SoSe 2014: M-TANI: Big Data Analytics

Lecture 4 – 21/05/2014

Sead Izberovic
Dr. Nikolaos Korfiatis
Agenda

• Recap from the previous session
• Clustering
  ▪ Introduction
  ▪ Distance measures
  ▪ Hierarchical Clustering
  ▪ Partitional Clustering
Introduction

• Given a set of objects, with a notion of distance between those objects. The task of a clustering algorithm is to group those objects into some number of clusters, so that:
  ▪ Members of a cluster are similar to each other
  ▪ Members of different clusters are dissimilar

• High dimensionality may be hard to interpret

from [2]
Introduction

• Clustering ≠ Classification

• Classification
  ▪ Assigning objects to predefined classes
  ▪ Requires supervised learning

• Clustering
  ▪ No predefined classes
  ▪ Assigning objects to clusters (based on distance)
Introduction

• Clustering applications
  ▪ Clustering DNA sequences
  ▪ Image segmentation
  ▪ Customer segmentation
  ▪ ...

Distance measures

• Calculate similarity
  ▪ Large distance = low similarity
  ▪ Small distance = high similarity

• Conditions
  1. \( dist(x, y) = d \) with \( d: X \times X \rightarrow R \) and \( x, y, z \in X \)
  2. \( dist(x, y) \geq 0 \)
  3. \( dist(x, y) = 0 \) if \( x = y \)
  4. \( dist(x, y) = dist(y, x) \)
  5. \( dist(x, z) \leq dist(x, y) + dist(y, z) \)

from [1] and [3]
Distance measures

• Euclidian distance: $dist_{euclid}(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

• Jaccard distance: $dist_{jaccard}(x, y) = 1 - SIM(x, y)$

• Cosine distance: $dist_{cos}(X, Y) = \frac{\sum_{i=1}^{n} x_i \times y_i}{\sqrt{\sum_{i=1}^{n}(x_i)^2} \times \sqrt{\sum_{i=1}^{n}(x_i)^2}}$

• Edit distance: smallest number of insertions and deletions of single characters that will convert string $x$ to string $y$ from [2] and [3]
Distance between clusters

Centroid

Medoid/Clustroid

adapted from [3]
Distance between clusters

Single-linkage

\[ \text{dist}_{SL}(C_x, C_y) = \min_{x \in C_x, y \in C_y} \text{dist}(x, y) \]

adapted from [3]
Distance between clusters

Complete-linkage

\[ \text{dist}_{CL}(C_x, C_y) = \max_{x \in C_x, y \in C_y} \text{dist}(x, y) \]

adapted from [3]
Distance between clusters

Average-linkage

$$\text{dist}_{AL}(C_x, C_y) = \frac{1}{|C_x| \cdot |C_y|} \cdot \sum_{x \in C_x, y \in C_y} \text{dist}(x, y)$$

adapted from [3]
Clustering approaches

- Hierarchical
- Partitional

adapted from [4]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between to clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between to clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster

2. Calculate the minimal distance between to clusters

3. Merge clusters with minimal distance

4. Repeat 2. and 3. until we have one big cluster

Single Link

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between to clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative

1. Every Object is a cluster
2. Calculate the minimal distance between to clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

adapted from [3]
Hierarchical Agglomerative Clustering

1. Every Object is a cluster
2. Calculate the minimal distance between two clusters
3. Merge clusters with minimal distance
4. Repeat 2. and 3. until we have one big cluster

Single Link

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

Single Link

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

adapted from [3]
Hierarchical Divisive

1. Find max. distance between objects
2. Split those clusters
3. Repeat 1. and 2. until we have clusters which contain just one object

Single Link

adapted from [3]
Cluster Visualization

Dendrogram

adopted from [3]
Cluster Visualization

Dendrogram

adopted from [3]
Cluster Visualization

Dendrogram

adopted from [3]
Cluster Visualization

adopted from [3]
Cluster Visualization

 adopted from [3]
Cluster Visualization

Dendrogram

adopted from [3]
Cluster Visualization

adopted from [3]
Cluster Visualization

Dendrogram

adopted from [3]
Clustering approaches

Clustering

Hierarchical

Partitional

adapted from [4]
K-Means

1. Place k centroids randomly
2. If distance to centroid min. merge to cluster
3. Move the k centroids to the new cluster center
4. Repeat 2. and 3. until we fulfil the stop criteria

$k=3$

adapted from [3]
K-Means

1. Place k centroids randomly

2. If distance to centroid min. merge to cluster

3. Move the k centroids to the new cluster center

4. Repeat 2. and 3. until we fulfil the stop criteria

adapted from [3]
K-Means

1. Place k centroids randomly
2. If distance to centroid min. merge to cluster
3. Move the k centroids to the new cluster center
4. Repeat 2. and 3. until we fulfil the stop criteria

adapted from [3]
K-Means

1. Place k centroids randomly

2. If distance to centroid min. merge to cluster

3. Move the k centroids to the new cluster center

4. Repeat 2. and 3. until we fulfil the stop criteria

adapted from [3]
K-Means

1. Place k centroids randomly
2. If distance to centroid min. merge to cluster
3. Move the k centroids to the new cluster center
4. Repeat 2. and 3. until we fulfil the stop criteria

adapted from [3]
K-Means

• How to choose the right $k$?
  
  ▪ Try different $k$, looking at the change in the average distance to centroid as $k$ increases
  
  ▪ Average falls rapidly until right $k$, then changes little

![Diagram showing the average distance to centroid as $k$ increases](image-url)
Clustering on graphs

- Create the MST
- Remove $k-1$ edges with the highest weights
- Create the clusters

$k=3$

adapted from [3]
Clustering on graphs

- Create the MST
- Remove k-1 edges with the highest weights
- Create the clusters

adapted from [3]
Clustering on graphs

- Create the MST
- Remove k-1 edges with the highest weights
- Create the clusters

adapted from [3]
Clustering on graphs

- Create the MST
- Remove k-1 edges with the highest weights
- Create the clusters

adapted from [3]
Literature

   Mining of Massive Datasets
   Cambridge University Press

   Slides: Mining Massive Data Sets
   URL: http://www.stanford.edu/class/cs246/slides/05-clustering.pdf

   Knowledge Discovery in Databases
   Springer – Verlag Berlin Heidelberg

   Data clustering: a review
   ACM Comput. Surv., 31(3):264-323