Fair Clustering

Outline

- Introduction to clustering and algorithmic fairness
- Demographic (group) fairness in clustering
- Individual fairness in clustering
- The two-stage approach for solving fair clustering
- Overlooked issues in fair clustering.

Clustering: Motivation

- Clustering (**ML + Data Analysis**)

Clustering: Motivation

- Clustering (**ML + Data Analysis**)

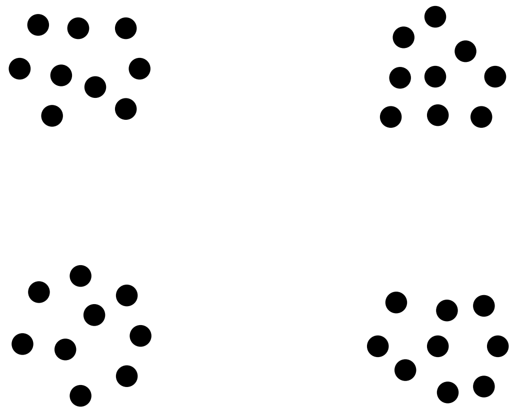
  Explore the data, Reveal existing structure, group similar points to one another, etc

Clustering: Motivation

- Clustering (**ML + Data Analysis**)

  Explore the data, Reveal existing structure, group similar points to one another, etc

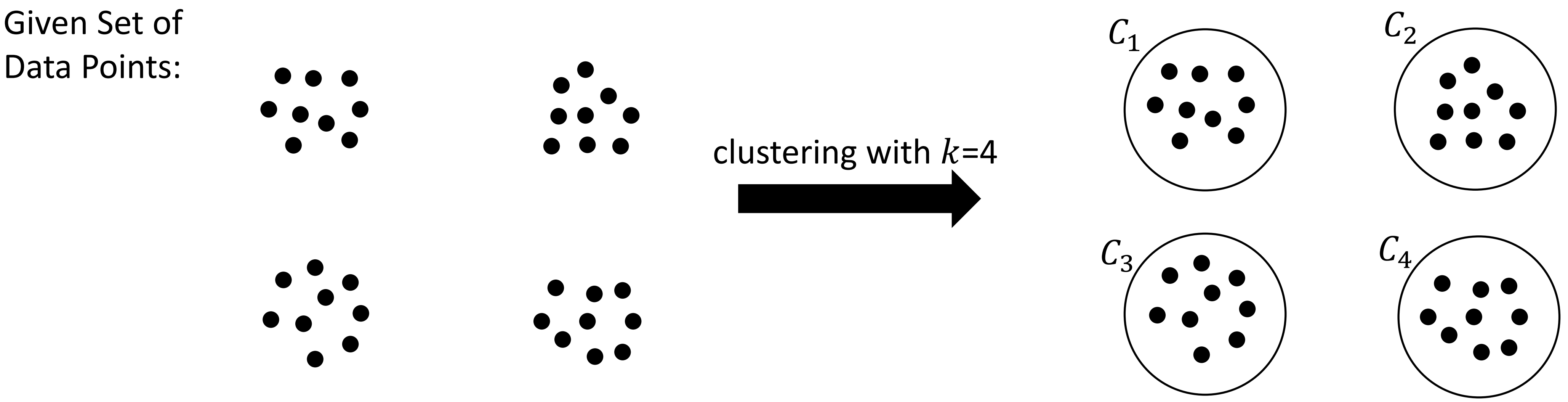
Given Set of
Data Points:



Clustering: Motivation

- Clustering (**ML + Data Analysis**)

  Explore the data, Reveal existing structure, group similar points to one another, etc



Clustering: Motivation

- Clustering (**ML + Data Analysis**)

  Explore the data, Reveal existing structure, group similar points to one another, etc

- Clustering (**Operations Research**)

Clustering: Motivation

- Clustering (**ML + Data Analysis**)

  Explore the data, Reveal existing structure, group similar points to one another, etc

- Clustering (**Operations Research**) ‘

  Allocating a collection of facilities or fire stations to serve a collection of users

Clustering: Motivation

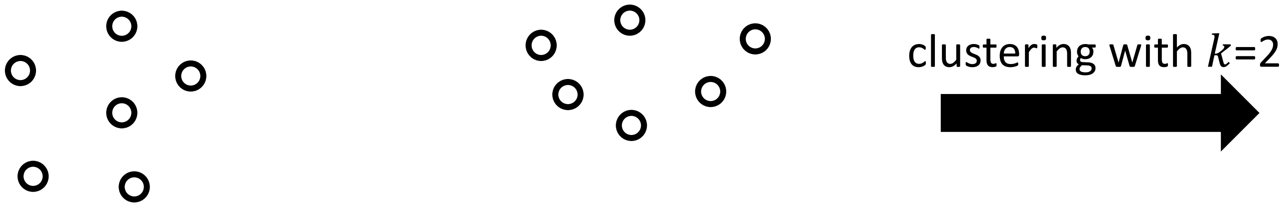
- Clustering (**ML + Data Analysis**)

  Explore the data, Reveal existing structure, group similar points to one another, etc

- Clustering (**Operations Research**) ‘

  Allocating a collection of facilities or fire stations for a collection of users

Set of Possible Locations:



Clustering: Motivation

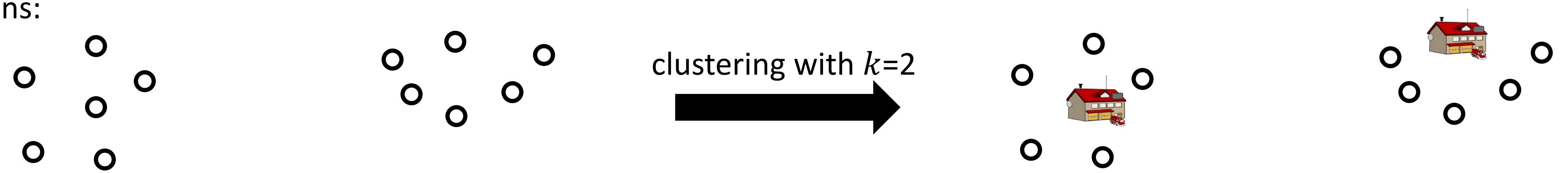
➢Clustering (**ML + Data Analysis**)

Explore the data, Reveal existing structure, group similar points to one another, etc

➢Clustering (**Operations Research**) ‘

Allocating a collection of facilities or fire stations for a collection of users

Set of Possible Locations:



Types of Clustering

Types of Clustering

- Center-Based Clustering
- Spectral Clustering
- Correlation Clustering
- Hierarchical Clustering

Center-Based Clustering

- the cluster is decided by choosing k ***centers*** → each point is then assigned to ***its closest center***

- Includes k-means, k-median, and k-center clustering

Formalizing Center-Based Clustering

- *Input:*
  - <u>Set of points</u>: $\mathcal{C}$
  - <u>Distance between points</u>: $\forall i, j \in \mathcal{C}$ we have $d(i, j)$ (which is a ***Metric***)
  - <u>Number of Clusters:</u> k

Formalizing Center-Based Clustering

- *Input:*
  - Set of points: $\mathcal{C}$
  - Distance between points: $\forall i, j \in \mathcal{C}$ we have $d(i, j)$ (which is a ***Metric***)
  - Number of Clusters: k
- *Output:*
  - Set of centers: S ($|S| \leq k$)
  - Assignment Function: $\varphi: \mathcal{C} \to S$ (assigning points to centers)

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in C} d(j, \varphi(j))$

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in C} d(j, \varphi(j))$

  Minimizes the maximum distance (sensitive to outliers)

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in \mathcal{C}} d(j, \varphi(j))$

Minimizes the maximum distance (sensitive to outliers)

- k-median: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d(j, \varphi(j))$

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the maximum distance (sensitive to outliers)

- k-median: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the sum of the distances (more noise-tolerant)

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the maximum distance (sensitive to outliers)

- k-median: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the sum of the distances (more noise-tolerant)

- k-means: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d^2(j, \varphi(j))$

Formalizing Center-Based Clustering

Objective Functions:

- *k*-center: $\min_{S,\varphi} \max_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the maximum distance (sensitive to outliers)

- *k*-median: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the sum of the distances (more noise-tolerant)

- *k*-means: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d^2(j, \varphi(j))$

  Minimizes the sum of the square of the distances (more weight on outliers because of squaring)

Formalizing Center-Based Clustering

Objective Functions:

- k-center: $\min_{S,\varphi} \max_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the maximum distance (sensitive to outliers)

- k-median: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d(j, \varphi(j))$

  Minimizes the sum of the distances (more noise-tolerant)

- k-means: $\min_{S,\varphi} \sum_{j \in \mathcal{C}} d^2(j, \varphi(j))$

  Minimizes the sum of the square of the distances (more weight on outliers because of squaring)

- In General: $\min_{S,\varphi} \left\| \left[d\big(1, \varphi(1)\big),\ d\big(2, \varphi(2)\big), \ldots ,\ d\big(n, \varphi(n)\big) \right] \right\|_p$

  $p = \infty \rightarrow k$-center, $p = 1 \rightarrow k$-median, $p = 2 \rightarrow k$-means

Spectral Clustering

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j\colon w_{ij} \geq 0$, w_{ij} is the similarity between i and j

Spectral Clustering

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j: w_{ij} \geq 0$, w_{ij} is the similarity between i and j

- Cluster (partition) the graph so that the edges *between points in **different** clusters* have low weight

- From the above, we have a graph cut problem

Spectral Clustering: Formal Definition

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j: w_{ij} \geq 0$, w_{ij} is the similarity between i and j

- $A, B \subset V, cut(A, B) = \sum_{i \in A, j \in B} w_{ij}$

- <u>Objective Functions</u>:
  - Given graph $G = (V, E)$ and number of clusters k → Partition V into $C_1, \ldots, C_k$

    - min $RatioCut(C_1, \ldots, C_k) = \sum_{i=1}^{k} \frac{cut(C_i, V \backslash C_i)}{|C_i|}$
    - min $NormalizedCut(C_1, \ldots, C_k) = \sum_{i=1}^{k} \frac{cut(C_i, V \backslash C_i)}{\sum_{j \in C_i} d_j}$

Correlation Clustering

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j: w_{ij}^{+} \geq 0, w_{ij}^{-} \geq 0,$

  -w_{ij}^{+} is the degree to which i and j are **similar**

  -w_{ij}^{-} is the degree to which i and j are **different**

Correlation Clustering

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j: w_{ij}^{+} \geq 0, w_{ij}^{-} \geq 0,$

  -w_{ij}^{+} is the degree to which i and j are **similar**

  -w_{ij}^{-} is the degree to which i and j are **different**
- Cluster (partition) the graph so that you get:

  -*large* w_{ij}^{+} *edges* within a cluster

  -*large* w_{ij}^{-} *edges* between *different clusters*

Correlation Clustering

- Points are vertices in a graph $G = (V, E)$
- $\forall i, j: w_{ij}^{+} \geq 0, w_{ij}^{-} \geq 0,$

  -w_{ij}^{+} is the degree to which i and j are **similar**

  -w_{ij}^{-} is the degree to which i and j are **different**
- Cluster (partition) the graph so that you get

  -*low weight edges* between *different clusters*
  -*high weight edges* within a cluster

Number of clusters k does NOT need to be given in correlation clustering

- Objective Function:

  Given graph $G = (V, E)$ → Partition V into $C_1, \ldots, C_k$

  $\max \sum_{i,j:same\ cluster} w_{ij}^{+} + \sum_{i,j:\ different\ clusters} w_{ij}^{-}$

Hierarchical Clustering

- Like correlation clustering, you don’t need to set k (the number of clusters)
- The output groups the points in a tree, giving groupings at different levels
- **agglomerative clustering** (common traditional method):

  -bottom-up approach, group similar points together forming a cluster,
  then group similar clusters and so on.

Hierarchical Clustering

- Like correlation clustering, you don't need to set k (the number of clusters)

- The output groups the points in a tree, grouping points at different levels

- **agglomerative clustering** (common traditional method):

  -*bottom-up approach*: group similar points together forming a cluster, then group similar clusters and so on.

Hierarchical Clustering

- agglomerative clustering has is NOT optimizing an objective function

  → Makes it hard to know the quality

Hierarchical Clustering

- agglomerative clustering has is NOT optimizing an objective function

  → Makes it hard to know the quality

- [Dasgupta, 2016] defines a cost function for hierarchical clustering:

  -Given $G = (V, E)$ with w_{ij} specifiying the similarity between i *and* j

  <u>Objective Function</u>:

  $$\min \sum_{i,j} w_{ij} \times (\#\text{of descendents of lowest commen ancestor of } i \text{ and } j)$$

  -the objective places higher penalty when separating points higher in the tree

- Further objectives for hierarchical clustering were also introduced [Moseley & Wang, 2017 ; Cohen-Addad et al, 2018].

Algorithmic Fairness

- Much of decision making is done at least partly using algorithms.

Algorithmic Fairness

- Much of decision making is done at least partly using algorithms.

  Examples: ML for recidivism prediction, loan approval

Algorithmic Fairness

- Much of decision making is done at least partly using algorithms.

    Examples: ML for recidivism prediction, loan approval

- Documented cases of algorithmic bias [O’Neil 2016; Kearns & Roth 2019].

- Substantial progress and interest in algorithmic fairness.

Some Considerations for Fairness in Clustering

- For a point i, its **distance from the center** $d(i, \phi(i))$:

  -closer to the center →more represented by the center (**ML**)

  -closer to the center →less travel distance (**OR**)

  → points want to be closer the center

- How does a fairness guarantee over the distance look like? How can we achieve that?

Some Considerations for Fairness in Clustering

- Clustering partitions the set of points $\mathcal{C}$ into clusters $C_1, C_2, \ldots, C_k$

  -different clusters will be processed differently, enjoy different outcomes, etc

  -suppose one demographic is under-represented in a cluster or over-represented in another.

  -suppose clustering assigns points which are not faraway from one another to different clusters.

- How do avoid the above situations and guarantee fairness?