

基于混合搜索方向的前向复值神经网络的学习算法

黄鹤[†], 张永亮

(苏州大学 电子信息学院, 江苏省 苏州市 215006)

摘要: 复值有限内存BFGS(CL-BFGS)算法能有效用于求解复数域的无约束优化问题,但其性能容易受到记忆尺度的影响.为了解决记忆尺度的选择问题,本文提出了一种混合搜索方向的CL-BFGS算法.其基本思想是,对于给定的记忆尺度候选集,采用滑动窗口法将其划分成有限个子集,并将每个子集中的元素作为记忆尺度计算得到一组混合方向,然后选择使目标函数值最小的混合方向作为当前迭代的搜索方向.在迭代过程中,采用混合搜索方向的策略有益于强化对最新曲率信息的利用,便于记忆尺度的选取,提高算法的收敛速度.本文提出的CL-BFGS算法适用于多层前向复值神经网络的高效学习.最后,通过在模式识别、非线性信道均衡和复函数逼近上的实验,验证了基于混合搜索方向的CL-BFGS算法能取得比一些已有算法更好的性能.

关键词: 前向复值神经网络; 复值L-BFGS算法; 记忆尺度; 混合搜索方向; 曲率; 高效学习

中图分类号: TP183

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.2213

引用格式: 黄鹤,张永亮. 基于混合搜索方向的前向复值神经网络的学习算法[J]. 控制与决策.

Hybrid search direction based learning algorithm for feedforward complex-valued neural networks

HUANG He[†], ZHANG Yong-liang

(School of Electronics and Information Engineering, Soochow University, Suzhou 215006, China)

Abstract: Complex limited-memory BFGS (CL-BFGS) algorithm can be efficiently applied to solve unconstrained optimization problems in complex domain. However, its performance is seriously affected by memory size. In this paper, to deal with the selection problem of memory size, an improved CL-BFGS algorithm with hybrid search directions is proposed. Basically, the candidate set of memory size is divided into several parts by the sliding window method and a group of hybrid directions are constructed by considering the elements of each subset as potential memory sizes. Then the hybrid direction achieving the minimum value of objective function is taken as the actual search direction at the current iteration. The advantage of the strategy of hybrid search direction is to strengthen the usage of the latest curvature information and facilitate the choice of memory size such that the performance of the CL-BFGS algorithm is improved. The proposed CL-BFGS algorithm is then applied for the efficient learning of multi-layer feedforward complex-valued neural networks. Finally, experiments are conducted on the tasks of pattern recognition, nonlinear channel equalization and complex function approximation to verify that better performance is guaranteed by our algorithm than by some existing ones.

Keywords: feedforward complex-valued neural networks; complex L-BFGS algorithm; memory size; hybrid search direction; curvature; efficient learning

0 引言

随着现代信息技术和大数据科学的不断发展,人工神经网络得到了越来越广泛的关注,在各种工程领域取得了非常成功的应用^[1-3].目前,人们对人工神经网络的研究大多集中于实值神经网络模型.已经知道,复值神经网络在很多实际应用中能发挥更好的作用.比如,在信号处理领域,需要同时考虑信号的幅度

和相位信息.如果将幅度与相位分别作为实值神经网络的输入进行处理,不能体现幅度与相位之间的内在关系,难以达到满意的效果^[4].在合成孔径雷达(SAR)图像处理与去噪中,需要充分利用相位信息,从而在数据重建和去噪时取得更准确的效果^[5-6].又比如在通信系统中,为了消除码间串扰和噪声干扰,需要在接收端设计非线性信道均衡器恢复原始信号.相位

收稿日期: 2021-12-22; 录用日期: 2022-05-17.

基金项目: 江苏省自然科学基金项目(BK20181431),江苏省“青蓝工程”资助.

[†]通讯作者. E-mail: hhuang@suda.edu.cn.

信息的合理利用同样能有效提高信道均衡器的性能^[7].近年来,复值神经网络已经在信号处理、图像处理等方面取得了广泛的应用.这极大地促进了复值神经网络的研究^[8-12].

前向复值神经网络的性能在很大程度上会受到学习算法的影响.根据所利用的目标函数的导数信息,前向复值神经网络的学习算法可分为两类:一阶复值优化算法和二阶复值优化算法^[13].一阶复值优化算法(如复梯度下降算法)只需要利用目标函数的一阶Wirtinger导数^[14]信息,易于计算,在实现过程中占用的内存单元少.但是,一阶复值优化算法往往收敛速度慢,容易陷入局部极小值点.一种克服这些不足的方法是采用复步长^[15].文献[15]探讨了复步长对复梯度下降算法的影响,相比于实步长,复步长能取得更快的收敛速度.文献[16]通过引入松弛参数解决了前向复值神经网络学习过程中自适应复步长的设计问题.

另一种解决方法是采用二阶复值优化算法训练前向复值神经网络.作为二阶优化算法的典型代表,Broyden-Fletcher-Goldfarb-Shanno(BFGS)算法被广泛用于无约束优化问题的求解^[17].为了避免牛顿法中海塞矩阵的计算,BFGS算法借助割线方程建立了近似海塞矩阵的更新公式,在保证算法收敛速度的同时有效减少了计算量.但是,与牛顿法一样,BFGS算法需要存储 n 阶对称矩阵,其中 n 为决策变量个数,从而在求解大规模优化问题时需要占用大量的内存单元.为了解决这一问题,提出了有限内存的BFGS(L-BFGS)算法^[17].在该算法中,只需要存储和利用一组向量对就可以实现近似海塞矩阵的计算,而且搜索方向能够采用两循环方式计算得到,有效降低了算法的空间复杂度.最近,L-BFGS算法已被推广到复数域,用于解决以复变量为决策变量的优化问题^[13].值得注意的是,研究表明,L-BFGS算法的性能容易受到所存储的向量对组数(即L-BFGS算法的记忆尺度)的影响.文献[18]首次研究了L-BFGS算法的记忆尺度选择问题,提出了两种自适应方法.但是,据了解,目前对复值L-BFGS(CL-BFGS)算法的记忆尺度选择问题的研究还很少,仍有一些有意义的问题值得探索.

另一方面,大多数优化算法都是沿着某一特定方向进行搜索.比如,梯度下降法总是采用当前位置的负梯度作为搜索方向,BFGS算法则一直都是沿着近似牛顿方向进行迭代.我们的实验结果表明,在迭代过程中,始终沿特定方向进行搜索并非总是最优的

策略,适当改变搜索方向可能会取得更好的效果.滑动窗口法常用于解决连续区间上满足一定条件的查找问题.当窗口每向前滑动一次,都会有一个新的数据进入到滑动窗内,同时有一个旧数据滑出.在传统的CL-BFGS算法中引入这一思想有益于记忆尺度的选取,同时可以强化对最新曲率信息的合理利用,提高算法的性能.

基于上述讨论,为了解决记忆尺度的选择问题,本文提出了一种混合搜索方向的CL-BFGS算法.具体地,该算法包含两个阶段:候选混合方向的计算和实际搜索方向的确定.在第一阶段中,对于给定的记忆尺度的候选集,采用滑动窗口法将其划分为不同的子集,将子集中的每个元素作为潜在的记忆尺度计算相应的方向,通过叠加得到该子集上的混合方向.依次进行,可以得到每个子集上的混合方向.但是,在算法实现过程中,只能选择一个混合方向作为搜索方向进行迭代.因此,第二阶段需要从备选的混合方向中选择最合适的作为实际搜索方向.为此,采用复值强Wolfe线搜^[13]确定各混合方向的步长,进而沿各混合方向计算目标函数值,选择使目标函数值下降最多的混合方向作为当前迭代的搜索方向.本文提出的混合搜索方向的CL-BFGS算法主要有以下优点:(1)混合搜索方向策略强化了在迭代过程中对最新曲率信息的利用,提高了CL-BFGS算法的性能和收敛速度;(2)采用滑动窗口法对候选集进行划分,有利于为CL-BFGS算法选取合适的记忆尺度;(3)混合搜索方向的CL-BFGS算法可以有效用于多层前向复值神经网络的学习.为了验证算法的有效性,我们在模式分类、非线性信道均衡和复函数逼近上做了大量的实验.实验结果表明,混合搜索方向的CL-BFGS算法能够取得比一些已有算法更好的性能.

1 预备知识

1.1 前向复值神经网络

考虑一个 $L + 1$ 层的前向复值神经网络,其中第0层为输入层,第 L 层为输出层,中间 $L - 1$ 层为隐层.假设第 $l + 1$ 层与第 l 层神经元之间的连接权值矩阵为 $\mathbf{W}_{l+1} \in \mathbb{C}^{n_{l+1} \times n_l}$, $\mathbf{b}_{l+1} \in \mathbb{C}^{n_{l+1}}$ 和 $f_{l+1}(z)$ 分别表示第 $l + 1$ 层神经元的偏置所构成的复向量和激励函数, n_l 为第 l 层神经元个数, $\mathbf{h}_{l+1} \in \mathbb{C}^{n_{l+1}}$ 表示第 $l + 1$ 层神经元的输出向量.通过前向传播,采用递归方式分别计算各层神经元的输出,并最终得到该前向复值神经网络的输出.因此,对于 $l =$

0, 1, 2, ..., L-1, 第l+1层神经元的输出可表示为

$$\mathbf{h}_{l+1} = f_{l+1}(\mathbf{W}_{l+1}\mathbf{h}_l + \mathbf{b}_{l+1}), \quad (1)$$

其中, $\mathbf{h}_0 \in \mathbb{C}^{n_0}$ 为前向复值神经网络的输入向量. 特别地, 当 $l = L-1$ 时, 得到前向复值神经网络的输出 $\mathbf{o} = f_L(\mathbf{W}_L\mathbf{h}_{L-1} + \mathbf{b}_L)$.

对于前向复值神经网络, 通常采用监督学习的方式进行训练. 给定N组训练样本 $\{(\mathbf{z}^n, \bar{\mathbf{o}}^n)\}_{n=1}^N$, $\mathbf{z}^n = [z_1^n, z_2^n, \dots, z_{n_0}^n]^T \in \mathbb{C}^{n_0}$ 和 $\bar{\mathbf{o}}^n \in \mathbb{C}^{n_L}$ 分别表示第n个训练样本及对应的期望输出. 在训练过程中, 目标函数定义为如下的均方误差函数:

$$J(\theta, \theta^*) = \frac{1}{2N} \sum_{n=1}^N (\bar{\mathbf{o}}^n - \mathbf{o}^n)^H (\bar{\mathbf{o}}^n - \mathbf{o}^n), \quad (2)$$

其中, 上标*和H分别表示复向量或复矩阵的共轭和共轭转置运算, $\mathbf{o}^n$ 表示第n个训练样本的实际输出, $\theta \in \mathbb{C}^M$ 表示前向复值神经网络所有权值和偏置构成的向量, $M = \sum_{l=1}^L n_l(n_{l-1} + 1)$.

1.2 CL-BFGS算法

CL-BFGS算法^[13]已被广泛用于求解带复变量的实值函数优化问题. 考虑复数域无约束优化问题

$$\min_{\mathbf{z} \in \mathbb{C}^n} f(\mathbf{z}, \mathbf{z}^*), \quad (3)$$

在第t次迭代时, 其搜索方向可表示为

$$\hat{\mathbf{d}}_t = -\mathbf{H}_t \hat{\mathbf{g}}_t, \quad (4)$$

其中, $\hat{\mathbf{d}}_t = [\mathbf{d}_t^T, \mathbf{d}_t^H]^T \in \mathbb{C}^{2n}$, 上标T表示转置运算, $\mathbf{H}_t \in \mathbb{C}^{2n \times 2n}$ 为近似复海塞矩阵的逆矩阵, $\hat{\mathbf{g}}_t = [2\nabla_{\mathbf{z}^*} f(\mathbf{z}_t, \mathbf{z}_t^*), 2\nabla_{\mathbf{z}} f(\mathbf{z}_t, \mathbf{z}_t^*)]^T \in \mathbb{C}^{2n}$ 为目标函数 $f(\mathbf{z}, \mathbf{z}^*)$ 关于变量 $\mathbf{z}$ 和 $\mathbf{z}^*$ 的Wirtinger梯度^[14].

与L-BFGS算法类似, 为了计算 $\mathbf{H}_t$, CL-BFGS算法需要存储有限组向量对 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-m}^{t-1}$, 这里 $\hat{\mathbf{s}}_i = \hat{\mathbf{z}}_{i+1} - \hat{\mathbf{z}}_i$, $\hat{\mathbf{y}}_i = \hat{\mathbf{g}}_{i+1} - \hat{\mathbf{g}}_i$, $\hat{\mathbf{z}}_i = [\mathbf{z}_i^T, \mathbf{z}_i^H]^T \in \mathbb{C}^{2n}$. 定义 $\rho_t = \frac{1}{\hat{\mathbf{y}}_t^H \hat{\mathbf{s}}_t}$, $\mathbf{V}_t = \mathbf{I} - \rho_t \hat{\mathbf{y}}_t \hat{\mathbf{s}}_t^H$ 以及 $\mathbf{H}_t^0 = \frac{\hat{\mathbf{y}}_{t-1}^H \hat{\mathbf{s}}_{t-1}}{\hat{\mathbf{y}}_{t-1}^H \hat{\mathbf{y}}_{t-1}} \mathbf{I}$, I 为单位矩阵, 则矩阵 $\mathbf{H}_t$ 可表示为

$$\begin{aligned} \mathbf{H}_t = & (\mathbf{V}_{t-1}^H \dots \mathbf{V}_{t-m}^H) \mathbf{H}_t^0 (\mathbf{V}_{t-m} \dots \mathbf{V}_{t-1}) \\ & + \rho_{t-m} (\mathbf{V}_{t-1}^H \dots \mathbf{V}_{t-m+1}^H) \hat{\mathbf{s}}_{t-m} \\ & \times \hat{\mathbf{s}}_{t-m}^H (\mathbf{V}_{t-m+1} \dots \mathbf{V}_{t-1}) \\ & + \rho_{t-m+1} (\mathbf{V}_{t-1}^H \dots \mathbf{V}_{t-m+2}^H) \hat{\mathbf{s}}_{t-m+1} \\ & \times \hat{\mathbf{s}}_{t-m+1}^H (\mathbf{V}_{t-m+2} \dots \mathbf{V}_{t-1}) \\ & + \dots + \rho_{t-1} \hat{\mathbf{s}}_{t-1} \hat{\mathbf{s}}_{t-1}^H. \end{aligned} \quad (5)$$

由式(4)-(5)可得, 第t次迭代时无约束优化问题(3)的参数更新公式为

$$\hat{\mathbf{z}}_{t+1} = \hat{\mathbf{z}}_t + \eta_t \hat{\mathbf{d}}_t, \quad (6)$$

其中, η_t 表示步长, 可采用线搜(如复值强Wolfe线搜^[13])方法求得. 在CL-BFGS算法的实现中, 利用双循环策略可以直接由 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-m}^{t-1}$ 得到搜索方向, 避免了矩阵 $\mathbf{H}_t$ 的计算与存储, 从而节省了存储空间.

2 基于混合搜索方向的CL-BFGS算法

正如前面所述, 在计算搜索方向时, 需要利用和存储m组向量对 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-m}^{t-1}$. 这就意味着CL-BFGS算法的性能容易受到记忆尺度m的影响. 研究表明, 并非越大的记忆尺度总能保证越好的性能, 有时大的记忆尺度反而会降低其性能. 因此, 根据实际问题, 如何为CL-BFGS算法选择合理的记忆尺度是值得研究的. 另一方面, 在CL-BFGS算法的迭代过程中, 与当前迭代越接近的向量对 $(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)$ 可能会隐含更多与当前状态相关的曲率信息^[17]. 在构建搜索方向时, 如果能够强化对这些曲率信息的利用, 则会提升CL-BFGS算法的性能. 基于这些考虑, 本文提出了一种基于混合搜索方向的CL-BFGS算法, 在有效加强对最新曲率信息利用的同时, 解决记忆尺度的选择问题.

基于混合搜索方向的CL-BFGS算法包括两个阶段. 第一阶段采用滑动窗口技术计算候选搜索方向; 第二阶段根据目标函数值的变化趋势确定实际的搜索方向.

2.1 候选搜索方向的计算

假设记忆尺度m的候选集合为 $\{1, 2, \dots, M\}$, 其中M为预先设定的正整数. 在第t次迭代时, 采用大小为 τ ($\tau \leq M$) 的滑动窗口对候选集合进行划分, 得到 $M-\tau+1$ 个子集 $\{k, k+1, \dots, k+\tau-1\}_{k=1}^{M-\tau+1}$. 考虑任一子集 $\{k, k+1, \dots, k+\tau-1\}$, 选择所有元素作为可能的记忆尺度. 根据式(4)-(5)分别计算方向, 得到 $\hat{\mathbf{d}}_{t,k}, \hat{\mathbf{d}}_{t,k+1}, \dots, \hat{\mathbf{d}}_{t,k+\tau-1}$. 在此基础上, 构建子集 $\{k, k+1, \dots, k+\tau-1\}$ 上的混合候选方向

$$\hat{\mathbf{d}}_{t,k}^{hd} = \sum_{j=k}^{k+\tau-1} \hat{\mathbf{d}}_{t,j}. \quad (7)$$

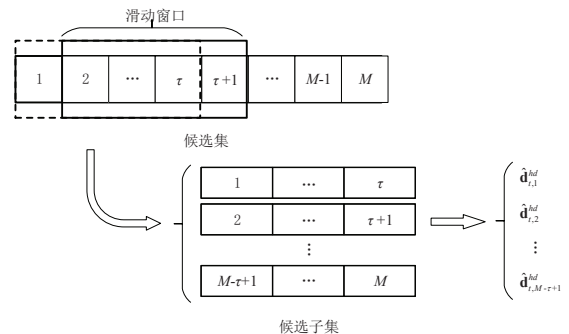


图1 候选搜索方向的计算过程

因此,采用滑动窗口技术,可以得到一组候选搜索方向 $\{\hat{\mathbf{d}}_{t,1}^{hd}, \hat{\mathbf{d}}_{t,2}^{hd}, \dots, \hat{\mathbf{d}}_{t,M-\tau+1}^{hd}\}$.候选搜索方向的计算过程如图1所示.

值得一提的是,在每个混合候选方向的计算中,连续选择了多个参数作为可能的记忆尺度.比如,为了得到 $\hat{\mathbf{d}}_{t,k}^{hd}$,分别将 $k, k+1, \dots, k+\tau-1$ 作为记忆尺度计算 $\hat{\mathbf{d}}_{t,j}$ ($j = k, k+1, \dots, k+\tau-1$).其目的是在构建搜索方向时强化对最新曲率信息的合理利用,提高CL-BFGS算法的性能.

为了进一步说明,不妨考虑一种简单情况.假设选取的记忆尺度分别为 m_1, m_2 ($m_1 < m_2$).根据式(4)-(5),可分别求得两个不同的方向 $\hat{\mathbf{d}}_{t,m_1}$ 和 $\hat{\mathbf{d}}_{t,m_2}$,进而构造出混合方向 $\hat{\mathbf{d}}_t^{hd} = \hat{\mathbf{d}}_{t,m_1} + \hat{\mathbf{d}}_{t,m_2}$.事实上,由式(5)可知, $\hat{\mathbf{d}}_{t,m_1}$ 和 $\hat{\mathbf{d}}_{t,m_2}$ 是分别利用 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-1}^{t-m_1}$ 和 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-1}^{t-m_2}$ 得到的.显然,这两组向量对同时包含了 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-1}^{t-m_1}$,差别在于前一组向量对不包含 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-m_1-1}^{t-m_2}$.也就是说,在 $\hat{\mathbf{d}}_{t,m_1}$ 和 $\hat{\mathbf{d}}_{t,m_2}$ 中都利用了向量对 $\{(\hat{\mathbf{s}}_i, \hat{\mathbf{y}}_i)\}_{i=t-1}^{t-m_1}$ 的信息.从而,在混合方向 $\hat{\mathbf{d}}_t^{hd}$ 中突出了最新曲率信息的重要性,可以有效提高CL-BFGS算法的性能.

2.2 实际搜索方向的确定

在第一阶段,利用滑动窗口技术和式(7)得到了一组候选混合搜索方向 $\{\hat{\mathbf{d}}_{t,k}^{hd}\}_{k=1}^{M-\tau+1}$.在算法实现时,我们需要从中选择一个最优的混合方向作为实际搜索方向,即剔除候选集中不好的数据.为此,在当前迭代处 $\hat{\mathbf{z}} = \hat{\mathbf{z}}_t$,针对每个混合方向 $\hat{\mathbf{d}}_{t,k}^{hd}$,采用复值强Wolfe线搜确定步长 $\eta_{t,k}$,即 $\eta_{t,k}$ 满足以下条件:

$$f(\hat{\mathbf{z}}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd}) \leq f(\hat{\mathbf{z}}_t) + c_1 \eta_{t,k} \left(\hat{\mathbf{d}}_{t,k}^{hd} \right)^T \frac{\partial f(\hat{\mathbf{z}}_t)}{\partial \hat{\mathbf{z}}}, \quad (8)$$

$$\left| \left(\hat{\mathbf{d}}_{t,k}^{hd} \right)^T \frac{\partial f(\hat{\mathbf{z}}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd})}{\partial \hat{\mathbf{z}}} \right| \leq c_2 \left| \left(\hat{\mathbf{d}}_{t,k}^{hd} \right)^T \frac{\partial f(\hat{\mathbf{z}}_t)}{\partial \hat{\mathbf{z}}} \right|, \quad (9)$$

其中, c_1, c_2 为满足 $0 < c_1 < c_2 < 1$ 的常数.

对 $k = 1, 2, \dots, M - \tau + 1$,分别计算目标函数值 $f(\hat{\mathbf{z}}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd})$,并将其从小到大进行排序.令

$$k^{opt} = \arg \min_k \left\{ f(\hat{\mathbf{z}}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd}) \right\}, \quad (10)$$

选择 $\hat{\mathbf{d}}_{t,k^{opt}}^{hd}$ 和 $\eta_{t,k^{opt}}$ 作为当前迭代的实际搜索方向和步长.也就是说,第 k^{opt} 个子集的元素 $\{k^{opt}, k^{opt} + 1, \dots, k^{opt} + \tau - 1\}$ 适合作为当前迭代时CL-BFGS算法的记忆尺度,因此被选择出来

用以计算混合搜索方向,而其它子集中的数据被舍弃.最终,复数域无约束优化问题(3)的参数更新公式为

$$\hat{\mathbf{z}}_{t+1} = \hat{\mathbf{z}}_t + \eta_{t,k^{opt}} \hat{\mathbf{d}}_{t,k^{opt}}^{hd}. \quad (11)$$

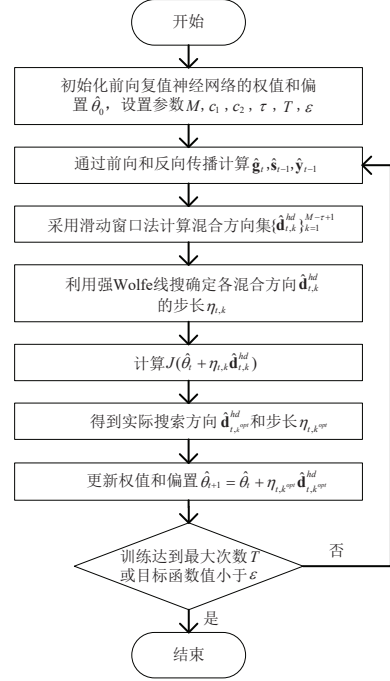


图2 基于混合搜索方向的前向复值神经网络训练流程图

3 前向复值神经网络的学习

上一节提出的基于混合搜索方向的CL-BFGS算法可用于多层前向复值神经网络的训练.考虑1.1节所述的前向复值神经网络模型,基于混合搜索方向的CL-BFGS算法的训练过程如下:

step 1: 初始化前向复值神经网络的权值和偏置 θ_0 ,设置参数,包括候选集上界 M 、窗口尺寸 τ 、复值强Wolfe线搜条件(8)-(9)中的参数 c_1 和 c_2 、最大训练次数 T 以及目标函数的阈值 ε .令 $t = 1$.

step 2: 利用前向和反向传播过程计算目标函数 $J(\theta, \theta^*)$ 关于 θ 和 θ^* 的Wirtinger梯度 $\hat{\mathbf{g}}_t$,计算 $\hat{\mathbf{s}}_{t-1} = \hat{\theta}_t - \hat{\theta}_{t-1}$ 和 $\hat{\mathbf{y}}_{t-1} = \hat{\mathbf{g}}_t - \hat{\mathbf{g}}_{t-1}$.

step 3: 采用大小为 τ 的滑动窗口对记忆尺度的候选集 $\{1, 2, \dots, M\}$ 进行划分,将每个子集的元素分别作为可能的记忆尺度,计算 τ 个方向 $\{\hat{\mathbf{d}}_{t,j}\}_{j=k}^{k+\tau-1}$,由式(7)得到一组候选混合搜索方向 $\{\hat{\mathbf{d}}_{t,k}^{hd}\}_{k=1}^{M-\tau+1}$.

step 4: 对每个混合搜索方向 $\hat{\mathbf{d}}_{t,k}^{hd}$,采用复值强Wolfe线搜确定步长 $\eta_{t,k}$.

step 5: 对 $k = 1, 2, \dots, M - \tau + 1$,分别计算目标函数值 $J(\hat{\theta}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd})$.

step 6: 令

$$k^{opt} = \arg \min_{k \in \{1, 2, \dots, M-\tau+1\}} \left\{ J \left(\hat{\theta}_t + \eta_{t,k} \hat{\mathbf{d}}_{t,k}^{hd} \right) \right\},$$

得到第 t 次迭代的搜索方向 $\hat{\mathbf{d}}_{t,k^{opt}}^{hd}$ 和步长 $\eta_{t,k^{opt}}$.

step 7: 更新前向复值神经网络的权值和偏置

$$\hat{\theta}_{t+1} = \hat{\theta}_t + \eta_{t,k^{opt}} \hat{\mathbf{d}}_{t,k^{opt}}^{hd}.$$

若达到最大训练次数 T 或者 $J(\hat{\theta}_{t+1}) \leq \varepsilon$, 则训练结束. 否则, 令 $t \leftarrow t + 1$, 返回step 2继续训练.

基于混合搜索方向的CL-BFGS算法的多层前向复值神经网络的训练过程如图2所示.

4 实验结果

为了验证本文所提出的基于混合搜索方向的CL-BFGS算法的有效性以及相比于其它算法的优势, 我们在模式分类、非线性信道均衡和复函数逼近等应用上做了大量实验, 其中模式分类的数据集来源于UCI机器学习数据库^[20], 具体信息见表1.

表1 模式分类数据集

数据集	类别数	特征数	训练样本数	测试样本数
Spambase	2	57	3221	1380
Biodeg	2	41	739	316
Glass	6	10	150	64
Ionosphere	2	34	246	105
Optical Digit	10	64	3823	1797
PIMA	2	8	538	230
Vehicle	4	18	592	254
Wine	3	13	125	53
Banana	2	2	3710	1590
Waveform21	3	21	3500	1500
Waveform40	3	40	3500	1500
Zoo	7	16	70	31
Liver Disorder	2	6	241	104
Page Blocks	5	10	3831	1642
Balance Scale	3	4	438	187

对于模式分类问题, 利用相位编码转换

$$z = \cos \theta + i \sin \theta \quad (12)$$

将实值特征转化为复值特征, 其中, $\theta = \frac{\pi(x-a)}{b-a}$, x 表示实值特征, i 为虚数单位, a, b 分别为实值特征 x 的上、下界.

在所有实验中, 目标函数(2)的Wirtinger梯度由 $\hat{\mathbf{g}}_t = \left[2\nabla_{\theta^*} J(\theta_t, \theta_t^*), 2\nabla_{\theta} J(\theta_t, \theta_t^*) \right]^T$ 计算得到, 复梯度下降算法(CGD)采用固定步长, 其余算法的步长均由式(8)-(9)得到.

4.1 参数分析

为了解决记忆尺度的选择问题, 在基于混合搜索方向的CL-BFGS算法中引入了两个参数 M 和 τ . 这两

个参数会直接影响算法的性能. 为此, 通过实验对它们进行分析.

4.1.1 候选集上界 M 的性能分析

候选集上界 M 表示了CL-BFGS算法记忆尺度的选择范围. M 值越大, 记忆尺度的选择范围越广. 相应地, CL-BFGS算法的性能通常会更好. 但是, 大的 M 值也意味着在候选搜索方向的计算过程中所需的计算量和存储单元都越多.

我们以模式分类的Wine数据集为例分析候选集上界 M 对混合搜索方向的CL-BFGS算法的影响. 在该实验中, 设计的模型为具有50个隐节点的三层前向复值神经网络, 激活函数为 $f(z) = \tanh(\text{Re}(z)) + i \tanh(\text{Im}(z))$, 其中 $\text{Re}(z)$ 和 $\text{Im}(z)$ 分别表示 z 的实部和虚部. 图3给出了目标函数 $J(\theta, \theta^*)$ 随候选集上界 M 的变化趋势, 这里 $\tau = 5$. 可以看出, 随着 M 的逐渐增大, 目标函数值总体上是逐渐减小的. 当 M 值较小时, 目标函数值有一个上升的变化. 这是因为在这种情况下CL-BFGS算法的性能难以完全体现, 从而可能导致目标函数值变大. 另一方面, 当 M 达到某个较大值后(比如15 ~ 20), 目标函数值减小的趋势并不明显, 意味着对混合搜索方向的CL-BFGS算法的性能没有产生明显的影响. 基于这些观察, 在实际应用中可以根据需要选择合适的 M 作为候选集上界, 以便在算法的性能与计算复杂度之间取得较好的折衷.

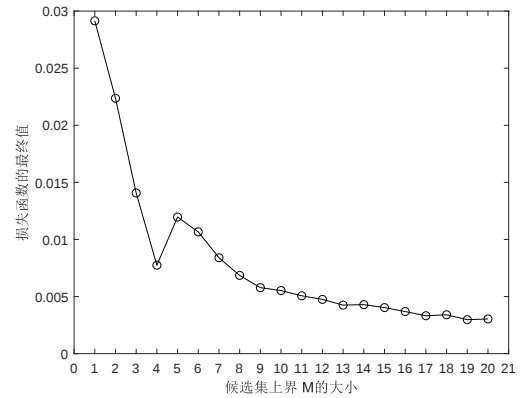


图3 目标函数值随候选集上界 M 的变化趋势

4.1.2 滑动窗口 τ 的性能分析

由式(7)知, 滑动窗口的大小 τ 直接关系到混合搜索方向的组合中方向的数量. 当 τ 取值太小时, 不能充分体现混合搜索方向策略的优势. 当 τ 太大时, 离当前迭代较远的曲率信息可能被过度利用, 使得最新曲率信息在构建混合搜索方向时占比减少, 从而不能有效强化最新曲率信息的利用, 影响算法的性能.

同样以Wine数据集为例. 图4给出了目标函数 $J(\theta, \theta^*)$ 随滑动窗口 τ 的变化趋势, 这里 $M = 20$. 由

图4的实验结果可以发现,随着 τ 的增大,目标函数值快速降低.当 $\tau > 13$ 时,目标函数值可能会上升,导致算法的性能减弱.这与我们前面的分析是一致的.因此,在实际中,相对于候选集的上界 M ,可以选择较小的正整数作为 τ 的值.

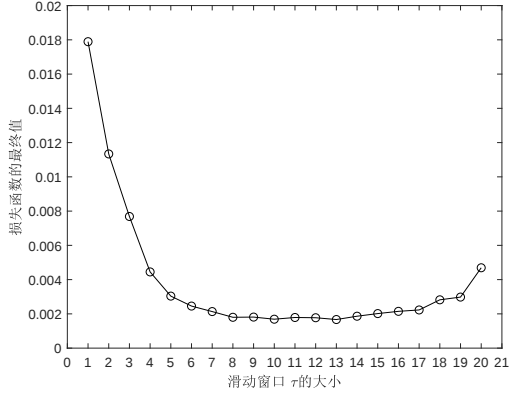


图4 目标函数值随滑动窗口 τ 的变化趋势

事实上,基于混合搜索方向的CL-BFGS算法中参数 M 和 τ 的选取是依赖于实际问题的.待解决问题的规模和复杂度不一样,应该选择不同的 M 和 τ .

4.2 性能比较

为了验证本文提出的混合搜索方向的CL-BFGS算法的有效性,结合非线性信道均衡、复函数逼近和模式分类实验,我们分别与CGD、复Barzilai-Borwein(CBBM)算法^[15]以及传统的CL-BFGS算法^[13]进行性能对比.为了方便,将混合搜索方向的CL-BFGS算法简记为CL-BFGS-HD.

4.2.1 非线性信道均衡

在通信系统中,信号在传输过程中容易受到外界噪声的干扰,需要在接收端设计信道均衡器以恢复原始信号.前向复值神经网络可有效用于非线性信道均衡器的设计^[7].考虑非线性信道模型^[19]:

$$y_n = u_n + 0.1u_n^2 + 0.05u_n^3 + v_n, \quad (13)$$

$$u_n = (0.34 - 0.27i)s_n + (0.87 + 0.43i)s_{n-1} + (0.34 - 0.21i)s_{n-2}, \quad (14)$$

其中, s_n 表示当前时刻的输入信号, y_n 为输出信号, s_{n-1} 和 s_{n-2} 分别为 s_n 延迟1个和2个单位的信号, v_n 表示均值为0、方差为0.01的高斯白噪声.输入信号 s_n 从 $\{-0.7 - 0.7i, -0.7 + 0.7i, 0.7 - 0.7i, 0.7 + 0.7i\}$ 中随机生成.

在该实验中,采用10个隐节点的三层前向复值神经网络构建信道均衡器,其输入为 $[y_n, y_{n-1}, y_{n-2}]^T$,对应的期望输出为 s_{n-2} .为了训练该前向复值神经网络,随机生成1000个训练

样本.图5的实验结果给出了输入信号的信噪比为20dB时不同训练算法下目标函数的变化趋势,其中CGD算法采用固定的复步长 $0.1 - 0.05i$,CL-BFGS算法的记忆尺度取为10,CL-BFGS-HD算法的候选集上界 M 和滑动窗口 τ 分别选为10和5.可以看出,与CGD、CBBM以及CL-BFGS算法相比,本文提出的CL-BFGS-HD算法能够找到更精确的解,且具有最快的收敛速度.

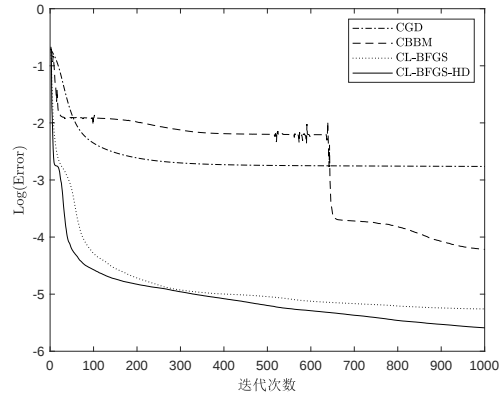


图5 信道均衡问题中不同算法的收敛曲线

为了更好验证本文提出的CL-BFGS-HD算法的有效性,我们测试了这些算法在输入信号的信噪比分别为10dB, 15dB, 20dB, 25dB, 30dB, 35dB和40dB情况下的训练误差.实验结果如表2所示.从中可以清楚地看到,在这7种不同信噪比下,我们提出的CL-BFGS-HD算法都得到了最小的训练误差.这些实验结果进一步表明CL-BFGS-HD算法能够取得比CGD、CBBM和CL-BFGS算法更好的性能.

表2 不同信噪比下各算法的训练误差

信噪比	CGD	CBBM	CL-BFGS	CL-BFGS-HD
10dB	0.1167	0.0829	0.0616	0.0571
15dB	0.0802	0.0357	0.0175	0.0127
20dB	0.0581	0.0093	0.0027	0.0014
25dB	0.0542	0.0039	0.0006	0.0002
30dB	0.0533	0.0038	6.0928e-04	6.4207e-05
35dB	0.0517	0.0032	3.8512e-04	6.2527e-05
40dB	0.0540	0.0288	5.0620e-04	6.6914e-05

4.2.2 复函数逼近

考虑如下的复函数逼近问题:

$$f(z_1, z_2) = \frac{1}{6} (z_1^2 + z_2^2), \quad (15)$$

其中, z_1, z_2 为复变量.为此,构建具有10个隐节点的三层前向复值神经网络,随机生成3000个训练样本, z_1, z_2 的实部和虚部分别从 $[-1, 1]$ 中选取.

图6给出了目标函数在不同训练算法下的收敛曲线,其中CGD算法的步长为0.01,CL-BFGS算法的

记忆尺度为10.在CL-BFGS-HD算法中,候选集上界为 $M = 10$,滑动窗口大小为 $\tau = 5$.由实验结果可知,相较于CGD、CBBM和CL-BFGS算法,本文提出的CL-BFGS-HD算法在复函数逼近问题(15)上同样具有最快的收敛速度,能够找到更加精确的解.

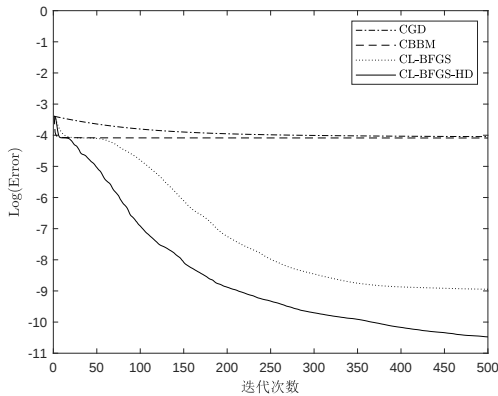


图6 复函数逼近问题中不同算法的收敛曲线

4.2.3 模式分类

为了比较不同训练算法的性能,我们以表1的模式分类问题为例做了相关实验,统计了不同训练算法取得的分类准确率,实验结果见表3.从中可以清楚地看到,和CGD算法相比,CL-BFGS-HD算法在这15个模式分类问题中都能取得更高的分类准确率.与CBBM算法相比,除了Waveform40数据集,采用CL-BFGS-HD算法训练的前向复值神经网络得到的分类准确率都有了明显的提升.和CL-BFGS算法相比,CL-BFGS-HD算法在其中11个数据集上取得了更高的分类准确率,在Wine数据集上的性能相同.因此,本文提出的CL-BFGS-HD算法在大多数模式分类数据集上能够取得更好的分类效果.

表3 不同训练算法在模式分类问题上的准确率(%)

数据集	CGD	CBBM	CL-BFGS	CL-BFGS-HD
Spambase	68.38	84.54	93.36	93.87
Biodeg	74.62	80.51	85.44	86.58
Glass	68.13	75.00	92.19	93.75
Ionosphere	90.67	93.71	95.05	95.43
Optical Digit	91.47	92.57	94.95	94.76
PIMA	62.87	72.09	71.83	72.52
Vehicle	70.31	77.24	82.91	82.76
Wine	95.47	96.60	96.98	96.98
Banana	58.08	57.22	83.55	89.20
Waveform21	84.79	85.92	85.88	86.13
Waveform40	86.28	86.88	86.77	86.61
Zoo	77.42	77.42	82.58	84.52
Liver Disorder	55.77	71.15	70.96	72.69
Page Blocks	85.38	86.42	87.99	88.40
Balance Scale	85.03	90.80	90.70	92.51

进一步,采用统计分析中的Wilcoxon符号秩检验^[21]对CL-BFGS-HD算法与CGD、CBBM和CL-BFGS算法进行两两对比,显著性水平为 $\alpha = 0.05$.表4给出了 R^+ 、 R^- 以及 p 值的结果.从中可以看出,对于这15个模式分类问题,本文提出的CL-BFGS-HD算法在整体上取得了比CGD、CBBM和CL-BFGS算法更好的效果.

表4 Wilcoxon符号秩检测结果

算法对比	R^+	R^-	p 值
CL-BFGS-HD vs CGD	120	0	0.00064
CL-BFGS-HD vs CBBM	119	1	0.00080
CL-BFGS-HD vs CL-BFGS	101.5	9.5	0.00413

5 结论

本文提出了一种混合搜索方向的策略解决CL-BFGS算法中记忆尺度的选择问题.对于给定的记忆尺度候选集,采用滑动窗口法对其进行划分,将每个子集的元素作为潜在的记忆尺度计算候选混合方向,然后根据目标函数值的变化趋势确定实际搜索方向.混合搜索方向策略的优势是在迭代过程中有益于强化最新曲率信息的利用,便于记忆尺度的选取,提升算法的性能.本文提出的基于混合搜索方向的CL-BFGS算法可用于解决多层前向复值神经网络的训练问题.通过模式分类、非线性信道均衡以及复函数逼近等应用,验证了混合搜索方向的CL-BFGS算法的有效性以及相比于一些已有算法的优势.

参考文献(References)

[1] Han Z, Qian X, Huang H, Huang T. Efficient design of multicolumn RBF networks[J]. Neurocomputing, 2021, 450: 253-263.

[2] 郭鑫, 李文静, 乔俊飞. 一种改进的在线自适应模块化神经网络[J]. 控制与决策, 2020, 35(7): 1597-1605. (Guo X, Li W J, Qiao J F. An improved online adaptive modular neural network[J]. Control and Decision, 2020, 35(7): 1597-1605.)

[3] 张伟, 黄卫民. 基于SAPSO算法的RBF神经网络设计[J]. 控制与决策, 2021, 36(9): 2305-2312. (Zhang W, Huang W M. Design of RBF neural network based on SAPSO algorithm[J]. Control and Decision, 2021, 36(9): 2305-2312.)

[4] Adali T, Schreier P J, Scharf L L. Complex-valued signal processing: The proper way to deal with impropriety[J]. IEEE Transactions on Signal Processing, 2012, 59(11): 5101-5125.

[5] 彭书娟, 曲长文, 李健伟. K 近邻优化估计的SAR图像建模与目标检测算法[J]. 控制与决策, 2020, 35(9): 2199-2206. (Peng S J, Qu C W, Li J W. K nearest neighbors optimized estimation algorithm for SAR image statistical

- modeling and target detection[J]. *Control and Decision*, 2020, 35(9): 2199-2206.)
- [6] 赵泉华, 高郡, 赵雪梅, 李玉. 结合KL距离与图像域分块的SAR图像分割[J]. *控制与决策*, 2018, 33(10): 1767-1774.
(Zhao Q H, Gao J, Zhao X M, Li Y. Segmentation of SAR image based on Kullback-Leibler distance and regular tessellation[J]. *Control and Decision*, 2021, 36(9): 2305-2312.)
- [7] Freire P J, Neskornuik V, Napoli A, et al. Complex-valued neural network design for mitigation of signal distortions in optical links[J]. *Journal of Lightwave Technology*, 2021, 39(6): 1696-1705.
- [8] Scardapane S, Vaerenbergh S V, Hussain A, et al. Complex-valued neural networks with nonparametric activation functions[J]. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2020, 4: 140-150.
- [9] 张昀, 于舒娟, 张志涌, 郭宇峰. 基于实虚型连续多值复数Hopfield神经网络的QAM盲检测[J]. *电子学报*, 2013, 41(2): 255-259.
(Zhang Y, Yu S J, Zhang Z Y, Guo Y F. Blind detection of QAM signals with a complex Hopfield neural network with real imaginary type soft multistate activation function[J]. *Acta Electronica Sinica*, 2013, 41(2): 255-259.)
- [10] 张代远. 复数前馈神经网络的全局最优和快速学习算法[J]. *电路与系统学报*, 2006, 11(4): 9-15.
(Zhang D Y. Algorithm with global minima and high learning speed for training complex-valued feedforward neural networks[J]. *Journal of Circuits and Systems*, 2006, 11(4): 9-15.)
- [11] Hirose A. *Complex-valued neural networks: Advances and applications*[M]. John Wiley & Sons, 2013.
- [12] Dong Z, Huang H. A training algorithm with selectable search direction for complex-valued feedforward neural networks[J]. *Neural Networks*, 2021, 137: 75-84.
- [13] Sorber L, Barel M V, Lathauwer L D. Unconstrained optimization of real functions in complex variables[J]. *SIAM Journal on Optimization*, 2012, 22(3): 879-898.
- [14] Kreutz-Delgado K. The complex gradient operator and the CR-calculus[J]. *arXiv preprint*, 2009, arXiv:0906.4835.
- [15] Zhang H, Mandic D P. Is a complex-valued stepsize advantageous in complex-valued gradient learning algorithms?[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, 27(12): 2730-2735.
- [16] Zhang Y, Huang H. Adaptive complex-valued stepsize based fast learning of complex-valued neural networks[J]. *Neural Networks*, 2020, 124: 233-242.
- [17] Nocedal J, Wright S. *Numerical optimization*. 2nd edition[M]. Springer, 2006.
- [18] Boggs P T, Byrd R H. Adaptive, limited-memory BFGS algorithms for unconstrained optimization[J]. *SIAM Journal on Optimization*, 2019, 29(2): 1282-1299.
- [19] Cha I, Kassam S A. Channel equalization using adaptive complex radial basis function networks[J]. *IEEE Journal on Selected Areas in Communications*, 1995, 13(1): 122-131.
- [20] Dua D, Graff C. *UCI Machine Learning Repository*. Irvine, CA: University of California, 2017. <http://archive.ics.uci.edu/ml>
- [21] Demsar J. Statistical comparisons of classifiers over multiple data sets[J]. *The Journal of Machine Learning Research*, 2006, 7: 1-30.

作者简介

黄鹤(1978—), 男, 教授, 博士, 从事神经网络、优化算法等研究, E-mail: hhuang@suda.edu.cn;

张永亮(1995—), 男, 硕士, 从事神经网络、模式识别的研究, E-mail: zylhard111@163.com.