维普资讯 http://www.cqvip.com

基于二进制 PSO 算法的特征选择及 SVM 参数同步优化*)

任江涛　赵少东　许盛灿　印　鉴
(中山大学计算机科学系　广州 510275)

摘　要　特征选择及分类器参数优化是提高分类器性能的两个重要方面，传统上这两个问题是分开解决的。近年来，随着进化优化计算技术在模式识别领域的广泛应用，编码上的灵活性使得特征选择及参数的同步优化成为一种可能和趋势。为了解决此问题，本文研究采用二进制 PSO 算法同步进行特征选择及 SVM 参数的同步优化，提出了一种 PSO-SVM 算法。实验表明，该方法可有效地找出合适的特征子集及 SVM 参数，并取得较好的分类效果；且与文[4]所提出的 GA-SVM 算法相比具有特征精简幅度较大、运行效率较高等优点。

关键词　特征选择，支持向量机，同步优化，粒子群算法

Simultaneous Feature Selection and SVM Parameters Optimization Algorithm Based on Binary PSO

REN Jiang-Tao　ZHAO Shao-Dong　XU Shen-Chan　YIN Jian
(Department of Computer Science, Zhongshan University, Guangzhou 510275)

Abstract　Feature selection and classifier parameter optimization are two important aspects for improving classifier performance and are solved separately traditionally. Recently, with the wide applications of evolutionary computation in pattern recognition area, simultaneous feature selection and parameter optimization become possible and tendency. To solve the problem, we propose a simultaneous feature selection and SVM parameter optimization algorithm based on binary PSO algorithm called PSO-SVM. The experiments show that the algorithm can efficiently find the suitable feature subsets and SVM parameters, which result in good classification performance. Compared with GA-SVM[4], PSO-SVM can get a more compact feature subset and run more efficiently.

Keywords　Feature selection, SVM, Simultaneous optimization, PSO

1　引言

分类问题是模式识别领域的核心问题，主要包括分类器模型的选择、特征选择及分类器参数优化等子问题。近年来，随着模式识别研究的深入，研究对象越来越复杂，对象的特征维数越来越高。大量高维数据对象的特征空间中含有许多冗余特征甚至噪声特征，这些特征一方面可能降低分类或聚类的精度，另一方面会大大增加学习及训练的时间及空间复杂度。因此，在面对高维数据进行分类或聚类时，通常需要运用特征选择算法找到具有较好可分性的特征子空间，从而实现降维，降低机器学习的时间及空间复杂度[1~3]。因此，模式识别领域研究出了大量的特征选择算法。另外，合适的参数设置对于提高分类器模型的分类精度非常重要，错误的参数设置往往会大幅降低分类器的性能，因此分类器参数优化也是模式识别领域重要的研究方向之一。

传统上，特征选择与分类器参数优化问题一般是分别研究的，但近年来出现了一种趋势，将这两个问题融合起来同步研究。文[4]提出了一种应用遗传算法(GA)同步进行特征选择及 SVM 参数优化的方法。文[5]提出了一种应用遗传规划(GP)同步进行特征选择及分类器参数优化的方法，都取得了不错的效果。

PSO 算法是一种新兴的优化技术，其思想来源于人工生命及演化计算理论。PSO 通过粒子追随自己找到的最优解核整个粒子群的最优解来进行优化[6,7]。近年来，粒子群算法(PSO)在模式识别与数据挖掘领域得到了许多应用，如二进制 PSO 算法用于特征选择[8,9]，PSO 算法还被应用于解决分类问题[10]。

针对特征选择及 SVM 参数同步优化问题，本文提出了一种基于二进制 PSO 算法同步进行特征选择及 SVM 参数同步优化的 PSO-SVM 算法。论文的第 2 部分首先介绍了 PSO 算法的基本原理，第 3 部分介绍了 SVM 算法，第 4 部分讨论了所提出的基于 PSO 同步进行特征选择及 SVM 参数优化的 PSO-SVM 算法，第 5 部分给出了实验研究结果，最后是对本文的总结。

2　PSO 算法

粒子群优化算法(PSO)是一种进化计算技术，源于对鸟群捕食的行为模拟，由 Eberhart 和 Kennedy 提出。系统初始化为一组随机解，通过迭代搜寻最优值。但是并不同于遗传算法使用的交叉以及变异，而是粒子在解空间追随当前最优的粒子进行搜索。同遗传算法相比，PSO 的优势在于简单易实现并且没有许多参数需要调整。目前已经广泛应用于函数优化，神经网络训练，模糊系统控制以及其他遗传算法的应用领域。

PSO 中，将每个粒子也就是优化问题的解看作是搜索空间中的一点。所有的粒子都有一个表示当前在解空间中位置的属性 $X_i=(x_{i1},x_{i2},\cdots,x_{iD})$，并由评价函数计算其适应度，每个粒子还有一个速度 $V_i=(v_{i1},v_{i2},\cdots,v_{iD})$ 决定它们运动的方向和距离。粒子之间通过共享当前最优粒子的信息，在解空间中搜索。

首先粒子群初始化为一群随机粒子(随机解)，然后通过迭代来寻找最优解。在每一轮的迭代中，粒子通过速度更新

*)本文研究得到国家自然科学基金资助(60573097)、广东省自然科学基金资助(04300462、05200302)。任江涛　博士，讲师，主要研究方向：数据挖掘与知识发现、生物信息学、商务智能。

维普资讯 http://www.cqvip.com

当前位置，并通过适应值函数计算出其适应值，然后粒子根据以下的速度更新公式进行计算，更新其当前速度。

$$v_{id}=v_{id}+c_1\times rand\times(p_{id}-x_{id})+c_2\times rand\times(p_{gd}-x_{id}) \tag{1}$$

$$x_{id}=x_{id}+v_{id} \tag{2}$$

式(1)是粒子的速度更新公式，$P_i=(p_{i1},p_{i2},\cdots,p_{iD})$表示粒子 i 的局部最优值，即粒子 i 目前为止在搜索空间中到过的最佳点，$P_g=(p_{g1},p_{g2},\cdots,p_{gD})$表示整个粒子群的全局最优值，即整个粒子群到目前为止在搜索空间中到过的最佳点。c_1 和 c_2 是两个正的常数，称为学习因子。$rand$ 表示 0 到 1 之间的随机数。公式(2)是粒子的位置更新公式。

为了适应 PSO 算法在离散问题中的应用，Kennedy 和 Eberhart 于 1997 年提出了二进制 PSO 算法。在二进制 PSO 算法中，每个粒子被编码为一个二进制向量。在二进制粒子中，速度定义了粒子的每个位置赋值为 1 的概率，因此要通过转换函数将速度转换到区间[0.0，1.0]。本研究中采用 sigmoid 函数。二进制的粒子更新公式如下：

$$p_{ij}=\begin{cases}1, & rand\ ()\leqslant S(v_{ij})\\ 0, & \text{otherwise}\end{cases} \tag{3}$$

为了获得更好的优化效率和效果，本研究对 PSO 算法进行了一些改进，在每轮迭代中将适应度最差的 10%粒子进行变异。实验证明这 10%的粒子浪费计算资源，而进行变异后粒子群能够更快地找到最优值，并且避免了容易聚集于局部最优的情况。

3 支持向量机

传统的统计模式识别方法都是在样本数目足够多的前提下进行研究，所提出的方法只有在样本数趋于无穷大时其性能才有理论上的保证。而实际上，样本数目通常是有限的。而近年来迅速发展起来的统计学习理论是一种专门研究小样本学习的模式识别理论，为研究有限样本情况下的统计模式识别和更广泛的机器学习问题建立了一个较好的理论框架，同时也发展了一种新的模式识别方法——支持向量机，能够较好地解决小样本学习问题。

支持向量机在解决小样本、非线性及高维模式识别问题中表现出许多特有的优势，决定使用它进行学习和分类。下面对支持向量机进行模式学习和分类的原理进行简要介绍[11]。

假设输入模式集 D 有 M 个 n 维样本，分为两类，如果样本属于第一类，标记为 1，如果属于第二类，则标记为 −1，数学定义如下：

$$D=\left\{\begin{matrix}(x_i,y_i)\mid i\in\{1,2,\cdots,M\},\\ x_i\in R^n,y_i\in\{+1,-1\}\end{matrix}\right\} \tag{4}$$

假定输入模式集 D 中的样本可以被下式所描述的一个超平面正确分类，

$$(w\cdot x)-b=0 \tag{5}$$

这个问题可表达为，

$$\begin{matrix}(w\cdot x_i)-b\geqslant 1 & \text{if } y_i=1\\ (w\cdot x_i)-b\leqslant -1 & \text{if } y_i=-1\end{matrix} \tag{6}$$

上式可以简化为

$$y_i[(w\cdot x_i)-b]\geqslant 1 \quad i\in\{1,2,\cdots,M\} \tag{7}$$

可以很容易得出结论，求解满足条件(4)的最优超平面问题可以表达为

$$Minimize\ \Phi(w)=\|w\|^2 \tag{8}$$

该问题还可转化为如下形式，

$$f(x)=sign(\sum_k a_k y_k K(x_k,x)+b) \tag{9}$$

其中，$K(x,y)$为核函数，b 是根据训练样本确定的阈值，a_k 由二次规划确定。在支持向量机中，其决策面的特性是由核函数确定的，而支持向量机正是通过决策面来实现样本分类的。核函数有很多种，如线性核函数、多项式核函数、径向基核函数等。

4 PSO−SVM 特征选择及参数优化同步算法

下面从粒子设计、适应度函数及算法流程等方面分别进行介绍。

4.1 粒子设计

粒子用一个二进制位串表示，其结构应该根据所采用的核函数来确定。本研究采用 RBF 函数作为 SVM 的核函数，RBF 核函数的定义如式(10)所示。

$$K(x,y)=\exp(-\gamma\|x-y\|^2),\gamma>0 \tag{10}$$

针对采用 RBF 核函数的 SVM，需要优化的参数有两个：核参数 $\gamma>0$ 和惩罚参数(penalty parameter)$C>0$，另外，还要同步进行特征选择。因此，粒子应该包含三个部分，即参数 C、参数 γ 和特征掩码，如图 1 所示。粒子的前 n_C 位代表参数 C，中间 n_γ 位表示参数 γ，最后 n_F 位代表特征掩码。n_C 和 n_γ 根据精度要求来确定，n_F 由数据集的特征数决定。

p_C^1	$\cdots p_C^i\cdots$	$p_C^{n_C}$	p_γ^1	$\cdots p_\gamma^i\cdots$	$p_\gamma^{n_\gamma}$	p_F^1	$\cdots p_F^i\cdots$	$p_F^{n_F}$

图 1 粒子结构示意图

粒子中代表参数 C 和 γ 的位串可以通过式(11)转换为十进制表示的实际参数。式(11)中 p 表示参数的十进制值，$\min_p$ 表示参数的最小值，$\max_p$ 表示参数的最大值，l 表示参数的二进制位串长度(对应于参数 C 和 γ 分别为 n_C 和 n_γ)，d 表示参数的二进制位串的十进制值。参数的精度(由代表该参数的位串的长度 l 决定)和范围(由参数的最小值和最大值确定)都可以根据需要设定。对于特征掩码部分，每一个二进制位对应特征集合中的一个特征，该位为 1 则表示对应的特征入选特征子集，而该位为 0 则表示对应的特征不在选出的特征子集中。

$$p=\min_p+\frac{\max_p-\min_p}{2^l-1}\times d \tag{11}$$

4.2 适应度函数

由于该算法目标为同步优化特征子集 SVM 参数以提高 SVM 算法的分类正确率，同时尽可能降低所选择的特征数，即一个粒子能够使分类器产生的分类精度越高，同时选出的特征数目越少，它的适应值就应该越高。因此，本研究采用如式(12)所示的粒子适应度函数定义。

$$fitness=w_a\times svm_accuracy+w_f\times(\sum_{i=1}^{n_F}f_i)^{-1} \tag{12}$$

其中，w_a 表示分类精度的权重；svm_accuracy 表示 SVM 的分类精度；w_f 表示特征数目倒数的权重；f_i 表示第 i 个特征在特征掩码中的对应位，1 表示特征被选中，0 表示未选中；w_a 和 w_f 根据实际需要确定。

4.3 算法流程

算法 1：$PSOSVM(D,c_1,c_2,PN,\min_c,\max_c,\min_\gamma,$

维普资讯 http://www.cqvip.com

$\max_\gamma, n_c, n_r, n_f, w_a, w_f, N$)

输入：数据集 D，学习因子 c_1、c_2，粒子数目 PN，参数 C、γ 的最小最大值 $\min_c, \max_c, \min_\gamma, \max_\gamma$ 及二进制位串长度 n_C 和 n_γ，特征二进制位串长度 n_f、权重参数 w_a, w_f、最大迭代次数 N

输出：优化的特征子集、SVM 参数 C、γ 及相应的分类精度

步骤：

1)初始化粒子群；

2)根据粒子编码方案及式(11)，将每个粒子的二进制表达转化为 SVM 的参数 C、γ 及所选择的特征子集，然后调用 LIBSVM 算法进行学习和训练，测试并记录分类精度，根据式(12)计算粒子适应度；

3)根据粒子适应度，更新 P_i 及 P_g；

4)根据式(1)，更新粒子的速度；

5)根据式(3)，更新粒子的二进制位；

6)判断是否达到最大迭代次数 N，若达到则输出当前最优的特征子集、参数 C、γ 及分类精度，否则返回步骤 2)继续迭代。

5 实验研究

5.1 实验数据及实验环境

为了评估算法的有效性，采用多个 UCI 数据集进行实验研究，数据集的名称及相关信息如表 1 所示。本研究的所有实验在采用 Intel 奔腾 4 3.0G CPU、内存为 512M 的计算机上进行，算法用 Matlab 编程实现，SVM 采用 LIBSVM[12]。

表 1 实验中用到的 UCI 数据集

Names	No. of classes	No. of instances	Nominal features	Numeric features	Total features
Pima-Indian diabetes	2	768	0	8	8
Heart disease	227	0	7	6	13
Ionosphere	2	351	0	34	34
Sonar	2	208	0	60	60
Iris	3	150	0	4	4
Vehicle	4	990	0	18	18

5.2 实验评价方法及指标

实验采用交叉验证法(Cross Validation)进行评价。数据集被随机分成 k 个子集，PSO-SVM 算法在每个数据集上运行 k 次，每次取一个子集作为测试集，其余 $k-1$ 个子集合并为训练集，然后取 k 次实验结果的均值作为该数据集的分类结果，本研究中取 $k=10$。

另外，为了与 PSO-SVM 算法进行对比，还采用既未进行特征选择，也未进行参数优化的 SVM 进行训练和测试。同时，也采用文[1]所提出的 GA-SVM 的实验结果进行对比。上述两种方法分别简记为 SVM 和 GA-SVM，其实验也采用交叉验证法进行评价。

下面描述粒子适应度函数中分类精度指标 svm_accuracy 的定义。

对于二类数据集(即数据集中的样本只有两类，分别称为正例和反例)，本文先定义三个命中率：正命中率、反命中率和总体命中率，见式(13)。

$$\begin{cases} r_p = num_p / Num_p \\ r_n = num_n / Num_n \\ r_a = (num_p + num_n)/(Num_p + Num_n) \end{cases} \tag{13}$$

其中 r_p 为正命中率；r_n 为反命中率；r_a 为总体命中率；num_p 表示被分类器正确分类的正例样本数目；num_n 表示被分类器正确分类的反例样本数目；Num_p 表示数据集中正例样本总数；Num_n 表示数据集中反例样本总数。

对于类别数为 2 的数据集，采用 $r_p \times r_n$ 而不是 r_a 来度量分类精度。因为一个好的分类器，不但总体的命中率要高，而且表现要均匀，即对于不同的类，其分类精度不能相差很大。这一点对二类数据集尤为重要。所以，采用 $r_p \times r_n$ 代表 svm_accuracy 更加科学。而对于多类数据集，则用总体命中率 r_a 代表其 svm_accuracy。

5.3 实验参数设置

实验中算法的参数设置如下：$c_1 = c_2 = 2$，粒子速度的初始值范围是$[-5, 5]$。粒子数目随着数据集维数的不同而不同，具体见表 2。这是因为当数据集的维数较高时，搜索空间大，所以需要较多的粒子数目；反之，则可以采用较少的粒子数目。当迭代次数达到 300 或者连续 100 次迭代全局最佳适应值没有改变时，PSO 算法停止。实验中用到的其它参数如下：$w_a = 0.8, w_f = 0.2, n_R = n_S = 20$。$C \in [2^{-5}, 500], \gamma \in [2^{-15}, 1]$。

表 2 各个数据集采用的粒子数目

Names	Total features	Particle No.
Pima-Indian diabetes	8	100
Heart disease	13	100
Ionosphere	34	200
Sonar	60	300
Iris	4	50
Vehicle	18	150

5.4 实验结果

表 3 所示是 SVM 算法、GA-SVM 算法和 PSO-SVM 算法在各个数据集上的实验结果，分类精度，其中平均值是算法在某一数据集上运行 10 次所得结果的平均值。

表 3 SVM、PSO-SVM 和 GA-SVM 三种方法在实验数据集上的性能比较

Datasets	SVM			PSO-SVM			GA-SVM		
	Average positive hit rate	Average negative hit rate	Average overall hit rate	Average positive hit rate	Average negative hit rate	Average overall hit rate	Average positive hit rate	Average negative hit rate	Average overall hit rate
Diabetes	0.7318	0.7440	0.7396	0.7433	0.7920	0.7748	0.7835	0.8704	0.8150
Heart disease	0.8000	0.8067	0.8037	0.9417	0.9267	0.9333	0.9447	0.9511	0.9480
Ionosphere	0.9532	0.9377	0.9431	0.9923	1.0000	0.9971	0.9963	0.9876	0.9856
Sonar	0.9477	0.6722	0.8187	1.0000	1.0000	1.0000	0.9863	0.9842	0.9800
Iris	NA	NA	0.9667	NA	NA	0.9800	NA	NA	1.0000
Vehicle	NA	NA	0.7634	NA	NA	0.9256	NA	NA	0.8406

维普资讯 http://www.cqvip.com

表4是GA-SVM方法和PSO-SVM方法在特征选择方面的比较，其中给出了两种算法在相同数据集上运行10次所选特征数目的平均值。第一列括号中的数字是该数据集原有的特征数目。

表4 PSO-SVM方法和GA-SVM方法在实验数据集上的平均特征选择数目比较

Datasets	PSO-SVM Average number of selected features	GA-SVM Average number of selected features
Diabetes(8)	1.4	3.7
Heart disease(13)	4.1	5.4
Ionosphere(34)	3.9	6.0
Sonar(60)	5.8	15.0
Iris(4)	1.0	1.0
Vehicle(18)	9.7	9.2

5.5 实验结果分析

下面将根据实验结果，从分类精度、特征选择能力和算法运行效率等三个方面对上述三种方法进行比较分析。

从表3可以看出，PSO-SVM方法的分类精度比起SVM方法有了很大的提高。在每个实验数据集上，前者的分类精度都高于后者。比如对于二类数据集Heart，SVM方法的正命中率、负命中率和总体命中率分别只有0.8000、0.8067和0.8037；而PSO-SVM方法的三个命中率分别达到0.9417、0.9267和0.9333，比SVM方法分别提高了17.71%、14.88%和16.13%。对于多类数据集Vehicle，SVM方法的平均命中率为0.7634；而PSO-SVM方法达到0.9256，比前者提高了21.25%。此外，由PSO-SVM方法得到的分类器更为均衡，也就是说对于不同的类，其命中率相差不大。这一点在二类数据集Sonar上表现最为明显。在该数据集上，SVM方法的负命中率和正命中率分别为0.6722和0.9477，相差40.98%；而PSO-SVM方法的正负命中率都是1.0000，相差0.00%。上述结果充分说明，在运用SVM分类器时，特征选择和参数优化的重要性，也证明了本文提出的PSO-SVM方法的有效性。对于PSO-SVM和GA-SVM而言，从总体上看，无论是分类器精度还是均衡性，两种方法都非常接近，都是较为有效的方法。

从表4可以看出，平均来说，PSO-SVM方法在大多数数据集上所选出的特征数目要少于GA-SVM方法，而且这种差别比较明显。以特征数目最多的Ionosphere(特征数目为34)和Sonar(特征数目为60)数据集为例，PSO-SVM选出的特征数目的均值只有3.9和5.8，而GA-SVM方法选出的特征数目均值分别为6.0和15.0，分别是前者的1.54倍和2.57倍。考虑到两种方法的分类精度很接近，可以得出结论说PSO-SVM方法比GA-SVM方法具有更好的特征选择效果。

在实际应用中，分类器有训练和使用两个阶段。在训练上，SVM方法是最快的，因为不像其它两种方法需要使用优化算法(PSO或GA)进行特征选择和参数优化。但是，训练阶段特征选择的缺失，将降低使用阶段SVM的分类精度；且过多的特征数目将使得SVM运行变慢。而一个分类器的使用阶段要长于训练阶段，所以从长远看，SVM方法的效率较差。从训练阶段看，在相同的数据集上达到类似的分类精度，GA-SVM方法需要进行600次迭代而PSO-SVM方法只需进行300次迭代。而且，GA-SVM方法对于每个数据集都采用了500个染色体，而PSO-SVM方法则根据数据集的特征数选择粒子数目，最多300，最少50。上述结果表明，PSO-SVM算法的运行效率优于GA-SVM算法。

综上所述，从分类精度、特征选择能力及算法运行效率等方面综合比较分析，PSO-SVM算法要优于另外两种算法。

结论 本研究受到文[4]所提出的GA-SVM算法启发，提出了一种基于PSO同步进行特征选择及SVM参数同步优化的PSO-SVM算法。实验表明，该方法可有效地找出合适的特征子集及SVM参数，并取得较好的分类效果；且与GA-SVM算法相比具有特征精简幅度较大、运行效率较高等优点。

参考文献

1 John G H, Kohavi R, Pfleger K. Irrelevant Features and the Subset Selection Problem [A]. In: Proceedings of the Eleventh International Conference on Machine Learning [C], New Brunswick, NJ, USA, Morgan Kaufmann, 1994. 121～129

2 Kohavi R, John G H. Wrappers for feature subset selection [J]. Artificial Intelligence, 1997, 97(1-2): 273～324

3 Liu Huan, Yu Lei. Toward Integrating Feature Selection Algorithms for Classification and Clustering [J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(5): 491～502

4 Huang C-L, Wang C-J. A GA-based feature selection and parameters optimization for support vector machines. Expert Systems with Applications, 2006, 31(2): 231～240

5 Muni D P, Pal N R, Das J. Genetic Programming for Simultaneous Feature Selection and Classifier Design. IEEE Transactions on Systems, Man, and Cybernetics-Part B, 2006, 36(1): 106～117

6 Kennedy J, Eberhart R C. Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ, 1995. 1942～1948

7 Kennedy J, Eberhart R C. A discrete binary version of the particle swarm algorithm. In: Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics 1997, Piscataway, NJ, 1997. 4104～4109

8 Liu Yu, Qin Zheng, Xu Zenglin, He Xingshi. Feature Selection with Particle Swarms, Computational and Information Science. In: First International Symposium, CIS 2004, Shanghai, China, December Proceedings, LNCS3314, 2004. 425～430

9 Wang Ling, Yu Jinshou. Fault Feature Selection Based on Modified Binary PSO with Mutation and Its Application in Chemical Process Fault Diagnosis, ICNC 2005, LNCS3612. 832～840

10 De Falco I, Della Cioppa A, Tarantino E. Facing classification problems with Particle Swarm Optimization. Applied Soft Computing, In Press, Available online 3 February 2006

11 Cristianini N, Shawe-Taylor J. An Introduction To Support Vector Machines And Other Kernel-Based Learning Methods. Cambridge University Press, 2000

12 Chang C-C, Lin C-J. LIBSVM: a library for support vector machines