无监督学习-聚类

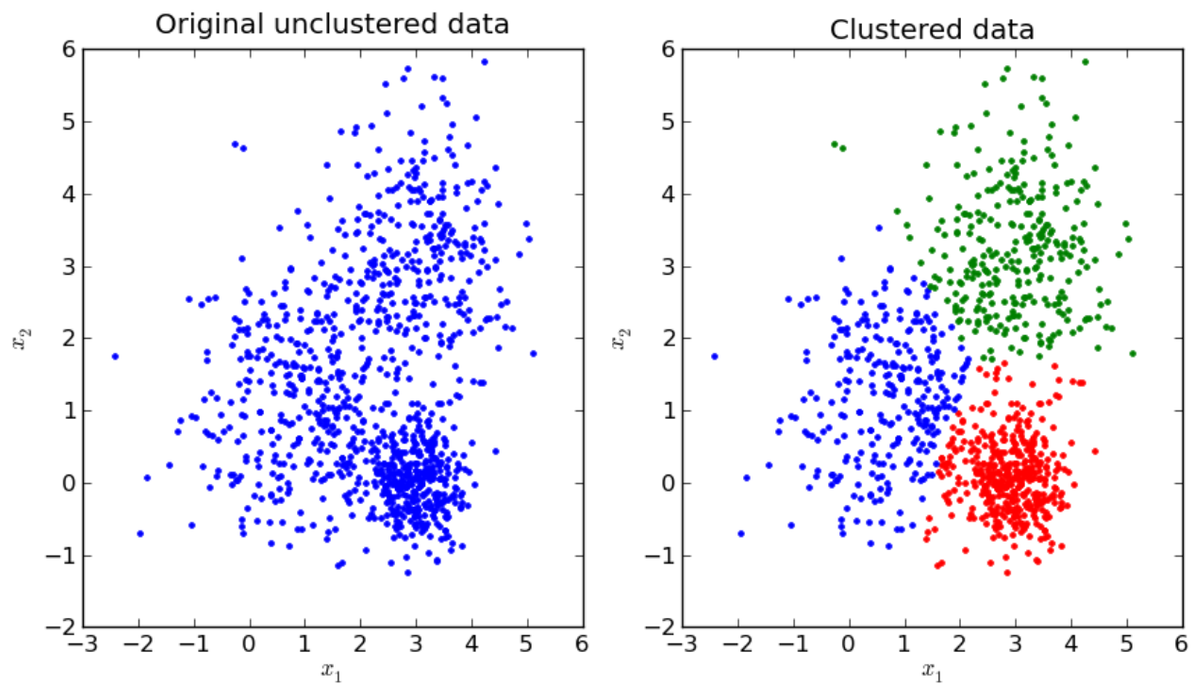
Unsupervised Learning Clustering

- 聚类
- K-Means（K均值）
- 层级聚类
- 高斯混合模型
- 实践

聚类

聚类的任务

- 聚类的目标：将数据集中的样本划分为若干个通常不相交的子集（有时称为簇或cluster）
- 聚类可以看作一个单独的任务，用来寻找数据内部的潜在特征，也可以作为分类任务的一个前处理
- 在聚类任务中，数据是没有被标记好的



聚类的性能指标

- 我们希望“物以类聚”，即同一簇的样本尽可能彼此相似，不同簇的样本尽可能不同。换言之，聚类结果的“簇内相似度”（intra-cluster similarity）高，且“簇间相似度”（inter-cluster similarity）低，这样的聚类效果较好.

如何衡量两个点相似

- 衡量x1与x2的相似度，其中

$$x_1 = (x_{11}, x_{12}, \ldots, x_{1d})$$
$$x_2 = (x_{21}, x_{22}, \ldots, x_{2d})$$

- 可以通过计算x1与x2的距离来衡量二者的相似度
  - 欧式距离 $d_{12} = \sqrt{\sum_{i=1}^{d} (x_{1i} - x_{2i})^2}$
  - 曼哈顿距离 $d_{12} = \sum_{i=1}^{d} |x_{1i} - x_{2i}|$
  - 余弦相似度 $cos\theta = \frac{|x_1 \cdot x_2|}{|x1||x2|}$

如何衡量两个点相似

- 可以通过计算x1与x2的距离来衡量二者的相似度
  - 皮尔逊相关系数

$$\rho_{x_1,x_2} = \frac{conv(x_1, x_2)}{\delta_{x_1}\delta_{x_2}} = \frac{E((x_1 - \overline{x_1})(x_2 - \overline{x_2}))}{\delta_{x_1}\delta_{x_2}} = \frac{E(x_1x_2) - E(x_1)E(x_2)}{\sqrt{E(x_1^2) - E(x_1)^2}\sqrt{E(x_2^2) - E(x_2)^2}}$$

K-Means(K均值)

K-Means

- 当给定数据集 $X = x_1, x_2, \ldots, x_N, x_i = (x_{i1}, x_{i2}, x_{id})$
- 我们自动的将X划分为k个簇中
- 每一个簇都有一个中心点 $c_k = (c_{k1}, c_{k2}, c_{kd})$
- k是用户提前指定的

K-Means算法流程

- 指定k

1. 随机初始化k个簇的中心点
2. 将X中的每个点规划到与它距离最近的那个中心点所在的簇
3. 使用2的结果重新计算每个簇中心点的坐标
4. 如果没达到收敛的条件，则重复2、3步

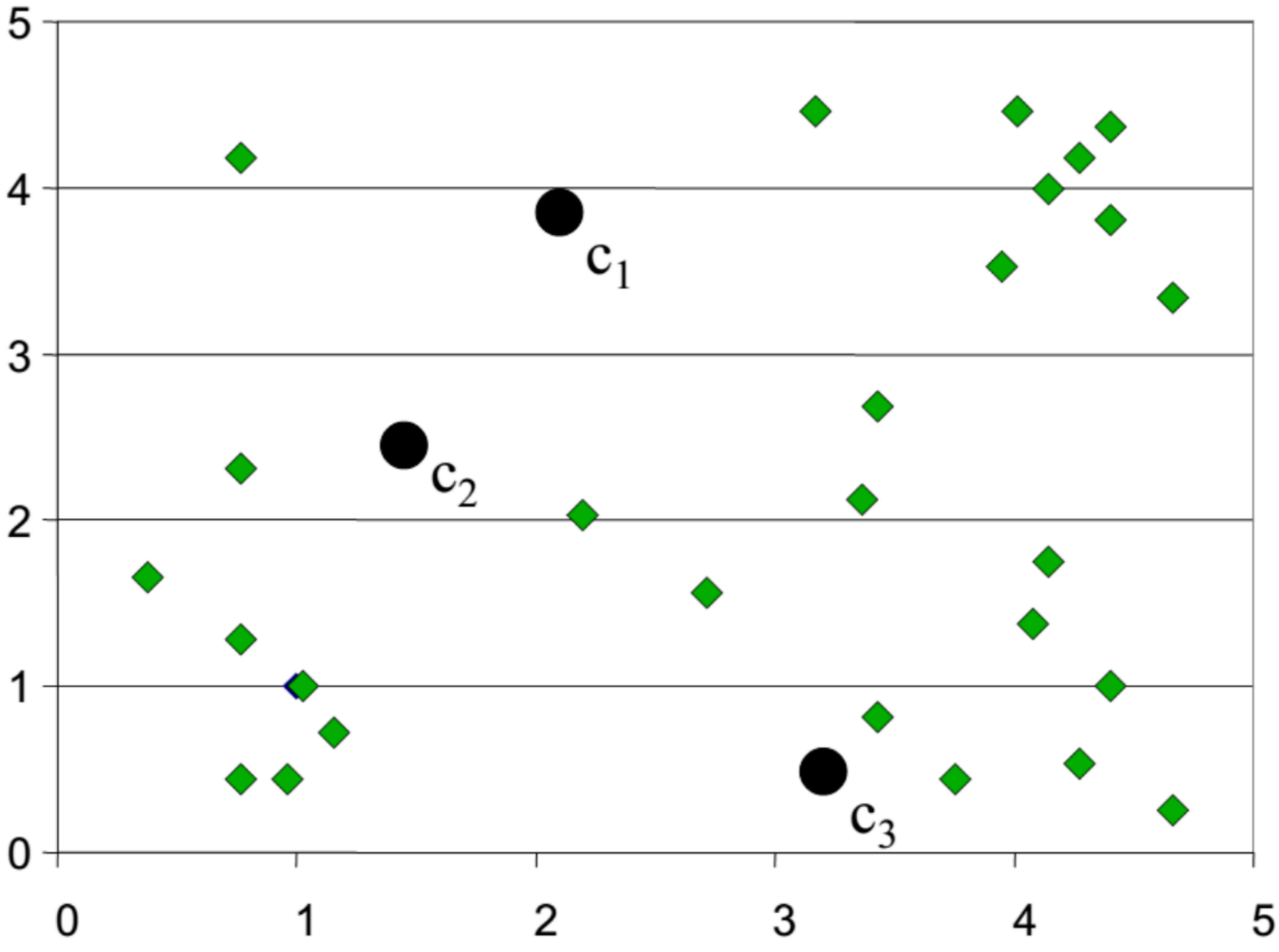
K-Means收敛（停止）条件

- 达到下列任意一条则k-means停止迭代

  1. 没有任何点被重新划分
  2. 中心点没有变化
  3. 达到迭代次数

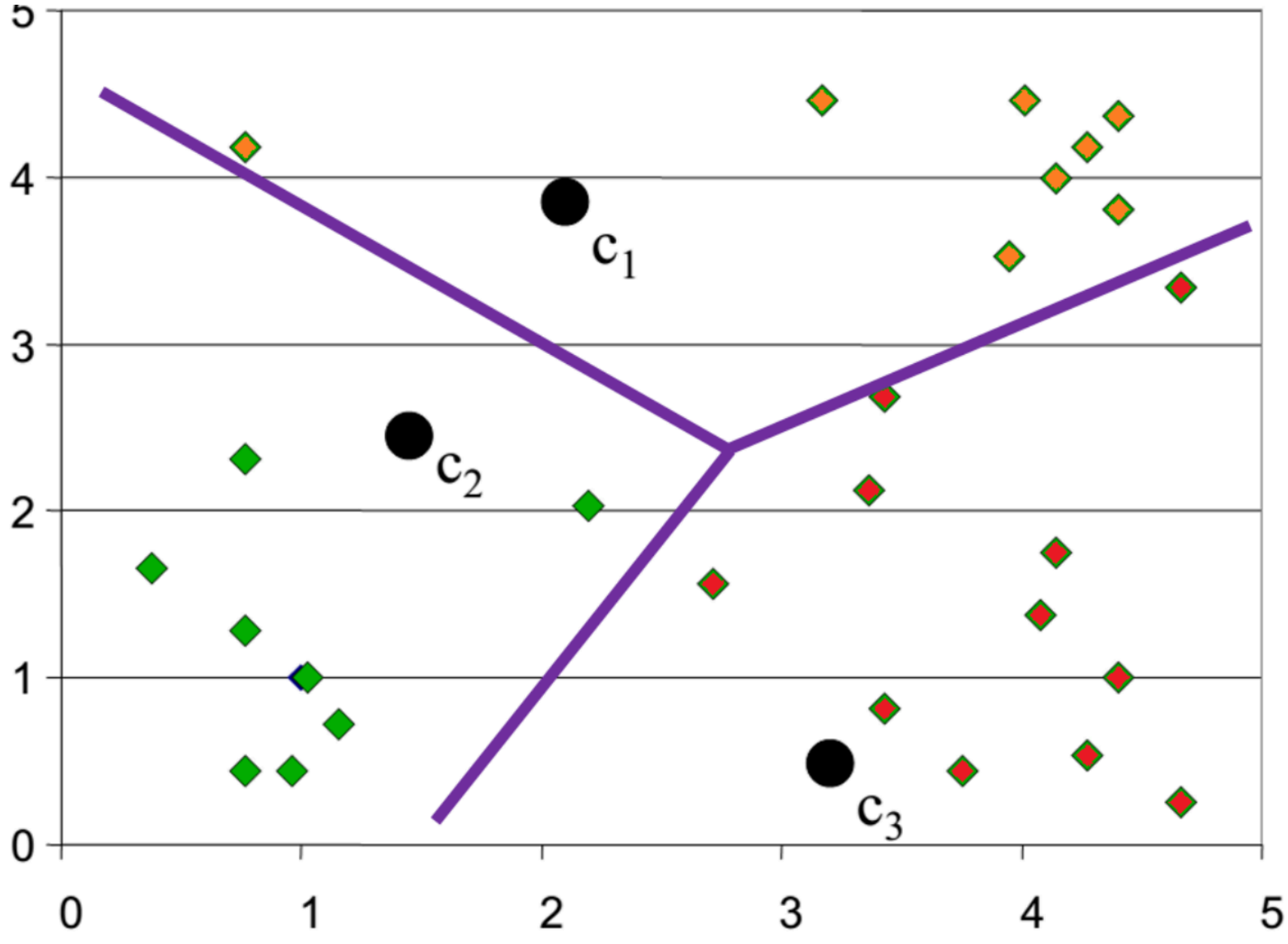
K-Means聚类过程

设定k=3，随机初始化三个中心点c1, c2, c3



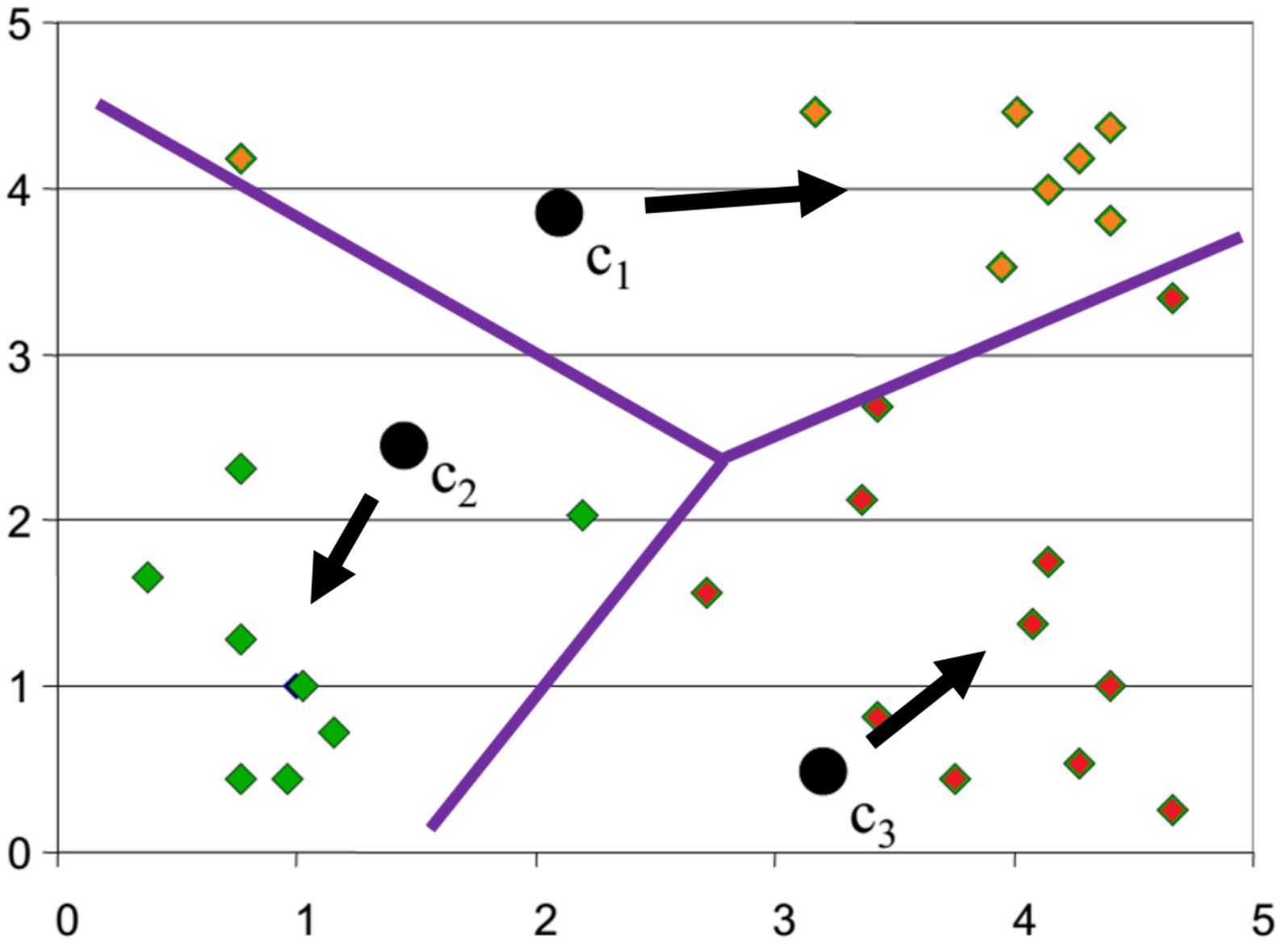
K-Means聚类过程

将数据集X中的每个点划分到与它最近的中心点所在的簇中



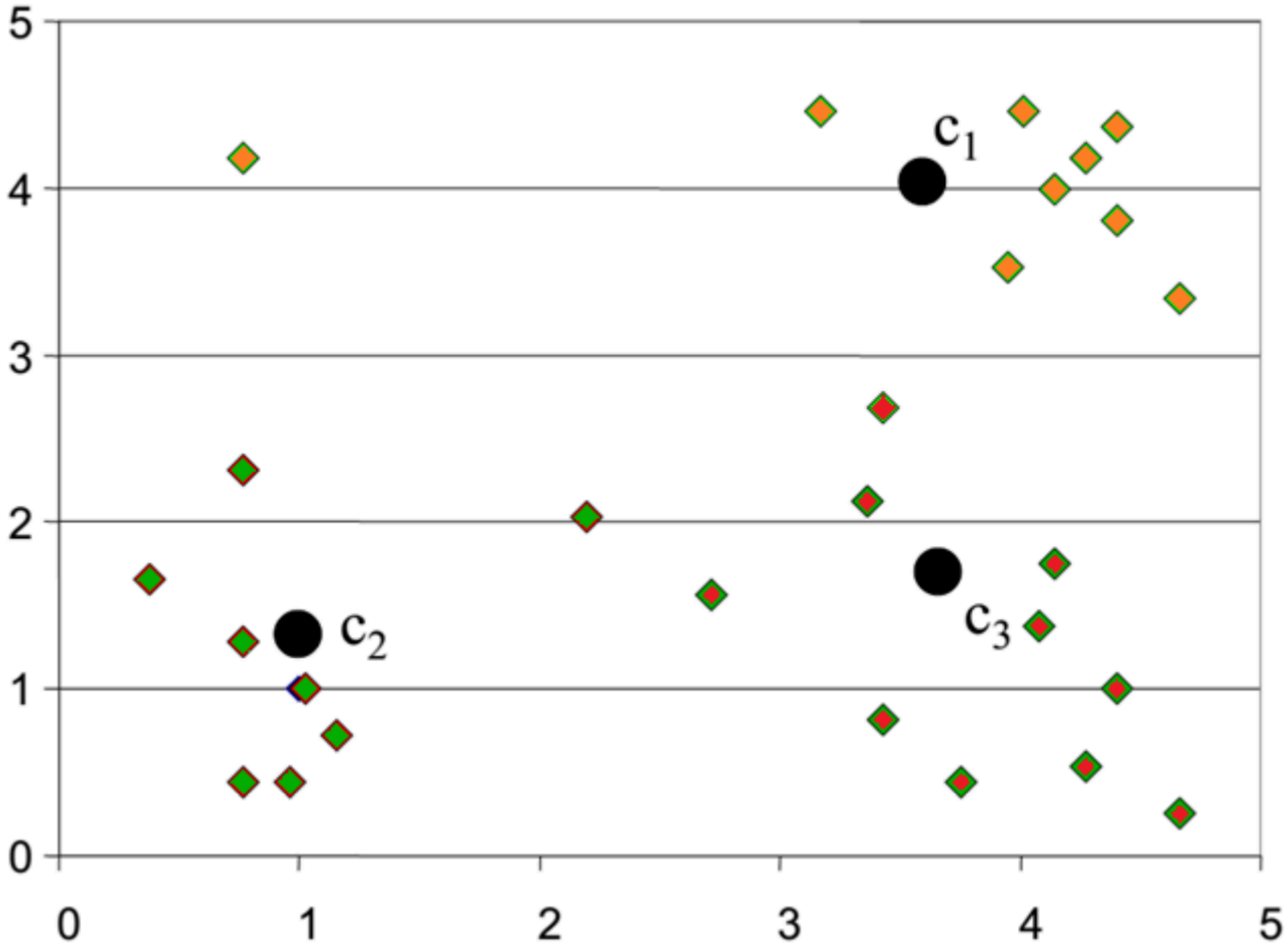
K-Means聚类过程

重新计算中心点坐标

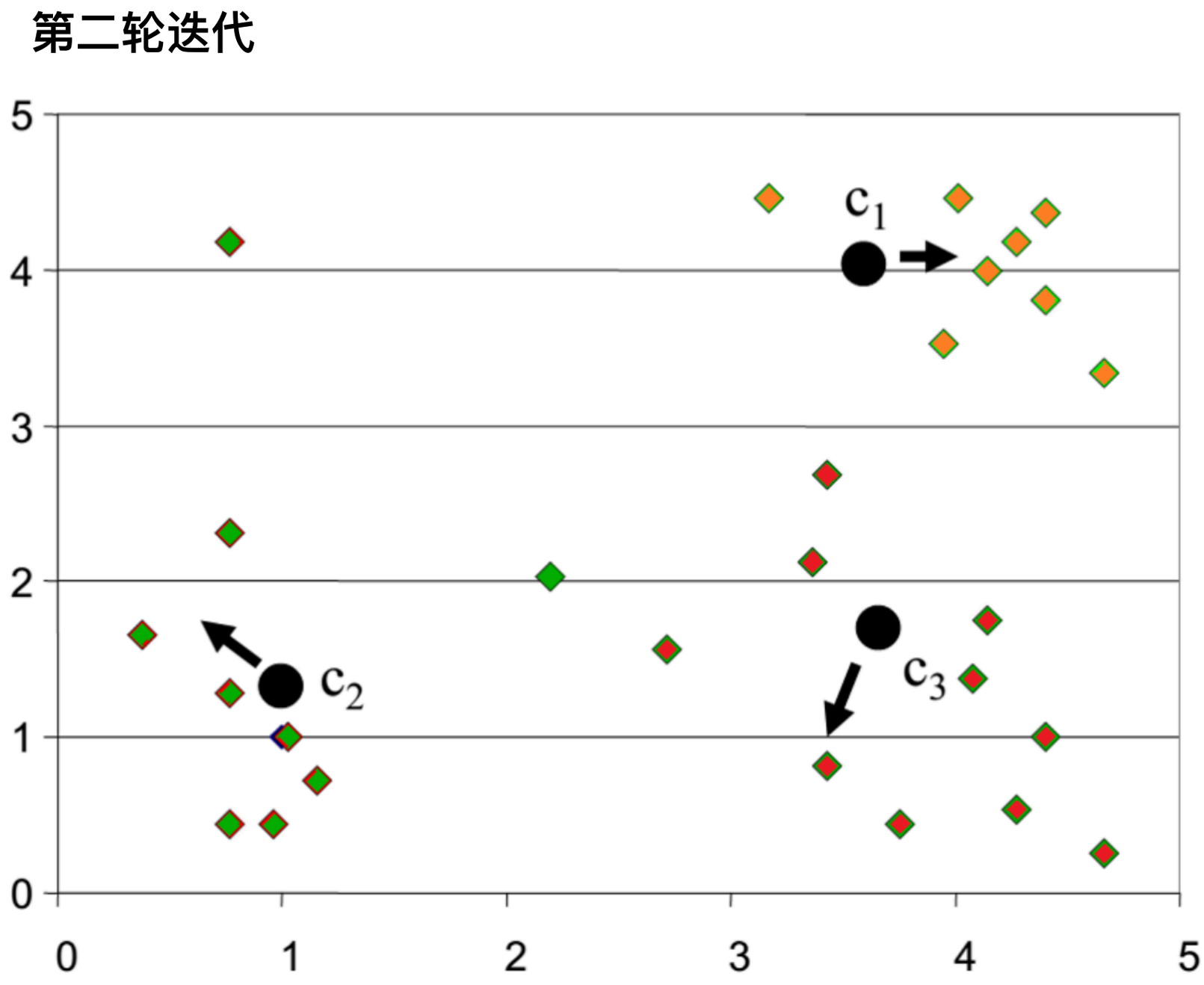


K-Means聚类过程

第一次迭代的结束

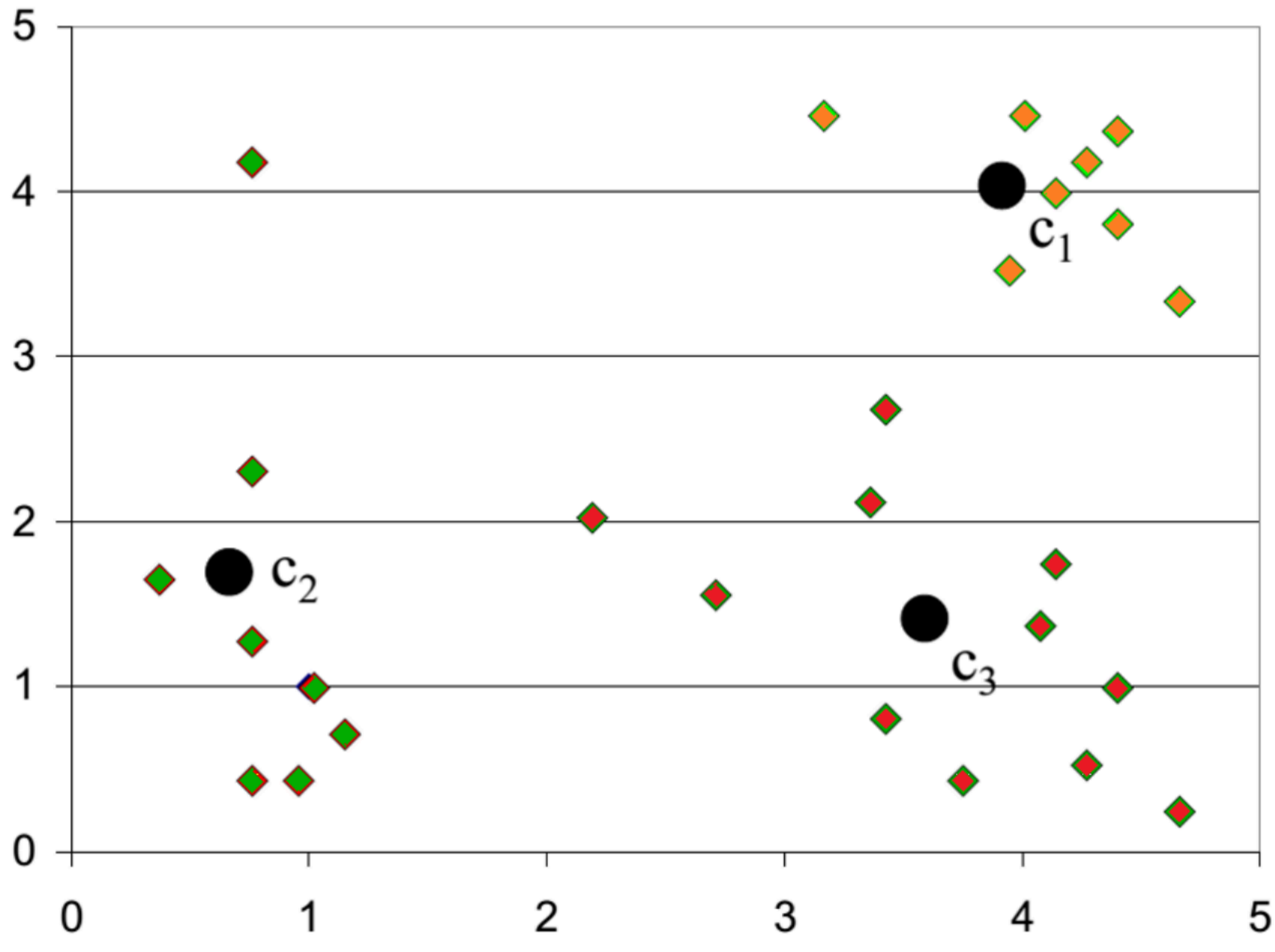


K-Means聚类过程



K-Means聚类过程

第二轮迭代结束

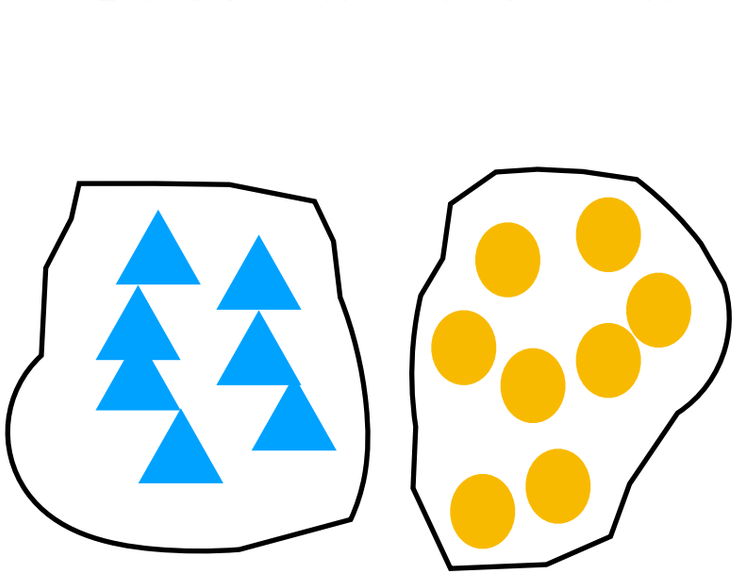


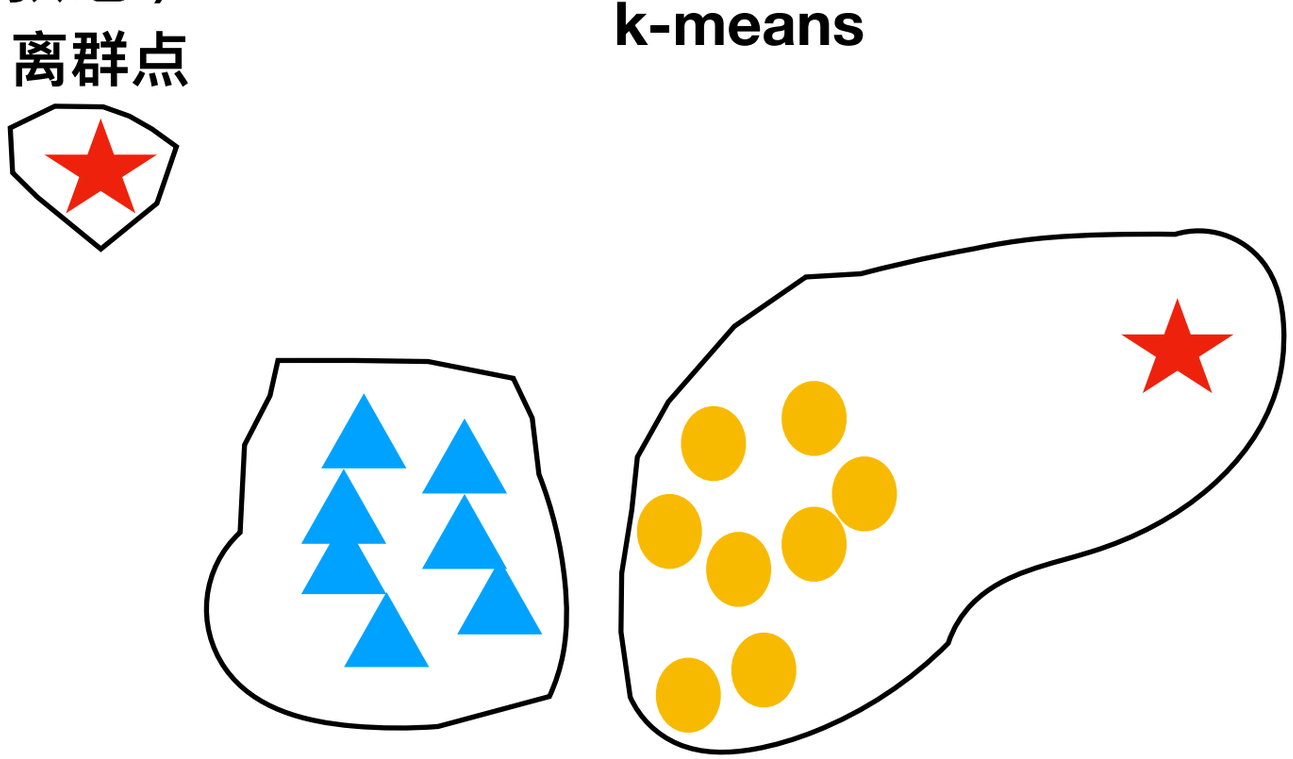
K-Means优点

- 算法简单
- 时间复杂度为o(tkn)
  - k是簇的数目
  - t是迭代数目
  - n是数据集中数据量
- K-Means是非常优秀的一个聚类算法

K-Means缺点

- 需要指定k
- 对离群点（异常点比较敏感）





如何选择k

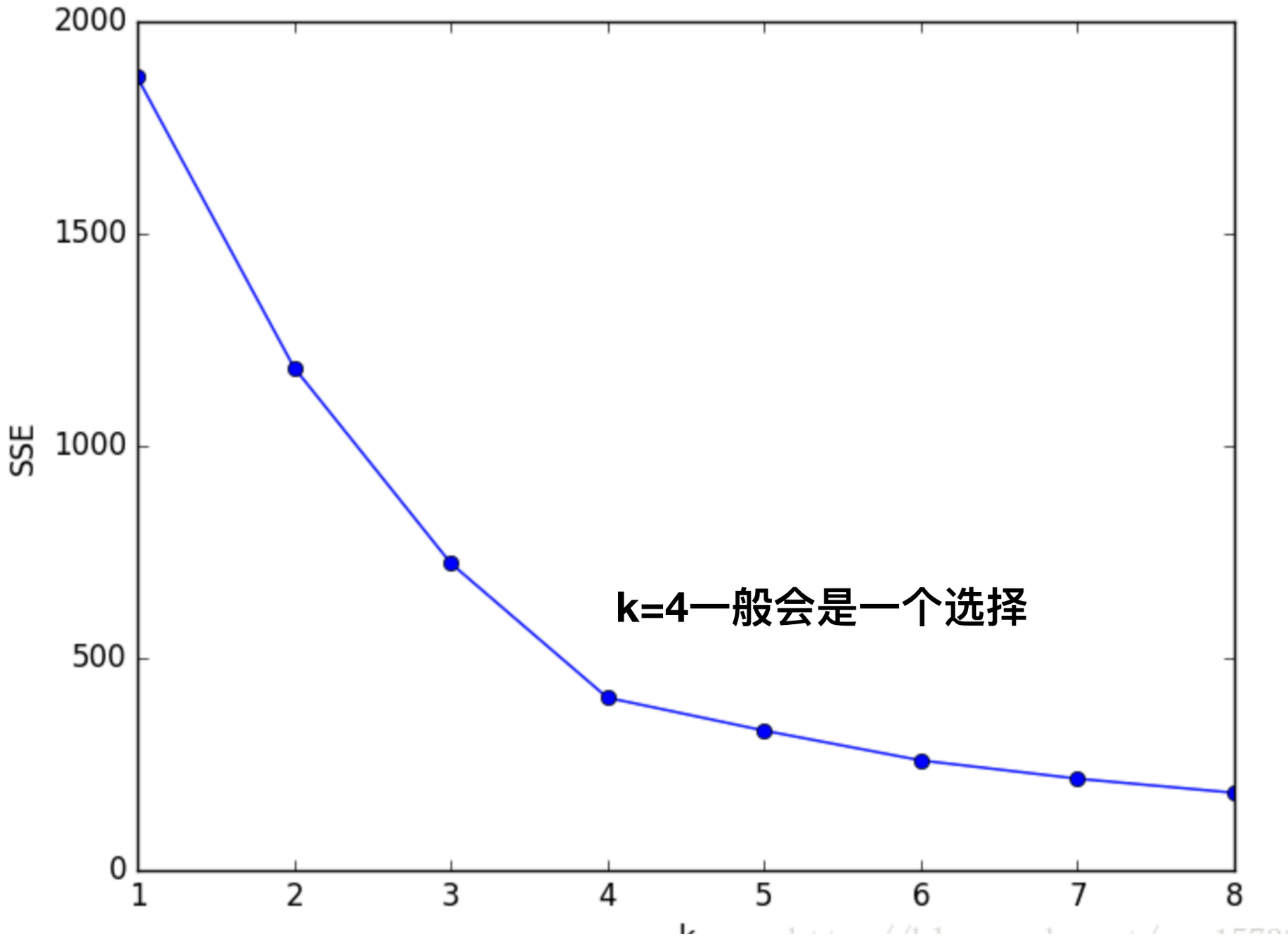
1. k=sqrt(N/2)

2. 手肘法

- 通过计算不同k下，聚类的误差合 $SSE = \sum_{i=1}^{k} \sum_{x \in C_i} |x - c_i|^2$

- 当k小于真实簇的数目时，随着k的增大SSE会大幅度的降低，当N接近真实簇的数目后，SSE会慢慢的变得很平缓

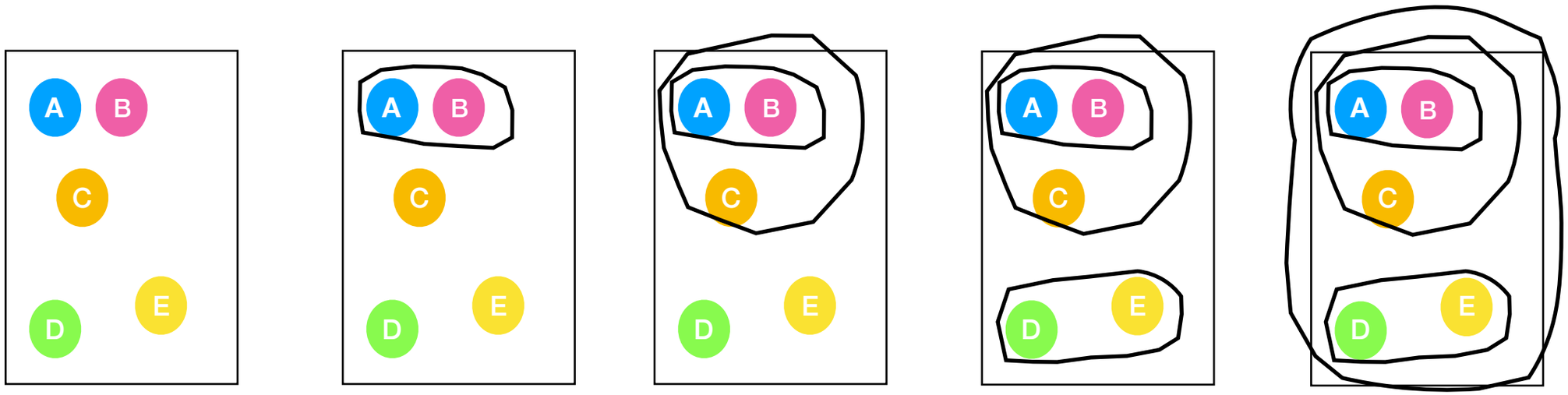
如何选择k



层级聚类

层级聚类

- 连续不断的将最相似的群组两两合并，来构造一个群组的层级结构



如何衡量两个群组相似

- 初始化时，可以认为每个群组只有一个点，那么群组的相似度可以认为是点之间的相似度

- 群组有多个点的时候：
  - average-linkage: 计算两个群组之间两两数据点之间的距离取平均（一般用average-linkage）
  - single-linkage/complete-linkage： 选择两个cluster中距离最短/最长的一对数据点的距离作为类的距离
  - ward 将有最小方差的簇合并

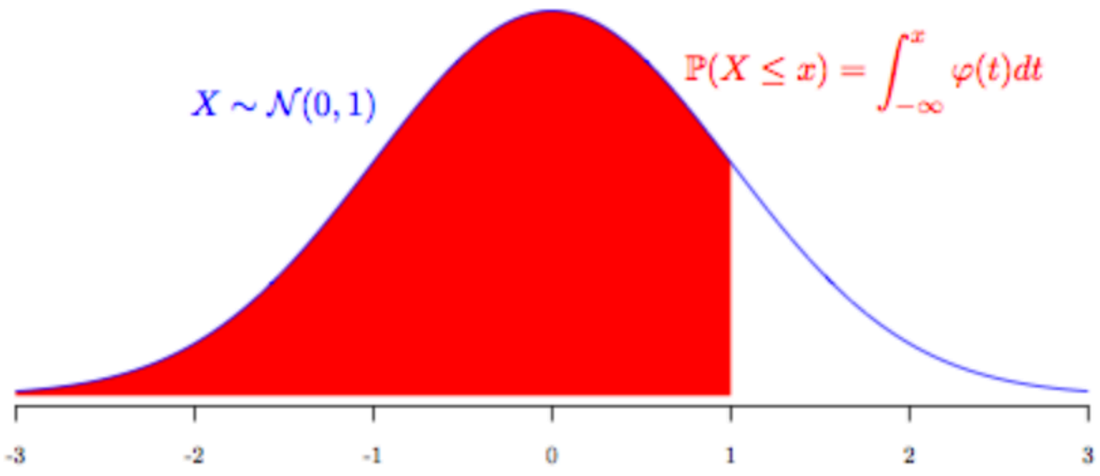
层级聚类

```
class sklearn.cluster.AgglomerativeClustering(
    n_clusters=2, # 目标类别数，默认是2
    affinity='euclidean', # 样本点之间距离计算方式
                          # euclidean(欧式距离)
                          # manhattan(曼哈顿距离)
                          # cosine(余弦距离)
                          # 如果参数linkage选择“ward”的时候只能使用euclidean
    memory=None, # 用于缓存输出的结果，默认为不缓存
    connectivity=None,
    compute_full_tree='auto', # 通常当训练了n_clusters后，训练过程就会停止，
                              # 但是如果compute_full_tree=True，则会继续训练从而生成一颗完整的树
    linkage='ward' # 链接标准，即样本点的合并标准，
                   # ward(默认)、complete、average、single
    )
```

高斯混合模型

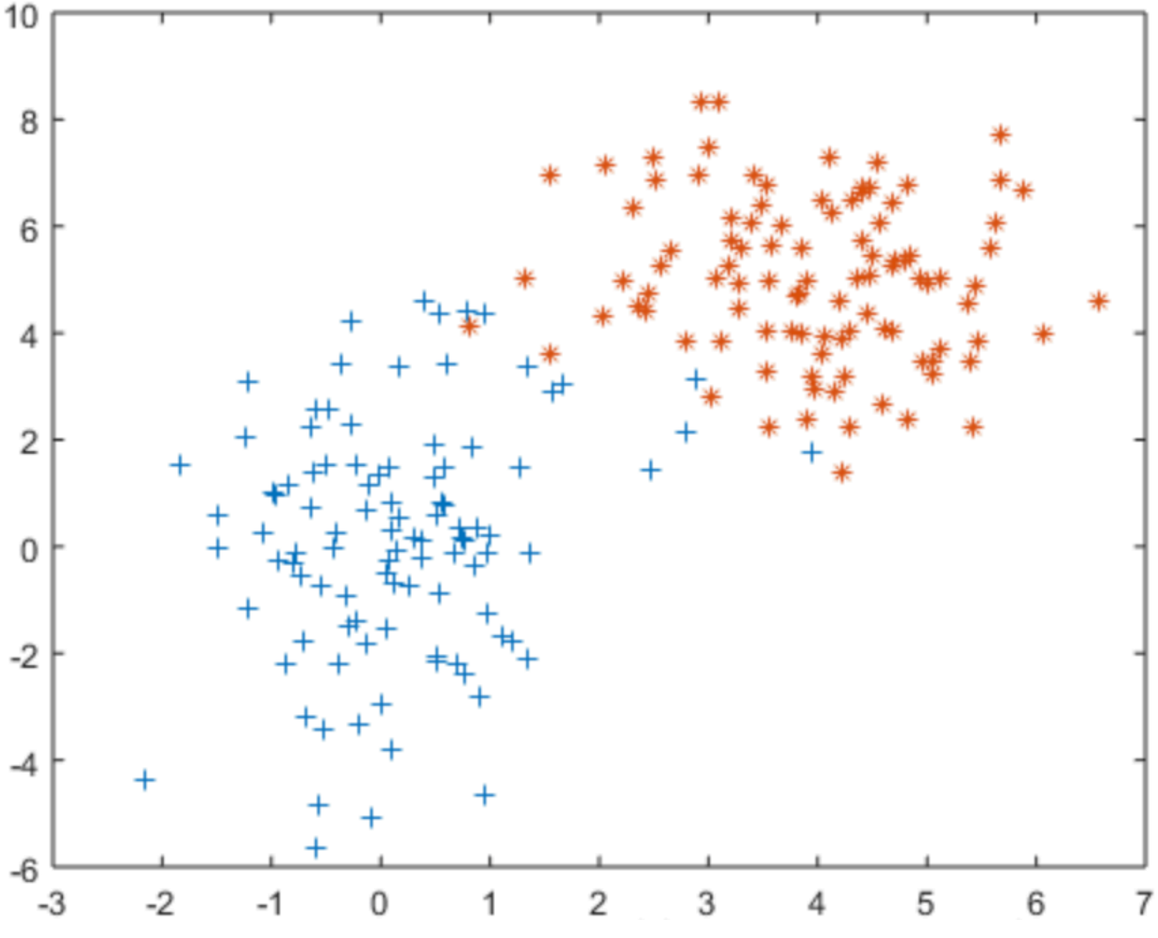
单变量高斯分布

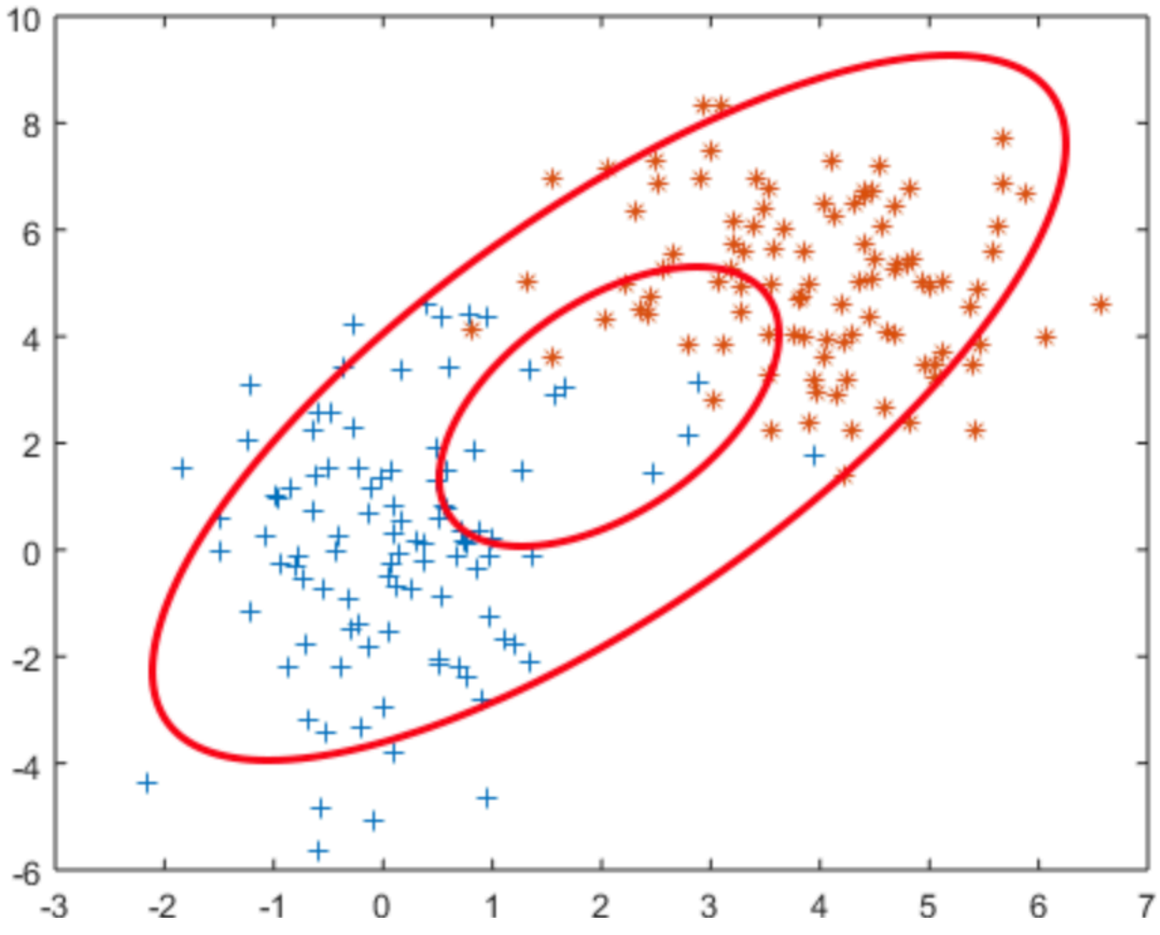
- 高斯分布又称为正态分布，有着广泛的应用
- 概率密度为 $f(x) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(x-\mu)^2}{2\sigma^2})$



高斯混合模型（Gaussian mixture model, GMM）

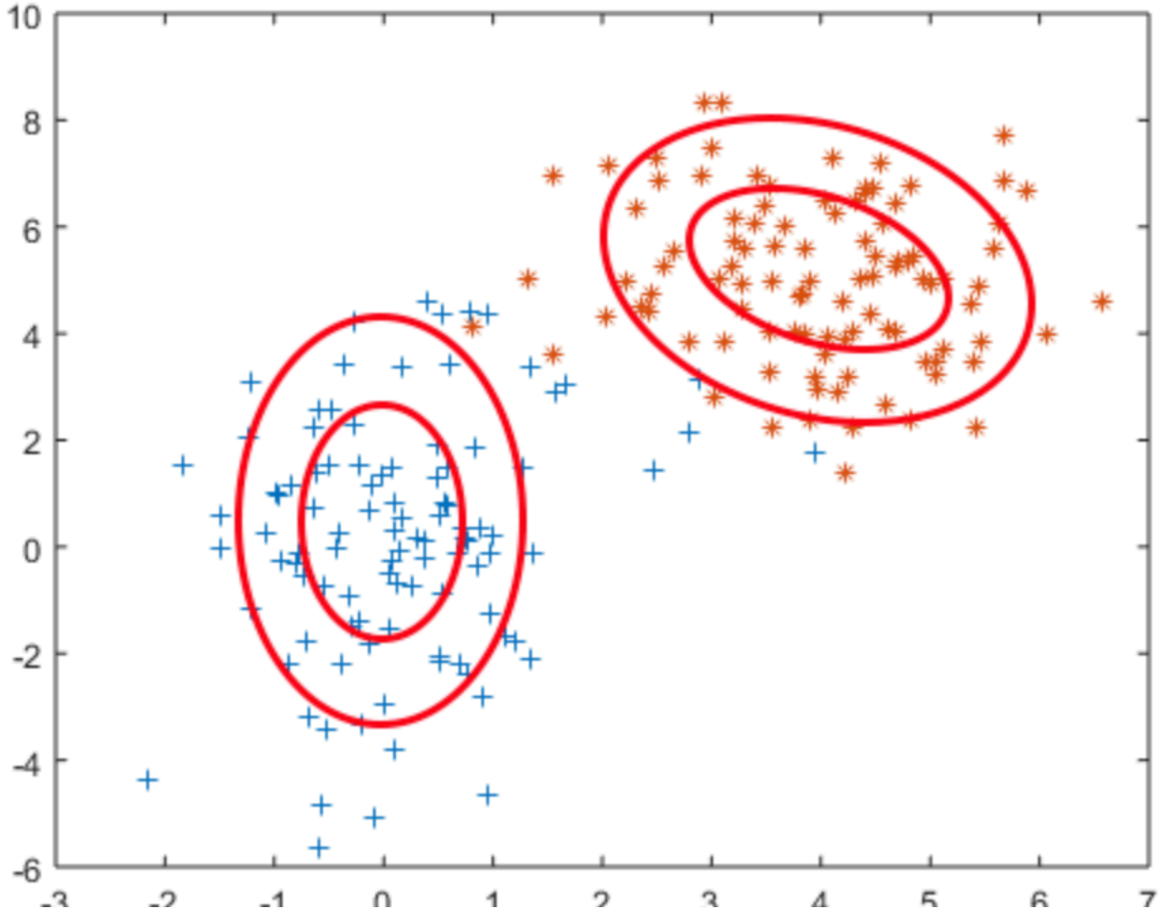
- 一般来讲越靠近椭圆的中心样本出现的概率越大，这是由概率密度函数决定的，但是这个高斯分布的椭圆中心的样本量却极少。显然样本服从单高斯分布的假设并不合理。单高斯模型无法产生这样的样本。





高斯混合模型（Gaussian mixture model, GMM）

- 但实际上它是由两个高斯分布同时产生的，对于这种问题，我们引入高丝混合模型



高斯混合模型（Gaussian mixture model, GMM）

- 高斯混合模型概率密度定义为

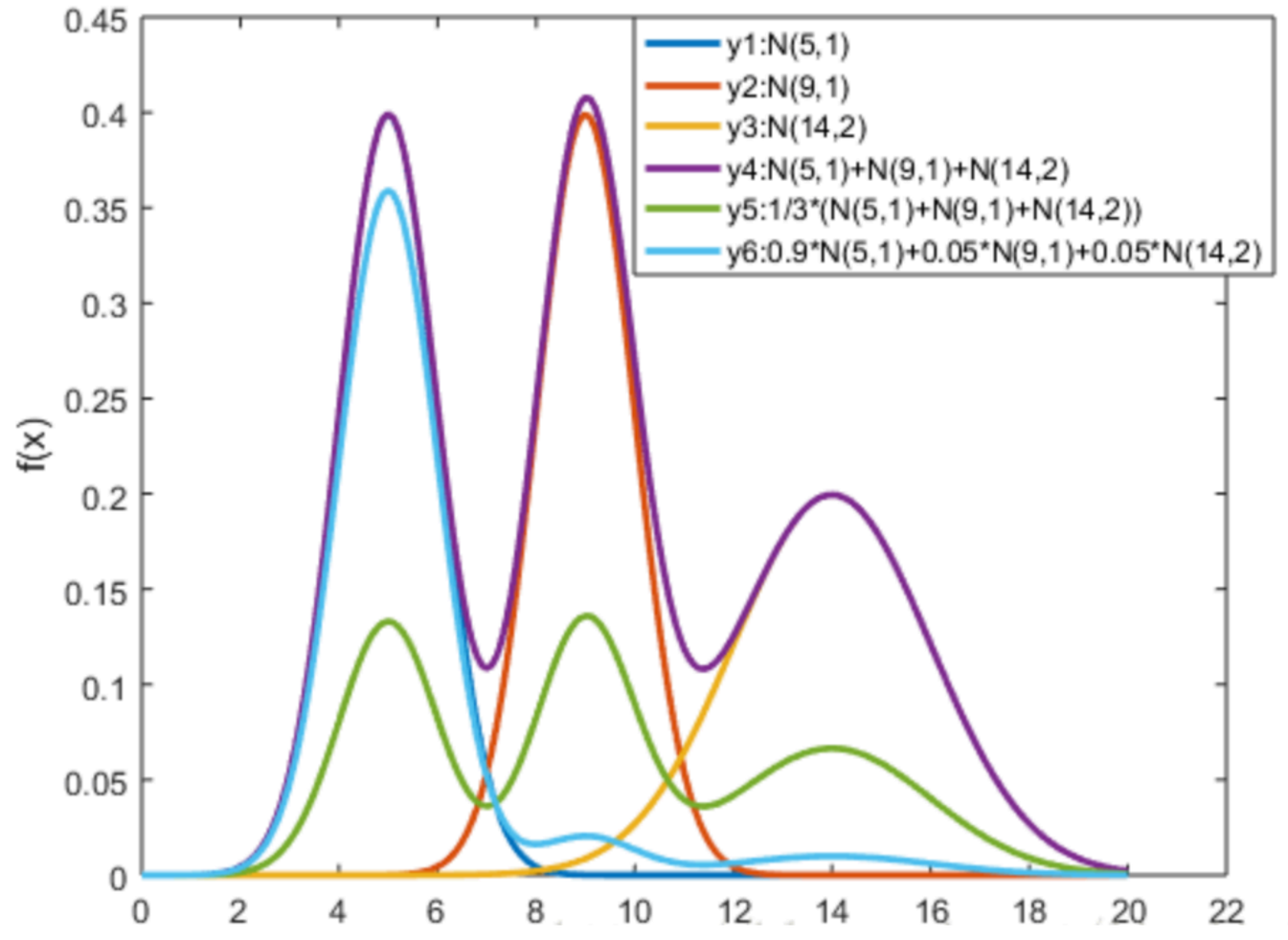
$$p(x) = \sum_{k=1}^{K} p(k)p(x \mid k) = \sum_{k=1}^{K} \pi_k N(x \mid \mu_k, \sigma_k^2)$$

- 其中 $p(x \mid k) = N(x \mid \mu_k, \sigma_k^2)$ 是第k个高斯分布的概率密度

- p(k)为第k个模型的权重 $p(k) = \pi_k, \ \sum_{k=1}^{K} \pi_k = 1$

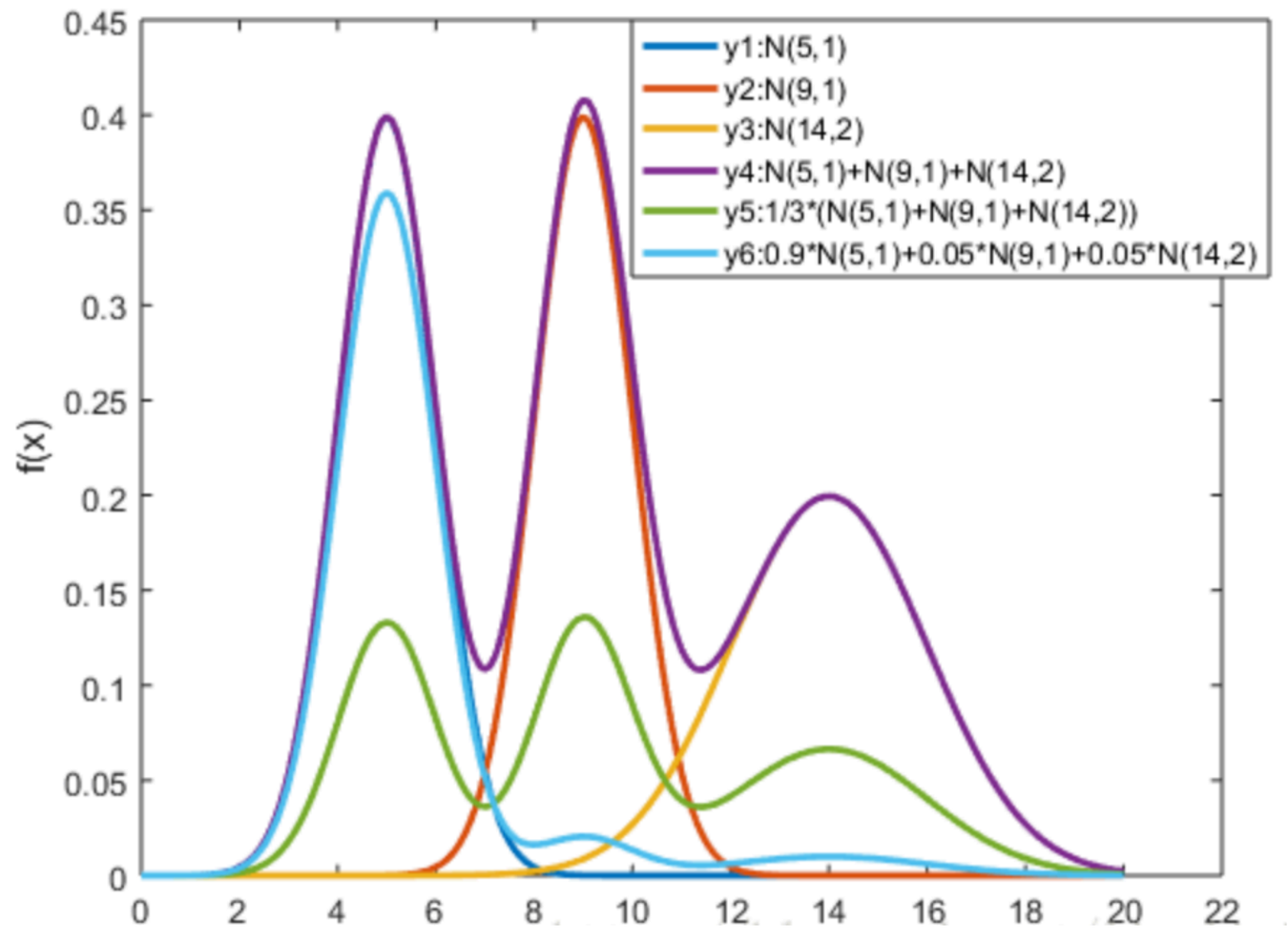
高斯混合模型（Gaussian mixture model, GMM）

- y1,y2和y3分别表示三个一维高斯模型



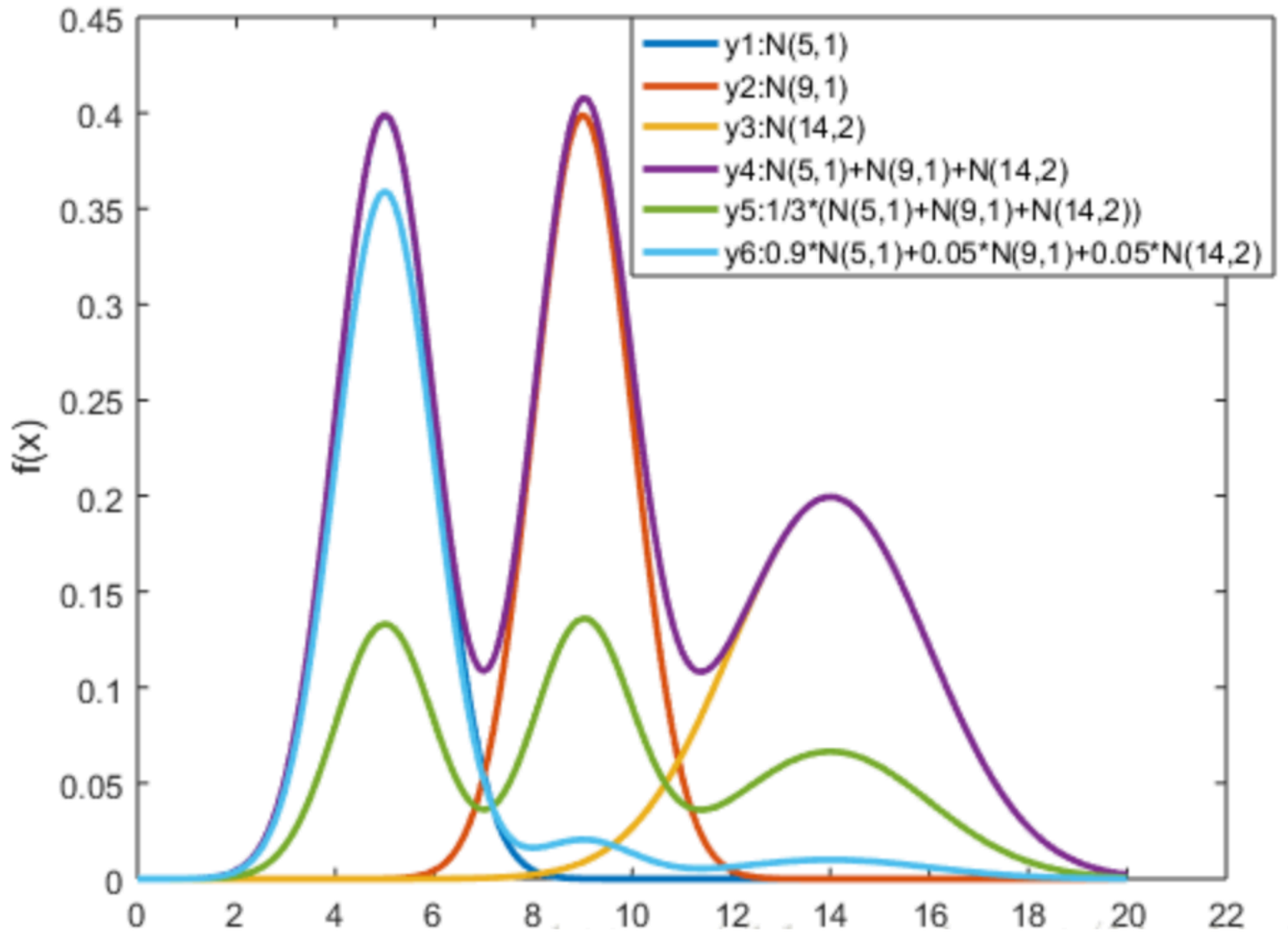
高斯混合模型（Gaussian mixture model, GMM）

- y4表示将三个模型的概率密度函数直接相加，注意的是这并不是一个混合高斯模型。因为不满足 $p(k) = \pi_k, \sum_{k=1}^{K} \pi_k = 1$



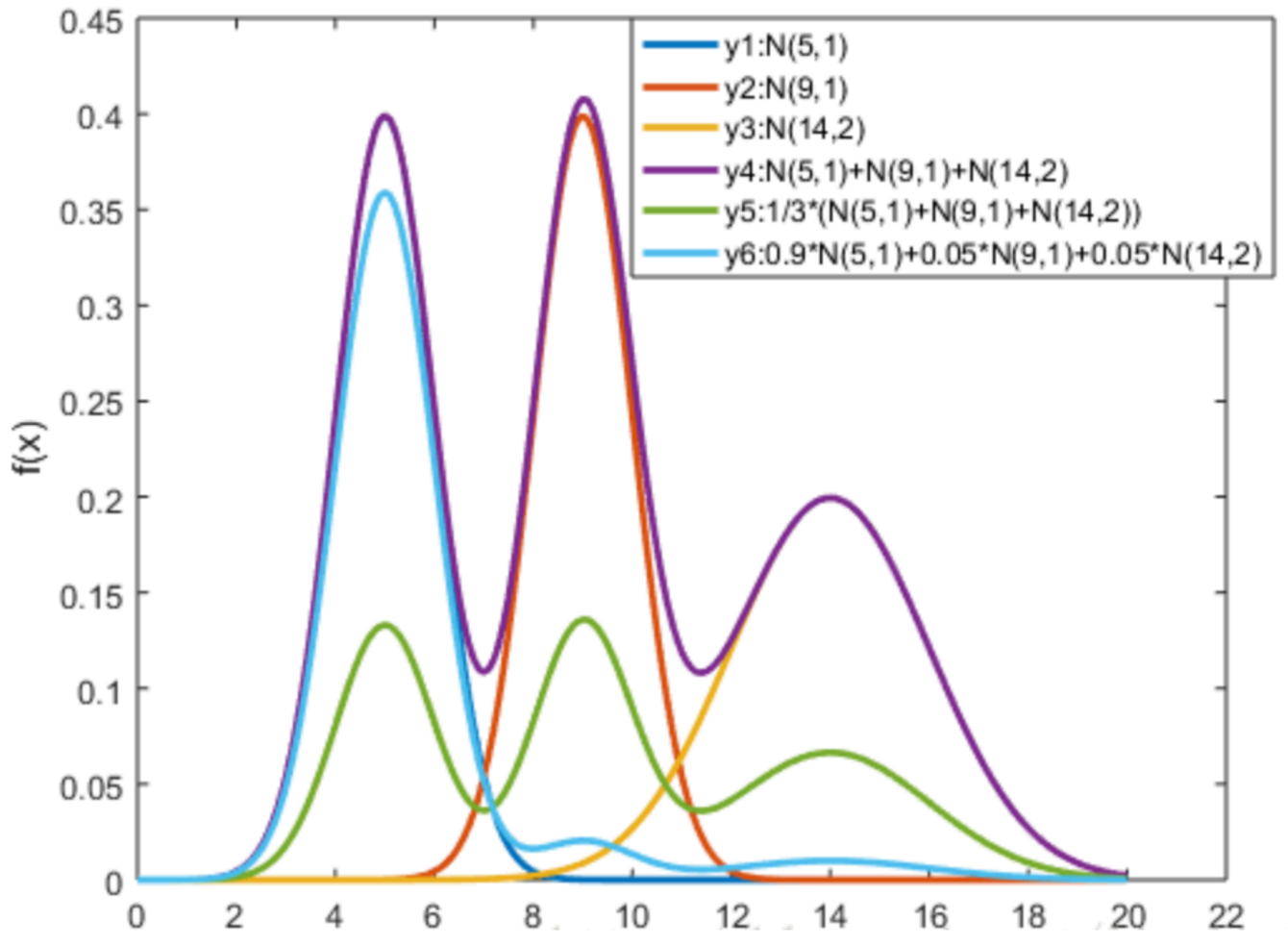
高斯混合模型（Gaussian mixture model, GMM）

- y5和y6分别是由三个相同的高斯模型融合生成的不同混合模型。



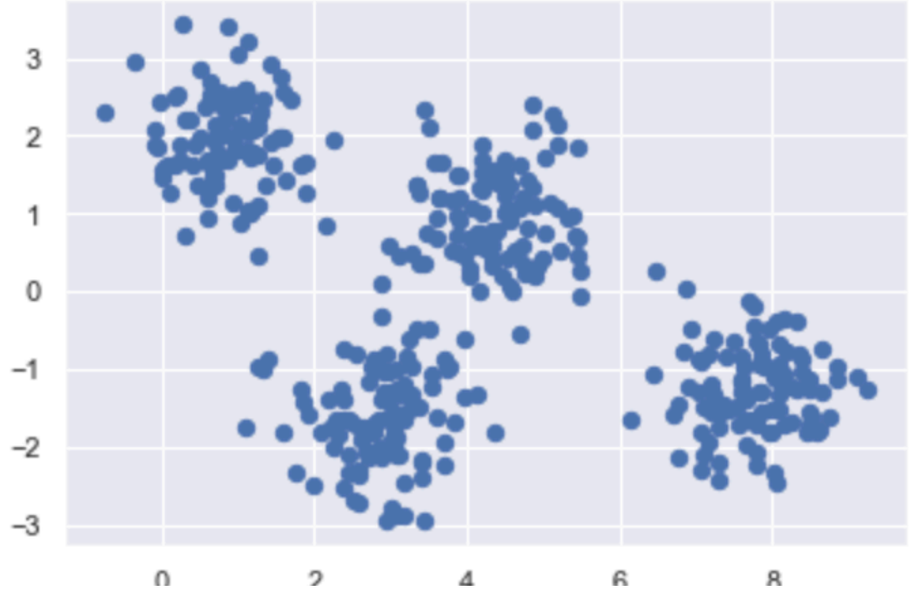
高斯混合模型（Gaussian mixture model, GMM）

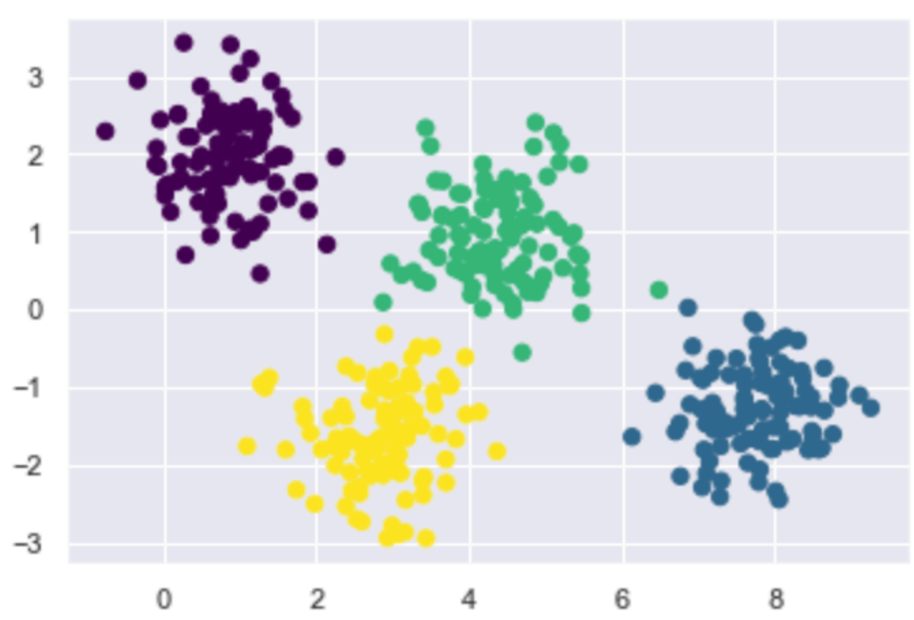
- 由此可见，调整权重将极大影响混合模型的概率密度函数曲线。另一方面也可以直观地理解混合高斯模型可以更好地拟合样本的原因：它有更复杂更多变的概率密度函数曲线。理论上，混合高斯模型的概率密度函数曲线可以是任意形状的非线性函数。



高斯混合模型（Gaussian mixture model, GMM）

- 当指定数据中高斯分布个数后，可以计算每个点属于某一个簇（高斯分布）的概率，将其 归属到概率最大的簇中





高斯混合模型（Gaussian mixture model, GMM）

- 使用EM算法来对高斯混合模型求解，估算高斯混合模型的参数

1) 初始化模型各个参数

2) 使用当前参数，计算样本属于某一个簇的概率

3) 根据2）的结果更新参数

4) 反复执行2）3）两步知道模型收敛

高斯混合模型（Gaussian mixture model, GMM）

- 高斯混合模型与k-means：
  - 高斯混合模型给出属于某一个簇的概率
  - k-means直接给出属于哪一类

高斯混合模型（Gaussian mixture model, GMM）

- 高斯混合模型优点：
  - 相对于k-mean上更具有一般性
  - 每个簇更具有描述性（均值与方差）
- 高丝混合模型缺点：
  - 收敛慢
  - 数据量少时效果不好