

DOI: 10.13382/j.jemi.B2306397

遗忘因子随机配置网络驱动自适应切换学习模型^{*}

乔景慧 张 岩 陈宇曦 张开济
(沈阳工业大学机械工程学院 沈阳 110870)

摘 要:随机配置网络(SCNs)具有通用逼近能力和快速建模特性,已成功应用于大数据分析。在 SCN 的基础上,块增量随机配置网络(BSC)使用块增量机制提高训练速度,但增加了模型结构的复杂程度。为了解决上述难题,提出遗忘因子随机配置网络(FSCN-I 和 FSCN-II)驱动的自适应切换学习模型(ASLM)。该模型利用正态分布配置隐含层节点的输入参数。FSCN-I 通过误差值和遗忘因子调整节点块的大小,提高训练速度。FSCN-II 引入节点移除机制降低模型结构的复杂程度。ASLM 由 FSCN-I 和 FSCN-II 构成,两者根据自适应变化的边界随机切换以提高模型的训练速度,并在 FSCN-I 的基础上降低模型结构的复杂程度。最后,通过基础数据集和工业实例,表明该方法的有效性。

关键词: 随机配置网络;遗忘因子;动态隐含层节点;自适应切换学习模型

中图分类号: TN911.7;TP24 **文献标识码:** A **国家标准学科分类代码:** 510.80

Adaptive switching learning model based on forgetting factor stochastic configuration networks

Qiao Jinghui Zhang Yan Chen Yuxi Zhang Kaiji
(School of Mechanical Engineering, Shenyang University of Technology, Shenyang 110870, China)

Abstract: Stochastic configuration networks (SCNs) have been successfully applied to big data analysis with their general approximation capability and fast modeling properties. Based on the SCNs, stochastic configuration networks with block increments (BSC) use block increment mechanism to improve the training speed, but increase the complexity of model structure. To solve the above challenges, an adaptive switching learning model based on forgetting factor stochastic configuration networks (FSCN-I and FSCN-II) with (ASLM) is proposed. FSCN-I adjusts the size of node blocks by error values and forgetting factors to improve the training speed, and FSCN-II introduces a node removal mechanism to reduce the complexity of the model structure. ASLM consists of FSCN-I and FSCN-II, both of which are randomly switched according to the adaptively changing boundaries to improve the training speed of the model and the complexity of the model structure is reduced based on FSCN-I. Finally, the effectiveness of the method is demonstrated with the underlying dataset and industrial examples.

Keywords: stochastic configuration networks; forgetting factor; dynamic hidden nodes; adaptive switching learning model

0 引 言

人工神经网络^[1-4]具有逼近任意非线性系统的能力,广泛应用于数据建模、工业控制等领域^[5],取得了显著的效果^[6]。但当数据量超过一定规模时,使用传统的神经

网络进行数据建模时存在训练效率低、模型结构复杂等问题。因此,建立一种高效的、结构简单的学习模型具有重要的意义。

20 世纪 90 年代,随机神经网络技术得到了发展。随机向量连接网络(random vector function-link networks, RVFLN)具有实现简单,建模速度快的优点。若随机参

数设置不正确, RVFLN 无法以很高的概率逼近目标函数^[7], 设计人工神经网络时无法准确的得到所需要的隐含层节点的个数。Liu 等^[8]提出了一种基于局部线性神经模糊网络来确定网络模型结构。文献^[9]提出了一种改进的带有尖峰神经网络的 Elman 神经网络结构, 使得训练算法具有动态跟踪的能力。张檬等^[10]提出一种单向耦合控制方法构建含有未知参量及未知拓扑结构的驱动-响应复杂网络模型。

Wang 等^[11]提出随机配置网络 (stochastic configuration networks, SCNs), SCNs 算法具有良好的泛化性^[12], 是一种增量学习算法, 在一个可变范围内随机分配隐含层节点的输入参数, 并根据约束不等式增加隐含层节点, 即在训练过程中构建模型的网络结构, 使其具有任意精度逼近能力。为了提高 SCNs 的性能, Zhu 等^[13]提出了两个新的约束不等式, 扩大了约束保持的可能性, 提高了训练速度。文献^[14]针对协作学习范式中的分散数据集, 提出了一种分布式 SCN。文献^[15]将正则化方法引入 SCNs, 提高了 SCNs 的泛化性能。文献^[16]提出了以矩阵为输入特征的 2DSCN, 在图像分类任务中有较好的表现。为解决传统集中式学习在海量数据计算和存储资源的可扩展性和效率方面的局限性, Zhao 等^[17]提出了一种分布式正则化随机配置网络。文献^[18]提出了一种基于混沌麻雀搜索算法 (CSSA) 的随机配置网络, 采用 CSSA 自动给出更优的 SCN 参数。文献^[19]和^[20]分别提出了两种鲁棒 SCN 算法, 用于处理数据受污染的建模问题。Dai 等^[21]提出了基于节点块增量的随机配置网络 (stochastic configuration networks with block increments, BSC), 减少训练模型的计算次数, 提高了模型的训练效率。文献^[22]提出了一种混合并行的随机配置网络, 以降低计算量。

BSC 虽然减少了计算量, 但会产生过多的冗余节点, 增加了模型结构的复杂程度; 过于复杂的结构会导致过拟合, 从而导致性能降低。部分节点的输出权重较小, 增加了模型的复杂性, 但不能提高模型的逼近精度。

针对上述问题, 本文提出了遗忘因子随机配置网络 (forgetting factor stochastic configuration networks, FSCN) 驱动的自适应切换学习模型 (adaptive switching learning model, ASLM), 该模型采用了遗忘因子控制网络权重的遗忘速度来减少已学知识对新知识的干扰, 随机配置网络能够控制系统的状态转移, 从而提高系统的鲁棒性和自适应能力, 该模型可以在非平稳数据流环境下进行自适应学习和切换, 并且具有较好的泛化能力, 同时还提高了模型的训练速度和减小隐含层节点的个数。本文的主要贡献在于:

首先, 使用正态分布配置隐含层节点的输入权重和偏置, 提高模型的训练速度。其次, 提出了两种 FSCN, 其

中 FSCN-I 根据逼近误差值, 将遗忘因子作为误差值的权重来调整节点块的尺寸, 提高训练速度。为移除输出权重较小的隐含层节点, 提出了 FSCN-II。Aghasi 等^[23]通过求解凸优化方法来移除每一层的连接并在每层寻找一组稀疏的权重, 以保持层输入和输出与最初训练的模型一致。Chin 等^[24]提出了一种全局剪枝算法, 把每个通道的局部重要性转化成全局重要性, 从网络模型全局角度对模型进行剪枝。因此, FSCN-II 在 FSCN-I 的基础上建立的节点移除机制, 与其他剪枝方法对权重或冗余通道进行移除不同。在 FSCN-II 中, 每次增加节点后对所有节点的输出权重进行观测; 若其中一个节点的所有的输出权重中的最小值小于设定值, 则将此节点移除, 继续添加新的节点。在不影响模型逼近能力的前提下, 具有简单的模型结构。为结合两种算法的优势, 提出了自适应切换学习模型。具体模型为: 首先, 设定一个边界值, 训练过程中产生一个随机数。若随机数小于边界值, 使用 FSCN-I 训练模型; 否则, 使用 FSCN-II 训练模型。训练过程中两者随机切换, 提高模型的训练速度并在 FSCN-I 的基础上降低模型结构的复杂程度。对所提出的 FSCN 与 ASLM 进行对比实验, 表明其有效性。

1 相关工作

1.1 随机配置网络

SCNs 首先建立具有一个隐含层节点的单层前馈网络 (single-hidden layer feed-forward network, SLFN), 然后根据约束不等式添加隐含层节点构造神经网络。

令 $\Gamma := \{g_1, g_2, g_3, \dots\}$ 表示一组实值函数, $\text{span}(\Gamma)$ 表示 Γ 张成的函数空间; 令 $L_2(D)$ 表示定义在 $D \subset \mathbf{R}^d$ 上的所有勒贝格可测函数 $f = [f_1, f_2, \dots, f_m]: \mathbf{R}^d \rightarrow \mathbf{R}^m$ 的空间。

给定一个目标函数 $f: \mathbf{R}^d \rightarrow \mathbf{R}^m$, 假设存在一个建立好的带有 $L-1$ 个隐含层节点的 SLFN, 那么其输出可以表示为:

$$f_{L-1}(\mathbf{x}) = \sum_{j=1}^{L-1} \beta_j g_j(\mathbf{w}_j^T \mathbf{x} + b_j) \quad (1)$$

式中: g_j 为第 j 个隐含层节点的激活函数, 隐含层节点的输入权值 $\mathbf{w}_j$ 和偏置 b_j 分别在区间 $[-\lambda, \lambda]^d$ 和 $[-\lambda, \lambda]$ 随机生成。其中, λ 在范围控制集 $Y = \{\lambda_1, \lambda_2, \dots, \lambda_{\max}\}$ 中自动生成。隐含层第 j 个节点和 m 个输出节点之间的输出权值表示为 $\beta = [\beta_{j1}, \beta_{j2}, \dots, \beta_{jm}]^T$ 网络的输出误差 e_{L-1} 如式 (2) 所示:

$$e_{L-1} = f - f_{L-1} \quad (2)$$

SCNs 根据下列约束不等式来增加第 L 个隐含层节点

$$\langle e_{L-1,q}, \mathbf{g}_L \rangle^2 \geq b_g^2 \delta_{L,q}, q = 1, 2, \dots, m \quad (3)$$

$$\delta_{L,q} = (1 - r - u_L) \|e_{L-1,q}\|^2, q = 1, 2, \dots, m \quad (4)$$

其中, $\forall \mathbf{g} \in \Gamma, \mathbf{b}_g \in \mathbf{R}^+, 0 < r < 1, u_L = 1 - r/L + 1$ 满足 $\lim_{L \rightarrow +\infty} u_L = 0$ 和 $0 < u_L < (1 - r)$ 。

输出权重采用式(5)计算:

$$\beta_{L,q} = \frac{\langle e_{L-1,q}, \mathbf{g}_L \rangle}{\|\mathbf{g}_L\|^2}, q = 1, 2, \dots, m \quad (5)$$

在每次迭代中,随机生成多个隐含层节点的输入权值和偏置,并使用约束不等式添加一个具有最优输入权重和偏置的节点,直到满足容许误差或无法生成新的节点,模型训练完成。

1.2 块增量随机配置网络(BSC)

BSC 是在 SCNs 的基础上每次训练时添加多个隐含层节点(节点块),提高了模型的训练速度。

引理 1^[21] 给定 $0 < r < 1$ 和一个非负实数序列 $\{u_L\}$, $\lim_{L \rightarrow +\infty} u_L = 0$ 并且 $u_L < 1 - r$ 。对于 $L = 1, 2, \dots, \Delta_k \in \{L\}$, $k = 1, 2, \dots$, 若节点块 $\mathbf{H}_{\Delta_k}$ 满足:

$$\delta_{L,q} = (1 - r - u_L) \|e_{L-\Delta_k,q}\|^2, q = 1, 2, \dots, m \quad (6)$$

$$\langle e_{L-\Delta_k,q}, \mathbf{H}_{\Delta_k} \tilde{\beta}_{\Delta_k,q} \rangle \geq \delta_{L,q}, q = 1, 2, \dots, m \quad (7)$$

然后,使用最小二乘法计算输出权重,如式(8)所示。

$$\beta^* = \operatorname{argmin} \left\| \mathbf{f} - \sum_{j=1}^L \beta_j \mathbf{g}_j \right\| \quad (8)$$

那么 $\lim_{L \rightarrow +\infty} \|\mathbf{f} - \mathbf{f}_{L-\Delta_k}\|$ 恒成立。

将式(6)代入(7)可以得到学习监督机制:

$$\xi_{L,q} = \langle e_{L-\Delta_k,q}, \mathbf{H}_{\Delta_k} \tilde{\beta}_{\Delta_k,q} \rangle - (1 - r - u_L) \langle e_{L-\Delta_k,q}, e_{L-\Delta_k,q} \rangle > 0 \quad (9)$$

其中, $q = 1, 2, \dots, m; \Delta_k$ 为第 k 次节点块的尺寸; $e_{L-\Delta_k} = \mathbf{f} - \mathbf{f}_{L-\Delta_k}$ 表示第 $k-1$ 次训练结束时的模型误差; $\mathbf{H}_{\Delta_k}$ 表示第 k 次迭代时的隐含层输出块,如式(10)所示。

$$\mathbf{H}_{\Delta_k} = [\mathbf{g}_{L-\Delta_k+1}, \dots, \mathbf{g}_L]_{N \times \Delta_k} \quad (10)$$

其中, Δ_k 是节点块的尺寸。根据节点块尺寸计算方法的不同,将 BSC 分为 BSC-I 和 BSC-II 两种。其中, BSC-I 中的 Δ_k 是一个固定值, BSC-II 中的 Δ_k 由模拟退火算法计算得出。

BSC 是建立具有一个隐含层节点的 SLFN。然后,进行多次迭代添加隐含层节点块。在每次迭代中,随机生成多个隐含层节点的输入权值和偏置,并通过学习监督机制(9)添加一个具有最优输入权和偏置的节点块,直到满足容许误差或无法生成新的节点块,模型训练完成。

2 遗忘因子随机配置网络驱动的自适应切换学习模型

BSC 可以提高模型的训练速度,但会增加冗余节点,

增加模型的复杂程度。过于复杂的结构会导致模型的过拟合^[21]。为了解决上述问题,并提高训练速度,使用正态分布随机配置隐含层节点的输入参数,提出了两种遗忘因子随机配置网络(FSCN-I 和 FSCN-II)和 ASLM。

2.1 遗忘因子随机配置网络(FSCN)

1) 正态分布节点配置方法

隐含层节点输入参数包括输入权值 $\mathbf{w}$ 和输入偏置 $\mathbf{b}$ 。传统的 SCN 设计了范围控制集,其中每个参数是平均分布的取值范围。平均分布节点配置方法是根据范围控制集在多个 $[-\lambda, \lambda]$ 区间内生成 $\mathbf{w}$ 和 $\mathbf{b}$ 。这种方法配置的 $\mathbf{w}$ 和 $\mathbf{b}$ 不能超过平均分布的边界。如果范围控制集设计的不合适,或采用不断修改 λ 的方法扩大取值范围,将增加算法的迭代次数,导致模型的训练速度下降。

正态分布的参数为期望 μ 和标准差 σ , μ 和 σ 分别影响生成随机数的平均数和取值的概率,有效地拓展了取值范围。本文使用正态分布代替平均分布生成 $\mathbf{w}$ 和 $\mathbf{b}$,其范围控制参数为 $\lambda_p = \{\mu, \sigma\}$ 。 $\mathbf{w}$ 在 $N(-\mu, \sigma)$ 和 $N(\mu, \sigma)$ 中随机生成, $\mathbf{b}$ 在 $N(0, \sigma)$ 中随机生成。 μ 和 σ 的取值对生成节点的质量有直接影响,范围参数从原来的一个变为两个,取值组合变得复杂,很难设计一组理想的参数。因此,只设计参数 λ_p 的初值和最大迭代次数 n_p ,训练过程中,如果当前的 λ_p 无法生成合适的隐含层节点,则根据误差值对 λ_p 进行更新,如式(11)所示。

$$\lambda_p^* = c_1 \lambda_p + c_2 (1 - e^{\Delta e_k}) \quad (11)$$

其中, c_1 和 c_2 均大于 0; e_k 为第 k 次训练的逼近误差, $\Delta e_k = e_k - e_{k-1}$ 。当 Δe_k 较小时,说明算法在此参数下收敛较慢,当前 λ_p 与理想的 λ_p 的距离较远需要对 λ_p 进行较大的调整以扩大搜索范围。当 Δe_k 较大时,说明算法在此参数下收敛较快,当前 λ_p 在理想的 λ_p 附近只对 λ_p 进行较小的调整即可。

λ_p 最大更新次数 n_p 为人工设置,若达到最大更新次数后,仍然无法产生新的节点,则训练停止。

2) FSCN-I

FSCN-I 是一种块增量学习算法。与 BSC-II 使用网络误差的变化值计算节点块的尺寸不同, FSCN-I 是根据后 N 次迭代的误差值和遗忘因子计算节点块的尺寸。其学习约束过程根据引理 1 获得。节点块的尺寸计算方法如下。

首先,设计如式(12)所示的误差累积变量 E 。

$$E = \sum_{i=1}^{i=N} \theta_i |e_{k-N+i}| \quad (12)$$

其中, $0 \leq \theta_i \leq 1$ 为遗忘因子, N 为观测窗口的尺寸。 θ_i 和 N 为人工设置的量。

函数 ϕ 计算节点块的尺寸 Δ_k 。当 E 较大时, Δ_k 为一个较大的值,使得模型快速收敛。当 E 逼近 0 时, Δ_k 快速逼近 1,以防产生过多的冗余节点。具体计算方法

如式(13)所示。

$$\phi(E) = \begin{cases} \Delta \text{ceil}(e^{E/d} - 1), \phi < M \\ M, \phi \geq M \end{cases} \quad (13)$$

式中: Δ_1 为初始块的大小, ceil 为向上取整函数, $d > 0$, M 为 Δ_k 的上界。

FSCN-I 训练模型的过程简述如下:

1) 给定训练输入输出数据。设定容许误差值 ε 和最大节点数 $L_{\max}$; 最大配置次数 $T_{\max}$; 初始节点块的大小 Δ_1 ; 范围控制 λ_p 的初值; λ_p 最大更新次数 n_p 。

2) 根据式(9), 在 λ_p 设定的区间内随机产生新的隐含层节点块的权值和偏置; 若设定区间内无法产生新的节点块, 则更新 λ_p 。若 n_p 达到最大值, 则训练停止。

3) 通过式(8), 使用全局最小二乘得到输出权重。

4) 计算当前训练误差 e_L 和最大节点数, 是否满足停止标准, 若满足其中一条则训练完成并返回模型参数。若不满足, 则根据式(13) 计算下一个节点块的大小 Δ_k , 返回步骤 2)。

3) FSCN-II

BSC 和 FSCN-I 均为块增量算法, 在一次训练中添加多个节点来减小误差值。然而, 块中的节点对减小误差的贡献一般是不同的。贡献小的节点增加了模型的复杂程度。因此, 需要根据输出权重的值对学习过程进行约束, FSCN-II 采用定理 1 在 FSCN-I 的基础上降低模型结构的复杂程度。

定理 1: 给定 $\tau_0, a \in \mathbf{R}^+, 0 < a \leq 1$, 使得输出权重 $\min[|\beta_{j1}|, |\beta_{j2}|, \dots, |\beta_{jm}|] \geq \tau_0 a^n, \lim_{k \rightarrow \infty} \|H_{\Delta k} \beta_{\Delta k}\| = 0$ 。其中, n 为当 $E < m_1$ 后模型的迭代次数, $n = 1, 2, \dots$ 。 $\beta_{\Delta k}$ 为节点块的输出权重, $H_{\Delta k}$ 为第 k 次迭代时的隐含层输出块, 并使用式(8) 计算输出权重。

可以得到等式 $\lim_{L \rightarrow +\infty} \|f - f_L^*\|$, 其中 $f_L^* = \sum_{j=1}^L \beta_j^* g_j$, $\beta_j^* = [\beta_{j1}^*, \dots, \beta_{jq}^*, \dots, \beta_{jm}^*]^T$ 最优残差为 $e_{L-\Delta k}^* = f - f_L^*$, $e_0^* = f$ 。

证明:

对于序列 $\|e_L^*\|$ 可得:

$$\|e_L^*\|^2 \leq \|\tilde{e}_L\|^2 = \langle e_{L-\Delta k}^* - H_{\Delta k} \tilde{\beta}_{\Delta k, q}, e_{L-\Delta k}^* - H_{\Delta k} \tilde{\beta}_{\Delta k, q} \rangle = \|e_{L-\Delta k}^*\|^2 - \|H_{\Delta k} \tilde{\beta}_{\Delta k}\|^2 \quad (14)$$

由式(14)得:

$$\|e_L^*\| - \|e_{L-\Delta k}^*\| \leq -\|H_{\Delta k}\|^2 \|\tilde{\beta}_{\Delta k}\| \quad (15)$$

进一步:

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

$$\sum_{q=1}^m (\delta_{L, q}^* - \langle e_{L-\Delta k, q}^{*T}, H_{\Delta k} \tilde{\beta}_{\Delta k, q} \rangle) \leq -\|H_{\Delta k}\|^2 \|\tilde{\beta}_{\Delta k}\|^2 \quad (16)$$

根据定理 1, $\min[|\beta_{j1}|, |\beta_{j2}|, \dots, |\beta_{jm}|] \geq \tau_0 a^n$, 可得式(17):

$$\sum_{q=1}^m (\delta_{L, q}^* - \langle e_{L-\Delta k, q}^{*T}, H_{\Delta k} \tilde{\beta}_{\Delta k, q} \rangle) \leq -m \tau_0^2 a^{2n} \|H_{\Delta k}\|^2 \quad (17)$$

式中: 当 $E < m_1$ 时, $\tau_0 a^n$ 不断减小, 进而放宽对输出权重的约束, 可以保证式(17) 成立。

$$\|e_L^*\|^2 \leq r \|e_{L-\Delta k}^*\|^2 + u_L \|e_{L-\Delta k}^*\|^2 \quad (18)$$

在式(18) 中由于 $\lim_{L \rightarrow \infty} \mu_L = 0$, 可得 $\lim_{L \rightarrow \infty} \|e_L^*\| = 0$ 即 $\lim_{L \rightarrow +\infty} \|f - f_L^*\|$, 证明完成。

定理 1 要求第 j 个节点的输出权重中的最小值大于设定的阈值 $\tau_0 a^n$ 。若不满足, 则将此节点移除, 以降低模型结构的复杂程度。当 $E < m_1$ 时, n 不断增加, 使得 $\tau_0 a^n$ 减小, 放宽对输出权重的要求以保证算法的收敛性。其中, n 为 $E < m_1$ 后的模型的迭代次数其初值为 0, $a \in (0, 1)$, τ_0 为输出权重的阈值, τ_0 和 a 为人为设置的值。

如果隐含层节点的输出权重过小, 那么其输出值也是一个较小的值, 说明该节点对提高模型的逼近性能贡献低, 但是增加了模型结构的复杂程度。因此, 对贡献低的节点进行移除并添加新的节点, 以保证模型的逼近能力。根据约束不等式(7) 和定理 1 对模型的结构进行更新, 在保证模型逼近能力的同时, 使模型具有最简单的结构。FSCN-II 在每次迭代中观测隐含层节点, 并移除输出权重小于阈值的隐含层节点, 并添加新的隐含层节点降低模型结构的复杂程度。当阈值设为固定值时, 若设置不当会影响算法的任意精度逼近能力。因此, 当模型误差值接近容许误差时, 适当放宽输出权重的取值范围, 保证模型收敛性和降低模型结构的复杂程度。

2.2 自适应切换学习模型 (ASLM)

为结合 FSCN-I 和 FSCN-II 的优势, 先使用 FSCN-I 训练模型提升训练速度, 然后再切换为 FSCN-II 降低模型结构的复杂程度。但这种方法会导致训练前期添加大量冗余节点, 后期又移除这些冗余节点导致逼近误差升高。为降低逼近误差, 模型需要继续训练, 增加迭代次数降低训练速度。因此, 借鉴比例选择法提出自适应切换学习模型。

自适应切换学习模型是由 FSCN-I 和 FSCN-II 两种算法构成。两者根据产生的随机数进行切换。首先设定一个边界值 $S, 0 < S < 1$ 。然后, 在 $(0, 1)$ 区间内生成一个随机数 R 。若 $R < S$, 则使用 FSCN-I 训练模型; 否则, 使用 FSCN-II 训练模型, 如图 1 训练过程中, 通过自适应

调整 S 选择使用 FSCN-I 或 FSCN-II 的概率。

训练过程中,对 S 自适应更新,如式(19)所示。

$$S^* = S - \frac{z_1 L}{L_{\max}} e^{-z_2 E} \quad (19)$$

式中: S 随 E 和 L 变化;其中, S^* 为更新后的边界值, L 为当前模型的节点数, $L_{\max}$ 为设定的最大节点数, $z_1, z_2 > 0$ 。 S 设定为一个较大值,使模型在训练初期增加使用 FSCN-I 的概率,以提高模型的训练速度;训练前期 E 较大, $e^{-z_2 E}$ 则为一个较小的值,使得后一项的作用变小,增加使用 FSCN-I 训练模型的概率提高模型训练速度。训练过程中 E 的值不断减小,使得 $e^{-z_2 E}$ 的值增加,后一项的作用增大。随着节点数 L 的增加, S 不断减小,增加使用 FSCN-II 的概率减小隐含层节点的个数。

ASLM 的具体实现步骤为:

- 1) 给定训练输入输出数据。设定 $\varepsilon, L_{\max}, T_{\max}, \Delta_1, \lambda_p$ 的初值、 n_p, S 的初值。
- 2) 随机生成 R 并与 S 比较。

3) 若 $R < S$,使用 FSCN-I 训练模型,否则使用 FSCN-II 训练模型。

4) 计算误差值。判断是否满足停止标准,若满足其中一条则训练完成并返回模型参数;否则执行步骤 5)。

5) 采用式(19)更新 S 。返回步骤 2)。

ASLM 由 FSCN-I 和 FSCN-II 两种算法构成,根据引理 1 采用块增量的方法添加隐含层节点是可以实现算法收敛的。进而说明 FSCN-I 是收敛的。FSCN-II 是在 FSCN-I 的基础上增加了隐含层节点移除机制,定理 1 证明采用节点移除机制算法仍具有收敛性。FSCN-I 和 FSCN-II 均是收敛,进而保证了 ASLM 的收敛性。

注 1: 停止标准为模型误差值达到容许误差;模型节点数达到设定的最大节点数;无法生成满足约束不等式的节点。如果训练过程中没有添加约束,很容易导致过拟合。停止标准是使用了早期停止技术避免可能出现的过拟合。3 个准则也是 SCN 算法的训练完成的标准。因此,设置停止标准并不会影响算法的收敛性。

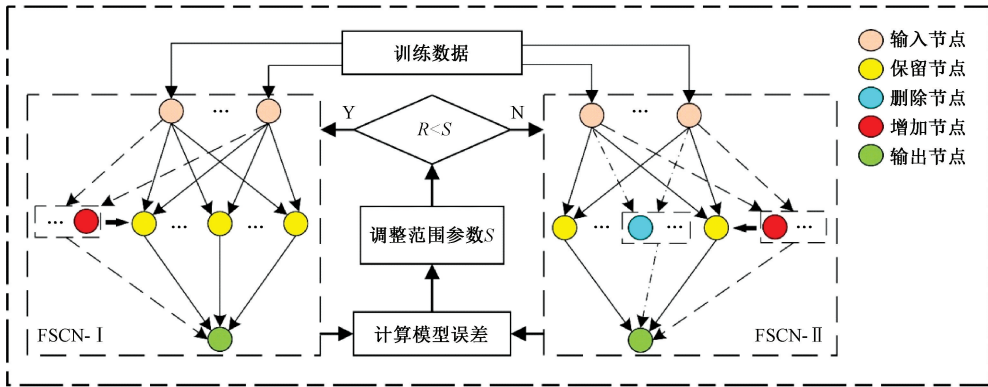


图 1 基于 FSCN 的自适应切换学习模型结构

Fig. 1 Structure of adaptive switching learning model (ASLM) based FSCN

3 实验与分析

在本章中,通过一个函数逼近,4 个真实的回归实例 DB1 (energy efficiency)、DB2 (electrical grid stability simulation)、DB3 (elevators)、DB4 (mv) 和一个工业实例将提出的算法和 SCN-III^[12]、BSC-II^[21] 进行对比分析。所选数据集的详细信息如表 1 所示。选取训练速度和模型的复杂程度作为评价指标。模型的复杂程度用达到容许误差时的隐含层节点数来衡量。仿真实验在 MATLAB 2016a 环境下运行,所用 PC 的 CPU 为 i5, 2.2 GHz, 内存为 8 GB RAM。

首先,对输入输出数据进行归一化处理。激活函数为 Sigmoid 函数。实验的参数选择:最大隐含层节点数

$L_{\max} = 150, r = \{0.9 \ 0.99 \ 0.999 \ \dots\}$ 、观测误差个数为 $N=5$ 、遗忘因子为 $\theta = \{1 \ 0.9 \ 0.8 \ 0.7 \ 0.6\}$ 、 $M=10, m_1 = 1.01\varepsilon, \varepsilon$ 为容许度误差。 λ_p 进行更新算法中的参数为 $c_1=2, c_2=5$ 。其他参数设置在具体的实验里给出。训练时间 t 、节点数 L 、迭代次数 k 、均方根误差 (root mean square error, RMSE) 的平均值和标准差是通过 50 个独立重复试验得到的。逼近函数如式(20)^[12] 所示。

$$y(x) = 0.2e^{-(10x-4)^2} + 0.5e^{-(80x-40)^2} + 0.3e^{-(80x-20)^2}, \quad x \in [0, 1] \quad (20)$$

首先根据式(20)得到的 1 000 个随机样本对模型进行训练。针对逼近函数(20)数据集,选择超参数如下。 $Y = \{1 \ 10 \ 20 \ 50 \ 100 \ 200 \ 400\}$ 。在 BSC-II 中, $\eta = 1$ 。 $\lambda_p = \{5 \ 50\}$ 、 $d = 1, a = 0.8, n_p = 7$, 容许误差为 $\varepsilon = 0.01$ 。

表 1 基础数据集属性

Table 1 Attributes of benchmark dataset

数据集	属性		样本数	任务
	输入	输出		
DB1	8	1	768	回归
DB2	12	1	10 000	回归
DB3	18	1	16 599	回归
DB4	10	1	40 768	回归

针对 4 个数据集模型,选择超参数如下。 $Y = \{0.1$
 $0.2 \ 0.3 \ 0.4 \ 0.5 \ 0.6 \ 0.7 \ 0.8 \ 0.9 \ 1\}$, $T_{\max} =$
20。在 BSC-II 中, $\eta = 1$ 。 $\lambda_p = \{0.1 \ 0.1\}$ 、 $d = 1$ 、 $a =$
 0.8 、 $n_p = 10$ 。容许误差和阈值参数如表 2 所示。

表 2 基础数据集参数

Table 2 Parameters of benchmark dataset

数据集	容许误差 ε	阈值 τ_0
DB1	0.05	1.55
DB2	0.08	0.3
DB3	0.045	8
DB4	0.03	0.2

当数据量超过一定的规模时,模型参数训练的成本会显著提高,增加了模型的训练时间。过于复杂的结构会导致过拟合,从而导致性能降低^[21]。因此,训练时间和模型结构的复杂程度均是衡量算法性能的重要指标。为综合衡量算法对训练时间和模型结构复杂程度的影响,提出一种衡量准则,如式(21)所示。

$$CT = \ln(L) + \ln(\ln(D_v))t \tag{21}$$

式中: L 为模型的节点数, t 是训练时间, D_v 是数据量, CT 表示算法在训练时间和模型结构复杂程度的综合性能。 CT 的值越小说明在训练时间和模型结构复杂程度上算法的性能越好。当数据量较大时应保证算法具有较快训练速度,因此在衡量准则中对 D_v 取两次对数作为 t 的权重。

3.1 正态分布节点配置方法性能验证

本节采用逼近函数(20)和 4 个基础数据集(DB1-DB4),对比平均分布(UD)和两个正态分布(ND1 和 ND2)随机配置节点的性能。其中,ND1 根据范围控制集在多个 $N(0,\lambda)$ 区间内生成 w 和 b 。ND2 为本节中提出的配置方法。UD 和 ND1 的范围控制集参数为 Y ,ND2 的参数为 λ_p 。三者均采用点增量的方式。

图 2 中, L_{UD} 和 t_{UD} 为 UD 方法训练模型得到的节点数和训练时间。 L_{ND1} 和 t_{ND1} 为 ND1 方法训练模型得到的节点数和训练时间。 L_{ND2} 和 t_{ND2} 为 ND2 方法训练模型得到的节点数和训练时间。

从图 2 中可以看出,在最大配置次数($T_{\max}$)不变的情况下,节点数 $L_{ND2} < L_{ND1} < L_{UD}$,训练时间 $t_{ND2} < t_{ND1} < t_{UD}$ 。

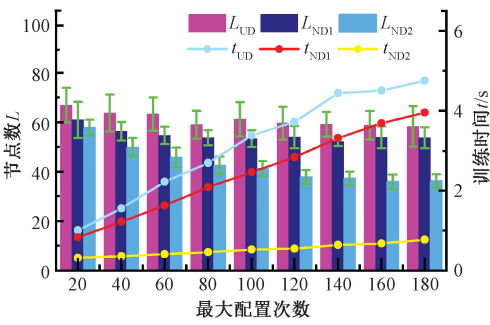


图 2 平均分布和正态分布性能对比

Fig. 2 Performance comparison of uniform distribution and normal distribution

随着 $T_{\max}$ 的增加,三者的差距也不断增大。说明正态分布可以获得更紧凑的模型结构和训练速度。本文提出的正态分布配置节点方法 ND2 的性能优于 UD 和 ND1。

表 3 为 3 种方法在训练集和测试集上得到的 RMSE 的值。可以看出采用本文提出的 ND2 配置节点参数的方法有更强的泛化性。

表 3 UD, ND1 和 ND2 在训练集和测试集中的 RMSE

Table 3 RMSE of UD, ND1and ND2 in training set and test set

$T_{\max}$	方法	训练集		测试集	
		均值	标准差	均值	标准差
20	UD	0.008 8	0.001 2	0.009 1	0.001 4
	ND1	0.009 0	0.001 3	0.009 3	0.001 5
	ND2	0.008 7	0.001 2	0.009 1	0.001 4
80	UD	0.009 1	0.001 1	0.009 4	0.001 3
	ND1	0.008 7	0.001 4	0.009 2	0.001 6
	ND2	0.008 5	0.001 6	0.008 8	0.001 9
140	UD	0.008 9	0.001 2	0.009 2	0.001 3
	ND1	0.008 9	0.000 9	0.009 2	0.001 2
	ND2	0.008 8	0.001 3	0.009 1	0.001 6

表 4 和 5 为 UD,ND1 和 ND2 在基础数据集上的训练时间、节点数、迭代次数和 RMSE 的对比。表 4 中可以看出在 4 个数据集中 ND2 方法的 CT 均是最低的,相比另外两种方法在训练时间和降低模型结构复杂程度上有更好的综合性能。

为得出不同的 λ_p 的取值对模型的训练时间和模型结构复杂程度的影响,在逼近函数(20)数据集上,使用具有不同 λ_p 的取值对模型进行训练, $T_{\max} = 20$,实验结果如图 3 所示。

从图 3(a)和(b)中可以看出随着 λ_p 中 μ 和 σ 两个参数的不断增大节点数和训练时间均不断降低。当 $\lambda_p = \{5 \ 1\}$ 的节点数是最小的,但训练时间较长。而且,实际应用中很难精确找到使得模型节点数最小的组合。因此,为尽量提高训练速度并降低模型结构的复杂程度, λ_p 可取较大值。

表 4 UD, ND1 和 ND2 在基础数据集上的性能对比

Table 4 Performance comparison of UD, ND1 and ND2 on benchmark datasets					
数据集	方法	t	L	k	CT
DB1	UD	0.43	38.44±2.2	38.44±2.2	4.46
	ND1	0.25	39.28±3.5	39.28±3.5	4.14
	ND2	0.25	35.42±2.4	35.42±2.4	4.04
DB2	UD	1.82	46.16±3.9	46.16±3.9	7.87
	ND1	1.32	46.58±4.2	46.58±4.2	6.77
	ND2	1.29	44.50±3.2	44.50±3.2	6.66
DB3	UD	1.84	38.14±3.1	38.14±3.1	7.83
	ND1	1.58	37.04±3.3	37.04±3.3	7.20
	ND2	1.43	37.50±2.8	37.50±2.8	6.88
DB4	UD	4.84	48.58±3.4	48.58±3.4	15.32
	ND1	4.54	50.60±3.5	50.60±3.5	14.65
	ND2	4.19	43.10±3.3	43.10±3.3	13.66

表 5 UD, ND1 和 ND2 的 RMSE 对比
Table 5 RMSE comparison of UD, ND1 and ND2

数据集	算法	训练集		测试集	
		均值	标准差	均值	标准差
DB1	UD	0.049 3	0.000 6	0.070 1	0.014 2
	ND1	0.049 1	0.000 9	0.069 0	0.016 7
	ND2	0.049 1	0.001 0	0.070 3	0.014 7
DB2	UD	0.079 5	0.000 4	0.081 2	0.001 0
	ND1	0.079 4	0.000 5	0.080 7	0.001 3
	ND2	0.079 4	0.000 6	0.080 9	0.001 1
DB3	UD	0.044 9	0.000 1	0.046 3	0.002 0
	ND1	0.044 9	0.000 1	0.045 4	0.001 3
	ND2	0.044 9	0.000 1	0.045 3	0.000 8
DB4	UD	0.029 4	0.000 6	0.029 6	0.000 6
	ND1	0.029 5	0.000 5	0.029 7	0.000 5
	ND2	0.029 4	0.000 7	0.029 4	0.000 8

c_1 、 c_2 是 λ_p 更新过程中的参数。其取值对于算法的性能具有一定的影响。在逼近函数(20)数据集上,使用具有不同 c_1 、 c_2 的取值对模型进行训练。 $T_{\max} = 20$ 。实验结果如图 4 所示。在图 4(a)中可以看出, c_1 、 c_2 的变化对节点数影响较小。根据图 4(b)可知, c_1 、 c_2 的变化对训练时间影响较大。当 c_1 的取值在 1.5~3 时,随 c_2 的增加,训练时间下降的趋势较为稳定。因此建议 c_1 的取值应于 [1.5,3], c_2 可以取较大值。

3.2 FSCN 性能验证

本节仍使用 3.1 节的逼近函数(20)和 4 个基础数据集(DB1-DB4),对比 SCN-III、BSC-II、FSCN-I 和 FSCN-II 四种算法的性能。参数选择为 $T_{\max} = 20$ 。对于 BSC-II、FSCN-I 和 FSCN-II,初始节点块为 $\Delta_1 = 5$ 。

从图 5 中可以看出,FSCN-I 模型添加的隐含层节点比 SCN-III 多,但比 BSC-II 少。虽然 BSC 和 FSCN-I 都使用块增量机制,但 FSCN-I 的块尺寸是随误差值而变化,

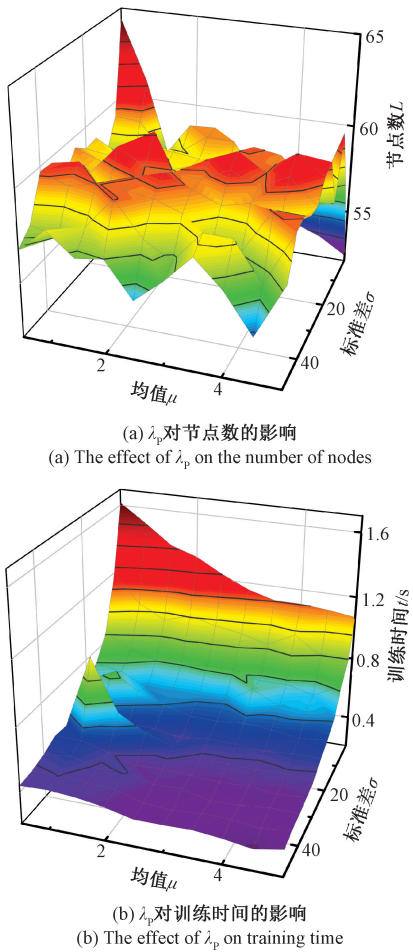


图 3 λ_p 取值对 ND2 方法性能的影响

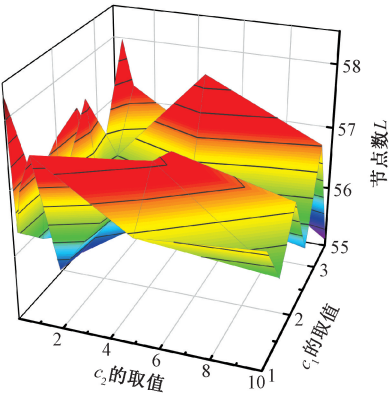
Fig. 3 The influence of λ_p on the performance of ND2 method

更利于降低模型结构的复杂程度。训练过程中,由于 FSCN-I 和 FSCN-II 使用正态分布配置输入权值和偏置,提高了生成节点的质量有效地降低了迭代次数。模型的训练速度得到了大幅提高。表 6 为 4 种算法在训练集和测试集上得到的 RMSE 的值。根据表 6,FSCN-II 泛化能力优于其他 3 种算法。FSCN-I 可以提高模型的训练速度,FSCN-II 可以降低模型结构的复杂程度。

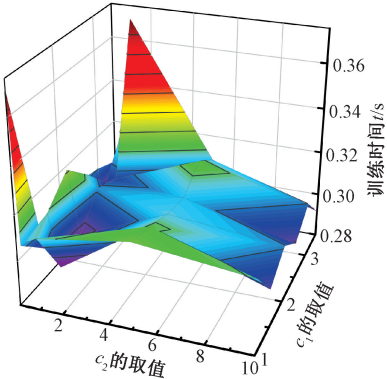
表 6 SCN-III、BSC-II、FSCN-I 和 FSCN-II 的 RMSE 对比
Table 6 RMSE comparison of SCN-III, BSC-II, FSCN-I and FSCN-II

算法	训练集		测试集	
	均值	标准差	均值	标准差
SCN-III	0.008 8	0.001 2	0.009 1	0.001 4
BSC-II	0.018 2	0.001 9	0.017 9	0.001 9
FSCN-I	0.017 3	0.002 7	0.017 3	0.002 8
FSCN-II	0.008 4	0.001 6	0.008 7	0.001 7

图 6 中也可以看出 FSCN-I 的误差值下降速度最快。



(a) c_1 、 c_2 的取值对节点数的影响
(a) The effect of the value of c_1 and c_2 on the number of nodes



(b) c_1 、 c_2 的取值对训练时间的影响
(b) The effect of the value of c_1 and c_2 on the training time

图 4 c_1 、 c_2 的取值对 ND2 方法性能的影响

Fig. 4 The influence of the c_1 , c_2 on the performance of ND2 method

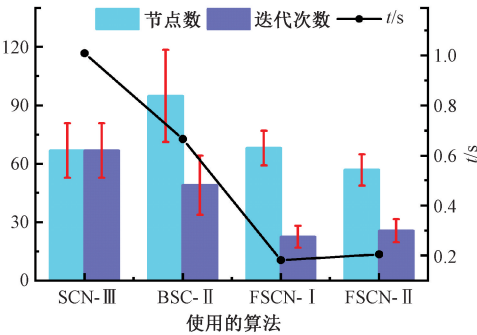


图 5 SCN-III, BSC-II, FSCN-I 和 FSCN-II 性能对比

Fig. 5 Performance comparison of SCN-III, BSC-II, FSCN-I and FSCN-II

说明 FSCN-I 可以有效地提高模型的训练速度和减小隐含层节点的个数。由于 FSCN-II 引入了节点移除机制,使得 FSCN-II 训练的模型的节点最少,但也增加了迭代次数导致其训练速度比 FSCN-I 慢。

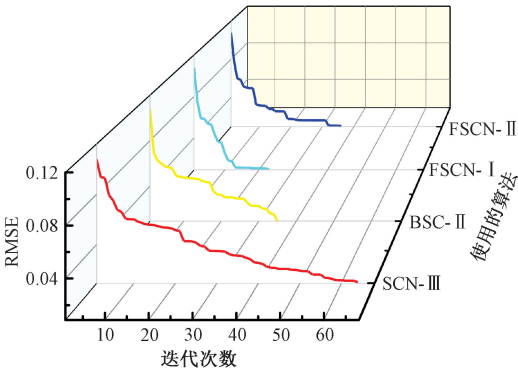


图 6 SCN-III, BSC-II, FSCN-I 和 FSCN-II 误差变化

Fig. 6 The error variation of SCN-III, BSC-II, FSCN-I and FSCN-II

表 7 和 8 为 SCN-III, BSC-II, FSCN-I 和 FSCN-II 在基础数据集上的训练时间、节点数、迭代次数和 RMSE 的对比。表 7 中可以看出 FSCN-II 可以最大程度的降低模型的复杂程度。FSCN-I 低于 SCN-III、BSC-II 和 FSCN-II。说明 FSCN-I 算法可以有效地提高训练速度。由于衡量准则考虑到数据量的影响,当进行大数据建模时,训练速度的权重更大,FSCN-I 的 CT 最小。因此,提出了自适应切换学习模型,使用两种算法训练模型以提高模型的训练速度并在 FSCN-I 的基础上降低模型结构的复杂程度。

表 7 SCN-III, BSC-II, FSCN-I 和 FSCN-II 在基础数据集上的性能对比

Table 7 Performance comparison of SCN-III, BSC-II, FSCN-I and FSCN-II on benchmark datasets

数据集	方法	t	L	k	CT
DB1	SCN-III	0.43	38.44±2.2	38.44±2.2	4.46
	BSC-II	0.22	49.12±4.4	20.06±2.2	4.31
	FSCN-I	0.16	35.48±2.9	24.08±3.1	3.87
	FSCN-II	0.16	34.48±2.6	24.88±2.6	3.84
DB2	SCN-III	1.82	46.16±3.9	46.16±3.9	7.87
	BSC-II	1.11	56.24±4.3	23.62±2.1	6.49
	FSCN-I	0.89	46.24±2.7	26.52±2.4	5.81
	FSCN-II	0.92	45.10±2.9	27.04±2.6	5.85
DB3	SCN-III	1.84	38.14±3.1	38.14±3.1	7.83
	BSC-II	1.39	47.04±6.2	19.02±3.1	7.01
	FSCN-I	1.33	38.42±2.8	31.72±2.8	6.67
	FSCN-II	1.38	37.56±3.0	32.24±3.0	6.76
DB4	SCN-III	4.84	48.58±3.4	48.58±3.4	15.32
	BSC-II	4.17	60.48±4.6	25.74±2.3	13.95
	FSCN-I	3.49	44.38±3.1	29.20±3.1	12.04
	FSCN-II	3.51	43.96±3.2	29.46±3.3	12.08

3.3 自适应切换学习模型性能对比

本节对 FSCN-I 和 FSCN-II 进行组合,设计了 4 种自适应切换学习模型(ASLM)。ASLM(1)为 FSCN-I $\Delta_1 = 5$

表 8 SCN-III, BSC-II, FSCN-I 和 FSCN-II 的 RMSE 对比

Table 8 RMSE comparison of SCN-III, BSC-II, FSCN-I and FSCN-II					
数据集	算法	训练集		测试集	
		均值	标准差	均值	标准差
DB1	SCN-III	0.049 3	0.000 6	0.070 1	0.014 2
	BSC-II	0.048 7	0.001 1	0.068 6	0.019 3
	FSCN-I	0.049 1	0.001 1	0.072 8	0.013 8
	FSCN-II	0.049 3	0.000 7	0.071 5	0.014 3
DB2	SCN-III	0.079 5	0.000 4	0.081 2	0.001 0
	BSC-II	0.079 1	0.000 7	0.080 7	0.001 1
	FSCN-I	0.079 1	0.000 7	0.080 9	0.001 3
	FSCN-II	0.079 2	0.000 7	0.080 5	0.001 3
DB3	SCN-III	0.044 9	0.000 1	0.046 3	0.002 0
	BSC-II	0.044 9	0.000 1	0.046 3	0.001 8
	FSCN-I	0.044 9	0.000 1	0.045 9	0.001 3
	FSCN-II	0.044 9	0.000 1	0.045 5	0.001 3
DB4	SCN-III	0.029 4	0.000 6	0.029 6	0.000 6
	BSC-II	0.029 2	0.000 7	0.029 2	0.000 8
	FSCN-I	0.029 4	0.000 5	0.029 5	0.000 6
	FSCN-II	0.029 4	0.000 5	0.029 5	0.000 5

和 FSCN-II $\Delta_1 = 5$ 的组合; ASLM(2) 为 FSCN-I $\Delta_1 = 10$ 和 FSCN-II $\Delta_1 = 5$ 的组合; ASLM(3) 为 FSCN-I $\Delta_1 = 5$ 和 FSCN-II $\Delta_1 = 10$ 的组合; ASLM(4) 为 FSCN-I $\Delta_1 = 10$ 和 FSCN-II $\Delta_1 = 10$ 的组合, 并与 SCN-III 和 BSC-II 在 4 个数据集上的性能进行对比。自适应切换学习模型的参数为 $S = 0.7, z_1 = 0.5, z_2 = 5$ 。初始节点块尺寸分别为 $\Delta_1 =$ 和 10。

SCN-III, BSC-II, ASLM(1), ASLM(2), ASLM(3) 与 ASLM(4) 性能对比如图 7 所示。4 种 ASLM 的迭代次数比 SCN-III 少, 训练速度比 SCN-III 快, 两者节点数大致相当。虽然 BSC-II 的迭代次数更少, 但 ASLM(2) 和 ASLM(4) 的训练速度比 BSC-II 快, 其原因是 BSC-II 隐含层节点过多导致计算伪逆的速度下降。而且, 4 种 ASLM 训练模型的隐含层节点数要比 BSC-II 少, ASLM 的模型结构更为简单。

表 7 中 FSCN-I 和 FSCN-II 的节点块均为 5, 与表 10 中的 ASLM(1) 相比, ASLM 方法在 FSCN-I 的基础上降低模型结构的复杂程度。通过在 4 个基础数据上对比 ASLM(4) 的 CT 值最小, 在模型复杂程度和训练速度方面具有最好的综合性能。表 9 为 SCN-III, BSC-II 和 ASLM 在训练集和测试集上得到的 RMSE 的值。

3.4 工业案例

将所提方法应用到新型干法水泥熟料生产过程^[25], 根据文献[25]对影响生产过程的各变量之间关系的分析, 建立以分解炉温度 $T_c(t)$ 、预热器 C5 出口温度 $T_{c5}(t)$ 、预热器 C5 锥部压力 $P_{c5}(t)$ 、下料管温度 $T_{bp}(t)$ 、

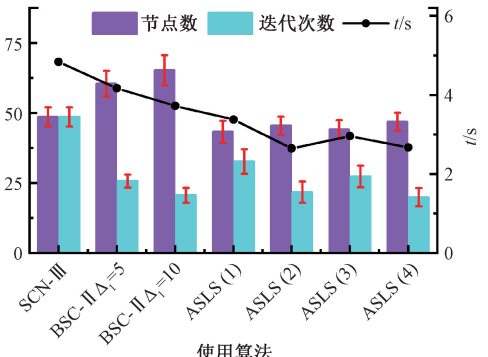
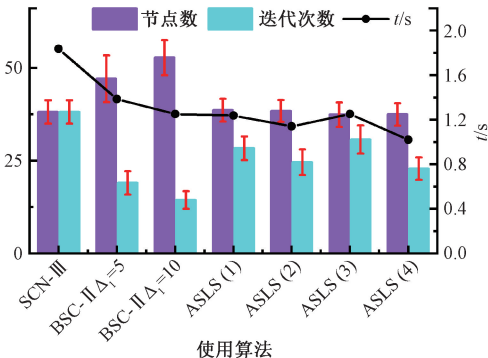
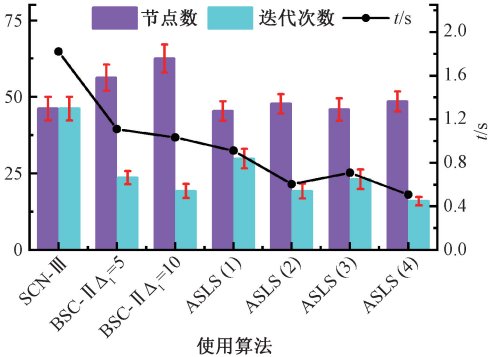
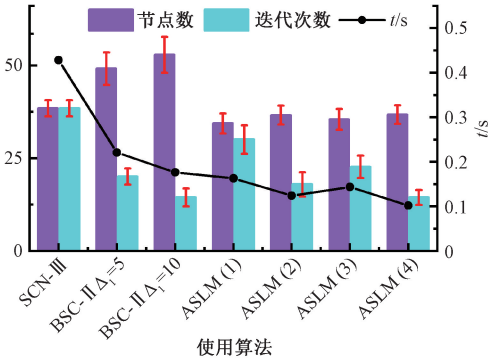


图 7 SCN-III、BSC-II 和 ASLM 性能对比

Fig. 7 Performance comparison of SCN-III, BSC-II and ASLM 分解炉给煤 $u(t)$ 和三次风温度 $T_{ia}(t)$ 为输入, 以生料分解率 $\gamma(t)$ 为输出的软测量模型, 如式(22)所示。

表 9 ASLM 的 RMSE 对比
Table 9 RMSE comparison of ASLM

数据集	算法	训练集		测试集	
		均值	标准差	均值	标准差
DB1	ASLM (1)	0.049 0	0.001 0	0.070 9	0.021 0
	ASLM (2)	0.049 1	0.001 0	0.070 7	0.014 9
	ASLM (3)	0.048 9	0.001 4	0.073 8	0.013 2
	ASLM (4)	0.049 2	0.000 7	0.069 8	0.011 7
DB2	ASLM (1)	0.079 0	0.000 7	0.080 6	0.001 0
	ASLM (2)	0.079 1	0.000 8	0.080 2	0.001 3
	ASLM (3)	0.078 9	0.000 7	0.080 6	0.001 1
	ASLM (4)	0.079 0	0.000 9	0.080 7	0.001 2
DB3	ASLM (1)	0.044 9	0.000 1	0.046 9	0.003 1
	ASLM (2)	0.044 9	0.000 1	0.046 0	0.002 3
	ASLM (3)	0.044 9	0.000 1	0.045 4	0.001 0
	ASLM (4)	0.044 9	0.000 1	0.045 7	0.002 8
DB4	ASLM (1)	0.029 3	0.000 5	0.029 4	0.000 6
	ASLM (2)	0.029 5	0.000 5	0.029 6	0.000 5
	ASLM (3)	0.029 3	0.000 5	0.029 4	0.000 5
	ASLM (4)	0.029 4	0.000 7	0.029 5	0.000 7

表 10 ASLM 在基础数据集上的性能对比
Table 10 Performance comparison of ASLM on benchmark datasets

数据集	方法	t	L	k	CT
DB1	ASLM (1)	0.16	34.38±2.7	30.04±3.8	3.84
	ASLM (2)	0.12	36.60±2.6	17.94±3.3	3.83
	ASLM (3)	0.14	35.46±2.8	22.68±3.0	3.83
	ASLM (4)	0.10	36.74±2.5	14.40±2.0	3.79
DB2	ASLM (1)	0.91	45.34±3.2	29.84±3.2	5.83
	ASLM (2)	0.60	47.72±3.2	19.26±2.5	5.20
	ASLM (3)	0.71	45.88±3.6	23.10±3.2	5.40
	ASLM (4)	0.51	48.48±3.2	15.92±1.4	5.01
DB3	ASLM (1)	1.24	38.06±3.0	35.14±3.0	6.46
	ASLM (2)	1.14	37.74±3.0	28.98±3.5	6.22
	ASLM (3)	1.25	37.76±2.9	31.34±3.3	6.47
	ASLM (4)	1.02	38.32±2.9	27.84±3.0	5.97
DB4	ASLM (1)	3.62	43.70±3.9	31.98±3.9	12.33
	ASLM (2)	3.03	46.72±3.3	23.00±4.0	11.00
	ASLM (3)	3.39	45.92±3.9	27.38±4.2	11.84
	ASLM (4)	2.67	47.06±3.3	19.22±3.3	10.16

$$\gamma(t)=f(T_c(t),T_{c5}(t),P_{c5}(t),T_{pb}(t),u(t),T_{ta}(t))$$

(22)

图 8 为生料分解工艺过程。其中,TT,PP 分别为温度和压力传感器,Ci 为第 i 个预热器。在实验中,随机选择的训练数据集不少于整个样本的 75%,以尽可能覆盖所有原始样本空间的特征。在文献[25]所述的正常运行的工况中采集数据样本 3 245 个,其中 2 600 个训练样本和 645 个测试样本。将训练和测试样本进行归一化处理,数据的采样周期为 1 s。模型的参数为 $\Upsilon=\{1\ 10\ 20\ 50\ 100\ 200\}$, $T_{\max}=40$, $\varepsilon=0.01$ 。在 BSC-II 中,初始节点

块为 $\Delta_1=5,\eta=1$,对于 FSCN-I 和 FSCN-II, $\Delta_1=5,\lambda_p=\{0.1\ 0.1\}$, $d=0.16,n_p=6,S=0.7,z_1=0.5,z_2=5$,阈值 $\tau=0.01,\alpha=0.8$,实验结果如表 11、12 和图 9 所示。

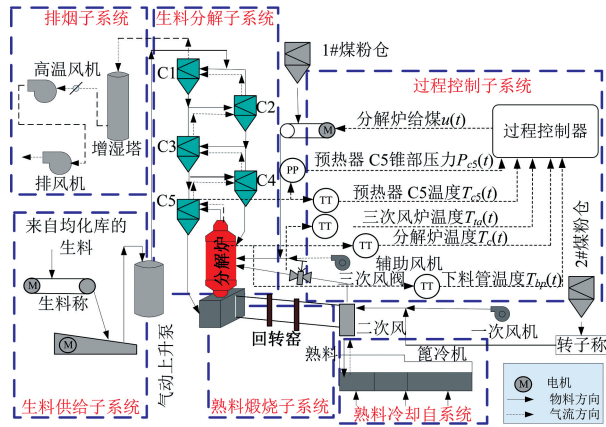


图 8 生料分解过程工艺流程

Fig. 8 Process flow diagram for raw meal calcination process

表 11 SCN-III, BSC-I, FSCN-, FSCN-II 和 ASLM 性能对比
ASLM 性能对比

Table 11 Performance comparisons among SCN-III, BSC-II, FSCN-I, FSCN-II and ASLM

算法	t	L	k	CT
SCN-III	0.78	36.78±9.2	36.78±9.2	5.21
BSC-II	0.40	40.82±10.3	16.02±5.2	4.53
FSCN-I	0.36	34.48±3.8	16.48±3.8	4.28
FSCN-II	0.46	32.60±3.0	20.68±4.6	4.43
ASLM (1)	0.43	35.48±3.4	18.16±4.6	4.46
ASLM (2)	0.12	34.04±2.3	6.44±5.2	3.78
ASLM (3)	0.35	32.96±2.9	15.92±5.5	4.22
ASLM (4)	0.08	35.80±1.8	4.28±2.6	3.74

表 12 SCN-III, BSC-II, FSCN-I, FSCN-II 和 ASLM 的 RMSE 对比

Table 12 RMSE comparison among SCN-III, BSC-II, FSCN-I, FSCN-II and ASLM

算法	训练集		测试集	
	均值	标准差	均值	标准差
SCN-III	0.009 0	0.001 0	0.009 1	0.001 2
BSC-II	0.008 9	0.000 8	0.010 5	0.001 4
FSCN-I	0.007 9	0.000 7	0.008 5	0.001 2
FSCN-II	0.008 0	0.000 7	0.008 2	0.001 8
ASLM (1)	0.008 2	0.000 9	0.008 4	0.001 9
ASLM (2)	0.008 3	0.000 8	0.008 4	0.001 5
ASLM (3)	0.008 3	0.000 9	0.008 5	0.001 4
ASLM (4)	0.007 9	0.000 7	0.008 6	0.001 5

表 11 中可以看出两种 FSCN 和 4 种 ASLM 在训练速度和模型的复杂程度上均优于 SCN-III,虽然 FSCN-II 和 ASLM (1) 的训练速度比 BSC-II 慢,但获得了更紧凑的模型结构。在工业数据集中,根据 CT 值可知 ASLM (4) 在

提高训练速度和降低模型结构的复杂程度上有更好的综合性能。为便于观察,图 9 只显示了 300 个测试数据。从图 9 中可以看出,模型输出值与实际值基本一致,可以准确估计生料分解率,表 12 为 5 种算法的 RMSE 的值。

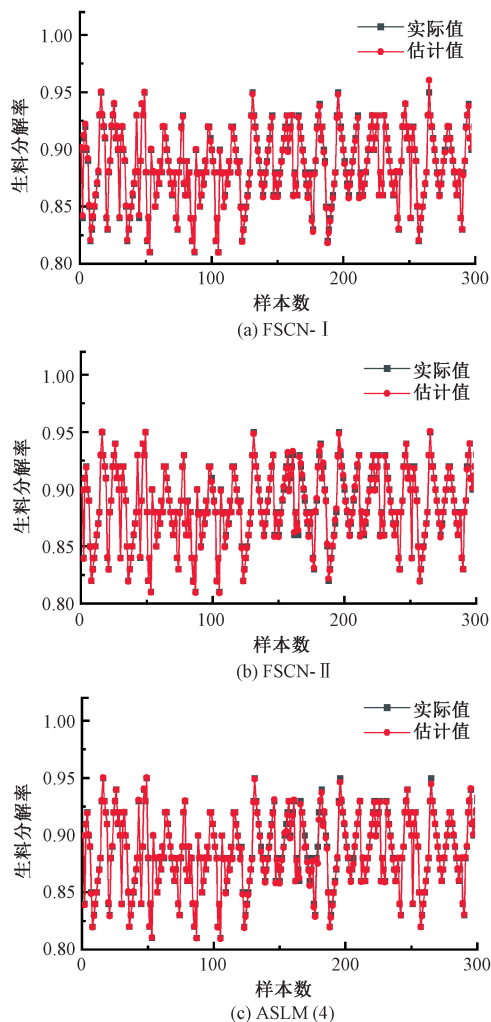


图 9 模型逼近特性

Fig. 9 Approximation characteristics of the model

4 结 论

本文提出了遗忘因子随机配置网络驱动的自适应切换学习模型,使用正态分布随机配置隐含层节点输入参数,提高了模型的训练速度。FSCN-I 根据逼近误差和遗忘因子计算节点块的尺寸,既提高了训练速度又保证了模型的收敛性;FSCN-II 具有节点移除机制,降低模型结构的复杂程度。自适应切换学习模型随机使用 FSCN-I 或 FSCN-II 训练模型,提高了训练速度并在 FSCN-I 的基础上降低模型结构的复杂程度。实际数据仿真研究表明:与 SCNs 和 BSC 相比,本文所提出的方法在提高模型的训练速度和降低模型结构的复杂程度方面,具有良好

的综合性能。本文使用离线数据对模型进行了测试,未来可以将该模型应用于实际问题中,以验证其在实际场景中的性能和可用性。

参考文献

- [1] 蔡奇志, 苗荣霞, 樊逸飞, 等. 基于粒子群优化神经网络的电梯群控算法[J]. 国外电子测量技术, 2019, 38(5): 114-119.
CAI Q ZH, MIAO R X, FAN Y F, et al. Elevator group control system based on improved PSO [J]. Foreign Electronic Measurement Technology, 2019, 38(5): 114-119.
- [2] 李渊朴, 王秀玲. 基于改进遗传算法和神经网络的大棚环境预测[J]. 电子测量技术, 2020, 43(7): 46-49, 166.
LI Y P, WANG X L. Greenhouse environment prediction based on improved genetic algorithm and neural network[J]. Electronic Measurement Technology, 2020, 43(7): 46-49, 166.
- [3] 赵泽宇, 杜明星. 多数据驱动人工神经网络的 IGBT 结温在线估计方法[J]. 电子测量与仪器学报, 2022, 36(7): 223-229.
ZHAO Z Y, DU M X. On line estimation of IGBT junction temperature based on multi data driven artificial neural network [J]. Journal of Electronic Measurement and Instrumentation, 2022, 36(7): 223-229.
- [4] 张志勇, 陆金桂, 张猛. 基于 WOA-BP 神经网络的磨煤机出粉量估算[J]. 电子测量技术, 2022, 45(22): 157-161.
ZHANG ZH Y, LU J G, ZHANG M. Estimation of powder output of coal mill based on WOA-BP neural network[J]. Electronic Measurement Technology, 2022, 45(22): 157-161.
- [5] 付东学, 赵希梅. 基于神经网络观测器的反推终端滑模位置控制[J]. 控制理论与应用, 2023, 40(1): 132-138.
FU D X, ZHAO X M. Backstepping terminal sliding mode position control based on neural network observer [J]. Control Theory & Applications, 2023, 40(1): 132-138.
- [6] 刘钊, 孙洁娣, 温江涛. 基于多层面压缩深度神经网络的轴承故障诊断[J]. 电子测量与仪器学报, 2022, 36(7): 189-198.
LIU ZH, SUN J D, WEN J T. Bearing fault diagnosis based on multi-level compression deep neural network [J]. Journal of Electronic Measurement and Instrumentation, 2022, 36(7): 189-198.
- [7] LI M, WANG D. Insights into randomized algorithms for neural networks: Practical issues and common pitfalls[J]. Information Sciences, 2017, 382: 170-178.

- [8] LIU Z, XU J, FANG H. Extracting inherent model structures and identifying parameters of time-varying systems using local linear neuro-fuzzy networks [J]. IEEE Transactions on Fuzzy Systems, 2022, 30 (1): 233-247.
- [9] AL-JAMALI N, ALRAWESHIDY H S. Modified elman spike neural network for identification and control of dynamic system [J]. IEEE Access, 2020, 8: 61246-61254.
- [10] 张檬, 韩敏. 基于单向耦合法的 uncertain 复杂网络间有限时间同步 [J]. 自动化学报, 2021, 47 (7): 1624-1632.
- ZHANG M, HAN M. Finite-time synchronization between uncertain complex networks based on unidirectional coupling method [J]. Acta Automatica Sinica, 2021, 47 (7): 1624-1632.
- [11] WANG D H, LI M. Stochastic configuration networks: fundamentals and algorithms [J]. IEEE Transaction on Cybernetics, 2017, 47 (10): 3466-3479.
- [12] 乔景慧, 潘忠, 熊宁康, 等. 数据与机理驱动的检定系统建模与参数辨识 [J]. 仪器仪表学报, 2022, 43 (6): 280-289.
- QIAO J H, PAN ZH, XIONG N K, et al. Modeling and parameter identification integrated data and mechanism for verification system [J]. Chinese Journal of Scientific Instrument, 2022, 43 (6): 280-289.
- [13] ZHU X H, FENG X C, WANG W W, et al. A further study on the inequality constraints in stochastic configuration networks [J]. Information Sciences, 2019, 487: 77-83.
- [14] WU A, WANG D H. Distributed stochastic configuration networks with cooperative learning paradigm [J]. Information Sciences, 2020, 540: 1-16.
- [15] 王前进, 杨春雨, 马小平, 等. 基于随机配置网络的井下供给风量建模 [J]. 自动化学报, 2021, 47 (8): 1963-1975.
- WANG Q J, YANG CH Y, MA X P, et al. Underground airflow quantity modeling based on SCN [J]. Acta Automatica Sinica, 2021, 47 (8): 1963-1975.
- [16] LI M, WANG D H. 2-D stochastic configuration networks for image data analytics [J]. IEEE Transaction on Cybernetics, 2021, 51 (1): 359-372.
- [17] ZHAO L J, ZOU S D, HUANG M Z, et al. Distributed regularized stochastic configuration networks via the elastic net [J]. Neural Comput & Applic, 2021, 33: 3281-3297.
- [18] ZHANG C, DING S. A stochastic configuration network based on chaotic sparrow search algorithm [J]. Knowledge-Based Systems, 2021, 220: 106924.
- [19] WANG D H, LI M. Robust stochastic configuration networks with kernel density estimation for uncertain data regression [J]. Information Sciences, 2017, 412-413: 210-222.
- [20] LI M, HUANG C Q, WANG D H. Robust stochastic configuration networks with maximum correntropy criterion for uncertain data regression [J]. Information Sciences, 2018, 473: 73-86.
- [21] DAI W, LI D P, ZHOU P, et al. Stochastic configuration networks with block increments for data modeling in process industries [J]. Information Sciences, 2019, 484: 367-386.
- [22] DAI W, ZHOU X, LI D S, et al. Hybrid parallel stochastic con-figuration networks for industrial data analytics [J]. IEEE Transactions on Industrial Informatics, 2022, 18 (4): 2331-2341.
- [23] AGHASI A, ABDI A, NGUYEN N, et al. Net-trim: Convex pruning of deep neural networks with performance guarantee [J]. Advances in Neural Information Processing Systems, 2017: 3177-3186.
- [24] CHIN T W, DING R, ZHANG C, et al. Towards efficient model compression via learned global ranking [J]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 1518-1528.
- [25] 乔景慧, 柴天佑. 数据与模型驱动的水泥生料分解率软测量模型 [J]. 自动化学报, 2019, 45 (8): 1564-1578.
- QIAO J H, CHAI T Y. Data and model-based soft measurement model of cement raw meal decomposition ratio [J]. Acta Automatica Sinica, 2019, 45 (8): 1564-1578.

作者简介



乔景慧 (通信作者), 2002 年于沈阳建筑大学获得学士学位, 2005 年于东北大学获得硕士学位, 2012 年于东北大学获得博士学位, 现为沈阳工业大学副教授, 主要研究方向为复杂工业过程建模与智能控制、智能视觉与机器学习。

E-mail: qiaojh2002@163.com

Qiao Jinghui received his B. Sc. degree from Shenyang Jianzhu University in 2002, M. Sc. degree from Northeastern University in 2005, and Ph. D. degree from Northeastern University in 2012, respectively. Now he is an associate professor with the School of Mechanical Engineering, Shenyang University of Technology. His main research interests include modeling and intelligent control for complex industry systems, intelligent vision and machine learning.



张岩, 2020 年于沈阳工业大学机械工程学院获得学士学位, 现为沈阳工业大学硕士研究生, 主要研究方向为工业控制、机器视觉和深度学习。

E-mail: 2803658353@qq. com

Zhang Yan received his B. Sc. degree from the School of Mechanical Engineering, Shenyang University of Technology in 2020. Now he is a M. Sc. candidate at Shenyang University of Technology. His main research interests include industrial control, machine vision and deep learning.



陈宇曦, 2020 年于沈阳工业大学机械工程学院获得学士学位, 现为沈阳工业大学硕士研究生, 主要研究方向为机器学习和复杂机电系统建模。

E-mail: cyx_double@163. com

Chen Yuxi received his B. Sc. degree from the School of Mechanical Engineering, Shenyang University of Technology in 2020. Now he is a M. Sc. candidate at Shenyang University of Technology. His main research interests include machine learning and modeling of complex electromechanical systems.



张开济, 2021 年于沈阳工业大学机械工程学院获得学士学位, 现为沈阳工业大学硕士研究生, 主要研究方向为机器学习与智能制造。

E-mail: 2248074858@163. com

Zhang Kaiji received his B. Sc. degree from the School of Mechanical Engineering, Shenyang University of Technology in 2021. Now he is a M. Sc. candidate at Shenyang University of Technology. His main research interests include machine learning and intelligent manufacturing.