

## 门店分组

笔记本： 算法

创建时间： 2020/8/10 13:01

更新时间： 2020/8/10 13:55

作者： 190072358@qq.com

---

## 1、背景说明

将群组管理的思想应用到计算门店配置、试销、铺货、补货等所有流程。基于以上目的，需要算法给出合理的门店分组方案。

## 2、模型设计

基于不同的分组场景，设计两种分组方案：

**分组a：按人流量相近对门店分组，适用于量的计算。**

对于人流量，可以用门店面积、销售利润等级等衡量。

**分组b：按客群相似对门店分组，适用于款的计算。**

对于客群，可以用商圈、爆款相似度等衡量

## 3、算法设计-分组b

如何衡量客群相似？可以通过两家店爆卖款重叠度衡量相似性。

### 3.1 相似门店计算的原理

如果A店的爆款推给B店，且在B店也是爆款，则认为A, B两店是相似的。

### 3.2 计算公式

A, B店的相似度=A, B店爆款的交集/A店爆款和B店的交集：

$$r_{A,B} = \frac{A_{bao} \cap B_{bao}}{A_{bao} \cup B_{bao}}$$

其中， $r_{A,B}$ 表示A, B店的相似度， $A_{bao}$ 表示A店爆款， $B_{bao}$ 表示B店爆款，A表示A店卖的所有款，B表示B店卖的所有款。

### 可以改进的优化点

在上面的公式中，每个款都是占有相同的权重（都为1）。设想对于热销款，如款001，它在全国大部分门店中都为爆款，那么如果在A, B店的爆款中都有款001，这并不能用来证明A, B店的相似性，因为该款在很多门店都是爆款。故我们应对每个款赋予不同的权重，越是热销款，权重越小。

### 加权版相似门店计算公式

$$A_{bao} \cap B_{bao} = \sum_{i \in A_{bao} \cap B_{bao}} w_i$$

$$r_{A,B} = \frac{A_{bao} \cap B_{bao}}{A_{bao} \cap B}$$

改进的相似门店计算，对每个款赋予不同的权重，款  $i$  属于  $A, B$  店爆款的交集， $w_i$  表示款  $i$  的权重。

**权重的计算公式**

$$w_i = \frac{1}{\log_2 \sum_{k=1}^n i \in A_{bao}^k}$$

$i \in A_{bao}^k$  表示款  $i$  在门店  $A^k$  中曾经被标记为爆款。 $n$  为门店总数。只要款  $i$  在查询时间段内，曾经成为门店爆款，则记 1，求和，并对其求对数倒数，即为款  $i$  的权重。该公式可以理解为，如果款 001 只在  $A, B$  两店被标记为爆款，则款 001 的权重为 1，如果款 001 在 64 家门店被标记为爆款，则权重为 1/6。这样可以保证热销款权重会较小。

### 3.3 改编版kmeans算法

相似度  $r_{A,B}$  的计算公式确定后，距离公式  $d_{A,B} = 1/r_{A,B}$ ，然后通过改编版kmeans算法对门店聚类。

改编版Kmeans聚类，不是直接计算类的均值作为中心点，而是遍历类中的每个点，选择其他点到该点的平均距离最小的点作为中心点。即k-means的中心点是欧式空间中任意一个连续的值，不一定在数据点中，但kmedoids是选择类中的点，该点一定在数据点中。k-means与k-mediods相当于均值与中位数的区别。

kmeans算法需要指定k值，即组数。这里我们不直接指定k值，把它当作模型的参数，通过交叉验证，取轮廓系数最大时的k值。

### 3.4 聚类评价指标-轮廓系数

对于单个样本，设a是与它同类别中其他样本的平均距离，b是与它距离最近不同类别中样本的平均距离，轮廓系数为：

$$s = \frac{b - a}{\max(a, b)}$$

对于一个样本集合，它的轮廓系数是所有样本轮廓系数的平均值。分值越大，聚类效果越好。