**UNIT V**

Cluster Analysis

* Cluster: A collection of data objects
* similar (or related) to one another within the same group
* dissimilar (or unrelated) to the objects in other groups
* Cluster analysis (or *clustering*, *data segmentation, …*)
  + Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
* Unsupervised learning: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)

Requirements for Cluster Analysis

* Scalability
* Ability to deal with different types of attributes
* Discovery of clusters with arbitrary shape
* Requirements for domain knowledge to determine input parameters
* Ability to deal with noisy data
* Incremental clustering and insensitivity to input order
* Capability of clustering high-dimensionality data
* Constraint-based clustering
* Interpretability and usability

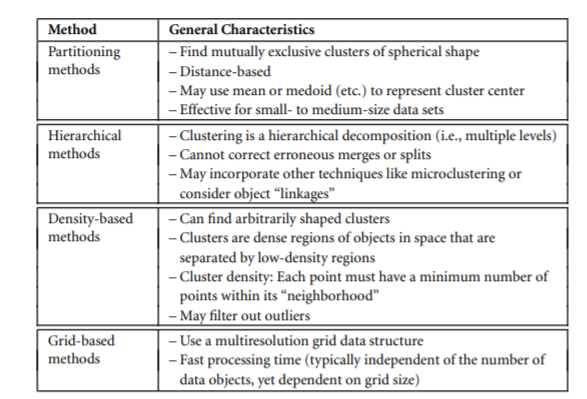
Overview of Basic Clustering Methods

Partitioning methods: Given a set of n objects, a partitioning method constructs k partitions of the data, where each partition represents a cluster and k ≤ n. That is, it divides the data into k groups such that each group must contain at least one object.

Density-based methods can divide a set of objects into multiple exclusive clusters, or a hierarchy of clusters

Grid-based methods: Grid-based methods quantize the object space into a finite number of cells that form a grid structure

Hierarchical methods: A hierarchical method creates a hierarchical decomposition of the given set of data objects.



**PARTITIONING METHODS**

Given a data set, D, of n objects, and k, the number of clusters to form, a partitioning algorithm organizes the objects into k partitions (k ≤ n), where each partition represents a cluster

**K-MEANS: A CENTROID-BASED TECHNIQUE**

Suppose a data set, D, contains n objects in Euclidean space. Partitioning methods distribute the objects in D into k clusters, C1,...,Ck , that is, Ci ⊂ D and Ci ∩Cj = ∅ for (1 ≤ i,j ≤ k).

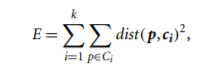
A centroid-based partitioning technique uses the centroid of a cluster, Ci , to represent the cluster.

The centroid of a cluster is its center point.

The centroid can be defined in various ways such as by the mean or medoid of the objects (or points) assigned to the cluster.

The difference between an object p ∈ Ci and ci , the representative of the cluster, is measured by dist(p,ci), where dist(x,y) is the Euclidean distance between two points x and y.

The quality of cluster Ci can be measured by the within-cluster variation, which is the sum of squared error between all objects in Ci and the centroid ci , defined as



where E is the sum of the squared error for all objects in the data set;

p is the point in space representing a given object; and ci is the centroid of cluster Ci (both p and ci are multidimensional).

**Working of the k-means algorithm**

The k-means algorithm defines the centroid of a cluster as the mean value of the points within the cluster.

It proceeds as follows.

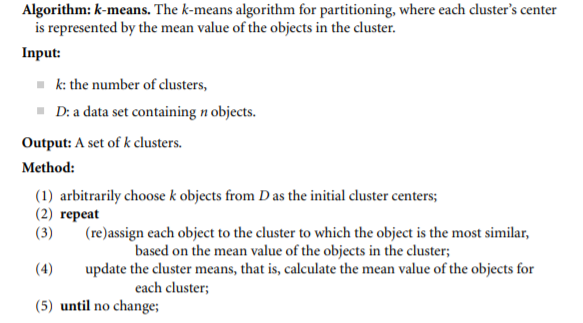
First, it randomly selects k of the objects in D, each of which initially represents a cluster mean or center. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar, based on the Euclidean distance between the object and the cluster mean.

The k-means algorithm then iteratively improves the within-cluster variation.

For each cluster, it computes the new mean using the objects assigned to the cluster in the previous iteration. All the objects are then reassigned 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.

Algorithm



The time complexity of the k-means algorithm is O(nkt), where n is the total number of objects, k is the number of clusters, and t is the number of iterations. Normally, and



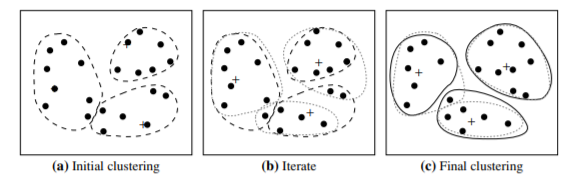
Example:

Clustering by k-means partitioning. Consider a set of objects located in 2-D space, as depicted in Figure (a). Let k = 3, that is, the user would like the objects to be partitioned into three clusters.

According to the algorithm in Figure , we arbitrarily choose three objects as the three initial cluster centers, where cluster centers are marked by a +. Each object is assigned to a cluster based on the cluster center to which it is the nearest. Such a distribution forms silhouettes encircled by dotted curves, as shown in Figure (a).

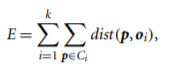
Next, the cluster centers are updated. That is, the mean value of each cluster is recalculated based on the current objects in the cluster. Using the new cluster centers, the objects are redistributed to the clusters based on which cluster center is the nearest. Such a redistribution forms new silhouettes encircled by dashed curves, as shown in Figure (b).

This process iterates, leading to Figure (c). The process of iteratively reassigning objects to clusters to improve the partitioning is referred to as iterative relocation. Eventually, no reassignment of the objects in any cluster occurs and so the process terminates. The resulting clusters are returned by the clustering process.



**k-Medoids: A Representative Object-Based Technique**

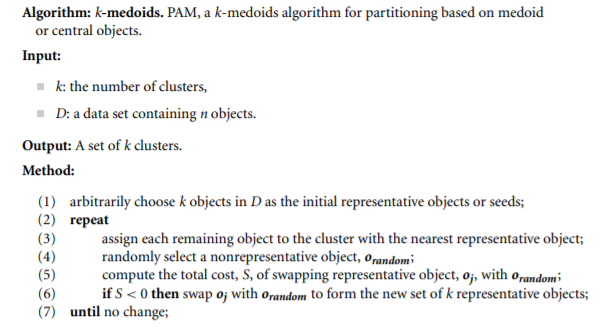
The partitioning method is then performed based on the principle of minimizing the sum of the dissimilarities between each object p and its corresponding representative object. That is, an absolute-error criterion is used, defined as



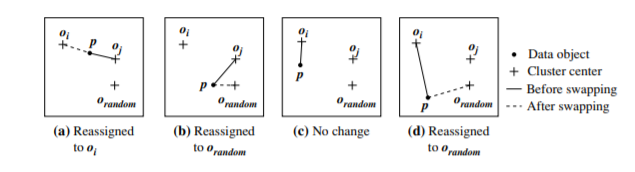
where E is the sum of the absolute error for all objects p in the data set, and oi is the representative object of Ci. This is the basis for the k-medoids method, which groups n objects into k clusters by minimizing the absolute error.

**Partitioning Around Medoids (PAM) algorithm**

Like the k-means algorithm, the initial representative objects (called seeds) are chosen arbitrarily. We consider whether replacing a representative object by a nonrepresentative object would improve the clustering quality.



Example



let o1,...,ok be the current set of representative objects (i.e., medoids).

To determine whether a nonrepresentative object, denoted by orandom, is a good replacement for a current medoid oj (1 ≤ j ≤ k), we calculate the distance from every object p to the closest object in the set {o1,...,oj−1,orandom,oj+1,...,ok}, and use the distance to update the cost function. The reassignments of objects to {o1,...,oj−1,orandom,oj+1,...,ok} are simple. Suppose object p is currently assigned to a cluster represented by medoid oj (Figure a or b).

Object p needs to be reassigned to either orandom or some other cluster represented by oi (i 6= j), whichever is the closest. For example, in Figure a, p is closest to oi and therefore is reassigned to oi .

In Figure b, however, p is closest to orandom and so is reassigned to orandom. What if, instead, p is currently assigned to a cluster represented by some other object oi.

Object o remains assigned to the cluster represented by oi as long as o is still closer to oi than to orandom (Figure c). Otherwise, o is reassigned to orandom (Figure d)

**Hierarchical Methods**

A hierarchical clustering method works by grouping data objects into a hierarchy or “tree” of clusters.

Types of Hierarchical **Methods**

* Agglomerative versus Divisive Hierarchical Clustering
* Distance Measures in Algorithmic Methods
* BIRCH: Multiphase Hierarchical Clustering Using Clustering Feature Trees
* Chameleon: Multiphase Hierarchical Clustering Using Dynamic Modeling

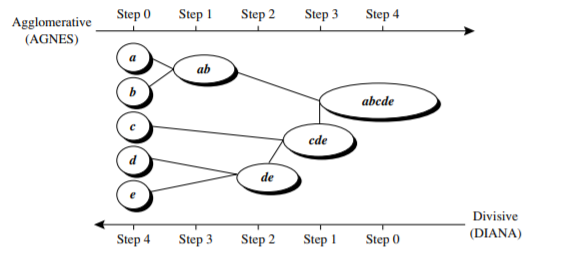
Agglomerative versus Divisive Hierarchical Clustering

An agglomerative hierarchical clustering method uses a bottom-up strategy. It typically starts by letting each object form its own cluster and iteratively merges clusters into larger and larger clusters, until all the objects are in a single cluster or certain termination conditions are satisfied.

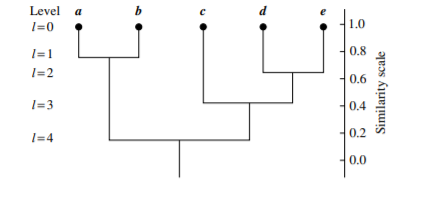
A divisive hierarchical clustering method employs a top-down strategy. It starts by placing all objects in one cluster, which is the hierarchy’s root. It then divides the root cluster into several smaller subclusters, and recursively partitions those clusters into smaller ones.

Example

Agglomerative versus divisive hierarchical clustering. Figure shows the application of AGNES (AGglomerative NESting), an agglomerative hierarchical clustering method, and DIANA (DIvisive ANAlysis), a divisive hierarchical clustering method, on a data set of five objects, {a,b,c,d, e}.



A tree structure called a **dendrogram** is commonly used to represent the process of hierarchical clustering.

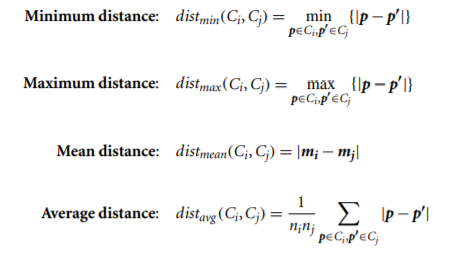


where l = 0 shows the five objects as singleton clusters at level 0. At l = 1, objects a and b are grouped together to form the first cluster (a,b).

**DISTANCE MEASURES IN ALGORITHMIC METHODS**

Four widely measures for distance between clusters are as follows, where |p − p 1 | is the distance between two objects or points, p and p 1 ; mi is the mean for cluster, Ci ; and ni is the number of objects in Ci .

They are also known as linkage measures.



When an algorithm uses the minimum distance, dmin(Ci ,Cj), to measure the distance between clusters, it is sometimes called **a nearest-neighbor clustering algorithm**.

If the clustering process is terminated when the distance between nearest clusters exceeds a user-defined threshold, it is called a **single-linkage algorithm.**

An agglomerative hierarchical clustering algorithm that uses the minimum distance measure is also called a **minimal spanning tree algorithm**, where a spanning tree of a graph is a tree that connects all vertices, and a minimal spanning tree is the one with the least sum of edge weights.

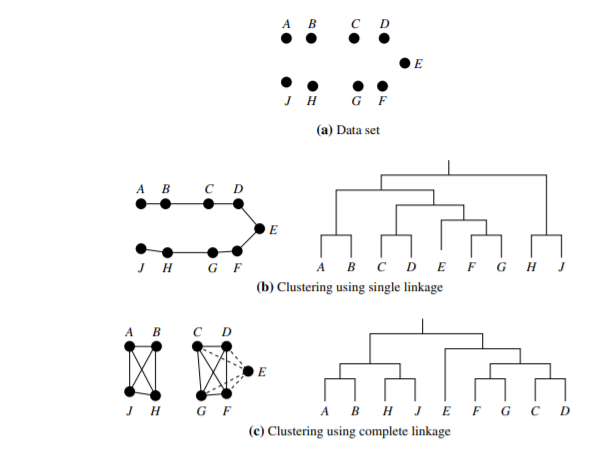
When an algorithm uses the maximum distance, dmax(Ci ,Cj), to measure the distance between clusters, it is sometimes called a **farthest-neighbor clustering algorithm**

If the clustering process is terminated when the maximum distance between nearest clusters exceeds a user-defined threshold, it is called a **complete-linkage algorithm.**

The **mean distance** is the simplest to compute, the **average distance** is advantageous in that it can handle categoric as well as numeric data.

**Example :**

**Single versus complete linkages.**



**BIRCH: MULTIPHASE HIERARCHICAL CLUSTERING USING CLUSTERING FEATURE TREES**

**Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH)** is designed for clustering a large amount of numeric data by integrating hierarchical clustering (at the initial microclustering stage) and other clustering methods such as iterative partitioning (at the later macroclustering stage).

It overcomes the two difficulties in agglomerative clustering methods: (1) scalability and (2) the inability to undo what was done in the previous step.

BIRCH uses the notions of clustering feature to summarize a cluster, and clustering feature tree (CF-tree) to represent a cluster hierarchy. These structures help the clustering method achieve good speed and scalability in large or even streaming databases

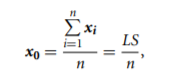
Consider a cluster of n d-dimensional data objects or points. The clustering feature (CF) of the cluster is a 3-D vector summarizing information about clusters of objects. It is defined as

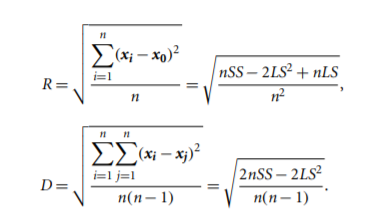
 

where LS is the linear sum of the n points ,

SS is the square sum of the data points

A clustering feature is essentially a summary of the statistics for the given cluster. Using a clustering feature, we can easily derive many useful statistics of a cluster. For example, the cluster’s centroid, x0, radius, R, and diameter, D, are





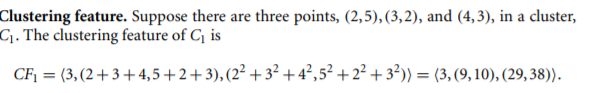
Here, R is the average distance from member objects to the centroid, and D is the average pairwise distance within a cluster. Both R and D reflect the tightness of the cluster around the centroid.

Clustering features are additive. That is, for two disjoint clusters, C1 and C2, with the clustering features CF1 = (n1,LS1,SS1)and CF2 = (n2,LS2,SS2), respectively, the clustering feature for the cluster that formed by merging C1 and C2 is simply

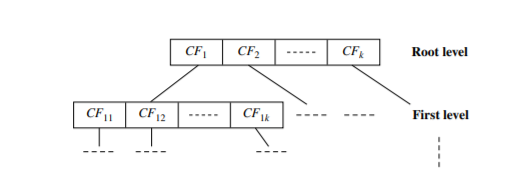
CF1 + CF2 = (n1 + n2,LS1 + LS2,SS1 + SS2).

Cf1= (n1,ls1,ss1) n=3 ls1=2+3+4,5+2+3, ss1=22+32+42, 52+22+32

cf1= 3,( 9,10)(29,38)

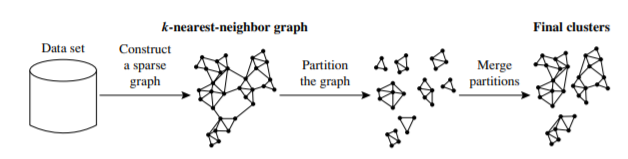


A CF-tree is a height-balanced tree that stores the clustering features for a hierarchical clustering



**CHAMELEON: MULTIPHASE HIERARCHICAL CLUSTERING USING DYNAMIC MODELING**

Chameleon is a hierarchical clustering algorithm that uses dynamic modeling to determine the similarity between pairs of clusters. In Chameleon, cluster similarity is assessed based on (1) how well connected objects are within a cluster and (2) the proximity of clusters.



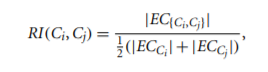
Chameleon uses a k-nearest-neighbor graph approach to construct a sparse graph, where each vertex of the graph represents a data object, and there exists an edge between two vertices (objects) if one object is among the k-most similar objects to the other.

The edges are weighted to reflect the similarity between objects. Chameleon uses a graph partitioning algorithm to partition the k-nearest-neighbor graph into a large number of relatively small subclusters such that it minimizes the edge cut.

That is, a cluster C is partitioned into subclusters Ci and Cj so as to minimize the weight of the edges that would be cut should C be bisected into Ci and Cj . It assesses the absolute interconnectivity between clusters Ci and Cj .

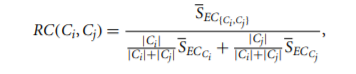
Chameleon determines the similarity between each pair of clusters Ci and Cj according to their relative interconnectivity, RI(Ci ,Cj), and their relative closeness, RC(Ci ,Cj).

The relative interconnectivity, RI(Ci ,Cj), between two clusters, Ci and Cj , is defined as the absolute interconnectivity between Ci and Cj , normalized with respect to the internal interconnectivity of the two clusters, Ci and Cj . That is,



where EC{Ci ,Cj} is the edge cut as previously defined for a cluster containing both Ci and Cj . Similarly, ECCi (or ECCj ) is the minimum sum of the cut edges that partition Ci (or Cj) into two roughly equal parts.

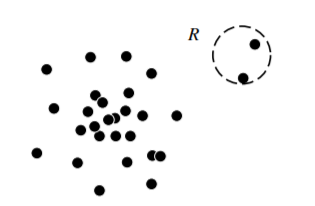
The relative closeness, RC(Ci ,Cj), between a pair of clusters, Ci and Cj , is the absolute closeness between Ci and Cj , normalized with respect to the internal closeness of the two clusters, Ci and Cj . It is defined as



where SEC{Ci ,Cj } is the average weight of the edges that connect vertices in Ci to vertices in Cj , and SECCi (or SECCj ) is the average weight of the edges that belong to the mincut bisector of cluster Ci (or Cj).

**OUTLIERS AND OUTLIER ANALYSIS**

Outliers:



**Types of Outliers**

**Global Outliers**

In a given data set, a data object is a global outlier if it deviates significantly from the rest of the data set. Global outliers are sometimes called point anomalies, and are the simplest type of outliers. Most outlier detection methods are aimed at finding global outliers.

**Contextual Outliers**

In a given data set, a data object is a contextual outlier if it deviates significantly with respect to a specific context of the object.

Contextual outlier detection is divided into two groups:

**Contextual attributes:** The contextual attributes of a data object define the object’s context. In the temperature example, the contextual attributes may be date and location.

**Behavioral attributes:** These define the object’s characteristics, and are used to evaluate whether the object is an outlier in the context to which it belongs. In the temperature example, the behavioral attributes may be the temperature, humidity, and pressure.

**Collective Outliers**

Given a data set, a subset of data objects forms a collective outlier if the objects as a whole deviate significantly from the entire data set. Importantly, the individual data objects may not be outliers.

**Challenges of Outlier Detection**

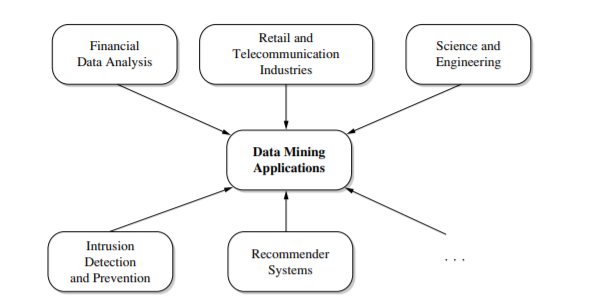
**Modeling normal objects and outliers effectively.** Outlier detection quality highly depends on the modeling of normal objects and outliers.

**Application-specific outlier detection.** Technically, choosing the similarity/distance measure and the relationship model to describe data objects is critical in outlier detection.

**Handling noise in outlier detection**

**Understandability.**

DATA MINING APPLICATIONS



**Data Mining for Financial Data Analysis**

* Design and construction of data warehouses for multidimensional data analysis and data mining
* Loan payment prediction and customer credit policy analysis
* Classification and clustering of customers for targeted marketing
* Detection of money laundering and other financial crimes

Data Mining for Retail and Telecommunication Industries

* Design and construction of data warehouses
* Multidimensional analysis of sales, customers, products, time, and region
* Analysis of the effectiveness of sales campaigns
* Customer retention—analysis of customer loyalty
* Product recommendation and cross-referencing of items
* Fraudulent analysis and the identification of unusual patterns

Data Mining in Science and Engineering

* Data warehouses and data preprocessing
* Mining complex data types
* Graph-based and network-based mining
* Visualization tools and domain-specific knowledge

Data Mining for Intrusion Detection and Prevention

* Signature-based detection
* Anomaly-based detection
* New data mining algorithms for intrusion detection
* Association, correlation, and discriminative pattern analyses help select and build discriminative classifiers
* Analysis of stream data
* Distributed data mining
* Visualization and querying tools

**Data Mining and Recommender System**

**Recommender systems** help consumers by making product recommendations that are likely to be of interest to the user.

**The content-based approach** recommends items that are similar to items the user preferred or queried in the past. It relies on product features and textual item descriptions.

**The collaborative approach** (or collaborative filtering approach) may consider a user’s social environment

**MINING TEXT DATA**

Text mining is an interdisciplinary field that draws on information retrieval, data mining, machine learning, statistics, and computational linguistics. A substantial portion of information is stored as text such as news articles, technical papers, books, digital libraries, email messages, blogs, and web pages

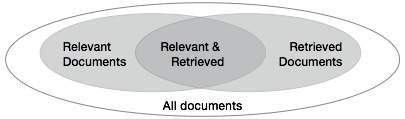
## **Information Retrieval**

Information retrieval deals with the retrieval of information from a large number of text-based documents. Some of the database systems are not usually present in information retrieval systems because both handle different kinds of data. Examples of information retrieval system include −

* Online Library catalogue system
* Online Document Management Systems
* Web Search Systems etc.

## **Basic Measures for Text Retrieval**

to check the accuracy of a system when it retrieves a number of documents on the basis of user's input. Let the set of documents relevant to a query be denoted as {Relevant} and the set of retrieved document as {Retrieved}. The set of documents that are relevant and retrieved can be denoted as {Relevant} ∩ {Retrieved}. This can be shown in the form of a Venn diagram as follows −



There are three fundamental measures for assessing the quality of text retrieval −

* Precision
* Recall
* F-score

### **Precision**

Precision is the percentage of retrieved documents that are in fact relevant to the query. Precision can be defined as −

Precision= |{Relevant} ∩ {Retrieved}| / |{Retrieved}|

### **Recall**

Recall is the percentage of documents that are relevant to the query and were in fact retrieved. Recall is defined as −

Recall = |{Relevant} ∩ {Retrieved}| / |{Relevant}|

### **F-score**

F-score is the commonly used trade-off. The information retrieval system often needs to trade-off for precision or vice versa. F-score is defined as harmonic mean of recall or precision as follows −

F-score = recall x precision / (recall + precision) / 2

# Mining World Wide Web

**web mining** can be organized into three main areas: web content mining, web structure mining, and web usage mining

**Web content mining**

Web content mining analyses web content such as text, multimedia data, and structured data. This is done to understand the content of web pages, provide scalable and informative keyword-based page indexing, entity/concept resolution, web page relevance and ranking, web page content summaries, and other valuable information related to web search and analysis. Web pages can reside either on the surface web or on the deep Web. The surface web is that portion of the Web that is indexed by typical search engines. The deep Web (or hidden Web) refers to web content that is not part of the surface web

**Web structure mining**

Web structure mining is the process of using graph and network mining theory and methods to analyze the nodes and connection structures on the Web. It extracts patterns from hyperlinks, where a hyperlink is a structural component that connects a web page to another location. It can also mine the document structure within a page (e.g., analyze the treelike structure of page structures to describe HTML or XML tag usage).

**Web usage mining**

Web usage mining is the process of extracting useful information (e.g., user click streams) from server logs. It finds patterns related to general or particular groups of users; understands users’ search patterns, trends, and associations; and predicts what users are looking for on the Internet.

## **Challenges in Web Mining**

The web poses great challenges for resource and knowledge discovery based on the following observations −

* **The web is too huge** − The size of the web is very huge and rapidly increasing. This seems that the web is too huge for data warehousing and data mining.
* **Complexity of Web pages** − The web pages do not have unifying structure. They are very complex as compared to traditional text document. There are huge amount of documents in digital library of web. These libraries are not arranged according to any particular sorted order.
* **Web is dynamic information source** − The information on the web is rapidly updated. The data such as news, stock markets, weather, sports, shopping, etc., are regularly updated.
* **Diversity of user communities** − The user community on the web is rapidly expanding. These users have different backgrounds, interests, and usage purposes. There are more than 100 million workstations that are connected to the Internet and still rapidly increasing.
* **Relevancy of Information** − It is considered that a particular person is generally interested in only small portion of the web, while the rest of the portion of the web contains the information that is not relevant to the user and may swamp desired results.

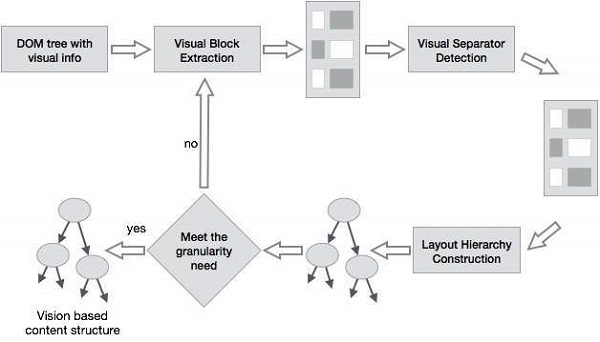
## **Mining Web page layout structure**

The basic structure of the web page is based on the Document Object Model (DOM). The DOM structure refers to a tree like structure where the HTML tag in the page corresponds to a node in the DOM tree. We can segment the web page by using predefined tags in HTML.

**Vision-based page segmentation (VIPS)**

* The purpose of VIPS is to extract the semantic structure of a web page based on its visual presentation.
* Such a semantic structure corresponds to a tree structure. In this tree each node corresponds to a block.
* A value is assigned to each node. This value is called the Degree of Coherence. This value is assigned to indicate the coherent content in the block based on visual perception.
* The VIPS algorithm first extracts all the suitable blocks from the HTML DOM tree. After that it finds the separators between these blocks.
* The separators refer to the horizontal or vertical lines in a web page that visually cross with no blocks.
* The semantics of the web page is constructed on the basis of these blocks.

The following figure shows the procedure of VIPS algorithm



## **Trends in Data Mining**

Data mining concepts are still evolving and here are the latest trends that we get to see in this field −

* Application Exploration.
* Scalable and interactive data mining methods.
* Integration of data mining with database systems, data warehouse systems and web database systems.
* Standardization of data mining query language.
* Visual data mining.
* New methods for mining complex types of data.
* Biological data mining.
* Data mining and software engineering.
* Web mining.
* Distributed data mining.
* Real time data mining.
* Multi database data mining.
* Privacy protection and information security in data mining.