一种基于假设检验的贝叶斯分类器

李锦善,王志海,王中锋
LI Jin-shan,WANG Zhi-hai,WANG Zhong-feng

北京交通大学 计算机与信息技术学院,北京 100044
Department of Computer and Information Technology,Beijing Jiaotong University,Beijing 100044,China
E-mail:zhhwang@bjtu.edu.cn

LI Jin-shan,WANG Zhi-hai,WANG Zhong-feng.Bayesian classifier based on hypothesis testing.Computer Engineering and Applications,2008,44(21):222-224.

Abstract: Classification is a main branch in Data Mining field.Bayesian classifier as an important technology in this branch has been widely used.Restricted Bayesian learning is a hotspot in these years.In this paper,a kind of hypothesis testing,called volume test is used to find the dependency between attributes.Based on these,propose a method of Bayesian classifier based on hypothesis testing,we call it Bayesian classifier based on Volume Test(BVT).It absorbs advantages of Naïve Bayes and idea of statistical hypothesis testing.Experiments show that this method outperforms Naïve Bayes,TAN,etc,especially when the dataset is large.
Key words: hypothesis testing;bayesian classifier;classification;machine learning

摘 要: 分类是数据挖掘领域的重要分支,而贝叶斯分类方法作为分类领域的重要技术得到了日益广泛的研究和应用。限制性贝叶斯网络在不牺牲太多精确性的前提下简化网络结构,是近几年分类领域的研究热点。论文采用统计学中理论较成熟的体积假设检验(Volume Testing)方法寻找属性间的依赖关系,同时结合假设检验的思想和朴素贝叶斯分类算法的优点构造限制性贝叶斯网络,提出了一种基于假设检验的贝叶斯分类算法,并命名为基于体积检验的贝叶斯分类算法。在 Weka 系统下进行的实验,结果表明,这种方法效果优于朴素贝叶斯方法、TAN 算法等,尤其对大数据集有更佳的表现效果。
关键词: 假设检验;贝叶斯分类器;分类;机器学习
DOI: 10.3778/j.issn.1002-8331.2008.21.060 **文章编号:** 1002-8331(2008)21-0222-03 **文献标识码:** A **中图分类号:** TP301.6

1 前言

朴素贝叶斯(Naïve Bayes)[1]方法具有效率高,计算简单的特点,且有较强的理论基础。但它所依赖的条件独立性假设在现实中是很容易违背的。后来人们发现适当地减弱独立性假设可以提高分类精度,出现了 TAN[2],LBR[3],SuperParent[4],AODE[5]等算法。这些方法归根到底是以不同的方式寻找属性间的依赖关系,构造贝叶斯网络。

统计假设检验的基本任务是根据样本所提供的信息,对未知总体某些方面的假设作出合理的判断。如果能够将此有效地应用到寻找属性间依赖关系的方法中,可能会进一步提高分类效果。而卡方检验是 20 世纪早期统计学领域的重要成就,可用来检验二维列联表两变量的独立性。但这种方法最大缺点在于当样本容量很大时,倾向于拒绝独立性假设,且当拒绝独立性假设时,卡方值不会提供更多的信息。针对这一缺点,1985 年,Diaconis 提出了另一种假设检验的方法[6],并命名为体积检验(Volume Testing),试图在非线性回归中使用几何性质去测验假设检验。

本文在朴素贝叶斯网络的基础上,使用统计学中的体积检验方法寻找属性间的依赖关系,并充分利用假设检验的思想以及统计学结论来源于数据的特点进一步改进分类算法。实验结果表明这个方法分类精度高, 而且对大数据集有更佳的表现效果。

2 体积检验

体积检验方法是利用卡方统计量的一种改进方法。卡方统计量的定义如下:

定义 1[7] 设总体 ξ 的分布形式未知。在实轴上,用 $k-1$ 个分点将随机变量 ξ 的值域划分为互不相交的 k 个区间 $A_1=[a_0,a_1),A_2=[a_1,a_2),\cdots,A_k=[a_{k-1},a_k)$,这些区间的长度可以不相等。设 $(x_1,x_2,\cdots,x_n)$ 是 ξ 的容量为 n 的样本观测值,v_i 为样本观测值落入区间 A_i 的频数, 随机变量 ξ 落到区间 A_i 的事件仍然用 A_i 表示,事件 A_i 发生的概率为 p_i,由此构造统计量:

$$\chi^2=\sum_{i=1}^{k}\left(\frac{v_i}{n}-p_i\right)^2\cdot\frac{n}{p_i}$$

这个统计量就称为卡方(K.Pearson)统计量。

体积检验的主要思想是假设列联表服从参数相同的多项

基金项目: 国家自然科学基金(the National Natural Science Foundation of China under Grant No.60673089)。
作者简介: 李锦善(1982-),女,工学硕士学位,主要研究领域为数据挖掘与机器学习;王志海(1963-),男,博士学位,特聘教授,主要研究领域为数据挖掘与机器学习;王中锋(1977-),男,工学硕士学位,主要研究领域为数据挖掘与机器学习。
收稿日期: 2008-04-30 **修回日期:** 2008-06-20

分布,即所有可能得到的列联表有相同的抽取概率,计算所得到的观测表的卡方值,然后计算在所有表中小于等于该卡方值的表所占的比例。将此比例作为显著性水平,若此比例较小,说明两随机变量独立,否则,说明相关。

计算观测表占总比例,即显著性水平的公式为:

$$\varepsilon(\chi^2)\doteq\left(\frac{\pi\chi^2}{n}\right)^{\frac{D}{2}}\left(\frac{c_{I,J}}{\prod_{h=1}^{IJ-1}\left(1+\frac{h}{n}\right)}\right) \tag{1}$$

其中,n 为样本容量 $D=(I-1)(J-1)$,$c_{I,J}=\frac{\Gamma(IJ)\Gamma((J+1)/2)^I\Gamma((I+1)/2)^J}{(D/2)!\ \Gamma(I(J+1)/2)\Gamma(J(I+1)/2)}$。

公式(1)中,若以边缘表作为条件,则可得到公式(2)。

$$\varepsilon(S|\boldsymbol{r},\boldsymbol{c})\doteq\left(\frac{\pi\chi^2}{n}\right)^{\frac{D}{2}}\frac{c_{I,J}(\boldsymbol{r},\boldsymbol{c})}{\widehat{V}} \tag{2}$$

其中,$S=\frac{\chi^2}{n}$,$c_{I,J}(\boldsymbol{r},\boldsymbol{c})=\left(\frac{1}{D/2}\right)!\ \left(I\prod_{i=1}^{I}r_i\right)^{\frac{J-1}{2}}\left(J\prod_{j=1}^{J}c_i\right)^{\frac{I-1}{2}}$。

行边缘表第 i 项的值 $r_i=p_{i+}=\sum_{j=1}^{J}p_{ij}$。

列边缘表第 j 项的值 $c_j=p_{+j}=\sum_{i=1}^{I}p_{ij}$。

p_{ij} 为二维列联表中第(i,j)项的观测频率。

$$\widehat{V}=\left(1+\frac{IJ}{2n}\right)^{D}\frac{I^{(J-1)/2}J^{(I-1)/2}\Gamma(Jk_{\bar{r}})}{\Gamma(J)^I\Gamma(k_{\bar{r}})^J}\left(\prod_{i=1}^{I}\bar{r}_i\right)^{J-1}\left(\prod_{j=1}^{J}\bar{c}_j\right)^{k_{\bar{r}}-1}$$

$$\bar{\boldsymbol{r}}=(1-\omega)\frac{\mathbf{1}_I}{I}+\omega\bar{\boldsymbol{r}},\bar{\boldsymbol{c}}=(1-\omega)\frac{\mathbf{1}_J}{J}+\omega\boldsymbol{c}$$

$$\omega=\frac{1}{1+IJ/2n}$$

$$k_{\bar{r}}=\frac{J+1}{J\|\bar{r}\|^2}-\frac{1}{J}$$

这种方法的优点在于所得的值不依赖于样本容量 n,只有在样本容量较小时,可能会受到一些影响,而对于大样本容量,不会像卡方检验那样产生极小的显著性概率值。

3 贝叶斯分类方法

贝叶斯网络能够较好地表示属性之间的依赖关系,成为分类的重要方法。使用这种方法进行分类主要包括两个阶段,贝叶斯网络的建立阶段和利用贝叶斯定理计算分类概率阶段。各种不同的贝叶斯分类算法的区别主要在于建网阶段。

朴素贝叶斯方法假设各属性之间在类属性条件下彼此独立,这种方法计算简单、精度较高,已经得到了广泛的认可。但在现实中,它的独立性假设很容易违背,从而影响了分类精度。由此出现了各种减弱独立性假设的方法。比如 TAN 算法,它是在朴素贝叶斯网络的基础上通过计算条件互信息为每一个属性又添加了另一个属性作为父结点,也取得了较好的效果。但这种方法限制了每个数据集都要选择 $n-1$ 条边(n 为非类属性个数),实现不够灵活。SuperParent 方法在这方面有所改进,但是其通过依次尝试的方法寻找 SuperParent 和 FavoriteChild,使其时间优势大打折扣。而 LBR 采用与前两者不同的懒惰式学习方式,提高了分类精度,但在建网时需要花费大量时间。

4 基于体积检验的贝叶斯分类算法

基于体积检验的贝叶斯分类算法(BVT,Bayesian classifier based on Volume Test)主要思想是:首先将贝叶斯网络初始化为朴素贝叶斯网络。再判断数据集大小,若较小则直接采用朴素贝叶斯方法进行分类,返回分类结果;否则按照体积检验公式(2)计算每两个属性之间的显著性水平,并将结果保存到一个 $n\times n$ 的显著性水平矩阵中。显然,该矩阵为对称阵。然后以所有属性为端点建立一个无向完全图,图中端点之间的权值为此两属性之间的显著性水平。下一步根据此无向完全图建立最大生成树。最后指定一个根节点形成有向图。通过以上步骤建立贝叶斯网络,再采用贝叶斯定理进行分类。算法具体步骤见图 1。

输入:一个具有若干非类属性和一个类属性的数据集,及一条待测实例
输出:待测实例的类属性取值
1.将网络初始化为朴素贝叶斯网络;
2.若 *sqrt*(数据集实例个数×非类属性个数)<*EDGEVALUE* 则转到步骤 7,否则进行步骤 3;
3.按照公式(2),计算每两个属性 i 和 $j(i\neq j)$之间的显著性水平;
4.以所有属性为端点建立一个无向完全图,端点间的权值为步骤 3 计算所得的显著性水平值;
5.根据此无向完全图建立最大生成树;
6.通过指定一个根节点将最大生成树形成一个有向图;
7.利用贝叶斯定理,按照以上步骤生成的网络对待测实例进行分类

图 1 基于体积检验的贝叶斯分类算法(BVT)

算法步骤 2 使用实例个数和非类属性个数作为数据集大小的衡量标准,这是有充分理论依据的:首先,统计学的结论强烈依赖于样本所提供的信息,提供的信息越充分,越能得到可靠的结果。而本文算法采用假设检验的方法判断属性间的依赖关系,数据集实例个数和属性个数越多,数据量就越大,抗噪音数据的能力就越强,由此所推出的结论就越可靠。其次,根据假设检验的思想,如果数据集数据量小,就没有充分的理由确定两属性存在依赖关系,则假设各属性间彼此独立,而直接采用朴素贝叶斯方法。实验也表明这种判断方法不但提高了分类精度,而且由于对于某些数据集省略了寻找属性间依赖关系的过程,加快了平均运行时间。

5 实验分析及结论

实验平台为 Weka 系统[8],数据集全部选自 UCI 数据库[9]。表 1 给出了本文使用的 40 个数据集的基本信息,表中 Size# 表示实例个数;C# 表示类属性取值个数;A# 表示非类属性个数,每个数据集均有一个类属性。选用 TAN 算法和朴素贝叶斯算法作为比较算法。所采用的实验方式是:采用 10 重交叉验证[10],按照 Weka 下参数设置方式[11],取随机数种子 s 为1,3,5,7,11,分别在每个数据集上运行 5 次,并将取得的平均错误率作为衡量标准。对于连续属性,所有算法都采取 Weka 下自带的离散化方法进行离散化处理。而对于数据集中缺损值不进行特殊处理,即采用忽略缺损值方式。

实验中取临界值 *EDGEVALUE* 为 80,所得实验结果如表 2。表中本文算法粗体部分表示小于临界值的数据集,在这些数据集上直接采用朴素贝叶斯方法分类。实验结果表明,本文算法的平均错误率为 16.07%,而朴素贝叶斯算法的平均错误率为 17.72%,平均错误率降低了 1.65%,在各个数据集上的错误率按朴素贝叶斯与 BVT 错误率差值由大到小排列,所得结果如图 2 所示。

表 1 数据集描述

No	Domain	Size#	C#	A#	No	Domain	Size#	C#	A#
1	Adult	48 842	4	18	21	Lymphography	148	4	18
2	Annealing Process	898	10	6	22	mfeat-mor	2 000	10	6
3	Audiology	226	10	16	23	Pen Digits	10 992	10	16
4	Balance Scale	625	2	8	24	Pima Indians Diabetes	768	2	8
5	Breast Cancer	699	3	8	25	Post-operative patient	90	3	8
6	Credit Screening	690	22	17	26	Primary tumor	339	22	17
7	Echocardiogram	131	2	57	27	Promoter gene sequences	106	2	57
8	German	1 000	6	36	28	Satellite	6 435	6	36
9	Heart	270	7	19	29	Segment	2 310	7	19
10	Glass identification	214	7	9	30	Shuttle	58 000	7	9
11	Heart Disease	303	3	8	31	Sign	12 546	3	8
12	Hepatitis Prognosis	155	3	10	32	Solar flare	1 389	3	10
13	Horse Colic	368	2	9	33	Solar flare	1 389	2	9
14	House Votes 84	435	2	60	34	Sonar classification	208	2	60
15	Hungarian	294	19	35	35	Soybean	683	19	35
16	Hypothyroid Diagnosis	3 163	3	60	36	Splice Junction Gene	3 177	3	60
17	Iris Classification	150	2	9	37	Tic-Tac-Toe end	958	2	9
18	Labor Negotiations	57	4	18	38	Vehicle	846	4	18
19	Letter Recognition	20 000	3	13	39	Wine recognition	178	3	13
20	Liver disorders	345	7	16	40	Zoology	101	7	16

表 2 实验结果

No	Domain	NB	TAN	BVT	No	Domain	NB	TAN	BVT
1	Adult	18.03	15.99	16.78	21	Lymphography	14.86	19.59	**14.86**
2	Annealing Process	5.70	7.51	4.54	22	mfeat-mor	30.40	29.01	29.36
3	Audiology	27.52	42.04	24.51	23	Pen Digits	12.92	4.35	5.24
4	Balance Scale	8.42	14.53	**8.42**	24	Pima Indians Diabetes	24.38	24.53	**24.38**
5	Breast Cancer	2.66	4.72	**2.66**	25	Post-operative patient	32.44	32.89	**32.44**
6	Credit Screening	15.19	16.72	16.29	26	Primary tumor	50.80	53.51	**50.80**
7	Echocardiogram	28.70	32.21	**28.70**	27	Promoter gene sequences	9.43	20.94	**9.43**
8	German	24.36	24.38	25.84	28	Satellite	19.13	12.55	14.28
9	Heart	12.52	6.64	**12.52**	29	Segment	10.97	6.34	6.18
10	Glass identification	15.78	17.93	**15.78**	30	Shuttle	9.67	5.74	7.26
11	Heart Disease	16.30	17.43	**16.30**	31	Sign	38.64	27.77	31.47
12	Hepatitis Prognosis	15.61	17.81	**15.61**	32	Solar flare	3.95	1.20	1.38
13	Horse Colic	20.38	20.11	20.54	33	Solar flare	18.66	15.97	16.34
14	House Votes 84	10.07	6.11	7.82	34	Sonar classification	23.85	23.65	25.48
15	Hungarian	15.65	19.05	**15.65**	35	Soybean	7.00	12.83	6.47
16	Hypothyroid Diagnosis	2.95	2.54	2.59	36	Splice Junction Gene	4.58	5.15	4.65
17	Iris Classification	5.33	9.07	**5.33**	37	Tic-Tac-Toe end	30.38	24.07	23.34
18	Labor Negotiations	6.67	14.04	**6.67**	38	Vehicle	39.31	28.61	29.62
19	Letter Recognition	29.96	17.59	17.96	39	Wine recognition	3.48	6.40	**3.48**
20	Liver disorders	36.06	33.97	**36.06**	40	Zoology	5.94	4.95	**5.94**
						平均错误率(%)	17.72	17.51	16.07

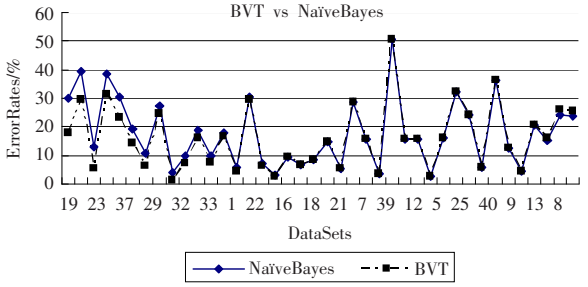

图 2 BVT 与 NaiveBayes 错误率比较

由图 2 可见,在前半部分的 17 个数据集上,本文算法有优势,尤其是在前 8 个数据集上优势较为明显。进一步观察发现,这些数据集或者实例个数较多,或者属性个数较多。如数据集 19(Letter Recognition),共有 20 000 条实例和 16 个非类属性;而数据集 31(Sign),共有 12 546 条实例和 8 个非类属性。图 2 中间部分的 18 个数据集实例个数或非类属性个数较少,故直接采用朴素贝叶斯分类器。而只有在 5 个数据集上本文算法的错误率略高于朴素贝叶斯分类器。

为了进一步验证算法的效果,又与 TAN 算法进行了比较,TAN 算法也采用与本算法相同的离散化方式,在 40 个数据集上的平均错误率为 17.51%,较本文算法平均高 1.44%。在各个数据集上的错误率按 TAN 与 BVT2 错误率差值由大到小排列,所得结果比较结果如图 3。

(下转 230 页)