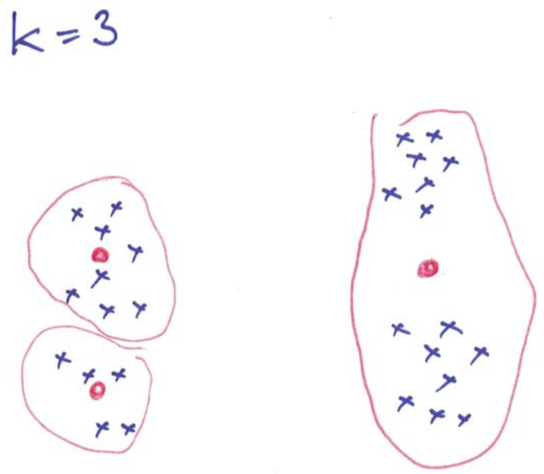
Common problems with k-means clustering

1. The algorithm attempts to optimize global error function J but instead converges to a local minimum



Common problems with k-means clustering

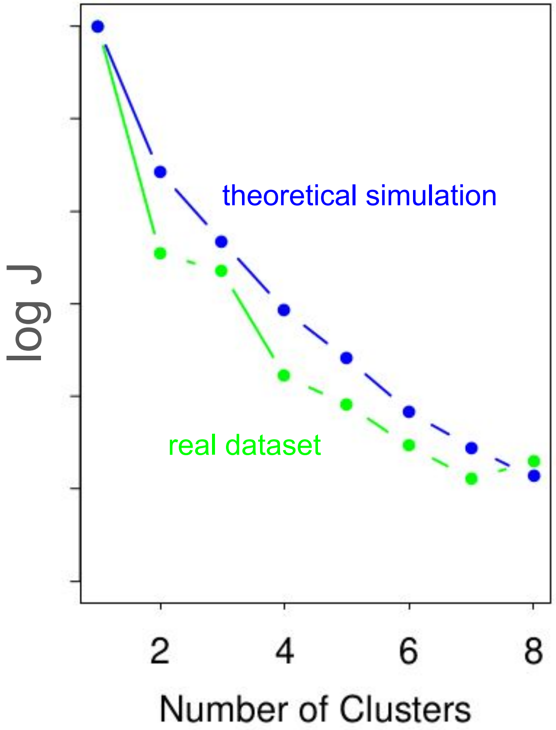
2. **How to determine number of clusters?**

can try to compare

- **theoretical error curve** (simulation on a dataset with no cluster structure)
- **error curve on the real dataset**

the error "drop" when increasing number of clusters should be larger than that on the theoretical curve until the "correct" number of clusters is reached

(k=2 or k=4 are good candidates here)



Common problems with k-means clustering

3. Features are categorical variables (not real numbers)

Examples:

- clustering of newspaper articles
- colors

Algorithm k-medians

Need to define a distance measure d(x,z):

- d(x,z)=0 if x=z
- d(x,z)=d(z,x) (symmetry)
- typically weighted sum of distance measures for individual features

Examples:

- quantitative features: Euclidean distance
- ordinal features: replace with real values from [0,1], treat as quantitative features
- categorical features: table

Differences from k-means algorithm:

- cannot easily define centers as new points in the space
  (what is the average of “chicken” and “pig”?)
- instead use existing points from the data set to characterize clusters
- **no guarantee of convergence**

Algorithm k-medians

1. [Initialization] Randomly choose k median points $m_1, ..., m_k$ out of all input vectors $x_1, \ldots, x_t$
2. Repeat until convergence:
   a. Assign each input vector to its closest median point:

$$c_i := \arg\min_j d(x_i, m_j)$$

   b. Choose new median points:

$$m_j := \arg\min_{m_i : c_i = j} \sum_{k : c_k = j} d(x_k, m_i)$$

Algorithm k-means

1. [Initialization] Randomly choose k centers $\mu_1, \ldots, \mu_k$ out of all input vectors $x_1, \ldots, x_t$
2. Repeat until convergence:
   a. Assign each input vector to its closest center:

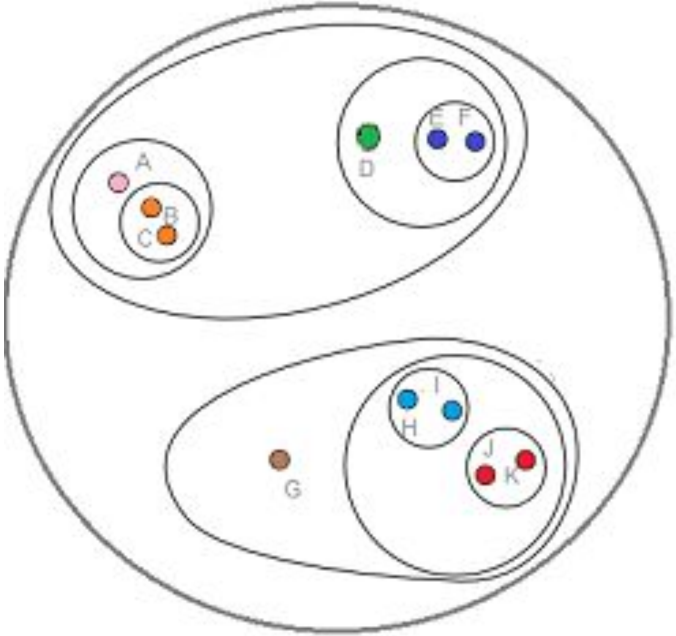
$$c_i := \arg\min_j ||x_i - \mu_j||_2$$

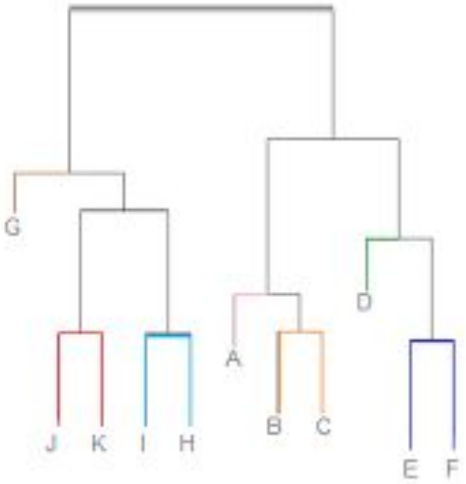
   b. Choose new centers:

$$\mu_j := \text{avg}_{i:c_i=j} x_i$$

Hierarchical clustering

not a single structure of clusters, instead a nested data structure:





Hierarchical clustering algorithms

Agglomerative clustering (bottom up)

- start with each data point being in a separate cluster
- in each step, **merge two clusters A, B** that are "closest" to each other based on:
    - **single linkage:** the distance of the two closest points from A and B
    - **group average:** distance of center of A to center of B
    - **complete linkage:** average or sum of all-to-all distances between points of A and B

Divisive clustering (top to bottom)

- star with all points being a single cluster
- in each step, choose one cluster **and divide it into two clusters**
  for example:
    - choose a particular cluster G
    - choose its "most distant" point and move it to a new cluster H
    - iterate through all points x in H and if $\mathrm{avg}_{z \in H} d(x, z) < \mathrm{avg}_{z \in G} d(x, z)$ move x to cluster H
    - continue until no more points can be moved from G to H