聚类算法对比研究

计算机学院 大数据与人工智能 21215122 何峙

摘 要 聚类是一种广泛应用于数据挖掘的算法。现实中的很多数据的类别标记可能是未知的，要揭示数据蕴含的性质和内在规律，则需要通过一定的数据分析技术，这一般称为“无监督学习”技术。而聚类则是其中占用重要地位的技术，其研究最多，应用最广，衍生出许许多多的算法。每种算法分为多种类别以适用在不同场景下。本文通过对这些聚类算法之间的对比阐述，让读者了解各种聚类算法的效用，以明晰其在特定场景下优缺点。

关键词 聚类，数据挖掘，无监督学习

1. 引言

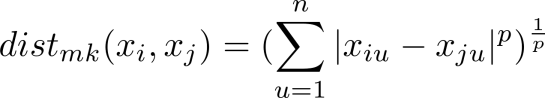
聚类是一种重要的无监督学习技术，它可以将没有类别标记的数据集划分为若干个互不相相交的子集，从而使数据样本分别位于不同的“簇”当中，这些“簇”也就是互不相交的类别，使得数据可进一步用于下游的数据分析任务。例如在一些图片应用中需要根据用户照片的内容进行类别划分，譬如海滩的照片归为一类，山脉的照片归为另一类，等。但用户并没有手工划分这些照片的类别，此时往往先对这些照片进行特征提取，根据图片特征进行聚类，再根据聚类结果将每个簇定义为一个类别，然后在基于这些类别训练模型，以进行下一步的图片细粒度识别等应用功能。这时，往往会考虑：使用哪种聚类算法更适合于特定场景？还有，如何衡量这些聚类算法的性能好坏？接下来本文将从如下几个方面进行相关阐述：

* 第二章阐述聚类算法所使用的性能度量，说明数据的相似度等概念。
* 第三章列举三种不同场景下常用的聚类算法概念及其设计思想。
* 第四章在传统的聚类算法基础上，展示某些前沿的聚类算法研究。
* 第五章聚类算法总结。

1. 性能度量

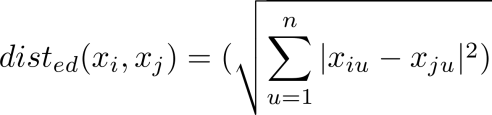
聚类算法的好坏直接影响着下游数据分析任务的优劣，那么如何衡量聚类算法的性能就相当关键。一般来说，聚类的核心性能指标是：使簇内相似度尽可能高，簇间相似度尽可能低。相似度有的文献称为“数据距离”，是一种衡量数据之间距离的度量。假设 wpsoffice 是两个样本对应的向量，以下列举的聚类中常用的相似度计算方法：

* 闵可夫斯基距离：



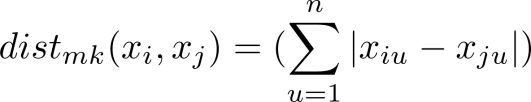
* 欧氏距离：

当闵可夫斯基距离的p=2时，即为欧氏距离：

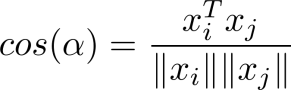


* 曼哈顿距离：

当闵可夫斯基距离的p=1时，即为曼哈顿距离：

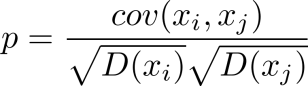


* 余弦相似度：



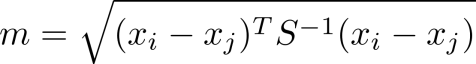
计算的结果是两个向量之间的夹角。所以值越小，可认为两个向量相似度越低。否则越大。

* 皮尔逊相关系数：



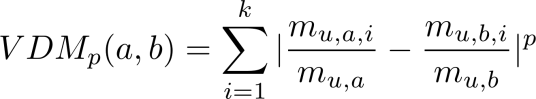
其中 wpsoffice 是两个向量的协方差，wpsoffice 是向量的方差。皮尔逊相关系数的取值范围在[-1,1]之间，值越趋于0表面向量越不相关，越趋于-1表明向量越负相关，而越趋于1明白向量越正相关。

* 马氏距离：



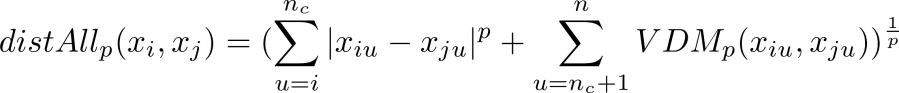
其中S为两个向量对应的协方差矩阵。

以上这些是常用于连续的属性，或者更确切的称为“有序的属性”。而对于样本中离散的、无序的属性，譬如属性的取值为学科类别={物理，化学，法律，经济}，又该如何衡量它们之间的相似度？可以考虑采用VDM数值（Value Difference Metric【hoho引用1】）衡量：



其中 /private/var/folders/4b/b2hj0zj524lbbtvttkfwrkbw0000gp/T/com.kingsoft.wpsoffice.mac/wpsoffice.akprRhwpsoffice表示在属性u上取值为a的样本数，wpsoffice 表示在第i个簇中在属性u上取值为a的样本数。

这时，可以将闵可夫斯基距离和VDM相结合，把含有有序属性和无序属性的样本相似度计算统一起来，如下计算：



有了如上这些距离度量，就可以进行性能度量。

性能度量指标又分为外部指标和内部指标两大类。外部指标是将聚类结果跟某个参考模型相比较，譬如人类专家定义的参考模型。内部指标是不利用外部参考模型而直接考虑聚类结果。

1. 外部指标

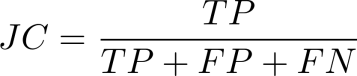
由于有外部参考模型的介入，聚类算法将各样本划分到对应簇的情况可能根外部参考模型不一致。先定义表1的混淆矩阵以方便说明问题。

Table 1 Definition of Confusion Matrix

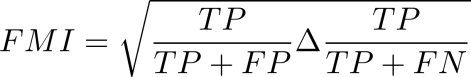
|  |  |  |  |
| --- | --- | --- | --- |
|  | Acutal Value | | |
| Predicted Value |  | Positive | Negative |
| Positive | TP | FP |
| Negative | FN | TN |

则有如下外部衡量指标：

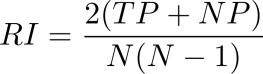
* Jaccard指数，也称Jaccard相似度：



* FM指数：



* Rand指数：



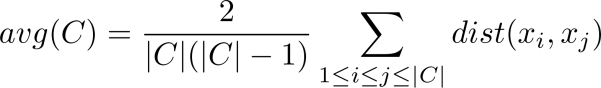
其中N为样本的数量。

显然上述性能度量的结果均在[0,1]区间，取值越大越好。

1. 内部指标

假设聚类结果的簇划分为 wpsoffice ,先定义如下计算量：

* 簇C内样本的平均距离：



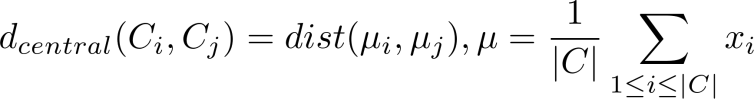
* 簇C内样本间的最远距离

wpsoffice

* 两个簇最近样本间的距离

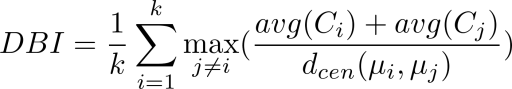


* 两个簇中心点的距离



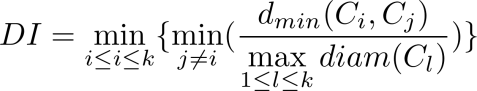
由此可导出以下常用的内部指标：

* DB指数：



显然值越小越好。

* Dunn指数



Dunn指数则相反，值越大越好。

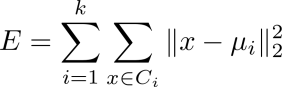
1. 聚类算法

聚类算法种类有很多，以下将根据数据聚类进行中的组织方式回顾四四大类常用的聚类算法。

1. 基于划分的聚类

这类聚类算法一般采用贪心策略，通过定义一个最优化的目标函数，通过迭代的方式使样本点不断归类于这K个代表点所属的类别，同时逐步提高聚类的效果。

K-means【hoho引用2】是其中的典型代表。其目标函数是最小化所有样本点的平方误差：



算法首先需要指定要划分的簇类数量K，随机选择K个代表点作为初始向量，然后不断迭代的判断每个样本点分别与这些代表点的距离，把与距离小的代表点所属的类别分配给迭代中的样本点，并且每迭代一次，重新计算代表点为其所归属的样本点的中心点为新的代表点，一直迭代到所有样本点均未更新为止。所以，从图1可见，K-means是一个不断将样本聚集在最佳簇中心点的过程。K-means的优点有：欧氏距离容易计算、算法不依赖与数据的先后排序，等。其缺点也是相当明显，主要有：（1）聚类的数量要预先指定，但是又难以指定合适的簇类数量K；（2）对异常点敏感；（3）对于不能线性可分的数据效果较差。这些通常也是基于划分的聚类算法的固有缺点。

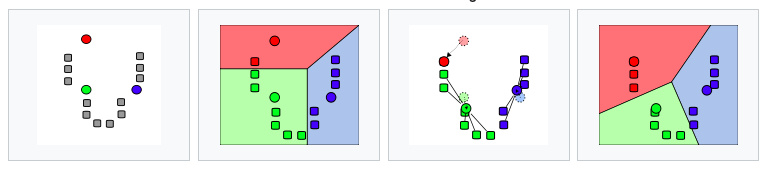
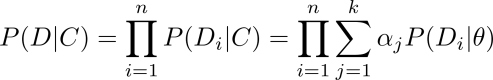


Fig.1. Procedure of K-means, from left to right. (picutre by wikipedia).

K-mediods【hoho引用3】是K-means算法的一个改进，其与K-means的最大区别是选择中心点的方法不同：K-mediods每次选取的中心点必须是样本点，而K-means选取的中心点可以是样本点以外的点。K-mediods选取中心点时，需要遍历簇中所有的点，在前簇中选择所有其他点到中心点距离之和最小的点为中心点。通过这样的方式K-mediods可以大大减少异常点对聚类结果影响。

还有一种常用的基于概率统计的聚类方法。可以假定数据空间中有一个隐藏的类别分布，分布的类别数量为K，数据样本就是通过这K个类别即簇产生的：先以一定概率选取一个簇，然后从簇中以一定的概率选取一个样本。所以接下来就是计算这K个簇和它们以概率产生数据集的似然。假定样本是独立采样，因此对于数据集 wpsoffice有：



其中 wpsoffice 是选择每个簇的概率， 所以选择K个簇的总概率为1，即 ，wpsoffice 是指定概率分布的参数，譬如实际应用中可选择高斯分布，则参数为n维均值向量和n x n阶的协方差矩阵。进行参数估计可使用EM算法【hoho引用4】或其他方法。得出参数后即可通过后验概率 wpsoffice 得出样本所属的簇类。

表2总结了基于划分的聚类算法的时间复杂度，其中GMM为高斯混合分布聚类算法，k表示聚类的数量，t为迭代次数，n为样本数量。

Table.2. Time complexity for partition alorithm

|  |  |  |
| --- | --- | --- |
| K-means | K-mediods | GMM |
| O(k\*t\*n) | O(k(n-k)2) | O(k\*t\*n2) |

1. 基于密度的聚类

这类算法以样本之间的紧密程度如可连接性为考察重点，基于可连接性将样本不断扩展聚类簇以获得最终的聚类效果。其中DBSCAN【hoho应用5】是此聚类算法典型代表。

DBSCAN需要预定参数（wpsoffice, MinPts），首先定义如下几个概念：

* wpsoffice-邻域：wpsoffice；
* 核心对象：若xj的wpsoffice-邻域至少包含MinPts个样本，则xj是一个核心对象；
* 密度直达：若xj位于xi的wpsoffice-邻域中，且xi是核心对象，则称xj由xi密度直达；
* 密度可达：对于xi与xj，若存在样本序列p1,p2,...,pn，其中p1=xi，pn=xj，且pi+1有pi密度可达，则称xj由xi密度可达；
* 密度相连：对xi与xj，若存在xk使得xi与xj均由xk密度可达，则称xi与xj密度相连。

基于以上定义，具体来说，该算法根据给定的参数（wpsoffice，MinPts）找出所有核心对象，以任一核心对象作为出发点，找出由其密度可达的样本生成聚类簇，不断迭代至所有核心对象都被访问过为止。如图2所示，该算法可看作是一个由种子点（即核心对象）不断扩大连通区域以形成簇的过程。DBSCAN的优点主要是其可用于不能线性可分，其聚类效果一般更加优秀，而且不需要指点聚类的簇数量。但其需要指定两个难以适用于样本数据集参数（wpsoffice，MinPts），甚至数据集的不同部分可能需要指定不同的参数以达到良好的聚类效果，所以其对于参数过于敏感。



Fig.2. Procedure of DBSCAN.(picture by wikipedia)

OPTICS【hoho引用6】是DBSCAN的改进算法，其也需要相同的两个参数（wpsoffice，MinPts），但对wpsoffice不再敏感，只要确定MinPts，wpsoffice 的轻微变化并不影响聚类结果。除了继承DBSCAN的定义，OPTICS还多了两个定义：

* 核心距离：是使一个样本点成为核心点（即核心对象）的最小半径，在给定邻域半径 wpsoffice 和MinPts参数的前提下，核心距离可以比给定的 wpsoffice 更小；
* 可达距离：样本点到核心点的距离。

其算法过程如下：

1. 创建两个队列：待处理队列用于存储核心点及其密度直达的点, 并按可达距离升序排列；结果队列用于存储样本点的输出次序。结果队列中的点为处理之后的样本；
2. 选取一个未处理的核心点, 将其放入结果队列，同时计算邻域内样本点的可达距离，按照可达距离升序将样本点依次放入待处理队列。
3. 若待处理队列为空，则回到步骤（2）以重新选取处理数据。否则，从待处理队列中提取第一个样本，如果为核心点, 则计算可达距离，将可达距离最小的点放入结果队列。如果不是核心点 则重复步骤（2）；如果待处理队列中已经存在直接密度可达点，如果此时新的可达距离小于旧的可达距离，则用新可达距离取代旧可达距离，待处理队列重新排序（因为一个对象可能有多个核心对象可达）；如果待处理队列中不存在该直接密度可达样本点，则插入该点，并对有序队列重新排序；
4. 迭代（2）、（3），直到数据集中所有点都处理完成，则算法结束，输出结果队列中有序样本点。

OPTICS不显式产生聚类簇，而是生成一个簇的排序，这个排序表示了个样本点基于密度的聚类结构。

DBSCAN与OPTICS的时间复杂度均为O(n\*logn)，属于比较高效的算法。

1. 基于层次的聚类

todo

1. 前景与展望

Todo

1. 总结

Todo

参考文献