Clustering
What is Clustering?

- Finding groups of objects such that objects in a group will be similar to one another and different from the objects in other groups
- A type of unsupervised learning and can be viewed as unsupervised segmentation (sometimes called classification by statisticians, sorting by psychologists, and segmentation by people in marketing)

Intra-cluster distances are minimized

Inter-cluster distances are maximized
What is a natural grouping among these objects?

Clustering is subjective

Simpson's Family
School Employees
Females
Males
Similarity is Subjective
Intuitions behind desirable distance measure properties

\[ D(A, B) = D(B, A) \quad \text{Symmetry} \]
Otherwise you could claim “Alex looks like Bob, but Bob looks nothing like Alex.”

\[ D(A, A) = 0 \quad \text{Constancy of Self-Similarity} \]
Otherwise you could claim “Alex looks more like Bob, than Bob does.”

\[ D(A, B) = 0 \text{ if and only if } A = B \quad \text{Positivity (Separation)} \]
Otherwise there are objects in your world that are different, but you cannot tell apart.

\[ D(A, B) \leq D(A, C) + D(B, C) \quad \text{Triangular Inequality} \]
Otherwise you could claim “Alex is very like Bob, and Alex is very like Carl, but Bob is very unlike Carl.”
Applications of Cluster Analysis

- **Understanding**
  - Group related documents for browsing
    - Business news articles about a particular stock
  - Group genes and proteins with similar functionality
  - Group stocks with similar price fluctuations
  - Group customers with similar buying habits
  - Group Whisky with similar flavors

- **Summarization**
  - Reduce the size of large data sets
  - Reduce the number of attributes
    - Cluster using a group of attributes and replace these attributes with one attribute that is the cluster label
Clustering Precipitation in Australia
Notion of a Cluster can be Ambiguous

So tell me how many clusters do you see?

How many clusters?

Two Clusters

Four Clusters

Six Clusters
Types of Clusterings

- A **clustering** is a set of clusters

- **Partitional Clustering**
  - A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset

- **Hierarchical clustering**
  - A set of nested clusters organized as a hierarchical tree
Partitional Clustering

Original Points

A Partitional Clustering
Hierarchical Clustering

Traditional Hierarchical Clustering

Traditional Dendrogram

Simpsonian Dendrogram
Other Distinctions Between Clusterings

- **Exclusive versus non-exclusive**
  - In non-exclusive clusterings points may belong to multiple clusters
  - Can represent multiple classes or ‘border’ points

- **Fuzzy versus non-fuzzy**
  - In fuzzy clustering, a point belongs to every cluster with some weight between 0 and 1
  - Weights must sum to 1
  - Probabilistic clustering has similar characteristics

- **Partial versus complete**
  - In some cases, we only want to cluster some of the data
Types of Clusters

- Well-separated clusters
- Center-based clusters (our main emphasis)
- Contiguous clusters
- Density-based clusters
- Described by an Objective Function
Types of Clusters: Well-Separated

- **Well-Separated Clusters:**
  - A clustering where any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.

3 well-separated clusters
Types of Clusters: Center-Based

- **Center-based**
  - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster.
  - The center of a cluster is often a centroid, the average of all the points in the cluster (assuming numerical attributes), or a medoid, the most “representative” point of a cluster (used if there are categorical features).

4 center-based clusters
Types of Clusters: Contiguity-Based

- Contiguous Cluster (Nearest neighbor or Transitive)
  - A clustering such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.
Types of Clusters: Density-Based

- Density-based
  - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
  - Used when the clusters are irregular or intertwined, and when noise and outliers are present.

6 density-based clusters
Clusters Defined by an Objective Function

- Finds clusters that minimize or maximize an objective function.
- Enumerate all possible ways of dividing the points into clusters and evaluate the `goodness' of each potential set of clusters by using the given objective function. (NP Hard)
- Example: Sum of squares of distances to cluster center
Clustering Algorithms

- K-means and its variants
- Hierarchical clustering
- Density-based clustering
K-means Clustering

- Partitional clustering approach
- Each cluster is associated with a **centroid** (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, $K$, must be specified
- The basic algorithm is very simple

1: Select $K$ points as the initial centroids.
2: repeat
3: Form $K$ clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don’t change
1. Ask user how many clusters they’d like. *(e.g. k=3)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it’s closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!
K-means Clustering: Step 1
K-means Clustering
K-means Clustering
K-means Clustering

The graph shows data points clustered into three distinct groups, represented by different colors and symbols. The clusters are labeled as $k_1$, $k_2$, and $k_3$. Each cluster has its centroid marked with a black circle. The x-axis represents expression in condition 1, while the y-axis represents expression in condition 2.
K-means Clustering – Details

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster
- ‘Closeness’ is measured by Euclidean distance, correlation, etc.
- K-means will converge for common similarity measures mentioned above.
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to ‘Until relatively few points change clusters’
Evaluating K-means Clusters

- Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.
  - We can show that to minimize SSE the best update strategy is to use the center of the cluster.
  - Given two clusters, we can choose the one with the smallest error
  - One easy way to reduce SSE is to increase K, the number of clusters
    - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K
Two different K-means Clusterings

Original Points

Optimal Clustering

Sub-optimal Clustering
Importance of Choosing Initial Centroids

If you happen to choose good initial centroids, then you will get this after 6 iterations
Importance of Choosing Initial Centroids

Good clustering
Importance of Choosing Initial Centroids ...
Starting with two initial centroids in one cluster of each pair of clusters
Starting with some pairs of clusters having three initial centroids, while other have only one.
Pre-processing and Post-processing

- **Pre-processing**
  - Normalize the data
  - Eliminate outliers

- **Post-processing**
  - Eliminate small clusters that may represent outliers
  - Split ‘loose’ clusters, i.e., clusters with relatively high SSE
  - Merge clusters that are ‘close’ and that have relatively low SSE
Limitations of K-means

- K-means has problems when clusters are of differing
  - Sizes (biased toward the larger clusters)
  - Densities
  - Non-globular shapes

- K-means has problems when the data contains outliers.
Limitations of K-means: Differing Sizes

Original Points

K-means (3 Clusters)
Limitations of K-means: Differing Density

Original Points

K-means (3 Clusters)
Limitations of K-means: Non-globular Shapes

Original Points

K-means (2 Clusters)
Overcoming K-means Limitations

One solution is to use many clusters. Find parts of clusters, but need to put together.
Overcoming K-means Limitations

Original Points  K-means Clusters
Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
  - A tree-like diagram that records the sequences of merges or splits
Hierarchal clustering can sometimes show patterns that are meaningless or spurious

• For example, in this clustering, the tight grouping of Australia, Anguilla, St. Helena etc is meaningful, since all these countries are former UK colonies.

• However the tight grouping of Niger and India is completely spurious, there is no connection between the two.
We can look at the dendrogram to determine the “correct” number of clusters. In this case, the two highly separated subtrees are highly suggestive of two clusters. (Things are rarely this clear cut, unfortunately)
One potential use of a dendrogram is to detect outliers.

The single isolated branch is suggestive of a data point that is very different to all others.
Hierarchical Clustering

- Build a tree-based hierarchical taxonomy \((dendrogram)\) from a set of unlabeled examples.

- Recursive application of a standard clustering algorithm can produce a hierarchical clustering.
Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by ‘cutting’ the dendogram at the proper level

- They may correspond to meaningful taxonomies
  - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, ...)
Hierarchical Clustering

- Two main types of hierarchical clustering
  - Agglomerative:
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
  - Divisive:
    - Start with one, all-inclusive cluster
    - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- Agglomerative is most common
Starting Situation

- Start with clusters of individual points
Intermediate Situation

- After some merging steps, we have some clusters
Intermediate Situation

- We want to merge the two closest clusters (C2 and C5)
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward’s Method uses squared error

Proximity Matrix
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward’s Method uses squared error

Proximity Matrix
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward’s Method uses squared error
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward’s Method uses squared error

Proximity Matrix

<table>
<thead>
<tr>
<th></th>
<th>p1</th>
<th>p2</th>
<th>p3</th>
<th>p4</th>
<th>p5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids

Proximity Matrix

|    | p1 | p2 | p3 | p4 | p5 | ...
|----|----|----|----|----|----|-----
| p1 |    |    |    |    |    |     
| p2 |    |    |    |    |    |     
| p3 |    |    |    |    |    |     
| p4 |    |    |    |    |    |     
| p5 |    |    |    |    |    |     

Proximity Matrix
Hierarchical Clustering: MIN

Nested Clusters

Dendrogram
Hierarchical Clustering: MAX

Nested Clusters

Dendrogram
Hierarchical Clustering: Problems and Limitations

- Once a decision is made to combine two clusters, it cannot be undone.
- No objective function is directly minimized.
- Different schemes have problems with one or more of the following:
  - Sensitivity to noise and outliers
  - Difficulty handling different sized clusters and convex shapes
  - Breaking large clusters
DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a core point if it has more than a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
  - A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point
  - A noise point is any point that is not a core point or a border point.
DBSCAN: Core, Border, and Noise Points
DBSCAN: Core, Border and Noise Points

Original Points

Point types: core, border and noise

Eps = 10, MinPts = 4
When DBSCAN Works Well

- Resistant to Noise
- Can handle clusters of different shapes and sizes
When DBSCAN Does NOT Work Well

• Varying densities

Original Points

(MinPts=4, Eps=9.75).

(MinPts=4, Eps=9.92)
Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is:
  - Accuracy, precision, recall

- For cluster analysis, the analogous question is how to evaluate the “goodness” of the resulting clusters?

- But “clusters are in the eye of the beholder”!

- Then why do we want to evaluate them?
  - To avoid finding patterns in noise
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To compare two clusters
Clusters found in Random Data

Random Points

K-means

DBSCAN
Internal Measures: SSE

- Clusters in more complicated figures aren’t well separated
- Internal Index: Used to measure the goodness of a clustering structure without respect to external information
  - SSE
- SSE is good for comparing two clusterings or two clusters (average SSE).
- Can also be used to estimate the number of clusters
Internal Measures: SSE

- SSE curve for a more complicated data set

SSE of clusters found using K-means
Final Comment on Cluster Validity

“The validation of clustering structures is the most difficult and frustrating part of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage.”

_Algorithms for Clustering Data_, Jain and Dubes
Understanding the Clustering

- How might you understand what the different clusters represent?
  - The names/identifiers *may* indicate a pattern if they are well known
  - You can represent the clusters by a prototype (centroid) and this can be insightful
  - You can provide statistics (e.g., mean) to describe the features of the objects in each cluster
    - In a class many years ago we had a cluster that you could name “Educated Single Women” based on the feature averages.

- But it is not always easy to truly understand the clusters
Using Supervised Learning to Generate Cluster Descriptions

- Supervised learning can help understand the clusters
  - Create a class variable that is assigned the cluster identifier
  - Then apply supervised learning using an interpretable classifier (rule-based or decision tree)
  - The classifier can help describe the cluster
Descriptive Data Mining

- Clustering is a descriptive task
- The problem/task that you are addressing may not be nearly as well defined as with a prediction task, where you must formulate a precise learning problem
- Even if your task is to do market segmentation for a marketing campaign, it is not clear what to do with the clustering
- You tend to have more work to do after the data mining phase
Clustering with WEKA

- Run Weka Explorer and open the vote dataset that comes with WEKA
  - This is the “Congressional Voting Records” data set from the UCI Machine Learning Repository
  - Votes for each of the U.S. House of Representatives Congressmen on 16 key votes
  - The class is “democrat” or “republican”

- Go to cluster tab and choose “SimpleKMeans”
  - Use default parameters, including N=2 clusters
### Clustering Output for “Vote”

**kMeans**

- **Number of iterations:** 3
- **Within cluster sum of squared errors:** 1510.0
- **Missing values globally replaced with mean/mode**

**Cluster centroids:**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Cluster #</th>
<th>Full Data (435)</th>
<th>0 (214)</th>
<th>1 (221)</th>
</tr>
</thead>
<tbody>
<tr>
<td>handicapped-infants</td>
<td></td>
<td>n n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>water-project-cost-sharing</td>
<td></td>
<td>y y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>adoption-of-the-budget-resolution</td>
<td></td>
<td>y n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>physician-fee-freeze</td>
<td></td>
<td>n y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>el-salvador-aid</td>
<td></td>
<td>y y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>religious-groups-in-schools</td>
<td></td>
<td>y y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>anti-satellite-test-ban</td>
<td></td>
<td>y n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>aid-to-nicaraguan-contras</td>
<td></td>
<td>y n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mx-missile</td>
<td></td>
<td>y n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>immigration</td>
<td></td>
<td>y y y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>synfuels-corporation-cutback</td>
<td></td>
<td>n n n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>education-spending</td>
<td></td>
<td>n y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>superfund-right-to-sue</td>
<td></td>
<td>y y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>crime</td>
<td></td>
<td>y y n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>duty-free-exports</td>
<td></td>
<td>n n y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>export-admin-act-south-africa</td>
<td></td>
<td>y y y</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Class**

<table>
<thead>
<tr>
<th></th>
<th>dem</th>
<th>rep</th>
<th>dem</th>
</tr>
</thead>
</table>

When reran with N=4

clusters appeared to split democrats into 3 groups

- **Cluster 1:** 40% (rep)
- **Cluster 2:** 12% (dem)
- **Cluster 3:** 13% (dem)
- **Cluster 4:** 35% (dem)