

上海交通大学博士学位论文

生成对抗网络的理论分析与优化

博士研究生：周志明

学 号：0140339031

导 师：俞勇教授

申 请 学 位：博士

学 科：计算机科学与技术

所 在 单 位：电子信息与电气工程学院

答 辩 日 期：2020 年 5 月 15 日

授 予 学 位 单 位：上海交通大学

Dissertation Submitted to Shanghai Jiao Tong University
for the Degree of Doctor

**ON THE THEORETICAL ANALYSIS AND
OPTIMIZATION OF GENERATIVE
ADVERSARIAL NETWORKS**

Candidate: ZHIMING ZHOU
Student ID: 0140339031
Supervisor: Prof. YONG YU
Academic Degree Applied for: Ph.D.
Speciality: Computer Science and Engineering
Affiliation: ELECTRONIC INFORMATION AND ENGINEERING
Date of Defence: May 15, 2020
Degree-Conferring-Institution: Shanghai Jiao Tong University

生成对抗网络的理论分析与优化

摘要

生成对抗网络 (GANs)，作为目前最为流行的生成模型之一，在各种各样的生成以及相关任务上带来了突破性的进展，然而生成对抗网络的训练却存在很大的困难，主要表现为：训练不稳定、样本质量差等。有很多工作试图去解释生成对抗网络的训练问题，但这些问题依然没有被很好地理解和解决。本文围绕生成对抗网络的这两个核心问题展开理论上的研究，旨在为生成对抗网络的训练提供理论基础、指导和解决方案。

本文首先从最优判别函数的角度研究生成对抗网络的训练不稳定性问题。我们发现，如果不对生成对抗网络的判别函数空间做任何限制，那么该类生成对抗网络均不能保证其收敛性。文中论证了，在这种情况下，生成器的优化所依赖的最优判别函数的梯度将一般性地无法反映真实分布的任何信息。与此相对的，基于为 Wasserstein 距离的生成对抗网络可以证明在训练过程中不会有此梯度无信息问题，其判别函数空间被约束在 1-Lipschitz 的空间。这样的发现驱使我们进一步探索 Lipschitz 条件在生成对抗网络中的作用和影响。通过研究一个基于 Lipschitz 条件的通用生成对抗网络目标函数，我们得到了一族新的可保证收敛的生成对抗网络的定义。我们把这族生成对抗网络称之为 LGANs (基于 Lipschitz 的 GANs)。我们证明 LGANs 保证最优判别函数的存在性和唯一性，并且生成器和最优判别器之间存在一个唯一的纳什均衡点。我们证明 LGANs 能够一般性地解决梯度无信息问题，并使得生成样本的梯度趋于指向真实样本。根据我们的实验，LGANs 的训练相比于 WGAN 更加稳定，同时，也能得到更高质量的生成样本。

基于对最优判别函数的深入理解，我们发现生成对抗网络通常都没有被很好的优化，因为在实践过程中的判别函数通常都没有达到最

优判别函数的性质。而与此同时，生成对抗网络的训练通常依赖于自适应学习率算法（以 Adam 为其主要代表），被发现在某些情况下存在不能正确收敛的问题。作为本文的第二部分，我们研究自适应学习率算法的收敛性问题，以更好地支撑对生成对抗网络的理解。我们发现，在自适应学习率算法中，梯度 g_t 和二阶矩量项 v_t 之间存在正相关性（下标 t 表示时刻），而这进一步导致：数值较大的梯度倾向于得到较小的更新步长，而数值较小的梯度倾向于得到较大的更新步长。文中论证了这样的步长偏向性是自适应学习率算法不收敛的根本原因之一。同时，我们证明，一旦 v_t 和 g_t 相互独立，自适应学习率算法的步长偏向性问题将得到解决，进而保证其收敛性和随机梯度下降类似。最后，我们提出一个新的自适应学习率算法，称之为 AdaShift。AdaShift 通过时序偏移使得二阶矩量项 v_t 和当前梯度 g_t 相互独立，即，利用 g_{t-n} 来计算 v_t 。实验表明，AdaShift 能有效地解决自适应学习率算法的不收敛性问题，同时，能保证和 Adam 相匹敌的训练速度和泛化能力，甚至在某些任务上能取得更好的结果。实验中我们也看到，AdaShift 能在一定程度上改善生成对抗网络的训练。

紧接着，作为本文的第三部分，我们研究生成对抗网络的第二个核心问题：样本质量问题。长期以来，生成对抗网络都难以在大而复杂数据集上达到令人满意的样本质量。但人们发现，在这种情况下，标签信息能有效地提升生成样本的质量。在该部分，我们系统地研究标签信息如何影响生成对抗网络的训练。通过分析标签信息如何影响判别器逻辑值上的梯度反传，以及多维交叉熵目标函数的分解，我们揭示了标签信息如何与生成对抗网络交互。这些分析也同时预示了当前各个试图利用标签信息的生成对抗网络模型所存在的不足和缺陷。基于此，我们提出了激活最大化生成对抗网络（AM-GAN），以改善生成对抗网络对标签信息的利用。我们对这些分析以及所提出的模型进行了系统的实验验证。实验中，所提出的模型以明显优势击败了当前的各个主流方法，样本质量达到了最先进水平。其实，一直以来，生成模型的样本质量的评估方法都存在争议。因此，在本文的该部分，我

我们也研究了样本质量的评估方法。我们发现主流的评估指标 Inception Score，在采用 Inception 模型作为其基础分类器时，主要衡量的是生成样本的多样性，而未发现有利的证据表明它能很好地衡量样本的质量。基于此，我们也进一步提出了一个新的评测指标，称之为 AM Score，以更加精准地衡量样本的质量。

关键词：生成对抗网络，收敛性，样本质量，利普希茨连续，自适应学习率算法，标签信息

ON THE THEORETICAL ANALYSIS AND OPTIMIZATION OF GENERATIVE ADVERSARIAL NETWORKS

ABSTRACT

Generative adversarial networks (GANs), as one of the most successful generative models, have shown promising results in various challenging tasks. However, GANs is also well-known for its difficulties in training. The common issues include training instability, low-quality sample, etc. The underlying obstacles, though have been heavily studied, are still not fully understood. In this paper, we study GANs theoretically, aimed at addressing the main issues in the training of GANs and providing theoretical guidance and solution for the training of GANs.

We first study the convergence of GANs from the view of the optimal discriminative function. We show that GANs without restriction on the discriminative function space commonly suffer from the problem that the gradient produced by the discriminator is uninformative to guide the generator. By contrast, Wasserstein GAN (WGAN), where the discriminative function is restricted to 1-Lipschitz, does not suffer from such a gradient un-informativeness problem. This implies the importance of Lipschitz condition and motivates us to study the general formulation of GANs with Lipschitz constraint, which leads to a new family of GANs that we call Lipschitz GANs (LGANs). We show that LGANs guarantee the existence and uniqueness of the optimal discriminative function as well as the existence of a unique Nash equilibrium. We prove that LGANs are generally capable of eliminating the gradient uninformativeness problem such that the gradient of generated sample tends to point to real sample. According to our empirical analysis,

LGANs are more stable and generate consistently higher quality samples compared with WGAN.

From the perspective of optimal discriminative function, we found that GANs are usually not well-optimized, i.e., the properties of the optimal discriminative function usually do not hold. In the meantime, it was shown that Adam, what current GANs heavily relies on for its optimization, not being able to converge to the optimal solution in certain cases. For the second part of this paper, we provide new insight into the non-convergence issue of Adam as well as other adaptive learning rate methods. We argue that there exists a positive correlation between gradient g_t and the second-moment term v_t in Adam (t is the time-step), which results in that a large gradient is likely to have small step size while a small gradient may have a large step size. We demonstrate that such biased step sizes are the fundamental cause of non-convergence of Adam, and we further prove that decorrelating v_t and g_t will lead to unbiased step size for each gradient, thus solving the non-convergence problem of Adam. Finally, we propose AdaShift, a novel adaptive learning rate method that achieves independence between v_t and g_t by temporal shifting, using temporally shifted gradient g_{t-n} to calculate v_t . The experiment results demonstrate that AdaShift is able to address the non-convergence issue of Adam, while still maintaining a competitive performance with Adam in terms of both training speed and generalization. And we also found evidences that AdaShift helps GANs training.

Then, as the third part of this paper, we study the second core problem of GANs, i.e., the sample quality issue. For a long time, GANs are hard to achieve satisfactory sample quality when facing large and complicated dataset. And class labels have been empirically shown useful in improving the sample quality of GANs. In this part, we mathematically study how class labels interact with GANs' optimization. With class aware gradient and cross-entropy decomposition, we uncover how class labels and

associated losses influence GANs' training. The analysis also indicates the disadvantages of existing GAN models that make use of class labels. Based on that, we propose Activation Maximization Generative Adversarial Networks (AM-GAN) as an advanced solution. Comprehensive experiments have been conducted to validate our analysis and evaluate the effectiveness of our solution, where AM-GAN outperforms other strong baselines and achieves state-of-the-art sample quality. Given the existence of controversy on the evaluation metrics of sample quality, in this part, we also investigate existing evaluation metrics for GANs. We demonstrate that, with the Inception ImageNet classifier, Inception Score mainly tracks the diversity of the generator, and there is, however, no reliable evidence that it can reflect the true sample quality. We thereby propose a new metric, called AM Score, to provide a more accurate estimation of the sample quality.

KEY WORDS: Generative adversarial networks (GANs), Convergence, Sample Quality, Lipschitz continuity, Adaptive Learning Rate Methods, Label Information

目 录

| | |
|--|----------|
| 第一章 绪论 | 1 |
| 1.1 研究背景和意义 | 1 |
| 1.2 研究现状 | 2 |
| 1.3 研究课题 | 3 |
| 1.3.1 训练稳定性问题 | 3 |
| 1.3.2 自适应学习率算法 | 4 |
| 1.3.3 样本质量问题 | 5 |
| 1.4 章节安排 | 6 |
| 第二章 生成对抗网络的收敛性 | 7 |
| 2.1 引言 | 7 |
| 2.2 预备知识 | 9 |
| 2.2.1 Lipschitz 连续与 Wasserstein 距离 | 9 |
| 2.2.2 生成对抗网络的一般表达式 | 10 |
| 2.2.3 梯度消失问题 | 11 |
| 2.3 梯度无信息问题 | 11 |
| 2.3.1 判别函数空间无限制 | 12 |
| 2.3.2 判别函数空间中有限制: 以 Fisher GAN 为例 | 12 |
| 2.3.3 Wasserstein GAN | 13 |
| 2.4 Lipschitz GANs | 14 |
| 2.4.1 理论分析 | 15 |
| 2.4.2 相互绑定关系的进一步结论 | 16 |
| 2.4.3 相互绑定关系的蕴意 | 16 |
| 2.5 Lipschitz 约束的实现 | 17 |
| 2.5.1 现有方法 | 17 |
| 2.5.2 全局约束的必要性 | 18 |
| 2.5.3 惩罚法中的冗余约束 | 19 |
| 2.5.4 最大梯度惩罚 | 19 |
| 2.6 实验分析 | 20 |
| 2.6.1 梯度无信息问题的实践表现 | 20 |

| | | |
|------------------------|--|-----------|
| 2.6.2 | Lipschitz 约束的实现方法对比 | 21 |
| 2.6.3 | 验证 LGANs 中最优判别函数的梯度性质 | 22 |
| 2.6.4 | 最优判别函数的唯一性以及稳定性 | 27 |
| 2.6.5 | 用无监督图片生成任务做基准测试 | 27 |
| 2.7 | 相关工作 | 29 |
| 2.8 | 结论 | 30 |
| 本章附录 | | 31 |
| 2.A | 证明 | 31 |
| 2.A.1 | 定理 2.2 的证明 | 31 |
| 2.A.2 | 定理 2.3 的证明 | 36 |
| 2.A.3 | 定理 2.4 的证明 | 38 |
| 2.A.4 | 定理 2.5 的证明 | 39 |
| 2.A.5 | Wasserstein 距离新对偶式的证明 | 40 |
| 2.B | 梯度无信息问题的实践表现 | 41 |
| 2.C | 技术细节与延拓 | 42 |
| 2.C.1 | 满足公式(2-11)的 ϕ 和 φ | 42 |
| 2.C.2 | 实验细节以及更多的结果 | 43 |
| 第三章 生成对抗网络的学习算法 | | 51 |
| 3.1 | 引言 | 51 |
| 3.2 | 准备知识 | 53 |
| 3.3 | 不收敛问题的原因: 步长的偏向性 | 55 |
| 3.3.1 | 净更新步长 | 55 |
| 3.3.2 | 在线优化问题中的反例 | 55 |
| 3.3.3 | 不收敛问题的一般性结论 | 57 |
| 3.4 | AdaShift: 通过时序偏移去除相关性 | 57 |
| 3.4.1 | 通过时序偏移去相关 | 59 |
| 3.4.2 | 利用空间上的元素 | 59 |
| 3.4.3 | 空间函数的选择: 降维与共享 | 60 |
| 3.4.4 | 一阶矩量的估计: 滑动平均窗口 | 60 |
| 3.5 | 实验 | 62 |
| 3.5.1 | 在线优化问题的反例 | 62 |
| 3.5.2 | 在 MNIST 上的逻辑回归和多层感知机 | 62 |

| | | |
|-------------|--|-----------|
| 3.5.3 | CIFAR-10 上的 DenseNet 和 ResNet | 63 |
| 3.5.4 | Tiny-ImageNet 上的 DenseNet | 65 |
| 3.5.5 | 生成模型以及循环网络结构 | 65 |
| 3.6 | 本章小结 | 66 |
| 本章附录 | | 69 |
| 3.A | AdaShift 算法的伪代码 | 69 |
| 3.B | 证明 | 69 |
| 3.B.1 | 定理 3.1 的证明 | 69 |
| 3.B.2 | 引理 3.2 的证明 | 70 |
| 3.B.3 | 引理 3.3 的证明 | 71 |
| 3.B.4 | 引理 3.4 的证明 | 73 |
| 3.B.5 | 引理 3.7 的证明 | 75 |
| 3.C | β_1 、 β_2 以及 C 之间的临界关系 | 76 |
| 3.D | 相关性测试 | 78 |
| 3.E | 仅时序偏移和仅空间操作 | 79 |
| 3.F | 算法中的超参数 | 80 |
| 3.F.1 | 实验中的超参数设定 | 80 |
| 3.F.2 | 学习率 α_t 的敏感性 | 80 |
| 3.F.3 | β_1 和 β_2 的敏感性 | 81 |
| 3.F.4 | n 和 m 的敏感性 | 82 |
| 3.G | 拓展对比实验 | 83 |
| 第四章 | 生成对抗网络的样本质量 | 87 |
| 4.1 | 引言 | 87 |
| 4.2 | 背景与相关工作 | 88 |
| 4.2.1 | 标签拓展的生成对抗网络 | 88 |
| 4.2.2 | 带有附属分类器的生成对抗网络 | 89 |
| 4.3 | 类别标签在梯度反传过程中的影响 | 90 |
| 4.4 | 激活最大化生成对抗网络 | 91 |
| 4.4.1 | 交叉熵分解以及模型之间的联系 | 93 |
| 4.4.2 | 层次化模型与非层次化模型 | 94 |
| 4.5 | 拓展内容 | 95 |
| 4.5.1 | 自动标签技术 | 95 |

| | | |
|--------------------------|---|------------|
| 4.5.2 | 从激活最大化的角度理解对抗训练 | 96 |
| 4.5.3 | 为附属分类器引入对抗训练 | 96 |
| 4.6 | 评测指标 | 97 |
| 4.6.1 | Inception Score | 97 |
| 4.6.2 | Inception Score 的一个等价变种: Mode Score | 99 |
| 4.6.3 | Inception Score 的分解项并不像想象中那样工作 | 99 |
| 4.6.4 | Inception Score 衡量样本多样性的能力 | 100 |
| 4.6.5 | AM Score | 101 |
| 4.7 | 实验分析 | 104 |
| 4.7.1 | 附属分类器的作用探究 | 104 |
| 4.7.2 | 不同模型之间的对比 | 104 |
| 4.7.3 | 和其他相关工作的对比 | 106 |
| 4.7.4 | 自动标签技术与预定义的标签 | 106 |
| 4.7.5 | 训练曲线 | 107 |
| 4.7.6 | 用 Tiny-ImageNet 做进一步的验证 | 108 |
| 4.8 | 结论 | 109 |
| 本章附录 | | 111 |
| 4.A | 将 AM-GAN 拓展到无标签数据 | 111 |
| 4.A.1 | 半监督学习 | 111 |
| 4.A.2 | 无监督学习 | 111 |
| 4.B | 网络结构与超参数 | 112 |
| 全文总结 | | 123 |
| 参考文献 | | 125 |
| 致 谢 | | 133 |
| 攻读博士学位期间已发表或录用的论文 | | 135 |

插图索引

| | |
|--|----|
| 图 2-1 梯度无信息问题的实践表现 | 20 |
| 图 2-2 梯度惩罚和最大梯度惩罚的对比：二维数据 | 23 |
| 图 2-3 梯度惩罚和最大梯度惩罚的对比：高维数据 | 24 |
| 图 2-4 在二维数据上验证 LGANs 的最优判别函数的梯度方向 | 25 |
| 图 2-5 在 CIFAR-10 数据上可视化最优判别函数的梯度 I | 26 |
| 图 2-6 在 LGANs 中判别函数更加稳定 | 27 |
| 图 2-7 FID 训练曲线 | 28 |
| 图 2-8 梯度无信息问题的实践表现 I | 41 |
| 图 2-9 梯度无信息问题的实践表现 II | 42 |
| 图 2-10 梯度无信息问题的实践表现 III | 42 |
| 图 2-11 梯度无信息问题的实践表现 IV | 42 |
| 图 2-12 梯度无信息问题的实践表现 V | 43 |
| 图 2-13 满足条件的 ϕ 和 φ 示例 | 43 |
| 图 2-14 CIFAR-10 上的 Inception Score 训练曲线 | 45 |
| 图 2-15 Tiny ImageNet 上的 Inception Score 训练曲线 | 45 |
| 图 2-16 不同 LGANs 训练实例的随机采样样本 I | 46 |
| 图 2-17 不同 LGANs 训练实例的随机采样样本 II | 47 |
| 图 2-18 不同 LGANs 训练实例的随机采样样本 III | 48 |
| 图 2-19 在 CIFAR-10 数据上可视化最优判别函数的梯度 II | 49 |
| 图 3-1 关于随机在线优化问题上的实验 | 63 |
| 图 3-2 MNIST 上的逻辑回归 | 63 |
| 图 3-3 MNIST 上的多层感知机 | 64 |
| 图 3-4 CIFAR-10 上的 ResNet | 64 |
| 图 3-5 CIFAR-10 上的 DenseNet | 64 |
| 图 3-6 Tiny-ImageNet 上的 DenseNet | 65 |
| 图 3-7 生成模型与循环网络结构 | 66 |
| 图 3-8 β_1 、 β_2 以及 C 的临界关系 | 77 |
| 图 3-9 仅时序偏移和仅空间操作 I | 80 |
| 图 3-10 仅时序偏移和仅空间操作 II | 80 |

| | |
|---|-----|
| 图 3-11 学习率敏感性实验 | 82 |
| 图 3-12 学习率敏感性实验 | 83 |
| 图 3-13 β_1 和 β_2 敏感性实验 I | 83 |
| 图 3-14 β_1 和 β_2 敏感性实验 II | 84 |
| 图 3-15 n 的敏感性实验 | 84 |
| 图 3-16 m 的敏感性实验 | 84 |
| 图 3-17 n 和 m 的敏感性实验 | 85 |
| 图 3-18 拓展对比试验 I | 85 |
| 图 3-19 拓展对比试验 II | 85 |
| 图 3-20 拓展对比试验 III | 86 |
| | |
| 图 4-1 梯度耦合问题的图示 | 92 |
| 图 4-2 AM-GAN 和 AC-GAN 的对比 | 94 |
| 图 4-3 Inception Score 以及它的分解项随训练的变化曲线 | 100 |
| 图 4-4 CIFAR-10 训练数据的统计信息 | 101 |
| 图 4-5 真实图片的 Inception 单样本熵 | 102 |
| 图 4-6 Inception Score 的模式崩塌测试 | 103 |
| 图 4-7 AM Score 以及它的分解项随训练的变化曲线 | 103 |
| 图 4-8 训练曲线的对比 | 108 |
| 图 4-9 随机采样的样本示例 I | 113 |
| 图 4-10 随机采样的样本示例 II | 114 |
| 图 4-11 随机采样的样本示例 III | 115 |
| 图 4-12 随机采样的样本示例 IV | 116 |
| 图 4-13 随机采样的样本示例 V | 117 |
| 图 4-14 随机采样的样本示例 VI | 118 |
| 图 4-15 随机采样的样本示例 VII | 119 |
| 图 4-16 随机采样的样本示例 VIII | 120 |
| 图 4-17 随机采样的样本示例 IX | 121 |
| 图 4-18 随机采样的样本示例 X | 122 |

表格索引

| | |
|-------------------------------------|-----|
| 表 2-1 不同生成对抗网络的目标函数之间的比较 | 8 |
| 表 2-2 在无监督图片生成任务下的量化对比 | 28 |
| 表 2-3 网络结构 | 44 |
| 表 3-1 不同时刻的梯度的时序相关系数 | 78 |
| 表 3-2 梯度不同维度间的空间相关系数 | 78 |
| 表 3-3 梯度与二阶项之间的的相关系数 I | 79 |
| 表 3-4 梯度与二阶项之间的的相关系数 II | 79 |
| 表 3-5 逻辑回归的超参数设定 | 81 |
| 表 3-6 MNIST 上的多层感知机的超参数设定 | 81 |
| 表 3-7 WGAN-GP 的超参数设定 | 81 |
| 表 3-8 神经机器翻译的超参数设定 | 81 |
| 表 3-9 其他实验的超参数设定 | 82 |
| 表 4-1 对比标签信息的各种用法 | 105 |
| 表 4-2 平均 MS-SSIM 的最大值 | 105 |
| 表 4-3 和其他流行的模型对比 | 107 |

算法索引

| | |
|---------------------------------|----|
| 算法 3-1 AdaShift | 59 |
| 算法 3-2 AdaShift (完整版) | 69 |

主要符号对照表

| | |
|----------------|---------------------------|
| P_r | 真实分布 |
| P_g | 真实分布 |
| S_r | 真实分布的支撑集 |
| S_g | 生成分布的支撑集 |
| G | 生成器 |
| D | 判别器 |
| f | 判别函数 |
| g | 生成函数 |
| \mathcal{G} | 生成函数空间 |
| \mathcal{F} | 判别函数空间 |
| f^* | 最优判别函数 |
| \mathbb{E} | 期望 |
| \mathbb{E}_x | 关于 x 期望 |
| H | 熵、交叉熵 |
| KL | 相对熵 (Kullback-Leibler 散度) |
| ∇ | 关于输入的梯度 |
| ∇_x | 关于 x 的梯度 |
| $v(\cdot)$ | 标签向量化算子 |
| σ | Sigmoid 函数 |
| g_t | 当前时刻的梯度 |
| v_t | 当前时刻的二阶矩量 |
| m_t | 当前时刻的一阶矩量 |

第一章 绪论

本章介绍生成对抗网络的研究背景以及它的主要功能与作用，同时介绍生成对抗网络有关的主要研究课题的来源和现状，以及我们的主要贡献和结论。

1.1 研究背景和意义

生成对抗网络^[1]（GANs）是机器学习领域近年来非常热门的课题，被认为是生成模型里目前最具有代表性和最具潜力的方法之一。

生成模型的基本任务是拟合给定数据的分布。它分为两种基本类型：一种是显示地学习分布的参数表示；另一种是学习生成样本，使得生成的样本的分布和给定的数据分布一致。

目前的主流研究是集中在后者，典型的模型包括变分自编码器^[2]（VAE）和生成对抗网络。生成对抗网络相对于变分自编码器能生成更真实的样本，但同时，生成对抗网络的训练相较于变分自编码器而言，目前显得困难的多。

生成模型是各种重要的机器学习算法的基本组成部分，尤其在计算机视觉和自然语言处理等方向有着重要的应用。典型的应用包括样本的生成与转化，数据的建模与探索，无监督与半监督学习等。

- **生成与转化任务。**生成对抗网络本身作为生成模型，可以用在各类以生成样本为目标的任务上，如：艺术图片生成^[3]、自然语言生成^[4]。同时，也可以用于一些转化任务，比如：文本到图像^[5]、风格转换^[6]、图片上色^[7]等。
- **数据建模与探索。**训练完成的生成模型是一个对数据的建模。基于数据模型人们可以做：数据差值^[8]、数据的语义运算^[9]、基于提示的数据的检索和浏览^[10]、数据的可靠编辑^[11]等。
- **无监督与半监督学习。**生成模型还可以用于无监督聚类^[12]，无监督的特征维度抽取^[13]，无监督的多模态输出^[14]。同时，生成模型的生产样本可以促进半监督学习，有数个代表性的工作基于生成对抗网络在半监督学习上取得了重要进展^[15, 16]。

实际上，凭借着生成对抗网络日益精湛的建模能力，它已经成为大多数生成类的任务的一个重要的组件。生成对抗网络虽然被广泛应用，但目前生成对抗网络的样本质量以及训练稳定性还有待提升^[17]，主要表现为在一些相对复杂的数据集上生成样本的质量较差，训练容易发散、不收敛，出现模式崩塌等。

1.2 研究现状

近年来，大量的研究工作都致力于提升生成对抗网络的训练稳定性和样本质量，而这些工作可以分为 4 个大类：

- **网络结构层面的方法。**其中有通过网络结构的改进增强训练的稳定性，如 [9] 提出一个规整的全卷积的网络结构模型。该类方法的本质在于通过网络结构来增加隐式的正则和约束，使得训练过程中生成函数，尤其判别函数，更加稳定。此外，还有许多工作提出层次递进或者从粗粒度到细粒度逐渐生成细节的方法来提升生成对抗网络的训练，该类的代表性论文包括 [5, 8, 18, 19]。其核心思想是通过将目标分解成多个较容易的子任务，降低训练的难度，使得每次从当前状态到当前子任务的最优解的距离更加接近，从而使得训练更加容易和稳定。
- **度量函数层面的方法。**因为从生成对抗网络的原始论文开始人们就已经注意到生成对抗网络的目标函数存在梯度消失的问题，所以很多研究人员认为生成对抗网络的训练不稳定性，可能源自度量函数上的问题，或者和梯度消失有关。大量的新的度量函数因此争相涌出，包括 [20-31]。除了在基础机构不变的前提下修改度量函数的工作，也有工作对整体架构了一定的修改的，如 [32] 将判别器的输入改为一个分布，表达为一族样本，而不是单一样本，而 [33] 将判别器的输入改为一个真实样本加一个生成样本，度量两个样本之间的真实程度的差异。度量函数层面的方法中，最广为接受的理论来自 [34]，该论文指出原始的生成对抗网络所以来的 JS 散度在实践过程中存在问题：因为生成分布和真实分布通常在初期差异很大，导致两者支撑集的覆盖率可能接近于 0，这将导致 JS 散度无法带来有效的梯度信息，这也是梯度消失的根本原因。在此基础上，[30] 提出基于 Wasserstein 距离的生成对抗网络，Wasserstein 距离是最优传输距离的一种，他在支撑集不相交的情况下也能很好地度量分布间的距离。实践中采用的 Wasserstein 距离的对偶式，因为对偶式中存在 Lipschitz 约束，后续也有许多论文针对于 Lipschitz 约束的实现方法提出改进，如 [35-37]。
- **基于显示正则的方法。**显示正则一直以来是机器学习和深度学习的热门课题，它对于训练的稳定性，防止过拟合，保证泛化能力等有着重要的作用。典型的显示正则包括：参数衰减 (Weight Decay)^[38, 39]，Dropout^[40]，Lipschitz^[41, 42] 等。值得一提的是，深度神经网络训练中常用的 BN^[43] 也被一般认为具有正则作用。引入显示正则也是一类常见的关于提升生成对抗网络的训练稳定性的尝试。代表工作包括 [36, 44-48]。

- **优化算法层面的方法。**生成对抗网络的优化是一个 minimax 优化问题，一般采用生成器和判别器的迭代优化。很多人认为普通的迭代优化可能没办法解得最优解，或者认为是训练不收敛性问题原因。由于 minimax 形式和博弈游戏（game）的相似性很多人将博弈论里的相关理论和优化方法借鉴到生成对抗网络的优化中来。如 [49-54]。
- **结合自编码器的方法。**有一类相对特殊的方法，其核心思想或是想结合 VAE 和 GANs，或者其他，最终都是通过引入自编码器来帮助 GANs 的训练。其内在的原理来自于自编码器本身的稳定性，而自编码器需要结合 GANs 是因为自编码器容易产生不清晰的糊的样本，类似 VAE。自编码器的稳定性有助于防止模式崩塌，能将训练稳定地拉下相对靠近最优解的位置。代表工作有 [55-61]。

此外，还有一些难以归类的技术性的处理，如 [8, 15, 62] 中所提到的一些小技巧。

即使存在上述各种各样的关于生成对抗网络的训练稳定性、样本质量等问题的尝试和思考，这些问题依旧没有得到系统性的解决，多数工作只是经验性的探索，许多问题在理论上的理解依旧存在明显不足。

1.3 研究课题

在本论文中，我们围绕生成对抗网络的两个核心问题，即训练稳定性问题和样本质量问题，从理论的角度展开研究，希望通过生成对抗网络的理论的完善，进一步发挥生成对抗网络在各相关领域和方向的关键性作用。

1.3.1 训练稳定性问题

生成对抗网络的目标函数通常被定义为真实分布 P_r 和生成分布 P_g 之间的一个距离度量，这同时也意味着 $P_r = P_g$ 是一个唯一的全局最优解。[34] 认为生成对抗网络的训练不稳定性问题是生成对抗网络所采用的距离度量的性质不好造成的。以原始 GAN 为例，当真实分布和生成分布的支撑集不相交的时候，原始 GAN 所衡量的 JS 散度是一个固定的常数，这将导致来自于距离度量的梯度无法指导生成器靠近真实分布。[30] 沿着这个思路做了进一步的工作，他们提出采用 Wasserstein 距离作为 GANs 的距离度量，因为 Wasserstein 距离即使在两个分布的支撑集不相交的情况下也能很好地度量分布之间的距离。

距离度量函数的性质是人们已知的一个影响收敛性的重要因素，这也代表着关于生成对抗网络训练稳定性的主流理解和解决思路。但是，我们发现从距离度

量函数的角度，并不能完美地解释生成对抗网络的训练稳定性问题。我们发现分布重叠的情况下，存在例子使得 JS 散度本应能很好地度量分布之间的距离，但基于 JS 散度的 GANs 依然无法正确优化。

这驱使我们从新的角度去观察和理解生成对抗网络的训练稳定性问题。我们发现如果不对判别器的函数空间做任何限制的话（包括原始 GAN 以及其他许多变种），最优判别函数在某个点的取值将仅仅和当前点上的真假分布的概率密度相关，从而不能反映任何其它点上的信息。而我们知道在生成对抗网络的训练过程中，通常生成分布和真实分布的支撑集是不相交的，在这种情况下，在判别器的函数空间没做任何限制的生成对抗网络模型必定都会存在一个我们称之为梯度无信息的问题，即，最优判别函数关于生成样本的梯度不包含任何关于真实分布的信息。因为生成器完全依赖于判别函数关于生成样本的梯度来更新，因此这样一个梯度无信息的问题，将导致生成器是否能收敛到真实分布没有任何保障。

按照 [35] 的分析，Wasserstein GAN 可以避免梯度无信息的问题。于是，我们深入研究为什么 Wasserstein GAN 可以避免梯度无信息问题。我们注意到 Wasserstein 距离的 Kantorovich-Rubinstein 对偶式中存在 Lipschitz 约束，而在 Lipschitz 约束被放松之后，梯度无信息问题就又出现了。这激发我们去思考是否 Lipschitz 约束才是解决梯度无信息问题的关键。

我们通过在生成对抗网的一般形式下分析 Lipschitz 约束的影响，最终得到了一个一般性的基于 Lipschitz 约束的 GAN 的表达式。我们发现一个很朴素的前提下，即，如果对 Lipschitz 常数进行惩罚，那它就能保证最优判别函数的存在性和唯一性，同时，它还保证存在一个唯一的关于最优判别函数和生成分布的纳什均衡，在这个纳什均衡处生成分布等于真实分布。由此便得出一族保证收敛的基于 Lipschitz 惩罚的生成对抗网络的目标函数。

这一系列发现从正则项的角度揭示了生成对抗网络的训练稳定性问题的影响因素，补充和完善了从距离度量函数性质视角的理解。

1.3.2 自适应学习率算法

对于最优判别函数的关注，使得我们意识到在生成对抗网络的实践训练中，尤其是在带有 Lipschitz 约束的情况下，判别函数远远没有达到最优。因为我们可以观察到判别函数的某些性质和最优判别函数本该具有的性质相差巨大。这驱使着我们进一步研究生成对抗网络的优化问题。

生成对抗网络在实践优化过程中，通常采用自适应学习率算法，如 Adam, RMSProp 等。人们发现相对于朴素的随机梯度下降，自适应学习率算法能使生成对

抗网络的训练变得更加稳定，同时也能达到更好的训练效果。而近期，有论文指出 Adam 等自适应学习率算法在某些情况下存在不能正确收敛的问题。这使得我们将注意的焦点转移到生成对抗网络优化所用的自适应学习率算法上来。

Adam 作为深度学习中被广泛采用的优化算法之一，一直以来优化表现稳定而优异。关于 Adam 的收敛性问题的提出，吸引了很多研究者的关注和兴趣。而现存的用以解决 Adam 收敛性问题的方法存在潜在的低效率问题。作为该论文的第二大块，我们深入探索 Adam 不收敛性问题的原因。

具体而言，我们发现在 Adam 算法中，二阶矩量 v_t 和当前梯度存在正相关性，而这会直接导致梯度更新的步幅存在偏向性：数值较大的梯度倾向于步幅较小，而数值较小的梯度倾向于步幅较大。我们论证步幅的偏向性是 Adam 不收敛问题的根本来源，并证明如果该偏向性完全去除，也即，若保证步幅和梯度无关，即可保证收敛性。我们从去相关性的角度，提出 AdaShift 算法，其核心思想是通过时序偏移使得二阶矩量不受当前梯度的影响。在假设梯度时序无关的前提下，实现去除二阶矩量和梯度相关性。

我们通过大量实验验证新算法的有效性。实验上，AdaShift 呈现出了颇具竞争力的表现，在各种设定下，收敛速度和泛化能力均不弱于 Adam。而在一些问题上，AdaShift 的表现明显优于 Adam，这个表现可能和 AdaShift 能更好地保证收敛性问题有关。同时，我们也观察到学习算法的改进确实有助于生成对抗网络的优化。

1.3.3 样本质量问题

生成对抗网络最近在各种各样的任务上都带来了很有竞争力的结果。尽管如此，目前的生成对抗网络模型还是难以生成非常令人信服的样本，尤其当训练数据集非常复杂的时候。与此同时，人们在实验中发现有效地利用类别标签信息能够显著地提高生成样本的质量。作为该论文的第三个重要组成部分，我们就标签信息如何影响生成对抗网络的训练以及如何更好地利用标签信息展开研究。

当前存在三种典型的运用类别标签信息的生成对抗网络模型：CatGAN^[12] 将判别器做成一个多类分类器；LabelGAN^[15] 将判别器在普通多类分类器的基础上额外引入一个平级的类别用以表示来自生成器的样本；AC-GAN^[63] 在真假二类分类的判别器的基础上，引入一个独立的分类器用以分类真实数据。通过将类别信息引入到训练过程中，这些模型在实验中得到了更高质量的生成样本。但是，这些现象背后的机制却没有得到很好的分析^[17]。

我们从数学的角度分析这些利用标签信息的生成对抗网络模型。通过分析标

签信息如何在 LabelGAN^[15] 的梯度反传中起作用，我们得到了一些重要的观察：1) LabelGAN 在优化过程中倾向于将每个样本朝着某个特定的类优化；2) 而与此同时，它存在梯度耦合的问题。此外，我们发现 AC-GAN^[63] 可以被看作是层次化的多类分类器，但它缺少在普通类别层上的对抗训练。

基于这些分析和对已有模型潜在缺陷的认识，我们相应地提出了新的解决方案。具体来说，我们主张任何时候我们都应该将样本朝着某个具体的类别优化，为了做到这点我们建议显示地为每个样本分配一个标签，因为具有显式目标类的模型将为生成器提供更清晰的梯度指导。

我们发现将代表真实数据的类别替换成 K 个具体的真实数据类别通常比简单地训练辅助分类器更好，因为在辅助分类器中缺少对抗训练会使模型更有可能模式崩溃并产生低质量的样本。我们还通过实验发现，预定义标签往往容易出现类内模式崩溃，并相应地提出动态标记作为解决方案。

我们从激活最大化的角度为这种类型的对抗训练提供了一种新的解释，并将所提出的模型被命名为激活最大化生成对抗网络（AM-GAN）。我们通过大量的对照实验证实 AM-GAN 的有效性，同时也验证了我们的分析。实验中，AM-GAN 达到了当前各种模型中最好的样本质量。

1.4 章节安排

本文的主体内容分为三个大章节，分别从训练稳定性、学习算法、样本质量等角度详尽的阐述生成对抗网络当前所面临的的问题以及我们就这些问题的研究成果。第一章为本绪论。第二章研究生成对抗网络的收敛性保证，以解释训练稳定性问题。第三章研究生成对抗网络所依赖的自适应学习率算法的收敛性问题，以改善生成对抗网络的优化。第四章研究生成对抗网络和标签信息的相互作用，以理解和改进标签信息对于样本质量的影响。最后一章总结全文。每个篇章所涉及的相关工作均会在各个章节的相关工作部分具体介绍。文中包含一些定理，证明较为复杂的部分，我们统一将证明放在了各个章节的附录部分，而相对简单的一些，则放于正文方便阅读。一些补充性的实验以及实验的具体设定也放于各章附录，以保证正文的简洁和流畅性。

第二章 生成对抗网络的收敛性

生成对抗网络^[1] (GANs)，作为目前最流行的生成模型之一，已经在各种各样的任务中取得了很有竞争力的结果。生成对抗网络因为它杰出的效果而被广泛使用，但是它同时也因为非常难训练而臭名昭著^[17]。有很多人试图研究为什么生成对抗网络很难训练，并尝试着提出了许多解决方案^[34, 49, 51, 52, 64, 65]。但目前为止，生成对抗网络的训练稳定性问题依旧没有一个足够深刻的理解。

本章中，我们从判别器的函数空间的角度对生成对抗网络的训练稳定性进行分析。我们关于此得出了一系列重要的结论。其中第一个结论是：如果没有对判别器的函数空间做任何约束，那么生成对抗网络的训练必然会出现问题。我们形式化地定义了其中一种较为典型的问题，即，判别器关于生成样本的梯度不包含任何真实分布的信息，我们称该问题为梯度无信息问题。其次，我们知道，基于 Wasserstein 距离对偶形式的 WGAN^[30] 并不存在梯度无信息的问题。我们深入研究了该问题，发现 Wasserstein 距离对偶形式中的 Lipschitz 条件其实是可以被放松的，放松后并不影响对于 Wasserstein 距离的估计，因而我们得到了一种新的放松之后的 Wasserstein 距离的对偶形式。我们发现，如果依据这个新的对偶形式构建一个生成对抗网络的模型，尽管该模型依旧是基于 Wasserstein 距离的，但是它却存在梯度无信息的问题。

这些发现暗示着 Lipschitz 条件的重要性，从而激励我们尝试着研究带有 Lipschitz 约束的更一般的生成对抗网络的形式。我们发现一旦对判别函数施加 Lipschitz 惩罚，一大类目标函数都将变成一个合理的生成对抗网络的目标函数。我们给出了这类生成对抗网络的一个通用表达形式，我们称之为 Lipschitz GANs (LGANs)。我们证明了 LGANs 中最优判别函数的存在性和唯一性。我们证明了 LGANs 存在一个唯一的纳什均衡，在该均衡点处生成分布等于真实分布。我们也证明了 LGANs 能保证消除梯度无信息问题。根据我们的实证分析，LGANs 相对于 WGAN 更稳定，生成样本的质量也更高。

2.1 引言

生成对抗网络的目标函数通常被定义为真实分布 P_r 和生成分布 P_g 之间的一个距离度量，这意味着 $P_r = P_g$ 是一个唯一的全局最优解。^[34] 认为生成对抗网络的训练不稳定性问题是所采用的距离度量的性质不好造成的。以原始 GAN 为例，当真实分布和生成分布的支撑集不相交的时候，原始 GAN 所衡量的 JS 散度

表 2-1 不同 GANs 目标函数之间的比较。

Table 2-1 Comparison of different objectives in GANs.

| | ϕ | φ | \mathcal{F} | $f^*(x)$ | Gradient Vanishing | Gradient Uninformative | $f^*(x)$ Uniqueness |
|-------------------|--|--------------------|--|--|--------------------|------------------------|---------------------|
| Vanilla GAN | $-\log(\sigma(-x))$ | $-\log(\sigma(x))$ | $\{f: \mathbb{R}^n \rightarrow \mathbb{R}\}$ | $\log \frac{P_r(x)}{P_g(x)}$ | Yes | Yes | Yes |
| Least-Squares GAN | $(x - \alpha)^2$ | $(x - \beta)^2$ | $\{f: \mathbb{R}^n \rightarrow \mathbb{R}\}$ | $\frac{\alpha P_r(x) + \beta P_g(x)}{P_r(x) + P_g(x)}$ | No | Yes | Yes |
| μ -Fisher GAN | x | $-x$ | $\{f: \mathbb{R}^n \rightarrow \mathbb{R}, \mathbb{E}_{x \sim \mu}[f(x)]^2 \leq 1\}$ | $\frac{1}{\mathcal{F}_\mu(P_r, P_g)} \frac{P_r(x) - P_g(x)}{\mu(x)}$ | No | Yes | Yes |
| Wasserstein GAN | x | $-x$ | $\{f: \mathbb{R}^n \rightarrow \mathbb{R}, k(f) \leq 1\}$ | N/A | No | No | No |
| Lipschitz GAN | any ϕ and φ satisfying Eq. (2-11) | | $\{f: \mathbb{R}^n \rightarrow \mathbb{R}\}; k(f)$ is penalized | N/A | No | No | Yes |

是一个固定的常数，这将导致来自于距离度量的梯度无法指导生成器靠近真实分布。[30] 沿着这个思路做了进一步的工作，他们提出采用 Wasserstein 距离作为生成对抗网络的距离度量，因为 Wasserstein 距离即使在两个分布的支撑集不相交的情况下也能很好地度量分布之间的距离。

在本章中，我们从最优判别函数梯度有效性的角度进一步研究生成对抗网络的收敛性。我们发现如果不对判别器的函数空间做任何限制的话（如原始 GAN 以及其他很多变种），最优判别函数在某个点的取值将仅仅和当前点上的真假分布的概率密度相关，从而不能反映任何其它点上的信息。而我们知道在生成对抗网络的训练过程中，通常生成分布和真实分布的支撑集是不相交的，在这种情况下，在判别器的函数空间没做任何限制的生成对抗网络模型必定都会存在一个我们称之为梯度无信息的问题，即，最优判别函数关于生成样本的梯度不包含任何关于真实分布的信息。因为生成器完全依赖于判别函数关于生成样本的梯度来更新，因此这样一个梯度无信息的问题，将导致生成器是否能收敛到真实分布没有任何保障。

值得一提的是，这个问题和梯度消失问题^[34] 本质上是一个完全不同的两个问题。梯度消失关注的是梯度的大小，而梯度有无信息问题关注的是梯度的方向，而一旦方向没有任何意义，不论梯度大小如何，生成器的收敛性都将没有任何保证。

按照 [35] 的分析，Wasserstein GAN 可以避免梯度无信息的问题。在本章中，我们深入研究为什么 Wasserstein GAN 可以避免梯度无信息问题。首先，我们发现在 Wasserstein 距离的 Kantorovich-Rubinstein 对偶式中，Lipschitz 约束是可以被进一步放松的而不影响对 Wasserstein 距离的正确估计。我们因此得到了一个新的 Wasserstein 距离的对偶式。然而，我们发现，在新的对偶式下，Wasserstein 距离的最优判别函数同样存在梯度无信息问题。这意味着是否采用 Wasserstein 距离可能并不是解决梯度无信息问题的关键。

我们注意到 Wasserstein 距离的 Kantorovich-Rubinstein 对偶式中存在 Lipschitz 约束，而在 Lipschitz 约束被放松之后，梯度无信息问题就又出现了。这激发我

们去思考是否 Lipschitz 约束才是解决梯度无信息问题的关键。我们通过在生成对抗网络的一般形式下分析 Lipschitz 约束的影响，最终得到了一个一般性的基于 Lipschitz 约束的生成对抗网络的表达式。我们发现一个很朴素的前提下，如果对 Lipschitz 常数进行惩罚，那它就能保证最优判别函数的存在性和唯一性，同时，它还保证存在一个唯一的关于最优判别函数和生成分布的纳什均衡，在这个纳什均衡处生成分布等于真实分布。我们把由此得出的一类生成对抗网络，称之为 Lipschitz GANs (LGANs)。我们证明了 LGANs 普遍可以解决梯度无信息问题，具体而言，最优判别函数关于生成样本的导数只要可导并且不是零，那么一定会指向一个真实样本。

本章的其余内容组织如下。在小节2.2中，我们先介绍一些相关工作和预备知识。在小节2.3中，我们细致地展开关于梯度无信息问题的研究。在小节2.4中，我们提出 Lipschitz GANs 并给出相关的理论分析。我们在小节2.6中做实验性的验证和分析，在小节2.7讨论相关工作，并最后在2.8中做本章小结。

2.2 预备知识

在本节中，我们首先介绍一些后文将会用到的符号和概念，然后给出生成对抗网络的一个一般性的表达式并介绍生成对抗网络的梯度消失问题。

2.2.1 Lipschitz 连续与 Wasserstein 距离

给定两个测度空间 (X, d_X) 和 (Y, d_Y) ，对于函数 $f: X \rightarrow Y$ ，如果存在一个常数 $k \geq 0$ 使得

$$d_Y(f(x_1), f(x_2)) \leq k \cdot d_X(x_1, x_2), \forall x_1, x_2 \in X. \quad (2-1)$$

那么我们称函数 f 是 Lipschitz 连续的。其中最小的常数 k 被称为 f 的 Lipschitz 常数或者最小 Lipschitz 常数，我们把它记作 $k(f)$ 。

在本文以及绝大多数生成对抗网络的文献中，测度 d_X 和 d_Y 都默认采用了欧氏距离。我们用 $\|\cdot\|$ 表示欧氏距离。

两个概率分布之间的一阶的 Wasserstein 距离 W_1 被定义为

$$W_1(P_r, P_g) = \inf_{\pi \in \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \pi} [d(x, y)], \quad (2-2)$$

其中 $\Pi(P_r, P_g)$ 表示所有边缘分布为概率 P_r 和 P_g 的概率测度的集合。它可以被解释成将分布 P_r 变换成分布 P_g 的最小传输代价。我们将最优传输方案记为 π^* 。

令 S_r 和 S_g 分别表示 P_r 和 P_g 的支撑集。有时为了简化起见，在不会引起混淆的情况下，我们也把“两个分布的支撑集不相交”简单说成“两个分布不相交”。

经典的 Kantorovich-Rubinstein (KR) 对偶式^[66] 提供了一个更有效的计算 Wasserstein 距离的方案。该对偶式写到：

$$\begin{aligned} W_1(P_r, P_g) &= \sup_f \mathbb{E}_{x \sim P_r} [f(x)] - \mathbb{E}_{x \sim P_g} [f(x)], \\ \text{s.t. } f(x) - f(y) &\leq d(x, y), \forall x, \forall y. \end{aligned} \quad (2-3)$$

值得注意的是，这个对偶式中的约束，也即公式(2-3)，以为这 f 是 Lipschitz 连续的，并且其 Lipschitz 常数 $k(f) \leq 1$ 。

有趣的是，我们发现该对偶式可以写的更紧凑，也即：

$$\begin{aligned} W_1(P_r, P_g) &= \sup_f \mathbb{E}_{x \sim P_r} [f(x)] - \mathbb{E}_{x \sim P_g} [f(x)], \\ \text{s.t. } f(x) - f(y) &\leq d(x, y), \forall x \in S_r, \forall y \in S_g. \end{aligned} \quad (2-4)$$

该新对偶式的证明在附录2.A.5中给出。

我们可以看到在这个新的对偶式中，Lipschitz 连续的条件被放松了。

2.2.2 生成对抗网络的一般表达式

典型的 GANs 都可以用如下公式形式化表达：

$$\begin{aligned} \min_{f \in \mathcal{F}} J_D &\triangleq \mathbb{E}_{z \sim P_z} [\phi(f(g(z)))] + \mathbb{E}_{x \sim P_r} [\varphi(f(x))], \\ \min_{g \in \mathcal{G}} J_G &\triangleq \mathbb{E}_{z \sim P_z} [\psi(f(g(z)))] \end{aligned} \quad (2-5)$$

其中 P_z 是生成器的输入的分布（通常是一个噪声分布），其维度是 \mathbb{R}^m ； P_r 表示真实数据分布，其维度为 \mathbb{R}^n 。生成函数 $g: \mathbb{R}^m \rightarrow \mathbb{R}^n$ 学习如何将输入的噪声转化成一个和真实数据相同维度的样本，与此同时，判别函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 学习给每个样本一个评分以辨识真假数据。 \mathcal{F} 和 \mathcal{G} 表示的分别是判别函数和生成函数的函数空间，而 $\phi, \varphi, \psi: \mathbb{R} \rightarrow \mathbb{R}$ 是损失度量函数。我们把生成样本的分布记为 P_g 。作为示例，我们在表2-1中列举了几个典型的生成对抗网络模型中 \mathcal{F}, ϕ 以及 φ 的选择。

在这些生成对抗网络中，对于生成样本 $x \in S_g$ 而言，生成器接收到的来自判别器的梯度是：

$$\nabla_x J_G(x) \triangleq \nabla_x \psi(f(x)) = \nabla_{f(x)} \psi(f(x)) \cdot \nabla_x f(x), \quad (2-6)$$

其中第一项 $\nabla_{f(x)} \psi(f(x))$ 是一个一维的标量，影响梯度的大小；第二项 $\nabla_x f(x)$ 是一个和样本 x 同维度的指示着样本优化方向的向量。

我们用 f^* 表示最优判别函数，即， $f^* \triangleq \arg \min_{f \in \mathcal{F}} J_D$ 。此外，我们定义 $\mathring{J}_D(x) \triangleq P_g(x)\phi(f(x)) + P_r(x)\varphi(f(x))$ 以便后面使用。它表示判别器关于样本点 x 的损失函数，有 $J_D = \int \mathring{J}_D(x) dx$ 。

2.2.3 梯度消失问题

梯度消失问题曾被认为是造成生成对抗网络不收敛问题的关键原因。梯度消失问题是指，在判别器被训练到最有的时候，生成器的所拿到的梯度变成零的问题。

[1] 通过将生成器的目标函数替换为 $-\log D$ 的形式解决这个问题。事实上，这个处理仅仅改变了公式2-6中的梯度标量项，即 $\nabla_{f(x)}\psi(f(x))$ 。Least-Squares GAN^[20]致力于解决梯度消失问题，但是其实也是在关注梯度的大小，而没有关注梯度方向的有效性。

[34] 提供了一种新的角度理解梯度消失问题。他们认为 S_r 和 S_g 通常是不相交的，而梯度消失问题源自于一些传统的距离度量在支撑集不相交的情况下病态表现，如：JS 散度在 P_r 和 P_g 不相交的时候将会保持为一个常数。他们因此提出 Wasserstein 距离作为一个新的目标函数^[30]。无论他们的支撑集是否相交，Wasserstein 距离都能很好地度量两个分布之间的距离。

2.3 梯度无信息问题

在本节中，我们关注最优判别函数的梯度方向，也即 $\nabla_x f^*(x)$ ，因为生成器将沿着这个方向更新生成样本。我们指出对于很多距离度量而言， $\nabla_x f^*(x)$ 可能不带有任何关于真实分布的信息。从而，沿这个梯度方向更新生成器，生成分布没有任何保证能收敛到真实分布。我们称这个现象为梯度无信息，并认为该问题是生成对抗网络的训练不稳定性问题以及不收敛性问题的根本源头。

梯度无信息问题和梯度消失问题有着本质的不同，梯度消失问题关注的是梯度中的标量项 $\nabla_{f(x)}\psi(f(x))$ 或者梯度 $\nabla_x J_G(x)$ 的整体大小。而梯度无信息问题关注的是梯度的方向，它由 $\nabla_x f^*(x)$ 决定。这两个问题因此是相互独立的，尽管他们有时候也同时发生。我们在表2-1中总结了一些典型的生成对抗网络中这两种问题的存在性。

接下来，我们从判别函数的函数空间的角度分类讨论梯度无信息问题。我们将说明对于在判别函数空间没有任何限制的生成对抗网络而言，梯度无信息问题普遍存在；而对于判别函数空间存在限制的生成对抗网络而言，梯度无信息问题也可能存在；而如果对判别函数空间施加 Lipschitz 约束的话，梯度无信息问题将普遍不存在。

2.3.1 判别函数空间无限制

对于很多生成对抗网络模型而言，判别函数空间是没有限制的。典型的例子包括：基于 f 散度的生成对抗网络，像原始GAN^[1]、Least-Squares GAN^[20]，以及 f -GAN^[23]都属于此类。

在这些生成对抗网络中，最优判别函数在每个点上的取值，也即 $f^*(x)$ ，是完全独立的，并且它仅仅反映该点局部的概率密度，也即 $P_r(x)$ 和 $P_g(x)$ 。具体而言：

$$f^*(x) = \arg \min_{f(x) \in \mathbb{R}} P_g(x)\phi(f(x)) + P_r(x)\varphi(f(x)), \forall x.$$

因此，对于每个生成样本 x ，如果它并没有被真实样本环绕，那么在 x 的附近的 f^* 的取值将不包含任何关于真实分布的信息。进一步地，可以得知 $\nabla_x f^*(x)$ ，也即样本 x 从最优判别函数得到的梯度，将不包含任何关于真实分布的信息。

所谓的不被真实样本环绕，严格来说可以定义如下：存在一个 $\epsilon > 0$ 使得对于任何的满足 $0 < \|y - x\| < \epsilon$ 的 y ，都有 $y \notin S_r$ 。典型的例子是 S_r 和 S_g 不相交的情形，而这按照[34]的分析，在生成对抗网络的训练中普遍存在。

为了进一步区分梯度无信息问题和梯度消失问题，我们考虑一个理想的情况： S_r 和 S_g 是完全重叠的，并且都包含 n 个离散的点，但是他们的概率密度并不一致。在这种情况下， $\nabla_x f^*(x)$ 是无信息的，但是梯度消失问题并不存在。

2.3.2 判别函数空间中有限制：以 Fisher GAN 为例

一些生成对抗网络模型在判别函数空间中加了限制。典型的例子包括基于积分概率度量（Integral Probability Metric；IPM）的生成对抗网络，如[26, 27, 31]以及Wasserstein GAN^[30]。

我们接下来说明在判别函数空间中加了限制的生成对抗网络也可能会存在梯度无信息问题。以Fisher GAN为例。Fisher GAN采用的是Fisher IPM，按照[26]的结论， μ -Fisher IPM $\mathcal{F}_\mu(P_r, P_g)$ 的最优判别函数的解析表达式：

$$f^*(x) = \frac{1}{\mathcal{F}_\mu(P_r, P_g)} \frac{P_r(x) - P_g(x)}{\mu(x)}, \quad (2-7)$$

其中 μ 是一个支撑集覆盖了 S_r 和 S_g 的分布。值得注意的是， $\frac{1}{\mathcal{F}_\mu(P_r, P_g)}$ 是一个常数。

可以观察到，在 μ -Fisher IPM的最优判别函数中， $f^*(x)$ 几乎是单点独立定义，也仅仅和该点局部的概率密度相关而不反应任何其他点概率密度的信息。按照上一节类似的论证方式，我们可以的结论：对于每个生成样本而言，如果它不被真实样本环绕，那么 $\nabla_x f^*(x)$ 将不能指导生成分布靠近真实分布。

2.3.3 Wasserstein GAN

论文 [35] 中给出过一个关于 Wasserstein 距离最优判别函数的命题。它指出在 Wasserstein 距离的 KR 对偶式中，最优判别函数的关于样本的梯度有如下性质：

命题 2.1 令 π^* 表示公式(2-2)中的最优传输计划， $x_t = tx + (1 - t)y$ 其中 $0 \leq t \leq 1$ 。如果公式(2-3)中的最优判别函数是可导的，并且对任何点 x 而言 $\pi^*(x, x) = 0$ ，那么，有：

$$\mathbb{P}_{(x,y) \sim \pi^*} \left[\nabla_{x_t} f^*(x_t) = \frac{y - x}{\|y - x\|} \right] = 1. \quad (2-8)$$

这个命题表明：1) 对于任何生成样本 x ，存在一个真实样本 y ，使得 $\nabla_{x_t} f^*(x_t) = \frac{y - x}{\|y - x\|}$ 对于任何 x 和 y 的线性插值点 x_t 都成立，也即，在 x 和 y 的连线上，任何一个点的梯度都是由 x 指向 y 的单位向量；2) 这些 (x, y) 点对和最优传输计划 π^* 是匹配的。

它表明 Wasserstein GAN 能够既能克服梯度无信息问题又能克服梯度消失问题。而我们想知道的是为什么 Wasserstein GAN 能够避免梯度消失。为了回答这个问题，我们对 Wasserstein 距离的放松后的对偶式即进行分析，研究新对偶式中最优判别函数的性质。因为公式(2-4)的最优判别函数通常是没有闭式解的，我们通过一个释例来分析这个问题，但我们认为这个结论是一般性的。

令 $Z \sim U[0, 1]$ 是一个服从 $[0, 1]$ 均一分布的变量， P_r 是一个由变量 $(1, Z)$ 代表的二维上的分布，而 P_g 是一个由变量 $(0, Z)$ 表征的另一个二维上的分布。按照公式(2-4)，我们可知以下定义的 f^* 是它的最优判别函数之一：

$$f^*(x) = \begin{cases} 1, & \forall x \in S_r; \\ 0, & \forall x \in S_g. \end{cases} \quad (2-9)$$

值得注意地，在真假分布的支撑集上的 $f^*(x)$ 值就足以确定新对偶式的中的 Wasserstein 距离。因而，即使存在约束 “ $f(x) - f(y) \leq d(x, y)$, $\forall x \in S_r, \forall y \in S_g$ ”，在该对偶式下，最优判别函数也仅仅只在 S_r 和 S_g 上有定义。对于任何一个生成样本 x ，如果它是孤立的或者是在边界上的，也即，不存在任何一个 $\epsilon > 0$ 使得对于任意的满足 $0 < \|y - x\| < \epsilon$ 的 y 都有 $y \in S_r \cup S_g$ ，那么 f^* 在该点的梯度是未定义的，从而也就不能提供任何关于真实分布的信息。我们也可以考虑更极端的情况，比如 S_g 是若干个孤立的点，此时，结论会更加清晰。

以上所讨论的所有情况均暗示着 Lipschitz 条件在解决梯度无信息问题时的重要性。因此，我们将在下一节中，研究一般的带有 Lipschitz 条件的生成对抗网络。我们将会发现，当损失度量函数发生变化时，Lipschitz 作用下的生成对抗网络将

不再是在测量 Wasserstein 距离，但它们依旧是一个好的距离度量，并且能在现实中工作的很好。

2.4 Lipschitz GANs

Lipschitz 连续最近在生成对抗网络中变得流行起来。人们发现把 Lipschitz 连续作为约束加入到判别器中能提高生成对抗网络的训练稳定性，同时也能带来样本质量的提升^[29, 30, 36, 45, 67]。

在本节中，我们分析带有 Lipschitz 约束的生成对抗网络的一般形式，具体而言，我们为判别函数引入了一个关于其 Lipschitz 常数的二次惩罚，以理论上分析这类生成对抗网络的性质。具体而言，我们将 Lipschitz GANs (LGANs) 定义为：

$$\begin{aligned} & \min_{f \in \mathcal{F}} \mathbb{E}_{z \sim P_z} [\phi(f(g(z)))] + \mathbb{E}_{x \sim P_r} [\varphi(f(x))] + \lambda \cdot k(f)^2, \\ & \min_{g \in \mathcal{G}} \mathbb{E}_{z \sim P_z} [\psi(f(g(z)))]. \end{aligned} \quad (2-10)$$

我们假设其中的损失度量函数 ϕ 和 φ 满足以下条件，我们后面会介绍这个假设的重要意义：

$$\begin{cases} \phi'(x) > 0, \phi''(x) \geq 0, \\ \varphi'(x) < 0, \varphi''(x) \geq 0, \\ \exists a, \phi'(a) + \varphi'(a) = 0. \end{cases} \quad (2-11)$$

生成对抗网络中常用的损失度量函数很多都满足这个假设。如在 WGAN 中， $\phi(x) = \varphi(-x) = x$ 。它就满足公式(2-11)。除此以外，还有很多其他的例子都满足这个条件，比如 $\phi(x) = \varphi(-x) = -\log(\sigma(-x))$ ， $\phi(x) = \varphi(-x) = x + \sqrt{x^2 + 1}$ ，以及 $\phi(x) = \varphi(-x) = \exp(x)$ 。其中， $\phi(x) = \varphi(-x) = -\log(\sigma(-x))$ 是原始 GAN 的目标函数。而同时，也有不少生成对抗网络的损失度量函数不满足我们的假设，如：[20] 中用到的二次惩罚损失 (square loss)，以及 [21, 24, 36] 中用到的 hinge loss。

如果我们想导出一个新的 LGANs 的目标函数，通常而言，我们可以直接令 $\phi(x) = \varphi(-x)$ ，然后为 ϕ 寻找一个单调递增而导数不减的函数。其实，如果找到了几组这样的函数之后，它们的任何一个线性组合都满足公式(2-11)。我们在图2-13中给出了这些损失度量函数的图示，以方便读者直观感受它们的性质。

值得注意的一点， $\phi(x) = \varphi(-x) = -\log(\sigma(-x))$ 是原始 GAN 的目标函数，而我们知道原始 GAN 存在梯度无信息问题，但是我们接下来会说明，当我们如公式(2-10)那样给它加入 Lipschitz 惩罚之后，该模型作为 LGANs 中的一员就不再存在梯度无信息问题了。

2.4.1 理论分析

我们接下来陈述关于 LGANs 的一些理论上的性质。首先，我们考虑最优判别函数的存在性和唯一性。

定理 2.2 假设 ϕ 和 φ 满足假设(2–11)，则最优判别函数一定存在。而如果 ϕ 和 φ 其中任何一个满足严格凸，则最优判别函数唯一。

值得注意的，WGAN 中 $\phi(x) = \varphi(-x) = x$ ，满足公式(2–11)，但 ϕ 和 φ 均不严格凸，按照定理2.2的结论，它的最优判别函数存在但是并不唯一。而我们其实可以简单地验证，对于 WGAN 而言，假设 f^* 是它的最优判别函数之一，那么对于任何的实数 α ， $f^* + \alpha$ 都是它的一个最优判别函数。也即，WGAN 中的最优判别函数是有一个自由的偏置的。这个自由的偏置在某些实验场景下也给人们填了不少麻烦。而 LGANs 中，只要 ϕ 和 φ 中的任何一个满足严格凸，最优判别函数就唯一，因此不会有这个困扰。

接下来的这个定理可以认为是命题2.1从 WGAN 到 LGANs 的一个拓展。

定理 2.3 假设 $\phi'(x) > 0$ ， $\varphi'(x) < 0$ ，最优判别函数存在 f^* 且平滑，那么我们有：

- (a) 对于任意的 $x \in S_r \cup S_g$ ，如果它满足 $\nabla_{f^*(x)} \mathring{J}_D(x) \neq 0$ ，则必定存在一个与 x 不同的点 $y \in S_r \cup S_g$ ，使得 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ ；
- (b) 对于任意的 $x \in S_r \cup S_g - S_r \cap S_g$ ，必定存在一个与 x 不同的点 $y \in S_r \cup S_g$ ，使得 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ ；
- (c) 如果 $S_r = S_g$ 且 $P_r \neq P_g$ ，那么必定存在一组在 (x, y) ，其中 x 和 y 都属于 $S_r \cup S_g$ 并且 $y \neq x$ ，使得 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ ，同时 $\nabla_{f^*(x)} \mathring{J}_D(x) \neq 0$ ；
- (d) 在目标函数 $J_D + \lambda \cdot k(f)^2$ 下， P_g 和 f^* 存在一个唯一的纳什均衡。在均衡点处， $P_r = P_g$ 且 $k(f^*) = 0$ 。

定理的证明在附录2.A.2中给出。这个定理陈述了 LGANs 的基本性质，纳什均衡的唯一性和存在性、纳什均衡出 $P_r = P_g$ 、以及相互绑定关系的存在性（即，两个不同的点 x 和 y 满足 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ ）。纳什均衡的性质保证所定义的目标函数是个良定义的距离测度，而后者则是消除梯度无信息问题的核心。

我们指出，惩罚判别函数的 Lipschitz 常数是对于一般性的损失度量函数得出结论(c)和(d)的关键。其内部原因是，在一般的损失度量函数下， $\nabla_{f^*(x)} \mathring{J}_D(x) = 0$ 可能在 $P_r(x) \neq P_g(x)$ 的情况下成立。而惩罚 $k(f)$ 能使得纳什均衡点处必定有 $P_r = P_g$ 。

在 WGAN 中，最小化 $k(f)$ 不是必须的。但是同时也因为 $k(f)$ 没有被最小化， $\nabla_x f^*(x)$ 在收敛时（即 $P_r = P_g$ 时）并不保证为 0。反而，当 $P_r = P_g$ 时，任何满足

1-Lipschitz 的函数都是 WGAN 的最优判别函数之一。这意味着，即使是在 WGAN 中，最小化 $k(f)$ 也是有好处的。

2.4.2 相互绑定关系的进一步结论

从定理2.3我们知道，对于任意的点 x ，只要 $\dot{J}_D(x)$ 关于 $f^*(x)$ 的梯度不为零，那么 $f^*(x)$ 的值必定被另一个点 y 绑定 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ 。我们接下来进一步澄清：在一个相互绑定关系必定同时涉及真实样本和生成样本。严格说来：

定理 2.4 在定理2.3的条件下，我们进一步有：

- 1) 对于任意的点 $x \in S_g$ ，如果 $\nabla_{f^*(x)} \dot{J}_D(x) > 0$ ，那么必定存在一个点与 x 不同的点 $y \in S_r$ 使得 $f^*(y) - f^*(x) = k(f^*) \cdot \|y - x\|$ ，并且 $\nabla_{f^*(y)} \dot{J}_D(y) < 0$ ；
- 2) 对于任意的点 $y \in S_r$ ，如果 $\nabla_{f^*(y)} \dot{J}_D(y) < 0$ ，那么必定存在一个点与 y 不同的点 $x \in S_g$ 使得 $f^*(y) - f^*(x) = k(f^*) \cdot \|y - x\|$ ，并且 $\nabla_{f^*(x)} \dot{J}_D(x) > 0$ 。

以上定理背后的直觉是来自同一个分布的样本不会因相互绑定而破坏 $\dot{J}_D(x)$ 的最优性，所以当存在一个严格的相互绑定关系的话，也即，如果绑定关系中存在一个点满足 $\nabla_{f^*(x)} \dot{J}_D(x) \neq 0$ ，那么它必定同时包含真实样本和生成样本。值得注意的是，只要不是重叠的情况，所有生成样本都满足 $\nabla_{f^*(x)} \dot{J}_D(x) > 0$ ，而所有真实样本都满足 $\nabla_{f^*(y)} \dot{J}_D(y) < 0$ 。

一个绑定关系中可能会包含一系列的真实样本和生成样本，他们都是互相绑定的关系。在 Lipschitz 连续的条件下，在 f^* 的值空间中的绑定关系是连接真实分布和生成分布的基本单元。每一个生成样本，只要 $\nabla_{f^*(x)} \dot{J}_D(x) \neq 0$ ，那么它必定存在于至少一个相互绑定关系中。

接下来我们进一步阐述相关绑定关系的含义，并说明相互绑定关系让其中所涉及的所有点的梯度都变得有意义。

2.4.3 相互绑定关系的蕴意

命题2.1陈述了在 WGAN 中最优判别函数的梯度有 $\nabla_{x_t} f^*(x_t) = \frac{y-x}{\|y-x\|}$ 的性质，我们接下来说明这样一个性质其实是相互绑定关系的直接的结果。我们把这个结论形式化地写成如下定理：

定理 2.5 假设函数 f 是可导的，并且其 Lipschitz 常数为 k ，对于任意的 x 和 y ，如果 $y \neq x$ 并且 $f(y) - f(x) = k \cdot \|y - x\|$ ，那么就有：对于任何的 $x_t = tx + (1-t)y$ ，如果 $0 \leq t \leq 1$ ，则 $\nabla_{x_t} f(x_t) = k \cdot \frac{y-x}{\|y-x\|}$ 。

也就是说，如果两个点在 $f(y) - f(x) = k \cdot \|y - x\|$ 的意义下相互绑定，那么在 f 的值空间中就存在一条从 x 到 y 的直线，直线上的任何一点都满足梯度大小为该函数的最大梯度值 k ，而梯度的方向都是从 x 指向 y 。证明见附录2.A.4。

结合定理2.3和2.4，我们有：当生成分布和真实分布不相交时，最优判别函数的在生成样本上的梯度都指向一个真实样本，这保证生成分布每步都在朝着真实分布移动。事实上，定理2.3在此基础上，提供了进一步的保证。性质（b）表明对于任意的生成样本点 x ，只要它不在真实分布的支撑集上，那么它的梯度也必定指向某个真实样本。而在完全重叠的情况下，性质（c）保证，除非两个分布已经完全相等，否则比如存在至少一组点 x 和 y ，它们相互绑定，并且 x 的梯度指向 y 。最后，性质（d）保证，唯一的纳什均衡点在于 $P_r = P_g$ ，而此时对于所有的生成样本 $\nabla_x f^*(x) = 0$ 。

2.5 Lipschitz 约束的实现

实现 Lipschitz 约束的典型方法包括：权重裁剪^[30]，梯度惩罚^[35]，Lipschitz 惩罚^[37]，以及谱归一化^[36]。其中，权重裁剪已经被证明是过度限制网络的表达能力，以至于会陷于很差的局部最优解^[37]。本章中，我们将讨论其余几种 Lipschitz 约束的实现方法的优点与不足，最后提出最大梯度惩罚作为本文中实现 Lipschitz 约束的方式。

2.5.1 现有方法

按照[68]的结论，函数的 Lipschitz 常数等于它的最大梯度的幅度。梯度惩罚^[35]和 Lipschitz 惩罚^[37]是基于这个思想的。他们通过梯度惩罚的方式实现 Lipschitz 约束。他们分别引入如下正则项：

$$L_{gp} = -\frac{\rho}{2} \mathbb{E}_{x \sim P_{\hat{x}}} [(\|\nabla_x f(x)\| - 1)^2], \quad (2-12)$$

$$L_{lp} = -\frac{\rho}{2} \mathbb{E}_{x \sim P_{\hat{x}}} [(\max(0, \|\nabla_x f(x)\| - 1))^2]. \quad (2-13)$$

这里 $P_{\hat{x}}$ 表示由采样策略决定的样本分布。Lipschitz 惩罚的提出是因为观察到梯度惩罚中的不足，即：，1-Lipschitz 仅要求梯度不大于 1，因此不应该惩罚梯度小于 1 的点的梯度。因此，他们对梯度惩罚做了简单的修正，改为仅仅惩罚梯度大于 1 的点。

而还有一个事实是，任何一个线性变换， $h(x) = Wx$ ，的 Lipschitz 常数等于它权重矩阵的最大特征值。谱归一化^[36]基于该思想，他们提出将每一层的权重矩阵

除上它的最大特征值，从而使得没一个线性变换都是 1-Lipschitz 的，即：

$$\bar{W}_{SN} = W / \sigma(W), \quad (2-14)$$

其中， $\sigma(W)$ 表示 W 的最大特征值。此时，如果激活层，也即神经网络中的非线性层，也满足 1-Lipschitz，那么整个网络就满足 1-Lipschitz。而我们通常所用的激活函数，如：RELU，Tanh，Sigmoid 均满足这个性质。

值得注意的是，谱归一化是一个硬的全局约束，而梯度惩罚和 Lipschitz 惩罚是软的局部正则。

2.5.2 全局约束的不必要性

通常梯度惩罚和 Lipschitz 惩罚将 $P_{\hat{x}}$ 选为由真假样本的随机线性插值点形成的分布。而为什么这个选择是一个合理的选择目前还不是很明确，人们通常认为这是一个实践中的迫不得已的选择，并且倾向于认为因为没有实现全局约束故而是有害的。接下来，我们为该选择提供理论上的辩护。

引理 2.6 令 $S_{\hat{x}}$ 表示真假样本线性插值点的分布的支撑集，我们有：在 $S_{\hat{x}}$ 上实现 1-Lipschitz 足以使命题2.1成立。

为了得到以上结论，我们需要深入分析 Wasserstein 距离的新对偶式（公式(2-4)）。值得注意的是，新对偶式中的约束是 KR 对偶式（公式(2-3)）中约束的一个子集，而其中对于的部分对 Wasserstein 距离的计算没有影响。更重要的是，任何一个新对偶式中的最优判别函数均对应于 KR 对偶式中的 KR 对偶式的一个最优判别函数，所对应的的最优判别函数保持其在 S_r 和 S_g 上的取值不变。

因此，新对偶式的任何一个最优判别函数都满足以下 Wasserstein 距离的关键性质^[35, 66]：

引理 2.7 令 π^* 表示公式(2-2)中的最优传输计划， f^* 表示公式(2-4)中的最优判别函数，则

$$P_{(x,y) \sim \pi^*} [f^*(x) - f^*(y)] = d(x, y) = 1. \quad (2-15)$$

其实，基于公式(2-15)和最优判别函数 Lipschitz 连续的性质即可得出命题2.1的结论。而我们可以进一步注意到证明过程中仅仅需要 f^* 的局部（在 $S_{\hat{x}}$ 上）的 Lipschitz 连续性。因此， $S_{\hat{x}}$ 的局部 Lipschitz 连续性是足够得出命题2.1的结论的。最后， f^* 在 $S_{\hat{x}}$ 局部 Lipschitz 连续是公式(2-4)的一个充分条件。因此，只要 f^* 在 $S_{\hat{x}}$ 上满足局部 Lipschitz 连续，命题2.1的结论即可成立。同理，我们可以论证在 Lipschitz GANs 中，惩罚在 $S_{\hat{x}}$ 上的 Lipschitz 常数即是足够的。

引理2.6意味着，对于生成对抗网络的训练而言，约束全局 Lipschitz 其实是没必要的。接下来我们说明，当前的局部 Lipschitz 约束方法存在冗余约束，会造成最优解的偏移。

2.5.3 惩罚法中的冗余约束

梯度惩罚和 Lipschitz 惩罚通过惩罚法约束 Lipschitz 常数。而惩罚法是一个软约束，它所得到 Lipschitz 常数通常存在一定的到目标常数的偏移。

具体而言，我们考虑如下目标函数，并假设我们可以直接优化 Lipschitz 常数 k :

$$L(k) = W_1(P_r, P_g, k) - \frac{\rho}{2}(k - 1)^2. \quad (2-16)$$

其中 $W_1(P_r, P_g, k)$ 表示在 k -Lipschitz 约束下的 Wasserstein 距离，也即， $\mathbb{E}_{x \sim P_r}[f(x)] - \mathbb{E}_{x \sim P_g}[f(x)]$ s.t. $\|f\|_L \leq k$.

首先，有一个显然的结论是 $W_1(P_r, P_g, k) = kW_1(P_r, P_g)$ 。当 P_r 和 P_g 固定式， $W_1(P_r, P_g)$ 是一个定值。因此 $L(k)$ 是关于 Lipschitz 常数 k 的一个二次函数，我们有： k 的最优解 $k^* = \frac{W_1(P_r, P_g)}{\rho} + 1$ 。注：将 $(k - 1)^2$ 换成 $\max\{0, k - 1\}^2$ 得到的 k 的最优解不变。

从上面我们可以看出，当 ρ 很小而 $W_1(P_r, P_g)$ 较大时，惩罚法得到的 Lipschitz 常数将远大于 1。在这种情况下，其实作为实现 Lipschitz 约束的方法，他们引入了额外的冗余的约束。

假设，这里得到的 Lipschitz 常数为 100，那么梯度惩罚和 Lipschitz 惩罚都会惩罚梯度幅度在 1 到 100 之间的点。而我们知道一个函数的 Lipschitz 常数，由其最大梯度的幅度决定，因此，对其他非最大梯度值的梯度惩罚是冗余的。我们将在试验中看到这些冗余约束会造成问题的最优解的偏移，并使得最优解的性质被破坏。

[37] 指出 Lipschitz 惩罚对应于带正则的 Wasserstein 距离。但不幸的是，带正则的 Wasserstein 距离的最优解就发生了偏移，并会造成最优传输计划上的模糊^[69]。因此，和我们这里的分布并不矛盾，反而是统一的。

2.5.4 最大梯度惩罚

LGANs 要求对判别函数的 Lipschitz 常数进行惩罚。然而，现有方法中：如何通过谱归一化惩罚 Lipschitz 常数还是一个有待进一步研究的问题；此外，谱归一化约束全局 Lipschitz 常数，按照我们的分析是没有必要的。而梯度惩罚和 Lipschitz 惩罚中存在冗余约束。

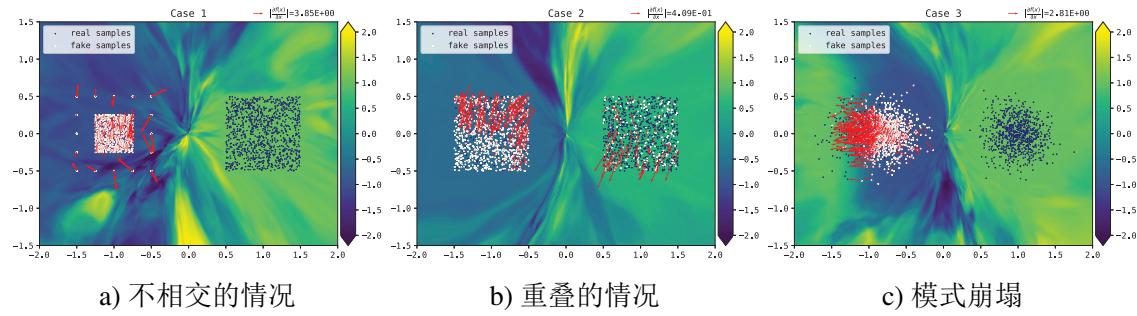


图 2-1 梯度无信息问题的实践表现：噪声梯度。局部贪心的梯度导致模式崩塌。

Figure 2-1 Practical behaviors of gradient uninformativeness: noisy gradient. Local greedy gradient leads to mode collapse.

为了直接惩罚 Lipschitz 常数，我们通过采样的方式近似当前的最大梯度：

$$k(f) \simeq \max_x \|\nabla_x f(x)\|. \quad (2-17)$$

然后把最大梯度的长度当作为 Lipschitz 常数的等价量进行惩罚，我们称该方法为最大梯度惩罚。

实践过程中，我们沿用 [35] 中的方法，样本 x 从真实样本和生成样本的插值点中随机得到。我们考虑过批量采用对最大梯度估计的稳定性问题。对此我们尝试了追踪记录历史采样中取得过最大梯度的点。具体操作如下，我们维护一个队列 S_{\max} ，其中包含当前样本中最大的 k 个梯度的样本点，其初始值可以设为随机样本。然后，每次计算最大梯度时把 S_{\max} 作为额外的点已补偿批量采样可能无法采到最大点的问题。在每次计算完每个样本点的梯度后更新 S_{\max} 。按照我们的实验，是否增加额外的 S_{\max} 通常影响不大。

2.6 实验分析

在本节中，我们通过实验分析梯度无信息问题，Lipschitz 条件各实现方法的差异，以及 Lipschitz GANs 的实践表现。

2.6.1 梯度无信息问题的实践表现

按照我们的分析， $\nabla_x f^*(x)$ 在大多数传统的生成对抗网络中是无信息的。这里我们坍缩梯度无信息问题的实践表现。实际上，任何在判别函数空间中不加约束的生成对抗网络，在梯度有无信息问题上是几乎等价的。这里，为了让可视化做的更容易，我们选取了 Least-Squares GAN 作为我们实验的对象。因为它的最优判别函数都去有限值（原始 GAN 则是趋于无限），比较容易可视化。

实验结果如图2-1所示，我们发现当判别器训练的足够好后（也即趋于最优判别函数时），判别函数的梯度是相当随机的。我们认为这应该就是梯度无信息问题的实践表现。给定最优判别函数在其他点上的不确定性，最优判别函数的梯度 $\nabla_x f^*(x)$ 受网络结构和超参数等的影响。为了进一步验证，我们做了一系列不同参数下的实验，这部分结果我们放在附录2.B。

在小2.3中，我们讨论梯度无信息的时候，假设生成样本不被真实样本环绕。事实上，最优判别函数的梯度问题是更一般的存在的。比如说，在图2-1b的例子中，真实分布和生成分布各自均匀地分布在两个区间内，但是密度不一样。这里最优判别函数在两个区间内都是常数，而在这两个区间外则是未定义的。所以理论上，最优判别函数在这两个区间内的梯度为零，而在边界和外面则是未定义的。这种情况下，它们其实也表现为噪声梯度。注意到，在完全重叠的情况下，最优判别函数的梯度可能也是病态的。如图2-1c所示，由于最优判别函数的局部性，梯度的也具有局部性，这可能是导致原始 GAN 以及类似的不带约束的生成对抗网络模式崩塌的原因。

2.6.2 Lipschitz 约束的实现方法对比

鉴于梯度惩罚和 Lipschitz 惩罚本质上以及实验结果上的高度相似性，我们这里将他们统一称为梯度惩罚。该部分试验中，我们采用 Wasserstein 距离作为目标函数，以在不引入 Lipschitz GAN 的前提下，讨论不同约束方法的差异。我们为梯度惩罚和最大梯度惩罚设置相同的目标 Lipschitz 常数，以使得对比变量得到控制。

为了直观对比不同 Lipschitz 约束的实现方法，我们首先在简单的二维数据上做对比实验。在该实验中，我们将 P_r 和 P_g 分别固定为二维空间中的两个随机点，然后我们分别用谱归一化、梯度惩罚、最大梯度惩罚训练判别器。按照理论，生成样本的梯度应该分别指向两个真实样本。我们通过检查生成样本的梯度的方向来验证各个方法是否能正确稳定收敛到最优解。

我们发现，在某些情况下，谱归一化无法解出最优解。我们在图2-2中给出了一个示例。谱归一化在这个例子上快速收敛但并没有收敛到最优解。我们目前无法断定为什么谱归一化会出现该问题。我们考虑过是否是因为全局 Lipschitz 约束太强导致网络能力太弱而造成该问题，但我们将网络能力大幅加强后依然发现谱归一化无法收敛到最优解。我们也怀疑是否是特征值近似的幂迭代（power iteration）算法出现了局部最优问题。但该问题很难验证，我们尝试了增加幂迭代的迭代次数，该问题依然存在。该问题似乎也不是因为学习率过高导致的，我们尝试了以很小的学习率长时间训练，但最终结果不变。至于该问题的真正原因，我们留以

后再进一步探索。但我们发现谱归一化确实存在一定的问题，不仅在简单的二维数据上，在真实数据上我们也经常发现梯度惩罚系的方法能得到较好的结果，而谱归一化样本质量无法达到正常范围。

在图2-2中，我们也注意到，梯度惩罚在训练过程中判别函数无法收敛到最优解且存在大幅波动。与之相对的，最大梯度惩罚能稳定收敛到最优判别函数。

我们进一步在高维数据上做对比实验。真实的高维数据集样本量过大，我们实验中发现判别函数很难收敛到最优解。因此，我们将 P_r 和 P_g 分别固定为十张真实图片和十张噪声图片。然后，和上面的实验一样，我们将判别器训练到趋于最优，然后检查判别函数的梯度方向，以对比不同方法的结果的差异。对于高维数据，用梯度渐进的方法可视化梯度方向，如图所示2-3。

从结果看来，最大梯度惩罚在高维数据中也能收敛到最优判别函数。而梯度惩罚的梯度并不能清晰地指向某个真实样本，它的梯度像是几个目标图片的混合状态，并且似乎有一定程度的模式崩塌（图中有多只猫、多只鸟），这也表明了冗余约束确实存在坏的影响。

以上实验中我们能看出最大梯度惩罚和梯度惩罚的明显差异。但在，数据规模较大时，我们在实验中很难看出最大梯度惩罚和梯度惩罚的区别。我们发现该问题可能是因为神经网络表达能力有限造成的，无论采用最大梯度惩罚还是梯度惩罚均无法达到最优解。此外，在一些实验中，我们发现如果采用梯度惩罚，训练会发散，但是如果采用最大梯度惩罚，训练则能正常收敛，比如说当 $\phi(x) = \varphi(-x) = \exp(x)$ 时。

2.6.3 验证 LGANs 中最优判别函数的梯度性质

LGANs 的一个重要的理论上的好处是保证任何生成样本的梯度都指向一个真实样本。这里，我们从实验中验证这个结论。我们测试了一组满足公式(2-11)的损失度量函数：(a) $\phi(x) = \varphi(-x) = x$; (b) $\phi(x) = \varphi(-x) = -\log(\sigma(-x))$; (c) $\phi(x) = \varphi(-x) = x + \sqrt{x^2 + 1}$; (d) $\phi(x) = \varphi(-x) = \exp(x)$ 。我们在以下两组数据中做了测试：二维合成数据以及高维的真实世界的数据。对于二维的合成数据，我们令真实分布由两个高斯分布组成，而生成分布是一个高斯分布。这些高斯分布具有相同的标准差，同时，生成分布的高斯更靠近真实分布中两个高斯分布中的一个，如图2-4所示。在高维数据的实验里，我们选用 CIFAR-10 作为我们的测试数据。为了让解最优判别函数较为容易，我们将真实分布定义为由十张 CIFAR-10 图片组成的分布，而生成分布固定为十张噪声图片。我们将生成分布固定，因为在研究最优判别函数的性质的时候，生成分布是没有必要去优化的，固定生成分

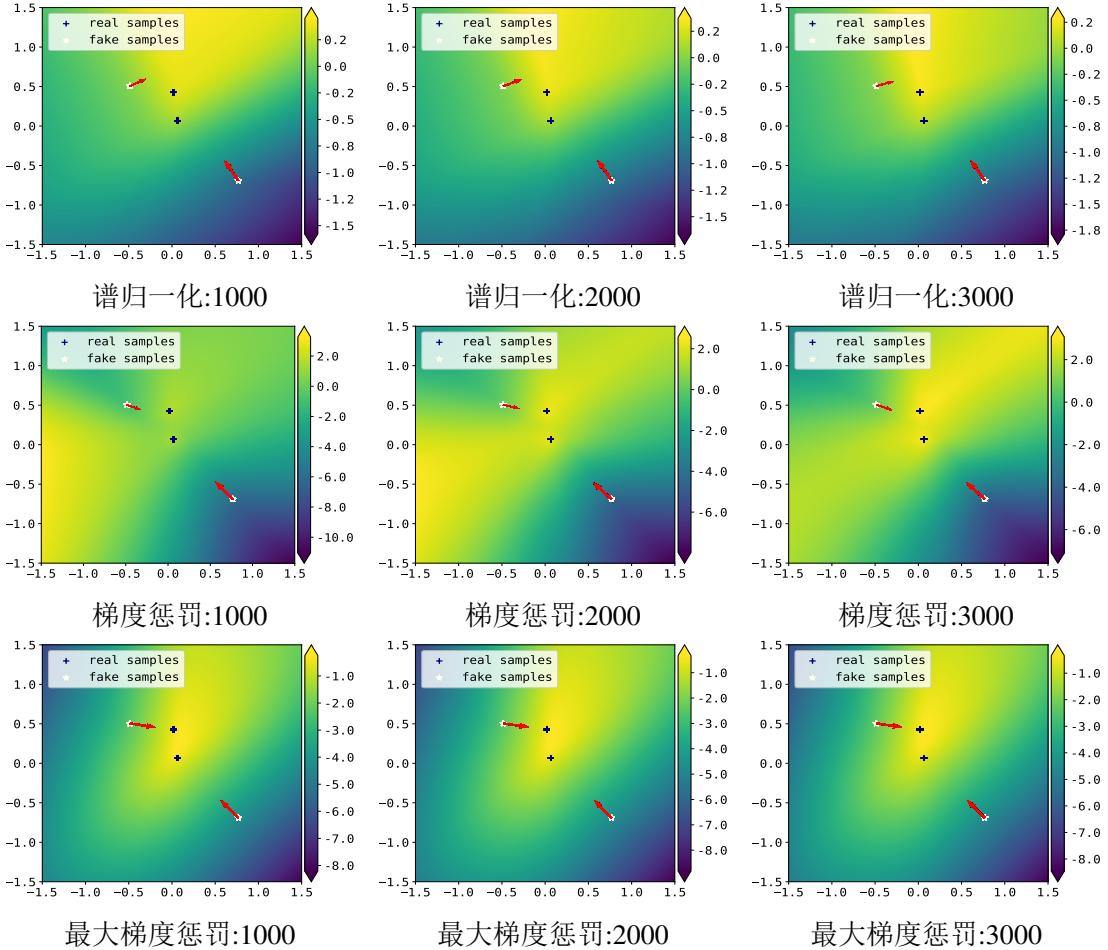
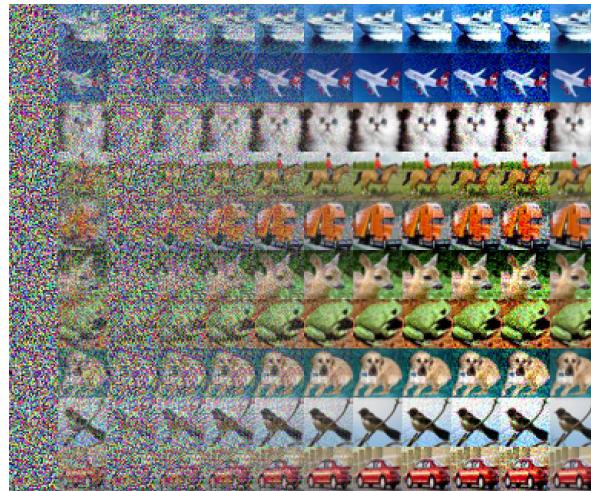


图 2-2 令 P_r 和 P_g 分别固定为二维空间中的两个随机点，我们分别用谱归一化、梯度惩罚、最大梯度惩罚训练判别器，结果如图所示。方法名后面的数字表示迭代次数。图中的箭头可是花了梯度的方向。从图中我们可以看出，(i) 谱归一化没有解到最优判别函数；(ii) 梯度惩罚的判别函数存在波动；(iii) 最大梯度惩罚稳定地收敛到最优判别函数。

Figure 2-2 With P_r and P_g both being two random sampled points in 2-dimensional space, we training the discriminator using SN, GP and MAXGP, respectively. The number after the name of the methods is the corresponding iteration number. The arrows in the figures indicate the gradient directions. From the results, we notice that: (i) SN in this case failed to achieve the optimal discriminator; (ii) the discriminator trained with GP is oscillatory; (iii) MAXGP stably converged to the optimal.



a) 梯度惩罚



b) 最大梯度惩罚

图 2-3 梯度惩罚（上）和最大梯度惩罚（下）的对比。该实验中真实分布和生成分布分别由十张真实图片和十张噪声图片组成。每行的最左侧为生成样本 $x \in S_g$ ；第二列为对应样本的梯度 $\nabla_x f^*(x)$ ；中间的其它图片为 $x + \epsilon \cdot \nabla_x f^*(x)$, 其中 ϵ 不断增大；最后一列是为真实分布中最靠近该梯度方向的样本 $y \in S_r$ 。如图所示，在最大梯度惩罚下，生成样本的梯度方向会经过一个真实样本，和理论相吻合；而梯度惩罚在实验中无法收敛到最优解且表现出大幅波动。

Figure 2-3 Comparison between gradient penalty (top) and maximum gradient penalty (bottom). P_r and P_g consist of ten real and noise images, respectively. The leftmost in each row is a $x \in S_g$ and the second is its gradient $\nabla_x f^*(x)$. The interiors are $x + \epsilon \cdot \nabla_x f^*(x)$ with increasing ϵ , and the rightmost is the nearest $y \in S_r$. We can see that with maximum gradient penalty, the gradient of each generated sample pass through a real sample, which is consistent with the theoretical result. However, gradient penalty can not achieve the theoretical optimal discriminative function. And in experiments, we notice that with gradient penalty, the discriminative function is highly unstable.

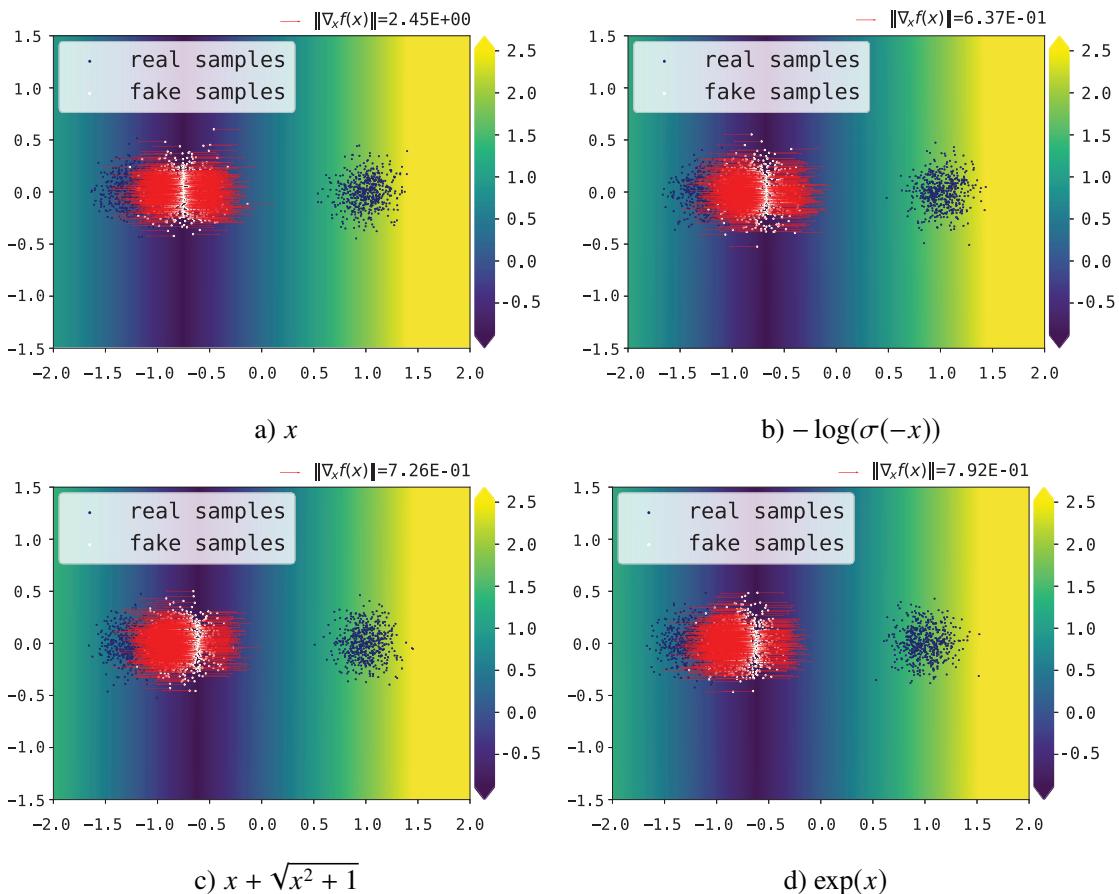


图 2-4 在二维数据上验证 LGANs 的最优判别函数的梯度方向指向真实样本。

Figure 2-4 Verifying that $\nabla_x f^*(x)$ in LGANs point towards real samples with two-dimensional data.

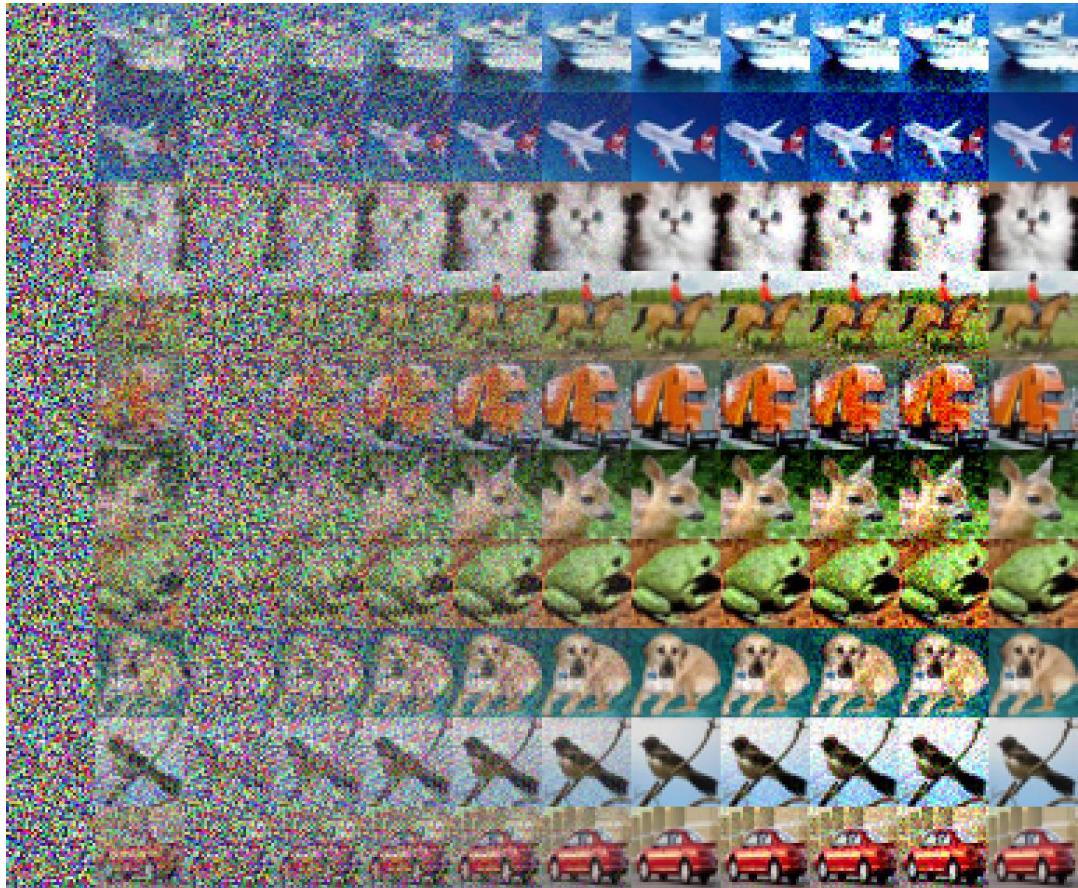


图 2-5 在 CIFAR-10 数据上可视化最优判别函数的梯度。每行的最左侧为生成样本 $x \in S_g$ ；第二列为对应样本的梯度 $\nabla_x f^*(x)$ ；中间的其它图片为 $x + \epsilon \cdot \nabla_x f^*(x)$ ，其中 ϵ 不断增大；最后一列是为真实分布中最靠近该梯度方向的样本 $y \in S_r$ 。

Figure 2-5 $\nabla_x f^*(x)$ gradation with CIFAR-10. The leftmost in each row is a $x \in S_g$ and the second is its gradient $\nabla_x f^*(x)$. The interiors are $x + \epsilon \cdot \nabla_x f^*(x)$ with increasing ϵ , and the rightmost is the nearest $y \in S_r$.

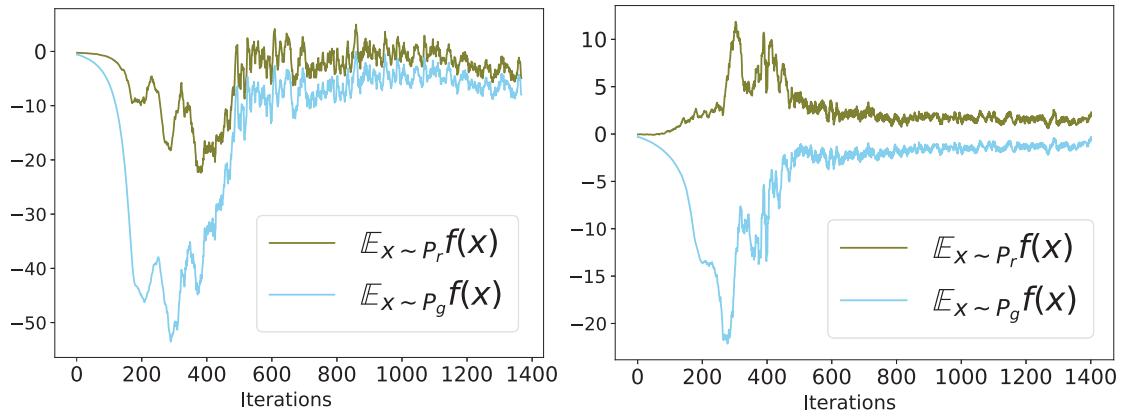


图 2-6 在 LGANs 中判别函数更加稳定。左图：WGAN；右图：LGANs。

Figure 2-6 $f(x)$ in LGANs is more stable. Left: WGAN. Right: LGANs.

布可以为我们省去很多麻烦。

结果分别呈现在图2-4和图2-5中。在图2-4中可以看到每个生成样本的梯度都是指向一个真实样本。因为高维数据的梯度难以直接可视化，我们将原图、梯度、原图在梯度方向上的渐变图可视化出来，并为这一组图寻找真实图片中最接近的图，如图2-5所示。图中，最左侧的是原图 x ，第二列的是它的梯度 $\nabla_x f(x)$ ，从第三列到倒数第二列是 $x + \epsilon \cdot \nabla_x f(x)$ ，其中 ϵ 不断增大，而最右边是真实图片中与原图在梯度方向上最近接的一张图片。在高维数据的这组实验中，不同的损失度量函数的最终结果都很类似。

2.6.4 最优判别函数的唯一性以及稳定性

Wasserstein GAN 所采用的损失度量函数是 Lipschitz GANs 中的一个特殊形式：它唯一一个 ϕ 和 φ 的二阶导都为零的情况。这导致的结果是，在 Wasserstein GAN 中 f^* 有一个自由度，也即给定一个最优判别函数 f^* ，对于任何实数 α 都有 $f^* + \alpha$ 也是它的一个最优判别函数。而这在实践过程中，通常表现为判别函数在训练过程中存在震荡。这个震荡似乎会对 WGAN 的训练造成一些坏的影响：[8] 以及^[68] 在论文中引入了一个正则项来防止判别函数在训练过程中出现波动。也此相对的，LGANs 里面的其他任何一个例子都有：最优判别函数是唯一的。因而不会出现震荡的问题。我们在图2-6中对比了这两个性质在实际训练中的区别。

2.6.5 用无监督图片生成任务做基准测试

为了量化地比较 LGANs 里面的不同目标函数，我们用无监督图片生成任务对它们做基准测试。在这部分的实验中，我们还加入了 hinge loss $\phi(x) = \varphi(-x) =$

表 2-2 在无监督图片生成任务下的量化对比

Table 2-2 Comparisons with unsupervised image generation.

| Objective | CIFAR-10 | | Tiny ImageNet | |
|----------------------|-----------------------------------|------------------------------------|-----------------------------------|------------------------------------|
| | IS | FID | IS | FID |
| x | 7.68 ± 0.03 | 18.35 ± 0.12 | 8.66 ± 0.04 | 16.47 ± 0.04 |
| $\exp(x)$ | 8.03 ± 0.03 | 15.64 ± 0.07 | 8.67 ± 0.04 | 14.90 ± 0.07 |
| $-\log(\sigma(-x))$ | 7.95 ± 0.04 | 16.47 ± 0.11 | 8.70 ± 0.04 | 15.05 ± 0.07 |
| $x + \sqrt{x^2 + 1}$ | 7.97 ± 0.03 | 16.03 ± 0.09 | 8.82 ± 0.03 | 15.11 ± 0.06 |
| $(x + 1)^2$ | 7.97 ± 0.04 | 15.90 ± 0.09 | 8.53 ± 0.04 | 15.72 ± 0.11 |
| $\max(0, x + 1)$ | 7.91 ± 0.04 | 16.52 ± 0.12 | 8.63 ± 0.04 | 15.75 ± 0.06 |

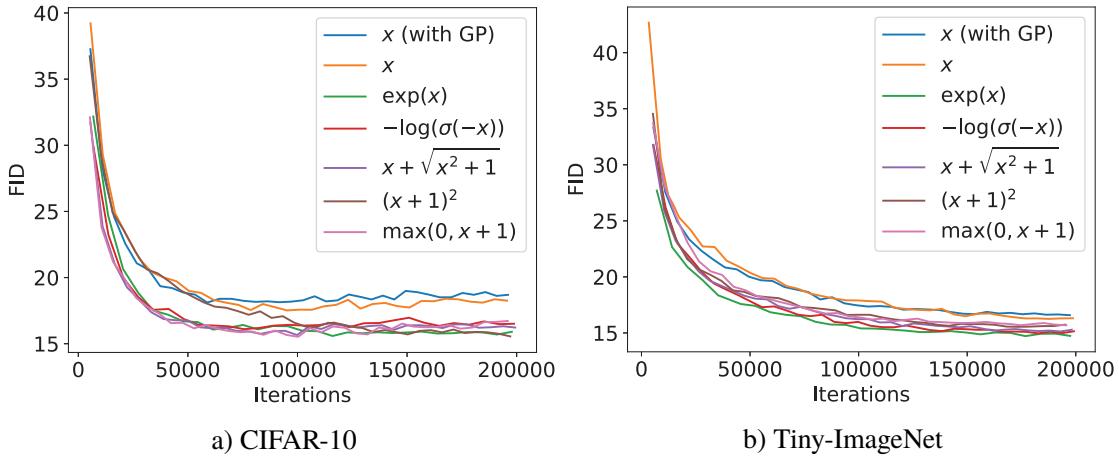


图 2-7 FID 训练曲线。

Figure 2-7 Training curves in terms of FID.

$\max(0, x + \alpha)$ 以及 quadratic loss^[20], 这两组函数并不满足我们的假设。但他们也被广泛使用, 我们把它们也加进来做进一步的分析和对比。对于 quadratic loss, 我们令 $\phi(x) = \varphi(-x) = (x + \alpha)^2$ 。为了让对比更加直接, 我们把生成器的损失度量函数 $\psi(x)$ 固定为 $-x$ 。我们令 $\alpha = 1.0$ 。

在定理2.3中, ϕ 和 φ 的严格单调性是保证相互绑定关系存在的重要前提。但是如果我们将进一步假设生成分布和真实分布的支撑集是有界的, 那么应该可以证明, 存在一个合适的 Lipschitz 惩罚系数 λ 使得真实数据和生成数据都处于损失度量函数 ϕ 和 φ 的单调区间中。我们猜测, 这可能是现实中 hinge loss, square loss 也能工作的原因之一。

我们用 Inception Score (IS)^[15] 以及 Frechet Inception Distance (FID)^[49] 作为衡量模型好坏的量化指标，所得实验结果呈现在表2–2中。对于这部分所有的实验而言，我们采用和 [35] 完全一致的网络结构和超参数。WGAN-GP 在我们的重现的代码中，能在 CIFAR-10 上达到 $IS = 7.71 \pm 0.03$ 和 $FID = 18.86 \pm 0.13$ 。我们在所有实验中都采用最大梯度惩罚，并且为每一组实验调试最好的梯度惩罚系数，其选择范围是 $[0.01, 0.1, 1.0, 10.0]$ 。我们每组实验跑 200,000 次迭代，并且用 $500k$ 个样本来算 IS 和 FID 已获得更准确的评测结果。我们必须指出虽然我们采用了 IS，但是 IS 是一个极其不稳定的指标，它在训练中存在大幅波动，并且甚至连初始化种子对它也有很大的影响。与此相对的，FID 整体上表现就相当稳定。

从表2–2，我们可以看出 LGANs 通常能比 WGAN 达到更好的效果。不同的 LGANs 最终结果都相对接近，而 $\phi(x) = \varphi(-x) = \exp(x)$ 和 $\phi(x) = \varphi(-x) = x + \sqrt{x^2 + 1}$ 在实验中结果最好。Hinge loss 和 quadratic loss 当 λ 选取合适的时候也能达到还不错的效果。我们把各个模型的 FID 训练曲线作图如2–7a和 2–7b。为了让正文更简洁，我们把更多的实验细节以及结果放在附录2.C。

2.7 相关工作

WGAN^[30] 是基于 Wasserstein 距离的 KR 对偶式建立的，它没有梯度无信息的问题。我们发现 Wasserstein 距离的 KR 对偶式中的 Lipschitz 条件可以被放松。如果采用 Wasserstein 距离放松之后的对偶式，它也存在梯度无信息问题。

我们发现惩罚 Lipschitz 常数可以使得一大类生成对抗网络的目标函数有收敛性保障。其中的关键是，它不再仅仅局限于 Wasserstein 距离。比如说，Lipschitz 约束也被引入到原始 GAN 中^[36, 45, 67]，并提高了生成样本的质量，同时也提高了训练的稳定性。事实上，按照我们的分析，原始 GAN 的目标函数其实是 LGANs 中的一员。所以，LGANs 也就解释了为什么 Lipschitz 约束能在原始 GAN 中起作用。同时，也在该如何正确地为原始 GAN 以及其他类似的生成对抗网络中加入 Lipschitz 约束给出了指导性的建议。[70] 也提供了一些关于 Lipschitz 约束如何和 f -散度结合的一些分析。然而，他们的分析局限于 f -散度并且还要求散度的对称性。

[67] 也断言过距离测度可能不是生成对抗网络的训练的核心指导。但是，他们认为原始 GAN 如果采用生成器的非饱和目标函数（也即，采用-log D 技巧）可以以某种方式工作。然而，按照我们的分析，原始 GAN 的最优判别函数在分布支撑集不相交的情况下无法有效指导生成器改进，进而不能保证手联系。

[71] 也分析过最优判别函数的梯度的不可靠性，这个分析促使他们提出了他

们的 Coulomb GAN。但是，他们关于最优判别函数的梯度的有效性这个问题的分析还不够透彻。与之相对的，我们定义了一个普遍存在的梯度无信息问题，并且把该问题与判别函数空间是否有约束联系了起来。以及，我们相应地提出了另一个解决该问题的方法，也即 LGANs。

一些相关工作研究生成对抗网络在次优情况下的性质^[51, 52, 70, 72, 73]。这是另一个很重要的从理论的角度研究生成对抗网络的方向。不过尽管最优情况下的分析和次优下的分析可能不太一样，我们认为最有情况下的优秀性质保证是最起码需要保证的，如收敛性、纳什均衡等等。

本章中，我们主要研究 Lipschitz 和判别器的关系。有人发现 Lipschitz 约束如果加到生成器上，也会对训练有所帮助^[74, 75]。我们认为 Lipschitz 如何在生成器上作用应该是一个完全不一样的机制。与本章讨论有关的，[29] 在数据服从 Lipschitz 密度的前提下，从损坏函数的敏感性的角度研究了 Lipschitz 条件的影响。

有一些生成模型不通过最优判别器的梯度更新生成器，如：[76] 直接按照最优传输计划更新生成器。不过，这类方法目前生成样本的质量还比较有限。

2.8 结论

在本章中，我们从梯度有无信息的角度研究了生成对抗网络的训练收敛性问题。我们提出了一个普遍存在与生成对抗网络中的梯度有无信息问题，它是指最优判别函数关于生成样本的梯度不反应任何关于真实分布的信息，该问题将直接导致生成对抗网络的收敛性没有任何保障。我们论证了，如果判别函数空间不存在任何限制，那么梯度无信息问题普遍存在。我们发现梯度无信息问题也存在于一些判别函数空间中存在约束的生成对抗网络中，但是如果在判别函数上引入了 Lipschitz 惩罚，那么梯度无信息问题就可以被消除。

我们由此定义了一类基于 Lipschitz 的生成对抗网络模型，我们称之为 Lipschitz GANs (LGANs)。我们证明了 LGANs 中任何一个生成样本的梯度都是指向一个真实样本的，从而消除了梯度无信息问题。我们也证明了 LGANs 中最优判别函数的存在性和唯一性。此外，我们还证明了 LGANs 中唯一纳什均衡的存在性，该纳什均衡点上生成分布等于真实分布，这相当于证明 LGANs 是有效的生成模型。我们的实验表明 LGANs 通常能得到比 WGAN 更好的样本质量，而且训练过程中判别函数更加稳定。

本章附录

2.A 证明

2.A.1 定理2.2的证明

令随机变量 X, Y 分别服从于分布 P_r 和 P_g 。假设 $\mathbb{E}_{X \sim P_g} \|X\| < \infty$ 以及 $\mathbb{E}_{Y \sim P_r} \|Y\| < \infty$ 。令 $\mathfrak{G}(f) = \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y))$ 。令 $\|f\|_{Lip}$ 表示 f 的 Lipschitz 常数。令 S_r 和 S_g 分别表示 P_r 和 P_g 的支撑集。令 $W_1(P_r, P_g)$ 表示 P_r 和 P_g 之间的 1-Wasserstein 距离。

引理 2.8 令 ϕ 和 φ 为两个定义域在 \mathbb{R} 的凸函数。假设 f 满足组 $\|f\|_{Lip} \leq k$ 。如果存在 $a_0 \in \mathbb{R}$ 使得 $\phi'(a_0) + \varphi'(a_0) = 0$, 那么 $\mathfrak{G}(f)$ 存在下界。

证明 给定 ϕ, φ 是凸函数, 我们有

$$\begin{aligned}
 \mathfrak{G}(f) &= \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y)) \\
 &\geq \mathbb{E}_{X \sim P_g} (\phi'(a_0)(f(x) - a_0) + \phi(a_0)) + \mathbb{E}_{Y \sim P_r} (\varphi'(a_0)(f(y) - a_0) + \varphi(a_0)) \\
 &= \phi'(a_0) \mathbb{E}_{X \sim P_g} f(x) + \varphi'(a_0) \mathbb{E}_{Y \sim P_r} f(y) + C \\
 &= (\phi'(a_0) + \varphi'(a_0)) \mathbb{E}_{X \sim P_g} f(X) + \varphi'(a_0) (\mathbb{E}_{Y \sim P_r} f(Y) - \mathbb{E}_{X \sim P_g} f(X)) + C \quad (2-18) \\
 &= k \varphi'(a_0) \left(\mathbb{E}_{Y \sim P_r} \frac{1}{k} f(Y) - \mathbb{E}_{X \sim P_g} \frac{1}{k} f(X) \right) + C \\
 &\geq -k \varphi'(a_0) W_1(P_r, P_g) + C.
 \end{aligned}$$

□

引理 2.9 令 ϕ 和 φ 为两个定义域在 \mathbb{R} 的凸函数。假设 f 满足 $\|f\|_{Lip} \leq k$ 。则:

- 如果存在 $a_1 \in \mathbb{R}$ 使得 $\phi'(a_1) + \varphi'(a_1) > 0$, 那么有: 如果 $f(0) \rightarrow +\infty$, 则 $\mathfrak{G}(f) \rightarrow +\infty$;
- 如果存在 $a_2 \in \mathbb{R}$ 使得 $\phi'(a_2) + \varphi'(a_2) < 0$, 那么有: 如果 $f(0) \rightarrow -\infty$, 则 $\mathfrak{G}(f) \rightarrow +\infty$ 。

证明 因为 ϕ, φ 是凸函数，我们有：

$$\begin{aligned}
\mathfrak{G}(f) &= \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y)) \\
&\geq \mathbb{E}_{X \sim P_g} (\phi'(a_1)(f(x) - a_1) + \phi(a_1)) + \mathbb{E}_{Y \sim P_r} (\varphi'(a_1)(f(x) - a_1) + \varphi(a_1)) \\
&= \phi'(a_1) \mathbb{E}_{X \sim P_g} f(x) + \varphi'(a_1) \mathbb{E}_{Y \sim P_r} f(Y) + C_1 \\
&= (\phi'(a_1) + \varphi'(a_1)) \mathbb{E}_{X \sim P_g} f(X) + \varphi'(a_1) (\mathbb{E}_{Y \sim P_r} f(Y) - \mathbb{E}_{X \sim P_g} f(X)) + C_1 \\
&= (\phi'(a_1) + \varphi'(a_1)) \mathbb{E}_{X \sim P_g} f(X) + k \varphi'(a_1) (\mathbb{E}_{Y \sim P_r} \frac{1}{k} f(Y) - \mathbb{E}_{X \sim P_g} \frac{1}{k} f(X)) + C_1 \\
&\geq (\phi'(a_1) + \varphi'(a_1)) \mathbb{E}_{X \sim P_g} f(X) - k \varphi'(a_1) W_1(P_r, P_g) + C_1 \\
&\geq (\phi'(a_1) + \varphi'(a_1)) f(0) - k(\phi'(a_1) + \varphi'(a_1)) \mathbb{E}_{X \sim P_g} \|X\| - k \varphi' W_1(P_r, P_g) + C_1.
\end{aligned} \tag{2-19}$$

因此，如果 $f(0) \rightarrow +\infty$ ，那么 $\mathfrak{G}(f) \rightarrow +\infty$ 。另一条同理可证。 \square

引理 2.10 令 ϕ 和 φ 为两个定义域在 \mathbb{R} 的凸函数。如果 ϕ 和 φ 满足如下性质：

- $\phi' \geq 0, \varphi' \leq 0$;
 - 存在 $a_0, a_1, a_2 \in \mathbb{R}$ 使得 $\phi'(a_0) + \varphi'(a_0) = 0, \phi'(a_1) + \varphi'(a_1) > 0, \phi'(a_2) + \varphi'(a_2) < 0$.
- 则， $\mathfrak{G}(f) = \mathbb{E}_{X \sim P_r} \phi(f(X)) + \mathbb{E}_{Y \sim P_g} \varphi(f(Y))$ ，其中 f 满足 $\|f\|_{Lip} \leq k$ ，有全局最小值。
也即， $\exists f^*, s.t.$
- $\|f^*\|_{Lip} \leq k$;
 - $\forall f s.t. \|f\|_{Lip} \leq k$, 有 $\mathfrak{G}(f^*) \leq \mathfrak{G}(f)$.

证明 按照引理2.8， $\mathfrak{G}(f)$ 存在下界，也即 $\inf(\mathfrak{G}(f)) > -\infty$ 。因此，我们可以得到一系列函数 $\{f_n\}_{n=1}^\infty$ 使得 $\lim_{n \rightarrow \infty} \mathfrak{G}(f_n) = \inf(\mathfrak{G}(f))$ 。假设 $\{r_i\}_{i=1}^\infty$ 是 $dom(f)$ 上的所有有理数的序列。依引理2.9可知，对于任意的 $x \in \mathbb{R}$ ， $\{f_n(x) | n \in \mathbb{R}\}$ 有界。根据 Bolzano-Weierstrass 定理，必定存在一个子列 $\{f_{1n}\} \subseteq \{f_n\}$ 使得 $\{f_{1n}(r_1)\}_{n=1}^\infty$ 收敛。以及存在一个子列 $\{f_{2n}\} \subseteq \{f_{1n}\}$ 使得 $\{f_{2n}(r_2)\}_{n=1}^\infty$ 收敛。对于 r_i ，存在一个子列 $\{f_{in}\} \subseteq \{f_{i-1n}\}$ 使得 $\{f_{in}(r_i)\}_{n=1}^\infty$ 收敛。从而， $\{f_{nn}\}_{n=1}^\infty$ 会收敛到 r_i 。

进一步地，对于任意的 $x \in dom(f)$ ，我们断言 $\{f_{nn}\}_{n=1}^\infty$ 收敛到 x 。实际上， $\forall \epsilon > 0$ ，找到 $r \in \{r_i\}$ 使得 $\|x - r\| \leq \frac{\epsilon}{10k}$ ，我们有：

$$\begin{aligned}
\lim_{m,l \rightarrow \infty} |f_{mm}(x) - f_{ll}(x)| &\leq \lim_{m,l \rightarrow \infty} (|f_{mm}(x) - f_{mm}(r)| + |f_{mm}(r) - f_{ll}(r)| + |f_{ll}(r) - f_{ll}(x)|) \\
&\leq \lim_{m,l \rightarrow \infty} \left(\frac{\epsilon}{10} + \frac{\epsilon}{10} + |f_{mm}(r) - f_{ll}(r)| \right) = \frac{\epsilon}{5}
\end{aligned} \tag{2-20}$$

令 $\epsilon \rightarrow 0$ ，则我们有 $\lim_{m,l \rightarrow \infty} |f_{mm}(x) - f_{ll}(x)| = 0$ 。

记 $\{f_{nn}\}_{n=1}^{\infty}$ 为 $\{g_n\}_{n=1}^{\infty}$, 令 $\{g_n\}_{n=1}^{\infty}$ 收敛到 g . 按照引理2.9, 我们知道 $\exists C'$ such that $|g_n(0)| \leq C'$, $\forall n \in \mathbb{N}$. 因为 $\phi' \geq 0, \varphi' \leq 0$, 我们有:

$$\begin{aligned}\phi(g_n(x)) &\geq \phi(g_n(0) - k\|x\|) \geq \phi(-C' - k\|x\|) \\ &\geq \phi'(a_0)(-C' - k\|x\| - a_0) + \phi(a_0) \\ &= -k\phi'(a_0)\|x\| + C''\end{aligned}\tag{2-21}$$

也即, $\phi(g_n(x)) + k\phi'(a_0)\|x\| - C'' \geq 0$.

由 Fatou's 引理:

$$\begin{aligned}\mathbb{E}_{X \sim P_g}(\phi(g(X)) + k\phi'(a_0)\|X\| - C'') &= \mathbb{E}_{X \sim P_g} \liminf_{n \rightarrow \infty} (\phi(g_n(X)) + k\phi'(a_0)\|X\| - C'') \\ &\leq \liminf_{n \rightarrow \infty} \mathbb{E}_{X \sim P_g}(\phi(g_n(X)) + k\phi'(a_0)\|X\| - C'') \\ &= \liminf_{n \rightarrow \infty} \mathbb{E}_{X \sim P_g} \phi(g_n(X)) + \mathbb{E}_{X \sim P_g}(k\phi'(a_0)\|X\| - C'')\end{aligned}\tag{2-22}$$

这意味着 $\mathbb{E}_{X \sim P_g} \phi(g(X)) \leq \liminf_{n \rightarrow \infty} \mathbb{E}_{X \sim P_g} \phi(g_n(X))$. 类似地, 我们有 $\mathbb{E}_{Y \sim P_r} \varphi(g(Y)) \leq \liminf_{n \rightarrow \infty} \mathbb{E}_{Y \sim P_r} \varphi(g_n(Y))$. 结合这两个不等式, 我们有:

$$\begin{aligned}\mathfrak{G}(g) &= \mathbb{E}_{X \sim P_g} \phi(g(X)) + \mathbb{E}_{Y \sim P_r} \varphi(g(Y)) \leq \liminf_{n \rightarrow \infty} \mathbb{E}_{X \sim P_g} \phi(g_n(X)) + \liminf_{n \rightarrow \infty} \mathbb{E}_{Y \sim P_r} \varphi(g_n(Y)) \\ &\leq \liminf_{n \rightarrow \infty} (\mathbb{E}_{X \sim P_g} \phi(g_n(X)) + \mathbb{E}_{Y \sim P_r} \varphi(g_n(Y))) = \inf_{\|f\|_{Lip} \leq k} \mathfrak{G}(f)\end{aligned}\tag{2-23}$$

对于任何的 $x, y \in dom(g)$, $|g(x) - g(y)| \leq \lim_{n \rightarrow \infty} (|g(x) - g_n(x)| + |g_n(x) - g_n(y)| + |g_n(y) - g(y)|) \leq k\|x - y\|$.

也即, $\|g\|_{Lip} \leq k$, $\mathfrak{G}(g) = \inf_{\|f\|_{Lip} \leq k} \mathfrak{G}(f)$. \square

引理 2.11 (Wasserstein 距离) 若存在 k 使得 f 满足 $\|f\|_{Lip} \leq k$, 则 $\mathfrak{T}(f) = \mathbb{E}_{X \sim P_g} f(X) - \mathbb{E}_{Y \sim P_r} f(Y)$ 有全局最小值。

证明 对于任何的 $C \in \mathbb{R}$, $\mathfrak{T}(f + C) = \mathfrak{T}(f)$. 和上面的引理类似, 我们可以得到一系列函数 $\{f_n\}_{n=1}^{\infty}$ 使得 $\lim_{n \rightarrow \infty} \mathfrak{T}(f_n) = \inf(\mathfrak{T}(f))$. 不失一般性地, 我们假设 $f_n(0) = 0, \forall n \in \mathbb{N}^+$. 因为 $\|f_n\|_{Lip} \leq k$, 我们可以断言对于任意的 $x \in \mathbb{R}$, $\{f_n(x) | n \in \mathbb{N}\}$ 是有界的. 然后模仿上面引理的证明方法, 可以找到 f^* 使得 $\mathfrak{T}(f^*) = \inf_{\|f\|_{Lip} \leq k} \mathfrak{T}(f)$. \square

引理 2.12 令 ϕ 和 φ 为两个定义域在 \mathbb{R} 的凸函数。我们进一步假设支撑集 S_r 和 S_g 是有界的。如果 ϕ 和 φ 满足如下性质:

- $\phi' \geq 0, \varphi' \leq 0$;

- 存在 $a_0 \in \mathbb{R}$ 使得 $\phi'(a_0) + \varphi'(a_0) = 0$ 。

则 $\mathfrak{G}(f) = \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y))$, 其中 f 满足 $\|f\|_{Lip} \leq k$, 有全局最小值。也即 $\exists f^*, s.t.$

- $\|f^*\|_{Lip} \leq k$
- $\forall f s.t. \|f\|_{Lip} \leq k$ 有 $\mathfrak{G}(f^*) \leq \mathfrak{G}(f)$ 。

证明 之前的引理其实已经包含了该定理的大部分的情况。这里我们仅需要进一步考虑以下情形：对于任意的 $x \in \mathbb{R}$, $\phi'(x) + \varphi'(x) \geq 0$ (或 $\phi'(x) + \varphi'(x) \leq 0$) 且存在 a_1 使得 $\phi'(a_1) + \varphi'(a_1) > 0$ (or $\phi'(a_1) + \varphi'(a_1) < 0$)。

不失一般性地，我们假设对于任意的 x , $\phi'(x) + \varphi'(x) \geq 0$ 且存在 a_1 使得 $\phi'(a_1) + \varphi'(a_1) > 0$ 。我们知道 $\forall x \leq a_0$, $\phi'(x) + \varphi'(x) = 0$, 这使得 $\forall x \leq a_0$, $\phi'(x) = -\varphi'(x)$ 。因此对于任意的 $x \leq a_0$, $0 \leq \phi''(x) = -\varphi''(x) \leq 0$, 这意味着 $\forall x \leq a_0$, $\phi(x) = -\varphi(x) = tx$, $t \geq 0$ 。

和之前的引理类似，我们可以得到一系列函数 $\{f_n\}_{n=1}^\infty$ 使得 $\lim_{n \rightarrow \infty} \mathfrak{G}(f_n) = \inf(\mathfrak{G}(f))$ 。事实上，我们可以假设对于任意的 $n \in \mathbb{N}^+$, 存在 $f_n(0) \in [-C, C]$, 其中 C 是一个常数。由引理2.9不难发现 $f_n(0) \leq C$ 。另一方面, 如果 $C > k \cdot diam(S_r \cup S_g) + a_0$, 则：如果 $f(0) < -C$, 我们有对于任意的 $X \in S_r \cup S_g$ $f(X) < a_0$ 。在这个情况下, $\mathfrak{G}(f) = \mathfrak{G}(f - f(0) - C)$ 。这是我们假设 $f_n(0) \in [-C, C]$ 的原因。因为 $\|f_n\|_{Lip} \leq k$, 我可以断言对于任意的 $x \in \mathbb{R}$, $\{f_n(x)|n \in \mathbb{R}\}$ 是有界的。因此，我们模仿引理2.10 并找到 f^* 使得 $\mathfrak{G}(f^*) = \inf_{\|f\|_{Lip} \leq k} \mathfrak{G}(f)$ 。 \square

引理 2.13 (定理2.2, 前半) 在引理2.12相同假设下, 我们有: 如果 $\lambda > 0$ 以及 $\alpha > 1$, 则 $\mathfrak{F}(f) = \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y)) + \lambda \|f\|_{Lip}^\alpha$ 存在下界。

证明 若 $\|f\|_{Lip} = \infty$, 显然 $\mathfrak{F}(f) = \infty$ 。而如果 $\|f\|_{Lip} < \infty$, 结合引理2.8, 我们有 $\mathfrak{F}(f) = \mathfrak{G}(f) + \lambda \|f\|_{Lip}^\alpha \geq -\|f\|_{Lip} \varphi'(a_0) W_1(P_r, P_g) + \lambda \|f\|_{Lip}^\alpha$ 。因为 $\lambda > 0$ 且 $\alpha > 1$, 右边是一个关于 $\|f\|_{Lip}$ 的凸函数, 它存在下界。所以, 我们可以的找到一个序列 $\{f_n\}_{n=1}^\infty$ 使得 $\lim_{n \rightarrow \infty} \mathfrak{F}(f_n) = \inf_{f \in \text{dom } \mathfrak{F}} \mathfrak{F}(f)$ 。无疑存在一个常数 C 使得对于任意的 f_n 有 $\|f_n\|_{Lip} \leq C$ 。然后, 可以得到对于任意的 x , $\{f_n(x)\}$ 是有界的。所以, 我们找到序列 $\{g_n\}$ 使得 $\{g_n\} \subseteq \{f_n\}$ and $\{g_n\}_{n=1}^\infty$ 在任意点 x 均收敛。假设 $\lim_{n \rightarrow \infty} g_n = g$, 那么由 Fatou's 引理, 我们有 $\mathfrak{G}(g) \leq \underline{\lim}_{n \rightarrow \infty} \mathfrak{G}(g_n)$ 。

接下来, 我们证明 $\|g\|_{Lip} \leq \underline{\lim}_{n \rightarrow \infty} \|g_n\|_{Lip}$ 。如果这个断言成立, 则 $\mathfrak{F}(g) = \mathfrak{G}(g) + \lambda \|g\|_{Lip}^\alpha \leq \underline{\lim}_{n \rightarrow \infty} \mathfrak{G}(g_n) + \underline{\lim}_{n \rightarrow \infty} \lambda \|g_n\|_{Lip}^\alpha \leq \underline{\lim}_{n \rightarrow \infty} (\mathfrak{G}(g_n) + \lambda \|g_n\|_{Lip}^\alpha) = \inf \mathfrak{F}(f)$ 。因此全局最小值存在。

事实上，如果 $\|g\|_{Lip} > \liminf_{n \rightarrow \infty} \|g_n\|_{Lip}$ ，那么存在 x, y 使得 $\frac{|g(x)-g(y)|}{\|x-y\|} \geq \liminf_{n \rightarrow \infty} |g_n(x)-g_n(y)| + \epsilon \geq \liminf_{n \rightarrow \infty} \frac{|g_n(x)-g_n(y)|}{\|x-y\|} + \epsilon$ ，也即， $|g(x)-g(y)| \geq \liminf_{n \rightarrow \infty} |g_n(x)-g_n(y)| + \epsilon\|x-y\| = |g(x)-g(y)| + \epsilon\|x-y\| > |g(x)-g(y)|$ 。这个矛盾说明 $\|g\|_{Lip} \leq \liminf_{n \rightarrow \infty} \|g_n\|_{Lip}$ 。□

引理 2.14(定理2.2, 后半) 令 ϕ 和 φ 为两个定义域在 \mathbb{R} 的凸函数。如果 ϕ 或者 φ 严格凸， $\lambda > 0$ ，且 $\alpha > 1$ ，则使 $\mathfrak{F}(f) = \mathbb{E}_{X \sim P_g} \phi(f(X)) + \mathbb{E}_{Y \sim P_r} \varphi(f(Y)) + \lambda \|f\|_{Lip}^\alpha$ 取得最小值的判别函数 f 在支撑集 $S_r \cup S_g$ 上的取值唯一。

证明 不失一般性地，我们假设 ϕ 严格凸。按照严格凸的定义我们有 $\forall x, y \in \mathbb{R}$, $\phi(\frac{x+y}{2}) < \frac{1}{2}(\phi(x) + \phi(y))$ 。假设 f_1 和 f_2 是两个不同的使 $\mathfrak{F}(f)$ 最小化的函数。

首先，我们有

$$\begin{aligned} \left\| \frac{f_1 + f_2}{2} \right\|_{Lip} &= \sup_{x,y} \frac{\frac{f_1(x)+f_2(x)}{2} - \frac{f_1(y)+f_2(y)}{2}}{\|x-y\|} \\ &\leq \sup_{x,y} \frac{1}{2} \frac{|f_1(x)-f_1(y)| + |f_2(x)-f_2(y)|}{\|x-y\|} \\ &\leq \frac{1}{2} \left(\sup_{x,y} \frac{|f_1(x)-f_1(y)|}{\|x-y\|} + \sup_{x,y} \frac{|f_2(x)-f_2(y)|}{\|x-y\|} \right) \\ &= \frac{1}{2} (\|f_1\|_{Lip} + \|f_2\|_{Lip}). \end{aligned} \tag{2-24}$$

因为 $\lambda > 0$ 且 $\alpha > 1$ ，我们进一步有：

$$\begin{aligned} \lambda \left\| \frac{f_1 + f_2}{2} \right\|_{Lip}^\alpha &\leq \lambda \left(\frac{1}{2} (\|f_1\|_{Lip} + \|f_2\|_{Lip}) \right)^\alpha \\ &\leq \lambda \frac{1}{2} (\|f_1\|_{Lip}^\alpha + \|f_2\|_{Lip}^\alpha). \end{aligned} \tag{2-25}$$

令 $\mathfrak{G}(f_1) = \mathfrak{G}(f_2) = \inf \mathfrak{G}(f)$ ，则我们有：

$$\begin{aligned} \mathfrak{G}\left(\frac{f_1 + f_2}{2}\right) &= \mathbb{E}_{X \sim P_g} \phi\left(\frac{f_1 + f_2}{2}\right) + \mathbb{E}_{Y \sim P_r} \varphi\left(\frac{f_1 + f_2}{2}\right) + \lambda \left\| \frac{f_1 + f_2}{2} \right\|_{Lip}^\alpha \\ &< \mathbb{E}_{X \sim P_g} \left(\frac{\phi(f_1) + \phi(f_2)}{2} \right) + \mathbb{E}_{Y \sim P_r} \varphi\left(\frac{f_1 + f_2}{2}\right) + \lambda \left\| \frac{f_1 + f_2}{2} \right\|_{Lip}^\alpha \\ &\leq \mathbb{E}_{X \sim P_g} \left(\frac{\phi(f_1) + \phi(f_2)}{2} \right) + \mathbb{E}_{Y \sim P_r} \left(\frac{\varphi(f_1) + \varphi(f_2)}{2} \right) + \lambda \left\| \frac{f_1 + f_2}{2} \right\|_{Lip}^\alpha \\ &\leq \mathbb{E}_{X \sim P_g} \left(\frac{\phi(f_1) + \phi(f_2)}{2} \right) + \mathbb{E}_{Y \sim P_r} \left(\frac{\varphi(f_1) + \varphi(f_2)}{2} \right) + \lambda \frac{1}{2} (\|f_1\|_{Lip}^\alpha + \|f_2\|_{Lip}^\alpha) \\ &= \frac{1}{2} (\mathfrak{G}(f_1) + \mathfrak{G}(f_2)) = \inf \mathfrak{G}(f) \end{aligned} \tag{2-26}$$

由此，我们得到矛盾 $\mathfrak{G}(\frac{f_1+f_2}{2}) < \inf \mathfrak{G}(f)$ 。故而使 $\mathfrak{G}(f)$ 最小化的判别函数唯一。□

2.A.2 定理2.3的证明

令 $J_D = \mathbb{E}_{x \sim P_g}[\phi(f(x))] + \mathbb{E}_{x \sim P_r}[\varphi(f(x))]$ 。

令 $\dot{J}_D(x) = P_g(x)\phi(f(x)) + P_r(x)\varphi(f(x))$, 有 $J_D = \int_{\mathbb{R}^n} \dot{J}_D(x)dx$ 。

令 $J_D^*(k) = \min_{f \in \mathcal{F}_{k\text{-Lip}}} J_D = \min_{f \in \mathcal{F}_{1\text{-Lip}}, b} \mathbb{E}_{x \sim P_g}[\phi(k \cdot f(x) + b)] + \mathbb{E}_{x \sim P_r}[\varphi(k \cdot f(x) + b)]$ 。

令 $k(f)$ 表示 f 的 Lipschitz 常数。

令 $J = J_D + \lambda \cdot k(f)^2$ 。

令 $f^* = \arg \min_f [J_D + \lambda \cdot k(f)^2]$ 。

引理 2.15 $\forall x, \frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$, 当且仅当, $k(f^*) = 0$ 。

证明

(i) 证明: 如果 $\forall x, \frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$, 则 $k(f^*) = 0$ 。

最优判别函数 f^* 满足 $\frac{\partial J}{\partial k(f^*)} = \frac{\partial J_D^*}{\partial k(f^*)} + 2\lambda \cdot k(f^*) = 0$ 。

对于任意的 x 有 $\frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$, 意味着 $\frac{\partial J_D^*}{\partial k(f^*)} = 0$ 。

因此, 我们有 $k(f^*) = 0$ 。

(ii) 证明: 如果 $k(f^*) = 0$, 则 $\forall x, \frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$ 。

最优判别函数 f^* 满足 $\frac{\partial J}{\partial k(f^*)} = \frac{\partial J_D^*}{\partial k(f^*)} + 2\lambda \cdot k(f^*) = 0$ 。

$k(f^*) = 0$ 可以推出 $\frac{\partial J_D^*}{\partial k(f^*)} = 0$ 。 $k(f^*) = 0$ 同时意味着 $\forall x, y, f^*(x) = f^*(y)$ 。

已知 $\forall x, y, f^*(x) = f^*(y)$, 如果存在某个 x 使得 $\frac{\partial \dot{J}_D(x)}{\partial f^*(x)} \neq 0$, 则显然 $\frac{\partial J_D^*}{\partial k(f^*)} \neq 0$ 。

这和 $\frac{\partial J_D^*}{\partial k(f^*)} = 0$ 矛盾。因此, 我们有 $\forall x, \frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$ 。 \square

引理 2.16 如果 $\forall x, y, f^*(x) = f^*(y)$, 那么 $P_r = P_g$ 。

证明 $\forall x, y, f^*(x) = f^*(y)$, 所以 $k(f^*) = 0$ 。根据引理2.15, 有 $\forall x, \frac{\partial \dot{J}_D(x)}{\partial f^*(x)} = 0$, 也即,

$P_g(x) \frac{\partial \phi(f^*(x))}{\partial f^*(x)} + P_r(x) \frac{\partial \varphi(f^*(x))}{\partial f^*(x)} = 0$ 。因此, $\frac{P_g(x)}{P_r(x)} = -\frac{\frac{\partial \varphi(f^*(x))}{\partial f^*(x)}}{\frac{\partial \phi(f^*(x))}{\partial f^*(x)}}$ 。因为 $f^*(x)$ 取常值, 所以

$\frac{P_g(x)}{P_r(x)}$ 是常值, 从而 $P_r = P_g$ 。 \square

证明 (定理2.3)

(a):

令 k 表示 f^* 的 Lipschitz 常数。

考虑满足 $\frac{\partial \dot{J}_D(x)}{\partial f^*(x)} \neq 0$ 的 x 。

定义 $k(x) = \sup_y \frac{|f(y) - f(x)|}{\|y - x\|}$ 。

(i) 假设 $\forall \delta, \forall \epsilon, \exists z, w \in B(x, \epsilon)$ 使得 $\frac{|f^*(z) - f^*(w)|}{\|z - w\|} \geq k - \delta$ 。这意味着, 存在 t 使得 $f'(t) \geq k - \delta$, 因为 $\frac{|f^*(z) - f^*(w)|}{\|z - w\|} = \frac{\int_w^z f'^*(t)dt}{\|z - w\|}$ 。令 $\epsilon \rightarrow 0$, 我们有 $t \rightarrow x$, 从而

$|f^{*'}(t)| \rightarrow |f^{*'}(x)|$ 。令 $\delta \rightarrow 0$, 我们有 $(k - \delta) \rightarrow k$ 。假设 f^* 是平滑的, 我们有 $|f'(x)| = k$, 这意味着存在 y 使得 $|f^*(y) - f^*(x)| = k\|y - x\|$ 。

(ii) 假设 $\exists \delta, \exists \epsilon, \forall z, w \in B(x, \epsilon)$, $\frac{|f^*(z) - f^*(w)|}{\|z - w\|} < k - \delta$ 。如果 $\forall \delta_2, \forall \epsilon_2 \in (0, \epsilon/2)$, $\exists y \in B(x, \epsilon_2)$, 使得 $k(y) > k - \delta_2$ 。存在 $\{y_n\}_{n=1}^\infty$ 使得 $\lim_{n \rightarrow \infty} \frac{|f(y) - f(y_n)|}{\|y - y_n\|} = k(y)$ 。所以存在 y' 使得 $\frac{|f(y) - f(y')|}{\|y - y'\|} \geq k - \delta_2$ 。根据假设, 我们有 $\|y - y'\| \geq \frac{\epsilon}{2}$ 。从而 $k(x) \geq \frac{|f^*(x) - f^*(y')|}{\|x - y\|} \geq \frac{|f^*(y) - f^*(y')| - |f^*(x) - f^*(y')|}{\|x - y\| + \|y - y'\|} \geq \frac{|f^*(y) - f^*(y')| - k\|x - y\|}{\|x - y\| + \|y - y'\|} \geq (k - \delta_2) \frac{\|y - y'\|}{\|x - y\| + \|y - y'\|} - k \frac{\|x - y\|}{\|x - y\| + \|y - y'\|} \geq (1 - \frac{\epsilon_2}{\epsilon_2 + \|y - y'\|})(k - \delta_2) - k \frac{\epsilon_2}{\|y - y'\|} \geq (1 - \frac{\epsilon_2}{\epsilon_2 + \|y - y'\|})(k - \delta_2) - k \frac{\epsilon_2}{\|y - y'\|}$ 。令 $\epsilon_2 \rightarrow 0$, 令 $\delta_2 \rightarrow 0$, 可得 $k(x) = k$, 故存在 y 使得 $|f^*(y) - f^*(x)| = k\|y - x\|$ 。

(iii) 现在我们可以假设 $\exists \delta_2, \exists \epsilon_2, \forall y \in B(x, \epsilon_2)$, 使得 $k(y) \leq k - \delta_2$ 。如果 $\frac{\partial \hat{J}_D(x)}{\partial f^*(x)} \neq 0$, 不失一般性地, 我们可以假设 $\frac{\partial \hat{J}_D(x)}{\partial f^*(x)} > 0$ 。那么, 对于任意的 $y \in B(x, \epsilon_2)$, 只要 ϵ_2 足够小, 我们有 $\frac{\partial \hat{J}_D(y)}{\partial f^*(y)} > 0$ 。现在我们改变在 $B(x, \epsilon_2)$ 里面的 y 的 $f^*(y)$ 的值。令

$$g(y) = \begin{cases} f^*(y) - \frac{\epsilon_2}{N}(1 - \frac{\|x-y\|}{\epsilon_2}), & y \in B(x, \epsilon_2); \\ f^*(y) & \text{otherwise.} \end{cases} \quad \text{因为 } \frac{\partial \hat{J}_D(y)}{\partial f^*(y)} > 0, \quad \forall y \in B(x, \epsilon_2), \quad \text{当 } \epsilon_2 \text{ 足够小。}$$

N 足够大, 不难得出 $J_D(g) < J_D(f^*)$ 。我们接下来验证 $\|g\|_{Lip} \leq k$ 。对于任意的 y, z , 如果 $y, z \notin B(x, \epsilon_2)$, 则 $\frac{|g(y) - g(z)|}{\|y - z\|} = \frac{|f^*(y) - f^*(z)|}{\|y - z\|} < k$ 。如果 $y \in B(x, \epsilon_2)$, $z \notin B(x, \epsilon_2)$, 则 $\frac{|g(y) - g(z)|}{\|y - z\|} \leq \frac{|f^*(y) - f^*(z)| + \frac{\epsilon_2}{N}(1 - \frac{\|x-y\|}{\epsilon_2})}{\|y - z\|} \leq \frac{|f^*(y) - f^*(z)|}{\|y - z\|} + \frac{\frac{\epsilon_2}{N}(1 - \frac{\|x-y\|}{\epsilon_2})}{\epsilon_2 - \|x-y\|} = \frac{|f^*(y) - f^*(z)|}{\|y - z\|} + \frac{1}{N} \leq k(y) + \frac{1}{N} \leq k - \delta_2 + \frac{1}{N} < k$ (when $N \gg \frac{1}{\delta_2}$)。If $y, z \in B(x, \epsilon)$, then $\frac{|g(y) - g(z)|}{\|y - z\|} \leq \frac{|f^*(y) - f^*(z)| + |\frac{\epsilon_2}{N}(1 - \frac{\|x-y\|}{\epsilon_2}) - \frac{\epsilon_2}{N}(1 - \frac{\|x-z\|}{\epsilon_2})|}{\|y - z\|} = \frac{|f^*(y) - f^*(z)|}{\|y - z\|} + \frac{\frac{\epsilon_2}{N}(\frac{\|x-y\| - \|x-z\|}{\epsilon_2})}{\|y - z\|} \leq \frac{|f^*(y) - f^*(z)|}{\|y - z\|} + \frac{1}{N} \frac{\|y - z\|}{\|y - z\|} = \frac{|f^*(y) - f^*(z)|}{\|y - z\|} + \frac{1}{N} \leq k - \delta_2 + \frac{1}{N} < k$ (当 $N \gg \frac{1}{\delta_2}$)。所以, 我们有 $\|g\|_{Lip} \leq k$ 。但是我们有 $J_D(g) < J_D(f^*)$, 也即, f^* 并非 J_D 的最优解。该矛盾表明必定存在 y 使得 $|f^*(y) - f^*(x)| = k\|y - x\|$ 。

(b): 对于 $x \in S_r \cup S_g - S_r \cap S_g$, 假设 $P_g(x) \neq 0$ 且 $P_r(x) = 0$, 我们有 $\frac{\partial \hat{J}_D(x)}{\partial f^*(x)} = P_g(x) \frac{\partial \phi(f^*(x))}{\partial f^*(x)} + P_r(x) \frac{\partial \varphi(f^*(x))}{\partial f^*(x)} = P_g(x) \frac{\partial \phi(f^*(x))}{\partial f^*(x)} > 0$, 因为 $P_g(x) > 0$ 且 $\frac{\partial \phi(f^*(x))}{\partial f^*(x)} > 0$ 。根据 (a) 可知, 必然存在一个点 y 使得 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ 。另一条同理可证。

(c): 按照引理2.16, 如果 $P_r \neq P_g$, 则对于最优判别函数 f^* , 必然存在至少一个点对 x 和 y 使得 $y \neq x$ and $f^*(x) \neq f^*(y)$ 。这也意味着 $k(f^*) > 0$ 。根据引理2.15, 存在一个点 x 使得 $\frac{\partial \hat{J}_D(x)}{\partial f^*(x)} \neq 0$ 。根据 (a) 可知, 存在点 y 满足 $y \neq x$ 使得 $|f^*(y) - f^*(x)| = k(f^*) \cdot \|y - x\|$ 。

(d): 在纳什均衡状态, 有对于任何的 $x \in S_r \cup S_g$, $\frac{\partial J}{\partial k(f)} = \frac{\partial \hat{J}_D^*}{\partial k(f)} + 2\lambda \cdot k(f) = 0$ 且 $\frac{\partial \hat{J}_D(x)}{\partial f(x)} \frac{\partial f(x)}{\partial x} = 0$ 。我们断言在纳什均衡态, $k(f)$ 必定为 0。如果 $k(f) \neq 0$, 按照引理2.15, 必然存在一个点 \hat{x} 使得 $\frac{\partial \hat{J}_D(\hat{x})}{\partial f(\hat{x})} \neq 0$ 。然后根据 (a) 有 $\exists \hat{y}$ fitting $|f(\hat{y}) - f(\hat{x})| = k(f) \cdot \|\hat{x} - \hat{y}\|$ 。按照定理2.5, 我们有 $\|\frac{\partial f(\hat{x})}{\partial \hat{x}}\| = k(f) \neq 0$ 。这和 $\frac{\partial \hat{J}_D(\hat{x})}{\partial f(\hat{x})} \frac{\partial f(\hat{x})}{\partial \hat{x}} = 0$ 矛盾。

因此 $k(f) = 0$ 。所以, $\forall x \in S_r \cup S_g$, $\frac{\partial f(x)}{\partial x} = 0$, 从而, $\forall x, y, f(x) = f(y)$ 。根据引理2.16, $\forall x, y, f(x) = f(y)$ 可以推出 $P_r = P_g$ 。因此, 系统的纳什均衡状态下, 必有 $k(f) = 0$ 以及 $P_r = P_g$ 。 \square

注 对于 Wasserstein 距离, $\nabla_{f^*(x)} \mathring{J}_D(x) = 0$ 当且仅当 $P_r(x) = P_g(x)$ 。对于 Wasserstein 距离, 惩罚 Lipschitz 常数可以使得在收敛时有 $\frac{\partial f^*(x)}{\partial x} = 0$ for all x 。

2.A.3 定理2.4的证明

引理 2.17 令 k 为 f 的 Lipschitz 常数。如果 $f(a) - f(b) = k\|a - b\|$ 并且 $f(b) - f(c) = k\|b - c\|$, 则 $f(a) - f(c) = k\|a - c\|$ 且 $(a, f(a)), (b, f(b)), (c, f(c))$ 共线。

证明 $f(a) - f(c) = f(a) - f(b) + f(b) - f(c) = k\|a - b\| + k\|b - c\| \geq k\|a - c\|$ 。因为 f 的 Lipschitz 常数为 k , 所以我们有 $f(a) - f(c) \leq k\|a - c\|$ 。因此 $f(a) - f(c) = k\|a - c\|$ 。因为三角不等式的等号成立, 所以我们有 a, b, c 共线。进一步地, 因为 $f(a) - f(b) = k\|a - b\|$ 、 $f(b) - f(c) = k\|b - c\|$ 、 $f(a) - f(c) = k\|a - c\|$, 我们有 $(a, f(a)), (b, f(b)), (c, f(c))$ 共线。 \square

引理 2.18 对于任意的点 x , 如果 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} > 0$, 则存在 y 满足 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} < 0$ 使得 $f^*(y) - f^*(x) = k(f^*)\|y - x\|$ 。

对于任意的点 y , 如果 $\frac{\partial \mathring{J}_D(y)}{\partial f^*(y)} < 0$, 则存在 x 满足 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} > 0$ 使得 $f^*(y) - f^*(x) = k(f^*)\|y - x\|$ 。

证明 考虑满足 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} > 0$ 的点 x 。按照定理2.3, 存在 y 使得 $|f^*(y) - f^*(x)| = k(f^*)\|y - x\|$ 。假设对于任意的满足 $|f^*(y) - f^*(x)| = k(f^*)\|y - x\|$ 的 y 都满足 $\frac{\partial \mathring{J}_D(y)}{\partial f^*(y)} \geq 0$ 。考虑集合 $S(x) = \{y \mid f^*(y) - f^*(x) = k(f^*)\|y - x\|\}$ 。根据引理2.17, 任何满足对于任何的 $y \in S(x)$ 均有 $f^*(z) - f^*(y) = k(f^*)\|z - y\|$ 的 z 也将在集合 $S(x)$ 中。类似定理2.3中第一条的证明, 我们可以降低所有在集合 $S(x)$ 中的点的值从而构造出一个更好的 f 。该矛盾表明, 必定存在一个 y 满足 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} < 0$ 并使得 $|f^*(y) - f^*(x)| = k(f^*)\|y - x\|$ 。给定 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} > 0$ 和 $\frac{\partial \mathring{J}_D(x)}{\partial f^*(x)} < 0$, 我们可以得出 $f^*(y) > f^*(x)$ 以及 $f^*(y) - f^*(x) = k(f^*)\|y - x\|$ 。否则, 如果 $f^*(x) - f^*(y) = k(f^*)\|y - x\|$, 则我们可以通过降低 $f^*(x)$ 升高 $f^*(y)$ 构造出一个更好的 f , 这个操作不会破坏 Lipschitz 常数为 k 的性质。另一条同理可证。 \square

引理 2.19 对于任意的 x , 如果 $\frac{\partial \mathring{J}_D(x)}{\partial f(x)} > 0$, 则 $P_g(x) > 0$ 。对于任意的 y , 如果 $\frac{\partial \mathring{J}_D(y)}{\partial f(y)} < 0$, 则 $P_r(y) > 0$ 。

证明 $\frac{\partial J_D^\circ(x)}{\partial f(x)} = P_g(x) \frac{\partial \phi(f(x))}{\partial f(x)} + P_r(x) \frac{\partial \varphi(f(x))}{\partial f(x)}$ 。已知 $\phi'(x) > 0$ 以及 $\varphi'(x) < 0$ 。自然地, $\frac{\partial J_D^\circ(x)}{\partial f(x)} > 0$ 可以推出 $P_g(x) > 0$ 。类似地, $\frac{\partial J_D^\circ(y)}{\partial f(y)} < 0$ 可以退出 $P_r(y) > 0$ 。 \square

证明 (定理2.4的证明)

对于任意的点 $x \in S_g$, 如果 $\frac{\partial J_D^\circ(x)}{\partial f^*(x)} > 0$, 按照引理2.18, 存在 y 满足 $\frac{\partial J_D^\circ(x)}{\partial f^*(x)} < 0$ 使得 $f^*(y) - f^*(x) = k(f^*)\|y - x\|$ 。按照引理2.19, 我们有 $P_r(y) > 0$ 。也即, 存在 $y \in S_r$ 使得 $f^*(y) - f^*(x) = k(f^*)\|y - x\|$ 。另一条同理可证。 \square

注 如果某个点 $x \in S_g$ 满足 $\frac{\partial J_D^\circ(x)}{\partial f^*(x)} < 0$, 根据引理2.19, 有 $P_r(x) > 0$, 这意味着 x 在 S_r 和 S_g 的交集中。它也可以被认为是一个点 $y \in S_r$ 。因此, 我们可以用定理中的另一条来保证存在一个 $x' \in S_g$ 来绑定这个点。

2.A.4 定理2.5的证明

令 x, y 为两个不相同的点, 定义 $x_t = x + t \cdot (y - x)$, 其中 $t \in [0, 1]$ 。

引理 2.20 如果 $f(x)$ 是关于 $\|\cdot\|_p$ 是 k -Lipschitz 的, 且 $f(y) - f(x) = k\|y - x\|_p$, 则 $f(x_t) = f(x) + t \cdot k\|y - x\|_p$ 。

证明 由 $f(x)$ 的 k -Lipschitz 性质, 我们有:

$$\begin{aligned} f(y) - f(x) &= f(y) - f(x_t) + f(x_t) - f(x) \\ &\leq f(y) - f(x_t) + k\|x_t - x\|_p = f(y) - f(x_t) + t \cdot k\|y - x\|_p \\ &\leq k\|y - x_t\|_p + t \cdot k\|y - x\|_p = k \cdot (1 - t)\|y - x\|_p + t \cdot k\|y - x\|_p \\ &= k\|y - x\|_p. \end{aligned} \tag{2-27}$$

而我们又已知 $f(y) - f(x) = k\|y - x\|_p$, 故上式中的不等式中的等号均成立, 从而 $f(x_t) = f(x) + t \cdot k\|y - x\|_p$ 。 \square

引理 2.21 令 v 表示单位向量 $\frac{y-x}{\|y-x\|_2}$ 。如果 $f(x_t) = f(x) + t \cdot k\|y - x\|_2$, 则 $\frac{\partial f(x_t)}{\partial v} = k$ 。

证明

$$\begin{aligned} \frac{\partial f(x_t)}{\partial v} &= \lim_{h \rightarrow 0} \frac{f(x_t + hv) - f(x_t)}{h} = \lim_{h \rightarrow 0} \frac{f(x_t + h \frac{y-x}{\|y-x\|_2}) - f(x_t)}{h} \\ &= \lim_{h \rightarrow 0} \frac{f(x_t + \frac{h}{\|y-x\|_2}) - f(x_t)}{h} = \lim_{h \rightarrow 0} \frac{\frac{h}{\|y-x\|_2} \cdot k\|y - x\|_2}{h} = k. \end{aligned} \tag*{\square}$$

证明 (定理2.5的证明) 假设 $p = 2$ 。根据 [68], 如果 $f(x)$ 关于 $\|\cdot\|_2$ 是 k -Lipschitz 的, 且 $f(x)$ 在 x_t 处可导, 则 $\|\nabla f(x_t)\|_2 \leq k$ 。令 v 为单位向量 $\frac{y-x}{\|y-x\|_2}$ 。我们有

$$k^2 = k \frac{\partial f(x_t)}{\partial v} = k \langle v, \nabla f(x_t) \rangle = \langle kv, \nabla f(x_t) \rangle \leq \|kv\|_2 \|\nabla f(x_t)\|_2 = k^2. \quad (2-28)$$

因为等式仅在 $\nabla f(x_t) = kv = k \frac{y-x}{\|y-x\|_2}$ 是成立, 所以我们有 $\nabla f(x_t) = k \frac{y-x}{\|y-x\|_2}$ 。□

2.A.5 Wasserstein 距离新对偶式的证明

Wasserstein 距离定义如下:

$$W_1(P_r, P_g) = \inf_{\pi \in \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \pi} [d(x, y)], \quad (2-29)$$

其中 $\Pi(P_r, P_g)$ 表示所有的满足边缘分布分别为 P_r 和 P_g 的概率测度。

Wasserstein 距离的 Kantorovich-Rubinstein (KR) 对偶式^[66] 写作:

$$\begin{aligned} W_{KR}(P_r, P_g) &= \sup_f \mathbb{E}_{x \sim P_r} [f(x)] - \mathbb{E}_{x \sim P_g} [f(x)], \\ &\text{s.t. } f(x) - f(y) \leq d(x, y), \forall x, \forall y. \end{aligned} \quad (2-30)$$

我们将证明 Wasserstein 距离的对偶式也可以被写作:

$$\begin{aligned} W_{LL}(P_r, P_g) &= \sup_f \mathbb{E}_{x \sim P_r} [f(x)] - \mathbb{E}_{x \sim P_g} [f(x)], \\ &\text{s.t. } f(x) - f(y) \leq d(x, y), \forall x \in S_r, \forall y \in S_g, \end{aligned} \quad (2-31)$$

注意这个新的对偶式的核心是进一步放松了 KR 对偶式中关于 f 的约束。

定理 2.22 若 $W_{KR}(P_r, P_g) = W_1(P_r, P_g)$, 则 $W_{KR}(P_r, P_g) = W_{LL}(P_r, P_g) = W_1(P_r, P_g)$ 。

证明

(i)

对于任意的 f , 如果它满足 “ $f(x) - f(y) \leq d(x, y), \forall x, \forall y$ ”, 那么它必定也满足 “ $f(x) - f(y) \leq d(x, y), \forall x \in S_r, \forall y \in S_g$ ”。因此, $W_{KR}(P_r, P_g) \leq W_{LL}(P_r, P_g)$ 。

(ii)

令 $F_{LL} = \{f \mid f(x) - f(y) \leq d(x, y), \forall x \in S_r, \forall y \in S_g\}$ 。

令 $A = \{(x, y) \mid x \in S_r, y \in S_g\}$ and $I_A = \begin{cases} 1, & (x, y) \in A; \\ 0, & \text{otherwise} \end{cases}$ 。

令 A^c 表示 A 的补集, 并相应地定义 I_{A^c} 。

对于任意的 $\pi \in \Pi(P_r, P_g)$, 我们有:

$$\begin{aligned}
W_{LL}(P_r, P_g) &= \sup_{f \in F_{LL}} \mathbb{E}_{x \sim P_r} [f(x)] - \mathbb{E}_{x \sim P_g} [f(x)] \\
&= \sup_{f \in F_{LL}} \mathbb{E}_{(x,y) \sim \pi} [f(x) - f(y)] \\
&= \sup_{f \in F_{LL}} \mathbb{E}_{(x,y) \sim \pi} [(f(x) - f(y)) I_A] + \mathbb{E}_{(x,y) \sim \pi} [(f(x) - f(y)) I_{A^c}] \\
&= \sup_{f \in F_{LL}} \mathbb{E}_{(x,y) \sim \pi} [(f(x) - f(y)) I_A] \\
&\leq \mathbb{E}_{(x,y) \sim \pi} [\|y - x\| I_A] \\
&\leq \mathbb{E}_{(x,y) \sim \pi} [d(x, y)].
\end{aligned}$$

$$\begin{aligned}
W_{LL}(P_r, P_g) &\leq \mathbb{E}_{(x,y) \sim \pi} [d(x, y)], \forall \pi \in \Pi(P_r, P_g) \\
\Rightarrow W_{LL}(P_r, P_g) &\leq \inf_{\pi \in \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \pi} [d(x, y)] = W_1(P_r, P_g). \\
(iii) \quad &\text{结合 (i) 和 (ii), 我们有 } W_{KR}(P_r, P_g) \leq W_{LL}(P_r, P_g) \leq W_1(P_r, P_g)。 \\
&\text{因为 } I(P_r, P_g) = W_1(P_r, P_g), \text{ 所以 } I(P_r, P_g) = W_{LL}(P_r, P_g) = W_1(P_r, P_g)。 \quad \square
\end{aligned}$$

2.B 梯度无信息问题的实践表现

为了研究梯度无信息问题在实际训练过程中的表现, 我们做了一些列实验, 观察在不同参数设定下梯度无信息问题的表现。我们采用 Least-Squares GAN^[20] 作为判别空间无限制 GAN 模型的代表。我们将训练趋于收敛后的判别函数以及它的梯度可视化如下。

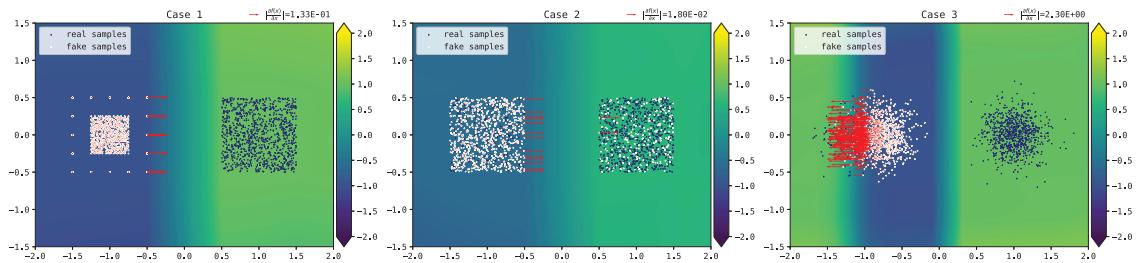


图 2-8 优化器: ADAM; 学习率: 1e-5; MLP 结构, RELU; 层数: 1; 隐层节点数: 1024。
Figure 2-8 Optimizer: ADAM; Learning Rate: 1e-5; MLP, RELU; #Layers: 1; #Hidden Units: 1024.

这些实验表明, 在实际训练过程中, 判别函数 f 的具体形态高度取决于网络结构以及其他各种超参数。给定有限的网络表达能力, 神经网络趋于学到在表达能力范围内的最佳判别函数。当神经网络可以近似学到最优判别函数时, 所学到的判别函数如何靠近最优判别函数, 以及判别函数的在未定义区域的取值高度取决于优化的细节和网络的特征。

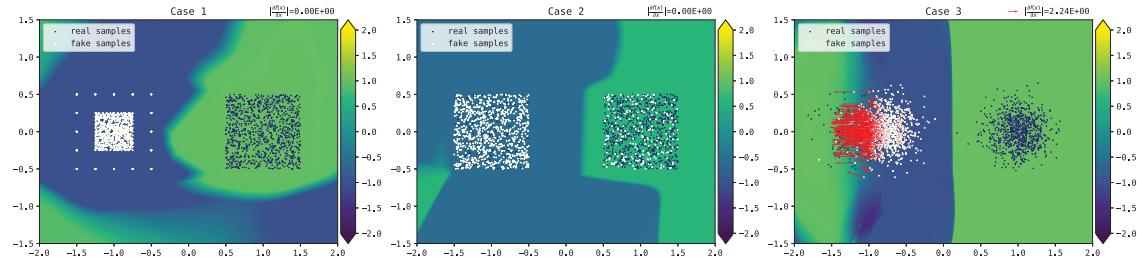


图 2-9 优化器: ADAM; 学习率: 1e-2; MLP 结构, RELU; 层数: 4; 隐层节点数: 1024。

Figure 2-9 Optimizer: ADAM; Learning Rate: 1e-2; MLP, RELU; #Layers: 4; #Hidden Units: 1024.

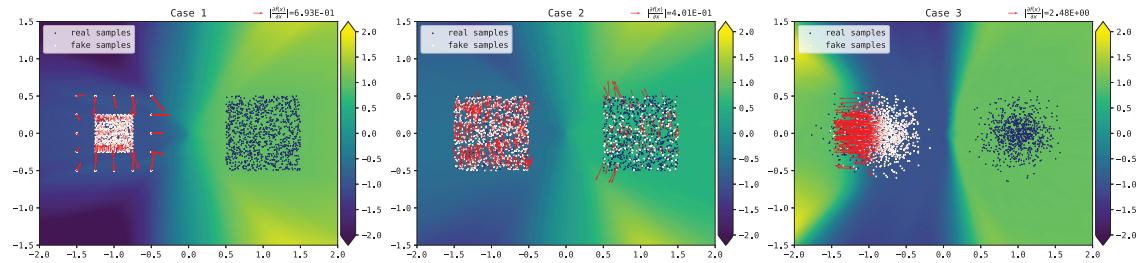


图 2-10 优化器: ADAM; 学习率: 1e-5; MLP 结构, RELU; 层数: 4; 隐层节点数: 1024。

Figure 2-10 Optimizer: ADAM; Learning Rate: 1e-5; MLP, RELU; #Layers: 4; #Hidden Units: 1024.

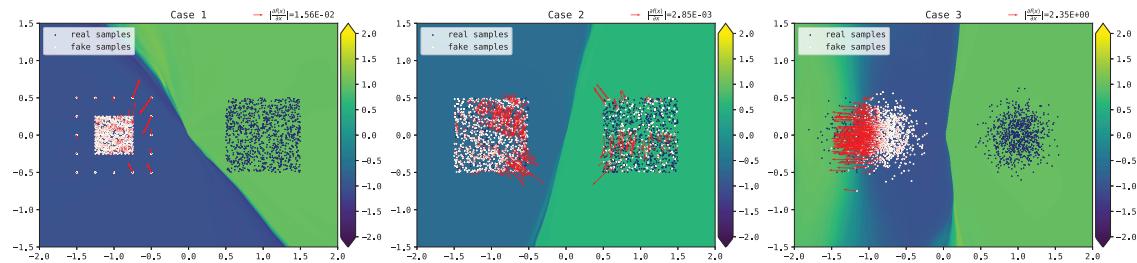


图 2-11 优化器: SGD; 学习率: 1e-3; MLP 结构, SELU; 层数: 64; 隐层节点数: 128。

Figure 2-11 Optimizer: SGD; Learning Rate: 1e-3; MLP, SELU; #Layers: 64; #Hidden Units: 128.

2.C 技术细节与延拓

2.C.1 满足公式(2-11)的 ϕ 和 φ

Lipschitz GANs 中 ϕ 和 φ 需要满足公式 (2-11)。公式(2-11)实际上不是一个很强的约束, 存在非常多满足条件的例子, 如: $\phi(x) = \varphi(-x) = x$, $\phi(x) = \varphi(-x) = -\log(\sigma(-x))$, $\phi(x) = \varphi(-x) = x + \sqrt{x^2 + \alpha}$ 其中 $\alpha > 0$, $\phi(x) = \varphi(-x) = \exp(x)$ 等等。我们讲这些例子的函数曲线可视化如图2-13。

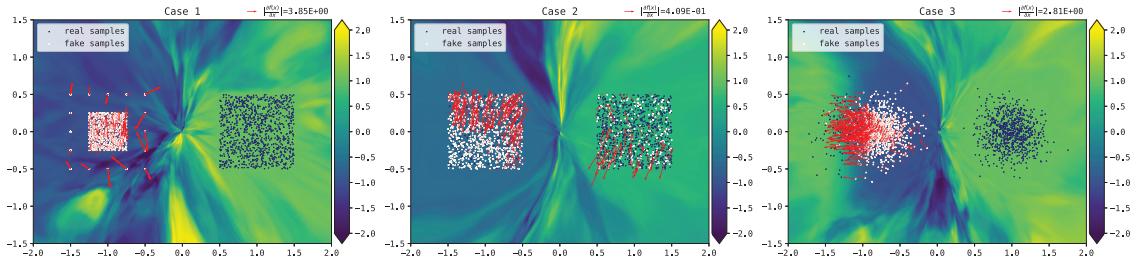


图 2-12 优化器: SGD; 学习率: 1e-4; MLP 结构, SELU; 层数: 64; 隐层节点数: 128。
Figure 2-12 Optimizer: SGD; Learning Rate: 1e-4; MLP, SELU; #Layers: 64; #Hidden Units: 128.

想要设计一个满足公式(2-11)的函数, 我们通常只需要寻找一个梯度不降的递增函数, 令其为 ϕ , 然后令 $\phi(x) = \varphi(-x)$ 即可。

可以注意到, 对一个函数的放缩以及偏移可以简单地得到更多的满足条件的同类型的 ϕ 和 φ 。而不同类型函数之间的线性组合也满足公式(2-11)。

2.C.2 实验细节以及更多的结果

在我们的实验中, 对于涉及真实数据 (CIFAR-10, Tiny Imagenet 以及 Oxford 102) 的部分实验, 我们沿用 WGAN-GP 中的 ResNet 结构^[35]。网络的具体结构我们列在了表2-3中。我们采用 Adam 优化器, 其中令 beta1=0.0, beta2=0.9。初始学习率为 0.0002, 学习率在 200000 个迭代中递减到 0。我们每更新一次生成器后, 更新 5 次判别器。我们在所有实验中, 采用最大梯度惩罚实验 Lipschitz 约束, 并为每一个实验在 [0.01, 0.1, 1.0, 10.0] 中选择最优的 λ 。对于所有列在表2-2中的实验, 我们仅仅改变 ϕ 和 φ , 以及数据集, 而保持其他所有参数和设定不变。

我们将 IS 训练曲线作图如图2-14和图2-15。我们将可视化的结构放在了图2-16和图2-17中。另外, 作为一个额外的实验, 我们也将 Oxford 102 的结果可视化在了图2-18中。

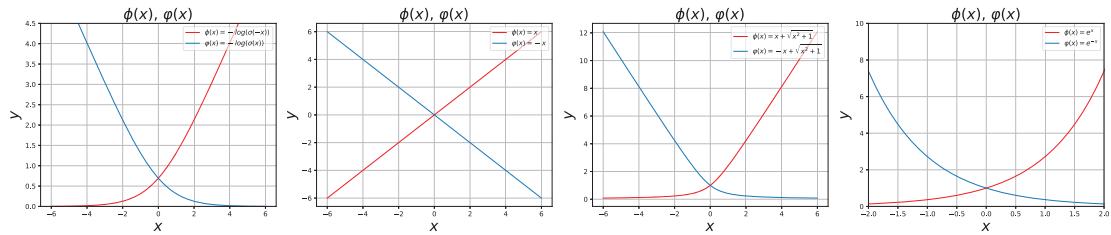


图 2-13 满足公式(2-11)的 ϕ 和 φ 示例。

Figure 2-13 Various ϕ and φ that satisfy Equation (2-11).

表 2-3 网络结构。

Table 2-3 Network structures.

生成器:

| 算子 | 卷积核 | 重采样 | 输出维度 |
|--------------|-----|-----|-----------|
| Noise | N/A | N/A | 128 |
| Linear | N/A | N/A | 128×4×4 |
| 残差模块 | 3×3 | UP | 128×8×8 |
| 残差模块 | 3×3 | UP | 128×16×16 |
| 残差模块 | 3×3 | UP | 128×32×32 |
| 卷积 & Tanh 激活 | 3×3 | N/A | 3×32×32 |

判别器:

| 算子 | 卷积核 | 重采样 | 输出维度 |
|---------------|-------|------|-----------|
| 残差模块 | 3×3×2 | Down | 128×16×16 |
| 残差模块 | 3×3×2 | Down | 128×8×8 |
| 残差模块 | 3×3×2 | N/A | 128×8×8 |
| 残差模块 | 3×3×2 | N/A | 128×8×8 |
| Relu & 激活均值池化 | N/A | N/A | 128 |
| 线性层 | N/A | N/A | 1 |

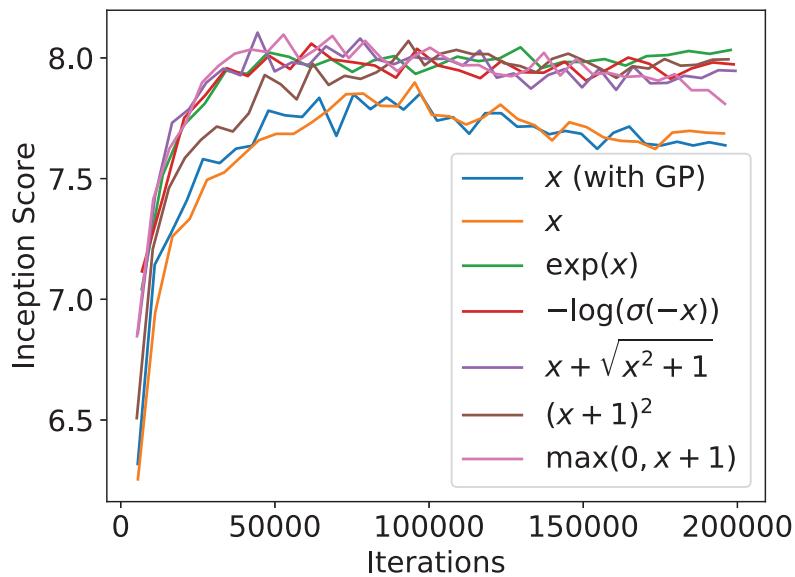


图 2-14 CIFAR-10 上的 Inception Score 训练曲线。

Figure 2-14

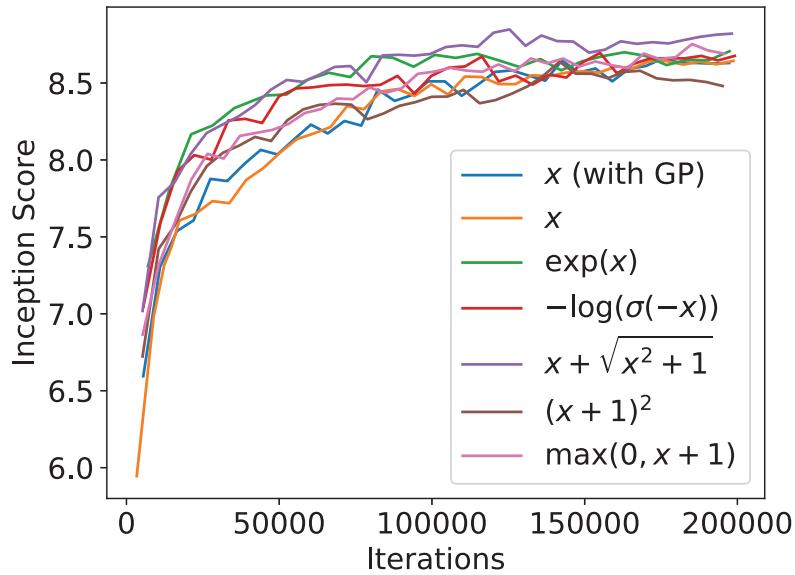


图 2-15 Tiny ImageNet 上的 Inception Score 训练曲线。

Figure 2-15



图 2-16 采用不同损失度量函数的 LGANs。随机采样的样本。数据集 CIFAR-10。

Figure 2-16 Random samples of LGANs with different loss metrics on CIFAR-10.



图 2-17 采用不同损失度量函数的 LGANs。随机采样的样本。数据集 Tiny Imagenet。

Figure 2-17 Random samples of LGANs with different loss metrics on Tiny Imagenet.



图 2-18 采用不同损失度量函数的 LGANs。随机采样的样本。数据集 Tiny Imagenet。

Figure 2-18 Random samples of LGANs with different loss metrics on Oxford 102.



图 2-19 在 LGANs 中判别函数关于生成样本的梯度指向真实样本。该实验中 P_r 由十张真实图片组成，而 P_g 是一个高维的高斯分布。上图：奇数列是来此生成分布的样本 $x \in S_g$ ，而与之相邻的偶数列是它们的梯度 $\nabla_x f^*(x)$ 。下图：最左侧是生成样本 $x \in S_g$ ，第二列是他们的梯度 $\nabla_x f^*(x)$ ，中间是 $x + \epsilon \cdot \nabla_x f^*(x)$ 伴随着 ϵ 递增，最右侧是真实分布中与之最接近样本 $y \in S_r$ 。
 Figure 2-19 The gradient of LGANs with real world data, where P_r consists of ten images and P_g is Gaussian noise. Up: Each odd column are $x \in S_g$ and the nearby column are their gradients $\nabla_x f^*(x)$. Down: the leftmost in each row is $x \in S_g$, the second are their gradients $\nabla_x f^*(x)$, the interiors are $x + \epsilon \cdot \nabla_x f^*(x)$ with increasing ϵ , and the rightmost is the nearest $y \in S_r$.

第三章 生成对抗网络的学习算法

生成对抗网络的训练不稳定性一直是令人头疼的问题之一。很多人从优化算法的角度解释和解决生成对抗的训练不稳定性问题。Adam 作为最典型的自适应学习率算法，被发现了能很好地稳定生成对抗网络的训练。而最近有研究发现，在生成对抗网络优化中非常好用的 Adam 算法在某些情况下存在不收敛性问题。本章中，我们就 Adam 的不收敛性问题展开研究，以期对优化算法深入理解，帮助我们进一步理解生成对抗网络的收敛性问题和优化问题。

Adam 作为深度学习中被广泛采用的优化算法之一，一直以来优化表现稳定而优异。关于 Adam 的收敛性问题的提出，吸引了很多研究者的关注和兴趣。而现存的用以解决 Adam 收敛性问题的方法存在潜在的低效率问题。在本章中，我们深入探索 Adam 不收敛性问题的原因，并基于此提出一套全新的关于 Adam 不收敛问题的解释和解决方案。

具体而言，我们发现在 Adam 算法中，二阶矩量 v_t 和当前梯度存在正相关性，而这会直接导致梯度更新的步幅存在偏向性：数值较大的梯度倾向于步幅较小，而数值较小的梯度倾向于步幅较大。我们论证步幅的偏向性是 Adam 不收敛问题的根本来源，并证明如果该偏向性完全去除，也即，若保证步幅和梯度无关，即可保证收敛性。我们从去相关性的角度，提出 AdaShift 算法，其核心思想是通过时序偏移使得二阶矩量不受当前梯度的影响。在假设梯度时序无关的前提下，实现去除二阶矩量和梯度相关性。

我们通过大量实验验证新算法的有效性。实验上，AdaShift 呈现出了颇具竞争力的表现，在各种设定下，收敛速度和泛化能力均不弱于 Adam。而在一些问题上，AdaShift 的表现明显优于 Adam，这个表现可能和 AdaShift 能更好地保证收敛性问题有关。同时，我们也观察到优化器的改进确实有助于生成对抗网络的优化。

3.1 引言

带有自适应学习率的一阶优化算法能很好地应用于大规模的优化问题，从而在深度学习中扮演着重要的角色。记 $g_t \in \mathbb{R}^n$ 为损失函数 f 在时刻 t 关于参数 $\theta \in \mathbb{R}^n$ 的梯度，则这类算法的通用更新规则可以被写作^[77]：

$$\theta_{t+1} = \theta_t - \frac{\alpha_t}{\sqrt{v_t}} m_t. \quad (3-1)$$

在上式中， $m_t \triangleq \phi(g_1, \dots, g_t) \in \mathbb{R}^n$ 是关于历史梯度的函数，用作对全局梯度的近似； $v_t \triangleq \psi(g_1, \dots, g_t) \in \mathbb{R}_+^n$ 是一个非负的 n 维的向量，扮演着为梯度的每个维度自动调节学习率的作用； α_t 是基础学习率，而 $\frac{\alpha_t}{\sqrt{v_t}}$ 可以看作是 m_t 的自适应更新步长。

$\phi(g_1, \dots, g_t)$ 的常见选择是梯度的指数平均，如 Momentum^[78] 和 Adam^[79]，可以认为是对梯度的一阶矩量的近似。指数平均计算代价低而同时能很好地平滑训练过程中的梯度抖动。 $\psi(g_1, \dots, g_t)$ 通常被定义为平方梯度的指数平均，如 Adadelta^[80]， RMSProp^[81]，Adam^[79] 以及 Nadam^[82]，可以认为是对梯度的二阶矩量的近似。

Adam^[79] 是一个典型的自适应学习率算法，它采用梯度的一阶和二阶的指数平均，在此基础上引入了初始偏差校正。通常而言，Adam 表现鲁邦而高效，无论是在密集梯度还是稀疏梯度的场景下，这使得它在深度学习中被广泛采用。然而，研究人员发现 Adam 在某些情况下无法收敛到最优解。^[77] 指出 Adam 不收敛问题的关键在于下面这个量：

$$\Gamma_t \triangleq \left(\frac{\sqrt{v_t}}{\alpha_t} - \frac{\sqrt{v_{t-1}}}{\alpha_{t-1}} \right), \quad (3-2)$$

在 Adam 的证明中，它被假设是正定的，但是不幸的是，这个假设在实践过程中并不总能成立。他们提供了一些反例并展示破坏 Γ_t 的正定性会导致 Adam 的不收敛性问题。

根据以上分析，^[77] 提出了 Adam 的两个变种：AMSGrad 和 AdamNC。其核心思想是保证 Γ_t 的正定性。具体而言，AMSGrad 定义 \hat{v}_t 为 v_t 的历史最大值，也即， $\hat{v}_t = \max \{v_i\}_{i=1}^t$ ，并将 Adam 更新公式中的 v_t 替换成 \hat{v}_t ，从而保证 v_t 不降，进而使得 Γ_t 的正定性得到保证。AdamNC 令 v_t 具有对历史梯度的长期记忆，并将 v_t 定义为历史梯度的加权平均，从而使得 v_t 在训练过程中趋于平稳。

尽管以上两个算法在一定程度上能解决 Adam 的不收敛性问题，他们算法中的问题也显而易见：过度强调 Γ_t 的正定性，必然会导致 v_t 的数值过大，不能及时自适应到梯度的当前幅度。带来的最直接的潜在问题是训练速度上的减缓，因为 v_t 过大将自然地使得学习率将减小。

在本章中，我们从全新的角度理解自适应学习率算法，它同时也为 Adam 的不收敛性问题提供了新的解决思路。具体而言，在小节3.3中，我们通过分析梯度 g_t 的累积更新步长来分析 Adam 的不收敛性问题。我们发现，在自适应学习率算法中，对于不同梯度的更新步长普遍存在偏向性：大梯度的步长倾向于较小，而小梯度的步长倾向于较大。我们说明这样一个偏向性的更新步长源自于二阶矩量 v_t 和梯度 g_t 之前的正相关性。我们认为这是 Adam 不收敛性问题的根本来源。

在小节3.4中，我们进一步证明去掉 v_t 和 g_t 的相关性，将能得到相等的不具有偏向性的期望步长，从而解决 Adam 的不收敛性问题。我们随即提出 AdaShift，一个 Adam 的去相关性的版本。AdaShift 通过在计算 v_t 的时候采用经时序偏移之后的 g_t 来去除 v_t 和 g_t 的相关性。最后，在小节3.5中，我们实验分析 AdaShift 的性能。我们观察到 AdaShift 在解决 Adam 不收敛性问题的同时具有和 Adam 相匹敌甚至稍微更有效的优化性能和泛化性能。

3.2 准备知识

Adam 在 Adam 中， m_t 和 v_t 被分别定义为 g_t 和 g_t^2 的指数平均：

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad \text{and} \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \quad (3-3)$$

其中 $\beta_1 \in [0, 1]$, $\beta_2 \in [0, 1)$ 分别为 m_t 和 v_t 的指数平均衰减系数。一般令 $m_0 = 0$, $v_0 = 0$ 。以上是它们的迭代表达式，其累积表达式也可以被写作：

$$m_t = (1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i} g_i \quad \text{and} \quad v_t = (1 - \beta_2) \sum_{i=1}^t \beta_2^{t-i} g_i^2. \quad (3-4)$$

指数平均如其名可以认为是在求数据的以指数衰减位系数的平均值（期望）。为了避免在初始时对于期望值的估计的误差，[79] 为指数平均引入了初始偏差校正。以 m_t 为例，初始偏差矫正工作如下：

$$m_t = \frac{(1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i} g_i}{(1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i}} = \frac{\sum_{i=1}^t \beta_1^{t-i} g_i}{\sum_{i=1}^t \beta_1^{t-i}} = \frac{(1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i} g_i}{1 - \beta_1^t}. \quad (3-5)$$

即，在 m_t 的基础上除以指数衰减系数的累积值，该数值等于 $1 - \beta_1^t$ 。

在线优化问题 一个在线优化问题（online optimization problem）包含一系列代价函数 $f_1(\theta), \dots, f_t(\theta), \dots, f_T(\theta)$ ，而优化器为每个时刻预测一个参数 θ_t 并在不提前知道的代价函数 $f_t(\theta)$ 上评测该参数的好坏。优化器的性能一般用遗憾值（regret）表示：

$$R(T)/T \triangleq \sum_{t=1}^T [f_t(\theta_t) - f_t(\theta^*)]$$

也即，各个时刻的在线预测 $f_t(\theta_t)$ 相较于一个固定的最优参数的代价 $f_t(\theta^*)$ 的差异总和，其中

$$\theta^* = \arg \min_{\theta \in \vartheta} \sum_{t=1}^T f_t(\theta)$$

是参数可行空间 ϑ 中最佳的固定参数。有时我们也用平均遗憾值来衡量有花期的性能，平均遗憾值定义为： $\frac{R(T)}{T}$ 。

反例 [77] 中指出对于任何的固定的 β_1 和 β_2 均存在一个在线优化问题使得 Adam 的平均遗憾值不为零，也即，Adam 不能收敛到最优解。反例的序列化的版本为：

$$f_t(\theta) = \begin{cases} C\theta, & \text{if } t \bmod d = 1; \\ -\theta, & \text{otherwise,} \end{cases} \quad (3-6)$$

这里 C 是一个相对较大的常数， d 是周期长度。在公式(3-6)中，大多数 $f_t(\theta)$ 的梯度是 -1 ，但是由于大梯度 C 的存在，整个周期的整体梯度是大于零的。这意味着，优化器应该减小 θ_t 以降低目标函数。然而，如 [77] 所说， θ 的累积更新在一些情况下是正好相反的，也即 θ_t 增大，因此，Adam 不能正确收敛。[77] 认为出现该问题的原因是 $\Gamma_t \triangleq (\sqrt{v_t}/\alpha_t - \sqrt{v_{t-1}}/\alpha_{t-1})$ 的正定性假设没有得到保证。

[77] 还提供了一个随机化版本的反例，其中有限的几个代价函数以随机的方式出现，相比于序列化版本，随机化版本更加一般化也更接近现实情况。作为其中最简单的例子，在每个时刻 t ，代价函数 $f_t(\theta)$ 独立地从以下分布中采样：

$$f_t(\theta) = \begin{cases} C\theta, & \text{with probability } p = \frac{1+\delta}{C+1}; \\ -\theta, & \text{with probability } 1-p = \frac{C-\delta}{C+1}, \end{cases} \quad (3-7)$$

其中 δ 是一个小的正常数，且 $\delta < C$ 。以上问题中的期望代价函数是 $F(\theta) = \frac{1+\delta}{C+1}C\theta - \frac{C-\delta}{C+1}\theta = \delta\theta$ ，因此优化器应该较小 θ 以降低目标函数。[77] 证明，当 C 足够大时，Adam 中的累积参数更新是正的而 θ 趋于增大。

基础解法 作为解法之一，[77] 的核心思路是保持 Γ_t 它的严格正定性，例如保持 v_t 不减或者采用递增且趋于 1 的 β_2 。实际上，保持 Γ_t 正定并不是解决该问题的唯一方法。举个例子，给定任何一个周期重复的在线优化问题，形如公式(3-6)，只要 β_1 足够大，Adam 也会收敛。形式化地，我们可以给出以下定理：

定理 3.1 (关于 β_1 的影响) 给定任何一个周期性地重复的在线优化问题，记其周期长度为 d 。如果 $\exists G \in \mathbb{R}$ 使得 $\|\nabla f_t(\theta)\|_\infty \leq G$ ，且 $\exists T \in \mathbb{N}, \exists \epsilon_2 > \epsilon_1 > 0$ 使得 $\epsilon_1 < \frac{\alpha_t}{\sqrt{v_t}}G^2 < \epsilon_2$ 对于所有 $t > T$ 均成立，则对于任何的固定的 $\beta_2 \in [0, 1)$ ，存在 $\beta_1 \in [0, 1)$ 使得 Adam 的平均遗憾值 $\leq \epsilon_2$ 。

以上定理背后的直觉是，假若 $\beta_1 \rightarrow 1$ ，则 $m_t \rightarrow \sum_{i=1}^d g_i/d$ ，也即， m_t 接近于周期内的平均梯度。因此，无论自适应学习率 $\alpha_t/\sqrt{v_t}$ 取值如何（只要是正数），Adam 均会朝着正确的方向收敛。

3.3 不收敛问题的原因：步长的偏向性

在这节中，我们结合上面给出的反例研究 Adam 的不收敛问题。我们发现自适应学习率算法中的根本的问题在于：二阶矩量 v_t 和 g_t 的大小是正相关的，这导致步长 $\alpha_t / \sqrt{v_t}$ 对于大的梯度倾向于较小，而对于小的梯度倾向于较大。我们断言这样的步长上的偏向性是不收敛问题的来源。

因为 Adam 中一阶矩量作为梯度的近似，任何一个梯度的影响其实都分散到了它的后续更新中。为了更严格地分析步长偏向性问题，我们首先定义每个梯度的净更新步长，以分析 g_t 对系统的累积影响。然后我们用净更新步长分析到 Adam 在反例中的步长偏向性表现。最后我们把，步长偏向性的讨论一般化。

3.3.1 净更新步长

当 $\beta_1 \neq 0$ 时，由于指数平均 m_t 的影响， g_t 的影响会存在于时刻 t 的所有后续更新中。对于时刻 i ($i \geq t$)，梯度 g_t 在 m_t 中的权重是 $(1 - \beta_1)\beta_1^{i-t}$ 。我们定义梯度的累积影响如下：

$$\begin{aligned} net(g_t) &\triangleq \sum_{i=t}^{\infty} \frac{\alpha_i}{\sqrt{v_i}} [(1 - \beta_1)\beta_1^{i-t} g_t] = k(g_t) \cdot g_t, \\ k(g_t) &\triangleq \sum_{i=t}^{\infty} \frac{\alpha_i}{\sqrt{v_i}} (1 - \beta_1)\beta_1^{i-t}. \end{aligned} \quad (3-8)$$

我们把 g_t 的净更新记为 $net(g_t)$ ，表示该梯度的累积影响，而 g_t 的净更新的步长记为 $k(g_t)$ 。注意 $k(g_t)$ 取决于 $\{v_i\}_{i=t}^{\infty}$ ，而在 Adam 中，如果 $\beta_2 \neq 0$ ，那么 $\{v_i\}_{i=t}^{\infty}$ 中的所有项都和 g_t 相关。因此， $k(g_t)$ 是 g_t 的一个函数。

值得一提的是，在 Momentum 算法中， v_t 可以认为是常数 1。如果进一步假设 α_t 是一个常数，则我们有 $k(g_t) = \alpha_t$ 且 $net(g_t) = \alpha_t g_t$ 。这意味着，在 Momentum 中，每个梯度的更新步长都是一致的，而且和原始 SGD (Stochastic Gradient Decent) 一样。因此，不难得出 Momentum 的收敛性和原始 SGD 相近。然而，在自适应学习率算法中， v_t 是关于历史梯度的一个函数，这就是它的收敛性变成了不平凡的。

3.3.2 在线优化问题中的反例

注意到梯度的净更新步长的表达式，公式(3-8)，中包含 v_t 。在进一步分析 Adam 的收敛性问题之前。我们先研究 v_t 的在线更新问题中的反例上的行为模式。我们首先看反例的序列化版本，因为它是确定性的，我们可以直接写出 v_t 的表达式如下：

引理 3.2 在序列在线优化问题(3–6)中, 记 $\beta_1, \beta_2 \in [0, 1)$ 为一阶矩量和二阶矩量的衰减系数, $d \in \mathbb{N}$ 为周期长度, $n \in \mathbb{N}$ 为当前周期数, 而 $i \in \{1, 2, \dots, d\}$ 为当前周期内的时间点, 则有:

$$\lim_{n \rightarrow \infty} v_{nd+i} = \frac{1 - \beta_2}{1 - \beta_2^d} (C^2 - 1) \beta_2^{i-1} + 1. \quad (3-9)$$

给定 v_t 的表达式(3–9), 我们现在研究每个梯度的净更新步长。从最简单的情况开始, 我们先假设 $\beta_1 = 0$, 此时, 我们有

$$\lim_{n \rightarrow \infty} k(g_{nd+i}) = \lim_{n \rightarrow \infty} \frac{\alpha_t}{\sqrt{v_{nd+i}}}. \quad (3-10)$$

可以看到, 在每个周期中 v_{nd+i} 单调递减, 因此, $k(g_{nd+i})$ 单调递增。具体而言, 对于每个周期里的第一个梯度, 也即对于梯度 $g_{nd+1} = C$, 它的步长最小, 但是它却代表着整体的梯度方向。相比之下, 后续的梯度值为 -1 的梯度的步长相对更大, 尽管他们与整体梯度方向相反。

我们进一步考虑一般性的情形: $\beta_1 \neq 0$ 。结果呈现在下面的引理中

引理 3.3 在序列在线优化问题(3–6)中, 当 $n \rightarrow \infty$, 第 n 个周期的第 i 个梯度的净更新步长 $k(g_{nd+i})$ 满足: $\exists 1 \leq j \leq d$ 使得

$$\lim_{n \rightarrow \infty} k(C) = \lim_{n \rightarrow \infty} k(g_{nd+1}) < \lim_{n \rightarrow \infty} k(g_{nd+2}) < \cdots < \lim_{n \rightarrow \infty} k(g_{nd+j}), \quad (3-11)$$

以及

$$\lim_{n \rightarrow \infty} k(g_{nd+j}) > \lim_{n \rightarrow \infty} k(g_{nd+j+1}) > \cdots > \lim_{n \rightarrow \infty} k(g_{nd+d+1}) = \lim_{n \rightarrow \infty} k(C), \quad (3-12)$$

其中 $k(C)$ 表示梯度值为 C 的梯度的净更新步长。

引理3.3中的结果表明, 在序列在线优化问题(3–6)中, 净更新步长始终是有偏向性的。具体而言, 大梯度 C 的净更新不尝试最小的, 而小梯度 -1 有着相对较大的净更新步长。这样一个有偏向性的净更新步长使得 Adam 有朝着错误方向收敛的可能。我们在附录3.C研究 Adam 出现不收敛问题的临界条件。

类似的分析可以作用于随机在线优化问题(3–7), 我们推导出每个梯度的净更新步长, 具体表示较为复杂, 我们整理有用的结论如下:

引理 3.4 在随机在线优化问题(3–7)中, 假设 $\alpha_t = 1$, 有 $k(C) < k(-1)$, 其中 $k(C)$ 表示梯度值为 C 的梯度的期望净更新步长, $k(-1)$ 表示梯度值为 -1 的梯度的净更新步长。

尽管在随机情况下，梯度的净更新步长具体表达式较为复杂，但其实分析起来反而更加容易：相同梯度值的梯度有着同样的期望净更新步长，所以我们仅需要分析 $k(C)$ 和 $k(-1)$ 。根据引理3.4，我们可以看出在期望更新步长的意义下， $k(C)$ 相较于 $k(-1)$ 更小，也即，梯度 C 对系统的影响相对 -1 更小。

3.3.3 不收敛问题的一般性结论

如我们在上一节中所观察到的，这些反例有一个共同的特征，也即，大梯度的净更新步长通常小于小梯度的净更新步长。以上事实可以被理解为是 v_t 和 g_t 正相关的直接结果。我们知道 $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$ 。假设 v_{t-1} 和 g_t 无关，那么：当新梯度 g_t 到达时，如果 g_t 较大， v_t 也会倾向于较大；而如果 g_t 较小，则 v_t 也会倾向于较小。如果 $\beta_1 = 0$ ，则 $k(g_t) = \alpha_t / \sqrt{v_t}$ 。进而，大梯度有较小的净更新步长，而小梯度有较大的净更新步长。

如果考虑 $\beta_1 > 0$ 的情况，其实论证方向也基本类似。因为 $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$ 。假设 v_{t-1} 和 $\{g_{t+i}\}_{i=1}^\infty$ 与 g_t 无关，则：不仅 v_t 和 g_t 的幅度正相关，而且后续所有的 $\{v_i\}_{i=t}^\infty$ 都与 g_t 的幅度正相关。因为净更新步长 $k(g_t) = \sum_{i=t}^\infty \alpha_i / \sqrt{v_i} (1 - \beta_1) \beta_1^{i-t}$ 与 $\{v_i\}_{i=t}^\infty$ 中的 v_i 负相关。所以， $k(g_t)$ 和 g_t 的幅度负相关。所以，幅度较大的梯度其净更新步长倾向于较小，幅度较小的梯度其净更新步长倾向于较大。

从上面的论证可以看出步长偏向性不仅仅是 Adam 中的问题，而是自适应学习率算法中普遍存在的问题。更进一步的，只要自适应项 v_t 和 g_t 有相关性，步长均会有偏向性，而这种偏向性将导致不收敛情况的出现。构造反例其实可以遵循一个固定的模式：有一个大梯度的梯度代表着正确（整体）的梯度方向，有（些）小的梯度指向。因为大梯度的净更新步长较大，而小梯度的净更新步长较小，整体更新的方向就可以呈现出问题。

最后，我们想要强调一点，就算 Adam 的步长偏向性有时候并不导致梯度反方向，在一维情况下我们可以看到这种相消的效果会阻碍收敛。而可以想象在高维情况下，由于每个梯度的步长的偏向性的存在，整体梯度方向势必和整体梯度方向存在偏差。

3.4 AdaShift：通过时序偏移去除相关性

按照前面的讨论，我们认为 Adam 的不收敛性问题源自于 v_t 和 g_t 之间的正相关性。当前，我们有两个可能的解决方案：(1) 使得 v_t 像一个常数，这样自然可以消除 v_t 和 g_t 之间的相关性，如：采用较大或递增趋于 1 的 β_2 ，保持 v_t 不减^[77]；(2) 采用较大或递增趋于 1 的 β_1 （定理3.1），也即通过具有长期记忆的一阶指数平均

来缓解净更新步长的偏向性。然而，这两者都没有从根本上解决补偿的偏向性问题。

由 v_t 造成的困境迫使我们重新思考它所扮演的角色。在自适应学习率算法中， v_t 扮演着近似二阶矩量的角色，它反映了梯度平均意义下的幅度。在自适应学习率 $\alpha_t/\sqrt{v_t}$ 的作用下， g_t 的更新步长被 $\sqrt{v_t}$ 缩放从而实现了 g_t 更新步长的尺度无惯性。这个性质在实际训练中有着重要的意义，它使得训练更加鲁邦且易于控制。然而，当前的 v_t 的更新策略，也即 $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$ ，带来了 v_t 和 g_t 之间的正相关性，进而导致大梯度的影响被削弱而小梯度的影响被增强，最终导致不收敛问题。因此，在不改变自适应学习率方法主要特性的基础上解决该问题的关键是要让 v_t 是一个能反映当前梯度幅度的量，而同时要与 g_t 无关。

为了强调去相关性的重要性，我们形式化地给出以下定理：

定理 3.5 (去相关保证收敛性) 给定任何一个周期性地重复的在线优化问题，假设其每个周期内的代价函数分别为 $\{f_1(\theta), \dots, f_t(\theta), \dots, f_n(\theta)\}$ ，假设 $\beta_1 = 0$ 且 α_t 是固定的，我们有：如果 v_t 服从一个固定分布且与当前梯度 g_t 无关，那么每个梯度的更新步长服从相同的分布，而期望更新步长一致。

令 P_v 表示 v_t 的分布，在周期性重复的在线优化问题下，每个梯度 g_t 的期望更新步长为：

$$\mathbb{E}[k(g_t)] = \sum_{i=t}^{\infty} \mathbb{E}_{v_i \sim P_v} \left[\frac{\alpha_i}{\sqrt{v_i}} (1 - \beta_1) \beta_1^{i-t} \right]. \quad (3-13)$$

给定 P_v 与 g_t 独立，期望更新步长 $\mathbb{E}[k(g_t)]$ 也和 g_t 独立，并且对于不同的梯度而言保持不变。在期望更新步长是一个固定常数的基础上，我们可以论证算法的收敛性和原始 SGD 接近。

Momentum^[78] 可以被看做是令 v_t 等于一个常数，使得 v_t 和 g_t 独立了。进一步的，在我们的视角下，使用递增的 β_2 (AdamNC) 或者令 \hat{v}_t 为 v_t 的最大值 (AMSGrad) 也是的 v_t 几乎固定为一个常数。将 v_t 固定为一个常数并不是很理想的一个解决方案，因为它损害了自适应学习率算法的自适应能力。

我们接下来介绍我们所提出的使 v_t 和 g_t 独立的方法。它基于一个假设：不同时间点上的梯度相互独立。我们将首先介绍时序偏移的基本想法，然后我们将方法进一步拓展以利用其他维度上信息，最后我们为所提出的算法补上一阶梯度近似。所提出的方法的伪代码如下：

算法 3-1 AdaShift

Require: $n, \beta_1, \beta_2, \phi, \theta_0, \{f_t(\theta)\}_{t=1}^T, \{\alpha_t\}_{t=1}^T, \{g_{-t}\}_{t=0}^{n-1}$,

```

1: set  $v_0 = 0$ 
2: for  $t = 1$  to  $T$  do
3:    $g_t = \nabla f_t(\theta_t)$ 
4:    $m_t = \sum_{i=0}^{n-1} \beta_1^i g_{t-i} / \sum_{i=0}^{n-1} \beta_1^i$ 
5:   for  $i = 1$  to  $M$  do
6:      $v_t[i] = \beta_2 v_{t-1}[i] + (1 - \beta_2) \phi(g_{t-n}^2[i])$ 
7:      $\theta_t[i] = \theta_{t-1}[i] - \alpha_t / \sqrt{v_t[i]} \cdot m_t[i]$ 
8:   end for
9: end for

```

10: // 出于简洁性的考虑，我们这里省略了偏差矫正等细节。完整算法见附录。

3.4.1 通过时序偏移去相关

在实际的训练场景中， $f_t(\theta)$ 通常会涉及不同的批量数据 x_t ，也即， $f_t(\theta) = f(\theta; x_t)$ 。因为批量数据采样的随机性，我们可以假设不同时刻的批量数据 x_t 是相互独立的。值得一提的是，如果假设 $f(\theta; x)$ 保持不变，则不同批量数据的梯度 $g_t = \nabla f(\theta; x_t)$ 是独立同分布的。

因此，我们可以改变 v_t 的更新法则，使得它的更新公式不再涉及 g_t 而是涉及 g_{t-n} ，这将使得 v_t 和 g_t 时序上存在偏移，从而不再相关：

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_{t-n}^2. \quad (3-14)$$

按照时序不相关的假设，任何两个时刻的梯度都不相关。因此 n 仅需要大于等于 1 即可。值得注意的是，在周期性重复的在线优化问题中， g_t 相互独立的假设并不成立。然而，在随机在线优化问题以及实践场景中，时序不相关的假设是一般成立的。

3.4.2 利用空间上的元素

大多数优化场景涉及大量的参数。 θ 的维度很高，因此 g_t 和 v_t 也是高维的。然而，在公式(3-14)中， v_t 是每个维度独立计算的，并且计算时仅考虑对应维上的信息。具体说来，我们仅适用 g_{t-n} 的第 i 维来计算 v_t 的第 i 维。换句话说，它仅仅利用了 $g_{t-n}[i]$ 和 $g_t[i]$ 之间的独立性，这里 $g_t[i]$ 表示 g_t 的第 i 维。

事实上，在多维情况下，我们可以进一步假设 g_{t-n} 的全部元素都和 g_t 的第 i 维独立。因此， g_{t-n} 的全部元素都可以被用到 v_t 的计算中来。我们从而提出在 v_t

的更新公式中引入函数 ϕ , 作用于 g_{t-n}^2 的全部元素上, 也即,

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) \phi(g_{t-n}^2). \quad (3-15)$$

我们方便指代, 我们称 g_{t-n} 中除 $g_{t-n}[i]$ 以外的其他元素为 g_{t-n} 的空间元素, 称 ϕ 为空间函数或者空间操作。

3.4.3 空间函数的选择: 降维与共享

空间操作 ϕ 原则上是可以任意选择的, 在我们的大部分实验中, 我们令 $\phi(x) = \max_i x[i]$, 这似乎是一个还不错的选择。

$\max_i x[i]$ 有一个附带的效果是将学习率自适应项 v_t 从一个向量变成了一个标量。这里, 一件重要的事情是, 我们不再将 v_t 解释为 g_t 的二阶矩量。此时, 我们仅把它看作一个和 g_t 独立的随机变量, 但同时反应整体梯度的大小。

在实际场景中, 如深度神经网络, θ 通常由许多个参数块组成, 例如: 每层的权重矩阵 (weight) 和偏差矫正量 (bias)。在深度神经网络中, 不同层的梯度的尺度 (方差) 通常是不一致的^[83, 84]。而梯度大小的不一样使得在采用 SGD 或者 Momentum 时很难为整个神经网络找到一个统一的合适的学习率。在传统自适应学习率算法中, 他们为每个维度分别做梯度尺度缩放, 从而实现了梯度的尺度不变性, 在某种意义下解决了以上问题。然而, Adam 在某些场景下并不能达到和 SGD 相当的泛化性能^[85, 86], 这可能和 Adam 过度自适应的学习率有关; 在 Adam 中各维度间的相对梯度大小也被去除了。

如果在时序去相关的基本上引入空间函数, 我们可以更自然地解决上面所提到的各层梯度尺度不一致的问题。做法如下: 对每一层 (或者每个参数块) 作用一个函数 ϕ , 该函数输出一个共享的标量, 作为该参数模块的自适应学习率 $v_t[i]$:

$$v_t[i] = \beta_2 v_{t-1}[i] + (1 - \beta_2) \phi(g_{t-n}^2[i]). \quad (3-16)$$

这个操作使得算法更接近 SGD, 因为每各模块中的相对梯度大小被保持住了。或者我们称之为自适应学习率 SGD, 因为它的学习率为 $\alpha_t / \sqrt{v_t[i]}$, 不受整体梯度尺度的影响。

如算法3-1所示, 参数 θ_t 、 g_t 以及 v_t 被分成了 M 个模块。每个模块包含相同类型或者属于神经网络同一层的参数。

3.4.4 一阶矩量的估计: 滑动平均窗口

一阶矩量估计, 通常是将 m_t 定义为 g_t 的指数平均, 是现代一阶优化算法的重要技巧。它能有效地缓解因批量数据带来的梯度波动。这一节中, 我们对我

们所提出的算法进行拓展，从而将一阶矩量的估计融合到算法中。

我们论证过二阶矩量 v_t 需要和 g_t 去相关。类似地，当引入一阶矩量估计的时候，我们需要保证 v_t 和 m_t 不相关，因为 m_t 替代了 g_t 的角色。基于我们的时序不相关的假设，最简单的实现 m_t 和 v_t 独立的方式是：让参与 m_t 的计算的梯度和 v_t 的独立。因此，我们使时序偏移量 $n > 1$ ，同时用时序偏移中不参与 v_t 计算的部分 ($\{g_{t-i}\}_{i=0}^{n-1}$) 估计一阶矩量：

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) \phi(g_{t-n}^2) \quad \text{and} \quad m_t = \frac{\sum_{i=0}^{n-1} \beta_1^i g_{t-i}}{\sum_{i=0}^{n-1} \beta_1^i}. \quad (3-17)$$

在公式(3-17)中， $\beta_1 \in [0, 1]$ 和 Adam 中类似，是不同时刻梯度权重的衰减系数。

该一阶矩量估计方法，可以看成是一个截断的指数平均，它只作用在做后的几个元素中。这似乎是被逼无奈的一个选择，因为我们不能像以前一样用到所有历史上的元素，但截断同时也是有好处的。截断的做法使得采用更大的 β_1 时不用担心会使得梯度估计时用到太过过时的梯度信息。在极端情况下，即 $\beta_1 = 1$ 时，这个截断的指数平均，退化成普通的算术平均。

我们在算法3-1中给出了简化版本的伪代码，并在附录中给出完整的版本。算法有以下参数：空间函数 ϕ ，时序偏移量 $n \in \mathbb{N}^+$ ，一阶矩量衰减系数 $\beta_1 \in [0, 1]$ ，二阶矩量衰减系数 $\beta_2 \in [0, 1]$ 以及学习率 α_t 。

小结 Adam 和所提出的算法的核心区别是后者在计算 v_t 时采用时序偏移 n 之后的梯度，然后用不参与计算 v_t 的部分梯度估计一阶矩量 m_t ，这使得 v_t 和 m_t 不相关，从而解决 Adam 因相关性导致的梯度步长存在偏向性的问题。

此外，基于我们所提出的新的理解自适应学习率方法的视角， v_t 不一定需要被定义为梯度二阶矩量估计。按照我们对该问题的分析， v_t 的核心作用是估计梯度的平均尺度，用以去除梯度的尺度，使得参数更新过程更加平滑稳定、学习率的设置更加容易等。因此，我们提出引入空间函数 ϕ ，它作用于时序偏移后的梯度，并且利用到空间上其他元素的信息；为了保留梯度不同维度间的相对大小关系，我们提议在空间函数 ϕ 中引入降维操作，使得不同维度共享自适应项 (v_t)。

我们将这个通过时序偏移实现 v_t 和 m_t 去相关的算法称为 **AdaShift**，用以指代其中的关键信息自适应学习率算法 (ADaptive learning rate method) 和时序偏移 (temporal SHIFT)。

3.5 实验

在本节中，我们通过实验通过实验分析和验证我们所提出的算法，并在各种测试任务上同 Adam、AMSGrad、SGD 等算法就训练和泛化性能上进行比较。在本节中，如果我们不额外指出，那么所报告的结果均为每个算法经过对于算法相关参数做网格搜索后得到的最优结果。该部分的实验代码公开于：<http://bit.ly/2NDXX6x>。

3.5.1 在线优化问题的反例

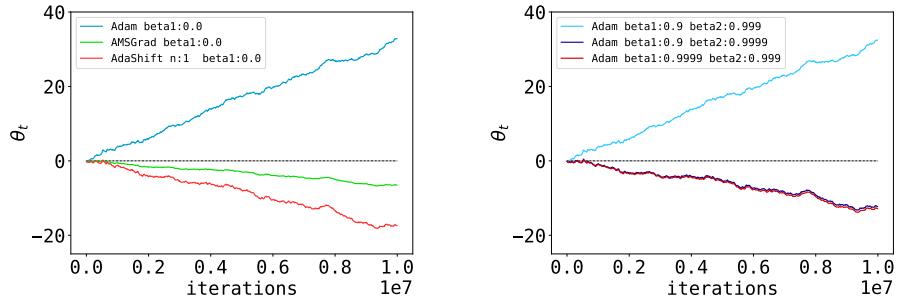
首先我们在随机在线优化问题(3–7)上测试和验证我们的算法。这里，我们令 $C = 101$ 且 $\delta = 0.02$ 。在这个试验中，我们比较了 Adam、AMSGrad 和 AdaShift 这几个自适应学习率算法。为了让比较是相对公平的，我们在所有算法中令 $\alpha = 0.001$, $\beta_1 = 0$, $\beta_2 = 0.999$ 。该实验的结果如图3–1a所示。我们可以看到 Adam 倾向于使得 θ 增大，也即，在 Adam 中， θ 的累积更新是朝着错误的方向的（因为在该问题中正确的优化方向是 θ 不断减小）。与之相对的，我们可以看到 AMSGad 和 AdaShift 中， θ 都是在不断地减小。除此之外，对于相同的学习率，在 AdaShift 中， θ 减少的速度更快（相较于 AMSGad）；这验证了我们关于 AMSGad 会因 v_t 过大而一定程度上减缓训练速度的论点。

在这个试验中，我们同时验证了定理3.1。如图 3–1b所示，当 β_1 或者 β_2 足够大时，Adam 均是可以朝着正确的方向收敛的。这里有几点值得注意：(1) 尽管此时 Adam 也能收敛，但 AdaShift 依然是收敛的最快的；(2) 如果 β_1 太小，如 $\beta_1 = 0.9$ （对应于图3–1b），Adam 并不朝着正确的方向收敛。

我们这里并没有做关于序列在线优化问题（公式(3–6)）的实验，因为该问题中我们时序独立的假设不成立。如果要使得算法在该问题下收敛，可以采用一个非常大的 β_1 或者 β_2 ，或者将 v_t 设为一个常数。

3.5.2 在 MNIST 上的逻辑回归和多层感知机

我们进一步将所提出的方法与 Adam、AMSGad 以及 SGD 等进行比较，这里我们基于 MNIST 数据集做逻辑回归以及多层感知机的训练。所采用的多层感知机包含两个隐藏层，每个隐藏层包含 256 个隐层单元。结果分别呈现在图3–2和图3–3中。我们发现在逻辑回归中，这些学习算法得到的结果在训练速度和泛化能力方面都很接近。在多层感知机的训练中，我们比较了 Adam、AMSGad 和 AdaShift。这里我们还测试了 AdaShift 有无空间操作算子时的区别，分别对应于 max-AdaShift（采用 reduce-max 空间操作），non-AdaShift（没有空间操作）。我们观察到 max-AdaShift 达到了更低的训练误差；而 non-AdaShift 在训练过程中有着轻微的抖动，



a) Adam, AMSGrad 以及 AdaShift。

b) 采用较大的 β_1 或 β_2 时的 Adam。

图 3-1 关于随机在线优化问题上的实验。

Figure 3-1 Experiments on stochastic counterexample.

与此同时，它达到了更好的泛化性能。这里 max-AdaShift 交叉的泛化性能可能来源于更好地拟合，因为拟合的更好可能就意味着过拟合；non-AdaShift 的训练抖动可能来源于它不太稳定的训练步长，而这可能也某种程度上导致它的泛化性能较好。

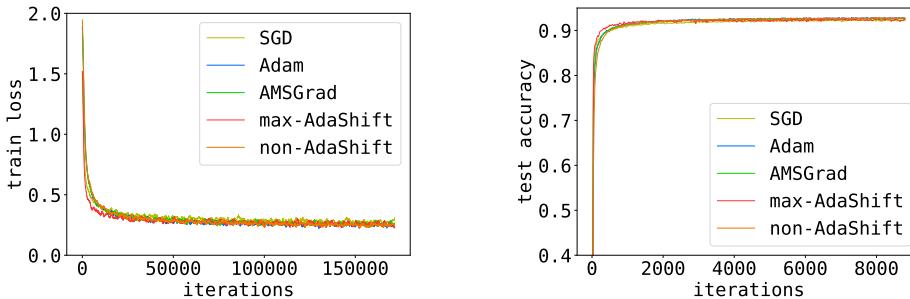


图 3-2 MNIST 上的逻辑回归。

Figure 3-2 Logistic Regression on MNIST.

3.5.3 CIFAR-10 上的 DenseNet 和 ResNet

ResNet^[87] 和 DenseNet^[88] 是两个典型的现代神经网络结构。它们通常具有很好的泛化性能也很容易训练，因此被广为使用。我们这里在 CIFAR-10 数据集上测试所提出的算法在训练 ResNet 和 DenseNet 时的性能。在实验中，我们分别采用 18 层的 ResNet 和 100 层的 DenseNet。我们将 Adam、AMSGrad 以及 AdaShift 经参数的网格搜索后得到的最优结果呈现在图3-4和图3-5中。我们可以看到 AMSGad 有着相对较差的训练速度和泛化表现。Adam 和 AdaShift 的结果相仿，尽管大体

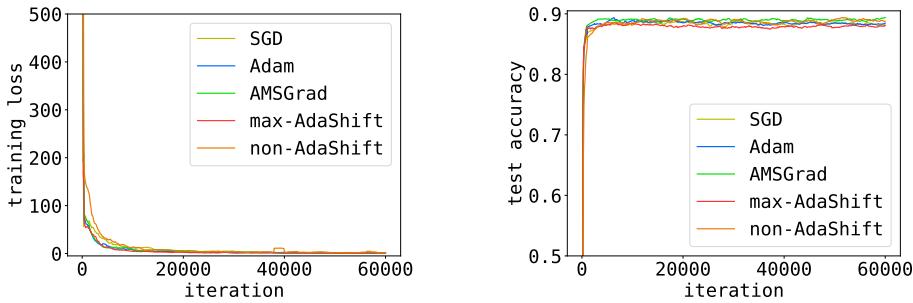


图 3-3 MNIST 上的多层感知机。

Figure 3-3 Multilayer Perceptron on MNIST.

上 AdaShift 稍微优于 Adam，尤其表现在 ResNet 的测试正确率（test accuracy）和 DenseNet 的训练误差（training loss）上。

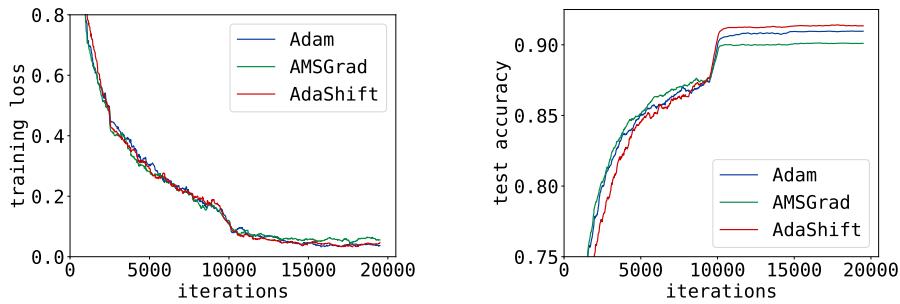


图 3-4 CIFAR-10 上的 ResNet。

Figure 3-4 ResNet on CIFAR-10.

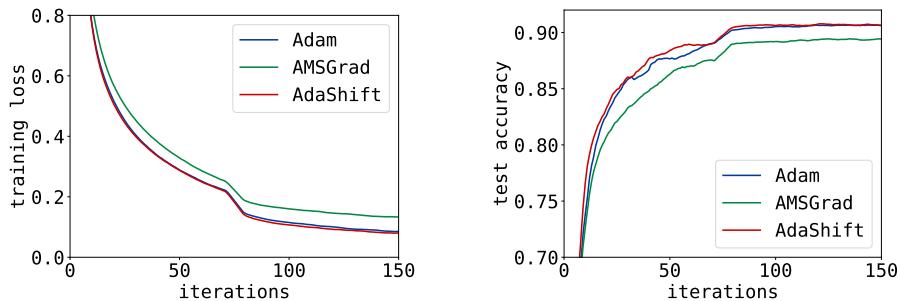


图 3-5 CIFAR-10 上的 DenseNet。

Figure 3-5 DenseNet on CIFAR-10.

3.5.4 Tiny-ImageNet 上的 DenseNet

我们进一步增加数据集的复杂程度：将数据集从 CIFAR-10 换成 Tiny-ImageNet。我们比较 Adam、AMSGrad 和 AdaShift 的算法表现，这里我们采用 DenseNet。实验结果呈现在图3–6中。从图中我们可以看出 Adam 和 AdaShift 的训练曲线基本重合，但是 AdaShift 达到了更好的测试准确率，也即表现出更好的泛化性质。AMSGrad 有着相对较高的训练误差，此外，它的测试正确率在初始阶段相对较低。

3.5.5 生成模型以及循环网络结构

我们也测试了所提出的算法在生成模型的训练以及循环网络结构上的表现。我们选择 WGAN^[35] 作为生成模型的代表，它涉及 Lipschitz 连续条件，因而可能优化的难度会较大。我们选择神经机器翻译（Neural Machine Translation, NMT^[89]）作为循环神经网络的测试任务，因为该任务中会用到典型的循环神经网络：LSTM。在图3–7a中，我们比较了 Adam、AMSGrad 和 AdaShift 在训练 WGAN-GP 判别器时的表现。这里我们固定了生成器，背后的考虑是：在生成对抗网络的训练中，生成器和判别器相互影响，固定生成器单独考虑判别器的优化时，优化表现上的差异才能清晰可见。我们发现 AdaShift 在判别器的优化任务上表现显著优于 Adam，而 AMSGad 的表现则相对不够好。神经机器翻译（NMT）任务上，如图3–7b所示，AdaShift 取得了相对于 Adam 和 AMSGad 更高的 BLEU 值（该指标越高越好）。

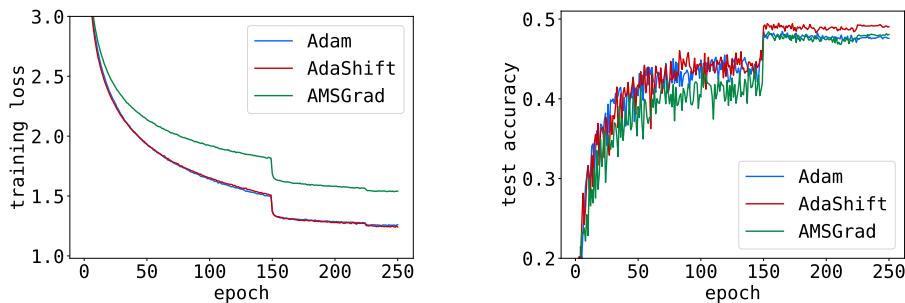


图 3–6 Tiny-ImageNet 上的 DenseNet。

Figure 3–6 DenseNet on Tiny-ImageNet.

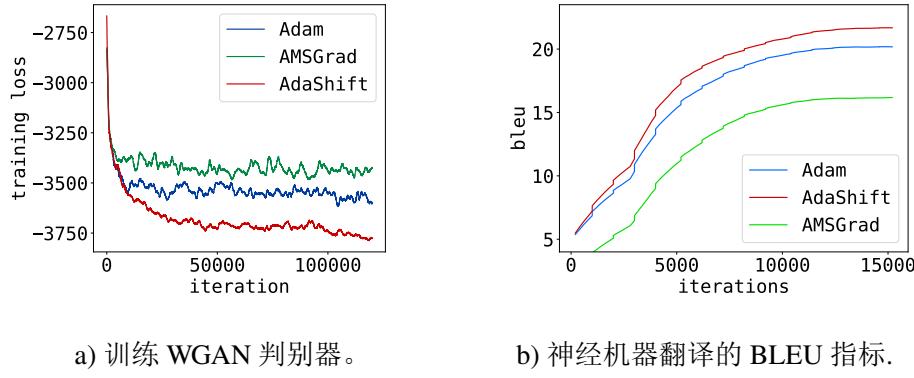


图 3-7 生成模型与循环网络结构。

Figure 3-7 Generative and Recurrent model.

3.6 本章小结

在本章中，我们研究自适应学习率算法的收敛性问题。我们从梯度的累积步长的角度出发，提出了二阶矩量 v_t 和梯度 g_t 之间存在相关性的问题，该问题导致梯度的累积步长存在一定的偏向性。我们论证了这种梯度的偏向性是导致存在不收敛现象的根本源头，此外，我们证明当 v_t 和 g_t 相互独立时，每个梯度的更新步长将服从一个相同的分布。由此，可以得出梯度的累积步长也服从同一个分布，所以在期望意义上，若 v_t 和 g_t 相互独立，那么算法的收敛性和 SGD 一致。最后，我们提出了通过时序偏移使得 v_t 和 g_t 相互独立，即，在计算 v_t 时利用的是时序偏移 n 步之后的梯度 g_{t-n} ，而不是直接使用 g_t 。

此外，按照我们对于自适应学习率算法的新的视角， v_t 可以不再是 g_t 的二阶矩量，它可以是任意的于 g_t 相互独立的变量，而为了达到自适应学习率的效果，它最好能反应整体的整体尺度（scale）。基于这样的理解，我们进一步为 v_t 的计算引入了空间操作。原本 v_t 的计算是在每个维度独立进行的，引入空间操作后，它可以利用 g_{t-n} 的所有维度的信息。这使得算法的可设计性进一步增强。实验中，我们令空间操作算法 ϕ 中包含降维操作，这样每个在 v_t 的计算上引入降维操作的模块都将得到一个共享的 v_t ，这将使得每个模块拥有各项的自适应学习率。所得出的算法在每个模块中保留了相对的梯度大小，而整体上有一个自适应的学习率，是经典 SGD 和传统自适应学习率算法的一个有机结合。

实验结果表面，所提出的算法 AdaShift 能够有效地解决 Adam 的不收敛性问题，同时，AdaShift 可以达到和 Adam 相匹敌甚至更好的训练和泛化性能。实验表面 AdaShift 在解决 Adam 的收敛性问题后，能明显改善生成对抗网络的训练。

本章中，我们所提出的 AdaShift， v_t 的计算公式采用类似 Adam 的方式更新，

这使得我们能更好地对比和分析空间操作和时序偏移的作用。事实上， v_t 的更新公式是可以任意定义的。作为一个很有意义的后续工作，我们可以进一步探索 v_t 的更新公式。

本章附录

3.A AdaShift 算法的伪代码

这里我们给出 AdaShift 的完整的伪代码，主要是对于时序偏移的处理。在正文的代码链接里有该算法的 Tensorflow 实现。

我们利用一个先进先出的队列 Q 来维度时序偏移的梯度，其最大长度为 n 。
 $Push(Q, g_t)$ 表示将元素 g_t 加入 Q 的队尾，而 $Pop(Q)$ 表示取出并返回 Q 的队首。

算法 3-2 AdaShift (完整版)

```

Require:  $n, \beta_1, \beta_2, \phi, \epsilon, \theta_0, \{f_t(\theta)\}_{t=1}^T, \{\alpha_t\}_{t=1}^T$ 
1: set  $v_0 = 0, p_0 = 1$ 
2:  $W = [\beta_1^{n-1}, \beta_1^{n-2}, \dots, \beta_1, 1] / \sum_{i=0}^{n-1} \beta_1^n$ 
3: for  $t = 1$  to  $T$  do
4:    $g_t = \nabla f_t(\theta_t)$ 
5:   if  $t \leq n$  then
6:      $Push(Q, g_t)$ 
7:   else
8:      $g_{t-n} = Pop(Q)$ 
9:      $Push(Q, g_t)$ 
10:     $m_t = W \cdot Q$ 
11:     $p_t = p_{t-1} \beta_2$ 
12:    for  $i = 1$  to  $M$  do
13:       $v_t[i] = \beta_2 v_{t-1}[i] + (1 - \beta_2) \phi(g_{t-n}^2[i])$ 
14:       $\theta_t[i] = \theta_{t-1}[i] - \alpha_t / (\sqrt{v_t[i]} / (1 - p_t) + \epsilon) \cdot m_t[i]$ 
15:    end for
16:  end if
17: end for

```

3.B 证明

3.B.1 定理3.1的证明

证明

在带偏差矫正的情况下， m_t 可表达如下：

$$m_t = \frac{(1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i} g_i}{(1 - \beta_1) \sum_{i=1}^t \beta_1^{t-i}} = \frac{\sum_{i=1}^t \beta_1^{t-i} g_i}{\sum_{i=1}^t \beta_1^{t-i}}. \quad (3-18)$$

根据洛必达法则，我们可以得到以下结论：

$$\lim_{\beta_1 \rightarrow 1} \sum_{i=1}^t \beta_1^{t-i} = \lim_{\beta_1 \rightarrow 1} \frac{1 - \beta_1^t}{1 - \beta_1} = t.$$

因此，

$$\lim_{\beta_1 \rightarrow 1} m_t = \frac{\sum_{i=1}^t g_i}{t}.$$

根据极限的定义，令 $g^* = \frac{\sum_{i=1}^t g_i}{t}$ ，我们有： $\forall \epsilon > 0, \exists \beta_1 \in (0, 1)$ ，使得，

$$\|m_t - g^*\|_\infty < \epsilon.$$

令 $\epsilon = |\frac{g^*}{2}|$ ，则对于 m_t 的每个维度，也即， $m_t[i]$ ，

$$\frac{g^*[i]}{2} \leq m_t[i] \leq \frac{3g^*[i]}{2}$$

因此， m_t 在各个维度上和 g^* 同号。

这是一个凸优化问题，假设最优解为 θ^* ，最大步长为 $\frac{\alpha_t}{\sqrt{v_t}} G$ ，满足 $\epsilon_1/G < \frac{\alpha_t}{\sqrt{v_t}} G < \epsilon_2/G$ ，我们有：

$$\lim_{t \rightarrow \infty} \|\theta_t - \theta^*\|_\infty < \epsilon_2/G. \quad (3-19)$$

给定 $\|\nabla f_t(\theta)\|_\infty \leq G$ ，我们有 $f_t(\theta) - f_t(\theta^*) < \epsilon_2$ ，从而：

$$R(T)/T = \sum_{t=1}^T [f_t(\theta_t) - f_t(\theta^*)]/T < \epsilon_2. \quad (3-20)$$

□

3.B.2 引理3.2的证明

证明 令 $\beta_1, \beta_2 \in [0, 1], d \in \mathbb{N}, 1 \leq i \leq d$ ，其中 $i \in \mathbb{N}$ 。

$$\begin{aligned}
 m_{nd+i} &= (1 - \beta_1) \sum_{j=1}^{nd+i} \beta_1^{nd+i-j} g_j \\
 &= (1 - \beta_1) \left[(C + 1) \sum_{j=0}^n \beta_1^{jd+i-1} - \sum_{j=0}^{nd+i-1} \beta_1^j \right] \\
 &= (1 - \beta_1) \left[\frac{1 - \beta_1^{(n+1)d}}{1 - \beta_1^d} \beta_1^{i-1} (C + 1) - \frac{1 - \beta_1^{nd+i}}{1 - \beta_1} \right] \\
 &= \frac{1 - \beta_1^{(n+1)d}}{1 - \beta_1^d} (1 - \beta_1) \beta_1^{i-1} (C + 1) - (1 - \beta_1^{nd+i})
 \end{aligned}$$

对于固定的 d , 随着 n 趋于无穷, 我们可以得到 m_{nd+i} 的极限:

$$\lim_{nd \rightarrow \infty} m_{nd+i} = \frac{1 - \beta_1}{1 - \beta_1^d} (C + 1) \beta_1^{i-1} - 1$$

类似地, 对于 v_{nd+i} , 我们有:

$$\begin{aligned}
 v_{nd+i} &= (1 - \beta_2) \sum_{j=1}^{nd+i} \beta_2^{nd+i-j} g_j^2 \\
 &= (1 - \beta_2) \left[(C^2 - 1) \sum_{j=0}^n \beta_2^{jd+i-1} + \sum_{j=0}^{nd+i-1} \beta_2^j \right] \\
 &= (1 - \beta_2) \left[\frac{1 - \beta_2^{(n+1)d}}{1 - \beta_2^d} \beta_2^{i-1} (C^2 - 1) + \frac{1 - \beta_2^{nd+i}}{1 - \beta_2} \right] \\
 &= \frac{1 - \beta_2^{(n+1)d}}{1 - \beta_2^d} (1 - \beta_2) \beta_2^{i-1} (C^2 - 1) - (1 - \beta_2^{nd+i})
 \end{aligned}$$

对于固定的 d , 随着 n 趋于无穷, 我们可以得到 v_{nd+i} 的极限:

$$\lim_{nd \rightarrow \infty} v_{nd+i} = \frac{1 - \beta_2}{1 - \beta_2^d} (C^2 - 1) \beta_2^{i-1} + 1 . \quad \square$$

3.B.3 引理3.3的证明

证明 首先, 我们定义 \tilde{V}_i 为:

$$\tilde{V}_i = \lim_{nd \rightarrow \infty} \frac{1}{\sqrt{v_{nd+i}}} = \frac{1}{\sqrt{\frac{1 - \beta_2}{1 - \beta_2^d} (C^2 - 1) \beta_2^{i-1} + 1}}$$

这里 $1 \leq i \leq d$, $i \in \mathbb{N}$ 。 \tilde{V}_i 的周期是 d 。令 $t' = t - nd$, 则我们有:

$$\begin{aligned}
 \lim_{nd \rightarrow \infty} k(g_{nd+i}) &= \sum_{t=nd+i}^{\infty} \frac{(1-\beta_1)\beta_1^{t-nd-i}}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{(t-1)\%d}+1}} \\
 &= \sum_{t'=i}^{\infty} \frac{(1-\beta_1)\beta_1^{t'-i}}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{(t'-1)\%d}+1}} \\
 &= \sum_{l=1}^{\infty} \sum_{j''=(l-1)d+i}^{ld+i-1} (1-\beta_1)\beta_1^{j''-i} \cdot \tilde{V}_{j''} \\
 &= \sum_{l=1}^{\infty} \beta_1^{(l-1)d} \sum_{j'=i}^{i+d-1} (1-\beta_1)\beta_1^{j'-i} \cdot \tilde{V}_{j'} \\
 &= \sum_{l=1}^{\infty} \beta_1^{(l-1)d} \sum_{j=0}^{d-1} (1-\beta_1)\beta_1^j \cdot \tilde{V}_{j+i} \\
 &= \sum_{l=1}^{\infty} \beta_1^{(l-1)d} \left[\beta_1 \sum_{j=0}^{d-1} (1-\beta_1)\beta_1^j \cdot \tilde{V}_{j+i+1} + (1-\beta_1)(1-\beta_1^d) \cdot \tilde{V}_i \right] \\
 &= \beta_1 \cdot \lim_{nd \rightarrow \infty} k(g_{nd+i+1}) + \sum_{l=1}^{\infty} \beta_1^{(l-1)d} (1-\beta_1)(1-\beta_1^d) \cdot \tilde{V}_i
 \end{aligned}$$

因此, 我们可以得到 $k(g_{nd+i})$ 的极限差异:

$$\begin{aligned}
 \lim_{nd \rightarrow \infty} k(g_{nd+i+1}) - \lim_{nd \rightarrow \infty} k(g_{nd+i}) &= \sum_{l=1}^{\infty} \beta_1^{(l-1)d} \left[(1-\beta_1)^2 \sum_{j=0}^{\infty} \beta_1^j \cdot \tilde{V}_{j+i+1} + (1-\beta_1)^2 \sum_{j=0}^{\infty} \beta_1^j \cdot \tilde{V}_i \right] \\
 &= (1-\beta_1)^2 \sum_{l=1}^{\infty} \beta_1^{(l-1)d} \sum_{j=0}^{d-1} \beta_1^j \cdot [\tilde{V}_{j+i+1} - \tilde{V}_i]
 \end{aligned}$$

因为 \tilde{V}_{nd+i} 在每个周期中单调递增, 并且权重 β_1^j 是固定的。所以加权和 $\sum_{j=0}^{d-1} \beta_1^j \cdot [\tilde{V}_{j+i+1} - \tilde{V}_i]$ 是单调递减的。也就是说, 极限差异单调递减, 使得存在 $j, 1 \leq j \leq d$, $\lim_{nd \rightarrow \infty} k(g_{nd+j})$ 取最大值。此外, 显然有 $\lim_{nd \rightarrow \infty} k(g_{nd+1})$ 是最小值。

因此, 我们有以下结论:

$\exists 1 \leq j \leq d$, 使得

$$\lim_{nd \rightarrow \infty} k(C) = \lim_{nd \rightarrow \infty} k(g_{nd+1}) < \lim_{nd \rightarrow \infty} k(g_{nd+2}) < \cdots < \lim_{nd \rightarrow \infty} k(g_{nd+j})$$

and

$$\lim_{nd \rightarrow \infty} k(g_{nd+j}) > \lim_{nd \rightarrow \infty} k(g_{nd+j+1}) > \cdots > \lim_{nd \rightarrow \infty} k(g_{nd+d+1}) = \lim_{nd \rightarrow \infty} k(C),$$

这里 $K(C)$ 表示梯度 $g_i = C$ 的净更新步长。.

□

3.B.4 引理3.4的证明

引理 3.6¹ 对于一个有界的随机变量 X , 可导函数 $f(x)$ 的期望可以表达如下:

$$\mathbb{E}[f(X)] = f(\mathbb{E}[X]) + \frac{f''(\mathbb{E}[X])}{2}D(X) + R_3 \quad (3-21)$$

其中 $D(X)$ 表示 X 的方差, 而 R_3 为:

$$R_3 = \frac{f^{[3]}(\alpha)}{3}\mathbb{E}(X - \mathbb{E}[X])^3 + \quad (3-22)$$

$$+ \int_{|x-\mathbb{E}[X]|>c} \left(f(\mathbb{E}[X]) + f'(\mathbb{E}[X])(x - \mathbb{E}[X])^2 + f''(x) \right) dF(x) \quad (3-23)$$

$F(x)$ 是 X 的分布函数。 R_3 在一定条件下是一个小量。 c 足够大, 使得: 对于任何的 $\epsilon > 0$,

$$P(X \in [\mathbb{E}[X] - c, \mathbb{E}[X] + c]) = P(|X - \mathbb{E}[X]| \leq c) \leq 1 - \epsilon \quad (3-24)$$

证明 (引理3.4 的证明) 在随机随机在线优化问题(3-7)中, 梯度满足如下分布:

$$g_i = \begin{cases} C, & \text{with probability } p := \frac{1+\delta}{C+1}; \\ -1, & \text{with probability } 1-p := \frac{C-\delta}{C+1}. \end{cases}, \quad (3-25)$$

由此, 我们可以得到 g_i 的期望如下:

$$\mathbb{E}[g_i] = \delta \quad (3-26)$$

$$\mathbb{E}[g_i^2] = C^2 \cdot \frac{1+\delta}{C+1} + (-1)^2 \cdot \frac{C-\delta}{C+1} = C + \delta(C+1) \quad (3-27)$$

$$D[g_i] = C + \delta(C+1) - \delta^2 \quad (3-28)$$

$$\mathbb{E}[g_i^4] = C(C^2 - C + 1) + \delta(C-1)(C^2 + 1) \quad (3-29)$$

$$D[g_i^2] = C^3 - 2C^2 + C + \delta(C^3 - 3C^2 - C - 1) - \delta^2(C+1)^2 \quad (3-30)$$

¹See detail in: <https://stats.stackexchange.com/questions/5782/variance-of-a-function-of-one-random-variable>

与此同时，假设梯度独立同分布，当 $nd \rightarrow \infty$ 时， v_i 的期望和方差可表达如下：

$$\mathbb{E}[v_i] = \lim_{i \rightarrow \infty} (1 - \beta_2) \sum_{j=1}^i \beta_2^{i-j} \mathbb{E}[g_j^2] = \lim_{i \rightarrow \infty} (1 - \beta_2^i) \mathbb{E}[g_j^2] = C + \delta(C + 1) \quad (3-31)$$

$$D[v_i] = \lim_{i \rightarrow \infty} (1 - \beta_2) \sum_{j=1}^i \beta_2^{i-j} D[g_j^2] = \lim_{i \rightarrow \infty} (1 - \beta_2^i) D[g_j^2] = D[g_j^2] \quad (3-32)$$

从而，梯度 g_i 的净更新步长为：

$$k(g_i) = \sum_{t=0}^{\infty} \frac{(1 - \beta_1) \beta_1^t}{\sqrt{\beta_2^{t+1} v_{i-1} + (1 - \beta_2) \beta_2^t \cdot g_i^2 + (1 - \beta_2) \sum_{j=1}^t \beta_2^{t-j} g_{i+j}^2}}$$

我们定义当 $t = 0$ 时 $\sum_{j=1}^t \beta_2^{t-j} g_{i+j}^2$ 等于 0。我们定义 X_t 为：

$$X_t = \beta_2^{t+1} v_{i-1} + (1 - \beta_2) \beta_2^t \cdot g_i^2 + (1 - \beta_2) \sum_{j=1}^t \beta_2^{t-j} g_{i+j}^2$$

$$\mathbb{E}[X_t] = \beta_2^{t+1} \mathbb{E}[v_{i-1}] + (1 - \beta_2) \beta_2^t \cdot g_i^2 + (1 - \beta_2) \sum_{j=1}^t \beta_2^{t-j} \mathbb{E}[g_{i+j}^2] \quad (3-33)$$

$$= \beta_2^{t+1} \mathbb{E}[g^2] + (1 - \beta_2) \beta_2^t \cdot g_i^2 + (1 - \beta_2^t) \mathbb{E}[g^2] \quad (3-34)$$

$$= (1 + \beta_2^{t+1} - \beta_2^t) \mathbb{E}[g^2] + (1 - \beta_2) \beta_2^t \cdot g_i^2 \quad (3-35)$$

$$D[X_t] = \beta_2^{2(t+1)} D[v_{i-1}] + \frac{(1 - \beta_2)^2 (1 - \beta_2^{2t})}{1 - \beta_2^2} D[g_{i+j}^2] \quad (3-36)$$

$$= \left[\beta_2^{2(t+1)} + \frac{(1 - \beta_2)^2 (1 - \beta_2^{2t})}{1 - \beta_2^2} \right] D[g^2] \quad (3-37)$$

对于函数 $f(x) = \frac{1}{\sqrt{x}}$ ：

$$f''(x) = \frac{3 \cdot x^{-5/2}}{8}$$

更具引理3.6， $f(X_t)$ 的期望可以表达如下：

$$\mathbb{E}[f(X_t)] = (\mathbb{E}[X_t])^{-1/2} + \frac{3}{8} (\mathbb{E}[X_t])^{-5/2} \cdot D[X_t] \quad (3-38)$$

$\mathbb{E}[X_t]$ 和 $D[X_t]$ 分别由公式(3-33) 和 公式(3-36) 表达。

从而，我们可以得到净更新步长的期望的表达如下：

$$k(g_i) = \sum_{t=0}^{\infty} (1 - \beta_1) \beta_1^t \left[\frac{1}{\sqrt{(1-\beta_2)\beta_2^t g_i^2 + (1+\beta_2^{t+1}-\beta_2^k)\mathbb{E}[g_i^2]}} + \frac{3D_t}{8[(1-\beta_2)\beta_2^t g_i^2 + (1+\beta_2^{t+1}-\beta_2^t)\mathbb{E}[g_i^2]]^{\frac{5}{2}}} \right] \quad (3-39)$$

这里 $D_t = D[X_k]$ 。

所以，梯度 C and -1 的净更新步长分别为：

$$k(C) = \sum_{t=0}^{\infty} (1 - \beta_1) \beta_1^t \left[\frac{1}{\sqrt{(1-\beta_2)\beta_2^t C^2 + (1+\beta_2^{t+1}-\beta_2^k)\mathbb{E}[g_i^2]}} + \frac{3D_t}{8[(1-\beta_2)\beta_2^t C^2 + (1+\beta_2^{t+1}-\beta_2^t)\mathbb{E}[g_i^2]]^{\frac{5}{2}}} \right] \quad (3-40)$$

$$k(-1) = \sum_{t=0}^{\infty} (1 - \beta_1) \beta_1^t \left[\frac{1}{\sqrt{(1-\beta_2)\beta_2^t + (1+\beta_2^{t+1}-\beta_2^k)\mathbb{E}[g_i^2]}} + \frac{3D_t}{8[(1-\beta_2)\beta_2^t + (1+\beta_2^{t+1}-\beta_2^t)\mathbb{E}[g_i^2]]^{\frac{5}{2}}} \right] \quad (3-41)$$

从以上表达式可以看出，在无穷级数中， $k(C)$ 的任何一项均小于 $k(-1)$ 中的对应项。因此， $k(C) < k(-1)$ 。□

3.B.5 引理3.7的证明

证明 根据引理3.2，我们有：

$$\lim_{nd \rightarrow \infty} \frac{m_{nd+i}}{\sqrt{v_{nd+i}}} = \frac{\frac{1-\beta_1}{1-\beta_1^d}(C+1)\beta_1^{i-1} - 1}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{i-1} + 1}}$$

定义极限条件下，每个周期中的所有更新的和为 $\mathcal{S}(\beta_1, \beta_2, C)$ ：

$$\mathcal{S}(\beta_1, \beta_2, C) = \sum_{i=1}^d \lim_{nd \rightarrow \infty} \frac{m_{nd+i}}{\sqrt{v_{nd+i}}}$$

假设 β_2 和 C 都足够大以使得 $v_t \gg 1$ ，我们可以得到 v_{nd+i} 的极限的近似值：

$$\lim_{nd \rightarrow \infty} v_{nd+i} \approx \frac{1 - \beta_2}{1 - \beta_2^d} (C^2 - 1) \beta_2^{i-1}$$

进而，我们可有将 $\mathcal{S}(\beta_1, \beta_2, C)$ 表达如下：

$$\begin{aligned}
\mathcal{S}(\beta_1, \beta_2, C) &= \sum_{i=1}^d \frac{\frac{1-\beta_1}{1-\beta_2^d}(C+1)\beta_1^{i-1}-1}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{i-1}}} \\
&= \sum_{i=1}^d \frac{\frac{1-\beta_1}{1-\beta_1^d}(C+1)\beta_1^{i-1}}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{i-1}}} - \sum_{i=1}^d \frac{1}{\sqrt{\frac{1-\beta_2}{1-\beta_2^d}(C^2-1)\beta_2^{i-1}}} \\
&= \sqrt{\frac{1-\beta_2^d}{(1-\beta_2)\beta_2^{d-1}}} \sqrt{\frac{C+1}{C-1}} \cdot \frac{1-\beta_1}{1-\beta_1^d} \cdot \frac{\sqrt{\beta_2^d}-\beta_1^d}{\sqrt{\beta_2}-\beta_1} - \sqrt{\frac{1-\beta_2^d}{(1-\beta_2)\beta_2^{d-1}}} \cdot \frac{1}{\sqrt{C^2-1}} \frac{\sqrt{\beta_2^d}-1}{\sqrt{\beta_2}-1} \\
&= \sqrt{\frac{1-\beta_2^d}{(1-\beta_2)\beta_2^{d-1}(C-1)}} \left[\frac{(1-\beta_1)(\beta_2^d-\beta_1^d)\sqrt{C+1}}{(1-\beta_1^d)(\sqrt{\beta_2}-\beta_1)} - \frac{\sqrt{\beta_2^d}-1}{\sqrt{C+1}(\sqrt{\beta_2}-1)} \right]
\end{aligned}$$

令 $\mathcal{S}(\beta_1, \beta_2, C) = 0$ ，我们得到以下临界条件：

$$C + 1 = \frac{(1-\beta_1^d)(\sqrt{\beta_2^d}-\beta_1^d)(1-\sqrt{\beta_2})}{(1-\beta_1)(\sqrt{\beta_2}-\beta_1)(1-\sqrt{\beta_2^d})}$$

□

3.C β_1 、 β_2 以及 C 之间的临界关系

为了直观地了解 C, d, β_1, β_2 等变量和 Adam 的收敛性之间的关系，我们令 $C = d = 6$ ，初始化 $\theta_1 = 0$ ，在区间 $[0, 1)$ 采样 β_1 和 β_2 ，观察 Adam 优化 2000 步之后的结果。实验结果如图3-8a所示。首先可以观察到的是，对于一个固定的序列在线优化问题， β_1 和 β_2 均影响和决定 Adam 的优化方向和速度。

进一步地，我们研究 C 和 d 使得 Adam 朝反方向收敛的临界值。实验中，我们令 $d = C$ ，这样可以使得每个 epoch 的总体梯度大小为 +1，我们在区间 $[0, 1)$ 采样 β_1 和 β_2 ，结果如图3-8b所示。实验表明，随着 β_1 和 β_2 的增大，序列在线优化问题需要更大的 C 来使得 Adam 朝着反方向优化。换句话说，大的 β_1 和 β_2 可以缓解 Adam 不收敛性问题。

我们同样对随机在线优化问题做了同样的实验，以分析 C, β_1, β_2 以及 Adam 收敛行为之间的关系。结果如图3-8c和图3-8d所示。所观察到的结果和之前的基本一致：大的 C 会更容易造成不收敛问题，而大的 β_1 或 β_2 可以一定程度上帮助解决不收敛性问题。在这个实验中，我们令 $\delta = 1$ 。

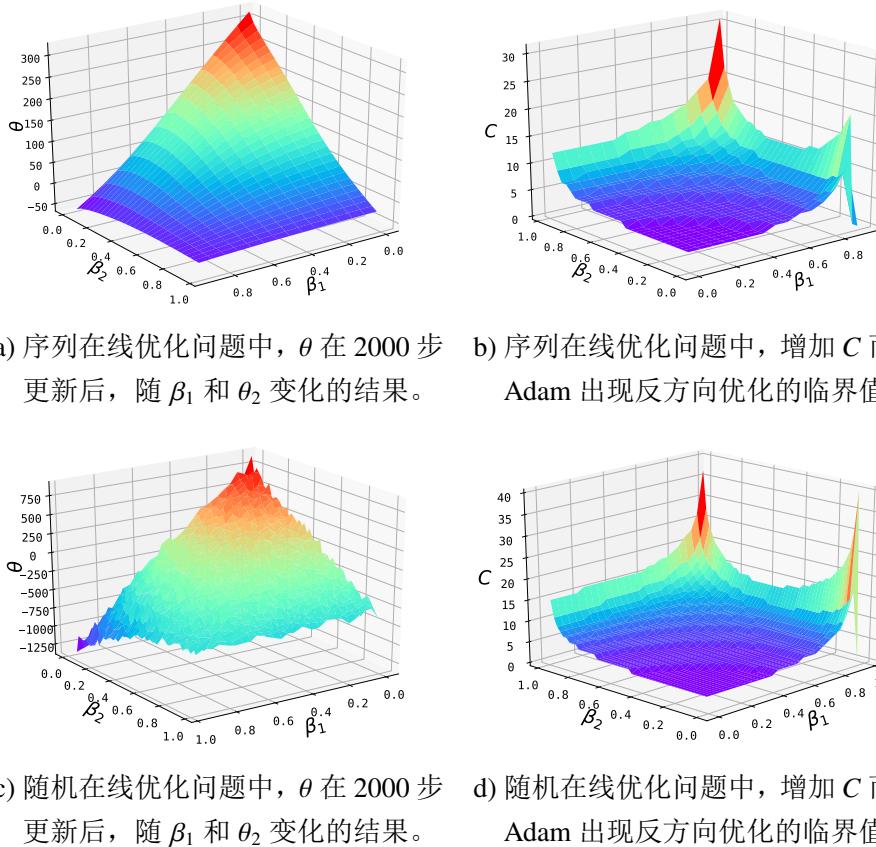


图 3-8 在 Adam 中, β_1 和 β_2 均影响收敛的方向和速度。 C_t 使得 Adam 反方向的临界值随着 β_1 和 β_2 的增大而增加。

Figure 3-8 Both β_1 and β_2 influence the direction and speed of optimization in Adam. Critical value of C_t , at which Adam gets into non-convergence, increases as β_1 and β_2 getting large.

引理 3.7 (临界条件) 在序列在线优化问题(3-6)中, 令 α_t 为固定量, 定义 $\mathcal{S}(\beta_1, \beta_2, C, d)$ 为每个周期 (epoch) 中更新步长的极限之和:

$$\mathcal{S}(\beta_1, \beta_2, C) \triangleq \sum_{i=1}^d \lim_{nd \rightarrow \infty} \frac{m_{nd+i}}{\sqrt{v_{nd+i}}} . \quad (3-42)$$

令 $\mathcal{S}(\beta_1, \beta_2, C) = 0$, 假设 β_2 和 C 足够大, 以使得 $v_t \gg 1$, 我们有:

$$C + 1 = \frac{(1 - \beta_1^d)(\sqrt{\beta_2^d} - \beta_1^d)(1 - \sqrt{\beta_2})}{(1 - \beta_1)(\sqrt{\beta_2} - \beta_1)(1 - \sqrt{\beta_2^d})} . \quad (3-43)$$

公式(3-43), 尽管有一点辅助, 依然很好地说明了 β_1 和 β_2 均和问题的收敛性质密切相关, 并且这些参数之间存在一个影响收敛方向正确性的临界条件。

3.D 相关性测试

为了验证正文中给出的一些相关性假设，我们做了一组实验，并计算各元素之间的相关系数。实验中，我们用 MNIST 数据集训练一个多层感知机直到收敛，并收集在训练过程中隐藏层单元在每一步的梯度。基于这些数据，我们计算 $g_t[i]$ 和 $g_{t-n}[i]$, $g_t[i]$ 和 $g_{t-n}[j]$, 以及 $g_t[i]$ 和 $v_t[i]$ 之间的相关系数。我们用最后十个训练周期的数据，计算 Pearson 相关系数。其计算公式如下：

$$\rho = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}.$$

为了测试 $g_t[i]$ 和 $g_{t-n}[i]$ 之间的相关性，我们令 n 从 1 变到 10，然后计算所有不同维度的平均时序相关系数。结果列于表3-1中。

表 3-1 $g_t[i]$ 和 $g_{t-n}[i]$ 之间的时序相关系数。

Table 3-1 Temporal correlation coefficient between $g_t[i]$ and $g_{t-n}[i]$.

| n | 1 | 2 | 3 | 4 | 5 |
|--------|--------------|--------------|--------------|--------------|--------------|
| ρ | -0.000368929 | -0.000989286 | -0.001540511 | -0.00116966 | -0.001613395 |
| n | 6 | 7 | 8 | 9 | 10 |
| ρ | -0.001211721 | 0.000357474 | -0.00082293 | -0.001755237 | -0.001267641 |

为了验证 $g_t[i]$ 和 $g_{t-n}[j]$ 的空间相关性，我们同样令 n 从 1 变到 10，然后随机采样 i 和 j ，计算所有采样的 (i, j) 对的平均的空间相关系数。结果列于表3-2中。

表 3-2 $g_t[i]$ 和 $g_{t-n}[j]$ 的空间相关系数。

Table 3-2 Spatial correlation coefficient between $g_t[i]$ and $g_{t-n}[j]$.

| n | 1 | 2 | 3 | 4 | 5 |
|--------|--------------|--------------|--------------|--------------|--------------|
| ρ | -0.000609471 | -0.001948853 | -0.001426661 | 0.000904615 | 0.000329359 |
| n | 6 | 7 | 8 | 9 | 10 |
| ρ | 0.000971337 | -0.000644563 | -0.00137805 | -0.001147973 | -0.000592037 |

为了测试在 Adam 中 $g_t[i]$ 和 $v_t[i]$ 之间的相关性，我们计算 v_t 并计算 g_t^2 和 v_t 之间的相关系数，在所有维度 i 之间去平均。其结果为 **0.435885276**。

我们进一步测试了：在不包含空间操作的 AdaShift 中， $g_{t-n}[i]$ 和 $v_t[i]$ 的相关性，以及在包含空间操作的 AdaShift 中（实验中采用的是 reduce-max 操作）， $g_{t-n}[i]$ 和 v_t 的相关性。我们令 n 从 1 变到 10，并计算各个维度的平均相关系数。结果分别列于表格3-3和3-4中。

表 3-3 在不包含空间操作的 AdaShift 中, $g_{t-n}^2[i]$ 和 $v_t[i]$ 的相关系数。Table 3-3 Correlation coefficient between $g_{t-n}^2[i]$ and $v_t[i]$ in non-AdaShift.

| n | 1 | 2 | 3 | 4 | 5 |
|--------|--------------|--------------|--------------|--------------|--------------|
| ρ | -0.010897023 | -0.010952548 | -0.010890854 | -0.010853069 | -0.010810747 |
| n | 6 | 7 | 8 | 9 | 10 |
| ρ | -0.010777789 | -0.01075946 | -0.010739279 | -0.010728553 | -0.010720019 |

表 3-4 在包含空间操作的 AdaShift 中, $g_{t-n}^2[i]$ 和 v_t 的相关系数。Table 3-4 Correlation coefficient between $g_{t-n}^2[i]$ and v_t in max-AdaShift.

| n | 1 | 2 | 3 | 4 | 5 |
|--------|--------------|--------------|--------------|--------------|--------------|
| ρ | -0.000706289 | -0.000794959 | -0.00076306 | -0.000712474 | -0.000668459 |
| n | 6 | 7 | 8 | 9 | 10 |
| ρ | -0.000623162 | -0.000566573 | -0.000542046 | -0.000598015 | -0.000592707 |

3.E 仅时序偏移和仅空间操作

在我们提出的算法中, 我们将空间操作作用在时序偏移后的梯度上 g_{t-n} , 即: $v_t[i] = \beta_2 v_{t-1}[i] + (1 - \beta_2)\phi(g_{t-n}^2[i])$ 。它基于梯度时序独立的假设, 也即, g_{t-n} 独立于 g_t , 对于任何的 $n! = 0$ 。依照我们在小节3.4.2中的论点, 我们可以进一步假设 g_{t-n} 中的每个元素均独立于 g_t 的第 i 个维度。

值得注意的是, 我们刻意避免了采用当前梯度 g_t 的其他维度的元素, 因为他们可能不满足独立性假设。比如, 当一个样本很罕见时, 它倾向于具有较大的损失, 进而带来较大的梯度, g_t 的整体梯度大小可能受此影响。然而, 对于时序已经做过偏移的梯度 g_{t-n} 而言, 进一步利用它的其他空间元素并不会带来相互独立性不成立的问题。

为了进一步研究时序偏移和空间操作的影响, 我们提出两个 AdaShift 的变种: (i) 仅时序偏移的 AdaShift, 它不引入空间操作算子, v_t 的更新公式为: $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_{t-n}^2$; (ii) 仅空间操作的 AdaShift, 它在 Adam 的基础上引入了空间操作, 但直接作用于不做时序偏移的梯度。

根据我们的实验, 仅时序偏移的 AdaShift 相对于 AdaShift 而言, 不太稳定。在某些任务中, 仅时序偏移的 AdaShift 可以正常工作, 但有些任务中, 仅时序偏移的 AdaShift 会出现梯度爆炸问题, 进而需要较小的学习率。而仅空间操作的 AdaShift 的表现和 Adam 接近。下一节中有更多的关于仅空间操作的 AdaShift 的实验。

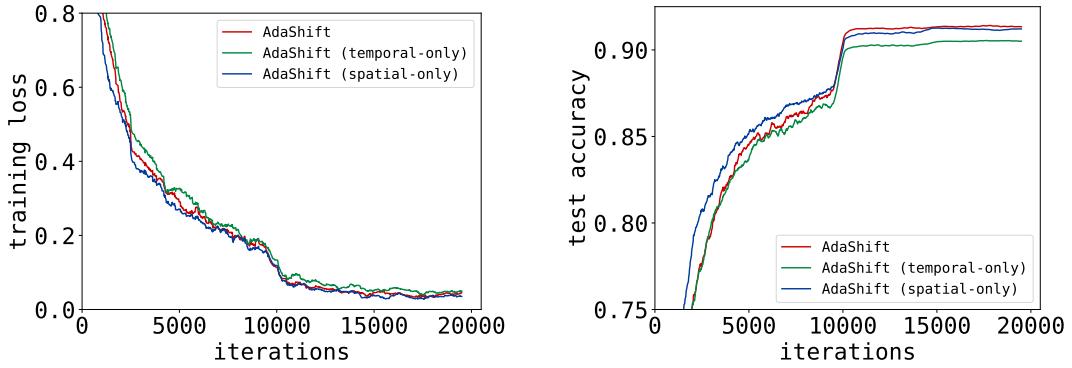


图 3-9 仅时序偏移和仅空间操作：CIFAR-10 上的 ResNet。

Figure 3-9 Temporal-only and Spatial-only: ResNet on CIFAR-10.

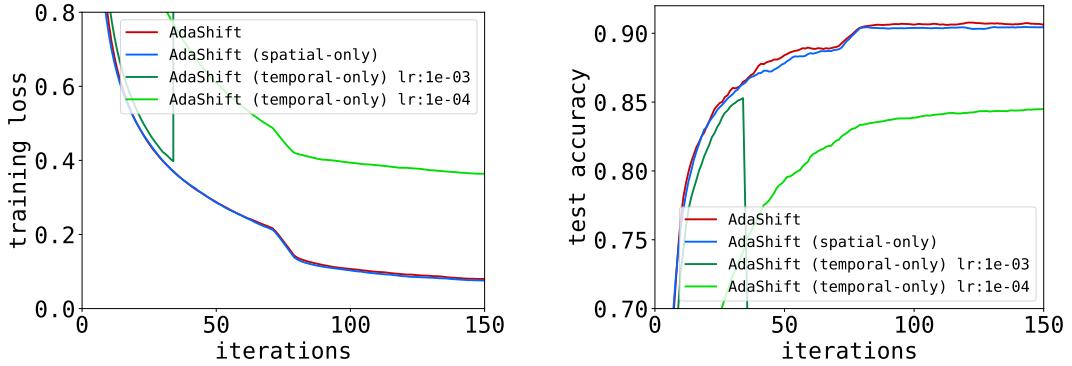


图 3-10 仅时序偏移和仅空间操作：CIFAR-10 上的 DenseNet。

Figure 3-10 Temporal-only and Spatial-only: DenseNet on CIFAR-10.

3.F 算法中的超参数

3.F.1 实验中的超参数设定

这里，我们列举以上所用实验中的算法参数。

3.F.2 学习率 α_t 的敏感性

这里我们通过实验探索 AdaShift 中学习率 α_t 的敏感性。实验中，我们令 $\alpha_t \in \{0.1, 0.01, 0.001\}$ ，令 $n = 10$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ 。结果呈现于图3-11 和图 3-12 中。

根据实验结果，当使用 \max 作为空间操作算子的时候，AdaShift 的最佳学习率大约是 Adam 的十倍。

表 3-5 逻辑回归的超参数设定。

Table 3-5 Hyper-parameter setting of Logistic Regression.

| Optimizer | learning rate | β_1 | β_2 | n |
|--------------|---------------|-----------|-----------|-----|
| SGD | 0.1 | N/A | N/A | N/A |
| Adam | 0.001 | 0 | 0.999 | N/A |
| AMSGrad | 0.001 | 0 | 0.999 | N/A |
| non-AdaShift | 0.001 | 0 | 0.999 | 1 |
| max-AdaShift | 0.01 | 0 | 0.999 | 1 |

表 3-6 MNIST 上的多层感知机的超参数设定。

Table 3-6 Hyper-parameter setting of MLP on MNIST.

| Optimizer | learning rate | β_1 | β_2 | n |
|--------------|---------------|-----------|-----------|-----|
| SGD | 0.001 | N/A | N/A | N/A |
| Adam | 0.001 | 0 | 0.999 | N/A |
| AMSGrad | 0.001 | 0 | 0.999 | N/A |
| non-AdaShift | 0.0005 | 0 | 0.999 | 1 |
| max-AdaShift | 0.01 | 0 | 0.999 | 1 |

表 3-7 WGAN-GP 的超参数设定。

Table 3-7 Hyper-parameter setting of WGAN-GP.

| Optimizer | learning rate | β_1 | β_2 | n |
|-----------|---------------|-----------|-----------|-----|
| Adam | 1e-5 | 0 | 0.999 | N/A |
| AMSGrad | 1e-5 | 0 | 0.999 | N/A |
| AdaShift | 1.5e-4 | 0 | 0.999 | 1 |

表 3-8 神经机器翻译的超参数设定。

Table 3-8 Hyper-parameter setting of Neural Machine Translation.

| Optimizer | learning rate | β_1 | β_2 | n |
|-----------|---------------|-----------|-----------|-----|
| Adam | 0.0001 | 0.9 | 0.999 | N/A |
| AMSGrad | 0.0001 | 0.9 | 0.999 | N/A |
| AdaShift | 0.01 | 0.9 | 0.999 | 30 |

3.F.3 β_1 和 β_2 的敏感性

这里，我们通过实验探索 AdaShift 中 β_1 和 β_2 的敏感性。实验中，我们令 $\alpha = 0.01$, $n = 10$, $\beta_1 \in \{0, 0.9\}$, $\beta_2 \in \{0.9, 0.99, 0.999\}$ 。结果呈现于图 3-13 和图

表 3–9 CIFAR-10 上的 ResNet 和 DenseNet 以及 Tiny-ImageNet 上的 DenseNet 的超参数设定。

Table 3–9 Hyper-parameter setting of ResNet, DenseNet on CIFAR-10 and Tiny-ImageNet.

| Optimizer | learning rate | β_1 | β_2 | n |
|-----------|---------------|-----------|-----------|-----|
| Adam | 0.001 | 0.9 | 0.999 | N/A |
| AMSGrad | 0.001 | 0.9 | 0.999 | N/A |
| AdaShift | 0.01 | 0.9 | 0.999 | 10 |

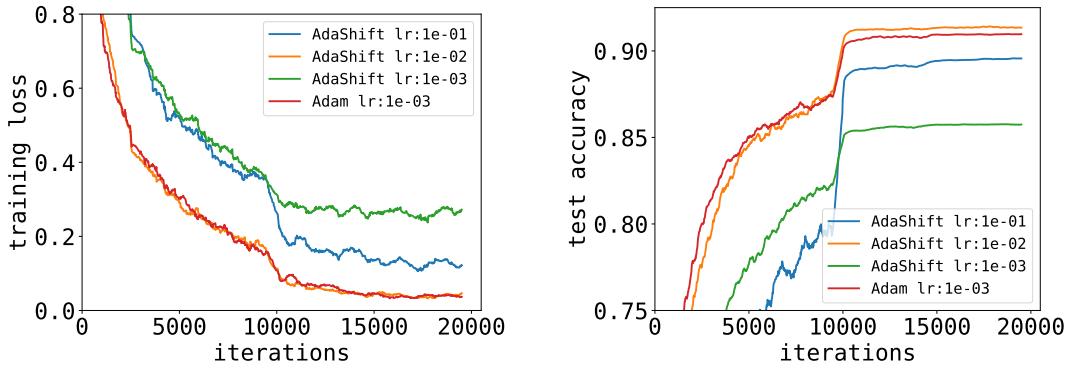


图 3–11 学习率敏感性实验：CIFAR-10 上的 ResNet。

Figure 3–11 Learning rate sensitivity experiment with ResNet on CIFAR-10.

3–14 中。

根据实验结果，AdaShift 具有相对较小的 β_1 和 β_2 敏感度。在某些任务中，采用一阶矩量估计（也即，当 $\beta_1 = 0.9$ 时）或者采用交大 β_2 （如 0.999）可以得到更好的性能。

根据我们的实验和经验，在第一次采用该算法时，建议默认参数为 $n = 10, \beta_1 = 0.9, \beta_2 = 0.999$ 。它能在多数任务中取得不错的结果。

3.F.4 n 和 m 的敏感性

现在我们实验研究 AdaShift 中的 n 敏感性。这里，我们提供一个 AdaShift 的拓展版本，也即在进行一阶矩量近似的时候，可以考虑仅仅使用最近的 m 个梯度 ($m \leq n$)：

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) \phi(g_{t-n}^2) \quad \text{and} \quad m_t = \frac{\sum_{i=0}^{m-1} \beta_1^i g_{t-i}}{\sum_{i=0}^{m-1} \beta_1^i}. \quad (3-44)$$

我们令 $\beta_1 = 0.9, \beta_2 = 0.999$ 。实验结果呈现在图 3–15，图 3–16，和图 3–17。在这些实验中我们可以看出，当 n 和 m 变化时，AdaShift 表现得相对稳定，这说

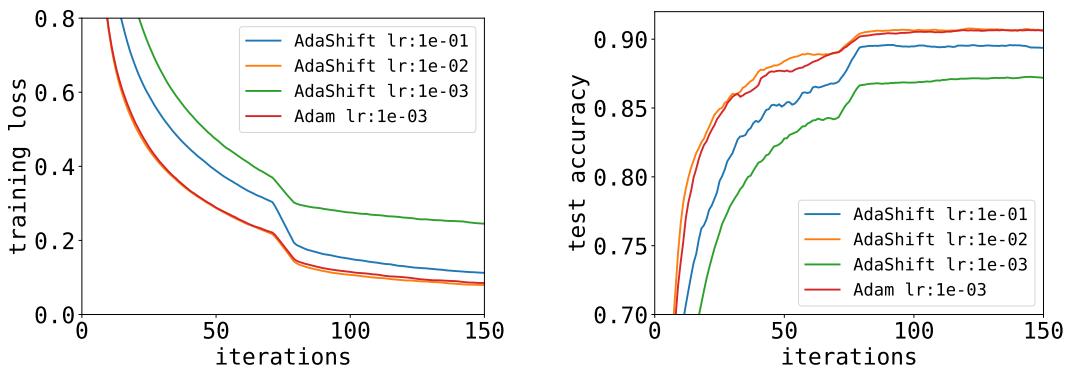
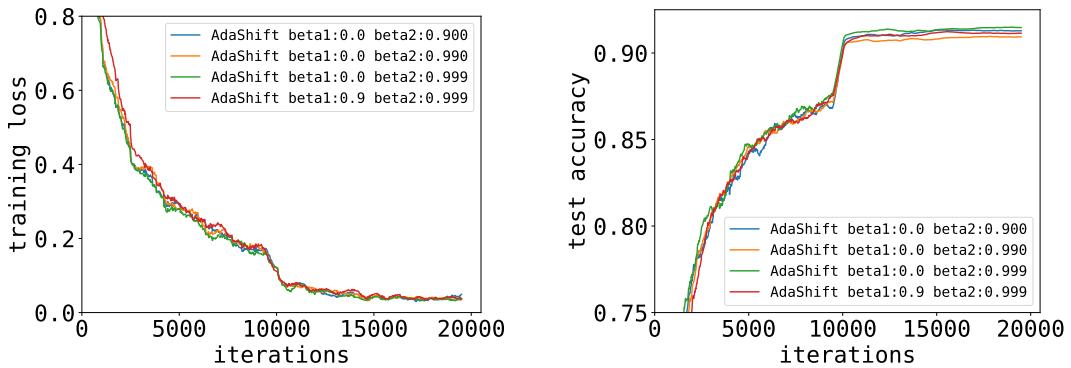


图 3-12 学习率敏感性实验：CIFAR-10 上的 DenseNet。

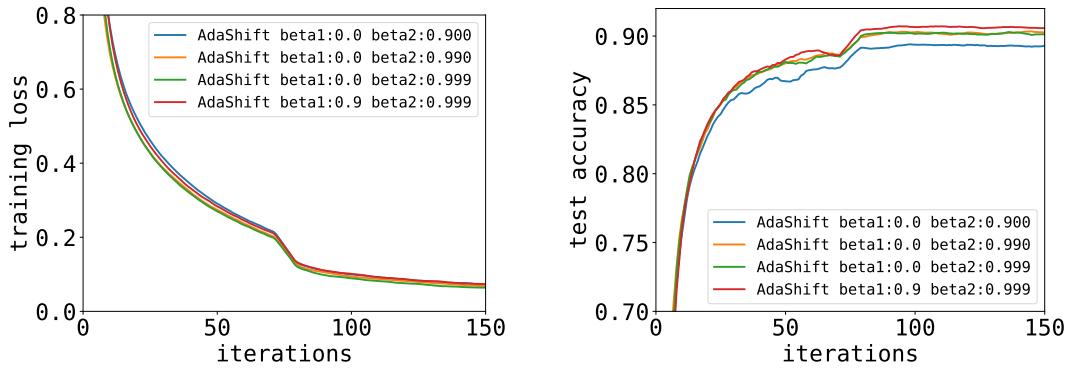
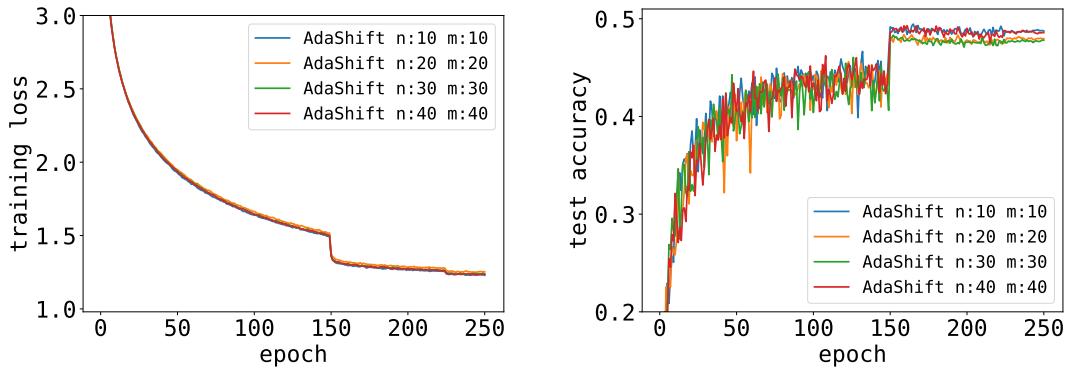
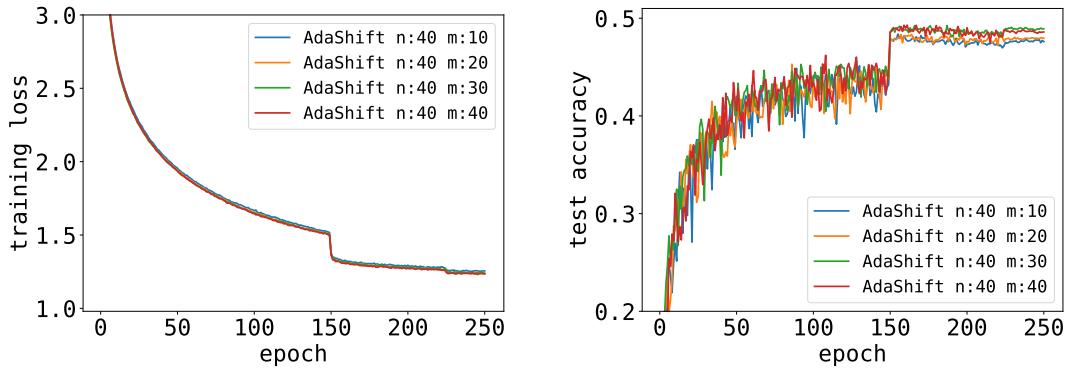
Figure 3-12 Learning rate sensitivity experiment with DenseNet on CIFAR-10.

图 3-13 β_1 和 β_2 敏感性实验：CIFAR-10 上的 ResNet。Figure 3-13 β_1 and β_2 sensitivity experiment with ResNet on CIFAR-10.

明了 AdaShift 对于这两个参数的鲁棒性。

3.G 拓展对比实验

本节中，我们拓展正文中的实验，并将 Nadam 和仅空间操作的 AdaShift 加入对比。实验结果见图3-18，图3-19 和图3-20。从这些实验结果看，Nadam 和仅空间操作的 AdaShift 和 Adam 的性能表现类似。

图 3-14 β_1 和 β_2 敏感性实验：CIFAR-10 上的 DenseNet。Figure 3-14 β_1 and β_2 sensitivity experiment with DenseNet on CIFAR-10.图 3-15 n 的敏感性实验：Tiny-ImageNet 上的 DenseNet。Figure 3-15 n sensitivity experiment with DenseNet on Tiny-ImageNet.图 3-16 m 的敏感性实验：Tiny-ImageNet 上的 DenseNet。Figure 3-16 m sensitivity experiment with DenseNet on Tiny-ImageNet.

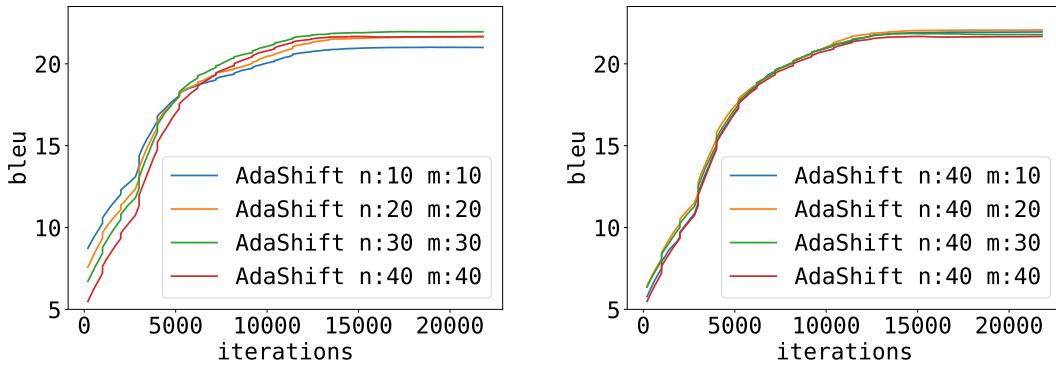


图 3-17 n 和 m 的敏感性实验：神经机器翻译。

Figure 3-17 n and m sensitivity experiment with Neural Machine Translation.

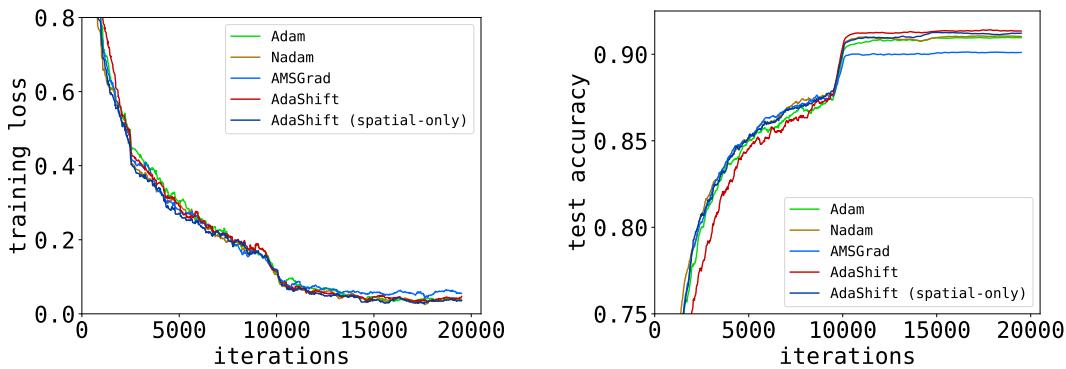


图 3-18 拓展对比试验：CIFAR-10 上的 ResNet。

Figure 3-18 Extended Experiments: ResNet on CIFAR-10.

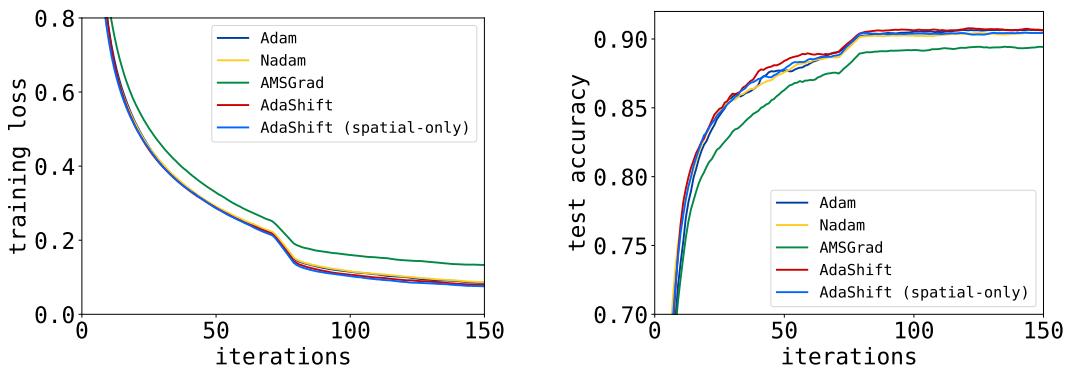


图 3-19 拓展对比试验：CIFAR-10 上的 DenseNet。

Figure 3-19 Extended Experiments: DenseNet on CIFAR-10.

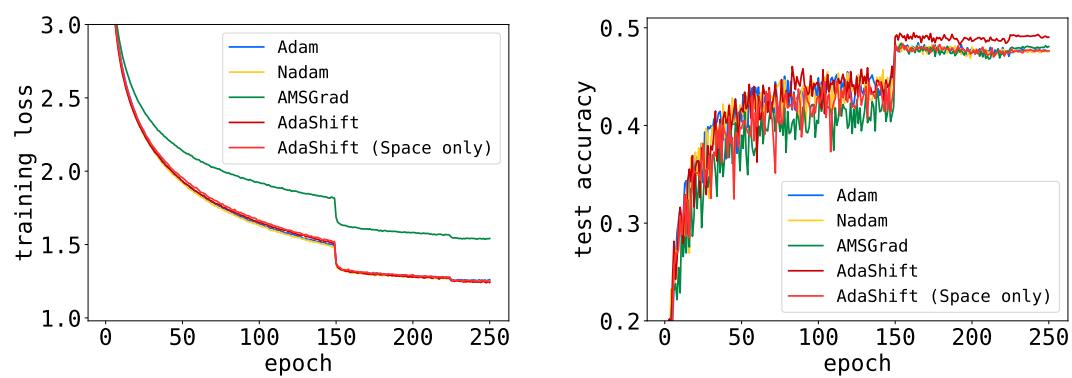


图 3–20 拓展对比试验：Tiny-ImageNet 上的 DenseNet。

Figure 3–20 Extended Experiments：DenseNet on Tiny-ImageNet.

第四章 生成对抗网络的样本质量

生成对抗网络^[1] (GANs)，作为一种新兴的生成模型，最近在各种各样的任务上都带来了很有竞争力的结果，比如自然图片生成^[5, 35, 90]，在条件控制下的图片生成^[7, 14, 19]，图片编辑^[10] 以及文本生成^[4]。尽管如此，目前的生成对抗网络模型还是难以生成非常令人信服的样本，尤其当训练数据集非常复杂的时候。与此同时，人们在实验中发现有效地利用类别标签信息能够显著地提高生成样本的质量。本章中，我们就标签信息如何影响生成对抗网络的训练以及如何更好地利用标签信息展开研究。

4.1 引言

当前存在三种典型的运用类别标签信息的生成对抗网络模型：CatGAN^[12] 将判别器做成一个多类分类器；LabelGAN^[15] 将判别器在普通多类分类器的基础上额外引入一个平级的类别用以表示来自生成器的样本；AC-GAN^[63] 在真假二类分类的判别器的基础上，引入一个独立的分类器用以分类真实数据。通过将类别信息引入到训练过程中，这些模型在实验中得到了更高质量的生成样本。但是，这些现象背后的机制却没有得到很好的分析^[17]。

在本章中，我们从数学的角度分析这些利用标签信息的生成对抗网络模型。通过分析标签信息如何在 LabelGAN^[15] 的梯度反传中起作用，我们得到了一些重要的观察：1) LabelGAN 在优化过程中倾向于将每个样本朝着某个特定的类优化；2) 而与此同时，它存在梯度耦合的问题。此外，我们发现 AC-GAN^[63] 可以被看作是层次化的多类分类器，但它缺少在普通类别层上的对抗训练。

基于这些分析和对已有模型潜在缺陷的认识，我们相应地提出了新的解决方案。具体来说，我们主张任何时候我们都应该将样本朝着某个具体的类别优化，为了做到这点我们建议显示地为每个样本分配一个标签，因为具有显式目标类的模型将为生成器提供更清晰的梯度指导。

我们认为将代表真实数据的类别替换成 K 个具体的真实数据类别通常比简单地训练辅助分类器更好，因为辅助分类器中缺少对抗训练会使模型更有可能模式崩溃并产生低质量的样本。我们还通过实验发现，预定义标签往往容易出现类内模式崩溃，并相应地提出动态标记作为解决方案。

我们从激活最大化的角度为这种类型的对抗训练提供了一种新的解释，并将所提出的模型被命名为激活最大化生成对抗网络 (AM-GAN)。我们通过大量的

对照实验实证 AM-GAN 的有效性，同时也验证我们的分析。值得一提的是，实验中，AM-GAN 取得了当前各种模型中最好的样本质量。

本章的其余部分组织如下。在小节4.2中，我们定义我们的主要符号，并介绍我们将会进一步研究的基线方法 LabelGAN^[15] 和 AC-GAN^[63]。在小节4.3中，我们分析标签信息在 LabelGAN 的梯度反传过程中的作用，以揭示类标签如何帮助其训练以及该模型中存在的问题。在小节4.4中，我们指出 LabelGAN 中的梯度耦合问题，并提出 AM-GAN 作为一种新的解决方案，我们还分析了 AM-GAN 的属性并建立了与相关工作的连接。在小节4.4.1和4.4.2中，我们通过交叉熵分解分析这几个模型之间的联系。在小节4.5.1，我们介绍了作为预定义标签的替代方案的动态标签。在小节4.5.2中，我们从激活最大化的角度解释对抗训练如何工作。在小节4.5.3中，我们为 AC-GAN 补充附属分类器中缺失的对抗训练。在小节4.7中，我们通过实验分析所提出的模型 AM-GAN 的具体表现，同时验证我们的分析。最后，我们在小节4.8总结本章内容并讨论可能的后续工作。

4.2 背景与相关工作

在原始 GAN^[1] 中，生成器 G 和判别器 D 的损失函数可以被写作：

$$L_G^{\text{ori}} = -\mathbb{E}_{z \sim p_z(z)}[\log D_r(G(z))] \triangleq -\mathbb{E}_{x \sim G}[\log D_r(x)], \quad (4-1)$$

$$L_D^{\text{ori}} = -\mathbb{E}_{x \sim p_{\text{data}}}[\log D_r(x)] - \mathbb{E}_{x \sim G}[\log(1 - D_r(x))], \quad (4-2)$$

其中， D 相当于是一个二类分类器，用以区分真实数据和生成数据； $D_r(x)$ 表示样本 x 来自于真实分布的概率。我们这里为生成器采用了 $-\log D_r(x)$ 的处理技巧。

4.2.1 标签拓展的生成对抗网络

在 [15] 中，原始 GAN 的框架被拓展到了多类的情况。经拓展后，每个样本 x 都有一个相应的类别标签 $y \in \{1, \dots, K, K+1\}$ ，其中第 $K+1$ 被定义为是生成样本的类别标签，其余的 K 个依次是所对应的数据集的 K 个真实类别。它的目标函数可以写作：

$$L_G^{\text{lab}} = -\mathbb{E}_{x \sim G}[\log \sum_{i=1}^K D_i(x)] \triangleq -\mathbb{E}_{x \sim G}[\log D_r(x)], \quad (4-3)$$

$$L_D^{\text{lab}} = -\mathbb{E}_{(x,y) \sim p_{\text{data}}}[\log D_y(x)] - \mathbb{E}_{x \sim G}[\log D_{K+1}(x)], \quad (4-4)$$

其中 $D_i(x)$ 表示样本 x 属于第 i 类的概率。为了方便后面的分析，我们把这个目标函数改成写交叉熵的形式：

$$L_G^{\text{lab}} = \mathbb{E}_{x \sim G}[H([1, 0], [D_r(x), D_{K+1}(x)])], \quad (4-5)$$

$$L_D^{\text{lab}} = \mathbb{E}_{(x,y) \sim p_{\text{data}}}[H(v(y), D(x))] + \mathbb{E}_{x \sim G}[H(v(K+1), D(x))], \quad (4-6)$$

这里 $D(x) = [D_1(x), D_2(x), \dots, D_{K+1}(x)]$ 。而 $v(y) = [v_1(y), \dots, v_{K+1}(y)]$ ，其中， $v_i(y) = 0$ 如果 $i \neq y$ ， $v_i(y) = 1$ 如果 $i = y$ 。 H 是交叉熵（cross-entropy）。它的定义是 $H(p, q) = -\sum_i p_i \log q_i$ 。我们称这个利用标签信息的生成对抗网络模型为 Label-GAN。

4.2.2 带有附属分类器的生成对抗网络

除了上述将二类分类器拓展成 $K + 1$ 分类器的做法外，[63] 提出在原始 GAN 的基础上额外引入一个分类器 C 。用 C 来分类真实数据。在不改变其核心的前提下¹，我们把它的目标函数改写如下，模型记为 AC-GAN：

$$L_G^{\text{ac}}(x, y) = \mathbb{E}_{(x,y) \sim G}[H([1, 0], [D_r(x), D_f(x)])] \quad (4-7)$$

$$+ \mathbb{E}_{(x,y) \sim G}[H(u(y), C(x))], \quad (4-8)$$

$$L_D^{\text{ac}}(x, y) = \mathbb{E}_{(x,y) \sim p_{\text{data}}}[H([1, 0], [D_r(x), D_f(x)])] \quad (4-9)$$

$$+ \mathbb{E}_{(x,y) \sim G}[H([0, 1], [D_r(x), D_f(x)])] \quad (4-10)$$

$$+ \mathbb{E}_{(x,y) \sim p_{\text{data}}}[H(u(y), C(x))], \quad (4-11)$$

这里 $D_r(x)$ 和 $D_f(x) = 1 - D_r(x)$ 表示二类判别器所输出的属于真实分布的概率和属于生成分布的概率，这部分和原始 GAN 没有区别。 $u(\cdot)$ 表示一个和 $v(\cdot)$ 类似的向量化操作，不同的是，它把类别信息向量化成一个长度为 K 的向量，而不是 $K + 1$ （因为这里用到的是一个普通的 K 类分类器）。 $C(x)$ 是样本 x 有分类器 C 判断得出的在 K 个类上的概率分布。

在 AC-GAN 当中，每个生成样本都有一个相对应的目标类别标签 y ，并且在附属分类器 C 上会相应地有一个关于 y 的损失函数以使得生成器能够利用到类别信息。我们把和附属分类器有关的损失函数，也即公式 (4-8) 和 (4-11)，称为附属分类器损失函数。

以上所定义的 AC-GAN 和 [63] 中的略有不同，主要区别包括：我们删掉了分类器 C 关于生成样本的一个损失函数 $\mathbb{E}_{(x,y) \sim G}[H(u(y), C(x))]$ ，它鼓励分类器 C 将

¹AC-GAN^[63] 的原始目标函数比较诡异、很难解释，后续也有很多人提出质疑并验证了一些更合理的修改会让它效果变得更好。我们后面也会给出进一步的解释，详见节 4.5.3。

生成样本 x 分为目标类别第 y 类，这并不合理。我们将在小节 4.5.3 更进一步地讨论这个问题。值得注意的，我们依旧为生成器采用了 $-\log(D_r(x))$ 的处理技巧。

4.3 类别标签在梯度反传过程中的影响

在本节中，我们分析类别标签在梯度反传过程中的影响。我们以 LabelGAN 为例，分析生成器在更新过程中如何利用标签信息。通过分析的梯度反传，我们发现 LabelGAN 的生成器在更新过程中，倾向于将每个样本优化到可以被判别器认为属于某个特定类别。这样一个性质暗示了标签信息如何帮助生成器得到更好的生成效果。

在展开分析的具体细节之前，我们需要引入以下关于交叉熵的引理，以便让后面的分析更加易读：

引理 4.1 令 $\sigma(l)$ 表示当前的 softmax 概率分布，其中 l 表示分类器的逻辑值向量， σ 表示 softmax 函数。令 \hat{p} 表示目标概率分布，那么

$$-\frac{\partial H(\hat{p}, \sigma(l))}{\partial l} = \hat{p} - \sigma(l). \quad (4-12)$$

证明

$$\begin{aligned} & -\left(\frac{\partial H(\hat{p}, \sigma(l))}{\partial l}\right)_k = -\frac{\partial H(\hat{p}, \sigma(l))}{\partial l_k} = \frac{\partial \sum_i \hat{p}_i \log \sigma(l)_i}{\partial l_k} = \frac{\partial \sum_i \hat{p}_i \log \frac{\exp(l_i)}{\sum_j \exp(l_j)}}{\partial l_k} \\ &= \frac{\partial \sum_i \hat{p}_i (l_i - \log \sum_j \exp(l_j))}{\partial l_k} = \frac{\partial \sum_i \hat{p}_i l_i}{\partial l_k} - \frac{\partial \log(\sum_j \exp(l_j))}{\partial l_k} = \hat{p}_k - \frac{\exp(l_k)}{\sum_j \exp(l_j)} \\ &\Rightarrow -\frac{\partial H(\hat{p}, \sigma(l))}{\partial l} = \hat{p} - \sigma(l). \end{aligned} \quad \square$$

在 LabelGAN 中，考虑单个生成样本 x 的话，它作为生成器整体目标函数的一部分，其目标函数是 $L_G^{\text{lab}}(x) = H([1, 0], [D_r(x), D_{K+1}(x)])$ ，参考公式 (4-5)。由引理 4.1 可知， $L_G^{\text{lab}}(x)$ 关于逻辑值向量 $l(x)$ 的梯度如下：

$$\begin{aligned} & -\frac{\partial L_G^{\text{lab}}(x)}{\partial l_k(x)} = -\frac{\partial H([1, 0], [D_r(x), D_{K+1}(x)])}{\partial l_r(x)} \frac{\partial l_r(x)}{\partial l_k(x)} \\ &= (1 - D_r(x)) \frac{D_k(x)}{D_r(x)}, \quad k \in \{1, \dots, K\}, \end{aligned} \quad (4-13)$$

$$\begin{aligned} & -\frac{\partial L_G^{\text{lab}}(x)}{\partial l_{K+1}(x)} = -\frac{\partial H([1, 0], [D_r(x), D_{K+1}(x)])}{\partial l_{K+1}(x)} \\ &= 0 - D_{K+1}(x) = -(1 - D_r(x)). \end{aligned} \quad (4-14)$$

基于以上结论，我们进一步有 $L_G^{\text{lab}}(x)$ 关于样本 x 的梯度是：

$$\begin{aligned} -\frac{\partial L_G^{\text{lab}}(x)}{\partial x} &= \sum_{k=1}^K -\frac{\partial L_G^{\text{lab}}(x)}{\partial l_k(x)} \frac{\partial l_k(x)}{\partial x} - \frac{\partial L_G^{\text{lab}}(x)}{\partial l_{K+1}(x)} \frac{\partial l_{K+1}(x)}{\partial x} \\ &= (1 - D_r(x))g\left(\sum_{k=1}^K \frac{D_k(x)}{D_r(x)} \frac{\partial l_k(x)}{\partial x} - \frac{\partial l_{K+1}(x)}{\partial x} g\right) \\ &= (1 - D_r(x)) \sum_{k=1}^{K+1} \alpha_k^{\text{lab}}(x) \frac{\partial l_k(x)}{\partial x}, \end{aligned} \quad (4-15)$$

其中

$$\alpha_k^{\text{lab}}(x) = \begin{cases} \frac{D_k(x)}{D_r(x)} & k \in \{1, \dots, K\} \\ -1 & k = K+1 \end{cases}. \quad (4-16)$$

从上面的公式，我们可以看出， $L_G^{\text{lab}}(x)$ 关于样本逻辑值向量的整体梯度是 $1 - D_r(x)$ ，这个和原始 GAN^[1] 是一致的。不同的是，这里表示真实数据的那个类，被拓展成了 K 个更具体的类别，因而在真实类别上的梯度被进一步分散到了各个具体类别上。而有趣的是，梯度分散到具体类别时以它们各自当前的概率比率 $\frac{D_k(x)}{D_r(x)}$ 为权重。

这样的梯度分配使得生成器在梯度更新的时候自然地利用到了判别器里面蕴含的类别信息。因为这个梯度分配有这样的性质：对于每个生成样本而言，它当前属于某个类别的概率越高，那么梯度中朝这个类别优化的权重就会越大。因此，如果考虑单个样本的话，来自判别器的梯度使得每个生成样本倾向于变成属于某一个特定类别的样本。也就是说，在 LabelGAN 中，每个样本是被朝着某个特定的类别优化，而不是像原始 GAN 那样朝着整体的真实的方向优化。因此，我们把 LabelGAN 当做一个隐式的带有目标类别标签的模型。

把每个样本朝着一个具体的类别优化能提升样本质量。回想一下，许多相关工作中也有类似的启示。^[18] 显示如果为每个类别单独训练一个生成对抗网络，那么最终效果将显著提升。而 AC-GAN^[63] 中，通过给生成器引入一个额外的损失函数以使每个生成样本能被分类器分作每个具体的类别，它得到了更好的生成效果。

4.4 激活最大化生成对抗网络

在 LabelGAN 中，生成器从判别器的逻辑值向量上拿到关于 K 个具体类别的信息，并倾向于把每个样本优化到属于某个特定类别。尽管如此，LabelGAN 的解决方案并不完美。我们可以观察到，它其实存在梯度耦合的问题，因为它在优化任何一个样本的时候，都是同时鼓励它把每个真实类别的逻辑值提高。尽管它把

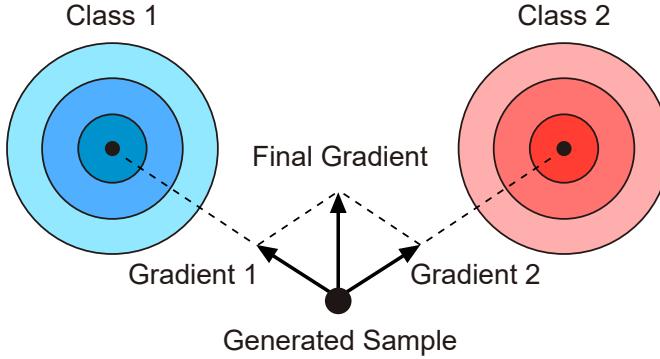


图 4-1 梯度耦合问题的图示。当一个样本被同时鼓励着朝两个或者更多的类别优化时，其整体梯度方向可能并不指向任何一个类别。这个问题可以通过给每个样本指定一个唯一的具体类别标签来解决。

Figure 4-1 An illustration of the overlaid-gradient problem. When two or more classes are encouraged at the same time, the combined gradient may direct to none of these classes. It could be addressed by assigning each generated sample a specific target class instead of the overall real class.

更多的权重分配给了当前概率较高的那些，它的最终梯度是一个来自各个标签的梯度的一个加权平均。如图 4-1 所示，加权平均之后的梯度可能并不指向任何一个类别。

在类别互斥的分类设定中，每个合理的样本都应该仅仅属于某一个类，并将被一个训练的很好的分类器以高概率分类到某个特定的类别。因此，一个直接而有效的解决方案是，在优化过程中，将每个样本朝着一个唯一的类别优化。

我们可以姑且认为每个样本都有一个预分配的标签，后面我们会进一步讨论标签分配的问题。在 LabelGAN 的基础上，给每个样本分配一个类别标签 y ，那么它的目标函数可以改写为：

$$L_G^{\text{am}} = \mathbb{E}_{(x,y) \sim G} [H(v(y), D(x))], \quad (4-17)$$

$$L_D^{\text{am}} = \mathbb{E}_{(x,y) \sim p_{\text{data}}} [H(v(y), D(x))] + \mathbb{E}_{x \sim G} [H(v(K+1), D(x))], \quad (4-18)$$

这里 $v(y)$ 和小节 4.2.1 中一样，是一个标签的向量化操作。我们把该模型称之为激活最大化生成对抗网络（Activation Maximization Generative Adversarial Networks），缩写为 AM-GAN。起这个名字其实是有更深一层的含义的，我们在小节 4.5.2 中揭示该方法和激活最大化（Activation Maximization）之间的联系。

AM-GAN 和 LabelGAN 的唯一区别在于生成器的目标函数。在 AM-GAN 中，每个样本都有一个具体的标签，它能解决 LabelGAN 中存在的梯度耦合问题。AC-GAN^[63] 也为每个样本显示地分配了一个标签，但是正如我们接下来要讲的：AM-

GAN 和 AC-GAN 其实有着重要的本质上的区别。

4.4.1 交叉熵分解以及模型之间的联系

LabelGAN 和 AM-GAN 都是具有 $K+1$ 个类的模型。我们接下来引入以下关于交叉熵分解的引理。这个引理将 $k+1$ 类的交叉熵分解为一个 2 类的交叉熵和一个 K 类的交叉熵。通过这个引理，我们将建立基于 $K+1$ 类分类器的生成对抗网络模型、基于 2 类分类器的生成对抗网络模型以及基于 K 类分类器的生成对抗网络模型之间的联系。

引理 4.2 给定一个 $K+1$ 维的向量 $v = [v_1, \dots, v_{K+1}]$ ，定义记号 $v_{1:K} \triangleq [v_1, \dots, v_K]$ ， $v_r \triangleq \sum_{k=1}^K v_k$ ， $R(v) \triangleq v_{1:K}/v_r$ ， $F(v) \triangleq [v_r, v_{K+1}]$ 。令 $\hat{p} = [\hat{p}_1, \dots, \hat{p}_{K+1}]$ ， $p = [p_1, \dots, p_{K+1}]$ ，则：

$$H(\hat{p}, p) = \hat{p}_r H(R(\hat{p}), R(p)) + H(F(\hat{p}), F(p)). \quad (4-19)$$

证明

$$\begin{aligned} H(\hat{p}, p) &= -\sum_{k=1}^K \hat{p}_k \log p_k - \hat{p}_{K+1} \log p_{K+1} = -\hat{p}_r \sum_{k=1}^K \frac{\hat{p}_k}{\hat{p}_r} \log \left(\frac{p_k}{p_r} p_r \right) - \hat{p}_{K+1} \log p_{K+1} \\ &= -\hat{p}_r \sum_{k=1}^K \frac{\hat{p}_k}{\hat{p}_r} \left(\log \frac{p_k}{p_r} + \log p_r \right) - \hat{p}_{K+1} \log p_{K+1} \\ &= -\hat{p}_r \sum_{k=1}^K \frac{\hat{p}_k}{\hat{p}_r} \log \frac{p_k}{p_r} - \hat{p}_r \log p_r - \hat{p}_{K+1} \log p_{K+1} \\ &= \hat{p}_r H(R(\hat{p}), R(p)) + H(F(\hat{p}), F(p)). \end{aligned} \quad \square$$

通过引理 4.2，AM-GAN 中生成器的损失函数可以被分解为：

$$L_G^{\text{am}}(x) = H(v(x), D(x)) = v_r(x) \cdot \underbrace{H(R(v(x)), R(D(x)))}_{\text{Auxiliary Classifier G Loss}} + \underbrace{H(F(v(x)), F(D(x)))}_{\text{LabelGAN G Loss}}. \quad (4-20)$$

我们发现，第二项正好等于 LabelGAN 中生成器的目标函数：

$$H(F(v(x)), F(D(x))) = H([1, 0], [D_r(x), D_{K+1}(x)]) = L_G^{\text{lab}}(x). \quad (4-21)$$

而其中第一项 $H(R(v(x)), R(D(x)))$ 恰好又等价于 AC-GAN 中的生成器关于附属分类器的损失函数。类似的分析也可以作用于判别器。

从这个角度看，AM-GAN 和 AC-GAN 的不同之处在于：AM-GAN 是 LabelGAN 和附属分类器的组合，而 AC-GAN 是原始 GAN 和附属分类器的组合。不是很正式地：

$$\text{AM-GAN} = \text{Auxiliary Classifier} + \text{LabelGAN}$$

$$\text{AC-GAN} = \text{Auxiliary Classifier} + \text{GAN}$$

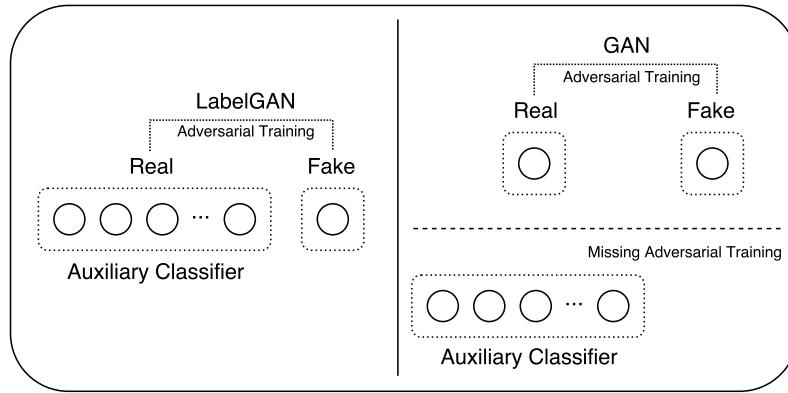


图 4-2 AM-GAN 的结构图（左侧）和 AC-GAN 的结构图（右侧）的对比。AM-GAN 可以被看作是 LabelGAN 和附属分类器的组合，而 AC-GAN 是原始 GAN 和附属分类器的组合。AM-GAN 是一个非层次化的结构，因而可以自然地在各个类别之间展开对抗训练，而 AC-GAN 是一个层次化的结构，它的对抗训练仅存在于顶层的真假二类分类器中。

Figure 4-2 AM-GAN (left) v.s. AC-GAN (right). AM-GAN can be viewed as a combination of LabelGAN and auxiliary classifier, while AC-GAN is a combination of vanilla GAN and auxiliary classifier. AM-GAN can naturally conduct adversarial training among all the classes, while in AC-GAN, adversarial training is only conducted at the real-fake level and missing in the auxiliary classifier.

其实，公式(4-20)中的附属分类器部分的损失函数，也可以看做是交叉熵版本的 CatGAN^[12]：CatGAN 直接优化 $H(R(D(x)))$ 以使得每个样本拥有一个高自信度被分类某一个类，而 AM-GAN 分解项的第一部分 $H(R(v(x)), R(D(x)))$ 与这个相对应，只不过这里将熵改成了交叉熵。因此，AM-GAN 也可以看作是一个交叉熵本版的 CatGAN 和 LabelGAN 的结合。

4.4.2 层次化模型与非层次化模型

利用引理 4.2，我们还可以把 AC-GAN 写成一个 $K+1$ 类的模型。以生成器的目标函数为例：

$$\begin{aligned} L_G^{ac}(x, y) &= \mathbb{E}_{(x,y) \sim G}[H([1, 0], [D_r(x), D_f(x)]) + H(u(y), C(x))] \\ &= \mathbb{E}_{(x,y) \sim G}[H(v(y), [D_r(x) \cdot C(x), D_f(x)])]. \end{aligned} \quad (4-22)$$

在这个 $K+1$ 类的模型中，这个 $K+1$ 类的分布被写成 $[D_r(x) \cdot C(x), D_f(x)]$ 。AC-GAN 最初的想法是引入一个附属的分类器来利用标签信息，然而事实证明 AC-GAN 可以被看作是一个层次化的 $K+1$ 类的模型，如图 4-2 所示。它包含一个顶层的二类

分类判别器和一个下层的 K 类分类器。与之相对的，AM-GAN 是一个非层次化的 $K+1$ 类模型，其中所有 $K+1$ 个类都处于判别器的同一层。

在层次化结构的 AC-GAN 中，对抗训练仅仅在顶层的二类分类判别器中存在，而在附属分类器中，对抗训练是缺失的。对抗训练是生成对抗网络收敛性保证的关键。以原始 GAN 为例，如果生成样本坍缩到了一个特定点 x ，那么有 $p_G(x) > p_{\text{data}}(x)$ 。按照概率分布的性质可以知道，必定存在另一个点 x' 满足 $p_G(x') < p_{\text{data}}(x')$ 。已知判别器的最优解满足 $D(x) = \frac{p_{\text{data}}(x)}{p_G(x)+p_{\text{data}}(x)}$ ，因此，坍缩的点 x 会有一个相对较小的判别器得分。又因为其他高分点的存在，如 x' ，最大化生成器的期望得分，在理论上能将生成器从坍缩状态中恢复过来。在实践过程中， p_G 和 p_{data} 的支撑剂通常是不相交的^[34]，尽管如此，生成对抗网络在训练过程中的整体表现保持一致：当样本坍缩到某个点时，该点就会得到一个相对较小的评分。

因为在附属分类器中缺少对抗训练，在 AC-GAN 中，即使生成器发生坍缩，它也不会受到来自于附属分类器损失函数的惩罚。在我们的实验中，我们发现 AC-GAN 更容易发生模式崩塌，并且如果减小附属分类器的损失函数的全中的话能一定程度缓解这个问题。^[35] 中也有类似的发现。在小节 4.5.3 中，我们会为 AC-GAN 的附属分类器引入额外的对抗训练，而实验表明该做法能显著提高 AC-GAN 的训练稳定性和样本质量。另一方面，AM-GAN 因为是一个非层次化的结构，所有类别的逻辑值之间能自然地形成对抗训练，因此不存在这个问题。

4.5 拓展内容

4.5.1 自动标签技术

在上面的讨论中，我们假设每个样本都有一个目标类别。一个可能的实现方式是像 AC-GAN^[63] 那样为每个样本预定义一个类别标签。这种做法实际上把模型变成了基于条件生成的生成对抗网络。而实际上，我们可以给每个样本自动地加上标签，并不需要预定义，这个过程中我们可以利用判别器当前对于生成样本的类别判断。给定判别器对样本的类别判断，一个自然地做法是直接选取当前概率最大的类作为该样本的目标类别：

$$y(x) \triangleq \arg \max_{i \in \{1, \dots, K\}} D_i(x), \forall x \quad (4-23)$$

我们把这种自动给生成样本加标签的技术称为，自动标签技术。

按照我们的实验，自动标签技术能为 AM-GAN 带来很大的样本质量的提升。自动标签技术其实也可以用到其他用到类别信息的模型中，比如 AC-GAN。

自动标签技术可以作为预定义标签的一个替换方案。我们实验中发现采用基于预定义标签的生成对抗网络模型较容易出现类内的模式崩塌，采用自动标签技术的生成对抗网络很少出现这个问题。此外，采用动态标签技术的生成对抗网络能保持从纯噪声中生成样本，这是有一定的潜在好处的，比如说，它可以支持平滑地对两个属于不同类别的样本进行线性插值。

4.5.2 从激活最大化的角度理解对抗训练

最大化激活（Activation maximization）是一个可视化神经网络神经元的传统方法^[91]，也被用于高可信度的图片生成^[90, 92]。这里方法里面，所要激活的神经元都是在某个已预训练好了的神经网络中。

而生成对抗网络的训练可以被看做一个带有对抗训练的激活最大化的过程。具体而言，生成器被训练以最大化生成样本在目标类别下的激活。而判别器被训练以区分真假样本，防止生成样本在真实类别下得到高的激活值。

值得注意的是，一个神经元的最大激活不一定是一张高质量的图片，它甚至可以是噪声。传统的利用最大化激活来做图片生成的方法都要引入一些关于激活图片的先验知识作为约束^[90, 92]。在生成对抗网络中，对抗训练能够检测到低质量图片（低质量的图片会被判别器高概率地认为是假的图片，从而不能达到目标类上的高激活值），因而能保证最大化激活的图片的质量。

对抗的最大化激活也能很好地解释 $-\log D_r(x)$ 技巧。而传统的 minimax 则会陷入困境，在两个分布的支撑集完全不重叠的情况下，minimax 目标函数的梯度将为零，无法训练。事实上，在这种情况下，生成对抗网络并不能保证收敛性，但对抗的最大化激活能很好地解释为什么可以使用 $-\log D_r(x)$ 技巧，以及 $-\log D_r(x)$ 技巧下，生成对抗网络是如何工作的。这并不能从 minimax 的角度很好地解释。

基于以上原因，我们为所有模型采用了 $-\log D_r(x)$ 技巧，并将我们的模型称之为最大化激活生成对抗网络（Activation Maximization Generative Adversarial Network），缩写为 AM-GAN。

4.5.3 为附属分类器引入对抗训练

实验上，我们发现 AC-GAN 很容易出现模式崩塌，并且如果减小生成器在附属分类器上的损失函数的权重将能有效地缓解这个问题。在小节4.5.2中，我们认为模式崩塌和附属分类器上缺少对抗训练有关。从对抗的激活最大化的角度，缺乏对抗训练还有一重问题：在附属分类器上的一个大的激活值不能保证样本的质量。因而，在 AC-GAN 中，原始 GAN 部分扮演着保证样本质量以及防止模式崩

塌的重要角色。

这里，我们为 AC-GAN 的附属分类器引入额外的对抗训练：

$$L_D^{ac+}(x, y) = \mathbb{E}_{(x,y) \sim G}[H(u(\cdot), C(x))], \quad (4-24)$$

这里 $u(\cdot)$ 表示一个均一分布。这个做法 CatGAN^[12] 中的对抗训练是类似的。实验中，我们发现引入附属分类器中的对抗训练后，AC-GAN 的稳定性和样本质量显著提升。

我们之前在定义 AC-GAN 的时候，去掉了附属分类器上 $\mathbb{E}_{(x,y) \sim G}[H(u(y))]$ 。按照我们的实验， $\mathbb{E}_{(x,y) \sim G}[H(u(y))]$ 确实可以提升 AC-GAN 的稳定性以较少模式崩塌，但它也会导致样本质量下降。因而，我们认为这并不是一个很好的做法，故将其舍去。我们建议将该项换成公式 (4-24) 所示的对抗训练。

4.6 评测指标

自 2014 年引入生成对抗网络以来，关于生成对抗网络的各种变体、拓展、改进不断涌现，它们带来了在理论和应用等方面的重要进展。但是，怎么衡量一个生成对抗网络的变体相较于其他方法的好坏一直以来是一个重要的挑战。

严格而准确地衡量生成模型的好坏通常而言较难，人们因此也常退而求其次用一些可视化的方法来直观展现模型的效果。典型的方法包括：可视化生成的样本、做模型参数或者样本做插值、做去噪或者填充任务做建模效果测试、寻找生成样本在真实数据中的最近邻^{[45][93]} 等等。但这类评判标准缺乏客观性和全面性，因而定性定量的评测指标必不可少。随着生成对抗网络等新兴生成模型的出现，一些用以评估生成模型的新指标也不断浮现。其中，Inception Score 是毋庸置疑地被最广泛使用的一个指标，但 Inception Score 同时也是最富有争议的一个指标。

在本节中，我们对广泛使用的评估指标 Inception Score 和其他相关指标进行了数学和实证分析。我们将说明目前被广泛接受的关于 Inception Score 如何工作的论断其实并不成立。与此同时，我们验证 Inception Score 具有一定的衡量样本多样性的能力。此外，基于以上分析，我们进一步提出 Inception Score 的一个修正版本，AM Score，来有效地衡量生成的样本质量。

4.6.1 Inception Score

作为新近提出的用于评估生成模型性能的评测指标，Inception Score 因能很好地吻合人对生成模型的评分，从而被广泛采用，但是 Inception Score 的可解释性一直是大家争辩的一个主题。

Inception Score 引入了一个在 ImageNet 上预先训练的公开可用的 Inception 模型。我们记该模型为 C 。通过将该 Inception 模型应用于每个生成的样本 x ，我们获得每个样本 x 相应的类概率分布 $C(x)$ 。然后 Inception Score 通过以下方式计算：

$$\text{Inception Score} = \exp(\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]), \quad (4-25)$$

这里 \mathbb{E}_x 是 $\mathbb{E}_{x \sim G}$ 的缩写； $\bar{C}^G = \mathbb{E}_x[C(x)]$ 表示所有生成样本经 C 判断得出的概率的平均；KL 则表示 Kullback-Leibler(KL) divergence。

我们接下来证明 $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 可以被分解为两项：

$$\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)] = H(\bar{C}^G) + (-\mathbb{E}_x[H(C(x))]). \quad (4-26)$$

首先，我们把 KL divergence 按照定义展开，有以下结论：

$$\begin{aligned} \text{KL}(p \parallel q) &= \sum_i p_i \log \frac{p_i}{q_i} = \sum_i p_i \log p_i - \sum_i p_i \log q_i \\ &= -H(p) + H(p, q). \end{aligned} \quad (4-27)$$

其次，我们可以证明以下引理：

引理 4.3 令 $p(x)$ 是样本 x 的一个在类别上的概率分布，其中 x 服从某个分布，而 \bar{p} 表示另一个概率分布，则

$$\mathbb{E}_x[H(p(x), \bar{p})] = H(\mathbb{E}_x[p(x)], \bar{p}). \quad (4-28)$$

证明

$$\begin{aligned} \mathbb{E}_x[H(p(x), \bar{p})] &= \mathbb{E}_x[-\sum_i p_i(x) \log \bar{p}_i] \\ &= -\sum_i \mathbb{E}_x[p_i(x)] \log \bar{p}_i = -\sum_i (\mathbb{E}_x[p(x)])_i \log \bar{p}_i \\ &= H(\mathbb{E}_x[p(x)], \bar{p}). \end{aligned}$$

□

从而，我们有：

$$\begin{aligned} \log(\text{Inception Score}) &= \mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)] \\ &= \mathbb{E}_x[H(C(x), \bar{C}^G)] - \mathbb{E}_x[H(C(x))] = H(\mathbb{E}_x[C(x)], \bar{C}^G) - \mathbb{E}_x[H(C(x))] \\ &= H(\bar{C}^G) + (-\mathbb{E}_x[H(C(x))]), \end{aligned} \quad (4-29)$$

4.6.2 Inception Score 的一个等价变种: Mode Score

作为 Inception Score 的一个拓展, mode_gan 指出 Inception Score 没有充分利用先验分布, 并提出将先验分布引入到 Inception Score, 从而提出了 Mode Score。其表达式如下:

$$\text{Mode Score} = \exp(\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^{\text{train}})] - \text{KL}(\bar{C}^G \parallel \bar{C}^{\text{train}})), \quad (4-30)$$

其中的主要区别是, Mode Score 将训练数据的平均类别概率分布 \bar{C}^{train} 引入作为 KL divergence 的参考值。

而事实上, 我们可以证明, Mode Score 其实等价于 Inception Score:

$$\begin{aligned} & \log(\text{Mode Score}) \\ &= \mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^{\text{train}})] - \text{KL}(\bar{C}^G \parallel \bar{C}^{\text{train}}) \\ &= \mathbb{E}_x[H(C(x), \bar{C}^{\text{train}})] - \mathbb{E}_x[H(C(x))] - H(\bar{C}^G, \bar{C}^{\text{train}}) + H(\bar{C}^G) \\ &= H(\bar{C}^G) + (-\mathbb{E}_x[H(C(x))]). \end{aligned} \quad (4-31)$$

$\Rightarrow \text{Inception Score} = \text{Mode Score}.$

4.6.3 Inception Score 的分解项并不像想象中那样工作

人们通常认为 Inception Score 是这样工作的: Inception Score 分为两项; 第一项 $H(\bar{C}^G)$ 越高表示生成样本有着更好的多样性, 因为它意味着整体分布是均匀的; 而第二项 $-\mathbb{E}_x[H(C(x))]$ 越高则表示生成样本的质量越高, 因为它意味着每个样本的概率分布都很集中, 这就进一步意味着这些样本能被很好地分类, 那么也就意味着样本质量很高。

然而, 如果以 Inception 模型作为分类器的话, 以 CIFAR-10 为例, 它的数据其实并不是均匀地分布在各个类上, 如图 4-4a 所示。我们不禁产生疑惑: 是否 $H(\bar{C}^G)$ 越大就表示模型的模式覆盖率越高, 是否 $H(C(x))$ 越小就表示样本质量越好。

我们基于此展开了进一步的探索。我们从实验中发现, 如图 4-3b 所示, $H(\bar{C}^G)$ 的值通常是随着训练的进行而不断下降的, 但按照分解项的理解, 我们期望 $H(\bar{C}^G)$ 是不断上升的。而与此同时, 我们可以看到, $H(C(x))$ 的值对于不同的数据而言存在相当程度的变化, 如图 4-4b 所示; 这意味着即使是在真实数据中, 基于 Inception 模型的 $H(C(x))$ 也会强烈的偏好一些样本。这样的偏向性直接就导致它不能是一个很好评测指标。

值得注意的是, Inception Score 最后有一个指数操作, 而 $H(C(x))$ 的数值范围在 0 到 7, 这足以让 Inception Score 产生巨大的变化。除了单个样本上 $H(C(x))$ 的分

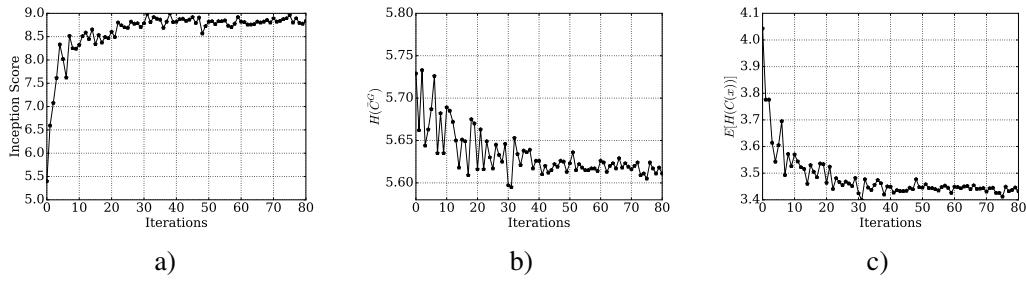


图 4-3 Inception Score 和它的分解项随训练的变化曲线。a) Inception Score, 也即 $\exp(H(\bar{C}^G) - \mathbb{E}_x[H(C(x))])$; b) $H(\bar{C}^G)$; c) $\mathbb{E}_x[H(C(x))]$ 。人们通常认为 Inception Score 中 $H(\bar{C}^G)$ 的值衡量着生成样本的多样性。按照预期, $H(\bar{C}^G)$ 在训练过程中应该逐渐增大。然而, 在实际训练过程中它的值却在不断减小, 如图 b) 所示。

Figure 4-3 Training curves of Inception Score and its decomposed terms. a) Inception Score, i.e. $\exp(H(\bar{C}^G) - \mathbb{E}_x[H(C(x))])$; b) $H(\bar{C}^G)$; c) $\mathbb{E}_x[H(C(x))]$. A common understanding of Inception Score is that: the value of $H(\bar{C}^G)$ measures the diversity of generated samples and is expected to increase in the training process. However, it usually tends to decrease in practice as illustrated in b).

布不均, 我们也能观察到类别上的一些偏向性, 如图 4-4b 所示: 对于卡车 (trucks) 而言 $\mathbb{E}_x[H(C(x))]=2.14$, 而对于鸟类而言 $\mathbb{E}_x[H(C(x))]=3.80$ 。

综上所述, 基于 Inception 模型, Inception Score 的两个指标似乎都不能正常工作。这意味着从分解项的角度理解 Inception Score 似乎是行不通的, 我们接下来将要从一种新的角度解释 Inception Score, 并说明它具有一定程度的衡量样本多样性的能力。

4.6.4 Inception Score 衡量样本多样性的能力

从分解项的角度理解 Inception Score 出现问题, 其实是可以理解的, 因为这两项本来就具有强相关性。单独的分析两个分解项的含义势必不够全面和准确。因此, 这里我们试着从整体的角度理解 Inception Score。我们回到 Inception Score 最初的定义: $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 。在这个形式下, 我们可以把 Inception Score 解释为: 它要求每个样本的分布和整体平均分布尽可能的不一致。这很自然的就和样本的多样性产生了联系, 我们下面进一步解释。

首先, 实验上我们经常能发现, 一个生成器出现模式崩塌, 我们所求得的 Inception Score 就会很低。我们可以考虑一个极端情况来理解这个现象: 假设所有生成样本都坍缩到了一个点, 那么 $C(x)=C^G$, 于是我们可以得出 Inception Score 等于 1.0, 这也是 Inception Score 所能取到的最小值。为了模拟更复杂的情况, 我们设计了以下实验: 给定一组包含 N 个点的数据 $\{x_0, x_1, x_2, \dots, x_{N-1}\}$, 其中每个点 x_i

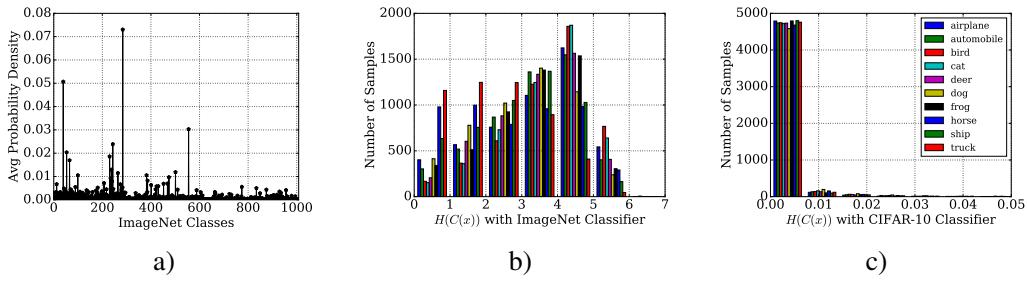


图 4-4 CIFAR-10 训练数据的统计信息。a) 在 ImageNet 类别上的 \bar{C}^G ; b) 各个类别的数据基于 ImageNet 分类器的 $H(C(x))$ 的分布; c) 各个类别的数据基于 CIFAR-10 分类器的 $H(C(x))$ 的分布。如果基于 ImageNet 分类器, 也即基于 Inception 模型, CIFAR-10 训练数据的 $H(C(x))$ 的值是多变的、不统一的, 这意味着即使在真实数据上, $H(C(x))$ 这样一个指标也会强烈的倾向于某一些样本。也即, $H(C(x))$ 有很强的偏向性, 从而不是一个好的指标。于此相对的, $H(C(x))$ 在基于 CIFAR-10 的分类器上, 几乎对于所有的样本而言都具有一个很小的值, 因此, 如果采用 CIFAR-10 分类器, $H(C(x))$ 是可以作为一个评判样本质量的指标的。

Figure 4-4 Statistics of the CIFAR-10 training images. a) \bar{C}^G over ImageNet classes; b) $H(C(x))$ distribution with ImageNet classifier of each class; c) $H(C(x))$ distribution with CIFAR-10 classifier of each class. With the Inception model, the value of $H(C(x))$ score of CIFAR-10 training data is variant, which means, even in real data, it would still strongly prefer some samples than some others. $H(C(x))$ on a classifier that pre-trained on CIFAR-10 has low values for all CIFAR-10 training data and thus can be used as an indicator of sample quality.

拥有分布 $C(x_i) = v(i)$ 并且表示类别 i 。这里的 $v(i)$ 是一个类别的向量化操作, 即长度为 N , 在第 i 个位置为 1, 其他位置为 0。我们随机地在这组数据中丢掉 m 个点, 然后计算 $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 的值并画曲线如图 4-6。我们可以考到, 当 $N - m$ 增加的时候, $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 的值基本上也是单调的上升, 这意味着这个指标可以很好地捕捉到模式崩塌或者说是衡量生成样本的多样性。

4.6.5 AM Score

按照我们的分析, Inception Score 是某种样本多样性的指标, 那么一个尚未解答的问题是: 生成样本多样性是否生成样本质量直接相关。从以上分析看来, 样本多样性和样本质量之间似乎没有什么必然的联系。不过一个可能的解释是: 在生成对抗网络中, 一般情况下, 样本的多样性和样本的质量通常呈很好的相关性。因为生成对抗网络的目标函数一般地要求生成分布等于真实分布, 这同时要求样本质量和样本多样性。

尽管如此, 就像我们之前在小节4.6.3所说的, 没有任何证据表明采用 ImageNet 分类器能够精确地衡量 CIFAR-10 数据的样本质量。为了弥补 Inception Score 不能

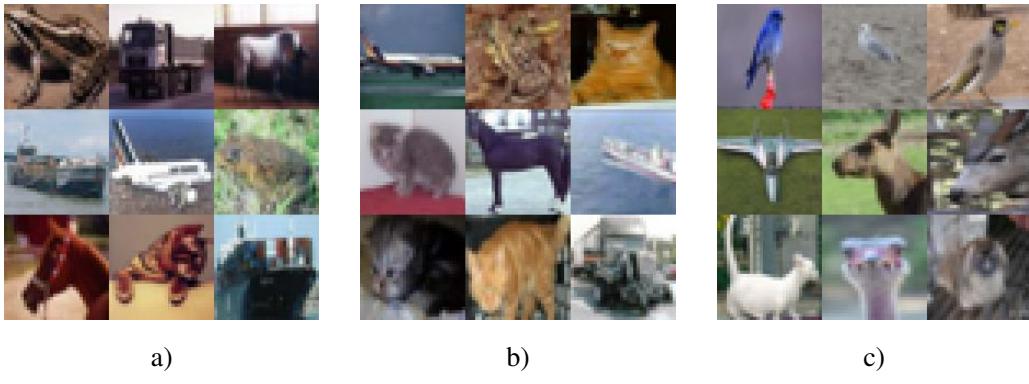


图 4-5 真实图片的 Inception 单样本熵。a) $0 < H(C(x)) < 1$; b) $3 < H(C(x)) < 4$; c) $6 < H(C(x)) < 7$ 。各组样本的质量没有明显差异，这意味着 $H(C(x))$ 不是一个好的表示样本质量的指标。

Figure 4-5 $H(C(x))$ of Inception Score in Real Images. a) $0 < H(C(x)) < 1$; b) $3 < H(C(x)) < 4$; c) $6 < H(C(x)) < 7$. There is no obvious difference in sample quality. It implies $H(C(x))$ is not a good indicator of sample quality.

准确衡量样本质量的问题，我们提出一个新指标，它将会采用一个和数据相匹配的分类器。

首先，我们考虑一个问题：为什么 Inception Score 选用的是 ImageNet 分类器。考虑情况如下，如果每个点 x_i 有多个变种，比如说 x_i^1, x_i^2, x_i^3 。考虑以下情况，模型仅仅生成 x_i^1 ，而不生成 x_i^2 和 x_i^3 。 $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 是无法检测到这种情况的。也即，如果采用一个和数据相匹配的分类器的话，Inception Score 势必会无法检测到类别内部的模式崩塌。这可能能够解释，为什么用于给 CIFAR-10 打分的 Inception Score 采用的不是 CIFAR-10 的分类器而是 ImageNet 数据的分类器。同时可以看出的一点是：采用一个别的分类器是可能可以避免无法检测类内模式崩塌这个问题的。

关于“怎样的分类器才是最好的”这个问题我们留作后续工作。而我们接下来要说明的是：一个和数据相匹配的分类器可以作为样本质量的测量指标。如图 4-4c 所示，如果采用 CIFAR-10 的分类器，CIFAR-10 的绝大多数样本都有一个很小并且很接近的 $H(C(x))$ 。具体来说，99.6% 的数据的的 $H(C(x))$ 小 0.05。这表明基于 CIFAR-10 分类器的 $H(C(x))$ 能用来评判 CIFAR-10 生成数据的质量的好坏。

除了将 ImageNet 分类器改为和训练数据相对应的分类器外，我们还注意到在 Inception Score 中 \bar{C}^G 实际上也是有问题的：当数据本身不是均匀分布的时候， $\arg \min H(\bar{C}^G)$ 是一个均一分布，这与最优解是本身不均一分部的数据相矛盾。因此，我们提出将训练数据的平均分布 \bar{C}^{train} 加入考虑，我们将 $H(\bar{C}^G)$ 换成 \bar{C}^{train} 和 \bar{C}^G 之间的 KL divergence。

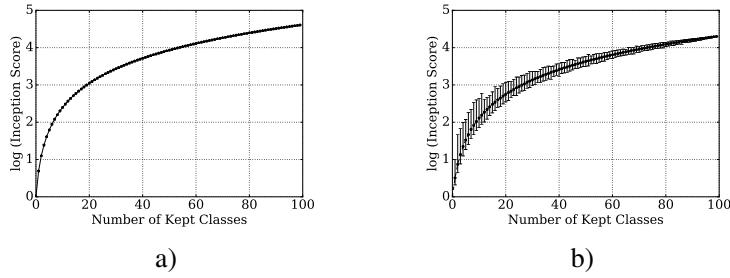


图 4-6 Inception Score 的模式崩塌测试。a) 数据在类别上均匀分布; b) 数据在类别上呈高斯分布。可以看到, $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ 随着被保留的模式的数目的增长而单调地上升。这意味着 Inception Score 能很好地捕捉模式崩塌。因为, 某种程度上能很好地刻画样本的多样性。图 b) 中的误差条形图表示的是在 1000 个随机实验中的最小和最大值。

Figure 4-6 Mode dropping analysis of Inception Score. a) Uniform density over classes; b) Gaussian density over classes. The value of $\mathbb{E}_x[\text{KL}(C(x) \parallel \bar{C}^G)]$ monotonically increases in general as the number of kept classes increases, which illustrates Inception Score is able to capture the mode dropping and the diversity of the generated distributions. The error bar indicates the min and max values in 1000 random dropping.

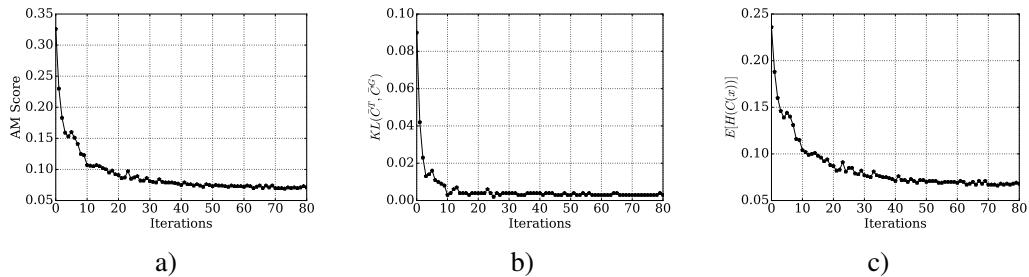


图 4-7 AM Score 和它的分解项随训练的变化曲线。a) AM Score, 也即 $\text{KL}(\bar{C}^{\text{train}}, \bar{C}^G) + \mathbb{E}_x[H(C(x))]$; b) $\text{KL}(\bar{C}^{\text{train}}, \bar{C}^G)$; c) $\mathbb{E}_x[H(C(x))]$ 。所有这些指标都正常工作: 在训练过程中趋于下降。

Figure 4-7 Training curves of AM Score and its decomposed terms. a) AM Score, i.e. $\text{KL}(\bar{C}^{\text{train}}, \bar{C}^G) + \mathbb{E}_x[H(C(x))]$; b) $\text{KL}(\bar{C}^{\text{train}}, \bar{C}^G)$; c) $\mathbb{E}_x[H(C(x))]$. All of them works properly (going down) in the training process.

综合起来, 所提出 AM Score 的表达式如下:

$$\text{AM Score} \triangleq \text{KL}(\bar{C}^{\text{train}}, \bar{C}^G) + \mathbb{E}_x[H(C(x))], \quad (4-32)$$

它要求 \bar{C}^G 尽可能低接近 \bar{C}^{train} , 并且每个样本 x 都有一个很小的 $C(x)$ 。这意味着, 只有当生成数据整体分布和真实数据接近, 并且样本质量很高时, AM Score 才会很小。值得注意的是, AM Score 的最小值是 0, 并且数值是越小越好, 这个和 Inception Score 不同。我们在图 4-7 中呈现了一个 AM Score 的训练曲线的实例,

从中我们可以看到 $\bar{C}^{\text{train}}, \bar{C}^G$ 和 $\mathbb{E}_x[H(C(x))]$, 以及 AM Score 整体, 在训练过程中都逐渐减小。

4.7 实验分析

为了经验地验证我们的分析和所提出的解决方案, 我们在 CIFAR-10 和 Tiny-ImageNet¹的生成任务上做了大量对比试验。CIFAR-10 是最经典的测试生成模型效果的数据集之一, 而 Tiny-ImageNet 是 ImageNet 的一个简化版, 拥有 200 个类别, 其中每个类别含有 500 张训练图片。我们采用 Inception Score^[15] 和 AM Score (详见小节4.6.5) 作为评测指标。Inception Score 中的分类器我们依照传统采用的是 ImageNet 上训练的 Inception 模型, 而对于 AM Score, 我们为每个数据集预训练了一个 DenseNet^[88] 分类器模型。Inception Score 越高越好, 而 AM Score 越小越好。此外, 我们也采用 [63] 中提出的 MS-SSIM^[94] 来检测是否存在类内模式崩塌。

实验中, 我们采用的网络是在 DCGAN^[9] 的基础上稍微修改过一个版本, 具体结构我们列在附录4.B中。我们把各个模型的部分图片样本包含在了图 4-9等等附录图中。

4.7.1 附属分类器的作用探究

在 AC-GAN 中附属分类器和判别器共享大部分参数。我们在实验部分想回答的第一个问题是: 在不引入额外损失函数的前提下, 训练一个附属分类器, 是否能提高生成样本的质量。也即在 AC-GAN 中, 如果生成器的损失函数仅保留原始 GAN 的部分。我们把这个模型记为 GAN*。

如表 4-1所示, 它确实一定程度上提升了生成对抗网络的生成样本的质量, 但是相比其他进一步的模型, 提升的幅度非常有限。这表明, 引入和类别相关的损失函数是充分利用类别信息的关键。

4.7.2 不同模型之间的对比

采用预定义的标签会让模型变成条件生成的版本, 导致和从单纯噪声生成样本的模型有本质区别, 为了相对公平地对别这些模型, 我们在这部分的实验中为所有需要生成样本标签的模型采用动态标签技术。实验中, 除了判别器最后一层的逻辑值数目外, 我们保持整体网络结构以及超参数在各个模型中一致, 以更公平地对比不同的模型。

¹<https://tiny-imagenet.herokuapp.com/>

表 4-1 各种模型之间的对比。在同一列的所有模型共享相同的网络结构和超参数。在如果模型中生成样本需要类别标签，我们根据它的所在列为它提供预定义的标签（predefined）或者动态标签（dynamic）。

Table 4-1 Comparisons among different models. Models in the same column share the same network structures & hyper-parameters. We applied dynamic / predefined labeling for models that require target classes.

| Model | Inception Score | | | | AM Score | | | |
|---------------------|--------------------|--------------------|---------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | CIFAR-10 | | Tiny ImageNet | | CIFAR-10 | | Tiny ImageNet | |
| | dynamic | predefined | dynamic | predefined | dynamic | predefined | dynamic | predefined |
| GAN | 7.04 ± 0.06 | 7.27 ± 0.07 | - | - | 0.45 ± 0.00 | 0.43 ± 0.00 | - | - |
| GAN* | 7.25 ± 0.07 | 7.31 ± 0.10 | - | - | 0.40 ± 0.00 | 0.41 ± 0.00 | - | - |
| AC-GAN | 7.41 ± 0.09 | 7.79 ± 0.08 | 7.28 ± 0.07 | 7.89 ± 0.11 | 0.17 ± 0.00 | 0.16 ± 0.00 | 1.64 ± 0.02 | 1.01 ± 0.01 |
| AC-GAN ⁺ | 8.56 ± 0.11 | 8.01 ± 0.09 | 10.25 ± 0.14 | 8.23 ± 0.10 | 0.10 ± 0.00 | 0.14 ± 0.00 | 1.04 ± 0.01 | 1.20 ± 0.01 |
| LabelGAN | 8.63 ± 0.08 | 7.88 ± 0.07 | 10.82 ± 0.16 | 8.62 ± 0.11 | 0.13 ± 0.00 | 0.25 ± 0.00 | 1.11 ± 0.01 | 1.37 ± 0.01 |
| AM-GAN | 8.83 ± 0.09 | 8.35 ± 0.12 | 11.45 ± 0.15 | 9.55 ± 0.11 | 0.08 ± 0.00 | 0.05 ± 0.00 | 0.88 ± 0.01 | 0.61 ± 0.01 |

表 4-2 CIFAR-10 上各个模型的平均 MS-SSIM 在十个类上的最大值。数值较高表明存在明显的类内模式崩塌。

Table 4-2 The maximum value of mean MS-SSIM of various models over the ten classes on CIFAR-10. High-value indicates obvious intra-class mode collapse.

| | AC-GAN | AC-GAN ⁺ | LabelGAN | AM-GAN |
|------------|-------------|---------------------|----------|--------|
| predefined | 0.61 | 0.39 | 0.35 | 0.36 |
| dynamic | 0.35 | 0.36 | 0.32 | 0.36 |

如表 4-1 所示，AC-GAN 达到了比原始 GAN 更优的样本质量，但是伴有明显的模式崩塌。表 4-2 中显示，AC-GAN 的 MS-SSIM 指标为 0.61，显著高于其他模型。而通过在它的基础上引入额外的对抗训练，AC-GAN⁺ 的表现显著优于 AC-GAN。我们也试着将 AC-GAN 中生成器的附属分类器上的损失函数的权重降低（变成原来的十分之一），结果是：它能达到 7.19 的 Inception Score，0.23 的 AM Score，和 0.35 的最大平均 MS-SSIM。最大平均 MS-SSIM 为 0.35 意味着类内模式崩塌问题得到了明显的缓解。这意味着类内模式崩塌问题确实和附属分类器上的损失函数有关。

我们在定义 AC-GAN 的时候省略了原文中的损失函数 $\mathbb{E}_{(x,y) \sim G}[H(u(y), C(x))]$ ，如果我们将这部分损失函数加回 AC-GAN，事实证明，它将得到一个更差的 Inception Score 和 AM Score，不过好的一点是，它能消除模式崩塌。具体而言，在动态标签下，Inception Score 从 7.41 降到 6.48，而 AM Score 从 0.17 上升到 0.43；在预

定义标签下，Inception Score 从 7.79 降到 7.66，而 AM Score 从 0.16 上升到 0.20。按照我们的理解：通过鼓励附属分类器把生成样本分到它们的目标类别，它其实变相地减小了附属分类器的作用。那么自然的结果就是它所带来的问题（模式崩塌）也减小了，而同时它所带来的好处（提升样本质量）也减弱了。

作为一个隐式的目标类别模型，按照我们的分析 LabelGAN 存在梯度耦合问题，它的生成样本在 AM Score 意义下的平均熵在 0.124，与之相对的，AM-GAN 通过在 LabelGAN 的基础上显示的给每个样本一个类别标签，实现了 0.079 的样本平均熵。作为另一个显示目标类别的模型，AC-GAN⁺ 的样本平均熵为 0.102，也明显优于 LabelGAN。在表 4-1 中我们可以看到，AM-GAN 在各个设定下都得到了全面优于其他模型的结果。

4.7.3 和其他相关工作的对比

在上面的实验中，AM-GAN 达到了 8.83 的 Inception Score，它显著地优于其他相关模型，不论是相比于他们论文中报告的结果还是我们的重现的结果，见表 4-3。通过进一步优化模型参数，即为判别器的每一层增加更多的滤波器，AM-GAN 可以达到 8.91 的 Inception Score。这个分数也高于 [95] 中报告的结果，该论文在 WGAN-GP^[35] 的基础上通过标签分裂的方式强化了类别信息。

4.7.4 自动标签技术与预定义的标签

我们在实验中发现如果采用预定义的标签，也即基于类别标签作条件生成，模型趋于发生类内的模式崩塌。这个现象在训练的起始阶段很明显，到后期通常会不断恶化。

在生成对抗网络的训练过程中，保持生成器和判别器的平衡是一件很重要的事情。我们发现在不改变生成器结构的前提下，如果将自动标签改成预定义标签，那么生成器和判别器之间的平衡将很难达成。可能的解释是：为了防止类内模式崩塌，判别器需要很强的能力，但是，这样一个判别器又通常会过于强大，以致于打破了生成器和判别器之间最佳的平衡状态，使得最终样本质量下降。

尽管如此，我们还是找到了一组还不错的参数设定，并在此基础上做了对比实验。结果列于表 4-1 中。基本结论还是和之前的类似，AC-GAN⁺ 的结果优于 AC-GAN，而 AM-GAN 达到了最好的效果。

在预定义类别标签的版本中，尽管网络结构和超参数已经经过了细致的调整，整体上，各个模型在 Inception Score 上的表现依旧不好。可能的解释是，基于预定义标签的模型就算不发生明显的类内模式崩塌，样本的多样性趋于减少。因为

表 4-3 与其他相关工作对比，AM-GAN 在 Inception Score 的意义下，显著地优于其他各种方法。Splitting GAN 将类别通过分裂细化来增强类别信息，和我们的工作正交互补；而 AM-GAN 也同样有优于它的表现。

Table 4-3 Comparing with reported Inception Score in related works, AM-GAN significantly outperforms all the baseline methods. AM-GAN also outperforms the orthogonal work Splitting GAN which enhances the class label information via class splitting.

| Model | Score ± Std. |
|-------------------------------|--------------------|
| DFM ^[96] | 7.72 ± 0.13 |
| Improved GAN ^[15] | 8.09 ± 0.07 |
| AC-GAN ^[63] | 8.25 ± 0.07 |
| WGAN-GP + AC ^[35] | 8.42 ± 0.10 |
| SGAN ^[19] | 8.59 ± 0.12 |
| AM-GAN (our work) | 8.91 ± 0.11 |
| Splitting GAN ^[95] | 8.87 ± 0.09 |
| Real data | 11.24 ± 0.12 |

Inception Score 会相对较低（缺乏多样性），而 AM Score 会相对较低（样本质量较高、坍缩之后的样本质量较好）。

值得注意的，LabelGAN 中生成样本是不需要类别标签的。所以两组实验下模型是一致的，只是网络和超参数不同。但是可以注意到 Inception Score 和 AM Score 在预定义标签下均有明显下降。这验证了在这类生成对抗网络模型中平衡生成器和判别器的重要性。我们发现如果不需要担心类内模式崩塌，那么能为生成器和判别器找到更好的平衡。这也是动态标签技术的一个优势。

4.7.5 训练曲线

我们将模型训练过程中的 Inception Score 和 AM Score 随时间变化的训练曲线作图，如4-8。Inception Score 和 AM Score 都是由 50k 个样本计算得到。AC-GAN 的训练曲线相比其他模型有着更大的波动，这个可能和 AC-GAN 中原始 GAN 目标函数和附属分类器目标函数的冲撞有关。在训练初期，AM-GAN 在 Inception Score 的意义下和 LabelGAN 以及 AC-GAN 不相上下，但是在 AM Score 的意义下，AM-GAN 依旧明显好于其他模型。

从图中我们可以看出，相比于 Inception Score，AM Score 的训练曲线整体上

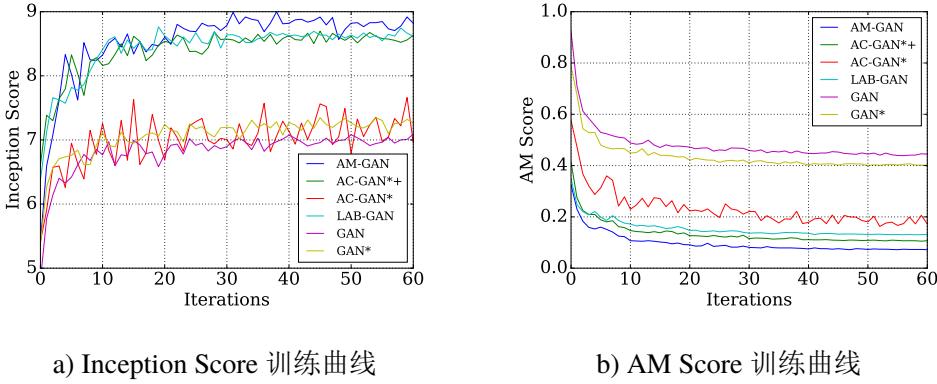


图 4-8 不同模型的训练曲线。需要标签的模型均采用的是自动标签技术。整体上 AM Score 相比 Inception Score 更加平滑稳定。相比 Inception Score，各个模型之间的区别在 AM Score 下可以更好地地区分出来。

Figure 4-8 The training curves of different models in the dynamic labeling setting. Comparing with Inception Score, AM Score is more stable in general. In terms of AM Score, different models are more distinguishable from each other.

更加平滑和稳定。值得一提的是，如果采用更多的样本的话，比如 $500k$ ，Inception Score 也会比现在更稳定，但 Inception Score 的计算代价其实很高，样本量倍增带来的计算代价不可忽略不计。总而言之，Inception Score 有一个比 AM Score 更高的存在一个样本复杂度，并且这样一个样本复杂度在它高昂的单样本计算代价下显得是个大问题。不过我们猜测选用其他分类器可能可以帮助缓解这个问题。

此外，我们可以看到 AM-GAN，LabelGAN 以及 AC-GAN⁺ 的 Inception Score 在初始阶段差异不大难以区分，但是在 AM Score 这个指标上，他们能被很好地区分出来。

4.7.6 用 Tiny-ImageNet 做进一步的验证

在 CIFAR-10 的实验中，各项实验的结果均和我们的分析一致，并且所提出的 AM-GAN 的结果显著优于其他模型。为了进一步验证，并不是因为实验过拟合到了 CIFAR-10 上，我们在数据集 Tiny-ImageNet 做相同的实验，看结论是否会发生变化。Tiny-ImageNet 包含更多的类别，而每个类别中的样本数量相对更少。我们将 Tiny-ImageNet 的样本从 64×64 下采样至 32×32 。实验中，我们直接采用 CIFAR-10 实验中的网络结构和超参数，仅仅修改数据集。结果列于表 4-1 中。从对比结果看，AM-GAN 依旧显著地优于其他方法。而 AC-GAN⁺ 也依旧比 AC-GAN 效果更好。

4.8 结论

在本章中，我们分析了当前的几个利用标签信息的生成对抗网络模型。按照我们的分析，LabelGAN 具有隐式地为每个生成样本分配标签的能力，但是同时也存在梯度耦合的问题。显示地为每个生成样本分配标签可以解决这个问题。我们指出直接将判别器中代表真实的类拓展成 K 个具体的类，比额外引入一个分类器要更好。因为前者可以自然地完成各个类之间的对抗训练，而后者在分类器中缺少对抗训练。我们通过大量实验验证了我们的分析，而所提出 AM-GAN 在实验中显著优于其他基线方法。

作为样本质量的评测指标，我们深入分析了被广泛采用，但同时又饱受争议的生成模型评测指标：Inception Score。我们指出目前比较普遍的关于 Inception Score 如何工作的理解其实并不正确，其主要原因是：Inception Score 采用的是基于 ImageNet 数据训练的分类器模型，而不是所测试数据相应的数据集。我们指出 Inception Score 其实一定程度上具有衡量样本多样性的能力，然而它是否能准确地衡量样本的质量却没有保证。我们仅仅能假设多样性和质量之间通常存在某种联系。为了更准确地衡量样本质量，我们进一步提出了 AM Score，并实验上对比了两种 Score 之间的联系和区别。

此外，我们提供了一个从对抗最大化激活角度理解生成对抗网络训练的方法，同时也强调了对抗在生成对抗网络训练中的作用。除此之外，我们还提出了动态标签技术，用以替换预定义标签的方案。动态标签有诸多好处，比如不容易出现模式崩塌、不需要改变模型的隐空间结构等等。

在本章中，我们把注意力集中在了生成器和它的样本质量上。而有一些相关工作关注的是判别器和半监督学习。作为后续工作，我们计划将我们的关注点拓展到判别器和半监督学习。我们在附录4.A里面将 AM-GAN 拓展到了无标签数据，可用于无监督学习和半监督学习。在本章中，我们的生成对抗网络模型均为原始 GAN 的变种。如何将标签信息较好的引入较新的生成对抗网络模型中也是一个非常值得探索的方向。

本章附录

4.A 将 AM-GAN 拓展到无标签数据

在正文部分，AM-GAN 假设数据集的所有数据都是带有标签的。然而在很多实际的场景中，更多的数据是无标签的（半监督学习），甚至所有数据都是无标签的（无监督学习）。在这章中，我们将 AM-GAN 拓展到无标签数据，是它能适应半监督学习和无监督学习的场景。我们的做法将延续自动标签的技术，而其核心思想也类似于 CatGAN^[12]。

4.A.1 半监督学习

在半监督学习的设定中，我们可以在 AM-GAN 中加入以下损失函数以利用无标签数据。

$$L_D^{\text{unl}} = \mathbb{E}_{x \sim p_{\text{unl}}} [H(v(x), D(x))]. \quad (4-33)$$

其中 $p_{\text{unl}}(x)$ 表示无标签数据的分布， $v(x)$ 和生成样本一样，采用动态标签技术为它分配标签。这里假设有一定的训练样本，分类器已经可以大致分清数据类型，该损失函数要求其将无标签数据也以高概率分到各个类别，将使得分类器借助无标签数据得到更好地分类边界。

4.A.2 无监督学习

在无监督学习的设定中，除了需要引入公式 (4-33) 的损失函数以使得每个样本被大概率地分为某个类外，我们还需要引入一个额外的损失函数来整体指导无标签数据利用各个类别：

$$L_D^{\text{unl}} = H(p_{\text{ref}}, R(\mathbb{E}_{x \sim p_{\text{unl}}} [D(x)])), \quad (4-34)$$

这里 p_{ref} 是一个参考的标签分布，它将指导模型如何去利用给定的这些类别。例如， p_{ref} 可以是一个均一分布，那么模型将尽可能平均地使用各个类别。

这项损失函数也可以选择性地加入到半监督学习中，可能的情绪包括：标签数据是不均衡的，而我们已知数据的整体类别分布。那么此时，可以将 p_{ref} 可以定义为已知的训练数据标签的分布。

4.B 网络结构与超参数

生成器:

| 算子 | 卷积核 | 卷及补偿 | 输出维度 | Dropout 概率 | 激活函数 | 批归一化 |
|------|-----|------|-----------|------------|-------------|------|
| 噪声 | N/A | N/A | 100 110 | 0.0 | N(0.0, 1.0) | |
| 线性层 | N/A | N/A | 4×4×768 | 0.0 | Leaky ReLU | 是 |
| 反卷积层 | 3×3 | 2×2 | 8×8×384 | 0.0 | Leaky ReLU | 是 |
| 反卷积层 | 3×3 | 2×2 | 16×16×192 | 0.0 | Leaky ReLU | 是 |
| 反卷积层 | 3×3 | 2×2 | 32×32×96 | 0.0 | Leaky ReLU | 是 |
| 反卷积层 | 3×3 | 1×1 | 32×32×3 | 0.0 | Tanh | |

判别器:

| 算子 | 卷积核 | 卷积步长 | 输出维度 | Dropout 概率 | 激活函数 | 批归一化 |
|-------|-----|------|--------------|------------|-------------|------|
| 加高斯噪声 | N/A | N/A | 32×32×3 | 0.0 | N(0.0, 0.1) | |
| 卷积层 | 3×3 | 1×1 | 32×32×64 | 0.3 | Leaky ReLU | 是 |
| 卷积层 | 3×3 | 2×2 | 16×16×128 | 0.3 | Leaky ReLU | 是 |
| 卷积层 | 3×3 | 2×2 | 8×8×256 | 0.3 | Leaky ReLU | 是 |
| 卷积层 | 3×3 | 2×2 | 4×4×512 | 0.3 | Leaky ReLU | 是 |
| 卷积层 * | 3×3 | 1×1 | 4×4×512 | 0.3 | Leaky ReLU | 是 |
| 平均池化 | N/A | N/A | 1×1×512 | 0.3 | N/A | |
| 线性层 | N/A | N/A | 10 11 12 | 0.0 | Softmax | |

带 * 的层仅仅在预定义标签的实验中才存在。

优化器: Adam, 其中 beta1=0.5, beta2=0.999。批量大小为 100。

学习率: 初始学习率 0.0004, 采用阶梯式指数递减。

我们对每一个层的参数都使用了参数归一化 (weight normalization)^[97]。

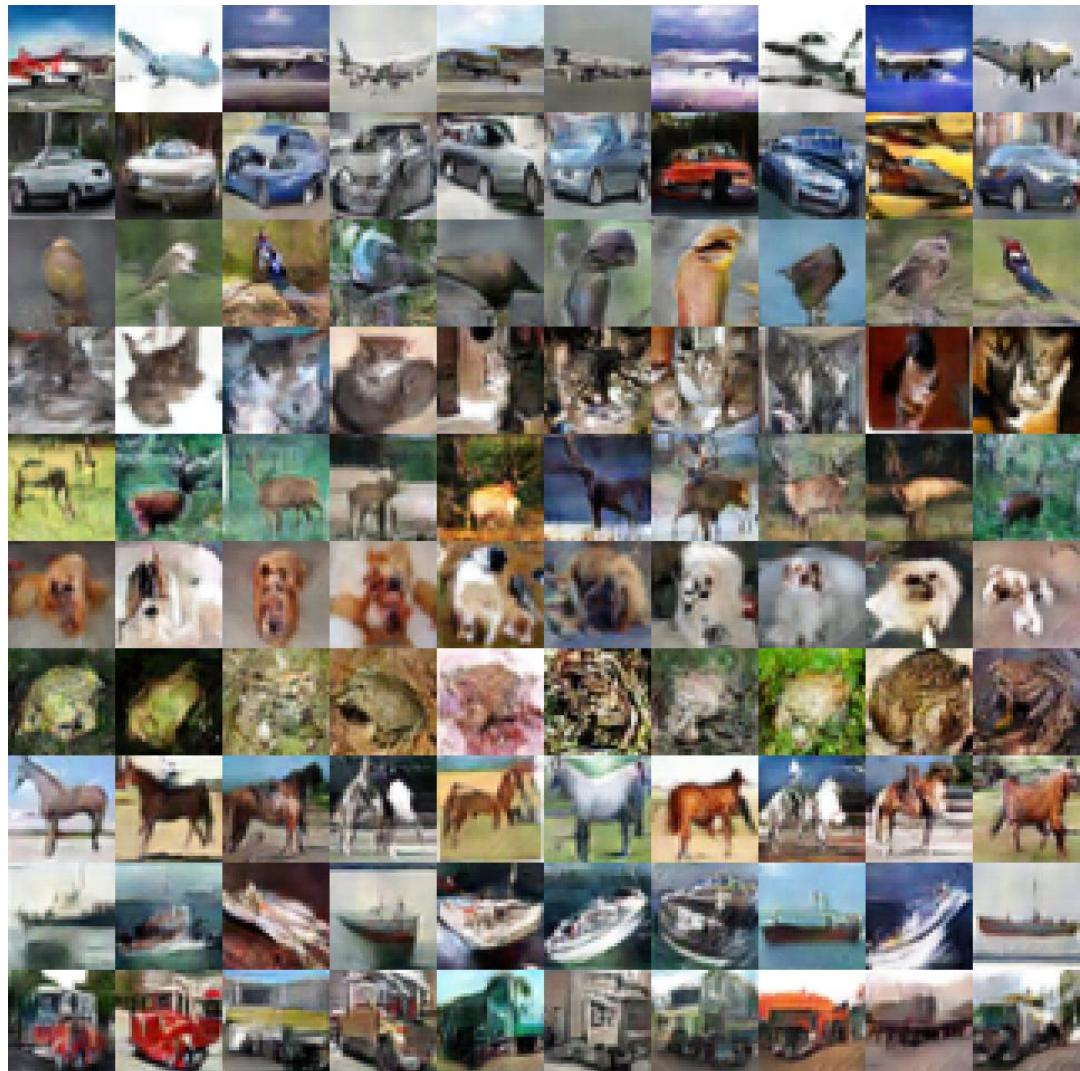


图 4-9 随机采样的样本示例：采用动态标签的 AM-GAN。

Figure 4-9 Random samples of AM-GAN with dynamic labeling.



图 4-10 随机采样的样本示例：采用动态标签的 AM-GAN。

Figure 4-10 Random samples of AM-GAN with predefined label.



图 4-11 随机采样的样本示例：采用动态标签的 AC-GAN。

Figure 4-11 Random samples of AC-GAN with dynamic labeling.



图 4-12 随机采样的样本示例：采用预定义标签的 AC-GAN。

Figure 4-12 Random samples of AC-GAN with predefined label.

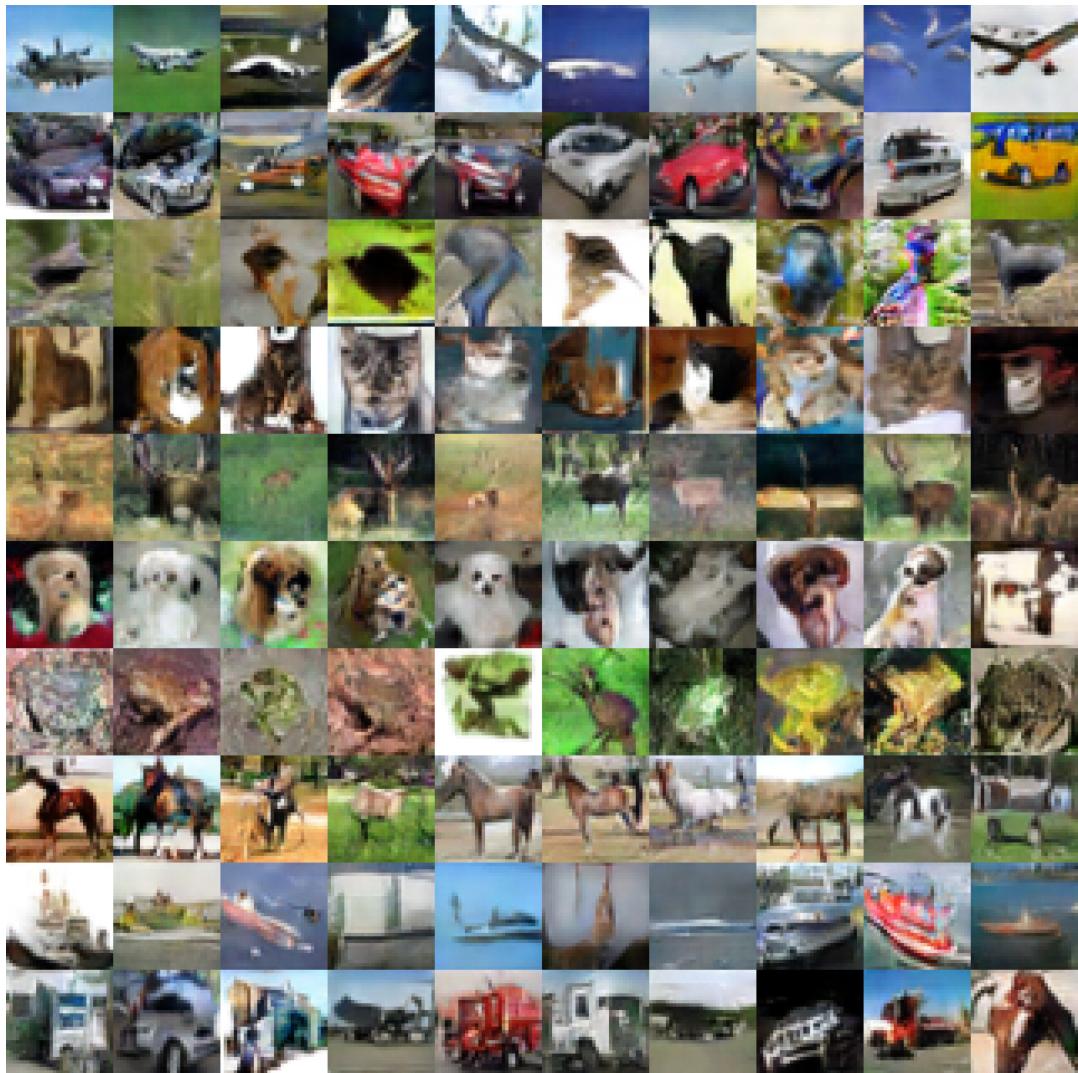


图 4-13 随机采样的样本示例：采用动态标签的 AC-GAN⁺。

Figure 4-13 Random samples of AC-GAN⁺ with dynamic labeling.



图 4-14 随机采样的样本示例：采用预定义标签的 AC-GAN⁺。

Figure 4-14 Random samples of AC-GAN⁺ with predefined label.

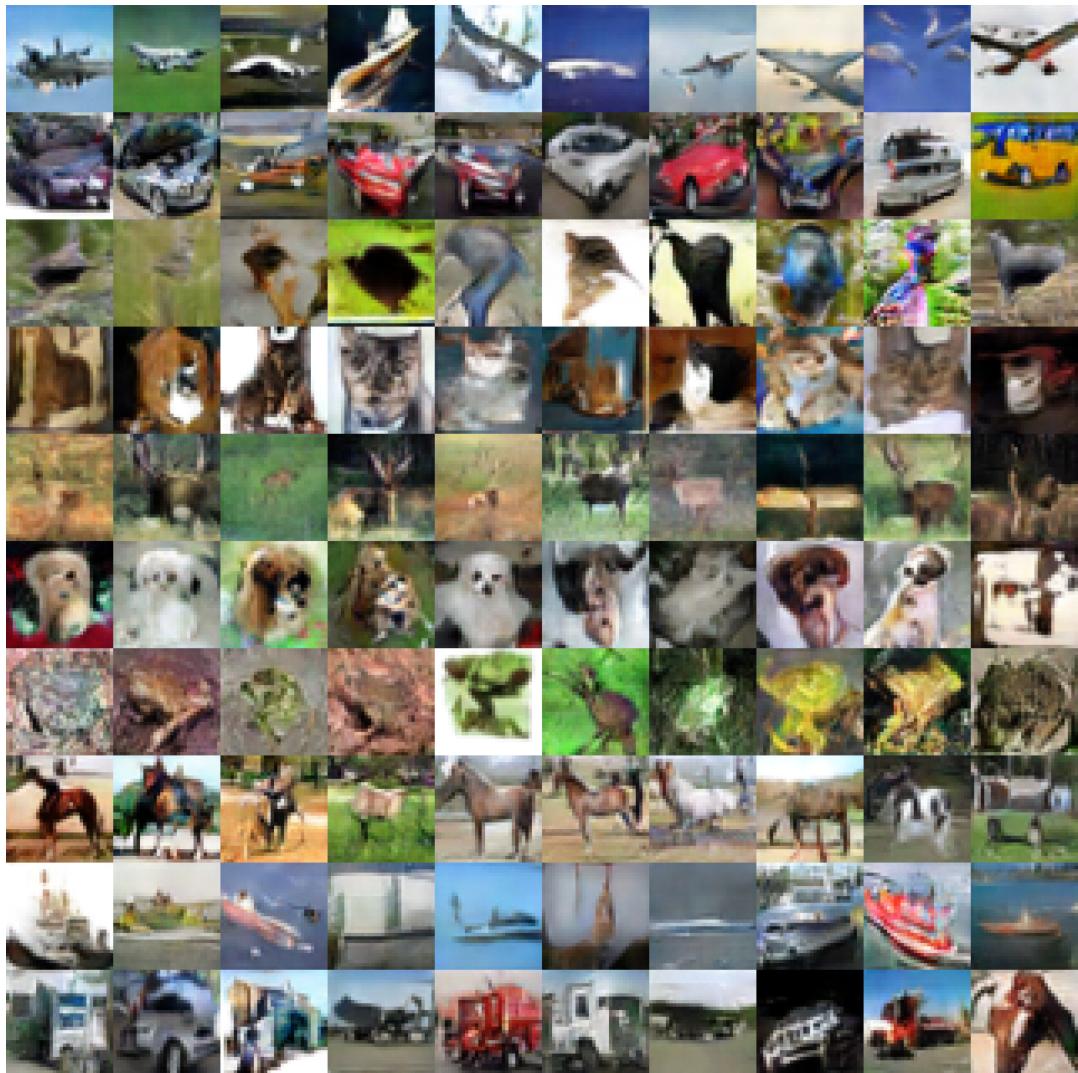


图 4-15 随机采样的样本示例：在动态标签参数设定下的 LabelGAN。

Figure 4-15 Random samples of LabelGAN under the parameter setting of dynamic labeling.



图 4-16 随机采样的样本示例：在预定义标签参数设定下的 LabelGAN。

Figure 4-16 Random samples of LabelGAN under the parameter setting of predefined label.

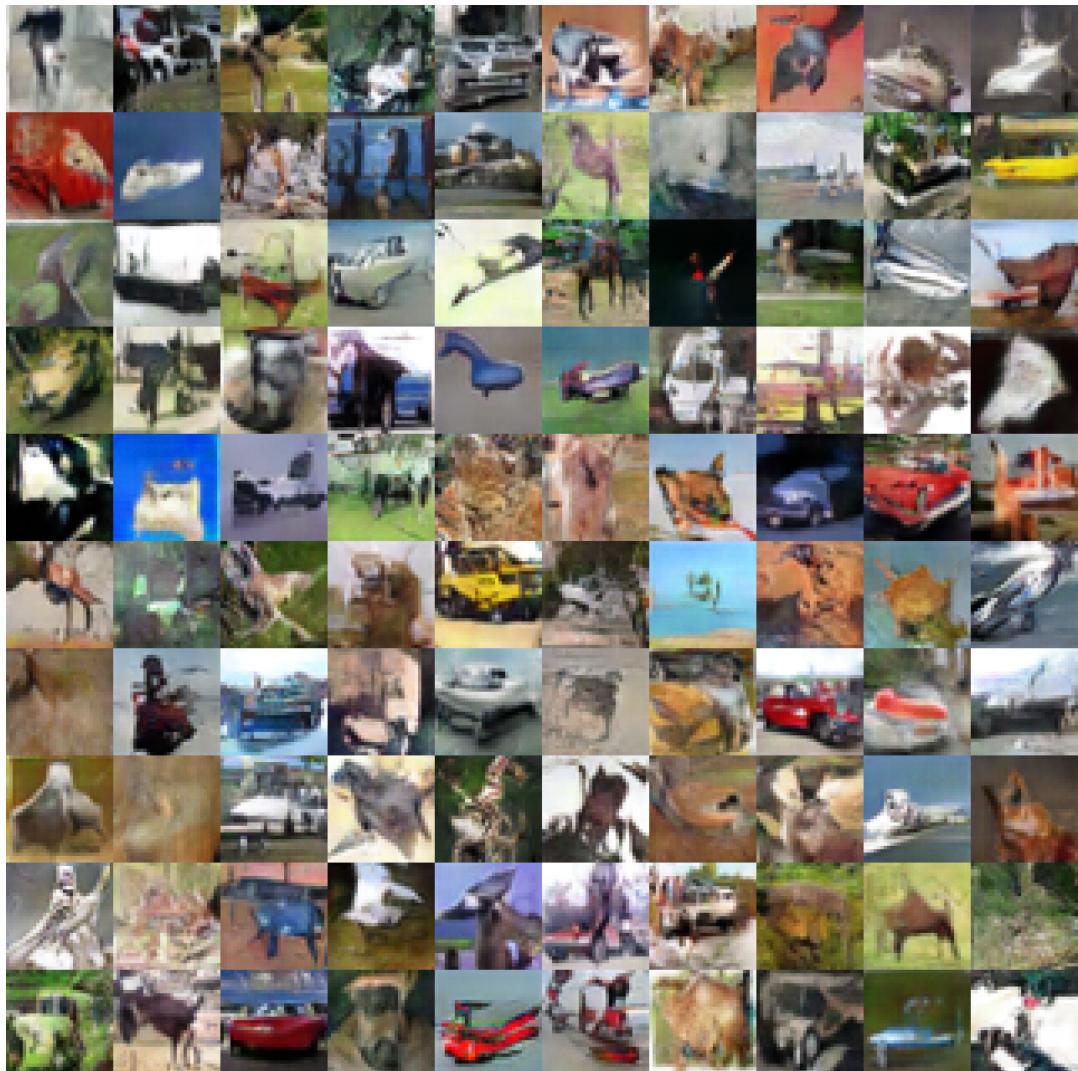


图 4-17 随机采样的样本示例：在动态标签参数设定下的 GAN。

Figure 4-17 Random samples of GAN under the parameter setting of dynamic labeling.

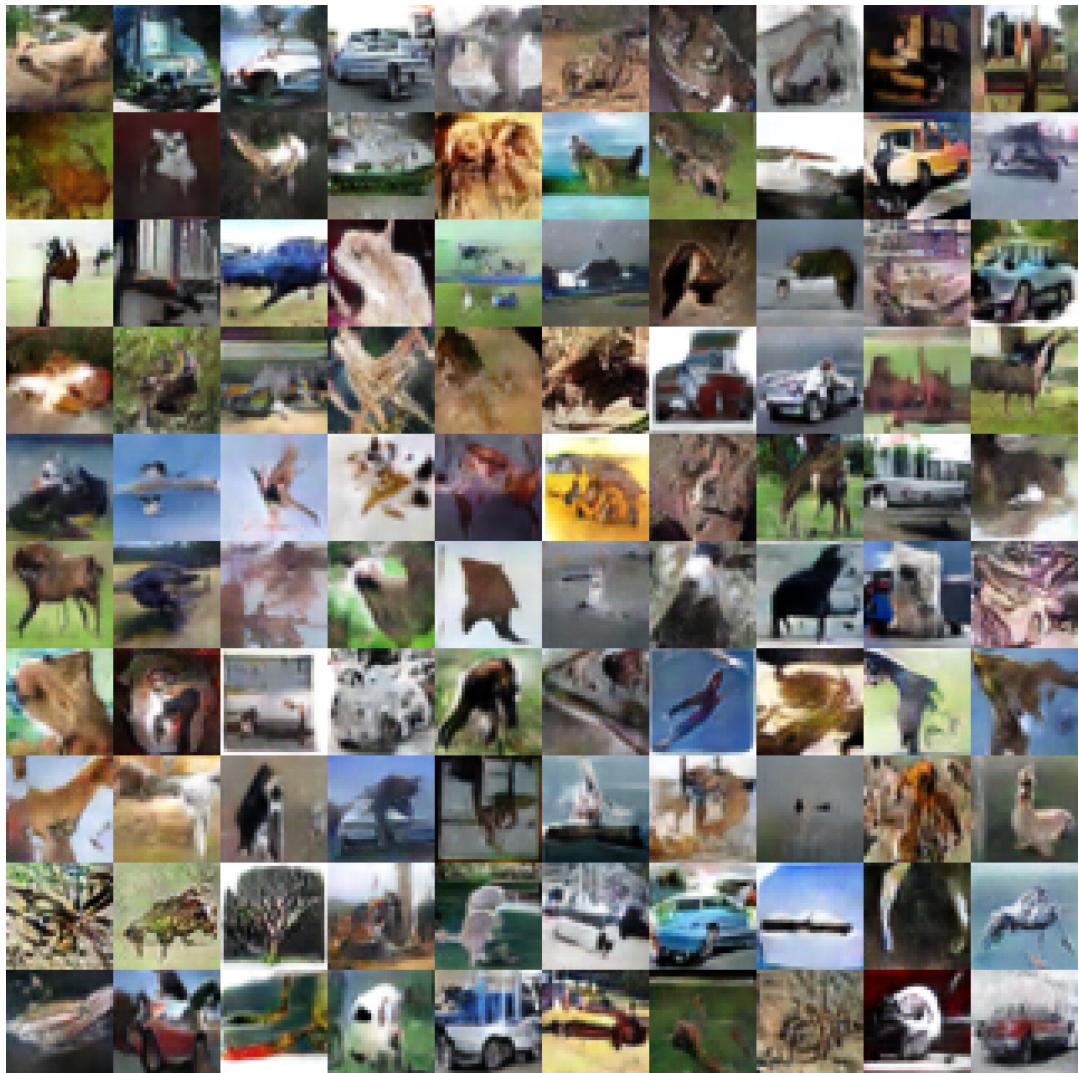


图 4-18 随机采样的样本示例：在预定义标签参数设定下的 GAN。

Figure 4-18 Random samples of GAN under the parameter setting of predefined label.

全文总结

在本文中，我们首先从最优判别函数的角度研究了生成对抗网络的收敛性问题，它很好地解释了生成对抗网络的训练不稳定性问题，并提供了一族新的保证收敛性的生成对抗网络模型，同时为生成对抗网络的收敛性问题和训练稳定性问题的研究提供了新的行之有效的思路。对于最优判别函数的深入研究，使我们意识到在实践过程中生成对抗网络的判别器通常是远远没有达到最优的。同一时期，生成对抗网络优化所依赖的自适应学习率算法被指出存在不收敛性问题。我们因此就自适应学习率算法展开了深入研究，进而指出了当前自适应学习率算法中存在的问题，并相应地提供了修正方案。除了收敛性问题，生成对抗网络的另一个巨大的难题在于它面对复杂数据集时，其样本质量常常不尽如人意。有证据表明标签信息能有效帮助提升生成对抗网络的生成样本质量，但标签信息如何作用于生成对抗网络还不甚明了。因此，我们就标签如何影响生成对抗网络的训练展开了研究，揭示了标签信息如何作用于生成对抗网络；基于此，提出了激活最大化的生成对抗网络模型，以最有效的方式利用标签信息，实现了对生成对抗网络样本质量的有效提升。

文中有许多重要的结论，如：（1）生成对抗网络的判别函数空间必须存在约束，否则其收敛性并将没有任何保障；（2）Lipschitz 约束能使得生产样本的梯度趋于指向真实样本；（3）自适应学习率算法中，二阶矩量项应与当前梯度去相关，否则将导致不收敛情形的存在；（4）自适应学习率算法中，原二阶矩量项其实广义上可以为任何与当前梯度独立的项，均能保证类似于随机梯度下降的收敛性；（5）标签信息的引入让判别器在反传的梯度中包含类别信息，使得生产样本的梯度更新方向更加明确；（6）简单的最大化激活并不能保证样本质量，标签分类器上需引入对抗训练。

生成对抗网络的理论研究还有许多值得探索的方向，比如：（1）生成对抗网络的泛化性：假设训练达到理想效果，生成对抗网络是否能够泛化，泛化的性质取决于哪些因素；（2）生成对抗网络的隐空间：一般将生成对抗网络的隐空间默认为高斯分布或均匀分布，但这绝不是最优的，所以怎样的隐空间才是最优的，和数据之间有怎样的关联，如何自动发掘最优的隐空间表示是一个很值得研究的课题；（3）生成器的目标函数：生成对抗网络通常定义为 minimax 的性质，minimax 有很好的解释性，但也有证据表明基于 minimax 的生成器目标函数并不是最优的，生成器的目标函数也是一个值得深入研究的课题。

参考文献

- [1] GOODFELLOW I, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial nets[C]//Advances in neural information processing systems. [S.l. : s.n.], 2014: 2672-2680.
- [2] KINGMA D P, WELLING M. Auto-encoding variational bayes[J]. ArXiv preprint arXiv:1312.6114, 2013.
- [3] TAN W R, CHAN C S, AGUIRRE H E, et al. ArtGAN: Artwork synthesis with conditional categorical GANs[C]//2017 IEEE International Conference on Image Processing (ICIP). [S.l. : s.n.], 2017: 3760-3764.
- [4] YU L, ZHANG W, WANG J, et al. SeqGAN: Sequence generative adversarial nets with policy gradient[C]//Thirty-First AAAI Conference on Artificial Intelligence. [S.l. : s.n.], 2017.
- [5] ZHANG H, XU T, LI H, et al. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks[C]//Proceedings of the IEEE International Conference on Computer Vision. [S.l. : s.n.], 2017: 5907-5915.
- [6] ZHANG L, JI Y, LIN X, et al. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier GAN[C]//2017 4th IAPR Asian Conference on Pattern Recognition (ACPR). [S.l. : s.n.], 2017: 506-511.
- [7] ISOLA P, ZHU J Y, ZHOU T, et al. Image-to-image translation with conditional adversarial networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l. : s.n.], 2017: 1125-1134.
- [8] KARRAS T, AILA T, LAINE S, et al. Progressive growing of GANs for improved quality, stability, and variation[J]. ArXiv preprint arXiv:1710.10196, 2017.
- [9] RADFORD A, METZ L, CHINTALA S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. ArXiv preprint arXiv:1511.06434, 2015.
- [10] ZHU J Y, KRÄHENBÜHL P, SHECHTMAN E, et al. Generative visual manipulation on the natural image manifold[C]//European Conference on Computer Vision. [S.l. : s.n.], 2016: 597-613.

- [11] BROCK A, LIM T, RITCHIE J M, et al. Neural photo editing with introspective adversarial networks[J]. ArXiv preprint arXiv:1609.07093, 2016.
- [12] SPRINGENBERG J T. Unsupervised and semi-supervised learning with categorical generative adversarial networks[J]. ArXiv preprint arXiv:1511.06390, 2015.
- [13] CHEN X, DUAN Y, HOUTHOOFT R, et al. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets[C]// Advances in neural information processing systems. [S.l. : s.n.], 2016: 2172-2180.
- [14] CAO Y, ZHOU Z, ZHANG W, et al. Unsupervised diverse colorization via generative adversarial networks[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. [S.l. : s.n.], 2017: 151-166.
- [15] SALIMANS T, GOODFELLOW I, ZAREMBA W, et al. Improved techniques for training GANs[C]//Advances in neural information processing systems. [S.l. : s.n.], 2016: 2234-2242.
- [16] KUMAR A, SATTIGERI P, FLETCHER P T. Improved semi-supervised learning with GANs using manifold invariances[J]. ArXiv preprint arXiv:1705.08850, 2017.
- [17] GOODFELLOW I. NIPS 2016 tutorial: Generative adversarial networks[J]. ArXiv preprint arXiv:1701.00160, 2016.
- [18] DENTON E L, CHINTALA S, FERGUS R, et al. Deep generative image models using a Laplacian pyramid of adversarial networks[C]//Advances in neural information processing systems. [S.l. : s.n.], 2015: 1486-1494.
- [19] HUANG X, LI Y, POURSAEED O, et al. Stacked generative adversarial networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l. : s.n.], 2017: 5077-5086.
- [20] MAO X, LI Q, XIE H, et al. Least squares generative adversarial networks[C]// Proceedings of the IEEE International Conference on Computer Vision. [S.l. : s.n.], 2017: 2794-2802.
- [21] ZHAO J, MATHIEU M, LECUN Y. Energy-based generative adversarial network[J]. ArXiv preprint arXiv:1609.03126, 2016.
- [22] DAI Z, ALMAHAIRI A, BACHMAN P, et al. Calibrating energy-based generative adversarial networks[J]. ArXiv preprint arXiv:1702.01691, 2017.

- [23] NOWOZIN S, CSEKE B, TOMIOKA R. F-GAN: Training generative neural samplers using variational divergence minimization[C]//Advances in neural information processing systems. [S.l. : s.n.], 2016: 271-279.
- [24] LIM J H, YE J C. Geometric GAN[J]. ArXiv preprint arXiv:1705.02894, 2017.
- [25] MROUEH Y, SERCU T, GOEL V. McGAN: Mean and covariance feature matching GAN[J]. ArXiv preprint arXiv:1702.08398, 2017.
- [26] MROUEH Y, LI C L, SERCU T, et al. Sobolev GAN[J]. ArXiv preprint arXiv:1711.04894, 2017.
- [27] MROUEH Y, SERCU T. Fisher GAN[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 2513-2523.
- [28] LI C L, CHANG W C, CHENG Y, et al. MMD GAN: Towards deeper understanding of moment matching network[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 2203-2213.
- [29] QI G J. Loss-sensitive generative adversarial networks on Lipschitz densities[J]. ArXiv preprint arXiv:1701.06264, 2017.
- [30] ARJOVSKY M, CHINTALA S, BOTTOU L. Wasserstein GAN[J]. ArXiv preprint arXiv:1701.07875, 2017.
- [31] BELLEMARE M G, DANIHELKA I, DABNEY W, et al. The Cramer distance as a solution to biased Wasserstein gradients[J]. ArXiv preprint arXiv:1705.10743, 2017.
- [32] LI C, ALVAREZ-MELIS D, XU K, et al. Distributional adversarial networks[J]. ArXiv preprint arXiv:1706.09549, 2017.
- [33] JOLICOEUR-MARTINEAU A. The relativistic discriminator: a key element missing from standard GAN[C/OL]//International Conference on Learning Representations. [S.l. : s.n.], 2019. <https://openreview.net/forum?id=S1erHoR5t7>.
- [34] ARJOVSKY M, BOTTOU L. Towards principled methods for training generative adversarial networks. arXiv, 2017[J]. ArXiv preprint arXiv:1701.04862,
- [35] GULRAJANI I, AHMED F, ARJOVSKY M, et al. Improved training of Wasserstein GANs[C]//Advances in neural information processing systems. [S.l. : s.n.], 2017: 5767-5777.

-
- [36] MIYATO T, KATAOKA T, KOYAMA M, et al. Spectral normalization for generative adversarial networks[J]. ArXiv preprint arXiv:1802.05957, 2018.
 - [37] PETZKA H, FISCHER A, LUKOVNICOV D. On the regularization of Wasserstein GANs[J]. ArXiv preprint arXiv:1709.08894, 2017.
 - [38] KROGH A, HERTZ J A. A simple weight decay can improve generalization[C]// Advances in neural information processing systems. [S.l. : s.n.], 1992: 950-957.
 - [39] LOSHCHILOV I, HUTTER F. Fixing weight decay regularization in adam[J]., 2018.
 - [40] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. The journal of machine learning research, 2014, 15(1): 1929-1958.
 - [41] YOSHIDA Y, MIYATO T. Spectral norm regularization for improving the generalizability of deep learning[J]. ArXiv preprint arXiv:1705.10941, 2017.
 - [42] OBERMAN A M, CALDER J. Lipschitz regularized deep neural networks converge and generalize[J]. ArXiv preprint arXiv:1808.09540, 2018.
 - [43] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. ArXiv preprint arXiv:1502.03167, 2015.
 - [44] ROTH K, LUCCHI A, NOWOZIN S, et al. Stabilizing training of generative adversarial networks through regularization[C]// Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 2018-2028.
 - [45] KODALI N, ABERNETHY J, HAYS J, et al. On convergence and stability of GANs[J]. ArXiv preprint arXiv:1705.07215, 2017.
 - [46] XIANG S, LI H. On the effects of batch and weight normalization in generative adversarial networks[J]. ArXiv preprint arXiv:1704.03971, 2017.
 - [47] JENNI S, FAVARO P. On Stabilizing Generative Adversarial Training with Noise[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l. : s.n.], 2019: 12145-12153.
 - [48] KURACH K, LUCIC M, ZHAI X, et al. A large-scale study on regularization and normalization in GANs[J]. ArXiv preprint arXiv:1807.04720, 2018.

-
- [49] HEUSEL M, RAMSAUER H, UNTERTHINER T, et al. GANs trained by a two time-scale update rule converge to a local nash equilibrium[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 6626-6637.
 - [50] METZ L, POOLE B, PFAU D, et al. Unrolled generative adversarial networks[J]. ArXiv preprint arXiv:1611.02163,
 - [51] MESCHEDER L, NOWOZIN S, GEIGER A. The numerics of GANs[C]// Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 1825-1835.
 - [52] MESCHEDER L, GEIGER A, NOWOZIN S. Which training methods for GANs do actually converge?[C]//International Conference on Machine Learning. [S.l. : s.n.], 2018: 3478-3487.
 - [53] DASKALAKIS C, ILYAS A, SYRGKANIS V, et al. Training gans with optimism[J]. ArXiv preprint arXiv:1711.00141, 2017.
 - [54] MERTIKOPOULOS P, LECOUAT B, ZENATI H, et al. Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile[J]. ArXiv preprint arXiv:1807.02629, 2018.
 - [55] LARSEN A B L, SØNDERBY S K, LAROCHELLE H, et al. Autoencoding beyond pixels using a learned similarity metric[J]. ArXiv preprint arXiv:1512.09300, 2015.
 - [56] MAKHZANI A, SHLENS J, JAITLEY N, et al. Adversarial autoencoders[J]. ArXiv preprint arXiv:1511.05644, 2015.
 - [57] DONAHUE J, KRÄHENBÜHL P, DARRELL T. Adversarial feature learning[J]. ArXiv preprint arXiv:1605.09782, 2016.
 - [58] DUMOULIN V, BELGHAZI I, POOLE B, et al. Adversarially learned inference[J]. ArXiv preprint arXiv:1606.00704, 2016.
 - [59] CHE T, LI Y, JACOB A P, et al. Mode regularized generative adversarial networks[J]. ArXiv preprint arXiv:1612.02136, 2016.
 - [60] ULYANOV D, VEDALDI A, LEMPITSKY V. It takes (only) two: Adversarial generator-encoder networks[C]//Thirty-Second AAAI Conference on Artificial Intelligence. [S.l. : s.n.], 2018.
 - [61] TOLSTIKHIN I, BOUSQUET O, GELLY S, et al. Wasserstein auto-encoders[J]. ArXiv preprint arXiv:1711.01558, 2017.

-
- [62] BROCK A, DONAHUE J, SIMONYAN K. Large scale gan training for high fidelity natural image synthesis[J]. ArXiv preprint arXiv:1809.11096, 2018.
 - [63] ODENA A, OLAH C, SHLENS J. Conditional image synthesis with auxiliary classifier GANs[C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. [S.l. : s.n.], 2017: 2642-2651.
 - [64] LUCIC M, KURACH K, MICHALSKI M, et al. Are GANs created equal? A large-scale study[C]//Advances in neural information processing systems. [S.l. : s.n.], 2018: 700-709.
 - [65] YADAV A, SHAH S, XU Z, et al. Stabilizing adversarial nets with prediction methods[J]. ArXiv preprint arXiv:1705.07364, 2017.
 - [66] VILLANI C. Optimal transport: old and new[M]. [S.l.]: Springer Science & Business Media, 2008.
 - [67] FEDUS W, ROSCA M, LAKSHMINARAYANAN B, et al. Many paths to equilibrium: GANs do not need to decrease a divergence at every step[J]. ArXiv preprint arXiv:1710.08446, 2017.
 - [68] ADLER J, LUNZ S. Banach Wasserstein GAN[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2018: 6754-6763.
 - [69] SEGUY V, DAMODARAN B B, FLAMARY R, et al. Large-scale optimal transport and mapping estimation[J]. ArXiv preprint arXiv:1711.02283, 2017.
 - [70] FARNIA F, TSE D. A convex duality framework for GANs[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2018: 5248-5258.
 - [71] UNTERTHINER T, NESSLER B, SEWARD C, et al. Coulomb GANs: Provably optimal Nash equilibria via potential fields[J]. International Conference on Learning Representations, 2018.
 - [72] ARORA S, GE R, LIANG Y, et al. Generalization and equilibrium in generative adversarial nets (GANs)[C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. [S.l. : s.n.], 2017: 224-232.
 - [73] LIU S, BOUSQUET O, CHAUDHURI K. Approximation and convergence properties of generative adversarial learning[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 5545-5553.

- [74] ZHANG H, GOODFELLOW I, METAXAS D, et al. Self-attention generative adversarial networks[J]. ArXiv preprint arXiv:1805.08318, 2018.
- [75] ODENA A, BUCKMAN J, OLSSON C, et al. Is generator conditioning causally related to GAN performance?[J]. ArXiv preprint arXiv:1802.08768, 2018.
- [76] SANJABI M, BA J, RAZAVIYAYN M, et al. Solving approximate Wasserstein GANs to stationarity[J]. ArXiv preprint arXiv:1802.08249, 2018.
- [77] REDDI S J, KALE S, KUMAR S. On the Convergence of Adam and Beyond[C/OL]//International Conference on Learning Representations. [S.l. : s.n.], 2018. <https://openreview.net/forum?id=ryQu7f-RZ>.
- [78] QIAN N. On the momentum term in gradient descent learning algorithms[J]. Neural networks, 1999, 12(1): 145-151.
- [79] KINGMA D P, BA J. Adam: A method for stochastic optimization[J]. ArXiv preprint arXiv:1412.6980, 2014.
- [80] ZEILER M D. ADADELTA: an adaptive learning rate method[J]. ArXiv preprint arXiv:1212.5701, 2012.
- [81] TIELEMANT, HINTON G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude[J]. COURSERA: Neural networks for machine learning, 2012, 4(2): 26-31.
- [82] DOZAT T. Incorporating nesterov momentum into adam[J]. International Conference on Learning Representations, Workshop track, 2016.
- [83] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks[C]//Proceedings of the thirteenth international conference on artificial intelligence and statistics. [S.l. : s.n.], 2010: 249-256.
- [84] HE K, ZHANG X, REN S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]//Proceedings of the IEEE international conference on computer vision. [S.l. : s.n.], 2015: 1026-1034.
- [85] WILSON A C, ROELOFS R, STERN M, et al. The marginal value of adaptive gradient methods in machine learning[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2017: 4148-4158.
- [86] KESKAR N S, SOCHER R. Improving generalization performance by switching from adam to sgd[J]. ArXiv preprint arXiv:1712.07628, 2017.

- [87] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l. : s.n.], 2016: 770-778.
- [88] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l. : s.n.], 2017: 4700-4708.
- [89] LUONG M, BREVDO E, ZHAO R. Neural Machine Translation (seq2seq) Tutorial[J]. [Https://github.com/tensorflow/nmt](https://github.com/tensorflow/nmt), 2017.
- [90] NGUYEN A, CLUNE J, BENGIO Y, et al. Plug & play generative networks: Conditional iterative generation of images in latent space[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l. : s.n.], 2017: 4467-4477.
- [91] ERHAN D, BENGIO Y, COURVILLE A, et al. Visualizing higher-layer features of a deep network[J]., 2009.
- [92] NGUYEN A, DOSOVITSKIY A, YOSINSKI J, et al. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2016: 3387-3395.
- [93] ZEMEL Y. Optimal transportation: continuous and discrete[D]. Master' s thesis, École polytechnique fédérale de Lausanne, 2012.
- [94] WANG Z, SIMONCELLI E P, BOVIK A C. Multiscale structural similarity for image quality assessment[C]//The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003:vol. 2. [S.l. : s.n.], 2003: 1398-1402.
- [95] GRINBLAT G L, UZAL L C, GRANITTO P M. Class-splitting generative adversarial networks[J]. ArXiv preprint arXiv:1709.07359, 2017.
- [96] WARDE-FARLEY D, BENGIO Y. Improving generative adversarial networks with denoising feature matching.(2017)[C]//International Conference on Learning Representations. [S.l. : s.n.], 2017.
- [97] SALIMANS T, KINGMA D P. Weight normalization: A simple reparameterization to accelerate training of deep neural networks[C]//Advances in Neural Information Processing Systems. [S.l. : s.n.], 2016: 901-909.

致 谢

时光荏苒，岁月如梭，转眼间在交大度过了将近十年的时光。往事历历在目，这十年见证了我的成长，见证了十年来的喜怒哀乐。在此期间，遇到了许许多多珍贵的老师、同学和朋友。是你们的陪伴、关心和照顾，让我的这十年，纵使有些崎岖坎坷，却又充实而美好。

在此，我要特别感谢我的导师俞勇教授。感谢您让我有机会进入 ACM 班这么美好的集体，感谢您在我困难时候的指导、教育和不放弃，感谢您十年来的无微不至的关心和照顾，感谢您总能在我感到困惑的时候给我方向和坚定地信念。

我也要特别感谢微软亚洲研究院的童欣老师。感谢您带我走进计算机图形学的世界，感谢您带我打开学术的大门，感谢您那严苛和严谨的治学风格深深地感染了我，感谢您在我学术遇到困难时对我的鼓励和肯定。此外，也要重重地感谢微软亚洲研究院的董悦和陈国军，感谢你们在我做第一篇论文时，给我的极其有耐心和细致的指导和帮助，感谢你们让我的这段实习时光充实而愉快。

我也要特别感谢实验室的张伟楠老师。感谢您在我转方向初期给我的有效沟通和指导，感谢您在我亲自执笔的第一篇论文上的给我的细心而细致的指导，感谢您一直以来对我学业和生活的关心和照顾。同时也要感谢汪军老师，感谢您紧急应援我的第一篇论文，让它从非常草的草稿变成了真正的论文，并成功投了出去还差点中了；感谢您对我后续论文的诸多的指导。

我还要特别感谢北大的张志华老师。感谢您在我论文遇到困难的时候帮我答疑解惑帮我重新梳理思路，感谢您对我学习和生活等诸多方面的关心和照顾，感谢您对我申请的鼎力支持。同时也要感谢北大的梁家栋同学，感谢你帮我梳理论文的证明。也要感谢林大超同学，感谢你给我的诸多非常有意义的讨论。

感谢实验室的同学们。感谢任侃、曹雪智、邱霖、朱晨浩、黄伟岳、蔡涵、曲彦儒、沈键等同学，感谢你们带我吃喝玩乐，一起度过美好的实验室时光。感谢我的组员们，宋宇轩、卢冠松、张庆儒、徐民凯、吴怡琳、曹贊、吴力胜、徐润泽、黄汝林、陈亦言等等，感谢你们这些小可爱和我一起度过的学术和闲聊的时光，这一切都让我获益匪浅，也是我生活的重要组成部分。

感谢在微软亚洲研究院遇到的小伙伴们。感谢饭团的肖文聪、周青宇、孟梦、星辰、谭传奇、贾伟、陈伟杰、肖雨佳、贺同、陈凯、吴宏、罗仪等同学，以及李俊杰、孙鑫等 foosball 玩家，陪我吃喝玩乐。感谢组里的刘一龙、夏睿、李潇、王鹏帅、任沛然、李琛、李昌健等同学，以及刘洋老师，在组内给我的帮助和照顾。

感谢交大轮滑协会和北大轮滑社的伙伴们。感谢讴哥、盗图、小蒙、实沉、三桂、镇韬、萍萍、洗脚、cookie、潇等，带我轮滑带我飞。感谢夫人、橙子、点子、宇航员等带我在 CC 畅玩。感谢我的舍友杨欢、黎彧君、王鸿伟，学校里最温暖和亲切的老朋友。感谢刘晓笑和黄丽明，感谢你们曾经陪在我身边，让这些年的时光更加的丰富多彩。

最后，我要感谢我的父母，感谢你们对我的无条件的支持，感谢你们无时无刻不在的默默地关心和关注。感谢我的哥哥，感谢你在我远道求学的期间，一直照顾着父母和处理家中的大小事务。

攻读博士学位期间已发表或录用的论文

- [1] Zhou Z, Liang J, Song Y, Yu L, Wang H, Zhang W, Yu Y, Zhang Z. Lipschitz Generative Adversarial Nets. International Conference on Machine Learning (ICML). 2019 May 24 (pp. 7584-7593).
- [2] Zhou Z, Chen G, Dong Y, Wipf D, Yu Y, Snyder J, Tong X. Sparse-as-Possible SVBRDF acquisition. ACM Transactions on Graphics (TOG). 2016 Nov 11;35(6):189.
- [3] Zhou Z, Zhang Q, Lu G, Wang H, Zhang W, Yu Y. AdaShift: Decorrelation and Convergence of Adaptive Learning Rate Methods. The Seventh International Conference on Learning Representations (ICLR). 2019.
- [4] Zhou Z, Cai H, Rong S, Song Y, Ren K, Zhang W, Wang J, Yu Y. Activation Maximization Generative Adversarial Nets. The Sixth International Conference on Learning Representations (ICLR). 2018.
- [5] Lu G, Zhou Z, Song Y, Ren K, Yu Y. Guiding the One-to-One Mapping in CycleGAN via Optimal Transport. Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). 2019 Jul 17 (Vol. 33, pp. 4432-4439).
- [6] Cao Y, Zhou Z, Zhang W, Yu Y. Unsupervised Diverse Colorization via Generative Adversarial Networks. Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML). 2017 Sep 18 (pp. 151-166).
- [7] Zhu Y, Wan J, Zhou Z, Chen L, Qiu L, Zhang W, Jiang X, Yu Y. Triple-to-Text: Converting RDF Triples into High-Quality Natural Languages via Optimizing an Inverse KL Divergence. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). July 2019. Pages 455–464.
- [8] Zhang H, Wang J, Zhou Z, Zhang W, Wen Y, Yu Y, Li W. Learning to Design Games: Strategic Environments in Reinforcement Learning. Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI). 2018 Jul 13 (pp. 3068-3074).

- [9] Song Y, Yu L, Cao Z, Zhou Z, Shen J, Shao S, Zhang W, Yu Y. Improving Unsupervised Domain Adaptation with Variational Information Bottleneck. The 24th European Conference on Artificial Intelligence(ECAI). 2019.