Association Rule Mining, as the name suggests, association rules are simple If/Then statements that help discover relationships between seemingly independent relational databases or other data repositories.

Most machine learning algorithms work with numeric datasets and hence tend to be mathematical. However, association rule mining is suitable for non-numeric, categorical data and requires just a little bit more than simple counting.

Association rule mining is a procedure which aims to observe frequently occurring patterns, correlations, or associations from datasets found in various kinds of databases such as relational databases, transactional databases, and other forms of repositories.

An association rule has 2 parts:

- **an antecedent (if) and**
- **a consequent (then)**

An antecedent is something that's found in data, and a consequent is an item that is found in combination with the antecedent. Have a look at this rule for instance:

> *"If a customer buys bread, he's 70% likely of buying milk."*

In the above association rule, bread is the antecedent and milk is the consequent. Simply put, it can be understood as a retail store's association rule to target their customers better. If the above rule is a result of a thorough analysis of some data sets, it can be used to not only improve customer service but also improve the company's revenue.
Association rules are created by thoroughly analyzing data and looking for frequent if/then patterns. Then, depending on the following two parameters, the important relationships are observed:

1. **Support**: Support indicates how frequently the if/then relationship appears in the database.

2. **Confidence**: Confidence tells about the number of times these relationships have been found to be true.

Types Of Association Rules In Data Mining

There are typically four different types of association rules in data mining. They are

- Multi-relational association rules
- Generalized Association rule
- Interval Information Association Rules
- Quantitative Association Rules

Multi-Relational Association Rule

Also known as MRAR, multi-relational association rule is defined as a new class of association rules that are usually derived from different or multi-relational databases. Each rule under this class has one entity with different relationships that represent the indirect relationships between entities.

Generalized Association Rule

Moving on to the next type of association rule, the generalized association rule is largely used for getting a rough idea about the interesting patterns that often tend to stay hidden in data.

Quantitative Association Rules

This particular type is actually one of the most unique kinds of all the four association rules available. What sets it apart from the others is the presence of numeric attributes in at least one attribute of quantitative association rules. This is in contrast to the generalized association rule, where the left and right sides consist of categorical attributes.

Algorithms Of Associate Rule In Data Mining

There are mainly three different types of algorithms that can be used to generate associate rules in data mining. Let’s take a look at them.

- Apriori Algorithm

  Apriori algorithm identifies the frequent individual items in a given database and then expands them to larger item sets, keeping in check that the item sets appear sufficiently often in the database.

- Eclat Algorithm

  ECLAT algorithm is also known as Equivalence Class Clustering and bottomup. Latice Traversal is another widely used method for associate rule in data mining. Some even consider it to be a better and more efficient version of the Apriori algorithm.

- FP-growth Algorirthm

  Also known as the recurring pattern, this algorithm is particularly useful for finding frequent patterns without the need for candidate generation. It mainly operates in two stages namely, FP-tree construction and extract frequently used item sets.

  Now that you have a basic understanding of what is association rule,

types Of Data Used In Cluster Analysis Are:

- **Interval-Scaled variables**
- **Binary variables**
- **Nominal, Ordinal, and Ratio variables**
- **Variables of mixed types**

Types Of Data In Cluster Analysis Are:

Interval-Scaled Variables

Interval-scaled variables are continuous measurements of a roughly linear scale.

Typical examples include weight and height, latitude and longitude coordinates (e.g., when clustering houses), and weather temperature.

The measurement unit used can affect the clustering analysis. For example, changing measurement units from meters to inches for height, or from kilograms to pounds for weight, may lead to a very different clustering structure.

In general, expressing a variable in smaller units will lead to a larger range for that variable, and thus a larger effect on the resulting clustering structure.

To help avoid dependence on the choice of measurement

units, the data should be standardized. Standardizing measurements attempts to give all variables an equal weight.

This is especially useful when given no prior knowledge of the data. However, in some applications, users may intentionally want to give more weight to a certain set of variables than to others.

For example, when clustering basketball player candidates, we may prefer to give more weight to the variable height.

Binary Variables

A binary variable is a variable that can take only 2 values.

For example, generally, gender variables can take 2 variables male and female.

Contingency Table For Binary Data

Let us consider binary values 0 and 1

	1	0	*sum*
1	a	b	$a+b$
0	c	d	$c+d$
sum	$a+c$	$b+d$	p

Let p=a+b+c+d

Simple matching coefficient (invariant, if the binary variable is symmetric):

$$d(i, j) = \frac{b + c}{a + b + c + d}$$

Jaccard coefficient (noninvariant if the binary variable is asymmetric):

$$d(i, j) = \frac{b + c}{a + b + c}$$

Nominal or Categorical Variables

A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green.

Method 1: Simple matching

The dissimilarity between two objects i and j can be computed based on the simple matching.

m: Let m be no of matches (i.e., the number of variables for which i and j are in the same state).

p: Let p be total no of variables.

$$d(i, j) = \frac{p - m}{p}$$

Method 2: use a large number of binary variables

Creating a new binary variable for each of the M nominal states.

Ordinal Variables

An ordinal variable can be discrete or continuous.

In this order is important, e.g., rank.

It can be treated like interval-scaled

By replacing xif by their rank,

$$r_{if} \in \{1, \ldots, M_f\}$$

By mapping the range of each variable onto [0, 1] by replacing the i-th object in the f-th variable by,

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

Then compute the dissimilarity using methods for interval-scaled variables.

Ratio-Scaled Intervals

Ratio-scaled variable: It is a positive measurement on a nonlinear scale, approximately at an exponential scale, such as Ae^Bt or A^e-Bt.

Methods:

- First, treat them like interval-scaled variables — not a good choice! (why?)
- Then apply logarithmic transformation i.e.y = log(x)
- Finally, treat them as continuous ordinal data treat their rank as interval-scaled.

Variables Of Mixed Type

A database may contain all the six types of variables symmetric binary, asymmetric binary, nominal, ordinal, interval, and ratio.

And those combinedly called as mixed-type variables.

Partitioning Method (K-Mean) in Data Mining

Partitioning Method:
This clustering method classifies the information into multiple groups based on the characteristics and similarity of the data. Its the data analysts to specify the number of clusters that has to be generated for the clustering methods.

In the partitioning method when database(D) that contains multiple(N) objects then the partitioning method constructs user-specified(K) partitions of the data in which each partition represents a cluster and a particular region. There are many algorithms that come under partitioning method some of the popular ones are K-Mean, PAM(K-Mediods), CLARA algorithm (Clustering Large Applications) etc.

In this article, we will be seeing the working of K Mean algorithm in detail.

K-Mean (A centroid based Technique):
The K means algorithm takes the input parameter K from the user and partitions the dataset containing N objects into K clusters so that resulting similarity among the data objects inside the group (intracluster) is high but the similarity of data objects with the data objects from outside the cluster is low (intercluster). The similarity of the cluster is determined with respect to the mean value of the cluster.

It is a type of square error algorithm. At the start randomly k objects from the dataset are chosen in which each of the objects represents a cluster mean(centre). For the rest of the data objects, they are assigned to the nearest cluster based on their distance from the cluster mean. The new mean of each of the cluster is then calculated with the added data objects.

Algorithm: K mean:

Input:

K: The number of clusters in which the dataset has to be divided

D: A dataset containing N number of objects

Output:

A dataset of K clusters

Method:

1. Randomly assign K objects from the dataset(D) as cluster centres(C)
2. (Re) Assign each object to which object is most similar based upon mean values.
3. Update Cluster means, i.e., Recalculate the mean of each cluster with the updated values.
4. Repeat Step 4 until no change occurs.

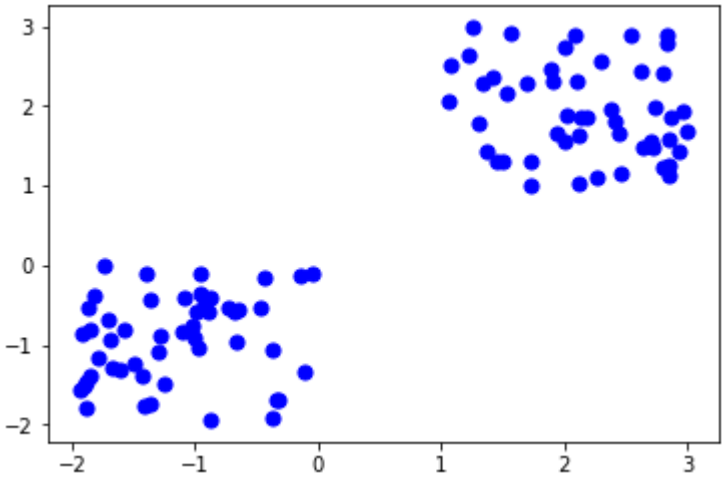

Figure – K-mean Clustering

Flowchart:

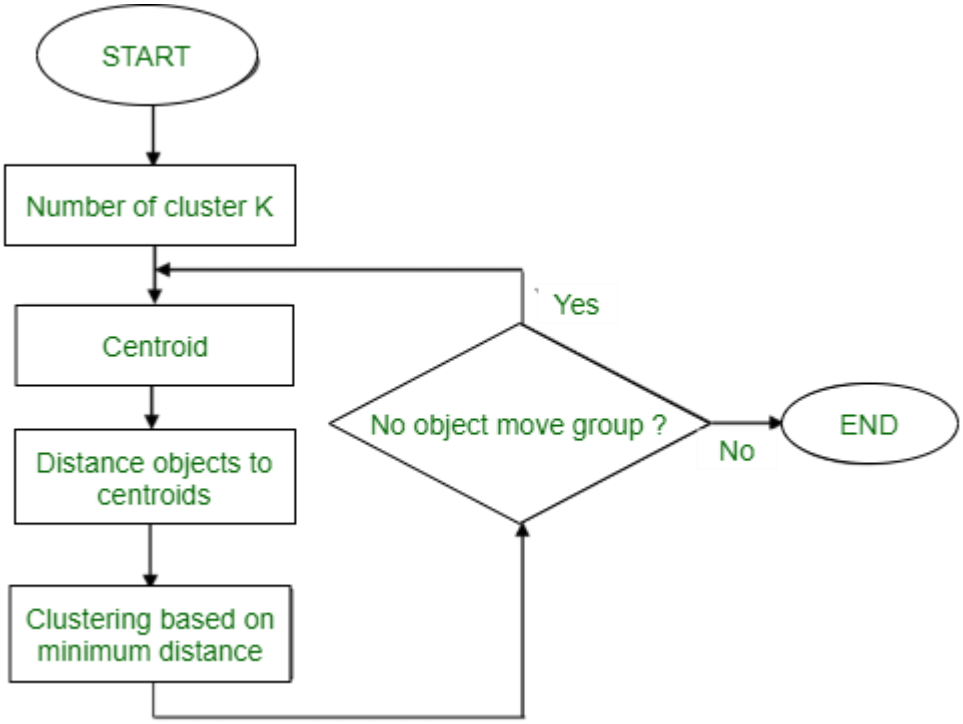

Figure – K-mean Clustering

Example: Suppose we want to group the visitors to a website using just their age as follows:

16, 16, 17, 20, 20, 21, 21, 22, 23, 29, 36, 41, 42, 43, 44, 45, 61, 62, 66

Initial Cluster:

K=2

Centroid(C1) = 16 [16]

Centroid(C2) = 22 [22]

Note: These two points are chosen randomly from the dataset.

Iteration-1:

C1 = 16.33 [16, 16, 17]

C2 = 37.25 [20, 20, 21, 21, 22, 23, 29, 36, 41, 42, 43, 44, 45, 61, 62, 66]

Iteration-2:

C1 = 19.55 [16, 16, 17, 20, 20, 21, 21, 22, 23]

C2 = 46.90 [29, 36, 41, 42, 43, 44, 45, 61, 62, 66]

Iteration-3:

C1 = 20.50 [16, 16, 17, 20, 20, 21, 21, 22, 23, 29]

C2 = 48.89 [36, 41, 42, 43, 44, 45, 61, 62, 66]

Iteration-4:

C1 = 20.50 [16, 16, 17, 20, 20, 21, 21, 22, 23, 29]

C2 = 48.89 [36, 41, 42, 43, 44, 45, 61, 62, 66]

Model-based clustering is a statistical approach to data clustering. The observed (multivariate) data is considered to have been created from a finite combination of component models. Each component model is a probability distribution, generally a parametric multivariate distribution.

For instance, in a multivariate Gaussian mixture model, each component is a multivariate Gaussian distribution. The component responsible for generating a particular observation determines the cluster to which the observation belongs.

Model-based clustering is a try to advance the fit between the given data and some mathematical model and is based on the assumption that data are created by a combination of a basic probability distribution.

There are the following types of model-based clustering are as follows −

Statistical approach − Expectation maximization is a popular iterative refinement algorithm. An extension to k-means −

- It can assign each object to a cluster according to weight (probability distribution).
- New means are computed based on weight measures.

The basic idea is as follows −

- It can start with an initial estimate of the parameter vector.
- It can be used to iteratively rescore the designs against the mixture density made by the parameter vector.
- It is used to rescored patterns are used to update the parameter estimates.
- It can be used to pattern belonging to the same cluster if they are placed by their scores in a particular component.

Algorithm

- Initially, assign k cluster centers randomly.

It can be iteratively refined the clusters based on two steps are as follows −

- **Expectation step** − It can assign each data point X_i to cluster C_i with the following probability

$$P(X_i \in C_k) = P(C_k \mid X_i) = P(C_k)P(X_i \mid C_k)P(X_i)$$

- **Maximization step** − It can be used to estimate of model parameter

$$m_k = \frac{1}{N}\sum_{i=1}^{N} X_i \frac{P(X_i \in C_k)}{P(X_i \in C_j)} X_j$$

Multilevel Association Rule in data mining

- **Last Updated :** 16 Dec, 2021

· Read

· Discuss

Multilevel Association Rule :
Association rules created from mining information at different degrees of reflection are called various level or staggered association rules.
Multilevel association rules can be mined effectively utilizing idea progressions under a help certainty system.
Rules at a high idea level may add to good judgment while rules at a low idea level may not be valuable consistently.

Utilizing uniform least help for all levels :

- At the point when a uniform least help edge is utilized, the pursuit system is rearranged.
- The technique is likewise straightforward, in that clients are needed to indicate just a single least help edge.
- A similar least help edge is utilized when mining at each degree of deliberation. (for example for mining from “PC” down to “PC”). Both “PC” and “PC” discovered to be incessant, while “PC” isn’t.

Needs of Multidimensional Rule :

- Sometimes at the low data level, data does not show any significant pattern but there is useful information hiding behind it.
- The aim is to find the hidden information in or between levels of abstraction.

Approaches to multilevel association rule mining :

1. Uniform Support(Using uniform minimum support for all level)
2. Reduced Support (Using reduced minimum support at lower levels)
3. Group-based Support(Using item or group based support)

Let’s discuss one by one.

1. **Uniform Support –**
   At the point when a uniform least help edge is used, the search methodology is simplified. The technique is likewise basic in that clients are needed to determine just a single least help threshold. An advancement technique can be adopted, based on the information that a progenitor is a superset of its descendant. the search keeps away from analyzing item sets containing anything that doesn't have minimum support. The uniform support approach however has some difficulties. It is unlikely that items at lower levels of abstraction will occur as frequently as those at higher levels of abstraction. If the minimum support threshold is set too high it could miss several meaningful associations occurring at low abstraction levels. This provides the motivation for the following approach.
2. **Reduce Support –**
   For mining various level relationship with diminished support, there are various elective hunt techniques as follows.

- **Level-by-Level independence –**
  This is a full-broadness search, where no foundation information on regular item sets is utilized for pruning. Each hub is examined, regardless of whether its parent hub is discovered to be incessant.
- **Level – cross-separating by single thing –**
  A thing at the I level is inspected if and just if its parent hub at the (I-1) level is regular .all in all, we research a more explicit relationship from a more broad one. If a hub is frequent, its kids will be examined; otherwise, its descendant is pruned from the inquiry.
- **Level-cross separating by – K-itemset –**
  A-itemset at the I level is inspected if and just if it's For mining various level relationship with diminished support, there are various elective hunt techniques.
- **Level-by-Level independence –**
  This is a full-broadness search, where no foundation information on regular item sets is utilized for pruning. Each hub is examined, regardless of whether its parent hub is discovered to be incessant.
- **Level – cross-separating by single thing –**
  A thing at the 1st level is inspected if and just if its parent hub at the (I-1) the level is regular .all in all, we research a more explicit relationship from a more broad one. If a hub is frequent, its kids will be examined otherwise, its descendant is pruned from the inquiry.
- **Level-cross separating by – K-item set –**
  A-item set at the I level is inspected if and just if its corresponding parents A item set (i-1) level is frequent.

3. **Group-based support –**
   The group-wise threshold value for support and confidence is input by the user or expert. The group is selected based on a product price or item set because often expert has insight as to which groups are more important than others.
   Example –
   For e.g. Experts are interested in purchase patterns of laptops or clothes in the non and electronic category. Therefore low support threshold is set for this group to give attention to these items' purchase patterns.

Data Mining Multidimensional Association Rule

- **Difficulty Level :** Expert
- **Last Updated :** 17 Dec, 2020

· Read

· Discuss

In this article, we are going to discuss Multidimensional Association Rule. Also, we will discuss examples of each. Let's discuss one by one.

Multidimensional Association Rules **:**

In Multi dimensional association rule Qualities can be absolute or quantitative.

- Quantitative characteristics are numeric and consolidates order.
- Numeric traits should be discretized.
- Multi dimensional affiliation rule comprises of more than one measurement.
- **Example –**buys(X, "IBM Laptop computer")buys(X, "HP Inkjet Printer")

Approaches in mining multi dimensional affiliation rules :
Three approaches in mining multi dimensional affiliation rules are as following.

1. **Using static discretization of quantitative qualities :**
   - Discretization is static and happens preceding mining.
   - Discretized ascribes are treated as unmitigated.
   - Use apriori calculation to locate all k-regular predicate sets(this requires k or k+1 table outputs). Each subset of regular predicate set should be continuous.

Example –
If in an information block the 3D cuboid (age, pay, purchases) is continuous suggests (age, pay), (age, purchases), (pay, purchases) are likewise regular.

Note –
Information blocks are appropriate for mining since they make mining quicker. The cells of an n-dimensional information cuboid relate to the predicate cells.

2. **Using powerful discretization of quantitative traits :**
   - Known as mining Quantitative Association Rules.
   - Numeric properties are progressively discretized.

Example –:

age(X, "20..25") Λ income(X, "30K..41K")buys (X, "Laptop Computer")

3. **Grid FOR TUPLES :**

Using distance based discretization with bunching –

This id dynamic discretization measure that considers the distance between information focuses. It includes a two stage mining measure as following.

- Perform bunching to discover the time period included.
- Get affiliation rules via looking for gatherings of groups that happen together.

The resultant guidelines may fulfill –

- Bunches in the standard precursor are unequivocally connected with groups of rules in the subsequent.
- Bunches in the forerunner happen together.
- Bunches in the ensuing happen together.