

## 基于BP神经网络的自适应伪最近邻分类

曾勇 舒欢\* 胡江平 葛月月

(电子科技大学自动化工程学院 成都 611731)

**摘要:**在伪最近邻(PNN)分类算法中,待分类样本点与每一类样本集中各个近邻的距离加权系数都是主观确定的,这就使得算法得不到最优距离加权值。针对这一问题,该文提出一种基于BP神经网络的自适应伪最近邻分类算法。首先通过计算待分类样本点与每一类样本集中各个近邻的距离值,并将其作为BP神经网络的输入。然后根据BP神经网络输入与输出之间的映射来自适应地确定相应的距离加权值。最后由BP神经网络的输出值判别样本类别号。实验结果表明,该算法能够自适应地调节距离加权系数,同时还能有效地改善分类准确率。

**关键词:** 伪最近邻分类; BP神经网络; 自适应

中图分类号: TP181

文献标识码: A

文章编号: 1009-5896(2016)11-2774-06

DOI: 10.11999/JEIT160133

## Adaptive Pseudo Nearest Neighbor Classification Based on BP Neural Network

ZENG Yong SHU Huan HU Jiangping GE Yueyue

(School of Automation Engineering, University of Electronic Science and  
Technology of China, Chengdu 611731, China)

**Abstract:** Distance-weighted coefficients between unlabeled sample point and its nearest neighbors belonging to same sample set are determined subjectively in the Pseudo Nearest Neighbor (PNN) classification algorithm, which makes it difficult to obtain optimal distance-weighted value. In this paper, an adaptive pseudo neighbor classification algorithm based on BP neural network is proposed. Firstly, the distance-weighted values between unlabeled sample point and its neighbors lying in the same sample set are regarded as the input of BP neural network. Secondly, the corresponding distance-weighted values are adaptively determined according to the mapping between the inputs and outputs of BP neural network. Finally, the classification of unlabeled sample point is judged by the outputs of BP neural network. Experimental results show that the proposed approach adaptively adjusts the distance-weighted coefficients. Moreover, the classification accuracy can be effectively improved.

**Key words:** Pseudo Nearest Neighbor (PNN) classification; BP neural network; Adaptive

### 1 引言

$k$ -近邻分类规则(the voting  $k$ -Nearest Neighbor Rule,  $k$ -NNR)是一种经典的非参数分类方法,作为数据挖掘领域的十大经典算法之一<sup>[1]</sup>,已在字符识别<sup>[2]</sup>、文本分类<sup>[3]</sup>、时间序列<sup>[4-6]</sup>、生物医学研究<sup>[7,8]</sup>、空间查询<sup>[9]</sup>等诸多领域<sup>[10,11]</sup>得到了广泛的应用。在 $k$ -近邻分类规则里,测试样本的 $k$ 个近邻类别信息被赋予相同的权值。事实上,不同近邻对测试样本的分类会产生不同的影响。基于此,文献

[12-14]提出了几种距离加权的 $k$ -近邻规则。在加权的 $k$ -近邻分类规则里,与测试样本距离较近的近邻被赋予较大的权值,反之,距离较远的近邻被赋予较小的权值。在某些场合,文献[12]已经发现加权的 $k$ -近邻规则可以比传统的 $k$ -近邻规则取得较好的性能。但是一些学者发现,在样本无限的情况下,加权的 $k$ -近邻规则的渐进分类误差率并不好于传统的 $k$ -近邻规则<sup>[15,16]</sup>。

受到距离加权的 $k$ -近邻分类规则的启发,文献[17]提出了伪最近邻分类规则(Pseudo Nearest Neighbor Rule, PNNR)。在伪最近邻分类规则中,一个称之为伪最近邻(Pseudo Nearest Neighbor, PNN)的新的最近邻取代原来的最近邻,伪最近邻由待分类样本点的属于同一类别的 $k$ 个近邻的距离加权和决定,待分类样本被指定为伪最近邻所属的

收稿日期: 2016-01-29; 改回日期: 2016-06-17; 网络出版: 2016-09-08

\*通信作者: 舒欢 shuhuan163@163.com

基金项目: 国家自然科学基金(61104104, 61473061), 四川省信号与信息重点实验室基金(SZJJ2009-002)

Foundation Items: The National Natural Science Foundation of China (61104104, 61473061), The Fund of Sichuan Provincial Key Laboratory of Signal and Information Processing (SZJJ2009-002)

类别。PNNR 充分利用了测试样本在每一类原型样本集里的多个近邻信息，大量的实验结果显示该算法的分类性能优于传统的最近邻分类方法与传统的  $k$ -近邻分类方法，也优于传统的距离加权的近邻分类算法。然而在 PNNR 算法中，对待分类样本点的属于同一类别的  $k$  个近邻的距离进行加权时，无论采用倒数距离加权，还是采用指数逆距离加权，距离加权系数都是人为主观确定的，并不能得到较优的距离加权值。

为了解决 PNNR 存在的不足，本文提出了一种新的分类方法：基于 BP 神经网络的自适应伪最近邻分类方法(Adaptive pseudo Nearest Neighbor classification based on BP neural network, BPANN)，根据同类样本特征相似，而不同样本特征值差异较大这一特性来计算测试样本在每一个类别中的  $k$  个近邻点，充分利用了测试样本在每一类原型样本集里的多个近邻信息，将计算出的测试样本与各近邻点间的距离值作为网络输入，并通过 BP 神经网络输入和输出之间的映射自适应地训练距离加权系数，使得分类器的分类精度得以提高。

本文其余部分组织如下：第 2 节介绍伪最近邻分类算法；第 3 节将给出本文所提出的 BPANN 的具体算法步骤以及分类器设计；第 4 节是实验及结果分析；最后是本文结论。

## 2 伪最近邻分类

对于  $N$  个原型样本， $N_1, N_2, \dots, N_M$  分别各自表示属于类  $1, 2, \dots, M$  的原型样本数，在伪最近邻分类规则里， $x_j^{(1)}, x_j^{(2)}, \dots, x_j^{(k_j)}$  表示测试样本  $x$  在第  $j$  类原型样本里的  $k_j$  个近邻， $d_j^{(1)}, d_j^{(2)}, \dots, d_j^{(k_j)}$  表示其对应按升序排列的与测试样本  $x$  的距离，为  $k_j$  个近邻分配权值  $w_j^1, w_j^2, \dots, w_j^{k_j}$ ，这些权值有多种分配方式，如加权距离赋值方式、指数逆距离加权方式、倒数距离加权等方式，并且  $w_j^1, w_j^2, \dots, w_j^{k_j}$  必须满足式(1)关系：

$$w_j^{k_j} \leq w_j^{k_j-1} \leq \dots \leq w_j^2 \leq w_j^1, \quad w_j^1 = 1 \quad (1)$$

式中，取测试样本  $x$  在不同类的原型样本集里的近邻数相等，即令  $k_j = k_i = k, i \neq j$ ，且  $k \leq \min(N_1, N_2, \dots, N_M)$ ，近邻数  $k_j$  是利用交叉验证方法得到的。同时，令相应的权值  $w_i^m = w_j^m$ ，这里  $m \in \{1, 2, \dots, k\}$ ， $w_i^m$  与  $w_j^m$  分别是测试样本  $x$  在第  $i$  类原型样本里的第  $m$  个近邻  $x_i^{(m)}$  的距离加权值及测试样本  $x$  在第  $j$  类原型样本里的第  $m$  个近邻  $x_j^{(m)}$  的距离加权值。令  $d_j$  表示测试样本  $x$  在第  $j$  类原型样本里的  $k$  个近邻的距离加权和，则有

$$d_j = w_j^1 \times d_j^1 + \dots + w_j^k \times d_j^k, \quad j = 1, 2, \dots, M \quad (2)$$

定义测试样本  $x$  的伪最近邻  $x_{PNN}$  为

$$x_{PNN} = \min(d_1, d_2, \dots, d_M) \quad (3)$$

必须指出的是，这里测试样本  $x$  的伪最近邻  $x_{PNN}$  是一个标量， $x_{PNN}$  并不是测试样本  $x$  真实的最近邻，它是一个假的最近邻，因而称为伪最近邻。伪最近邻  $x_{PNN}$  代表的是测试样本  $x$  在某一类原型样本里的  $k$  个近邻的距离加权值的最小值，实际上最后代表的是类别信息，因此可用它来进行分类，即伪最近邻规则。

由文献[17]提出的伪最近邻规则 PNNR(Pseudo Nearest Neighbor Rule)如下：一个测试样本对  $(x, \theta)$  给定，则伪最近邻分类规则把测试样本  $x$  分配为其伪最近邻  $x_{PNN}$  所属的类别，如果有多个伪最近邻，则在其中随机选择一个，并把其对应的类别指定给测试样本。

实验表明伪最近邻分类方法的分类性能优于传统的最近邻分类方法与传统的  $k$ -近邻分类方法，也优于传统的距离加权的近邻分类算法<sup>[17]</sup>，但由于其中的距离加权系数都是人为主观确定的，并不能得到较优的距离加权值，为此本文提出了一种基于 BP 神经网络的自适应伪最近邻分类的方法。

## 3 基于 BP 神经网络的自适应伪最近邻分类

基于 BP 神经网络的自适应伪最近邻分类是伪最近邻分类方法的扩展，与 PNN 分类算法不同的是，BP 神经网络分类器的输入不是待分类样本点的特征值，而是待分类样本在每一类样本集中的各个近邻的距离值。并且距离加权值不需要人为确定，而是由 BP 神经网络输入和输出之间的映射自适应确定，同时对每类原型样本自适应的设计其相应的分类器，从有效的样本资料中得到尽可能多的信息，使其获得更好的分类效果。图 1 显示的是自适应伪最近邻分类器的训练原理图，其中输入数据是训练数据集， $N_1, N_2, \dots, N_M$  分别各自表示属于类  $1, 2, \dots, M$  的原型样本数， $d_j^{(1)}, d_j^{(2)}, \dots, d_j^{(k)}$  表示其对应  $k$  个近邻按升序排列的与测试样本  $x$  的距离， $\varphi_1, \varphi_2, \dots, \varphi_M$  表示  $M$  个分类器的输出值。

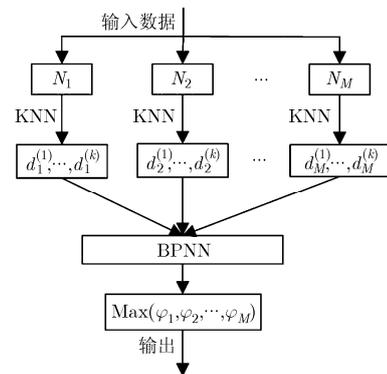


图 1 自适应伪最近邻分类器训练原理图

### 3.1 BPANN 模型的参数设置

BP 神经网络具有良好的容错性、与人脑相似的高度并行性以及联想记忆功能,容错能力和自适应学习都较强,可以实现从输入到输出的非线性映射。应用于近邻分类的 BP 神经网络分类器,必须结合数据集的情况设计,并在试验中不断改进,才能训练出泛化性能好的模式分类器。所以,必须选择适当大小的网络结构,网络太小不能解决问题,太大则推广能力差。本文中,BP 神经网络在样本训练阶段通过附加动量法来调整层与层之间的权值和阈值,从而通过网络输入输出之间的映射自适应的调节距离加权重值,同时针对样本数据的数据类别自适应的设计其相应的分类器,以便选出较优的分类器。

(1)输入与输出层节点数确定: BP 网络的输入、输出层维数需要根据实际要求而定,本实验中,若样本预处理时采用的是  $k$  近邻,那么输入层的维数就为  $k$ ; 而输出层输出的则是样本的相似度即新的距离加权和,因此输出层的维数为 1。

(2)隐层层数的选择及隐层节点数: 根据戈尔莫戈罗夫(Kolmogorov)定理,一个 3 层的 BP 网络足以完成任何从输入到输出的连续映射,因此,我们采用具有一个隐层的 3 层 BP 神经网络。隐节点数目的选择是一个比较复杂的问题,目前确定隐节点数的方法有很多种,主要有修剪方法、复杂性调整方法、增益方法、进化方法、自适应方法<sup>[18]</sup>等。在大量实验的基础上,这里选择式(4)作为参考。

$$N_{\text{Hidden}} = (N_{\text{In}} + \max(N_{\text{Out}}, N_{\text{Class}})) / 2 \quad (4)$$

其中,  $N_{\text{Hidden}}$  为隐层节点数,  $N_{\text{In}}$ ,  $N_{\text{Out}}$ ,  $N_{\text{Class}}$  分别为 BP 网络的输入、输出层节点数和所需分的目标分类数即样本数据的类别数。

(3)激活函数的选择: 神经元的激活函数一般选用 Sigmoid 函数,经过大量实验对比,最终我们选取式(5)所示 logistic 函数作为激活函数。

$$\varphi_j(v_j(n)) = \frac{1}{1 + \exp(-v_j(n))} \quad (5)$$

其中,  $v_j(n)$  是神经元  $j$  的诱导局部域,其范围位于  $(-\infty, +\infty)$ , 输出  $\varphi_j(v_j(n))$  的范围位于  $[0, 1]$  区间内。

(4)初始权值的选取: 因系统的非线性性使初始权值对学习是否收敛关系很大,故而希望初始权值在输入累加时使每个神经元的状态值接近于零。一般,初始权值取随机数,而且权的值要求比较小。

(5)学习率以及冲量项的选择: 原则上,只要学习率足够小以保证收敛,但实际上学习率可以影响到最后的网络性能。而冲量项的目的在于: 允许当误

差曲面中存在平坦区时,网络可以以更快的速度学习,增加了学习过程的稳定性。对于我们所用的 Sigmoid 型网络,可以首先将学习率  $\eta$  设为 0.2,冲量项  $\alpha$  设为 0.9,然后可以在学习过程中适当的改动。

### 3.2 BPANN 分类方法的实现

基于 BP 神经网络的自适应伪最近邻分类方法实现的流程图如图 2 所示。

由于在实际应用中用于训练的样本各元素之间取值范围不可能完全一致,这就给网络的训练带来很大不便,不仅加大了逼近函数的波动性,使网络训练速度下降,而且容易造成网络训练失败。因此先对数据进行适当的预处理是非常重要的,这在一定程度上可以加速训练,提高训练的成功率。样本集经过预处理后,便将其送入 BP 神经网络中进行网络训练。对于一个数据集中  $N$  个可得的训练样本,令  $N_1, N_2, \dots, N_M$  分别表示对应于属于类  $1, 2, \dots, M$  的训练样本数。

BPANN 的具体步骤如下:

步骤 1 将样本数据集(data)分为训练集(trainset)和测试集(testset),进行数据预处理。计算每个训练样本点  $x$  在每一类训练样本中的  $k$  个近邻  $x_j^{(1)}, x_j^{(2)}, \dots, x_j^{(k)} (j = 1, 2, \dots, M)$ , 以及到各个近邻的距离,并将其按升序排列为  $d_j^{(1)}, d_j^{(2)}, \dots, d_j^{(k)} (j = 1, 2, \dots, M)$ , 对每一类样本数据经过归一化处理后计算对应的正例、反例,类  $1, 2, \dots, M$  的正例、反例个数分别为  $N_1, N_2, \dots, N_M$ , 其中正例指在类内寻找的样本点,反例指从类间寻找的样本。将每一类的正例、反例作为 BP 神经网络的输入,表示为  $X$ , 并且数据集中每一类样本分别对应一个 BP 网络分类器,类  $1, 2, \dots, M$  共对应  $M$  个分类器。

步骤 2 for  $n = 1$  to  $M$

{

初始化 BP 神经网络,即设定网络参数。其中输入层到隐含层与隐含层到输出层的权值和偏置分别表示为  $w_1, w_2, b_1, b_2$ , 将它们分别取随机数。

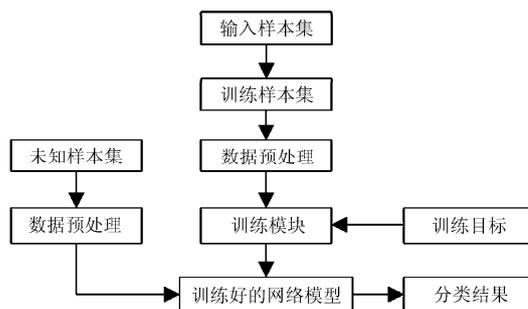


图 2 基于 BP 神经网络的自适应伪最近邻分类方法实现流程

完成步骤 3-步骤 5。

}

步骤 3 预处理后的数据送入网络中进行训练，得到新的权值以及偏置。

本文采用附加动量法来作为权值和偏置的学习算法，其权值学习公式为

$$w(t) = w(t-1) + \Delta w(t) + \alpha[w(t-1) - w(t-2)] \quad (6)$$

式中， $w(t), w(t-1), w(t-2)$  分别为  $t, t-1, t-2$  时刻的权值，其中  $\eta$  为学习率， $\alpha$  为冲量项， $\Delta w(t) = \eta \times dw(t)$ ， $dw(t)$  表示对相应的权值求导。权值和偏置  $w1, w2, b1, b2$  的变化率分别为

$$db1 = d[\text{logsig}(I)] \times (w2 \times e), \quad db2 = e,$$

$$dw1 = d[\text{logsig}(I)] \times X \times (w2 \times e), \quad dw2 = I_{\text{out}} \times e \quad (7)$$

式中  $e$  是网络输出的误差，用于权值和偏置的修正， $I$  是网络隐含层的输出， $I = w1 \times X + b1$ ， $I_{\text{out}} = \text{logsig}(w1 \times X + b1)$ ，网络输出层的输出为  $y = w2^T \times I_{\text{out}} + b2$ 。这个训练过程由 BP 神经网络输入输出间的映射自适应调节权值，从而替代了 PNN 中人为计算距离加权系数。

步骤 4 将预处理后的测试样本送入已经训练好的网络中进行分类，找出网络输出值中的最大值，也就是新的伪最近邻，并将测试样本分到最大值对应的索引类。

步骤 5 分类结果与期望值对比计算误差率  $\text{Err}$ ， $\text{Err} = \frac{N_{\text{df}}}{N_n}, n = 1, 2, \dots, M$ ，其中  $N_{\text{df}}$  表示实际分类输出值与期望值不同的样本数据个数。

步骤 6 选择出  $M$  个 BP 网络分类器中误差率最小即分类效果最好的一个。

## 4 实验结果及分析

实验是在 MATLAB7.11.0 环境下实现，采用了

机器学习库 UCI<sup>[19]</sup>上的 9 个数据集。所使用的数据介绍见表 1。数据集 Letter, Pen, Thyroid, Optdigits, Landsat-Satellite 和 Image-Segmentation，其训练样本集与测试样本集已被预先指定。而其余的 3 个数据集，通过 5 倍交叉验证来选择训练集与测试集，其中对于数据集采用的距离度量是欧几里得距离。

现在用 BPANN 与 PNN 以及传统的 KNN 一起对机器学习库 UCI<sup>[19]</sup>上的 9 个数据集进行分类，其中 PNN1, PNN2, PNN3, PNN4 分别是距离逆加权、指数衰减距离加权、线性距离逆加权、倒数距离加权的伪最近邻分类，分类结果见表 2。对每一个数据集，几种分类方法中最好的分类结果用黑体表示。

从表 2 可以看到，对数据集 Letter, Pen, Optdigits, Image-Segmentation, Landsat-Satellite 以及 Wine，基于 BP 神经网络的自适应伪最近邻分类的分类性能明显好于传统的近邻分类以及伪最近邻分类。而在数据集 Thyroid 上，BPANN 也取得了较好的分类效果。对于数据集 Iris 和 Glass，最终分类效果没有得到明显改善，这是由于它们属于小样本数据集，先前没有分出训练集和测试集。而 BPANN 分类算法在分类器设计阶段，是通过 5 倍交叉验证来选择训练集与测试集，交叉分组的训练数据每次的变化会对权值优化产生较大影响，进而影响分类结果。并且该算法在小样本数据集上的误差率是 5 次分类结果的平均值，避免了实验结果由于训练集与测试集选择的随机性引起的偶然性。

表 3 为表 2 中几种算法取得相应分类结果的分时间(因为 KNN 直接计算测试样本与训练样本之间的距离，不需要进行训练，因此实验中记录的是各个算法的测试时间，即分类时间)，表中时间单位均为秒，几种分类算法分别在每个数据集上最少的分类时间用黑体表示。

表 1 所使用数据集的一些特征

仿真所使用的数据集	特征维数	样本数	类数	误差估计
Letter	16	16000 个训练样本	26	测试样本 4000 个
Pen	16	7494 个训练样本	10	测试样本 3498 个
Thyroid	21	3772 个训练样本	3	测试样本 3428 个
Optdigits	64	3823 个训练样本	10	测试样本 1797 个
Landsat-Satellite	36	4435 个训练样本	3	测试样本 2000 个
Image-Segmentation	19	210 个训练样本	7	测试样本 2100 个
Iris	4	150	3	5CV
Glass	9	214	6	5CV
Wine	13	178	3	5CV

表 2 在 9 个数据集上的分类误差(%)

数据集	KNN	PNN1	PNN2	PNN3	PNN4	BPANN
Letter	4.12 $k=3$	3.80	3.75	4.30	3.93	<b>3.67</b>
Pen	2.12 $k=4$	1.94	2.26	2.26	1.97	<b>1.92</b>
Thyroid	<b>6.33</b> $k=5$	6.65	6.42	8.02	6.42	<b>6.33</b>
Optdigits	2.00 $k=1$	1.67	1.84	2.00	1.67	<b>1.56</b>
Landsat-Satellite	10.60	10.55	10.35	10.55	9.90	<b>8.90</b>
Image-Segmentation	12.33	12.33	12.33	12.33	12.33	<b>12.14</b>
Wine	30.72	28.57	27.47	28.60	28.01	<b>25.73</b>
Iris	<b>2.67</b>	<b>2.67</b>	<b>2.67</b>	3.33	<b>2.67</b>	3.33
Glass	<b>35.05</b>	37.17	37.61	37.20	36.28	35.27
平均误差率	11.77	11.71	11.63	12.36	11.38	<b>10.98</b>

表 3 不同算法在各数据集上的分类时间(s)

数据集	KNN	PNN1	PNN2	PNN3	PNN4	BPANN
Letter	35.9965	38.1466	44.4699	45.7013	41.6946	<b>23.9958</b>
Pen	14.3465	18.0775	14.7245	13.4599	12.5943	<b>8.0977</b>
Thyroid	16.9841	13.5748	11.9098	11.4273	11.0418	<b>3.6130</b>
Optdigits	14.9394	18.9481	14.3248	16.6299	13.9480	<b>4.1931</b>
Landsat-Satellite	13.4912	13.3566	11.0128	11.6741	11.7278	<b>2.7176</b>
Image-Segmentation	3.9030	4.0412	4.0186	3.3453	3.7430	<b>3.3353</b>
Wine	0.0163	0.0090	0.0296	0.0091	<b>0.0085</b>	0.0234
Iris	<b>0.0119</b>	0.0216	0.0342	0.0378	0.0269	0.0187
Glass	<b>0.0091</b>	0.0100	0.0126	0.0190	0.0257	0.0344

由表 3 可得, BPANN 算法在数据集 Letter, Pen, Thyroid, Optdigits, Landsat-Satellite 以及 Image-Segmentation 上的分类时间明显的小于 PNN 以及 KNN, 从算法原理可对此作出解释, PNN 是通过计算距离加权和来分类, 而 BPANN 是由神经网络输入与输出间的映射来调节权值并分类, 使得分类时间相对较少。对于小样本数据集 Wine, Iris 和 Glass, BPANN 在分类时间上没有取得明显改善, 这是由于为了避免实验的偶然性, 其分类时间计算的是交叉验证次数的平均值。

## 5 结论

针对 PNN 算法中距离加权系数的确定问题, 本文提出了一种新的伪最近邻方法: 基于 BP 神经网络的自适应伪最近邻分类方法。在该分类方法中, BP 神经网络的输入不是待分类样本点的特征值, 而是待分类样本在每一类样本集中的各个近邻的距离值, 同时距离加权值不需要人为确定, 而是由 BP 神经网络输入和输出之间的映射自适应确定, 并且对每一个数据集自适应的设计其相应分类器。因此

在整个分类过程中进一步减少了主观因素的参与成分, 这使得分类器性能具有更好的稳定性和推广性。在多个 UCI 数据集上的实验结果表明, 该算法与传统的 KNN 算法以及 PNN 算法相比, 取得了更好的分类性能。

## 参考文献

- [1] WU Xindong, KUMAR V, QUINLAN J R, *et al.* Top 10 algorithms in data mining[J]. *Knowledge and Information Systems*, 2008, 14(1): 1-37. doi: 10.1007/s10115-007-0114-2.
- [2] MATEI O, POP P C, and VĂLEAN H. Optical character recognition in real environments using neural networks and k-nearest neighbor[J]. *Applied Intelligence*, 2013, 39(4): 739-748. doi: 10.1107/s10489-013-0456-2.
- [3] WAN C H, LEE L H, RAJKUMAR R, *et al.* A hybrid text classification approach with low dependency on parameter by integrating k-nearest neighbor and support vector machine[J]. *Expert Systems with Applications*, 2012, 39(15): 11880-11888. doi: 10.1016/j.eswa.2012.02.068.
- [4] CARAWAY N M, MCCREIGHT J L, and RAJAGOPALAN

- B. Multisite stochastic weather generation using cluster analysis and  $k$ -nearest neighbor time series resampling[J]. *Journal of Hydrology*, 2014, 508: 197–213. doi: 10.1016/j.jhydrol.2013.10.054.
- [5] RAHMAN S A, HUANG Y, CLAASSEN J, *et al.* Combining Fourier and lagged  $k$ -nearest neighbor imputation for biomedical time series data[J]. *Journal of Biomedical Informatics*, 2015, 58: 198–207. doi: 10.1016/j.jbi.2015.10.004.
- [6] GONZÁLEZ Mabel, BERGMEIR Christoph, TRIGUERO Isaac, *et al.* On the stopping criteria for  $k$ -Nearest Neighbor in positive unlabeled time series classification problems[J]. *Information Sciences*, 2016, 328: 42–59. doi: 10.1016/j.ins.2015.07.061.
- [7] WANG A, AN N, CHEN G, *et al.* Accelerating wrapper-based feature selection with  $k$ -nearest-neighbor[J]. *Knowledge-Based Systems*, 2015, 83: 81–91. doi: 10.1016/j.knosys.2015.03.009.
- [8] CHEN C H, HUANG W T, Tan T H, *et al.* Using  $K$ -nearest neighbor classification to diagnose abnormal lung sounds[J]. *Sensors*, 2015, 15(6): 13132–13158. doi: 10.3390/s150613132.
- [9] HAN Y, PARK K, HONG J, *et al.* Distance-constraint  $k$ -nearest neighbor searching in mobile sensor networks[J]. *Sensors*, 2015, 15(8): 18209–18228. doi: 10.3390/s150818209.
- [10] TOMAŠEV N and MLADENIĆ D. Hubness-aware shared neighbor distances for high-dimensional  $k$ -nearest neighbor classification[J]. *Knowledge and Information Systems*, 2014, 39(1): 89–122. doi: 10.1007/s10115-012-0607-5.
- [11] CHOI Sangil, YOUN Ik-hyun, LEMAY Richelle, *et al.* Biometric gait recognition based on wireless acceleration sensor using  $k$ -nearest neighbor classification[C]. 2014 IEEE International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, 2014: 1091–1095. doi: 10.1109/ICCNC.2014.6785491.
- [12] DUDANI S A. The distance-weighted  $k$ -nearest-neighbor rule[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1976, 6(4): 325–327. doi: 10.1109/TSMC.1976.5408784.
- [13] GOU Jianping, XIONG Taisong, and KUANG Yin. A novel weighted voting for  $k$ -nearest neighbor rule[J]. *Journal of Computers*, 2011, 6(5): 833–840. doi: 10.4304/jcp.6.5.833-840.
- [14] GOU Jianping, DU Lan, ZHANG Yuhong, *et al.* A new distance-weighted  $k$ -nearest neighbor classifier[J]. *Journal of Information & Computational Science*, 2012, 9(6): 1429–1436.
- [15] BAILY T and JAIN A K. A note on distance-weighted  $k$ -nearest neighbor rules[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1978, 8(4): 311–313. doi: 10.1109/TSMC.1978.4309958.
- [16] MORIN R L and RAESIDE B E. A reappraisal of distance-weighted  $k$ -nearest-neighbor classification for pattern recognition with missing data[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1981, 11(3): 241–243. doi: 10.1109/TSMC.1981.4308660.
- [17] ZENG Yong, YANG Yupu, and ZHAO Liang. Pseudo nearest neighbor rule for pattern classification[J]. *Expert Systems with Applications*, 2009, 36: 3587–3595. doi: 10.1016/j.eswa.2008.02.003.
- [18] 杨凡, 赵建民, 朱信忠. 一种基于BP神经网络的车牌字符分类识别方法[J]. *计算机科学*, 2005, 32(8): 192–195.
- YANG Fan, ZHAO Jianmin, and ZHU Xinzong. A new method of license plate characters classified recognition based on BP neural networks[J]. *Computer Science*, 2005, 32(8): 192–195.
- [19] BACHE K and LICHMAN M. UCI repository of machine learning databases[OL]. <http://www.ics.uci.edu/~mllearn/MLRepository.html>. 2014.
- 曾 勇： 男，1968 年生，博士，副教授，研究方向为智能信息处理、智能控制理论与应用、计算机视觉与模式识别。
- 舒 欢： 女，1991 年生，硕士生，研究方向为模式识别。
- 胡江平： 男，1977 年生，教授，博士生导师，研究方向为多智能系统建模与控制、传感器网络信息融合、智能飞行控制等。