Clustering

What is clustering

- Unsupervised classification of patterns into groups.
- Grouping of similar patterns
- Partitioning of patterns into groups such that patterns in a particular group are more similar than patterns in another group.
- Groups contain patterns that are highly similar in some way and different from patterns in another group.

Components

- Pattern presentation.
- Similarity measure.
- Clustering or grouping of patterns.
- Assessment of output.
Pattern presentation

- Number of clusters?
- Number of patterns? (representative)
- Features?
  - Used to determines cluster membership.
  - Curse of dimensionality.
  - Data types?
  - Use all features or only some?
    * Feature selection: Choose features
    * Feature extraction: Create new features.

Similarity measures

- Used to determine the similarity of patterns.
- Euclidean distance:

\[
Euclidean(X_i, X_j) = \sqrt{\sum_{k=1}^{n} (X_{ik} - X_{jk})^2}
\]

- Manhattan distance:

\[
Manhattan(X_i, X_j) = \sum_{k=1}^{n} |X_{ik} - X_{jk}|
\]
• Cosine distance metric:

\[ \text{Cosine}_{\text{dist}} = \frac{\sum_{k=1}^{n} x_{ik} x_{jk}}{||x_i|| ||x_j||} \]

- Difference in angles between 2 vectors, not magnitude.

• Only works for numbers.
• What about larger scaled features?
• What about other data types?
• Hamming distance?

## Clustering Methods

• Hierarchical:
  - Tree.
  - Splitting: divisive, top down.
  - Merging: agglomerative, bottom up.
  - Computationally expensive.
  - Static (patterns can’t move to other clusters)
  - Overlapping clusters...

• Partitioned:
  - Number of clusters are (usually) specified beforehand.
  - Minimize heuristic.
  - Inter and intra cluster differences.
  - K-means and K-medoids:
Generic Clustering Algorithm

1. Randomly initialize K centroids, $m_k$

2. while not stopping condition
   a) for each pattern $x_i$
      i. compute $\mu(m_k|x_i)$ for each centroid $m_k$
      ii. compute weight $w(x_i)$
   b) for each $m_k$
      i. Recalculate centroid

• What is $\mu(m_k|x_i)$? It is the membership function.
• Does does $x_i$ belong to $m_k$
• The following must hold:
  1. $\mu(m_k|x_i) \geq 0, i = 1,..,n$ and $k = 1,..,K$
  2. $\sum_{k=1}^{K} \mu(m_k|x_i) = 1, i = 1,..,n$

K-means

• Find an “average" (representative) between patterns as the cluster centroid.
• Patterns are assigned to the closest centroid.
• Minimizes intra cluster differences (distance between furthest patterns in cluster).
• Crisp membership, pattern belongs ONLY to the closest centroid.
Algorithm

Initialize k means (centroids)
Repeat
  1. Assign patterns to means using a distance metric
  2. Update means to become the ’’average’’ of all patterns associated with it
Until stopping condition reached

• Calculating new mean?
• Membership function:

\[
\mu(m_k|x_i) = \begin{cases} 
1 & \text{if } E(m_k, x_i) = \min_{l=1,\ldots,K} \{E(m_l, x_i)\} \\
0 & \text{otherwise}
\end{cases}
\]

where \(E\) is the Euclidean distance.

• Minimize:

\[
E_{k\text{means}} = \sum_{k-1}^{K} \sum_{\forall x_i \in C_k} \text{Euclidean}(x_i, m_k)^2
\]

• Stopping condition?
• Sets don’t change anymore. Means don’t change anymore?
• Depends on initialization of means, could get stuck in local minima.
K-medoids

- Centroids are patterns from data set.
- Try to find centroids (aka medoids) that are most centrally located in data set.

**Algorithm:**

Choose k medoids from your data set

Repeat

  Assign each pattern to closest medoid.
  For all medoids m
  
  Swap m with each pattern in the data set.
  Compute total cost of this configuration
  Select configuration with lowest cost.

Until stopping condition not reached.
• What is cost?
  – The sum of the distances for each medoid from the patterns associated with it.

• And total cost?
  – The sum of the costs.

• Stopping condition(s)?

**Incremental clustering**

• Number of clusters is determined by the algorithm.

**Pseudo code:**

Assign first pattern to the only cluster
For all remaining patterns.
   Assign pattern to an existing cluster.
   OR create a new cluster to assign the pattern to.

**Cluster quality**

• Want clusters to be homogeneous. More homogeneous requires more clusters.

• Inter and intra cluster distance.
- Intra: average of all distances between each pattern and the cluster centroid.
- Inter: Minimum distance between centroids.

- Dunn index:
  - Maximize inter and minimizes intra.
  - The configuration which has the higher Dunn index is better.

\[
\text{index} = \min_{i=1,...,K}\{\min_{j=1,...,K,i \neq j}(\frac{\text{distance}(C_i, C_j)}{\max_{m=1,...,K}d(C_m)})\}
\]

with \( \text{distance} \) the distance of the closest two patterns in clusters \( C_i \) and \( C_j \) and \( d \) the distance between the two furthest patterns in cluster \( C_m \)

**Quantization Error:**

\[
Q_{\text{error}} = \frac{\sum_{k=1}^{K}[\sum_{\forall x_i \in C_k} \text{dist}(x_i, m_k)]/|C_k|}{K}
\]

With:
- \( K \) = number of clusters.
- \( C_k \) = the \( k^{th} \) cluster.
- \( m_k \) = the centroid for \( C_k \).
- \( x_i \) is a pattern from the data set, member of \( C_k \).
Additional stopping conditions

- Centroids don’t change anymore.
- Quality metric is satisfactory.
- Maximum number of iterations reached.
- Anything else?

What is clustering used for?

- Document clustering.
- Classification.
- Pattern recognition (speech).
- Image analysis (characters, faces, medical).
- Profiling (Criminals).
- Identifying outliers.
- Data set reduction (for large data sets): Can do stratified sampling.
- Exploratory data analysis?
- Any other uses?
Homework:

- Research the Fuzzy C means algorithm.
- Find out about using Quantization error as a metric for clustering quality.

Quantization Error:

\[ Q_{error} = \frac{\sum_{k=1}^{K} [\sum_{\forall x_i \in C_k} dist(x_i, m_k)]/|C_k|}{K} \]

With:

- \( K \) = number of clusters.
- \( C_k \) = the \( k^{th} \) cluster.
- \( m_k \) = the centroid for \( C_k \)
- \( x_i \) is a pattern from the data set.