

一种随机配置网络的模型与数据混合并行学习方法

代伟¹ 李德鹏¹ 杨春雨¹ 马小平¹

摘要 随机配置网络 (Stochastic configuration networks, SCNs) 在增量构建过程引入监督机制来分配隐含层参数以确保其无限逼近特性, 具有易于实现、收敛速度快、泛化性能好等优点. 然而, 随着数据量的不断扩大, SCNs 的建模任务面临一定的挑战性. 为了提高神经网络算法在大数据建模中的综合性能, 本文提出了一种混合并行随机配置网络 (Hybrid parallel stochastic configuration networks, HPSCNs) 架构, 即: 模型与数据混合并行的增量学习方法. 所提方法由不同构建方式的左右两个 SCNs 模型组成, 以快速准确地确定最佳隐含层节点, 其中左侧采用点增量网络 (PSCN), 右侧采用块增量网络 (BSCN); 同时每个模型建立样本数据的动态分块方法, 从而加快候选“节点池”的建立、降低计算量. 所提方法首先通过大规模基准数据集进行了对比实验, 然后应用在一个实际工业案例上, 表明其有效性.

关键词 增量学习方法, 随机配置网络, 模型并行, 数据并行, 大数据建模

引用格式 代伟, 李德鹏, 杨春雨, 马小平. 一种随机配置网络的模型与数据混合并行学习方法. 自动化学报, 2021, 47(10): 2427-2437

DOI 10.16383/j.aas.c190411



开放科学(资源服务)标识码(OSID):

A Model and Data Hybrid Parallel Learning Method for Stochastic Configuration Networks

DAI Wei¹ LI De-Peng¹ YANG Chun-Yu¹ MA Xiao-Ping¹

Abstract Stochastic configuration networks (SCNs) that employ a supervisory mechanism to assign hidden-node parameters in the incremental construction process can work successfully in building a universal approximator, which indicates remarkable merits in simplicity of implementation, fast convergence and sound generalization. However, an increasing amount of data makes the modeling task with SCNs a challenge. In order to improve the comprehensive performance of neural network algorithms in large-scale data modeling, this paper proposes a hybrid parallel stochastic configuration networks (HPSCNs) architecture by incorporating dual parallelism of model and data. The proposed architecture consists of two SCN models with different construction methods to fast determine the required hidden nodes. The first one is point-incremental SCN (PSCN) which uses point incremental algorithm, and another one is block-incremental SCN (BSCN) which adopts block incremental algorithm. Besides, a dynamic block method of sample data is established and applied for each model, which accelerates the establishment of candidate node pool and reduces the computational load. Comparative experiments were first conducted through large-scale benchmark data sets and then a real-world industrial application case, indicating the effectiveness of the proposed method.

Key words Incremental learning method, stochastic configuration networks (SCNs), model parallelism, data parallelism, large-scale data modeling

Citation Dai Wei, Li De-Peng, Yang Chun-Yu, Ma Xiao-Ping. A model and data hybrid parallel learning method for stochastic configuration networks. *Acta Automatica Sinica*, 2021, 47(10): 2427-2437

收稿日期 2019-05-27 录用日期 2019-12-02

Manuscript received May 27, 2019; accepted December 2, 2019

国家自然科学基金 (61603393, 61973306), 江苏省自然科学基金 (BK20160275), 中国博士后科学基金 (2018T110571), 流程工业综合自动化国家重点实验室开放基金资助 (PAL-N201706)

Supported by National Natural Science Foundation of China (61603393, 61973306), Natural Science Foundation of Jiangsu Provinces (BK20160275), the Postdoctoral Science Foundation of China (2018T110571), State Key Laboratory of Synthetical Automation for Process Industries (PAL-N201706)

本文责任编辑 张军平

Recommended by Associate Editor ZHANG Jun-Ping

1. 中国矿业大学信息与控制工程学院 徐州 221116

1. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116

信息技术的迅速发展使得生产制造行业进入大数据时代, 这为数据建模提供了大量的数据样本, 使得数据驱动建模在不同领域产生广阔的应用空间^[1-3]. 然而, 系统复杂度和数据规模的日益增大为数据建模算法带来新的挑战. 模型精度取决于样本的质量与数量, 但超过一定规模的样本数据, 会显著增加网络参数训练与优化的成本, 且难以有效学习, 导致模型的整体性能下降^[4]. 在采用传统神经网络算法进行大数据建模时, 所得到的模型往往存在训练耗时、网络结构复杂等问题, 难以满足实际应

用的需求. 因此, 建立一种能够从大量数据中快速、高效学习的策略具有重要意义.

上世纪 90 年代, 文献 [5] 提出的随机向量函数链接网络, 与文献 [6] 提出的另一种具有随机权值的单层前馈神经网络等被统称为随机权神经网络 (Random weight neural networks, RWNNs)^[7-8]. 其特征在于隐含层参数 (输入权值和偏置) 在给定的区间内随机产生, 只需解析求解网络输出权值. 因 RWNNs 实现简单、建模速度快等优势受到了广泛的关注. 文献 [9] 设计了一种二维 RWNNs 分类器, 用于人脸识别, 其使用两组输入权值直接以特征矩阵作为输入而不用转换成向量形式, 能够有效保留图像数据的空间信息, 具有良好的分类性能. 文献 [10] 将非参数的核密度估计方法与加权最小二乘估计引入到 RWNNs 的学习算法, 通过评估每个训练样本对于建模的贡献度来抑制噪声数据或离群点的负面影响, 从而建立了鲁棒 RWNN 以及在线学习算法. 在此基础上, 文献 [11] 进一步引入正则项来防止过拟合现象, 有效减小输出权值, 在一定程度上降低了模型复杂度. 然而, 两个本质缺陷使得 RWNNs 的应用受到一定的限制: 1) 隐含层节点数不能先于训练而确定, 即难以设定合适的网络结构; 2) 隐含层参数在固定不变的区间内产生, 影响其实际逼近特性^[12-13].

为了解决上述问题, Wang 等首次提出了一种随机增量学习技术, 即随机配置网络 (Stochastic configuration networks, SCNs)^[14]. SCNs 在一个可调区间内随机分配隐含层参数, 并创新性地引入监督机制加以约束, 确保了其无限逼近特性. 作为一种增量学习算法, SCNs 在每次的增量学习过程中, 均建立候选“节点池”以选出最佳节点, 加快了收敛速度; 同时, 网络结构也可在增量学习过程中确定. 针对 SCNs 监督机制中关键参数的选择问题, 文献 [15] 指导性地给出了两种不等式约束, 从而提高了随机参数配置的效率, 有助于候选“节点池”的建立. 文献 [16] 以 SCNs 作为基模型, 采用负相关学习策略来配置输出权值, 提出了一种快速解除相关异构神经元的集成学习方法. 文献 [17] 针对隐含层输出矩阵可能存在的多重共线性等不适定情况, 在计算输出权值时采用截断奇异值分解对重新构建的输出矩阵进行广义逆运算, 建立了 TSVD-SCN 模型. 文献 [18] 通过引入正则化技术, 进一步提升 SCNs 的泛化性能. 文献 [19] 通过理论分析和算法实现将 SCNs 推广到深度结构, 即 DeepSCNs. 结果表明, 相比其他深度神经网络, DeepSCNs 可以快速有效地生成, 且学习特性和泛化性能之间具有一致性. 此外, 文献 [20] 和 [21] 分别提出了两种不同的鲁棒

SCNs 算法用于受污染数据的建模问题.

然而, 现有的 SCNs 及其相关算法均采用点增量的构建方式. 在大数据建模任务中, 往往需要大量的迭代和耗时的伪逆运算. 为解决采用传统计算方式将产生过大计算损耗的问题, 本文研究 SCNs 并行学习方法, 以提高 SCNs 在大数据应用中的效率.

现有面向大数据的神经网络并行建模的研究成果可分为数据或者模型并行两类^[22-24]. 数据并行是将数据集分成若干数据块以同时处理, 模型并行是将参数进行并行划分以同时求解. 本文针对 SCNs 增量学习过程中网络结构动态变化的特点, 提出一种新颖的模型与数据混合并行的增量学习方法. 主要贡献在于: 将点增量和块增量两种增量模型构建方法进行并行, 左侧为点增量 SCN (PSCN), 右侧为块增量 SCN (BSCN). 其中, PSCN 每次迭代隐含层节点单个增加, 以准确地找到最佳网络结构, BSCN 每次迭代隐含层节点按块添加, 具有较快的收敛速度. 同时针对样本数据维数高、数据量大导致的计算耗时问题, 采用一种基于动态分块策略的数据并行方法, 其在增量学习过程的不同阶段, 根据当前节点数将训练数据划分为不同的数据块, 从而采用并行计算方式, 同步进行运算; 然后, 在 PSCN 与 BSCN 计算输出权值时整合数据块, 并通过比较残差, 择优选取模型参数. 对所形成的混合并行随机配置网络 (Hybrid parallel stochastic configuration networks, HPSCNs) 进行对比实验, 结果表明, 其具有资源利用率高、建模速度快、网络模型紧致等优点.

1 增量式随机权网络

1.1 传统 RWNNs

对于给定的 N 组训练集样本 $(\mathbf{x}_i, \mathbf{t}_i)$, 其输入 $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, $\mathbf{x}_i = \{x_{i,1}, \dots, x_{i,d}\} \in \mathbf{R}^d$, 相应的输出 $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_N\}$, $\mathbf{t}_i = \{t_{i,1}, \dots, t_{i,m}\} \in \mathbf{R}^m$. 其中 $i = 1, 2, \dots, N$. 具有 L 个隐含层节点的 RWNNs 模型可以表示为:

$$f_L(\mathbf{X}) = \sum_{j=1}^L \beta_j \mathbf{g}_j(\mathbf{v}_j, b_j, \mathbf{X}) \quad (1)$$

其中, $\mathbf{g}_j(\cdot)$ 表示隐含层第 j 个节点的激活函数, 隐含层参数 $(\mathbf{v}_j$ 和 $b_j)$ 分别在区间 $[-\lambda, \lambda]^d$ 和 $[-\lambda, \lambda]$ 随机生成, $\beta_j = [\beta_{j,1}, \dots, \beta_{j,q}, \dots, \beta_{j,m}]^T$ 为隐含层第 j 个节点与 m 个输出节点之间的输出权值, f_L 即当前网络的输出. 模型参数可以通过求解如下的二次型优化问题获得.

$$\{\beta_1, \dots, \beta_L\} = \min \sum_{i=1}^N \left\| \sum_{j=1}^L \beta_j g_j(\mathbf{v}_j, b_j, \mathbf{x}_i) - \mathbf{t}_i \right\|^2 \quad (2)$$

上述等式可以进一步表示为矩阵形式:

$$\beta^* = \arg \min_{\beta} \|H\beta - T\|^2 = H^\dagger T \quad (3)$$

其中

$$H = \begin{bmatrix} g(\mathbf{v}_1, b_1, \mathbf{x}_1) & \cdots & g(\mathbf{v}_L, b_L, \mathbf{x}_1) \\ \vdots & \cdots & \vdots \\ g(\mathbf{v}_1, b_1, \mathbf{x}_N) & \cdots & g(\mathbf{v}_L, b_L, \mathbf{x}_N) \end{bmatrix} \quad (4)$$

为隐含层输出矩阵, H^\dagger 为其 Moore-Penrose 广义逆^[25].

通常, 传统 RWNNs 所需要的节点数往往只能针对特定建模任务采用交叉验证法确定, 实验过程耗时, 且网络结构的确定取决于交叉验证时所选择的参数.

1.2 增量构造学习算法

增量构造学习算法是解决网络结构的一种有效方法, 从随机产生第一个节点 $\mathbf{g}_1 = g(\mathbf{v}_1, b_1, \mathbf{x})$ 开始, 向网络中逐渐增加节点; 相应地, 输出权值为 $\beta_1 = \langle \mathbf{e}_0, \mathbf{g}_1 \rangle / \|\mathbf{g}_1\|^2$, 其中 $\mathbf{e}_0 = f$; 更新 $\mathbf{e}_1 = f - \beta_1 \mathbf{g}_1$ ^[2]. 通过重复上述过程可以同时解决网络结构和参数优化问题, 增量 RWNNs 的构造过程如下.

首先, 设定增量学习的期望精度, 其输出 f_L 可以表示为先前网络 f_{L-1} 与新增节点 $\mathbf{g}_L(\mathbf{v}_L$ 和 $b_L)$ 的特定组合, 即:

$$f_L(x) = f_{L-1}(x) + \beta_L \mathbf{g}_L(x) \quad (5)$$

其中, 新增节点的输出权值依据下式

$$\beta_L = \frac{\langle \mathbf{e}_{L-1}, \mathbf{g}_L \rangle}{\|\mathbf{g}_L\|^2} \quad (6)$$

且先前网络的残差

$$\mathbf{e}_{L-1} = f - f_{L-1} = [e_{L-1,1}, \dots, e_{L-1,m}] \quad (7)$$

增量 RWNNs 虽然解决了传统 RWNNs 难以确定网络结构的问题, 但网络的输入权值和偏置通常选择在固定区间 (如 $[-1, 1]$) 内产生, 这种与样本数据无关的参数设定方式导致 RWNNs 的逼近特性具有不确定性. 因此, 随机参数的产生应该依赖于建模数据并加以约束.

2 混合并行增量学习方法

SCNs 作为一种先进的随机学习技术, 其本质特征在于随机产生的输入权值和偏置需要满足监督

机制, 并采用数据依赖策略动态调整随机区间, 有效解决了 RWNNs 及其增量构造学习算法的本质缺陷. 然而, 传统 SCNs 均是采用点增量的方式来构建网络 (PSCN), 即一次迭代过程只能添加一个节点. 采用这种方法, 每一个新增节点都需要重新建模, 当所需节点数较多时, 网络的构建就会变得相对复杂、耗时. 从特征学习的角度来说, 前馈神经网络从输入样本空间去逼近复杂非线性函数的能力很大程度上取决于隐含层的特征映射, 即从低维输入数据到高维特征空间. 而节点可以被认为是高维空间的特征. 那么, 每次只增加一个节点则在很大程度上限制了特征的获取. 因此, 使用点增量方法构建 SCNs 的过程往往需要较多的迭代, 不可避免地增加了计算量、影响了建模效率. 鉴于此, 我们最近的工作^[26] 将传统点增量 SCNs 推广为允许同时添加多个隐含层节点 (节点块) 的块增量网络 (BSCN), 用于加快构建过程.

点增量与块增量算法分别具有网络结构紧致但收敛速度慢和收敛速度快但模型不紧致的特点. 为了进一步提高算法的综合性能, 本文对传统 SCNs 进行改进, 提出了一种新的混合并行增量学习方法, 即: HPSCNs, 以应对大数据建模. 主要思想是: 在同一个建模任务中, 使用点和块两种增量构建模型方法进行并行学习, 且在学习过程中将样本数据随学习过程动态划分为多个数据块, 从而采用并行计算择优选取模型参数.

2.1 模型并行策略

HPSCNs 由 PSCN 和 BSCN 组成, 以单输出为例, 其模型并行结构如图 1 所示. 这里添加节点的过程被称为迭代. 在训练过程中, PSCN 的构建方式采用点增量方法 (隐含层节点每次迭代单个增加), 从而保证相对准确地找到最佳隐含层节点; BSCN 的构建方式采用块增量方法 (隐含层节点每次迭代按块添加), 以提高模型的收敛速度.

该策略采用平行网络并行计算. 即在增量学习过程中, 平行的 PSCN 与 BSCN 通过并行计算被独立地构建; 每次迭代结束后, 以当前网络残差为指标, 保留其中较优的网络, 并将其模型参数作为本次迭代的最终结果; 同时将该结果赋值给另一个网络以更新其节点数, 然后进行下一次迭代.

2.1.1 PSCN

图 1 所示的模型并行增量学习方法, 左侧采用传统点增量随机配置网络 PSCN, 其学习过程约束根据引理 1 获得.

引理 1^[14]. 令 $\Gamma := \{g_1, g_2, g_3, \dots\}$ 表示一组实值

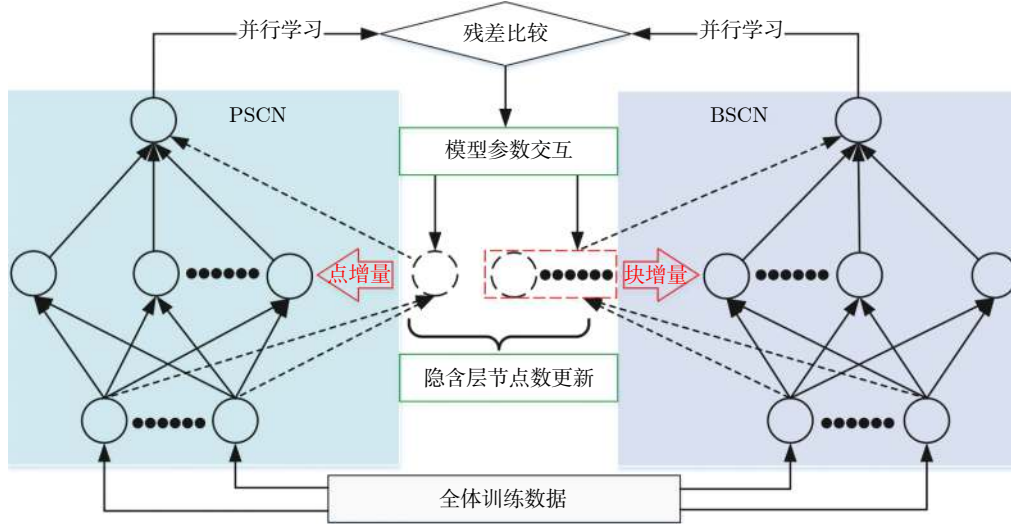


图1 模型并行结构图

Fig.1 The structure diagram of model parallelism

函数, $\text{span}(\Gamma)$ 表示由 Γ 组成的函数空间. 假设 $\text{span}(\Gamma)$ 稠密于 L_2 空间且 $\forall g \in \Gamma$, $0 < \|g\| < b_g$, 其中 $b_g \in \mathbf{R}^+$. 给定 $0 < r < 1$ 以及非负实值序列 $\{\mu_L\}$, 其中 $\lim_{L \rightarrow \infty} \mu_L = 0$, $\mu_L \leq (1-r)$. 对于 $L = 1, 2, \dots$, 定义

$$\delta_{L,q} = (1-r-\mu_L) \|e_{L-1,q}\|^2, q = 1, 2, \dots, m \quad (8)$$

若激活函数 g_L 满足下列不等式约束:

$$\langle e_{L-1,q}, g_L \rangle^2 \geq b_g^2 \delta_{L,q}, q = 1, 2, \dots, m \quad (9)$$

并且输出权值通过式 (3) 计算, 那么 $\lim_{L \rightarrow \infty} \|f - f_L\| = 0$ 恒成立.

在上述定理中, 将式 (8) 代入 (9), 可以得到点增量学习的监督机制:

$$\xi_{L,q} = \frac{\langle e_{L-1,q}, g_L \rangle^2}{g_L^T g_L} - (1-r-\mu_L) \langle e_{L-1,q}, e_{L-1,q} \rangle > 0 \quad (10)$$

其中, $q = 1, 2, \dots, m$, $\mu_L = (1-r)/(L+1)$.

网络构建过程简述如下:

1) 在可调区间内随机产生隐含层参数 (v_L 和 b_L). 将满足式 (10) 的节点作为候选节点, 并保留最大的 $\xi_{L,q}$ 所对应的随机参数;

2) 输出权值 β 通过公式 (3) 求解全局最小二乘得到;

3) 计算 PSCN 当前训练残差 e_L , 判断是否满足停止条件.

2.1.2 BSCN

按照块增量方法, 将引理 1 进行推广, 有如下引理 2.

引理 2^[26]. 给定 $0 < r < 1$ 以及非负实值序列 $\{\mu_L\}$, 其中 $\lim_{L \rightarrow \infty} \mu_L = 0$, $\mu_L \leq (1-r)$. 对于 $L = 1, 2, \dots$, $\Delta_k \in \{L\}$, 若隐含层输出块 H_{Δ_k} 满足下列块增量形式的监督机制:

$$\xi_{L,q}^B = \langle e_{L-\Delta_k,q}, H_{\Delta_k} \tilde{\beta}_{\Delta_k,q} \rangle - (1-r-\mu_L) \langle e_{L-\Delta_k,q}, e_{L-\Delta_k,q} \rangle > 0 \quad (11)$$

那么 $\lim_{L \rightarrow \infty} \|f - f_{L-\Delta_k}\| = 0$ 恒成立. 其中块宽 Δ_k 表示第 k 次迭代中新加入的节点块; $e_{L-\Delta_k} = f - f_{L-\Delta_k}$ 表示第 $k-1$ 次迭代结束时的训练残差; $H_{\Delta_k} = [g_{L-\Delta_k+1}, \dots, g_L]_{N \times \Delta_k}$ 表示第 k 次迭代时的隐含层输出块; 且

$$\tilde{\beta}_{\Delta_k,q} = (H_{\Delta_k}^T H_{\Delta_k})^\dagger H_{\Delta_k}^T e_{L-\Delta_k,q} \quad (12)$$

表示第 k 次迭代中的输出权值的中值.

BSCN 的增量学习过程在 PSCN 的基础上, 根据式 (11) 在可调区间内随机产生新增节点块 (v_{Δ_k} 和 b_{Δ_k}), 从而实现了批量分配隐含层参数和按块增加隐含层节点.

2.1.3 收敛性分析

模型并行增量学习过程当前网络节点数为 L 时, 令新增节点数为 $l \in \{1, \Delta_k\}$ 有:

$$\begin{aligned} \|e_{L+l}\|^2 - (r + \mu_{L+l}) \|e_L\|^2 = \\ \min \left\{ \|e_{L+1}\|^2, \|e_{L+\Delta_k}\|^2 \right\} - (r + \mu_{L+l}) \|e_L\|^2 = \\ \|e_{L+l}^*\|^2 - (r + \mu_{L+l}) \|e_L\|^2 \end{aligned} \quad (13)$$

其中, e_{L+l}^* 表示模型并行学习时不同构建方式下新增节点的最佳残差.

1) 点增量 ($l = 1$) 时:

$$\begin{aligned} & \|e_{L+1}^*\| - (r + \mu_{L+1}) \|e_L\|^2 \leq \\ & \|\tilde{e}_{L+1}\|^2 - (r + \mu_{L+1}) \|e_L\|^2 = \\ & \sum_{q=1}^m \langle e_{L,q} - \tilde{\beta}_{L+1,q} g_{L+1}, e_{L,q} - \tilde{\beta}_{L+1,q} g_{L+1} \rangle - \\ & \sum_{q=1}^m (r + \mu_{L+1}) \langle e_{L,q}, e_{L,q} \rangle = \\ & (1 - r - \mu_{L+1}) \|e_L\|^2 - \frac{\sum_{q=1}^m \langle e_{L,q}, g_{L+1} \rangle^2}{\|g_{L+1}\|^2} = \\ & \delta_{L+1} - \frac{\sum_{q=1}^m \langle e_{L,q}, g_{L+1} \rangle^2}{\|g_{L+1}\|^2} \leq \\ & \delta_{L+1} - \frac{\sum_{q=1}^m \langle e_{L,q}, g_{L+1} \rangle^2}{b_g^2} \end{aligned} \quad (14)$$

根据引理 1 可得 $\|e_{L+1}^*\| - (r + \mu_{L+1}) \|e_L\|^2 \leq 0$.

2) 块增量 ($l = \Delta_k$) 时:

$$\begin{aligned} & \|e_{L+\Delta_k}^*\| - (r + \mu_{L+\Delta_k}) \|e_L\|^2 \leq \\ & \|\tilde{e}_{L+\Delta_k}\|^2 - (r + \mu_{L+\Delta_k}) \|e_L\|^2 = \\ & \sum_{q=1}^m \langle e_{L,q} - H_{\Delta_k} \tilde{\beta}_{\Delta_k,q}, e_{L,q} - H_{\Delta_k} \tilde{\beta}_{\Delta_k,q} \rangle - \\ & \sum_{q=1}^m (r + \mu_{L+\Delta_k}) \langle e_{L,q}, e_{L,q} \rangle = \\ & \delta_{L+\Delta_k} - \sum_{q=1}^m (e_{L,q}^T H_{\Delta_k} (H_{\Delta_k}^T H_{\Delta_k})^\dagger H_{\Delta_k}^T e_{L,q}) = \\ & \delta_{L+\Delta_k} - \sum_{q=1}^m \langle e_{L,q}, H_{\Delta_k} \tilde{\beta}_{\Delta_k,q} \rangle \end{aligned} \quad (15)$$

根据引理 2 可得 $\|e_{L+\Delta_k}^*\| - (r + \mu_{L+\Delta_k}) \|e_L\|^2 \leq 0$.

由以上分析可知, 式 (13) $\|e_{L+i}\| - (r + \mu_{L+i}) \|e_L\|^2 \leq 0$, 残差序列 $\|e_{L+i}\|$ 单调递减且有界. 进一步地,

$$\|e_{L+i}\|^2 \leq r \|e_L\|^2 + \mu_{L+i} \|e_L\|^2 \quad (16)$$

注意到 $\lim_{L \rightarrow \infty} \mu_{L+i} \|e_L\|^2 = 0$, 其中 $\lim_{L \rightarrow \infty} \mu_{L+i} = 0$. 根据式 (16) 可以进一步得到 $\lim_{L \rightarrow \infty} \|e_{L+i}\|^2 = 0$, 即 $\lim_{L \rightarrow \infty} \|e_{L+i}\| = 0$. 因此残差序列 $\|e_{L+i}\|$ 单调递减且收敛于 0, 说明本文所提 HPSCNs 模型具有无

限逼近特性.

2.2 数据并行策略

在增量学习过程中, 将数据进行分块处理, 以建立一种数据并行计算策略, 可进一步提高学习速度. 因此, 将训练输入样本 \mathbf{X} 、训练输出样本 \mathbf{T} 等分为 M 块, 即

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_M \end{bmatrix}, \mathbf{T} = \begin{bmatrix} \mathbf{T}_1 \\ \mathbf{T}_2 \\ \vdots \\ \mathbf{T}_M \end{bmatrix} \quad (17)$$

每个数据块的隐含层输出矩阵通过并行计算获得, 第 p 个数据块与隐含层输出矩阵的对应关系:

$$\mathbf{H}_p = [\mathbf{g}_{p,1}(\mathbf{v}_1, b_1, \mathbf{X}_p), \dots, \mathbf{g}_{p,L}(\mathbf{v}_L, b_L, \mathbf{X}_p)] \quad (18)$$

这里 $\mathbf{g}_{p,L}$ 表示节点数为 L 时第 p 个数据块的隐含层输出. 此时, PSCN 的监督机制为:

$$\begin{aligned} \xi_{L,q} &= \frac{\langle e_{L-1,q}, [\mathbf{g}_{1,L}^T, \dots, \mathbf{g}_{M,L}^T] \rangle^2}{\sum_{p=1}^M \mathbf{g}_{p,L}^T \mathbf{g}_{p,L}} - \\ & (1 - r - \mu_L) \langle e_{L-1,q}, e_{L-1,q} \rangle > 0 \end{aligned} \quad (19)$$

相应地, BSCN 的监督机制为:

$$\begin{aligned} \xi_{L,q}^B &= \langle e_{L-\Delta_k,q}, [\mathbf{h}_{1,k}^T, \dots, \mathbf{h}_{M,k}^T]^T \tilde{\beta}_{k,q} \rangle - \\ & (1 - r - \mu_L) \langle e_{L-\Delta_k,q}, e_{L-\Delta_k,q} \rangle > 0 \end{aligned} \quad (20)$$

式中, $\mathbf{h}_{p,k} = [\mathbf{g}_{p,L-\Delta_k+1}, \dots, \mathbf{g}_{p,L}]$ 表示 BSCN 在第 k 次迭代中数据块 p 对应的隐含层输出块, 且

$$\tilde{\beta}_{k,q} = \left(\sum_{p=1}^M \mathbf{h}_{p,k}^T \mathbf{h}_{p,k} \right)^\dagger \left(\begin{bmatrix} \mathbf{h}_{1,k} \\ \vdots \\ \mathbf{h}_{M,k} \end{bmatrix}^T e_{L-\Delta_k,q} \right) \quad (21)$$

表示第 k 次迭代时输出权值的中值.

根据不同数据规模的学习任务, 输出权值 β 可由下式计算获得:

$$\beta = \begin{cases} (\mathbf{H}^T \mathbf{H})^\dagger \mathbf{H}^T \mathbf{T}, & N \geq L \\ \mathbf{H}^T (\mathbf{H} \mathbf{H}^T)^\dagger \mathbf{T}, & N < L \end{cases} \quad (22)$$

对于大数据建模, 这里只考虑 $N \geq L$ 的情况, 因此, 数据并行学习方法中的输出权值可以表示为:

$$\beta = \left(\sum_{p=1}^M \mathbf{H}_p^T \mathbf{H}_p \right)^\dagger \left(\sum_{p=1}^M \mathbf{H}_p^T \mathbf{T}_p \right) \quad (23)$$

式 (23) 将不同数据块整合并得到输出权值.

注意到, 随着学习过程中隐含层节点数的增加, 网络尺寸逐渐增大, 计算负荷也随之增加, 因此, 在增量学习过程的不同阶段, 本文采用动态分块策略, 逐渐增加数据块数量, 以充分利用并行计算资源, 提高学习的效率. 所形成的数据并行策略如图 2 所示, 其中从隐含层到输入层的连接起到反馈作用, 实现数据块 M 随学习过程中隐含层节点数 L 的增加而变化, 具体动态分块方法如下:

$$M_{k+1}(L) = M_0 \times \frac{L_{\max}^k}{L_{en}^k}, L_{\min}^k < L \leq L_{\max}^k \quad (24)$$

其中, M_0 表示初始样本数据分块基数; L_{en}^k 表示第 k 次迭代时分块数递增区间长度, L_{\max}^k 和 L_{\min}^k 分别表示相应递增区间的上下界, 则 $L_{en}^k = L_{\max}^k - L_{\min}^k$. L_{\max}^k 和 L_{\min}^k 的选取详见性能评估部分.

注 1. HPSCNs 继承了原始 SCNs 自组织学习的特点, 其隐含层节点在监督机制下自主构建, 无需采用传统自组织学习的剪枝等算法即可保证模型的紧致性. 此外, HPSCNs 中 PSCN 和 BSCN 的监督机制分别不同于传统 SCNs 和块增量 SCNs. 以 PSCN 为例, 若根据 M 个不同的数据块同步获取一组隐含层输出 $g_{1,L}, \dots, g_{p,L}, \dots, g_{M,L}$ 后, 直接带入 SCNs 中得到 M 组满足条件的不等式约束, 尽管该网络仍具有无限逼近特性, 但收敛性会变差. 这是因为监督机制的数据依赖特性. 每组监督机制只包含当前数据块的信息而忽略了其他数据块, 不可避免地放宽了约束的作用, 导致所得到的输入权值和偏置"质量"变低, 从而影响到模型收敛性. 因此, 隐含层参数的分配需要考虑全体数据块.

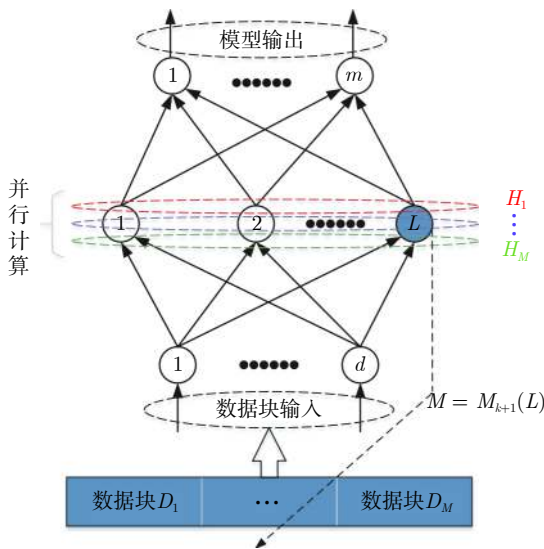


图 2 数据并行策略

Fig.2 Strategy of data parallelism

注 2. RWNNs 与 SCNs 均直接使用式 (3) 求解输出权值, 导致广义逆运算成为构建过程中较为耗时环节, 尤其针对大数据建模, 即 $N \gg L$ 的情况, 则需要对 $N \times L$ 阶的矩阵进行广义逆运算; 而基于数据并行采用式 (23), 则只需要计算 $L \times L$ 阶矩阵的广义逆, 大大降低了计算量.

2.3 HPSCNs 实现步骤

混合并行增量学习算法的具体实现描述如下:

给定训练输入 $\mathbf{X} \in \mathbf{R}^{N \times d}$, 输出 $\mathbf{T} \in \mathbf{R}^{N \times m}$. 设置增量构建过程中的容忍误差 ε 和最大迭代次数 k_{\max} ; 隐含层参数分配区间集 $\Upsilon = \{\lambda_1, \lambda_2, \dots, \lambda_{end}\}$; 最大随机配置次数 T_{\max} ; BSCN 中的块宽 Δ_k 等.

步骤 1. 初始化训练集数据分块数 $M = M_1$;

步骤 2. 开始模型与数据并行学习;

PSCN:

1) 在区间 $[-\lambda_i, \lambda_i]^d$ 和 $[-\lambda_i, \lambda_i]$ 内随机生成隐含层参数 \mathbf{v}_L 和 \mathbf{b}_L , $\lambda_i \in \Upsilon, i = 1, 2, \dots, end$;

2) 使用不同数据块同步获取 PSCN 的隐含层输出 $g_{1,L}, \dots, g_{p,L}, \dots, g_{M,L}$;

3) 根据式 (19) 建立候选“节点池”, 并找出使得 $\xi_{L,q}$ 最大的随机参数 \mathbf{v}_L^* 和 \mathbf{b}_L^* ;

4) 根据式 (23) 整合数据块并评估 PSCN 的输出权值.

BSCN:

1) 在区间 $[-\lambda_i, \lambda_i]^{\Delta_k \times d}$ 和 $[-\lambda_i, \lambda_i]^{\Delta_k \times 1}$ 内随机生成隐含层参数 \mathbf{v}_{Δ_k} 和 \mathbf{b}_{Δ_k} , $\lambda_i \in \Upsilon, i = 1, 2, \dots, end$;

2) 使用不同数据块同步获取 BSCN 的隐含层输出块 $h_{1,k}, \dots, h_{p,k}, \dots, h_{M,k}$;

3) 根据式 (20) 建立候选“节点池”, 并找出使得 $\xi_{L,q}$ 最大的随机参数 $\mathbf{v}_{\Delta_k}^*$ 和 $\mathbf{b}_{\Delta_k}^*$;

4) 根据式 (23) 整合数据块并评估 BSCN 的输出权值.

步骤 3. 计算 PSCN 和 BSCN 的训练残差;

步骤 4. 通过比较残差择优选取模型参数并用于更新另一个模型的隐含层节点数 L ;

步骤 5. 更新下一次迭代中训练数据的分块数 $M = M_{k+1}(L)$;

步骤 6. 重复上述步骤, 直至模型达到停止标准;

步骤 7. 返回最终模型参数 \mathbf{v}^* , \mathbf{b}^* 和 β^* .

注 3. 神经网络的模型精度会随隐含层节点数的增加而逐渐提高, 但当节点数过多时会使训练误差变小而测试误差增大, 即过拟合现象. 本文所提方法达到停止条件时立即结束迭代, 不再继续增加隐含层节点, 在一定程度上避免了过拟合现象.

注 4. 为尽可能地提高学习效率, 需要多次随机

配置隐含层参数, 以建立候选“节点池”, 从而寻找出最佳的隐含层节点. T_{\max} 决定着随机配置隐含层参数的次数, 过大增加计算损耗、过小不利于候选“节点池”的建立^[14]. 此外, 隐含层参数分配区间集 $\Upsilon = \{\lambda_1, \lambda_2, \dots, \lambda_{end}\}$ 可设置从 λ_1 到 λ_{end} 逐渐增长, 学习过程从 λ_1 开始逐次选择. 以 PSCN 为例: 首先在 $[-\lambda_1, \lambda_1]^d$ 和 $[-\lambda_1, \lambda_1]$ 内随机生成 T_{\max} 组隐含层参数 v_L 和 b_L , 并从后续所建立的候选“节点池”中找出最佳的一组随机参数 v_L^* 和 b_L^* ; 若候选“节点池”为空, 则从参数集 Υ 中选择 λ_2 , 调节分配区间, 进而重复上述步骤.

3 性能评估

本文选取收敛性和紧致性作为衡量模型质量的评价指标. 其中收敛性在数值上表现为残差序列的递减程度, 可采用达到期望容忍误差时的学习时间衡量, 收敛性好有利于建模的快速性. 紧致性表现为网络尺寸, 可采用达到期望容忍误差时的隐含层节点数来度量, 模型不紧致往往包含不必要的冗余节点, 不利于泛化性能, 且会显著增加计算空间和硬件实现的成本. 此外, 采用均方根误差 (Root mean squares error, RMSE) 计算建模精度, 均值 (Mean) 和标准差 (Standard deviation, Std) 则分别用来反映模型性能的平均值和离散程度.

为验证本文所提混合并行增量学习模型的有效性, 将传统 SCNs (SC-III)^[14]、块增量 SCNs (BSC-I)^[26] 和混合并行随机配置网络 (HPSCNs) 分别对大数据建模并根据评价指标进行性能评估. 仿真实验在 MATLAB 2016a 环境下运行, 所用 PC 的 CPU 为 i5, 3.4 GHz, 内存为 16 GB RAM.

3.1 建模数据

4 个不同领域的基准数据: DB1 (SGEMM GPU Kernel Performance Data Set)、DB2 (Electrical Grid Stability Simulated Data Set)、DB3 (Mv Data Set) 和 DB4 (Pole Telecommunications Data Set) 来自 UCI (University of California at Irvine)^[27] 和 KEEL (Knowledge Extraction based on Evolutionary Learning)^[28].

所选数据集信息见表 1. 针对每个大规模数据, 我们随机选取 80% 的样本作为训练集, 余下的 20% 作为测试集.

3.2 比较实验

在数据预处理阶段, 输入输出样本均被归一化至 $[-1, 1]$. 实验参数设定如下:

最大迭代次数 $k_{\max} = 300$;

表 1 基准数据集说明
Table 1 Specification of benchmark data sets

数据集	属性		样本数
	输入变量	输出变量	
DB1	14	4	241 600
DB2	12	1	10 000
DB3	10	1	40 768
DB4	26	1	14 998

最大随机配置次数 $T_{\max} = 10$;

随机参数范围 $\lambda \in \{1, 10, 50, 100, 150, 200\}$;

学习参数 $r \in \{0.9, 0.99, 0.999, \dots\}$;

容忍误差 $\varepsilon = 0.1$ (DB1 和 DB2), $\varepsilon = 0.01$ (DB3 和 DB4);

不同 BSCN 每次迭代的块宽分别取固定值 $\Delta_k = 1, 3, 5$;

大规模训练集分块基数 $M_0 = 4$, 第 k 次迭代时分块数递增区间长度 $L_{en}^k = L_{\max}^k - L_{\min}^k$ 取 50, 100, 150, ..., 具体对应关系见表 2.

因此, 根据式 (24) 可得

$$M = M_{k+1}(L) = \begin{cases} 4, & 0 < L \leq 50 \\ 6, & 50 < L \leq 150 \\ 8, & 150 < L \leq 300 \\ \vdots & \vdots \end{cases} \quad (25)$$

表 2 分块数递增区间长度及其上下界
Table 2 Incremental interval length of block number and its upper and lower bounds

L_{en}^k	L_{\max}^k	L_{\min}^k
50	50	0
100	150	50
150	300	150
...

为了便于描述, 本文以下标的形式表示 BSC-I 算法中的块宽, 以上下标的形式分别表示 HPSCNs 中左侧随机配置网络 PSCN 和右侧随机配置网络 BSCN 的块宽. 如 BSC-I₃ 表示其每次迭代添加 3 个隐含层节点; HPSCN₃¹ 表示本文所提方法中 PSCN 每次迭代添加 1 个隐含层节点, 而 BSCN 每次迭代添加 3 个隐含层节点. 需要指出的是, 点增量构建方式是块增量中 Δ_k 取 1 的特例.

表 3 记录了不同算法 50 次独立实验结果的均值和标准差 (Mean±Std). 通过比较 SC-III、BSC-I₃ 和 BSC-I₅ 可以看出, 随着块宽 Δ_k 取值的增加, 达到期望容忍误差 ε 时所需要的迭代次数 k 明显降低, 建模时间 t 显著减少, 即块增量的构建方式可以有

效加快传统 SCNs 的收敛速度;然而,块增量 SCNs 需要更多的节点数 L ,导致网络复杂度增加,不利于模型泛化性能.这是因为从特征学习的角度来说,残差序列单调递减,尚未学习的特征也会随着增量学习的过程减少,然而 BSC-I₃ 和 BSC-I₅ 在每次迭代中采用固定的块宽,不可避免地导致了特征(节点)的冗余.尤其在构建过程后期,每次迭代中节点块对于建模的贡献近似于单个节点,却徒增了网络复杂度.因此,块增量 SCNs 是一种以牺牲模型紧致性为代价的极速建模方法.

相同的结论可以见算法 HPSCN₁¹、HPSCN₃¹ 和 HPSCN₅¹.不同的是,本文所提方法可以获得更好的综合性能,即相比于 SC-III 和 BSC-I, HPSCN₁¹ 和 HPSCN₃¹ 能够保证模型较为紧致的同时进一步减少建模时间,原因如下:

1) 数据并行通过使用动态分块策略并同步获取隐含层输出,加快了候选“节点池”的建立;同时,针对大数据本文所提方法采用式(23)整合数据块并计算输出权值,在很大程度上降低了广义逆运算负担;

2) 模型并行通过单次迭代中 PSCN 与 BSCN 独立地同步建模,完成当前迭代后选择获得残差较小的模型参数.其中,HPSCNs 中的 BSCN 采用块增量的构建方式,具有较快的收敛性;PSCN 采用点增量的构建方式,从而准确地找到最佳隐含层节点数.模型并行增量学习也有效减少了迭代次数.

为了评估模型的综合性能,图 3 给出不同算法对数据集 DB1-DB4 的建模时间 (t) 和隐含层节点数 (L).其中, t 和 L 均归一化处理至 $[0, 1]$,并表示对应性能指标的得分情况.如:在 DB1 建模任务中,SC-III 相比其他算法用时最长,其 t 对应的得分为 1、BSC-I₅ 所需节点数最多,其 L 对应的得分为 1.显然,分数越低,性能越好.因此,由图 3 可以看出,本文所提混合并行增量学习方法具有较好的

表 3 不同算法性能比较

Table 3 Performance comparison of different algorithms

数据集	算法	$t(s)$	k	L
DB1	SC-III	24.35 ± 1.69	164.40 ± 7.76	164.40 ± 7.76
	BSC-I ₃	12.60 ± 1.21	69.20 ± 3.03	207.60 ± 9.09
	BSC-I ₅	9.41 ± 1.33	44.00 ± 3.24	220.00 ± 16.20
	HPSCN ₁ ¹	3.48 ± 0.38	122.40 ± 8.02	122.40 ± 8.02
	HPSCN ₃ ¹	3.03 ± 0.28	63.40 ± 4.16	162.80 ± 7.90
	HPSCN ₅ ¹	2.96 ± 0.19	45.00 ± 2.83	215.00 ± 9.71
DB2	SC-III	26.97 ± 2.54	300.00 ± 14.18	300.00 ± 14.18
	BSC-I ₃	14.66 ± 1.33	120.40 ± 3.98	361.20 ± 11.93
	BSC-I ₅	11.01 ± 1.07	78.80 ± 2.91	394.00 ± 14.87
	HPSCN ₁ ¹	7.22 ± 0.95	239.30 ± 14.55	239.3 ± 14.55
	HPSCN ₃ ¹	5.47 ± 0.33	123.50 ± 3.34	301.90 ± 10.99
	HPSCN ₅ ¹	4.39 ± 0.42	81.80 ± 3.74	378.60 ± 16.54
DB3	SC-III	18.04 ± 2.15	106.60 ± 3.36	106.60 ± 3.36
	BSC-I ₃	8.96 ± 1.21	39.80 ± 2.28	119.40 ± 6.84
	BSC-I ₅	6.81 ± 0.55	25.20 ± 1.10	126.00 ± 5.48
	HPSCN ₁ ¹	3.45 ± 0.24	97.00 ± 2.65	97.00 ± 2.65
	HPSCN ₃ ¹	2.05 ± 0.13	41.20 ± 2.17	106.40 ± 4.39
	HPSCN ₅ ¹	1.88 ± 0.12	25.00 ± 1.22	121.00 ± 6.44
DB4	SC-III	9.16 ± 0.34	161.20 ± 2.56	161.20 ± 2.56
	BSC-I ₃	3.79 ± 0.68	54.20 ± 0.84	162.60 ± 2.51
	BSC-I ₅	2.59 ± 0.13	33.40 ± 0.89	167.00 ± 4.47
	HPSCN ₁ ¹	4.23 ± 0.13	154.80 ± 2.59	154.80 ± 2.59
	HPSCN ₃ ¹	2.01 ± 0.13	59.00 ± 2.00	162.60 ± 2.41
	HPSCN ₅ ¹	1.36 ± 0.11	34.20 ± 1.09	166.20 ± 3.03

综合性能,且 HPSCN₃¹ 最佳.

3.3 参数讨论与选择

需要指出的是,HPSCN₅¹ 虽然所需时间最少但已经失去了模型紧致性上的优势,是一种折中的算法.为了进一步研究不同块宽对本文所提方法的影

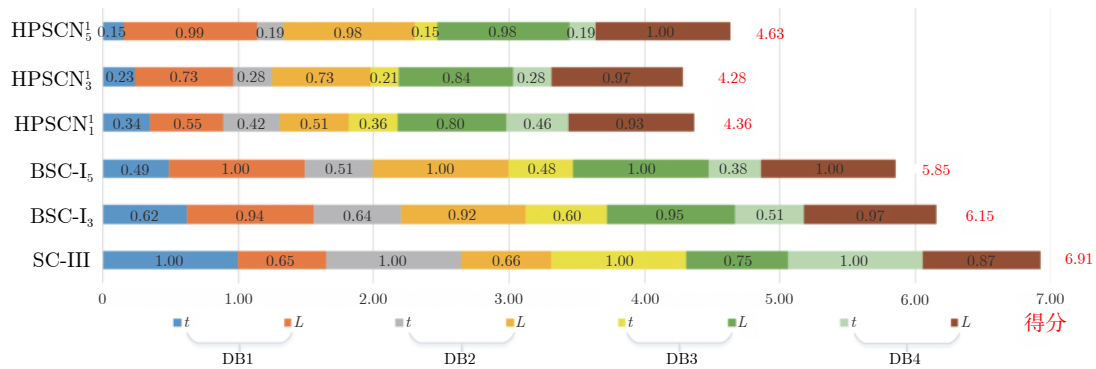


图 3 不同算法综合性能比较

Fig. 3 Comparison of comprehensive performance of different algorithms

响, 我们定义 HPSCNs 的有效性 (Effectiveness, Eff):

$$Eff = \frac{nL}{nL + nR} \times 100\% \quad (26)$$

其中 nL 和 nR 分别表示每次迭代 PSCN 和 BSCN 贡献最佳模型参数的次数, 则 Eff 正比于 nL .

表 4 记录了 HPSCNs 中 BSCN 选择不同块宽 Δ_k 时的模型并行情况. 可以看出, 对于 DB1-DB4, 随着 Δ_k 的增加, Eff 的数值越来越小, 如 $\Delta_k = 10$ 时, $HPSCN_{10}^1$ 的 Eff 最低, 此时 PSCN 基本上失去了其在模型并行中的作用. 因此, 考虑模型的综合性能, 使用 HPSCNs 进行大数据建模时, 其 BSCN 的块宽建议取 $\Delta_k \in \{1, 2, 3\}$.

3.4 工业案例

将所提方法应用在典型一段磨矿过程, 建立以磨机给矿量、磨机入口给水量和分级机溢流质量浓度为输入, 以磨矿粒度为输出的软测量模型. 从磨矿半实物仿真平台^[29]中采集 20 000 训练样本和 5 000 测试样本, 设置容忍误差 $\varepsilon = 0.05$, 块宽 $\Delta_k = 3$, 其余实验参数设定同比较实验部分.

表 4 不同块宽的算法性能比较

Table 4 Performance comparison of algorithms with different block sizes

数据集	算法	nR	nL	Eff (%)
DB1	$HPSCN_1^1$	61.3	61.1	49.9
	$HPSCN_2^1$	63.8	22.4	26.0
	$HPSCN_3^1$	52.8	12.6	19.3
	$HPSCN_5^1$	42.5	2.5	5.6
	$HPSCN_{10}^1$	24.2	0.6	2.4
DB2	$HPSCN_1^1$	119.2	120.1	50.2
	$HPSCN_2^1$	115.0	56.4	32.9
	$HPSCN_3^1$	99.2	24.3	19.7
	$HPSCN_5^1$	74.2	7.6	9.3
	$HPSCN_{10}^1$	44.6	0.4	0.9
DB3	$HPSCN_1^1$	48.4	48.6	50.1
	$HPSCN_2^1$	40.8	23.4	36.4
	$HPSCN_3^1$	33.6	7.6	18.4
	$HPSCN_5^1$	24.0	1.0	4.0
	$HPSCN_{10}^1$	13.6	0.2	1.4
DB4	$HPSCN_1^1$	77.3	77.5	50.0
	$HPSCN_2^1$	64.2	29.4	31.4
	$HPSCN_3^1$	51.8	7.2	12.2
	$HPSCN_5^1$	33.0	1.2	3.5
	$HPSCN_{10}^1$	17.0	0.2	1.1

图 4 和图 5 分别为基于 HPSCNs 的磨矿粒度软测量模型的收敛特性和逼近特性. 由图 4 可以看出, 在训练过程中所提方法具有较好的收敛效果, 且在接近容忍误差前残差序列能够快速下降. 为了便于观察, 这里仅取 500 个测试数据的逼近效果 (如图 5 所示), 从中可以看出模型输出与真实值基本吻合, 可以实现磨矿粒度的准确估计.

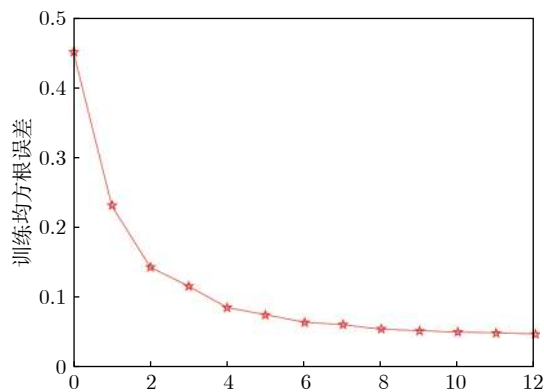


图 4 模型的收敛曲线

Fig.4 Convergence curve of HPSCNs

此外, 从图 5 可知, 所建立磨矿粒度软测量模型在达到期望精度时, 建模时间较快, 网络结构简单, 具有良好的泛化性能; $Eff = 25.0\%$, 其中左侧点增量网络 (PSCN)、右侧块增量网络 (BSCN) 贡献最佳模型参数的次数分别为 3 和 9, 说明模型具有较高的有效性. 需要指出的是, 上述矿粒度软测量模型是通过输入特性维数较少的样本建立的, 本文所提方法应用到其他数据维数较多的工业案例中, 将会取得更好的优势. 同时在多维大数据应用时, 还可以考虑并行节点池构建策略, 以提高学习速度.

4 结语

本文基于随机配置网络提出了一种模型与数据混合并行的增量学习方法. 实际数据仿真研究表明: 与传统 SCNs 和块增量 SCNs 相比, 本文所提方法大大降低了计算量, 提高了资源利用率, 所建立模型综合性能好, 且利于硬件实现, 具有良好的实用价值. 然而, HPSCNs 直接根据 PSCN 和 BSCN 的残差选择新增节点, 在 BSCN 获得残差优势不明显的情况下, 选择了增加节点块构建模型, 不利于模型紧致性. 下一步的研究目标是进一步提高算法的综合性能, 如针对 HPSCNs 模型的有效性, 建立新的模型参数交互准则; 在保留块增量 SCNs 极速建模优势的情况下, 削减其中的冗余隐含层节点.

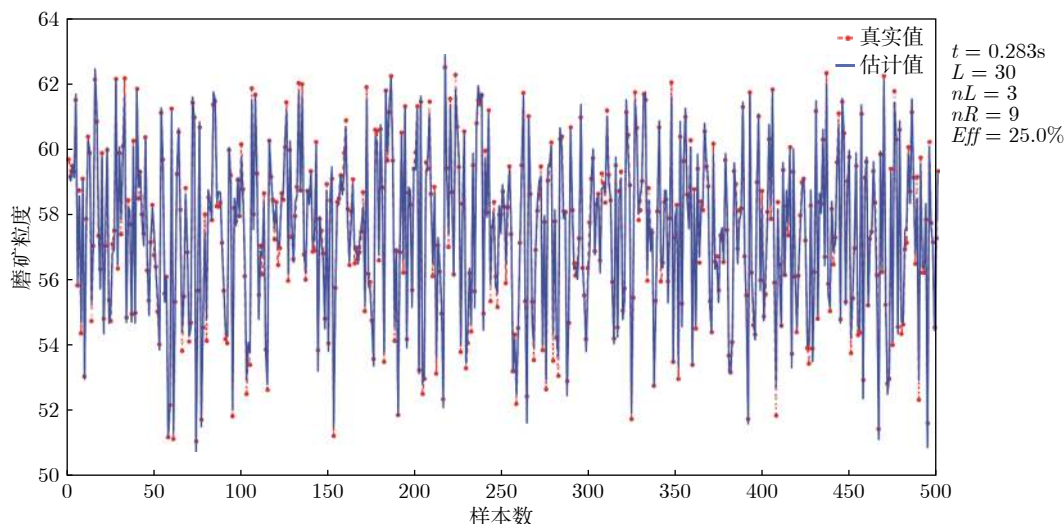


图 5 模型的逼近特性

Fig.5 Approximation performance of HPSCNs

References

- Chen Xing, Niu Ya-Wei, Wang Guang-Hui, Yan Gui-Ying. MKRMDA: Multiple kernel learning-based Kronecker regularized least squares for MiRNA-disease association prediction. *Journal of Translational Medicine*, 2013, **15**(1): 251–264
- Zhou Ping, Liu Ji-Ping. Data-driven multi-output ARMAX modeling for online estimation of central temperatures for cross temperature measuring in blast furnace ironmaking. *Acta Automatica Sinica*, 2018, **44**(3): 552–561
(周平, 刘记平. 基于数据驱动多输出ARMAX建模的高炉十字测温中心温度在线估计. *自动化学报*, 2018, **44**(3): 552–561)
- Tang Jian, Qiao Jun-Fei, Chai Tian-You, Liu Zhuo, Wu Zhi-Wei. Modeling multiple components mechanical signals by means of virtual sample generation technique. *Acta Automatica Sinica*, 2018, **44**(9): 1569–1589
(汤健, 乔俊飞, 柴天佑, 刘卓, 吴志伟. 基于虚拟样本生成技术的多组分机械信号建模. *自动化学报*, 2018, **44**(9): 1569–1589)
- Witten Ian H, Frank E, Hall M A. *Data Mining: Practical Machine Learning Tools and Technique, 3rd Edition*. Amsterdam: Morgan Kaufmann, 2011
- Pao Y H, Takefji Y. Functional-link net computing: Theory, system architecture, and functionalities. *Computer*, 1992, **25**(5): 76–79
- Schmidt W F, Kraaijveld M A, Duin R P W. Feedforward neural networks with random weights. *International Conference on Pattern Recognition IEEE Computer Society*, 1992
- Cao Wei-Peng, Wang Xi-Zhao, Ming Zhong, Gao Jin-Zhu. A review on neural networks with random weights. *Neurocomputing*, 2018, **275**: 278–287
- Scardapane S, Wang Dian-Hui. Randomness in neural networks: An overview. *Wiley Interdisciplinary Reviews-Data Mining and Knowledge Discovery*, 2017, **7**(2): e1200
- Lu Jing, Zhao Jian-Wei, Cao Fei-Long. Extended feed forward neural networks with random weights for face recognition. *Neurocomputing*, 2014, **136**: 96–102
- Dai Wei, Liu Qiang, Chai Tian-You. Particle size estimate of grinding processes using random vector functional link networks with improved robustness. *Neurocomputing*, 2015, **169**: 361–372
- Dai Weo, Chen Qi-Xin, Chu Fei, Chai Tian-You. Robust regularized random vector functional link network and its industrial application. *IEEE Access*, 2017, **5**: 16162–16172
- Li Ming, Wang Dian-Hui. Insights into randomized algorithms for neural networks: practical issues and common pitfalls. *Information Sciences*, 2017, **382**: 170–178
- Gorban A N, Tyukin I Y, Prokhorov D V, Soseikov K I. Approximation with random bases: Pro et contra. *Information Sciences*, 2016, **364**: 129–145
- Wang Dian-Hui, Li Ming. Stochastic configuration networks: Fundamentals and algorithms. *IEEE Transactions on Cybernetics*, 2017, **47**(10): 3466–3479
- Zhu Xiao-Long, Feng Xiang-Chu, Wang Wei-Wei, Jia XiXi, He Rui-Qiang. A further study on the inequality constraints in stochastic configuration networks. *Information Sciences*, 2019, **487**: 77–83
- Wang Dian-Hui, Cui Cai-Hao. Stochastic configuration networks ensemble for large-scale data analytics. *Biochemical Treatment Process. Information Sciences*, 2017, **417**: 55–71
- Sheng Zhui-Yong, Zeng Zhi-Qiang, Qu Hong-Quan, Zhang Yuan. Optical fiber intrusion signal recognition method based on TSVD-SCN. *Optical Fiber Technology*, 2019, **48**: 270–277
- Wang Qian-Jin, Yang Chun-Yu, Ma Xiao-Ping, Zhang Chun-Fu, Peng Si-Min. Underground Airflow Quantity Modeling Based on SCN. *Acta Automatica Sinica*, 2021, **47**(8): 1963–1975
(王前进, 杨春雨, 马小平, 张春富, 彭思敏. 基于随机配置网络的井下供给风量建模. *自动化学报*, 2021, **47**(8): 1963–1975)
- Wang Dian-Hui, Li Ming. Deep stochastic configuration networks with universal approximation property. In: Proceedings of the 2018 International Joint Conference on Neural Networks, 2018.
- Wang Dian-Hui, Li Ming. Robust stochastic configuration networks with kernel density estimation for uncertain data regression. *Information Sciences*, 2017, **412-413**: 210–222
- Li Ming, Huang Chang-Qin, Wang Dian-Hui. Robust stochastic configuration networks with maximum correntropy criterion for uncertain data regression. *Information Sciences*, 2019, **473**: 73–86
- He Qing, Shang Tang-Feng, Zhuang Fu-Zhen, Shi Zhon-Zhi. Parallel extreme learning machine for regression based on MapReduce. *Neurocomputing*, 2013, **102**: 52–58
- Duan Ming-Xing, Li Ken-Li, Liao Xiang-Ke, Li Ke-Qin. A parallel multiclassification algorithm for big data using an extreme learning machine. *IEEE Transactions on Neural Networks and Learning Systems*, 2018, **29**(6): 2337–2351
- Wang Yue-Qing, Dou Yong, Liu Xin-Wang, Lei Yuan-Wu. PR-

- ELM: Parallel regularized extreme learning machine based on cluster. *Neurocomputing*, 2016, **173**: 1073–1081
- 25 Lancaster P, Tismenetsky M. *The Theory of Matrices: With Applications*. Elsevier, 1985
- 26 Dai Wei, Li De-Peng, Zhou Ping, Chai Tian-You. Stochastic configuration networks with block increments for data modeling in process industries. *Information Sciences*, 2019, **484**: 367–386
- 27 UCI Machine Learning Repository. [Online], available: <https://archive.ics.uci.edu>, 2013
- 28 KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. [Online], available: <http://www.keel.es/>, 2011
- 29 Dai Wei, Zhou Ping, Zhao Da-Yong, Lu Shao-Wen, Chai Tian-You. Hardware-in-the-loop simulation platform for supervisory control of mineral grinding process. *Powder technology*, 2016, **288**: 422–434



代 伟 中国矿业大学信息与控制工程学院副教授. 主要研究方向为复杂工业过程建模, 运行优化与控制. 本文通信作者.

E-mail: weidai@cumt.edu.cn

(DAI Wei Associate professor at the School of Information and control Engineering, China University of Mining and Technology. His research interest covers modeling, operational optimization, and control for complex industrial process. Corresponding author of this paper.)



李德鹏 中国矿业大学信息与控制工程学院硕士研究生. 主要研究方向为数据驱动建模, 机器学习算法.

E-mail: dpli@cumt.edu.cn

(LI De-Peng Master student at the School of Information and Control Engineering, China University of Mining and Technology. His research interest covers data-driving modeling and machine learning algorithms.)



杨春雨 中国矿业大学信息与控制工程学院教授. 于 2009 年获得东北大学博士学位. 主要研究方向为广义系统, 鲁棒控制.

E-mail: chunyuyang@cumt.edu.cn

(YANG Chun-Yu Professor at the School of Information and Control Engineering, China University of Mining and Technology. He received his Ph.D. degree from Northeastern University in 2009. His research interest covers descriptor systems and robust control.)



马小平 中国矿业大学信息与控制工程学院教授. 主要研究方向为过程控制, 网络控制, 故障诊断.

E-mail: xpma@cumt.edu.cn

(MA Xiao-Ping Professor at the School of Information and Control Engineering, China University of Mining and Technology. His research interest covers process control, networked control, and fault detection.)