的实际应用至关重要。尽管有大量的工作提出了各种防御方法来提升模型的鲁棒性，进而希望可以正确分类对抗样本，但是，在实际中，目前效果最好的防御方法仍然无法实现足够可靠的对抗鲁棒性。按照对抗领域内权威的基准评测榜单 RobustBench[36] ①，在 CIFAR-10（ℓ_∞-范数下 8/255 扰动限制）上，鲁棒性最好的模型在对抗样本上的预测准确率都没有超过 67%。注意到，这一结果还是在使用了大量额外训练数据（8 千万额外图片[173]，相比之下 CIFAR-10 原本仅有 5 万张图片）以及大模型（WRN-70-16[174]）的情况下。因此，在对抗样本预测准确率的指标上，现有的模型还远远不足以达到实际中可靠的水平。除了正确分类对抗样本，另一种退而求其次的策略是检测出对抗样本，并允许模型拒绝在检测出的对抗样本上返回预测，减小潜在的风险。例如在自动驾驶场景中，系统拒绝预测并提示驾驶者主动介入的代价，会远低于系统错误识别行人或者路障带来的风险。因此，在本章中，我们提出了一种新的模型训练方式，来帮助检测算法更好地区分出对抗样本。具体来说，我们设计了反交叉熵（reverse cross-entropy，缩写为 RCE）训练方法，可以将干净样本特征映射到低维流形上，从而与对抗样本的特征更明显地区分开来。反交叉熵训练相比于传统的交叉熵训练不引入额外的计算量，且极易实现。实验上，我们在 MNIST 以及 CIFAR-10 数据集上测试了我们的方法，并展示了相比于基线模型显著更优的检测效率。

4.1　本章引言

由于深度学习模型的对抗鲁棒性缺陷，大量的工作尝试提高模型的鲁棒性，使得模型可以正确分类对抗样本。然而，更强的攻击方法例如自适应攻击[48]构造的对抗样本仍然可以成功欺骗之前的很多防御方法，诱导模型返回错误预测。另一方面，一些从模型可验证性（model verification）出发的防御方法被提出，可以保证模型在某一扰动大小下一定不会存在对抗样本。然而，这类可验证性方法需要进行搜索计算，计算开销大，很难扩展到大数据集或者大模型上。并且可验证性方法得到的安全扰动范围（certified bound）是点对点的（point-wise），无法得到一致的（uniform）安全扰动范围。

由于正确分类对抗样本极具挑战性，导致基于检测（detection）对抗样本的防

① 网页链接https://robustbench.github.io/

御策略在近几年受到关注。文献[86]在分类器中引入一个额外的类，用于检测对抗样本；类似地，文献[175]训练一个额外的二元分类器来确定一个输入是否是对抗性的。文献[89]训练检测对抗样本的神经网络，其输入为原始分类网络的中间层表征。文献[176]减少原始分类网络的输入维数，并在压缩后的输入上训练一个全连接的神经网络。文献[177]构建一个级联分类器，其中每个分类器都用线性 SVM 实现，且输入为原始分类网络内部卷积层的 PCA 表征。然而，上述的这些方法都需要大量的额外计算开销，其中一些方法还会导致模型损失在干净样本上的预测准确率。相比之下，文献[93]提出了一种核密度（kernel density，缩写为 KD）估计的方法来检测最后一层特征空间中远离数据流形的样本点，同时该方法不会改变原始分类网络的结构，计算开销也很低。尽管如此，文献[47]表明，这些防御方法中的每一种都可以被针对特定防御的攻击（即自适应攻击）所攻破。

在本章中，我们提出一种新的模型训练方法，使得模型学到的特征表示可以更好地与之前提出的各种检测指标（例如核密度估计方法[93]）相配合。具体来说，我们提出了反交叉熵训练（RCE）准则，来替代传统的交叉熵训练准则（CE）。通过最小化反交叉熵损失函数，模型被鼓励在真实类别上返回低的概率值，同时在其他类别上返回均匀分布的概率值。此外，反交叉熵训练准则还可以促使模型将干净样本的特征分布在低维流形上，这样当有对抗样本输入时，其对应的特征大概率会偏离低维流形，从而更容易被检测指标所发现。反交叉熵训练相比于交叉熵训练不引入额外的计算量，且非常容易实现（一行代码即可实现）。

实验中，我们在 MNIST[133]以及 CIFAR-10[126]数据集上测试了检测多种攻击构造的对抗样本。我们考虑不同的攻击设定[47]，包括灰盒攻击（oblivious adversaries）、白盒攻击（white-box adversaries）以及黑盒攻击（black-box adversaries）。我们选用核密度估计方法作为检测指标，并且展示当模型经过反交叉熵训练后，核密度估计方法可以更加有效地检测出各类对抗样本，提升模型预测的鲁棒性。此外，我们还发现如果想要使用自适应攻击来攻破我们的检测方法，则攻击者需要添加幅度很大的扰动，从而可以被人类观察者所察觉。

4.2　算法设计

我们首先明确一下本章所使用的符号定义，以免有歧义（大部分符号定义与上一章相同）。我们将深度神经网络模型表示为 $F(X,\theta):\mathbb{R}^d\to\mathbb{R}^L$，其中 $X\in\mathbb{R}^d$ 是输入变量，θ 表示模型参数，L 是分类问题中类别数目。下面为了符号简洁，在不引起歧义的情况下我们省略对 θ 的依赖关系。本章中我们聚焦于有 softmax 输出层的神经网络。softmax 函数定义为 $\mathbb{S}(z):\mathbb{R}^L\to\mathbb{R}^L$，其中 $\mathbb{S}(z)_i=\frac{\exp(z_i)}{\sum_{i=1}^{L}\exp(z_i)}, i\in[L]$,

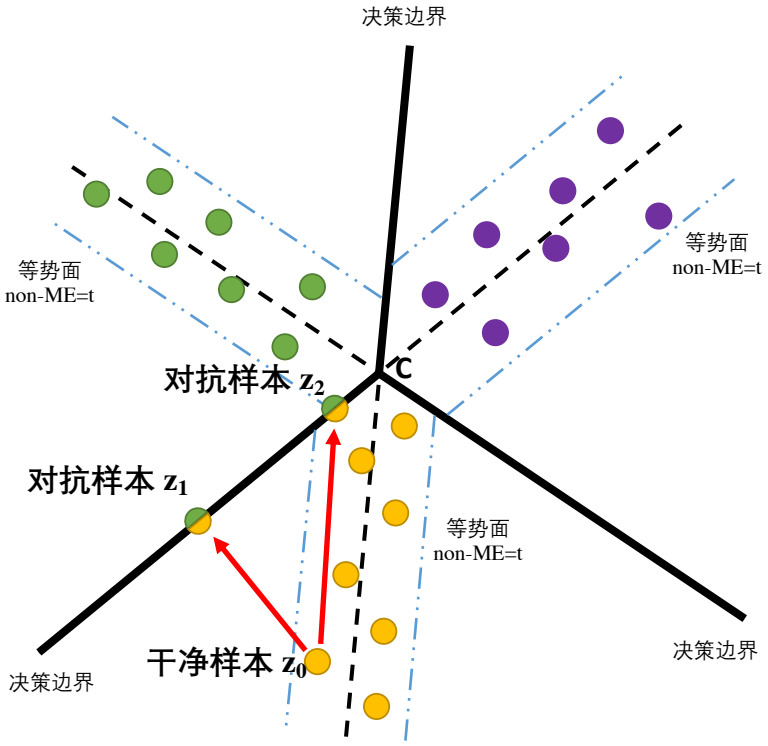


图 4.1　非最大熵机理示意图

以及 $[L] := \{1, \cdots, L\}$。我们用 Z 表示模型的特征表示，则模型可以写为 $F(X) = \mathbb{S}(W_s Z + b_s)$，其中 W_s 和 b_s 分别是 softmax 层的权重矩阵和偏置。此处，分对数（logits）可以写为 $Z_{pre} = W_s Z + b_s$。给定一个输入变量 X 的实例化 x，模型在其上的预测类别为 $\hat{y} = \text{argmax}_{i \in [L]} F(x)_i$。在深度学习的术语中，$F(x)_{\hat{y}}$ 常被称作置信度（confidence）[1]。训练中常用的损失函数为交叉熵（CE）损失函数，定义为

$$\mathcal{L}_{CE}(x, y) = -1_y^{\top} \log F(x) = -\log F(x)_y, \tag{4.1}$$

其中 y 为 x 的真实类别，1_y 为独热码。交叉熵训练准则即为最小化模型在训练数据上的交叉熵损失函数值。

4.2.1　非最大熵

对抗样本检测方法需要依赖于检测指标来决定一个输入对于分类器模型 $F(X)$ 是否是对抗样本。一个常用的检测指标就是预测的置信度 $F(x)_{\hat{y}}$，其可以在一定程度上代表预测的确定性（certainty）[1]。然而，之前的工作表明模型预测的置信度可以被对抗攻击者所轻易地欺骗[16,178]。

基于此，我们首先构造一个更鲁棒的检测指标。具体来说，我们定义非最大熵（non-maximal entropy，缩写为 non-ME）。非最大熵是 $F(x)$ 中所有非最大预测经过

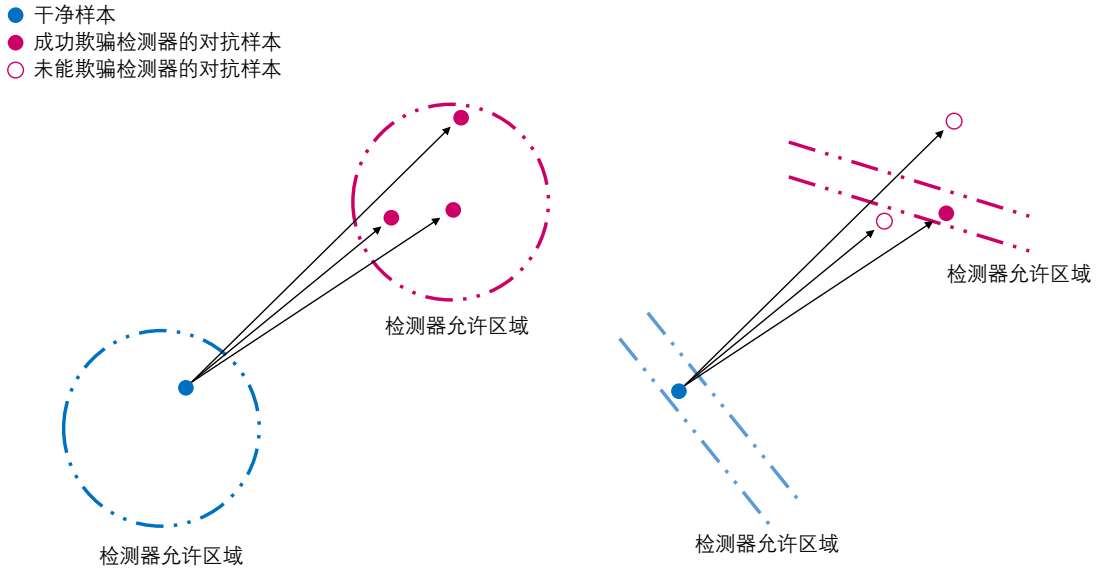


图 4.2　攻击基于 CE 或者 RCE 训练的检测器的不同机制示意图

归一化之后的香农熵，数学上定义为

$$\text{non-ME}(x) = -\sum_{i \neq \hat{y}} \hat{F}(x)_i \log(\hat{F}(x)_i), \tag{4.2}$$

其中 $\hat{F}(x)_i = \frac{F(x)_i}{\sum_{j \neq \hat{y}} F(x)_j}$ 为 $F(x)$ 中经过归一化之后的非最大预测。在下面的分析中，我们用 $F(z)$ 表示与 $F(x)$ 相同的意思，其中 z 为输入 x 所对应的特征。为了直观地解释非最大熵的意义，我们在图 4.1中展示了一个二维特征空间中的三类别分类器示例，即 $z \in \mathbb{R}^2$ 以及 $L = 3$。图中黑色实线为分类器的决策边界，黑色虚线为决策边界的反向延长线，每种颜色代表一个类别。注意到蓝色的虚线为非最大熵等于某一常数 t（即 $\text{non-ME}(x) = t$）时的等势面。令 $Z_{pre,i}, i \in [L]$ 表示分对数 Z_{pre} 的第 i 个分量，则分类器的任意两类 i 与 j 之间的决策边界数学上可以写为 $db_{ij} = \{z : Z_{pre,i} = Z_{pre,j}\}$，且令 $DB_{ij} = \{Z_{pre,i} = Z_{pre,j} + C, C \in \mathbb{R}\}$ 表示所有与 db_{ij} 平行的超平面族。在图 4.1中，db_{ij} 表示一条黑色的直线（包括实线与虚线部分）。在此基础上，我们将半空间（half space）$Z_{pre,i} \geqslant Z_{pre,j}$ 表示为 db^+_{ij}。则我们可以形式上写出类别 $\hat{y}$ 的决策域（decision region）为 $dd_{\hat{y}} = \bigcap_{i \neq \hat{y}} db^+_{\hat{y}i}$，以及其对应的决策边界 $\overline{dd_{\hat{y}}}$。注意到，预测向量 $F(z)$ 在低维流形 $S_{\hat{y}} = (\bigcap_{i,j \neq \hat{y}} db_{ij}) \bigcap dd_{\hat{y}}$ 上的任意点都具有 $L - 1$ 个相等的非最大元素。由此我们得出以下引理[33]：

引理 4.1： 在类别 $\hat{y}$ 的决策域 $dd_{\hat{y}}$ 中，对于 $\forall i, j \neq \hat{y}, \widetilde{db_{ij}} \in DB_{ij}$，非最大熵 non-ME 在低维流形 $\bigcap_{i,j \neq \hat{y}} \widetilde{db_{ij}}$ 上的值是一个常数。特别地，non-ME 取得其全局最大值 $\log(L-1)$ 当且仅当特征处于 $S_{\hat{y}}$ 上。

引理 4.1告诉我们在类别 $\hat{y}$ 的决策域中，如果我们沿着低维流形 $\bigcap_{i,j\neq\hat{y}}\widetilde{db_{ij}}$ 移动一个干净样本特征，则其对应的非最大熵的值不变，反之亦然。这个结论可以进一步推出如下定理：

定理 4.1： 在类别 $\hat{y}$ 的决策域 $dd_{\hat{y}}$ 中，对于 $\forall i,j\neq\hat{y}, z_0\in dd_{\hat{y}}$，存在唯一的 $\widetilde{db_{ij}^0}\in DB_{ij}$，使得 $z_0\in Q_0$，其中 $Q_0=\bigcap_{i,j\neq\hat{y}}\widetilde{db_{ij}^0}$。令 $Q_0^{\hat{y}}=Q_0\bigcap\overline{dd_{\hat{y}}}$，则问题

$$\operatorname{argmin}_{z_0}(\max_{z\in Q_0^{\hat{y}}}F(z)_{\hat{y}}) \tag{4.3}$$

的解集为 $S_{\hat{y}}$。此外，对于 $\forall z_0\in S_{\hat{y}}$，有 $Q_0=S_{\hat{y}}$ 成立；以及对于 $\forall z\in S_{\hat{y}}\bigcap\overline{dd_{\hat{y}}}$，有 $F(z)_{\hat{y}}=\frac{1}{L}$ 成立。

证明： 给定一个点和一个法向量，我们可以唯一地确定一个超平面。因此，$\forall i,j\neq\hat{y}, z_0\in dd_{\hat{y}}$，存在唯一的 $\widetilde{db_{ij}^0}\in DB_{ij}$，使得 $z_0\in\bigcap_{i,j\neq\hat{y}}\widetilde{db_{ij}^0}=Q_0$ 成立。根据引理 4.1，我们知道 $\forall i,j\neq\hat{y}, z^*\in Q_0^{\hat{y}}$，有 $Z_{pre,i}-Z_{pre,j}=C_{ij}$ 成立，以及存在 $k\neq\hat{y}$，使得 $Z_{pre,\hat{y}}=Z_{pre,k}$。基于此，我们可以推出

$$\begin{aligned}
F(z^*)_{\hat{y}}&=\frac{\exp(Z_{pre,\hat{y}})}{\sum_i\exp(Z_{pre,i})}\\
&=\frac{1}{1+\sum_{i\neq\hat{y}}\exp(Z_{pre,i}-Z_{pre,\hat{y}})}\\
&=\frac{1}{1+\exp(Z_{pre,k}-Z_{pre,\hat{y}})(1+\sum_{i\neq\hat{y},k}\exp(Z_{pre,i}-Z_{pre,k}))}\\
&=\frac{1}{2+\sum_{i\neq\hat{y},k}\exp(C_{ik})}.
\end{aligned} \tag{4.4}$$

令 $M=\{i: C_{ij}\geqslant 0,\forall j\neq\hat{y}\}$，则必有 $k\in M$ 使得集合 M 非空。则我们可以得到

$$\begin{aligned}
\max_{z^*\in Q_0^{\hat{y}}}F(z^*)_{\hat{y}}&=\max_{l\in M}\frac{1}{2+\sum_{i\neq\hat{y},l}\exp(C_{il})}\\
&=\frac{1}{2+\min_{l\in M}\sum_{i\neq\hat{y},l}\exp(C_{il})}\\
&=\frac{1}{2+\sum_{i\neq\hat{y},\tilde{k}}\exp(C_{i\tilde{k}})},
\end{aligned} \tag{4.5}$$

其中 $\tilde{k}$ 为集合 M 中的任意元素。公式（4.5）成立由于 $\forall k_1,k_2\in M$，且 $C_{k_1k_2}\geqslant 0$，$C_{k_2k_1}\geqslant 0$ 以及 $C_{k_1k_2}=-C_{k_2k_1}$，从而导出 $C_{k_1k_2}=C_{k_2k_1}=0$。因此，对于 $\forall l\in M$，$\sum_{i\neq\hat{y},l}\exp(C_{il})$ 拥有相同的值。从公式（4.5）我们可以进一步导出

$$\begin{aligned}
\operatorname{argmin}_{z_0}(\max_{z^*\in Q_0^{\hat{y}}}F(z^*)_{\hat{y}})&=\operatorname{argmin}_{z_0}\frac{1}{2+\sum_{i\neq\hat{y},\tilde{k}}\exp(C_{i\tilde{k}})}\\
&=\operatorname{argmax}_{z_0}\sum_{i\neq\hat{y},\tilde{k}}\exp(C_{i\tilde{k}}).
\end{aligned} \tag{4.6}$$

根据引理 4.1中的结论，我们知道当 $C_{i\tilde{k}} = 0, \forall i \neq \hat{y}, \tilde{k}$ 时，$\sum_{i\neq\hat{y},\tilde{k}} \exp(C_{i\tilde{k}})$ 取到其最大值。因此问题（4.3）的解集合为 $S_{\hat{y}}$。此外，我们还可以得出 $\forall z^* \in S_{\hat{y}} \bigcap \overline{dd_{\hat{y}}}$，有 $F(z^*)_{\hat{y}} = \frac{1}{2+L-2} = \frac{1}{L}$ 成立。 ∎

令 z_0 表示某一干净样本的特征，且其预测类别为 $\hat{y}$。当攻击者基于 z_0 构造对抗样本时，攻击者需要将 z_0 扰动出决策域 $\overline{dd_{\hat{y}}}$，从而改变模型预测。定理 4.1说明在类别 $\hat{y}$ 的决策域中存在唯一的低维流形 Q_0，如果我们可以限制攻击者在扰动样本时不允许改变非最大熵的值，则样本必然被限制在 Q_0 上。在这种情况下，距离干净样本 z_0 最近的对抗样本 z^* 必然处在集合 $Q_0^{\hat{y}}$ 中[136]。此时，最大值 $\max_{z\in Q_0^{\hat{y}}} F(z)_{\hat{y}}$ 是预测置信度 $F(z^*)_{\hat{y}}$ 的上界。定理 4.1进一步告诉我们，如果 $z_0 \in S_{\hat{y}}$，则上述的上界将取到其最小值 $\frac{1}{L}$，从而可得 $F(z^*)_{\hat{y}} = \frac{1}{L}$（置信度的下界也是 $\frac{1}{L}$）。这种情况下我们可以很容易地检测出对抗样本 z^*。

实际中，我们可以通过基于非最大熵的检测器来限制攻击者。在图 4.1中，任何集合 $S_{\hat{y}}$（黑色虚线）上的点都对应最大的非最大熵值 non-ME = $\log 2$。假设模型学到的特征映射 $X \mapsto Z$ 可以将所有的干净样本特征聚集到 $S_{\hat{y}}$ 的邻域，其中邻域的边界为非最大熵的等势面 non-ME = t（蓝色虚线）。这表明所有的干净样本都对应于 non-ME $\geqslant t$。当我们不使用检测器时，距离干净样本 z_0 最近的对抗样本为 z_1。相比之下，当我们使用基于非最大熵的检测器时，z_1 将很容易被检测器发现为对抗样本，而在此情况下能同时欺骗模型和检测器的最近对抗样本为 z_2。在更一般的情况下，我们可以推出 $\|z_0 - z_2\| > \|z_0 - z_1\|$ 几乎处处成立。这表明由于使用了对抗样本检测器，攻击者需要更大的扰动来避免对抗样本被检测出来。并且根据定理 4.1，对抗样本 z_2 拥有较低的预测置信度，当我们使用多个检测指标（例如非最大熵以及置信度的结合）时，仍然可以检测出 z_2。

4.2.2　反交叉熵训练

基于上一小节的分析，我们开始着手于设计新的训练损失函数来提升模型的鲁棒性。我们的设计思路主要遵循着希望将所有干净样本特征映射到低维流形 $S_{\hat{y}}$ 的邻域内。根据引理 4.1，这一目标可以通过鼓励模型预测 $F(x)$ 的非最大元素尽量相等来实现，从而使得干净样本上的非最大熵尽量取得最大值。具体来说，我们令 R_y 来表示反独热码（reverse one-hot encoding），其第 y 个元素为零，其他位置的元素均为 $\frac{1}{L-1}$。一种直观的方式来鼓励非最大预测间的相等就是去引入类别平滑（label smoothing[179]）操作。数学上类别平滑操作的损失函数可以写为

$$\mathcal{L}_{CE}^{\lambda}(x, y) = \mathcal{L}_{CE}(x, y) - \lambda \cdot R_y^{\top} \log F(x), \tag{4.7}$$

其中 λ 为一个超参数。然而，我们很容易推出最小化 $\mathcal{L}_{CE}^{\lambda}$ 等价于最小化模型预测 $F(x)$ 与向量 P^{λ} 之间的交叉熵，其中 P^{λ} 定义为：

$$P_i^{\lambda} = \begin{cases} \frac{1}{\lambda+1}, & i = y, \\ \frac{\lambda}{(L-1)(\lambda+1)}, & i \neq y. \end{cases} \tag{4.8}$$

注意到上式中 $1_y = P^0$ 以及 $R_y = P^{\infty}$。当 $\lambda > 0$ 时，令 $\theta_{\lambda}^* = \operatorname{argmin}_{\theta} \mathcal{L}_{CE}^{\lambda}$，则模型预测 $F(x, \theta_{\lambda}^*)$ 会趋向于向量 P^{λ}，而非趋向于独热码 1_y。这使得模型预测的最优解是有偏的（biased）。为了得到无偏的最优解（即 $F(x)$ 趋向于 1_y），同时又可以鼓励非最大预测相等，我们提出了*反交叉熵*（RCE）训练损失函数，定义为

$$\mathcal{L}_{CE}^{R}(x, y) = -R_y^{\top} \log F(x). \tag{4.9}$$

最小化反交叉熵损失函数等价于最小化 $\mathcal{L}_{CE}^{\infty}$。注意到当我们直接最小化 $\mathcal{L}_{CE}^{R}$ 时，最优参数 $\theta_R^* = \operatorname{argmin}_{\theta} \mathcal{L}_{CE}^{R}$，此时我们将得到一个*反模型*（reverse model）$F(X, \theta_R^*)$。该反模型趋向于给真实类别最低的预测概率，同时给其他类别相等的预测概率。这一现象引出了我们反交叉熵训练准则的整体流程，包括两个主要部分：

- **反训练：**给定训练集 $\mathcal{D} := \{(x^i, y^i)\}_{i \in [N]}$，最小化反交叉熵损失函数得到

$$\theta_R^* = \operatorname*{argmin}_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{CE}^{R}(x^i, y^i); \tag{4.10}$$

- **分对数取反：**对 softmax 层的输入分对数取反，即

$$F_R(X, \theta_R^*) = \operatorname{softmax}(-Z_{pre}(X, \theta_R^*)). \tag{4.11}$$

通过反训练以及分对数取反这两个流程，我们得到最终的模型 $F_R(X, \theta_R^*)$。下述定理表明了模型 $F_R(X, \theta_R^*)$ 的性质：

定理 4.2： 令 x 为输入，y 为其真实类别。在 ℓ_{∞}-范数意义下，我们定义训练误差 $\alpha \ll \frac{1}{L}$，即 $\left\|\operatorname{softmax}(Z_{pre}(x, \theta_R^*)) - R_y\right\|_{\infty} \leqslant \alpha$，$\alpha \ll \frac{1}{L}$，则我们可以得到

$$\left\|\operatorname{softmax}(-Z_{pre}(x, \theta_R^*)) - 1_y\right\|_{\infty} \leqslant \alpha(L-1)^2, \tag{4.12}$$

且对于 $\forall j, k \neq y$，我们有

$$\left|\operatorname{softmax}(-Z_{pre}(x, \theta_R^*))_j - \operatorname{softmax}(-Z_{pre}(x, \theta_R^*))_k\right| \leqslant 2\alpha^2(L-1)^2. \tag{4.13}$$

证明： 为了简单起见，我们符号上省略 Z_{pre} 对于输入 x 以及参数 θ_R^* 的依赖关系。令 $G = (g_1, g_2, ..., g_L)$，其中 $g_i = \exp(Z_{pre,i})$。根据条件 $\left\|\operatorname{softmax}(Z_{pre}) - R_y\right\|_{\infty} \leqslant \alpha$，

表 4.1　在 MNIST 以及 CIFAR-10 数据集上的测试错误率（%）

模型结构（方法）	MNIST	CIFAR-10
Resnet-32 (CE)	0.38	7.13
Resnet-32 (RCE)	**0.29**	**7.02**
Resnet-56 (CE)	0.36	**6.49**
Resnet-56 (RCE)	**0.32**	6.60

我们得出

$$\begin{cases} \frac{g_y}{\sum_i g_i} \leqslant \alpha \\ \left| \frac{g_j}{\sum_i g_i} - \frac{1}{L-1} \right| \leqslant \alpha, \quad j \neq y. \end{cases} \tag{4.14}$$

令 $C = \sum_i g_i$，我们可以进一步写出

$$\begin{cases} g_y \leqslant \alpha C \\ (\frac{1}{L-1} - \alpha)C \leqslant g_j \leqslant (\frac{1}{L-1} + \alpha)C, \quad j \neq y. \end{cases} \tag{4.15}$$

基于此，对于 $L \geqslant 2$ 的情形（囊括所有非平凡的情况），我们可以推出

$$\begin{aligned} \text{softmax}(-Z_{pre})_y &= \frac{\frac{1}{g_y}}{\frac{1}{g_y} + \sum_{i \neq y} \frac{1}{g_i}} \\ &\geqslant \frac{1}{1 + \sum_{i \neq y} \frac{\alpha C}{(\frac{1}{L-1} - \alpha)C}} \\ &= 1 - \frac{\alpha(L-1)^2}{1 - \alpha(L-1) + \alpha(L-1)^2} \\ &\geqslant 1 - \alpha(L-1)^2 \end{aligned} \tag{4.16}$$

以及 $\forall j \neq y$，我们有

$$\begin{aligned} \text{softmax}(-Z_{pre})_j &= \frac{\frac{1}{g_j}}{\frac{1}{g_y} + \sum_{i \neq y} \frac{1}{g_i}} \\ &\leqslant \frac{1}{1 + \frac{g_j}{g_y}} \\ &\leqslant \frac{1}{1 + \frac{(\frac{1}{L-1} - \alpha)C}{\alpha C}} \\ &= \alpha(L-1) \\ &\leqslant \alpha(L-1)^2. \end{aligned} \tag{4.17}$$

至此，我们证明了 $\left\|\text{softmax}(-Z_{pre}) - 1_y\right\|_\infty \leqslant \alpha(L-1)^2$。此外，我们还可以得出 $\forall j, k \neq y$，

$$\begin{aligned}
\left|\text{softmax}(-Z_{pre})_j - \text{softmax}(-Z_{pre})_k\right| &= \frac{\left|\frac{1}{g_j} - \frac{1}{g_k}\right|}{\frac{1}{g_y} + \sum_{i\neq y}\frac{1}{g_i}} \\
&\leqslant \frac{\frac{1}{(\frac{1}{L-1}-\alpha)C} - \frac{1}{(\frac{1}{L-1}+\alpha)C}}{\frac{1}{\alpha C} + \sum_{i\neq y}\frac{1}{(\frac{1}{L-1}+\alpha)C}} \\
&= \frac{\frac{L-1}{1-\alpha(L-1)} - \frac{L-1}{1+\alpha(L-1)}}{\frac{1}{\alpha} + \frac{(L-1)^2}{1+\alpha(L-1)}} \\
&= \frac{2\alpha^2(L-1)^2}{1+\alpha(L-1)^2(1-\alpha L)} \\
&\leqslant 2\alpha^2(L-1)^2.
\end{aligned} \tag{4.18}$$

∎

定理 4.2证明了反交叉熵训练的两个重要的性质。第一，当训练误差趋于零（$\alpha \to 0$）时，模型的预测 $F_R(x, \theta_R^*)$ 趋于独热码 1_y，即模型预测是一致无偏的；第二，反交叉熵训练中任意两个非最大预测之间的差值随着训练误差以 $\mathcal{O}(\alpha^2)$ 量级缩小，相比之下，交叉熵训练中相应的下降速度为 $\mathcal{O}(\alpha)$ 量级。这两个性质使得反交叉熵训练满足我们的设计目标。

4.3　实验结果

实验中，我们在 MNIST[133] 以及 CIFAR-10[126] 数据集上验证我们的方法。我们将输入的像素值归一化到 $[-0.5, 0.5]$ 的范围。我们使用核密度估计[93] 作为检测指标，具体来说给定预测类别 $\hat{y}$，核密度定义为 $KD(x) = \frac{1}{|X_{\hat{y}}|}\sum_{x_i \in X_{\hat{y}}} k(z_i, z)$，其中 $X_{\hat{y}}$ 表示训练样本中真实类别为 $\hat{y}$ 的子集，$k(z_i, z) = \exp(-\left\|z_i - z\right\|^2/\sigma^2)$ 为高斯核密度函数，σ 为超参数。①

我们首先在正常环境中测试模型在干净样本上的预测错误率。我们使用两种模型结构，包括 Resnet-32 以及 Resnet-56[180]。在 MNIST 我们训练 20000 步，在 CIFAR-10 上我们训练 90000 步。在表 4.1中我们报告了使用交叉熵训练（CE）或者使用反交叉熵训练（RCE）的情况下模型的测试错误率。在表 4.1中我们未使用任何检测机制，仅仅是为了检验模型的预测能力。从结果中可以看到，使用反交叉熵训练的模型预测能力和使用交叉熵训练的模型基本是可比的，甚至在很多情况

① 源代码参见https://github.com/P2333/Reverse-Cross-Entropy。

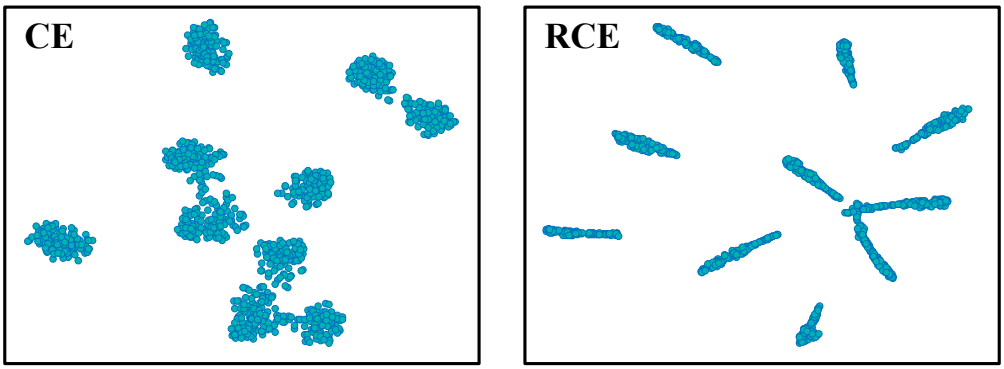


图 4.3　基于 CE 或者 RCE 训练学到的模型特征 t-SNE 可视化

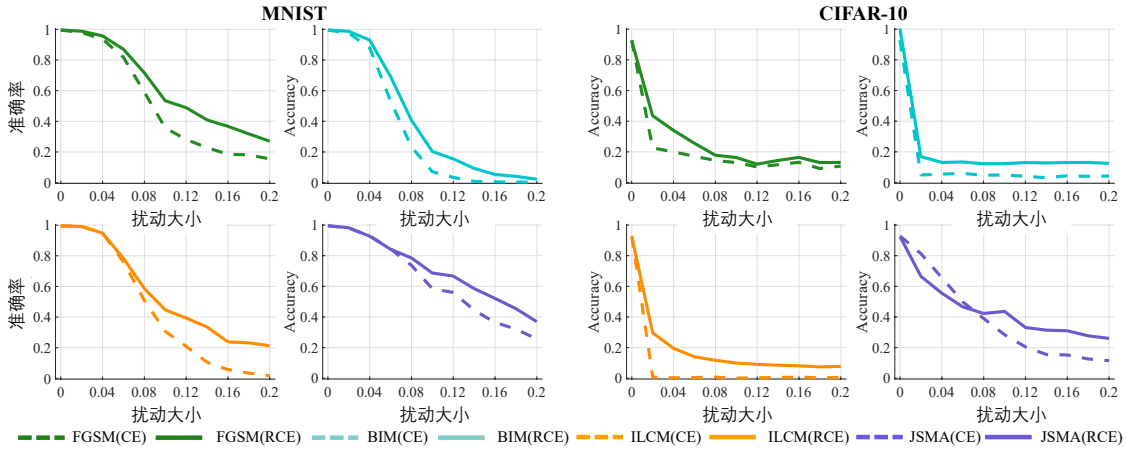


图 4.4　迭代攻击下模型预测准确率随扰动大小的变化

下可以得到更低的错误率。值得注意的是，反交叉熵训练相比于传统的交叉熵训练不需要额外的计算量，也不需要进行参数调整，且极易实现（仅需要一行代码）。为了验证反交叉熵训练过程可以将干净样本特征映射到低维流形 $S_{\hat{y}}$ 附近，我们在图 4.3中提供了模型特征的 t-SNE[181]可视化，其中左图为交叉熵训练（CE）的结果，右图为反交叉熵训练（RCE）的结果。每个图中包含 1000 个 CIFAR-10 上的干净测试样本。我们可以明显地看到反交叉熵训练可以有效地将干净样本特征映射到低维流形附近。

4.3.1　灰盒攻击下的鲁棒性

在对抗环境中，我们首先测试模型在灰盒攻击（oblivious attack）下的鲁棒性。灰盒攻击设定假设攻击者知道预测模型且可以得到其参数，但是攻击者不知道检测器的存在。我们使用 Resnet-32 模型结构，并先测试了在不使用检测器的情况下，模型在对抗攻击下的预测能力。我们考虑四种基于迭代的攻击方法，包括 FGSM[16]、

表 4.2　检测不同攻击构造的对抗样本得到的 AUC 分数（10^{-2}）

攻击	训练	MNIST			CIFAR-10		
		置信度	非最大熵	核密度	置信度	非最大熵	核密度
FGSM	CE	79.7	66.8	98.8	71.5	66.9	**99.7**
	RCE	98.8	98.6	**99.4**	92.6	91.4	98.0
BIM	CE	88.9	70.5	90.0	0.0	64.6	**100.0**
	RCE	91.7	90.6	**91.8**	0.7	70.2	**100.0**
ILCM	CE	98.4	50.4	96.2	16.4	37.1	84.2
	RCE	100.0	97.0	**98.6**	64.1	77.8	**93.9**
JSMA	CE	98.6	60.1	97.7	99.2	27.3	85.8
	RCE	100.0	99.4	**99.0**	99.5	91.9	**95.4**
C&W	CE	98.6	64.1	99.4	99.5	50.2	95.3
	RCE	100.0	99.5	**99.8**	99.6	94.7	**98.2**
C&W-hc	CE	0.0	40.0	91.1	0.0	28.8	75.4
	RCE	0.1	93.4	**99.6**	0.2	53.6	**91.8**

BIM[17]、ILCM[17] 以及 JSMA[148]。在图 4.4中，我们报告了在不同扰动大小（ℓ_∞-范数意义）下模型的预测准确率。可以看到就算不使用检测机制，反交叉训练出来的模型依旧比交叉熵训练出的模型更鲁棒，即在对抗攻击下拥有更高的预测准确率。

我们也测试了基于优化的攻击算法 C&W[47]。具体来说，C&W 攻击引入一个额外变量 ω，并且定义 $x^* = \frac{1}{2}(\tanh(\omega)+1)$。C&W 攻击求解问题 $\min_\omega \|\frac{1}{2}(\tanh(\omega)+1) - x\|_2^2 + c \cdot f(\frac{1}{2}(\tanh(\omega)+1))$，其中 c 为需要二分查找的超参数，$f(x) = \max(\max\{Z_{pre}(x)_i : i \neq y\} - Z_{pre}(x)_i, -\kappa)$，$\kappa$ 为控制置信度的超参数，默认值为 0。在实现中，我们设定对 c 执行最多 9 次二分查找，每次查找过程中 C&W 攻击的优化步数为 10000，优化步长为 0.01。我们同时引入了高置信度版本的 C&W 攻击（high-confidence C&W，缩写为 C&W-hc），其中 κ 设置为 10。在图 4.5中，我们报告了 C&W 和 C&W-hc 攻破交叉熵、反交叉熵训练的模型所需的平均最小扰动。这里扰动大小（distortion）是通过公式 $\|x - x^*\|_2/\sqrt{d}$ 计算的，其中 x^* 的像素值保持在 $[0, 255]$ 之中。从结果上看，成功攻破反交叉熵训练的模型需要更大的对抗扰动。

进一步地，我们加入检测机制，即预设一个阈值，当检测指标高于该阈值时返回预测，低于该阈值时则认为输入为对抗样本，从而拒绝预测。在表 4.2中，我

表 4.3　$f_2(x^*) > 0$ 比例以及攻击成功所需最小扰动

训练	MNIST		CIFAR-10	
	$f_2(x^*) > 0$ 比例	最小扰动	$f_2(x^*) > 0$ 比例	最小扰动
CE	0.01	17.12	0.00	1.26
RCE	**0.77**	**31.59**	**0.12**	**3.89**

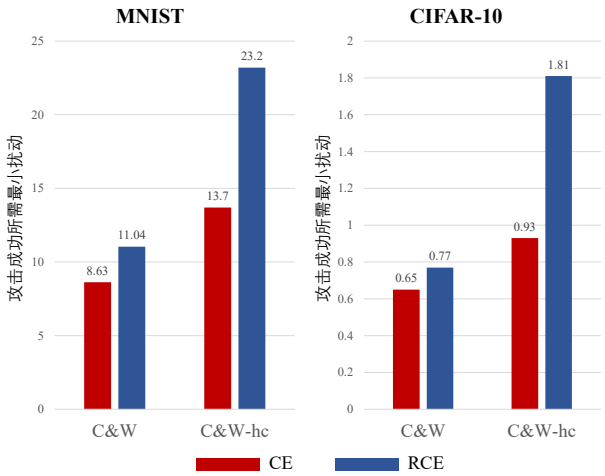


图 4.5　成功攻击模型所需最小扰动大小

们报告了在各类攻击算法下，交叉熵和反交叉熵训练得到的特征结合置信度、非最大熵或者核密度为检测指标时，检测结果的 ROC-AUC 分数。ROC-AUC 分数即为 ROC 曲线覆盖的面积，其中 ROC 曲线也被称作受试者工作特征曲线或者感受性曲线。从表 4.2的结果可以看出，反交叉熵训练结合核密度估计可以一致地在各类攻击下到达最好的 AUC 分数，既可以灵敏地检测出对抗样本，同时也降低误检率（把干净样本检测成对抗样本）。此外，我们观察到反交叉熵训练出的模型的置信度也更加可靠，可以比交叉熵训练的情况更加好地区分对抗样本与干净样本。

4.3.2　白盒攻击下的鲁棒性

在白盒攻击设定下，攻击者不光可以知道分类模型的参数，还可以知道检测模型的机制。在这种场景下，最典型的攻击为自适应或者白盒版本的 C&W 攻击算法（white-box version of the C&W attack[47]，缩写为 C&W-wb）。相比于原本的 C&W 算法，C&W-wb 引入了额外的损失函数项 $f_2(x^*) = \max(-\log(KD(x^*)) - \eta, 0)$ 来攻击核密度估计检测器，其中 η 被设为核密度 $-\log(KD(\cdot))$ 在训练集上的中位数（median）。具体来说，C&W-wb 构造的对抗样本 x^* 在欺骗分类模型的同时，还要避免被核密度估计检测器检测到。在表 4.3中，我们计算了成功攻破分类模型且

表 4.4　检测黑盒迁移对抗样本的 AUC 分数（10^{-2}）

	Resnet-32 (CE)	ResNet-32 (RCE)
Resnet-56 (CE)	75.0	90.8
Resnet-56 (RCE)	89.1	84.9

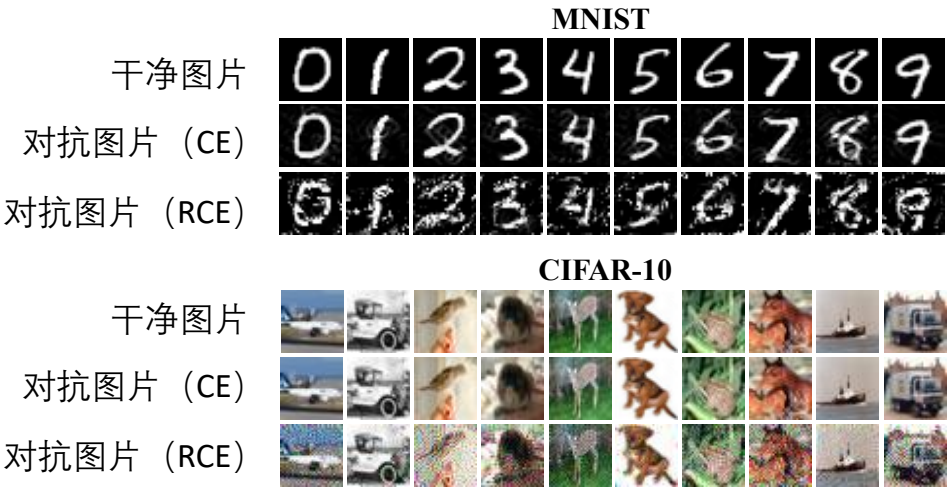


图 4.6　对抗样本可视化

检测指标超过阈值所需要的最小对抗扰动（distortion），以及所有对抗样本中满足 $f_2(x^*) > 0$ 的比例。具体来说，$f_2(x^*) > 0$ 说明对抗样本 x^* 的检测指标未能超过干净样本上的中位数，因此容易被检测器检测出来。从表 4.3的结果中我们可以看到，在交叉熵训练的基线模型上，几乎所有对抗样本都可以达到 $f_2(x^*) \leqslant 0$（即 $f_2(x^*) > 0$ 的比例几乎为零），说明检测器很容易被欺骗。相比之下，反交叉熵训练的模型可以维持更高的 $f_2(x^*) > 0$ 比例，说明在我们的方法里检测器显著地更加鲁棒。在图 4.6中我们展示了 C&W-wb 算法构造的对抗样本。可以看到为了成功攻击反交叉熵训练的模型，攻击者构造的对抗噪音显著大于攻击交叉熵训练的模型，且已经达到了肉眼可见的水平。

4.3.3　黑盒攻击下的鲁棒性

为了进行全面的评估，我们也测试了模型在黑盒迁移攻击下的检测能力。我们使用 Resnet-56 模型结构作为目标模型（target models），同时用 Resnet-32 模型结构作为替代模型（substitute models）。黑盒迁移攻击在替代模型上构造对抗样本，之后输入到目标模型中进行攻击。我们假设攻击者知道检测器的存在，因此使用 C&W-wb 攻击算法构造对抗样本。然而，我们发现 C&W-wb 攻击算法构造的

对抗样本迁移性很差，在 MNIST 上只有不到 50% 的迁移成功率，在 CIFAR-10 上不到 15% 的迁移成功率。此外，在表 4.4中我们报告了检测黑盒迁移对抗样本的 ROC-AUC 分数。可以看到我们的方法可以有效地检测出黑盒迁移对抗样本。

4.4　本章小结

在本章中，我们考虑如何训练模型，使得学到的特征分布可以更好地与现有的检测指标（例如核密度估计）相结合。通过理论分析，我们提出了反交叉熵训练准则。相比于传统使用的交叉熵训练准则，我们的方法可以将干净样本的特征映射到低维流形上，从而当输入为对抗样本时，检测器可以更加灵敏地发现对抗特征。实现上，反交叉熵训练准则不增加额外的计算量，代码实现极其简单，且不影响模型在干净样本上的预测准确率。我们在 MNIST 以及 CIFAR-10 上的各类攻击算法和攻击场景下验证了我们方法的有效性。

第 5 章　基于置信度修正的互耦对抗样本检测

在上一章中，我们主要聚焦于如何训练模型可以使得其特征分布与某一检测指标更好地配合。除此之外，本章我们探索如何使用多个（相互关联的）检测指标来更加可靠地区分出对抗样本。在对抗检测指标方面，*置信度*（confidence）是最常用的预测确定度（certainty）衡量指标之一[1]。沿着这一思路，我们发现*修正置信度*（rectified confidence，缩写为 R-Con）可以和置信度形成一对互耦（coupled）检测指标，也就是说修正置信度和置信度结合起来可以保证（provably）区分出分类错误样本和分类正确样本。这一有趣的结论展示了互耦检测指标潜在的应用价值。实验中，我们在 CIFAR-10，CIFAR-10-C 以及 CIFAR-100 数据集上验证了我们的*修正滤除*（rectified rejection，缩写为 RR）方法，测试了多种攻击下（包括自适应攻击）模型的鲁棒性。此外，我们的方法可以和各种对抗训练方法相结合，且不引入额外计算量。

5.1　本章引言

机器学习模型的对抗鲁棒性缺陷因其反直觉性和对安全层面的潜在影响而被广泛研究[15-16,125]。为此，之前的工作提出了许多防御措施来增强模型的鲁棒性，但大多数防御方法都可以被自适应攻击欺骗[48,182]。在之前提出的防御策略中，对抗训练（adversarial training，缩写为 AT）被公认为是最有效的防御方法之一[29-30]。尽管如此，在对抗领域内权威的基准评测榜单 RobustBench[36] 上，即使在利用大量额外数据之后，最鲁棒的模型在 CIFAR-10 上的对抗样本预测准确率仍然难以超过 67%[123,127,183-184]，这相比实际应用场景中所要求的鲁棒性水平还有很大的距离。

为了能进一步得到更可靠的预测，我们可以考虑引入对抗样本检测机制，通过允许模型在不确定的输入上拒绝返回预测来规避潜在的严重风险[185-187]。尽管之前的相关工作提出训练额外的基于边际距离（margin-based）或者置信度校准（confidence calibration）的检测方法，但是这些方法会过度估计（overestimate）模型预测的确定度，特别是在被模型错误分类的输入样本上。此外，文献[188] 也提出学习一个鲁棒的检测器并不比学习一个鲁棒的分类器更容易，且这些鲁棒学习过程都会受到数据量不足[122] 或者泛化性不够[189] 等问题的影响。

为了解决这一问题，我们首先观察到*真实的*（true）交叉熵损失函数 $-\log f_\theta(x)[y]$ 可以反映模型 $f_\theta(x)$ 在输入 x 上的泛化好坏[1]（这里假设我们可以知道真实类别 y）。因此，我们提出将*真实置信度*（true confidence，缩写为 T-Con）

表 5.1　CIFAR-10 上在 TPR-95 前提下的预测准确率（%）

	输入样本	All	TPR-95	
			Confidence	T-Con
标准训练	Clean	95.36	98.40	**100.0**
	PGD-10	0.22	0.18	**100.0**
对抗训练	Clean	82.67	87.39	**96.55**
	PGD-10	53.58	57.23	**88.75**
可获得性			✓	✗

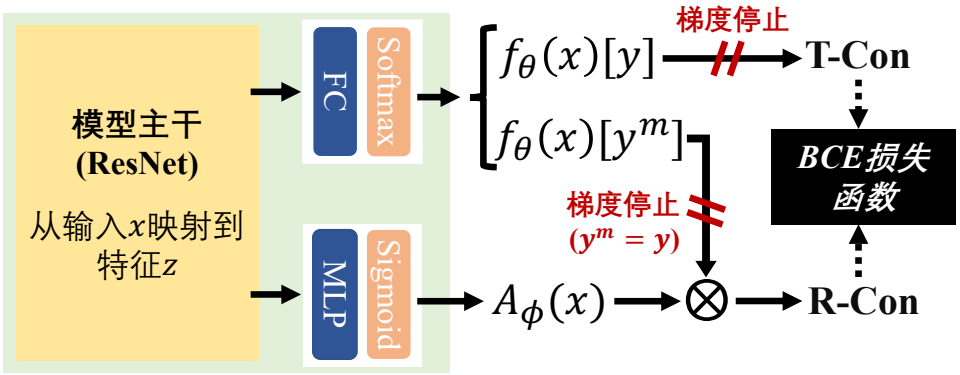


图 5.1　修正滤除方法训练及模型结构示意图

$f_\theta(x)[y]$ 当做表征模型预测确定度的衡量（oracle）。注意到，真实置信度和我们常用的*置信度*（confidence）不同，具体来说，置信度是通过对预测向量取最大元素 $\max_l f_\theta(x)[l]$ 获得的，并不需要知道真实类别。在表 5.1中我们可以看到，假设我们可以在测试阶段知道真实置信度的话，我们就可以得到很高的预测准确率。其中 TPR-95 为 95% 真正例率（true positive rate）的情况；All 为不进行对抗样本检测（即对所有输入都返回预测类别）的情况。

很巧妙的是，我们发现如果首先只保留置信度大于 $\frac{1}{2}$ 的输入样本，那么这些输入样本中，如果模型可以正确分类该样本，则其真实置信度必然大于 $\frac{1}{2}$；反之若模型会错误分类该样本，则其真实置信度必然小于 $\frac{1}{2}$。因此，置信度和真实置信度可以构成一对*互耦的*（coupled）检测指标，它们可以完全区分正确和错误分类的样本。这一性质对于检测对抗样本来说非常有用，但是由于实际中我们不可能在测试阶段预先知道真实类别 y，所以我们也无法在测试阶段计算真实置信度。因此，我们构造*修正置信度*（R-Con）来在训练过程中学习预测真实置信度，并应用在测试阶段。我们证明如果修正置信度被训练到 ξ-误差（ξ-error，其中 $\xi \in [0, 1)$），那么一个 ξ-误差的修正置信度和阈值为 $\frac{1}{2-\xi}$ 的置信度同样可以构成一对互耦的检测指标，它们可以完全区分出正确和错误分类的样本。

技术上来讲，如图 5.1中所示，我们采用双分支结构来分别建模分类器和我们

的修正滤除（RR）模块，并且同时对这两个模块进行对抗训练。具体来说，我们的修正滤除模块通过最小化修正置信度（R-Con）和真实置信度（T-Con）之间的双值交叉熵（binary cross-entropy，缩写为 BCE）损失函数来进行训练。分类器和滤除检测模块共享模型主干，从而节省参数规模和计算量。当 $y^m = y$ 时，我们使用梯度停止（stopping gradients）作用在 $f_\theta(x)[y^m]$ 上，这样可以保证分类器的最优解不受影响（即保持无偏估计），且避免模型专注于简单样本。

实验上，我们在 CIFAR-10、CIFAR-10-C 以及 CIFAR-100[119,126] 上验证了我们方法的有效性，特别是当使用置信度和修正置信度作为互耦检测指标时，可以严格区分正确与错误分类的样本。此外，为了和之前的基准模型进行对比，我们也可以选择仅用修正置信度作为单独的检测指标，并设置固定的真正例率（TPR）来评测模型的预测准确率。我们还进行了各类消融实验来评估我们的修正滤除模块中各个组成部分的作用。实验结果表明，我们的方法可以和各种对抗训练方法兼容，并且在多种对抗攻击下均可以相比基线模型进一步提高鲁棒性，不需要额外的计算量，且代码上容易实现。

5.2　相关工作

在标准训练（standard training）的设定下，文献[190] 首先提出联合训练分类器和检测器的策略，并随后被扩展到深度学习领域[191-192]。最近，文献[186] 以及文献[185] 在对抗训练分类器的过程中联合训练一个额外的检测器，并利用基于边际距离的（margin-based）损失函数。然而这些方法舍弃了置信度中自带的有用信息，造成了信息丢失[193]。另一方面，文献[187] 提出了基于置信度校准的对抗训练方法（confidence-calibrated AT，缩写为 CCAT），通过自适应的标签平滑技术，使得检测器可以在训练中未见过的攻击上更加灵敏。然而，CCAT 聚焦于校正真实置信度，而忽略了在测试阶段置信度与真实置信度之间的差别，特别是在错误预测的样本上。相比之下，我们将真实置信度视为模型预测确定度的衡量，并通过修正置信度来学习真实置信度。原理上，我们的方法与 CCAT 是相互兼容的，可以进一步提升检测或者滤除操作的可靠性。

5.3　算法设计

考虑输入 $x \in \mathbb{R}^d$，其真实类别为 y。我们将分类器模型定义为 $f_\theta(x) : \mathbb{R}^d \Rightarrow \Delta^L$，其中 θ 为模型参数，Δ^L 为 L 类别概率分布的定义域。根据文献[192]，一个分类器

结合上一个检测或者滤除模块 $\mathcal{M}$ 的情况可以写为

$$(f_\theta, \mathcal{M})(x) \triangleq \begin{cases} f_\theta(x), & \text{if } \mathcal{M}(x) \geqslant t; \\ \text{don't know}, & \text{if } \mathcal{M}(x) < t, \end{cases} \tag{5.1}$$

其中 t 为预设的阈值，$\mathcal{M}(x)$ 为某一确定度的度量（即检测指标）。

我们应该滤除什么样的输入样本? 检测指标 $\mathcal{M}$ 的设计很大程度上取决于我们的目的，即我们应该滤除什么样的输入样本。在对抗环境中，大部分之前的工作聚焦于滤除对抗样本，且这些对抗样本中的大多数可以成功欺骗分类模型[146]。在这种情况下，“错误分类样本”和“对抗样本”可以认为是近似相同的概念。然而，对于对抗训练之后的分类模型，至少 50% 左右的对抗样本是无法欺骗分类模型的（即会被正确分类）。因此在这种情况下，正确分类的对抗样本不应该被检测器滤除。基于此，在本章中我们聚焦于滤除错误分类的输入样本。

5.3.1　真实置信度的性质

为了滤除错分样本，有很多现成的选择可以用来作为检测指标 $\mathcal{M}(x)$。我们用 $f_\theta(x)[l]$ 来代表分类模型预测向量的第 l 个分量，并且其相应的预测类别为 $y^m = \operatorname{argmax}_l f_\theta(x)[l]$。我们将 $f_\theta(x)[y^m]$ 称作*置信度*（confidence[1]）。在标准训练设定下，置信度常被用来衡量模型预测的确定度[191]。然而在对抗环境中，标准训练得到的模型其置信度指标很容易被攻击者操控[136]。

和通过取最大值 $\max_l f_\theta(x)[l]$ 得到的置信度不同，我们聚焦于*真实置信度*（T-Con），定义为 $f_\theta(x)[y]$，即分类模型在真实类别 y 上的预测概率。当模型通过最小化交叉熵损失函数 $\mathbb{E}[-\log f_\theta(x)[y]]$ 进行训练时，相应地 $-\log f_\theta(x)[y]$ 的值也可以更加准确地反映出模型在 x 上预测的确定度，尽管在实际的测试阶段我们只能计算 $-\log f_\theta(x)[y^m]$。注意到在错分样本（即 $y^m \neq y$）上指标 $-\log f_\theta(x)[y^m]$ 会高估真正的确定度 $-\log f_\theta(x)[y]$。

如表 5.1中所示，我们在 CIFAR-10 上分别使用标准训练和对抗训练得到两种模型，并且测试了置信度和真实置信度作为检测指标 $\mathcal{M}$ 的效果。这里我们假设在测试阶段可以计算真实置信度。我们报告了不使用检测机制的情况（All）以及真正例率在 95% 的情况（TPR-95），即最多 5% 的正确分类样本会被滤除掉。可以看到，真实置信度可以很有效地分辨正确和错误分类的样本。为了解释这一现象，注意到标准训练的模型通常会返回很高的置信度，例如 0.95，不论输入是干净样本还是对抗样本[178]。在这种情况下，若输入 x 被正确分类，则其对应的真实置信度 $\text{T-Con}(x) = 0.95$；反之若输入 x 被错误分类，则其对应的真实置信度 $\text{T-Con}(x) < 1 - 0.95 = 0.05$。因此我们会看到在表 5.1中的标准训练情况下，使用真

实置信度可以得到 100% 的 TPR-95 预测准确率。由此，我们将真实置信度视作模型预测确定度的衡量，且在训练过程中真实置信度为检测器提供监督学习信号。

除了使用单个检测指标之外，我们发现置信度和真实置信度可以构成一对互耦的检测指标，其性质如下所述：

引理 5.1： 给定分类器 f_θ，对于 $\forall x_1, x_2$ 满足对应的置信度大于 $\frac{1}{2}$，即

$$f_\theta(x_1)[y_1^m] > \frac{1}{2}, \text{ and } f_\theta(x_2)[y_2^m] > \frac{1}{2}. \tag{5.2}$$

若 x_1 被正确分类 $y_1^m = y_1$，而 x_2 被错误分类 $y_2^m \neq y_2$，那么我们有 T-Con$(x_1) > \frac{1}{2} >$ T-Con(x_2)。

证明： 因为 x_1 被正确分类，即 $y_1^m = y_1$，所以 $f_\theta(x_1)[y_1] = f_\theta(x_1)[y_1^m] > \frac{1}{2}$。另一方面，由于 x_2 被错误分类，即 $y_2^m \neq y_2$，所以我们有 $f_\theta(x_2)[y_2] \leqslant 1 - f_\theta(x_2)[y_2^m] < \frac{1}{2}$。最后可得 T-Con$(x_1) > \frac{1}{2} >$ T-Con(x_2)。 ∎

直观上来讲，引理 5.1说明若我们先按照置信度的值设定一个阈值 $\frac{1}{2}$，即预测置信度大于 $\frac{1}{2}$ 的输入我们允许模型返回预测类别，否则模型则拒绝返回预测类别。那么在任何模型返回预测类别的输入 x 上，若进一步有其真实置信度 T-Con$(x) < \frac{1}{2}$，那么 x 一定是被错误分类的；若 T-Con$(x) > \frac{1}{2}$ 那么 x 一定是被正确分类的。注意到这里我们没有对错误分类的原因进行任何假设，即错误分类可以是对抗攻击造成的，也可以是任何其他原因（例如图片旋转或者模糊等等）造成的。

5.3.2　通过置信度修正来学习真实置信度

当输入 x 被分类器 f_θ 正确分类时（即 $y^m = y$），其对应的置信度和真实置信度相等。因此为了学习真实置信度，我们可以通过对置信度进行修正来建模，而非从头建模。具体来说，我们引入一个额外的修正函数 $A_\phi(x) \in [0, 1]$，其参数为 ϕ。由此我们构造修正置信度（R-Con）如下：

$$\text{R-Con}(x) = f_\theta(x)[y^m] \cdot A_\phi(x). \tag{5.3}$$

在训练过程中，我们鼓励修正置信度和真实置信度保持一致。为了达到这一目的，我们最小化修正置信度与真实置信度之间的双值交叉熵（BCE）损失函数。这里双值交叉熵损失函数定义为

$$\text{BCE}(f \parallel g) = -g_\dagger \cdot \log f - (1 - g_\dagger) \cdot \log(1 - f), \tag{5.4}$$

其中下角标 $\dagger$ 表示梯度停止（stopping gradients）。下面可以写出训练我们的修正滤除（RR）模块所需的目标函数

$$\mathcal{L}_{\text{RR}}(x, y; \theta, \phi) = \text{BCE}\left(f_\theta(x)[y^m] \cdot A_\phi(x) \parallel f_\theta(x)[y]\right), \tag{5.5}$$

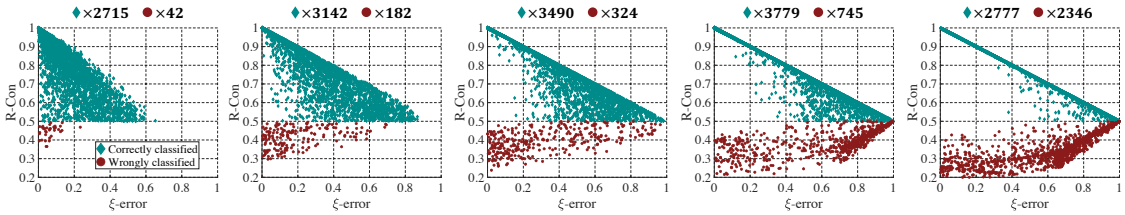


图 5.2　置信度与修正置信度区分 CIFAR-10 上构造的 PGD 对抗样本

这里最小化 $\mathcal{L}_{\mathrm{RR}}$ 的最优解为

$$A_\phi^*(x) = \frac{f_\theta(x)[y]}{f_\theta(x)[y^m]}. \tag{5.6}$$

修正函数 $A_\phi(x)$ 可以和分类器 $f_\theta(x)$ 一同训练，整体的训练目标函数写为

$$\begin{aligned} &\min_{\theta,\phi} \mathbb{E}_{p(x,y)}\big[\underbrace{\mathcal{L}_{\mathrm{T}}(x^*, y; \theta)}_{\textbf{classification}} + \lambda \cdot \underbrace{\mathcal{L}_{\mathrm{RR}}(x^*, y; \theta, \phi)}_{\textbf{rectified rejection}}\big], \\ &\text{其中} x^* = \operatorname{argmax}_{x' \in B(x)} \mathcal{L}_{\mathrm{A}}(x', y; \theta). \end{aligned} \tag{5.7}$$

这里 λ 为超参数，$B(x)$ 代表 x 周围攻击者的扰动范围（例如 $\|x' - x\|_p \leqslant \epsilon$），$\mathcal{L}_{\mathrm{T}}$ 和 $\mathcal{L}_{\mathrm{A}}$ 分别为对抗训练中的训练目标函数和对抗样本构造函数[31]。更一般地，公式（5.7）中的最小化问题可以包含干净样本 x，从而包括了 TRADES 这种混合对抗训练框架[30]。类似地，公式（5.7）中最大化问题部分也可以包含 ϕ。

A_ϕ 的结构：我们考虑包含 softmax 层的分类器结构，写为 $f_\theta(x) = \mathbb{S}(Wz + b)$，其中 z 为特征，W 和 b 分别为 softmax 层的权重矩阵和偏置向量。如图 5.1中所示，我们使用一个额外的浅全连接网络（shallow MLP）来建模 $A_\phi(x) = \mathrm{MLP}_\phi(z)$。其他网络结构例如限制玻尔兹曼机等等也可以在这里使用[194-195]。注意到，我们在信息流路径 $f_\theta(x)[y] \to$ BCE loss 和 $f_\theta(x)[y^m] \to$ R-Con（$y^m = y$ 时）上加入了梯度停止操作。这些梯度停止操作可以防止模型过度聚焦于简单样本，且可以保证分类器 $f_\theta(x)[y]$ 的最优解与真实分布 $p_{\mathrm{data}}(y|x)$ 相一致。

A_ϕ 的训练误差：在实际的训练过程中，修正函数 $A_\phi(x)$ 与其最优解 $A_\phi^*(x)$ 之间会有一定的误差。数学上，我们定义 $A_\phi(x)$ 与 $A_\phi^*(x)$ 在 x 点上的误差满足如下

定义 5.1： 若下列两个条件满足至少任意一条：

$$\begin{aligned} &\text{条件 (i):} \quad \left|\log\left(\frac{A_\phi(x)}{A_\phi^*(x)}\right)\right| \leqslant \log\left(\frac{2}{2-\xi}\right); \\ &\text{条件 (ii):} \quad \left|A_\phi(x) - A_\phi^*(x)\right| \leqslant \frac{\xi}{2}. \end{aligned} \tag{5.8}$$

其中 $\xi \in [0, 1)$，那么我们称 $A_\phi(x)$ 在 x 点上达到了 ξ-误差。

我们可以证明对于任何的比随机猜想（random guess）学得更好的修正函数 A_ϕ，

我们总可以找到 $\xi \in [0,1)$ 满足定义 5.1。具体来说，假设 A_ϕ 在点 x 上是随机猜想，即 $A_\phi(x) = \frac{1}{2}$。那么由于 $A_\phi^*(x) \in [0,1]$，我们有 $\left|A_\phi(x) - A_\phi^*(x)\right| = \left|\frac{1}{2} - A_\phi^*(x)\right| \leqslant \frac{1}{2}$ 成立，说明一个随机猜想的 A_ϕ 也可以满足定义 5.1中的条件 (ii)，其中 $\xi = 1$。

5.3.3　置信度与修正置信度互耦

回忆我们在引理 5.1中展示了置信度（confidence）与真实置信度（T-Con）可以构成一对互耦的检测指标，从而区分任何的正确与错误分类的样本。然而在实际的测试阶段，我们无法计算真实置信度。因此，在测试阶段，我们使用修正置信度（R-Con），并且证明置信度与修正置信度也可以构成一对互耦的检测指标，具体如下述定理所示：

定理 5.1： 给定分类器 f_θ，对于任何的 x_1，x_2 满足其预测置信度大于 $\frac{1}{2-\xi}$，即

$$f_\theta(x_1)[y_1^m] > \frac{1}{2-\xi}, \text{以及} f_\theta(x_2)[y_2^m] > \frac{1}{2-\xi}, \tag{5.9}$$

其中 $\xi \in [0,1)$。若 x_1 被正确分类（$y_1^m = y_1$），而 x_2 被错误分类（$y_2^m \neq y_2$），且 A_ϕ 在 x_1，x_2 上是 ξ-误差的，则可以得出 $\text{R-Con}(x_1) > \frac{1}{2} > \text{R-Con}(x_2)$。

证明： 定理 5.1中的条件可以写为 $f_\theta(x_1)[y_1^m] > \frac{1}{2-\xi}, y_1^m = y_1$ 以及 $f_\theta(x_2)[y_2^m] > \frac{1}{2-\xi}$，$y_2^m \neq y_2$，其中 $\xi \in [0,1)$。由于 $A_\phi(x)$ 在 x_1，x_2 上是 ξ-误差的，所以根据定义 5.1，下述两个条件至少有一个成立：

$$\begin{aligned} &\text{条件 (i):} \ \left|\log\left(\frac{A_\phi(x)}{A_\phi^*(x)}\right)\right| \leqslant \log\left(\frac{2}{2-\xi}\right); \\ &\text{条件 (ii):} \ \left|A_\phi(x) - A_\phi^*(x)\right| \leqslant \frac{\xi}{2}. \end{aligned} \tag{5.10}$$

对于 x_1，我们有 $A_\phi^*(x_1) = 1$。那么若条件 (i) 成立，我们可以得到

$$\begin{aligned} \text{R-Con}(x_1) &= f_\theta(x_1)[y_1^m] \cdot A_\phi(x_1) \\ &> f_\theta(x_1)[y_1^m] \cdot \frac{2-\xi}{2} \\ &> \frac{1}{2-\xi} \cdot \frac{2-\xi}{2} = \frac{1}{2}, \end{aligned} \tag{5.11}$$

且若条件 (ii) 成立，我们可以得到

$$\begin{aligned} \text{R-Con}(x_1) &= f_\theta(x_1)[y_1^m] \cdot A_\phi(x_1) \\ &> f_\theta(x_1)[y_1^m] \cdot \left(1 - \frac{\xi}{2}\right) \\ &> \frac{1}{2-\xi} \cdot \frac{2-\xi}{2} = \frac{1}{2}. \end{aligned} \tag{5.12}$$

类似地，对于 x_2 有 $f_\theta(x_2)[y_2^m] \cdot A_\phi^*(x_2) = f_\theta(x_2)[y_2]$。那么若条件 (i) 成立，我们可

以得到

$$\begin{aligned}\text{R-Con}(x_2) &= f_\theta(x_2)[y_2^m] \cdot A_\phi(x_2) \\ &= f_\theta(x_2)[y_2^m] \cdot A_\phi^*(x_2) \cdot \frac{A_\phi(x_2)}{A_\phi^*(x_2)} \\ &< f_\theta(x_2)[y_2] \cdot \frac{2}{2-\xi} \\ &< \left(1 - \frac{1}{2-\xi}\right) \cdot \frac{2}{2-\xi} \\ &= \frac{2-2\xi}{(2-\xi)^2} < \frac{1}{2},\end{aligned} \tag{5.13}$$

其中我们很容易验证 $\frac{2-2\xi}{(2-\xi)^2}$ 在区间 $\xi \in [0, 1)$ 是单调递减的。若条件 (ii) 成立，我们可以得到

$$\begin{aligned}&\text{R-Con}(x_2) \\ &= f_\theta(x_2)[y_2^m] \cdot A_\phi(x_2) \\ &< f_\theta(x_2)[y_2^m] \cdot \left(\frac{f_\theta(x_2)[y_2]}{f_\theta(x_2)[y_2^m]} + \frac{\xi}{2}\right) \\ &= f_\theta(x_2)[y_2] + f_\theta(x_2)[y_2^m] \cdot \frac{\xi}{2} \\ &= f_\theta(x_2)[y_2] \cdot \left(1 - \frac{\xi}{2}\right) + \left(f_\theta(x_2)[y_2] + f_\theta(x_2)[y_2^m]\right) \cdot \frac{\xi}{2} \\ &< \left(1 - \frac{1}{2-\xi}\right) \cdot \left(1 - \frac{\xi}{2}\right) + \frac{\xi}{2} = \frac{1}{2}.\end{aligned} \tag{5.14}$$

由此我们证明了必有 $\text{R-Con}(x_1) > \frac{1}{2} > \text{R-Con}(x_2)$ 成立。 ∎

定理 5.1说明当我们先对预测置信度设定一个阈值 $\frac{1}{2-\xi}$ 时，所有保留下来的输入样本（即置信度大于阈值）中，若修正函数 A_ϕ 达到 ξ-误差，则任何错误分类的样本对应的修正置信度都一定小于任何正确分类的样本。这一性质可以防止自适应攻击者同时欺骗分类类别和修正置信度的值。在图 5.2中，我们在 CIFAR-10 上用修正滤除（RR）训练方法得到了一个鲁棒的 ResNet-18 模型，且对其构造 PGD 对抗样本。对每个输入样本 x，我们首先算出修正函数在 x 上对应的 ξ-误差（这里我们假设知道真实置信度是多少），并且相应地对 x 设置置信度滤除的阈值为 $\frac{1}{2-\xi}$（预测置信度大于阈值则保留；反之滤除）。我们使用绿色代表被模型正确分类的样本，用红色代表被模型错误分类的样本。可以看到，正如我们在定理 5.1中所证明的，修正置信度可以完全地区分出两者。注意到，尽管在实际的测试场景中我们无法显式地计算出 ξ，但是这一互耦机制仍然隐式地发挥作用，提高检测的灵敏度。此外，在图 5.2中，我们定义 softmax 层的形式为 $f_\theta(x) = \text{softmax}(\frac{Wz+b}{\tau})$，其中 $\tau > 0$ 为一超参数，通常称作“温度”。从结果中可以看到，调整温度 τ 的大小可

以影响通过检测器的样本数目，控制正负样本比例。

5.3.4　修正函数的学习难度

文献[188]认为学习一个鲁棒的检测器并不比学习一个鲁棒的分类器更容易。所以一个自然的疑问就是：想要学到一个 ξ-误差的修正函数 A_ϕ 大概是什么样的难度? 在本小节中我们尝试回答这个问题。具体来说，由于 $A_\phi(x)$ 的值域限制在 $[0,1]$ 之中，所以我们可以将“学习 ξ-误差修正函数”这个回归（regression）任务转换成一个等价的分类（classification）任务：

定理 5.2： 学习一个 ξ-误差的修正函数 $A_\phi(x)$ 的任务可以转换成一个等价的分类任务，且该分类任务的类别数目为 N_{sub}，其中

$$N_1 = \frac{\log \rho^{-1}}{\log\left(\frac{2}{2-\xi}\right)} + 1,\ N_2 = \frac{2}{\xi},\ \text{以及} N_{\text{sub}} = \min(N_1, N_2). \tag{5.15}$$

这里 ρ 是预设的舍入误差。

证明： 由于 $A^*_\phi(x)$ 与 $A_\phi(x)$ 的值域均限定在 $[0,1]$ 内，所以我们用 $\{B_0, B_1, \cdots, B_S\}$ 来表示区间 $[0,1]$ 中的 $S+1$ 个点，其中 $B_0 = 0$ 且 $B_s = 1$。这些点可以诱导出 S 个子区间，即 $I_s = [B_{s-1}, B_s]$，其中 $s = 1, \cdots, S$。当 $A_\phi(x)$ 在 x 点上达到 ξ-误差时，我们分别考虑 $A_\phi(x)$ 满足定义 5.1中条件 (i) 和条件 (ii) 的情况。

条件 (i) 满足： 在此情况下，我们用等比数列的方式构造子区间，即 $B_s = \frac{2}{2-\xi} \cdot B_{s-1}$ 且我们设置 $B_1 = \rho$。注意到我们有

$$\rho \cdot \left(\frac{2}{2-\xi}\right)^{S-2} < 1 \leqslant \rho \cdot \left(\frac{2}{2-\xi}\right)^{S-1}, \tag{5.16}$$

因此我们可以推出

$$S = \frac{\log \rho^{-1}}{\log\left(\frac{2}{2-\xi}\right)} + 1. \tag{5.17}$$

我们很容易看出 $A_\phi(x)$ 和 $A^*_\phi(x)$ 落在相同的子区间内，即条件 (i) 得到满足。因此，满足条件 (i) 这一回归任务可以等效成类别数目为 $N_1 = \frac{\log \rho^{-1}}{\log\left(\frac{2}{2-\xi}\right)} + 1$ 的分类任务。

条件 (ii) 满足： 在此情况下，我们用等差数列的方式构造子区间，即 $B_s = B_{s-1} + \frac{\xi}{2}$，则我们得出

$$(S-1) \cdot \frac{\xi}{2} < 1 \leqslant S \cdot \frac{\xi}{2}, \tag{5.18}$$

由此可以进一步推出

$$S = \frac{2}{\xi}. \tag{5.19}$$

我们很容易看出 $A_\phi(x)$ 和 $A_\phi^*(x)$ 落在相同的子区间内，即条件 (ii) 得到满足。因此，满足条件 (ii) 这一回归任务可以等效成类别数目为 $N_2 = \frac{2}{\xi}$ 的分类任务。 ∎

直观上，定理 5.2提供了一种途径来预先估计出修正函数在多少测试样本上可以达到 ξ-误差。这是基于在类似的数据分布下，分类问题的难度随着分类类别数目的增加而增长[159]。例如，一个在 CIFAR-10 数据集上可以达到 90% 准确率的模型结构，在 CIFAR-100 上可能只能达到 70% 的准确率。根据定理 5.2，若我们想要修正函数在 CIFAR 数据集上达到 0.1-误差，那么这一任务可以等效地转化成一个类别数目为 20 的分类问题，那么我们可以近似认为 20 分类问题的难度介于 10 分类问题（CIFAR-10）和 100 分类问题（CIFAR-100）之间[196]。因此，同样的模型结构在 20 分类问题上的测试准确率大概在 90% 到 70% 之间，即 CIFAR 数据集或者数据分布上修正函数在大概 70%~90% 的测试样本上可以满足 $\xi = 0.1$。

5.4　实验结果

我们在 CIFAR-10, CIFAR-100 以及 CIFAR-10-C[119] 数据集上评测我们的方法。①我们选用两种最常用的模型结构：ResNet-18[144] 以及 WRN-34-10[174]。根据文献[35]中的训练建议，对于所有的防御方法，默认的训练参数包括批量大小 128；动量随机梯度下降优化器（SGD momentum optimizer），初始学习率 0.1；权重衰减 5×10^{-4}。我们对模型训练 110 轮，其中学习率在第 100 和 105 轮的时候乘以衰减因子 0.1。对于每个模型，我们报告其 PGD-10 攻击下准确率最高的存储点（checkpoint）[197]。

实验中使用的对抗训练框架：我们主要采用三种常用的对抗训练框架，包括 PGD-AT[29]，TRADES[30] 以及 CCAT[187]。对于 PGD-AT 和 TRADES，我们在训练中使用 PGD-10 算法构造对抗样本，扰动大小为 ℓ_∞-范数下的 8/255，扰动步长为 2/255。TRADES 中的平衡参数为 6[30]，而 CCAT 的实现上我们遵循其原始代码。在下述报告的结果中，我们用“**RR**”表示依照公式（5.7）训练出的模型，且使用修正置信度 R-Con 作为检测滤除的指标。在公式（5.7）中我们设定 $\lambda = 1$。

基线方法：我们与两类最常用的基线检测方法进行比较[198]。第一大类检测方法基于分类模型学出的特征，根据这些特征的统计信息来作为检测指标。这一类方法的典型代表包括 KD（kernel density）[93]、LID（local intrinsic dimensionality）[98]、GDA（Gaussian discriminant analysis）[106] 以及 GMM（Gaussian mixture model）[199]。第二大类检测方法基于训练一个额外的检测器模型，典型代表包括 SNet（SelectiveNet）[192]、EBD（energy-based detection）[200]、CARL[186]、ATRO[185]

① 源代码参见https://github.com/P2333/Rectified-Rejection。

表 5.2　CIFAR-10 上在 PGD-100 攻击下的 TPR-95 准确率（%）以及 ROC-AUC 分数

对抗训练	检测指标	干净样本		ℓ_∞, 8/255		ℓ_∞, 16/255		ℓ_2, 128/255	
		TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC
PGD-AT	KD	82.59	0.618	53.12	0.588	31.97	0.535	64.60	0.599
	LID	84.02	0.712	54.92	0.660	32.75	0.621	66.07	0.663
	GDA	82.35	0.453	52.67	0.461	31.89	0.454	64.13	0.459
	GDA*	84.51	0.664	53.88	0.589	31.94	0.527	65.71	0.605
	GMM	85.44	0.703	54.35	0.607	31.96	0.532	66.54	0.635
CARL	Margin	85.54	0.682	51.67	0.539	30.41	0.516	65.98	0.645
ATRO	Margin	73.42	0.669	36.04	0.654	21.37	0.644	41.52	0.655
TRADES	Con.	86.07	0.837	57.62	0.774	37.55	0.739	67.88	0.781
CCAT	Con.	92.44	0.806	51.68	0.637	45.12	0.683	67.07	0.772
PGD-AT	Con.	86.52	0.857	57.30	0.768	34.77	0.685	69.10	0.783
PGD-AT	SNet	84.19	0.796	56.41	0.730	35.25	0.692	67.49	0.741
PGD-AT	EBD	85.34	0.832	57.04	0.763	34.96	0.690	67.82	0.774
TRADES	**RR**	86.47	0.849	**58.52**	**0.786**	38.06	**0.748**	68.97	0.793
CCAT	**RR**	**94.12**	**0.909**	53.89	0.662	**48.02**	0.688	67.98	0.785
PGD-AT	**RR**	86.91	0.861	58.21	0.776	35.32	0.705	**70.24**	**0.796**

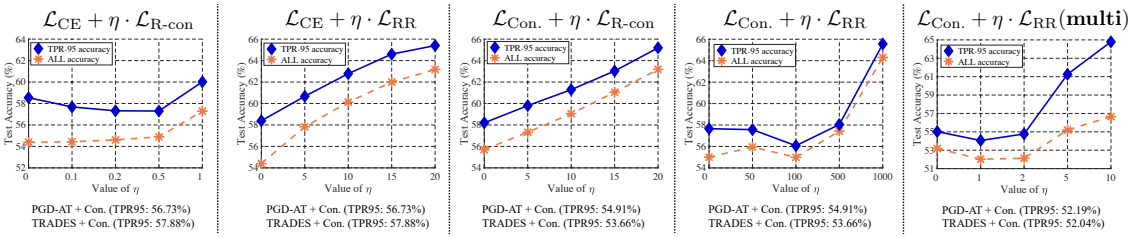


图 5.3　不同自适应攻击下的防御效果

以及 CCAT[187]。值得注意的是，这些基线模型在提出时大多是基于标准训练框架的。这里为了更加公平的比较，我们将这些基线模型移植到对抗训练框架中，并且对这些基线方法的超参数进行重新的调整（详情参见[124]），从而获得更加鲁棒的基线模型。

对抗攻击：作为鲁棒性的评测方法，实验中我们使用的对抗攻击包括 PGD[29]、C&W[146]、AutoAttack[28]、多目标攻击（multi-target attack）[201] 以及 GAMA[202]。我们同时在 CIFAR-10-C[119] 上测试了模型对于一般变换的鲁棒性。

表 5.3　CIFAR-10-C 上的 TPR-95 准确率（%）

对抗训练	检测指标	CIFAR-10-C									
		Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contra	Elastic	JPEG
PGD-AT	SNet	77.74	75.52	78.72	79.77	75.81	61.32	81.75	42.97	78.59	82.08
PGD-AT	EBD	78.47	77.92	80.47	81.17	79.14	61.16	83.98	42.10	80.86	83.34
CARL	Margin	77.45	74.94	78.00	79.86	74.16	56.09	81.28	40.33	78.17	82.64
ATRO	Margin	55.36	53.74	54.59	50.84	41.12	42.82	50.13	33.54	54.48	56.82
CCAT	Con.	83.04	85.47	89.33	**89.38**	88.21	76.32	**92.71**	55.99	89.34	91.94
TRADES	Con.	79.89	78.48	80.92	78.75	71.61	63.53	80.97	45.22	80.53	84.50
PGD-AT	**RR**	80.87	79.42	81.90	81.89	76.95	63.49	84.02	44.03	82.18	85.12
CCAT	**RR**	**85.03**	**86.26**	**89.83**	89.22	**88.41**	**77.45**	92.62	**58.95**	**89.59**	**92.06**
TRADES	**RR**	80.03	79.15	81.00	80.16	74.18	63.55	82.13	45.99	80.98	84.64

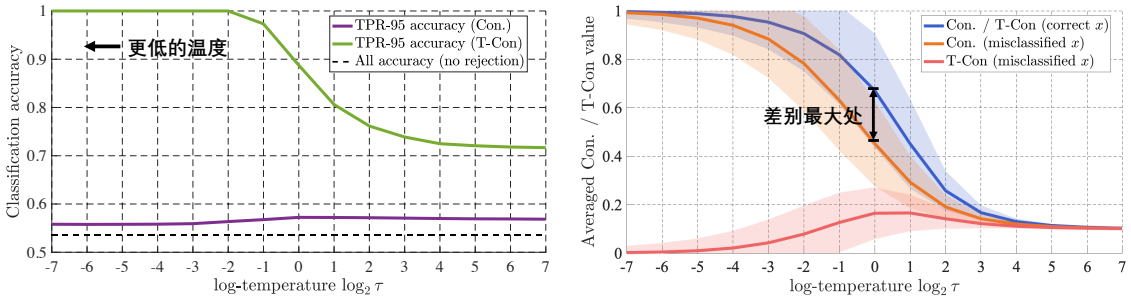


图 5.4　softmax 层中温度 τ 的效果

5.4.1　非自适应攻击下的鲁棒性

本小节中我们测试模型在非自适应攻击（也称作灰盒攻击）下的鲁棒性，即攻击者仅攻击分类器，而不攻击检测器或者检测指标。

PGD 攻击上的测试结果：在表 5.2中，我们报告了 CIFAR-10 上的结果，包括 TPR-95（真正例率 95%）准确率以及 ROC-AUC 分数。我们使用 PGD-100 攻击（$\ell_\infty, \epsilon = 8/255$）作为测试攻击，且包含了其他模型在训练阶段未见过的攻击设定，例如更大的扰动（$\epsilon = 16/255$）以及不一样的范数设定（ℓ_2-范数）。我们采用无目标攻击模式，且每次构造的对抗样本选取 10 次重启（restarts）中损失函数最大的。

更先进的攻击上的测试结果：在表 5.4中，我们测试了多类别攻击（multi-target attack）以及 GAMA 攻击。对于 AutoAttack，其攻击算法会返回成功攻击模型的对抗样本；若无法成功攻击模型则返回原始的干净样本。通过使用我们的 **RR** 方法训练一个 ResNet-18 模型，其在对于 AutoAttack 攻击下的 TPR-95 准确率为 84.32%

表 5.4　CIFAR-10 上在更先进的攻击下的 TPR-95 准确率（%）

对抗训练	检测指标	Multi-target	GAMA (PGD)	GAMA (FW)
PGD-AT	SNet	55.02	55.79	51.37
PGD-AT	EBD	55.40	56.15	53.24
CARL	Margin	46.17	48.49	44.78
ATRO	Margin	32.53	31.74	28.31
CCAT	Con.	34.21	49.78	38.01
TRADES	Con.	53.69	56.89	50.88
PGD-AT	**RR**	**56.18**	57.57	**54.08**
CCAT	**RR**	36.48	51.30	40.72
TRADES	**RR**	54.83	**57.93**	51.48

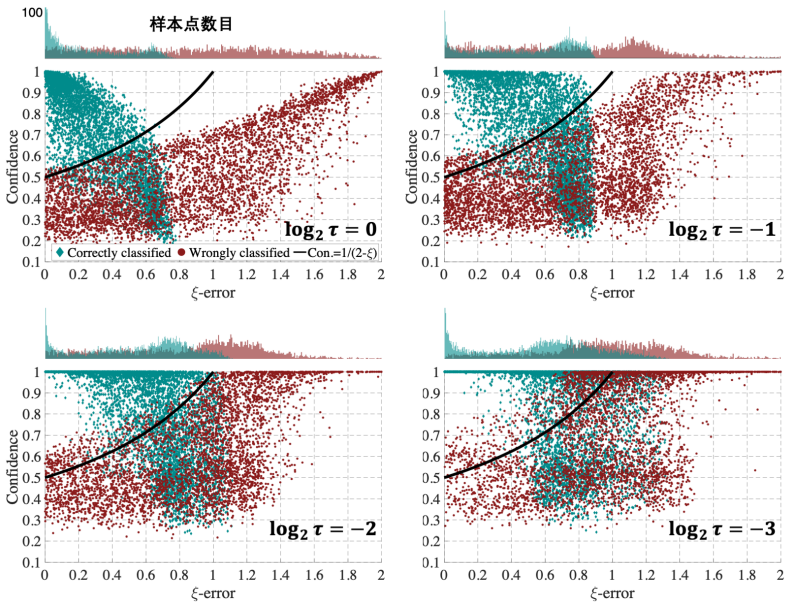


图 5.5　样本预测置信度随 ξ-误差的变化

（CIFAR-10 上）以及 70.99%（CIFAR-100 上）。

一般扰动模式下的测试结果：我们也在数据集 CIFAR-10-C 上测试了模型的表现。该数据集包含各种一般扰动（common corruptions）模式，具体结果在表 5.3中展示。综合上述结果，可以看到我们的 RR 方法可以和各种对抗训练框架相结合，且表现显著超过之前的基线方法。

表 5.5　不同温度 τ 下的 TPR-95 准确率（%）和 ROC-AUC 分数

$\log_2 \tau$	干净样本		PGD-10 对抗样本	
	TPR-95	AUC	TPR-95	AUC
−1	**86.86**	0.866	59.11	**0.770**
−2	86.62	0.865	60.63	0.762
−3	85.18	**0.868**	**61.12**	0.741
−4	80.22	0.836	55.15	0.740

表 5.6　关于修正滤除（RR）各模块的消融实验

检测指标	干净样本		PGD-10 对抗样本	
	TPR-95	AUC	TPR-95	AUC
$A_\phi(x)$	85.77	0.844	56.97	0.765
RR	**86.91**	**0.861**	**58.39**	**0.776**
$f_\theta(x)[y^m]$	86.76	0.865	57.42	0.768
RR (Con.)	**87.12**	**0.868**	**58.49**	**0.777**

5.4.2　自适应攻击下的鲁棒性

根据文献[46]的建议，我们设计自适应攻击来尝试同时欺骗分类模型和检测指标，具体如下所述。

评测自适应攻击下的准确率：在第一种自适应攻击模式中，我们考虑最常用的 ℓ_∞-范数扰动，且扰动大小为 8/255。为了充分测试我们的防御在自适应攻击下的效果，我们设计了五种不同的自适应攻击目标函数，包括 $\mathcal{L}_{\text{CE}}+\eta\cdot\mathcal{L}_{\text{R-Con}}$，$\mathcal{L}_{\text{CE}}+\eta\cdot\mathcal{L}_{\text{RR}}$，$\mathcal{L}_{\text{Con.}}+\eta\cdot\mathcal{L}_{\text{RR}}$，$\mathcal{L}_{\text{Con.}}+\eta\cdot\mathcal{L}_{\text{R-Con}}$ 以及 $\mathcal{L}_{\text{Con.}}+\eta\cdot\mathcal{L}_{\text{RR}}$(multi)，其中 $\mathcal{L}_{\text{Con.}}$ 代表直接优化置信度的值，$\mathcal{L}_{\text{R-Con}}=\log \text{R-Con}(\cdot)$，且 multi 代表多目标版本。实验结果如图 5.3中所示。可以看到，在这五种自适应攻击目标函数下，我们的防御方法都可以得到比基线方法更好的鲁棒性（即更高的 TPR-95 准确率）。

自适应攻击下所需最小扰动：在第二种自适应攻击模式中，我们根据之前的攻击思路[47]，将额外的最大化修正置信度的目标函数项添加到 C&W 攻击中，并且找到成功攻击分类器且使得修正置信度大于训练集上中位数（median）所需的最小扰动大小。在 C&W 攻击的每次二分查找中，我们执行 1000 步迭代，且总共进行 9 次二分查找。从表 5.7中的结果我们可以看出，自适应攻击（Ada.）相比非自适应攻击（Nor.）需要更大的最小扰动才能成功。此外，成功攻击我们的方法需要的最小扰动大于攻击基线方法需要的最小扰动。

表 5.7　自适应攻击所需的最小扰动大小

检测指标	CIFAR-10				CIFAR-100			
	CW-ℓ_∞		CW-ℓ_2		CW-ℓ_∞		CW-ℓ_2	
	Nor.	Ada.	Nor.	Ada.	Nor.	Ada.	Nor.	Ada.
SNet	14.30	30.48	0.84	2.70	8.20	23.05	0.56	2.37
EBD	14.70	37.54	0.85	2.42	8.58	25.69	0.60	1.81
RR	14.99	**38.58**	0.87	**3.28**	8.53	**28.67**	0.61	**3.21**

表 5.8　PGD-1000 攻击下的 TPR-95 准确率（%）以及 ROC-AUC 分数

检测指标	CIFAR-10		CIFAR-100	
	TPR-95	AUC	TPR-95	AUC
SNet	55.83	0.725	32.69	0.744
EBD	56.12	0.763	33.35	0.769
RR	**57.57**	**0.773**	**34.48**	**0.776**

5.4.3　消融实验

温度 τ 的影响：在图 5.4中我们展示了 TPR-95 准确率以及平均的置信度、真实置信度的值随温度 τ 的变化。此外，在图 5.5我们画出了 ξ-误差与置信度值的样本分布。可以看到，当温度 τ 变低时，真实置信度在正确与错误分类样本上的差别更大，从而可以更好地区分它们。然而这也造成数值上真实置信度会提供更少的监督信号。实际应用中，我们可以平衡温度 τ 的大小，进行交叉验证获得最优的温度值。在表 5.5中，我们探索了不同温度对于 $f_\theta(x)[y]$ 以及 $f_\theta(x)[y^m]$ 的影响。可以看到适度地降低温度可以提升模型鲁棒性，但是会影响在干净样本上的正常准确率。

修正置信度的构成：在表 5.6中，我们对修正置信度的构成进行消融实验。具体来说，我们考虑训练中的修正滤除项只有修正函数 $A_\phi(x)$ 或者只有置信度 $f_\theta(x)[y^m]$ 的情况。可以看到，修正滤除项中的每个部分都对最终的鲁棒性提升有所贡献。

5.5　本章小结

在本章中，我们引入真实置信度作为模型预测确定性的衡量，并且在训练过程中构造修正置信度，使其学会预测真实置信度。我们发现一个 ξ-误差的修正置信度指标和一个阈值为 $\frac{1}{2-\xi}$ 的置信度检测器可以构成一对互耦的检测方案，从而可以区分任何正确与错误分类的输入样本。此外，在实验中我们也验证了只使用修正置信度的效果，结果表明我们的方法相比于基线方法可以在各种攻击场景下

提升模型的鲁棒性。

局限性：尽管互耦检测指标这一性质很吸引人，但是正如我们上述分析中提到的，这一性质只有大概 50% 的输入样本可以满足。此外，在实际应用中，我们很难显式地控制互耦指标得到的真正例率。然而，本章中提出的置信度与修正置信度只是众多潜在互耦指标中的一个例子，之后的工作中可以继续探索性质更好、更加易用的互耦检测指标。

第 6 章　对抗训练中的技巧及参数设定

对抗训练是各类鲁棒学习策略中最有效的一种。然而，最近的工作表明，之前提出的各种对抗训练方法带来的性能提升都不如简单地对对抗训练过程实施早停（early stopping）带来的提升[197]。这一反直觉的现象促使我们深入探究多个最近提出的对抗训练方法，发现对抗训练中使用的不同训练技巧及参数设定对于模型的鲁棒性有很大的影响。受此启发，在本章中我们综合地评估了之前工作中经常忽略的各种训练技巧及参数设定对于模型鲁棒性的影响。我们的结果表明相比于标准学习（standard learning）来说，鲁棒学习（robust learning）对于这些训练设定更加敏感。例如，简单地修改权重衰减系数可以提升至多 7% 的鲁棒准确率。基于本章中的大量实验结果，我们总结出了一套标准的训练设定，这样可以更加公平地比较不同对抗训练方法的性能，且不受潜在训练设定因素的影响。

6.1　本章引言

对抗训练是各类防御方法中最有效的一种[15-16,125]。基于主流的对抗训练框架如 PGD-AT[29]，各种变体方法相继被提出并且进一步提升了模型的鲁棒性。然而，最近的工作[28,117]发现，简单地早停对抗训练过程就可以显著地提升模型在对抗样本上的预测准确率[197]，并且可以获得超越更先进的对抗训练框架（如 TRADES[30]）的性能。然而，经过仔细阅读 TRADES 的实现代码①后我们发现，其实现中也采用了早停策略。此外，文献[197]中报告的 PGD-AT 训练鲁棒性（未使用早停策略的情况）远高于文献[29]中的结果。这些实验结果上的前后不一致性促使我们深入检查这些工作的具体代码实现。我们发现 TRADES 的原始代码中使用的权重衰减系数为 2×10^{-4}，对抗扰动的初始化为高斯标准分布 $\delta_0\sim\mathcal{N}(0,\alpha I)$，并在批量归一化层（batch normalization layer，下面简称为 BN 层）使用 eval 模式；相比之下，文献[197]的代码中②使用的权重衰减系数为 5×10^{-4}，对抗扰动的初始化为均匀分布 $\delta_0\sim\mathcal{U}(-\epsilon,\epsilon)$，并在 BN 层使用 train 模式。在我们的实验中，上述两种训练参数设定在同样使用 TRADES 框架训练时，得到的模型鲁棒准确率可以相差 $\sim5\%$，这相比于方法改进上带来的提升（通常为 1% 到 2% 左右）还要显著。

这一现象启发我们对训练中的参数设定等方面进行综合全面的评估，测试每一项对于模型鲁棒性的影响大小。为此，我们查看了二十余种防御方法的训练代

① https://github.com/yaodongyu/TRADES
② https://github.com/locuslab/robust_overfitting

表 6.1　之前的防御方法所使用的训练参数设定

防御方法	学习率	总训练轮数 (衰减)	批量大小	权重衰减	早停 (训练 / 攻击)	预热 (学习率 / 扰动)
文献[29]	0.1	200 (100, 150)	128	2×10^{-4}	No / No	No / No
文献[203]	0.1	300 (150, 250)	200	5×10^{-4}	No / No	No / Yes
文献[30]	0.1	76 (75)	128	2×10^{-4}	Yes / No	No / No
文献[74]	0.01	120 (60, 100)	128	1×10^{-4}	No / Yes	No / No
文献[204]	0.1	110 (100, 105)	256	2×10^{-4}	No / No	No / Yes
文献[60]	0.1	80 (50, 60)	50	2×10^{-4}	No / No	No / No
文献[65]	0.1	100 (cosine anneal)	256	5×10^{-4}	No / No	No / No
文献[66]	0.2	64 (38, 46, 51)	128	5×10^{-4}	No / No	No / No
文献[72]	0.1	200 (100, 150)	128	2×10^{-4}	No / No	No / No
文献[73]	0.05	105 (79, 90, 100)	256	5×10^{-4}	No / No	No / No
文献[205]	0.1	200 (60, 90)	60	2×10^{-4}	No / No	No / No
文献[206]	0.01	100 (50)	32	1×10^{-4}	No / No	No / No
文献[76]	0∼0.2	30 (one cycle)	128	5×10^{-4}	No / No	Yes / No
文献[197]	0.1	200 (100, 150)	128	5×10^{-4}	Yes / No	No / No
文献[207]	0.3	128 (51, 77, 102)	128	2×10^{-4}	No / No	No / No
文献[34]	0.01	200 (100, 150)	50	1×10^{-4}	No / No	No / No
文献[75]	0.1	120 (60, 90, 110)	128	2×10^{-4}	No / Yes	No / No
文献[208]	0.1	200 (cosine anneal)	256	5×10^{-4}	No / No	Yes / No
文献[209]	0.1	200 (80, 140, 180)	128	5×10^{-4}	No / No	No / No
文献[210]	0.1	200 (100, 150)	128	2×10^{-4}	No / No	No / No
文献[211]	0.1	120 (60, 90)	256	1×10^{-4}	No / No	No / No

码和细节，具体总结于表 6.1中。可以看到，尽管很多防御方法会使用公认的几种常用模型结构（例如 ResNet-18、WRN-28-10、WRN-34-10 等等）来验证其有效性，然而其他训练设定例如训练轮数，批量大小，权重衰减等等却非常的不一致。这些参数设定的不一致会导致不同防御方法之间无法公平地比较。此外，不合适的训练参数设定也会导致原本有效的方法无法得到更加鲁棒的模型。这些都给筛选出真正有效的鲁棒策略带来了阻碍。

在本章中，我们评估了多个训练技巧及参数设定对于对抗训练性能的影响，包括预热、早停、权重衰减系数、批量大小、BN 模式等等。我们的结果表明经常被

表 6.2　批量大小以及初始学习率对鲁棒性的影响

ResNet-18					**WRN-34-10**				
批量	基础学习率		放缩学习率		批量	基础学习率		放缩学习率	
大小	Clean	PGD-10	Clean	PGD-10	大小	Clean	PGD-10	Clean	PGD-10
64	80.08	51.31	82.44	52.48	64	84.20	54.69	85.40	54.86
128	82.52	**53.58**	-	-	128	86.07	**56.60**	-	-
256	83.33	52.20	82.24	52.52	256	86.21	52.90	85.89	56.09
512	83.40	50.69	82.16	53.36	512	86.29	50.17	86.47	55.49

研究者忽略的一些实验设定可以很大程度上影响模型的鲁棒性，从而混淆外界对防御方法本身有效性的评估。基于我们的实验结果，我们在 CIFAR-10 上给出了一套标准的训练参数设定，并且按照此设定训练的 TRADES 模型可以达到更好的效果。尽管这套训练参数设定不能保证泛化到其他数据集，然而我们揭示了鲁棒学习对于训练参数设定的敏感性，从而为之后的鲁棒性评测等工作提供帮助。

6.2　消融实验

本章中，我们的主要研究对象为通常被研究者所忽略的（或者与防御方法本身无关的）超参数或者训练技巧。这些对抗训练中使用的训练设定通常取定为某些默认值，而并未进一步做交叉验证。具体来说，我们考虑 CIFAR-10 数据集[126]上 ℓ_∞-范数攻击，扰动大小为 $\epsilon = 8/255$。我们使用 10 步的 PGD 攻击（缩写为 **PGD-10**）[29]以及 AutoAttack（缩写为 **AA**）[28]作为评测手段。对于 PGD 攻击，我们使用无目标模式且允许攻击者预先知道输入的真实类别；每步的扰动大小为 2/255，并使用五次重启（restarts）来找到能成功攻击模型的样本。对于 AutoAttack，我们使用标准模式①，即执行 AutoPGD 和 FAB 时不使用重启策略。

默认设定：本章中我们采用一些之前工作中验证过有效的训练设定[197]作为默认选择，包括 128 的批量大小；动量批量梯度下降优化器（SGD momentum optimizer），初始学习率 0.1；权重衰减系数 5×10^{-4}；ReLU 激活函数以及不使用标签平滑（label smoothing）；训练中构造对抗样本时 BN 层使用 train 模式。我们对所有的模型训练 110 轮，其中学习率在第 100 以及 105 轮的时候乘以衰减因子 0.1。

① https://github.com/fra31/auto-attack

表 6.3　早停以及预热策略对鲁棒性的影响（ResNet-18 模型结构）

	基线	攻击迭代早停			学习率预热			攻击扰动预热		
		40 / 70	40 / 100	60 / 100	10	15	20	10	15	20
Clean	82.52	86.52	86.56	85.67	82.45	82.64	82.31	82.64	82.75	82.78
PGD-10	53.58	52.65	53.22	52.90	53.43	53.29	53.35	53.65	53.27	53.62
AA	48.51	46.6	46.04	45.96	48.26	48.12	48.37	48.44	48.17	48.48

表 6.4　早停以及预热策略对鲁棒性的影响（WRN-34-10 模型结构）

	基线	攻击迭代早停			学习率预热			攻击扰动预热		
		40 / 70	40 / 100	60 / 100	10	15	20	10	15	20
Clean	86.07	88.29	88.25	88.81	86.35	86.63	86.41	86.66	86.43	86.73
PGD-10	56.60	56.06	55.49	56.41	56.31	56.60	56.28	56.25	56.37	55.65
AA	52.19	50.19	49.44	49.81	51.96	52.13	51.75	51.88	52.06	51.70

6.2.1　早停及预热策略

训练迭代早停：训练迭代轮数的早停策略最早出现在 TRADES[30] 的实现代码中，其学习率在第 75 轮的时候衰减，且训练在第 76 轮停止。之后文献[197]对于训练迭代早停策略进行了详细的研究，并且认为其可以成为通用的提高模型鲁棒性的训练技巧，防止模型过拟合。因此，在本章中我们默认使用训练迭代早停策略。

攻击迭代早停：另一个层面的早停为攻击迭代早停，即在构造训练用的对抗样本时，根据对抗攻击是否成功或者损失函数大小等来决定是否继续迭代攻击。该策略最早被应用在 NeurIPS 2018 对抗比赛防御赛道的第二名方案中[57]。此后一系列工作提出了攻击迭代早停的不同触发机制[74-75]。在表 6.3和表 6.4的左边部分，我们评测了文献[75]中的攻击迭代早停方案。可以看到，攻击迭代早停可以提高模型在干净样本上的正常准确率，且保持相似的 PGD 攻击下的鲁棒准确率。然而当使用更强的 AutoAttack 测试时，模型的鲁棒性出现了明显的下降。

学习率预热：深度学习中学习率预热是最常用的训练技巧之一[1]。在对抗学习领域，FastAT[76] 发现使用循环学习率（cycle learning rate）可以有效提升单步对抗训练的鲁棒性。因此我们测试学习率预热对于 PGD-AT 训练的有效性。在训练过程中，我们分别在前 10、15、20 轮内将学习率从零线性增加到初始学习率。如表 6.3和表 6.4的中间部分所示，学习率预热对于鲁棒模型的性能影响不显著。

攻击扰动预热：在对抗训练过程中，预热策略也可以应用在对抗扰动上。文

表 6.5　标签平滑对鲁棒性的影响

ResNet-18					WRN-34-10				
标签平滑	Clean	PGD-10	AA	RayS	标签平滑	Clean	PGD-10	AA	RayS
0	82.52	53.58	48.51	53.34	0	86.07	56.60	52.19	60.07
0.1	82.69	54.04	48.76	53.71	0.1	85.96	56.88	52.74	59.99
0.2	82.73	54.22	49.20	53.66	0.2	86.09	57.31	**53.00**	60.28
0.3	82.51	54.34	**49.24**	53.59	0.3	85.99	57.55	52.70	61.00
0.4	82.39	54.13	48.83	53.40	0.4	86.19	57.63	52.71	60.64

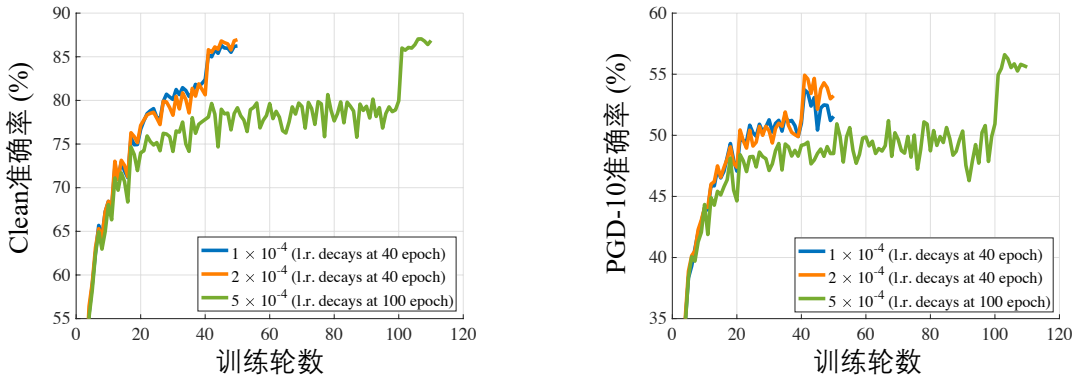


图 6.1　使用小权重衰减系数时对训练进行提前停止的效果

献[203]提出了课程对抗训练（curriculum AT），其在训练过程中逐渐增大对抗扰动，并且在验证集上监测模型是否发生过拟合。此外，文献[204]在前 15 轮内将对抗扰动从零逐渐增加到 8/255。在表 6.3和表 6.4的右边部分，我们测试在前 10、15、20 轮内将对抗扰动从零逐渐增加到 8/255。可以看到攻击扰动预热对于模型鲁棒性的影响同样不显著。

6.2.2　训练超参数设定

批量大小：在大规模数据集例如 ImageNet[38]上，批量大小（batch size）是影响模型性能的重要因素之一[212]。大批量学习可以更快地遍历数据集，然而则需要占用更多的显存。在对抗训练中，文献[213]在 ImageNet 上使用了 4096 的批量大小，并得到了目前 ImageNet 上最鲁棒的模型之一。在 CIFAR-10 数据集上，批量大小通常设定为 128 或者 256（如表 6.1中所列）。在表 6.2中，我们测试使用不同批量大小对于模型鲁棒性的影响。由于批量大小会改变每轮训练中的迭代步数，所以我们也考虑两种学习率设定，一种是基础学习率（即 0.1），另一种是放缩学习率

表 6.6　梯度下降优化器对鲁棒性的影响（ResNet-18 模型结构）

	Mom	Nesterov	Adam	AdamW	SGD-GC	SGD-GCC
Clean	82.52	82.83	83.20	81.68	82.77	82.93
PGD-10	53.58	53.78	48.87	46.58	53.62	53.40
AA	48.51	48.22	44.04	42.39	48.33	48.51

表 6.7　梯度下降优化器对鲁棒性的影响（WRN-34-10 模型结构）

	Mom	Nesterov	Adam	AdamW	SGD-GC	SGD-GCC
Clean	86.07	86.80	81.00	80.72	86.70	86.67
PGD-10	56.60	56.34	52.54	50.32	56.06	56.14
AA	52.19	51.93	46.52	45.79	51.75	51.65

（即以 128 为基准，当批量大小变大 k 倍时，学习率设定为 $0.1 \times k$）。可以看到在 CIFAR-10 上，过大或者过小的批量都会降低模型性能。此外放缩学习率相比于基础学习率可以显著提升模型鲁棒性。

标签平滑：文献[214]提出使用标签平滑（label smoothing，缩写为 LS）来模拟对抗训练。文献[59]发现对集成模型预测使用标签平滑操作可以抑制单模型间对抗样本迁移性。然而，在标准学习范式下使用标签平滑并不能帮助模型抵抗自适应攻击[182]或者更大的攻击步数[215]。基于之前的发现，我们进一步评估标签平滑对于对抗训练的影响。在表 6.5和表 6.9中我们报告了在多种攻击以及不同步数下标签平滑对于鲁棒性的影响。从结果中可以看出，适度的标签平滑（LS 等于 0.2 或者 0.3）可以将鲁棒性提升 $0.5 \sim 1\%$，且不影响干净样本上的正常准确率。这一现象可以视作对抗训练模型的一种置信度校准（confidence calibration）[187]。此外，我们还可以看到过大的标签平滑（LS 大于 0.4）会降低模型的鲁棒性，这一发现与之前的工作相一致[216]。

优化器：大部分的对抗训练算法的实现中选用动量批量梯度下降（SGD with momentum）优化器。其动量因子默认为 0.9，且无阻尼项（dampening）。文献[65]使用 Nesterov 动量项的 SGD，而文献[197]使用循环学习率下的 Adam 优化器。在表 6.6和表 6.7中我们测试了几种常用的优化器对于鲁棒性的影响，包括 AdamW 优化器[217]以及最近提出的 SGD-GC、SGD-GCC 优化器[218]。可以看到，基于 SGD 的优化器及其变体的效果相似，而 Adam、AdamW 优化器会得到的更差的模型性能（特别是在使用线性学习率衰减的情况下）。

表 6.8　激活函数对鲁棒性的影响

	ReLU	Leaky.	ELU ‡	CELU ‡	SELU ‡	GELU	Softplus	Tanh ‡
Clean	82.52	82.11	82.17	81.37	78.88	80.42	**82.80**	80.13
PGD-10	53.58	53.25	52.08	51.37	49.53	52.21	**54.30**	49.12

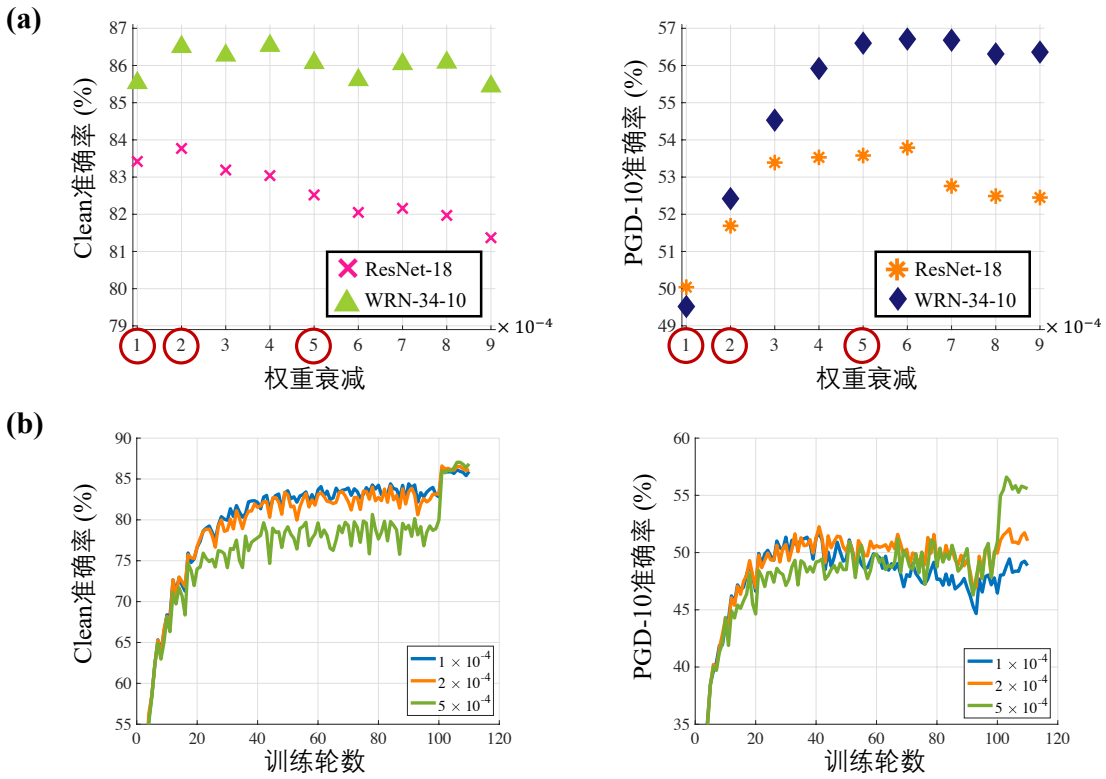


图 6.2　权重衰减对鲁棒性的影响

权重衰减：如表 6.1中所总结的，之前的防御方法所使用的权重衰减（weight decay）系数主要为三个值，即 1×10^{-4}、2×10^{-4} 以及 5×10^{-4}。其中 5×10^{-4} 的权重衰减常用在标准学习范式中，而 2×10^{-4} 被用在 PGD-AT 的论文中[29]。在图 6.2(a)中，我们报告不同权重衰减系数下模型的准确率。可以看到，模型的鲁棒准确率对于权重衰减系数的选择十分敏感。例如在 WRN-34-10 模型上，权重衰减系数选为 1×10^{-4} 和选为 5×10^{-4} 的情况下的鲁棒准确率可以相差 ~ 7%。此外，在图 6.2(b)中，我们画出了模型准确率在训练过程中的变化曲线。从结果中我们注意到，小的权重衰减系数促使模型学习得更快（准确率增长得更快），然而这也导致了模型更容易发生过拟合。在图 6.1中，我们尝试在使用小权重衰减系数时在第 50 轮提前停止训练（学习率在第 40 和 45 轮的时候衰减，而不是默认的第 100 和 105 轮），这样从干净样本准确率上看确实可以防止过拟合，然而却仍然会在鲁棒准确率层

表 6.9　标签平滑在 PGD-1000 以及 SPSA-10000 攻击下的鲁棒性

评测设定			标签平滑				
攻击	重启	步长	0	0.1	0.2	0.3	0.4
PGD-1000 (CE 目标函数)	1	2/255	52.45	52.95	53.08	53.10	**53.14**
	5	2/255	52.41	52.89	53.01	**53.04**	53.03
	10	2/255	52.31	52.85	52.92	**53.02**	52.96
	10	0.5/255	52.63	52.94	**53.33**	53.30	53.25
PGD-1000 (C&W 目标函数)	1	2/255	50.64	50.76	**51.07**	50.96	50.54
	5	2/255	50.58	50.66	**50.93**	50.86	50.44
	10	2/255	50.55	50.59	**50.90**	50.85	50.44
	10	0.5/255	50.63	50.73	51.03	**51.04**	50.52
SPSA-10000	1	1/255	61.69	61.92	**61.93**	61.79	61.53

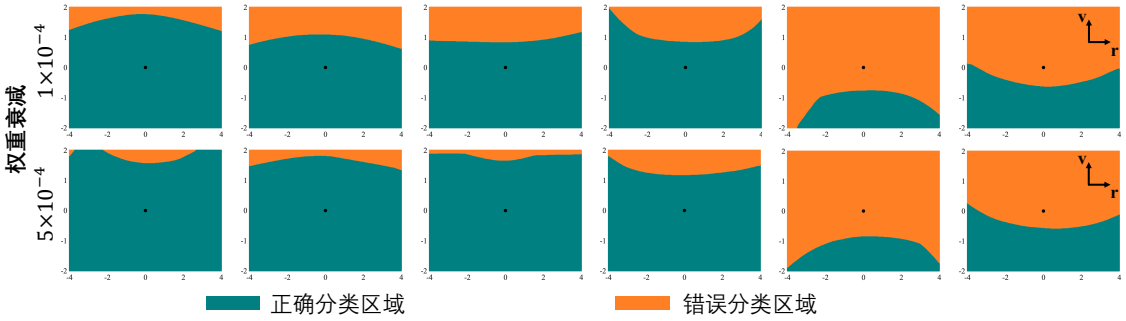


图 6.3　权重衰减对模型决策边界的影响

面发生过拟合。在图 6.3中，我们可视化了多个样本附近的决策边界，其中每一列的两张图对应于同一个输入样本。我们选取竖直方向为对抗扰动方向，水平方向为随机方向。从结果可以看到，合适的权重衰减系数（例如 5×10^{-4}）可以增大样本与决策边界之间的距离，从而提升鲁棒性。然而，从最右边的两列可以看到，对于原本错误分类的样本，权重衰减系数的改变对其影响相对较小。从图 6.4中可以看到，标准学习范式对于权重衰减系数的不同值不太敏感，其最终收敛的模型准确率基本一致。

激活函数：大部分对抗训练工作中都采用默认的 ReLU 激活函数，然而文献[219]从实验上表明光滑的（smooth）激活函数可以帮助模型在 ImageNet 上达到更好的鲁棒准确率。沿用该工作的思路，我们测试 CIFAR-10 上激活函数对于模型鲁棒性的影响。在表 6.8和表 6.11中我们探究了在 PGD-AT 和 TRADES 框架下使

表 6.10　训练中构造对抗样本时使用的 BN 模式对鲁棒性的影响

	BN 模式	模型结构					
		ResNet-18	SENet-18	DenseNet-121	GoogleNet	DPN26	WRN-34-10
Clean	train	82.52	82.20	85.38	83.97	83.67	86.07
	eval	83.48	84.11	86.33	85.26	84.56	87.38
	-	+0.96	+1.91	+0.95	+1.29	+0.89	+1.31
PGD-10	train	53.58	54.01	56.22	53.76	53.88	56.60
	eval	53.64	53.90	56.11	53.77	53.41	56.04
	-	+0.06	-0.11	-0.11	+0.01	-0.47	-0.56
AA	train	48.51	48.72	51.58	48.73	48.50	52.19
	eval	48.75	48.95	51.24	48.83	48.30	51.93
	-	+0.24	+0.23	-0.34	+0.10	-0.20	-0.26

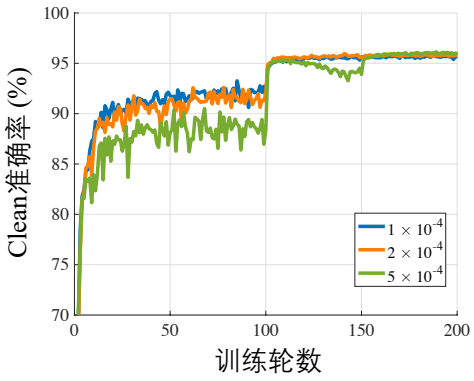


图 6.4　标准学习范式下权重衰减对模型性能的影响

用不同激活函数的效果。可以看到，光滑激活函数（例如 Softplus）确实可以显著提升模型的鲁棒性。然而如表 6.11中所示，这一提升对于小模型（例如 ResNet-18）更加显著，而在大模型（例如 WRN-34-10）上相对不显著。此外，CIFAR-10 上训练的模型在 $\sigma(x) \geqslant 0$ 的情况表现更好，且使用具有负的返回值（negative return values）的激活函数 ELU、LeakyReLU、Tanh 等反而会导致比使用 ReLU 更差的模型性能。

模型结构：文献[220]对标准学习范式下不同模型结构对鲁棒性的影响进行了研究。在对抗学习范式中，业内普遍认为更大的模型表达能力（model capacity）有助于更好的鲁棒性[29]。基于此，一些工作基于自动化机器学习（AutoML）的方法来搜索鲁棒的模型结构。在图 6.5中，我们聚焦于之前研究者手工设计的神经网络模型。我们选取模型参数量可比的模型结构，每个模型结构对应的圆圈大小正比

表 6.11　TRADES 上使用 ReLU 和 Softplus 激活函数的效果比较

攻击者：ℓ_∞-范数，$\epsilon = 8/255$ 扰动大小						
模型结构	权重衰减	BN 模式	激活函数	Clean	PGD-10	AA
ResNet-18	5×10^{-4}	train	ReLU	80.23	53.60	48.96
	5×10^{-4}	train	Softplus	81.26	54.58	50.35
	5×10^{-4}	eval	ReLU	81.45	53.51	49.06
	5×10^{-4}	eval	Softplus	82.37	54.37	50.51

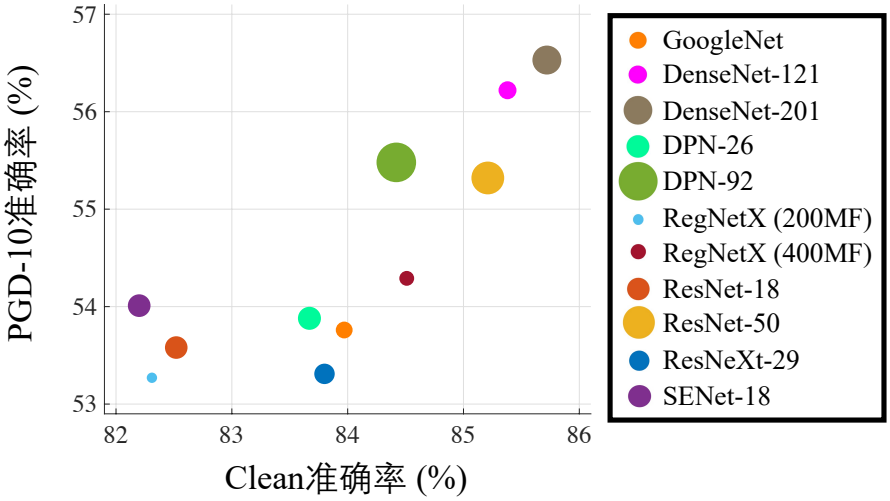


图 6.5　模型结构对于鲁棒性的影响

于其参数量。从结果中我们发现 DenseNet 的结构用于对抗训练可以同时提升干净样本准确率以及鲁棒准确率，且需要更少的存储空间（但是需要更久的计算时间）。这一现象与之前工作中发现残差连接（residual connections）有助于构成鲁棒的模型结构的结论相吻合。此外，文献[51]提出残差连接有助于构造迁移性好的对抗样本，而在对抗训练中，迁移性好的训练对抗样本可以避免模型陷入局部最优解。

BN 层模式：当在训练过程中构造对抗样本时，TRADES 的实现中使用了 eval 模式的 BN 层[30]，而 PGD-AT 中使用了 train 模式的 BN 层[29,197]。由于在构造训练对抗样本时 BN 层的参数不会被更新，所以使用 eval 与 train 模式的区别就在于 BN 层记录的移动平均（moving average）的数据均值与方差。正如文献[221]中所强调的，BN 层记录的移动平均统计量对于测试阶段模型的性能有很大影响。因此在表 6.10中，我们测试了不同模型结构上使用 train 和 eval 模式的 BN 层情况下的鲁棒性。可以看到，使用 eval 模式可以增加干净样本准确率，同时保持鲁棒准确率基本不下降。从原理上来讲，我们同样提倡使用 eval 模式的 BN 层来构造训练对

表 6.12　PGD-AT 框架下参数组合的影响

模型结构	标签平滑	权重衰减	激活函数	BN 模式	准确率（%）		
					Clean	PGD-10	AA
WRN-34-10	0	1×10^{-4}	ReLU	train	85.87	49.45	46.43
	0	2×10^{-4}	ReLU	train	86.14	52.08	48.72
	0	5×10^{-4}	ReLU	train	86.07	56.60	52.19
	0	5×10^{-4}	ReLU	eval	**87.38**	56.04	51.93
	0	5×10^{-4}	Softplus	train	86.60	56.44	52.70
	0.1	5×10^{-4}	Softplus	train	86.42	57.22	**53.01**
	0.1	5×10^{-4}	Softplus	eval	86.34	56.38	52.21
	0.2	5×10^{-4}	Softplus	train	86.10	56.55	52.91
	0.2	5×10^{-4}	Softplus	eval	86.98	56.21	52.10
WRN-34-20	0	1×10^{-4}	ReLU	train	86.21	49.74	47.58
	0	2×10^{-4}	ReLU	train	86.73	51.39	49.03
	0	5×10^{-4}	ReLU	train	86.97	57.57	53.26
	0	5×10^{-4}	ReLU	eval	87.62	57.04	53.14
	0	5×10^{-4}	Softplus	train	85.80	57.84	53.64
	0.1	5×10^{-4}	Softplus	train	85.69	57.86	**53.66**
	0.1	5×10^{-4}	Softplus	eval	**87.86**	57.33	53.23
	0.2	5×10^{-4}	Softplus	train	84.82	57.93	53.39
	0.2	5×10^{-4}	Softplus	eval	87.58	57.19	53.26

抗样本，这样可以避免在多步攻击迭代的过程中记录每一步的均值与方差，从而防止模糊化（blur）BN 层的统计量，这样测试阶段使用的 BN 层统计量依旧满足对抗样本分布。

6.2.3　标准化的训练设定

在上小节中，我们分别评估了各种参数设定对于模型鲁棒性的影响。[①]在本小节中，我们选取一些对鲁棒性影响显著的参数，包括标签平滑、权重衰减、激活函数以及 BN 层模式，并且探究这些参数选择的组合对于模型鲁棒性的影响。从表 6.13中的结果我们看到，每种参数调整在组合后其效果并不是简单地相加。具体来说，标签平滑以及光滑的激活函数对鲁棒性有帮助，然而在组合中并不算显

① 源代码参见https://github.com/P2333/Bag-of-Tricks-for-AT。

表 6.13　TRADES 框架下参数组合的影响

攻击者：ℓ_∞-范数，$\epsilon = 0.031$ 扰动大小						
模型结构	权重衰减	BN 模式	激活函数	Clean	PGD-10	AA
WRN-34-10	2×10^{-4}	train	ReLU	83.86	54.96	51.52
	2×10^{-4}	eval	ReLU	85.17	55.10	51.85
	5×10^{-4}	train	ReLU	84.17	57.34	53.51
	5×10^{-4}	eval	ReLU	**85.34**	58.54	**54.64**
	5×10^{-4}	eval	Softplus	84.66	58.05	54.20
WRN-34-20	5×10^{-4}	eval	ReLU	**86.93**	57.93	**54.42**
	5×10^{-4}	eval	Softplus	85.43	57.94	54.32
攻击者：ℓ_∞-范数，$\epsilon = 8/255$ 扰动大小						
模型结构	权重衰减	BN 模式	激活函数	Clean	PGD-10	AA
WRN-34-10	2×10^{-4}	train	ReLU	84.50	54.60	50.94
	2×10^{-4}	eval	ReLU	85.17	54.58	51.54
	5×10^{-4}	train	ReLU	84.04	57.41	53.83
	5×10^{-4}	eval	ReLU	**85.48**	57.45	53.80
	5×10^{-4}	eval	Softplus	84.24	57.59	**53.88**
WRN-34-20	2×10^{-4}	train	ReLU	84.50	53.86	51.18
	2×10^{-4}	eval	ReLU	85.48	53.21	50.59
	5×10^{-4}	train	ReLU	85.87	57.40	54.22
	5×10^{-4}	eval	ReLU	**86.43**	57.91	**54.39**
	5×10^{-4}	eval	Softplus	85.51	57.50	54.21

著，特别是对于大模型（例如 WRN-34-10 和 WRN-34-20）来说。我们将对应于文献[197]中使用的参数设定用蓝色标注出来。可以看到，文献[197]中报告的 PGD-AT 具有很高的鲁棒性，这主要是因为其使用了较为合理的参数设定（权重衰减 5×10^{-4} 以及训练迭代早停）。基于此，我们下面提供一套 CIFAR-10 上训练 PGD-AT 的建议参数设定。

建议参数设定（CIFAR-10）：批量大小 128；动量批量梯度下降优化器；权重衰减系数 5×10^{-4}；构造训练对抗样本时使用 eval 模式的 BN 层；不需要预热；适度的标签平滑（0.1 ~ 0.2）；使用光滑的激活函数；使用残差连接多的模型结构。

表 6.14　建议参数设定下复现 TRADES 的结果

攻击者：ℓ_∞-范数，$\epsilon = 8/255$ 扰动大小			
方法	模型结构	Clean	AA
TRADES（建议参数设定）	WRN-34-20	86.43	54.39
TRADES（建议参数设定）	WRN-34-10	85.49 ± 0.24	53.94 ± 0.10
文献[31]	WRN-34-20	85.14	53.74
文献[75]	WRN-34-10	84.52	53.51
文献[197]	WRN-34-20	85.34	53.42
文献[204]	WRN-40-8	86.28	52.84
攻击者：ℓ_∞-范数，$\epsilon = 0.031$ 扰动大小			
方法	模型结构	Clean	AA
TRADES（建议参数设定）	WRN-34-10	85.45 ± 0.09	54.28 ± 0.24
文献[208]	WRN-34-10	83.48	53.34
文献[30]	WRN-34-10	84.92	53.08

为了验证上述建议参数设定在不同对抗训练框架下的泛化性,我们在 TRADES 上进行了验证，结果报告于表 6.13中。我们将 TRADES 论文[30]中使用的原始参数设定用蓝色标注出。可以看到，简单地将权重衰减系数从 2×10^{-4} 改为 5×10^{-4}, TRADES 的干净样本准确率提升了 ~ 1%，鲁棒准确率（在 AutoAttack 攻击下）提升了 ~ 4%，使其重新超越了很多基线模型，如表 6.14中所示。这一结果说明了训练参数设定对于合理公平地评估各种防御方法的重要性。

除了最常用 PGD-AT 与 TRADES 这两种对抗训练框架以外，我们还测试了其他对抗训练框架, 包括 FastAT[76] 以及 FreeAT[72]。实现上我们基于 FastAT 提供的代码①。具体来说, 对于 FastAT, 我们使用循环学习率, 其中 $l_{\min} = 0$ 且 $l_{\max} = 0.2$, 对模型训练 15 轮。对于 FreeAT, 我们同样使用循环学习率, 其中 $l_{\min} = 0$ 且 $l_{\max} = 0.04$, 对模型训练 24 轮。在表 6.15中我们报告了实验结果。可以看到我们给出的建议参数设定可以很好地泛化到其他对抗训练框架例如 FastAT 以及 FreeAT 上。

① https://github.com/locuslab/fast_adversarial

表 6.15　建议参数设定下复现 FastAT 以及 FreeAT 的结果

方法	标签平滑	权重衰减	BN 模式	准确率 Clean	PGD-10	AA
FastAT [76]	0	2×10^{-4}	train	82.19	47.47	42.99
	0	5×10^{-4}	train	82.93	48.48	44.06
	0	5×10^{-4}	eval	**84.00**	48.16	43.66
	0.1	5×10^{-4}	train	82.83	**48.76**	**44.50**
FreeAT [72]	0	2×10^{-4}	train	87.42	47.66	44.24
	0	5×10^{-4}	train	88.17	48.90	45.66
	0	5×10^{-4}	eval	**88.26**	48.50	45.49
	0.1	5×10^{-4}	train	88.07	**49.26**	**45.91**

6.3　本章小结

在本章中，我们从实验上系统地验证了鲁棒学习特别是对抗训练对于各种训练参数设定的敏感程度。与之前的相关工作不同，本章中我们主要聚焦于独立于防御方法本身的参数设定，这些参数通常在实现过程中被设置为某些默认值（取决于选用的基线方法及其代码实现），而未被充分地重视。我们的实验结果表明，这些我们通常忽略的参数设定（例如权重衰减系数、BN 层模式等等）会很大程度上影响模型的鲁棒性，这与我们在标准学习框架下的普遍认知是不同的。本章的工作为之后对各类防御方法进行公平的鲁棒性评测提供了帮助，并且也从实验现象出发提供了很多理论工作的启发性思路（例如残差连接对于对抗训练的影响等等）。

第 7 章　总结与展望

7.1　本文总结

为了提高深度学习模型在对抗环境中的鲁棒性，本文系统地就对抗防御算法进行了研究与开发，取得的主要创新性成果如下:

第 2 章提出最大化马氏距离学习，通过提高现有训练数据的使用效率，促进了样本间信息的传输，提高了模型的鲁棒性。此外，该方法还具有收敛速度快、不受小批量情况影响等良好的性质。

第 3 章提出集成模型的多样性增强鲁棒学习，通过鼓励单模型间非最大预测的多样性来抑制对抗样本在单模型间迁移，从而提升集成模型的整体鲁棒性。

第 4 章提出基于反交叉熵训练的对抗样本检测方法，通过将传统的交叉熵训练换成反交叉熵训练过程，模型可以将正常的干净样本输入映射到特征空间中的低维流形上。这样当输入样本为对抗样本时，检测指标可以更加灵敏地将其检测出来。

第 5 章提出互耦的双检测指标方法，理论上证明了模型预测置信度与修正置信度可以构成一对互耦的检测指标，在特定条件下可以完全区分出任何正确分类样本与错误分类样本，因此可以抵御潜在的自适应攻击。

第 6 章基于对抗训练框架系统地进行了大量的对比消融实验，评估多种训练参数设定对于对抗训练效果的影响程度，发现鲁棒学习对于一些基础参数设定十分敏感。基于此，第 6 章提供了一套建议的标准参数设定，帮助后续的防御方法快速达到理想的训练效果。

7.2　未来工作展望

本文系统地研究了深度学习的对抗鲁棒性，特别是在对抗防御方面，提出了多种高效的算法，一定程度上缓解了深度学习模型鲁棒性不足的问题，取得了阶段性成果。但是深度学习的对抗鲁棒性方面还有很多极具挑战性的问题待解决，或者很有潜力的方向值得探索，具体可以概括为以下三方面:

缩小鲁棒准确率与正常准确率的差距：尽管在过去几年的研究中，模型的对抗鲁棒性得到了实质性的提升，然而鲁棒准确率与正常准确率之间仍然有巨大的差距。例如在 CIFAR-10 数据集上，在限制攻击者扰动为 ℓ_∞-范数下 8/255 的情况，鲁棒性最好的模型在对抗样本上的预测准确率都没有超过 67%。注意到，这一结

果还是在使用了大量额外训练数据（8 千万额外图片[173]，相比之下 CIFAR-10 原本仅有 5 万张图片）以及大模型（WRN-70-16[174]）的情况。相比之下，CIFAR-10 数据集上最先进的模型正常准确率可以达到 99% 以上[222]。因此，如何缩小鲁棒准确率与正常准确率的差距是未来对抗学习领域重要的课题之一。

探究鲁棒模型展现出的性质：之前的工作发现，对抗鲁棒的模型具有很多性质。例如，对抗训练或者随机平滑得到的鲁棒模型的输入梯度具有明显的语义特征[39,138]，且具有更好的可解释性[41]。此外，对抗鲁棒的特征在迁移学习任务上效果更好[40]。相比于标准学习，鲁棒学习过程更容易产生过拟合[197]。这些鲁棒模型展现出的性质目前还仅仅停留在实验现象阶段，尚未有严格的理论来解释这些现象。因此，从理论上解释这些鲁棒模型展现出的性质对于我们理解深度学习模型的内部机理具有重要意义。

对抗学习技术的正面应用：除了安全性以外，对抗攻防技术还有很多潜在的正面应用。例如对抗攻击可以用来保护人脸隐私[43]、防止网络传输的个人照片等被未授权的第三方使用；对抗训练可以用来提升模型在 ImageNet 上的正常准确率[44]，且对抗训练的变体还可以用来进行半监督学习[42]。沿着这一思路，探索出对抗学习技术更多的正面应用场景也是未来重要的研究课题之一。

参考文献

[1] Goodfellow I, Bengio Y, Courville A. Deep learning[M/OL]. MIT Press, 2016. http://www.deeplearningbook.org.

[2] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems (NIPS), 2012.

[3] Sun Y, Wang X, Tang X. Deeply learned face representations are sparse, selective, and robust [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015: 2892-2900.

[4] Wen Y, Zhang K, Li Z, et al. A discriminative feature learning approach for deep face recognition [C]//European Conference on Computer Vision (ECCV). Springer, 2016: 499-515.

[5] Schroff F, Kalenichenko D, Philbin J. Facenet: A unified embedding for face recognition and clustering[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015: 815-823.

[6] Koehn P. Statistical machine translation[M]. Cambridge University Press, 2009.

[7] Wu Y, Schuster M, Chen Z, et al. Google's neural machine translation system: Bridging the gap between human and machine translation[J]. arXiv preprint arXiv:1609.08144, 2016.

[8] Lü L, Medo M, Yeung C H, et al. Recommender systems[J]. Physics Reports, 2012, 519(1): 1-49.

[9] Ricci F, Rokach L, Shapira B. Introduction to recommender systems handbook[M]// Recommender systems handbook. Springer, 2011: 1-35.

[10] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the kitti vision benchmark suite[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2012: 3354-3361.

[11] Shen D, Wu G, Suk H I. Deep learning in medical image analysis[J]. Annual Review of Biomedical Engineering, 2017, 19: 221-248.

[12] Silver D, Hubert T, Schrittwieser J, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play[J]. Science, 2018, 362(6419): 1140-1144.

[13] Jumper J, Evans R, Pritzel A, et al. Highly accurate protein structure prediction with alphafold [J]. Nature, 2021, 596(7873): 583-589.

[14] Moult J, Pedersen J T, Judson R, et al. A large-scale experiment to assess protein structure prediction methods[J]. Proteins: Structure, Function, and Bioinformatics, 1995, 23(3): ii-iv.

[15] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing properties of neural networks[C]// International Conference on Learning Representations (ICLR). 2014.

[16] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[C]// International Conference on Learning Representations (ICLR). 2015.

[17] Kurakin A, Goodfellow I, Bengio S. Adversarial examples in the physical world[C]//The International Conference on Learning Representations (ICLR) Workshops. 2017.

[18] Eykholt K, Evtimov I, Fernandes E, et al. Robust physical-world attacks on deep learning visual classification[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 1625-1634.

[19] Jin D, Jin Z, Zhou J T, et al. Is bert really robust? a strong baseline for natural language attack on text classification and entailment[C]//Proceedings of the AAAI conference on artificial intelligence: volume 34. 2020: 8018-8025.

[20] Dai H, Li H, Tian T, et al. Adversarial attack on graph structured data[C]//International Conference on Machine Learning (ICML). 2018.

[21] Lin Y C, Hong Z W, Liao Y H, et al. Tactics of adversarial attack on deep reinforcement learning agents[J]. arXiv preprint arXiv:1703.06748, 2017.

[22] Carlini N, Wagner D. Audio adversarial examples: Targeted attacks on speech-to-text[C]//2018 IEEE Security and Privacy Workshops (SPW). IEEE, 2018: 1-7.

[23] Tu J, Ren M, Manivasagam S, et al. Physically realizable adversarial examples for lidar object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 13716-13725.

[24] Lang I, Kotlicki U, Avidan S. Geometric adversarial attacks and defenses on 3d point clouds [C]//International Conference on 3D Vision (3DV). IEEE, 2021: 1196-1205.

[25] Anand M, Kayal P, Singh M. On adversarial robustness of synthetic code generation[J]. arXiv preprint arXiv:2106.11629, 2021.

[26] Cao Y, Chen X, Yao L, et al. Adversarial attacks and detection on reinforcement learning-based interactive recommender systems[C]//ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 1669-1672.

[27] Dong Y, Liao F, Pang T, et al. Boosting adversarial attacks with momentum[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.

[28] Croce F, Hein M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks[C]//International Conference on Machine Learning (ICML). 2020.

[29] Madry A, Makelov A, Schmidt L, et al. Towards deep learning models resistant to adversarial attacks[C]//International Conference on Learning Representations (ICLR). 2018.

[30] Zhang H, Yu Y, Jiao J, et al. Theoretically principled trade-off between robustness and accuracy [C]//International Conference on Machine Learning (ICML). 2019.

[31] Pang T, Yang X, Dong Y, et al. Boosting adversarial training with hypersphere embedding[C]// Advances in Neural Information Processing Systems (NeurIPS). 2020.

[32] Pang T, Du C, Zhu J. Max-mahalanobis linear discriminant analysis networks[C]//International Conference on Machine Learning (ICML). 2018.

[33] Pang T, Du C, Dong Y, et al. Towards robust detection of adversarial examples[C]//Advances in Neural Information Processing Systems (NeurIPS). 2018: 4579-4589.

[34] Pang T, Xu K, Dong Y, et al. Rethinking softmax cross-entropy loss for adversarial robustness [C]//International Conference on Learning Representations (ICLR). 2020.

[35] Pang T, Yang X, Dong Y, et al. Bag of tricks for adversarial training[C]//International Conference on Learning Representations (ICLR). 2021.

[36] Croce F, Andriushchenko M, Sehwag V, et al. Robustbench: a standardized adversarial robustness benchmark[J]. arXiv preprint arXiv:2010.09670, 2020.

[37] Dong Y, Fu Q A, Yang X, et al. Benchmarking adversarial robustness on image classification [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 321-331.

[38] Deng J, Dong W, Socher R, et al. Imagenet: A large-scale hierarchical image database[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2009.

[39] Ilyas A, Santurkar S, Tsipras D, et al. Adversarial examples are not bugs, they are features[C]// Advances in Neural Information Processing Systems (NeurIPS). 2019.

[40] Salman H, Ilyas A, Engstrom L, et al. Do adversarially robust imagenet models transfer better? [J]. Advances in Neural Information Processing Systems (NeurIPS), 2020, 33: 3533-3545.

[41] Dong Y, Su H, Zhu J, et al. Towards interpretable deep neural networks by leveraging adversarial examples[J]. arXiv preprint arXiv:1708.05493, 2017.

[42] Miyato T, Maeda S i, Koyama M, et al. Virtual adversarial training: a regularization method for supervised and semi-supervised learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2018, 41(8): 1979-1993.

[43] Yang X, Dong Y, Pang T, et al. Towards face encryption by generating adversarial identity masks [C]//IEEE International Conference on Computer Vision (ICCV). 2021: 3897-3907.

[44] Xie C, Tan M, Gong B, et al. Adversarial examples improve image recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

[45] Pang T, Lin M, Yang X, et al. Robustness and accuracy could be reconcilable by (proper) definition[J]. arXiv preprint arXiv:2202.10103, 2022.

[46] Carlini N, Athalye A, Papernot N, et al. On evaluating adversarial robustness[J]. arXiv preprint arXiv:1902.06705, 2019.

[47] Carlini N, Wagner D. Adversarial examples are not easily detected: Bypassing ten detection methods[C]//ACM Workshop on Artificial Intelligence and Security (AISec). 2017.

[48] Athalye A, Carlini N, Wagner D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples[C]//International Conference on Machine Learning (ICML). 2018.

[49] Athalye A, Carlini N. On the robustness of the cvpr 2018 white-box adversarial example defenses [J]. arXiv preprint arXiv:1804.03286, 2018.

[50] Xie C, Zhang Z, Zhou Y, et al. Improving transferability of adversarial examples with input diversity[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 2730-2739.

[51] Wu D, Wang Y, Xia S T, et al. Skip connections matter: On the transferability of adversarial examples generated with resnets[C]//International Conference on Learning Representations (ICLR). 2020.

[52] Chen P Y, Zhang H, Sharma Y, et al. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models[C]//ACM Workshop on Artificial Intelligence and Security (AISec). ACM, 2017.

[53] Ilyas A, Engstrom L, Athalye A, et al. Black-box adversarial attacks with limited queries and information[C]//International Conference on Machine Learning (ICML). 2018.

[54] Wierstra D, Schaul T, Glasmachers T, et al. Natural evolution strategies[J]. Journal of Machine Learning Research (JMLR), 2014, 15(1): 949-980.

[55] Brendel W, Rauber J, Bethge M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models[J]. arXiv preprint arXiv:1712.04248, 2017.

[56] Kurakin A, Goodfellow I, Bengio S, et al. Adversarial attacks and defences competition[J]. arXiv preprint arXiv:1804.00097, 2018.

[57] Brendel W, Rauber J, Kurakin A, et al. Adversarial vision challenge[M]//The NeurIPS'18 Competition. Springer, 2020: 129-153.

[58] Tramèr F, Kurakin A, Papernot N, et al. Ensemble adversarial training: Attacks and defenses [C]//International Conference on Learning Representations (ICLR). 2018.

[59] Pang T, Xu K, Du C, et al. Improving adversarial robustness via promoting ensemble diversity [C]//International Conference on Machine Learning (ICML). 2019.

[60] Mao C, Zhong Z, Yang J, et al. Metric learning for adversarial robustness[C]//Advances in Neural Information Processing Systems (NeurIPS). 2019: 478-489.

[61] Li P, Yi J, Zhou B, et al. Improving the robustness of deep neural networks via adversarial training with triplet loss[C]//International Joint Conference on Artificial Intelligence (IJCAI). 2019.

[62] Jiang H, Chen Z, Shi Y, et al. Learning to defense by learning to attack[J]. arXiv preprint arXiv:1811.01213, 2018.

[63] Wang H, Yu C N. A direct approach to robust deep learning using adversarial networks[C]// International Conference on Learning Representations (ICLR). 2019.

[64] Deng Z, Dong Y, Pang T, et al. Adversarial distributional training for robust deep learning[C]// Advances in Neural Information Processing Systems (NeurIPS). 2020.

[65] Carmon Y, Raghunathan A, Schmidt L, et al. Unlabeled data improves adversarial robustness [C]//Advances in Neural Information Processing Systems (NeurIPS). 2019.

[66] Alayrac J B, Uesato J, Huang P S, et al. Are labels required for improving adversarial robustness? [C]//Advances in Neural Information Processing Systems (NeurIPS). 2019: 12192-12202.

[67] Zhai R, Cai T, He D, et al. Adversarially robust generalization just requires more unlabeled data [J]. arXiv preprint arXiv:1906.00555, 2019.

[68] Hendrycks D, Lee K, Mazeika M. Using pre-training can improve model robustness and uncertainty[C]//International Conference on Machine Learning (ICML). 2019.

[69] Chen K, Chen Y, Zhou H, et al. Self-supervised adversarial training[C]//IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 2218-2222.

[70] Chen T, Liu S, Chang S, et al. Adversarial robustness: From self-supervised pre-training to fine-tuning[C]//Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 699-708.

[71] Naseer M, Khan S, Hayat M, et al. A self-supervised approach for adversarial robustness[C]// Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 262-271.

[72] Shafahi A, Najibi M, Ghiasi A, et al. Adversarial training for free![C]//Advances in Neural Information Processing Systems (NeurIPS). 2019.

[73] Zhang D, Zhang T, Lu Y, et al. You only propagate once: Accelerating adversarial training via maximal principle[C]//Advances in Neural Information Processing Systems (NeurIPS). 2019.

[74] Wang Y, Ma X, Bailey J, et al. On the convergence and robustness of adversarial training[C]// International Conference on Machine Learning (ICML). 2019: 6586-6595.

[75] Zhang J, Xu X, Han B, et al. Attacks which do not kill training make adversarial learning stronger[C]//International Conference on Machine Learning (ICML). 2020.

[76] Wong E, Rice L, Kolter J Z. Fast is better than free: Revisiting adversarial training[C]// International Conference on Learning Representations (ICLR). 2020.

[77] Liu G, Khalil I, Khreishah A. Using single-step adversarial training to defend iterative adversarial examples[J]. arXiv preprint arXiv:2002.09632, 2020.

[78] Vivek B S, Venkatesh Babu R. Single-step adversarial training with dropout scheduling[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

[79] Andriushchenko M, Flammarion N. Understanding and improving fast adversarial training[C]// Advances in Neural Information Processing Systems (NeurIPS). 2020.

[80] Li B, Wang S, Jana S, et al. Towards understanding fast adversarial training[J]. arXiv preprint arXiv:2006.03089, 2020.

[81] Floudas C A, Lin X. Mixed integer linear programming in process scheduling: Modeling, algorithms, and applications[J]. Annals of Operations Research, 2005, 139(1): 131-162.

[82] Wong E, Kolter Z. Provable defenses against adversarial examples via the convex outer adversarial polytope[C]//International Conference on Machine Learning (ICML). 2018: 5283-5292.

[83] Wong E, Schmidt F, Metzen J H, et al. Scaling provable adversarial defenses[C]//Advances in Neural Information Processing Systems (NeurIPS). 2018: 8400-8409.

[84] Cohen J M, Rosenfeld E, Kolter J Z. Certified adversarial robustness via randomized smoothing [C]//International Conference on Machine Learning (ICML). 2019.

[85] Crecchi F, Melis M, Sotgiu A, et al. Fader: Fast adversarial example rejection[J]. arXiv preprint arXiv:2010.09119, 2020.

[86] Grosse K, Manoharan P, Papernot N, et al. On the (statistical) detection of adversarial examples [J]. arXiv preprint arXiv:1702.06280, 2017.

[87] Liu X, Li Y, Wu C, et al. Adv-bnn: Improved adversarial defense through robust bayesian neural network[C]//International Conference on Learning Representations (ICLR). 2019.

[88] Lu J, Issaranon T, Forsyth D. Safetynet: Detecting and rejecting adversarial examples robustly [C]//International Conference on Computer Vision (ICCV). 2017: 446-454.

[89] Metzen J H, Genewein T, Fischer V, et al. On detecting adversarial perturbations[C]// International Conference on Learning Representations (ICLR). 2017.

[90] Roth K, Kilcher Y, Hofmann T. The odds are odd: A statistical test for detecting adversarial examples[C]//International Conference on Machine Learning (ICML). 2019.

[91] Zhang C, Ye Z, Wang Y, et al. Detecting adversarial perturbations with saliency[C]//2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP). IEEE, 2018: 271-275.

[92] Gondara L. Detecting adversarial samples using density ratio estimates[J]. arXiv preprint arXiv:1705.02224, 2017.

[93] Feinman R, Curtin R R, Shintre S, et al. Detecting adversarial samples from artifacts[J]. arXiv preprint arXiv:1703.00410, 2017.

[94] Xu W, Evans D, Qi Y. Feature squeezing: Detecting adversarial examples in deep neural networks[J]. arXiv preprint arXiv:1704.01155, 2017.

[95] Sheikholeslami F, Jain S, Giannakis G B. Minimum uncertainty based detection of adversaries in deep neural networks[J]. arXiv preprint arXiv:1904.02841, 2019.

[96] Smith L, Gal Y. Understanding measures of uncertainty for adversarial example detection[C]// Conference on Uncertainty in Artificial Intelligence (UAI). 2018.

[97] Zhao C, Fletcher P T, Yu M, et al. The adversarial attack and detection under the fisher information metric[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 33. 2019: 5869-5876.

[98] Ma X, Li B, Wang Y, et al. Characterizing adversarial subspaces using local intrinsic dimensionality[C]//International Conference on Learning Representations (ICLR). 2018.

[99] Ma S, Liu Y. Nic: Detecting adversarial samples with neural network invariant checking[C]// Proceedings of the 26th Network and Distributed System Security Symposium (NDSS 2019). 2019.

[100] Tao G, Ma S, Liu Y, et al. Attacks meet interpretability: Attribute-steered detection of adversarial samples[C]//Advances in Neural Information Processing Systems (NeurIPS). 2018.

[101] Yang P, Chen J, Hsieh C J, et al. Ml-loo: Detecting adversarial examples with feature attribution. [C]//Thirty-First AAAI Conference on Artificial Intelligence (AAAI). 2020.

[102] Carrara F, Becarelli R, Caldelli R, et al. Adversarial examples detection in features distance spaces[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[103] Cohen G, Sapiro G, Giryes R. Detecting adversarial samples using influence functions and nearest neighbors[C]//Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

[104] Sperl P, Kao C Y, Chen P, et al. Dla: Dense-layer-analysis for adversarial example detection [C]//IEEE European Symposium on Security and Privacy (EuroS&P). 2020.

[105] Ahuja N A, Ndiour I, Kalyanpur T, et al. Probabilistic modeling of deep features for out-of-distribution and adversarial detection[J]. arXiv preprint arXiv:1909.11786, 2019.

[106] Lee K, Lee K, Lee H, et al. A simple unified framework for detecting out-of-distribution samples and adversarial attacks[C]//Advances in Neural Information Processing Systems (NeurIPS). 2018.

[107] Ma C, Wu B, Xu S, et al. Effective and robust detection of adversarial examples via benford-fourier coefficients[J]. arXiv preprint arXiv:2005.05552, 2020.

[108] Anirudh R, Thiagarajan J J, Kailkhura B, et al. Mimicgan: Robust projection onto image manifolds with corruption mimicking[J]. International Journal of Computer Vision (IJCV), 2020: 1-19.

[109] Dubey A, Maaten L v d, Yalniz Z, et al. Defense against adversarial images using web-scale nearest-neighbor search[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 8767-8776.

[110] Samangouei P, Kabkab M, Chellappa R. Defense-gan: Protecting classifiers against adversarial attacks using generative models[C]//International Conference on Learning Representations (ICLR). 2018.

[111] Croce F, Gowal S, Brunner T, et al. Evaluating the adversarial robustness of adaptive test-time defenses[J]. arXiv preprint arXiv:2202.13711, 2022.

[112] Alfarra M, Pérez J C, Thabet A, et al. Combating adversaries with anti-adversaries[J]. arXiv preprint arXiv:2103.14347, 2021.

[113] Mao C, Chiquier M, Wang H, et al. Adversarial attacks are reversible with natural supervision [C]//IEEE International Conference on Computer Vision (ICCV). 2021: 661-671.

[114] Chen Z, Li Q, Zhang Z. Towards robust neural networks via close-loop control[J]. arXiv preprint arXiv:2102.01862, 2021.

[115] Wang D, Ju A, Shelhamer E, et al. Fighting gradients with gradients: Dynamic defenses against adversarial attacks[J]. arXiv preprint arXiv:2105.08714, 2021.

[116] Dong Y, Fu Q A, Yang X, et al. Benchmarking adversarial robustness[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

[117] Chen J, Gu Q. Rays: A ray searching method for hard-label adversarial attack[C]//International Conference on Knowledge Discovery & Data Mining (KDD). 2020.

[118] Mu N, Gilmer J. Mnist-c: A robustness benchmark for computer vision[J]. arXiv preprint arXiv:1906.02337, 2019.

[119] Hendrycks D, Dietterich T. Benchmarking neural network robustness to common corruptions and perturbations[C]//International Conference on Learning Representations (ICLR). 2019.

[120] Engstrom L, Tran B, Tsipras D, et al. Exploring the landscape of spatial robustness[C]// International Conference on Machine Learning (ICML). 2019.

[121] Tramèr F, Boneh D. Adversarial training and robustness for multiple perturbations[C]// Advances in Neural Information Processing Systems (NeurIPS). 2019: 5858-5868.

[122] Schmidt L, Santurkar S, Tsipras D, et al. Adversarially robust generalization requires more data [C]//Advances in Neural Information Processing Systems (NeurIPS). 2018: 5019-5031.

[123] Gowal S, Qin C, Uesato J, et al. Uncovering the limits of adversarial training against norm-bounded adversarial examples[J]. arXiv preprint arXiv:2010.03593, 2020.

[124] Pang T, Zhang H, He D, et al. Two coupled rejection metrics can tell adversarial examples apart [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2022.

[125] Biggio B, Corona I, Maiorca D, et al. Evasion attacks against machine learning at test time [C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, 2013: 387-402.

[126] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[R]. Technical report, University of Toronto, 2009.

[127] Rebuffi S A, Gowal S, Calian D A, et al. Fixing data augmentation to improve adversarial robustness[J]. arXiv preprint arXiv:2103.01946, 2021.

[128] Gowal S, Rebuffi S A, Wiles O, et al. Improving robustness using generated data[C]//Advances in Neural Information Processing Systems (NeurIPS). 2021.

[129] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[C]//Advances in Neural Information Processing Systems (NeurIPS). 2020.

[130] Efron B. The efficiency of logistic regression compared to normal discriminant analysis[J]. Journal of the American Statistical Association, 1975, 70(352): 892-898.

[131] Chen T, Kornblith S, Norouzi M, et al. A simple framework for contrastive learning of visual representations[C]//International Conference on Machine Learning (ICML). 2020.

[132] Sun Y, Chen Y, Wang X, et al. Deep learning face representation by joint identification-verification[C]//Advances in Neural Information Processing Systems (NIPS). 2014: 1988-1996.

[133] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[134] Friedman J, Hastie T, Tibshirani R. The elements of statistical learning[M]. Springer series in statistics New York, 2001.

[135] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in Neural Information Processing Systems (NIPS). 2014: 2672-2680.

[136] Moosavi-Dezfooli S M, Fawzi A, Frossard P. Deepfool: a simple and accurate method to fool deep neural networks[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 2574-2582.

[137] Santurkar S, Ilyas A, Tsipras D, et al. Image synthesis with a single (robust) classifier[C]// Advances in Neural Information Processing Systems (NeurIPS). 2019.

[138] Kaur S, Cohen J, Lipton Z C. Are perceptually-aligned gradients a general property of robust classifiers?[J]. arXiv preprint arXiv:1910.08640, 2019.

[139] Wan W, Zhong Y, Li T, et al. Rethinking feature distribution for loss functions in image classification[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 9117-9126.

[140] Nielsen F, Sun K. Guaranteed bounds on the kullback–leibler divergence of univariate mixtures [J]. IEEE Signal Processing Letters, 2016, 23(11): 1543-1546.

[141] Loskot P, Beaulieu N C. On monotonicity of the hypersphere volume and area[J]. Journal of Geometry, 2007.

[142] Wang F, Xiang X, Cheng J, et al. Normface: l 2 hypersphere embedding for face verification [C]//ACM International Conference on Multimedia (ACM MM). ACM, 2017: 1041-1049.

[143] Deng J, Guo J, Xue N, et al. Arcface: Additive angular margin loss for deep face recognition [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.

[144] He K, Zhang X, Ren S, et al. Identity mappings in deep residual networks[C]//European Conference on Computer Vision (ECCV). Springer, 2016: 630-645.

[145] Qian N. On the momentum term in gradient descent learning algorithms[J]. Neural networks, 1999, 12(1): 145-151.

[146] Carlini N, Wagner D. Towards evaluating the robustness of neural networks[C]//IEEE Symposium on Security and Privacy (S&P). 2017.

[147] Uesato J, O'Donoghue B, Oord A v d, et al. Adversarial risk and the dangers of evaluating against weak attacks[C]//International Conference on Machine Learning (ICML). 2018.

[148] Papernot N, McDaniel P, Jha S, et al. The limitations of deep learning in adversarial settings [C]//IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, 2016: 372-387.

[149] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale[C]//International Conference on Learning Representations (ICLR). 2017.

[150] Kurakin A, Boneh D, Tramèr F, et al. Ensemble adversarial training: Attacks and defenses[J]. International Conference on Learning Representations (ICLR), 2018.

[151] Kannan H, Kurakin A, Goodfellow I. Adversarial logit pairing[J]. arXiv preprint arXiv:1803.06373, 2018.

[152] Liao F, Liang M, Dong Y, et al. Defense against adversarial attacks using high-level representation guided denoiser[J]. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[153] Papernot N, McDaniel P, Goodfellow I, et al. Practical black-box attacks against deep learning systems using adversarial examples[J]. arXiv preprint arXiv:1602.02697, 2016.

[154] Dauphin Y N, Pascanu R, Gulcehre C, et al. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization[C]//Advances in Neural Information Processing Systems (NIPS). 2014: 2933-2941.

[155] Li Y, Yosinski J, Clune J, et al. Convergent learning: Do different neural networks learn the same representations?[C]//International Conference on Learning Representations (ICLR). 2016.

[156] Liu Y, Yao X. Ensemble learning via negative correlation[J]. Neural Networks, 1999, 12(10): 1399-1404.

[157] Liu Y, Yao X. Simultaneous training of negatively correlated neural networks in an ensemble [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 1999, 29(6): 716-725.

[158] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors [J]. Cognitive modeling, 1988, 5(3): 1.

[159] Russakovsky O, Deng J, Su H, et al. Imagenet large scale visual recognition challenge[J]. International Journal of Computer Vision (IJCV), 2015, 115(3): 211-252.

[160] Kulesza A, Taskar B, et al. Determinantal point processes for machine learning[J]. Foundations and Trends® in Machine Learning, 2012, 5(2–3): 123-286.

[161] Kwok J T, Adams R P. Priors for diversity in generative latent variable models[C]//Advances in Neural Information Processing Systems (NIPS). 2012: 2996-3004.

[162] Mariet Z, Sra S. Diversity networks[J]. International Conference on Learning Representations (ICLR), 2016.

[163] Islam M M, Yao X, Murase K. A constructive algorithm for training cooperative neural network ensembles[J]. IEEE Transactions on Neural Networks, 2003, 14(4): 820-834.

[164] Krogh A, Vedelsby J. Neural network ensembles, cross validation, and active learning[C]// Advances in Neural Information Processing Systems (NIPS). 1995: 231-238.

[165] Kuncheva L I, Whitaker C J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy[J]. Machine Learning, 2003, 51(2): 181-207.

[166] Bernstein D S. Matrix mathematics: Theory, facts, and formulas with application to linear systems theory: volume 41[M]. Princeton university press Princeton, 2005.

[167] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators[J]. Neural networks, 1989, 2(5): 359-366.

[168] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 770-778.

[169] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[170] Maaten L v d, Hinton G. Visualizing data using t-sne[J]. Journal of Machine Learning Research (JMLR), 2008, 9(Nov): 2579-2605.

[171] Papernot N, McDaniel P, Goodfellow I, et al. Practical black-box attacks against deep learning systems using adversarial examples[J]. arXiv preprint arXiv:1602.02697, 2016.

[172] Chen P Y, Sharma Y, Zhang H, et al. Ead: elastic-net attacks to deep neural networks via adversarial examples[C]//AAAI Conference on Artificial Intelligence (AAAI). 2018.

[173] Torralba A, Fergus R, Freeman W T. 80 million tiny images: A large data set for nonparametric object and scene recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2008, 30(11): 1958-1970.

[174] Zagoruyko S, Komodakis N. Wide residual networks[C]//The British Machine Vision Conference (BMVC). 2016.

[175] Gong Z, Wang W, Ku W S. Adversarial and clean data are not twins[J]. arXiv preprint arXiv:1704.04960, 2017.

[176] Bhagoji A N, Cullina D, Mittal P. Dimensionality reduction as a defense against evasion attacks on machine learning classifiers[J]. arXiv preprint arXiv:1704.02654, 2017.

[177] Li X, Li F. Adversarial examples detection in deep networks with convolutional filter statistics [J]. arXiv preprint arXiv:1612.07767, 2016.

[178] Nguyen A, Yosinski J, Clune J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015: 427-436.

[179] Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the inception architecture for computer vision [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 2818-2826.

[180] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 770-778.

[181] Maaten L v d, Hinton G. Visualizing data using t-sne[J]. Journal of Machine Learning Research (JMLR), 2008, 9(Nov): 2579-2605.

[182] Tramer F, Carlini N, Brendel W, et al. On adaptive attacks to adversarial example defenses[C]// Advances in Neural Information Processing Systems (NeurIPS). 2020.

[183] Sehwag V, Mahloujifar S, Handina T, et al. Improving adversarial robustness using proxy distributions[J]. arXiv preprint arXiv:2104.09425, 2021.

[184] Wu D, Xia S T, Wang Y. Adversarial weight perturbation helps robust generalization[J]. Advances in Neural Information Processing Systems (NeurIPS), 2020, 33.

[185] Kato M, Cui Z, Fukuhara Y. Atro: Adversarial training with a rejection option[J]. arXiv preprint arXiv:2010.12905, 2020.

[186] Laidlaw C, Feizi S. Playing it safe: Adversarial robustness with an abstain option[J]. arXiv preprint arXiv:1911.11253, 2019.

[187] Stutz D, Hein M, Schiele B. Confidence-calibrated adversarial training: Generalizing to unseen attacks[C]//International Conference on Machine Learning (ICML). 2020.

[188] Tramer F. Detecting adversarial examples is (nearly) as hard as classifying them[C]//ICML 2021 Workshop on Adversarial Machine Learning. 2021.

[189] Yang Y Y, Rashtchian C, Zhang H, et al. A closer look at accuracy vs. robustness[C]//Advances in Neural Information Processing Systems (NeurIPS). 2020.

[190] Cortes C, DeSalvo G, Mohri M. Learning with rejection[C]//International Conference on Algorithmic Learning Theory. Springer, 2016: 67-82.

[191] Geifman Y, El-Yaniv R. Selective classification for deep neural networks[C]//Advances in Neural Information Processing Systems (NIPS). 2017.

[192] Geifman Y, El-Yaniv R. Selectivenet: A deep neural network with an integrated reject option [C]//International Conference on Machine Learning (ICML). 2019.

[193] Wu X, Jang U, Chen J, et al. Reinforcing adversarial robustness using model confidence induced by adversarial training[C]//International Conference on Machine Learning (ICML). PMLR, 2018: 5334-5342.

[194] Sotgiu A, Demontis A, Melis M, et al. Deep neural rejection against adversarial examples[J]. EURASIP Journal on Information Security, 2020, 2020: 1-10.

[195] Zadeh P H, Hosseini R, Sra S. Deep-rbf networks revisited: Robust classification with rejection [J]. arXiv preprint arXiv:1812.03190, 2018.

[196] Zhang C, Bengio S, Hardt M, et al. Understanding deep learning requires rethinking generalization[C]//International Conference on Learning Representations (ICLR). 2017.

[197] Rice L, Wong E, Kolter J Z. Overfitting in adversarially robust deep learning[C]//International Conference on Machine Learning (ICML). 2020.

[198] Bulusu S, Kailkhura B, Li B, et al. Anomalous instance detection in deep learning: A survey [J]. arXiv preprint arXiv:2003.06979, 2020.

[199] Zheng Z, Hong P. Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks[C]//Advances in Neural Information Processing Systems (NeurIPS). 2018.

[200] Liu W, Wang X, Owens J, et al. Energy-based out-of-distribution detection[J]. Advances in Neural Information Processing Systems (NeurIPS), 2020.

[201] Gowal S, Uesato J, Qin C, et al. An alternative surrogate loss for pgd-based adversarial testing [J]. arXiv preprint arXiv:1910.09338, 2019.

[202] Sriramanan G, Addepalli S, Baburaj A, et al. Guided adversarial attack for evaluating and enhancing adversarial defenses[C]//Advances in Neural Information Processing Systems (NeurIPS). 2020.

[203] Cai Q Z, Liu C, Song D. Curriculum adversarial training[C]//International Joint Conference on Artificial Intelligence (IJCAI). 2018: 3740-3747.

[204] Qin C, Martens J, Gowal S, et al. Adversarial robustness through local linearization[C]// Advances in Neural Information Processing Systems (NeurIPS). 2019: 13824-13833.

[205] Zhang H, Wang J. Defense against adversarial attacks using feature scattering-based adversarial training[C]//Advances in Neural Information Processing Systems (NeurIPS). 2019: 1829-1839.

[206] Atzmon M, Haim N, Yariv L, et al. Controlling neural level sets[C]//Advances in Neural Information Processing Systems (NeurIPS). 2019: 2034-2043.

[207] Ding G W, Sharma Y, Lui K Y C, et al. Mma training: Direct input space margin maximization through adversarial training[C]//International Conference on Learning Representations (ICLR). 2020.

[208] Huang L, Zhang C, Zhang H. Self-adaptive training: beyond empirical risk minimization[J]. arXiv preprint arXiv:2002.10319, 2020.

[209] Cheng M, Lei Q, Chen P Y, et al. Cat: Customized adversarial training for improved robustness [J]. arXiv preprint arXiv:2002.06789, 2020.

[210] Lee S, Lee H, Yoon S. Adversarial vertex mixup: Toward better adversarially robust generalization[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2020: 272-281.

[211] Xu Z, Shafahi A, Goldstein T. Exploring model robustness with adaptive networks and improved adversarial training[J]. arXiv preprint arXiv:2006.00387, 2020.

[212] Goyal P, Dollár P, Girshick R, et al. Accurate, large minibatch sgd: Training imagenet in 1 hour [J]. arXiv preprint arXiv:1706.02677, 2017.

[213] Xie C, Wu Y, van der Maaten L, et al. Feature denoising for improving adversarial robustness [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2019.

[214] Shafahi A, Ghiasi A, Huang F, et al. Label smoothing and logit squeezing: A replacement for adversarial training?[J]. arXiv preprint arXiv:1910.11585, 2019.

[215] Summers C, Dinneen M J. Logit regularization methods for adversarial robustness[J/OL]. ICLR submission, 2018. https://openreview.net/forum?id=BJlr0j0ctX.

[216] Jiang L, Ma X, Weng Z, et al. Imbalanced gradients: A new cause of overestimated adversarial robustness[J]. arXiv preprint arXiv:2006.13726, 2020.

[217] Loshchilov I, Hutter F. Decoupled weight decay regularization[C]//International Conference on Learning Representations (ICLR). 2019.

[218] Yong H, Huang J, Hua X, et al. Gradient centralization: A new optimization technique for deep neural networks[C]//European Conference on Computer Vision (ECCV). 2020.

[219] Xie C, Tan M, Gong B, et al. Smooth adversarial training[J]. arXiv preprint arXiv:2006.14536, 2020.

[220] Su D, Zhang H, Chen H, et al. Is robustness the cost of accuracy? – a comprehensive study on the robustness of 18 deep image classification models[C]//The European Conference on Computer Vision (ECCV). 2018.

[221] Xie C, Yuille A. Intriguing properties of adversarial training at scale[C]//International Conference on Learning Representations (ICLR). 2020.

[222] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[C]//International Conference on Learning Representations (ICLR). 2021.