人工智能
——划分式与基于密度的聚类

Yanghui Rao
Assistant Prof., Ph.D
School of Data and Computer Science,
Sun Yat-sen University
raoyangh@mail.sysu.edu.cn

聚类

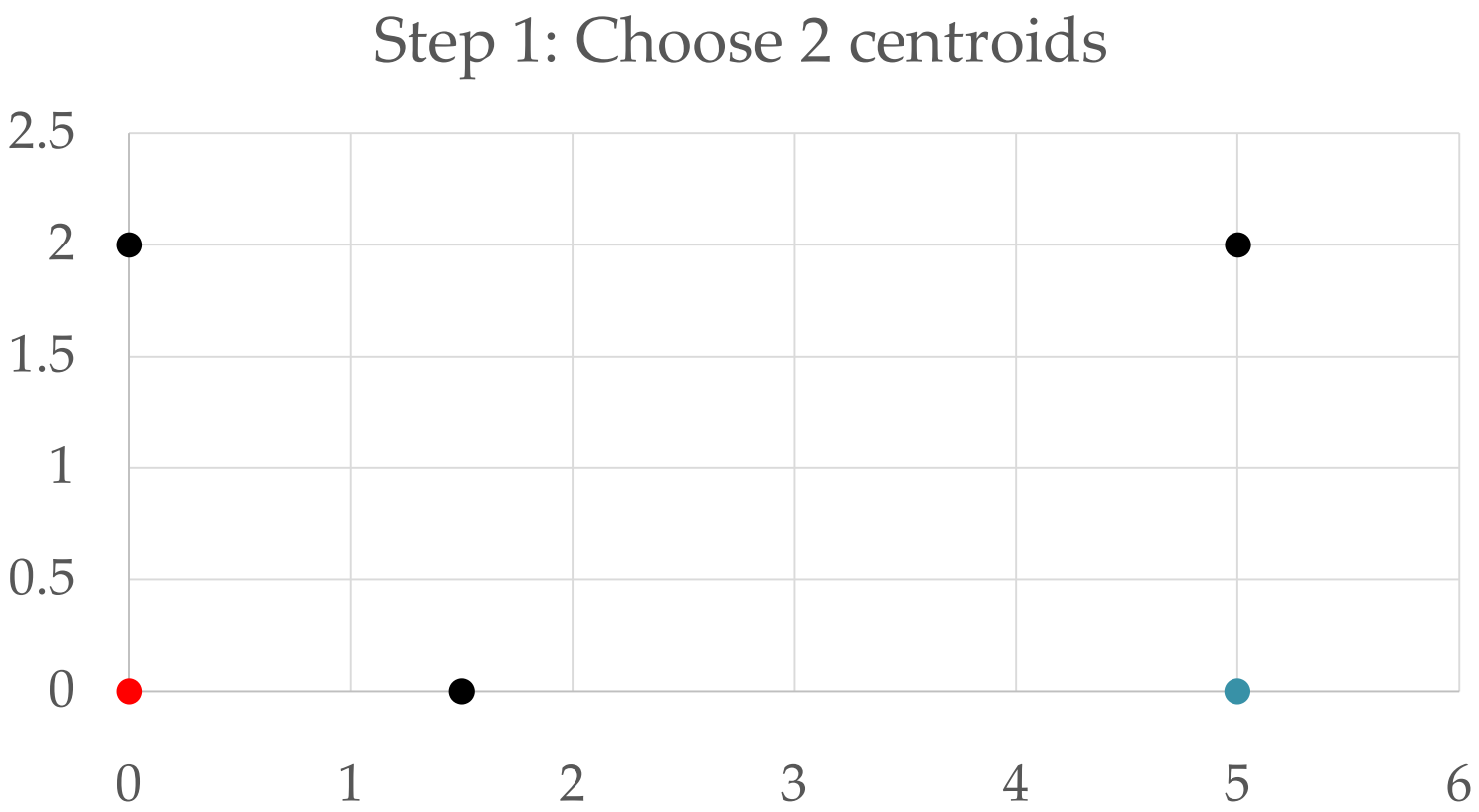
- 目标：基于相似性度量，将文本（也可以是词）集合聚类到簇(cluster)中，使得簇内样例的相似度大于簇之间样例的相似度.
- 假定：给定集合的“合适”簇群，如果用户对某篇文本（或者关键词）感兴趣，则该用户可能会对那篇文本（或者关键词）所属的簇群的其他样例感兴趣.
- 相似度指标：
  - 用矢量表示文本
    - 文本向量之间的距离
    - 文本向量之间的夹角余弦

划分式聚类

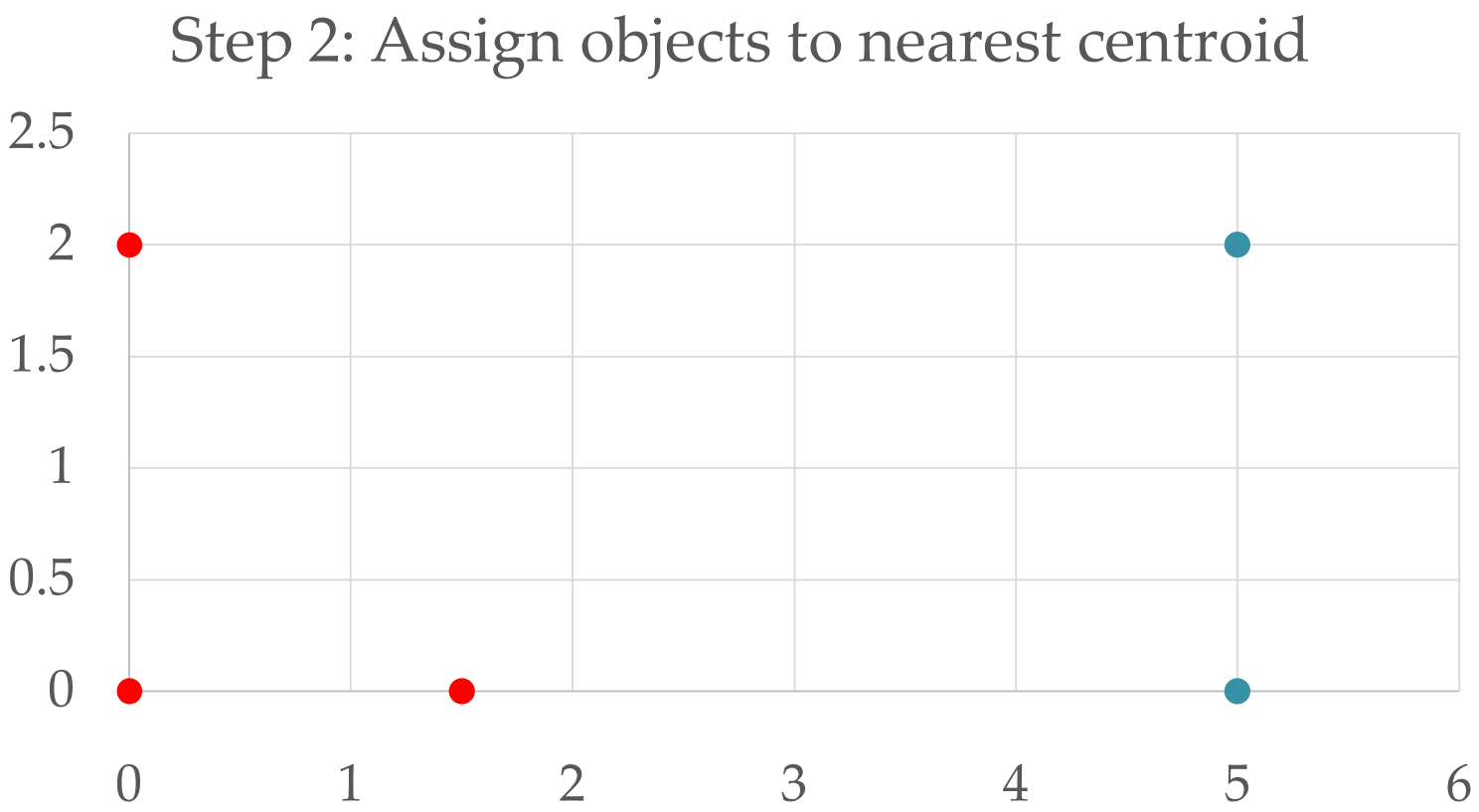
- *k*-Means: 重复如下步骤…
  - 选择任意 *k* 个质心(centroids)
  - 将每个文档分配到最近的质心
  - 重新计算质心

- *k*-Means (划分法) 示例：
  - x1 = (0, 2), x2 = (0, 0), x3 = (1.5, 0), x4 = (5, 0), x5 = (5, 2)
  - $k = 2$

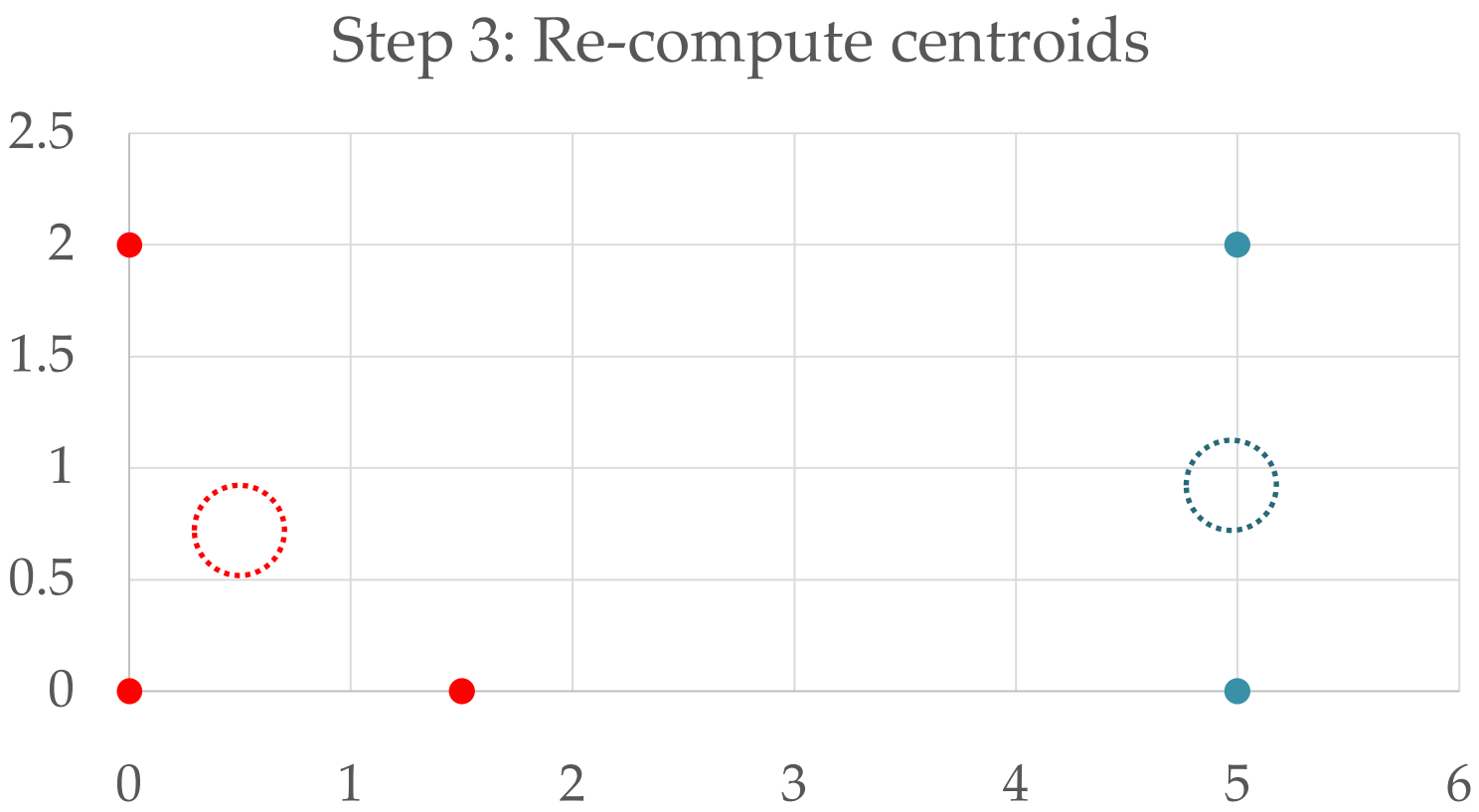
k-Means



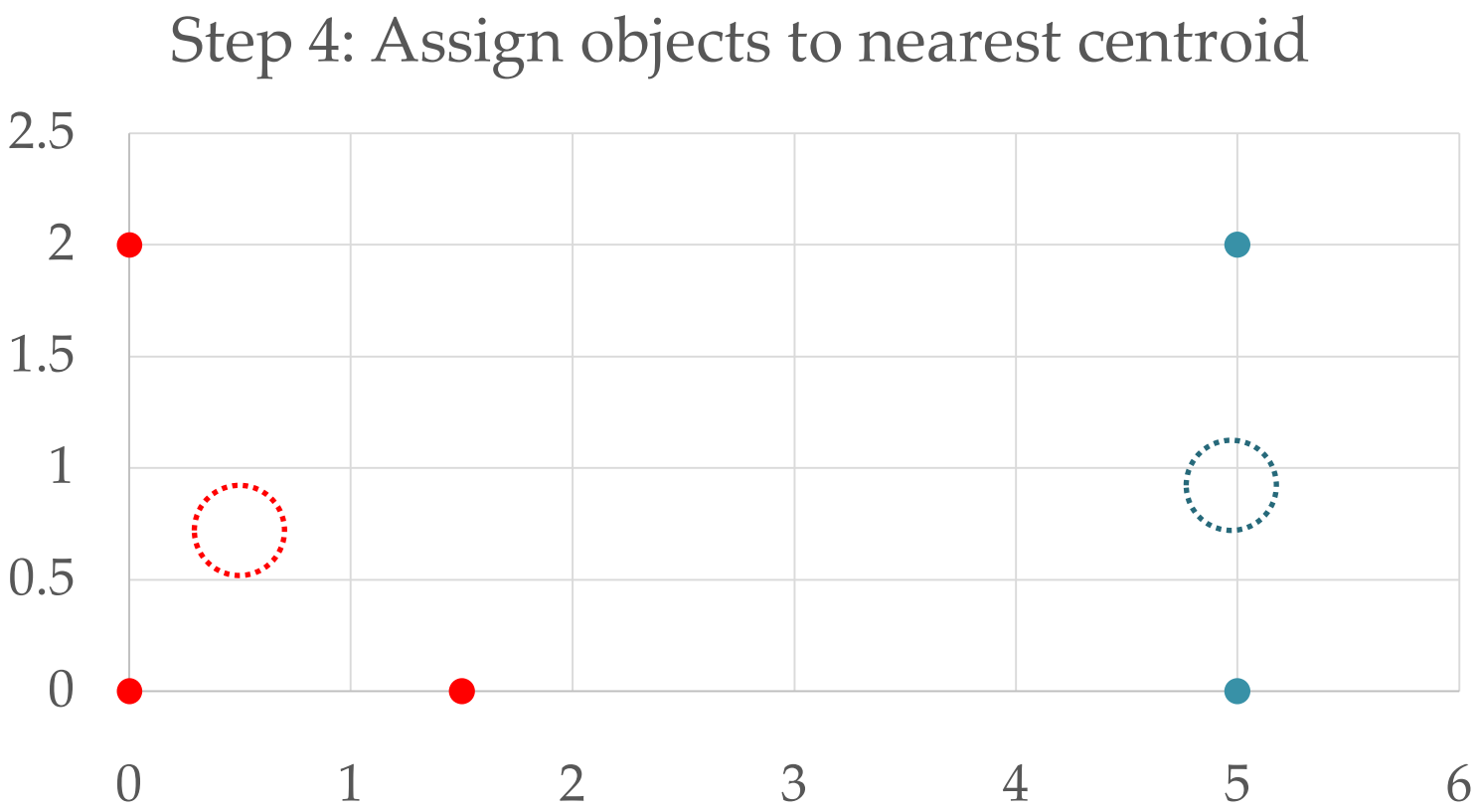
k-Means


Step 2: Assign objects to nearest centroid
2.5
2
1.5
1
0.5
0
0
1
2
3
4
5
6

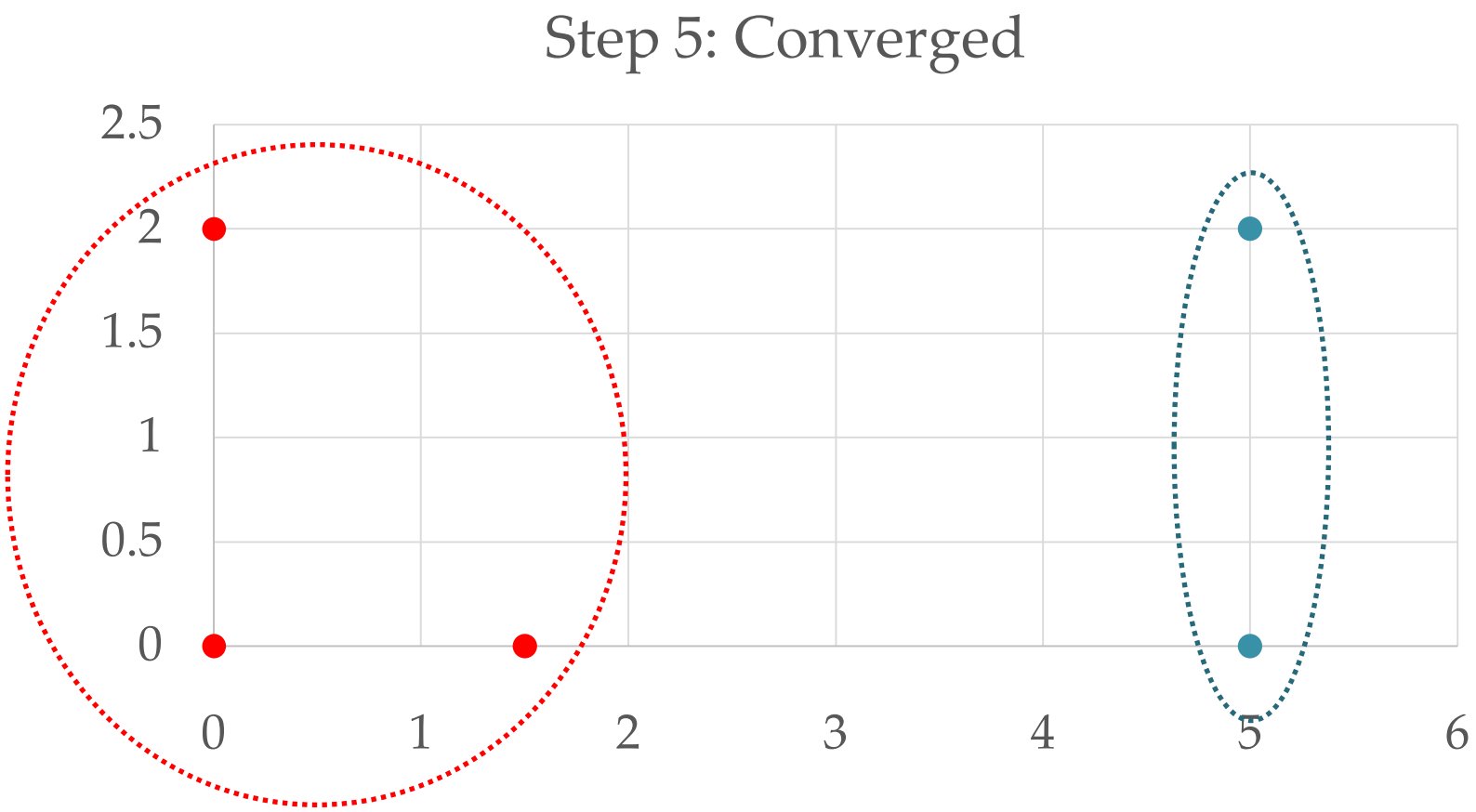
k-Means



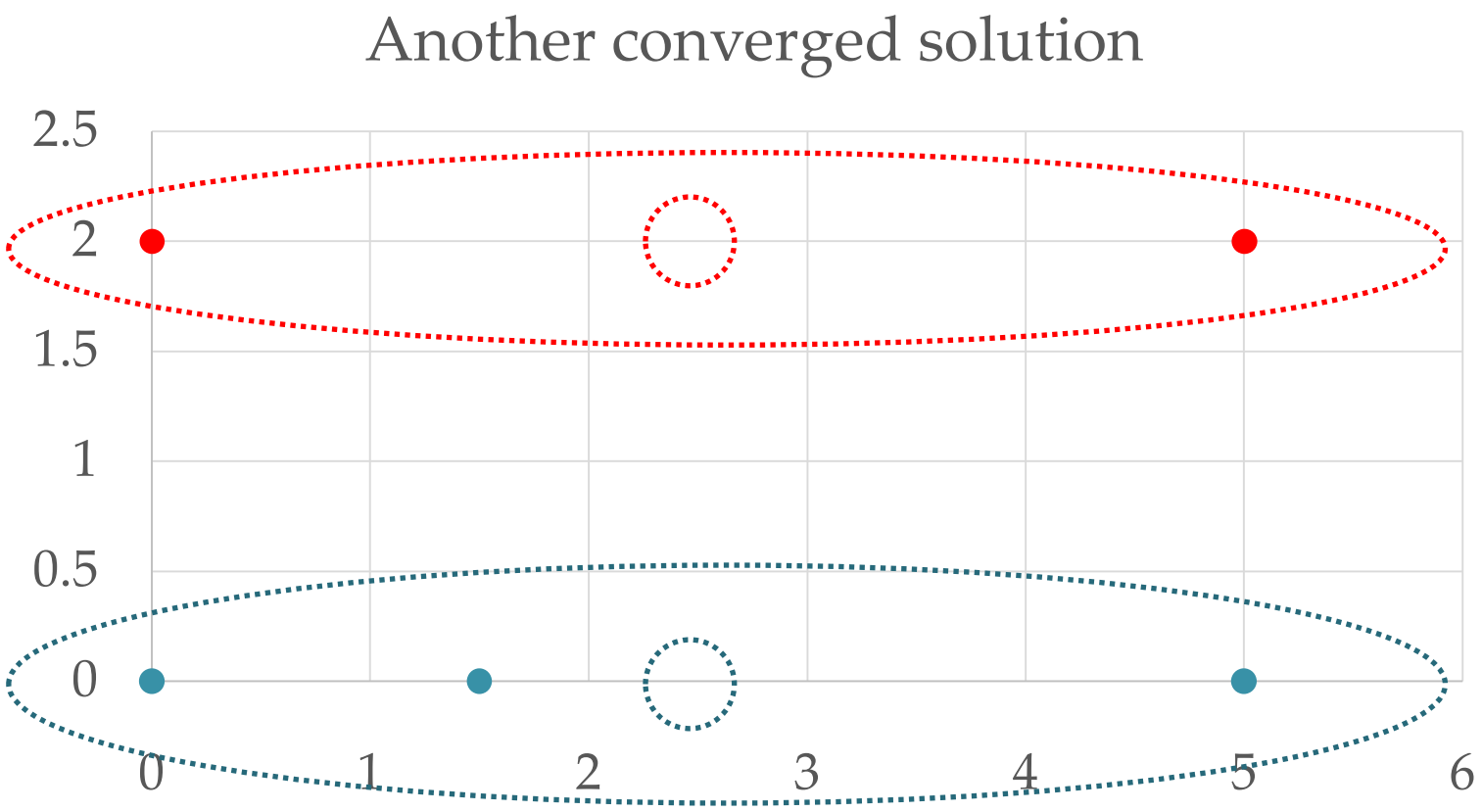
k-Means


Step 4: Assign objects to nearest centroid
2.5
2
1.5
1
0.5
0
0
1
2
3
4
5
6

k-Means



k-Means



k-Means: Choosing *k*

- 主要是“问题驱动”（problem driven）
- 也可以是“数据驱动”（data driven），但条件如下：
  - 数据不稀疏
  - 输入的属性没有太多噪音

基于密度的聚类

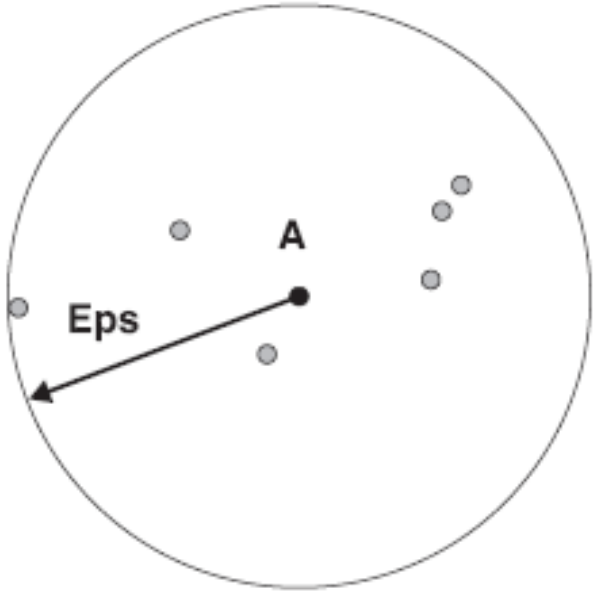
- 基于密度的聚类旨在检测高密度的区域，这些区域由低密度区域相互分开.
- DBSCAN（Density-Based Spatial Clustering of Applications with Noise）是一种简单有效的基于密度的聚类算法.

DBSCAN

- 对于DBSCAN，我们需要估计数据集中特定点的密度（density）
- 这是通过计算在该点的指定半径Eps内的点数来执行的.
- 计算时需要包括当前这个点.

DBSCAN

- 下图说明了这种技术。
- A点的Eps或半径内的点数是7，包括A点本身。



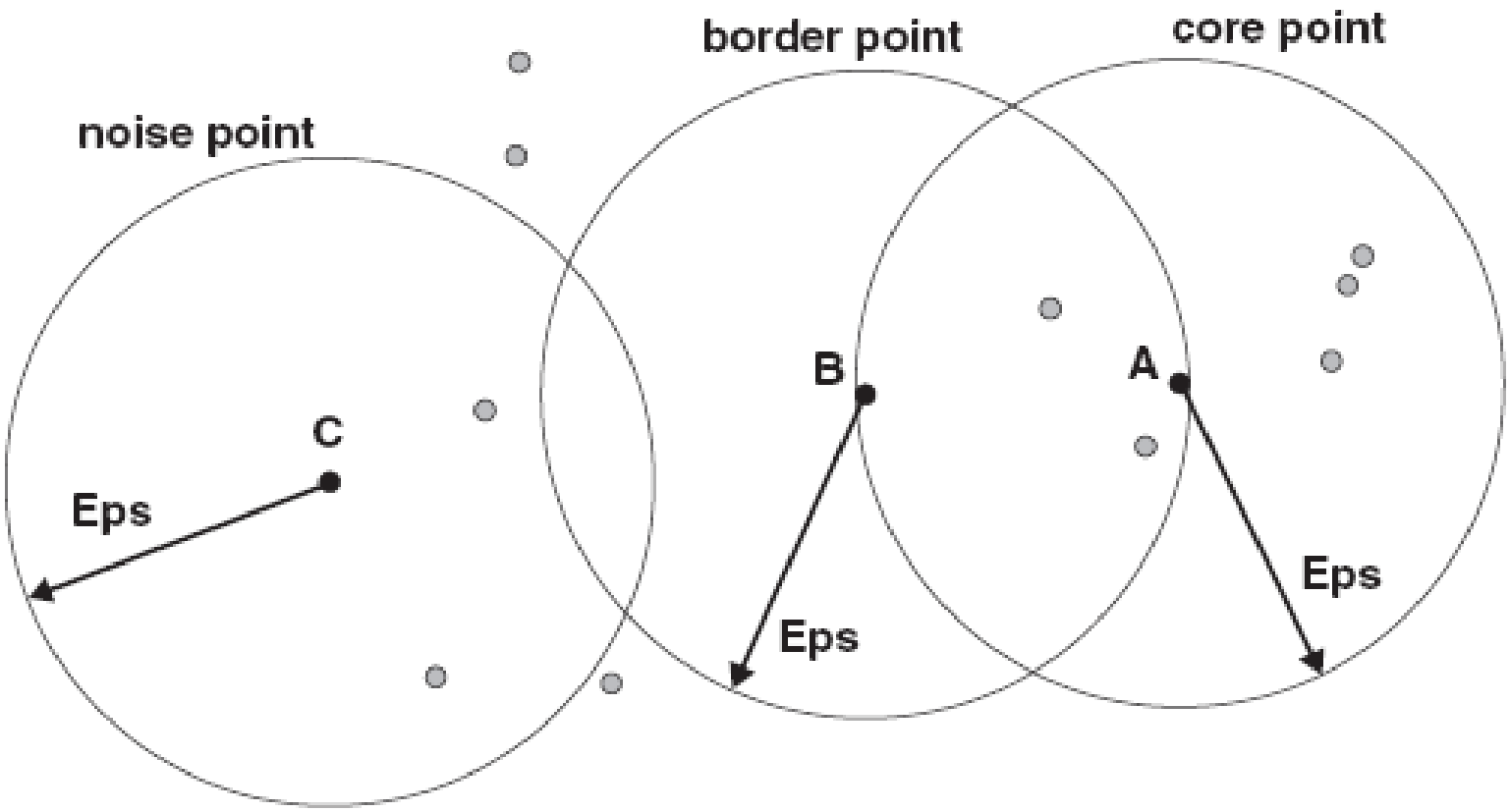
DBSCAN

- 任何点的密度将取决于指定的半径.
- 假设数据集中的点数是m.
- 如果半径足够大，那么所有点的密度都是m.
- 如果半径太小，那么所有点的密度都是1.

DBSCAN

- 我们需要把一个点归类为
  - 在一个密集区域的内部(a core point, 核心点).
  - 在密集区域的边缘(a border point, 边界点).
  - 在一个稀疏的地区(a noise or background point, 噪音点).
- 核心，边界和噪声点的概念说明如下.

DBSCAN



DBSCAN

- 核心点位于密集区域的内部.
- 如果点的给定邻域内或边界上的点的数量大于或等于某个阈值MinPts ，则该点是核心点.
- 邻域的大小由距离函数和用户指定的半径参数Eps确定.
- 阈值MinPts也是用户指定的参数.
- 在上图中，如果MinPts = 7，则A是指定半径（Eps）的核心点.

DBSCAN

- 边界点不是核心点，而是落在核心点附近或边界附近.
- 在上图中，B是边界点.
- 边界点可以落在几个核心点的邻域内.

- 噪声点是指既不是核心点又不是边界点的任何一点.
- 在上图中，C是一个噪声点.

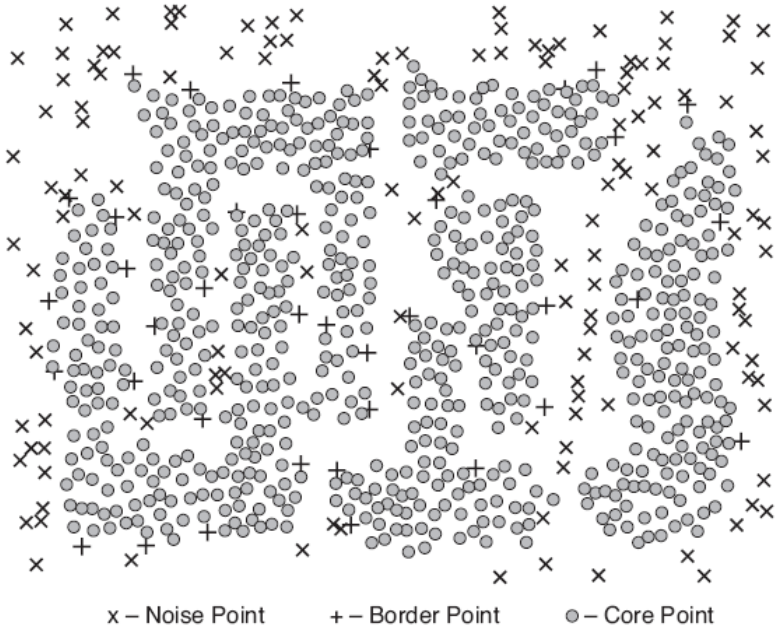
DBSCAN

- DBSCAN可以概括如下:
  - 如果所有的点已经处理，停止
  - 对于以前没有处理的特定点，检查它是否是核心点
  - 如果不是核心点
    - 将其标记为噪音点（此标签可能稍后会更改）
  - 如果是核心点，将其标记并
    - 使用这一点形成一个新的聚类C_{new}，并包括集群内的Eps-邻域内或边界上的所有点
    - 将所有这些相邻点插入队列中
    - 当队列不为空时
      - 从队列中删除第一个点
      - 如果这一点不是一个核心点，将其标记为边界点
      - 如果这个点是一个核心点，则标记它并检查其邻居中以前没有分配给类的每个点。对于每一个未分配的相邻点
        - 将该点分配给当前类C_{new}
        - 将该点插入队列

DBSCAN

- 下一张幻灯片的左图显示了具有3000个二维点的示例数据集.
- 右图显示了由DBSCAN找到的结果簇群.
- 核心点，边界点和噪声点也显示出来.

DBSCAN



DBSCAN

- DBSCAN相对来说更抗噪声，可以处理任意形状和大小的簇集.
- 因此，它可以找到许多无法用k-Means等算法获得的聚类结果.

参考资料

- S.J. Rizvi and J.R. Haritsa. Maintaining data privacy in association rule mining. *Proceedings of the 28th VLDB Conference*, 34(6):682-693, 2002.
- A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(2):264-323, 1999.
- A. Rodriguez and A. Laio. Clustering by fast search and find of density peaks. *Science*, 344(6191):1492-1496, 2014.