



大数据成就未来



聚类分析

张敏

17/11/29

目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



概述

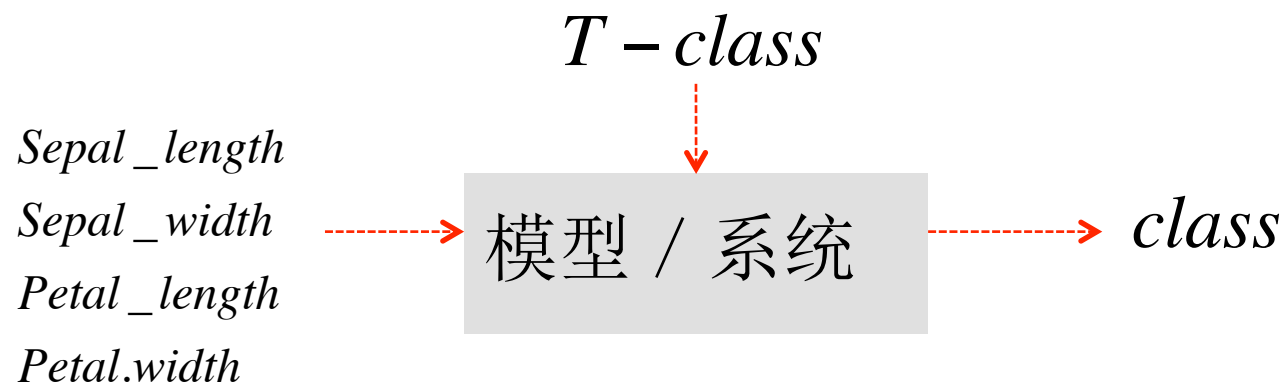
分类与聚类

分类：学习 / 训练过程

有监督，训练样本有明

确标签

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	T-class
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
7	3.2	4.7	1.4	versicolor
6.4	3.2	4.5	1.5	versicolor
6.3	3.3	6	2.5	virginica
5.8	2.7	5.1	1.9	virginica
6.5	3	5.8	2.2	?
6.2	2.9	4.3	1.3	?



概述

分类与聚类

聚类：学习 / 训练过程

无监督，样本无明确标签

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	
5.1	3.5	1.4	0.2	
4.9	3	1.4	0.2	
7	3.2	4.7	1.4	
6.4	3.2	4.5	1.5	
6.3	3.3	6	2.5	
5.8	2.7	5.1	1.9	
6.5	3	5.8	2.2	
6.2	2.9	4.3	1.3	

Sepal_length

Sepal_width

Petal_length

Petal.width



模型 / 系统



class



概述

聚类的概念

- 聚类是把各不相同的个体分割为有更多相似性的子集合的工作。
- 聚类生成的子集合称为簇

聚类的要求

- 生成的簇内部的任意两个对象之间具有较高的相似度
- 属于不同簇的两个对象间具有较高的相异度

聚类与分类的区别在于聚类不依赖于预先定义的类，没有预定义的类和样本——聚类是一种无监督的数据挖掘任务



概述

聚类的概念

聚类通常作为其他数据挖掘或建模的前奏。

例如，聚类可以作为市场划分研究的第一步：

- 不是对“客户对哪些促销反应最好”提出一个统一的适合所有人的标准
- 而是首先将客户划分为有相似购物习惯的人群，然后研究对每个人群用哪种促销最好。

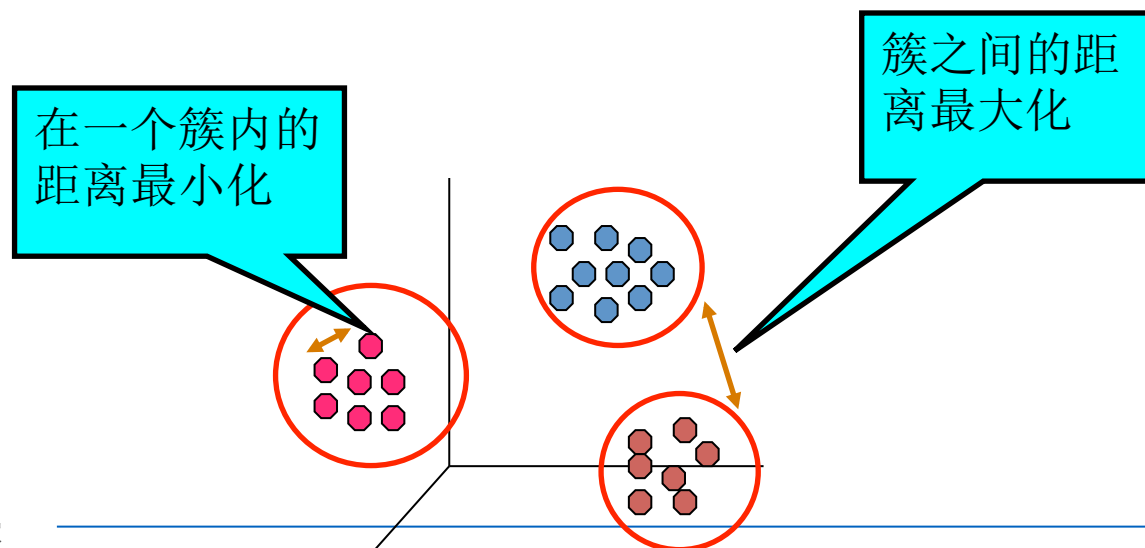
聚类能够促进我们对数据的理解，刻画部分用户的特征



概述





聚类的概念

- 仅根据在数据中发现的描述对象及其关系的信息，将数据对象分组。
- 与分类模型需要使用有类标记样本构成的训练数据不同，聚类模型可以建立在无类标记的数据上，是一种非监督的学习算法。
- 聚类的输入是一组未被标记的样本，聚类根据数据自身的距离或相似度将他们划分为若干组，划分的原则是组内样本最小化而组间（外部）距离最大化：



概述

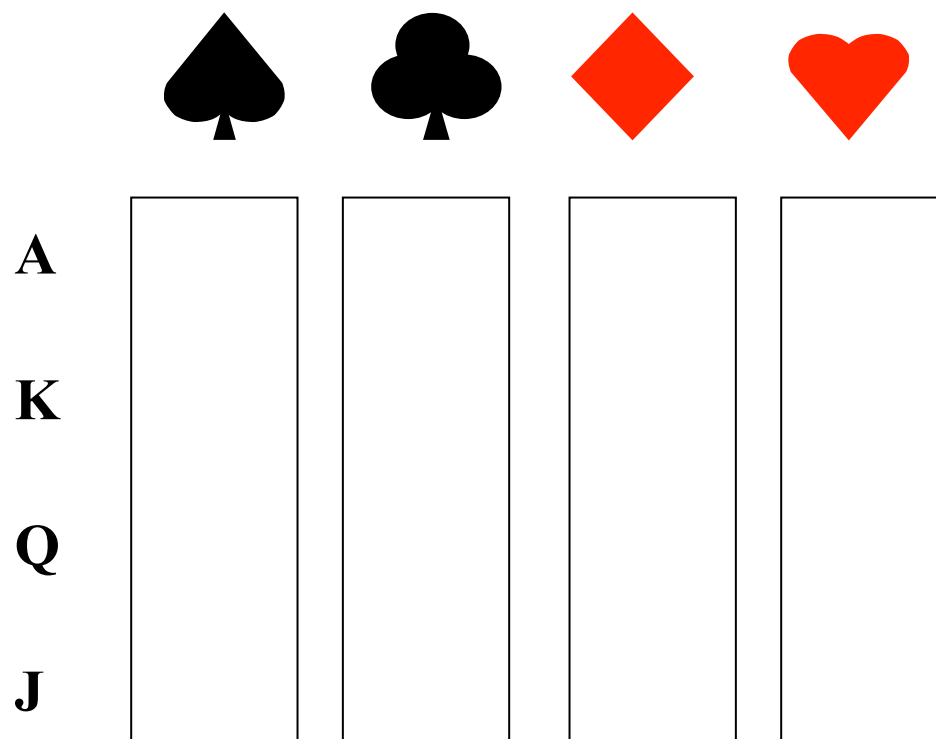
有16张牌如何将他们分组

				
A	<div></div>	<div></div>	<div></div>	<div></div>
K	<div></div>	<div></div>	<div></div>	<div></div>
Q	<div></div>	<div></div>	<div></div>	<div></div>
J	<div></div>	<div></div>	<div></div>	<div></div>

概述

分成四组





- 每组里花色相同
- 组与组之间花色相异



概述

分成两组

- 每组里符号相同
- 组与组之间符号相异

				
A	<input type="text"/>			
K	<input type="text"/>			
Q	<input type="text"/>			
J	<input type="text"/>			



概述

聚类分析：物以类聚、人以群分

应用领域：

- 客户价值分析
- 文本分类
- 基因识别
- 空间数据处理
- 卫星图片分析

数据分析、统计学、机器学习、空间数据库技术、生物学和市场学也推动了聚类分析研究的进展



概述

常用聚类算法

聚类算法种类繁多，且其中绝大多数可以用R实现。下面将选取普及性最广、最实用、最具有代表性的5中聚类算法进行介绍，其中包括：

- K-均值聚类(K-Means)
- K-中心点聚类(K-Medoids)
- 密度聚类(Densit-based Spatial Clustering of Application with Noise, DBSCAN)
- 层次聚类(系谱聚类 Hierarchical Clustering, HC)
- 期望最大化聚类(Expectation Maximization, EM)

需要说明的是，这些算法本身无所谓优劣，而最终运用于数据的效果却存在好坏差异，这在很大程度上取决于数据使用者对于算法的选择是否得当。



目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



相似性度量

距离

聚类分析是研究对样本或变量的聚类，在进行聚类时，可使用的方法有很多，而这些方法的选择往往与变量的类型是有关系的，由于数据的来源及测量方法的不同，变量大致可以分为两类：

1. 定量变量，也就是通常所说的连续变量。
2. 定性变量，这些量并非真有数量上的变化，而只有性质上的差异。这些量可以分为两种，一种是有序变量，另一种是名义变量。



相似性度量

连续型变量距离

典型的距离定义

距离	定义式	说明
绝对值距离	$d_{ij}(1)=\sum_{k=1}^p x_{ik}-x_{jk} $	绝对值距离是在一维空间下进行的距离计算
欧式距离	$d_{ij}(2)=\sqrt{\sum_{k=1}^p(x_{ik}-x_{jk})^2}$	欧式距离是在二维空间下进行的距离计算
闵可夫斯基距离	$d_{ij}(q)=\left[\sum_{k=1}^p(x_{ik}-x_{jk})^q\right]^{1/q},\quad q>0.$	闵可夫斯基距离是在 q 维空间下进行的距离计算
切比雪夫距离	$d_{ij}(\infty)=\max_{1\leq k\leq p} x_{ik}-x_{jk} .$	切比雪夫距离是 q 取正无穷大时的闵可夫斯基距离，即切比雪夫距离是在 $+\infty$ 维空间下进行的距离计算
Lance 距离	$d_{ij}(L)=\sum_{k=1}^p\frac{ x_{ik}-x_{jk} }{x_{ik}+x_{jk}}$	减弱极端值的影响能力
归一化距离	$d_{ij}=\sum_{k=1}^p\frac{ x_{ik}-x_{jk} }{\max(x_k)-\min(x_k)}$	自动消除不同变量间的纲量影响，其中每个变量 k 的距离取值均是 $[0,1]$

相似性度量

相似系数

两个仅包含二元属性的对象之间的相似性度量也称相似系数

两个对象的比较导致四个：f00 = x取0并且y取0的属性个数；f01 = x取0并且y取1的属性个数；f10 = x取1并且y取0的属性个数；f11 = x取1并且y取1的属性个数

简单匹配系数：SMC = 值匹配的属性个数 / 属性个数
$$= (f11 + f00) / (f01 + f10 + f11 + f00)$$

Jaccard(雅卡尔)系数：J = 匹配的个数 / 不涉及0-0匹配的属性个数
$$= (f11) / (f01 + f10 + f11)$$



相似性度量

相似系数

两个二元向量： $x = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0)$

$y = (0, 0, 0, 0, 0, 0, 1, 0, 0, 1)$

$f_{00} = 7$ (x 取0并且 y 取0的属性个数)

$f_{01} = 2$ (x 取0并且 y 取1的属性个数)

$f_{10} = 1$ (x 取1并且 y 取0的属性个数)

$f_{11} = 0$ (x 取1并且 y 取1的属性个数)

简单匹配系数： $SMC = (f_{11} + f_{00}) / (f_{01} + f_{10} + f_{11} + f_{00})$

$$= (0 + 7) / (2 + 1 + 0 + 7) = 0.7$$

Jaccard系数： $J = (f_{11}) / (f_{01} + f_{10} + f_{11})$

$$= 0 / (2 + 1 + 0) = 0$$



相似性度量

相似系数

余弦相似系数（如计算两文档间相似系数）：

$$\cos(x_1, x_2) = (x_1 \cdot x_2) / \|x_1\| \|x_2\|,$$

其中 \cdot 表示向量的点积(内积)， $\|x\|$ 表示向量的范数。

例向量： $x_1 = (3, 2, 0, 5, 0, 0, 0, 2, 0, 0)$

$x_2 = (1, 0, 0, 0, 0, 0, 0, 1, 0, 2)$

则余弦相似系数为： $\cos(x_1, x_2) = 5 / (6.481 * 2.245) = 0.3436$



目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



K-Means

例：某餐饮公司欲通过客户消费记录寻找VIP客户，进行精准营销。

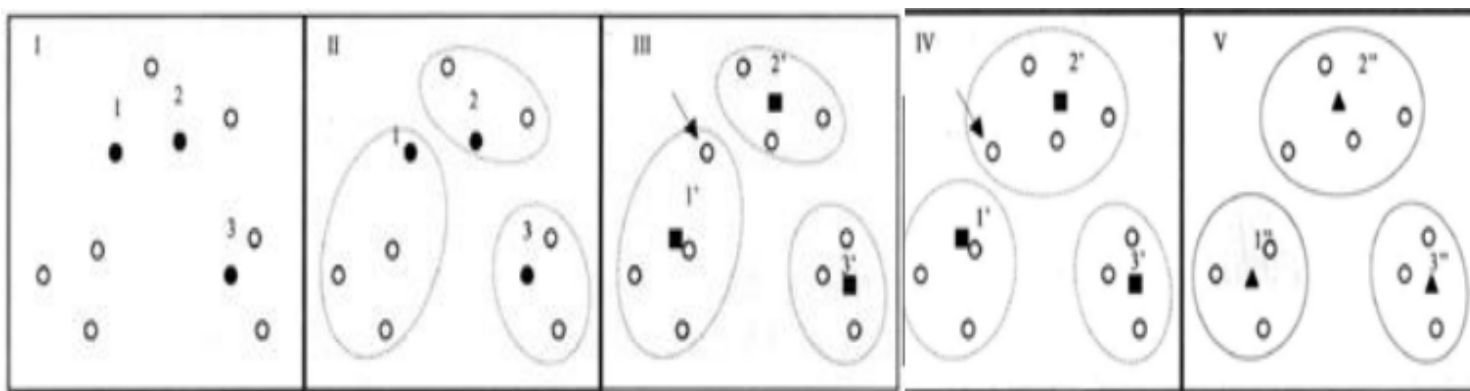
客户id	客单价
a	1
b	2
c	4
d	5



K-Means

算法步骤

1. 随机选取K个样本作为类中心；
2. 计算各样本与各类中心的距离；
3. 将各样本归于最近的类中心点；
4. 求各类的样本的均值，作为新的类中心；
5. 判定：若类中心不再发生变动或达到迭代次数，算法结束，否则回到第2步。



K-Means

选定样本a和b为初始类中心，中心值分别为1、2

a	1
b	2
c	4
d	5

	1	2	class
a	0	1	1
b	1	0	2
c	3	2	2
d	4	3	2

center1=1
center2=11/3

	1	11/3	class
a	0	8/3	1
b	1	5/3	1
c	3	1/3	2
d	4	4/3	2

center1=3/2
center2=9/2

	3/2	9/2	class
a	1/2	7/2	1
b	1/2	5/2	1
c	5/2	1/2	2
d	7/2	1/2	2

center1=3/2
center2=9/2

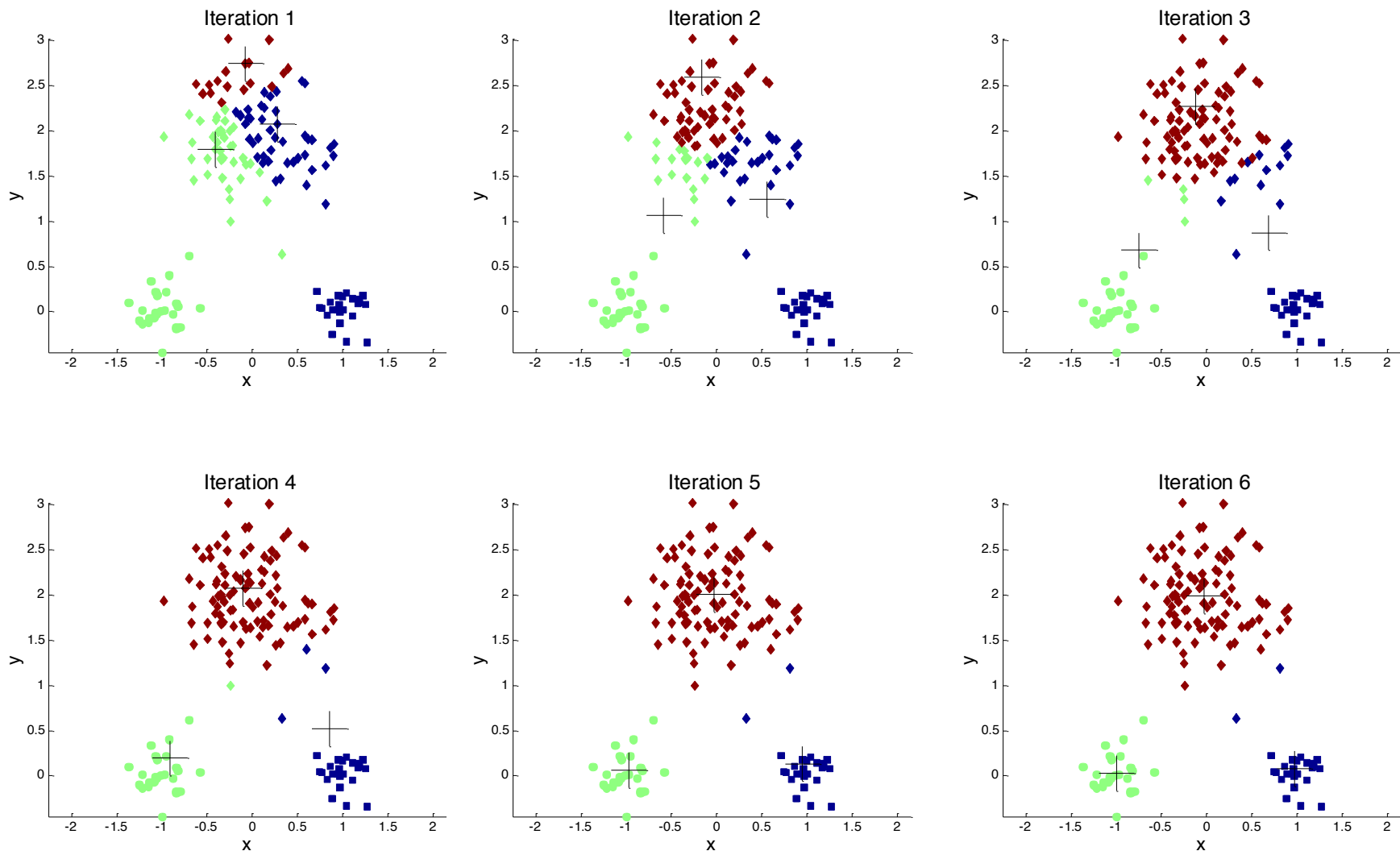
结束

1. 选中心
2. 求距离
3. 归类
4. 求新类中心
5. 判定结束



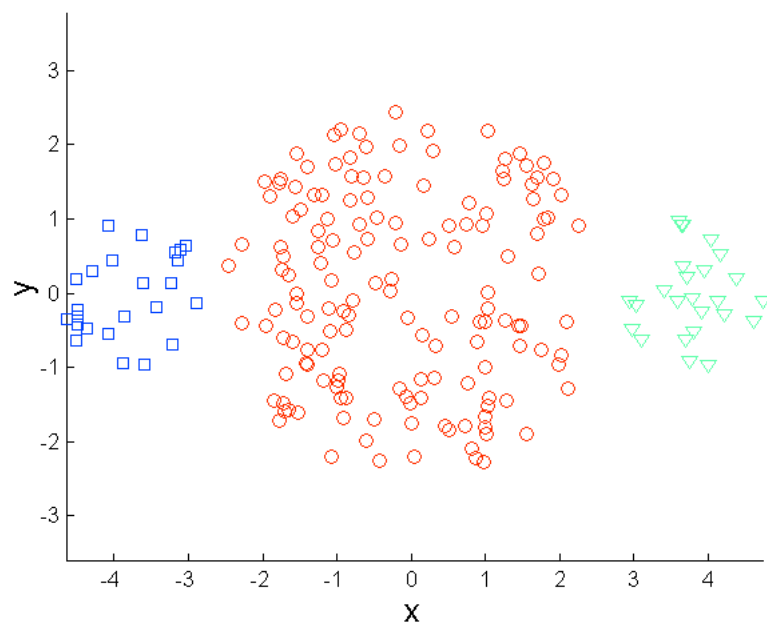
K-Means

示例

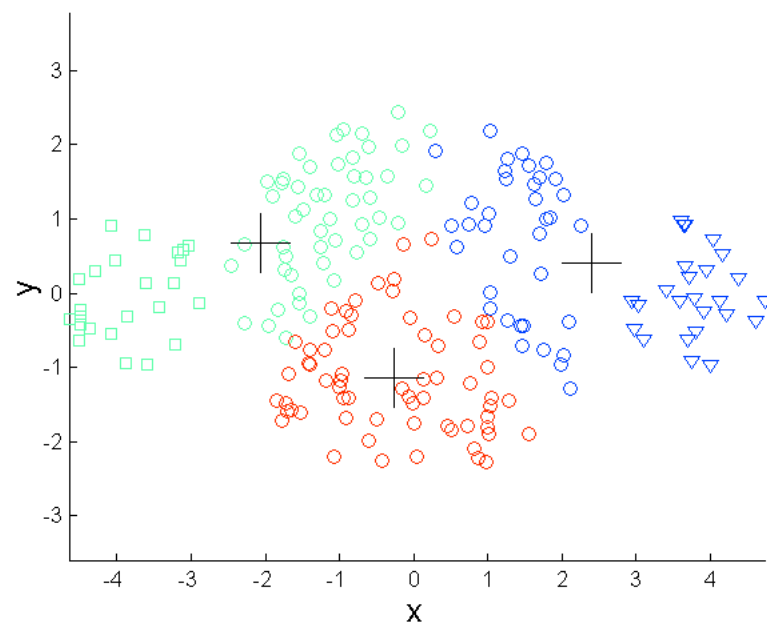


K-Means

思考：K-means聚类的特点是什么？



Original Points

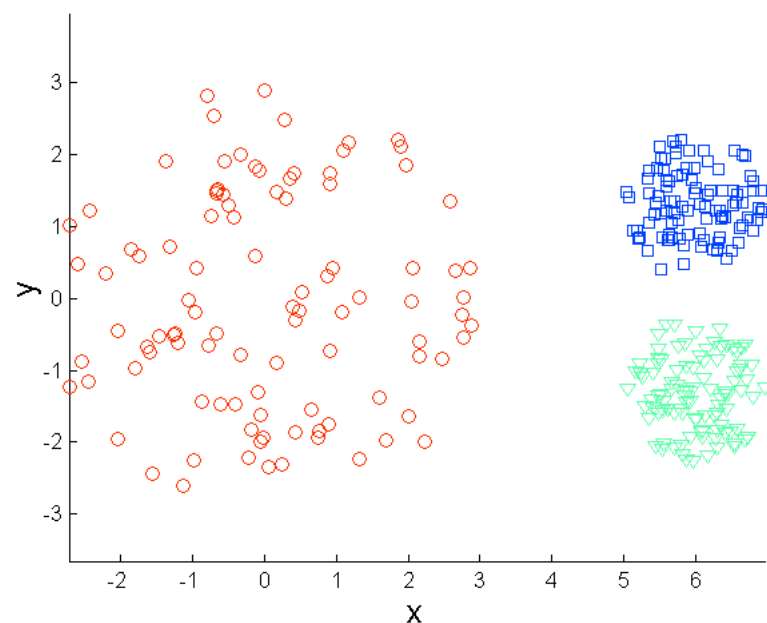


K-means (3 Clusters)

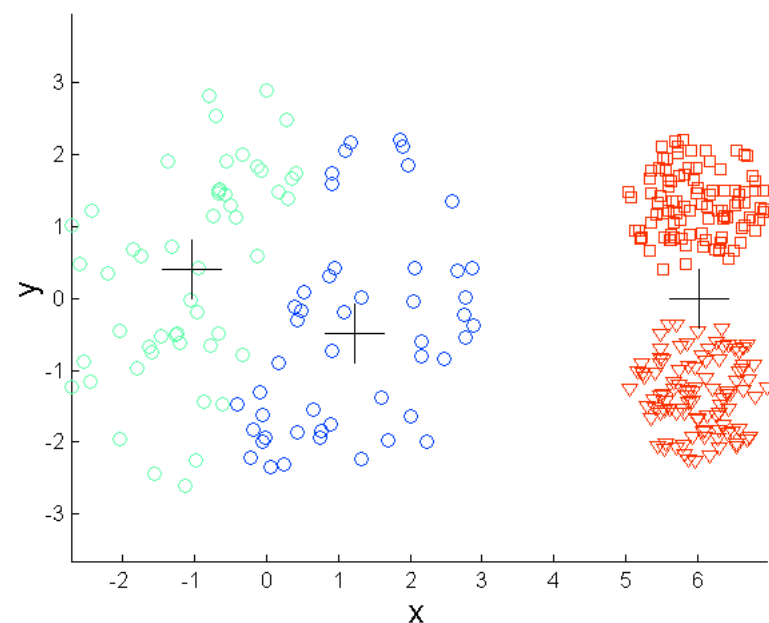


K-Means

思考：K-means聚类的特点是什么？



Original Points



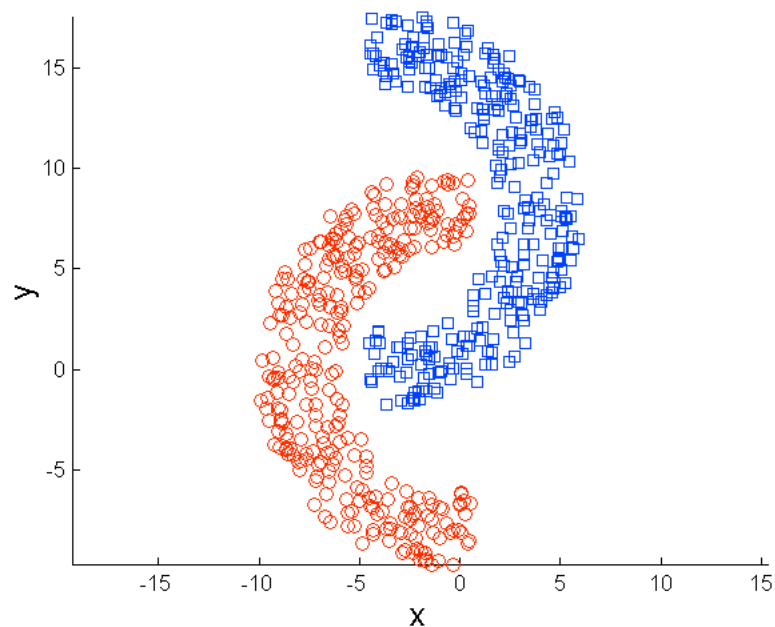
K-means (3 Clusters)



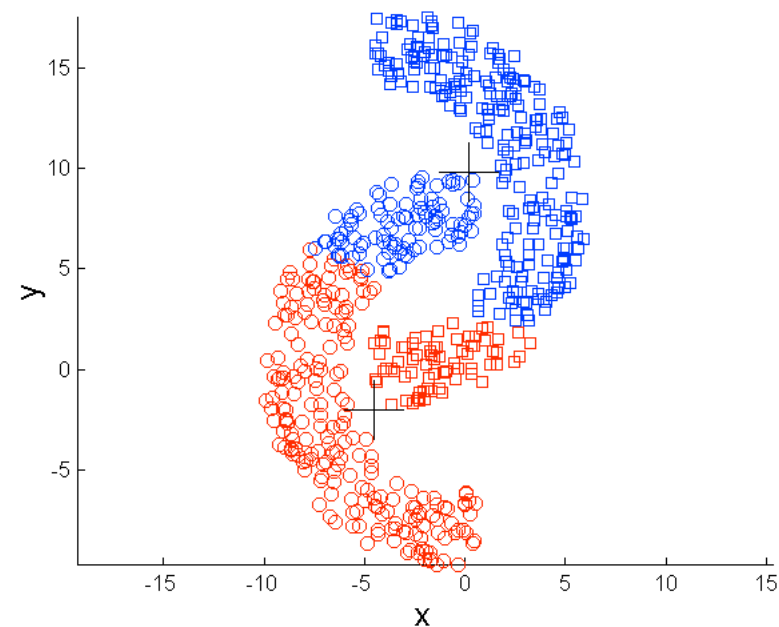
K-Means

思考：K-means聚类的特点是什么？

适用于球状簇



Original Points



K-means (2 Clusters)



K-Means

优缺点

优点：

- 算法简单
- 适用于球形簇
- 二分k均值等变种算法运行良好，不受初始化问题的影响。

缺点：

- 不能处理非球形簇、不同尺寸和不同密度的簇
- 对离群点、噪声敏感



K-Means

练习：将平面上100个点聚为2类

- x坐标为1到100
- y坐标为101到200



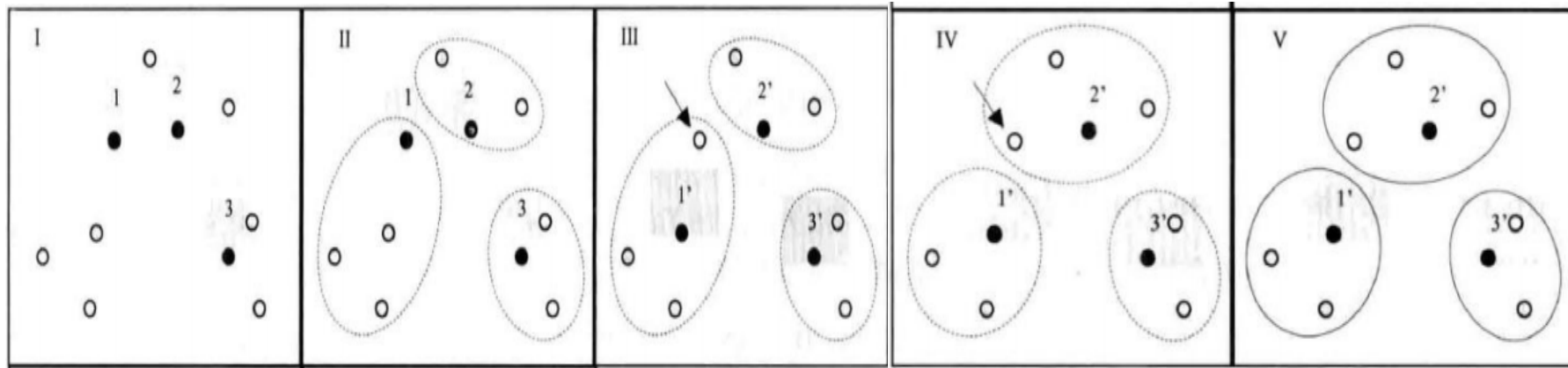
目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



K-Medoids

- 1、随机选取K个样本作为类中心；
- 2、计算各样本与各类中心的距离；
- 3、将各样本归于最近的类中心点；
- 4、在各类别内选取到其余样本距离之和最小的样本作为新的类中心；
- 5、判定：若类中心不再发生变动或达到迭代次数，算法结束，否则回到第2步。



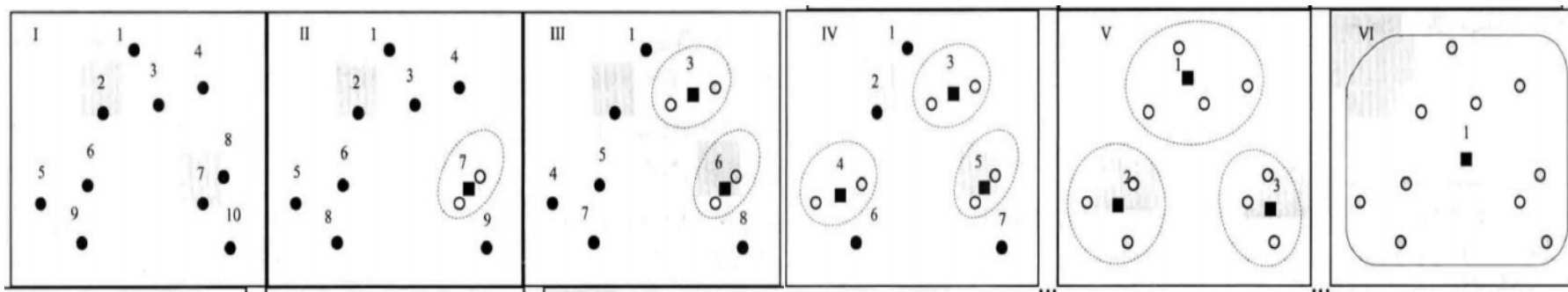
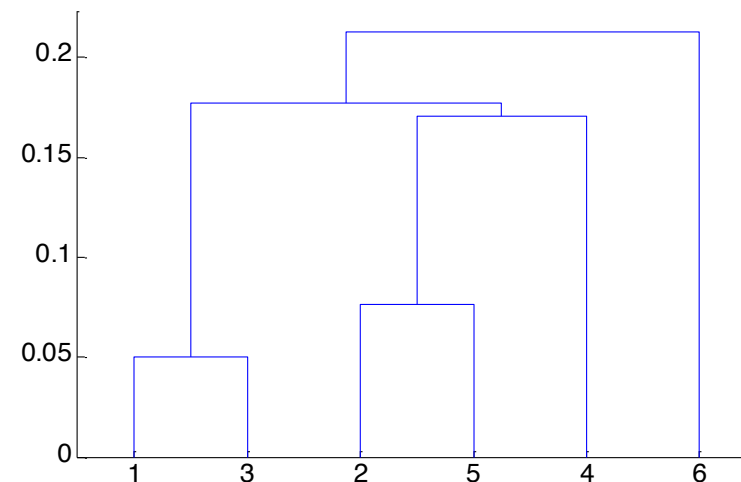
目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



层次聚类(系谱聚类 Hierarchical Clustering, HC)

- 不需事先设定类别数 k
- 每次迭代过程仅将距离最近的两个样本/簇聚为一类
- 得到 $k=n$ 至 $k=1$ (n 为待分类样本总数)个类别的聚类结果



层次聚类(系谱聚类 Hierarchical Clustering, HC)

优缺点

优点：

- 某些应用领域需要层次结构。如：系统发生树，基因芯片
- 有些研究表明，这种算法能够产生较高质量的聚类

缺点：

- 计算量、存储量大
- 对噪声、高维数据敏感



目录

1	概述
2	相似性度量
3	K-Means
4	K-Medoids
5	层次聚类
6	密度聚类



密度聚类(DBSCAN)

算法简介

- 主要是依赖两个主要的参数来进行聚类的，即对象点的区域半径Eps和区域内点的个数的阈值MinPts
- DBSCAN算法通过查找数据聚类的，即对象点的区域半径Eps和区域内点的个数的阈值M集中任意一个点的距离在Eps区域来进行聚类，如果这个区域内的点数大于MinPts，则将这些点放在同一个簇中，形成新的一类。



密度聚类(DBSCAN)

基础概念

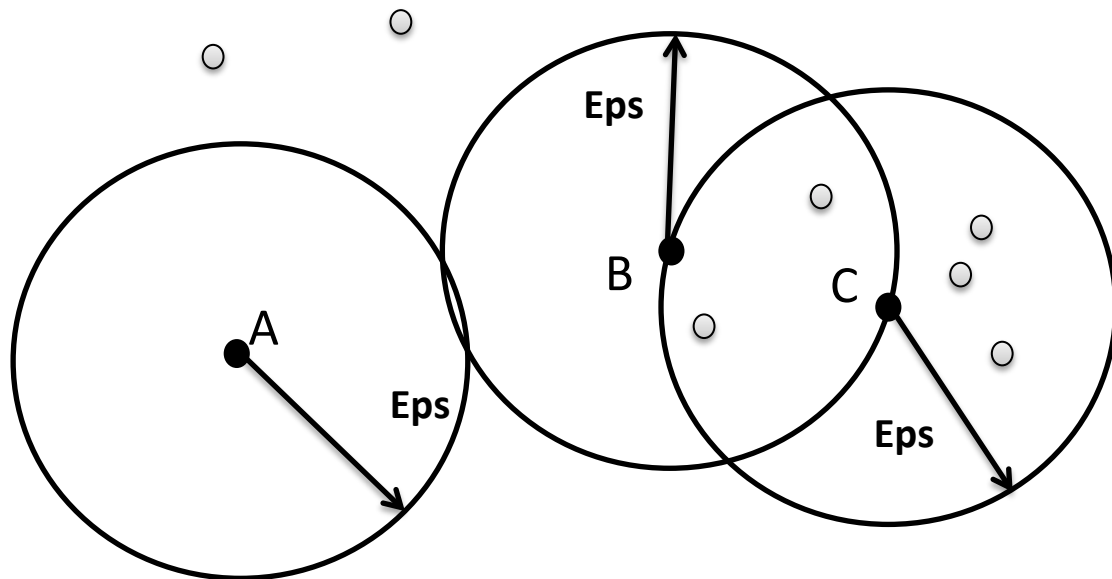
1. 点的Eps区域：以空间中任意一点 p 为圆心， Eps 为半径的区域中的点的集合 D 。
2. 点的密度：集合 D 中点的个数即为点 P 的密度。
3. 阈值 $MinPts$ ：在集合 D 中使点 p 成为核心点的限定值。
4. 核心点：如果点 p 的密度等于或者大于阈值 $MinPts$ ，则 P 为核心点。
5. 边界点：如果点 p 不是核心点，但落在其他核心点的区域内，那么 p 点为边界点。
6. 噪声点：如果点 p 既不是核心点，也不是边界点，则 p 点为噪声点。
7. 密度直达：存在空间任意一点 q 在集合 D 中，且 p 是核心点，则称点 q 从点 p 密度直达。
8. 密度可达：空间存在点 m 在集合 D 中，如果 m 到 p 密度直达，且 m 到 q 也是密度直达，那么点 p 从点 q 密度可达。



密度聚类(DBSCAN)

样本（对象点）的区域半径Eps和区域内点的个数的阈值MinPts

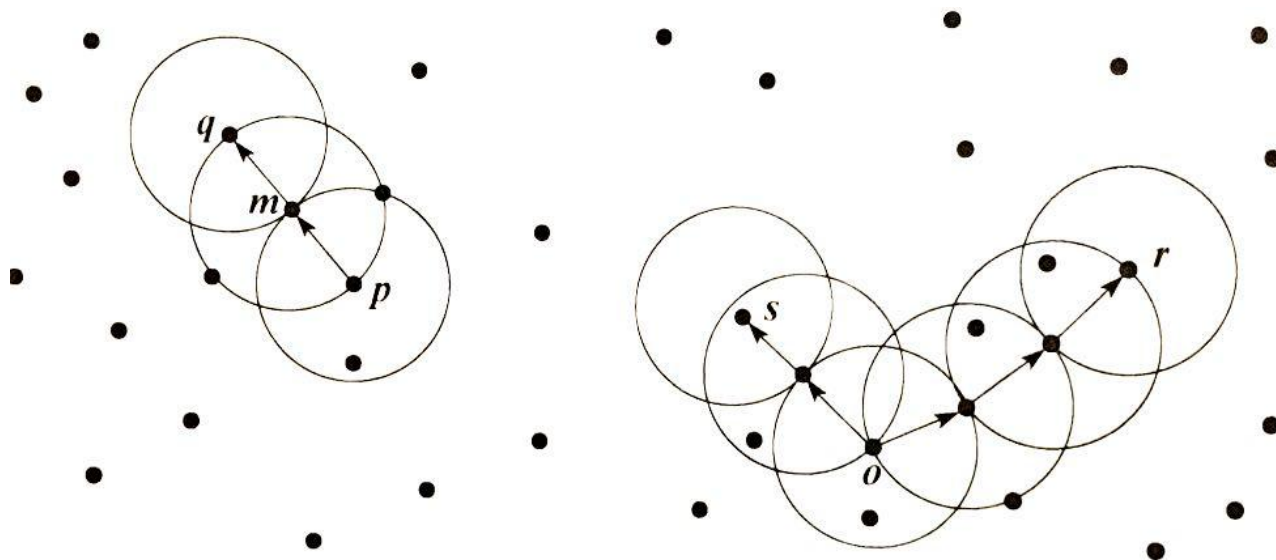
- 核心点：如果点p的密度等于或者大于阈值MinPts，则P为核心点。
- 边界点：如果点p不是核心点，但落在其他核心点的区域内，那么p点为边界点。
- 噪声点：如果点p既不是核心点，也不是边界点，则p点为噪声点。



A为噪声点
B为边界点
C为核心点

密度聚类(DBSCAN)

DBSCAN算法原理直观图



- 1.密度直达：点 q 距核心点 m 距离小于等于 ϵ ，从 m 到 q 密度直达。不对称
- 2.密度可达：若从 p 到 m 密度直达，从 m 到 q 密度直达，则从 p 到 q 密度可达。不对称
- 3.密度相连：若从 o 到 s 密度可达，且从 o 到 r 密度可达的，所有 o, r 和 s 都是密度相连的。对称

密度聚类(DBSCAN)

算法步骤

1. 定义半径和MinPts
2. 从对象集合D中抽取未被访问过的样本点q
3. 检验该样本点是否为核心对象，如果是则进入下一步，否则返回上一步
4. 找出该样本点所有从该点密度可达的对象，构成聚类
5. 如果全部样本点都已被访问，则结束算法，否则返回第2步骤



密度聚类(DBSCAN)

优缺点

优点：

- 因为DBSCAN使用簇的基于密度的定义，因此它是相对抗噪声的，并且能够处理任意形状和大小的簇。

缺点:

- 当簇的密度变化太大时，DBSCAN就会有麻烦。
- 当近邻计算需要计算所有的点对近邻度时，DBSCAN可能是开销很大的。





大数据成就未来



Thank you!

泰迪科技 : www.tipdm.com
热线电话 : 40068-40020

