文章编号:1001 - 9081(2011)02 - 0501 - 03
doi:10.3724/SP. J. 1087. 2011. 00501

基于动量粒子群的混合核 SVM 参数优化方法

王　佳,徐蔚鸿
(长沙理工大学 计算机与通信工程学院,长沙 410114)
(jiajia123a@163.com)

摘　要:支持向量机(SVM)可以很好地用来解决分类问题,参数优化尤其重要。混合核函数的引入,使得 SVM 又多了一个可调参数。针对该参数用人工或经验的方法获取具有局限性,采用动量粒子群(MPSO)对 SVM 基本参数、混合可调核参数进行综合寻优,来寻找最佳参数组合。通过 UCI 数据仿真,对比结果表明:所提优化方法能够快速有效地提取最佳参数组合,所得 SVM 性能明显提高,分类效果更好。

关键词:混合核;动量粒子群优化;参数优化;分类
中图分类号: TP181　　**文献标志码**:A

Parameter optimization of mixed kernel SVM based on momentum particle swarm optimization

WANG Jia, XU Wei-hong
(*School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha Hunan* 410114, *China*)

Abstract: Support Vector Machine (SVM) can be used to solve classification problems, and it is very important to optimize its parameters. With the introduction of mixed kernels, SVM has one more adjustable parameter. Because it is hard to obtain the parameter by manual or experience, Momentum Particle Swarm Optimization (MPSO) was used to find the best combination of the basic parameters and mixed adjustable nuclear parameter of SVM. Finally, the simulations of UCI data show that the proposed algorithm provides an effective way to search the best parameters combination, and makes SVM have higher performance and better classification accuracy.

Key words: mixed kernel; Momentum Particle Swarm Optimization (MPSO); parameter optimization; classification

0　引言

支持向量机[1](Support Vector Machine, SVM)在解决小样本、非线性及高维模式识别问题中表现出了许多特有的优势,因而得到极大重视。目前已广泛应用于模式识别、信号处理和生物发酵软测量等众多领域[2-5]。

支持向量机的核心在于核函数,核函数的构造对于支持向量机的性能起着至关重要的作用。混合核函数[6-7]是其中的一种构造方法,一般由一个局部性核函数和一个全局性核函数线性组合而成,权系数的确定对混合核函数性起关键作用。目前,关于混合核 SVM 参数优化的方法都是针对惩罚系数和核参数,而对于混合核函数中可调参数如何选取的文献却很少,一般均采用经验值,这无疑使得 SVM 的参数优化只能达到局部最优,而无法达到全局最优。

本文采用一种改进的粒子群算法,即动量粒子群(Momentum Particle Swarm Optimization, MPSO)算法对 3 个参数进行综合寻优,该算法不但保持了基本 PSO 算法的简单、易实现等优点,而且能有效提高算法的收敛速度,在寻优的过程中能在较少的进化代数内达到较好的寻优效果,并能部分避免算法的后期振荡,以取得最佳的参数组合,为混合核 SVM 参数优化提供了一种新方法。

1　SVM 的理论基础[8]及参数分析

SVM 就是通过核函数把低维的线性不可分问题映射到高维空间,从而转化为线性可分问题。图 1 假设是已经映射好的高维空间,空心的方格和圆圈各代表一类。

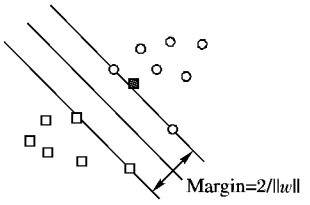

图 1　svm 的高维映射

此时有一个新的样本需要添加,结果其映射到高维空间后成为图 1 中的实心方格,这样的点无疑给分类问题带来了一定的偏差,称这样的点为离群点,此时就要放低对一些点到分类平面的距离不满足最初的分类要求,最初的优化问题是:

$$\min \frac{1}{2}\|\omega\|^2$$

$$\text{s.t. } y_i[(\omega x_i) + b] \geqslant 1$$

$i = 1,2,\cdots,l$;l 是样本数

$\|\omega\|^2$ 是目标函数,希望它越小越好。但由于离群点的出现,务必会使其变大,造成一定的损失。此时的优化问题就转化为:

$$\min \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{l}\zeta_i$$

$$\text{s.t. } y_i[(\omega x_i) + b] \geqslant 1 - \zeta_i$$

收稿日期:2010 - 07 - 27;**修回日期**:2010 - 09 - 17。
基金项目:教育部重点科研基金资助项目(208098);湖南省教育厅重点项目(07A056)。
作者简介:王佳(1987 -),女,河南西华人,硕士研究生,主要研究方向:人工智能;　徐蔚鸿(1963 -),男,湖南湘潭人,教授,博士,主要研究方向:人工智能、模式识别。

$i = 1,2,\cdots,l$;l 是样本数

用 $\sum_{i=1}^{l}\zeta_i$ 来表示损失,将其加入目标函数时,就需要一个惩罚因子 C,此参数就是 SVM 中提到的需要优化的参数。惩罚因子 C 决定了重视离群点带来损失的程度。当 $\sum_{i=1}^{l}\zeta_i$ 一定时,定的 C 越大,对目标函数的损失也就越大,此时就暗示着对离群点重视的程度比较大,最极端的情况就是把 C 定为无限大,这样只要稍有一个点离群,目标函数值就会变成无限大,使问题无解。因此在对参数组合进行寻优时,当有几组组合都可以取得相同的识别率时,一般会取 C 值较小的那组,可以有效避免 SVM 的过学习现象,即训练集分类准确率很高而测试集分类准确率很低(分类器的泛化能力降低)。

一般 SVM 默认核函数为 RBF 核,该核中核参数 γ[9] 是众多参考文献中 SVM 需要优化的另一个重要参数,它反应了支持向量之间的相关程度。γ 太小,支持向量间的联系就比较松弛;γ 太大,支持向量间的影响就会过强,模型精度难以满足要求。当 γ 很小时,惩罚因子 C 可以相应取小一点,以保证模型的推广能力。

2 MPSO 优化混合核 SVM 参数

2.1 混合核函数

鉴于局部性核函数学习能力强、泛化性能弱,而全局性核函数泛化性能强、学习能力弱,为了得到学习能力与泛化能力都较强的核函数,将这两类核函数混合起来。RBF 核函数是一个典型的局部性核函数,使用比较普遍,而多项式核函数是一个典型的全局性核函数,因此选择将二者线性组合,构造混合核函数[10] 如式(1)所示。

$$K_{\text{mix}} = \lambda K_{\text{poly}} + (1-\lambda)K_{\text{rbf}} \tag{1}$$

其中:$\lambda \in (0,1)$

$$K_{\text{poly}} = [(\boldsymbol{x}\cdot\boldsymbol{x}_i)+1]^q$$

$$K_{\text{rbf}} = (-\gamma\|\boldsymbol{x}-\boldsymbol{x}_i\|^2)$$

此混合核函数满足 Mercer 条件,核函数的混合使需要优化的 SVM 参数增一。实验得到的 λ 的值一般在 0.50 ~ 0.99,因此可以在此范围内对其寻优。

2.2 MPSO 算法

PSO 算法[11] 采用的是速度—位置搜索模型,在每一次迭代中,粒子通过跟踪 2 个极值(全局极值 g_{best} 与个体极值 p_{best})和前一时刻的状态,来不断地更新自己当前在解空间中的位置,从而找到问题的最优解。其迭代公式如式(2)、(3)所示。

$$v_i(k+1) = \omega\cdot v_i(k) + c_1\cdot rand_1(k)\cdot(p_{\text{ibest}} - x_i(k)) + c_2\cdot rand_2(k)\cdot(g_{\text{best}} - x_i(k)) \tag{2}$$

$$x_i(k+1) = x_i(k) + v_i(k+1) \tag{3}$$

其中:$v(k)$,$v(k+1)$,$x(k)$,$x(k+1)$ 分别是粒子当前时刻、下一时刻的速度及所处位置;$rand_1(k)$,$rand_2(k)$ 是介于 0 和 1 之间的随机数;c_1,c_2 是学习因子,一般取为 2;ω 是惯性权重。

但基本的 PSO 算法存在收敛缓慢、后期振荡等缺陷,因此在基本 PSO 的基础上,引入动量项[12],构造 MPSO 算法,从而有效提高算法的收敛速度,同时起到部分避免算法后期振荡的作用。其改进主要是在速度更新公式中的改进,令 $\Delta v_i(k) = c_1\cdot rand_1(k)\cdot(p_{\text{ibest}} - x_i(k)) + c_2\cdot rand_2(k)\cdot(g_{\text{best}} - x_i(k))$,则更新后的速度公式如式(4)所示。

$$v_i(k+1) = \omega\cdot v_i(k) + \Delta v_i(k) + \alpha\Delta v_i(k-1) \tag{4}$$

此时粒子速度的修正量由两项组成,第一项是基本粒子群算法的速度修正量;第二项为动量项,其与微粒的历史修正量线性相关,α 为动量因子常数,$0 \leqslant |\alpha| < 1$,α 可取正也可取负,一般取为正数。当 $\Delta v_i(k)$ 与前次符号相同时,在稳定调节时能增加 $v_i(k+1)$ 的调节速度,加快算法的进化速度;当 $\Delta v_i(k)$ 与前次符号相反时,说明算法有一定的振荡,会使修正后的速度修正量减少,从而起到稳定算法的作用。

2.3 基于 MPSO 优化混合核 SVM 算法流程

2.3.1 SVM 性能指标的选取

交叉验证(Cross Validation,CV)是用来验证分类器性能的一种统计分析方法。常见的 CV 方法有 Hold-OutMethod、K-fold Cross Validation (K-CV)、Leave-One-Out Cross Validation。

鉴于 Hold-OutMethod 只是将原始数据分为两组,最终验证集分类准确率太依赖于原始数据的分组,而 Leave-One-Out Cross Validation 计算成本太高,样本数量较多时,实际操作很困难。因此,本文采用 K- CV 的方法,即将原始数据分为 K 组,将每个子集数据分别做一次验证集,其余的 $K-1$ 组子集作为训练集,这样得到 K 个模型,用这 K 个模型最终的验证集的分类准确率的平均值作为此 K- CV 下分类器的性能指标。

2.3.2 算法流程

MPSO 的适应度函数为 K- CV 下的分类准确率。

输入:粒子的维数,粒子的个数。

输出:最优的 SVM 参数(c,γ,λ)组合。

流程按如下步骤进行:

1)初始化 MPSO 中的各个参数,并确定 SVM 各个参数的解空间,如 $c_1 = 1.5$,$c_2 = 1.7$,$\omega = 0.9$ 等。

2)在解空间中随机初始化(c,γ,λ)的位置,在限定范围内随机初始化粒子的初始速度,并计算初始的适应度。

```
for i = 1: pso_option. maxgen
```

3)for j = 1: pso_option. sizepop

根据式(4)进行速度更新,式(3)进行种群更新:

```
4) cmd = ['-v', num2str(pso_option. v),
   '-c', num2str(pop(j,1)), '-g', num2str(pop(j,2)), '-a',
   num2str(pop(j,3))];
   fitness(j) = svmtrain(train_label, train, cmd);
   //计算适应度值
5)//更新每个粒子的新局部最优位置
   if fitness(j) > local_fitness(j)
     local_x(j,:) = pop(j,:);
     local_fitness(j) = fitness(j);
   end
6)//更新群体最优位置:
   if fitness(j) > global_fitness
     global_x = pop(j,:);
     global_fitness = fitness(j);
   end
   end
7)//确定最优解: fit_gen(i) = global_fitness; 当使得取最优解的
   //参数组合不止一组时,选取 C 值最小的那组
8)//输出最优解组合(c, γ, λ)
   end
```

此算法需要在 faruto 编写的 Libsvm-mat 加强工具箱[13] 的辅助下实现,由于此工具箱默认核函数为 RBF 类型,因此,在 4)中使用的混合核函数需要在工具箱中按式(1)进行修改或添加,文中采用的是在 RBF 核的基础上进行修改,由 return

exp(- gamma * (x_square[i] + x_square[j] - 2 * dot(x[i], x[j])))修改为 *return* (*exp*(- gamma * (x_square[i] + x_square[j] -2 * dot(x[i],x[j]))) * (1 - a) + powi(dot(x[i], x[j]) +1,1) * a),其中的 a 代表可调参数 λ。

3　实验与仿真

该实验采用的是 UCI 中的数据[14],在 Matlab 7.9.0 的环境下编程实现。实验中的参数设置如下:MPSO 中,c_1 =1.5,c_2 =1.7,最大进化代数设置为 200,种群数量为 20,ω =0.9,m = 0.30,SVM 中 C 的范围为 [0.1,100],γ 的范围为 [0.01,1 000],λ 的范围为 [0.50,0.99],对于 K-CV 中的 K 值,当数据量较小时取值为 5,数据量较大时取值为 9,默认值为 5。表 1 给出了针对 UCI 中 wine、iris、machine 数据,网格搜索算法与 MPSO 算法测试所得到的最优参数组合及 CV 下训练集的准确率及采用优化后参数组合进行实际测试的准确率,注意在实验中选用相同的训练集与测试集,K 取 5。

表 1　网格搜索算法与 MPSO 优化参数组合及分类数据相关信息对比

数据集	网格搜索算法		MPSO	
	(C,γ)	$A_{网}/A_{网}'$	(C,γ,λ)	A_M/A_M'
wine	(2.30,4)	0.988/0.988	(2.75,1.90,0.65)	0.978/0.994
iris	(12.13,2.29)	1.000/0.962	(14.32,2.07,0.88)	1.000/0.981
machine	(5.66,2)	0.918/0.806	(4.31,1.00,0.632)	0.901/0.852

表 1 中,$A_{网}$、$A_{网}'$ 分别代表网格搜索算法下得到的 CV 下的训练集准确率、实际测试集准确率;A_M、A_M' 分别代表 MPSO 算法下得到的 CV 下的训练集准确率、实际测试集准确率。

网格搜索算法实质上是一种穷举法,常用来在小范围内寻找最佳 SVM 参数组合。通过表 1 可知,本文提出的算法在最优参数组合上与网格搜索算法相当,并且在训练准确率相同或稍低于网格搜索算法的情况下,在实际测试中仍能得到比其更好的识别率,由此可知,所得的 SVM 泛化能力有所提高。

启发式算法不必遍历所有参数点,也能找到全局最优解,因而得以广泛应用。表 2 为本文的 MPSO 与基本 PSO 寻优参数组合及进化代数对比情况,此时取 $C \in [0.1,1\,000]$。图 2 ~ 3 分别给出了 balance-scale 数据在基本 PSO、MPSO 算法下训练的适应度曲线 Accuracy 图,图 4 为不同分类算法下的分类准确率对比图。

表 2　基本 PSO 与 MPSO 下的最优参数组合及进化代数对比

数据集	基本 PSO		MPSO	
	(C,γ)	进化代数	(C,γ,λ)	进化代数
wine	(12.25,4.350)	59	(7.16,1.790,0.586)	15
balance-scale	(29.90,2.620)	120	(25.50,1.160,0.780)	80
sonar	(19.30,0.362)	70	(15.86,0.769,0.980)	46
vehicle	(75.50,1.180)	158	(15.60,1.040,0.622)	97
diabetes	(47.87,0.625)	162	(10.90,1.070,0.820)	62

在第 1 章的 SVM 参数分析时,知道过高的 C 会导致过学习状态的发生。由表 2 知,MPSO 算法能找到优于 PSO 算法的最优参数组合,且进化速度明显快于 PSO。由图 2 ~ 3 对比得知,MPSO 算法能够部分缓解 PSO 存在的后期振荡现象。由图 4 可知 MPSO 算法下优化参数后得到 SVM 的分类准确率明显高于 PSO 下得到的 SVM 分类准确率,且高于最近邻算法(1NN)、C4.5 等其他常见分类算法的准确率。

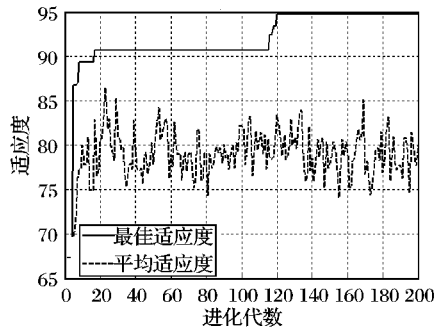

图 2　balance-scale 数据在基本 PSO 下训练的适应度曲线 Accuracy 图

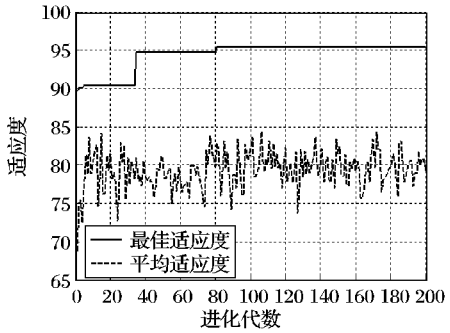

图 3　balance-scale 数据在 MPSO 下训练的适应度曲线 Accuracy 图

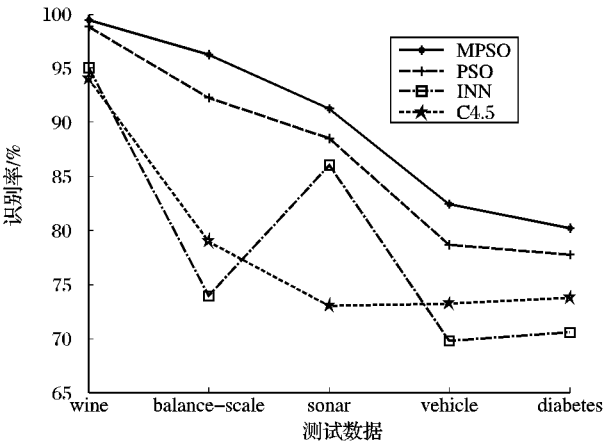

图 4　测试数据不同分类算法下的识别率对比图

4　结语

SVM 的最优参数组合选取问题一直是影响 SVM 性能的关键问题,目前关于其选取仍没有统一的方法。本文给出的带有动量项的粒子群算法,对具有混合核的 SVM 进行参数寻优,通过 UCI 数据进行测试,得出该算法能快速有效地提取出最优参数组合,所得的 SVM 泛化能力上优于 RBF 核的网格搜索算法,进化速度快于 PSO,测试所得分类准确率优于其他常见分类算法,为选取最优 SVM 参数组合提供了一种新方法。

参考文献:

[1]　边肇祺, 张学工. 模式识别[M]. 2 版. 北京: 清华大学出版社, 2000: 296 - 304.

[2]　刘松. 基于 SVM 信息融合的图像识别与并行实现[J]. 计算机工程与应用, 2009, 45(33): 168 - 182.

[3]　MELGANI F, BAZI Y. Classigication of electroca-rdiogram signals with support vector machines and particle swarm optimization[J]. Information Technology in Biomedicine, 2008, 12(5): 667 - 677.

[4]　LATRY CH, PANEM C, DEJEAN P. Cloud detecti - on with SVM technique[C]// 2007 IEEE Iternational Geoscience and Remote Sensing Symposium. Washington, DC: IEEE Computer Society, 2007: 448 - 451.

(下转第 516 页)

次数与冲突数的关系方面进一步讨论,如图2~3所示。

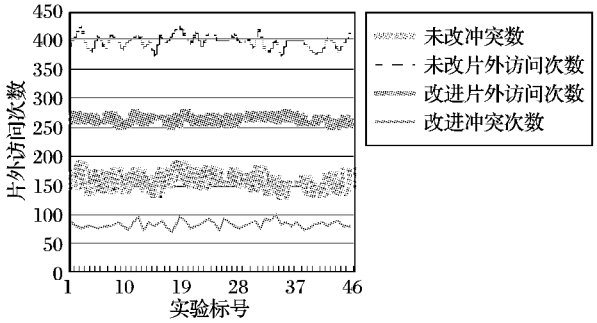

图1 片外访问次数与冲突数对比

由图2可知,随着冲突次数的增加,片外访问次数也急剧增加,访问次数为370~430,即每一个键值的片外访问次数的波动范围约为(1.53,1.77)。

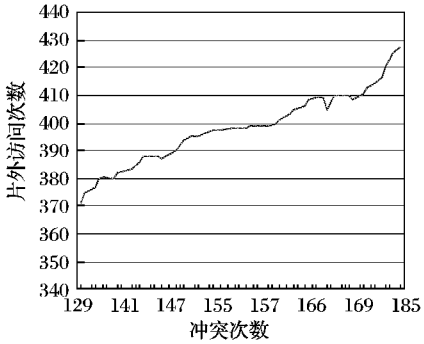

图2 原始孔雀哈希算法关系

由图3可知,随着冲突次数的增加,片外访问次数增加并不急剧,而是出现了波动,这种波动仍处于访问次数250~280,即每一个键值的片外访问次数的波动范围为(1.03,1.15)。通过两图对比,说明了改进哈希算法较原始孔雀哈希算法稳定性更强。

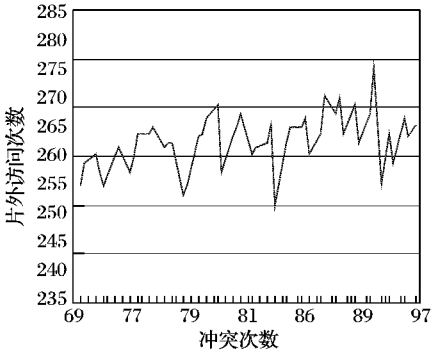

图3 改进哈希算法关系

3.2 空间性能仿真结果

由于实验中关键字大小固定,故未将其加入内存开销的计算。实际上,哈希表的内存开销主要是其他数据结构在表中应用带来的额外开销,如表1所示。

表1 不同表长的内存开销对比

最大表长	表的个数	原始算法内存开销/B	改进算法内存开销/B
512	4	450	304
1024	4	900	730
2048	4	1800	1328
4096	4	3600	2640

从表1可知,原始孔雀哈希算法的内存开销在各种表长情况下,都会比改进哈希算法的内存开销要大一些,改进后的内存开销平均节省了约27.7%。

3.3 实验分析

通过时间性能仿真可以得出改进后的哈希表在片外访问次数和冲突次数方面都有很大的优化。在空间性能方面,实验也证明改进哈希算法的内存开销确实小于原始孔雀哈希算法的开销。

4 结语

改进哈希表的目标在于能够有好的决策进行高速查找及改善最坏情况。在孔雀哈希的基础上,通过引入位图数组,提出了一种新的倒插入分段哈希表。改进后的哈希表较原始孔雀哈希算法对片外的访问次数减少了约34%,内存开销降低了约27.7%,降低了能耗。

参考文献:

[1] KUMAR S, TURNER J, CROWLEY P. Peacock hashing: Deterministic and updatable hashing for high performance networking [C]// The 27th Conference on Computer Communications. Washington, DC: IEEE Computer Society, 2008: 101-105.

[2] AHMADI M, WONG S. A memory-optimized bloom filter using an additional hashing function[C]// IEEE Global Telecommunications Conference. Washington, DC: IEEE Computer Society, 2008: 1-5.

[3] KUMAR S, CROWLEY P. Segmented hash: An efficient hash table implementation for high performance networking subsystems [C] // Proceedings of the 2005 ACM Symposium on Architecture for Networking and Communications Systems. New York: Springer, 2005: 91-103.

[4] DHARMAPURIKAR S, KRISHNAMURTHY P. Deep packet inspection using parallel bloom filters[J]. IEEE Micro, 2004, 24(1): 52-61.

[5] 潘登,张大方,谢鲲,等.一种基于折半层次搜索的包分类算法[J].计算机应用, 2009, 29(2): 500-506.

(上接第503页)

[5] 常玉清,王福利,王小刚,等.基于支持向量机的生物发酵过程软测量建模[J].东北大学学报:自然科学版, 2005, 26(11): 1025-1028.

[6] 张拥华,曾凡仔.基于混合核支持向量机的金融时间序列分析[J].计算机工程与应用, 2008, 44(19): 220-222.

[7] 业巧林,业宁,张训华.基于极分解下的混合核函数及改进[J].模式识别与人工智能, 2009(3): 366-373.

[8] 邓乃扬,田英杰.支持向量机:理论、算法与拓展[M].北京:科学出版社, 2009: 81-111.

[9] 陈林,潘丰.基于量子PSO的SVM参数选择及其应用[J].自动化与仪表, 2009, 24(1): 5-8.

[10] 张芬,陶亮,孙艳.基于混合核函数的SVM及其应用[J].计算机技术与发展, 2006, 16(2): 176-178.

[11] SALLEH S M, TOKHI M O, JULAI S, *et al*. PSO-based parametric modelling of a thin plate structure[C]// Proceedings of the 2009 Third UKSim European Symposium on Computer Modeling and Simulation. Washington, DC: IEEE Computer Society, 2009: 43-48.

[12] 黄福员.一种改进的动量粒子群算法及实验分析[J].计算机应用与软件, 2009, 26(10): 57-59.

[13] faruto编写的Libsvm加强工具箱[EB/OL]. [2010-02-02]. http://www.ilovematlab.cn/viewthread.php?tid=65333&extra=&page=1.

[14] UCI中的数据来源[EB/OL]. [2010-07-02]. http://lamda.nju.edu.cn/yuy/files/download/UCI_arff.zip or http://archive.ics.uci.edu/ml/.