

"练习一万小时成天才!" - 摘自《异类》

昵称: bourneli

园龄: 4年1个月

粉丝: 75

关注: 9

+加关注

< 2015年12月 >						
日	一	二	三	四	五	六
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

搜索

谷歌搜索

- 最新随笔
1. Spark随机深林扩展—OOB错误评估和变量权重

2. Spark随机森林实现学习

3. RDD分区2GB限制

4. Spark使用总结与分享

5. Spark核心—RDD初探

6. 机器学习技法--学习笔记04--Soft SVM

7. 机器学习技法--学习笔记03--Kernel技巧

8. 机器学习基石--学习笔记02--Hard Dual SVM

9. 机器学习基石--学习笔记01--linear hard SVM

10. 特征工程(Feature Enginnerin g)学习记要

- 我的标签
- coursera(4)

C/C++(3)

kmeans聚类理论篇

前言

kmeans是最简单的聚类算法之一，但是运用十分广泛。最近在工作中也经常遇到这个算法。kmeans一般在数据分析前期使用，选取适当的k，将数据分类后，然后分类研究不同聚类下数据的特点。

本文记录学习kmeans算法相关的内容，包括算法原理，收敛性，效果评估聚，最后带上R语言的例子，作为备忘。

- 算法原理
- kmeans的计算方法如下：
- 1 随机选取k个中心点

2 遍历所有数据，将每个数据划分到最近的中心点中

3 计算每个聚类的平均值，并作为新的中心点

4 重复2-3，直到这k个中线点不再变化（收敛了），或执行了足够多的迭代
- 时间复杂度：O(I*n*k*m)
- 空间复杂度：O(n*m)

其中m为每个元素字段个数，n为数据量，I为跌打个数。一般I,k,m均可认为是常量，所以时间和空间复杂度可以简化为O(n)，即线性的。

算法收敛

从kmeans的算法可以发现，SSE其实是一个严格的坐标下降（Coordinate Decendet）过程。设目标函数SSE如下：

$SSE(c_1, c_2, ..., c_k) = \sum (x - c)^2$

采用欧式距离作为变量之间的聚类函数。每次朝一个变量 c_i 的方向找到最优解，也就是求偏倒数，然后等于0，可得

$c_i = \frac{1}{m} \sum x$ 其中m是 c_i 所在的簇的元素个数

也就是当前聚类的均值就是当前方向的最优解（最小值），这与kmeans的每一次迭代过程一样。所以，这样保证SSE每一次迭代时，都会减小，最终使SSE收敛。

由于SSE是一个非凸函数（non-convex function），所以SSE不能保证找到全局最优解，只能确保局部最优解。但是可以重复执行几次kmeans，选取SSE最小的一次作为最终的聚类结果。

0-1规格化

由于数据之间量纲的不相同，不方便比较。举个例子，比如游戏用户的在线时长和活跃天数，前者单位是秒，数值一般都是几千，而后者单位是天，数值一般在个位或十位，如果用这两个变量来表征用户的活跃情况，显然活跃天数的作用基本上可以忽略。所以，需要将数据统一放到0~1的范围，将其转化为无量纲的纯数值，便于不同单位或量级的指标能够进行比较和加权。具体计算方法如下：

$v_i = \frac{v_i - \min(A)}{\max(A) - \min(A)}$

其中 v_i 属于A。

轮廓系数

轮廓系数（Silhouette Coefficient）结合了聚类的凝聚度（Cohesion）和分离度（Separation），用于评估聚类效果。该值处于-1~1之间，值越大，表示聚类效果越好。具体计算方法如下：

决策树(2)
data analysis(2)
dlopen(1)
jsoncpp(1)
k fold(1)
MapReduce(1)
MOOC(1)
singleton(1)
更多

随笔分类(92)
C/C++(4)
R(7)
Web前端开发(26)
大数据(7)
机器学习(6)
数据分析&挖掘(42)

随笔档案(119)
2015年5月 (2)
2015年4月 (2)
2015年3月 (1)
2015年1月 (4)
2014年11月 (1)
2014年9月 (1)
2014年8月 (1)
2014年4月 (1)
2014年3月 (1)
2013年10月 (1)
2013年9月 (1)
2013年8月 (3)
2013年7月 (1)
2013年6月 (1)
2013年4月 (3)

1. 对于第*i*个元素 x_i ，计算 x_i 与其同一个簇内的所有其他元素距离的平均值，记作 a_i ，用于量化簇内的凝聚度。
2. 选取 x_i 外的一个簇**b**，计算 x_i 与**b**中所有点的平均距离，遍历所有其他簇，找到最近的这个平均距离,记作 b_i ，用于量化簇之间分离度。
3. 对于元素 x_i ，轮廓系数 $s_i = (b_i - a_i) / \max(a_i, b_i)$
4. 计算所有*x*的轮廓系数，求出平均值即为当前聚类的整体轮廓系数

从上面的公式，不难发现若 s_i 小于0，说明 x_i 与其簇内元素的平均距离小于最近的其他簇，表示聚类效果不好。如果 a_i 趋于0，或者 b_i 足够大，那么 s_i 趋近与1，说明聚类效果比较好。

K值选取

在实际应用中，由于Kmean一般作为数据预处理，或者用于辅助分类贴标签。所以k一般不会设置很大。可以通过枚举，令k从2到一个固定值如10，在每个k值上重复运行数次kmeans(避免局部最优解)，并计算当前k的平均轮廓系数，最后选取轮廓系数最大的值对应的k作为最终的集群数目。

实际应用

下面通过例子（R实现，完整代码见附件）讲解kmeans使用方法，会将上面提到的内容全部串起来

```
1 library(fpc) # install.packages("fpc")
2 data(iris)
3 head(iris)
```

加载实验数据iris，这个数据在机器学习领域使用比较频繁，主要是通过画的几个部分的大小，对花的品种分类，实验中需要使用fpc库估计轮廓系数，如果没有可以通过install.packages安装。

```
1 # 0-1 正规化数据
2 min.max.norm <- function(x){
3   (x-min(x))/(max(x)-min(x))
4 }
5 raw.data <- iris[,1:4]
6 norm.data <- data.frame(s1 = min.max.norm(raw.data[,1]),
7                           sw = min.max.norm(raw.data[,2]),
8                           pl = min.max.norm(raw.data[,3]),
9                           pw = min.max.norm(raw.data[,4]))
```

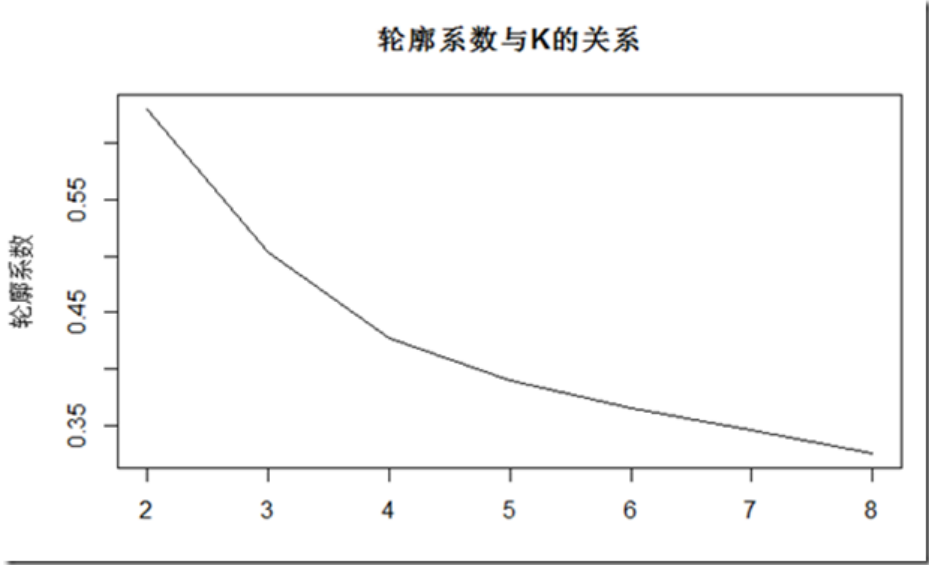
对iris的4个feature做数据正规化，每个feature均是花的某个不为的尺寸。

```
1 # k取2到8，评估K
2 K <- 2:8
3 round <- 30 # 每次迭代30次，避免局部最优
4 rst <- sapply(K, function(i){
5   print(paste("K=",i))
6   mean(sapply(1:round,function(r){
7     print(paste("Round",r))
8     result <- kmeans(norm.data, i)
9     stats <- cluster.stats(dist(norm.data), result$cluster)
10    stats$avg.silwidth
11  })))
12 })
13 plot(K,rst,type='l',main='轮廓系数与K的关系', ylab='轮廓系数')
```

评估k，由于一般K不会太大，太大了也不易于理解，所以遍历K为2到8。由于kmeans具有一定随机性，并不是每次都收敛到全局最小，所以针对每一个k值，重复执行30次，取并计算轮廓系数，最终取平均作为最终评价标准，可以看到如下的示意图，

2013年3月 (4)
2013年2月 (1)
2013年1月 (4)
2012年12月 (4)
2012年11月 (17)
2012年10月 (12)
2012年9月 (10)
2012年8月 (5)
2012年7月 (4)
2012年6月 (3)
2012年5月 (7)
2012年4月 (10)
2012年3月 (1)
2012年2月 (3)
2012年1月 (4)
2011年12月 (6)

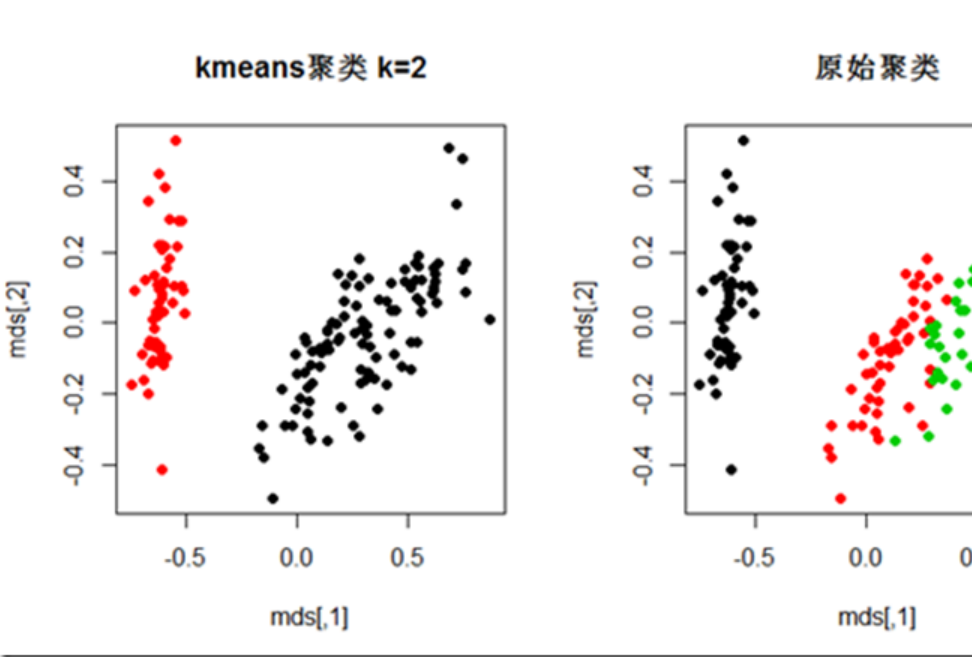
文章分类(23)
apache
C/C++
gbk(1)
gtest&gmock
js(2)
LAMP(2)
Linux(4)
mysql(3)
php(2)
shell(4)
单元测试
多进程(1)
工作感悟(1)
开源软件
设计模式



当k取2时，有最大的轮廓系数，虽然实际上有3个种类。

```
1 # 降纬度观察
2 old.par <- par(mfrow = c(1,2))
3 k = 2 # 根据上面的评估 k=2最优
4 clu <- kmeans(norm.data,k)
5 mds = cmdscale(dist(norm.data,method="euclidean"))
6 plot(mds, col=clu$cluster, main='kmeans聚类 k=2', pch = 19)
7 plot(mds, col=iris$Species, main='原始聚类', pch = 19)
8 par(old.par)
```

聚类完成后，有源原始数据是4纬，无法可视化，所以通过多维定标(Multidimensional scaling)将纬度将至2为，查看聚类效果，如下



可以发现原始分类中和聚类中左边那一簇的效果还是拟合的很好的，右侧原始数据就连在一起，kmeans无法很好的区分，需要寻求其他方法。

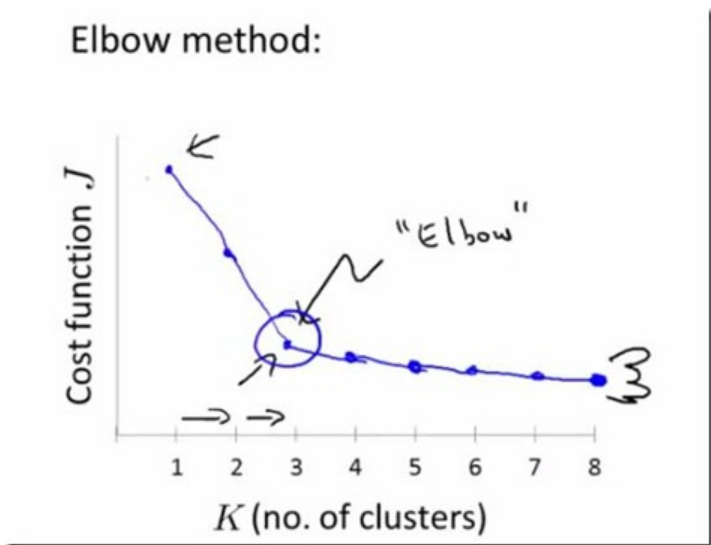
kmeans最佳实践

1. 随机选取训练数据中的k个点作为起始点
2. 当k值选定后，随机计算n次，取得到最小开销函数值的k作为最终聚类结果，避免随机引起的局部最优解
3. 手肘法选取k值：绘制出k--开销函数闪点图，看到有明显拐点（如下）的地方，设为k值，可以结合

数据库事务
字符编码(3)
文章档案(17)
2013年8月 (1)
2013年1月 (1)
2012年10月 (2)
2012年9月 (2)
2012年8月 (3)
2012年7月 (5)
2012年6月 (2)
2012年4月 (1)
友情链接
R博客（英文）
R中文网
TOWER -- 思想，智慧，人生
DM, Hadoop
数据科学与R语言
统计之都
统计学中文论坛
小虫织网
郑纪blog
积分与排名
积分 - 155984
排名 - 1049
最新评论
1. Re:Spark使用总结与分享
楼主，spark sql 通过自定义bean来实现创建表，需要序列化，而这个序列化应该是十分消耗内存的，有没有什么方法解决。
--晓枫
2. Re:MySQL全文检索初探
@conqweal是分词，英文可以按

轮廓系数。

4. k值有时候需要根据应用场景选取，而不能完全的依据评估参数选取。




参考

- [1] [kmeans 讲义by Andrew NG](#)
- [2] [坐标下降法 \(Coordinate Decendent\)](#)
- [3] [数据规格化](#)
- [4] [维基百科--轮廓系数](#)
- [5] [kmeans算法介绍](#)
- [6] [降为方法一多维定标](#)
- [7] [Week 8 in Machine Learning, by Andrew NG, Coursera](#)

声明：如有转载本博文章，请注明出处。您的支持是我的动力！文章部分内容来自互联网，本人不负任何法律责任。

分类: [数据分析&挖掘](#), [R](#)



 [bourneli](#)
关注 - 9
粉丝 - 75

[±加关注](#)

2 0

(请您对文章做出评价)

« 上一篇: [PCA主成份分析学习记要](#)
» 下一篇: [R绘制3D散点图](#)

posted @ 2014-04-04 13:59 bourneli 阅读(10544) 评论(1) 编辑 收藏

评论列表

#1楼 2015-09-24 19:58 fanfan123

请问附件在哪里呢？谢谢

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

空格分词，中文就比较麻烦了...

--抠脚大汉

3. Re:kmeans聚类理论篇

请问附件在哪里呢？谢谢

--fanfan123

4. Re:Linux上使用gtest

亲测可行！

--www.点esoso.点com

5. Re:数据挖掘学习08 - 实验：使用R评估kmeans聚类的最优K

谢谢博主！

这个方法挺好的，回头也找这本书看看

不过# 遍历计算kmeans的SSE这里，pam的运行速度好慢，是kmeans的好几倍。。。

--尾巴AR

阅读排行榜

1. 决策树学习笔记整理(42554)

2. kmeans聚类理论篇(10543)

3. Django静态文件配置(9544)

4. PHP判断键值数组是否存在，使用empty或isset或array_key_exists(8462)

5. Apache alias目录配置(7544)

评论排行榜

1. PHP多进程处理并行处理任务实例(5)

2. 数据挖掘学习05 - 使用R对文本进行hierarchical cluster并验证结果(4)

3. Linux上使用gtest(4)

4. PHPUnit学习05---Mock使用进阶(3)

5. MySQL全文检索初探(3)

推荐排行榜

1. 决策树学习笔记整理(6)

2. 5分钟回忆正则表达式(3)

【推荐】极光推送30多万开发者的选择，SDK接入量超过30亿了，你还没注册？

【精品】高性能阿里云服务器+SSD云盘，支撑I/O密集型核心业务、极高数据可靠性



最新IT新闻：

- 作为员工，如何识别初创企业健康状况
 - 如何招到靠谱的产品经理？
 - 创业跟风者的15项特征
 - 在网上没人知道你是一条狗的时候，你会怎么做？
 - 看完豆瓣读书这份年度榜单，才知道今年错过了多少好书
- » 更多新闻...

JavaWeb教程，进大公司的捷径

阿里巴巴、京东、滴滴...只有大公司才能用的起的JavaWeb人才



最新知识库文章：

- Git协作流程
 - 企业计算的终结
 - 软件开发的核心理解
 - Linux概念架构的理解
 - 从涂鸦到发布——理解API的设计过程
- » 更多知识库文章...

历史上的今天：

2013-04-04 利用Minhash和LSH寻找相似的集合

3. epoll学习(2)

4. Spark核心—RDD初探(2)

5. kmeans聚类理论篇(2)