Unsupervised Learning: K-Means & Agglomerative Clustering

These slides are partially based on slides assembled by Eric Eaton, with grateful acknowledgement of the many others who made their course materials freely available online.
## Types of Learning

<table>
<thead>
<tr>
<th></th>
<th>from input $x$, output:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>unsupervised</strong></td>
<td>summary $z$</td>
</tr>
<tr>
<td><strong>supervised</strong></td>
<td>prediction $y$</td>
</tr>
<tr>
<td><strong>reinforcement</strong></td>
<td>action $a$ to maximize reward $r$</td>
</tr>
</tbody>
</table>
Types of Learning

Unsupervised Learning
- What happened?
- Descriptive Analytics

Supervised Learning
- Predictive Analytics
- Insight
- Hindsight
- Information

How can we make it happen?
- Prescriptive Analytics

Value

Difficulty

From Gartner, Recht
Unsupervised Learning

• Supervised learning used labeled data pairs \((x, y)\) to learn a function \(f: X \rightarrow Y\)
  – But, what if we don’t have labels?

• No labels = **unsupervised learning**
Clustering:
group together similar points and represent them with a single token

Key Challenges:
1) What makes two data points similar?
2) How do we compute an overall grouping from pairwise similarities?
How might we cluster?

• K-means
  – Iteratively re-assign points to the nearest cluster center

• Agglomerative clustering
  – Start with each point as its own cluster and iteratively merge the closest clusters
K-Means Clustering
Clustering Data
K-Means Clustering

K-Means \((k, X)\)

- Randomly choose \(k\) cluster center locations (centroids)
- Loop until convergence
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster
K-Means Clustering

K-Means \(( k, X )\)

- Randomly choose \( k \) cluster center locations (centroids)
- Loop until convergence
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster
K-Means Clustering

K-Means \((k, X)\)
- Randomly choose \(k\) cluster center locations (centroids)
- Loop until convergence
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster
K-Means Objective Function

• K-means finds a local optimum of the following objective function:

$$\arg \min_{\mathcal{S}} \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2_2$$

where $\mathcal{S} = \{S_1, \ldots, S_k\}$ is a partitioning over $X = \{x_1, \ldots, x_n\}$ s.t. $X = \bigcup_{i=1}^{k} S_i$ and $\mu_i = \text{mean}(S_i)$
K-means Demo
K-Means pros and cons

- Pros
  - Finds cluster centers that minimize conditional variance (good representation of data)
  - Easy to implement

- Cons
  - Need to choose K
  - Sensitive to outliers
  - Prone to local minima
  - All clusters have the same parameters (e.g., distance measure is non-adaptive)
K-means Demo
K-Means: initialization

• Very sensitive to the initial points
  – Do many runs of K-Means, each with different initial centroids
  – Seed the centroids using a better method than randomly choosing the centroids
    • e.g., Farthest-first sampling

• Must manually choose \( k \)
  – Learn the optimal \( k \) for the clustering
    • Note that this requires a performance measure
K-medoids

• Just like K-means except
  – Represent the cluster with one of its members, rather than the mean of its members
  – Choose the member (data point) that minimizes cluster dissimilarity

• Applicable when a mean is not meaningful
  – E.g., clustering values of hue
How might we cluster?

• **K-means**
  – Iteratively re-assign points to the nearest cluster center

• **Agglomerative clustering**
  – Start with each point as its own cluster and iteratively merge the closest clusters
Agglomerative clustering

1. Say “Every point is its own cluster”
Agglomerative clustering

1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
Agglomerative clustering

1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster
Agglomerative clustering

1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster
4. Repeat

Copyright © 2001, 2004, Andrew W. Moore
Agglomerative clustering

1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster
4. Repeat
Agglomerative clustering

How to define cluster similarity?
- Average distance between points, maximum distance, minimum distance
- Distance between means or medoids

How many clusters?
- Clustering creates a dendrogram (a tree)
- Threshold based on max number of clusters or based on distance between merges
Agglomerative clustering demo