文章编号: 1001-0920 (2015) 10-1903-04 DOI: 10.13195/j.kzyjc.2014.1100

一种基于属性关系的特征选择算法

胡 静[1], 华 俊[1], 姜 羽[1], 宋铁成[1], 刘世栋[2], 郭经红[2]

(1. 东南大学 信息科学与工程学院，南京 210096；2. 国网智能电网研究院 信息通信研究所，南京 210003)

摘 要: 对于包含大量特征的数据集, 特征选择已成为一个研究热点, 能剔除无关和冗余特征, 将会有效改善分类准确性. 对此, 在分析已有文献的基础上, 提出一种基于属性关系的特征选择算法 (NCMIPV), 获取优化特征子集, 并在 UCI 数据集上对 NCMIPV 算法进行性能评估. 实验结果表明, 与原始特征子集相比, 该算法能有效降低特征空间维数, 运行时间也相对较短, 分类差错率可与其他算法相比, 在某些场合下性能明显优于其他算法.

关键词: 特征选择；属性关系；分类

中图分类号: TP391 **文献标志码:** A

A feature selection algorithm based on relationship between attributes

HU Jing[1], *HUA Jun*[1], *JIANG Yu*[1], *SONG Tie-cheng*[1], *LIU Shi-dong*[2], *GUO Jing-hong*[2]

(1. School of Information Science and Engineering，Southeast University，Nanjing 210096，China；2. Research Institute of Information Technology & Communication，State Grid Smart Grid Research Institute，Nanjing 210003，China. Correspondent：HU Jing，E-mail：louy@seu.edu.cn)

Abstract: Feature selection has become a heated research issue for datasets that contain large numbers of features and has the ability to remove irrelevant and redundant features and improve classification accuracy in an effective fashion. A feature selection algorithm based on relationship between attributes named NCMIPV is proposed to acquire the optimized feature subset based on the analysis of existing relevant literatures, and the performance of NCMIPV on UCI datasets is evaluated. Experiment results show that compared with original datasets, this algorithm tends to shrink the dimension of feature space effectively in a comparatively shorter length of time. Moreover, the misclassification rate appears to rival other algorithms. Overall performance of the proposed algorithm is obviously superior to its counterparts in certain situation.

Keywords: feature selection；attribute relationship；classification

0 引 言

随着获取数据的能力不断增强, 来自现实世界的数据维数也不断增加, 多元性与复杂性并生. 在现代信息社会中, 各式各样的数据正变得越来越常见, 囊括了多维度的特征属性. 一方面, 数据量庞大对于数据存储构成挑战; 另一方面, 对于数据的处理也存在相应困难, 计算复杂度大幅提升[1]. 有时, 数据分析需要经过若干步骤, 前面环节出现的难题若不能得到有效解决, 则将直接导致后面算法无法运行, 进而导致整个流程受阻. 虽然特征数量巨大, 但是特征之间的冗余不可避免[2], 这严重影响了分类效果.

特征选择是从一组给定的特征中选择出一些有效特征从而降低特征空间大小的过程[3], 广泛应用于数据挖掘、机器学习及模式识别等领域, 通常作为分类问题的预处理步骤. 通过特征选择, 能够剔除无关和冗余特征, 降低计算代价. 目前, 特征选择的一个研究热点是如何进行高效特征选择, 即如何以尽量小的代价从备选特征集合中选择特征并构成特征子集.

本文提出一种基于属性关系的特征选择算法——NCMIPV. 首先通过分析特征属性与类别标签之间的相关性剔除无关特征, 然后对缩减后的特征子集进行冗余性分析. 为验证算法的有效性与合理性, 选用神经网络作为分类器, 在数据集上进行测试. 实验结果表明了该算法的有效性.

1 相关工作

特征选择研究通常集中于搜索相关特征, 最优特征子集应包括所有强相关特征与一部分弱相关特征, 而不包括任何非相关特征[4].

特征选择算法中经常运用一些评价准则, 主要有距离度量、信息度量、一致性度量等标准[5]. 距离度

收稿日期: 2014-07-15；**修回日期:** 2015-03-09.

作者简介: 胡静(1975–), 女, 副教授, 博士, 从事信息处理等研究；宋铁成(1967–), 男, 教授, 博士生导师, 从事无线通信等研究.

量基于一些重要的距离测度, 例如: 欧氏距离和马氏距离等. 一致性度量的目标在于找出与完整特征集分类效果一致的最小特征子集[6]; 而信息度量则借助于信息论中的基本概念来衡量变量之间的关系. 信息度量不需假定分布已知, 能以量化的形式度量特征间的不确定程度, 而且能有效度量非线性关系[7], 正是由于这样的特点, 信息度量被广泛采用, 并已在实践中表明了其性能. 下面对一些常见的特征选择算法进行简要介绍.

Relief 算法是 Kira 等[8]提出的一种采用欧氏距离作为度量标准的特征选择算法. 特征属性与类别属性的相关性越高, 对应的权重越大, 因此, 应选择权值大于预设阈值的特征, 而将权值小于等于阈值的特征删除. 但是, Relief 算法仅适用于二值分类问题, 而且不能区分冗余特征.

针对 Relief 算法的局限性, 文献 [9] 对其进行了改进, 对于从数据集中随机选择的样本, 从同类以及其他类中均选择 k 个最近邻样本, 并计算特征权重, 选择权重大于阈值的特征. 该方法可以处理多分类问题, 但是仍然无法解决特征冗余的问题.

BIF (最优个体特征)[10]是一种最直接的特征选择方法, 算法思路简单: 将互信息作为评价函数, 即

$$J(f) = I(C; f), \tag{1}$$

其中 $I(C; f)$ 表示类别标签 C 与特征属性 f 之间的互信息量. 对所有特征计算相应评价函数值 $J(f)$, 再按值的大小降序排列, 选择排名靠前的特征构成选择子集 S. 显然, 该算法未考虑所选特征间的相关性, 常常会引入冗余.

文献 [11] 针对 BIF 算法存在的不足提出了改进方法 MIFS (互信息量特征选择), 通常使用备选特征 f 与单个已选特征 s 的相关性作为惩罚项并加以修正, 相应的评价函数为

$$J(f) = I(C; f) - \beta \sum_{s \in S} I(s; f). \tag{2}$$

其中: $I(s; f)$ 表示已选特征 s 与备选特征 f 之间的互信息量, β 为调节参数. 但是, 参数 β 的具体取值对算法性能影响较大.

FCBF (基于相关性的快速特征选择) 方法[4]基于对称不确定度

$$\mathrm{SU}(X, Y) = 2\Big[\frac{\mathrm{IG}(X|Y)}{H(X) + H(Y)}\Big] \tag{3}$$

来度量特征与类别的相互关系以及特征之间的相互关系, 去除相关值小于给定阈值 δ 的特征, 再借助于 Markov blanket 技术对其余特征分析冗余性, 从而快速删除冗余特征. 该算法中阈值难以设定, 且对称不确定度可能会给出一些错误或非确定信息. 式 (3) 中

$$H(X) = -\sum_i P(x_i)\log_2(P(x_i)), \tag{4}$$

$$H(X|Y) = -\sum_j P(y_j)\sum_i P(x_i|y_j)\log_2(P(x_i|y_j)), \tag{5}$$

$$\mathrm{IG}(X|Y) = H(X) - H(X|Y). \tag{6}$$

其中: $\mathrm{IG}(X|Y)$ 为信息增益, $H(X)$、$H(Y)$ 分别为对应变量 X、Y 的熵值.

CMI (基于条件互信息量的特征选择) 算法[12]采用条件互信息量

$$I(x_i; C|S_{i-1}) = H(C|S_{i-1}) - H(C|x_i, S_{i-1}),\ i \geqslant 2 \tag{7}$$

作为评价标准, 并利用近似 Markov blanket 删除冗余特征, 从而得到简化后的特征子集.

式 (7) 中: $I(x_i; C|S_{i-1})$ 表示在给定 S_{i-1} 条件下 x_i 与 C 之间的互信息, $H(C|S_{i-1})$ 表示在给定 S_{i-1} 条件下 C 的条件熵, $H(C|x_i, S_{i-1})$ 表示在给定 S_{i-1} 与 x_i 条件下 C 的条件熵.

2 NCMIPV 算法

基于上述分析结果, BIF 算法并未考虑所选特征之间的相关性, 而 MIFS 算法中引入参数, 会对算法性能产生影响, 且这两种算法都需要事先确定选择特征的个数, 如果特征个数发生变化, 则选择结果也受影响. 同时, 这两种算法没有考虑类别已知条件下, 特征属性之间的相关性. 实际上, 特征属性之间的相关性或冗余性与类别有着较强的关系.

鉴于此, 本文给出如下参考公式:

$$\mathrm{fval} = \sum_{\forall k \in \boldsymbol{S}} \mathrm{MI}(f_k, C) - \sum_{\forall k \in \boldsymbol{S}, k \neq m, m \in \mathbf{fV} - S} \mathrm{RI}(f_k, f_m, C). \tag{8}$$

其中: $\mathrm{MI}(f_k, C)$ 表示已选特征 f_k 与类别 C 之间的互信息, $\mathrm{RI}(f_k, f_m, C)$ 表示已选特征 f_k、备选特征 f_m 与类别 C 之间的冗余程度.

对于子集而言, 将特征属性与类别标签间的互信息作为评价函数的奖励项, 而将各个特征属性与已选特征属性间的冗余关系作为惩罚项. 惩罚项引入条件互信息量是为了考虑类别已知条件下特征之间的冗余性. 由式 (8) 可知, fval 取值越大表明子集性能愈佳.

下面本文将提出基于属性关系的特征选择算法 (NCMIPV). 该算法首先挑选相关特征集合, 然后按照评价准则 (8) 去除冗余特征, 最终得到一个优化特征子集. 采用该算法可以降低数据集合的大小, 提升后续分类算法的运行效率.

NCMIPV 算法的流程如图 1 所示.

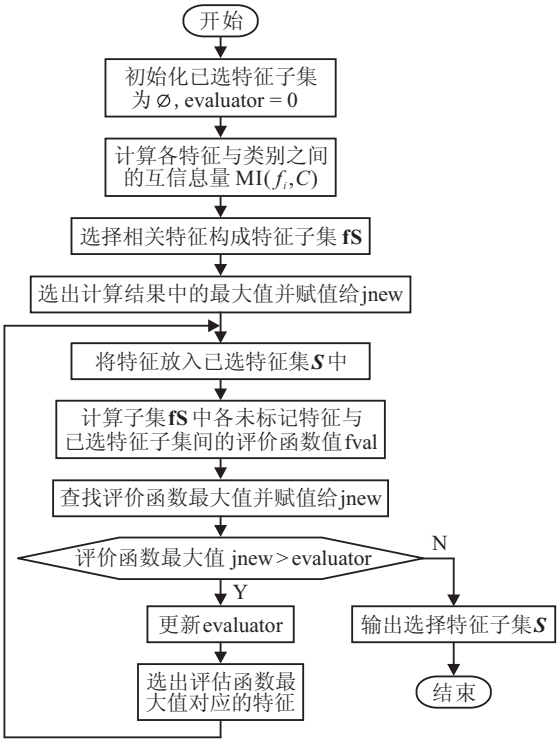

图 1　NCMIPV 算法流程图

算法的具体执行步骤如下.

输入: 特征向量 $\mathbf{fV} = \{f_1, f_2, \cdots, f_n\}, n = |\mathbf{fV}|$, 类别 C;

输出: 特征属性选择结果 $\boldsymbol{S}$.

Step 1: 初始化已选特征子集为 $\varnothing$, 评价函数最大值 evaluator $= 0$.

Step 2: 计算特征集合中各个特征与类别 C 之间的相关性度量 $\mathrm{MI}(f_i, C), 1 \leqslant i \leqslant n$.

Step 3: 选择符合要求的对应特征构成特征子集 **fS** (注意与原始特征向量区分), 并找出上述结果的最大值, 将其赋值给 jnew, 同时将其对应的特征属性 f_j $(1 \leqslant j \leqslant n)$ 加入 $\boldsymbol{S}$ 中, 在 **fS** 中将其打上已读标签.

Step 4: 对于 **fS** 中的其他属性, 根据式 (8) 计算 **fS** 中未读属性与 $\boldsymbol{S}$ 特征子集对应的评价函数值 fval.

Step 5: 根据评价函数寻找最大值, 将其赋值给 jnew, 并与评价函数最大值 evaluator 比较.

Step 6: 若 jnew 大于 evaluator, 则将 jnew 对应的特征属性加入 $\boldsymbol{S}$ 中, 并在 **fS** 中将该属性打上已读标签, 同时更新 evaluator 值为 jnew, 重复 Step 4; 否则, 将 $\boldsymbol{S}$ 作为算法的返回结果.

由上述算法执行步骤可知, 时间开销主要花费在子集搜索过程中, 而该过程直接受评价函数的影响, 不需手动设定选择特征个数即可完成特征子集的构建, 灵活性较强. 不同于 BIF 算法, NCMIPV 算法中引入冗余性度量描述特征之间的冗余程度, 考虑了特征与类别之间的冗余, 而且相较于 MIFS 算法, 评价函数中并无可调节参量, 减少了参量对最终性能的影响. NCMIPV 算法的研究对象是把特征子集作为整体, 比单独考虑某一特征属性更完整、更确切. 总之, NCMIPV 算法兼顾了特征属性与类别之间的相关性以及特征之间的冗余, 对特征子集采用评价函数作为准则选择出符合要求的特征并构成新的特征子集. 整个过程不需要人为设定阈值或其他参量即可完成特征选择, 达到了化简特征的目的.

3　实验及结果分析

实验中的测试数据集选自 UCI 数据集, 如表 1 所示. 数据集的具体说明可参见相关网页[13]. 这些数据集所含的特征数目不等, 特征数据类型也不同. 部分数据集只包含离散型或连续型数据, 部分数据集则既包含连续型数据又包括离散型数据. 对于连续属性数据, 采用文献 [14] 对其进行预处理之后再运行相关算法. 其中, MIFS 算法的参数 $\beta = 0.5$, 未对其进行优化.

表 1　实验数据集

数据集名称	实例数	特征数	离散属性	连续属性	类别数
wine	178	12	12	0	3
zoo	101	17	16	1	7
sonar	208	60	0	60	2
soybean	687	36	36	0	19
anneal	898	38	32	6	6
glass	214	9	0	9	7

实验中将神经网络作为分类器, 在各数据集上分别运行各种特征选择算法, 记录各算法选择的特征子集的大小以及算法运行的时间开销, 如表 2 和表 3 所示. 实验运行环境如表 4 所示. 选择特征子集的大小衡量的是算法去除无关和冗余特征的能力, 运行时间开销则表明算法的时间复杂度, 而分类差错率则间接给出了算法对于数据分类的准确度. 其中, 分类差错率定义为

$$\mathrm{err} = \frac{\text{输出分类结果与实际分类结果不一致的数目}}{\text{分类总数}}. \tag{9}$$

表 2　特征选择算法的选择结果

数据集名称	FCBF	CMI	NCMIPV
wine	10	4	12
zoo	7	1	6
sonar	1	5	10
soybean	14	16	8
anneal	7	10	6
glass	5	3	4

表 3　特征选择算法的运行时间　　s

数据集名称	FCBF	CMI	NCMIPV	BIF	MIFS
wine	0.016 6	0.018 8	0.017 6	0.006 2	0.014 1
zoo	0.015 3	0.014 7	0.012 2	0.006 1	0.012 8
sonar	0.024 3	0.072 9	0.018 6	0.009 3	0.043 3
soybean	0.076 8	0.173 8	0.131 8	0.012 5	0.049 5
anneal	0.053 6	0.222 3	0.049 8	0.016 0	0.055 0
glass	0.013 3	0.014 6	0.016 7	0.005 9	0.010 9

表 4 实验基本环境

项目	具体信息
操作系统	Microsoft Windows XP Professional Service Pack 3
测试平台	Matlab R2012b
CPU	(Intel)Pentium(R) Dual-Core CPU E5300 @ 2.60 GHz (2600 MHz)
内存	4.00 GB (宇瞻 PC2-6400 DDR2 SDRAM 800 MHz)

由表 2 可知: 在 zoo/soybean/anneal/glass 数据集上 NCMIPV 的选择特征数小于 FCBF, 但是在 wine/sonar 数据集上特征数大于该算法; 而与 CMI 相比, NCMIPV 算法选择的特征个数除 soybean/anneal 数据集外均比 CMI 多. 需要说明的是, 设定 BIF 与 MIFS 特征选择算法的选择特征个数与 NCMIPV 算法一致, 这里并未给出.

从运行时间来看, 表 3 给出了各算法的运行时间. 为便于讨论, 不考虑 BIF 算法, 因为该算法仅考虑属性之间的相关性, 从算法复杂度上低于其他算法, 不具有同类可比性. 在 zoo/sonar/anneal 数据集上 NCMIPV 花费时间最少, 结合选择特征, NCMPV 在部分数据集上不仅时间消耗少, 而且选择特征个数也少, 相比之下, CMI 算法的时间开销则大一些.

对选择特征构成的数据子集进行测试, 结果如图 2 所示. 可以看出: NCMIPV 算法在部分数据集上分类差错率较低, 优于其他算法; 在部分数据集上分类差错率与 FCBF 算法相差不大.

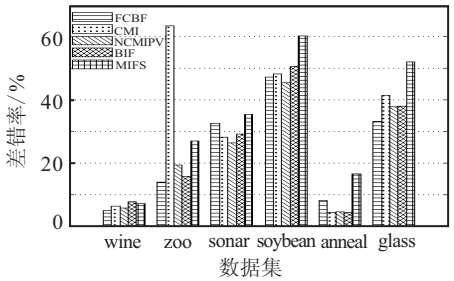

图 2 算法分类差错率

总之, NCMIPV 算法在时间开销、分类差错率、选择特征等方面表现良好, 与已有算法相当, 在某些场合下性能甚至优于部分算法.

4 结 论

本文提出了一种基于属性关系的特征选择算法 NCMIPV. 首先通过相关性分析, 剔除与类别无关的特征; 然后采用子集评价与前向搜索相结合的方法去除冗余特征. 对若干数据集进行实验测试, 测试结果表明与已有算法相比, 该算法呈现出较好的性能, 运行时间开销较小, 选择特征集合大小适中, 选择特征构成的数据子集上测试分类效果良好.

参考文献(References)

[1] 范雪莉, 冯海泓, 原猛. 基于互信息的主成分分析特征选择算法[J]. 控制与决策, 2013, 28(6): 915-919.
(Fan X L, Feng H H, Yuan M. PCA based on mutual information for feature selection[J]. Control and Decision, 2013, 28(6): 915-919.)

[2] Guyon I, Elisseeff A. An introduction to variable and feature selection[J]. J of Machine Learning Research, 2003, 3: 1157-1182.

[3] 肖健华. 智能模式识别方法[M]. 广州: 华南理工大学出版社, 2006: 30-50.
(Xiao J H. Intelligent pattern recognition methods[M]. Guangzhou: South China University of Technology Press, 2006: 30-50.)

[4] Yu L, Liu H. Efficient feature selection via analysis of relevance and redundancy[J]. J of Machine Learning Research, 2004, 5: 1205-1224.

[5] Dash M, Liu H. Feature selection for classification[J]. Intelligent Data Analysis, 1997, 1(3): 131-156.

[6] Dash M, Liu H, Motoda H. Consistency based feature selection[C]. Knowledge Discovery and Data Mining. Berlin: Springer, 2000: 98-109.

[7] 姚旭, 王晓丹, 张玉玺, 等. 特征选择方法综述[J]. 控制与决策, 2012, 27(2): 161-166.
(Yao X, Wang X D, Zhang Y X, et al. A maximum relevance minimum redundancy hybrid feature selection algorithm based on particle swarm optimization[J]. Control and Decision, 2012, 27(2): 161-166.)

[8] Kira K, Rendell L A. The feature selection problem: Traditional methods and a new algorithm[C]. Proc of the 9th National Conf on Artificial Intelligence. San Jose, 1992: 129-134.

[9] Kononenko I. Estimation attributes: Analysis and extensions of REL IEF[C]. Proc of the 1994 European Conf on Machine Learning. Catania: Springer Verlag, 1994: 171-182.

[10] Jain A K, Duin R P W, Mao J. Statistical pattern recognition: A review[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2000, 22(1): 4-37.

[11] Battiti R. Using mutual information for selecting features in supervised neural net learning[J]. IEEE Trans on Neural Networks, 1994, 5(4): 537-550.

[12] Lee J, Kim D W. Efficient multivariate feature filter using conditional mutual information[J]. Electronics Letters, 2012, 48(3): 161-162.

[13] Asuncion A, Newman D J. UCI Machine Learning Repository[EB/OL]. [2014-03-10]. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[14] Fayyad U, Irani K. Multi-interval discretization of continuous-valued attributes for classification learning[C]. Proc of the 13th Int Joint Conf on Artificial Intelligence. Chambéry, 1993: 1022-1029.

(责任编辑: 李君玲)