聚类算法-Kmeans

OutLine

- 聚类基本知识
- 层次聚类法
- Kmeans聚类
- 【实践】基于MLlib的Kmeans聚类

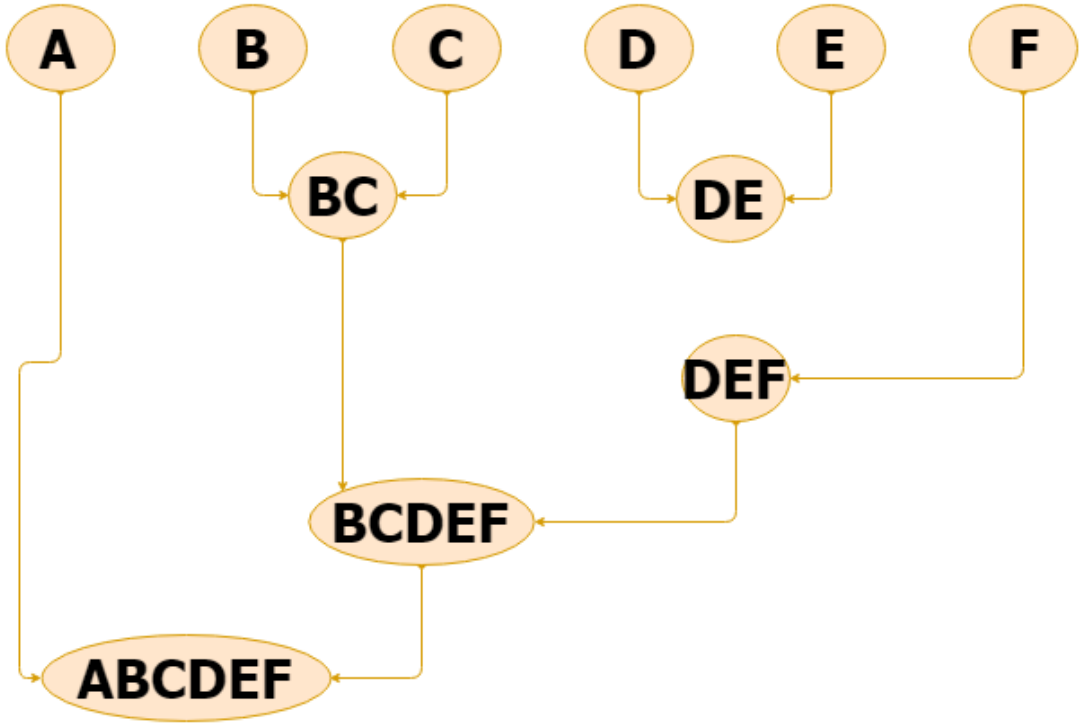
基础知识

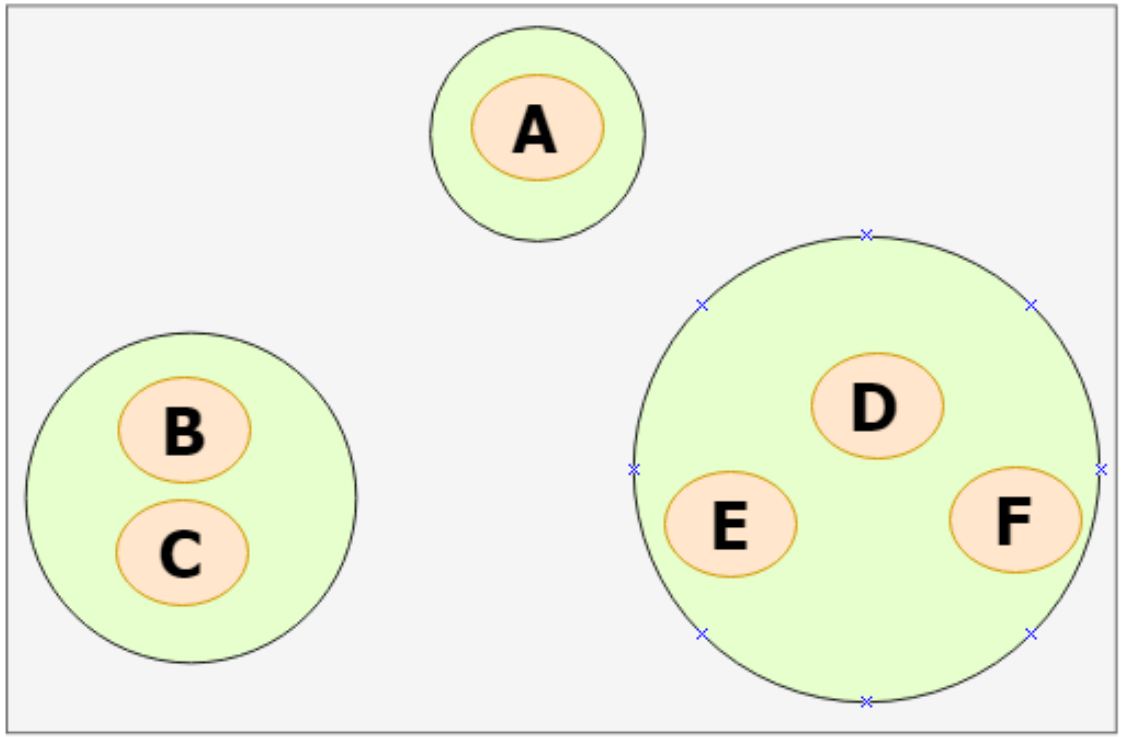
- 将数据划分到不同的类里，使相似的数据在同一类里，不相似的数据在不同类里
- 无监督学习

- 应用：文本聚类、图像聚类和商品聚类
  - 便于发现规律，以解决数据稀疏问题

基 础 知 识

- 层次聚类 vs. 非层次聚类
  - 不同类之间有无包含关系



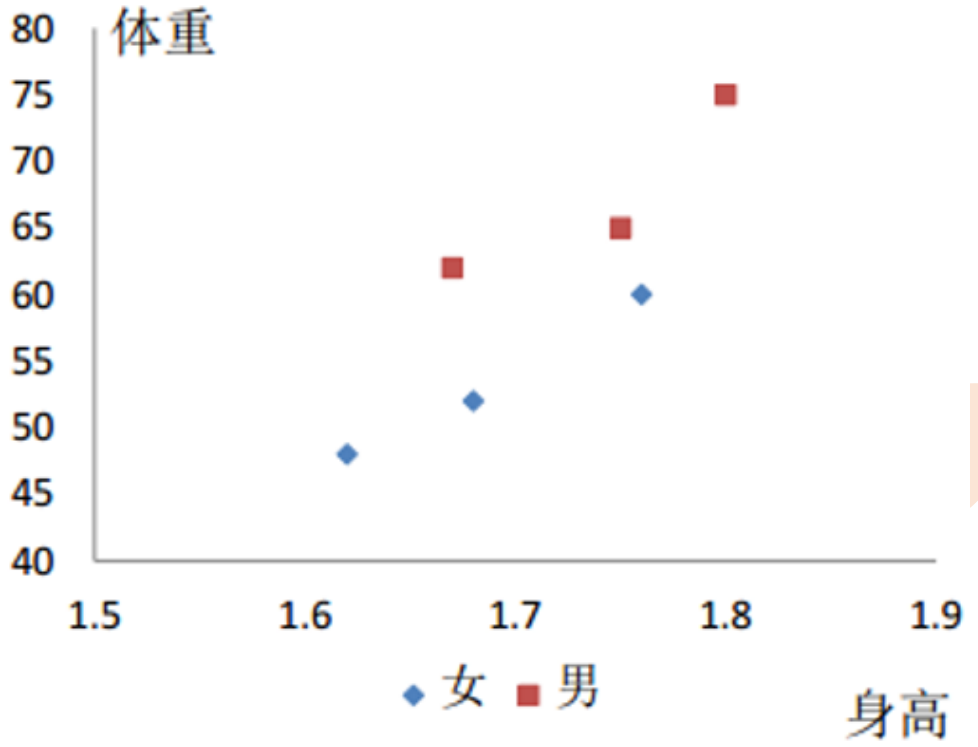


基 础 知 识

- 硬聚类 vs. 软聚类
  - 硬聚类：每个对象只属于一个类
    - A：class1
    - B：class2
    - C：class3
  - 软聚类：每个对象以某个概率属于每个类
    - A：class1:0.5, class2:0.2, class3:0.3
    - B：class1:0.3, class2:0.3, class3:0.5
    - C：class1:0.4, class2:0.5, class3:0.1

基础知识

- 向量表示
  - 每个对象用一个向量表示，可以视为高维空间的一个点
- 例如：根据身高体重来判断性别
  - ①　男(1.67, 62)
  - ②　男(1.75, 65)
  - ③　男(1.8, 75)
  - ④　女(1.62, 48)
  - ⑤　女(1.68, 52)
  - ⑥　女(1.76, 60)



基 础 知 识

- 向量表示
  - 每个对象用一个向量表示，可以视为高维空间的一个点
  - 所有对象形成数据空间
- 相似度计算：
  - Cosine、点积
  - 计算质心

$$\begin{pmatrix} & T_1 & T_2 & \cdots & T_t \\ D_1 & w_{11} & w_{21} & \cdots & w_{t1} \\ D_2 & w_{12} & w_{22} & \cdots & w_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ D_n & w_{1n} & w_{2n} & \cdots & w_{tn} \end{pmatrix}$$

- 距离矩阵、相似度矩阵
  - 给出聚类对象之间的距离（或相似性度量）

$$D_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} 0.0 & & & & \\ 2.0 & 0.0 & & & \\ 6.0 & 5.0 & 0.0 & & \\ 10.0 & 9.0 & 4.0 & 0.0 & \\ 9.0 & 8.0 & 5.0 & 3.0 & 0.0 \end{pmatrix} \end{array}$$

基 础 知 识

- 评价方法：
  - 内部评价法（Internal Evalution）：
    - 没有外部标准，非监督式
    - 同类是否相似，跨类是否相异
  - 外部评价法（External Evalution）：
    - 外部标准，监督式
    - 跟外部标准的一致性如何

基 础 知 识

- Davies-Bouldin index

$$DB = \frac{1}{n}\sum_{i=1}^{n}\max_{i\neq j}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right)$$

C_x – 表示类x的质心

σ_x – 表示类x内的所有对象到质心的平均距离

基 础 知 识

- 分类指标：
  - 准确度（accuracy）
  - 精度（Precision）
  - 召回（Recall）
  - F值（F-measure）

$$F_\beta = \frac{(\beta^2 + 1) \cdot P \cdot R}{\beta^2 \cdot P + R}$$

- 混淆矩阵
  - 准确度（accuracy）：(C11+C22) / (C11 + C12 + C21 + C22)
  - 精度（Precision）：C11 / (C11 + C21)
  - 召回（Recall）：C11 / (C11 + C12)

混淆表(confusion table)		分类器预测的类别	
		y1	y2
实际的类别	y1	C11	C12
	y2	C21	C22

基 础 知 识

- 混淆矩阵
  - 准确度（accuracy）：(50+35) / (35+5+10+50) = 85%
  - 精度（Precision）：50 / (50+5) = 90.9%
  - 召回（Recall）：50 / (50 + 10) = 83.3%

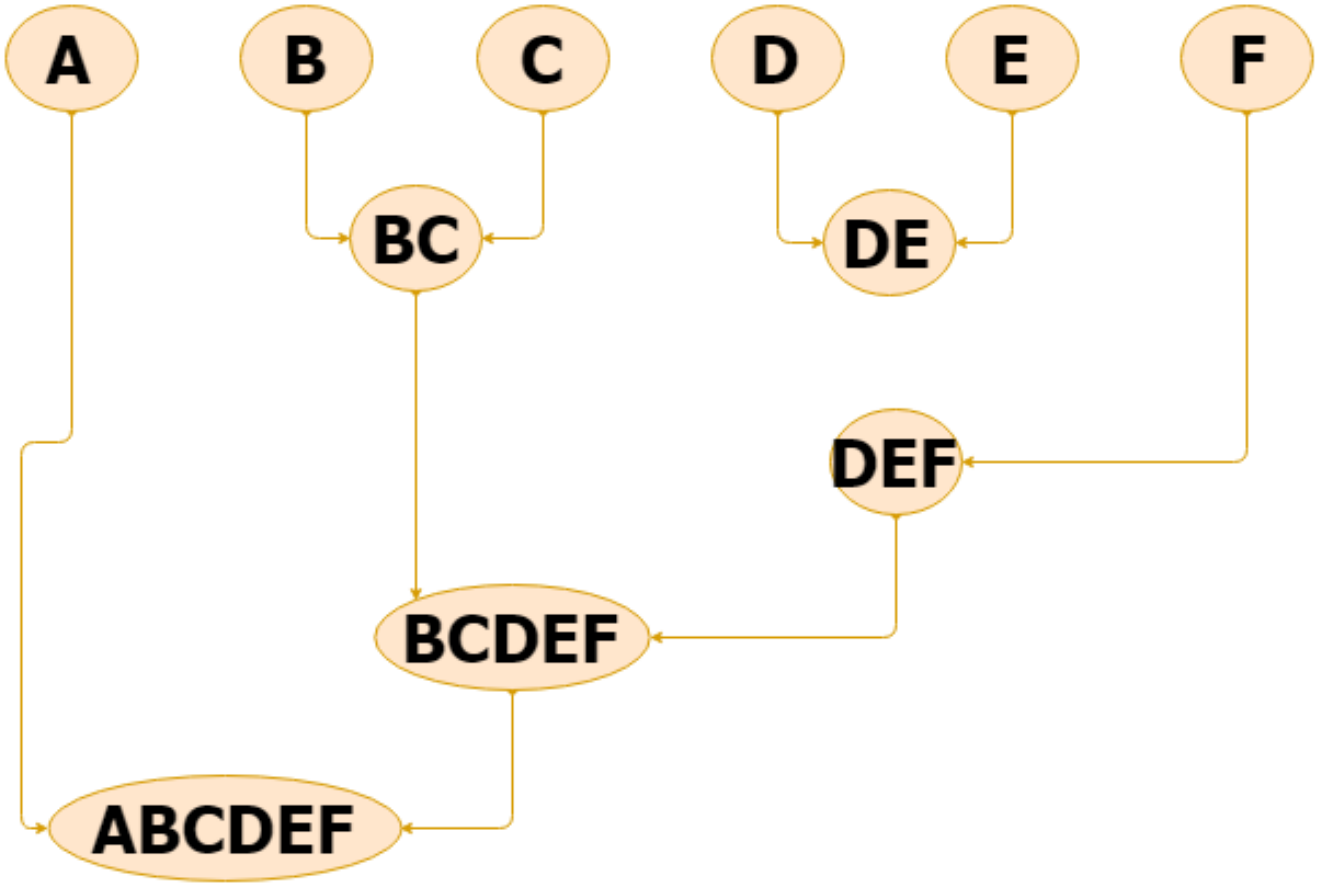
混淆表(confusion table)		分类器预测的类别	
		军事	科技
实际的类别	军事(60)	50	10
	科技(40)	5	35

OutLine

- 聚类基本知识
- 层次聚类法
- Kmeans聚类
- 【实践】基于MLlib的Kmeans聚类

层次聚类算法

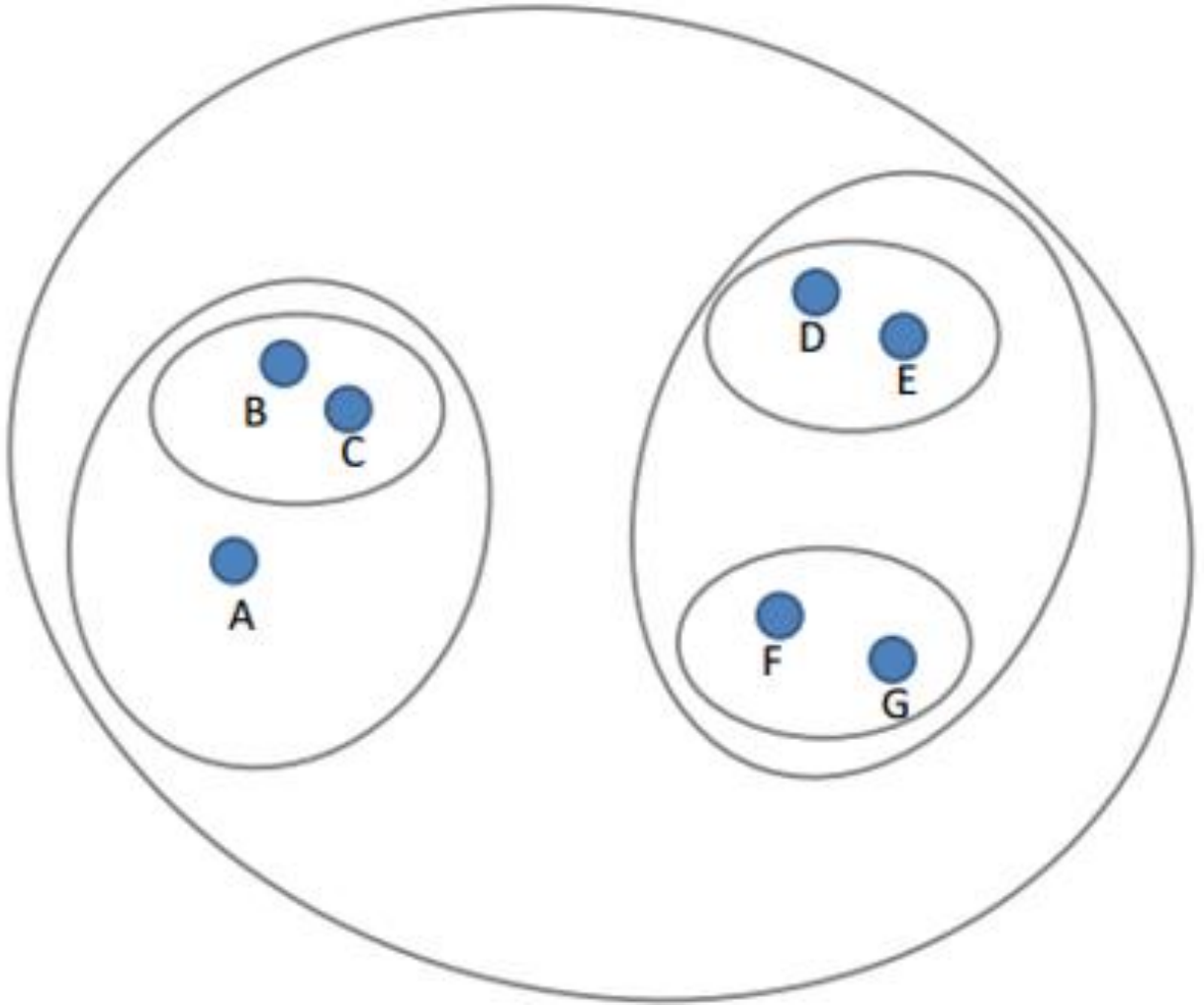
- 自底向上
  - 凝聚层次聚类
- 自顶向下
  - 分裂层次聚类



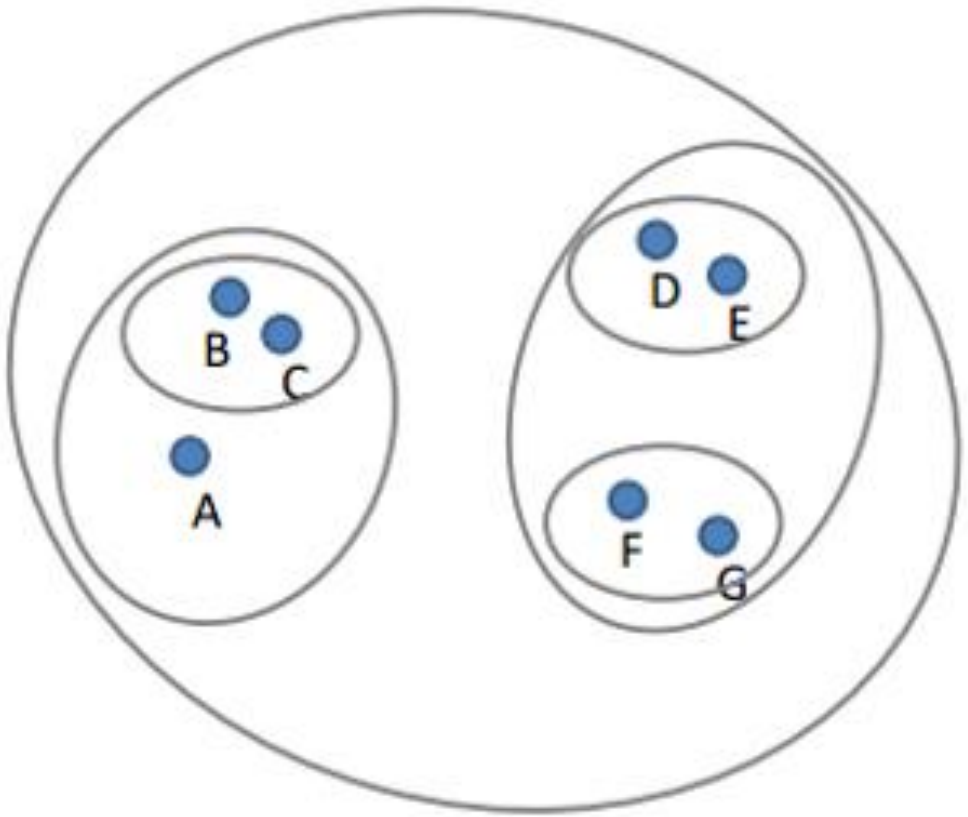
凝聚层次聚类算法

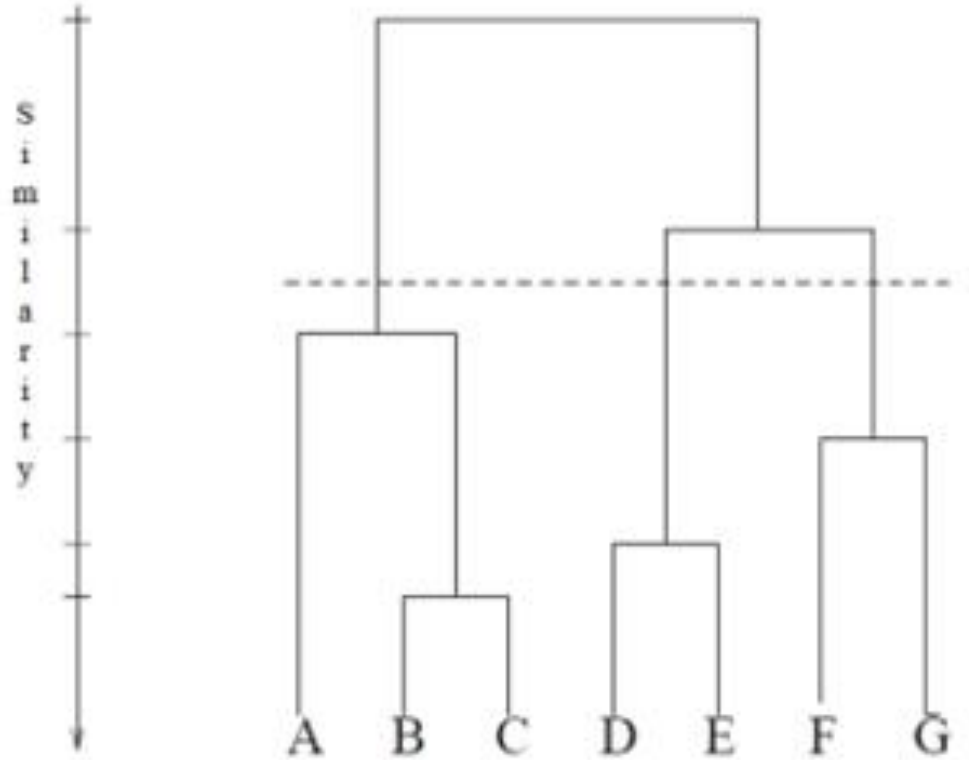
- 算法描述：
  - 1、将每一个对象归为一类，共得到N类，每类仅包含一个对象
  - 2、找到最接近的两个类合并成一个类
  - 3、重新计算新的类与所有旧类之间的距离
  - 4、重复2/3，直到最后合并为一个类为止（此类包含N个对象）

凝聚层次聚类过程



凝聚层次聚类树状图描述



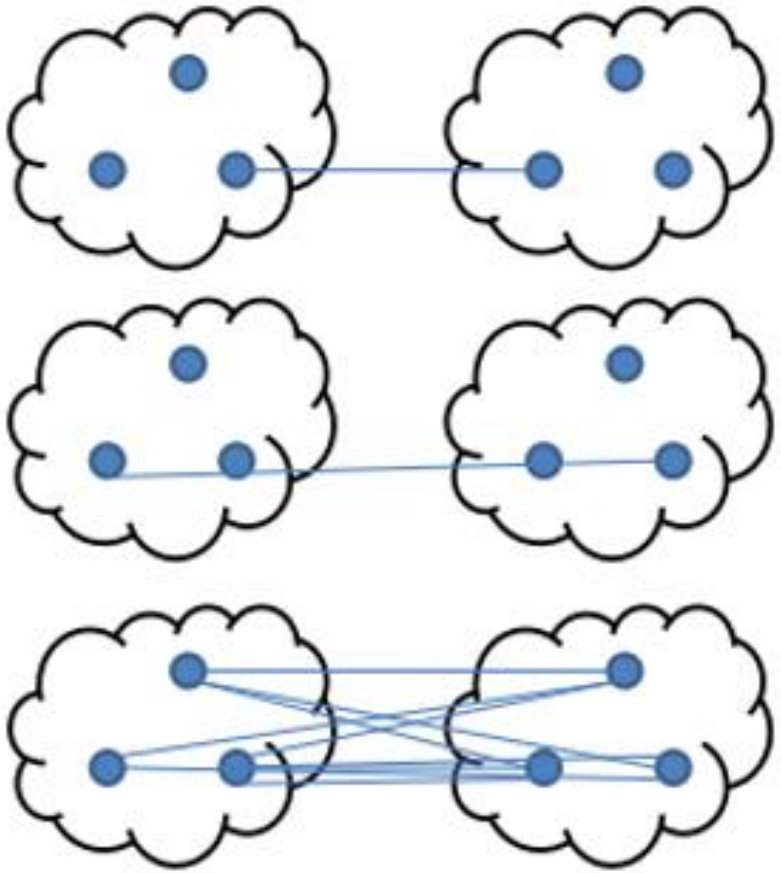


凝聚层次聚类算法

- 每次迭代需要重新计算新的类与旧类之间的距离
- 计算方式：
  - 单链（Single linkage）：最近距离
  - 全链（complete linkage）：最远距离
  - 组合链（average linkage）：平均距离

凝聚层次聚类算法

- 类间距离计算：



- **Min**
- **Max**
- **Group average**

层次聚类算法总结

- 算法简单
- 层次用于概念聚类（生成概念、文档层次树）
- 聚类对象的两种表示法都适用
- 处理大小不同的簇
- 簇选取步骤在树状图生成之后

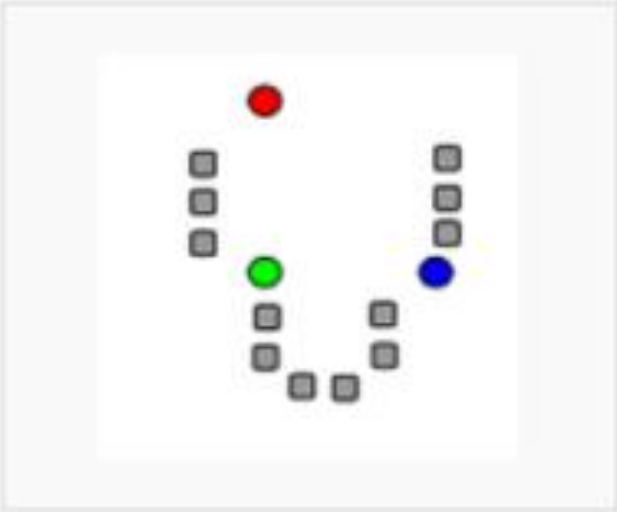
OutLine

- 聚类基本知识
- 层次聚类法
- Kmeans聚类
- 【实践】基于MLlib的Kmeans聚类

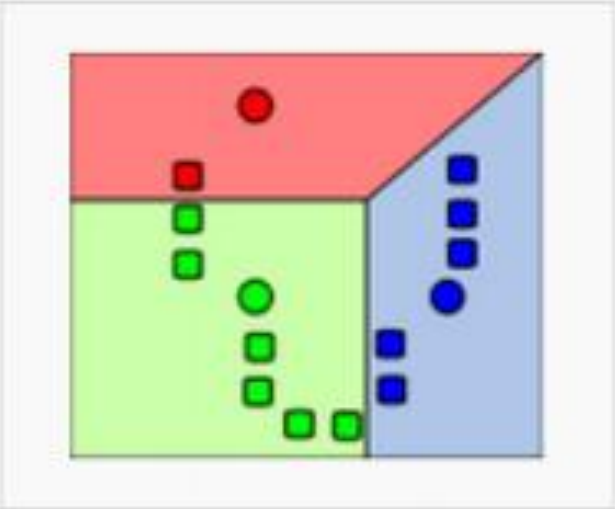
K平均（K-means）聚类

- 算法描述：
  - 任意选择K个点作为初始聚类中心
  - 根据每个聚类的中心，计算每个对象与这些中心的距离，并根据最小距离重新对相应对象进行划分
  - 重新计算每个聚类的中心
  - 当满足一定条件，如类别划分不再发生变化时，算法终止，否则继续步骤2和3

K平均（K-means）聚类

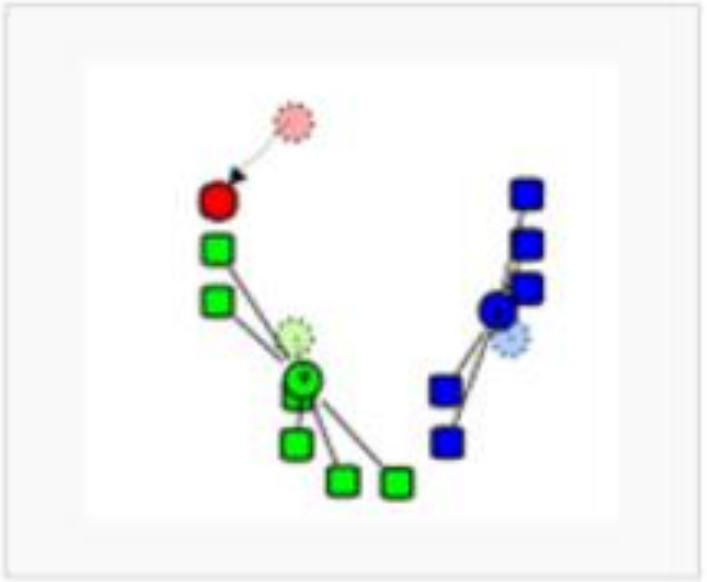


1) *k* initial "means" (in this case *k*=3) are randomly selected from the data set (shown in color).

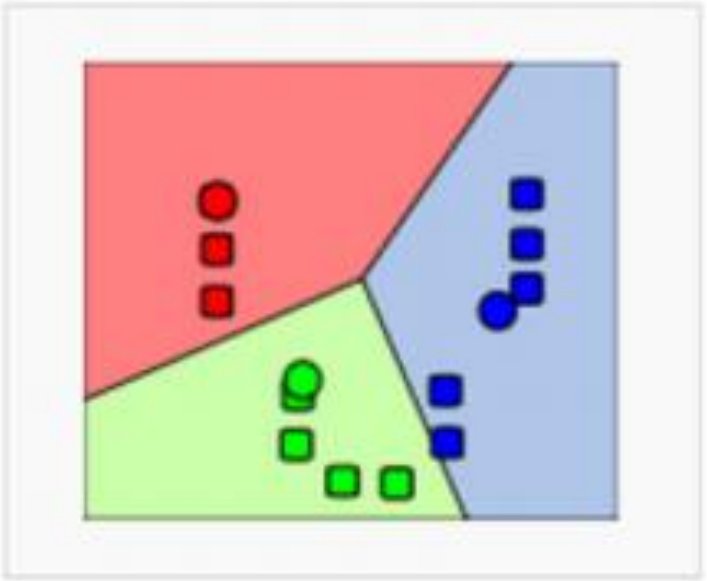


2) *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

K平均（K-means）聚类



3) The centroid of each of the *k* clusters becomes the new means.



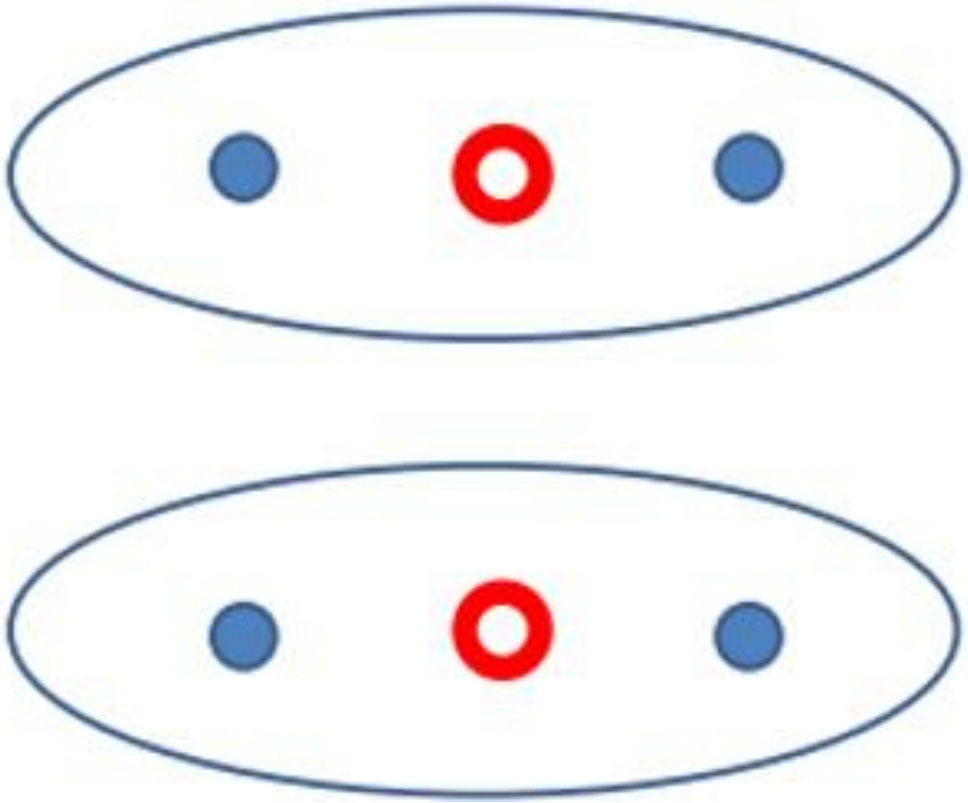
4) Steps 2 and 3 are repeated until convergence has been reached.

K平均（K-means）聚类

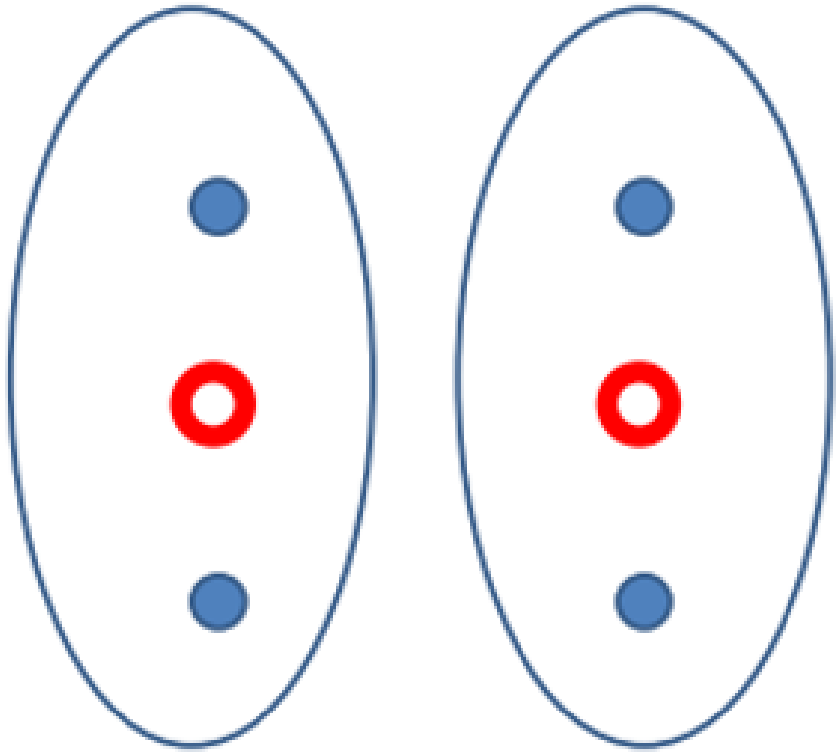
- 计算每个对象与这些中心的距离：
  - 欧式距离
- 重新计算每个聚类的中心对象：
  - 中心对象：均值
- 当满足一定条件，则算法终止：
  - 损失函数：WCSS
  - 步骤3：最小化簇内对象到质心的距离
  - 步骤4：重新计算质心，从而最小化WCSS

$$L(C)=\sum_{k\in K}\sum_{i\in k}\| x_i-c_k \|^2$$

K平均（K-means）聚类



K平均（K-means）聚类



非常依赖初始点选择！

K平均（K-means）聚类

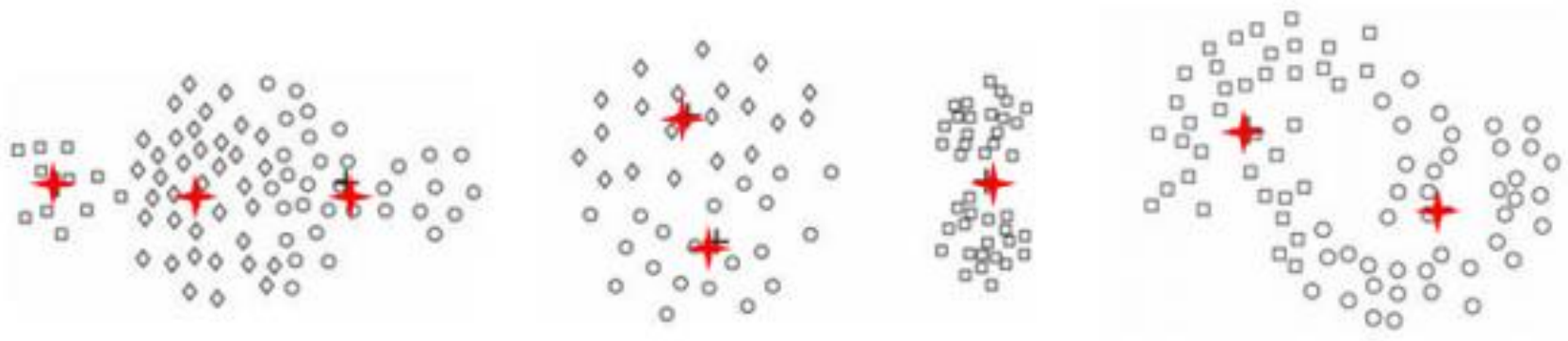
距离函数	中心	标准测度函数
L1	median	最小化对象到簇中心的L1距离和
L2	mean	最小化对象到簇中心的L2距离的平方之和
cosine	mean	最小化对象到簇中心的cosine距离和

K平均（K-means）聚类小结

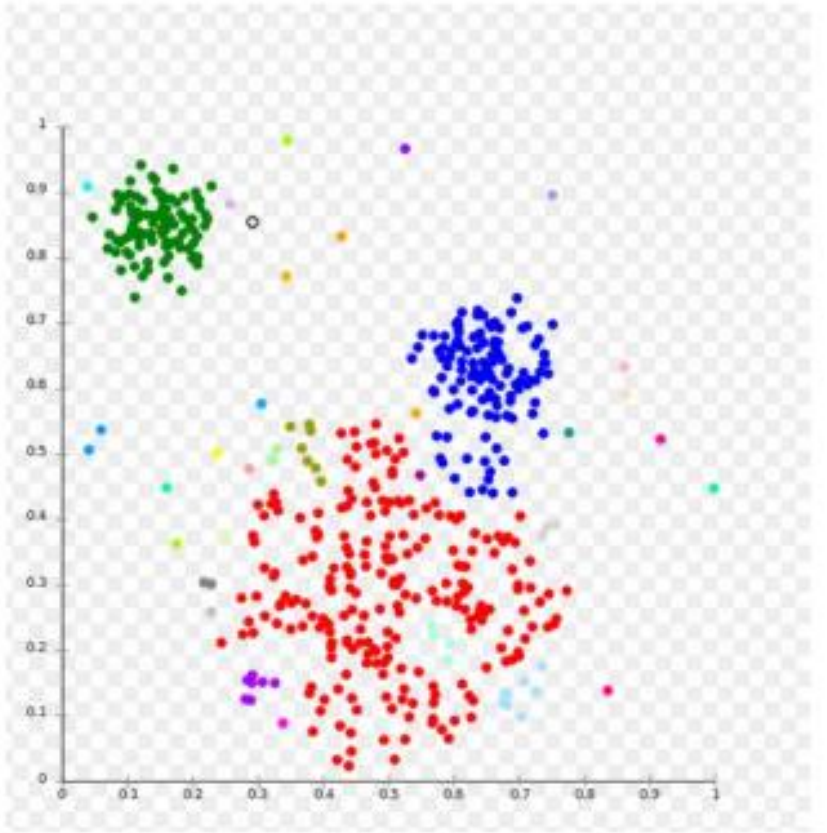
- K的选择
- 中心点的选择
  - 随机
  - 多轮随机：选择最小的WCSS
- 优点
  - 算法简单、有效
  - 时间复杂度：O(nkt)
    - N个聚类对象，K个类，T个迭代次数

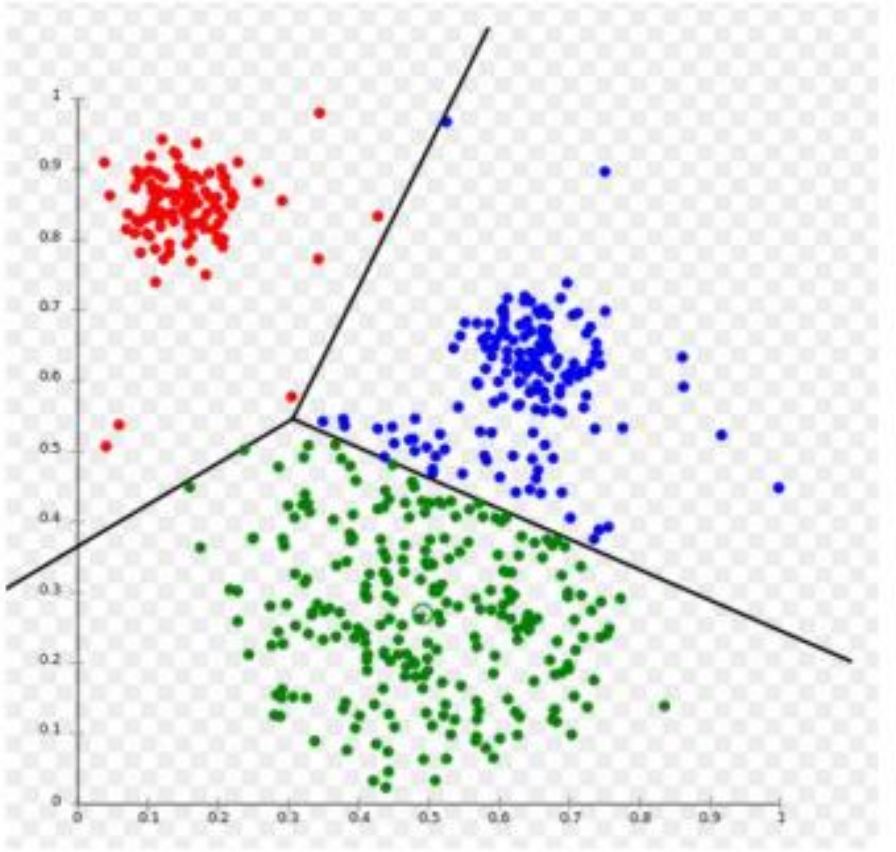
K平均（K-means）聚类小结

- 处理非球面（凸型）聚类
- 密度，大小不同的聚类（受K的限制，难于发现自然的聚类）
  - 部分解决方法：增加聚类个数

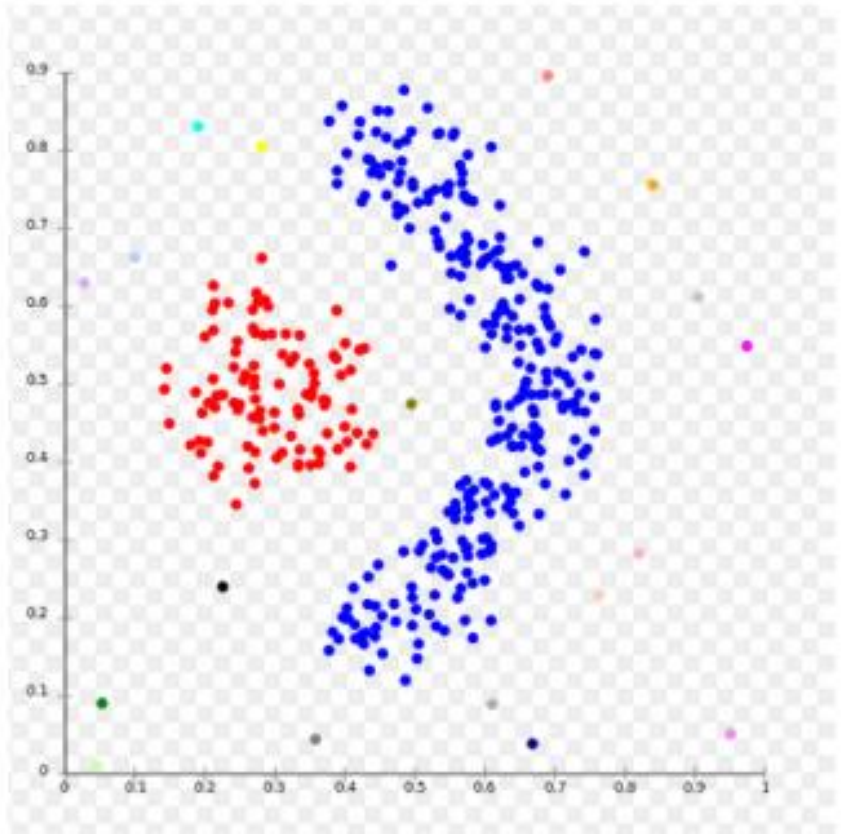


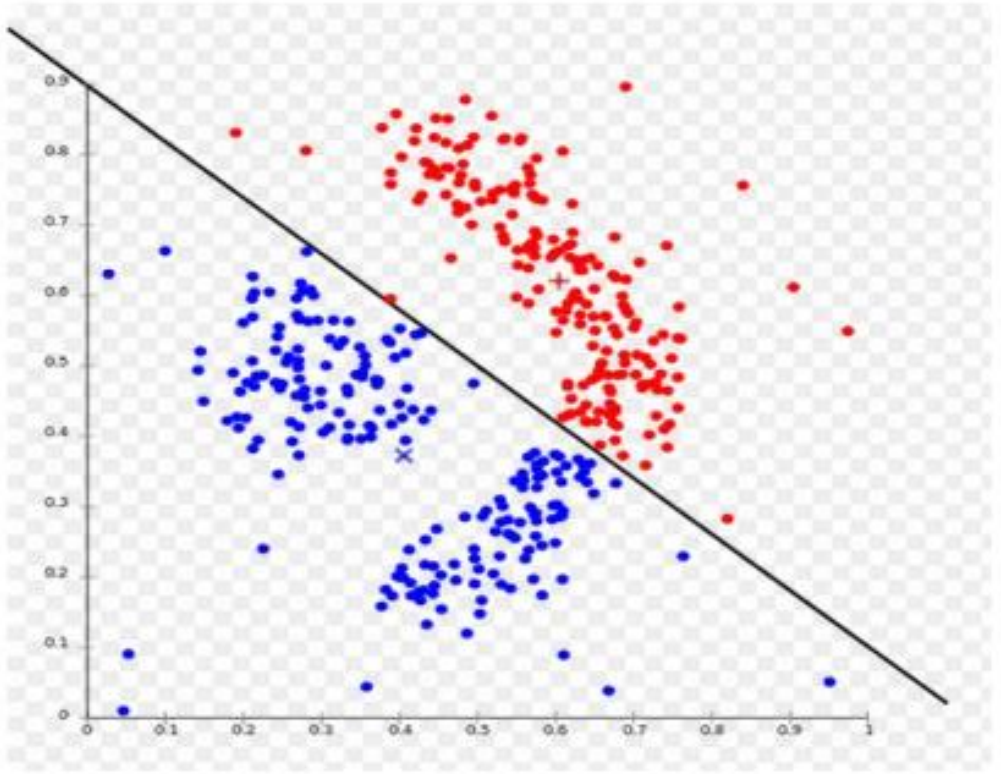
K-means VS. 层次聚类





K-means VS. 层次聚类





OutLine

- 聚类基本知识
- 层次聚类法
- Kmeans聚类
- 【实践】基于MLlib的Kmeans聚类

Q & A

@八斗学院