

文章编号: 2095-2163(2022)10-0150-04

中图分类号: TP183

文献标志码: A

含有 L_{21} 范数正则化的在线顺序 RVFL 算法

季江飞, 郭久森

(浙江理工大学 信息学院, 杭州 310018)

摘要: 单隐层前馈神经网络(SLFN)以其量级轻、参数量少、训练成本低等优点,目前被广泛地运用于函数逼近处理、模式识别和控制领域中。随机向量函数连接网络(RVFL)作为 SLFN 的一种,能够将输入层与输出层做直接相连,加强输出层与输入层的关联。然而目前的预测任务中,已经训练好的网络在面对批量数据会随时间不断变化的情况时,则容易显露出泛化能力不足问题。为了提升网络的泛化能力,并防止重复训练,本文提出了一种在线顺序的 RVFL 算法,使用 L_{21} 范数实现正则化。在 UCI 数据集上经过对多种相关参数的最佳选择后,与同类型的 RVFL 算法和 LR_{21} - RVFL 算法相比,本文提出的 LR_{21} - OSRVFL 算法在多种评价指标下均有更优表现。

关键词: 单隐层前馈神经网络; 随机向量功能连接网络; 在线顺序; L_{21} 范数

Online sequential RVFL algorithm with L_{21} norm regularization

JI Jiangfei, GUO Jiusen

(School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China)

[Abstract] Single-hidden layer feedforward neural network (SLFN) is widely used in function approximation processing, pattern recognition and control due to its advantages of light weight, few parameters and low training cost. Random vector functional-link neural network (RVFL), as a kind of SLFN, can directly connect the input layer with the output layer to strengthen the association between the input layer and the output layer. However, in the current prediction task, the trained network is prone to show insufficient generalization ability in the face of batch data changing over time. In order to improve the generalization ability of networks and prevent repeated training, an online sequential RVFL algorithm is proposed in this paper, and is regularized by L_{21} norm. After the optimal selection of a variety of related parameters on UCI dataset, the LR_{21} - OSRVFL algorithm proposed in this paper has better performance under a variety of evaluation indexes compared with the same type of RVFL algorithm and LR_{21} - RVFL algorithm.

[Key words] single-hidden layer feedforward neural network (SLFN); random vector functional-link neural network (RVFL); online sequence; L_{21} norm

0 引言

迄今为止,在与机器学习有关的诸多研究方向^[1-2]中,人工神经网络正起着越来越重要的作用,常见的相关应用包括图像处理、模式识别和控制等领域^[3-5]。当下的研究也已表明,针对不同的预测问题,只要选定了单隐层前馈网络(SLFN)^[6]的激活函数,就能够确定相应的决策边界。

单隐层前馈网络结构简单,训练代价小,在许多场合中都有不俗表现。目前经常见到的 SLFN 有极限学习机(Extreme Learning Machine, ELM)^[7]和随机向量函数连接网络(Random vector functional-link network, RVFL)^[8]。其中,ELM 网络结构简单,训练和预测时计算速度较快,在做分类时有较好的性能。然而现今的批处理系统在执行任务过程时数

据会不断更新,如果反复进行批量训练,就会使计算成本偏高。基于此,则有学者提出了在线顺序的极限学习机(OSELM)^[9],能够在线地训练模型,不断更新输出权重。RVFL 也是一种 SLFN 网络,在结构上与 ELM 类似,最主要的不同就在于 RVFL 直接将输入层与输出层做了直接映射。尽管 RVFL 在一定程度上提升了网络复杂度,但也提升了网络的泛化能力。为了降低奇异点和噪声的影响,引入了核函数^[10]来替代激活函数,再无需对隐藏层神经元数量进行调整,但是核函数的参数选择上却又面临了灵活多样、难以固定的困扰。Zhang 等人^[11]使用了在线的 RVFL 网络来预测高炉铁水质量,但是并未引入正则化思想,神经网络的泛化能力略有不足。Zhou 等人^[12]使用 L_{21} 范数的方式提升结构的稀缺性,通过整体消除神经元来降低模型的内在复杂性,

作者简介: 季江飞(1996-),男,硕士研究生,主要研究方向:计算机技术;郭久森(1994-),男,硕士研究生,主要研究方向:软件工程。

通讯作者: 季江飞 Email:2986831797@qq.com

收稿日期: 2022-03-09

降低损失函数^[13]中的经验损失和结构损失。

综合前述研究, 本文将在线顺序模式引入 RVFL, 使其具有即时处理新输入的小块训练集的能力, 而不用重复计算整体大量的数据集。同时, 利用 L_{21} 范数对输出权重的表达进行正则化, 可有效降低模型的内在复杂性。

1 相关知识

1.1 随机向量功能连接网络

随机向量功能连接网络(RVFL)是一种单隐层神经网络, 具体网络结构如图 1 所示。假设有 N 个任意的样本, 这里的第 i 个样本的输入为 $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{in}] \in \mathbb{R}^n$, 第 i 个样本的输出为 $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{im}] \in \mathbb{R}^m$ 。对于一个有 L 个隐含层节点的 RVFL 网络的原理公式可写为:

$$\sum_{j=1}^L \beta_j g(\mathbf{w}_j \mathbf{x}_i + \mathbf{b}_j) + \sum_{j=L+1}^{L+d} \beta_j x_{ij} = o_i \quad (1)$$

其中, $g(x)$ 为激活函数; \mathbf{w}_j 为输入权重, $\mathbf{w}_j = [w_{j1}, w_{j2}, \dots, w_{jn}]^T$; \mathbf{b}_j 为隐含层增强节点的偏置; $\beta_j = [\beta_{j1}, \beta_{j2}, \dots, \beta_{jm}]^T$ 为输出权重。 \mathbf{w}_j 和 \mathbf{b}_j 一般是随机确定的。

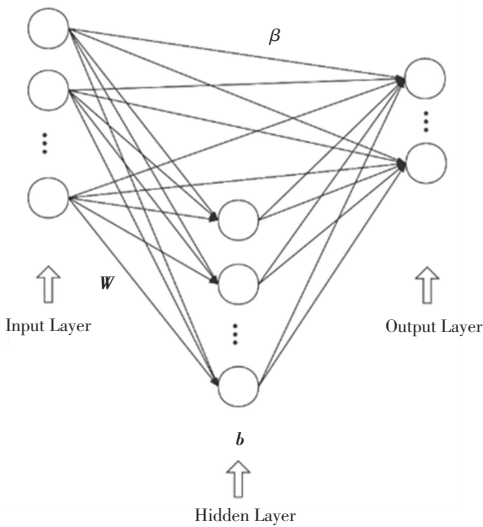


图 1 RVFL 的网络结构

Fig. 1 Network structure of RVFL

公式(1)可以简化成矩阵, 矩阵形式为:

$$[\mathbf{G} \ \mathbf{X}] \boldsymbol{\beta} = \mathbf{O} \quad (2)$$

其中,

$$\mathbf{G} = \begin{bmatrix} g(\mathbf{w}_1 \mathbf{x}_1 + \mathbf{b}_1) & \cdots & g(\mathbf{w}_L \mathbf{x}_1 + \mathbf{b}_L) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_1 \mathbf{x}_N + \mathbf{b}_1) & \cdots & g(\mathbf{w}_L \mathbf{x}_N + \mathbf{b}_L) \end{bmatrix}_{N \times L},$$

$$\boldsymbol{\beta} = [\boldsymbol{\beta}_1 \cdots \boldsymbol{\beta}_{L+n}]^T_{(L+n) \times m}, \quad \mathbf{O} = [\mathbf{O}_1 \cdots \mathbf{O}_N]^T_{N \times m} \quad (3)$$

研究可知, 公式(3)是公式(2)的细节表达, 这里的 \mathbf{G} 为隐含层增强节点的输出, $\boldsymbol{\beta}$ 为输出权重, \mathbf{O} 为实际输出。对于 RVFL 网络来说, 要使得输出的误差 $\mathbf{E} = \sum_{i=1}^N (\mathbf{O}_i - \mathbf{Y}_i)^2$ 最小, 可转化为求解 $\hat{\boldsymbol{\beta}} = \mathbf{H}^+ \mathbf{Y}$, 此处的 \mathbf{H}^+ 是矩阵 \mathbf{H} 的 Moore-Penrose 广义逆^[14]。

相较于传统的 RVFL 网络, 正则化的 RVFL 能够提升网络的泛化能力, 并能有效地预防模型的过拟合问题。常用的做法是求出训练误差和输出权重的最小值:

$$\text{最小化: } L_{RVFL} = \frac{1}{2} C \|\mathbf{E}\|_2^2 + \frac{1}{2} \|\boldsymbol{\beta}\|_2^2 \quad (4)$$

其中, 输出的误差 $\mathbf{E} = \mathbf{O} - \mathbf{Y}$, C 是正则化系数, 用于权衡训练误差和模型复杂度之间的影响。为了求得该最小值, 可以通过将 RVFL 与 $\boldsymbol{\beta}$ 有关的梯度设为零, 进而推得 $\boldsymbol{\beta}$ 的封闭形式解为:

$$\boldsymbol{\beta} = \begin{cases} \left(\mathbf{H}^T \mathbf{H} + \frac{\mathbf{I}}{\lambda} \right)^{-1} \mathbf{H}^T \mathbf{T} & \text{if } N > L + n \\ \mathbf{H}^T \left(\mathbf{H}^T \mathbf{H} + \frac{\mathbf{I}}{\lambda} \right)^{-1} \mathbf{T} & \text{if } N < L + n \end{cases} \quad (5)$$

其中, \mathbf{I} 是维度为 $(L + n)$ 的单位矩阵。

1.2 正则化 RVFL 和损失函数的 L_{21} 范数

在数据集的收集和整理的过程中, 难免会录入一些受到噪声影响或者偶然性较大的数据点, 这些由于人为因素或环境因素造成的误差, 可能会使由常规样本训练而来的神经网络的性能难以达到预期。为了减少或者消除奇异点所带来的影响, 这里定义了正则化和损失函数在 L_{21} 模的最小化表达:

$$\text{最小化: } L_{RVFL} = \frac{1}{2} C \|\mathbf{E}\|_2^2 + \frac{1}{2} \|\boldsymbol{\beta}\|_{21}$$

$$\text{限制条件: } h(x_i) \boldsymbol{\beta} = \mathbf{y}_i^T - \mathbf{e}_i^T, i = 1, 2, \dots, N$$

(6)

从上文对 L_{21} 模的定义可知, 对矩阵处理时先按行求其 L_2 范数, 再对结果求 L_1 范数, 如此操作能够将权重矩阵的部分行值减小到零。从而达到剔除可忽略不计的特征值的目的, 同时也能减少网络的复杂性。基于 KKT 条件, 可以得到最优解:

$$L_{LR21-RVFL}(\lambda, \boldsymbol{\beta}, \mathbf{e}) = \frac{\lambda}{2} \|\mathbf{e}_i\|^2 + \frac{1}{2} \text{Tr}(\boldsymbol{\beta}^T \mathbf{D} \boldsymbol{\beta}) - \sum_{i=1}^N \sum_{k=1}^m \alpha_{i,k} (h(x_i) \beta_k - y_{ik} + e_{ik}) \quad (7)$$

其中, $\alpha_{i,k}$ 是第 i 个训练样本和第 k 个输出神经

元的拉格朗日乘数; β_k 是输出权重的第 k 列; $Tr(\mathbf{A})$ 操作表示求方阵 \mathbf{A} 的迹; 矩阵 \mathbf{D} 是一个对角矩阵, 其

元素 $d_{ii} = \frac{1}{2 \|\beta_i\|_2}$ 。而在有些情况中, $\|\beta_i\|_2$ 的计算结果可能会趋近于零, 为此引入一个足够小的量 ε ,

$$\text{此时有 } d_{ii} = \frac{1}{2 \sqrt{\beta_i^T \beta_i + \varepsilon}}.$$

为了计算输出权重 β 的解, 分别对 α_i, β_k, e_i 求偏导并使其为零, 那么就可以得到输出权重的最终解为:

$$\beta = \begin{cases} \left(\mathbf{H}^T \mathbf{H} + \frac{\mathbf{D}}{\lambda} \right)^{-1} \mathbf{H}^T \mathbf{O} & \text{if } N > L + n \\ \mathbf{D}^{-1} \mathbf{H}^T \left(\mathbf{H}^T \mathbf{D}^{-1} \mathbf{H} + \frac{\mathbf{I}}{\lambda} \right)^{-1} \mathbf{O} & \text{if } N < L + n \end{cases} \quad (8)$$

2 含有 L_{21} 范数的 OSRVFL

RVFL 能够批量训练全部 N 个样本值, 而在实时场景中, 样本数据可能会不断地更新, 此时 RVFL 网络则要去重新计算所有的训练数据。为了能够对更新的数据进行实时处理, 故将研究重点关注在即时的数据上, 引入了在线顺序的方式。通过不断地调整新参与的训练数据与已有数据之间的关联关系, 计算和更新输出权重。让网络对新样本也能有好的预测效果, 同时又大大减少了重复训练整块样本数据的计算量。

文中不妨假设初始训练样本数 $N_0 > L + n$, 那么初始输出权重 β_0 可以由初始输出矩阵 \mathbf{H}_0 和输出向量 \mathbf{Y}_0 得到:

$$\begin{aligned} \beta_0 &= \mathbf{M}_0^{-1} \mathbf{H}_0^T \mathbf{T}_0 \\ \mathbf{M}_0 &= \left(\mathbf{H}_0^T \mathbf{H}_0 + \frac{\mathbf{D}_0}{\lambda} \right) \end{aligned} \quad (9)$$

其中, \mathbf{D}_0 是一个 $L \times L$ 的对角矩阵。当一个大小为 N_1 的新数据块加入时, 此时的输出权重有:

$$\beta_1 = \mathbf{M}_1^{-1} \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{T}_0 \\ \mathbf{T}_1 \end{bmatrix}$$

$$\mathbf{M}_1 = \frac{\mathbf{D}_1}{\lambda} + \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix} = \frac{\mathbf{D}_1}{\lambda} + \mathbf{H}_0^T \mathbf{H}_0 + \mathbf{H}_1^T \mathbf{H}_1 =$$

$$\frac{\mathbf{D}_1 - \mathbf{D}_0}{\lambda} + \mathbf{M}_0 + \mathbf{H}_1^T \mathbf{H}_1 = \frac{\delta \mathbf{D}}{\lambda} + \mathbf{M}_0 + \mathbf{H}_1^T \mathbf{H}_1 \quad (10)$$

不妨把 β_0 和 β_1 之间的前后递推关系应用于第 k 和第 $(k+1)$ 个数据块之间, 求 \mathbf{M}_k^{-1} 时使用

Woodbury 公式, 令 $\mathbf{K}_k = \mathbf{M}_k^{-1}$, 令 $\mathbf{R}_k = \left(\frac{\delta \mathbf{D}}{\lambda} + \mathbf{M}_k \right)^{-1}$

做展开时得到中间量 $\left(\frac{\delta \mathbf{D}}{\lambda} + \mathbf{M}_k \right)^{-1}$, 那么可推出:

$$\beta_1 = \mathbf{M}_1^{-1} \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{T}_0 \\ \mathbf{T}_1 \end{bmatrix}$$

$$\mathbf{M}_1 = \frac{\mathbf{D}_1}{\lambda} + \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix}^T \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \end{bmatrix} = \frac{\mathbf{D}_1}{\lambda} + \mathbf{H}_0^T \mathbf{H}_0 + \mathbf{H}_1^T \mathbf{H}_1 =$$

$$\frac{\mathbf{D}_1 - \mathbf{D}_0}{\lambda} + \mathbf{M}_0 + \mathbf{H}_1^T \mathbf{H}_1 =$$

$$\frac{\delta \mathbf{D}}{\lambda} + \mathbf{M}_0 + \mathbf{H}_1^T \mathbf{H}_1 \quad (11)$$

3 实验与分析

3.1 数据集

为了评估 L_{21} -OSRVFL 算法的效果, 本节将提出的算法和一些已有的 RVFL 相关算法进行比较, 选用的数据集来自于 UCI 机器学习存储库。使用的数据集中, 随机选取数据集 80% 样本用作于训练集, 将剩余 20% 的样本作为测试集。在数据被划分为训练集和测试集之前, 则会将数据集的顺序重新打乱并随机选取出新的训练集和测试集。

3.2 评价指标

使用不同的统计学上的指标来对算法的性能做出评价, 主要包括相关系数、均方根误差。对此拟做研究分述如下。

(1) 相关系数。数学定义公式具体如下:

$$r = \frac{\sum_{i=1}^n [(y_i - \bar{y})(o_i - \bar{o})]}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2} \sqrt{\sum_{i=1}^n (o_i - \bar{o})^2}} \quad (12)$$

(2) 均方根误差。数学定义公式具体如下:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2} \quad (13)$$

相关系数越接近 1 则表示模型预测效果越好, 而均方根误差衡量的是实际值和观测值之间的误差分布, 取值越小, 说明模型预测效果越好。

3.3 不同算法对比

表 1 给出了几种算法运用不同的激活函数, 运行在不同的数据集上的性能比较结果。由表 1 中看出, 对于不同的数据集而言, 使用不同的激活函数能够有不同的均方根误差。

在表 1 基础上, 确定了适用于每个数据集的最

佳激活函数后,对相关系数进行了评估。研究得到的不同算法的相关系数见表 2。

表 1 不同激活函数结果

Tab. 1 Results of different active functions

数据集	激活函数	RMSE		
		RVFL	LR_{21} - RVFL	LR_{21} - OSRVFL
abalone	sig	0.078 5	0.079 4	0.077 0
	hardlim	0.082 3	0.078 9	0.076 6
	radbas	0.080 2	0.078 5	0.075 7
fried	sig	0.083 5	0.083 1	0.082 8
	hardlim	0.102 4	0.096 0	0.097 2
	radbas	0.089 6	0.102 4	0.104 6
wine	sig	0.129 2	0.129 9	0.127 9
	hardlim	0.129 7	0.128 2	0.126 6
	radbas	0.132 1	0.129 4	0.127 2
machine_cpu	sig	0.053 5	0.064 3	0.047 5
	hardlim	0.041 0	0.056 3	0.039 2
	radbas	0.045 6	0.054 1	0.038 1
puma8NH	sig	0.182 2	0.180 8	0.181 1
	hardlim	0.192 5	0.188 7	0.188 6
	radbas	0.181 9	0.190 0	0.189 6

表 2 不同算法的相关系数

Tab. 2 Correlation coefficients of different algorithms

数据集	RMSE		
	RVFL	LR_{21} - RVFL	LR_{21} - OSRVFL
abalone	0.711	0.741	0.751
fried	0.849	0.855	0.854
wine	0.611	0.609	0.630
machine_cpu	0.938	0.932	0.969
puma8NH	0.612	0.617	0.627

从实验结果来看,本文提出的 L_{21} - OSRVFL 方法与经典的 RVFL 和 L_{21} - RVFL 相比,在 UCI 的多个数据集中,多数情况下均取得了最佳表现。本文算法的预估值更为贴近真实值,与真实值的相关性更强。

4 结束语

针对单隐层前馈神经网络在批处理问题中,需要反复训练网络、更新权重的问题,本文将在线顺序机制与 RVFL 相结合,并且为了降低模型复杂度,引入了 L_{21} 范数对输出权重进行正则化。在 UCI 部分分类数据集中,与另外 2 种同源算法进行了比较。从实验结果来看,本文提出的算法的预测表现更为出色。

参考文献

- [1] 黎英. 迁移学习在医学图像分析中的应用研究综述[J]. 计算机工程与应用,2021, 57 (20): 42-52.
- [2] 王灵钰, 刘子昂, 朱兴杰, 等. 基于深度学习的新型冠状病毒肺炎趋势预测研究综述[J]. 电脑知识与技术,2021, 17 (23): 106-109.
- [3] 王晓宾, 马泉, 王新承. 基于神经网络的水彩笔油墨红外光谱模式识别[J]. 激光与光电子学进展,2020, 57 (15): 282-286.
- [4] 谢晴. 神经网络在医学图像处理中的部分应用体会[J]. 影像研究与医学应用,2018, 2 (15): 228-229.
- [5] 陈亚琴, 彭浩, 冯诗愚. 基于神经网络的微重力流动冷凝换热预测[J]. 南京航空航天大学学报,2021, 53 (06): 989-995.
- [6] 李娜, 刘冰, 王伟. 基于单隐层前馈神经网络的优化算法[J]. 科学技术与工程,2019, 19 (01): 136-141.
- [7] 徐睿, 梁循, 齐金山, 等. 极限学习机前沿进展与趋势[J]. 计算机学报,2019, 42 (07): 1640-1670.
- [8] 吴天宇, 王士同. 快速多视角特权协同随机向量函数连接网络[J/OL]. 计算机科学与探索: 1-11 [2021-03-23]. <https://kns.cnki.net/kcms/detail/11.5602.TP.20210322.1003.002.html>.
- [9] 尹刚, 张英堂, 李志宁, 等. 改进在线序贯极限学习机在模式识别中的应用[J]. 计算机工程,2012, 38 (08): 164-166,169.
- [10] MAJUMDER I, DASH P K, BISOI R. Short-term solar power prediction using multi-kernel-based random vector functional link with water cycle algorithm-based parameter optimization [J]. Neural Computing and Applications,2020, 32 (12): 8011-8029.
- [11] ZHANG Li, ZHOU Ping, SONG Heda, et al. Multivariable dynamic modeling for molten iron quality using incremental random vector functional-link networks [J]. Journal of Iron and Steel Research, International,2016, 23 (11): 1151-1159.
- [12] ZHOU S, LIU X, LIU Q, et al. Random Fourier extreme learning machine with $L_2, 1$ -norm regularization [J]. Neurocomputing, 2016, 174: 143-153.
- [13] 伍权, 朱萌, 陈磊, 等. 基于贝叶斯正则化 BP 神经网络的 $S-CO_2$ 流动阻力特性预测[J/OL]. 热力发电: 1-10 [2022-01-29]. <https://doi.org/10.19666/j.rld.202112259>.
- [14] 尹钊, 贾尚晖. Moore-Penrose 广义逆矩阵与线性方程组的解[J]. 数学的实践与认识,2009, 39 (09): 239-244.