

基于分段损失的生成对抗网络

刘其开¹, 姜代红², 李文吉³

(1. 中国矿业大学 信息与控制工程学院, 江苏 徐州 221116; 2. 徐州工程学院 信电工程学院, 江苏 徐州 221111;
3. 中国国土资源航空物探遥感中心 国土资源部航空地球物理与遥感地质重点实验室, 北京 100083)

摘要: 生成对抗网络 (GAN) 在训练过程中未能有效进行生成器与鉴别器间的同步更新, 导致模型训练不稳定并出现模式崩溃的现象。为此, 提出一种基于分段损失的生成对抗网络 PL-GAN。生成器在不同的训练时期采用不同形式的损失函数, 同时引入真实样本与生成样本之间的特征级损失, 从而使鉴别器提取的特征更具有鲁棒性。MNIST 和 CIFAR-10 数据集上的实验结果表明, 与 regular GAN、feature-wise GAN 相比, PL-GAN 具有更高的分类精度与运行效率。

关键词: 生成对抗网络; 模式崩溃; 特征级损失; 分段损失; 半监督学习

中文引用格式: 刘其开, 姜代红, 李文吉. 基于分段损失的生成对抗网络[J]. 计算机工程, 2019, 45(5): 155-160, 168.

英文引用格式: LIU Qikai, JIANG Daihong, LI Wenji. Generative adversarial network based on piecewise loss[J]. Computer Engineering, 2019, 45(5): 155-160, 168.

Generative Adversarial Network Based on Piecewise Loss

LIU Qikai¹, JIANG Daihong², LI Wenji³

(1. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China;
2. School of Information and Electrical Engineering, Xuzhou University of Technology, Xuzhou, Jiangsu 221111, China;
3. Key Laboratory of Airborne Geophysics and Remote Sensing Geology, Ministry of Land and Resources, China Aerospace Geophysical Survey and Remote Sensing Center for Land and Resources, Beijing 100083, China)

[Abstract] Generative Adversarial Network (GAN) fails to effectively execute the synchronous update between generator and discriminator during training, resulting in unstable model training and mode collapse. To solve this problem, a generative adversarial network PL-GAN based on piecewise loss is proposed. The generator uses different loss functions in different training periods, and introduces the feature-wise loss between the real sample and the generated sample, which makes the feature extracted by the discriminator more robust. Experimental results on MNIST and CIFAR-10 datasets show that PL-GAN has higher classification accuracy and operation efficiency than regular GAN and feature-wise GAN.

[Key words] Generative Adversarial Network (GAN); mode collapse; feature-wise loss; piecewise loss; Semi-Supervised Learning (SSL)

DOI: 10.19678/j.issn.1000-3428.0050529

0 概述

根据训练样本有无标签, 机器学习算法可分为带标签的监督学习和无标签的无监督学习。在实际应用中, 监督学习的标签数据获取成本较高, 无监督学习的性能较差, 因此, 半监督学习 (Semi-Supervised Learning, SSL)^[1] 成为人们广泛关注的研究热点之一。SSL 利用海量无标签样本和少量带标签样本学习具有鲁棒性的特征, 在图像分类方面具

有较好的性能表现。文献[2]提出一种对无标签数据进行伪标签的方法, 以进行模型训练。文献[3]构建基于自动编码器的阶梯网络, 其中, 编码器用于监督学习, 解码器的每一层与编码器一一对应形成阶梯, 用于无监督学习训练。

近年来, 生成对抗网络 (Generative Adversarial Network, GAN)^[4] 与深度生成模型在 SSL 中得到应用。文献[5]提出灵活的深度生成模型算法, 其中包括隐含特征判别模型 M1 和生成半监督模型 M2, 前

基金项目: 国家自然科学基金 (51574232); 国土资源部航空地球物理与遥感地质重点实验室航空青年创新基金 (2016YFL02); 徐州市科技计划项目 (KC16SQ78)。

作者简介: 刘其开 (1992—), 男, 硕士, 主研方向为深度学习、计算机视觉; 姜代红, 教授、博士; 李文吉, 工程师、硕士。

收稿日期: 2018-02-27 **修回日期:** 2018-04-13 **E-mail:** liuqikai_0127728@163.com

者学习输入样本在高维空间中的流形,后者结合标签信息推断样本的后验概率 $p_\theta(y|x)$,算法最后将两者结合堆叠为生成半监督模型 M1 + M2。文献[6]建立辅助深度生成网络(Auxiliary Depth Generation Network, ADGN),其通过引入辅助的隐变量来改进变分近似算法。文献[7]构建对抗学习推理模型 ALI,该模型通过引入一个推理器将样本数据空间映射到隐变量空间,输出样本与隐变量的联合分布,相应地,生成器输出隐变量与生成样本的联合分布,鉴别器判断输入的联合分布的来源。文献[8]中的 Cat-GAN 在目标函数中引入权衡输入样本与对应预测类别的互信息,通过最大化生成数据类别的条件交叉熵来训练分类器。文献[9]将鉴别器的二分类概率输出扩展到 $N+1$ 类别概率输出。文献[10]提出的 Triple GAN 通过引入额外的分类器,以解决生成器与鉴别器在训练时无法同时达到最优的问题。文献[11]利用无标签数据对 GAN 的鉴别器进行预训练,对少量有标签数据微调后用于分类任务。

GAN 在训练过程中易出现模型不稳定以及模式崩溃现象,对此,文献[12]从理论上进行了分析。当生成样本分布与真实样本分布之间没有重叠或重叠部分可忽略为 0 时,生成器的损失梯度近似常数。常规 GAN 采用的 JS 散度要求分布之间没有重叠。Wasserstein-GAN^[13]通过设计满足一定约束条件的神经网络逼近函数来近似度量 2 个分布间的距离。文献[14]针对模式崩溃问题,提出一种基于梯度损失的 Unrolled GAN,其通过计算二阶梯度来指示生成器一阶梯度变化的方向。

本文提出一种基于分段损失(Piecewise Loss, PL)的生成对抗网络 PL-GAN。通过设置时间参数使生成器在不同的训练阶段采用不同形式的损失函数。此外,引入生成样本与真实样本在特征空间中的均方差损失,以增强鉴别器所提取特征的鲁棒性。

1 PL-GAN 的设计与实现

GAN 可以理解为 2 个玩家进行相互博弈的游戏,即生成器(G)与鉴别器(D)之间的“零和游戏”。 G 的输入是噪声变量 z ,目的是拟合真实样本的数据分布,输出生成样本 $G(z)$ 。 D 判断输入的样本是来自真实样本还是生成样本,输出输入样本来自真实样本的概率 $D(x)$ 。因此, D 的优化目标是使输入为真实样本的概率尽可能为 1,输入为生成样本的概率尽可能为 0。 G 的优化目标是最大化 D 给出错误判断的概率。算法最终的优化目标是寻找两者之间的一个纳什均衡,该优化过程是一个极大极小的对抗

过程,目标函数 $V(D, G)$ 为:

$$\min_G \max_D V(D, G) = E_{x \sim p_{\text{data}}(x)} [\ln D(x)] + E_{z \sim p_{\text{noise}}(z)} [\ln(1 - D(G(z)))] \quad (1)$$

其中, $p_{\text{data}}(x)$ 表示真实样本的分布, $p_{\text{noise}}(z)$ 表示随机噪声的分布, E 表示期望值。

1.1 算法思想

GAN 的最终目标是通过双玩家游戏策略隐式训练一个逼近真实样本的生成器 $p_g = G_\theta(z)$,其中, D 要尽可能地鉴别出真实样本与生成样本,其损失函数为:

$$D_{\text{loss}} = -E_{x \sim p_{\text{data}}} [\ln D(x)] - E_{z \sim p_{\text{noise}}} [\ln(1 - D(G(z)))] \quad (2)$$

G 的损失函数有以下 2 种形式:

$$G_{\text{loss_first}} = E_{z \sim p_{\text{noise}}} \ln(1 - D(G(z))) = E_{x \sim p_g} \ln(1 - D(x)) \quad (3)$$

$$G_{\text{loss_second}} = -E_{z \sim p_{\text{noise}}} \ln D(G(z)) \quad (4)$$

当 $p_{\text{data}}(x) = p_g(x)$ 时,最优鉴别器表示为:

$$D^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)} = \frac{1}{2} \quad (5)$$

在最优鉴别器下,将式(5)代入式(1)得:

$$\min_G \max_D V(D, G) = E_{x \sim p_g} \ln(1 - D^*(x)) + E_{x \sim p_{\text{data}}} \ln(D^*(x)) \quad (6)$$

将式(6)代入 $D^*(x)$ 后,引入 2 个衡量相似度的指标 KL 散度与 JS 散度,计算如下:

$$KL(P_1 \parallel P_2) = E_{x \sim P_1} \ln \frac{P_1}{P_2}$$

$$JS(P_1 \parallel P_2) = \frac{1}{2} KL(P_1 \parallel \frac{P_1 + P_2}{2}) + \frac{1}{2} KL(P_2 \parallel \frac{P_1 + P_2}{2})$$

在最优鉴别器下, G 的第 1 种损失函数等价于:

$$G_{\text{loss_first} \parallel D^*} = 2JS(p_{\text{data}} \parallel p_g) - 2\ln 2 \quad (7)$$

由式(7)可以看出,在最优鉴别器下,最小化生成器的损失等价于最小化生成样本与真实样本之间的 JS 散度。文献[12]研究表明,在第 1 种损失函数下,用 JS 散度衡量分布差异的前提是 2 个分布间存在不可忽略的重叠,否则 JS 散度将会是一个常数。但网络初始化后的生成样本分布很难与真实样本分布存在不可忽略的重叠。

同样,由式(6)、式(7)可得 G 的第 2 种损失函数为:

$$G_{\text{loss_second}} = KL(p_g \parallel p_{\text{data}}) - E_{x \sim p_g} \ln(1 - D^*(x)) = KL(p_g \parallel p_{\text{data}}) - 2JS(p_{\text{data}} \parallel p_g) + 2\ln 2 + E_{x \sim p_{\text{data}}} \ln(D^*(x)) \quad (8)$$

由于式(8)右边的后 2 项不依赖 G , 即最小化式(4)等价于最小化式(9)。

$$G_{\text{loss_second}|D^*} = KL(p_g \parallel p_{\text{data}}) - 2JS(p_{\text{data}} \parallel p_g) \quad (9)$$

该目标形式一方面要求最小化生成分布与真实分布间的 KL 散度, 另一方面又要求最大化两者的 JS 散度, 即优化目标相互矛盾。此外, $KL(p_g \parallel p_{\text{data}})$ 不是一个对称的度量, 当 p_{data} 与 p_g 的取值相对改变时, KL 散度也会变化, 这就迫使生成器生成大量重复且置信度较高的样本, 最终导致了模式崩溃现象。

为解决生成器第 1 种损失函数存在的问题, WGAN 提出对生成样本与真实样本添加噪声的方法, 使得原本的 2 个低维流形弥散到整个高维空间, 迫使它们产生不可忽略的重叠。此时, JS 散度将发挥作用, 从而避免了梯度消失。随着训练的进行, 逐渐对噪声退火, JS 散度继续产生有意义的梯度, 拉近 2 个低维流形直到完全重合。

本文借鉴噪声退火的思想, 通过引入时间参数 $w(t) = \exp[-10(1-t)^2]$ 来控制生成器在不同的训练阶段采用不同形式的损失函数。训练前期以 $E_{z \sim p_{\text{noise}}}[-\ln D(G(z))]$ 为主, 随着训练的进行, 真实样本和生成样本将发生重叠, 训练进行到某一阶段再切换到以 $E_{z \sim p_{\text{noise}}}[\ln(1 - D(G(z)))]$ 为主, 此时, JS 散度将发挥作用。同时, 为给生成器提供足够的梯度, 引入生成样本与真实样本之间特征级的均方差损失。最终, 生成器的损失函数如下:

$$\begin{aligned} \min_G V(G) = & \alpha \{ w(t) E_{z \sim p_{\text{noise}}} [\ln(1 - D(G(z)))] + \\ & (1 - w(t)) E_{z \sim p_{\text{noise}}} \ln(-D(G(z))) \} + \\ & \beta \{ \| E_{x \sim p_{\text{data}}} D_f(x) - E_{z \sim p_{\text{noise}}} D_f(G(z)) \|_2^2 \} \quad (10) \end{aligned}$$

其中, $D_f(\cdot)$ 表示鉴别器特征层的输出, α 表示对抗损失的权重系数, β 表示特征级损失系数。PL-GAN 算法流程如图 1 所示, 其中, $epoch$ 表示当前迭代次数, T 表示损失切换迭代次数, 实线表示前向传播, 虚线表示后向的参数更新。

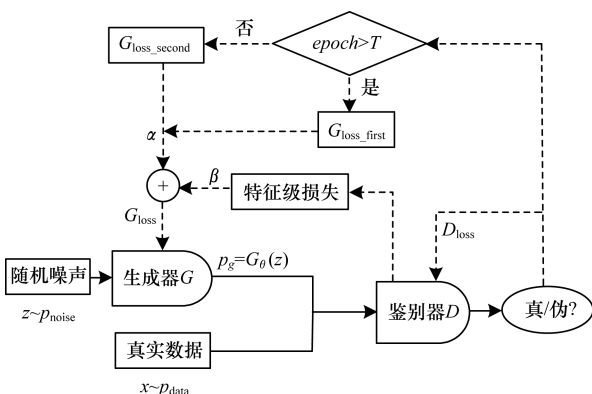


图 1 PL-GAN 算法流程

考虑到 GAN 的监督损失, 假设标准的分类器输出是 N 维向量 $\text{logits} = \{l_1, l_2, \dots, l_N\}$, N 为样本的类别数。则用 softmax 计算的输出概率为:

$$P_{\text{model}}(y = j | x) = \exp(l_j) / \sum_{n=1}^N \exp(l_n) \quad (11)$$

将生成样本所属的类别定义为第 $N + 1$ 类, 则来自生成样本的概率为 $P_{\text{model}}(y = N + 1 | x)$, 对应常规 GAN 的 $1 - D(x)$ 。假设鉴别器的训练样本一半来自真实样本, 一半来自生成样本, 则 D 的损失函数为:

$$\begin{aligned} C(D) = & -E_{x, y \sim p_{\text{data}}(x, y)} [\ln P_{\text{model}}(y | x)] - \\ & E_{x \sim p_g} [\ln P_{\text{model}}(y = N + 1 | x)] = \\ & C_{\text{sup}} + C_{\text{adv}} \quad (12) \end{aligned}$$

其中, 真实样本由带标签样本与不带标签样本组成。带标签样本参与的监督损失为:

$$C_{\text{sup}} = -E_{x, y \sim p_{\text{data}}(x, y)} [\ln P_{\text{model}}(y | x, y < N + 1)] \quad (13)$$

不带标签的真实样本与生成样本参与的无监督对抗损失为:

$$\begin{aligned} C_{\text{adv}} = & -E_{x \sim p_{\text{data}}(x)} [\ln(1 - P_{\text{model}}(y = N + 1 | x))] - \\ & E_{x \sim p_g} [\ln P_{\text{model}}(y = N + 1 | x)] \quad (14) \end{aligned}$$

令 $D(x) = 1 - P_{\text{model}}(y = N + 1 | x)$, 则有:

$$C_{\text{adv}} = -E_{x \sim p_{\text{data}}(x)} \ln(1 - D(x)) - E_{x \sim p_g} \ln(1 - D(x)) \quad (15)$$

计算监督损失与无监督对抗损失成为算法实现的关键。图 2 所示为 GAN 半监督图像分类流程。其中, 标签样本为 D 贡献监督损失, 无标签样本为 D 贡献无监督对抗损失。

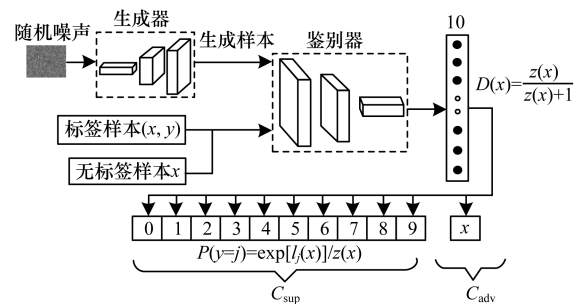


图 2 GAN 半监督图像分类流程

从优化目标方面分析, 存在一个未知的映射函数 $f(x)$, 使 $\forall j < N + 1, P(y = j, x) = f(x) \cdot \exp[l_j(x)]$ 、 $P_G(x) = f(x) \cdot \exp[l_{N+1}(x)]$ 成立^[9]。由于鉴别器输出维度为 $N + 1$ 的概率是过参数化的, 假设 $\forall x, l_{N+1}(x) = 0$ 成立, 则不改变鉴别器 softmax 的概率值。此时, $D(x) = z(x) / (z(x) + 1), z(x) = \sum_{n=1}^N \exp[l_n(x)]$ 。

1.2 算法步骤

本文基于分段损失的生成对抗网络半监督分类算法 PL-GAN 描述如下。确定最大迭代次数与损失切

换迭代次数后,设时间参数 $w(t) = \exp(-10(1-t)^2)$, t 等于当前迭代次数 $epoch$ 与损失切换迭代次数 T 的比值。

算法1 PL-GAN 算法

初始化:设批大小 (Batchsize, 即每一次参数更新时所需的样本数) $m = 100$; iterations 为批迭代次数, 表示总训练样本与批大小 m 的整数比; epoch 为迭代次数, 即遍历整个训练集的循环次数; 超参数 $k = 1$, 表示训练鉴别器 k 次才训练生成器 1 次; 对抗损失和特征级均方差损失权重系数设置为 $\alpha = \beta = 0.5$; 用 Xavier 方法进行参数初始化; 噪声分布 $P_{\text{noise}}(z)$ 满足 $Z \sim U[-1, 1]$;

for number of training iterations do

for k steps do

1. 抽样 m 个服从噪声分布的样本 $\{z^{(1)}, z^{(2)}, \dots, z^{(m)}\}$

2. 抽样 m 个无标签真实样本 $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$

3. 抽样 m 个有标签真实样本 $\{(x_1^{(1)}, y^{(1)}), (x_1^{(2)}, y^{(2)}), \dots, (x_1^{(m)}, y^{(m)})\}$

4. 计算鉴别器的无监督对抗损失:

$$C_{\text{adv}} = -\frac{1}{m} \sum_{i=1}^m [\ln D(x^{(i)}) + \ln(1 - D(G(z^{(i)})))]$$

5. 计算鉴别器的监督损失:

$$C_{\text{sup}} = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \ln(\exp(x_1^{(i)})/z(x_1^{(i)})) + (1 - y^{(i)}) \ln(1 - \exp(x_1^{(i)})/z(x_1^{(i)}))]$$

6. 通过 Adam 优化算法更新鉴别器的参数:

$$\nabla \theta_d [C_{\text{adv}} + C_{\text{sup}}]$$

end for

7. 抽样 m 个服从噪声分布的噪声样本 $\{z^{(1)}, z^{(2)}, \dots, z^{(m)}\}$

8. 抽样 m 个无标签样本 $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$

9. 计算生成器的特征级均方差损失:

$$V_{\text{fea}}(x, z) = \|E_{x \sim p_{\text{data}}(x)} D_f(x) - E_{z \sim p_{\text{noise}}(z)} D_f(G(z))\|_2^2$$

10. 计算生成器的对抗损失:

$$V_{\text{adv}}(z) = w(t) E_{z \sim p_{\text{noise}}} \ln(1 - D(G(z))) + (1 - w(t)) E_{z \sim p_{\text{noise}}} \ln(-D(G(z)))$$

11. 通过 Adam 优化算法更新生成器的参数:

$$\nabla \theta_g \frac{1}{m} \sum_{i=1}^m \alpha V_{\text{adv}}(z^{(i)}) + \beta V_{\text{fea}}(x^{(i)}, z^{(i)})$$

end for

2 实验结果与分析

2.1 实验数据集

本文采用 MNIST^[15]、CIFAR-10^[16] 数据集进行实验验证。MNIST 是深度学习领域常见的手写字体数据集, 共 10 类(用数字 0~9 表示), 每类包含单通道的 6 000 个训练样本和 1 000 个测试样本。CIFAR-10 包含 60 000 张、共 10 类的三通道样本, 每类有 5 000 个训练样本和 1 000 个测试样本。

2.2 参数设置

实验在单块 GPU 型号为 GTX1080 的计算机上运行, 基于 theano^[17] 深度学习框架实现。对于生

成器的损失函数而言, 损失切换迭代次数 T 以及学习率衰减策略非常重要。本次实验数据集中参数 T 与学习率衰减因子的设置如表 1 所示, 其中, 实验所用学习率为初始学习率与衰减因子的乘积。

表 1 参数设置

数据集	总迭代次数	T	初始学习率	衰减因子
MNIST	300	60	0.003 0	$\min(2 - epoch/250, 1)$
CIFAR-10	600	100	0.000 3	$\min(2 - epoch/450, 1)$

2.3 性能分析

2.3.1 MNIST 数据集

本次实验的网络结构主要由多层感知机组成, PL-GAN 模型的参数配置如表 2 所示, 为加速模型的训练速度, 在鉴别器结构中使用了权重归一化 (Weight Normalization, WN)^[18] 策略。

表 2 MNIST 数据集网络参数配置

鉴别器 D	生成器 G
输入: 28×28 gray image one-hot labels $y \in \mathbb{R}^{10}$	Input Noise $\in \mathbb{R}^{100}$
Denselayer 1 000 Units lReLU, Guassian noise, WN	Denselayer 500 Units
Denselayer 500 Units, lReLU, Guassian noise, WN	Softplus, batch norm
Denselayer 250 Units, lReLU, Guassian noise, WN	Denselayer 500 Units
Denselayer 250 Units, lReLU, Guassian noise, WN	Softplus, batch norm
Denselayer 10 Units, lReLU, Guassian noise, WN	Denselayer 784 Units, sigmoid

将生成样本的质量与半监督的分类精度作为实验结果的评价标准。对比算法主要包括:

1) 常规 GAN (regular GAN), 其生成器的损失为对抗损失 $E_{z \sim p_{\text{noise}}} \ln(-D(G(z)))$ 。

2) 特征级损失 GAN (feature-wise GAN), 其生成器的损失仅为特征级均方差损失 $\|E_{x \sim p_{\text{data}}(x)} D_f(x) - E_{z \sim p_{\text{noise}}(z)} D_f(G(z))\|_2^2$ 。

PL-GAN 在 regular GAN 的基础上改变对抗损失的形式并引入特征级损失, 经过加权后作为最终损失。图 3 所示为各算法的生成样本对比结果。由图 3 可以看出, 与 feature-wise GAN 相比, PL-GAN 生成的样本质量较好, 与 regular GAN 相比, PL-GAN 生成的样本虽然质量稍差, 但其多样性得到明显提升, 即 PL-GAN 解决了 regular GAN 的模式崩溃问题。

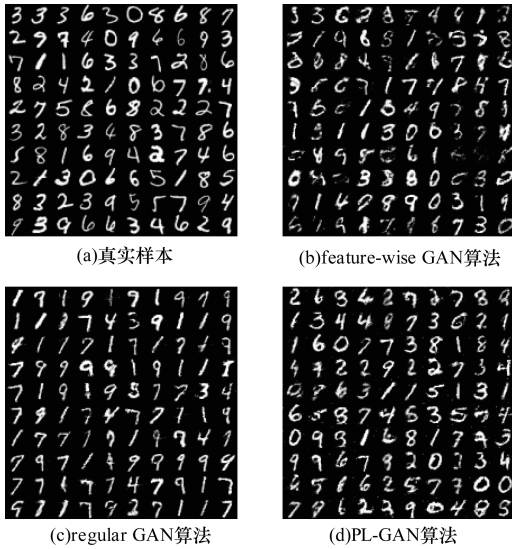


图 3 MNIST 数据集中 3 种算法的生成样本对比结果

图 4 所示为各算法在训练过程中的损失变化对比。由图 4 可以看出,对生成器的损失而言,相比 feature-wise GAN,PL-GAN 的损失下降得更加平稳;与 regular GAN 相比,PL-GAN 的损失呈现逐渐下降的趋势,而并非趋近于一个常数。对鉴别器的损失而言,相比 regular GAN,PL-GAN 的损失值较低,其与 feature-wise GAN 相当。

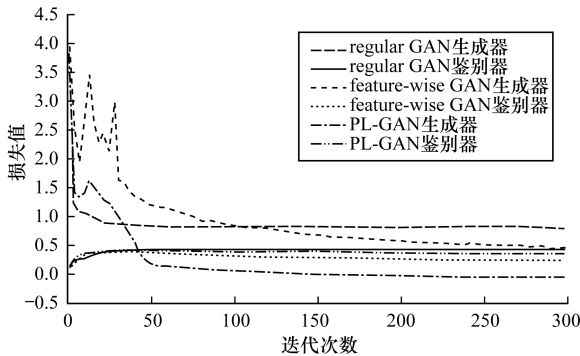


图 4 MNIST 数据集中算法的损失变化对比结果

在保证模型结构相同的情况下,当标签样本数为 100 时,比较各算法的半监督分类性能,分类对比结果如图 5 所示。由图 5 可以看出,PL-GAN 比 regular GAN 分类错误率低,其分类性能与 feature-wise GAN 基本接近,但收敛速度更快。

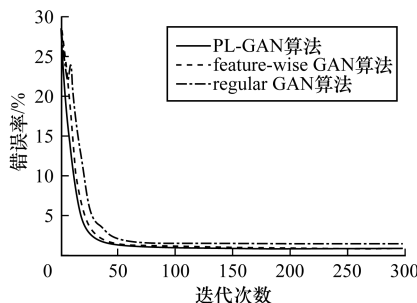


图 5 MNIST 数据集中 3 种算法的错误率对比结果

同样设标签样本数为 100,将 PL-GAN 与传统的半监督学习算法 ADGM、M1 + M2、VAT、CatGAN 等进行比较,结果如表 3 所示。从表 3 可以看出,PL-GAN 具有最低的分类错误率。

表 3 MNIST 数据集中 8 种算法分类错误率对比 %

算法	错误率
Ladder ^[3]	1.06 (±0.37)
M1 + M2 ^[5]	3.33 (±0.14)
ADGM ^[6]	0.96 (±0.02)
CatGAN ^[8]	1.39 (±0.28)
Improved-GAN ^[9]	0.93 (±0.07)
Triple-GAN ^[10]	0.91 (±0.58)
VAT ^[19]	2.33
PL-GAN	0.90 (±0.06)

2.3.2 CIFAR-10 数据集

本次实验的生成器结构以 DCGAN^[20] 框架为基准,初始学习率设为 0.000 3,为加速模型训练并防止出现过拟合现象,鉴别器使用 WN 和 Dropout 策略。网络模型参数配置如表 4 所示。

表 4 CIFAR-10 数据集网络参数配置

鉴别器 D	生成器 G
输入: 32 × 32 Colored image one-hot labels classy ∈ ℝ ¹⁰	Input Noise ∈ ℝ ¹⁰⁰
Dropout = 0.2 3 × 3 conv, 96, lReLU, WN	MLP 8 192 units ReLU, batch norm Reshape 512 × 4 × 4 5 × 5 deconv, 256, stride 2 ReLU, batch norm
Dropout = 0.2 3 × 3 conv, 192, lReLU, WN	5 × 5 deconv. 128. stride 2 ReLU, batch norm
Dropout = 0.2 3 × 3 conv, 192, lReLU, WN NIN, 192, lReLU, WN Global pool layer Denselayer 10 Units with WN	5 × 5 deconv. 3. stride 2, tanh, weight norm

为定量衡量生成样本质量的优劣,利用 IS (Inception Score) 指标依次对不同算法生成的 5 000 个样本进行 10 次评估,得到相应的 IS 值,结果如表 5 所示。图 6 所示为各算法在 CIFAR-10 数据集上生成样本的视觉对比。

表 5 CIFAR-10 数据集中 3 种算法的 IS 值对比结果

算法	IS 值
regular GAN	6.45 ± 0.22
feature-wise GAN	5.46 ± 0.20
PL-GAN	7.30 ± 0.17
真实样本	10.71 ± 0.44

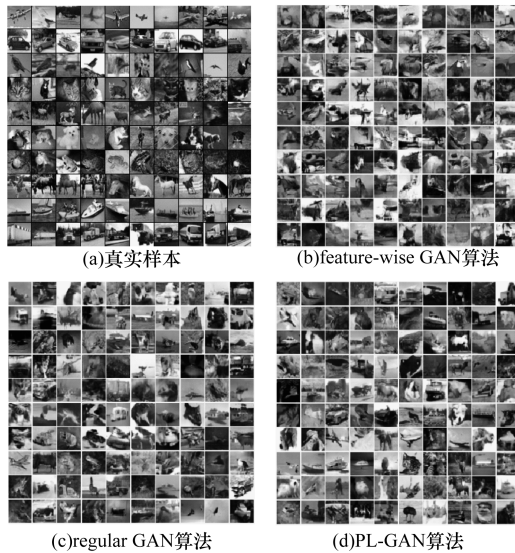


图6 CIFAR-10数据集中3种算法的生成样本视觉对比

在保证模型结构框架相同的情况下,当标签样本数为4 000时,各算法的半监督分类错误率对比结果如图7所示。由图7可以看出,PL-GAN比regular GAN分类错误率低,与feature-wise GAN分类性能基本接近,但收敛性更好。设带标签的训练样本数为4 000,将PL-GAN与传统半监督分类算法进行对比,结果如表6所示,可以看出,PL-GAN有较好的性能表现,优于大多数传统半监督分类算法。

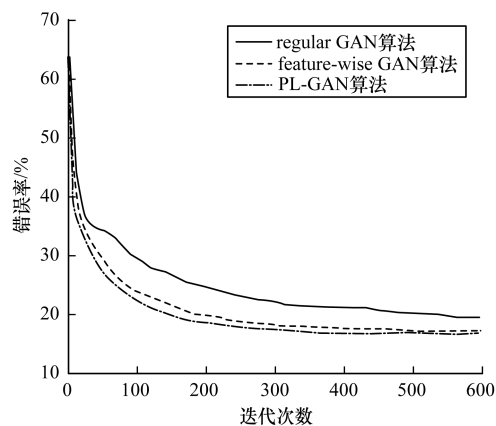


图7 CIFAR-10数据集中3种算法的错误率对比结果

表6 CIFAR-10数据集中7种算法的错误率对比 %

算法	错误率
Ladder	20.40(±0.47)
ALI ^[7]	18.30
CatGAN	19.58(±0.58)
Improved-GAN	18.63(±2.32)
Triple GAN	16.99(±0.36)
VAT	24.65
PL-GAN	17.30(±0.56)

3 结束语

本文提出一种基于分段损失的生成对抗网络PL-GAN。通过设置时间参数来改变生成器与鉴别器的训练过程,使衡量生成分布与真实分布之间差异的JS散度能够更好地发挥作用。在生成器中引入真实样本与生成样本之间的特征级损失,使训练过程更加稳定。实验结果表明,PL-GAN具有较高的分类精度与较快的收敛速度。

参考文献

- [1] CHAPPELLE O, SCHOLKOPF B, ZIEN A. Semi-supervised learning [J]. IEEE Transactions on Neural Networks, 2009, 20(3): 542-546.
- [2] LEE D H. Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks [EB/OL]. [2018-02-01]. http://deeplearning.net/wp-content/uploads/2013/03/pseudo_label_final.pdf.
- [3] RASMUS A, BERGLUND M, HONKALA M, et al. Semi-supervised learning with ladder networks [EB/OL]. [2018-02-01]. <http://papers.nips.cc/paper/5947-semi-supervised-learning-with-ladder-networks.pdf>.
- [4] KINGMA D P, MOHAMED S, REZENDE D J, et al. Semi-supervised learning with deep generative models [EB/OL]. [2018-02-05]. <https://arxiv.org/pdf/1406.5298.pdf>.
- [5] GOODFELLOW I J, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial nets [C]//Proceedings of International Conference on Neural Information Processing Systems. Cambridge, USA: MIT Press, 2014: 2672-2680.
- [6] MAALØE L, SØNDERBY C K, SØNDERBY S K, et al. Auxiliary deep generative models [EB/OL]. [2018-02-06]. <https://arxiv.org/pdf/1602.05473.pdf>.
- [7] DUMOULIN V, BELGHAZI I, POOLE B, et al. Adversarially learned inference [EB/OL]. [2018-02-04]. <https://arxiv.org/pdf/1606.00704.pdf>.
- [8] SPRINGENBERG J T. Unsupervised and semi-supervised learning with categorical generative adversarial networks [J]. Computer Science, 2015(6): 2321-2330.
- [9] SALIMANS T, GOODFELLOW I, ZAREMBA W, et al. Improved techniques for training GANs [C]//Proceedings of NIPS'16. Barcelona, Spain: NIPS, 2016: 2234-2242.
- [10] LI Chongxuan, XU Kun, ZHU Jun, et al. Triple generative adversarial nets [C]//Proceedings of NIPS'17. Barcelona, Spain: NIPS, 2017: 4091-4101.
- [11] 王坤峰, 苟超, 段艳杰, 等. 生成式对抗网络 GAN 的研究进展与展望 [J]. 自动化学报, 2017, 43(3): 321-332.
- [12] ARJOVSKY M, BOTTOU L. Towards principled methods for training generative adversarial networks [EB/OL]. [2018-01-25]. <https://leon.bottou.org/publications/pdf/iclr-2017.pdf>.
- [13] ARJOVSKY M, CHINTALA S, BOTTOU L. Wasserstein GAN [EB/OL]. [2018-02-05]. <https://arxiv.org/pdf/1701.07875.pdf>.

(下转第168页)

综上,本文提出的自适应邻域更新策略与传统固定邻域相比,在处理复杂、凸面测试问题时,既能保证算法的收敛性又能兼顾解的分布性。

4 结束语

为了避免经典 MOEA/D 算法固定邻域导致的种群进化效率低的问题,本文提出一种基于自适应邻域策略的改进型 MOEA/D 算法。设计一种基于邻域更新能力的种群进化状态判断机制,并基于该机制提出自适应邻域策略,使邻域规模根据种群和子问题进化状态自适应地进行调整,达到平衡种群收敛性与分布性的目的。实验结果表明,该算法较好的分布性和收敛性。下一步将根据子问题的邻域更新能力采取相应的邻域更新策略,并通过理论分析寻找更优的 ε_m 和 ε_M 阈值组合。

参考文献

- [1] ZITZLER E, THIELE L. Multiobjective optimization using evolutionary algorithms—a comparative case study [C] // Proceedings of International Conference on Parallel Problem Solving from Nature. Berlin, Germany: Springer, 2006:292-301.
- [2] LI Yuanyong, ZHOU Yuren, ZHAN Zhihui, et al. A primary theoretical study on decomposition-based multiobjective evolutionary algorithms [J]. IEEE Transactions on Evolutionary Computation, 2016, 20(4):563-576.
- [3] WANG Rui, ZHANG Qingfu, ZHANG Tao. Decomposition based algorithms using Pareto adaptive scalarizing methods [J]. IEEE Transactions on Evolutionary Computation, 2016, 20(6):821-837.
- [4] JIANG Siwei, ZHANG Jie, ONG Y S, et al. A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm [J]. IEEE Transactions on Cybernetics, 2015, 45(10):2202-2213.
- [5] ZHANG Qingfu, LI Hui. MOEA/D: a multiobjective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6):712-731.
- [6] LI Hui, ZHANG Qingfu. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2009, 13(2):284-302.
- [7] ZHANG Qingfu, LIU Wudong, LI Hui. The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances [C] // Proceedings of 2009 IEEE Congress on Evolutionary Computation. Washington D. C., USA: IEEE Press, 2009:203-208.
- [8] ISHIBUCHI H, HITOTSUYANAGI Y, TSUKAMOTO N, et al. Use of biased neighborhood structures in multiobjective memetic algorithms [J]. Soft Computing, 2009, 13(8/9):795-810.
- [9] ZHAO Shizheng, SUGANTHAN P N, ZHANG Qingfu. Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes [J]. IEEE Transactions on Evolutionary Computation, 2012, 16(3):442-446.
- [10] 刘志君, 高亚奎, 章卫国, 等. 变邻域分解多目标自适应差分进化算法 [J]. 控制理论与应用, 2014, 31(11):1492-1501.
- [11] WANG Zhenkun, ZHANG Qingfu, ZHOU Aimin, et al. Adaptive replacement strategies for MOEA/D [J]. IEEE Transactions on Cybernetics, 2016, 46(2):474-486.
- [12] 刘璐, 郑力明. 基于邻域和变异算子组合优化的 MOEA/D 算法 [J]. 计算机工程, 2017, 43(3):232-240.
- [13] HUBAND S, HINGSTON P, BARONE L, et al. A review of multiobjective test problems and a scalable test problem toolkit [J]. IEEE Transactions on Evolutionary Computation, 2006, 10(5):477-506.
- [14] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multiobjective genetic algorithm; NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2):182-197.
- [15] WANG Hu, YEN G G. Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(1):1-18.
- [14] METZ L, POOLE B, PFAU D, et al. Unrolled generative adversarial networks [EB/OL]. [2018-02-07]. <https://arxiv.org/pdf/1611.02163.pdf>.
- [15] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.
- [16] KRIZHEVSKY A, HINTON G. Learning multiple layers of features from tiny images [J]. Handbook of Systemic Autoimmune Diseases, 2009, 1(4):32-33.
- [17] BASTIEN F, LAMBLIN P, PASCANU R, et al. Theano: new features and speed improvements [EB/OL]. [2018-01-25]. <http://export.arxiv.org/pdf/1211.5590>.
- [18] SALIMANS T, KINGMA D P. Weight normalization: a simple reparameterization to accelerate training of deep neural networks [C] // Proceedings of NIPS '16. Barcelona, Spain: NIPS, 2016:901-909.
- [19] MIYATO T, MAEDA S, KOYAMA M, et al. Distributional smoothing with virtual adversarial training [EB/OL]. [2018-01-26]. <https://arxiv.org/pdf/1507.00677.pdf>.
- [20] RADFORD A, METZ L, CHINTALA S. Unsupervised representation learning with deep convolutional generative adversarial networks [EB/OL]. [2018-02-10]. <https://arxiv.org/pdf/1511.06434.pdf>.

编辑 樊丽娜

编辑 吴云芳

(上接第 160 页)