Introduction to Clustering

TOP: Data Clustering 076/091

Instructor: Sayan Bandyapadhyay

Portland State University

Outline

Clustering of Social Network

Dividing the customers into similar groups

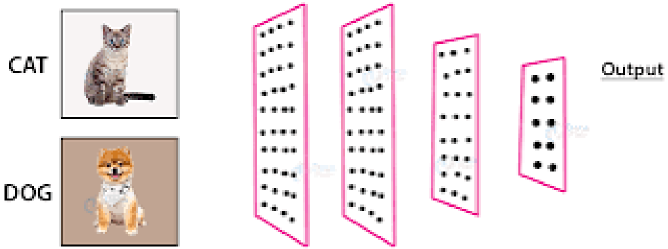
Applications

- *grouping of genes and proteins*, and *cancer and tumor detection* in Biology
- *speech recognition*, and *text summarization* in Natural Language Processing
- *grouping images*, and *image segmentation* in Computer Vision

Applications

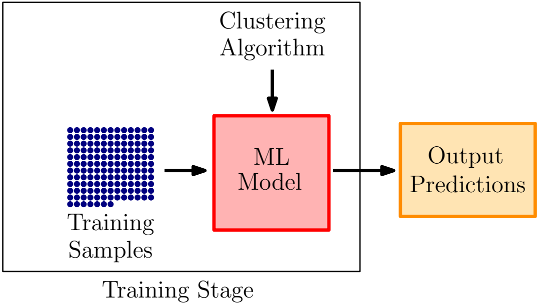
- *grouping of genes and proteins*, and *cancer and tumor detection* in Biology
- *speech recognition*, and *text summarization* in Natural Language Processing
- *grouping images*, and *image segmentation* in Computer Vision
- *Collaborative filtering*
- *Data summarization*
- *Dynamic trend detection*
- *Social network analysis*
- *Unsupervised learning*

Unsupervised learning



Building a classifier to identify cats and dogs images

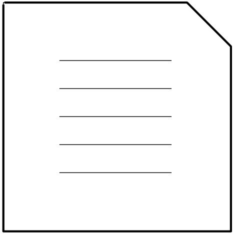
The ML Pipeline



Training the classifier: feature engineering/labeling of samples

Outline

The Mapping

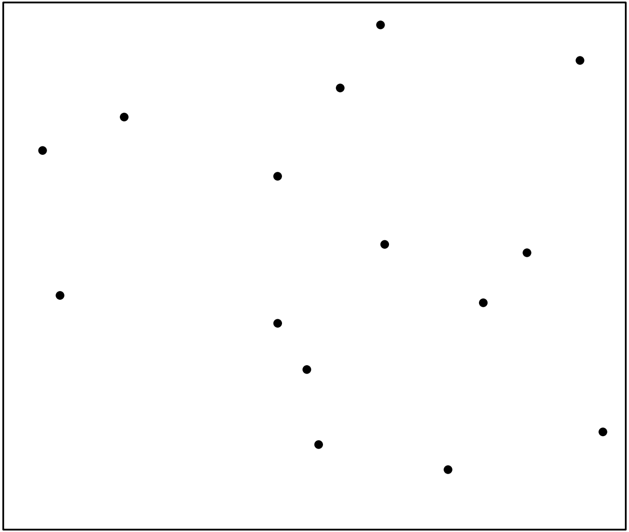


Profile

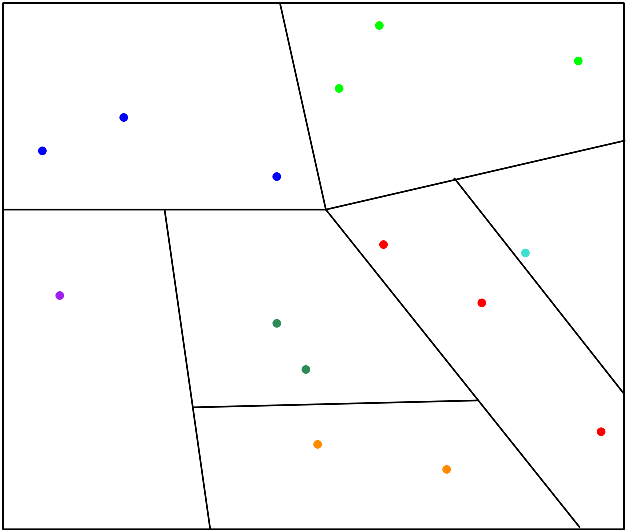
$$\longrightarrow \quad (F_1, F_2, \ldots, F_d)$$

Point in real space

Points in Real Space



Partition of Points



Drawback: Abstract Data Types

Not all data can be represented in numerical forms

- Categorical data: Gender, Address
- Text data
- Biological data: Gene expressions, Gene ontology annotations

Drawback: Abstract Data Types

Not all data can be represented in numerical forms

- Categorical data: Gender, Address
- Text data
- Biological data: Gene expressions, Gene ontology annotations

We will try to represent data in an abstract way

Outline

Metric

X is a set of points

Metric

X is a set of points

A function $d : X \times X \longrightarrow \mathbb{R}^+$ is said to be a **distance metric** if it has the following properties:

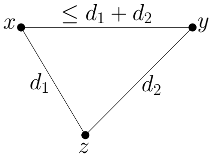
- *Reflexivity*: $\forall x, y \in X, d(x, y) = 0 \iff x = y$
- *Symmetry:* $\forall x, y \in X, d(x, y) = d(y, x)$
- *Triangle Inequality:*
  $\forall x, y, z \in X, d(x, y) \leq d(x, z) + d(z, y)$

Metric

X is a set of points

A function $d : X \times X \longrightarrow \mathbb{R}^+$ is said to be a **distance metric** if it has the following properties:

- *Reflexivity*: $\forall x, y \in X, d(x, y) = 0 \iff x = y$
- *Symmetry:* $\forall x, y \in X, d(x, y) = d(y, x)$
- *Triangle Inequality:* $\forall x, y, z \in X, d(x, y) \leq d(x, z) + d(z, y)$

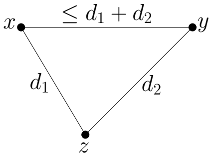


Metric

X is a set of points

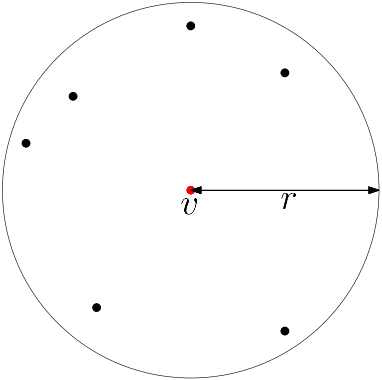
A function $d : X \times X \longrightarrow \mathbb{R}^+$ is said to be a **distance metric** if it has the following properties:

- *Reflexivity*: $\forall x, y \in X, d(x, y) = 0 \iff x = y$
- *Symmetry:* $\forall x, y \in X, d(x, y) = d(y, x)$
- *Triangle Inequality:*
  $\forall x, y, z \in X, d(x, y) \leq d(x, z) + d(z, y)$



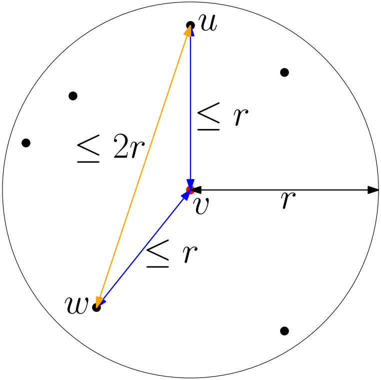
X along with the metric d is called a metric space (X, d)

The Idea of Metric Spaces



Ball $B(v, r)$ with center v and radius r

The Idea of Metric Spaces



Diameter of $B(v, r)$ is $\leq 2r$

Examples

Metric

- Euclidean distance: $X = \mathbb{R}^d : d(x, y) = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$
- Manhattan distance: $X = \mathbb{R}^d : d(x, y) = \sum_{i=1}^{d} |x_i - y_i|$
- $X = \Sigma^*$ is the set of finite length strings over an alphabet Σ, d is the edit distance
- X is a set of vertices in a graph G, d is the shortest path distance

Outline

Distances to clustering

C is a set of points

- Diameter of C: $\Delta(C) = \max_{x,y \in C} d(x, y)$

Distances to clustering

C is a set of points

- Diameter of C: $\Delta(C) = \max_{x,y \in C} d(x, y)$
- Is a measure of goodness of a cluster

Distances to clustering

C is a set of points

- Diameter of C: $\Delta(C) = \max_{x,y \in C} d(x, y)$
- Is a measure of goodness of a cluster

A partition of X, $\Pi(X) = \{C_1, C_2, \ldots, C_k\}$ such that

- $C_i \subset X; \forall i$
- $C_i \cap C_j = \emptyset; \forall i \neq j$
- $\Pi(X)$ is a cover: $\cup_{i=1}^{k} C_i = X$

Measuring Goodness via Cost

Cost of a partition $\Pi(X)$: $\text{Cost}(\Pi(X)) = \max_{i=1}^{k} \Delta(C_i)$

Measuring Goodness via Cost

Cost of a partition $\Pi(X)$: $\text{Cost}(\Pi(X)) = \max_{i=1}^{k} \Delta(C_i)$

***k*-partition problem**: Given a metric space (X, d), find a partition $\Pi(X)$ of size k that minimizes $\text{Cost}(\Pi(X))$

Measuring Goodness via Cost

Cost of a partition $\Pi(X)$: $\mathrm{Cost}(\Pi(X)) = \max_{i=1}^{k} \Delta(C_i)$

***k*-partition problem**: Given a metric space (X, d), find a partition $\Pi(X)$ of size k that minimizes $\mathrm{Cost}(\Pi(X))$

- Why k is needed?
- An example of a *model selection*

Cluster Representatives

- center of a cluster
- data compression/summarization

Cluster Representatives

- center of a cluster
- data compression/summarization

Should the center be in X?

- key patterns of variations of brain scans: Yes
- words that capture different topics: No (continuous)

Cluster Representatives

- center of a cluster
- data compression/summarization

Should the center be in X?

- key patterns of variations of brain scans: Yes
- words that capture different topics: No (continuous)

We use a universe $\mathcal{U}$: $X \subset \mathcal{U}$ and centers are also in $\mathcal{U}$

- (discrete) centers are from $X = \mathcal{U}$
- (continuous) centers are from $\mathcal{U}$ and not-necessarily in X

Outline

Center-Based Clustering: Voronoi property

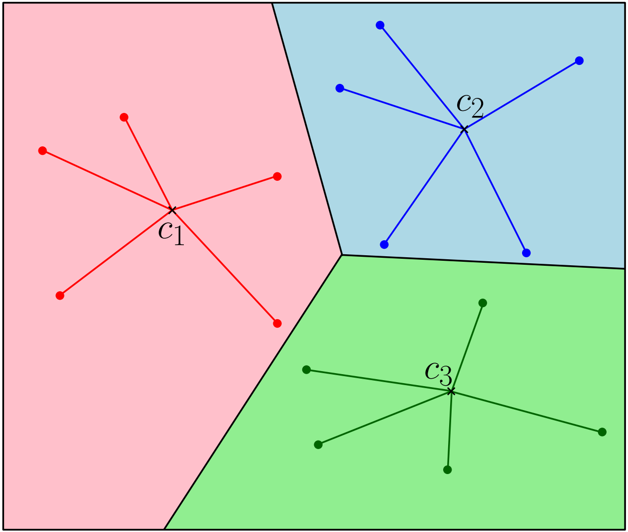
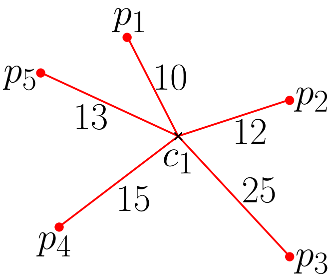


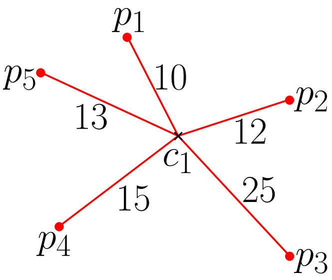
Figure: 3-cluster example

k-means Clustering



- Cost(p_1)=10^2, Cost(p_2)=12^2,. . ., Cost(p_5)=13^2

k-means Clustering



- Cost(p_1)=10^2, Cost(p_2)=12^2,. . ., Cost(p_5)=13^2
- Choose a set of cluster centers to minimize the sum of point costs

k-means Clustering

Given a set X of n points in the metric space $(\mathcal{U}, d)$

- Find a set C of k points (cluster centers) in $\mathcal{U}$ that minimizes,

$$\text{cost}(C) = \sum_{p} d(p, NearestCenter(p))^2$$

Popular Clustering Objectives

Find a set C of k points (cluster centers) in $\mathcal{U}$ that minimizes

$$k\text{-means: } \mathrm{cost}(C) = \sum_p d(p, \mathit{NearestCenter}(p))^2$$

$$k\text{-median: } \mathrm{cost}(C) = \sum_p d(p, \mathit{NearestCenter}(p))$$

$$k\text{-center: } \mathrm{cost}(C) = \max_p d(p, \mathit{NearestCenter}(p))$$

Outline

Finding the Best Clustering

All these problems are NP-hard

Finding the Best Clustering

All these problems are NP-hard

Solving exactly

- Discrete: $|X| = |\mathcal{U}| = n$. Pick the best k centers in X in $n^{O(k)}$ time
- Continuous: Pick the best k centers in $\mathcal{U}$ in $|\mathcal{U}|^{O(k)}$ time

Finding the Best Clustering

All these problems are NP-hard

Solving exactly

- Discrete: $|X| = |\mathcal{U}| = n$. Pick the best k centers in X in $n^{O(k)}$ time
- Continuous: Pick the best k centers in $\mathcal{U}$ in $|\mathcal{U}|^{O(k)}$ time

We can solve more efficiently if we are allowed to have some error in our solution

Coping with NP-hardness

- **Heuristics**
  - Simple (easy to implement)
  - Very time-efficient
  - Works well in practice
  - No quality control

Coping with NP-hardness

- **Heuristics**
  - Simple (easy to implement)
  - Very time-efficient
  - Works well in practice
  - No quality control
- **Approximation Algorithms**
  - Simple most of the time
  - Time-efficient
  - Works well in general
  - Quality control

Approximation Algorithms

- α-approximation algorithm: cost is within α-factor
  - minimum cost M; our cost $\leq \alpha \cdot M$;

Approximation Algorithms

- α-approximation algorithm: cost is within α-factor
  - minimum cost M; our cost $\leq \alpha \cdot M$;
  - 1-approximation is optimal
  - 2-approximation is 100% error

Approximation Algorithms

- α-approximation algorithm: cost is within α-factor
  - minimum cost M; our cost $\leq \alpha \cdot M$;
  - 1-approximation is optimal
  - 2-approximation is 100% error
- Approximation scheme
  - approximation to any desired precision: our cost $\leq (1 + \varepsilon) \cdot M$, for any $\varepsilon > 0$

Approximation Algorithms

- α-approximation algorithm: cost is within α-factor
  - minimum cost M; our cost $\leq \alpha \cdot M$;
  - 1-approximation is optimal
  - 2-approximation is 100% error
- Approximation scheme
  - approximation to any desired precision: our cost $\leq (1 + \varepsilon) \cdot M$, for any $\varepsilon > 0$
  - error is controlled


Thank You