文章编号：1001－9081(2007)11－2812－02

# 基于 Fisher 准则和特征聚类的特征选择

王 飒，郑 链
(北京理工大学 宇航科学技术学院，北京 100081)
(sharonws@bit.edu.cn)

摘 要：特征选择是机器学习和模式识别等领域的重要问题之一。针对高维数据，提出了一种基于 Fisher 准则和特征聚类的特征选择方法。首先基于 Fisher 准则，预选出鉴别性能较强的特征子集，然后在预选所得到的特征子集上对特征进行分层聚类，从而最终达到去除不相关和冗余特征的目的。实验结果表明该方法是一种有效的特征选择方法。

关键词：特征选择；Fisher 准则；特征聚类
中图分类号：TP181　　文献标识码：A

# Feature selection method based on Fisher criterion and feature clustering

WANG Sa, ZHENG Lian
(School of Aerospace Science and Engineering, Beijing Institute of Technology, Beijing 100081, China)

Abstract: Feature selection is one of the important issues in the machine learning and pattern recognition. For high-dimensional data, the new feature selection method based on Fisher criterion and feature clustering was proposed. Firstly, the features that have more discrimination information based on Fisher criterion were selected. Then hierarchical clustering in the pre-selected subset was adopted. Finally, the irrelevant and redundant features were removed. The experimental results show that the proposed algorithm is an effective method for feature selection.

Key words: feature selection; Fisher criterion; feature clustering

## 0 引言

特征选择是指从原始特征中挑选出一些最有效的特征以降低特征空间的维数。大量高维数据中含有许多冗余特征甚至是噪声特征，这些特征的存在不仅会大大增加学习算法的训练时间和计算复杂度，而且可能降低分类的准确度。因此，在高维数据中合理地选择特征可以有效去除不相关和冗余的特征，从而提高学习算法的效率，减少计算复杂度[1,2]。

从理论上讲，最为可靠的最优特征子集的搜索方法是穷举法，但穷举法往往因为特征空间的维数较大而计算量过大，从而难以实现。根据特征选择准则是否依赖于学习算法，特征选择方法可以分为 Wrapper 和 Filter 两大类[3]。Filter 特征选择的评估标准直接由数据集求得，独立于学习算法，具有计算代价小，效率高等特点。Wrapper 特征选择可能会比 Filter 特征选择的降维效果好，但该算法因为计算代价大，所以效率较低[1,2]。

本研究从特征选择的目的，即去除不相关和冗余特征出发，对特征进行选择，从而得到一组次优的特征子集。去除不相关特征方面采用单个特征的 Fisher 比作为特征选择准则，去除噪声特征以及鉴别性能较差的特征；去除冗余特征方面，提出了一种基于特征之间的相关性系数的相似度度量，对特征进行分层聚类，最后从每一个聚类中选择一维特征作为最后的特征子集。

## 1 基于 Fisher 准则的特征选择

Fisher 准则是特征选择的有效方法之一，其主要思想是鉴别性能较强的特征表现为类内距离尽可能小，类间距离尽可能大[4]。这里采用单个特征的 Fisher 比作为准则，对特征进行排序，选出那些鉴别性能较强的特征，从而达到降维的目的并得到较优的识别性能。

定义数据集中共有 $n$ 个样本属于 $C$ 个类 $\omega_1, \omega_2, \cdots, \omega_C$，每一类分别包含 $n_i$ 个样本，$x^{(k)}$，$m_i^{(k)}$，$m^{(k)}$ 分别表示样本 $x$ 第 $k$ 类样本的均值，所有样本的均值在第 $k$ 维上的取值。单个特征的 Fisher 准则表示为：

$$J_{fisher}(k) = S_B^{(k)} / S_w^{(k)} \tag{1}$$

其中 $S_B^{(k)}$ 和 $S_w^{(k)}$ 分别表示该维特征在训练样本集上的类间方差和类内方差。

$$S_B^{(k)} = \sum_{i=1}^{C} \frac{n_i}{n} \left( m_i^{(k)} - m^{(k)} \right)^2 \tag{2}$$

$$S_w^{(k)} = \frac{1}{n} \sum_{i=1}^{C} \sum_{x \in \omega_i} \left( x^{(k)} - m_i^{(k)} \right)^2 \tag{3}$$

式(2)和(3)分别为第 $k$ 维特征的类内方差和类间方差的表达式。

$J_{fisher}$ 称为特征的 Fisher 比或 Fisher 判据，某维特征在训练样本集上的 Fisher 比越大说明该维特征的类别区分度越好，即包含越多的鉴别信息，而噪声特征的 $J_{fisher}$ 趋近于 0。

## 2 基于特征间相关性的特征聚类

### 2.1 特征间冗余度度量

本文采用相关系数来度量特征之间的冗余度：

$$\rho_{xy} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (y_i - \bar{y})^2}} \tag{4}$$

收稿日期：2007－05－11；修回日期：2007－07－12。　基金项目：国防预研基金资助项目(××××0202)。

作者简介：王飒(1981－)，女，河北定州人，博士研究生，主要研究方向：高维数据特征选择及分类方法；郑链(1942－)，男，江苏南京人，教授，博士生导师，主要研究方向：目标图像识别与智能信息处理技术，微机应用与仿真测试技术。

如式(4)所示, ρ值取值范围在 -1到 1之间, ρ的绝对值越大, 就表示变量 x, y之间的相关度越大, 也就是冗余度越大。ρ值为零, 说明两个变量相互独立。

### 2.2 特征聚类

聚类算法是非监督模式识别的一种重要方法, 它根据某种相似度度量, 对样本空间进行分组, 使组内数据之间彼此相似, 使组间数据相似距离较大, 从而实现自动分类[4]。

本研究将聚类算法应用到特征空间中, 即对特征进行聚类。聚类方法中常用的有分层聚类和 k均值聚类, 在 k均值聚类中需要指定 k值, 即最后的聚类个数。而在把聚类算法应用到特征空间时, 一般情况下我们不能确定有用特征的个数。所以在本研究中, 聚类方法采用分层聚类中的合并(自底而上)方法, 先使得每维特征各成一类, 然后通过合并相似度最大的两类, 来减少类别数目。当两个聚类 $\phi_i$, $\phi_j$中分别包含 m和 n个特征时, 这两个聚类的相似度度量定义为:

$$SM_{ij}=\frac{1}{mn}\sum_{x\in\phi_i}\sum_{y\in\phi_j}|\rho_{xy}| \tag{5}$$

分层聚类过程中每次将最相似的两个聚类, 即 $SM_{ij}$值最大的两个聚类聚为一类。聚类停止的条件是类别之间的相似度的最大值到达某个阈值。

## 3 基于特征间相关性的特征聚类

利用 Fisher准则可以删除无关的和鉴别性能较差的特征, 但 Fisher准则不能去除冗余的特征。为了克服这个缺点, 我们提出了一种基于 Fisher准则和特征聚类的组合式特征选择方法。该算法第一步采用 Fisher准则对初始特征集 S中的每一维特征进行评估, 滤掉 Fisher值较小的特征得到特征子集 $S_1$; 然后采用分层聚类的方法, 删除冗余特征。基于 Fisher准则和特征聚类的特征选择方法(Feature Selection based on Fisher Criterion and Feature Clustering, FSFCFC)具体描述如下:

算法　FSFCFC(Tr, Te, ρ, Ct)

输入　训练数据集 Tr, 测试数据集 Te, 聚类算法中的相似度阈值 Ct, Fisher准则中比值 ρ。

初始特征集 $S=\{F_i \mid i=1, 2, \cdots, D\}$, $S_1=S_2=NULL$

步骤:

1) 在训练数据集上计算每一维特征的 $J_{fisher}(F_i)$;

2) 计算所有特征 Fisher比的总和 $FisherSum=\sum_{F\in S}J_{fisher}(F_i)$;

3) 按照 Fisher比从大到小对 S中的特征进行排序, 按照排好的顺序将 S中的特征顺序选入到 $S_1$中, 直到 $\sum_{F\in S_1}J_{fisher}(F_i)>\rho*FisherSum$停止。

4) 对 $S_1$中的特征进行分层聚类, 聚类时类别之间的相似度度量采用式(5), 直到类别之间的相似度 $SM_{ij}$的最大值小于阈值 Ct停止聚类。

5) 对于特征聚类的聚类结果, 从每一个聚类中选择 Fisher比最大的那维特征加入到 $S_2$中, 最后 $S_2$中特征的个数就是分层聚类的类别数。

输出　特征子集 $S_1$, $S_2$

## 4 实验结果与分析

### 4.1 实验数据集

实验数据集中 USPS是 10个手写数字的数据集[5], 其他数据集均来自 UCI数据集[6]。其中数据集 waveform40数据是人造数据集, 是在 waveform21数据的基础上加入 19维噪声特征得到的。实验数据集如表 1所示。

### 4.2 实验方法

实验中用到两种分类器, 分别为最近邻分类器(1-Nearest Neighbor, 1NN)和支持向量机(Support Vector Machine, SVM), 其中支持向量机采用 4阶多项式核函数。每个数据集上, 先后运行基于 Fisher准则的特征选择和特征聚类, 记录下得到的特征子集的大小以及对应的测试集上的识别率。在用 Fisher准则对特征进行选择时, 参数 ρ设为 0.99; 在用特征之间的相似性对特征进行聚类时, 迭代到最相似的两组特征之间的 $SM_{ij}$值小于 0.8时终止, 并从每一聚类中选择一维 Fisher比最大的特征组成最后的特征选择结果。

表 1　实验数据集

| 数据编号 | 数据集名称 | 数据维数 | 训练/测试样本个数 | 类别数 |
|---|---|---|---|---|
| 1 | waveform40 | 40 | 1000/4000 | 2 |
| 2 | isolet | 617 | 6238/1559 | 26 |
| 3 | USPS | 256 | 7291/2007 | 10 |
| 4 | mfeat | 649 | 500/1500 | 10 |

### 4.3 实验结果及讨论

表 2列出了在各个数据集上运行基于 Fisher准则的特征选择后得到的特征子集 $S_1$以及通过特征聚类最后得到的特征子集 $S_2$的大小。表 3和 4分别给出了最近邻分类器和支持向量机的测试识别率, 为比较起见, 分别列出了在原始特征空间 S上以及在 $S_1$, $S_2$特征子集上的测试识别率。由表 2可以看出, 通过基于 Fisher准则的特征选择和特征聚类后使得特征空间的维数大大降低了; 而由表 3和 4可见, 在特征空间维数大大降低的情况下, 分类器的识别性能没有下降, 在有些数据集上反而有所提高。

表 2　特征选择前后特征维数

| 数据编号 | 原始特征空间 S | Fisher准则后 $S_1$ | 特征聚类后 $S_2$ |
|---|---|---|---|
| 1 | 40 | 19 | 19 |
| 2 | 617 | 571 | 287 |
| 3 | 256 | 234 | 173 |
| 4 | 649 | 594 | 362 |

表 3　特征选择前后测试识别率(%)(1NN)

| 数据编号 | 原始特征空间 S | Fisher准则后 $S_1$ | 特征聚类后 $S_2$ |
|---|---|---|---|
| 1 | 74.7 | 76.47 | 76.47 |
| 2 | 88.58 | 89.42 | 88.71 |
| 3 | 94.77 | 94.87 | 94.62 |
| 4 | 96.93 | 97.07 | 97.27 |

表 4　特征选择前后测试识别率(%)(SVM)

| 数据编号 | 原始特征空间 S | Fisher准则后 $S_1$ | 特征聚类后 $S_2$ |
|---|---|---|---|
| 1 | 84.72 | 86.33 | 86.33 |
| 2 | 96.73 | 96.79 | 96.28 |
| 3 | 96.01 | 95.86 | 95.96 |
| 4 | 98.27 | 98.33 | 98.33 |

另外可以发现大部分情况下分类器的识别率在 $S_1$特征子集上都较在原始特征空间 S上有一定程度的提高, 这说明

(下转第 2840页)

差为 1。这样根据三 σ 原则，每次获得的内存大小在 2 KB到 8 KB之间的概率为 99%，这是一个可以接受的范围。每次释放内存使用表中的哪一个内存则由一系列符合均匀分布的随机数决定。

系统的硬件配置如下：CPU主频 551 239 583 Hz，66 MHz系统总线。

4.2 检测结果

运行测试程序获得的测试结果如表 1所示。其中场景 1表示整个测试过程，包括全部的内存申请情况；场景 2是开始阶段，系统进行连续的内存申请，没有内存释放的情况；场景 3是内存申请与释放交替进行，存在内存碎片的情况。均值是在该场景中进行内存分配的平均耗时，方差表示每次申请时间对均值的偏离程度。

表 1 测试数据

| 场景 | 次数 | 最小值/μs | 最大值/μs | 均值/μs | 方差 |
|---|---|---|---|---|---|
| 场景 1 | 8160 | 0.422 | 12.153 | 0.681 | 0.307 |
| 场景 2 | 4082 | 0.422 | 2.809 | 0.647 | 0.114 |
| 场景 3 | 4078 | 0.451 | 12.153 | 0.715 | 0.416 |

在对场景 2中获得的数据进行统计后发现，申请时间小于 0.6μs的累积频数为 2.2%，申请时间在 0.68μs处的累积频数为 91.2%，申请时间大于 2μs的次数只有 2次，很明显在绝大多数的内存申请中耗时都在 0.68μs左右。图 2是场景 2的内存申请耗时的频数直方图，其中横坐标表示申请的时间，单位μs；纵坐标代表申请时间出现的频数。从图上可以清晰地看到，均值附近的申请时间出现频数非常高。

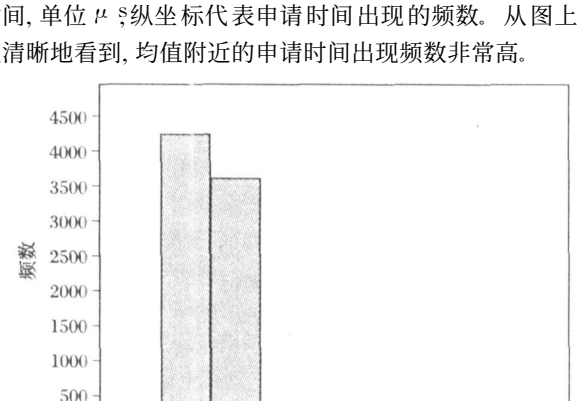


图 2 整个测试过程的内存申请耗时频数直方图

场景 3的数据统计结果为，申请时间在 0.66μs处的累积频数为 7.3%，申请时间在 0.816μs处的累积频数为 91.8%，而申请时间大于 26μs的有 24次，频数为 5.88‰，最大值达到 12.156μs。在场景 3中内存申请时间主要集中在 0.86μs附近的区域，平均值较之场景 2的有所增加，而且申请时间大于 26μs的频数明显增多。说明内存分配过程中遇到了状况。

图 2是整个测试过程的内存申请耗时的频数直方图。从图上可以看到有两个凸现的直方形，一个跨度在 0.45到 0.65，另一个跨度在 0.65到 0.85。这验证了操作系统的内存分配算法在前后两个场景中的表现有所不同。另一方面两个直方形紧密相连，说明操作系统在两个场景中的内存申请耗时的总体水平相差无几。

4.3 结果分析

通过测试数据的统计分析，可以得到如下结论：

1)测试场景的设置确实对 VxWorks的内存分配产生了影响；

2) VxWorks在系统启动或复位时进行内存分配时，内存分配算法的性能稳定，每次内存申请时间绝大部分落在均值附近，且幅度很小；

3)当 VxWorks处于内存碎片繁多的情况时，每次内存申请的时间仍然落在均值附近，但是均值较无碎片情况时有所增大，且偏离均值的程度也增大，但是内存分配性能仍然保持稳定。

参考文献：

[1] 陈育君，温彦军，陈琪. VxWorks程度开发实践[M]. 北京：人民邮电出版社，2004.

[2] WindRiver. Benchmark methodology report[R]. USA WindRiver Systems, Inc, 1997.

[3] QNX. QNX neutrino real-time OS: kernel benchmark methodology[Z]. USA QNX Software Systems Ltd, 2003.

[4] 袁萌棠. 概率论与数理统计[M]. 2版 北京 中国人民大学出版社，1995.

[5] SAMEK M. Practical statecharts in C/C++ quantum programming for embedded systems[M]. USA: CMP Books, 2002.

[6] KEUTZER K. Software environments for embedded systems[M]. USA: CMP Books, 2000.

[7] MAEBE J, RONSSE M, De BOSSCHERE K. Precise detection of memory leaks[C/OL]// Proceedings of the Second International Workshop on Dynamic Analysis [2007-05-01]. http://www.cs.virginia.edu/woda2004/papers/maebe.pdf.

[8] NETHERCOTE N, SEWARD J. Valgrind: a program supervision framework[C]// SOKOLSKY O, VISWANATHAN M. Proceedings of the Third Workshop on Runtime Verification. Electronic Notes in Theoretical Computer Science 89. Amsterdam: Elsevier, 2003.

(上接第 2813页)

删除掉噪声特征和鉴别性能较差的特征可以使学习算法的性能得到一定程度的提高。对于数据集 waveform40经过特征聚类后的特征空间 $S_2$相对于 $S_1$特征维数没有减少，这说明 $S_1$中的特征之间的冗余度较低。对于 isolet, USPS和 mfeat 3个数据集在经过特征聚类后，特征维数大幅度减少，说明在这三个数据集中特征之间的冗余度较大，而删除这些冗余特征，对学习算法的性能影响并不大。

参考文献：

[1] LANGLY P. Selection of relevant features in machine learning[C]// Proceedings of the AAAI Fall Symposium on Relevance. New Orleans, LA: AAAI Press, 1994: 140-144.

[2] JOHN G, KOHAVI R, PFLEGER K. Irrelevant feature and the subset selection problem[C]// Proceedings of the 11th International Conference on Machine Learning. New Brunswick, NJ, USA: Morgan Kaufmann Publishers, 1994: 121-129.

[3] KOHAVI R, JOHN G. Wrappers for feature subset selection[J]. Artificial Intelligence, 1997, 97(1/2): 273-324.

[4] DUDA R O, HART P E, STORK D G. Pattern classification[M]. 2nd ed. New York: John Wiley & Sons, 2001.

[5] LECUN Y, BOSER B, DENKER J S, et al. USPS database[DB/OL]. [2007-05-01]. http://www.kernel-machines.org/data.html.

[6] MERZ C J, MURPHY P M. UCI repository of machine learning databases[DB/OL]. [2007-05-01]. http://www.ics.uci.edu/~mlearn/MLRepository.html.