

AAIL100: Introduction to Data Science

Ragesh Jaiswal, IITD

- What is data clustering?

- What is data clustering?
 - The task of grouping a set of objects such that objects in the same group (called *cluster*) are more similar than objects in different groups.
 - Given a *representation* of n objects, find k groups based on a measure of *similarity* (dissimilarity) such that the similarities between objects in the same group are high while similarities between objects in different groups are low.

- Why study data clustering?
 - Clustering is usually the first step when trying to make sense of Big Data.
 - Common tool across many different fields such as image analysis, pattern recognition, bioinformatics, information retrieval etc.
- Data clustering has been used for the following three main purposes [Jain'09]:
 - *Underlying structure*: to gain insight into data, generate hypotheses, detect anomalies, and identify salient features.
 - *Natural classification*: to identify the degree of similarity among forms or organisms.
 - *Compression*: as a method for organizing the data and summarizing it through cluster prototypes.

- How is clustering different from “classification”?

- How is clustering different from “classification”?

Classification	Clustering
The task of assigning objects to predefined “classes”.	The task of grouping objects without any knowledge of the “classes”.

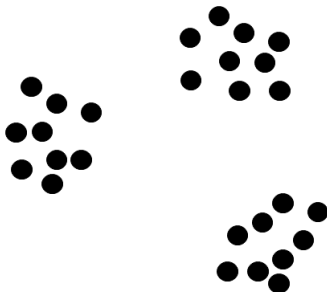
- The more general terms used are *supervised* learning versus *unsupervised* learning.

Supervised Learning	Unsupervised Learning
- Classes/labels known - Expert available - Training data available	- Classes/labels unknown - No experts - No training data

- What is data clustering?
 - Given a *representation* of n objects, find k groups based on a measure of *similarity* (dissimilarity) such that the similarities between objects in the same group are high while similarities between objects in different groups are low.
- Suppose the given objects to be clustered can be *represented* as points in two-dimensional space (i.e., $\mathbb{R}^2$).
 - What is a reasonable notion of *similarity* between objects?

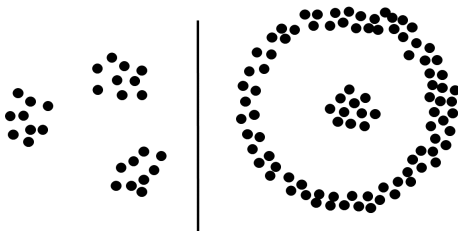
Introduction

- What is data clustering?
 - Given a *representation* of n objects, find k groups based on a measure of *similarity* (dissimilarity) such that the similarities between objects in the same group are high while similarities between objects in different groups are low.
- Suppose the given objects to be clustered can be *represented* as points in two-dimensional space (i.e., $\mathbb{R}^2$).
 - What is a reasonable notion of *similarity* between objects?
 - Distance between points.



Introduction

- What is data clustering?
 - Given a *representation* of n objects, find k groups based on a measure of *similarity* (dissimilarity) such that the similarities between objects in the same group are high while similarities between objects in different groups are low.
- Suppose the given objects to be clustered can be *represented* as points in two-dimensional space (i.e., $\mathbb{R}^2$).
 - What is a reasonable notion of *similarity* between objects?
 - Distance between points.
 - The notion of similarity/dissimilarity has to be defined carefully.



Data Representation

- What is data clustering?
 - Given a *representation* of n objects, find k groups based on a measure of *similarity* (dissimilarity) such that the similarities between objects in the same group are high while similarities between objects in different groups are low.
- The “data” for cluster analysis can be described by two standard formats:
 - Pattern Matrix
 - Proximity Matrix

- The “data” for cluster analysis can be described by two standard formats:
 - Pattern Matrix
 - Data is represented as an $n \times d$ matrix where each row corresponds to a “pattern/item/object” and each column denotes a “feature/measurement”.
 - Example: For patient records in a hospital each row corresponds to a patient and each column denotes a feature such as age, weight, height, measurement for certain magical tests etc.
 - The d features are usually visualised as a set of orthogonal axes. Given this, the items then are points in a d -dimensional space called pattern space.
 - Proximity Matrix

- The “data” for cluster analysis can be described by two standard formats:
 - Pattern Matrix
 - Proximity Matrix
 - This is an $n \times n$ matrix where n denotes the number of items/patterns. The entries in this matrix is called proximity indices. The $(i, j)^{th}$ entry in this matrix denotes the proximity between the i^{th} and j^{th} item.
 - Proximity could indicate similarity or dissimilarity. For example for dissimilarity $D(i, i) = 0$ and for similarity $D(i, i) \geq \max_k D(i, k)$.

- The “data” for cluster analysis can be described by two standard formats:
 - *Pattern Matrix*: $n \times d$ matrix denoting the data points.
 - *Proximity Matrix*: $n \times n$ matrix denoting the pairwise proximity between data points.
- How to interpret the numbers in the above matrices? The numbers can be of the following nature
 - *Nominal*: The numbers are used as names. For example, a yes/no response can be encoded as 0/1 or 500/1000 etc.
 - *Ordinal*: The numbers have meaning with respect to each other. That is, column entries 1, 2, 3 is equivalent to the column entries 1, 20, 300.
 - *Ratio scale*: The numbers have absolute meaning. For example, distance between two cities, temperature etc.

Data Representation

- The “data” for cluster analysis can be described by two standard formats:
 - *Pattern Matrix*: $n \times d$ matrix denoting the data points.
 - *Proximity Matrix*: $n \times n$ matrix denoting the pairwise proximity between data points.
- How to interpret the numbers in the above matrices? The numbers can be of the following nature
 - *Nominal*:
 - *Ordinal*:
 - *Ratio scale*:
- Can we obtain the proximity matrix from the pattern matrix?
 - The most common dissimilarity measure using the pattern matrix (with ratio-scaled data) is the *Minkowski* metric which is defined as follows: Let x denote the pattern matrix. Then we have

$$D(i,j) = \left(\sum_{j=1}^d |x(i,j) - x(k,j)|^r \right)^{1/r} \quad \text{where } r \geq 1$$

- Can we obtain the proximity matrix from the pattern matrix?
 - The most common dissimilarity measure using the pattern matrix (with ratio-scaled data) is the *Minkowski* metric which is defined as follows: Let x denote the pattern matrix. Then we have

$$D(i, j) = \left(\sum_{j=1}^d |x(i, j) - x(k, j)|^r \right)^{1/r} \quad \text{where } r \geq 1$$

- Specific instances of Minkowski metric is given below:
 - *Euclidean distance*: $r = 2$
 - *Manhattan distance*: $r = 1$
 - *Sup distance*: $r \rightarrow \infty$. This means that
$$D(i, j) = \max_{1 \leq j \leq d} |x(i, j) - x(k, j)|$$
- The Euclidean distance is the most commonly used distance measure in Engineering.

- Can we obtain the proximity matrix from the pattern matrix?
 - In a number of settings the pattern matrix contains binary nominal values (i.e., 0/1 indicating yes/no)
 - In such cases the proximity index $D(i, j)$ is calculated in the following manner:
 - Let $a_{00} = |\{k : x(i, k) = 0 \text{ and } x(j, k) = 0\}|$
 - Let $a_{01} = |\{k : x(i, k) = 0 \text{ and } x(j, k) = 1\}|$
 - Let $a_{10} = |\{k : x(i, k) = 1 \text{ and } x(j, k) = 0\}|$
 - Let $a_{11} = |\{k : x(i, k) = 1 \text{ and } x(j, k) = 1\}|$
 - *Simple Matching Coefficient:* $D(i, j) = \frac{a_{00} + a_{11}}{a_{00} + a_{11} + a_{01} + a_{10}}$
 - *Jaccard Coefficient:* $D(i, j) = \frac{a_{11}}{a_{11} + a_{01} + a_{10}}$

- What can we do for “missing data”?
 - In a number of settings some data entries might be missing. For example, missing medical test for some individual etc.
 - Missing data is handled in the following manner:
 - Delete the items or features that have missing entries.
 - Suppose the j^{th} entry of the i^{th} item is missing. Find the k “nearest neighbours” of the i^{th} item and replace the missing entry with the average value of the j^{th} feature of these nearest items.
 - Skip the missing features while calculating the distance between pair of items.
 - Try computing the missing entries by assuming certain reasonable properties of the pattern matrix.

- How do we normalize across different features?
 - Some features might be recorded using a larger range of numbers (e.g. distance in inches) compared other features (e.g., distance in miles). When calculating dissimilarity using Minkowski metric, one feature might dominate the dissimilarity.
 - Here is the standard way to normalise the $n \times d$ pattern matrix x . Let y denote the normalised matrix.
 - For all j , let $m_j = \frac{1}{n} \sum_{i=1}^n x_{ij}$
 - For all j , let $s_j^2 = \frac{1}{n} \sum_{i=1}^n (x_{ij} - m_j)^2$
 - $y_{ij} = \frac{x_{ij} - m_j}{s_j}$
 - All features in y have 0 mean and unit variance.
- Is such normalisation always desirable when used for clustering purposes?
 - Can you think of an example?

- Can we reduce the dimensionality of the data?
 - Discussed later

The k -means Problem

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the 1-means problem?
 - Given $x_1, \dots, x_n \in \mathbb{R}^d$ find a point $z \in \mathbb{R}^d$ such that $f(z) = \sum_i \|x_i - z\|^2$ is minimized.

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the 1-means problem?
 - Given $x_1, \dots, x_n \in \mathbb{R}^d$ find a point $z \in \mathbb{R}^d$ such that $f(z) = \sum_i \|x_i - z\|^2$ is minimized.
 - What is $\frac{\partial f(z)}{\partial z_i}$?
 - So, for what z , $\sum_i \|x_i - z\|^2$ gets minimized?

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the 1-means problem?
 - Given $x_1, \dots, x_n \in \mathbb{R}^d$ find a point $z \in \mathbb{R}^d$ such that $f(z) = \sum_i \|x_i - z\|^2$ is minimized.
 - What is $\frac{\partial f(z)}{\partial z_i}$? $\frac{\partial f(z)}{\partial z_i} = nz_i - \sum_j x_{ji}$
 - So, for what z , $\sum_i \|x_i - z\|^2$ gets minimized? $z = \frac{\sum_i x_i}{n}$
 - $\frac{\sum_i x_i}{n}$ is called the *centroid* of the points $x_1, \dots, x_n$.

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the k -means problem for $k > 1$ when $d = 1$? In other words, how hard is the 1-dimensional k -means problem?

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the k -means problem for $k > 1$ when $d = 1$? In other words, how hard is the 1-dimensional k -means problem?
 - There is a simpler Dynamic Programming algorithms for this problem!

The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How hard is the k -means problem for $k > 1$ and $d > 1$?
 - NP-hard.

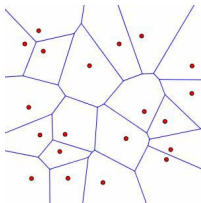
The k -means Problem

k -means

Given a set of points $S \subset \mathbb{R}^d$ in a d dimensional Euclidean space, and an integer k , output a set $T \subset \mathbb{R}^d$ of points (called *centers*) such that $|T| = k$ and the following cost function is minimised:

$$\text{cost}(T) = \sum_{x \in S} \min_{z \in T} \|x - z\|^2.$$

- How is this problem related to clustering?
 - The k centers induce a *voronoi* partition of $\mathbb{R}^d$ (and hence the given data points).
 - The voronoi partition corresponding to a center z is the region of space whose nearest center (among the k centers) is z .



The k -means Problem

The k -means Algorithm

- The most popular heuristic that used to solve the k -means problem in practice is the k -means algorithm (also known as Lloyd's Algorithm).

k -means Algorithm

- Initialize centers $z_1, \dots, z_k \in \mathbb{R}^d$.
- Repeat until there is no further change in cost:
 - For each j : $C_j \leftarrow \{x \in S \mid z_j \text{ is the closest center of } x\}$.
 - For each j : $z_j \leftarrow \text{Centroid}(C_j)$.

End