DOI:10.11992/tis.201603038
网络出版地址:http://www.cnki.net/kcms/detail/23.1538.TP.20160513.0925.028.html

基于置换检验的聚类结果评估

谷飞洋,田博,张思萌,陈征,何增有
(大连理工大学 软件学院, 辽宁 大连 116621)

摘 要:对聚类结果,传统的评估方法不能从统计意义上对结果评估。ECP 是一种新颖的基于置换检验的评估算法。ECP 直接对聚类结果进行置换检验从而计算出 p -value。为了测试 ECP 的效果,利用了 UCI 中的 iris, wine, yeast 数据集对算法进行评测。实验结果表明,ECP 可以在能够接受的时间内运算出比较准确的实验结果。

关键词:聚类;聚类评估; 统计检验;置换检验
中图分类号:TP393 **文献标志码**:A **文章编号**:1673-4785(2016)03-0301-09

中文引用格式:谷飞洋,田博,张思萌,等.基于置换检验的聚类结果评估[J]. 智能系统学报, 2016, 11(3): 301-309.
英文引用格式:GU Feiyang, TIAN Bo, ZHANG Simeng, et al. Statistical evaluation of the clustering results based on permutation test[J]. CAAI transactions on intelligent systems, 2016,11(3): 301-309.

Statistical evaluation of the clustering results based on permutation test

GU Feiyang, TIAN Bo, ZHANG Simeng, CHEN Zheng, HE Zengyou
(Software School, Dalian University of Technology, Dalian 116621, China)

Abstract: For the result of clustering, tranditional methods of evalution couldn´t assess the result in statistics. We propose a new algorithm called ECP(Statistical evaluation of Clustering based on Permutation test) which uses permutation test to evaluate the result of clustering. To evaluate the performance of the algorithm, we use the data sets, iris, wine, yeast, from UCI datasets. Experimental results show that the performance of the algorithm is good.

Keywords: clustering; clustering evaluation; statistical test; permutation test

随着获得的数据越来越多,利用机器学习、数据挖掘[1-3]等手段从数据中获取潜在的知识变得越来越重要。然而如何评估挖掘出来的信息,即评估数据挖掘结果的质量是一个十分重要的问题。只有一个好的评估方法,才能保证挖掘算法发现高质量的信息。聚类[4-5]是数据挖掘领域一个很重要的分支。同时,聚类的应用也越来越广泛。随着聚类的广泛应用,如何有效地评估聚类结果的质量[6-7]成为一个重要的研究课题。虽然评估聚类结果的重要性一点不亚于挖掘算法本身,但是评估方面却没有受到它应有的重视。

针对聚类,现有的方法主要是用评价函数对聚类结果评估。这种函数一般分 3 种类型:紧密型、分散型和连接型。常见的评估函数有 DB-Index, Silhouette-Index, Dunn-Index 等。这些函数能够评估聚类结果,但是这些函数评估出来的结果往往没有一个比较好的可以参考的值。即一个评估值计算出来之后得到的只是一个评估值,至于这个值达到什么标准能够接受并不能确定。利用统计方法评估聚类结果的算法很少,其主要原因是聚类的特殊性与复杂性使传统的统计方法很难用到聚类质量评估上。近年来有一些利用随机方法来评估聚类结果的研究,但也存在一定的问题。本文根据存在的问题提出了一种基于置换检验的评估方法。

1 相关研究

1.1 利用簇结构评估聚类质量

该方法先对原始数据聚类,然后将原始数据集按照一定的约束随机置换抽样构造新的数据集。抽样之后用同样的聚类算法对样本数据集进行聚类。

收稿日期:2016-03-19. 网络出版日期:2016-05-13.
基金项目:国家自然科学基金项目(61572094).
通信作者:何增有. E-mail:zyhe@ dlut.edu.cn.

这样重复大量的次数后,再用评估函数(如 DB-Index)计算每个样本的函数值。如果原始数据集聚类结果的函数值小于大部分随机构造的数据集聚类结果的函数值,那么说明挖掘出来的信息是可靠的,否则说明聚类结果不可靠。更通俗一点,如果原来数据集没有好的簇结构,那么无论怎么聚类,结果都是不好的。代表性的方法有最大熵模型抽样[8]、矩阵元素交换[9]等。利用数据集簇结构来评估聚类质量[10]的方法能很好地评估出簇结构不好的聚类结果。实验证实对不同数据集进行聚类,有明显簇结构数据集的 p-value 会比没有明显簇结构的 p-value 小很多。但是这种方法并不能准确评估聚类的质量。从某种意义上讲,这种方法更适合评估一个数据集是否有好的簇结构。

1.2 SigClust

SigClust[11] 认为如果一个数据集符合高斯分布,那么对这个数据集的任何分割都是不合理的。因此这个方法的前提假设是:一个单一的簇的元素符合高斯分布。SigClust 主要是针对 $k=2$ 的聚类评估。对于 $k>2$ 的情况,还没有比较好的解决办法。

1.3 层次聚类的 p -value 计算

这种方法主要针对层次聚类的评估[12,13]。层次聚类后会形成一个二叉树。对二叉树上的每个节点都进行置换检验,算出每个节点划分对应的 p -value。这种算法的空假设为:当前节点的左子树和右子树应该属于一个簇。如果算出 p -value 足够小就说明空假设是一个小概率事件,应该拒绝。该方法是将当前节点的左子树和右子树打乱,按照一定的约束随机分配左子树和右子树的元素。抽样若干次后形成的随机样本集按照某种指标与原始划分对比计算出 p -value。这个评估只能针对层次聚类,不能对其他的聚类算法进行评估。另外这样计算出的 p -value 只是每个节点上的 p -value,并不是全局聚类的 p -value。

2 基本概念

2.1 无监督聚类质量评估函数

如果数据集中的元素没有类标签,聚类结果的评价就只能依赖数据集自身的特征和量值。在这种情况下,聚类的度量追求有 3 个目标:紧密度、分离度和链接度。

紧密度 簇中的每个元素应该彼此尽可能接近。紧密度的常用度量是方差,方差越小说明紧密度越大。

分离度 簇与簇之间应该充分分离。有 3 种常用方法来度量两个不同簇之间的距离。单连接:度量不同簇的两个最近成员的距离。全连接:度量不同簇的两个最远成员的距离。质心比较:度量不同簇的中心点的距离。

链接度 链接度指簇中的元素成员至少要跟同一个簇内的元素比较像。这个可以用来评估簇模型不是圆形或者球形的聚类结果,比如 DBSCAN 的聚类结果。

本文用一种无监督评估聚类质量的方法, Davies-Bouldin Index,即 DB_Index。

$$\mathrm{DBI}=\frac{1}{k}\sum_{i=1}^{k}\max\left(\frac{S_i+S_j}{D_{ij}}\right).$$

式中:S_i 表示第 i 个簇内的元素与质心的标准方差,D_{ij} 表示第 i 个簇与第 j 个簇质心间的欧几里德距离,k 表示簇的数目。

DBI 的思想是一个高质量的聚类结果需要满足:同一个簇的各元素间相似度大,不同类之间的相似度小。在 DBI 中,分子越小意味着簇内元素相似度越大,分母越大意味着簇间相似度越小。

2.2 聚类评估的 p -value

给一个数据集 X,用 DB-Index 计算聚类结果的函数值为 x_0x_0。数据集 X 所有可能的聚类结果的函数值为 $x_1,x_2,\cdots x_{N_{\mathrm{all}}}$。置换检验的 p -value 定义为

$$P_{perm}=\frac{\sum_{n=1}^{N_{\mathrm{all}}}I(x_n\leqslant x_0)}{N_{\mathrm{all}}}$$

式中 I 是一个逻辑函数。当 $x_n\leqslant x_0$ 的情况下为 1,否则为 0。由于要枚举出所有的聚类方案的复杂度是指数级别的,所以需要采取其他的策略。抽样出所有情况的一个子集 Y,并计算子集 Y 中所有元素的函数值为 $x_1,x_2,\cdots x_N$,其中 $N\ll N_{\mathrm{all}}$。这时候置换检验的 p -value 被定义为

$$P_{\mathrm{perm0}}=\frac{\sum_{n=1}^{N}I(x_n\leqslant x_0)}{N}.$$

一些研究为了避免 p -value 为 0 的情况,将 p -value 的定义修改为

$$P_{\mathrm{perm1}}=\frac{1+\sum_{n=1}^{N}1(x_n\leqslant x_0)}{N+1}$$

这种方法把分子加 1 的理由是把 x_0 也看作置换检验一个样本的函数值。这就避免了得到 p -value 为 0 的试验结果。然而这种做法事实上是不太合理的。试想如果抽样 999 次没有发现比 x_0 更小的统计值,这样草率地得出结论当前置换检验的结果为 0.001 显然太武断了。因为可能抽样 99 999 次依旧没有比 x_0 更优的样本。那么依照这个计算公式 p -value 又为0.000 01。而实际上 p -value 的值可

能更小。因此本文把 p -value 的定义为 $P_{\text{perm0}}P_{\text{ecdf0}}$。

置换检验的准确性取决于抽样的数目,一般的置换检验抽样的次数都在 1 000 次以上。为了得到更精确的 p -value 抽样的次数越多越好,理想的情况是置换所有的可能。然而对于不同的数据集合,甚至很难预测需要执行多少次置换才能够得到比较好的结果。往往为了得到更精确的值就会增大抽样次数,但是增加抽样次数的代价是增加计算的复杂性。对于普通的数据集往往抽样次数达到 10 000 次之后就不太容易提高抽样次数。而这样做又产生出了一个问题。如果一个聚类结果真实的 p -value 为 0.000 001。而抽样的次数只有 10 000 次的话,那么 p -value 为就为 0 了。针对这些问题,本文提出了一种新的聚类评估方法,ECP,该方法能比较好地解决上文提到的问题。

3　基于置换检验的聚类结果评估

3.1　基本思想

本文提出的置换检验方法将关注点锁定在了聚类的结果上。评估聚类结果的本质是看聚类算法对数据集中元素的划分质量。从这个角度出发,可以枚举对数据集的划分,然后用评估函数算出枚举划分的函数值。如果绝大部分划分都没有要评估的聚类结果质量好的话,那么就说明要评估的聚类结果质量比较好。相反地,就说明要评估的聚类结果质量并不好。

因此对于一个聚类结果,本文定义了零假 $H0$:当前聚类结果不是一个高质量的聚类。然后计算这个零假设的 p-value。如果这个 p-value 非常小,就认为这个划分结果可以接受,可以拒绝 $H0$。否则认为这个聚类结果不能接受。

定义数据集 $\boldsymbol{X}$ 是一个包含 n 个元素的 d 维数值型矩阵。首先对数据集聚类,聚成 k 簇后每个元素都会归属于一个簇。我们对每个簇进行标号。标号从 0 开始,往后依次是 1,2, …,$k-1$。定义 CI_i 为第 i 个元素所属的簇标号。比如 $CI_3=2$ 表示第 3 个元素属于标号为 2 的簇。

接下来是抽样。抽样要满足一定约束。本文定义的约束是:样本中簇包含元素的数目要与待评估聚类结果中簇中元素的数目保持一致。举个例子,假设数据集元素数目 n 为 100。划分成 3 簇,划分簇中的数目分别是 40、33、27。那么抽样出来的样本也要满足这些条件,也就是要划分成 3 簇,并且簇中元素的数目也必须是 40、33、27。具体的抽样方法:首先搜集所有元素的簇标号,然后将这些簇标号随机地分配给每个元素。其实这个过程是洗牌算法。算法 1 描述了抽样的过程。

算法 1　Shuffle(CI, n)

for $i\leftarrow 0$ to $n-1$ do

index $\leftarrow$ rand( ) mod ($i+1$)

swap(CI_i, $CI_{\text{index}}CI_i$, CI_{index})

可以用数学归纳法进行证明算法 1 保证了每个元素获得同一簇标号的概率是一样的。抽样的复杂度为 $O(n)$。这样进行抽样 N 次,就得到了 N 个样本。然后利用样本对原始聚类结果进行评估。用 DB-Index 算出原始聚类的函数值 x_0 与样本的函数值 $x_1,x_2,\cdots,x_N$。有了这些值就能计算 p -value 了。具体算法如下。

算法 2　ECP1

用 DB-Index 计算聚类结果的函数值 x_0。

for　$i\leftarrow 1$ to N do

Shuffle(CI, n)

用 DB-Index 计算样本的函数值 x_i

计算 p -value

一般情况下 $k\ll n$,因此 DB-Index 的复杂度为 $O(n\times d)$。抽样一次的复杂度是 $O(n)$,容易算出总体复杂度为 $O(N\times n\times d)$。这个复杂度还是比较高的。所以需要想一些方法来降低复杂度。N 是抽样次数,期望越大越好。可以看到 DB-Index 是影响复杂度的主要因素。如果降低 DB-Index 计算的复杂性,那么就可以在相同的时间内抽取更多的样本来提高 p -value 的准确度。本文发现了 DB-Index 公式的特点,对上文提到的算法做了改进。

3.2　加速技巧

首先选取聚类结果作为初始状态。然后随机交换一对簇标号不同的元素的簇标号。交换后把此时的划分作为一个样本,直接计算 DB-Index 的函数值。接下来继续交换一对簇标号不同的元素的簇标号,交换后计算 DB-Index 的值。这样迭代 N 次后就会得到 N 个样本的函数值。利用这 N 个值就可以计算出 p -value。整个算法流程如下。

算法 3　ECP2

用 DB-Index 计算聚类结果的函数值 x_0

for $i\leftarrow 1$ to N do

随机交换一对簇标号不同元素的簇标号

用 DB-Index 计算抽样结果的函数值 x_i

计算 p -value

对比 ECP1,ECP2 只是修改了第 3 步的抽样方法。为什么修改了抽样方法就可以增大抽样次数?下面将仔细讨论 DB-Index 的计算过程。DB-Index 的计算公式为

$$\mathrm{DBI} = \frac{1}{k}\sum_{i=1}^{k}\max\left(\frac{S_i + S_j}{D_{ij}}\right).$$

由 S_i 的定义可以得出：

$$S_i = \sqrt{\frac{\sum_{j=1}^{m_i} \| z_j - \bar{z} \|^2}{m_i}}.$$

式中 m_i 是簇 zi 中元素的数目。z_j 是簇 i 中第 j 个元素的属性向量，$\bar{z}$ 是簇 i 质心的属性向量。由于数据是 d 维的，所以 $\| z_j - \bar{z} \|^2$ 就是各个维度的平方和。因此可以单独对每一维计算，然后再把所有维度的平方相加即可：

$$\sum_{j=1}^{m_i} \| z_j - \bar{z} \|^2 = \sum_{t=1}^{d}\sum_{j=1}^{m_i} (a_{jt} - \bar{a}_t)^2,$$

式中：a_{jt} 是簇 i 中第 j 个元素的第 t 个属性值，$\bar{a}_t$ 是簇 i 质心的第 t 个属性值。下面直接讨论第 t 维的计算方法：

$$\frac{\sum_{j=1}^{m_i} \| z_j - \bar{z} \|^2}{m_i} = \frac{\sum_{t=1}^{d}\sum_{j=1}^{m_i} (a_{jt} - \bar{a}_t)^2}{m_i} = \sum_{t=1}^{d} \frac{\sum_{j=1}^{m_i} (a_{jt} - \bar{a}_t)^2}{m_i}$$

其中：

$$\frac{\sum_{j=1}^{m_i} (a_{jt} - \bar{a}_t)^2}{m_i} = \frac{\sum_{j=1}^{m_i} a_{jt}^2}{m_i} - \bar{a}_t^2$$

因此

$$\frac{\sum_{j=1}^{m_i} \| z_j - \bar{z} \|^2}{m_i} = \sum_{t=1}^{d} \frac{\sum_{j=1}^{m_i} a_{jt}^2}{m_i} - \bar{a}_t^2$$

$\sum_{j=1}^{m_i} a_{jt}^2$ 是簇 i 中所有元素中第 t 维的平方和，$\bar{a}_t$ 是簇 i 中所有元素第 t 维的平均值。所以为了计算 S_i，每一维只需要维护两个值就可以了：平方和与平均值。当簇标号交换的话，能在 $O(1)$ 复杂度内修正这两个值。修改完每个维度的这两个值后，就可以用 DB-Index 算出函数值了。

可以看出修改一个簇的平方和与平均值复杂度是 $O(d)$ 的。因此 DB-Index 的计算复杂度就是 $O(k\times k\times d)$ 了。没有加速的 DB-Index 的计算复杂度是 $O(n\times d)$。一般情况下，$k \ll n$。所以这种方法的效率有明显的提升。

3.3 更准确的 p -value

上边提到计算 DB-Index 的方法的复杂度为 $O(k\times k\times d)$。虽然相比于原先的计算方法已经优化很多，但是对于 p -value 非常小的情况，可能依旧由于抽样数目有限而无法算出精确的 p -value。这种情况下算出的 p -value 就会为 0，然而这样的结果是不准确的。

如果知道了样本 DB-Index 函数值的概率分布就可以根据原始聚类结果的函数值算出精确的 p -value 了[14]。聚类是一种半监督的机器学习，其本质对元素所属类别的划分。如果对元素随机划分无穷次。那么质量特别高的划分的比例会很小。同样的，质量极端差的划分占的比例也会很小。很大比重的划分都介于它们之间。而正态分布的特点是：极端概率很小，中间的概率很大。经过对数据的分析，聚类划分的 DB-Index 函数值比较符合正态分布。因此可以假设抽样样本 DB-Index 的函数值符合正态分布。实际上正态分布符合很多自然概率分布的指标。下面要做的就是得到正态分布的参数。对于一维的正态分布均值和方差用式(1)和(2)得到：

$$\mu = \frac{\sum_{i=1}^{N} x_i}{N} \tag{1}$$

$$\partial = \sqrt{\frac{(x_i - \bar{x})^2}{N-1}} \tag{2}$$

有了概率分布函数，就能将原始聚类结果 x_0 代入概率分布算出 p -value 了。

这样估出概率分布函数实现了在整体复杂度没有增加的前提下用较少的抽样得到更为精确 p -value 的目的了。

本文利用公式 $P_{\mathrm{perm0}} \dfrac{\sum_{n=1}^{N} I(y_n > x_0)}{N}$ 计算 p -value 实际上是利用了大数定律。大数定律的本质是如果有无穷次试验，事件出现的频率就会无限趋近于事件发生的概率。而由于抽样次数有限，本文假设了 DB-Index 的函数值符合正态分布。不过对于抽样 N 次后发现，已经有足够的样本可以精确算出 p -value 的话，就不需要用正态分布计算了。然而如果抽样 N 次后没有足够的样本可以用大数定律精确地计算 p -value 的话就要拟合正态概率分布函数了。对于有多少个样本满足 $x_i \leqslant x_0$ 算是足够呢？这是一个阈值问题。上边的过程总结起来如算法 4。

算法 4 ECP

抽样 N 次，算出每次的函数值 x_i

统计 $x_i \leqslant x_0$ 的数目 M

如果 $M \geqslant$ Limit 利用公式 P_{perm0} 计算 p -value

否则，拟合正态概率分布算出 p -value

其中 Limit 是 ECP 的一个参数，是用 P_{perm0} 计算出 p -value 的最低数目限制。ECP 不同于很多其他的置换检验方法。这种方法实现了用较少的抽样计

算出更为精确 p -value 的目的,在效率上有了非常大的飞跃。

4　实验

实验选取了 iris、wine 和 yeast 等 3 个数据集。这 3 个数据集都来自 UCI 数据库[15]。iris、wine 和 yeast 数据集的属性都是数值型的,并且这 3 个数据集都带有类标签。

4.1　利用 p -value 选择合适的聚类算法

从聚类这个概念提出以来出现了很多聚类算法。对于一个具体的应用,选择合适的聚类算法是一个很重要的问题。本文认为对于同一个数据集用不同的算法聚类, p -value 小的那个结果更为可靠。为此本文对同一数据集选用多种算法聚类来验证 p -value 对选择聚类算法的有效性。实验结果如表 1。从实验结果可以看出,对于同一数据集 p -value 小的聚类算法对应的 f -score 和 accuracy 比较大。这说明利用 p -value 选择聚类算法是可靠的。本文还计算了 p -value 与 f -score 和 accuracy 的相关系数。本文用 k-means 对同一数据集聚类 100 次。通过控制 k-means 的迭代次数来控制划分的质量。这样就避免了正常 k-means 聚类只会出现若干个固定情况的问题。

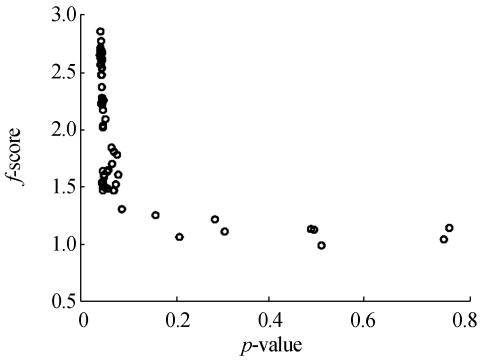
表 1　不同聚类方法的 p -value, f -score, accuracy

Table 1　The p -value, f -score, accuracy of different cluster algorithms

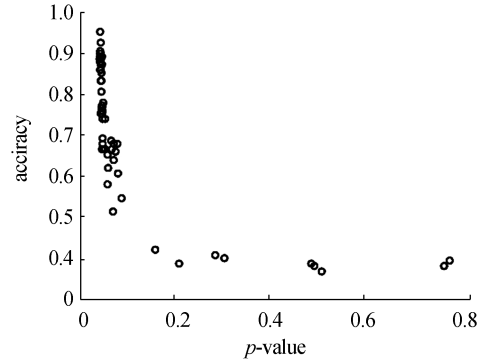
数据	算法	p-value	f-score	accuracy
Iris	Random	0.456 254	1.134 140	0.380 000
	Hierarchical Clustering	0.100 548	1.656 570	0.666 667
	DBSCAN	0.042 825	2.714 400	0.906 667
	k-means	0.042 751	2.655 840	0.886 667
Wine	Random	0.559 588	1.095 420	0.410 112
	Hierarchical Clustering	0.001 574	1.666 460	0.657 303
	DBSCAN	1.892 991e-05	2.833 750	0.943 820
	k-means	1.818 384e-05	2.832 200	0.943 820
Yeast	Random	0.688 145	1.078 260	0.357 198
	Hierarchical Clustering	0.003 871	0.835 371	0.360 277
	DBSCAN	0.000 711	1.304 800	0.434 950
	k-means	7.544 556 e-05	1.881 950	0.480 370

针对 iris 数据集,利用 ECP 计算出的 p -value 与 f -score 的相关系数为−0.578 018,与 accuracy 的相关系数为−0.699 331。具体的结果如图 1。针对 wine 数据集,利用 ECP 计算得到的 p -value 与 f -score 的相系数为−0.535 734,与 accuracy 的相关系数为−0.538 754。具体的结果为图 2。对于 yeast 数据集,利用 ECP 计算得到的 p -value 与 f -score 的相关系数为−0.500 340,与 accuracy 的相关系数为−0.167 325。具体结果为图 3。

从实验结果可以看出用本文方法算出来的 p -value 是可靠的。需要注意的是 yeast 的数据集簇结构比较明显,聚类的结果比较集中。



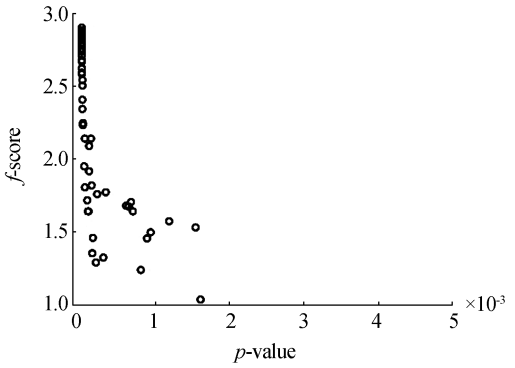
(a)p-value 与 f-score 的关系



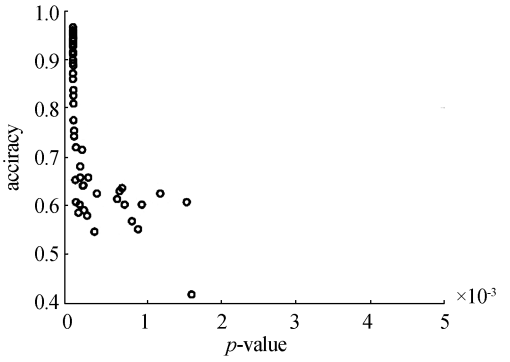
(b)p-value 与 accuracy 的关系

图 1　Iris 数据集 p-value 与 f-score 和 accuracy 的关系

Fig.1　The relationship between p-value and f-score, accuracy of iris dataset



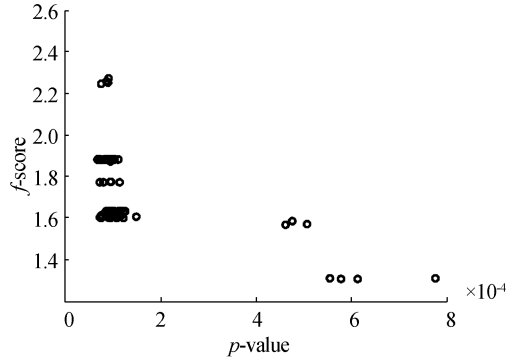
(a)p-value 与 f-score 的关系



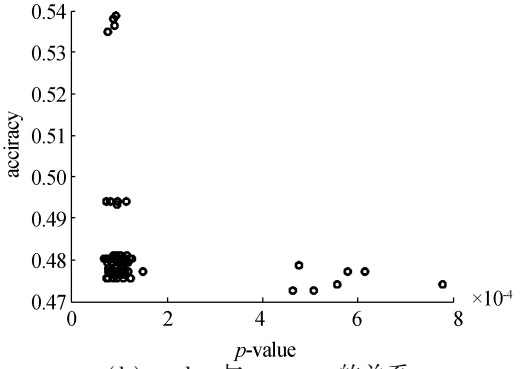
(b)p-value 与 accuracy 的关系

图 2 Wine 数据集 p-value 与 f-score 和 accuracy 的关系

Fig.2 The relationship between p-value and f-score, accuracy of wine dataset



(a)p-value 与 f-score 的关系



(b)p-value 与 accuracy 的关系

图 3 Yeast 数据集 p-value 与 f-score 和 accuracy 的关系

Fig.3 The relationship between p-value and f-score, accuracy of yeast dataset

4.2 利用 p -value 决定数据集簇的数目 k

很多聚类算法需要预先设定划分数目 k。本文研究了 p -value 与 k 的关系。对于同一数据集,选择不同的 k 用 k-means 分别聚类,然后计算对应的 p -value。计算结果如表 2。

从表 2 中看出随着 k 的增加, p -value 的值变小。因为 k 越大,对数据集划分得越细,同一个簇内的元素就会越相似, p -value 自然就会越小。然而划分的越细并不意味着就一定越好。举个极端的例子,将一个数据量为 n 的数据集划分成 n 个簇是毫无意义的。

本文研究了一种利用 p -value 的变化幅度来确定 k 的新方法。这里给出一个定义:

$$R(i)=\frac{p(i-1)}{p(i)},$$

式中:$p(i-1)$是当 k 取 $i-1$ 时聚类结果的 p -value,$p(i)$是当 k 取 i 时的聚类结果的 p -value。$R(i)$ 的意义是当 k 增加 1 时 p -value 的变化幅度。将表 2 的结果按照公式计算的结果如表 3。

由实验结果可以看出,对于 iris 数据集,当 k 取 3 的时候,$R(3)=$ 2.538 900 最大。事实上 iris 的类别数目就是 3。接着看 wine 数据集,当 i 取 3 的时候 $R(3)=$ 97.836 510 最大。真实情况 wine 的类别数目就是 3。对于 yeast 数据集当 i 取 4 的时候 $R(4)=$ 14.991 890 最大,以此来确定簇的数目为 4。而事实上 yeast 的类别数目就是 4。

利用本文提出的定义能正确算出数据集中的簇数目 k。因此可以说明计算聚类的 p -value 对于确定聚类数目 k 也是有一定意义的。不过对于 $R(i)$ 这个定义还存在一定的问题。根据 R 的定义,i 的取值不小于 3。因此对于簇数目为 2 的情况还不能够做出合适的处理。

表 2 不同 k 下的 p -value

Table 2 The p -value of clusters for different$k$

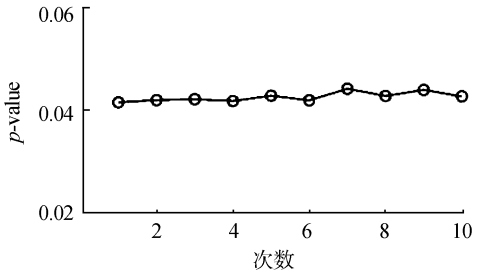
数据	2	3	4	5	6	7
Iris	0.108 518	0.042 742	0.020 435	0.017 261	0.006 991	0.003 208
Wine	0.001 946	1.988 773e-05	7.579 904e-07	2.381 891e-08	2.125 773e-09	1.537 855e-09
Yeast	0.006 911	0.001 040	6.937 873e-05	9.647 412e-06	1.327 582e-06	3.264 579e-06

表 3 不同 k 下的 $R(k)$

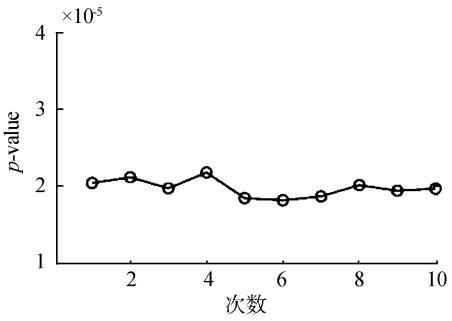
Table 3 The $R(k)$ of clusters for different$k$

数据	3	4	5	6	7
Iris	2.538 900	2.091 640	1.183 870	2.469 150	2.179 010
Wine	97.836 510	26.237 440	31.823 050	11.204 820	1.382 300
Yeast	6.644 860	14.991 890	7.191 430	7.266 900	0.406 660

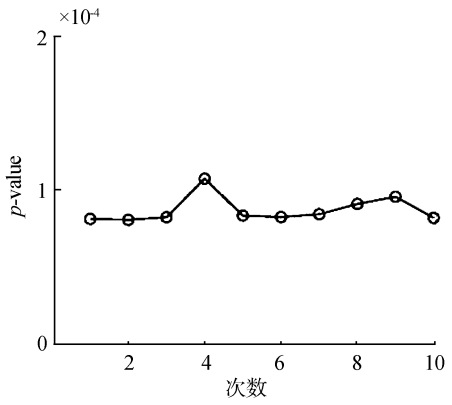
研究了对于 iris、wine 和 yeast 数据集需要多少样本能保证 p -value 不会因样本数目的增加而改变。对于每个数据集用不同数目样本计算 p -value,结果如图 5。



(a) Iris



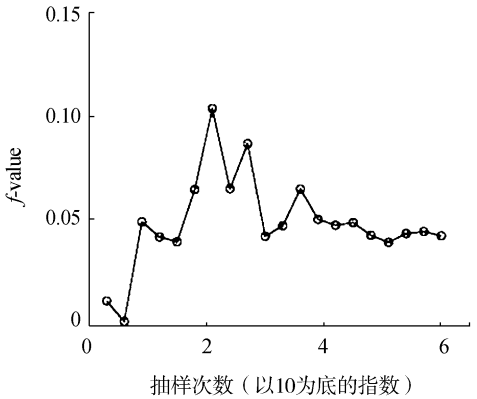
(b) Wine



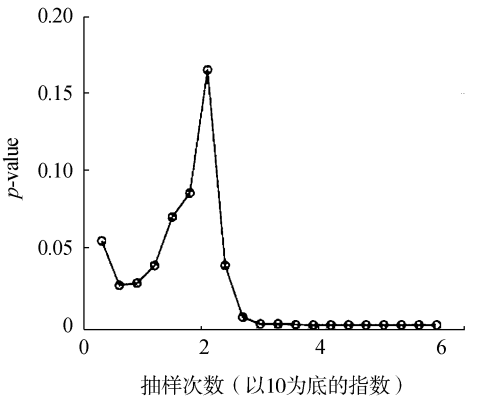
(c) Yeast

图 4　*p*-value 稳定性

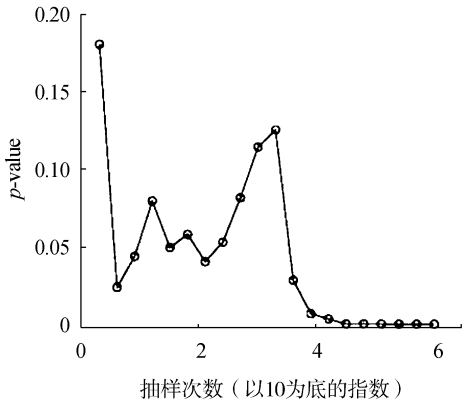
Fig.4　The stability of *p*-value



(a) Iris



(b) Wine



(c) Yeast

图 5　*p*-value 与抽样次数的关系

Fig.5　The relationship between *p*-value and the number of samples

实验最多抽取1 000 000个样本。对于这 3 个数据集,当抽样数目达 10 000 时 p -value 就基本稳定了。这一结果证实该方法具有很强的可行性。

4.3　与相关算法对比

4.3.1　ECP 与最大熵模型比较

本文重复了最大熵模型的评估方法,这 3 个数据集算出的 p -value 都为 $1/N$。这是因为样本太少,算法把原始聚类结果也当做一个样本。前文分析了这种做法的不合理性。利用 ECP 就可以避免这样的情况。除此之外,本文也尝试将最大熵方法的抽样评估值拟合出正态分布。实验结果如表 4。从实验结果可以看出,对于 wine 数据集,最大熵方法算出的 p-value 为 0.001,拟合正态后的 p-value 为 0.370 035 2。这两者差距比较大,这说明将最大熵方法拟合成正态分布是不合适的。这一实验说明利用 ECP 评估聚类结果更为可靠。

4.3.2　ECP 与 SigClust 对比

SigClust 算法是主要针对 k 为 2 聚类结果的评估。本文从每个数据集中选出了两类用 k-means 进行聚类(比如 iris 数据集中选出了 Setosa、Versicolour

两类进行对比)。为了让聚类质量有层次的差距,对 k-means 的聚类结果进行不同程度的破坏。破坏的程度越大,聚类的质量越差。实验结果如表 5。从实验看 SigClust 与 ECP 都能够区别出很好和很差的聚类。但是可以很明显地看出,SigClust 对聚类质量的区分度不够大。比如对于 iris 数据集计算的 f_1 为 2 和 1.8,SigClust 算出的 p-value 都是 0,没有区分开这 2 个不同划分的质量。同样地 iris 数据集 f_1 为 1.36 和1.158 65,SigClust 算出的 p-value 都为 1。实验可以看出 ECP 能很好地区分聚类质量的差距。因此,与 SigClust 相比,ECP 不仅能处理 $k>2$ 的情况,而且能更好地评估聚类质量。

表 4 ECP 与最大熵方法对比

Table 4 The comparison of ECP and maximum entropy method

算法	iris	wine	yeast
最大熵	0.001	0.001	0.001
最大熵拟合正态	4.891 817e-05	0.370 035 2	0.002 626 655
ECP	0.042 742 13	1.988 773e-05	6.937 873e-05

表 5 ECP 与 Sigclust 对比

Table 5 The comparison of ECP and Sigclust

数据	p-value/ECP	p-value/Sigclust Sigclust	f-score	accuracy
Iris	0.114 572 8	0	2	1
	0.121 688 1	0	1.8	0.9
	0.157 168 9	1	1.36	0.68
	0.228 296 5	1	1.158 65	0.58
wine	0.001 534 783	0	1.876 81	0.938 462
	0.002 878 496	0.199 2	1.673 66	0.838 462
	0.006 082 356	1	1.430 74	0.715 385
	0.221 656	1	1.011 64	0.546 154
yeast	0.006 761 993	0	1.130 05	0.567 265
	0.010 775 1	1	1.077 86	0.539 238
	0.012 549 87	1	1.073 48	0.536 996
	0.256 406 2	1	1.044 03	0.522 422

4.3.3 ECP 与 ECP1 对比

这一部分说明 ECP 比加速的 ECP1 在效率上有很大提高。ECP1 是未加速的 ECP 算法。本文将这两种算法进行了效率上的对比。实验结果如表 6。实验分别用两种算法抽样 100 000 次并得到对应的统计值。可以看出,对于 iris 数据集,ECP 比 ECP1 快了 60 倍。可见 ECP 在效率上有质的提升。

表 6 ECP 与 ECP1 效率对比

Table 6 The comparison of ECP and ECP1

算法	iris	wine	yeast
ECP1	18 s	50 s	56s
ECP	1109 s	734 s	280 m

5 结束语

本文提出了一种新的基于置换检验的聚类结果评估方法 ECP。为了增大抽样的数目,利用 DB-Index 的计算特点减小了对样本函数值计算的复杂度。为了得到更精确的 p -value,根据聚类划分的特点,假设了 DB-Index 的函数值是符合高斯分布的,进而可以用较少的抽样估出更为准确的 p -value。从实验的结果来看,ECP 对评估聚类结果有很好的效果,并且具有很强的实用性。

参考文献:

[1] TAN Pangning, STEINBACH M, KUMAR V. Introduction to data mining[M]. Boston: Addison-Wesley, 2005.

[2] HAN Jiawei, KAMBER M, PEI Jian. Data mining: concepts and techniques[M]. 3rd ed. Burlington, MA, USA: Elsevier, 2012: 1-33.

[3] 尹宏伟, 李凡长. 谱机器学习研究综述[J]. 计算机科学与探索, 2015, 9(12): 1409-1419.
YIN Hongwei, LI Fanzhang. Survey on spectral machine learning[J]. Journal of frontiers of computer science and technology, 2015, 9(12): 1409-1419.

[4] JAIN A K, MURTY M N, FLYNN P J. Data clustering: a review[J]. ACM computing surveys, 1999, 31(3): 264-323.

[5] WU Xindong, KUMAR V, QUINLAN J R, et al. Top 10 algorithms in data mining[J]. Knowledge and information systems, 2008, 14(1): 1-37.

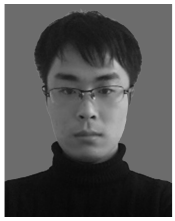
[6] HALKIDI M, BATISTAKIS Y, VAZIRGIANNIS M. On clustering validation techniques[J]. Journal of intelligent information systems, 2001, 17(2-3): 107-145.

[7] HANDL J, KNOWLES J, KELL D B. Computational cluster validation in post-genomic data analysis[J]. Bioinformatics, 2005, 21(15): 3201-3212.

[8] KONTONASIOS K N, VREEKEN J, DE BIE T. Maximum entropy modelling for assessing results on real-valued data [C]//Proceedings of the 11th international conference on data mining. Vancouver, BC, Canada, 2011: 350-359.

[9] OJALA M. Assessing data mining results on matrices with randomization[C]//Proceedings of international conference on data mining. Sydney, Australia, 2010: 959-964.
[10] OJALA M, VUOKKO N, KALLIO A, et al. Randomization methods for assessing data analysis results on real-valued matrices[J]. Statistical analysis and data mining, 2009, 2(4): 209-230.
[11] LIU Yufeng, HAYES D N, NOBEL A, et al. Statistical significance of clustering for high-dimension, low-sample size data[J]. Journal of the American statistical association, 2008, 103(483): 1281-1293.
[12] PARK P J, MANJOURIDES J, BOONETTI M, et al. A permutation test for determining significance of clusters with applications to spatial and gene expression data[J]. Computational statistics & data analysis, 2009, 53(12): 4290-4300.
[13] 张刚, 刘悦, 郭嘉丰, 等. 一种层次化的检索结果聚类方法[J]. 计算机研究与发展, 2008, 45(3): 542-547.
ZHANG Gang, LIU Yue, GUO Jiafeng, et al. A Hierarchical search result clustering method[J]. Journal of computer research and development, 2008, 45(3): 542-547.
[14] KNIJNENBURG T A, WESSELS L F A, REINDERS M J T, et al. Fewer permutations, more accurate p-values[J]. Bioinformatics, 2009, 25(12): i161-i168.
[15] ASUNCION A, NEWMAN D J. UCI machine learning repository[EB/OL]. 2007. http://archive.ics.uci.edu/ml/.

作者简介:



谷飞洋,男,1991 年生,硕士研究生,主要研究方向是数据挖掘和生物信息。

田博,女,1992 年生,硕士研究生,主要研究方向为数据挖掘和生物信息。

何增有,男,1976 年生,副教授,主要研究方向为数据挖掘和生物信息学,学术论文均发表在该领域的顶级期刊或会议上,出版学术专著 1 部。

2016 年第九届 SPIE 机器学习国际会议
2016 The 9th International Conference on Machine Vision (ICMV 2016)

Welcome to the official website for 2016 The 9th International Conference on Machine Vision (ICMV 2016). ICMV conference is initiated by School of Electronics, Si Chuan University, China, assisted by Halmstad University, Sweden, University of Barcelona, Spain. This is the annual conference started in 2007(Islamabad,Pakistan),ICMV 2009 (Dubai, UAE), ICMV 2010 (Hong Kong), ICMV 2011 (Singapore), ICMV 2012 (Wuhan, China), ICMV 2013 (London, UK),ICMV 2014 (Milano, Italy), ICMV 2015 (Barcelona, Spain). ICMV 2016 will take place in Nice, France during November 18-20,2016, the conference chairs are Prof. Antanas Verikas, Halmstad University, Sweden, Prof. Petia Radeva, University of Barcelona, Spain and Prof. Dmitry Nikolaev, Russian Academy of Science, Russia.

The emergence of Machine Vision as a ubiquitous platform for innovations has laid the foundation for the rapid growth of the Information. Side-by-side, the use of mobile and wireless devices such as PDA, laptop, and cell phones for accessing the Internet has paved the ways for related technologies to flourish through recent developments. In addition, the Machine Vision Technology is promoting better integration of the digital world with physical environment. This conference serves to foster communication among researchers and practitioners working in a wide variety of scientific areas with a common interest in improving Machine Vision related techniques.

High quality, original papers are solicited in all areas of Machine Vision. The final program will be the result of a highly selective review process designed to include the best work of its kind in every category. The program will include invited talks as well as oral and poster presentations of refereed papers.

Website: http://www.icmv.org/index.html