

集成 Dropout 极限学习机数据分类方法

翟俊海^{1,2}, 臧立光³, 周昭一¹

(1.河北省机器学习与计算智能重点实验室,河北大学数学与信息科学学院,河北 保定 071002)

(2.浙江师范大学数理与信息工程学院,浙江 金华 321004)

(3.河北大学计算机科学与技术学院,河北 保定 071002)

[摘要] 极限学习机(Extreme Learning Machine, ELM)是一种速度快,泛化能力强的训练单隐层前馈神经网络(Single-hidden Layer Feed-forward Neural-network, SLFN)的算法.但是在应用 ELM 解决实际问题时,需要先确定合适的 SLFN 结构.然而,对于给定的问题,确定合适的 SLFN 结构是非常困难的.针对这一问题,本文提出了一种集成学习方法.用该方法解决问题时,不需要事先确定 SLFN 的结构.提出的方法包括 3 步:(1)初始化一个比较大的 SLFN;(2)用 ELM 重复训练若干个 Dropout 掉若干个隐含层结点的 SLFNs;(3)用多数投票法集成训练好的 SLFNs,并对测试样例进行分类.在 10 个数据集上进行了实验,比较了本文提出的方法和传统的极限学习机方法.实验结果表明,本文提出的方法在分类性能上优于传统的极限学习机算法.

[关键词] 极限学习机,随机化方法,重复训练,泛化能力,集成

[中图分类号] TP181 **[文献标志码]** A **[文章编号]** 1001-4616(2017)03-0059-08

An Approach of Integrating Dropout Extreme Learning Machine for Data Classification

Zhai Junhai^{1,2}, Zang Liguang³, Zhou Zhaoyi¹

(1.Key Lab of Machine Learning and Computational Intelligence, College of Mathematics and Information Science, Hebei University, Baoding 071002, China)

(2.College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China)

(3.College of Computer Science and Technology, Hebei University, Baoding 071002, China)

Abstract: Extreme learning machine (ELM) is an algorithm for training single-hidden layer feed-forward neural-network (SLFN) with fast speed and good generalization. It is indispensable to firstly select an appropriate architecture of SLFN when applying ELM to solve practical problems. However, it is very difficult to select an appropriate architecture of SLFN. In order to deal with this problem, an ensemble learning method is proposed in this paper. It is unnecessary to select the appropriate architecture of SLFN when using the proposed method to solve the problems. The proposed method includes 3 steps: (1) initialize a big SLFN; (2) retrain some SLFNs with dropout hidden nodes with ELM; (3) the trained SLFNs are integrated by majority voting method, and the integrated model is used to classify testing instances. We experimentally compared the proposed approach with traditional ELM on 10 data sets, and the experimental results confirm that the proposed approach outperforms the traditional ELM on performance of classification.

Key words: extreme learning machine, randomization methods, retrain, generalization, ensemble

极限学习机(Extreme Learning Machine, ELM)是 Huang 等人^[1]于 2004 年提出的一种训练单隐层前馈神经网络的算法,它用随机化方法初始化输入层连接权和隐含层偏置,用分析的方法确定输出层连接权.因为 ELM 不需要迭代调整权参数,所以具有学习速度非常快的优点.另外, Huang 等人^[2]从理论上证明了 ELM 具有很强的泛化能力. ELM 已经成为近几年机器学习领域的研究热点,已成功应用于模式识别^[3-4]、预测预报^[5-6]、分类与回归等领域^[7-8].

收稿日期:2017-03-18.

基金项目:国家自然科学基金项目(71371063)、河北省自然科学基金项目(F2017201026)、浙江省计算机科学与技术重中之重学科(浙江师范大学)基金项目.

通讯联系人:翟俊海,博士,教授,研究方向:数据挖掘与模式识别. E-mail:mczjh@126.com

本文研究分类问题,研究人员提出了许多基于传统极限学习机的分类算法^[7,9-12].用极限学习机解决分类问题时,需要事先确定单隐含层前馈神经网络的结构,即确定合适的隐含层结点个数.但对于给定的实际问题,确定合适的网络结构是非常困难的.研究人员开展了不少这方面的研究工作,提出了许多网络结构选择算法.这些算法大致可分为两类:增量方法和剪枝方法.

增量方法从一个小的网络开始,逐渐增加隐含层结点个数,直到满足预定义的停止条件为止.在 ELM 框架内,第一个选择网络结构的增量方法是 Huang 等人提出的 I-ELM (Incremental-ELM)^[13-14],I-ELM 采用一个接一个地增加隐含层结点的方式,但保持原有的输出层权值不变,停止条件是预定义的最大隐含层结点数.在 I-ELM 的基础上,Feng 等人提出了误差最小化极限学习机 EM-ELM (Error Minimized-ELM),EM-ELM 可以采用一个接一个,或一组接一组地增加隐含层结点的方式,不同于 I-ELM 的是,EM-ELM 递归地更新所有输出层权值,停止条件是误差达到预定义的一个值.增量方法的缺点是难于自动获得最优的网络结构,研究的相对较少.

剪枝方法是从一个初始化的大的网络开始,利用某种启发式将不重要的结点剪掉.在 ELM 框架中,第一个网络结构选择的剪枝方法是 Rong 等人提出的剪枝极限学习机 P-ELM (Pruned-ELM)^[15].P-ELM 用信息论判据(信息增益和卡方统计量)度量隐含层结点的重要性,用 AIC 统计量^[16]作为终止条件.用 P-ELM 处理连续值数据集,需要离散化连续值属性,但是离散化不可避免地会造成信息丢失.针对 P-ELM 的这一不足,Zhai 等人将粗糙集中相容依赖度的概念引入到 ELM 网络结构选择中,提出了改进的 P-ELM 剪枝算法^[17].另外,Zhai 等人还提出了基于结点敏感度的网络结构选择方法^[18].基于最小角回归,Miche 等人提出了最优剪枝极限学习机 OP-ELM (Optimally Pruned ELM)^[19].

无论是增量方法还是剪枝方法,它们共同的不足是计算复杂度都比较高.增量方法计算复杂度高的原因是迭代更新输出层权值,每增加一个(或一组)隐含层结点,都要重复计算输出层权值.剪枝方法计算剪枝高主要是由于计算度量隐含层结点的启发式,如计算隐含层结点的信息增益、卡方统计量、敏感度等.

具有大量参数的深度网络具有非常强的学习能力,但是深度网络很容易产生过拟合,而且大而复杂的网络训练速度很慢.Dropout 是 Hinton 等人于 2012 年提出的解决这一问题的新技术^[20],其核心概念是在训练网络的过程中,从神经网络中随机删除一些隐含层结点及其连接,其动机就是为了防止由于网络隐含层结点之间的相互适应而产生的过拟合.研究结果已经证明,在计算机视觉、语音识别、文本分类等许多监督学习任务中,Dropout 都能够改进神经网络的学习性能^[21]. Srivastava 等人的研究还发现,Dropout 技术不仅适用于前馈神经网络,也适用于波尔兹曼机和受限波尔兹曼机等图模型^[21].近几年,Dropout 技术在深度学习领域应用非常广泛,受到了研究人员的广泛关注. Baldi 和 Sadowski 利用 Bernoulli 变量分析了 Dropout 的各种性质,包括浅层网络和深度网络中 Dropout 的收敛性质、优化性质及逼近性质等^[22],具有很高的参考价值. Wager 等人从正则化的角度,研究了基于 Dropout 的神经网络训练机制^[23]. Iosifidis 等人也是从正则化的角度,研究了基于 Dropout 的极限学习机模型^[24]. Ba 和 Frey 研究了深度神经网络训练中的自适应 Dropout^[25]. Li 等人针对浅层和深层两种神经网络的训练,提出了基于多项式分布的 Dropout 训练方法^[26]. Yang 等人将 Dropout 的思想应用于训练样本,提出了一种针对大数据集的深度卷积神经网络的训练方法,并应用于中文字符识别^[27]. Klein 等人对如何理解 Dropout,提出了一种通用的框架^[28].

本文将 Dropout 技术引入到极限学习机中,提出了一种集成 Dropout 极限学习机数据分类方法,该方法包括 3 步:(1)初始化一个大的 SLFN;(2)用 ELM 重复训练若干个 Dropout 若干个隐含层结点的 SLFNs;(3)用多数投票法集成训练好的 SLFNs,并对测试样例进行分类.在 10 个数据集上进行了实验,比较了本文提出的方法和传统的极限学习机方法.实验结果表明,本文提出的方法在分类性能上优于传统的极限学习机算法.

1 ELM 和 Dropout

1.1 ELM

ELM 是训练如图 1 所示的 SLFNs 的算法.在图 1 所示的 SLFN 中,输入层结点的个数为 d , d 为描述样例的特征或属性个数;隐含层结点个数为 m , m 为特征空间的维数;输出层结点个数为 k , k 为样例的类别个数.SLFN 的这种网络结构可用 1 个三元组 (d, m, k) 表示.

给定训练集 $D = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \mathbf{x}_i \in R^d, \mathbf{y}_i \in R^k, 1 \leq i \leq n\}$, n 是数据集 D 中样例个数. 结构为 (d, m, k) 的 SLFN 可用式(1)表示.

$$f(\mathbf{x}_i) = \sum_{j=1}^m \beta_j g(\mathbf{w}_j \cdot \mathbf{x}_i + \mathbf{b}_j), i=1, 2, \dots, n. \quad (1)$$

式中, $\mathbf{w}_j = (w_{j1}, w_{j2}, \dots, w_{jd})^T$ 是连接输入层结点和隐含层第 j 个结点的权向量, \mathbf{b}_j 是隐含层第 j 个结点的偏置. 在 ELM 中, \mathbf{w}_j 和 \mathbf{b}_j ($j=1, 2, \dots, m$) 是用随机化的方法产生的. $\beta_j = (\beta_{j1}, \beta_{j2}, \dots, \beta_{jk})^T$ 是连接隐含层第 j 个结点和输出层结点的权向量, β_j ($j=1, 2, \dots, m$) 是用分析的方法确定的. 具体地, 可以用给定的训练集 D 通过最小二乘拟合来估计, 即 β_j 满足下面的式(2).

$$f(\mathbf{x}_i) = \sum_{j=1}^m \beta_j g(\mathbf{w}_j \cdot \mathbf{x}_i + \mathbf{b}_j) = y_i, \quad i=1, 2, \dots, n. \quad (2)$$

式(2)可写成下面的矩阵形式.

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

式中,

$$\mathbf{H} = \begin{pmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + \mathbf{b}_1) & \cdots & g(\mathbf{w}_m \cdot \mathbf{x}_1 + \mathbf{b}_m) \\ \vdots & \vdots & \vdots \\ g(\mathbf{w}_1 \cdot \mathbf{x}_n + \mathbf{b}_1) & \cdots & g(\mathbf{w}_m \cdot \mathbf{x}_n + \mathbf{b}_m) \end{pmatrix}, \quad (4)$$

$$\beta = (\beta_1^T, \beta_2^T, \dots, \beta_m^T)^T, \quad (5)$$

$$\mathbf{Y} = (y_1^T, y_2^T, \dots, y_n^T)^T, \quad (6)$$

\mathbf{H} 是 SLFN 隐含层输出矩阵, 它的第 j 列是相对于输入样例 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ 第 j 个隐含层结点的输出向量, 它的第 i 行是相对于输入样例 \mathbf{x}_i 的隐含层的输出向量. 如果隐含层结点数 m 等于训练集中包含的不同样例个数 n , 那么矩阵 \mathbf{H} 是一个可逆方阵. 在这种情况下, SLFN 可以以零误差逼近这 n 个样例. 但一般情况下, 隐含层结点数 m 远远小于训练集中包括的样例个数 n . 因此, 矩阵 \mathbf{H} 不是方阵, 方程(3)一般很难得到精确解. 但是, 方程(3)可以用式(7)的解来逼近.

$$\min_{\beta} \|\mathbf{H}\beta - \mathbf{Y}\|. \quad (7)$$

方程(7)的最小范数最小二乘解可表示为式(8).

$$\hat{\beta} = \mathbf{H}^+ \mathbf{Y}, \quad (8)$$

式中, \mathbf{H}^+ 是矩阵 \mathbf{H} 的 Moore-Penrose 伪逆矩阵. ELM 算法^[1]可用下面的算法伪代码描述:

算法 1: 极限学习机算法

- 1 输入: 训练集 $D = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \mathbf{x}_i \in R^d, \mathbf{y}_i \in R^k, 1 \leq i \leq n\}$; 激活函数 g ; 隐含层结点数 m .
- 2 输出: 权值矩阵 $\hat{\beta}$.
- 3 for ($j=1; j \leq m; j=j+1$) do
- 4 随机给定输入权值 \mathbf{w}_j 和偏置 \mathbf{b}_j ;
- 5 end
- 6 计算隐含层输出矩阵 \mathbf{H} ;
- 7 计算矩阵 \mathbf{H} 的广义逆矩阵 \mathbf{H}^+ ;
- 8 计算权矩阵 $\hat{\beta} = \mathbf{H}^+ \mathbf{Y}$
- 9 输出权矩阵 $\hat{\beta}$.

1.2 Dropout

本节通过简单的线性元, 简要介绍 Dropout 的基本思想^[21-22, 28]. 关于 Dropout 更详细的内容, 感兴趣的读者可参考文献[21-22].

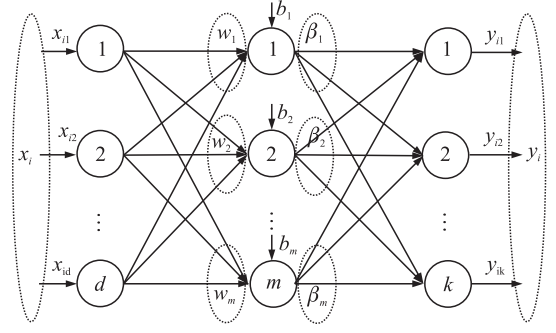


图 1 单隐含层前馈神经网络

Fig. 1 Single-hidden layer feed-forward neural networks

给定输入向量 $\mathbf{I}=(I_1,I_2,\cdots,I_n)$, 线性神经元的输出为 $S(\mathbf{I})=\sum_{i=1}^n w_i I_i$. 如果我们以均匀分布删除 $I_i(1\leq i\leq n)$, 或以等概率 0.5 删除 $I_i(1\leq i\leq n)$, 这样共有 2^n 种可能, 每一种对应一个子网络(包括空网络). 所有子网络的输出平均值为:

$$E(S)=\frac{1}{2^n}\sum_{\mathcal{N}}S(\mathcal{N},\mathbf{I}),$$

式中, \mathcal{N} 表示所有的子网络.

设 $\delta_i(1\leq i\leq n)$ 是服从 Bernoulli 分布且相互独立的随机变量, $p_i=P(\delta_i=1)$, $q_i=P(\delta_i=0)=1-p_i$. 如果以概率 q_i 删除 $I_i(1\leq i\leq n)$ 或连接权 $w_i(1\leq i\leq n)$, 则线性元的输出可表示为:

$$S(\mathbf{I})=\sum_{i=1}^n w_i \delta_i I_i.$$

根据数学期望的线性性质, 可得:

$$E(S)=\sum_{i=1}^n w_i E(\delta_i) I_i=\sum_{i=1}^n w_i p_i I_i.$$

如果 $p_1=p_2=\cdots=p_n=p$, 那么上面的公式变为:

$$E(S)=\sum_{i=1}^n w_i E(\delta) I_i=\sum_{i=1}^n w_i p I_i.$$

如果线性元有确定的偏置 b , 则线性元的输出可表示为:

$$S(\mathbf{I})=\sum_{i=1}^n w_i \delta_i I_i+b \delta_b.$$

类似地, 有:

$$E(S)=\sum_{i=1}^n w_i p_i I_i+b p_b,$$

式中, $p_b=P(\delta_b=1)$.

2 集成 Dropout 极限学习机数据分类方法

本节介绍提出的集成 Dropout 极限学习机数据分类方法. 该方法将 Dropout 技术引入到极限学习机中, 对于给定的实际问题, 可以避免选择合适的网络结构环节, 也能提高数据分类的精度. 该方法首先初始化一个比较大的单隐层前馈神经网络(SLFN), 然后重复训练若干个 Dropout 掉若干个隐含层结点的 SLFNs, 这些训练好的 SLFNs 用作集成的基分类器, 最后用简单的多数投票法集成这些基分类器 SLFNs, 并用于数据分类. 算法的基本思想可用图 2 表示, 算法的伪代码如算法 2 所示.

在分类器集成中, 基分类器的多样性和互补性对集成分类系统的性能具有较大的影响. 多样性是指基分类器的误差多样性, 即被基分类器错误分类的样例应该是不同的样例. 如果被每一个基分类器错误分类的样例都是不同的, 那么基分类器的多样性最好. 显然, 这样的基分类器互补性也最好. 换句话说, 它们在对样例进行分类时, 能够取长补短. 构造基分类器的经典方法是将训练集划分成若干数据子集, 然后在这些数据子集上用某种训练算法训练基分类器. 因为基分类器是用不同的数据子集训练的, 所以人们认为这样训练出来的基分类器具有较好的多样性. 本文构造基分类器的方法与经典方法不同, 我们是用 Dropout 技术构造基分类器, 这些基分类器是用相同的数据集训练的, 但是基分类器的结构不同. 换句话说, 刻画基分类器的参数是不同的, 自然也有理由相信它们具有较好的多样性, 在对样例进行分类时, 能够较好地取长补短.

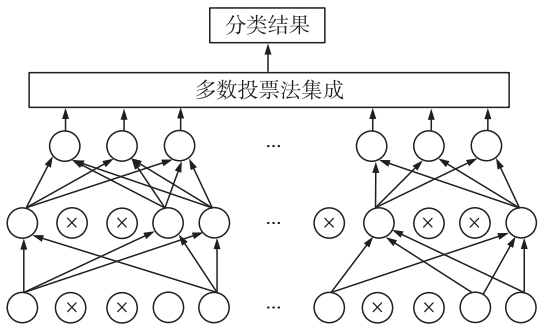


图 2 提出的算法基本思想示意图

Fig. 2 The diagram of basic idea of the proposed algorithm

算法 2 集成 Dropout 极限学习机数据分类算法

```

1 输入:训练集  $D = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \mathbf{x}_i \in R^d, \mathbf{y}_i \in R^k, 1 \leq i \leq n\}$ ; 重复训练次数  $p$ ; 测试样例  $\mathbf{x}$ .
2 输出:测试样例  $\mathbf{x}$  的类标  $y(\mathbf{x})$ .
3 初始化一个比较大的 SLFN, 假设它包含  $m$  个隐含层结点;
4 for( $i = 1; i \leq p; i = i + 1$ ) do
5   按均匀分布生成一个  $m$  维 0-1 向量;
6   按生成的  $m$  维 0-1 向量, 将对应分量为 0 的隐含层结点 dropout 掉, 得到一个  $\text{SLFN}_i$ ;
7   运用极限学习机算法训练  $\text{SLFN}_i$ . 设计  $\text{SLFN}_i$  的输出为  $k$ -维 0-1 二元组  $(y_{i1}(\mathbf{x}), y_{i2}(\mathbf{x}), \dots, y_{ik}(\mathbf{x}))^T \in \{0, 1\}^k$ . 其中,  $y_{ij}(\mathbf{x}) = 1$ , 如果  $\text{SLFN}_i$  分类样例  $\mathbf{x}$  为第  $j$  类, 否则,  $y_{ij}(\mathbf{x}) = 0$ .  $1 \leq j \leq k$ ;
8 end
9 for( $i = 1; i \leq p; i = i + 1$ ) do
10  运用训练好的  $\text{SLFN}_i$ , 计算测试样例  $\mathbf{x}$  的输出, 即  $\mathbf{x}$  的类标向量  $(y_{i1}(\mathbf{x}), y_{i2}(\mathbf{x}), \dots, y_{ik}(\mathbf{x}))^T$ ;
11  运用多数投票法确定测试样例  $\mathbf{x}$  的类别:  $y(\mathbf{x}) = \max_{1 \leq j \leq k} \sum_{i=1}^p y_{ij}(\mathbf{x})$ ;
12 end
13 输出测试样例  $\mathbf{x}$  的类标  $y(\mathbf{x})$ .

```

3 实验结果

为了验证本文方法的有效性,我们在 10 个 UCI 数据集^[29]上进行了实验,与经典的 ELM 在测试精度上进行了比较. 实验所用数据集的基本信息列于表 1 中. 实验环境是 Intel(R) Core(TM) i5-2400 CPU @ 3.10 GHz 处理器, 4 G 内存, 32 位 Windows 操作系统, Matlab R2013a.

表 1 实验所用数据集的基本信息

Table 1 The basic information of data sets used in the experiments

数据集	样例数	属性数	类别数	m 值
Fertility	100	10	2	20
Forest	325	28	4	80
Glass	214	11	7	60
Heart	2 126	22	3	80
Ionosphere	337	34	2	40
Iris	150	4	3	50
Mammographic	961	6	2	100
Optical	1 797	64	10	100
Parkinsons	195	22	2	60
Pima	768	8	2	80

在实验中,我们用 70%的样例作为训练集,用 30%的样例作为测试集. 对于不同的数据集,参数 m 设置为不同的值(见表 1 的最右一列),原则是 m 的值要大于数据集的维数. 而对于不同的数据集,参数 p 均设置为 5. 即对于每一个数据集,通过 Dropout 技术训练 5 个分类器 $\text{SLFN}_i (1 \leq i \leq 5)$. 在每一个数据集上,Dropout 掉的隐含层结点数及训练的基本分类器 $\text{SLFN}_i (1 \leq i \leq 5)$ 的测试精度列于表 2-表 12 中,与经典的 ELM 在测试精度上的比较结果列于表 13 中.

表 2 在数据集 Fertility 上实验的结果($m=20$)

Table 2 The experimental results on data set Fertility($m=20$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN_1	2	0.855 1
SLFN_2	7	0.843 2
SLFN_3	1	0.846 7
SLFN_4	11	0.827 2
SLFN_5	6	0.849 6

表 3 在数据集 Forest 上实验的结果($m=80$)

Table 3 The experimental results on data set Forest($m=80$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN_1	31	0.852 5
SLFN_2	42	0.825 1
SLFN_3	17	0.861 9
SLFN_4	49	0.844 7
SLFN_5	2	0.853 2

表 4 在数据集 Glass 上实验的结果 ($m=60$)

Table 4 The experimental results on data set Glass($m=60$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	8	0.923 8
SLFN ₂	37	0.882 7
SLFN ₃	4	0.891 0
SLFN ₄	21	0.903 2
SLFN ₅	27	0.884 6

表 6 在数据集 Ionosphere 上实验的结果 ($m=40$)

Table 6 The experimental results on data set Ionosphere($m=40$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	29	0.889 1
SLFN ₂	22	0.893 4
SLFN ₃	1	0.912 5
SLFN ₄	18	0.904 3
SLFN ₅	10	0.901 8

表 8 在数据集 Mammographic 上实验的结果 ($m=100$)

Table 8 The experimental results on data set Mammographic($m=100$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	62	0.778 0
SLFN ₂	9	0.805 4
SLFN ₃	14	0.810 9
SLFN ₄	27	0.801 1
SLFN ₅	10	0.812 4

表 10 在数据集 Parkinsons 上实验的结果 ($m=60$)

Table 10 The experimental results on data set Parkinsons($m=60$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	3	0.896 6
SLFN ₂	23	0.882 4
SLFN ₃	11	0.877 1
SLFN ₄	22	0.906 3
SLFN ₅	34	0.864 5

表 5 在数据集 Heart 上实验的结果 ($m=80$)

Table 5 The experimental results on data set Heart($m=80$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	11	0.923 7
SLFN ₂	8	0.897 3
SLFN ₃	28	0.928 2
SLFN ₄	53	0.902 2
SLFN ₅	19	0.918 1

表 7 在数据集 Iris 上实验的结果 ($m=50$)

Table 7 The experimental results on data set Iris($m=50$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	23	0.942 3
SLFN ₂	3	0.963 4
SLFN ₃	19	0.952 1
SLFN ₄	13	0.944 4
SLFN ₅	21	0.933 5

表 9 在数据集 Optical 上实验的结果 ($m=100$)

Table 9 The experimental results on data set Optical($m=100$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	51	0.928 4
SLFN ₂	8	0.953 6
SLFN ₃	17	0.942 1
SLFN ₄	45	0.932 7
SLFN ₅	24	0.937 8

表 11 在数据集 Pima 上实验的结果 ($m=80$)

Table 11 The experimental results on data set Pima($m=80$)

基本分类器	Dropout 的隐结点数	测试精度
SLFN ₁	59	0.742 1
SLFN ₂	37	0.765 1
SLFN ₃	19	0.806 4
SLFN ₄	18	0.791 3
SLFN ₅	11	0.824 2

从列于表 12 的实验结果可以看出,本文提出的方法在 10 个数据集上的测试精度均优于经典的 ELM,主要原因是从平均意义上,集成分类器的结果要优于每一个基本分类器的结果. 我们还对实验结果用成对 T-检验进行了统计分析,显著性水平参数 α 的值设置为 0.05,统计分析的结果列于表 13 中,从表 13 中的 P 值可以得出:“上述结论以 95%的结论成立”. 因此,可以肯定本文提出的方法优于经典 ELM.

表 12 与经典 ELM 比较的实验结果

Table 12 The experimental results compared with classical ELM

基本分类器	经典 ELM	本文方法
Fertility	0.826 7	0.861 4
Forest	0.836 4	0.860 0
Glass	0.878 3	0.931 5
Heart	0.889 6	0.943 3
Ionosphere	0.880 6	0.925 1
Iris	0.946 2	0.964 9
Mammographic	0.780 8	0.823 9
Optical	0.920 8	0.950 4
Parkinsons	0.864 6	0.890 6
Pima	0.767 6	0.823 3

表 13 实验结果的统计分析 ($\alpha=0.05$)

Table 13 The statistical analysis of experimental results($\alpha=0.05$)

基本分类器	P 值
Fertility	2.66E-10
Forest	3.13E-06
Glass	4.76E-05
Heart	2.38E-04
Ionosphere	3.19E-04
Iris	4.59E-03
Mammographic	2.09E-03
Optical	5.17E-06
Parkinsons	5.66E-05
Pima	1.28E-03

4 结论

本文提出了一种集成 Dropout 极限学习机的数据分类方法,该方法利用 Dropout 方法随机删除掉若干个 SLFNs 的隐含层结点,这样可以得到一个原 SLFN 的子网络. 对于给定的一个比较大的 SLFN,重复 p 次可以得到 p 个子网络. 以这 p 个子网络作为基本分类器,用多数投票法进行集成,并对测试样例进行分类. 本文提出的方法思想简单、易于实现,而且不用进行 SLFNs 的网络结构选择. 与传统的极限学习机在 10 个数据集上的实验结果及对实验结果的统计分析,证明了本文提出的方法在测试精度上均优于传统极限学习机. 本文对 ELM 的随机初始化用的都是服从均匀分布的随机数,下一步工作将研究用服从其他概率分布(如高斯分布、指数分布)的随机数进行初始化及其对测试精度是否会有本质的影响? 另外,进一步研究用其他的集成方法进行基本分类器的集成.

[参考文献]

- [1] HUANG G B, ZHU Q Y, SIEW C K. Extreme learning machine: a new learning scheme of feedforward neural networks[C]//Proceedings of International Joint Conference on Neural Networks (IJCNN2004), vol. 2, Budapest, Hungary, 25–29 July, 2004: 985–990.
- [2] HUANG G B, CHEN L, SIEW C K. Universal approximation using incremental constructive feedforward networks with random hidden nodes[J]. IEEE transactions on neural networks, 2006, 17(4): 879–892.
- [3] MOHAMMED A A, MINHAS R, JONATHAN Q M, et al. Human face recognition based on multidimensional PCA and extreme learning machine[J]. Pattern recognition, 2011, 44(10/11): 2 588–2 597.
- [4] CHACKO B P, VIMAL V R V, RAJU G, et al. Handwritten character recognition using wavelet energy and extreme learning machine[J]. International journal of machine learning and cybernetics, 2012, 3(2): 149–161.
- [5] 张弦, 王宏力. 基于贯序正则极端学习机的时间序列预测及其应用[J]. 航空学报, 2011, 32(7): 1 302–1 308.
- [6] YANG H, THOMAS P, OLGA F. Fault detection based on signal reconstruction with auto-associative extreme learning machines[J]. Engineering applications of artificial intelligence, 2017, 57: 105–117.
- [7] ZHU H, TSANG E C, WANG X Z, et al. Monotonic classification extreme learning machine[J]. Neurocomputing, 2017, 225: 205–213.
- [8] CHEN Y L, WU W. Mapping mineral prospectivity using an extreme learning machine regression[J]. Ore geology reviews, 2017, 80: 200–213.
- [9] XU S, WANG J. A fast incremental extreme learning machine algorithm for data streams classification[J]. Expert systems with applications, 2016, 65: 332–344.
- [10] ANAM K, AL-JUMAILY A. Evaluation of extreme learning machine for classification of individual and combined finger movements using electromyography on amputees and non-amputees[J]. Neural networks, 2017, 85: 51–68.
- [11] IOSIFIDIS A, TEFAS A, PITAS I. Approximate kernel extreme learning machine for large scale data classification[J]. Neurocomputing, 2017, 219: 210–220.
- [12] 裘日辉, 刘康玲, 谭海龙, 等. 基于极限学习机的分类算法及在故障识别中的应用[J]. 浙江大学学报(工学版), 2016, 50(10): 1 965–1 972.
- [13] HUANG G B, CHEN L, SIEW C K. Universal approximation using incremental constructive feedforward networks with random hidden nodes[J]. IEEE transactions on neural networks, 2006, 17(4): 879–892.
- [14] HUANG G B, LI M B, CHEN L, et al. Incremental extreme learning machine with fully complex hidden nodes[J]. Neurocomputing, 2008, 71(4/5/6): 576–583.
- [15] RONG H J, ONG Y S, TAN A H, et al. A fast pruned-extreme learning machine for classification problem[J]. Neurocomputing, 2008, 72(1/2/3): 359–366.
- [16] AKAIKE H. Information theory and an extension of the maximum likelihood principle[C]//Second International Symposium on Information Theory. Budapest: Akademiai Kiado, 1992: 267–281.
- [17] ZHAI J H, SHAO Q Y, WANG X Z. Improvements for P-ELM1 and P-ELM2 pruning algorithms in extreme learning machines[J]. International journal of uncertainty, fuzziness and knowledge-based systems, 2016, 24(3): 327–345.
- [18] ZHAI J H, SHAO Q Y, WANG X Z. Architecture selection of ELM networks based on sensitivity of hidden nodes[J]. Neural

- processing letters, 2016, 44(2): 1–19.
- [19] MICHE Y, SORJAMAA A, BAS P, et al. OP-ELM: optimally pruned extreme learning machine[J]. IEEE transactions on neural networks, 2010, 21(1): 158–162.
- [20] HINTON G E, SRIVASTAVA N, KRIZHEVSKY A, et al. Improving neural networks by preventing co-adaptation of feature detectors[DB/OL]. <https://arxiv.org/abs/1207.0580>, 2012.
- [21] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. Journal of machine learning research, 2014, 15(1): 1 929–1 958.
- [22] BALDI P, SADOWSKI P. The dropout learning algorithm[J]. Artificial intelligence, 2014, 210(210): 78–122.
- [23] WAGER S, WANG S, LIANG P. Dropout training as adaptive regularization[C]//Advances in Neural Information Processing Systems, Laka Tahoe, Nevada, 2013: 351–359.
- [24] IOSIFIDIS A, TEFAS A, PITAS I. DropELM: fast neural network regularization with dropout and drop connect[J]. Neurocomputing, 2015, 162: 57–66.
- [25] BA L J, FREY B. Adaptive dropout for training deep neural networks[C]//Advances in Neural Information Processing Systems, Laka Tahoe, Nevada, 2013.
- [26] LI Z, GONG B, YANG T. Improved dropout for shallow and deep learning[C]//Advances in Neural Information Processing Systems, Barcelona, Spain, 2016: 1–10.
- [27] YANG W, JIN L, TAO D, et al. Drop sample: a new training method to enhance deep convolutional neural networks for large-scale unconstrained handwritten Chinese character recognition[J]. Pattern recognition, 2016, 58: 190–203.
- [28] KLEIN E B, STONE W N, HICKS M W, et al. Understanding Dropouts[J]. Advances in neural information processing systems, 2013, 26(2): 89–100.
- [29] FRANK A, ASUNCION A. UCI machine learning repository[DB/OL]. [<http://archive.ics.uci.edu/ml>], 2013.

[责任编辑:顾晓天]

(上接第58页)

- [11] 公茂果, 焦李成, 杨咚咚, 等. 进化多目标优化算法研究[J]. 软件学报, 2009, 20(2): 271–289.
- [12] 王明昭, 王宇平, 王晓丽, 等. 一种均匀聚集距离的改进 NSGA-II 算法[J]. 西安电子科技大学学报(自然科学版), 2016, 43(3): 49–54.
- [13] DEB K, AGRAWAL S, PRATAP A, et al. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II[J]. Lecture notes in computer science, 2002, 1917: 849–858.
- [14] 陈婕, 熊盛武, 林婉如, 等. NSGA-II 算法的改进策略研究[J]. 计算机工程与应用, 2011, 47(19): 42–45.
- [15] AHMED F, DEB K. Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms[J]. Soft computing, 2013, 17(7): 1283–1299, .
- [16] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part ii: handling constraints and extending to an adaptive approach[J]. IEEE transactions on evolutionary computation, 2014, 18(4): 602–622.

[责任编辑:黄 敏]