Chapter 15: Clustering Algorithms
Hierarchical clustering
- Single-linkage algorithm (minimum method)
- Complete-linkage algorithm (maximum method)
- Average linkage algorithm
- Minimum-variance method (Ward’s method)

Partitional clustering
- K-means algorithm
- Fuzzy c-means algorithm
- Isodata algorithm
A cluster is a collection of data points with similar properties.
Chapter 15: Clustering

Hierarchical Clustering

Agglomerative clustering (bottom up)
1. Begin with n clusters; each containing one sample
2. Merge the most similar two clusters into one.
3. Repeat the previous step until done

Divisive clustering (top down)
Hierarchical Clustering

Single-linkage algorithm (minimum method)

\[ D_{\text{min}}(C_i, C_j) = \min_{a \in C_i, b \in C_j} d(a, b) \]

Complete-linkage algorithm (maximum method)

\[ D_{\text{max}}(C_i, C_j) = \max_{a \in C_i, b \in C_j} d(a, b) \]

Average-linkage algorithm (average method)

\[ D_{\text{ave}}(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{a \in C_i, b \in C_j} d(a, b) \]
Hierarchical Clustering

Ward’s method (minimum-variance method)

\[ D_{\text{Ward}}(C_i, C_j) = \sum_{j=1}^{d} \sigma_j^2 = \sum_{j=1}^{d} \sum_{i=1}^{m} (X_{ij} - \mu_j)^2 \]

d: the number of features
m: the number of samples in \( C_i \) and \( C_j \)
Chapter 15: Clustering

Partitional Clustering: K-means

K-means algorithm:
1. Begin with K cluster centers
2. For each sample, find the cluster center nearest to it. Put the sample in the cluster represented by the just-found cluster center.
3. If no samples changed clusters, stop.
4. Recompute cluster centers of altered clusters and go back to step 2.

Properties:
- The number of cluster K must be given in advance.
- The goal is to minimize the square error, but it could end up in a local minimum.
Chapter 15: Clustering

Partitional Clustering: Isodata

Similar to K-means with some enhancements:

- Clusters with too few elements are discarded.
- Clusters are merged if the number of clusters grows too large or if clusters are too close together.
- A cluster is split if the number of clusters is too few or if the cluster contains very dissimilar samples.

Properties:

- The number of clusters K is not given exactly in advance.
- The algorithm may requires extensive computations.
Chapter 15: Clustering

Fuzzy C-means Clustering

Properties:

- Similar to K-means, but each sample can belong to various clusters with degrees from 0 to 1.
- For a given sample, the degrees of membership to all clusters sum to 1.
- Computationally more extensive than k-means, but usually reach a better result.
Fuzzy C-Means Clustering

In fuzzy clustering, a point can belong to many clusters with various degrees.
Mountain Clustering

Mountain clustering is based on using a special function to determine cluster means

- A grid is constructed in data space. The grid points are used as possible cluster centers.
- A mountain function is defined. We can consider this function as an estimate of the density of data points, i.e., if many points are at a specific location, the function takes a large value.
- Cluster centers are determined by destroying the mountain function.
Mountain Clustering

Typical mountain function

$$m(v) = \sum_{i=1}^{N} \exp \left( -\frac{\|v - x_i\|^2}{2\sigma^2} \right)$$

Peaks removed

$$m_{\text{new}}(v) = m(v) - m(c_1) \exp \left( -\frac{\|v - c_1\|^2}{2\beta^2} \right)$$
a) 2-D data, b) mountain function based on $\sigma = 0.02$, c) $\sigma = 0.1$ and d) $\sigma = 0.2$. 

Chapter 15: Clustering
Mountain Clustering: Destruction

a) Original function, b) first peak removed, c) second peak removed and d) third peak removed.
Subtractive Clustering

Subtractive clustering is based on a similar idea as mountain clustering

- Subtractive clustering is fast. The computational burden does not increase with the number of dimensions as in mountain clustering.
- The computational burden increases with the number of data points.
- Data points are used as possible cluster centers.
- Subtractive clustering is used in genfis2.m in the Fuzzy Logic Toolbox (FLT) of Matlab
Subtractive Clustering

Typical construction function

Peaks removed

\[ D_i = \sum_{j=1}^{n} \exp\left( -\frac{\|x_i - x_j\|^2}{(r_a/2)^2} \right) \]

\[ D_i = D_i - D_{c_1} \exp\left( -\frac{\|x_i - x_{c_1}\|^2}{(r_b/2)^2} \right) \]
### Chapter 15: Clustering

#### From Data Sets to FIS

**Flow chart: From data sets to FIS**

- **Training data**
- **FLT**
- **GUI tools**
- **genfis1.m**
- **genfis2.m**
- **Initial FIS**
- **anfis.m**
- **Final FIS**

**Steps:**
1. Training data
2. FLT GUI tools
3. genfis1.m
4. genfis2.m
5. Initial FIS
6. anfis.m
7. Final FIS

**Inputs:**
- Training data
- Checking data
Chapter 15: Clustering

FIS data structure in FLT

FIS file and FIS matrix

FIS file (on disk)

writefis

FIS matrix (in workspace)

readfis

FIS Data Structure

[System]
Name: mam21
Type: mamdani

[Input 1]
Name: Position
Range: [-10 10]
MF #: 2

[Input 2]
Name: Velocity
Range: [-50 50]
MF #: 2

[Output 1]
Name: Force
Range: [-10 10]
MF #: 4

[Rules]
Rule list
Rule weights
Rule types

MF 1
Label: Small
Type: Gaussian
Params: [5 -10]

MF 2
Label: Large
Type: Triangle
Params: [-5 10 20]

Input 1
Name: Position
Range: [-10 10]
MF #: 2

MF 1
Label: Small
Type: Gbell
Params: [5 2 -40]

MF 2
Label: Large
Type: S
Params: [-50 0 50]

Input 2
Name: Velocity
Range: [-50 50]
MF #: 2

MF 1
Label: Neg. Big
Type: Z
Params: [-10 -5 0]

MF 2
Label: Neg. Small
Type: Gbell
Params: [5 2 -3]

MF 3
Label: Pos. Small
Type: Gbell
Params: [5 2 3]

MF 4
Label: Pos. Big
Type: S
Params: [0 5 10]

Output 1
Name: Force
Range: [-10 10]
MF #: 4

readfis

writefis