CS522 Advanced Database Systems

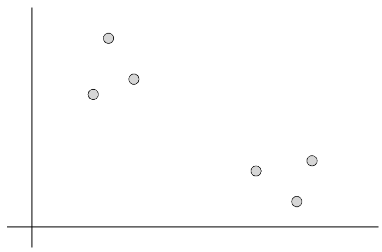
Clustering: K-Means

Chengyu Sun
California State University, Los Angeles

K-Means

- Input: dataset `D` and number of clusters `k`
- Algorithm
  1. Randomly choose k objects as cluster centers
  2. Assign each object to the closest cluster center
  3. Update each cluster center
  4. Repeat 2 until there is no reassignment occurs

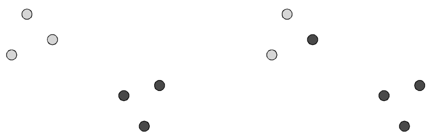
K-Means Example



Key Issues in K-Means

- Distance measure?
  - Euclidean, Manhattan, Cosine ...
- Cluster center?
  - Mean, median

Need for Objective Function



(a) (b)

- The best clustering is the one that minimize the "errors" defined by an *objective function*

Notations

`D`	Dataset
`k`	The number of clusters
C_i	ith cluster
c_i	The center of the ith cluster
`x`	An object

Objective Functions

Sum of the Squared Error (SSE):

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} dist_{L_2}(x, c_i)^2$$

Sum of the Absolute Error (SAE):

$$SAE = \sum_{i=1}^{k} \sum_{x \in C_i} dist_{L_1}(x, c_i)$$

Minimize an Object Function

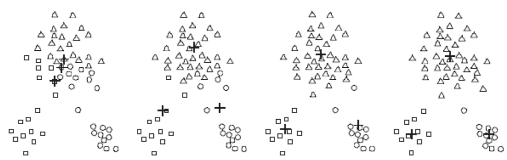
- Example:
  - One dimensional data
  - One cluster
  - SSE

$$SSE(c) = \sum_{x \in C} (c - x)^2 \quad \Rightarrow \quad \frac{\partial}{\partial c} SSE(c) = 0$$

Distances, Centroids, and Objective Functions

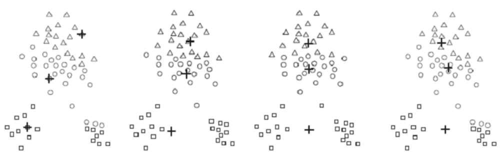
Distance Function	Centroid	Objective Function
Manhattan (L_1)	Median	Sum of L_1 distance
Squared Euclidean (L_2)	Mean	Sum of squared L_2 distance
Cosine	Mean	Sum of cosine distance
Bregman Divergence	Mean	Sum of Bregman divergence

Another K-Means Example ...



(a) Iteration 1. (b) Iteration 2. (c) Iteration 3. (d) Iteration 4.

... Another K-Means Example



(a) Iteration 1. (b) Iteration 2. (c) Iteration 3. (d) Iteration 4.

Dealing with the Problem of Initial Centroid Selection

- Perform several runs of K-Means and select the clustering with the smallest SSE
  - What's the probability of picking K objects, each from a different cluster??
- Use a hierarchical clustering algorithm on a sample to get K initial clusters
- Select centroid one by one, and each one is the farthest away from previously selected ones

Postprocessing

- Escape local SSE minimum by performing alternate clustering *splitting* and *merging*

Postprocessing – Splitting

- Splitting the cluster with the largest SSE on the attribute with the largest variance
- Introduce another centroid
  - The point that is farthest from current centroids
  - Randomly chosen

Postprocessing – Merging

- Disperse a cluster and reassign its objects
- Merge two clusters that are closest to each other

Bisecting K-Means

1. Initialize a list of clusters with one cluster containing all the objects
2. Choose one cluster from the list
3. Split the cluster into two using basic K-Means, and add them back to the list
4. Repeat Step 2 until `k` clusters are reached
5. Perform one more basic K-Means using the centroids of the `k` clusters as initial centriods

About Bisecting K-Means

- Step 2
  - Choose the largest cluster
  - Choose the cluster with the largest SSE
- Step 3
  - Perform basic K-Means several times and choose the clustering with the smallest SSE
- Less susceptible to initialization problems
  - Why??

Handling Empty Clusters

- Choose a replacement centroid
  - The point that's farthest away from any current centroid
  - A point from the cluster with the highest SSE

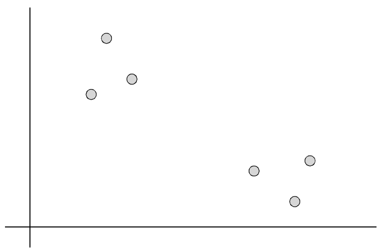
K-Medoids

- Instead of using mean/centroid, use medoid, i.e. representative object
- Objective function: sum of the distances of the objects to their medoid
- Differs from K-Means in how the medoids are updated

PAM (Partition Around Medoids)

1. Randomly choose `k` objects as initial medoids
2. For each non-medoid object `x`
   For each medoid c_i
   calculate the reduction of the total distance if c_i is replaced by `x`
3. Replace the c_i with `x` that results in maximum total distance reduction
4. Repeat Step 2 until the total distance cannot be reduced
5. Assign each object to its closest mediod

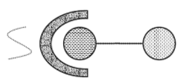
PAM Example



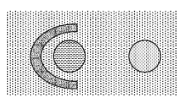
K-Means vs. K-Medoids

K-Means	K-Medoids
Requires the notion of mean/centroid	Works for all distance measures
More susceptible to outliers	Less susceptible to outliers
$O(k(n-k))$ per iteration	$O(k(n-k)^2)$ per iteration

Limitations of K-Means – Different Types of Clusters

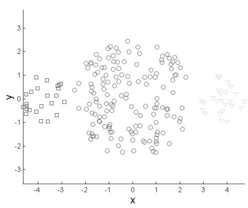


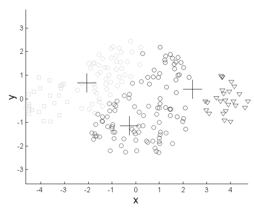
Continuity-based



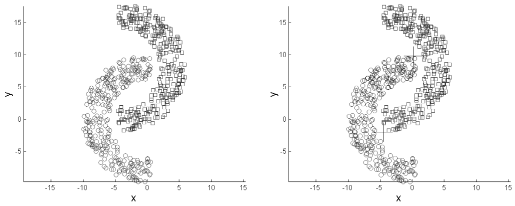
Density-based

Limitations of K-Means – Differing Sizes





Limitations of K-Means –Non-globular Shapes



Readings

- Textbook 7.4