Pure Mathematics 理论数学, 2023, 13(5), 1255-1266
Published Online May 2023 in Hans. https://www.hanspub.org/journal/pm
https://doi.org/10.12677/pm.2023.135129

Hans汉斯

极小极大优化问题的一类自适应三次正则化牛顿法

高瑞成

重庆师范大学数学科学学院，重庆

收稿日期：2023年4月12日；录用日期：2023年5月15日；发布日期：2023年5月22日

摘　要

许多机器学习问题，如对抗学习、强化学习和图像处理的一些典型问题大都可归结为极小极大优化问题。因此，极小极大优化问题最近已经成为最优化领域与机器学习交叉领域中的一个重要研究前沿和热点，吸引了众多学者的关注和研究。如何有效求解极小极大优化问题，是优化领域及应用中的关键科学问题之一。考虑到极小极大三次正则化牛顿法中，三次正则化项的系数是相对固定的，会影响算法的收敛性。本文首先针对非凸 – 强凹极小极大优化问题，采用信赖域半径的选取策略，提出了一类自适应极小极大三次正则化牛顿法。然后证得了算法的迭代复杂度为$\mathcal{O}\left(T^{-1/3}\right)$。最后，通过一类对抗攻击神经网络问题，验证了该算法的有效性。

关键词

极小极大优化问题，三次正则化牛顿法，非凸优化，自适应性，复杂度分析

One Class of Cubic Regularized Newton Methods for Min-Max Optimization Problems

Ruicheng Gao

School of Mathematical Sciences, Chongqing Normal University, Chongqing

Received: Apr. 12th, 2023; accepted: May 15th, 2023; published: May 22nd, 2023

Abstract

Many machine learning problems, such as adversarial learning, reinforcement learning and image processing, can be reduced to min-max optimization problem. Therefore, min-max optimization

文章引用: 高瑞成. 极小极大优化问题的一类自适应三次正则化牛顿法[J]. 理论数学, 2023, 13(5): 1255-1266.
DOI: 10.12677/pm.2023.135129

has recently become an important research frontier and hot spot in the intersection of optimization and machine learning, attracting the attention and research of many scholars. How to effectively solve min-max problem is one of the key scientific problems in optimization field and application. Considering that the coefficient of cubic regularization term in min-max cubic regularization Newton method is relatively fixed, it will affect the convergence of the algorithm. In this paper, a class of adaptive min-max cubic regularization Newton method is proposed based on the selection strategy of the radius of the trust region for the nonconvex-strongly concave minimax optimization problem. Then, the iterative complexity of the algorithm is bounded by $\mathcal{O}\left(T^{-1/3}\right)$. Finally, the effectiveness of the proposed algorithm is verified by a class of adversarial attack neural network problems.

Keywords

Min-Max Optimization Problem, Cubic Regularized Newton Method, Nonconvex Optimization, Adaptability, Complexity Analysis

1. 引言

许多机器学习问题，如对抗学习、强化学习的一些典型问题大都归结为极小极大优化问题[1] [2]。还有一些著名的数学和优化难题也可看成是某类极小极大问题的特例，例如，著名数学家 S. Smale 在 1998 年提出的 18 个数学公开问题的第 7 个：球面点散布问题。它可看成是在三维空间单位球面上的一个特殊的非凸极小极大问题。带等式或不等式的约束优化问题，对应的拉格朗日对偶问题也可看成是一类极小极大优化问题。因此，极小极大优化问题最近已经成为最优化领域与机器学习领域中的一个重要研究前沿和热点，吸引了众多学者的关注和研究。如何有效求解极小极大优化问题是优化领域及应用中的关键问题之一。

目前针对凸－凹极小极大优化问题的算法和理论研究已经取得许多较为成熟的研究结果。但是，机器学习中的大多数极小极大优化问题的目标函数都是非凸的，相比于凸－凹极小极大优化问题，非凸极小极大优化问题在理论研究和求解难度上都更具有挑战性。针对非凸极小极大优化问题的算法研究，目前大致包含两类，分别为单循环算法和多循环算法。单循环算法的好处是不需要求解子问题，因其实现起来简单，受到众多学者的关注和研究。其中最简单的是梯度下降－上升算法(GDA)。但是，Takeuchi 等人[3]在 2019 年的研究表明：算法 GDA 即使是对于简单的双线性极小极大优化问题也难以保证算法收敛。Lu 等人[4]借鉴类似于块上界梯度下降－上升算法的思想，提出混合分块序列近似算法求解非凸－凹极小极大问题，该算法找到一阶平稳点的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-4}\right)$。

另外一大类极小极大优化是多循环算法。Sanjabi 等人[5]在 2018 年提出一种基于非凸随机梯度下降算法求解 GANs 问题，并证得算法找到近似一阶平稳点的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-2}\right)$。Nouiehed 等人[6]在 2019 年提出一种多步上升－下降梯度算法，并给出该算法内层循环的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-1/2}\right)$，外层循环的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-3}\right)$，最终算法得到目标函数的一阶纳什均衡点的复杂度为$\mathcal{O}\left(\varepsilon^{-3.5}\right)$。Kong 等人[7]提出了一种加速非精确临近点光滑化方法求解非凸－凹极小极大优化问题，得到算法的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-3}\right)$，但该方法子问题的求解复杂度没有计算在内。2020 年，Lin 等人[8]提出了一类加速算法求解光滑非凸－

凹极小极大优化问题，并得到该算法可以达到$\mathcal{O}\left(\varepsilon^{-2.5}\right)$的迭代复杂度。但是，以上这些算法大都属于一阶优化算法。

Nesterov 等人[9]在 2006 年通过引入一个目标函数的三次正则化项的二次近似模型，设计出了一类三次正则化牛顿法来求解无约束非凸优化问题，并得到该算法的全局收敛性和局部二次收敛性。由于三次正则化牛顿法具有良好的鲁棒性，全局收敛性和逃离鞍点的能力，这使得它已经成为求解无约束优化问题的一类重要方法。但是该算法中的三次正则化系数是固定的。于是，Cartis 等人[10] [11]引入信赖域半径选取的方法，提出一类自适应三次正则化牛顿法，其好处是能自适应调节三次正则化项系数。为求解非凸－强凹极小极大优化问题，Luo 等人[12]借助于三次正则化牛顿方法思想，设计了一类双循环算法，其中外层循环使用了三次正则化牛顿法(简称为：极小极大三次正则化牛顿法，MCN)，并证得算法达到二阶平稳点时的迭代复杂度为$\mathcal{O}\left(\varepsilon^{-3/2}\right)$。尽管该算法在外层循环使用了三次正则化牛顿法，但是其三次正则化项的系数是相对固定的。

受文献[10] [11] [12]工作的启发，本文采用信赖域半径选取的策略，提出一类自适应极小极大三次正则化牛顿法(A-MCN)来求解非凸－强凹极小极大优化问题，并得到了算法的迭代复杂度为$\mathcal{O}\left(T^{-1/3}\right)$。最后，通过数值实验验证了该算法的有效性。相比于文献[12]提出的算法 MCN，本文提出的算法 A-MCN 能自适应调节三次正则化项的系数，数值结果表明算法 A-MCN 优于算法 MCN。

本文在第 2 节中，介绍考虑的问题及其相关假设。在第 3 节中，给出相应的算法迭代步骤。在第 4 节中，针对算法进行收敛性分析，并证明算法的收敛速率。在第 5 节中，给出数值实验结果。最后在第 6 节中，总结本文的主要研究内容。

2. 极小极大优化问题

考虑如下一类极小极大优化问题：

$$\min_{x\in\mathbb{R}^{d_x}}\left\{Q(x)\triangleq\max_{y\in\mathbb{R}^{d_y}}f(x,y)\right\},\tag{1}$$

其中$f(x,y)$是一个二次连续可微函数，且对任意固定的 y 关于 x 是非凸的，对任意固定的 x 关于 y 是 μ-强凹的。

下面给出关于问题(1)的一些假设。

假设 1 a) 对任意给定的$y\in\mathbb{R}^{d_y}$，函数$f(x,y)$的梯度$\nabla f(x,y)$关于 x 是 l-Lipschitz 连续的，即存在一个常数$l>0$，使得对于$\forall x,x'\in\mathbb{R}^{d_x}$，有$\left\|\nabla f(x,y)-\nabla f(x',y)\right\|\le l\left\|x-x'\right\|$。

b) 对任意给定的$y\in\mathbb{R}^{d_y}$，函数$f(x,y)$关于 x 是二次可微的。并且关于 x 的 Hessian 矩阵$\nabla_{xx}^2 f(x,y)$是 L_H-Lipschitz 连续的，即对$\forall x,x'\in\mathbb{R}^{d_x}$，存在一个常数$L_H>0$，有

$$\left\|\nabla_{xx}^2 f(x,y)-\nabla_{xx}^2 f(x',y)\right\|\le L_H\left\|x-x'\right\|.$$

c) 对任意给定的$x\in\mathbb{R}^{d_x}$，$f(x,y)$关于 y 是 μ-强凹的，即对$\forall y,y'\in\mathbb{R}^{d_y}$，存在一个常数$\mu>0$，有

$$f(x,y)\le f(x,y')+\nabla_y f(x,y)^{\top}(y-y')-\frac{\mu}{2}\left\|y-y'\right\|^2.$$

假设 2：函数$Q(x)\triangleq\max_{y\in\mathbb{R}^{d_y}}f(x,y)$有下界，即$Q^*=\inf_{x\in\mathbb{R}^{d_x}}Q(x)>-\infty$。

下面的两个引理给出了$Q(x)$的一些性质。

引理 1 [13]若 f 满足假设 1 中的 a)和 c)，那么有：a) $Q(x)$的梯度$\nabla Q(x)=\nabla_x f\left(x,y^*(x)\right)$是$(\zeta+1)l$-Lipschitz 连续的，这里$\zeta=l/\mu$；b) $y^*(x)=\arg\max_{y\in\mathbb{R}^{d_y}}f(x,y)$是唯一，且是$\zeta$-Lipschitz 连续的。

引理 2 [12]若 f 满足假设 1，那么有：a) $Q(x)$在点$y^*(x)$处有

$$\nabla^2 Q(x)=\nabla^2_{xx} f(x,y)-\nabla^2_{xy} f(x,y)\left[\nabla^2_{yy} f(x,y)\right]^{-1}\nabla^2_{yx} f(x,y),$$

b) $\nabla^2 Q(x)$ 是 ω-Lipschitz 连续的，其中 $\omega=4\sqrt{2}\zeta^3 L_H$，即 $\forall x,x'\in\mathbb{R}^{d_x}$，有

$$\left\|\nabla^2 Q(x)-\nabla^2 Q(x')\right\|\le\omega\|x-x'\|.$$

3. 自适应极小极大三次正则化牛顿法

为求解一类非凸－强凹极小极大优化问题，基于 Luo 等人[12]提出的极小极大三次正则化牛顿法(MCN)框架，为调节该算法中固定的三次正则化项的系数，借助 Cartis 等人[10] [11]提出的算法中信赖域半径的选取策略，本文设计了一类自适应极小极大三次正则化牛顿法(A-MCN)求解问题(1)，其中针对外层循环设计了一种自适应的三次正则化牛顿法，针对内层子问题设计了一种加速梯度算法。下面给出具体算法框架：

算法 1. Adaptive Min-Max Cubic Regularized Newton Method (A-MCN)

初始化： $x_0\in\mathbb{R}^{d_x}$ 和 $y_0\in\mathbb{R}^{d_y}$。选取合适的 $T\ge1$，非负序列 $\{\sigma_t\}$ 满足 $\sigma_t>\sigma_{\min}>0$，$\gamma_1>1\ge\gamma_2>\gamma_3>0$，$1>\eta_2>\eta_1>0$，以及正整数序列 $\{K_t\}$.

for $t=1$ to T **do**

更新 y_t：$y_t=\mathrm{AGD}(x_{t-1},y_{t-1},K_t,\alpha,\beta)$，见算法 2。

$g_t=\nabla_x f(x_{t-1},y_t)$,

$H_t=\nabla^2 Q(x)$,

$$d_{t-1}\in\arg\min_d\left\{m_{t-1}(d):=Q(x_{t-1})+g_t^{\top}d+\frac{1}{2}d^{\top}H_t d+\frac{\sigma_{t-1}}{6}\|d\|^3\right\},$$

计算 $Q(x_{t-1}+d_{t-1})$ 和 $\rho_{t-1}=\dfrac{Q(x_{t-1})-Q(x_{t-1}+d_{t-1})}{Q(x_{t-1})-m_{t-1}(d_{t-1})}$,

if $\rho_{t-1}>\eta_1$ (成功更新) **then**

更新 x_t：$x_t=x_{t-1}+d_{t-1}$,

if $\rho_{t-1}>\eta_2$ (非常成功的更新) **then**

$\sigma_t=\max(\sigma_{\min},\gamma_3\sigma_{t-1})$,

else

$\sigma_t=\gamma_2\sigma_{t-1}$,

end if

else (不成功的更新)

$x_t=x_{t-1}$,

$\sigma_t=\gamma_1\sigma_{t-1}$,

end if

end for

输出： (x_t,y_t).

算法 2. $y_t=\mathrm{AGD}(x_{t-1},y_{t-1},K_t,\alpha,\beta)$ (Accelerated Gradient Descent Algorithm, AGD)

初始化： $\hat{y}_0=y_{t-1}$ 和 $\tilde{y}_0=y_{t-1}$。设置 $\alpha=1/l$ 和 $\beta=\left(\sqrt{\zeta}-1\right)\Big/\left(\sqrt{\zeta}+1\right)$。

对于 $m=0$ to K_t，按照如下规则更新：

$\hat{y}_{m+1}=\tilde{y}_m+\alpha\nabla_y f(x_{t-1},\tilde{y}_m)$,

$\tilde{y}_{m+1}=\hat{y}_{m+1}+\beta(\hat{y}_{m+1}-\hat{y}_m)$.

输出： $y_t=\tilde{y}_{K_t+1}$.

注记：1) 算法 A-MCN 是先固定变量 x_{t-1}，利用算法 AGD 更新 y_t，然后固定 $y=y_t$ 利用信赖域的概念来判断每次迭代所得的牛顿更新步骤是否成功，只有当更新步骤成功时，才会使用得到的牛顿更新步骤替代上一次的迭代，以此更新 x_t，同时在更新判断的过程中，算法的三次正则化项的系数 σ_t 是可以自适应调节的。

2) 对比算法 MCN [12]，本文借助信赖域半径的选取策略，通过信赖域半径作为判断准则，判断牛顿更新步是否成功，并在判断过程中调节三次正则化项的系数。与文献[12]中算法 MCN 相比，本文提出的自适应三次正则化牛顿法，可以自适应的调节三次正则化项的系数。

4. 收敛性分析

下面为证明全局收敛性，介绍如下两个引理。

引理 3 [10] [11]对所给的初始迭代点 x_0，存在的一个闭凸集 $\mathcal{F}$，使得水平集 $\mathcal{L}(x_0):=\{x:Q(x)\le Q(x_0)\}\subseteq\mathcal{F}$，对 $\forall x,x'\in\mathcal{F}$，$\exists\kappa_H\ge 0$，有 $\|\nabla^2Q(x)\|\le\kappa_H$。

引理 4 [9] [10]如果 $d_{t-1}\in\arg\min_d\left\{m_{t-1}(d):=Q(x_{t-1})+g_t^{\top}d+\frac{1}{2}d^{\top}H_td+\frac{\sigma_{t-1}}{6}\|d\|^3\right\}$，则有

$$Q(x_{t-1})-m_{t-1}(d_{t-1})\ge\frac{1}{12}\sigma_{t-1}\|d_{t-1}\|^3. \tag{2}$$

此外，设 $\mathcal{S}:=\{t:\rho_t>\eta_1\}$ 表成功更新的索引集合，则对任意 $t\in\mathcal{S}$，有

$$Q(x_{t-1})-Q(x_t)\ge\eta_1\left(Q(x_{t-1})-m_t(s_{t-1})\right)\ge\frac{\eta_1}{12}\sigma_{t-1}\|d_{t-1}\|^3. \tag{3}$$

下面给出自适应惩罚参数 σ_t 的一个上界。

引理 5 在假设 1 和 2 下，自适应惩罚参数 σ_t 不能任意大，即

$$\sigma_t\le\max\{\omega\gamma_1,\sigma_{\min}\}=\sigma_{\max}. \tag{4}$$

证明：令：

$$r_t=Q(x_t+d_t)-m_t(d_t)+(1-\eta_2)\left(m_t(d_t)-Q(x_t)\right),$$

若 x_t 不是局部最小解，由文献[10] [11]知

$$m_t(d_t)<Q(x_t),$$

再由引理 1 有

$$\left|Q(x)-Q(x')-\nabla Q(x')(x-x')-\frac{1}{2}(x-x')^{\top}\nabla^2Q(x)(x-x')\right|\le\frac{\omega}{6}\|x-x'\|^3, \tag{5}$$

若 $\sigma_t\ge\omega$ 时，由式(5)可得

$$Q(x_t+d_t)-m_t(d_t)\le\left(\frac{\omega}{6}-\frac{\sigma_t}{6}\right)\|d_t\|^3\le 0$$

此时 $r_t<0$，等价于

$$\rho_t>\eta_2,$$

意味着此时迭代非常成功，迭代点 $t\in\mathcal{S}$.

若 $t-1\notin\mathcal{S}$，则有

$$\sigma_{t\text{-}1}<\omega,$$

那么

$$\omega \le \sigma_t = \gamma_1 \sigma_{t-1} \le \gamma_1 \omega, t \in \mathcal{S},$$

故

$$\sigma_t \le \max\{\gamma_1 \omega, \sigma_{\min}\} = \sigma_{\max}.$$ 证毕

下面给出不成功更新集合的一个上界和成功更新总集合的一个下界。

引理 6 记 $\mathcal{S}_j := \{t < j : \rho_t > \eta_1\}$，$\mathcal{U}_j := \{t < j : \rho_t \le \eta_1\}$，有

$$|\mathcal{U}_j| \le \left\lceil \log \frac{\max\{\gamma_1 \omega, \sigma_{\min}\}}{\sigma_{\min}} \right\rceil |\mathcal{S}_j|, \tag{6}$$

和

$$|\mathcal{S}_T| \ge \frac{T}{1 + \left\lceil \log \frac{\max\{\gamma_1 \omega, \sigma_{\min}\}}{\sigma_{\min}} \right\rceil}. \tag{7}$$

证明：根据引理 5，若当前迭代是成功的，则距下一次成功迭代，最多需要

$$\left\lceil \log \frac{\sigma_{\max}}{\sigma_{\min}} \right\rceil$$

步不成功迭代，那么式(6)成立。

下证式(7)成立，由式(6)以及

$$|\mathcal{S}_T| + |\mathcal{U}_T| = T,$$

即证得式(7)成立。证毕

在这里给出收敛性证明。

引理 7 对于所有给出的 d_t 以及 $\mathcal{S}_T$，有以下关系成立

$$\max_{t \in \mathcal{S}_T} \|d_t\| \le \left(\frac{12\left(Q(x_0) - Q^*\right)}{\eta_1 \sigma_{\min}} \right)^{\frac{1}{3}}, \tag{8}$$

$$\min_{t \in \mathcal{S}_T} \|d_t\| \le \left(\frac{12\left(Q(x_0) - Q^*\right)}{|\mathcal{S}_T| \eta_1 \sigma_{min}} \right)^{\frac{1}{3}}. \tag{9}$$

证明：由引理 4，有

$$\begin{aligned} \sum_{t \in \mathcal{S}_T} \frac{1}{12} \eta_1 \sigma_{t-1} \|d_{t-1}\|^3 &\le \sum_{t \in \mathcal{S}_T} Q(x_{t-1}) - Q(x_t) \\ &= \sum_{t=1}^{T} \left(Q(x_{t-1}) - Q(x_t) \right) \\ &\le Q\left(x_0 - Q^*\right), \end{aligned} \tag{10}$$

其中等式成立是因为对任意的 $t \notin \mathcal{S}_T$，有

$$Q(x_t) = Q(x_{t-1}),$$

因此

$$\sum_{t\in\mathcal{S}_T}\|d_t\|^3 \le \frac{12\left(Q(x_0)-Q^*\right)}{\eta_1\sigma_{\min}}, \sigma_{t-1}>\sigma_{\min}.$$

最后式(10)结合

$$\max_{t\in\mathcal{S}_T}\|d_t\|^3 \le \sum_{t\in\mathcal{S}_T}\|d_t\|^3,$$

得到式(8)，由

$$\min_{t\in\mathcal{S}_T}\|d_t\|^3 \le \frac{1}{|\mathcal{S}_T|}\sum_{t\in\mathcal{S}_T}\|d_t\|^3,$$

得到式(9)。证毕

为证明该算法求解问题(1)的收敛速率，给出如下引理。

引理 8 [9]假设 1 和 2 成立，序列$\{(x_t,y_t)\}$由算法 1 产生，则有

$$\|\nabla Q(x_t)\| \le \frac{1}{2}(\sigma_t+\omega)\|d_t\|^2,$$

以及

$$\lambda_{\min}\left(\nabla^2 Q(x_t)\right) \ge -\left(\frac{1}{2}\sigma_t+\omega\right)\|d_t\|.$$

定义 1 下面定义如下局部优化度量

$$\tau(x)=\max\left\{\sqrt{\frac{2}{\sigma_t+\omega}\|\nabla Q(x_t)\|}, -\frac{1}{\frac{1}{2}\sigma_t+\omega}\lambda_{\min}\left(\nabla^2 Q(x_t)\right)\right\}.$$

定理 1 若假设 1 成立，序列$\{(x_t,y_t)\}$由算法 1 产生，则对任意给定的$T\ge 1$，有

$$\min_{1\le t\le T}\tau(x_{t-1}) \le \left(\frac{12\left(Q(x_0)-Q^*\right)}{\eta_1\sigma_{\min}T}\left(1+\left\lceil\log\frac{\max\{\gamma_1\omega,\sigma_{\min}\}}{\sigma_{\min}}\right\rceil\right)\right)^{\frac{1}{3}}.$$

证明：由引理 8 可得$\forall t\in\mathcal{S}_T$，有

$$\tau(x_t)\le\|d_t\|,$$

则有

$$\min_{1\le t\le T}\tau(x_{t-1}) \le \min_{t\in\mathcal{S}_T}\|d_t\|.$$

结合式(9)和(7)，有

$$\begin{aligned}\min_{1\le t\le T}\tau(x_{t-1}) &\le \min_{t\in\mathcal{S}_T}\|d_t\| \\ &\le \left(\frac{12\left(Q(x_0)-Q^*\right)}{|\mathcal{S}_T|\eta_1\sigma_{min}}\right)^{\frac{1}{3}} \\ &\le \left(\frac{12\left(Q(x_0)-Q^*\right)}{T\eta_1\sigma_{\min}}\left(1+\left\lceil\log\frac{\max\{\gamma_1\omega,\sigma_{\min}\}}{\sigma_{\min}}\right\rceil\right)\right)^{\frac{1}{3}} \\ &= \mathcal{O}\left(T^{-1/3}\right).\end{aligned}$$

证毕

5. 数值实验

下面通过一个对抗神经网络模型(DANN) [14]来测试本文提出的算法 A-MCN 的数值表现。

DANN 是一种经典的迁移学习方法，假设源域数据集 $\mathcal{S}=\left\{\left(\boldsymbol{a}_i^{\mathcal{S}},b_i^{\mathcal{S}}\right)\right\}$，其中 $\boldsymbol{a}_i^{\mathcal{S}}$ 是第 i 次采样的特征向量，$b_i^{\mathcal{S}}$ 是对应的标签。目标域数据集是 $\mathcal{T}=\left\{\boldsymbol{a}_i^{\mathcal{T}}\right\}_{i=1}^{N_{\mathcal{T}}}$，该数据集只包含特征向量。实验环境为笔记本电脑 AMD Ryzen 5 4600U 处理器，16GB RAM，所有算法均使用 Python3.8.3 (TensorFlow 框架)编码实现。

考虑 DANN 中具有如下形式的一类非凸－强凹极小极大优化问题[14]：

$$\min_{[x_1;x_2]\in\mathbb{R}^{d_x}}\max_{y\in\mathbb{R}^{d_y}}L_1\left(x_1,x_2\right)-q\cdot L_2\left(x_1,y\right),$$

其中

$$L_1\left(x_1,x_2\right)=\frac{1}{N_{\mathcal{S}}}\sum_{i=1}^{N_{\mathcal{S}}}\theta\left(x_2;\Phi\left(x_1;a_i^{\mathcal{S}}\right),b_i^{\mathcal{S}}\right),$$

是监督学习损失总和，且有分类损失总和：

$$L_2\left(x_1,y\right)=\frac{1}{N_{\mathcal{S}}}\sum_{i=1}^{N_{\mathcal{S}}}D_{\mathcal{S}}\left(h\left(y;\Phi\left(x_1;\boldsymbol{a}_i^{\mathcal{S}}\right)\right)\right)-\frac{1}{N_{\mathcal{T}}}\sum_{i=1}^{N_{\mathcal{T}}}D_{\mathcal{T}}\left(h\left(y;\Phi\left(x_1;\boldsymbol{a}_i^{\mathcal{T}}\right)\right)\right)+\lambda\|y\|^2,$$

其中

$$\left(\boldsymbol{y};z\right)=1\Big/\left(1+\exp\left(-\boldsymbol{y}^{\top}z\right)\right),$$

$$D_{\mathcal{S}}\left(z\right)=1-\log\left(z\right),$$

以及

$$D_{\mathcal{T}}\left(z\right)=\log\left(1-z\right)$$

作为 L_2 的常用逻辑损失函数。这里 Φ 是一个以 x_1 为参数，大小为(28 × 28) × 200 的单层神经网络；θ 是一个以 x_2 为参数，大小为 200 × 20 × 10 的三层神经网络。神经网络中选择 sigmoid 函数作为激活函数，并选择 $q=1,\lambda=0.2$ 作为模型参数，算法 GDA 和算法 A-MCN 中 AGD 步的学习速率设为 $\left\{c\cdot10^{-i}:c\in\{1,5\},i\in\{1,2,3,4,5\}\right\}$，由于在实验前无法获取 $Q(x)$ 的显示解，因此在实验中使用 AGD 算法预估了 $Q(x)=\max_y f\left(x,y\right)$ 的值。

实验中用到的数据集为 MNIST [15]和 MNIST-M 数据集[14]。MNIST 数据集包含 60,000 个用于训练的示例和 10000 个用于测试的示例，其中包含的手写数字图像已经过尺寸标准化且位于图像中心，图像是固定大小(28 × 28 像素)，其值为 0 到 1。并且每个图像都被平展并转换为 784 (28 × 28)个特征的一维 numpy 数组。MNIST-M 数据集是由 MNIST 中的数字图像与 BSDS500 数据集中的随机色块混合而成。

5.1. 参数对算法的影响分析

超参数的调节是影响机器学习算法数值效果的重要因素，合适的超参数设置往往能够提升算法对数据的表达效果，然而参数的调节是很困难的。在工程实践中，通常利用数值实验的方式确定算法最佳超参数。考虑到算法 A-MCN 中参数 η_1,η_2 以及求解三次正则化牛顿法子问题的 Lanczos 方法终止步条件对算法 A-MCN 的收敛性影响较大，这里的 η_1,η_2 为算法中检验牛顿迭代步是否成功的判断标准，且 $1>\eta_2>\eta_1>0$，Lanczos 方法在每次迭代中的终止步为记为 *Lanmax*。

本节数值实验目的是，分析算法 A-MCN 中的几个参数选取对算法性能的影响。在实验中选择精度

$\varepsilon = 1e-4$。固定$\eta_2 = 0.8$， *Lanmax* 设为 5，分析参数η_1对算法 A-MCN 性能的影响。如表 1 所示。可以看到在最终迭代步时，算法 A-MCN 选取参数$\eta_1 = 0.1$的损失函数值均低于其他情形下的损失函数值，且η_1选取在 0.2~0.5 之间对算法的性能影响不大。因此算法 A-MCN 选取参数η_1的最优值为 0.1。

Table 1. Analysis of the influence of parameters η_1 on the performance of algorithm A-MCN
表 1. 参数η_1对算法 A-MCN 性能影响分析

$\varepsilon = 1e-4$ 参数选取	**A-MCN** $\eta_1 = 0.1$	A-MCN $\eta_1 = 0.2$	A-MCN $\eta_1 = 0.3$	A-MCN $\eta_1 = 0.4$	A-MCN $\eta_1 = 0.5$
迭代次数	损失函数值	损失函数值	损失函数值	损失函数值	损失函数值
10,752	**8.1642885**	8.294297	8.294297	8.388242	8.388242

接着固定上轮得到的最优参数$\eta_1 = 0.1$， *Lanmax* 设为 5，分析参数η_2对算法 A-MCN 性能的影响。如表 2 所示。可以看到在在最终迭代步时，算法 A-MCN 选取参数$\eta_2 = 0.8$的损失函数值均低于其他情形下的损失函数值，且η_2选取在 0.5~0.7 之间对算法的性能影响不大。因此算法 A-MCN 选取参数η_2的最优值为 0.8。

Table 2. Analysis of the influence of parameters η_2 on the performance of algorithm A-MCN
表 2. 参数η_2对算法 A-MCN 性能影响分析

$\varepsilon = 1e-4$ 参数选取	A-MCN $\eta_2 = 0.5$	A-MCN $\eta_2 = 0.6$	A-MCN $\eta_2 = 0.7$	**A-MCN** $\eta_2 = 0.8$	A-MCN $\eta_2 = 0.9$
迭代次数	损失函数值	损失函数值	损失函数值	损失函数值	损失函数值
10,752	8.18876	8.222577	8.222577	**8.1642885**	8.226245

最后固定选取的最优参数$\eta_1 = 0.1, \eta_2 = 0.8$，分析参数 *Lanmax* 对算法 A-MCN 性能的影响，如表 3 所示，可以看到在最终迭代步时，算法 A-MCN 选取参数 *Lanmax* = 5 的损失函数值均低于其他情形下的损失函数值，因此在固定最优参数$\eta_1 = 0.1, \eta_2 = 0.8$下，参数 *Lanmax* 最优选取为 5。在通过本节数值实验之后，本文选取的最佳参数为：$\eta_1 = 0.1, \eta_2 = 0.8, Lanmax = 5$。

Table 3. Analysis of the influence of parameters *Lanmax* on the performance of algorithm A-MCN
表 3. 参数 *Lanmax* 对算法 A-MCN 性能影响分析

$\varepsilon = 1e-4$ 参数选取	A-MCN *Lanmax* = 3	A-MCN *Lanmax* = 4	**A-MCN** *Lanmax* = 5	A-MCN *Lanmax* = 6	A-MCN *Lanmax* = 7
迭代次数	损失函数值	损失函数值	损失函数值	损失函数值	损失函数值
10,752	8.62405	8.588853	**8.1642885**	8.304988	8.588853

5.2. 正文算法对比实验

本节对比算法 A-MCN、随机梯度上升下降算法(SGDA) [16]和极小极大随机三次正则化牛顿法(MSCR) [12]在两个不同的数据集：MNIST 和 MNIST-M 之间的自适应学习问题，由前文参数对算法 A-MCN 性能影响实验可知，算法 A-MCN 中参数$\eta_1 = 0.1, \eta_2 = 0.8, Lanmax = 5$。并将算法 A-MCN、算法 SGDA 和算法 MSCR 计算得到的损失函数值与迭代次数进行对比，数值结果如图 1 和图 2 所示。图 1 为

数据集 MNIST 到数据集 MNIST-M 的自适应学习结果，图 2 为数据集 MNIST-M 到数据集 MNIST 的自适应学习结果，在两次实验中均通过迭代次数来比较算法 A-MCN 和算法 SGDA、算法 MSCR 的性能。无论是图 1 还是图 2 均可以看到，在达到相同计算精度时，算法 A-MCN 在预测比较中显著优于算法 SGDA 和算法 MSCR。

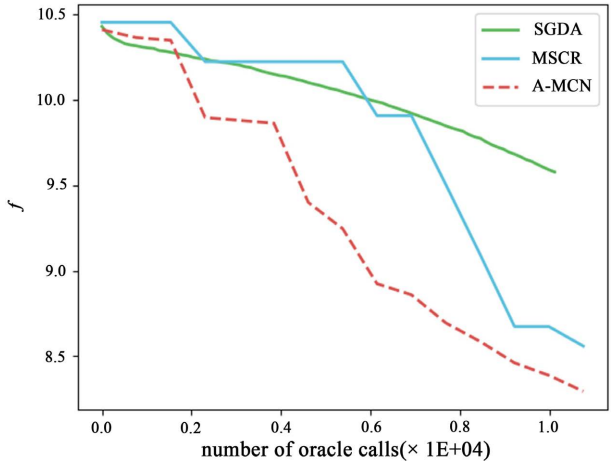

Figure 1. Adaptive learning results from MNIST to MNIST-M
图 1. MNIST 到 MNIST-M 的自适应学习结果

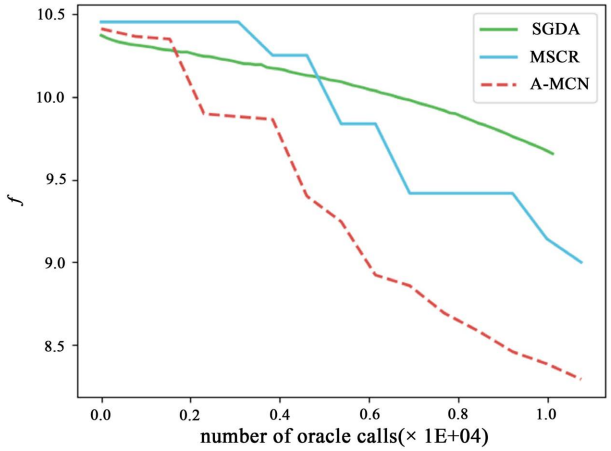

Figure 2. Adaptive learning results from MNIST-M to MNIST
图 2. MNIST-M 到 MNIST 的自适应学习结果

6. 总结及展望

首先对本文的工作进行总结，然后对未来的进一步工作进行展望。

6.1. 总结

针对非凸－强凹极小极大优化问题，采用信赖域半径的选取策略，首先提出了一类自适应极小极大三次正则化牛顿法(A-MCN)，其中针对外层循环设计了一种自适应的三次正则化牛顿法，针对内层子问题设计了一种加速梯度算法。

然后，得到了算法的迭代复杂度为$\mathcal{O}\left(T^{-1/3}\right)$。最后，通过 DANN 中的一类对抗攻击神经网络问题验

证了该算法的有效性。相比于文献[12]提出的算法 MCN，本文提出的算法 A-MCN 能自适应调节三次正则化项的系数。

6.2. 展望

可行的未来研究工作主要有以下几个方面。

1) 在极小极大优化问题的算法设计中，提高算法的收敛速度是非常必要的。受文献[17]工作的启发，本文下一步研究的工作是借鉴文献[17]中的动量加速机制，来加速本文提出的自适应极小极大三次正则化牛顿法，并建立算法的收敛性理论。

2) 本文主要研究了针对非凸－凹极小极大优化问题的两类算法和理论，但针对非凸－非凹极小极大优化问题的相关算法和理论研究还有待进一步研究[18]。

参考文献

[1] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., *et al*. (2018) A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go through Self-Play. *Science*, **362**, 1140-1144. https://doi.org/10.1126/science.aar6404

[2] Young, T., Hazarika, D., Poria, S. and Cambria, E. (2018) Recent Trends in Deep Learning Based Natural Language Processing. *IEEE Computational Intelligence Magazine*, **13**, 55-75. https://doi.org/10.1109/MCI.2018.2840738

[3] Letcher, A., Balduzzi, D., Racaniere, S., Martens, J., Foerster, J., Tuyls, K. and Graepel, T. (2019) Differentiable Game Mechanics. *Journal of Machine Learning Research*, **20**, 3032-3071.

[4] Lu, S., Tsaknakis, I., Hong, M. and Chen, Y. (2020) Hybrid Block Successive Approximation for One-Sided Nonconvex Min-Max Problems: Algorithms and Applications. *IEEE Transactions on Signal Processing*, **68**, 3676-3691. https://doi.org/10.1109/TSP.2020.2986363

[5] Sanjabi, M., Ba, J., Razaviyayn, M. and Jason, D.L. (2018) On the Convergence and Robustness of Training Gans with Regularized Optimal Transport. *Proceedings of the* 32*nd International Conference on Neural Information Processing Systems*, Montréal, 3-8 December 2018, 7091-7101.

[6] Nouiehed, M., Sanjabi, M., Huang, T., Lee, J.D. and Razaviyayn, M. (2019) Solving a Class of Nonconvex Min-Max Games Using Iterative First Order Methods. *Proceedings of the* 33*rd International Conference on Neural Information Processing Systems*, Vancouver, 8-14 December 2019, 14934-14942.

[7] Kong, W. and Monteiro, R.D.C. (2021) An Accelerated Inexact Proximal Point Method for Solving Nonconvex-Concave Min-Max Problems. *SIAM Journal on Optimization*, **31**, 2558-2585. https://doi.org/10.1137/20M1313222

[8] Lin, T., Jin, C. and Jordan, M.I. (2020) Near-Optimal Algorithms for Minimax Optimization. *Conference on Learning Theory PMLR*, 9-12 July 2020, 2738-2779.

[9] Nesterov, Y. and Polyak, B.T. (2006) Cubic Regularization of Newton Method and Its Global Performance. *Mathematical Programming*, **108**, 177-205. https://doi.org/10.1007/s10107-006-0706-8

[10] Cartis, C., Gould, N.I.M. and Toint, P.L. (2011) Adaptive Cubic Regularisation Methods for Unconstrained Optimization. Part I: Motivation, Convergence and Numerical Results. *Mathematical Programming*, **127**, 245-295. https://doi.org/10.1007/s10107-009-0286-5

[11] Cartis, C., Gould, N.I.M. and Toint, P.L. (2011) Adaptive Cubic Regularisation Methods for Unconstrained Optimization. Part II: Worst-Case Function- and Derivative-Evaluation Complexity. *Mathematical Programming*, **130**, 295-319. https://doi.org/10.1007/s10107-009-0337-y

[12] Luo, L., Li, Y. and Chen, C. (2022) Finding Second-Order Stationary Point in Nonconvex-Strongly-Concave Minimax Optimization. 36*th Conference on Neural Information Processing Systems* (*NeurIPS* 2022), New Orleans, 28 November-9 December 2022, 36667-36679.

[13] Lin, T., Jin, C. and Jordan, M.I. (2020) On Gradient Descent Ascent for Nonconvex-Concave Minimax Problems. *International Conference on Machine Learning*. *PMLR*, 13-18 July 2020, 6083-6093.

[14] Ganin, Y., Ustinova, E., Ajakan, H., *et al*. (2016) Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, **17**, 2096-2030.

[15] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, **86**, 2278-2324. https://doi.org/10.1109/5.726791

[16] Geerts, D., Nouwen, M., Beek, E.V., Slegers, K., Miranda, F.C. and Bleumers, L. (2019) Using the SGDA Framework

to Design and Evaluate Research Games. *Simulation and Gaming*, **50**, 272-301. https://doi.org/10.1177/1046878118808826

[17] Wang, Z., Zhou, Y., Liang, Y. and Lan, G. (2020) Cubic Regularization with Momentum for Nonconvex Optimization. *Uncertainty in Artificial Intelligence*. *PMLR*, 3-6 August 2020, 313-322.

[18] 徐姿, 张慧灵. 非凸极小极大优化问题的优化算法与复杂度分析[J]. 运筹学学报, 2021, 25(3): 74-86.