The "Memory-Load" of Points

华中科技大学 计算机科学与技术学院 School of Computer Science &Technology, HUST

- **Processing the "Memory-Load" of points (1):**
- **1)**Find those points that are "**sufficiently close**" to a cluster centroid and add those points to that cluster and the **DS**
  - These points are so close to the centroid that they can be summarized and then discarded
- **2)**Use any main-memory clustering algorithm to cluster the remaining points and the old **RS**
  - Clusters go to the **CS**; outlying points to the **RS**

Discard set (DS, 废弃集): Close enough to a centroid to be summarized.
Compression set (CS, 压缩集): Summarized, but not assigned to a cluster
Retained set (RS, 留存集): Isolated points

The "Memory-Load" of Points

- **Processing the "Memory-Load" of points (2):**
- **3) DS set:** Adjust statistics of the clusters to account for the new points
  - Add ***N***s, ***SUM***s, ***SUMSQ***s
- **4)** Consider merging compressed sets in the **CS**
- **5)** If this is the last round, merge all compressed sets in the **CS** and all **RS** points into their nearest cluster

Discard set (DS, 废弃集): Close enough to a centroid to be summarized.
Compression set (CS, 压缩集): Summarized, but not assigned to a cluster
Retained set (RS, 留存集): Isolated points

Example: N-SUM-SUMSQ

- Example for update N-SUM-SUMSQ in DS set when adding point (6,0):
  - ***N***=3
  - ***SUM***=[18,-1]
  - ***SUMSQ***=[110,5]

x (5,1)

x (6,0)   x (7,0)

x (6,-2)

- Ans: 新增点加入DS废弃集后统计信息更新为
  - ***N***=4
  - ***SUM***=[24,-1]
  - ***SUMSQ***=[146,5]

Discard set (DS, 废弃集): Close enough to a centroid to be summarized.
Compression set (CS, 压缩集): Summarized, but not assigned to a cluster
Retained set (RS, 留存集): Isolated points

Details in BFR

- **Q1)** How do we decide if a point is "**close enough**" to a cluster that we will add the point to that cluster?

- **Q2)** How do we decide whether two compressed sets (**CS**) deserve to be combined into one?

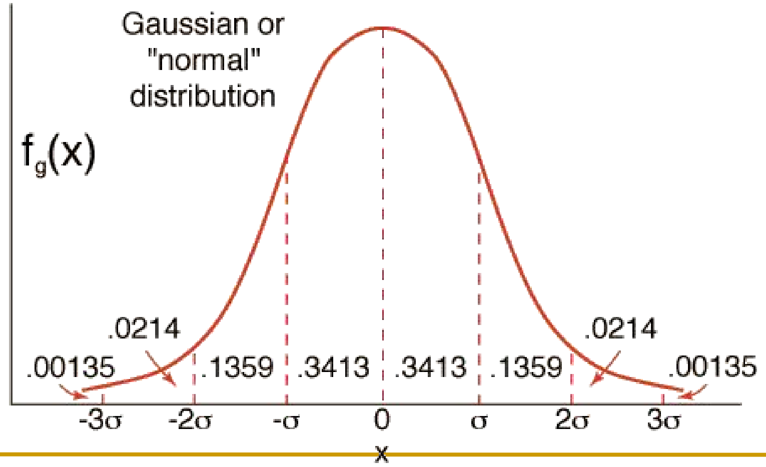
Discard set (DS, 废弃集): Close enough to a centroid to be summarized.
Compression set (CS, 压缩集): Summarized, but not assigned to a cluster
Retained set (RS, 留存集): Isolated points

How Close is Close Enough?

- **Q1) We need a way to decide whether to put a new point into a cluster (and discard)**

- **BFR suggests:**
  - **High likelihood of the point belonging to currently nearest centroid**
  - The **Mahalanobis distance (马氏距离)** is less than a threshold

fact: each cluster normally distributed



Mahalanobis Distance

- **Mahalanobis Distance(马氏距离),** is normalized Euclidean distance from centroid(欧氏距离的修正)
- For point ***(x_1, …, x_d)*** and centroid ***(c_1, …, c_d)***
  - 1、Normalize in each dimension (每个维度的点减去对应簇质心的点，然后除以标准差): $y_i = (x_i - c_i) / \sigma_i$
  - 2、Take sum of the squares of the y_i (每个维度对应求平方和)
  - 3、Take the square root (每个维度再开平方根)

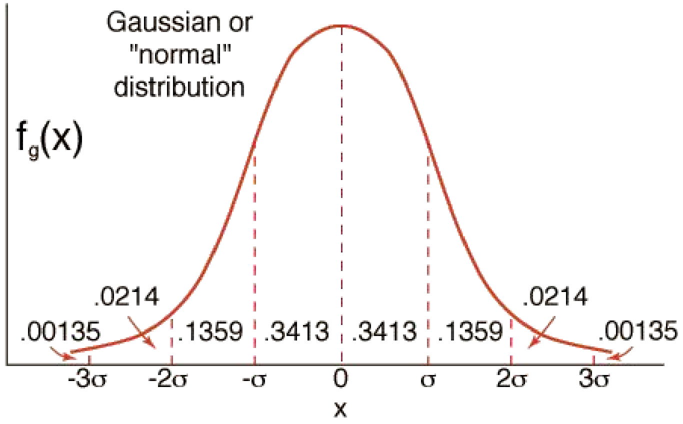
$$d(x,c) = \sqrt{\sum_{i=1}^{d} \left(\frac{x_i - c_i}{\sigma_i}\right)^2}$$

σ_i … standard deviation (标准差) of points in the cluster in the i^{th} dimension

Mahalanobis Distance

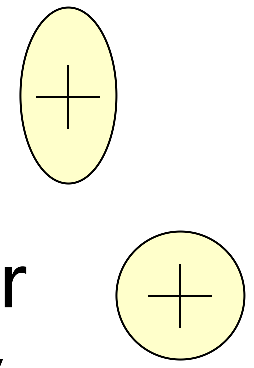
- If clusters are normally distributed in $\boldsymbol{d}$ dimensions, then after transformation, one standard deviation = $\sqrt{\boldsymbol{d}}$
  - i.e., 68% of the points of the cluster will have a Mahalanobis distance $< \sqrt{\boldsymbol{d}}$

- Accept a point for a cluster if its mahalanobis distance is < some threshold, e.g. **2** standard deviations



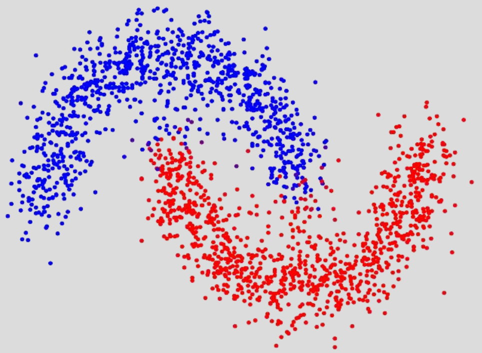
Should 2 CS clusters be combined?

- **Q2) Should 2 CS subclusters be combined?**
- **Approach 1:**
- Compute the variance (方差) of the combined subcluster
  - ***N***, ***SUM***, and ***SUMSQ*** allow us to make that calculation quickly
- Combine if the combined variance is below some threshold



- **Approach 2:** Treat dimensions differently, consider density

Discard set (DS, 废弃集): Close enough to a centroid to be summarized.
Compression set (CS, 压缩集): Summarized, but not assigned to a cluster
Retained set (RS, 留存集): Isolated points

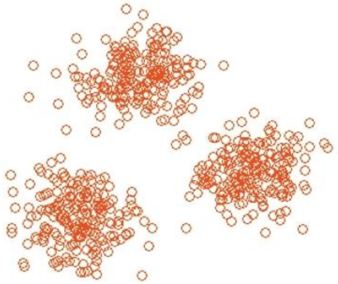


Section 5.5: CURE Algorithm

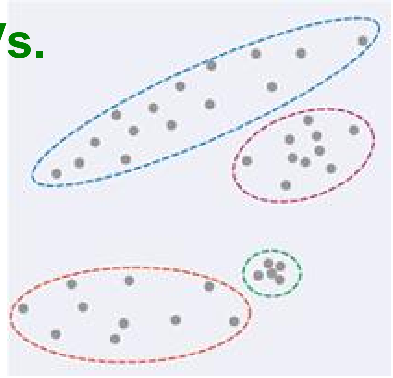
Extension of *k*-means to clusters of arbitrary shapes

The CURE Algorithm

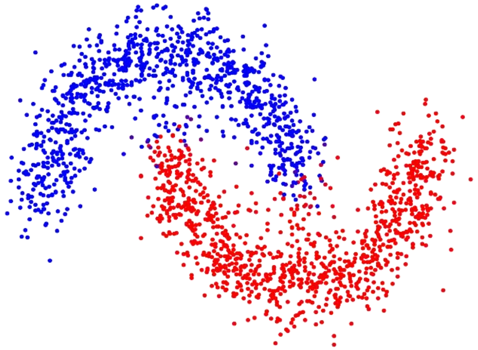
- **Problem with BFR/*k*-means:**
  - Assumes clusters are normally distributed in each dimension
  - And axes are fixed – ellipses at an angle are ***not OK***



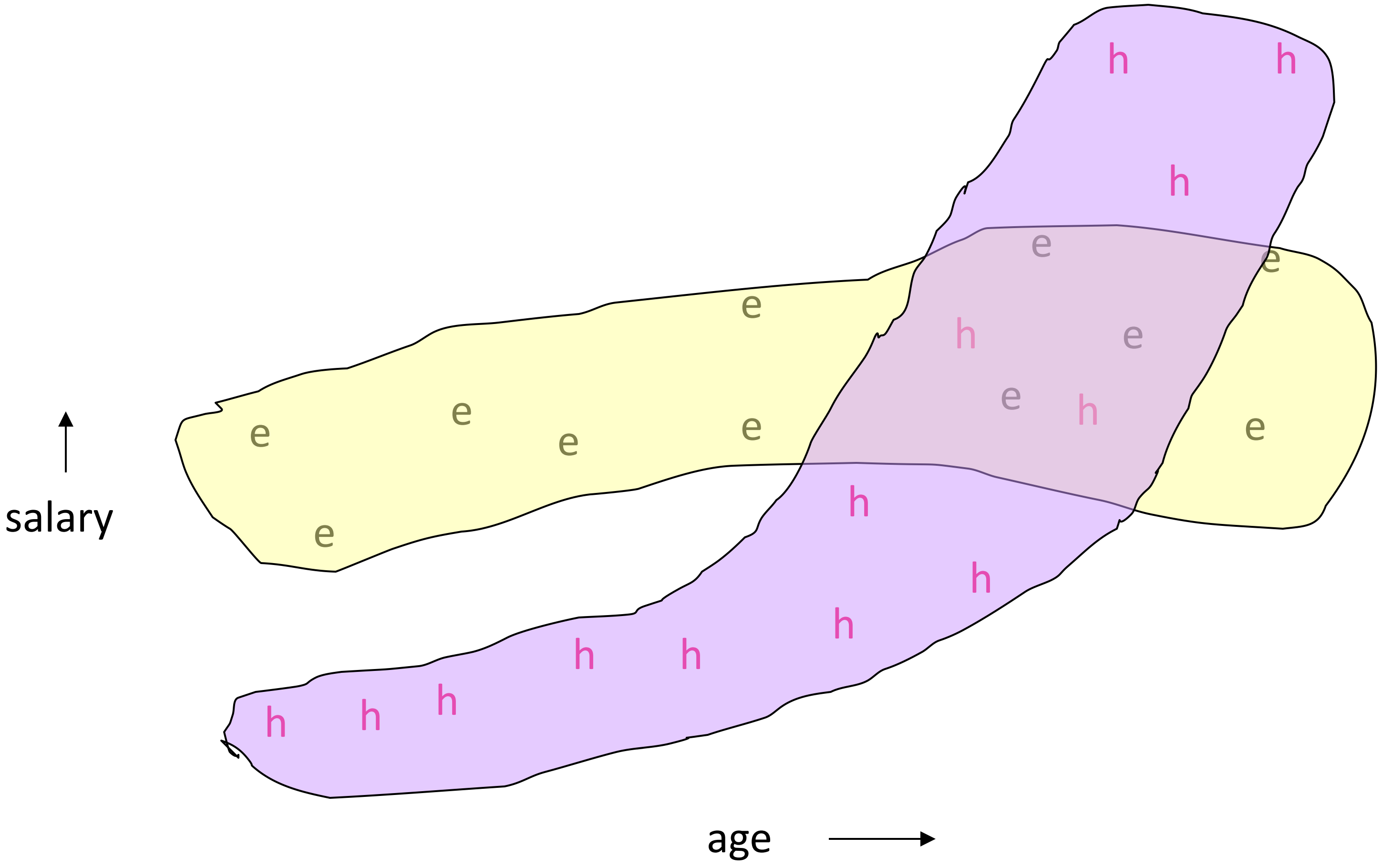
Vs.



- **CURE (Clustering Using REpresentatives):**
  - Assumes a Euclidean distance
  - Allows clusters to assume any shape
  - **Uses a collection of representative points to represent clusters**



Example: Stanford Salaries

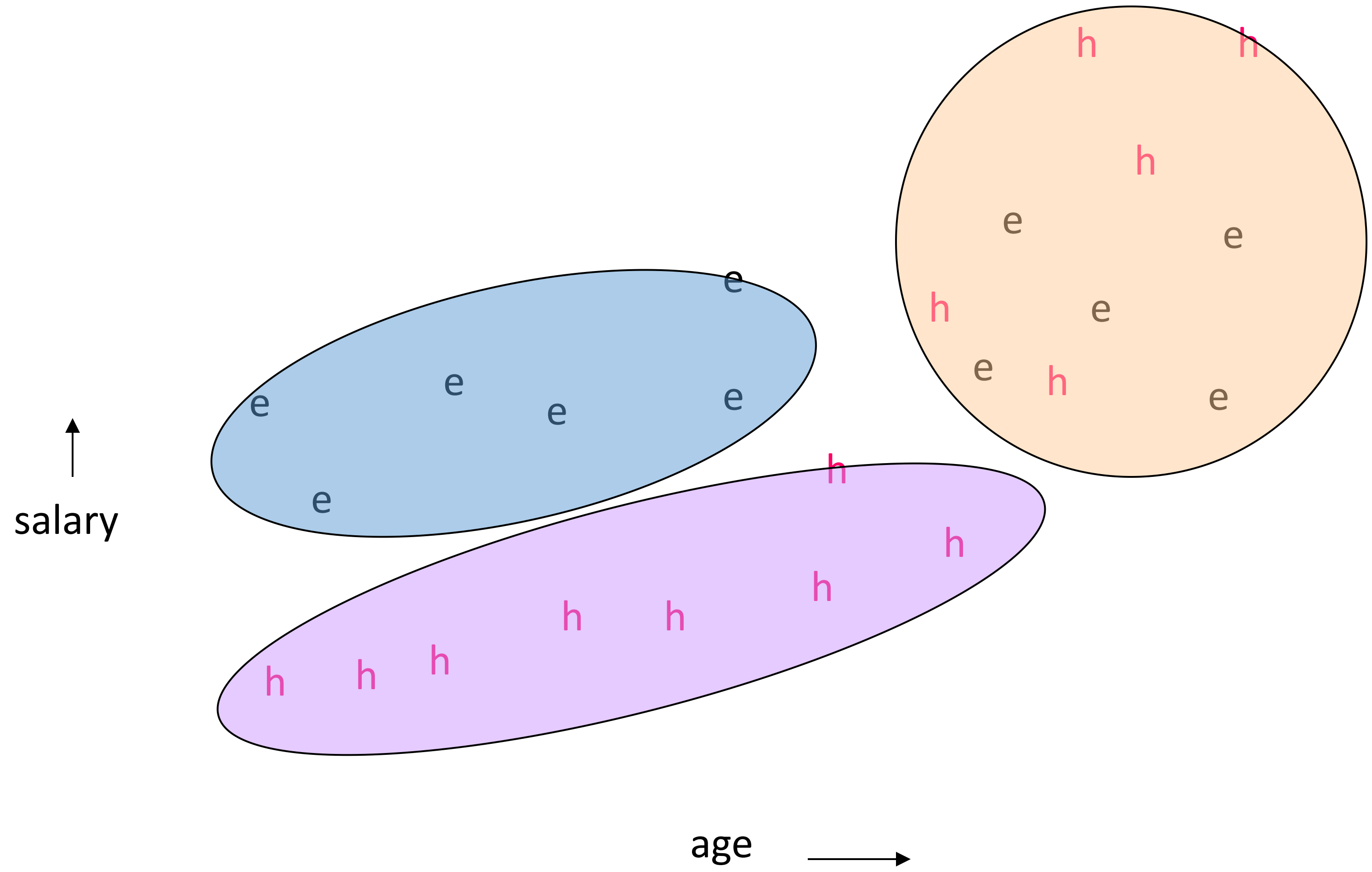


e: 人文学科教师
h: 工科教师

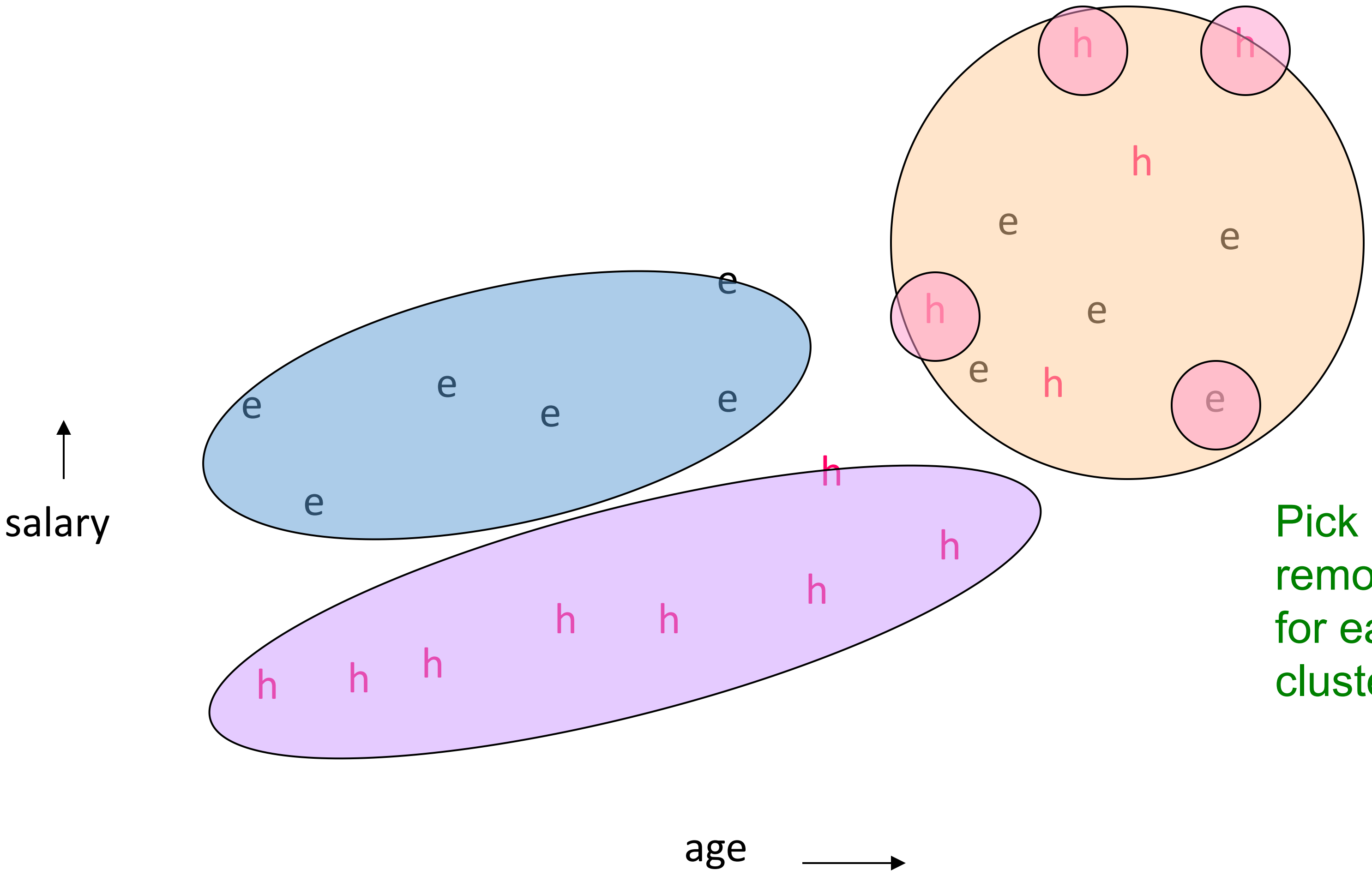
Starting CURE

- **CURE algorithm has 2 passes.**
- **Pass 1:**
- **0) Pick a random sample of points that fit in main memory**
- **1) Initial clusters:**
  - Cluster these points hierarchically – group nearest points/clusters
- **2) Pick representative points:**
  - For each cluster, pick a sample of points, as dispersed as possible
  - From the sample, pick representatives by moving them (say) 20% toward the centroid of the cluster

Example: Initial Clusters

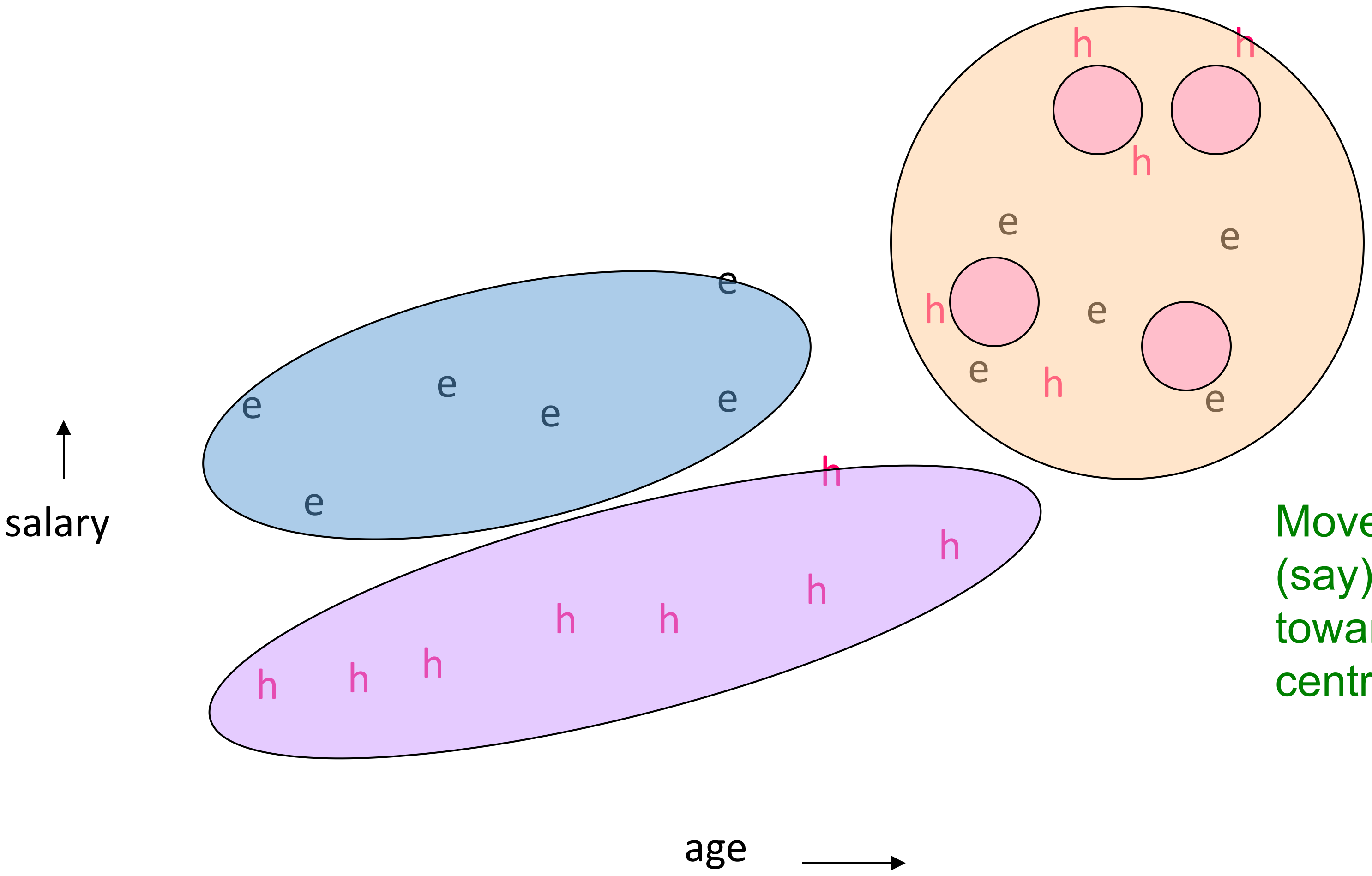


Example: Pick Dispersed Points



Pick (say) 4 remote points for each cluster.

Example: Pick Dispersed Points



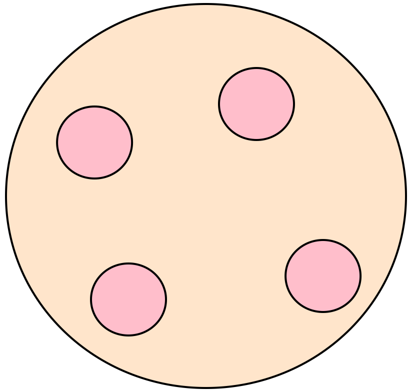
Move points (say) 20% toward the centroid.

Finishing CURE

<ins>**Pass 2:**</ins>

- Now, rescan the whole dataset and visit each point ***p*** in the data set

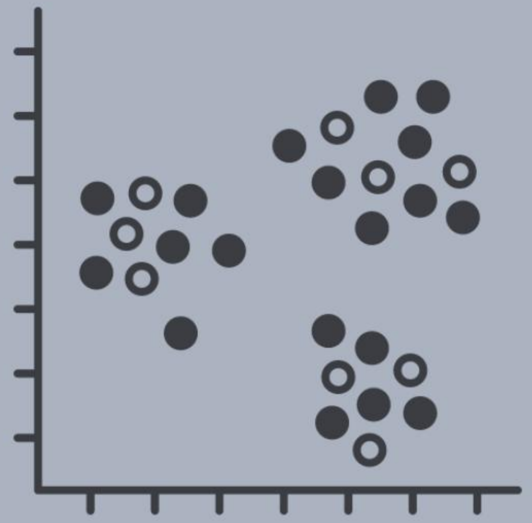
- **Place it in the "closest cluster"**
  - Normal definition of "closest": Find the closest representative to ***p*** and assign it to representative's cluster



p

CURE: Approach

- CURE is positioned between centroid based (d_{ave}) and all point (d_{min}) extremes.
  - A constant number of well scattered points is used to capture the shape and extend of a cluster.
  - The points are shrunk towards the centroid of the cluster by a factor α.
  - These well scattered and shrunk points are used as representative of the cluster. Scattered points approach alleviates shortcomings of d_{ave} and d_{min}.
    - Since multiple representatives are used the splitting of large clusters is avoided.
    - Multiple representatives allow for discovery of non spherical clusters.
    - The shrinking phase will affect outliers more than other points since their distance from the centroid will be decreased more than that of regular points.



Cluster Validity

Cluster Validity

- 评估聚类结果的有效性, 即**聚类评估**(Cluster Validity, 或称**聚类验证**), 对于聚类应用程序的成功至关重要.
  - 可以确保聚类算法在数据中识别出有意义的聚类
  - 还可以用来确定哪种聚类算法最适合特定的数据集和任务, 并调优这些算法的超参数.
  - To compare clustering algorithms
  - To compare two clusters
  - To avoid finding patterns in noise

- For cluster analysis, the analogous question is how to evaluate the "goodness" of the resulting clusters?

Measures of Cluster Validity

华中科技大学 计算机科学与技术学院 School of Computer Science &Technology, HUST

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following three types:
  - Internal Index: Used to measure the goodness of a clustering structure *without* respect to external information.
    - E.g., Sum of Squared Error (SSE)、 Cohesion and Separation
  - External Index: Used to measure the extent to which cluster labels match externally supplied class labels.
    - E.g., Purity、Entropy
  - Relative Index: Used to compare two different clusterings or clusters.
    - Often an external or internal index is used for this function, e.g., SSE or entropy

Internal Measures: SSE

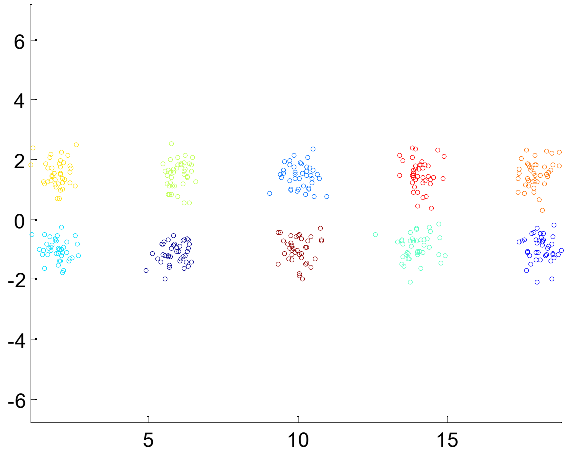
- **Internal Index**: Used to measure the goodness of a clustering structure without respect to external information
- **SSE (平方误差和)** is good for comparing two clusterings or two clusters (average SSE):

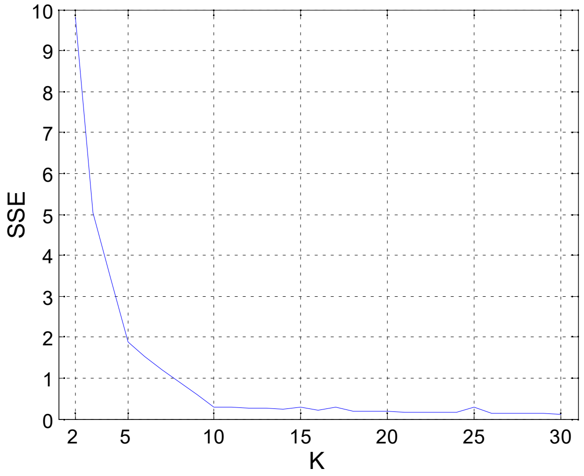
$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist^2(m_i, x)$$

  - where x is a data point in cluster c_i, and m_i is a representative point for cluster c_i . If we' re given two sets of clusters, we prefer the one with the smallest error

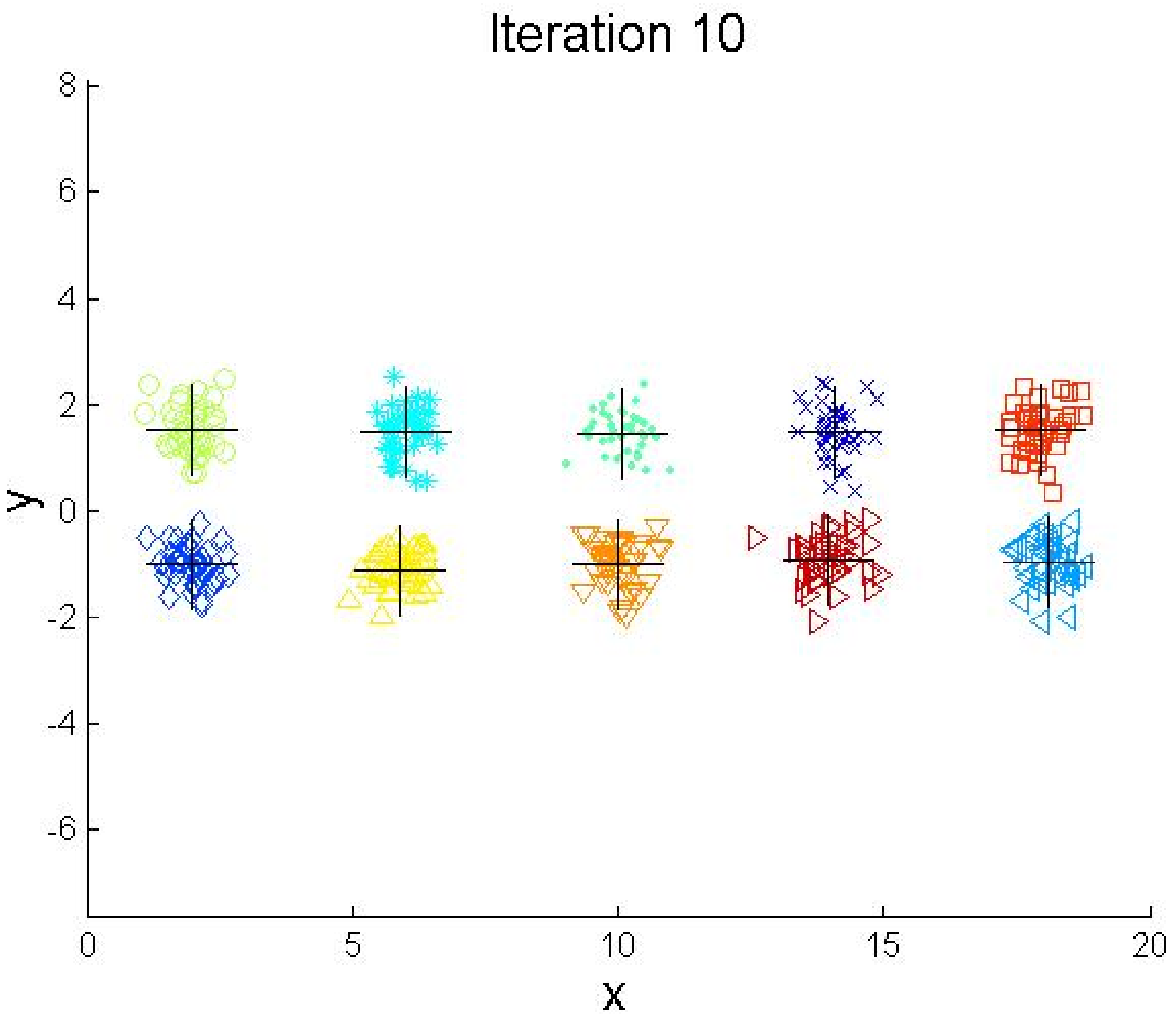
Internal Measures: SSE

- **SSE** Can also be used to estimate the number of clusters



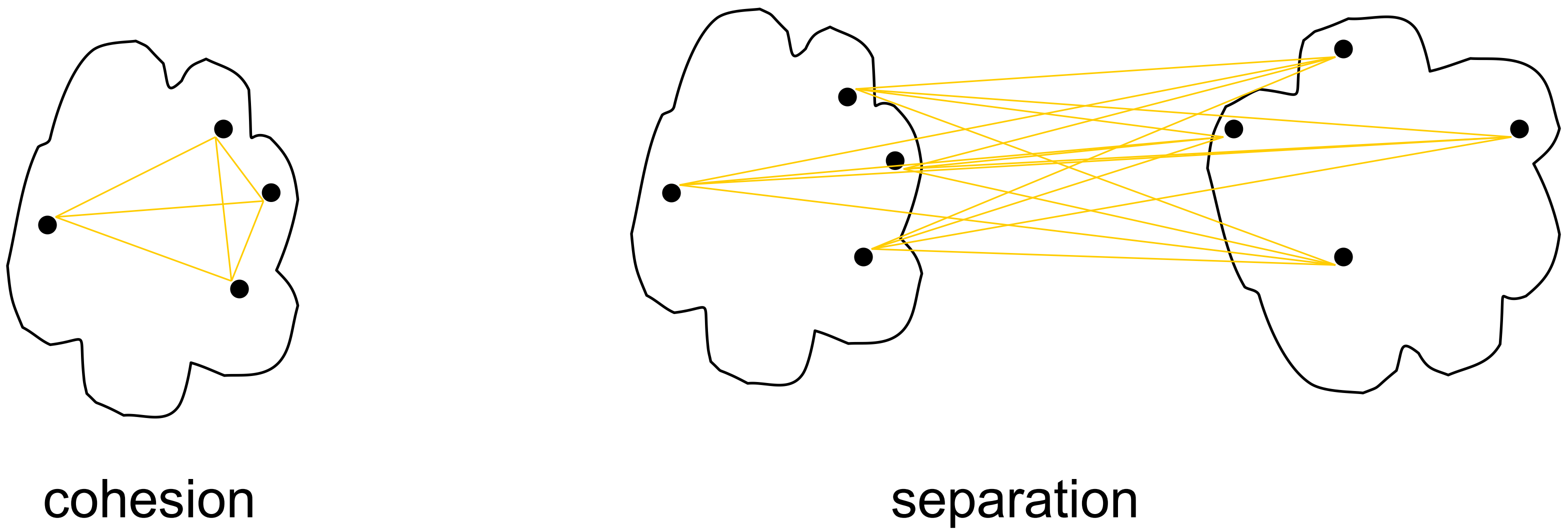


Internal Measures: SSE



Internal Measures: Cohesion and Separation

- A proximity graph based approach can also be used for **cohesion and separation**.
  - Cluster cohesion is the sum of the weight of all links within a cluster.
  - Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



What Is A Good Clustering?

- Internal criterion: A good clustering will produce high quality clusters in which:
  - the intra-class (that is, intra-cluster) similarity is high
  - the inter-class similarity is low
  - The measured quality of a clustering depends on both the point representation and the similarity measure used

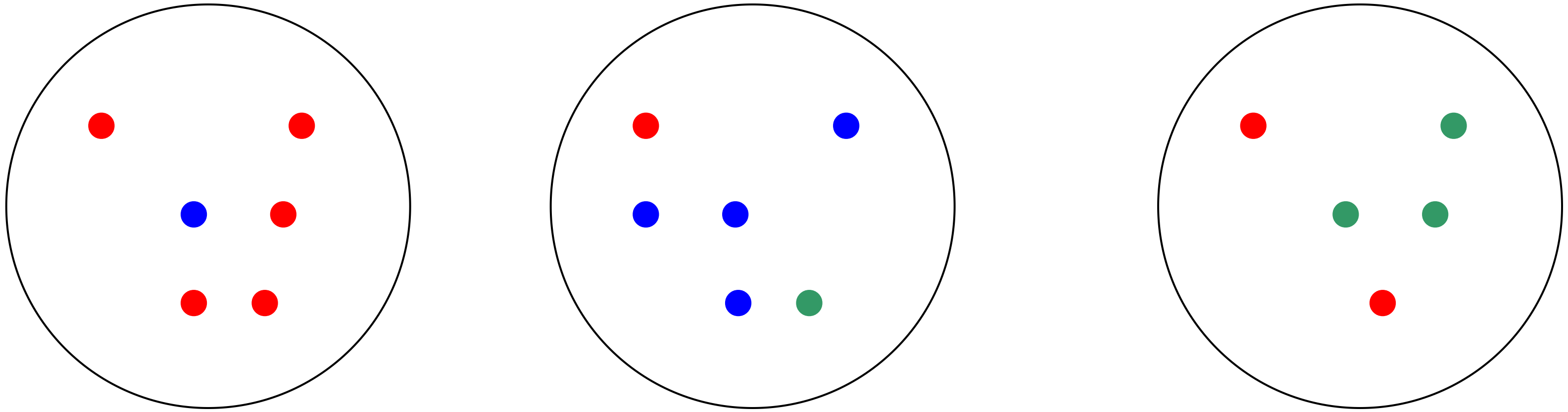
External Evaluation of Cluster Quality

- **External Index**: Used to measure the extent to which cluster labels match externally supplied class labels.

- Simple measure: **Purity(纯度),** the ratio between the dominant class in the cluster and the size of cluster ω_i

$$Purity(\omega_i) = \frac{1}{n_i} \max_j (n_{ij}) \quad j \in C$$

Example: Purity



Cluster I Cluster II Cluster III

- Cluster I: Purity = 1/6 (max(5, 1, 0)) = 5/6
- Cluster II: Purity = 1/6 (max(1, 4, 1)) = 4/6
- Cluster III: Purity = 1/5 (max(2, 0, 3)) = 3/5

External Evaluation of Cluster Quality

- Others: e.g., entropy (熵) of classes in clusters (or mutual information between classes and clusters), rand index (兰德系数), F value, adjusted rand index (调整兰德系数), et al.

Final Comment

- In clustering, clusters are inferred from the data **without human input** (**unsupervised learning**).
- However, in practice, it’ s a bit less clear: there are many ways of influencing the outcome of clustering: number of clusters, similarity measure, representation of points, . . .
- “The validation of clustering structures is the **most difficult and frustrating part** of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage.” -Algorithms for Clustering Data, Jain and Dubes

- **Clustering:** Given a **set of points**, with a notion of **distance** between points, **group the points** into some number of ***clusters***
- **Algorithms:**
  - **Hierarchical Clustering**:
    - Centroid and clustroid
  - ***k*-means:**
    - Initialization, picking *k*
  - **BFR**
  - **CURE**