金融科技学

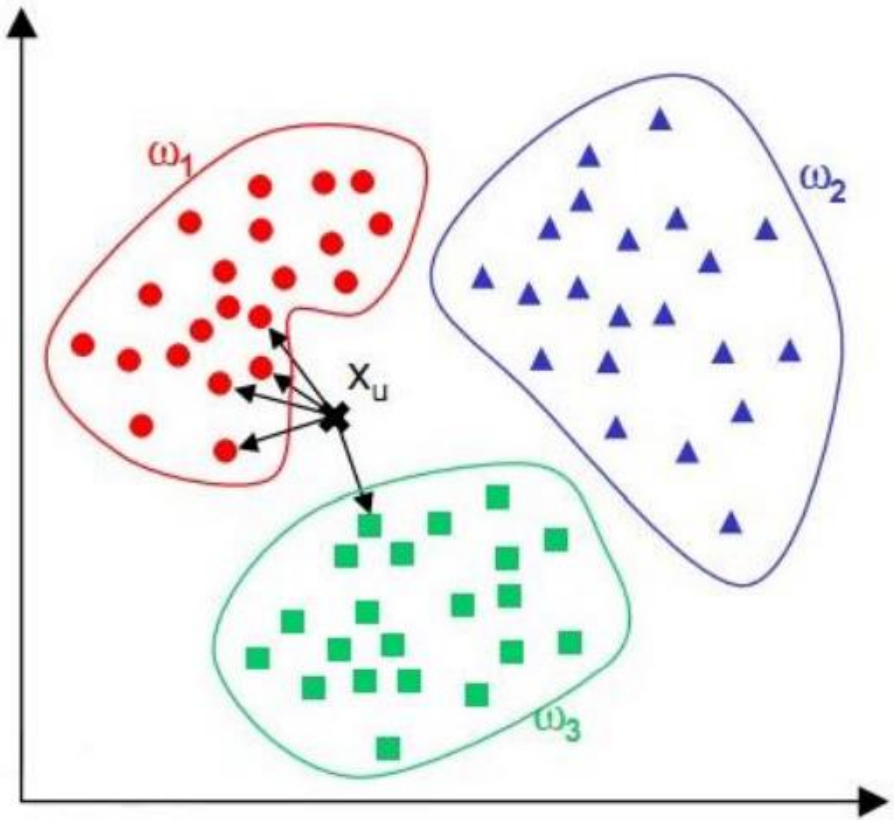
李彦

liyan_zjgsu.163.com

目录

KNN算法

- **K近邻(KNN，k-NearestNeighbor)分类算法**是分类技术中最简单的方法之一：
  - K近邻，即k个最近的邻居，表示每个样本都可以用它最接近的k个邻居样本来代表。
  - 核心思想：若一个样本在特征空间中的k个最相邻样本中的大多数属于某一个类别，则该样本也属于这个类别，并具有该类别样本的特性。
  - **距离+少数服从多数**



kNN算法实现

- 给定一个未知样本，k-最临近分类法搜索模式空间，找出最接近未知样本的k个训练样本；然后使用k个最临近者中最公共的类来预测当前样本的类标签：

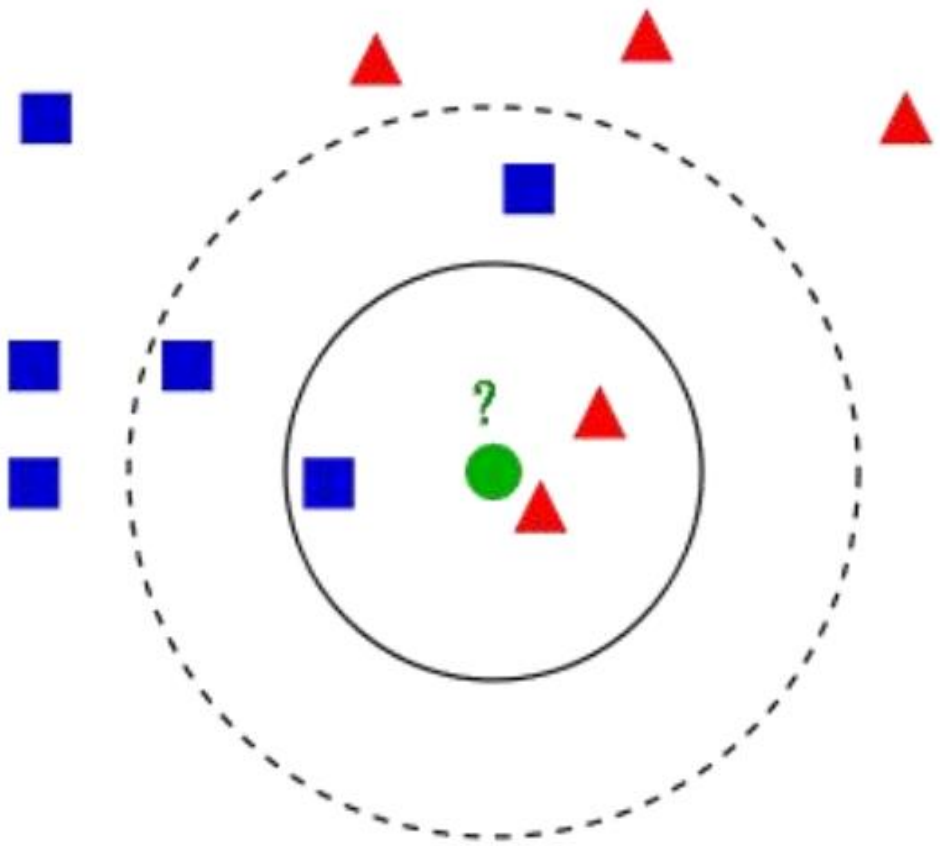
  1. 产生训练集，使得训练集按照已有的分类标准划分成离散型数值类，或者是连续型数值类输出。

  2. 以训练集的分类为基础，对测试集每个样本寻找K个近邻，采用**欧式距离**作为样本间的相似程度的判断依据，相似度大的即为最近邻。

  3. 当类为连续型数值时，测试样本的最终输出为近邻的平均值；当类为离散型数值时，测试样本的最终为近邻类中个数最多的那一类

KNN算法

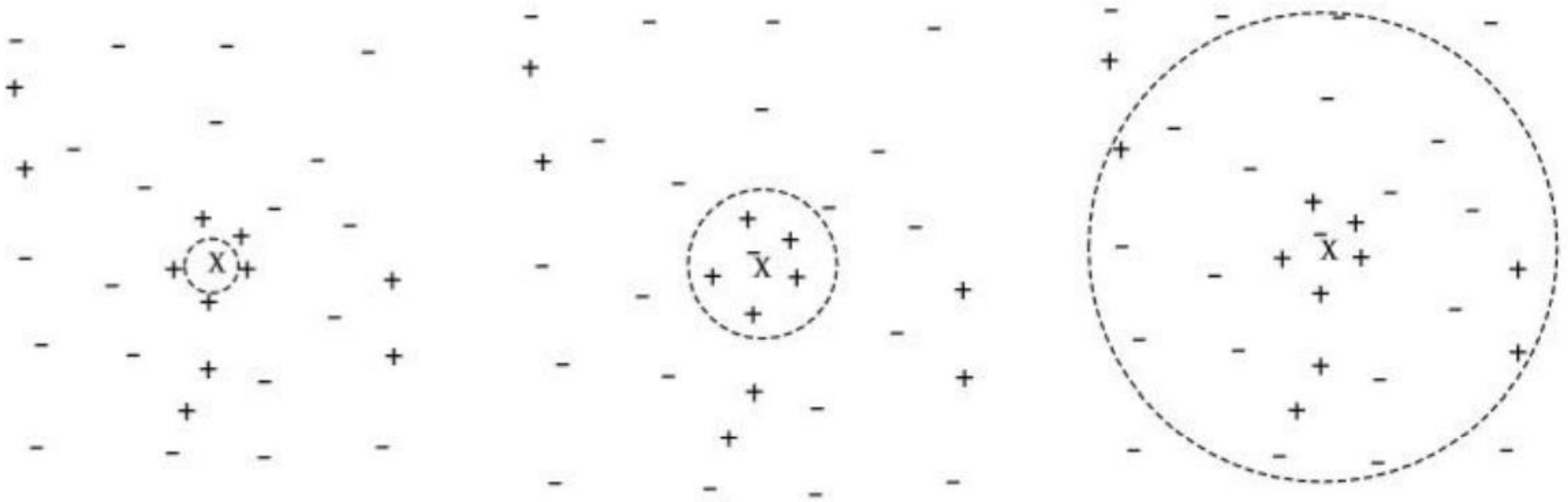
- 如图所示，有两类不同的样本数据，分别用蓝色的正方形和红色的三角形表示。
- 问题：图中的绿色的圆点属于哪一类？
  - 如果K=3，绿色圆点属于红色三角形一类。
  - 如果K=5，绿色圆点属于蓝色正方形一类。



KNN算法——K 值的选择

- **K 值的选取**对KNN学习模型有很大的影响
  - 若K值过小，得到的近邻数过少，会放大噪声数据的干扰，降低分类精度
  - 若k值过大，会有较多的邻域训练样本用来进行预测， 距离较远的训练样本会对预测结果会有贡献，以至于实际上并不相似的数据也可能被包含进来，从而导致分类或者预测效果的降低

K值选取对于预测结果的影响



(a) Neighborhood too small.

(b) Neighborhood just right.

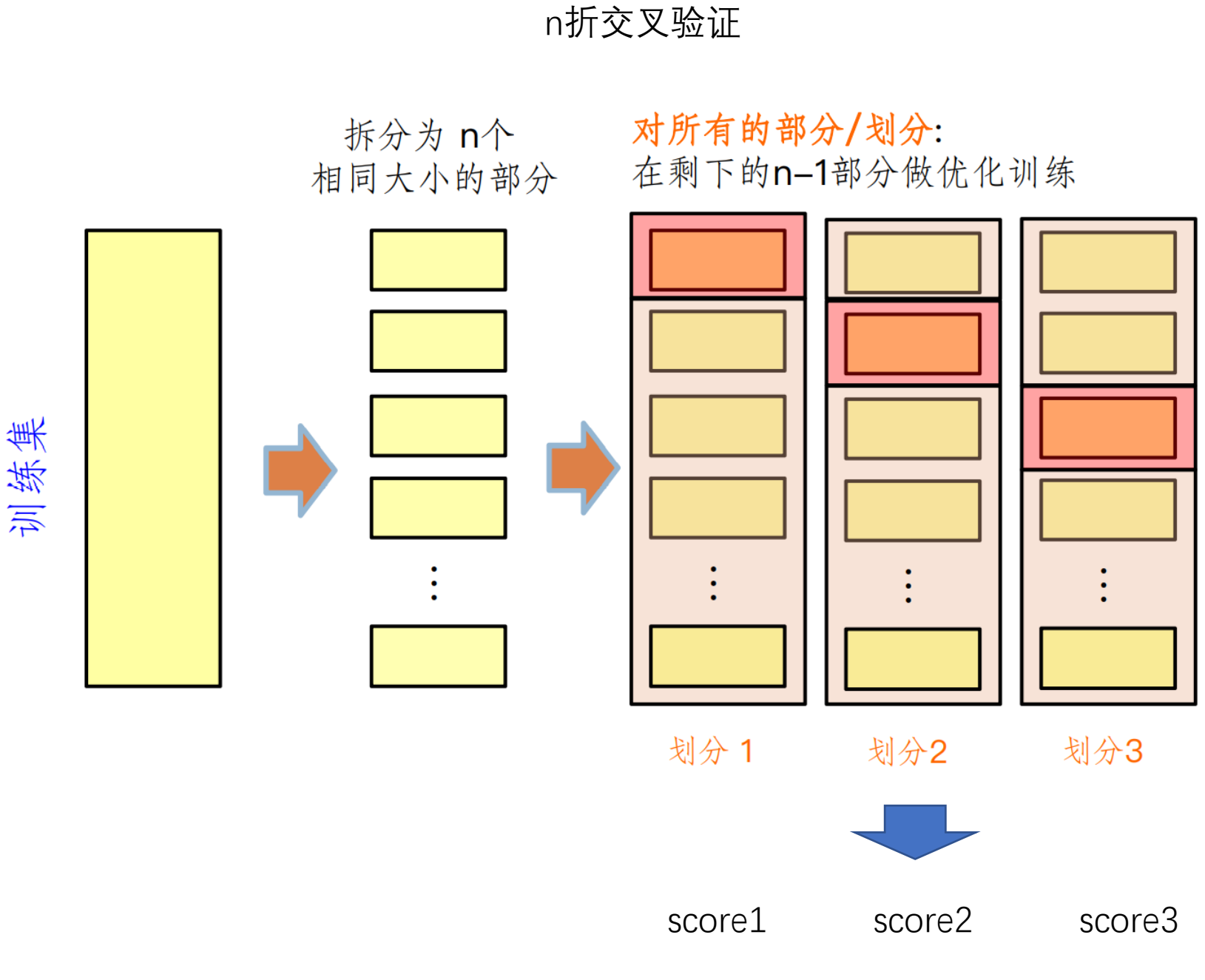
(c) Neighborhood too large.

K值的选择

- K值的设定通常采用**交叉验证**的方式（以K=1为基准）
- 经验规则：K一般低于训练样本数的平方根。
- 交叉验证：亦称循环估计， 是一种统计学上将数据样本切割成较小子集的实用方法。可以先在一个子集上做分析，而其它子集则用来做后续对此分析的确认及验证。开始的子集被称为训练集，而其它的子集则被称为验证集或测试集。
- 交叉验证误差统计选择法就是比较不同K值时的交叉验证平均误差率，选择误差率最小的那个K值。例如选择 K=1,2,3,... ， 对每个 K=i 做交叉验证，计算出平均误差，然后进行比较并选出最小值。

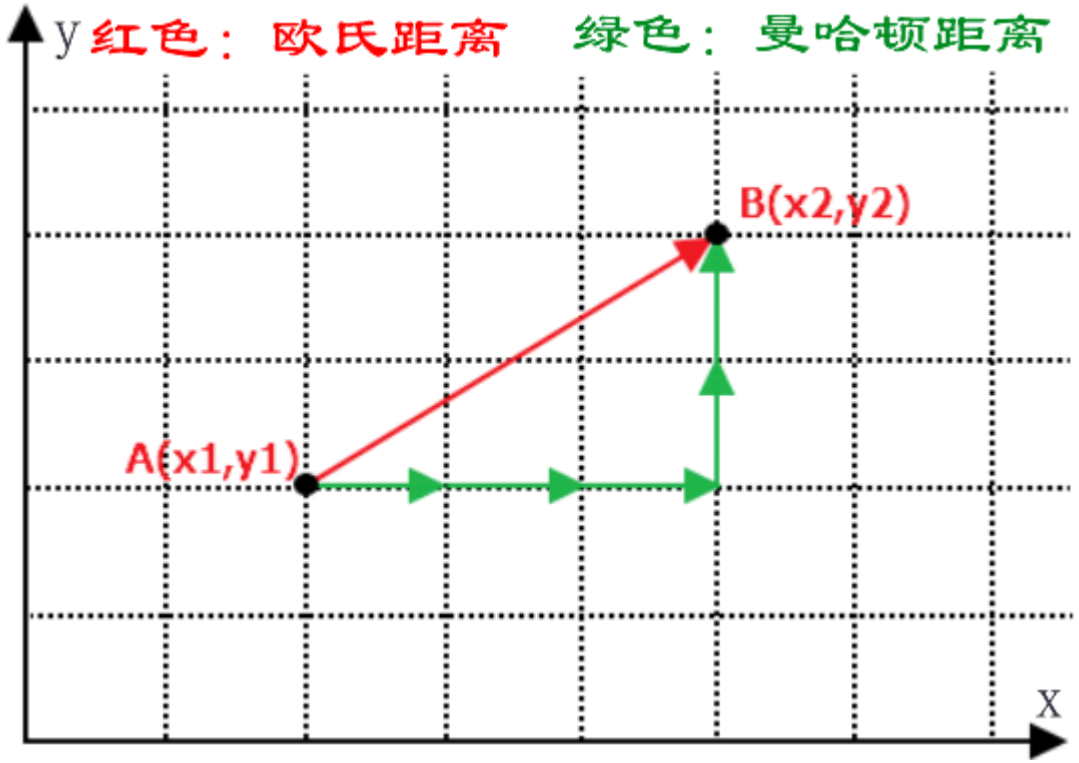
交叉验证

- 更稳健：不仅仅依靠一个测试集来评估方法（或优化参数）
- 需将计算开销乘以n（必须训练n个模型而不是一个）
- 最常见的n包括5和10



距离度量

- 计算距离有许多种不同的方法，如欧氏距离、余弦距离、 汉明距离、曼哈顿距离等等，传统上，kNN算法采用的是欧式距离。



$$\text{欧氏距离}(d) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$\text{曼哈顿距离}(d) = |x_2 - x_1| + |y_2 - y_1|$$

类别判定

- 投票决定：少数服从多数，近邻中哪个类别的点最多就分为该类
- 如果训练数据大部分都属于某一类（不均衡），投票算法就有很大问题了。这时候就需要考虑设计每个投票者的权重。
- **加权投票法**：根据距离的远近，对近邻的点进行加权，距离越近则权重越大（权重为距离平方的倒数）

kNN算法的评价

优点

- 简单，易于理解和实现，无需估计参数，无需训练；
- 适合对稀有事件进行分类；
- 特别适合于多分类问题(multi-modal, 样本具有多个类别标签)
- 对于类域交叉或重叠较多的待分样本集来说，KNN方法较其他方法更为适合

缺点

- 需要存储全部训练样本，计算量较大 （lazy algorithm）
- 可解释性较差，无法给出决策树那样的规则
- 当样本不均衡时（如一个类的样本容量很大，而其他类样本容量很小时），有可能导致当输入一个新样本时，该样本的K个邻居中大容量类的样本占多数。

sklearn中的K近邻分类器

- 在sklearn库中，可以使用*sklearn.neighbors.KNeighborsClassifier* 创建一个K近邻分类器。
- 参见实例

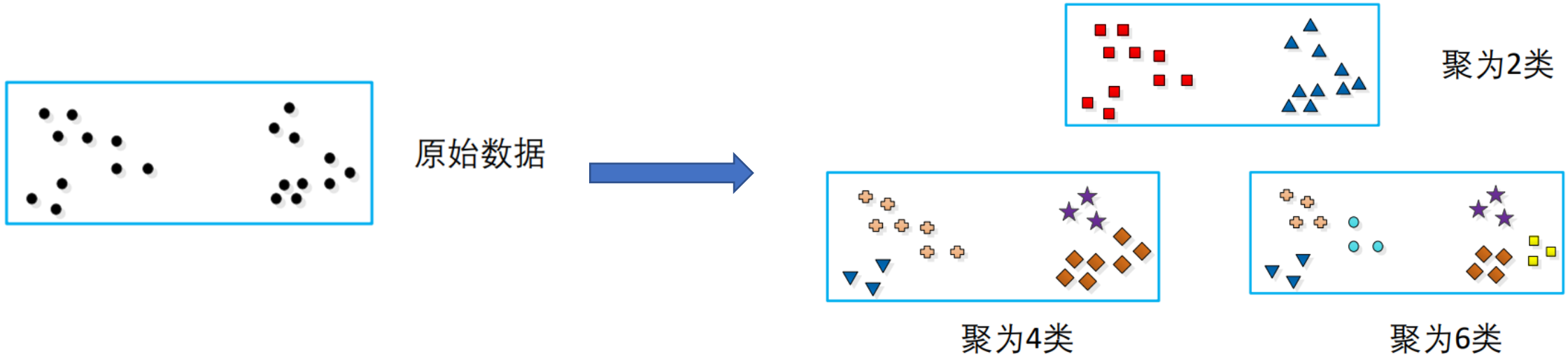
分类分析

- Naive Bayes
- Logistic Regression
- Decision Tree
- kNN
- ……

什么是聚类分析?

- 聚类是无监督学习方法
  - 根据数据内在性质及规律将其划分为若干个不相交的子集，每个子集称为一个“簇”（Cluster）
  - 自动获得的簇需要人为对应“类别”概念
- 聚类可作为分类等其他任务的预处理过程
  - 如电商网站，将用户聚类后，根据簇特性定义用户类，分类进行商品促销
- **目标：*是使同一个簇中的样本相似度较高，而不同簇间的样本相似度较低***

示意图



聚类的金融应用

- 对企业客户进行聚类
  - 贷款、存款、其他业务等
- 为个人客户（家庭）推荐金融服务
  - 收入情况、消费习惯、风险偏好等
- 为保险客户推荐适宜的保险合约等
  - 保险意识、健康需求、年龄层
- 聚类分析在金融投资分析中的应用
  - 股票特征等

聚类分析：变量 vs 样本

聚类分析 聚类对象

- Q型聚类—针对样本
- R型聚类—针对变量

样本

变量

R型聚类

表 3.1 某市 2001 年城镇居民户主个人收入数据

X1 职工标准工资收入
X2 职工奖金收入
X3 职工津贴收入
X4 其他工资性收入
X5 单位得到的其他收入
X6 其他收入
X7 性别
X8 就业身份

X1	X2	X3	X4	X5	X6	X7	X8
540.00	0.0	0.0	0.0	0.0	6.00	男	国有
1137.00	125.00	96.00	0.0	109.00	812.00	女	集体
1236.00	300.00	270.00	0.0	102.00	318.00	女	国有
1008.00	0.0	96.00	0.0	86.0	246.00	男	集体
1723.00	419.00	400.00	0.0	122.00	312.00	男	国有
1080.00	569.00	147.00	156.00	210.00	318.00	男	集体
1326.00	0.0	300.00	0.0	148.00	312.00	女	国有
1110.00	110.00	96.00	0.0	80.00	193.00	女	集体
1012.00	88.00	298.00	0.0	79.00	278.00	女	国有
1209.00	102.00	179.00	67.00	198.00	514.00	男	集体
1101.00	215.00	201.00	39.00	146.00	477.00	男	集体

Q型聚类

表 3.1 某市 2001 年城镇居民户主个人收入数据

X1 职工标准工资收入
X2 职工奖金收入
X3 职工津贴收入
X4 其他工资性收入
X5 单位得到的其他收入
X6 其他收入
X7 性别
X8 就业身份

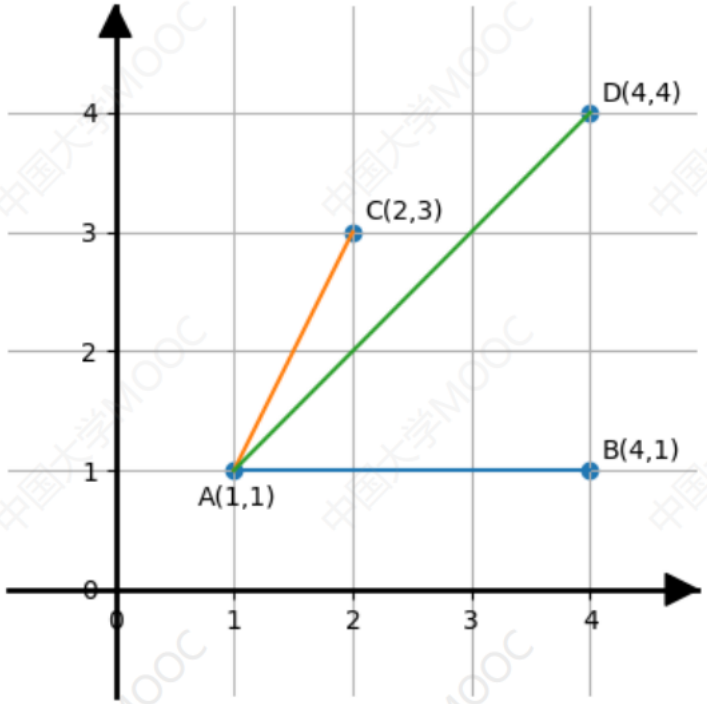
X1	X2	X3	X4	X5	X6	X7	X8
540.00	0.0	0.0	0.0	0.0	6.00	男	国有
1137.00	125.00	96.00	0.0	109.00	812.00	女	集体
1236.00	300.00	270.00	0.0	102.00	318.00	女	国有
1008.00	0.0	96.00	0.0	86.0	246.00	男	集体
1723.00	419.00	400.00	0.0	122.00	312.00	男	国有
1080.00	569.00	147.00	156.00	210.00	318.00	男	集体
1326.00	0.0	300.00	0.0	148.00	312.00	女	国有
1110.00	110.00	96.00	0.0	80.00	193.00	女	集体
1012.00	88.00	298.00	0.0	79.00	278.00	女	国有
1209.00	102.00	179.00	67.00	198.00	514.00	男	集体
1101.00	215.00	201.00	39.00	146.00	477.00	男	集体

相似性度量

- 聚类：在**未知样本类别**的情况下，通过计算样本彼此间的**相似性**来估计样本所属类别
- 样本-距离
  - 欧几里得距离
  - 曼哈顿距离
  - 切比雪夫距离
  - 闵可夫斯基距离
  - ……
- 变量-相似性
  - 相关系数
  - 夹角余弦

相似性度量

- 不同的相似性度量方法可能会带来不同的评价结果，使用合适的相似性度量方法是非常重要的



	(A,B)	(A,C)	(A,D)
曼哈顿距离	3	3	6
欧氏距离	3	2.24	4.24
夹角余弦	0.86	0.98	1

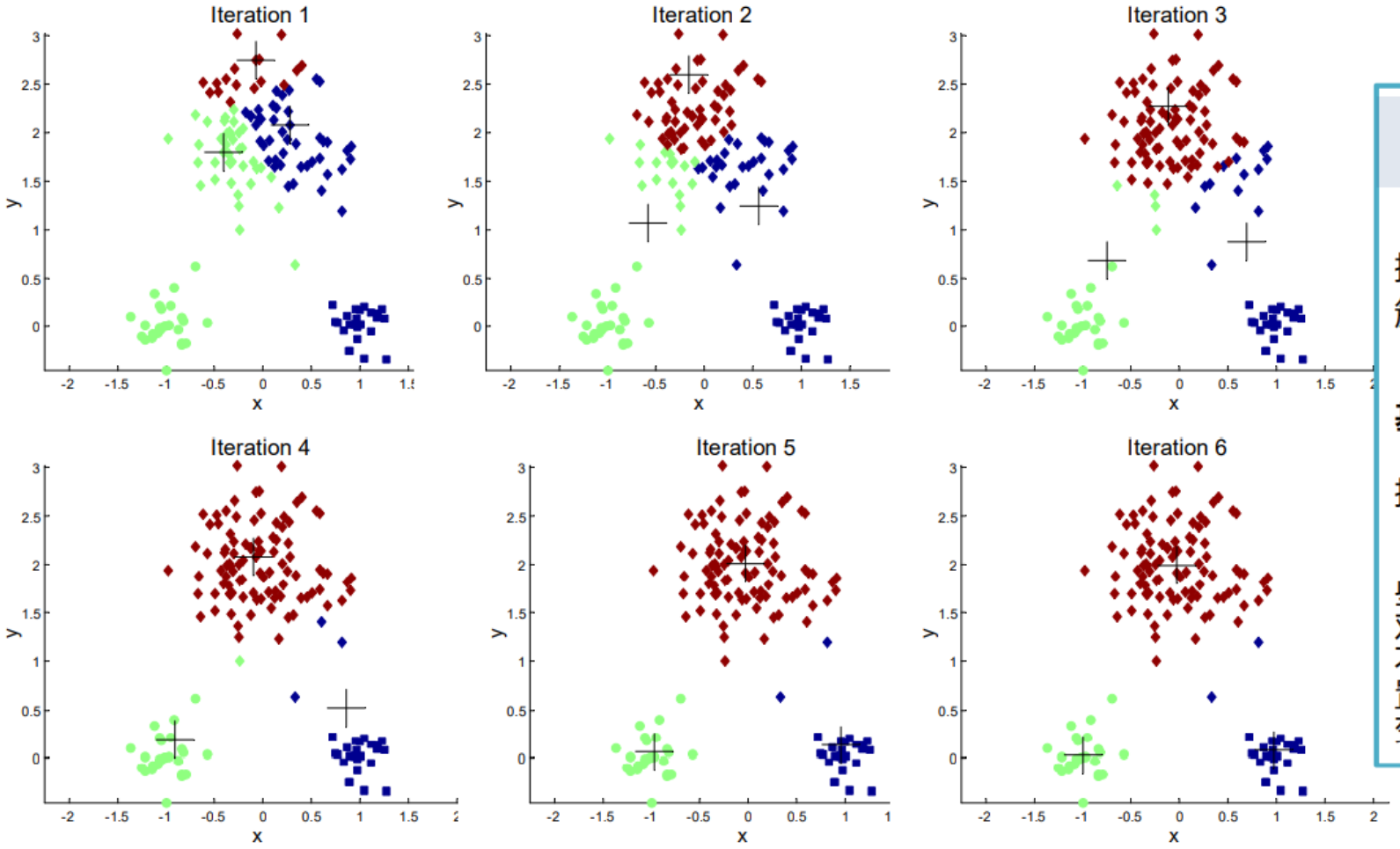
聚类算法分类

- 划分法（Partition）
  - k-means，k-medoids
- 层次法（Hierarchical）
  - 聚合层次聚类（agglomerative hierarchical clustering）： AGNES、BIRCH、ROCK
  - 分裂层次聚类（divisive hierarchical clustering）： DIANA
- 基于密度聚类（Density based）
  - dbscan、optics、denclue
- 基于图/网格聚类（Graph/Grid based）
  - STING、WaveCluster
- 基于模型聚类（Model based）
  - EM、基于神经网络模型、基于概率模型

K-means——均值聚类

- 基于划分的方法通过将样本划分为**互斥**的簇进行聚类，每个样本属于且仅属于一个簇（基于划分的聚类常用算法有k-means、k-medoids）
- k-means算法以k为参数，把n个样本分成k个簇，使簇内具有较高的相似度，而簇间的相似度较低。
- 其处理过程如下：
  - 1.随机选择k个点作为初始的聚类中心；
  - 2.对于剩下的点，根据其与聚类中心的距离，将其归入最近的簇
  - 3.对每个簇，计算所有点的**均值**作为新的聚类中心
  - 4.重复2、3**直到聚类中心不再发生改变**

K-means聚类过程图示



应用实例

利用K-means聚类算法，把原始数据聚成三个不同的簇的应用实例如左图示（K=3）。

基本思路：

（1）首先，随机选择k个数据点做为聚类中心；

（2）然后，计算其它点到这些聚类中心点的距离，通过对簇中距离平均值的计算，不断改变这些聚类中心的位置，直到这些聚类中心不再变化为止。

K-means聚类算法停止条件

- k-means聚类算法的停止条件一般有以下几种:
  - 设定迭代次数。
  - 聚类中心不再变化 。
  - 前后两次聚类结果的目标函数值变化很小。

- 目标：最小化簇内距离的平方和

- 目标函数：$SSE = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} d(\mathbf{x}_i, \mathbf{v}_k)^2$，式中$v_k$是$C_k$的质心（$C_k$中所有样本的均值）

- 设迭代次数为I，给定一个很小的正数δ，如果前后两次迭代结果$|SSE(I) - SSE(I+1)| < \delta$，算法结束；否则继续执行算法。

K-means距离计算-标准化

- K-means算法的核心是相似度的计算
  - 数值型数据，通常采用欧式距离

$$d(A,B)=\sqrt{\sum_{i=1}^{d}(a_i-b_i)^2}$$

- 数据需要先进行标准化处理
  - 将数据按比例缩放，使之落入一个小的特定区间
  - 正规化（Normalization）[-1,1]：$x_i^*=\frac{x_i-\bar{x}}{S}$，其中$\bar{x}$为均值，$S$ 为标准差
  - min-max法[0,1]：$x_i^*=\frac{\max_i(x_i)-x_i}{\max_i(x_i)-\min_i(x_i)}$

余弦相似度

- 离散（nominal）数据，余弦相似度
  - 如文本的相似度
  - 余弦值范围：[-1,1]，越接近1，两个样本相似度越高

$$cos(A,B) = \frac{A \cdot B}{|A| \times |B|} = \frac{\sum_{i=1}^{d} a_i \times b_i}{\sum_{i=1}^{d}(a_i)^2 \times \sum_{i=1}^{d}(b_i)^2}$$

例：　$A = [3\ 2\ 0\ 5\ 0\ 0\ 0\ 2\ 0\ 0]$

$B = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 2]$

值：特征词（如：我、住、上海、……）10个
在A、B两段文本中出现的次数

$A \bullet B = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$

$|A| = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)^{0.5} = (42)^{0.5} = 6.481$

$|B| = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{0.5} = (6)^{0.5} = 2.245$

$\cos(A, B) = .3150$

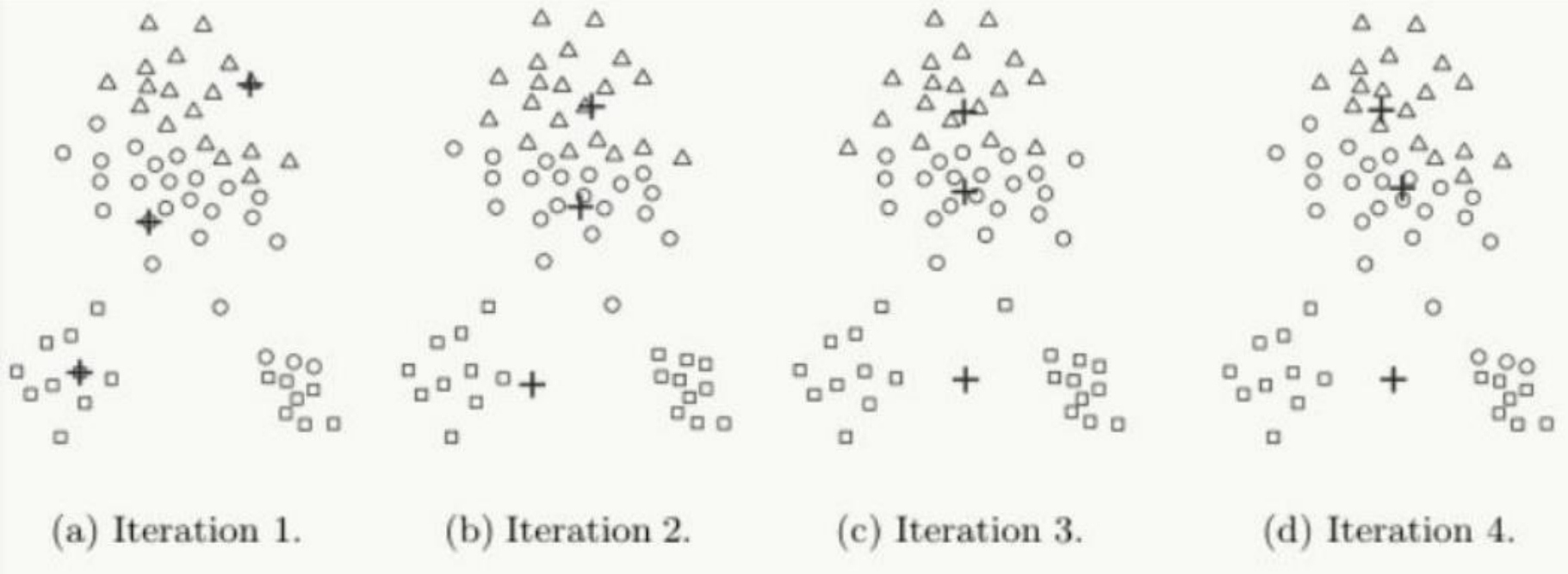
K-means算法评价

- 优点:
- 原理简单，容易实现，且运行效率比较高
- 聚类结果容易解释，适用于高维数据的聚类

- 缺点:
- 需要人为设置簇的个数与随机初始化质心点可能影响聚类的最终效果
- 算法对孤立点（离群点）特别敏感，会对最终的聚类结果产生明显扰动。
- 采用贪心策略，导致容易局部收敛，在大规模数据集上求解较慢

K-means算法局限性-初始质心位置

- k-means是局部最优的，容易受到初始质心的影响。比如在下图中，因**选择初始质心不恰当**而造成次优的聚类结果 （SSE较大）：



K-means算法改进——K-means++

- k-均值算法中初始聚类中心的选取对算法结果影响很大，不同的初始中心可能会导致不同的聚类结果。对此，研究人员提出k-均值++算法，其核心思想是：**使初始的聚类中心之间的相互距离尽可能远**。

k-均值++算法步骤如下：

- 从样本集χ中随机选择一个样本点c_1作为第1个聚类中心；
- 计算其它样本点x到最近的聚类中心的距离$d(x)$；
- 以概率$\frac{d(x)^2}{\sum_{x\in\chi} d(x)^2}$选择一个新样本点$c_i$加入聚类中心点集合中，其中距离值$d(x)$越大，被选中的可能性越高；
- 重复步骤(2)和(3)选定k个聚类中心；
- 基于这k个聚类中心进行k-均值运算

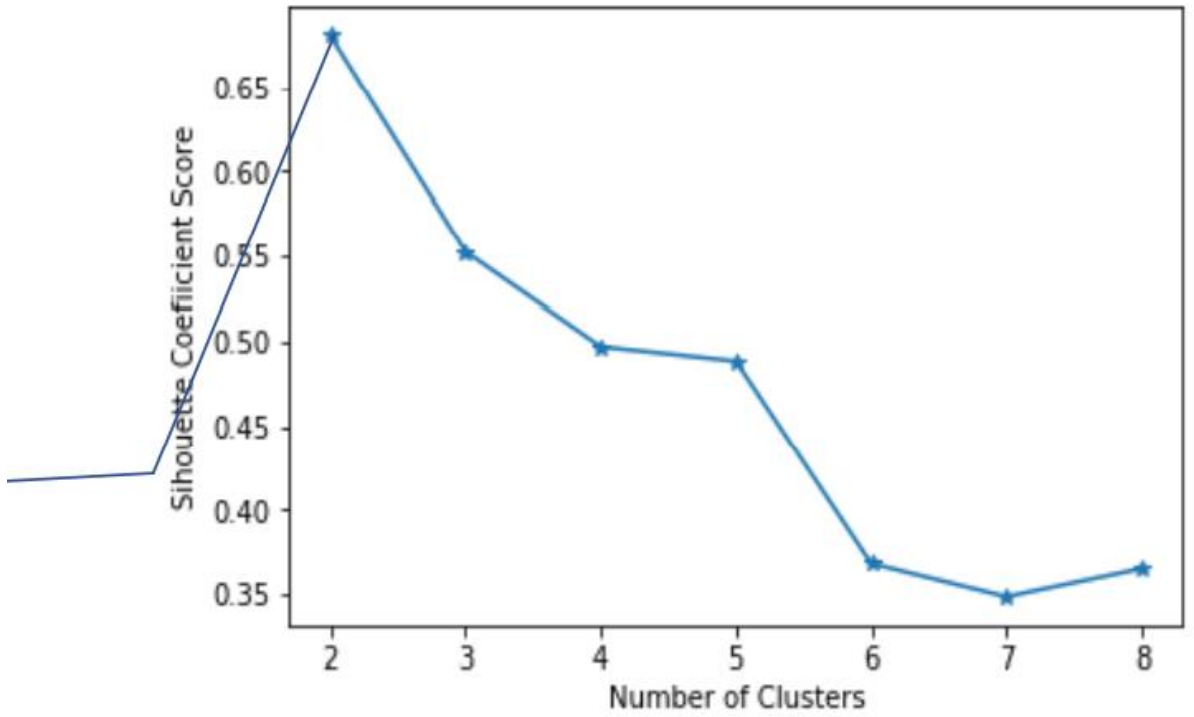
K-medoids

- k-均值算法簇的聚类中心选取受到噪声点的影响很大，因为噪声点与其他样本点的距离远，在计算距离时会严重影响簇的中心。
- k-中心算法克服了 k-均值算法的这一缺点， k-中心算法**不通过计算簇中所有样本的平均值得到簇的中心，而是通过选取原有样本中的样本点作为代表对象代表这个簇**，计算剩下的样本点与代表对象的距离，将样本点划分到与其距离最近的代表对象所在的簇中。
- 步骤：在k-中心算法的迭代过程中，不断用非中心点替换中心点，尝试所有可能的替换情况，直到聚类的质量不能再被提升为止 。

k值选择：轮廓系数

- 最优的类别个数未知，不同的类别个数会导致不一样的结果。
- 尝试用多种类别个数分别进行聚类，通过分析聚类结果的质量，推测最优的K值。

选取轮廓系数最大的k值

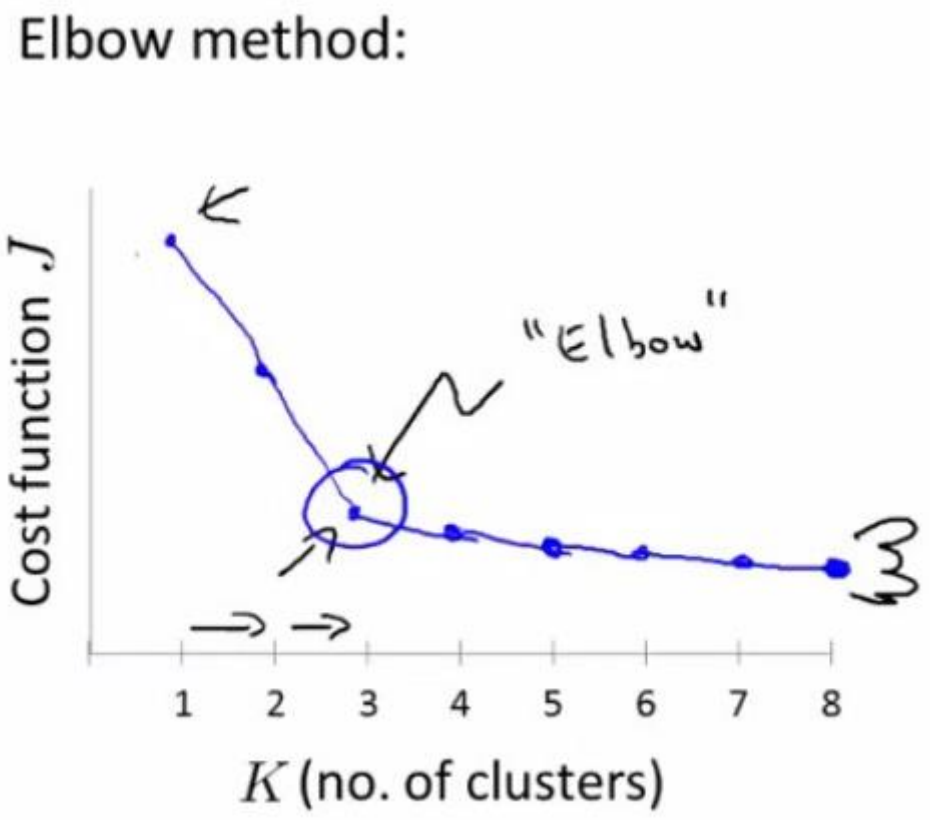


k值选择：肘部法则

- Cost Function:

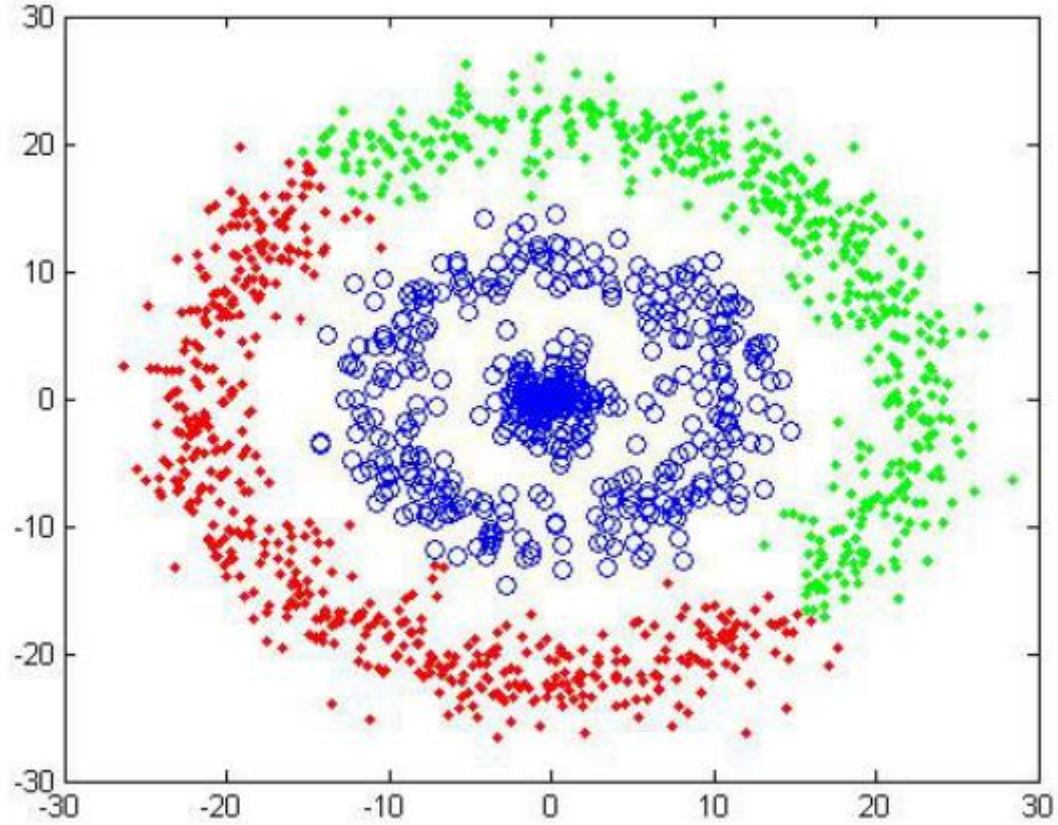
$$SSE = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} d(\mathbf{x}_i, \mathbf{v}_k)^2$$

- 核心思想：随着聚类数k的增大，样本划分会更加精细，每个簇的聚合程度会逐渐提高，那么误差平方和SSE自然会逐渐变小。当k小于真实聚类数时，由于k的增大会大幅增加每个簇的聚合程度，故SSE的下降幅度会很大；而当k到达真实聚类数时，再增加k所得到的聚合程度回报会迅速变小，所以SSE的下降幅度会骤减，然后随着k值的继续增大而趋于平缓。



K-means算法局限性

- k-均值算法**不适用于非凸面形状（非球形）的数据集**，例如图中例子，k-均值算法的聚类结果就与初始目标有非常大的差别



K-means聚类实现

- Scikit-learn的聚类： Cluster 类

模型初始化：kmeans = KMeans(n_clusters)

模型学习：kmeans.fit(X)

参数说明：	
n_clusters	簇的个数
X	特征二维数组，数值型

- 数据标准化方法： preprocessing类

```
from sklearn import preprocessing

X_scale = preprocessing.scale(X)    #scaled之后的数据均值为0，方差为1
```

聚类方法性能评估

- **有分类标签**的数据集
  - 使用**兰德指数**（ARI，Adjusted Rand Index）
  - 计算真实标签与聚类标签两种分布之间的相似性，取值范围为[0,1]
  - 1表示最好的结果，即聚类类别和真实类别的分布完全一致
  - 鸢尾花数据集带有标签

```
from sklearn import metrics
metrics.adjusted_rand_score(y, kmeans.labels_)
```

- **没有分类标签**的数据集
  - 使用**轮廓系数**（Silhouette Coefficient）来度量聚类的质量
  - 轮廓系数同时考虑聚类结果的簇内凝聚度和簇间分离度
  - 取值范围：[-1,1]，轮廓系数越大，聚类效果越好

```
from sklearn import metrics
metrics.silhouette_score( X,kmeans.labels_,met
    ric='euclidean' )
```

兰德指数（Rand index, RI）

- RI取值范围为[0,1]，值越大意味着聚类结果与真实情况越吻合。
- 如果有了类别标签，那么聚类结果也可以像分类那样计算准确率和召回率。
- 假设U是外部评价标准，即true_label，而V是聚类结果，根据[元素对]在类标签和聚类结果中的表现设定4个统计量：

	Same Cluster	Different Cluster	Sum U
Same Class	TP / a	FN / b	a+b
Different Class	FP / c	TN / d	c+d
Sum V	a+c	b+d	a+b+c+d

- RI定义为“正确决策”的比率：

$$RI = \frac{TP + TN}{TP + FP + TN + FN} = \frac{TP + TN}{C_N^2} = \frac{a + b}{C_2^{n_{samples}}}$$

轮廓系数（Silhouette Coefficient）

- 对于n个样本的数据集D，假设D被划分为k个簇。
- 对于D中的每个样本o，计算它和它所属的簇的其他样本的平均距离a(o)。类似的，计算它和它不属于的簇中样本的最小平均距离b(o)。
- **对数据集中的每个样本计算轮廓系数，然后取平均值作为聚类的质量度量**。
- 轮廓系数的取值范围是[-1,1]，**越靠近1代表聚类质量越好，越靠近-1代表聚类质量越差**。

轮廓系数（Silhouette Coefficient）

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

$$s(i)=\begin{cases}1-\dfrac{a(i)}{b(i)}, & a(i)<b(i)\\ 0, & a(i)=b(i)\\ \dfrac{b(i)}{a(i)}-1, & a(i)>b(i)\end{cases}$$

- 计算样本i到同簇其他样本的平均距离ai，ai越小，说明样本i越应该被聚类到该簇。将ai称为样本i的**簇内不相似度**。
- 计算样本i到其他某簇Cj的所有样本的平均距离bij，称为样本i与簇cj的不相似度。定义bi为样本i的**簇间不相似度**：bi=min{bi1,bi2,.....bik}
- si接近1说明样本i聚类合理；si接近-1说明样本i应该分类到另外的簇；若si近似为0说明样本i在两个簇的边界上。

Thank you