

# 一种基于正则优化的批次继承极限学习机算法

刘彬<sup>①</sup> 杨有恒<sup>①</sup> 赵志彪<sup>②</sup> 吴超<sup>②</sup> 刘浩然<sup>②</sup> 闻岩<sup>①</sup>

<sup>①</sup>(燕山大学电气工程学院 秦皇岛 066004)

<sup>②</sup>(燕山大学信息科学与工程学院 秦皇岛 066004)

**摘要:** 极限学习机(ELM)作为一种新型神经网络, 具有极快的训练速度和良好的泛化性能。针对极限学习机在处理高维数据时计算复杂度高, 内存需求巨大的问题, 该文提出一种批次继承极限学习机(B-ELM)算法。首先将数据集均分为不同批次, 采用自动编码器网络对各批次数据进行降维处理; 其次引入继承因子, 建立相邻批次之间的关系, 同时结合正则化框架构建拉格朗日优化函数, 实现批次极限学习机数学建模; 最后利用MNIST, NORB和CIFAR-10数据集进行测试实验。实验结果表明, 所提算法具有较高的分类精度, 并且有效降低了计算复杂度和内存消耗。

**关键词:** 极限学习机; 高维数据; 批次学习; 继承因子; 正则化

中图分类号: TN911.7; TP391

文献标识码: A

文章编号: 1009-5896(2020)07-1734-09

DOI: 10.11999/JEIT190502

## A Batch Inheritance Extreme Learning Machine Algorithm Based on Regular Optimization

LIU Bin<sup>①</sup> YANG Youheng<sup>①</sup> ZHAO Zhibiao<sup>②</sup> WU Chao<sup>②</sup>

LIU Haoran<sup>②</sup> WEN Yan<sup>①</sup>

<sup>①</sup>(School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China)

<sup>②</sup>(School of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China)

**Abstract:** As a new type of neural network, Extreme Learning Machine (ELM) has extremely fast training speed and good generalization performance. Considering the problem that the Extreme Learning Machine has high computational complexity and huge memory demand when dealing with high dimensional data, a Batch inheritance Extreme Learning Machine (B-ELM) algorithm is proposed. Firstly, the dataset is divided into different batches, and the automatic encoder network is used to reduce the dimension of each batch. Secondly, the inheritance factor is introduced to establish the relationship between adjacent batches. At the same time, the Lagrange optimization function is constructed by combining the regularization framework to realize the mathematical modeling of batch ELM. Finally, the MNIST, NORB and CIFAR-10 datasets are used for the test experiment. The experimental results show that the proposed algorithm not only has higher classification accuracy, but also reduces effectively computational complexity and memory consumption.

**Key words:** Extreme Learning Machine(ELM); High dimensional data; Batch learning; Inheritance factor; Regularization

### 1 引言

极限学习机(Extreme Learning Machine, ELM)是一种高效的单隐层前馈神经网络(Single hidden Layer Feedforward neural Network, SLFN), 其输入权重和隐藏层偏置是随机产生的,

只需要设定隐藏层节点个数就可以快速有效地完成网络的训练<sup>[1-3]</sup>。然而, 随着大数据时代的到来, ELM算法对高维数据的处理面临巨大挑战。主要表现为: 在低维数据集上表现良好而在高维数据集上学习性能却急剧下降。其主要原因为: (1)维数增加导致数据体积急剧膨胀而占据大量的内存, 降低了训练效率; (2)高维数据中包含大量冗余信息对特征学习造成干扰, 增加了计算复杂度<sup>[4]</sup>。因此, 研究大数据背景下新型极限学习机算法对提高学习效率和节省算法研究成本具有十分重要的现实意义。

收稿日期: 2019-07-05; 改回日期: 2019-12-12; 网络出版: 2019-12-20

\*通信作者: 刘彬 liubin@ysu.edu.cn

基金项目: 河北省自然科学基金(F2019203320, E2018203398)

Foundation Items: The Natural Science Foundation of Hebei Province (F2019203320, E2018203398)

针对上述问题，很多学者以特征降维方法为基础，从不同方面提出了众多改进的ELM，例如：文献[5]基于主成分分析法，提出一种主成分分析极限学习机(Principal Component Analysis ELM, PCA-ELM)，减少了输入数据的维度，提高了算法的鲁棒性。文献[6]以核函数为特征映射，提出一种加权极限学习机(Weighted ELM, W-ELM)，解决了数据分布不均衡的问题，降低了数据搜索空间。文献[7]基于随机投影法(Random Projection, RP)，提出一种部分连接的极限学习机(Partially Connected ELM, PC-ELM)，减少了训练参数，在计算复杂度和分类精度之间达到了良好的平衡。文献[8]结合自动编码器(Auto Encoder, AE)，提出堆栈式自动编码极限学习机(Stacked Auto Encoder ELM, SAE-ELM)，实现了更好的稀疏性表达，提高了算法的学习速度。

上文提到的ELM算法，在一定程度上取得较为成功的改进，但仍具有一定的局限性，其中学者们在建立ELM算法模型时，大都采用整批样本数据进行建模，这在一次训练过程中，必定会伴随高维数据的存储和调取，使其计算量呈指数增长，从而导致维数灾难<sup>[4]</sup>。目前，现代深度神经网络训练通常基于批次进行随机梯度下降(Stochastic Gradient Descent, SGD)，将训练集划分为一些固定大小的批次，每个批次在一个训练时期内按顺序处理，使用批次训练已被证明可提供更好的泛化性能和更小的内存占用<sup>[9,10]</sup>。Liang等人<sup>[11]</sup>结合批次学习的思想提出一种在线序列极限学习机(Online Sequential ELM, OS-ELM)，不仅可以逐个训练数据，还可以逐批训练数据，并丢弃已经完成训练的数据，大大节省了内存空间。OS-ELM使用批次学习可以实时根据新到来的数据对输出层权重进行矫正更新，具有更好的泛化性能，这为高维数据背景下ELM的研究增添了一个全新的视角。

批次学习为神经网络训练提供较低的内存需求，并提供更好的计算效率。因此，本文在深入研究ELM的基础上引入批次学习的思想，提出一种能融合二者优势的批次继承极限学习机(Batch inheritance ELM, B-ELM)算法，其创新点包括：(1)将高维数据集平均分割为可以独立优化求解的不同批次，利用极限学习机自动编码器(ELM-AE)分批次处理，降低数据维度，减少计算复杂度；(2)提出继承因子，建立各批次的隐藏层权重联系，通过正则化代价函数求取新的隐藏层输出权值，最大程度上继承各批次的学习特征。与深度神

经网络比较，本文使用的ELM模型结构简单，训练速度快，在处理一些简单场景的分类任务时同样能够获得较高的精度，但是在复杂场景中特征提取能力仍然有限，且在处理相同任务时往往需要更多节点，造成模型训练过程中矩阵维数过高，导致维数灾难。为此，本文探讨了ELM算法的批次学习方法，并在3个标准数据集上进行了大量实验，结果表明所提算法能够缓解高维计算问题，降低内存消耗。

## 2 极限学习机-自动编码器

### 2.1 极限学习机

ELM是由Huang<sup>[12]</sup>提出的一种非迭代学习的神经网络算法，其网络结构如图1所示。

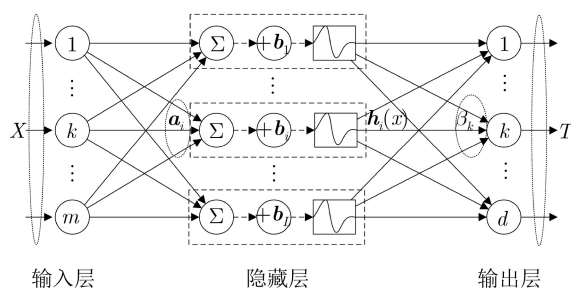


图1 ELM网络结构示意图

对于具有 $L$ 个隐藏层节点的ELM，其隐藏层输出函数可以表示为

$$f_L(\mathbf{x}) = \sum_{i=1}^L G_i(\mathbf{x}, \mathbf{a}_i, \mathbf{b}_i) \cdot \beta_i, \mathbf{a}_i \in R^d, \mathbf{b}_i, \beta_i \in R \quad (1)$$

其中 $G_i(\cdot)$ 为第 $i$ 个隐藏层节点的激活函数， $\mathbf{a}_i$ 为输入层和第 $i$ 个隐藏层的输入权重， $\mathbf{b}_i$ 为随机产生的第 $i$ 个隐藏层偏置， $\beta_i$ 为输出权重。

定义隐藏层关于 $\mathbf{x}$ 的输出向量为 $\mathbf{H} = [\mathbf{h}_1(\mathbf{x}), \mathbf{h}_2(\mathbf{x}), \dots, \mathbf{h}_L(\mathbf{x})]$ ，隐藏层节点和输出层节点之间的输出权重向量为 $\beta = [\beta_1, \beta_2, \dots, \beta_L]^T$ ，则整个系统的输出可由式(1)改写为

$$f_L(\mathbf{x}) = \sum_{i=1}^L \beta_i \mathbf{h}_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\beta \quad (2)$$

在ELM中，若输入权重和偏置被随机确定，隐藏层的输出矩阵 $\mathbf{H}$ 就被确定，设 $\mathbf{T}$ 为训练数据目标矩阵，则其训练过程可由式(2)转换为式(3)的线性回归方程

$$\mathbf{H}\beta = \mathbf{T} \quad (3)$$

$$\text{其中, } \mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \mathbf{h}(\mathbf{x}_2) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) & h_2(\mathbf{x}_1) & \cdots & h_L(\mathbf{x}_1) \\ h_1(\mathbf{x}_2) & h_2(\mathbf{x}_2) & \cdots & h_L(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(\mathbf{x}_N) & h_2(\mathbf{x}_N) & \cdots & h_L(\mathbf{x}_N) \end{bmatrix}_{N \times L}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1^T \\ \beta_2^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m}, \quad \mathbf{T} = \begin{bmatrix} t_1^T \\ t_2^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m}$$

即求使线性方程

$$\min = \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|_2 \quad (4)$$

成立, 则可得方程的最小二乘解为

$$\boldsymbol{\beta} = \mathbf{H}^+ \mathbf{T} \quad (5)$$

式(4)中 $\|\cdot\|$ 表示最小二乘范数解, 式(5)中 $\mathbf{H}^+$ 表示矩阵 $\mathbf{H}$ 的广义逆矩阵。

正交投影方法可有效地用于ELM: 如果 $\mathbf{H}^T \mathbf{H}$ 是非奇异的,  $\mathbf{H}^+ = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T$ , 如果 $\mathbf{H} \mathbf{H}^T$ 是非奇异的,  $\mathbf{H}^+ = \mathbf{H}^T (\mathbf{H} \mathbf{H}^T)^{-1}$ 。Huang<sup>[12]</sup>提出在求广义逆矩阵时, 将正值 $1/C$ 加到 $\mathbf{H}^T \mathbf{H}$ 或 $\mathbf{H} \mathbf{H}^T$ 的对角线上可以得到更稳定的泛化性能。因此, ELM的输出权重 $\boldsymbol{\beta}$ 为

$$\left. \begin{aligned} \boldsymbol{\beta} &= \left( \frac{1}{C} + \mathbf{H}^T \mathbf{H} \right)^{-1} \mathbf{H}^T \mathbf{T}, N \geq L \\ \boldsymbol{\beta} &= \mathbf{H}^T \left( \frac{1}{C} + \mathbf{H} \mathbf{H}^T \right)^{-1} \mathbf{T}, N < L \end{aligned} \right\} \quad (6)$$

## 2.2 自动编码器

自动编码器(AE)是基于神经网络的降维算法, 它通过最小化重建误差来近似原始输入<sup>[13]</sup>。对于高维数据的处理, ELM-AE将输入的高维数据 $\mathbf{x}$ 通过编码变成低维的, 在完成特征提取后通过解码映射返回到 $\mathbf{x}^*$ 。ELM-AE使用随机映射的输出 $\mathbf{x}^*$ 作为输入 $\mathbf{x}$ 的重构表示, 并且将隐藏层随机产生的权重和偏置正交化, 可以获得更高的抗干扰性能, 在表示输入特征时可分为: 压缩表示、稀疏表示和等维表示3种方法。

自动编码器的隐藏层输出结果为

$$\mathbf{h}_i = g(\mathbf{a}_i \mathbf{x} + \mathbf{b}_i) \quad (7)$$

对特征提取结果进行解码可得

$$\mathbf{x}^* = f'(\mathbf{a}_i \mathbf{h}_i + \mathbf{b}_i) \quad (8)$$

其中 $g(\cdot)$ 和 $f'(\cdot)$ 为激活函数,  $\mathbf{h}_i$ 为隐藏层输出,  $\mathbf{a}_i$ ,  $\mathbf{b}_i$ 为隐藏层对应偏置, 且满足 $\mathbf{a}_i^T \mathbf{a}_i = \mathbf{I}$ ,  $\mathbf{b}_i^T \mathbf{b}_i = \mathbf{I}$ 。为了训练能够对输入特征等效编码的权重矩阵, 让 $\mathbf{x}$ 作为输出特征来训练网络输出权重矩阵 $\boldsymbol{\beta}$ , 对于压缩表示与稀疏表示, ELM-AE输出权重为

$$\left. \begin{aligned} \boldsymbol{\beta} &= \left( \frac{1}{C} + \mathbf{H}^T \mathbf{H} \right)^{-1} \mathbf{H}^T \mathbf{X}, N \geq L \\ \boldsymbol{\beta} &= \mathbf{H}^T \left( \frac{1}{C} + \mathbf{H} \mathbf{H}^T \right)^{-1} \mathbf{X}, N < L \end{aligned} \right\} \quad (9)$$

对于等维表示时, ELM-AE输出权重为

$$\boldsymbol{\beta} = \mathbf{H}^{-1} \mathbf{X} \quad (10)$$

由ELM原理可知, 样本数量越多, 矩阵 $\mathbf{H}$ 的维数也就越高, 如果一次性处理所有数据样本, 矩阵 $\mathbf{H}$ 的维数将会呈指数增加, 严重影响极限学习机的学习速度。为此, 本文将给出一种基于正则优化的批次继承极限学习机算法。

## 3 基于正则优化的批次继承极限学习机算法

### 3.1 B-ELM算法设计

批次学习适用于大样本数据集, 为提高批次处理效率, 本文提出一种批次继承极限学习机(B-ELM)算法。其主要方法是: 将整批数据分割为大小相同的不同批次, 引入继承因子建立关系, 利用推导出的带有继承因子的循环更新公式来训练网络权重, 提高网络的分类能力。

基于ELM的训练过程即为求解隐藏层输出权重矩阵的过程, 其B-ELM训练过程如图2所示。首先将高维数据 $X$ 均分为 $M$ 批次, 把第1批数据载入ELM-AE中得到隐藏层输出矩阵 $\mathbf{H}$ 并计算得到初始输出权重 $\boldsymbol{\omega}$ ; 然后将第2批数据载入到ELM-AE中得到该批数据的隐藏层输出矩阵 $\mathbf{H}_1$ , 通过含有继承因子的范式得到第2层的输出权值 $\beta_1$ , 以此类推直到得到最后一批数据对应的输出权值 $\beta_{M-1}$ ; 最后得到最终的输出权重 $\boldsymbol{\beta}$ 。

批次继承极限学习机算法是利用前批次的系统输出来修正当前批次的系统输入, 从而使相邻批次之间的输出权重差值最小以达到继承学习的目的。其中主要涉及到一个问题, 相邻批次之间的输出权重 $\boldsymbol{\beta}$ 该如何继承融合, 下面给出解决方法。

### 3.2 B-ELM算法输出权重求解

批次处理是典型的多阶段处理。由于批次处理阶段之间的相互作用, 单一的高精度不能保证整批数据处理的高精度。为解决这一问题, 设计了能够继承相邻批次特征学习的正则项, 并以线性等式的形式给出最终的输出权重, 从而使得B-ELM具有较好的迁移性能, 具体的求解过程如下。

由于ELM可以近似任何连续目标函数, 对于第 $i$ 批样本数据为 $\mathbf{x}_i$ , ELM的输出函数可表示为

$$f_L(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i) \boldsymbol{\beta} \quad (11)$$

为建立各批次的输出权重关系, 在正则化模型

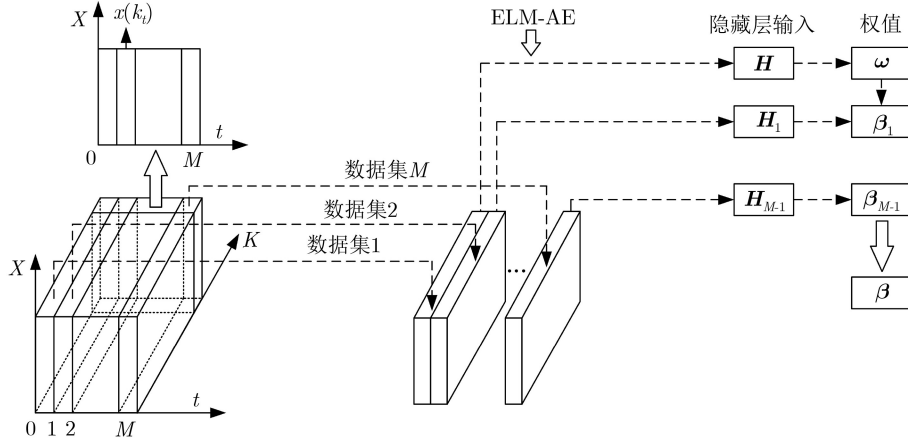


图2 B-ELM训练过程示意图

中引入继承因子 $u$ , 根据结构风险最小化原则<sup>[14]</sup>, 在输出函数中加入权重 $\beta$ 的L2范数, 构造基于约束优化的B-ELM函数范式, 则目标函数转化为求解最小值优化的问题

$$\min L = \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \sum_{i=1}^N \|\xi_i\|^2 + \frac{1}{2} u \|\beta - w\|^2 \quad (12)$$

$$\text{s.t. } \mathbf{h}(\mathbf{x}_i)\beta = \mathbf{t}_i - \xi_i, \quad i = 1, 2, \dots, N$$

其中,  $\xi_i$ 为理论输出 $T$ 相对于实际输出 $f(\mathbf{x})$ 的误差,  $\frac{1}{2} \|\beta\|^2$ 表示结构风险,  $\frac{1}{2} \sum_{i=1}^N \|\xi_i\|^2$ 表示经验风险和正则化最小均方参数, 用以权衡结构风险和正则化最小均方参数之间的比例。继承正则项, 它通过前批次权重来修正当前批次的权重, 旨在使得相邻批次之间的输出权重差异最小,  $u$ 为继承因子, 用以调节各批次之间输出权值的比例,  $w$ 为第1批数据得到的初始输出权重。

式(12)所示的最小值优化问题是一个凸优化问题, 因此可以利用拉格朗日乘子法解决此约束优化问题, 则建立拉格朗日函数为

$$L = \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \sum_{i=1}^N \|\xi_i\|^2 + \frac{1}{2} u \|\beta - w\|^2 - \sum_{i=1}^N \sum_{j=1}^M a_{ij} [\mathbf{h}(\mathbf{x}_i)\beta_j - t_{ij} + \xi_{ij}] \quad (13)$$

其中 $a_{ij}$ 为拉格朗日乘子, 为得到最小值, 根据KKT(Karush-Kuhn-Tucker)最优性条件, 分别对式(13)中变量 $\beta$ ,  $\xi$ ,  $a$ 求偏导, 并令偏微分为0得

$$\frac{\partial L}{\partial \beta_j} = 0 \Rightarrow \beta_j + u(\beta_j - w_j) - \sum_{i=1}^N a_{ij} \mathbf{h}^T(\mathbf{x}_i) = 0 \quad (14)$$

$$\frac{\partial L}{\partial \xi_i} = 0 \Rightarrow \mathbf{a}_i = C\xi_i, \quad i = 1, 2, \dots, N \quad (15)$$

$$\frac{\partial L}{\partial \mathbf{a}_i} = 0 \Rightarrow \mathbf{h}(\mathbf{x}_i)\beta - \mathbf{t}_i^T + \xi_i^T = 0, \quad i = 1, 2, \dots, N \quad (16)$$

由式(14)化简整理可以得到式(17)和式(18)

$$\beta = \frac{1}{1+u} (uw + \mathbf{H}^T \mathbf{a}) \quad (17)$$

$$\mathbf{a} = (\mathbf{H}^T)^+ (\beta + u\beta - uw) \quad (18)$$

为方便对目标函数的求解, 其输出权重 $\beta$ 可分为两种形式,

当 $N \geq L$ 时, 将式(15)和式(18)代入式(16)得到

$$\beta = \left( \mathbf{H}^T \mathbf{H} + \frac{1+u}{C} \right)^{-1} \left( \mathbf{H}^T \mathbf{T} + \frac{uw}{C} \right) \quad (19)$$

当 $N < L$ 时, 将式(15)和式(17)代入式(16)得到

$$\beta = \mathbf{H}^T \left( \mathbf{H} \mathbf{H}^T + \frac{1+u}{C} \right)^{-1} \left( \mathbf{T} - \frac{\mathbf{H}uw}{1+u} \right) + \frac{uw}{1+u} \quad (20)$$

式(19)和式(20)即为B-ELM的权值递推公式, 在训练样本很大的数据集中,  $N \gg L$ , 故本文采用式(19)作为B-ELM的输出, 由式(11)可知, B-ELM的输出表达式为

$$f_L(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i) \left( \mathbf{H}^T \mathbf{H} + \frac{1+u}{C} \right)^{-1} \left( \mathbf{H}^T \mathbf{T} + \frac{uw}{C} \right) \quad (21)$$

### 3.3 B-ELM算法步骤

本文提出的B-ELM算法, 通过分割数据集, 利用ELM-AE逐批训练, 以降低输入数据的矩阵维度, 同时构建含有继承因子 $u$ 的正则项, 使各批次数据最大程度上保留并继承上一批次的的数据特征, 并使误差达到最小, 以保证整批数据信息完整性。同时通过引入正则化框架建立拉格朗日优化函数关系, 求得网络隐藏层输出权重 $\beta$ , 从而完成训练。

总结B-ELM算法的训练步骤分为6步:

步骤 1 设高维数据集为 $X$ , 将样本数据集平均分割为 $M$ 份, 其中第 $i$ 份样本数据集为 $X_i$ ;

步骤 2 随机生成输入层和隐藏层之间的输入权值 $\mathbf{a}_i$ 以及隐藏层的偏置 $\mathbf{b}_i$ ;

步骤 3 计算网络隐藏层的输出矩阵 $\mathbf{H}$ ;

步骤 4 将第1批数据 $X_1$ 导入ELM-AE, 根据式 $\mathbf{w} = \left(\frac{1}{C} + \mathbf{H}^T \mathbf{H}\right)^{-1} \mathbf{H}^T \mathbf{X}$ , 得到初始输出权重 $\mathbf{w}$ ;

步骤 5 将剩余的 $M-1$ 批数据有序导入, 计算隐藏层的输出矩阵 $\mathbf{H}_i$ , 输出矩阵 $\mathbf{T}_i$ , 根据含有继承因子的范式 $\beta_i = \left(\mathbf{H}_i^T \mathbf{H}_i + \frac{1+u}{C}\right)^{-1} \left(\mathbf{H}_i^T \mathbf{T}_i + \frac{u\mathbf{w}}{C}\right)$ , 得到最终隐藏层输出权重矩阵 $\beta$ ;

步骤 6 根据B-ELM的输出 $f_L(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i)\beta$ , 得到输出值并进行精度测试。

### 3.4 算法时间复杂度分析

基于ELM算法的时间复杂度大小, 区别在于输出权重的求解部分。首先分析传统ELM, 设 $d$ 为输入数据维度,  $L$ 为隐藏层神经元个数, 对于给定的 $N$ 个样本, 根据式(6)可知 $\mathbf{H}^T \mathbf{H}$ 为维度 $N \times N$ 的对角阵, 复杂度为 $O(N^2L)$ , 其中 $(\cdot)^{-1}$ 的复杂度最高, 大小为 $O(N^3)$ , 则可得ELM的时间复杂度为 $O(N^3 + N^2L + NdL + 1)$ 。本文算法引入继承因子, 虽增加了相邻批次权重继承的复杂度, 其大小为 $O(NdL)$ , 但批次学习将数据集分割为 $M$ 份进行训练, 使得输入样本 $N$ 的矩阵维度降低为 $N/M$ , 故本文提出的B-ELM算法的时间复杂度为 $O((N/M)^3 + (N/M)^2L + 2(N/M)dL + 1)$ 。由于 $(N/M) \ll N$ , 因此本文算法的时间复杂度远小于传统ELM算法的时间复杂度。

## 4 实验

本文算法的实验平台为: 燕山大学超级计算中心, 其主机处理器为Intel E5-2683v3(28核)@2.0 GHz, 64 GB RAM, 算法操作系统和软件环境分别为Centos7.2, Matlab R2018a。

### 4.1 数据集介绍

MNIST是一种用于测试神经网络的手写体数据集, 它由10类手写数字的 $28 \times 28$ 像素图像及其相应的标签构成, 每个标签是0到9之间的数字。其中60000个为训练数据, 10000个为测试数据。NORB比MNIST数据集更具挑战性, 它由50个不同对象的 $32 \times 32$ 像素灰色图像及其相应的标签(汽车、卡车、飞机、动物、人类)组成, 其训练样本和测试样本都包含24300个3D物体图像对。CIFAR-10是神经网络常用的测试数据集, 它由10类分类对象

的 $32 \times 32 \times 3$ 像素RGB图像组成, 包含60000张RGB图像, 其中50000张用于训练, 10000张用于测试。图3为3种数据集的部分图像示例。

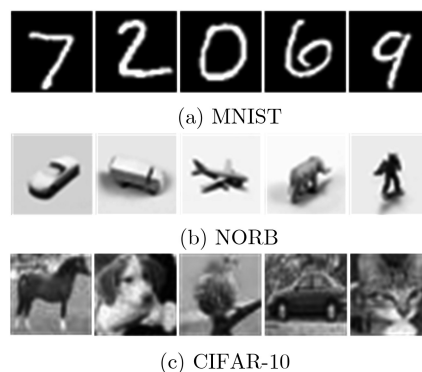


图3 数据集图像示例

### 4.2 实验结果与分析

为全面评价B-ELM算法的性能, 本文采用MNIST, NORB和CIFAR-10标准数据集对算法进行测试。B-ELM算法通过对数据集划分批次进行训练, 对于MNIST数据集, 设定 $M=3$ ; 对于NORB数据集, 设定 $M=2$ ; 对于CIFAR-10数据集, 设定 $M=2$ , 在实验中, 激活函数选择Sigmoid函数。

**实验1** 为证明B-ELM算法的有效性, 本实验以MNIST数据集为例, 比较B-ELM和ELM超参数的选取对算法性能的影响, 主要包括隐藏层节点数 $L$ 和正则化最小均方参数 $C$ , 并用不同的超参数绘制了测试精度曲线。

图4和图5分别为不同超参数对算法测试精度影响的变化曲线。由图4(a)可以看出, B-ELM的精度随着节点数 $L$ 增大而增大, 并且 $L$ 在1600-2000的范围内精度趋于稳定, 相比于图4(b), B-ELM具有更高的收敛精度, 当 $L=2000$ ,  $C=10^{-8}$ 时, B-ELM的精度为98.35%, ELM的精度为94.62%, B-ELM的精度比ELM高了3.73%, 且在节点数 $L$ 较少时, B-ELM的精度同样比ELM高。由图5(a)可以看出, 随着 $1/C$ 的增加, B-ELM的精度也随之升高, 在一定范围内, B-ELM的测试精度曲线光滑且趋于稳定, 由图5(b)可以看出, 随着 $1/C$ 的增加, ELM的精度变化整体上呈上升趋势, 但曲线有所波动。综上所述, B-ELM的精度曲线比ELM更为平滑, 两者在相同条件下的精度曲线趋势非常相似, 但B-ELM的收敛精度更高且具有更好的鲁棒性。

**实验2** 为进一步验证本文算法的有效性, 采用如下5种对比算法: SAE<sup>[15]</sup>(Stacked Auto Encoders), SDA<sup>[16]</sup>(Stacked Denoise Autoencoder), DBM<sup>[17]</sup>(Deep Boltzmann Machines), ML-ELM<sup>[18]</sup>

(Multi-Layer ELM)和H-ELM<sup>[19]</sup>(Hierarchical ELM)。其中SAE, SDA和DBM是非ELM的深度神经网络, ML-ELM, H-ELM是基于ELM的深度神经网络, 分别在MNIST, NORB和CIFAR-10数据集上对分类精度和训练时间的最佳效果进行统计, 如表1所示。

由表1可以发现, 在分类精度上, B-ELM在3种数据集上均取得了最高的分类准确率, 在MINST数据集上, 本文算法与最先进的H-ELM相比, 提高了0.31%, 在NORB数据集上, 比H-ELM高了0.62%, 在CIFAR-10数据集上, 比H-ELM高了0.17%。在训练时间上, 本文算法训练时间比H-ELM稍长, 但由于其中ELM-AE具有降低数据搜索空间的作用, 训练时间仍然非常快, 而且比其他非ELM训练方法要快得多。

同时, 通过表1对B-ELM的性能分析如下: (1) B-ELM分类精度高于SAE, SDA和DBM, 训练时间快了数百倍, 这是由于B-ELM具有ELM的一致逼近性, 所提取的特征更加有效, 不易陷入局部最优, 且结构简单, 参数无需迭代微调, 训练耗时更短; (2) B-ELM分类精度高于ML-ELM, 这是因为B-ELM的自动编码器结构具有更强的高维数据处理能力, 较ML-ELM提取的特征更加符合样本本质, 更加鲁棒; (3) B-ELM分类精度略高于H-ELM, 这是因为B-ELM融合自动编码结构与批次学习, 通过批次编码降低了数据搜索空间, 使训练获得了更有效的样本特征。

**实验3** 为验证B-ELM算法具有降低运行内存的优势, 比较B-ELM与ELM的分类精度与内存消

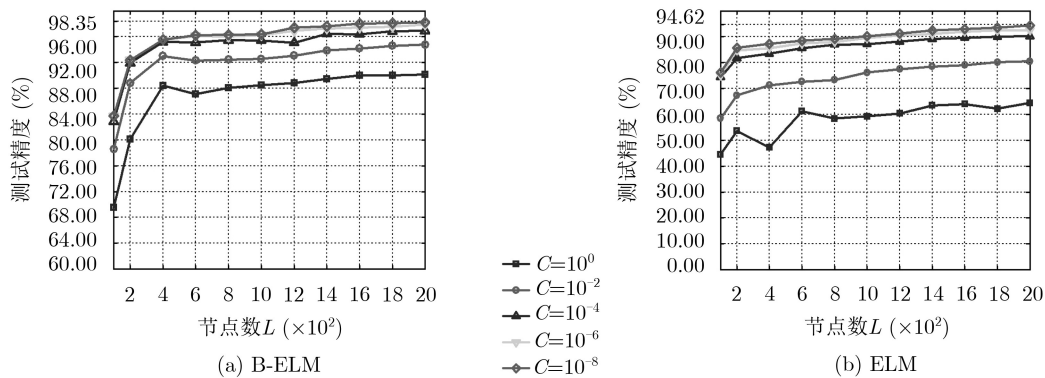


图4 节点数L对测试精度的影响

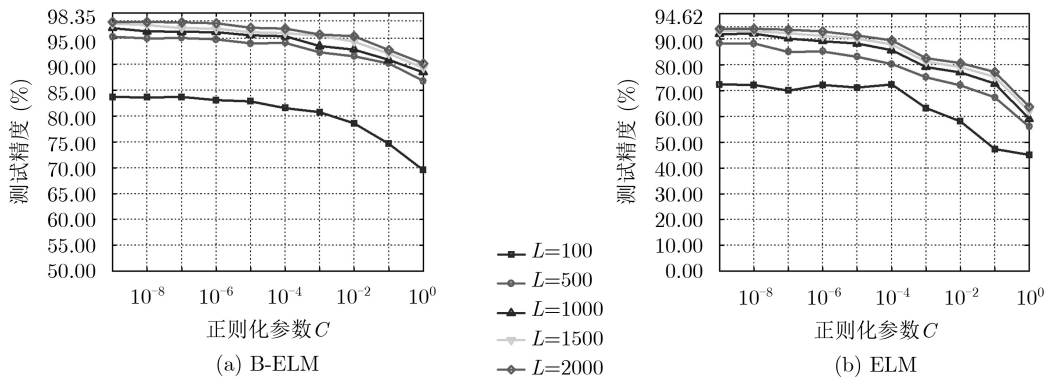


图5 正则化参数C对测试精度的影响

表1 不同数据集上的性能比较

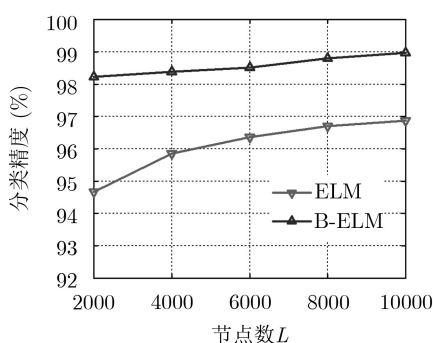
分类方法	MNIST		NORB		CIFAR-10	
	精度(%)	训练时间(s)	精度(%)	训练时间(s)	精度(%)	训练时间(s)
SAE	98.60	4042.36	86.28	6438.56	43.37	60514.26
SDA	98.72	3892.26	87.62	6572.14	43.61	87289.59
DBM	99.05	14505.14	89.65	18496.64	43.12	90123.53
ML-ELM	98.21	51.83	88.91	78.36	45.42	74.06
H-ELM	99.12	28.97	91.28	42.74	50.21	62.76
B-ELM	99.43	42.67	91.90	55.96	50.38	69.06

耗大小，其中ELM和B-ELM的正则化最小均方参数 $C$ 分别设置为 $C=10^{-8}$ 。为定量评价算法的峰值内存消耗量，每组实验基于MATLAB程序重复20次，选取均值和标准差(Standard Deviation, SD)两种数学统计量作为评判依据，均值和标准差越小则说明算法运行越稳健，反之，亦然。在实验过程中，使用超算中心的作业系统，通过设定不同的节点统计算法分类精度和平均峰值内存消耗量，如图6-图8所示。

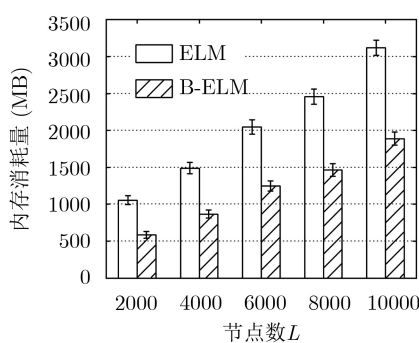
图6-图8分别为3个数据集上ELM与B-ELM的分类精度曲线和平均峰值内存消耗量。由3个图中

的(a)图可以看出，当节点 $L$ 相同时，B-ELM的精度远高于ELM。由3个图中的(b)图可以看出，在隐藏层节点个数 $L$ 相同的情况下，采用B-ELM普遍要比ELM的内存消耗量低，且标准差更小。由图6-图8可以看出，B-ELM不仅精度高，而且内存消耗量更小，可以用更少的节点达到较高的精度。相比于ELM，本文提出的算法可以通过分批训练减少内存消耗，降低内存需求，从而缓解传统ELM训练带来的维数灾难问题。因此，本文算法在降低内存需求方面具有一定的优势。

综合以上3个实验可以看出，本文算法在分类

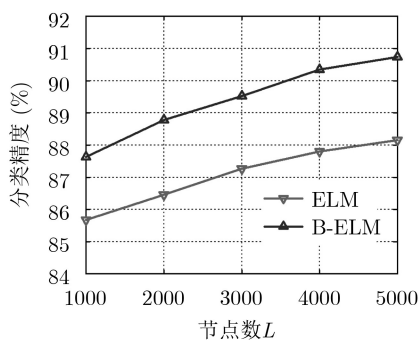


(a) MNIST数据集精度曲线

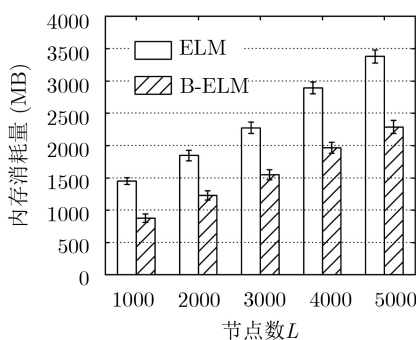


(b) MNIST数据集平均峰值内存消耗

图6 MNIST数据集算法性能比较

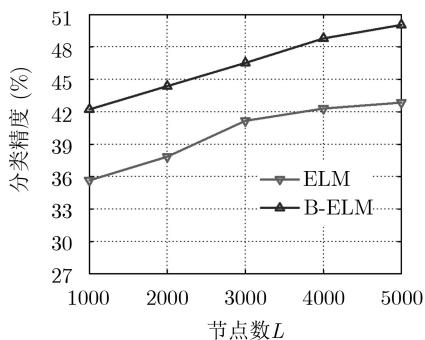


(a) NORB数据集精度曲线

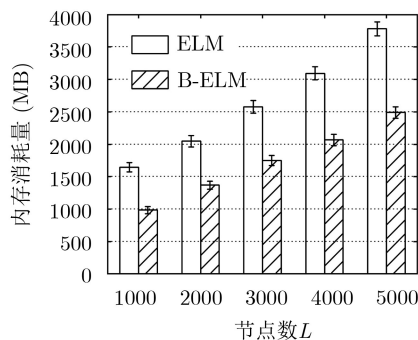


(b) NORB数据集平均峰值内存消耗

图7 NORB数据集算法性能比较



(a) CIFAR-10数据集精度曲线



(b) CIFAR-10数据集平均峰值内存消耗

图8 CIFAR-10数据集算法性能比较

精度和内存消耗上明显优于其他算法。因此, 将ELM与批次学习相结合, 构造B-ELM用来完成分类的机制是可行的。

## 5 结束语

与利用整批数据进行训练的思路不同, 本文从分批训练的角度出发, 提出了一种高效的批次继承极限学习机算法。该算法首先通过对数据分批处理, 并利用ELM-AE逐批训练, 大大降低了输入数据的矩阵维度。其次, 为将各批次输出权重进行有效地融合, 引入继承因子设计了具有扩展信息, 能够继承相邻批次特征学习的正则项, 保证了整批数据信息的完整性。最后, 根据正则化框架中求解约束优化的问题, 构建目标函数, 得到了一种新的求解输出权值的方法, 增强了算法的泛化性能。理论分析和实验结果表明, 本文算法B-ELM相较于原始ELM具有计算量小, 内存消耗低的特性, 同时批次学习也为ELM处理高维数据提供了一种新的思路。

## 参 考 文 献

- [1] HUANG Guangbin, ZHU Qinyu, and SIEW C K. Extreme learning machine: Theory and applications[J]. *Neurocomputing*, 2006, 70(1/3): 489–501. doi: [10.1016/j.neucom.2005.12.126](https://doi.org/10.1016/j.neucom.2005.12.126).
- [2] 李佩佳, 石勇, 汪华东, 等. 基于有序编码的核极限学习顺序回归模型[J]. *电子与信息学报*, 2018, 40(6): 1287–1293. doi: [10.11999/JEIT170765](https://doi.org/10.11999/JEIT170765).  
LI Peijia, SHI Yong, WANG Huadong, *et al.* Ordered code-based kernel extreme learning machine for ordinal regression[J]. *Journal of Electronics & Information Technology*, 2018, 40(6): 1287–1293. doi: [10.11999/JEIT170765](https://doi.org/10.11999/JEIT170765).
- [3] HUANG Guangbin, ZHOU Hongming, DING Xiaojian, *et al.* Extreme learning machine for regression and multiclass classification[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2012, 42(2): 513–529. doi: [10.1109/tsmcb.2011.2168604](https://doi.org/10.1109/tsmcb.2011.2168604).
- [4] WANG Yongchang and ZHU Ligu. Research and implementation of SVD in machine learning[C]. The 2017 16th IEEE/ACIS International Conference on Computer and Information Science, Wuhan, China, 2017: 471–475. doi: [10.1109/ICIS.2017.7960038](https://doi.org/10.1109/ICIS.2017.7960038).
- [5] CASTAÑO A, FERNÁNDEZ-NAVARRO F, and HERVÁS-MARTÍNEZ C. PCA-ELM: A robust and pruned extreme learning machine approach based on principal component analysis[J]. *Neural Processing Letters*, 2013, 37(3): 377–392. doi: [10.1007/s11063-012-9253-x](https://doi.org/10.1007/s11063-012-9253-x).
- [6] ZONG Weiwei, HUANG Guangbin, and CHEN Yiqiang. Weighted extreme learning machine for imbalance learning[J]. *Neurocomputing*, 2013, 101: 229–242. doi: [10.1016/j.neucom.2012.08.010](https://doi.org/10.1016/j.neucom.2012.08.010).
- [7] ZHAO Rui and MAO Kezhi. Semi-random projection for dimensionality reduction and extreme learning machine in high-dimensional space[J]. *IEEE Computational Intelligence Magazine*, 2015, 10(3): 30–41. doi: [10.1109/MCI.2015.2437316](https://doi.org/10.1109/MCI.2015.2437316).
- [8] LUO Xiong, XU Yang, WANG Weiping, *et al.* Towards enhancing stacked extreme learning machine with sparse autoencoder by correntropy[J]. *Journal of the Franklin Institute*, 2018, 355(4): 1945–1966. doi: [10.1016/j.jfranklin.2017.08.014](https://doi.org/10.1016/j.jfranklin.2017.08.014).
- [9] WU Shuang, LI Guoqi, DENG Lei, *et al.* L1-norm batch normalization for efficient training of deep neural networks[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2019, 30(7): 2043–2051. doi: [10.1109/TNNLS.2018.2876179](https://doi.org/10.1109/TNNLS.2018.2876179).
- [10] LI Yanghao, WANG Naiyan, SHI Jianping, *et al.* Adaptive batch normalization for practical domain adaptation[J]. *Pattern Recognition*, 2018, 80: 109–117. doi: [10.1016/j.patcog.2018.03.005](https://doi.org/10.1016/j.patcog.2018.03.005).
- [11] LIANG Nanying, HUANG Guangbin, SARATCHANDRAN P, *et al.* A fast and accurate online sequential learning algorithm for feedforward networks[J]. *IEEE Transactions on Neural Networks*, 2006, 17(6): 1411–1423. doi: [10.1109/TNN.2006.880583](https://doi.org/10.1109/TNN.2006.880583).
- [12] HUANG Guangbin. What are extreme learning machines? Filling the gap between frank Rosenblatt’s dream and john von Neumann’s puzzle[J]. *Cognitive Computation*, 2015, 7(3): 263–278. doi: [10.1007/s12559-015-9333-0](https://doi.org/10.1007/s12559-015-9333-0).
- [13] YI Yugen, QIAO Shaojie, ZHOU Wei, *et al.* Adaptive multiple graph regularized semi-supervised extreme learning machine[J]. *Soft Computing*, 2018, 22(11): 3545–3562. doi: [10.1007/s00500-018-3109-x](https://doi.org/10.1007/s00500-018-3109-x).
- [14] CHENG Kai and LU Zhenzhou. Adaptive sparse polynomial chaos expansions for global sensitivity analysis based on support vector regression[J]. *Computers & Structures*, 2018, 194: 86–96. doi: [10.1016/j.compstruc.2017.09.002](https://doi.org/10.1016/j.compstruc.2017.09.002).
- [15] HINTON G E and SALAKHUTDINOV R R. Reducing the dimensionality of data with neural networks[J]. *Science*, 2006, 313(5786): 504–507. doi: [10.1126/science.1127647](https://doi.org/10.1126/science.1127647).
- [16] VINCENT P, LAROCHELLE H, BENGIO Y, *et al.* Extracting and composing robust features with denoising autoencoders[C]. The 25th International Conference on Machine Learning, Helsinki, Finland, 2008: 1096–1103. doi: [10.1145/1390156.1390294](https://doi.org/10.1145/1390156.1390294).
- [17] SALAKHUTDINOV R and HINTON G. An efficient

- learning procedure for deep Boltzmann machines[J]. *Neural Computation*, 2012, 24(8): 1967–2006. doi: [10.1162/NECO\\_a\\_00311](https://doi.org/10.1162/NECO_a_00311).
- [18] CAMBRIA E, HUANG Guangbin, KASUN L L C, *et al.* Extreme learning machines[trends & controversies][J]. *IEEE Intelligent Systems*, 2013, 28(6): 30–59. doi: [10.1109/MIS.2013.140](https://doi.org/10.1109/MIS.2013.140).
- [19] TANG Jiexiong, DENG Chenwei, and HUANG Guangbin. Extreme learning machine for multilayer perceptron[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2016, 27(4): 809–821. doi: [10.1109/TNNLS.2015.2424995](https://doi.org/10.1109/TNNLS.2015.2424995).
- 刘 彬: 男, 1953年生, 教授, 博士生导师, 研究方向为数据挖掘、信号估计与识别算法.
- 杨有恒: 男, 1996年生, 硕士生, 研究方向为数据挖掘、机器学习.
- 赵志彪: 男, 1989年生, 博士生, 研究方向为人工智能优化算法.
- 吴 超: 男, 1990年生, 博士生, 研究方向为计算机视觉.
- 刘浩然: 男, 1980年生, 教授, 博士生导师, 研究方向为无线传感器网络、信号处理.
- 闻 岩: 男, 1963年生, 教授, 博士生导师, 研究方向为数据挖掘、人工智能优化算法.

责任编辑: 阮 望