Cluster Analysis

Dr. Dewan Md. Farid

Computational Modeling Lab (CoMo), Department of Computer Science
Vrije Universiteit Brussel, Belgium

March 14, 2016

What is Clustering?

Clustering is the process of grouping a set of instances (data points or examples or vectors) into clusters (subsets or groups) so that instances within a cluster have high similarity in comparison to one another, but are very dissimilar to instances in other clusters.
Clustering may be found under different names in different contexts, such as:

- Unsupervised Learning
- Data Segmentation
- Automatic Classification
- Learning by Observation

What is Clustering? (con.)

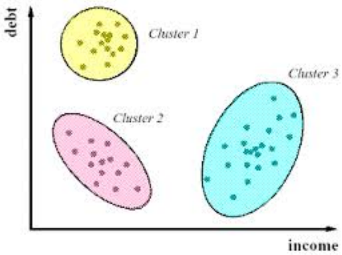

Figure: Clustering of a set of instances.

Similarities and dissimilarities of instances are based on the predefined features of the data. The most similar instances are grouped into a single cluster.

Area of Applications

Clustering has been widely used in many real world applications, such as:

- Human genetic clustering
- Medical imaging clustering
- Market research
- Field robotics
- Crime analysis
- Pattern recognition

Clustering Instances

Let X be the unlabelled data set, that is,

$$X = \{x_1, x_2, \cdots, x_N\}; \tag{1}$$

The partition of X into k clusters, $C_1, \cdots, C_k$, so that the following conditions are met:

$$C_i \neq \emptyset, i = 1, \cdots, k; \tag{2}$$

$$\cup_{i=1}^{k} C_i = X; \tag{3}$$

$$C_i \cap C_j = \emptyset, i \neq j, i, j = 1, \cdots, k; \tag{4}$$

Requirements for Clustering

The goal of clustering is to group a set of unlabelled data. There are many typical requirements of clustering in machine learning and data mining, such as:

- Dealing with large data sets containing different types of attributes.
- Find the clusters with arbitrary shape.
- Ability to deal with noisy data in data streaming environment.
- Handling with high-dimensional data sets.
- Constraint-based clustering.

Types of Clustering Methods

The basic clustering methods are organised into the four categories:

1. Partitioning methods
2. Hierarchical methods
3. Density-based methods
4. Grid-based methods

Partitioning Method

- The partitioning method constructs k clusters of the given set of N instances, where $k \leq N$. It finds mutually exclusive clusters of spherical shape using the traditional distance measures (Euclidean distances).
- To find the cluster center, it may use mean or medoid (etc.) and apply iterative relocation technique to improve the clustering by moving instances from one cluster to another such as *k-means* clustering.
- The partitioning algorithms are ineffective for clustering high-dimensional big data.

Hierarchical Method

The hierarchical methods create a hierarchical decomposition of N instances. It can be divided into two categories:

1. top-down (or divisive) approache.
2. bottom-up (or agglomerative) approache

The **top-down** approach starts with a single cluster having all the N instances and then split into smaller clusters in each successive iteration, until eventually each instance is in one cluster, or a termination condition holds.

The **bottom-up** approach starts with each instance forming a separate cluster and then successively merges the clusters close to one another, until all the clusters are merged into a single cluster, or a termination condition holds.

Density-based method

The density-based methods cluster instances based on the distance between instances, which can find arbitrarily shaped clusters. It can cluster instances as dense regions in the data space, separated by sparse regions.

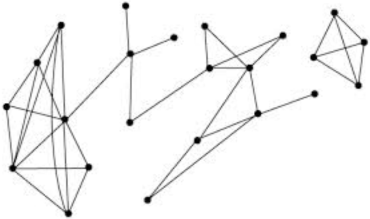

Figure: Clustering of a set of instances using density-based clustering.

Grid-based method

The grid-based methods use a multi-resolution grid data structure. It's fast processing time that typically independent of the number of instances, yet dependent on the grid size.

Similarity Measure

A similarity measure (SM), $sim(x_i, x_l)$, defined between any two instances, $x_i, x_l \in X$; An integer value k, the clustering problem is to define a mapping $f : X \rightarrow 1, \cdots, k$, where each instance, x_i is assigned to one cluster C_i, $1 \leq i \leq k$;
Given a cluster, $C_i, \forall x_{il}, x_{im} \in C_i$, and $x_j \notin C_i, sim(x_{il}, x_{im}) > sim(x_{il}, x_j)$;
A good clustering is that instances in the same cluster are "close" or related to each other, whereas instances of different clusters are "far apart" or very different from one another, which together satisfy the following requirements:

- Each cluster must contain at least one instance.
- Each instance must belong to exactly one cluster.

Distance Measure

A distance measure (DM), $dis(x_i, x_l)$, where $x_i, x_l \in X$, as opposed to similarity measure, is often used in clustering. Let's consider the well-known Euclidean distance or Euclidean metric (i.e. straight-line) between two instances in Euclidean space in Eq. 5.

$$dis(x_i, x_l) = \sqrt{\sum_{i=1}^{m}(x_i - x_l)^2} \tag{5}$$

Where, $x_i = (x_{i1}, x_{i2}, \cdots, x_{im})$ and $x_l = (x_{l1}, x_{l2}, \cdots, x_{lm})$ are two instances in Euclidean m-space.

k-Means or c-Means

It defines the centroid of a cluster, C_i as the mean value of the instances $\{x_{i1}, x_{i2}, \cdots, x_{iN}\} \in C_i$. It proceeds as follows. First, it randomly selects k instances, $\{x_{k1}, x_{k2}, \cdots, x_{kN}\} \in X$ each of which initially represents a cluster mean or center. For each of the remaining instances, $x_i \in X$, x_i is assigned to the cluster to which it is most similar, based on the Euclidean distance between the instance and the cluster mean. It then iteratively improves the within-cluster variation. For each cluster, C_i, it computes the new mean using the instances assigned to the cluster in the previous iteration. All the instances, $x_i \in X$ are then reassigned into clusters using the updated means as the new cluster centers. The iterations continue until the assignment is stable, that is the clusters formed in the current round are the same as those formed in the previous round.

Cluster Mean

A high degree of similarity among instances in clusters is obtained, while a high degree of dissimilarity among instances in different clusters is achieved simultaneously. The cluster mean of $C_i = \{x_{i1}, x_{i2}, \cdots, x_{iN}\}$ is defined in equation 6.

$$Mean = C_i = \frac{\sum_{j=1}^{N}(x_{ij})}{N} \tag{6}$$

Algorithm 1 k-Means Clustering

Input: $X = \{x_1, x_2, \cdots, x_N\}$ // A set of unlabelled instances.
k // the number of clusters
Output: A set of k clusters.
Method:

1: arbitrarily choose k number of instances, $\{x_{k1}, x_{k2}, \cdots, x_{kN}\} \in X$ as the initial k clusters center;
2: **repeat**
3: (re)assign each $x_i \in X \rightarrow k$ to which the x_i is the most similar based on the mean value of the $x_m \in k$;
4: update the k means, that is, calculate the mean value of the instances for each cluster;
5: **until** no change

Drawbacks of k-Means Clustering

The k-Means clustering is not guaranteed to converge to the global optimum and often terminates at a local optimum (as the initial cluster means are assigned randomly). It may not be used in some application such as when data with nominal features are involved. The k-Means method is not suitable for discovering clusters with non-convex shapes or clusters of very different size.

The time complexity of the k-Means algorithm is $O(nkt)$, where n is the total number of instances, k is the number of clusters, and t is the number of iterations. Normally, $k \ll n$ and $t \ll n$.

K-Means - An Example

Viewer

Relation: weather-weka.filters.unsupervised.attribute.Remove-R5

No.	outlook Nominal	temperature Numeric	humidity Numeric	windy Nominal
1	sunny	85.0	85.0	FALSE
2	sunny	80.0	90.0	TRUE
3	overcast	83.0	86.0	FALSE
4	rainy	70.0	96.0	FALSE
5	rainy	68.0	80.0	FALSE
6	rainy	65.0	70.0	TRUE
7	overcast	64.0	65.0	TRUE
8	sunny	72.0	95.0	FALSE
9	sunny	69.0	70.0	FALSE
10	rainy	75.0	80.0	FALSE
11	sunny	75.0	70.0	TRUE
12	overcast	72.0	90.0	TRUE
13	overcast	81.0	75.0	FALSE
14	rainy	71.0	91.0	TRUE

Undo OK Cancel

Figure: Weather Numeric Data.

K-Means using Weka 3

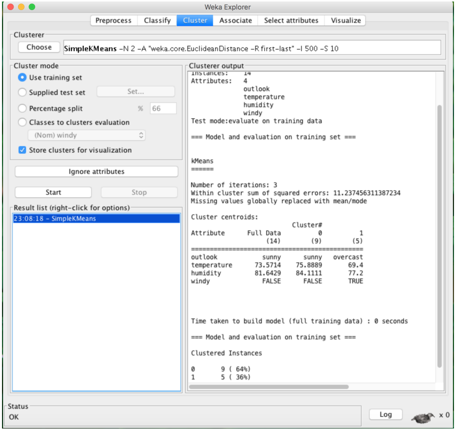

Figure: SimpleKMeans on Weather Nominal Data.

Run Information

```
=== Run information ===
Scheme:weka.clusterers.SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R
first-last" -I 500 -S 10
Relation:     weather.symbolic-weka.filters.unsupervised.attribute.Remove-R5
Instances:    14
Attributes:   4
              outlook
              temperature
              humidity
              windy
Test mode:evaluate on training data

=== Model and evaluation on training set ===

kMeans
======
Number of iterations: 4
Within cluster sum of squared errors: 21.000000000000004
Missing values globally replaced with mean/mode
Cluster centroids:
                          Cluster#
Attribute      Full Data         0          1
                    (14)      (10)        (4)
=============================================
outlook            sunny     sunny   overcast
temperature         mild      mild       cool
humidity            high      high     normal
windy              FALSE     FALSE       TRUE

Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===
Clustered Instances
0      10 ( 71%)
1       4 ( 29%)
```

Weka Cluster Visualize

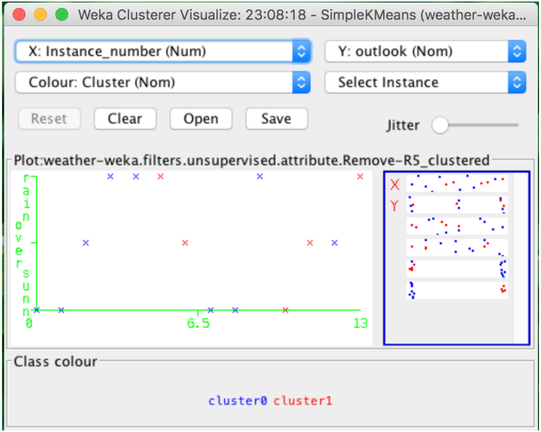

Figure: Clustering Weather Nominal Data.

k-Means: Another Example

Table: Height Data

Name	Gender	Height	Output
Kristina	F	1.6 m	Short
Jim	M	2 m	Tall
Maggie	F	1.9 m	Medium
Martha	F	1.88 m	Medium
Stephanie	F	1.7 m	Short
Bob	M	1.85 m	Medium
Kathy	F	1.6 m	Short
Dave	M	1.7 m	Short
Worth	M	2.2 m	Tall
Steven	M	2.1 m	Tall
Debbie	F	1.8 m	Medium
Todd	M	1.95 m	Medium
Kim	F	1.9 m	Medium
Amy	F	1.8 m	Medium
Wynette	F	1.75 m	Medium

Similarity-Based Clustering

A similarity-based clustering method (SCM) is an effective and robust clustering approach based on the similarity of instances, which is robust to initialise the cluster numbers and efficient to detect different volumes of clusters. SCM is a method for clustering a data set into most similar instances in the same cluster and most dissimilar instances in different clusters. The instances in SCM can self-organise local optimal cluster number and volumes without using cluster validity functions.

Similarity between Instances

Let's consider $sim(x_i, x_l)$ as the similarity measure between instances x_i and the lth cluster center x_l. The goal is to find x_l to maximise the total similarity measure shown in Eq. 7.

$$J_s(C) = \sum_{l=1}^{k} \sum_{i=1}^{N} f(sim(x_i, x_l)) \tag{7}$$

Where, $f(sim(x_i, x_l))$ is a reasonable similarity measure and $C = \{C_1, \cdots, C_k\}$. In general, the similarity-based clustering method uses feature values to check the similarity between instances. However, any suitable distance measure can be used to check the similarity between the instances.

Algorithm 2 Similarity-based Clustering

Input: $X = \{x_1, x_2, \cdots, x_N\}$ // A set of unlabelled instances.
Output: A set of clusters, $C = \{C_1, C_2, \cdots, C_k\}$.
Method:

$C = \emptyset$;
$k = 1$;
$C_k = \{x_1\}$;
$C = C \cup C_k$;
for $i = 2$ to N **do**
  for $l = 1$ to k **do**
    find the lth cluster center $x_l \in C_l$ to maximize the similarity measure, $sim(x_i, x_l)$;
  end for
  if $sim(x_i, x_l) \geq threshold\_value$ **then**
    $C_l = C_l \cup x_i$
  else
    $k = k + 1$;
    $C_k = \{x_i\}$;
    $C = C \cup C_k$;
  end if
end for

SCM - An Example

Viewer

Relation: weather.symbolic-weka.filters.unsupervised.attribute.Remove-R5

No.	outlook Nominal	temperature Nominal	humidity Nominal	windy Nominal
1	sunny	hot	high	FALSE
2	sunny	hot	high	TRUE
3	overcast	hot	high	FALSE
4	rainy	mild	high	FALSE
5	rainy	cool	normal	FALSE
6	rainy	cool	normal	TRUE
7	overcast	cool	normal	TRUE
8	sunny	mild	high	FALSE
9	sunny	cool	normal	FALSE
10	rainy	mild	normal	FALSE
11	sunny	mild	normal	TRUE
12	overcast	mild	high	TRUE
13	overcast	hot	normal	FALSE
14	rainy	mild	high	TRUE

Undo OK Cancel

Figure: Weather Nominal Data.

Nearest Neighbor (NN) Clustering

Instances are iteratively merged into the existing clusters that are closest. In NN clustering a threshold, t, is used to determine if instances will be added to existing clusters or if a new cluster is created. The complexity of the NN clustering algorithm is depends on the number of instances in the dataset. For each loop, each instance must be compared to each instance already in a cluster.

Thus, the time complexity of NN clustering algorithm is $O(n^2)$. We do need to calculate the distance between instances often, we assume that the space requirement is also $O(n^2)$.

Algorithm 3 Nearest Neighbor Clustering

Input: $D = \{x_1, x_2, \cdots, x_n\}$ // A set of instances.
A // Adjacency matrix showing distance between instances
Output: A set of C clusters.
Method:

$C_1 = \{x_1\}$;
$C = \{C_1\}$;
$k = 1$;
for $i = 2$ to n **do**
  find x_m in some cluster C_m in C so that $dis(x_i, x_m)$ is the smallest;
  if $dis(x_i, x_m) \leq$t, *threshold_value* **then**
    $C_m = C_m \cup x_i$
  else
    $k = k + 1$;
    $C_k = \{x_i\}$;
    $C = C \cup C_k$;
  end if
end for

Euclidean Vs. Manhattan distance

The distance between the two points in the plane with coordinate (x,y) and (a,b) is given by:

$$EuclideanDistance, (x, y)(a, b) = \sqrt{(x - a)^2 + (y - b)^2} \tag{8}$$

$$ManhattanDistance, (x, y)(a, b) = |x - a| + |y - b| \tag{9}$$

Ensemble Clustering

Ensemble clustering is a process of integrating multiple clustering algorithms to form a single strong clustering approach that usually provides better clustering results. It generates a set of clusters from a given unlabelled data set and then combines the clusters into final clusters to improve the quality of individual clustering.

- No single cluster analysis method is optimal.
- Different clustering methods may produce different clusters, because they impose different structure on the data set.
- Ensemble clustering performs more effectively in high dimensional complex data.
- It's a good alternative when facing cluster analysis problems.

Ensemble clustering (con.)

Generally three strategies are applied in ensemble clustering:

1. Using different clustering algorithms on the same data set to create heterogeneous clusters.
2. Using different samples/ subsets of the data with different clustering algorithms to cluster them to produce component clusters.
3. Running the same clustering algorithm many times on same data set with different parameters or initialisations to create homogeneous clusters.

The main goal of the ensemble clustering is to integrate component clustering into one final clustering with a higher accuracy.

Subspace Clustering

The subspace clustering finds subspace clusters in high-dimensional data. It can be classified into three groups:

1. Subspace search methods.
2. Correlation-based clustering methods
3. Biclustering methods.

A subspace search method searches various subspaces for clusters (set of instances that are similar to each other in a subspace) in the full space. It uses two kinds of strategies:

- Bottom-up approach - start from low-dimensional subspace and search higher-dimensional subspaces.
- Top-down approach - start with full space and search smaller subspaces recursively.

Subspace Clustering (con.)

A correlation-based approach uses space transformation methods to derive a set of new, uncorrelated dimensions, and then mine clusters in the new space or its subspaces. It uses PCA-based approach (principal components analysis), the Hough transform, and fractal dimensions.

Biclustering methods cluster both instances and features simultaneously, where cluster analysis involves searching data matrices for sub-matrices that show unique patterns as clusters.

Weka 3: Data Mining Software in Java

Weka (Waikato Environment for Knowledge Analysis) is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, **clustering**, association rules, and visualization.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.

Clustering Algorithms in Weka 3

1. SimpleKMeans - Cluster using the k-Means method.
2. XMeans - Extension of k-Means.
3. DBScan - Nearest-neighbor-based clustering that automatically determines the number of clusters.
4. OPTICS - Extension of DBScan to hierarchical clustering.
5. HierarchicalClusterer - Agglomerative hierarchical clustering.
6. MakeDensityBasedCluster - Wrap a clusterer to make it return distribution and density.
7. EM - Cluster using expectation maximization.
8. CLOPE - Fast clustering of transactional data.
9. Cobweb - Implements the Cobweb and Classit clustering algorithms.
10. FarthestFirst - Cluster using the farthest first traversal algorithm.
11. FilteredClusterer - Runs a clusterer on filtered data.
12. sIB - Cluster using the sequential information bottleneck algorithm.

Weka GUI Chooser

Figure: Weka GUI Chooser.

Weka Explorer

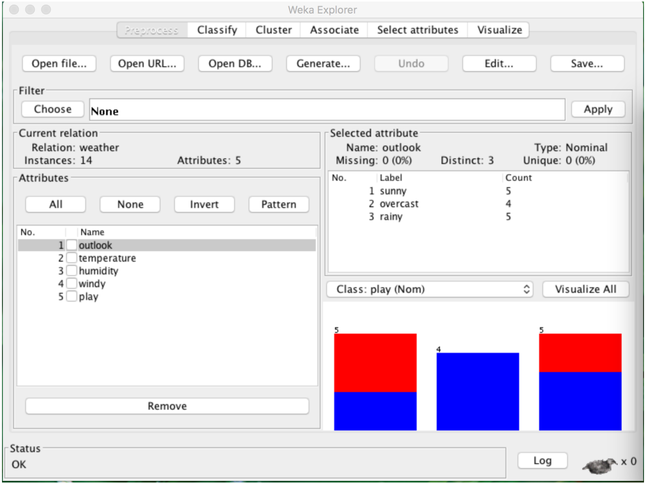

Figure: Weka Explorer.

Clustering using Weka

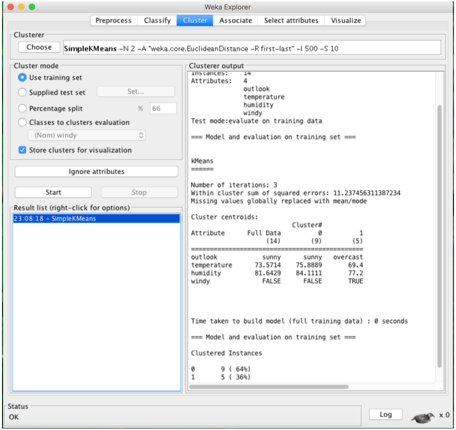

Figure: Cluster - Weka Explorer.

Reference Books

1. Data Mining Concepts and Technique, by Jiawei Han, Micheline Kamber, and Jian Pei (Third Edition)
2. Data Mining Practical Machine Learning Tools and Techniques, by Ian H. Witten, Eibe Frank, and Mark A. Hall (Third Edition)
3. Data Mining Knowledge Discovery and Applications, Edited by Adem Karahoca
4. Mining Complex Data, by Djamel A. Zighed, Shusaku Tsumoto, Zbigniew W. Ras, and Hakim Hacid