CLUSTER ANALYSIS OF DECOMPRESSION ILLNESS

(DEKOMPRESYON HASTALIĞININ CLUSTER ANALİZİ)

by

Barış Aksoy, B.S

Thesis

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

in the

INSTITUTE OF SCIENCE AND ENGINEERING

of

GALATASARAY UNIVERSITY

June 2009

CLUSTER ANALYSIS OF DECOMPRESSION ILLNESS

(DEKOMPRESYON HASTALIĞININ CLUSTER ANALİZİ)

by

Barış Aksoy, B.S

Thesis

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Date of Submission : May 15, 2009

Date of Defense Examination : June 15, 2009

Supervisors : Dr. Vincent LABATUT

Asst. Prof. Dr. S. Murat EGİ

Committee Members : Prof. Dr. Hocine CHERIFI

Assoc. Prof. Dr. Esra ALBAYRAK

Asst. Prof. Dr. Burak ARSLAN

ACKNOWLEDGEMENTS

I would like to thank my supervisors Ass. Prof. Vincent Labatut and Ass. Prof. Murat Egi for their kind support during the development of the ideas of this thesis.

I would also like to thank Tamer Özyiğit for his helpful comments on my thesis and DAN Europe and Divers Alert Network who provided us the data.

And lastly I'd like to thank to Pınar, Seda, my parents, and my friends at Anadolu Hayat Emeklilik for their encouragements and for their understanding.

Barış Aksoy,

İstanbul, May 15th, 2009

TABLE OF CONTENTS

LIST OF TABLES

ABSTRACT

There have been many classifications of Decompression Illness (DCI) that is seen in divers as the result of bubbles which expand in human body causing local damage in tissues or which block blood circulation because of decompression.

The diagnosis and classification of DCI is made observing the patient's symptoms and signs. The treatment is performed in a hyperbaric chamber where the conditions are reversed (recompression) and the combination of pressure and time is determined by the type of the disease.

The problem is that DCI has a lot of signs and symptoms, resulting in a lot of different classifications of the illness requiring different treatment plans. Sometimes, the treatment is to be initiated by the chamber operators without the presence of a doctor, which makes the correct classification of DCI extremely important and data mining techniques can be used as decision support tools to determine the type of DCI.

In this thesis we classified empirically the DCI patients using the sign and symptom list of the Diving Injury Reporting Forms (DIRF) of Divers Alert Network with different clustering algorithms (k-means algorithm, COBWEB algorithm, and EM algorithm) and compared our results with recent statistical studies on DCI classification and other classifications and outcome of treatment. And we have also found association rules which will contribute differential diagnosis.

Consequently, the classes we have obtained after clustering have the characteristics of hierarchy from mild to severe as in other classifications and as in recent classifications of DCI.

RESUME

Il y a différents types de classification de la maladie de décompression (MDC) observée chez les plongeurs et causée par les bulles qui se dilatent dans le corps humain. Elles causent des dommages locaux ou dans les tissus, ce qui bloque la circulation du sang.

Le diagnostic et la classification de la MCD est fait en observant les symptômes du patient. Le traitement est effectué dans une chambre hyperbare où les conditions sont inversées (récompression) et la combinaison de pression et de temps est déterminée par le type de la maladie.

Le problème est que la MDC présente beaucoup de symptômes, ce qui cause différentes classifications de la maladie nécessitant des traitements différents. Parfois, le traitement doit être initié par les opérateurs de la chambre hyperbare sans la présence d'un médecin, ce qui rend la classification de la DCI extrêmement importante. Les techniques de fouilles de données peuvent être utilisées comme outils d'aide à la décision pour déterminer le type de MDC.

Dans cette thèse, nous avons classifié empiriquement les patients atteints de la MDC en implémentant différents algorithmes de clustering (k-moyenne, le COBWEB, l'EM) sur la liste de symptômes décrits dans les formulaires de déclaration des blessures de plongée de Divers Alerts Network (DAN). Nous avons réalisé une étude statistique de la classification de la MCD pour comparer nos résultats avec d'autres classifications et avec les résultats du traitement. Nous avons aussi cherché des règles d'association afin d'aider à la réalisation d'un diagnostic différentiel.

Les classes que nous avons obtenues après le clustering ont pour caractéristique de respecter la hiérarchie des symptômes légers à graves rencontrée dans les classifications manuelles classiques et dans les classifications automatiques récentes

ÖZET

Dekompresyon nedeniyle insan vücudunda dokularda lokal zararlara yol açan veya kan dolaşımını engelleyen hava kabarcıkları sebebiyle dalgıçlarda görülen Dekompresyon hastalığının çeşitli sınıflandırmaları mevcuttur

Dekompresyon hastalığının teşhisi ve sınıflandırması hastanın farklı belirti ve bulgularının değerlendirilmesi ile yapılır. Tedavisi basınç odalarında yapılır ve koşulların hastalığın tipine göre belirlenen basınç ve zamanın ters çevrilmesiyle (rekompresyon) yapılır.

Dekompresyon hastalığının birçok belirti ve bulguya sahip olması farklı sınıflandırmalara ve dolayısıyla farklı tedavi şekillerine yol açmaktadır.

Kimi zaman, doktor olmadığı ortamlarda tedavi basınç odası operatörleri tarafından başlatılmaktadır ve bu da dekompresyon hastalığının doğru sınıflandırılmasının ne kadar önemli olduğunu ve veri madenciliği tekniklerinin karar destek aracı olarak hastalığın tipini belirlemede kullanılabileceğini göstermektedir.

Bu tez çalışmasında farklı clustering algoritmaları (k-ortalama, COBWEB, EM) ile Divers Alert Network(Dalgıçların Acil Durum Ağı)'nın dalış yaralanmaları bildirim formlarından elde ettiğimiz belirti ve bulgu listelerini kullanarak dekompresyon hastalığını sınıflandırdık ve sonuçlarımızı klasik sınıflandırma yöntemleri, yeni yapılan istatiksel sınıflandırma yöntemleri ve tedavi sonuçları ile karşılaştırdık. Ayrıca teşhiste yardımcı olabilecek birliktelik kuralları (association rules) elde ettik.

Sonuç olarak, clustering yöntemleriyle elde ettiğimiz sınıfların yeni yapılan istatistiksel sınıflandırmalarla ve klasik sınıflandırmalarla uyumlu olduğunu ve hafiften şiddetli vakalara giden hiyerarşik yapıda olduğunu gözlemledik.

1. INTRODUCTION

Decompression illness (DCI) is seen in divers as the result of bubbles which expand in human body causing local damage in tissues or which block blood circulation because of decompression.

The main risk factor for DCI is a reduction in ambient pressure, but there are other risk factors that will increase the chance of DCI occurring. These known risk factors are deep long dives, cold water, hard exercise at depth, and rapid ascents. Decompression illness affects scuba divers, aviators, astronauts and compressed-air workers. It occurs in approximately 1,000 U.S. scuba divers each year. Since DCI is a random event, almost any dive profile can result in DCI, no matter how safe it seems. The reason is that the risk factors, both known and unknown, can influence the probability of DCI in myriad ways. Because of this, evaluation of a diver for possible decompression illness must be made on a case-by-case basis by evaluating the diver's signs and symptoms and not just based on the dive profile [1].

The diagnosis and classification of DCI is made observing the patient's symptoms and signs. The treatment is performed in a hyperbaric chamber where the conditions are reversed (recompression) and the combination of pressure and time is determined by the type of the disease.

There are several classifications of DCI each of them presenting the DCI with different classes and there are also recent studies on classification of DCI with statistical methods.

The problem is that DCI has a lot of signs and symptoms, resulting in a lot of different classifications of the illness requiring different treatment plans. Sometimes, the treatment is to be initiated by the chamber operators without the presence of a doctor,

which makes the correct classification of DCI extremely important. Data mining techniques can be used as decision support tools to determine the type of DCI.

In this work our goal is to classify empirically the DCI patients using the signs and symptoms list of the Diving Injury Reporting Forms (DIRF) of DAN with different clustering algorithms (k-means algorithm, COBWEB algorithm, and EM algorithm) and to compare our results with recent statistical studies on DCI classification. Another goal is to find decision rules which will contribute differential diagnosis.

In the following pages we will first describe the Decompression Illness, giving the causes, symptoms, different classifications in literature and treatment methods. We will then give a general definition of data mining, explain the processes of data mining and will be focusing on the tasks and algorithms which will be used later. Then we will introduce classification and clustering. We will start by a general definition and then explain the clustering algorithms that we have used (k-means algorithm, COBWEB algorithm, and EM algorithm) for classification. After this phase we will continue with association rules and will introduce the *A priori* algorithm that we used to find out the relations within the data we have used. Application of these algorithms to DCI is the section which will be following the explanation of the algorithms. In that section, we will give a definition of the data we have used which is a collection on scuba diving injuries from hyperbaric chambers all around the world. Then we will give the results of clustering by k-means algorithm, COBWEB algorithm, and EM algorithm and compare them with the classical classification of DCI and with the recent studies on classification of DCI using statistical analysis which strengthens the strong relationship among the clusters of different classification algorithms and classifications. Another section is the implementation of *A priori* algorithm to DCI data and presentation of the association rules found. We will end with conclusion section where we will comment the results.

2. DECOMPRESSION ILLNESS

Decompression illness (DCI) is a term used to describe illness that results from a reduction in the ambient pressure surrounding a body. DCI encompasses two diseases, decompression sickness (DCS) and arterial gas embolism (AGE). DCS is thought to result from bubbles growing in tissue and causing local damage, while AGE results from bubbles entering the lung circulation, traveling through the arteries and causing tissue damage at a distance by blocking blood flow at the small vessel level [1].

Bubble damage causes the signs and symptoms of DCI. The bubbles appear following the reduction in ambient pressure as the diver ascends towards the surface. They can be intravascular or extravascular. The former can originate from pulmonary barotrauma or from the release of excess dissolved gas. The latter are also thought to originate from the release of excess dissolved gas and are particularly associated with more severe decompression stress in the presence of a significant inert gas load. Whatever the location and source of the bubbles, if present in quantities to cause sufficient damage, they can cause a clinical phenomenon known as acute DCI. The bubbles can act as emboli causing ischemia, they can injure the tissues within which they appear and they can act as foreign bodies that damage vascular endothelium, disrupt the blood–brain barrier and initiate pathophysiological processes such as the complement cascade. Even after the bubbles have been cleared from the vasculature or resorbed from the tissues they can leave residual damage that later causes vasospasm, reperfusion injury, thrombotic deposits, extravasation of blood components, inflammatory changes, and release of locally and systemically active substances. The wide range of mechanisms of injury and the fact that target tissues are not necessarily restricted to normal anatomical boundaries means that there are many manifestations and patterns of presentation. The precise targets of injury and the amount of gas involved will dictate whether the DCI becomes a trivial problem or a life threatening multi-system disorder [2].

There are several classifications of DCI. According to Benton & Glover [2] there are seven classes of manifestations: limb pain, neurological, vestibular, cardiopulmonary, cutaneous, lymphatic and constitutional (non specific symptoms). Golding et al. [3] classified decompression illness as Type I, which are the cases exhibiting only pain, and Type II which have the characteristics of neurological manifestations, abnormal physical signs and pain. This sub-classification has been used by the US Navy as a guide to diagnosis and treatment of decompression illness [3]. Buch et al. [4] made a classification according to the clinical severity: mild, moderate and severe. Diver's Alert Network's (DAN) Perceived Severity Index (PSI) [5] has six classes from most severe to least severe, based on physicians' diagnosis: serious neurological, cardiopulmonary, mild neurological, pain, lymphatic/skin and constitutional/non-specific.

Ozyigit et al. [6, 7] implemented Ward's method [8] and two-step cluster analysis [9] on Medical Reports of the SSS (Sub-aquatic Safety Services) Recompression Chamber Network and DAN Europe Diving Injury Reports. Their study is a milestone as it is the first attempt of using clustering techniques on classification of DCI.

Whatever the classification is, a diver with DCI must be recompressed at the earliest available opportunity, especially if the symptoms are severe or the disease is progressive. More severe DCI can be expected to develop in cases that present with pain in the distribution of a thoracolumbar dermatome (known as girdle pain) and in a proportion of cases with cutaneous manifestations where the skin develops a marbled discoloration and becomes tender (known as cutis marmorata). First aid measures differ little from any other medical emergency. Basic life support considerations take priority: Airway, Breathing and Circulation. Rehydration is also considered to be very important. Intravenous administration is indicated if the airway is unprotected or if oral fluids cannot be tolerated for any other reason. Otherwise the casualty should be encouraged to drink copious clear fluids.

Definitive treatment for DCI is recompression. There are many recompression tables, each requiring different combinations of pressure and time. Both pressure and oxygen have beneficial effects. Pressure will reduce bubble size and will also reduce or

reverse the pressure gradient that is encouraging free gas to leave the tissues and form new bubbles or cause existing ones to enlarge. Pressure alone will never completely eliminate a bubble but this can be achieved by the oxygen-accelerated diffusion.

Some 55% of cases of DCI resolve completely after the first recompression treatment. 75% will resolve on completion of all therapies required. Twenty five percent, however, will be left with some form of deficit which might require considerable medical care or rehabilitation in future [2].

According to 2002 DCI Report, over 50% of the divers received only one hyperbaric treatment and the mean was more than 2 and the highest number was 14. Half of the injured divers had complete relief after the initial recompression, and 43 percent were improved. Only 6.7 percent had no improvement. Ninety-six percent of injured divers had resolved by six months, 98 percent by nine months, and 99 percent by 12 months. The remainder reported improvement [5].

In this section we defined DCI, presented different classifications of the illness, gave a brief explanation of treatment of the illness and gave the statistics on the success of treatment. We will continue with a brief definition of data mining that we have used to cluster DCI.

3. DATA MINING

3.1. GENERAL DEFINITION

"The amount of data in the world goes on increasing every day. As the volume of data increases, inexorably, the proportion of it that people understand decreases. Lying hidden in all this data is information that is rarely made explicit or taken advantage of. So this big need of finding out the hidden patterns in data which give birth to a new domain called data mining. Data mining is defined as the process of discovering patterns in data" [10]. And data is consisted of elements called *instances* which are characterized by the values of *attributes* which measure different aspects of instances. For example a customer is an instance of a data set which has different attributes such as age, sex, salary. There are different types of attributes such as *nominal* (values that represent categories with no intrinsic ranking such as place of birth), *ordinal* (values represent categories with some intrinsic ranking such as economic status: low, medium, high), *interval* (values represent ordered categories with a meaningful metric such as salary in thousand of liras), *numeric* (such as age=25), *boolean* (values TRUE, FALSE for an attribute. As we stated above there is a big need for extracting information from data in different domains from banking to scientific researches and data mining is in the center for solving this need. It is used in many different areas such as in marketing, fraud detection, manufacturing and science. The process of data mining has 3 phases:

1. Preprocessing or data preparation
2. The actual mining
3. Interpretation of the results

Preprocessing or data preparation: In this phase, data is prepared for processing. Preparing input for a data mining investigation usually consumes the bulk of the effort invested in the entire data mining process. When beginning work on a data mining problem, it is first necessary to bring all the data together into a set of instances. Then

the data must be assembled, integrated, and cleaned up [10]. The analyzer must also take decisions on handling of missing data. Another process may be changing the format of the data if it is required for the data mining tool.

The actual mining: It is the phase where we can apply different algorithms which involve mainly 4 tasks:

1. *Clustering* is a common descriptive task where one seeks to identify a finite set of categories or clusters (a subset of the data set that groups similar instances together) to describe the data [11]. In the clustering problem, we group similar instances together. This creates segments of the data which have considerable similarity within a group of points. Depending upon the application, each of these clusters may be treated differently. For example, in image and video databases, clustering can be used to detect interesting spatial patterns and features and support content based retrievals of images and videos using low-level features such as texture, color histogram, shape descriptions, etc. In insurance applications, the different clusters may represent the different demographic segments of the population each of which have different risk characteristics, and may be analyzed separately [12].

2. *Classification* consists in learning a function that maps (classifies) a data instance into one of several predefined cluster[11]. In the classification problem, the attributes are divided into two categories: a multiplicity of feature attributes, and a single class label. The training data is used in order to model the relationship between the feature attributes and the class label. This model is used in order to predict the class label of a test example in which only the feature attributes are known [12].

3. *Association rules* searches for relationships between different variables. This task often occurs in the process of finding relationships between different attributes in large customer databases. The idea in the association rule problem is to find the nature of the causalities between the values of the different

attributes[12].

4. *Regression* searches for the best function that models the data which can be used for prediction. It consists in learning a function that maps a data item to a real-valued prediction variable. Regression applications are many, for example, predicting the amount of biomass present in a forest given remotely sensed microwave measurements, estimating the probability that a patient will survive given the results of a set of diagnostic tests, predicting consumer demand for a new product as a function of advertising expenditure, and predicting time series where the input variables can be time-lagged versions of the prediction variable [11].

Interpretation of the results: The found patterns are evaluated in this phase. It is the phase where happens the acting on the discovered knowledge: One can use the knowledge directly, or it can be taken into another system for further action such as changing the format, presenting it differently or implementing different analysis, or it can be simply documented and reported it to interested parties for their interpretation. This process also includes checking for and resolving potential conflicts with previously believed (or extracted) knowledge. This phase can also involve visualization of the extracted patterns and models or visualization of the data given the extracted models [11].

In our work described in section 3, we will use clustering for unsupervised classification of data and association rules to find relationships between variables of data. For this reason, the rest of this section will consist in further definition of clustering and association rules.

3.2. CLASSIFICATION AND CLUSTERING

Classification consists in partitioning a set of instances into several classes. The classification quality depends on the intra-class and extra-class similarities: the first should be high, whereas the latter should be low. In other terms: two instances of the same class should be very similar, while two instances from different classes should be different.

From a more formal point of view [13], we can consider an instance x defined by n real attributes to be a value defined on R^n where R stands for the set of all real attributes. A data set X containing m instances would then be defined on $R^{m \times n}$ as $x = (x_1, \ldots, x_m)$ where each x_i represents an instance. A classification consists in partitioning X in k subsets C_j called classes.

The partition is processed thanks to a classification algorithm. This algorithm implements an injective mapping $X \rightarrow \{C_J\}$ of data set X to classes C_j. As stated in [10] "These classes should reflect some mechanism at work in the domain from which instances or data points are drawn, a mechanism that causes some instances to bear a stronger resemblance to one another than they do to the remaining instances."

When we don't have data for testing the k-fold cross validation method is used. In the k-fold cross validation method, the original data is partitioned into k subsets. One set is left as validation data and validated on the remaining $k - 1$ subsets as they are used as training data. This process is repeated *k times (*$k = 10$ which give the best results in general [14]) so that each subset is used once as the validation data[15].

Once the partition has been defined, it can be used on new data (i.e. other than X). In *supervised classification*, the actual class of each instance in X is known and used by the algorithm. In *non-supervised classification*, also called *clustering*, the actual classes (also called clusters) are unknown. The problem of clustering data points can be defined as follows: Given a set of points in multidimensional space, find a partition of the points into clusters so that the points within each cluster are close to one another [11]. Clustering has usage in many data mining applications such as segmentation, medical diagnostic, web analysis, computational biology, etc. In this work, we use clustering to process DCI data (that we will present below), because we do not know the actual nature of the decompression problems.

3.2.1. K-Means Clustering

The k-means method is a widely used geometric clustering algorithm based on the article proposed by Lloyd in 1982 [16]. Given a set of n data points, the algorithm uses a local search approach to partition the points into k clusters. A set of k initial

cluster centers is chosen arbitrarily. Each point is then assigned to the center closest to it, and the centers are recomputed as centers of mass of their assigned points. This is repeated until the process stabilizes. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early grouping is done. At this point we need to re-calculate k new centroids of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not change any more.

Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function is

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} \left| x_n - \mu_j \right|^2 \qquad (3.1)$$

where x_n is a vector representing the *n*th data point and μ_j is the geometric centroid of the data points in S_j which are disjoint subsets. The algorithm works as follows:

1. *Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.*

2. *Assign each object to the group that has the closest centroid.*

3. *When all objects have been assigned, recalculate the positions of the K centroids*

4. *Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated* [17].

It can be shown that no partition occurs twice during the course of the algorithm, and so the algorithm is guaranteed to terminate. The k-means method is still very popular today, as it is easy to implement to data from different domains which gives successful results, and it has been applied in a wide variety of areas ranging from computational biology to computer graphics [9].

One of the shortcomings of the k-means algorithm is the necessity to specify the number of clusters. Another is the impact of the high dimensionality on the performance of k-means. The traditional euclidean notion of proximity is not very effective for k-means on high-dimensional data sets, such as gene expression data sets and document data sets. And also outliers and noise in the data can degrade the performance of this algorithm [15].

3.2.2. COBWEB Algorithm

Unlike the k-means algorithm which iterates over the whole dataset, the COBWEB Algorithm [18] incrementally incorporates objects into a classification tree. At any stage the clustering forms a tree with instances at the leaves and a root node that represents the entire data set. In the beginning the tree consists of the root alone. Instances are added one by one, and the tree is updated appropriately at each stage. Updating may merely be a case of finding radical restricting of the part of the tree that is affected by the new instance. The key to deciding how and where to update is a quantity called *category utility* which measures the overall quantity of a partition of instances. Category utility works both for nominal and numeric attribute (based on an estimate of mean and standard deviation of the value of an attribute). When estimating the standard deviation of an attribute for a particular node, the result will be zero if the node contains only one instance. Zero variances produce infinite values in the category utility formula. A simple heuristic solution is to impose a minimum variance on each attribute. It can be argued that since no measurement is completely precise, it is reasonable to impose such a minimum: it represents the measurement error in a single sample. This parameter is called *acuity*. Another parameter, *cutoff* is used to suppress growth. Some instances are deemed sufficiently similar to others not to warrant formation of their own child, and this parameter governs the similarity threshold. Cutoff is specified in terms of category utility: when the increase in

category utility from adding a new node is sufficiently small, that node is cut off. And the definition of category utility is:

$$CU(C_1, C_2, \dots, C_k) = \frac{\sum_l \Pr[C_l] \sum_i \sum_j (\Pr[a_i = v_{ij} | C_l]^2 - \Pr[a_i = v_{ij}]^2}{k} \quad (3.2)$$

where $C_1, C_2, \dots, C_k$ are the k clusters; the outer summation is over the clusters; the next inner one sums over the attributes; a_i is the ith attribute, and it takes on values $v_{i1}, v_{i2}, \dots$ which are dealt with by the sum over j. The probabilities themselves are obtained by summing over all instances; thus there is a further implied level of summation.

The point of having a cluster is that it will give some advantage in predicting the values of attributes of instances in that cluster, that is

$$\Pr[a_i = v_{ij} | C_l] \quad (3.3)$$

is a better estimate of the probability that attribute a_i has value v_{ij}, for an istance in cluster C_l, than the probability

$$Pr[a_i = v_{ij}] \quad (3.4)$$

because it takes account of what cluster the instance is in. So what the above measure calculates, inside the multiple summations, is the amount by which that information does help, in terms of differences between squares of probabilities. This measure sums the difference of squares and the differences between squares of probabilities are summed over all clusters, weighted by their probabilities, in the outer summation.

The overall division by k provides a "per cluster" figure for the category utility which discourages overfitting. And this formula can be extended to numeric attributes by assuming their distribution is normal, with a given (observed) mean μ and standard deviation σ. So the category utility formula becomes

$$CU(C_1, C_2, \dots, C_k) = \frac{1}{k} \sum_l \Pr[C_l] \frac{1}{2\sqrt{\pi}} \sum_i \left(\frac{1}{\sigma_{il}} - \frac{1}{\sigma_i}\right) \quad (3.5)$$

where σ_i is the standard deviation of the attribute a_i. The need for the parameter acuity becomes apparent: a zero standard deviation produces an infinite value of category utility formula. Imposing a prespecified minimum variance on each attribute, the acuity, is a rough-and-ready solution to the problem [10].

The incorporation of an object is a process of classifying the object by descending the tree along an appropriate path, updating counts along the way, and performing one of several operators at each level. These operators include:

- Classifying the object with respect to an existing class
- Creating a new class
- Combining two classes into a single class (merging)
- Dividing a class into several classes (splitting)

The algorithm works as follows:

COBWEB (Object, Root)

1) Update counts of the Root
2) *IF* Root is a leaf

   THEN

   Return the expanded leaf to accommodate the new object

   ELSE

   Find that child of Root that best hosts Object

   and *perform one of the following*

   a) Consider *creating a new class* and do so if appropriate

   b) Consider *node merging* and do so if appropriate

   And call COBWEB (Object, Merged node)

   c) Consider node *splitting* and do so if appropriate

   And call COBWEB (Object, Root)

   d) IF none of the above (a, b or c) were performed

3.2.3. EM Algorithm

Some of the shortcomings of the heuristic clustering described in COBWEB algorithm are: the arbitrary division by k in the category utility formula which is necessary to prevent overfitting, the need to supply an artificial minimum value for the standard deviation of clusters, the *ad hoc* cutoff value to prevent every single instance from becoming a cluster in its own right.

A more principled statistical approach can overcome some of these shortcomings. From a probabilistic perspective, the goal of clustering is to find the most likely set of clusters given the data (and, inevitably, prior expectations). Because no finite amount of evidence is enough to make a completely firm decision on the matter, instances- even training instances- should not be placed categorically in one cluster or the other: instead they have certain probability of belonging to each cluster. This helps to eliminate the brittleness that is often associated with schemes that make hard and fast judgments.

The foundation for statistical clustering is a statistical model called *finite mixtures*. A mixture is a set of k probability distributions, representing k clusters that govern the attribute values for members of that clusters. In other words, each distribution gives the probability that a particular instance would have a certain set of attribute values if it were known to be a member of that cluster. Each cluster has a certain distribution. Any particular instance belongs to one and only one of the clusters, but it is not known which one. Finally, the clusters are not equally likely: there is some probability distribution that reflects their relative populations.

The simplest finite mixture situation is when there is only one numeric attribute, which has Gaussian or normal distribution for each cluster but with different means and variances. The clustering problem is to take a set of instances and a prespecified number of clusters, and workout each cluster's mean and variance and the population distribution between the clusters. This approach is the main idea of the Expectation-Maximization Algorithm[10].

The Expectation-Maximization (EM) algorithm [19] is a statistical model that makes use of the finite Gaussian mixtures mode. The algorithm is similar to the k-means procedure in that a set of parameters are re-computed until a desired convergence value is achieved.

A mixture is a set of N probability distributions where each distribution represents a cluster. An individual instance is assigned a probability that it would have a certain set of attribute values given it was a member of a specific cluster.

In the simplest case $N = 2$, the probability distributions are assumed to be normal and data instances consist of a single real-valued attribute. Using this scenario, the job of the algorithm is to determine the value of five parameters, specifically:

1. The mean and standard deviation for cluster 1
2. The mean and standard deviation for cluster 2
3. The sampling probability P for cluster 1 (the probability for cluster 2 is $1 - P$)

Here is the general procedure:

1. Guess initial values for the five parameters.
2. Use the probability density function for a normal distribution to compute the cluster probability for each instance.
3. Use the probability scores to re-estimate the five parameters.
4. Return to Step 2.

The algorithm terminates when the calculation of the distribution parameters (which is maximization of the likelihood of the distributions given the data) that measures cluster quality no longer shows significant increases. One measure of cluster quality is the likelihood that the data came from the dataset determined by the clustering. The likelihood computation is simply the multiplication of the sum of the probabilities for each of the instances.

The number of clusters could be set by the user or it can be done by the algorithm. Other parameters are the minimum allowable standard deviation, maximum number of iterations for the algorithm.

3.3. ASSOCIATION RULES

An *association* is the relationship of items in a transaction in such a way that items imply the presence of other items in the same transaction. Agrawal et al. [20] introduced *association rules* by presenting an algorithm which generates significant rules involving items in a large database of customer transactions. An example of association rule is: 80% of customers who buy beef and onions also buy coke. So the main idea is to predict the outcome using any attribute by constructing an association rule. Because so many different association rules can be derived from even a tiny

dataset, interest is restricted to those applying to a reasonably large number of instances, and having a reasonably high accuracy on the instances they apply to.
The coverage of an association rule is the number of instances for which it predicts correctly is called *support.* And its accuracy -often called *confidence*- is the number of instances correctly predicted, expressed as a proportion of all instances it applies to [10]. It may be formulated using the association rule "if X then Y" such as:

$$Confidence(if\ X\ then\ Y) = \frac{Support(X\ and\ Y)}{Support\ (X)} \tag{3.6}$$

The problem is usually decomposed into two sub problems. One is to find itemsets whose occurrences exceed a predefined threshold (minimum support) in the database. Those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets is L_k,

$$L_k = \{ I_1, I_2, \dots , I_{k-1} \} \Rightarrow \{I_k\} \tag{3.7}$$

By checking the confidence, this rule can be determined as interesting or not. Then other rules are generated by deleting the last items in the antecedent and inserting it to the consequent, further the confidences of the new rules are checked to determine the interestingness of them. This process is iterated until the antecedent becomes empty [21].

The formal definition of *A priori* [20] algorithm is as follows: Let

$$I = \{ I_1, I_2, \dots , I_m \} \tag{3.8}$$

be a set of binary attributes, called items. Let T be a database of transactions. Each transaction t is represented as a binary vector, with $t_k = 1$ if t bought the item I_k, and $t_k = 0$ otherwise. There is one tuple in the database for each transaction. Let X be a set of some items in I. We say that a transaction t satisfies X if for all items I_k in X, $t_k = 1$. By an association rule, we mean an implication of the form $X \Rightarrow I_j$, where X is a set of some items in I, and I_j is a single item in I that is not present in X. The rule X => I_j, is satisfied in the set of transactions T with the *confidence factor* $0 \leq c \leq 1$ if at least c% of transactions in T that satisfy X also satisfy I_j . We will use the notation $X \Rightarrow I_j \mid c$ to specify that the rule $X \Rightarrow I_j$ has a *confidence factor* of c.

Given the set of transactions T, we are interested in generating all rules that satisfy certain additional constraints of two different forms:

1. Syntactic Constraints: These constraints involve restrictions on items that can appear in a rule. For example, we may be interested only in rules that have a specific item I_x appearing in the consequent or rules that have a specific item I_y appearing in the antecedent. Combinations of the above constraints are also possible - we may request all rules that have items from some predefined item set X appearing in the consequent, and items from some other item set Y appearing in the antecedent.

2. Support Constraints: These constraints concern the number of transactions in T that support a rule. The support for a rule is defined to be the fraction of transactions in T that satisfy the union of items in the consequent and antecedent of the rule. In this formulation, the problem of rule mining can be decomposed into two sub problems:

a. Generate all combinations of items that have fractional transaction support above a certain threshold, called minimum support. Call those combinations large item sets, and all other combinations that do not meet the threshold small item sets. Syntactic constraints further constrain the admissible combinations. For example, if only rules involving an item I_x in the antecedent are of interest, then it is sufficient to generate only those combinations that contain I_x.

b. For a given large item set

$$Y = \{ I_1 I_2 \ \dots \ I_k \}, k \geq 2, \tag{3.9}$$

generate all rules (at the most k rules) that use items from the set $I_1 I_2 \ \dots \ I_k$. The antecedent of each of these rules will be a subset X of Y such that X has $k - 1$ items, and the consequent will be the item $Y - X$.

To generate a rule $X \Rightarrow I_j \mid c$ where

$$X = I_1 I_2 \dots I_{j-1} I_{j+1} \dots I_k \tag{3.10}$$

and c is the confidence factor, take the support of Y and divide it by the support of X. If the ratio is greater than c then the rule is satisfied with the confidence factor c; otherwise it is not.

3.4. RESULTS ANALYSIS

We have used the correlation and association to find out the relationship among the clusters and among the classifications. For correlation we have used the *Pearson product-moment correlation*:

$$\rho_{XY} = \frac{Cov(X,Y)}{\sqrt{(Var\ X)(Var\ Y)}} \tag{3.11}$$

where ρ_{XY} is the correlation between X and Y where X and Y are random variables with means μ_X and μ_Y and variances σ_X^2 and σ_Y^2 respectively [22].

Association refers to coefficients which gauge the strength of a relationship [23]. We have nominal data such as the cluster names: cluster1, cluster2, we would like to know if knowing the cluster of an instance let us to guess its cluster in another classification. So we have used *Goodman-Kruskal lambda* that its value reflects the percentage reduction in errors in predicting the dependent given knowledge of the independent. This probability is defined as the chance that an observation is in a category other than the most common (modal) one. That is, with no knowledge of the independent, a blind forecaster would guess that each observation of the dependent would have the value of its modal category. Thus the marginal of the modal category is the number of correct guesses one would expect by chance. This forms the denominator of the equation for lambda. The numerator reflects the number of correct guesses knowing the independent variable. Values range from 0 (no association) to 1 (the theoretical maximum possible association) [23]. So if we put all these in formula:

$$\lambda = \frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1} \tag{3.12}$$

where ε_1 is the overall non-modal frequency and ε_2 is the sum of non-modal frequencies for each value of the independent variable. We have used SPSS to compute lambda which print out three versions: a symmetric version, and two asymmetric versions, one with each of the two variables considered as dependent. Lambda symmetric is simply the average of the two asymmetric lambdas.

4. APPLICATION TO DCI

We will introduce our work in three parts:

1. First part: Software and data; where we will give information about the data we used in our study which is retrieved from DAN's database and the software we used for data mining.

2. Second part: Classification, where we will implement three different clustering techniques on the data that we will describe at the very beginning of this section. The algorithms that we will apply on our data are: k-means algorithm, COBWEB algorithm and EM algorithm.

3. Third part: Association rules, where we used the *A priori* algorithm to find out the association rules that may help us to find the relations among the signs and symptoms.

The second and third parts are symmetrical: we will start with the data we have used, continue with the methods, followed by the results and ends with the discussions. Finally, we will end with a conclusion part.

4.1. SOFTWARE

We have used WEKA 3.5.7 as data mining tool. Weka 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. It is also well-suited for developing new machine learning schemes. Weka is open source software issued under the GNU General Public License [24].

In order to work with WEKA, data should be converted to a compatible file format, such as .csv or ARFF (Attribute-Relation File Format). A direct connection to a

database can be established as well. An ARFF file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files have two distinct sections. The first section is the header information, which is followed by the data information.

The header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types [10].

```
@RELATION dirf

@ATTRIBUTE Age  NUMERIC
@ATTRIBUTE Weight   NUMERIC
```

The data of the ARFF file looks like the following:

```
@DATA
25,80
14,70
```

We have also used SPSS 16.0.1 for calculating the associations among the classifications and data transformation. SPSS is software used for statistical analysis. SPSS includes many tools to perform descriptive statistics, prediction for numerical outcomes, bivariate statistics, etc.

4.2. DATA

In our study, we have used DAN's database which hosts the data from the DIRF, which is a collection on scuba diving injuries from hyperbaric chambers all around the world, especially in US, Latin America and Caribbean. The database contains the signs and symptoms of patients and details of the dive accidents.

We have used 1929 DIRFs (1368 males – 561 females) which were collected between 1998 and 2002. The average age of the patients is 37.94 with a range of 13 to 73. We have used twenty-five different signs and symptoms for our analysis which are: unconsciousness, mental problems, pulmonary problems, cardiovascular (CV) signs, pain, skin, lymphatic problems, abnormal sensations, hearing, vision, coordination troubles, muscular weakness, muscular problems, skin sensitivity, bladder-bowel

problems, headache, fatigue, nausea, dizziness, vertigo, paresthesia, tinnitus, numbness, paralysis and weakness.

4.3. CLUSTERING

Classifying patients into clusters is a classical approach, and there have been many studies for clustering in the medical field. Cluster analysis is used mainly for empirical grouping of patients by the signs and symptoms of a disease, but there are also some studies on the classification of symptoms and signs by their observed frequencies on patients [6]. DCI has many signs and symptoms and there are different classifications of the illness. There are classical approaches for classifying DCI as we mentioned while introducing DCI. The treatment of DCI, *recompression* depends on the classification which proves the importance of classification. There are the cases where this treatment is applied without a presence of a doctor or with remote assistance. So the first attempts by Ozyigit et al. [6, 7] to classify DCI using Ward's method and two-step clustering give us an empirical grouping of signs and symptoms of DCI. We will use different clustering algorithms to classify signs and symptoms of DCI which may help as a decision support tool for the diagnosis and facilitate to improve the treatment.

4.3.1. K-Means Clustering

As our aim was to cluster the patients and compare it to the results of Ozyigit et al.[7] we have set the number of classes to 4 after trying different values: We have set the number of clusters to different values such as 3 which give Cluster 1 with numbness, paresthesia, skin sensitivity, weakness, Cluster 3 with weakness and muscular weakness, and mostly other symptoms and signs are nearly equal in at least two clusters, pain is seen in all clusters but most frequent in Cluster 2. What we observed when we have set the number of clusters to 6 as in PSI, Cluster 2 and Cluster 4 with numbness, paresthesia and skin sensitivity where pain has no occurrence in Cluster 4 but in Cluster 2. Cluster 5 holds nearly all the symptoms and signs like Cluster 1, Cluster 3 and Cluster 6 where pain is the most frequent one in Cluster 5. The data is binary and we don't have any missing data so the data we have used in our study is not noisy. As a result, we expected that k-means clustering would give reliable results.

So finally, we obtained the following clusters: Cluster 1 with 708 patients, Cluster 2 with 637 patients, Cluster 3 with 300 patients and Cluster 4 with 204 patients.

The results of clustering with k-means algorithm are given in Table 4.1. Pain is the symptom that is seen in all clusters. Numbness, paresthesia and skin sensitivity are the signs and symptoms of Cluster 1. Cluster 2 has the characteristics of the signs and symptoms such as fatigue, headache, skin, mental, nausea, pulmonary, unconsciousness, vertigo, lymphatic, abnormal sensations, hearing problems, vision problems, muscular problems and tinnitus. Cluster 3 is the pain only cluster. Weakness, muscular weakness, paralysis and bladder bowel are the characteristics of Cluster 4.

When we further analyze Cluster 1 and Cluster 4 we see that the major symptoms of spinal cord are the same as in the Type 2 decompression sickness defined by Golding et al.[3]. We can also observe that there is a hierarchical classification from mild to severe similar to Perceived Severity Index [5] and other classifications in literature.

Table 4.1 Signs and symptoms by K-Means clusters

	Cluster1	Cluster2	Cluster3	Cluster4
Consciousness	5	28	0	10
Mental	32	78	0	32
Pulmonary	27	45	0	20
CV	2	1	0	1
Pain	431	159	300	45
Skin	20	89	0	13
Lymphatic	4	17	0	1
Abnormal Sensations	12	17	0	8
Hearing	2	13	0	1
Vision	16	26	0	11
Coordination	19	24	0	17
Muscular Weakness	63	7	2	186
Muscular Problems	22	25	0	5
Skin Sensitivity	140	48	0	60
Bladder Bowel	7	2	0	27
Headache	40	96	0	21
Fatigue	59	110	0	26
Nausea	29	57	0	10

Dizziness	46	95	0	27
Vertigo	11	28	0	8
Paresthesia	615	51	0	74
Tinnitus	1	3	0	0
Numbness	753	135	0	115
Paralysis	44	8	0	71
Weakness	74	28	0	204

4.3.2. COBWEB Clustering

We obtained 3 clusters using the COBWEB Algorithm by setting the acuity (minimum value to avoid infinite values [10]) to 1 and cutoff (a parameter to suppress growth to avoid overwhelmingly large hierarchy [10]) to 0.2740947917738781. We have set the cutoff to this value after trying several values which ended with three clusters: We obtained 58 clusters, Cluster 2 with 721 instances and Cluster 3 with 944 instances when cutoff is set to 0.2420947917738781 and we observed a convergence of the remaining 56 clusters to one cluster. Cluster 1 has 264 instances while Cluster 2 and Cluster 3 have 721 and 944 instances respectively.

The results of clustering with COBWEB Algorithm are given in Table 3. Weakness, paralysis, muscular weakness and bladder bowel are the characteristic signs and symptoms of Cluster 1. We see that numbness, paresthesia and skin sensitivity are the signs and symptoms of Cluster 2. While pain is one of the most frequent sign of Cluster 3 other signs and symptoms of Cluster 3 are mental, unconsciousness, pulmonary, skin, lymphatic, abnormal sensations, hearing, vision, headache, fatigue, nausea, dizziness, vertigo.

Table 4.2 Signs and symptoms by COBWEB clusters

	Cluster1	**Cluster2**	**Cluster3**
Unconsciousness	11	8	20
Mental	35	28	77
Pulmonary	27	10	54
CV	2	1	1
Pain	96	179	652
Skin	15	30	76
Lymphatic	1	5	16
Abnormal Sensations	10	10	17
Hearing	1	4	11
Vision	15	13	25

Coordination	21	18	21
Muscular Weakness	237	10	11
Muscular Problems	7	19	26
Skin Sensitivity	73	95	76
Bladder Bowel	33	2	1
Headache	24	49	81
Fatigue	32	60	100
Nausea	12	35	46
Dizziness	31	55	78
Vertigo	9	13	25
Paresthesia	128	565	26
Tinnitus	0	1	2
Numbness	163	649	167
Paralisys	107	5	11
Weakness	255	20	31

4.3.3. EM Clustering

Using the EM Algorithm we had 4 clusters: Cluster 1 with 905 patients, Cluster 2 with 253 patients, Cluster 3 with 471 patients, Cluster4 with 300 patients after several attempts by changing the number of clusters and by setting the minimum standard deviation to 0.1 and maximum number of iterations to 100.

We have started with the default settings where standard deviation is set to 1.0E-6, maximum number of iterations is set to 100 and number of clusters is set by the algorithm and we obtained 10 clusters, cluster 0 with Weakness, paralysis, muscular weakness and bladder bowel, cluster 7 with numbness, paresthesia and skin sensitivity, cluster 9 with unconsciousness, mental, pulmonary, skin, lymphatic, abnormal sensations, headache, fatigue, nausea, dizziness and cluster 2 with vision, coordination, vertigo and tinnitus. Other clusters does not show a significative symptom or alike these clusters. We have implemented the algorithm then by changing the number of clusters and we've seen that there is a convergence to 4 clusters.

Table 4.3 Signs and symptoms by EM clusters

	cluster1	cluster2	cluster3	cluster4
Consciousness	6	10	27	0
Mental	35	36	71	0

Pulmonary	19	26	47	0
CV	1	2	1	0
Pain	370	102	163	300
Skin	27	15	80	0
Lymphatic	4	1	17	0
Abnormal Sensations	10	10	17	0
Hearing	1	2	13	0
Vision	12	16	25	0
Coordination	18	21	21	0
Muscular Weakness	9	243	4	2
Muscular Problems	23	9	20	0
Skin Sensitivity	152	72	24	0
Bladder Bowel	1	34	1	0
Headache	59	22	76	0
Fatigue	77	29	89	0
Nausea	41	11	44	0
Dizziness	58	31	79	0
Vertigo	9	10	28	0
Paresthesia	593	132	15	0
Tinnitus	1	0	3	0
Numbness	846	148	9	0
Paralysis	8	111	4	0
Weakness	43	237	26	0

The Table 4.3 shows the occurrences of symptoms and signs in clusters. We see that numbness, paresthesia and skin sensitivity are the signs and symptoms of Cluster1. Weakness, paralysis, muscular weakness and bladder bowel are the characteristic signs and symptoms of Cluster 2. Signs and symptoms of Cluster 3 are mental, unconsciousness, mental, pulmonary, skin, lymphatic, abnormal sensations, hearing, vision, headache, fatigue, nausea, dizziness, vertigo. Cluster 4 is the pain only cluster.

4.3.4. Statistical Analysis of Clusters

We have made an association and correlation analysis of the clusters and the classifications we obtained using PSI diagnosis, final classical diagnosis and outcome of the instances that we used for clustering. Firstly we have calculated the Pearson product-moment correlations among the clusters of EM, k-means, COBWEB, two-steps, and PSI classifications. The results are given in table 4.4. We have chosen to keep the correlations which is bigger than 0.5 [25].

Table 4.4 Correlations of clusters

	Cluster1	Cluster2	Cluster3	Cluster4
Cluster1	E1:P1=0,6991 C1:P1=0,9235 K1:P1=0,7387 E1 :K1=0.9910	E1:C2=0,9817 E1:K2=0,5832 E1:T2=0,9957 E1:P2= 0,5031 K1:T2=0,9937	E1:P3=0,9951 C1:T3=0,9954 K1 :P3=0,9933	T1:P4=0,9913 C1:K4=0,9881
Cluster2	E2:C1=0,9975 E2:P1=0,9151 C2:K1=0,9687 T2:P1=0,6753	C2:T2=0,9815 K2:P2=0,6230	E2:T3=0,9916 C2:P3=0,9730 K2:P3=0,6202 T2:P3=0,9950	E2:K4=0,9814 K2:T4=0,9765 K2:P4=0,6042
Cluster3	T3:P1=0,9455 C3:T1=0,9518	C3:P2=0,5802 E3:K2=0,8120 C3:K2=0,7489	E3:C3=0,7864 K3:T1=1 C3:K3=0,9518	E3:T4=0.8693 E3:P4=0,7458 C3:T4=0,7441 C3:P4=0,9704 K3:P4=0,9913
Cluster4	K4:P1=0,8808 E4:T1=1	T4:P2=0,6065	K4:P3=0,5521 K4:T3=0.9738 E4:C3=0,9518 E4:K3=1	T4:P4=0,6137 E4:P4=0,9913

The results in red indicates the highest correlation, the results in are those ranked second between the correlations of the clusters. The letters E, C, K, T and P are the abbreviations for the EM, COBWEB, K-means, two-steps and PSI classifications respectively.

We see that there is a strong correlation among the clusters we have found as the correlations are near to 1 or even 1 for the pain only classes.

After finding out that there is a large correlation among the clusters we have studied associations among the classifications of different algorithms used for clustering the DCI and other classifications.

The results are given in Table 4.5. When we further analyze the table we see that the association between different classification methods is strong. The highest association is between the classes of EM and two-steps classification when it is two steps dependent. If we have a look at the symmetric results we see that it is between EM and two steps classifications with a lambda of 0.829. The lowest lambda is 0.292 when PSI is k-means dependent and the symmetric values is 0.343. The correlations among the class 5 and class 6 of PSI and the classes of k-means are also lower than 0.5 but this is not the case for other classes so one of the reasons of this lowliness might be this difference of detail in classes of two methods. A part from this association the symmetric lambdas are high: EM_COBWEB (0.621), EM_K-Means (0.759), EM_Two-Steps (0.827), Two-Steps_K-means (0.802), COBWEB_K-Means (0.449) and COBWEB_Two-Steps (0.549). If we also take into account the high correlations approximately near to 1 among the clusters we may say that different clustering algorithms and other classifications are in accordance in classifying DCI.

Table 4.5 Associations of classifications

		EM	COBWEB	K-Means	Two-Steps	PSI
EM	Symmetric		0.621	0.759	0.827	0.405
	EM dependent		0.481	0.745	0.813	0.348
	Classification dependent		0.765	0.771	0.839	0.459
COBWEB	Symmetric	0.778		0.449	0.549	0.350
	COBWEB dependent	0.775		0.557	0.655	0.363
	Classification dependent	0.781		0.355	0.461	0.339
K-Means	Symmetric	0.759	0.449		0.802	0.343
	K-Means dependent	0.771	0.355		0.798	0.292
	Classification	0.745	0.557		0.806	0.397

	dependent					
Two-steps	Symmetric	0.827	0.549			0.444
	Two-steps dependent	0.839	0.461			0.426
	Classification dependent	0.813	0.655			0.464

Consequently, we have compared different clusters obtained from different classification methods and also different classifications by using the Goodman-Kruskal lambda and correlation and these statistical analysis yields us to the result that there is a strong relationship among the different classifications and also among the clusters of different classifications.

4.3.5. Discussion

When we compare clusters constructed using the EM Algorithm with the two other algorithms explained above we observe that Cluster 1 have the same signs and symptoms (numbness, paresthesia and skin sensitivity) as the Cluster 2 of COBWEB and Cluster 1 of k-means. Cluster 2 is in accordance with Cluster 4 of k-means and Cluster 1 of COBWEB having weakness, paralysis, muscular weakness and bladder bowel as signs and symptoms. Finally, Cluster3 is in consistency with Cluster 3 of COBWEB and Cluster 2 of k-means. Cluster 4 is the pain only cluster as in cluster 3 of k-means.

The clusters are hierarchical from mild to severe like the other algorithms cited above and pain is seen all clusters while there are pain only clusters.

In two-step cluster analysis of DCI [7] Ozyigit et al. has found pain which is not associated with another sign or symptom, is the characteristic symptom of cluster 1. Numbness, paresthesia and skin sensitivity are the characteristic signs and symptoms with a presence of more than 50% of their total present counts are in Cluster 2. Weakness, paralysis, muscular weakness and bladder bowel problems are the characteristic signs and symptoms for the Cluster 3 and unconsciousness, mental, pulmonary, skin, lymphatic, muscular problems, abnormal sensations, hearing, vision troubles, headache, fatigue, dizziness, nausea, vertigo and tinnitus, for the Cluster 4.

Comparing the signs and symptoms we also see that Clusters 2 to Cluster 4 are alike wherein our study there is no pain only group in COBWEB cluster.

Table 4.6 Classification of DCI

Classification/Clusters				
Ozyigit	Pain	Numbness Paresthesia Skin Sensivity	Weakness Paralysis Muscular Weakness Bladder Bowel	Unconsciousness Mental Pulmonary Skin Lymphatic Hearing Vision Other signs and symptoms
k-means	Pain	Numbness Paresthesia Skin Sensivity	Weakness Paralysis Muscular Weakness Bladder Bowel	Unconsciousness Mental Pulmonary Skin Lymphatic Hearing Vision Other signs and symptoms
Benton&Glover	Limb pain	Neurological	Neurological	Cardiopulmonary Lymphatic Constitutional Cutaneous Vestibular
Buch et al	Mild	Moderate	Moderate/Severe	Severe
COBWEB		Numbness Paresthesia Skin Sensivity	Weakness Paralysis Muscular Weakness Bladder Bowel	Unconsciousness Mental Pulmonary Skin Lymphatic Hearing Vision Other signs and

				symptoms
EM	Pain	Numbness Paresthesia Skin Sensivity	Weakness Paralysis Muscular Weakness Bladder Bowel	Unconsciousness Mental Pulmonary Skin Lymphatic Hearing Vision Other signs and symptoms

When we compare our results with other classifications in table 2, we see that all classifications have a hierarchy from mild to severe and see that there is a pain only group which is followed by mild neurological signs and symptoms and severe neurological signs and symptoms. In Cluster 4 we find out many symptoms and signs such as hearing problems, vision problems, and muscular problems, pulmonary, skin which are the case in Cluster 4 of Ozyigit and in classical classification of those cited in the table. Clusters we have obtained using k-means are hierarchical as the clusters of Ozyigit et al. and also as in classical classifications previously cited. We have observed that the two different clustering algorithms (k-means algorithm and two-step cluster analysis) have resulted in clusters corresponding to nearly the same symptoms and signs of DCI. Moreover, these statistically formed clusters are coherent with the other expert-based classifications of DCI.

We may summarize the results of clustering of DCI with different algorithms in Table 4.7. Different algorithms have given similar results with the other expert-based classification and of Ozyigit et al. But we have not found a “pain only” class unlike Ozyigit et al.[7] and PSI[5]. Pain is one of the significant symptoms and signs in one of the clusters as it is also the case in two-steps clustering but we have not found a pain only cluster by COBWEB algorithm which might be due to incremental characteristic of the algorithm. Numbness, paresthesia and skin sensitivity are signs and symptoms in one of the classes. Another cluster has weakness, paralysis,

muscular weakness and bladder bowel as symptoms and signs. Unconsciousness, mental, pulmonary, skin, lymphatic, muscular problems, abnormal sensations, hearing, vision troubles, headache, fatigue, dizziness, nausea, vertigo and tinnitus are the signs and symptoms of Cluster 4 (Two-steps), of Cluster 2 (k-means) and of Cluster 3 (COBWEB, EM). There is a hierarchy found out by all clustering methods from mild to severe [26].

We have also studied the correlations among the clusters and associations among different classification methods as we stated in the previous section which have signaled a strong relationship among the clusters and the classifications. We think that these statistical results can be taken into account as a proof that clustering algorithms can successfully classify DCI.

Table 4.7 Clusters of DCI with different algorithms

Classification/Clusters				
Two-steps	Cluster 1	Cluster 2	Cluster 3	Cluster 4
k-means	Cluster 3	Cluster 1	Cluster 4	Cluster 2
COBWEB		Cluster 2	Cluster 1	Cluster 3
EM	Cluster 4	Cluster 1	Cluster 2	Cluster 3

The results presented in this work can be seen as confirmatory relatively to the work performed by Ozyigit *et al*. [7]: clustering algorithms can be efficiently used to classify DCI symptoms. This automatic approach may ease the diagnosis of patients and may result with better recompression and medical treatment. The clusters may be useful in the research of outcome of the treatment which is linked with the age, sex, time passed till treatment. The associations may be useful in large databases and the decision tables may be used as a decision support tool.

The limitation of the clustering methods used in this work and in Ozyigit's is the obtained clusters are not easily interpretable. Clustering with k-means, COBWEB and EM is a first step for classification which results with classes of symptoms and signs but it is a process which ends only with results. As a post-processing phase, we decided to construct a decision tree that might help to use the classification easily as a

decision support tool. The benefit of a decision tree is that it is easy to interpret and understand the classes as it can be visualized easily. And association rules might help finding out the relations among the signs and symptoms within the cluster. The next step of this study will focus on these relations, trying to find out the links between the outcome of the treatment and these variables. A first step was to find out the correlations among the clusters and the outcome of therapy that we have found strong correlations but we will not interpret the results due to lack of metadata for outcome.

4.4. ASSOCIATON RULES

4.4.1. A priori algorithm

We used the *A priori* algorithm to find out the association rules that may help us to find the relations among the signs and symptoms. We have set the minimum support to 0.35 and minimum confidence to 0.5 after implementing the algorithm with different minimum support and minimum confidence to avoid numerous relationships with many variables which are hard to interpret. Best rules found with these parameters for k-means clustering are as follows:

1. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE Numbness=TRUE Weakness=FALSE 809 ==> Cluster=cluster0 679 conf:(0.84)

2. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE Numbness=TRUE Weakness=FALSE 808 ==> Cluster=cluster0 678 conf:(0.84)

3. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE BladderBowel=FALSE Numbness=TRUE Weakness=FALSE 808 ==> Cluster=cluster0 678 conf:(0.84)

4. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 808 ==> Cluster=cluster0 678 conf:(0.84)

5. Consciousness=FALSE Hearing=FALSE Numbness=TRUE Weakness=FALSE 814 ==> Cluster=cluster0 683 conf:(0.84)

6. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE BladderBowel=FALSE Numbness=TRUE Weakness=FALSE 807 ==> Cluster=cluster0 677 conf:(0.84)

7. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 807 ==> Cluster=cluster0 677 conf:(0.84)

8. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE BladderBowel=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 807 ==> Cluster=cluster0 677 conf:(0.84)

9. Consciousness=FALSE CV=FALSE Hearing=FALSE Numbness=TRUE Weakness=FALSE 813 ==> Cluster=cluster0 682 conf:(0.84)

10. Consciousness=FALSE Hearing=FALSE BladderBowel=FALSE Numbness=TRUE Weakness=FALSE 813 ==> Cluster=cluster0 682 conf:(0.84)

11. Consciousness=FALSE Hearing=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 813 ==> Cluster=cluster0 682 conf:(0.84)

12. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE BladderBowel=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 806 ==> Cluster=cluster0 676 conf:(0.84)

13. Consciousness=FALSE CV=FALSE Hearing=FALSE BladderBowel=FALSE Numbness=TRUE Weakness=FALSE 812 ==> Cluster=cluster0 681 conf:(0.84)

14. Consciousness=FALSE CV=FALSE Hearing=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 812 ==> Cluster=cluster0 681 conf:(0.84)

15. Consciousness=FALSE Hearing=FALSE BladderBowel=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 812 ==> Cluster=cluster0 681 conf:(0.84)

16. Consciousness=FALSE Lymphatic=FALSE Numbness=TRUE Weakness=FALSE 811 ==> Cluster=cluster0 680 conf:(0.84)

17. Consciousness=FALSE CV=FALSE Hearing=FALSE BladderBowel=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 811 ==> Cluster=cluster0 680 conf:(0.84)

18. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Numbness=TRUE Weakness=FALSE 810 ==> Cluster=cluster0 679 conf:(0.84)

19. Consciousness=FALSE Lymphatic=FALSE BladderBowel=FALSE Numbness=TRUE Weakness=FALSE 810 ==> Cluster=cluster0 679 conf:(0.84)

20. Consciousness=FALSE Lymphatic=FALSE Tinnitus=FALSE Numbness=TRUE Weakness=FALSE 810 ==> Cluster=cluster0 679 conf:(0.84)

We see from this association rules that absence of muscular weakness, paresthesia, bladder bowel, tinnitus and CV yields to cluster 4 of k-means which is confirmatory to

the classification as muscular weakness, paresthesia, bladder bowel are the characteristics of cluster 2 and cluster 3. CV and tinnitus are very rare in the dataset both with 4 occurrences so another analysis may be done by excluding these 2 attributes.

Association rules found with minimum support 0.45 and minimum confidence 0.5 for COBWEB clustering are:

1. Muscular Weakness=FALSE Paresthesia=FALSE Weakness=FALSE 1012 ==> Cluster=cluster3 877 confidence: (0.87)
2. CV=FALSE Muscular Weakness=FALSE Paresthesia=FALSE Weakness=FALSE 1011 ==> Cluster=cluster3 876 confidence: (0.87)
3. Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Weakness=FALSE 1011 ==> Cluster=cluster3 876 confidence: (0.87)
4. Muscular Weakness=FALSE Paresthesia=FALSE Tinnitus=FALSE Weakness=FALSE 1010 ==> Cluster=cluster3 875 confidence: (0.87)
5. CV=FALSE Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Weakness=FALSE 1010 ==> Cluster=cluster3 875 confidence: (0.87)
6. CV=FALSE Muscular Weakness=FALSE Paresthesia=FALSE Tinnitus=FALSE Weakness=FALSE 1009 ==> Cluster=cluster3 874 confidence: (0.87)
7. Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Tinnitus=FALSE Weakness=FALSE 1009 ==> Cluster=cluster3 874 confidence: (0.87)
8. CV=FALSE Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Tinnitus=FALSE Weakness=FALSE 1008 ==> Cluster=cluster3 873 confidence: (0.87)
9. Muscular Weakness=FALSE Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1006 ==> Cluster=cluster3 871 confidence: (0.87)
10. Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1006==> Cluster=cluster3 871 confidence: (0.87)
11. CV=FALSE Muscular Weakness=FALSE Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1005 ==> Cluster=cluster3 870 confidence: (0.87)

12. CV=FALSE Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1005 ==> Cluster=cluster3 870 confidence: (0.87)
13. Muscular Weakness=FALSE Paresthesia=FALSE Tinnitus=FALSE Paralysis=FALSE Weakness=FALSE 1004 ==> Cluster=cluster3 869 confidence: (0.87)
14. Muscular Weakness=FALSE Bladder Bowel=FALSE Paresthesia=FALSE Tinnitus=FALSE Paralysis=FALSE Weakness=FALSE 1004 ==> Cluster=cluster3 869 confidence: (0.87)
15. Paresthesia=FALSE Weakness=FALSE 1025 ==> Cluster=cluster3 887 confidence: (0.87)
16. CV=FALSE Paresthesia=FALSE Weakness=FALSE 1024 ==> Cluster=cluster3 886 confidence: (0.87)
17. Bladder Bowel=FALSE Paresthesia=FALSE Weakness=FALSE 1024 ==> Cluster=cluster3 886 confidence: (0.87)
18. Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1016 ==> Cluster=cluster3 879 confidence: (0.87)
19. Bladder Bowel=FALSE Paresthesia=FALSE Paralysis=FALSE Weakness=FALSE 1016 ==> Cluster=cluster3 879 confidence: (0.87)
20. Paresthesia=FALSE Tinnitus=FALSE Weakness=FALSE 1023 ==> Cluster=cluster3 885 confidence: (0.87)

Muscular weakness, weakness, paresthesia, bladder bowel and paralysis are the symptoms and signs of cluster 1 and cluster 2 of in classification of DCI with COBWEB algorithm, so the absence of these symptoms and signs yields us to cluster 3 which is in accordance with the clusters.

And lastly, the association rules we have found by setting minimum support to 0.45 and minimum confidences to 0.5 for EM clustering are as follows:
1. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE Paralysis=FALSE 1550 ==> Cluster=cluster0 869 conf:(0.56)

2. Skin=FALSE Hearing=FALSE MuscularWeakness=FALSE 1553 ==> Cluster=cluster0 869 conf:(0.56)

3. Consciousness=FALSE Lymphatic=FALSE Vision=FALSE MuscularWeakness=FALSE Vertigo=FALSE 1554 ==> Cluster=cluster0 869 conf:(0.56)

4. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1565 ==> Cluster=cluster0 875 conf:(0.56)

5. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1567 ==> Cluster=cluster0 876 conf:(0.56)

6. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE Tinnitus=FALSE 1564 ==> Cluster=cluster0 874 conf:(0.56)

7. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE Tinnitus=FALSE 1566 ==> Cluster=cluster0 875 conf:(0.56)

8. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE 1568 ==> Cluster=cluster0 876 conf:(0.56)

9. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE 1570 ==> Cluster=cluster0 877 conf:(0.56)

10. Consciousness=FALSE CV=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE Tinnitus=FALSE 1567 ==> Cluster=cluster0 875 conf:(0.56)

11. Consciousness=FALSE Lymphatic=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE Tinnitus=FALSE 1569 ==> Cluster=cluster0 876 conf:(0.56)

12. Consciousness=FALSE CV=FALSE Abnormal Sensations=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1558 ==> Cluster=cluster0 869 conf:(0.56)

13. Consciousness=FALSE Abnormal Sensations=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1560 ==> Cluster=cluster0 870 conf:(0.56)
14. Consciousness=FALSE CV=FALSE Hearing=FALSE Vision=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1560 ==> Cluster=cluster0 870 conf:(0.56)
15. Consciousness=FALSE Hearing=FALSE Vision=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE 1562 ==> Cluster=cluster0 871 conf:(0.56)
16. Consciousness=FALSE Abnormal Sensations=FALSE Hearing=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE Tinnitus=FALSE 1559 ==> Cluster=cluster0 869 conf:(0.56)
17. Consciousness=FALSE CV=FALSE Hearing=FALSE Vision=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE Tinnitus=FALSE 1559 ==> Cluster=cluster0 869 conf:(0.56)
18. Consciousness=FALSE CV=FALSE Abnormal Sensations=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE 1561 ==> Cluster=cluster0 870 conf:(0.56)
19. Consciousness=FALSE Hearing=FALSE Vision=FALSE MuscularWeakness=FALSE BladderBowel=FALSE Vertigo=FALSE Tinnitus=FALSE 1561 ==> Cluster=cluster0 870 conf:(0.56)
20. Consciousness=FALSE Abnormal Sensations=FALSE Hearing=FALSE MuscularWeakness=FALSE Vertigo=FALSE 1563 ==> Cluster=cluster0 871 conf:(0.56)

The symptoms and signs on the left side of the association rules are the symptoms and signs of cluster 2 and cluster 3 of the classification with EM algorithm, and their absence yields us to the cluster 1 which is concordant with the signs and symptoms of the clusters.

4.4.2. Discussion

The rules we have found may be trivial for experts, or they may be useful in the presence or absence of some of the symptoms or signs. They are in accordance with

the clusters we have found. The minimum support is 0.45 for all of the 3 classifications. And the lowest confidence is 0.77, 0.87 and 0.56 for k-means, COBWEB and EM respectively. But there may be further analysis by using other algorithms and with different parameters and only within the clusters which may help to find multi-level association rules which might be more interesting than the ones that we found using Apriori algorithm.

5. CONCLUSION

In this work we have used three different clustering algorithms which have different approaches to the problem of clustering: incremental, categorical and probabilistic. These three approaches found out similar classes for the DCI with the other expert-based classification and of Ozyigit et al. but no "pain only" class unlike Ozyigit et al.[7] and PSI [5]. Pain is one of the significant symptoms and signs in all clusters as it is also the case in two-steps clustering. Numbness, paresthesia and skin sensitivity are signs and symptoms in one of the classes. Another cluster has weakness, paralysis, muscular weakness and bladder bowel as symptoms and signs. Unconsciousness, mental, pulmonary, skin, lymphatic, muscular problems, abnormal sensations, hearing, vision troubles, headache, fatigue, dizziness, nausea, vertigo and tinnitus are the signs and symptoms of Cluster 4 (Two-steps, k-means) and Cluster 3 (COBWEB, EM). There is a hierarchy found out by all clustering methods from mild to severe.

We have also applied a statistical analysis on the classifications we have found and compared them with themselves and with two-steps algorithm classification and with PSI. We found that there is a strong correlation among the clusters and the associations among the classifications are high which is confirmatory to say that different algorithms can successfully classify DCI. The association of outcome and classifications are nearly zero which yields us to add more variables to this relationship such as age, sex, treatment variables which may help us to find out association rules between these elements of DCI.

Association rules that we have found might help finding out the relations among the signs and symptoms within the cluster or they may be confirmatory for different classifications.

Further studies might focus on to find out the relations among different classifications, treatment and outcome and these variables, or a supervised classification.

Consequently, cluster analysis is one of the suitable techniques that can be used to classify DCI according to their signs and symptoms which groups empirically the DCI and highlights the difference of classes and the important signs and symptoms of each class that may help the diagnosis of DCI and it can be used for further studies to find out the relationship of diagnosis, treatment and outcome of treatment.

REFERENCES

[1] Decompression Illness: What Is It and What Is The Treatment? Available from:http://www.diversalertnetwork.org/medical/articles/article.asp?articleid=65 (2009).

[2] Benton, M. and M. Glover, Dive Medicine. Travel Med Infect Dis, (4): 238-254, (2006).

[3] Golding, F., et al. Decompression sickness during construction of the Dartford Tunnel, Brit. J. Indus. Med. (17), 167-180, (1960).

[4] Buch, A. et al. Cigarette Smoking and Decompression Illness Severity: A Retrospective Study in Recreational Divers, Aviat Space EnvironMed, 74: 1271-1278, (2003).

[5] Vann, R., et al., DAN's Annual Review on Decompression Illness, Diving Fatalities, and Project Dive Exploration, (2002).

[6] Özyigit, T. et al. Empirical Classification of DCS patients using Cluster Analysis in 33rd International Meeting of European Underwater and Baromedical Society: Sharm el-Sheikh, Egypt. (2007)

[7] Ozyigit, T. et al. Empirical Classification of DCI using Cluster Analysis, (2009).

[8] Ward, J.H. Hierarchical Grouping to Optimize an Objective Function. J Am Stat Assoc. (63): 236-244,(1963).

[9] Chiu, T. et al. A Robust and Scalable Clustering Algorithm for Mixed Type of Attributes in Large Database Environment, in Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (2001).

[10] Witten, I.H. and E. Frank, Data Mining: Practical machine learning tools and techniques with Java implementations, San Francisco, CA, USA, Morgan Kaufman, (2000).

[11] Fayyad, U., G. Piatetsky-shapiro, and P. Smyth, From Data Mining to Knowledge Discovery In Databases AI Magazine. (17), 37-54,(1996).

[12] Aggarwal, C.C. and P.S. Yu, Data Mining Techniques for Associations, Clustering and Classification, in Proceedings of the Third Pacific-Asia Conference on Methodologies for Knowledge Discovery and Data Mining, Springer-Verlag, (1999).

[13] Graepel, T., Statistical physics of clustering algorithms. Technical Report 171822, FB Physik, Institut fur Theoretische Physic (1998).

[14] Kohavi, R. , A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection in Proceedings of the International Joint Conference on Artificial Intelligence (1995).

[15] Xiong, H., J. Wu, and J. Chen., K-means Clustering versus Validation Measures, A Data Distribution Perspective,in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Philadelphia, PA, USA (2006).

[16] Lloyd., S.P., Least Squares Quantization in PCM, IEEE Transactions on Information Theory, 28(2), 129-137, (1982).

[17] Matteucci, M., K-Means Clustering, Available from: http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html, (2009).

[18] Douglas, H.F., Knowledge acquisition via incremental conceptual clustering, Machine Learning Volume 2. (2), 139-172, (1987).

[19] Kotsiantis, S. and D. Kanellopoulos, Gamma Ray Burst Search,(2009).

[20] Agrawal, R. and R. Srikant, Fast Algorithms for Mining Association Rules, in Int'l Conf. Very Large Data Bases, Santiago, Chile,487-499, (1994).

[21] Kotsiantis, S. and D. Kanellopoulos, Association Rules Mining: A Recent Overview, GESTS International Transactions on Computer Science and Engineering, 32(1), 71-82, (2006).

[22] Milton, J.S. and J.C. Arnold, Introduction to probability and statistics, McGraw Hill, (2003).

[23] Garson,G.D. Measures of Association, Available from http://faculty.chass.ncsu.edu/garson/PA765/association.htm. (2009).

[24] Witten,I.H. and E. Frank, Available from: http://www.cs.waikato.ac.nz/~ml/weka/index.html, (2009).

[25] Cohen.J, Statistical Power Analysis for the Behavioral-Science, (1990).

[26] Catanzaro, M., M. Boguna, and R. Pastor-Satorras, Generation of uncorrelated random scale-free networks, Physical Review E, 71(2), (2005).

BIOGRAPHICAL SKETCH

Barış Aksoy was born in Erzincan in 13 February 1979. He is graduated from Vefa High School. Between 1998 and 2004 he was a student at the Computer Engineering Department of Galatasaray University where he had his B.Sc. He is a student of Master of Computer Enginering at the University of Galatasaray. In 2004 he joined the Galatasaray University as research assistant at the Faculty of Engineering and Technology, where he worked till 2006. After the military service, in 2007 he started to work at the IT Department of Anadolu Hayat Emeklilik.