Data Mining derives from searching valuable information in a large database, which requires either sifting through a huge amount of data or intelligently investigate it to find exactly where the value is. Data Mining is concerned with finding new patterns in large amounts of data and it can be applied to several areas, like Business, Medicine or Education (EDM), being the latter the object of this study.

## 1.1. The definition of data mining

Data mining and knowledge discovery in databases are two terms, but these two terms are often used interchangeably. Basically, data mining is the process of the extraction of patterns or models from observed data. The simple definition of knowledge discovery in databases is that knowledge discovery in databases is the process of identifying valid, potential, useful and ultimately understandable patterns in data [1]. Originally, data mining was just one step in the overall knowledge discovery in database process. A common acceptable process of knowledge discovery in databases consists of the following steps:

- understanding of the application domain and requirements
- selecting a target data set
- integrating and checking the data set
- data cleaning and preprocessing
- model development and hypothesis building
- choosing suitable data mining algorithms
- interpreting results
- verifying and testing results
- using the discovered knowledge

In the remainder of the thesis we will use the term "data mining" to refer both narrowly to the actual discoveries of patterns in the data and broadly to include the entire above process.

## 1.2 Data Mining Tasks

Goebel and Gruenwald (1999) survey data mining goals in the following categories. We should note that several methods may be applied together to achieve a desired result in real applications.

- Prediction: Given a data item and a predictive model, predict the value for a specific attribute of the data set.

- Regression: analysis of the dependency of some attribute values over other attribute values and automatic production of a model that can predict these attribute values for new records.

- Classification: Given a set of predefined categorical classes, determine to which of these classes a specific data item belongs.

- Clustering: Given a set of data items, partition or divide this set into a set of classes such that items with similar characteristics are grouped together.

- Association: Identify relationships between attributes and items such as the presence of one pattern implies the presence of another pattern.

- Exploratory data analysis: Exploratory data analysis is the interactive exploration of a data set without dependense on predifined assumltion and models, thus attempting to indentify intersting patterns.

## 1.3 Categories of Data Mining Algorithms

The categories of data mining algorithms mainly include classification and prediction, clustering, and mining association rules. Here, we briefly list primary data mining algorithms, and outline details and descriptions combined with educational data mining applications in the following sections.

- Classification and Prediction
  - ✓ Classifiy nominal data：Decision Tree, Neural Networks, Bayes Classifier, Instance –based Reasoning, Support Vector Machines (Kernel Machines)
  - ✓ Predict continuous data：Linear Regression, Non-linear Regression, Neural Networks, Kernel Models
  - ✓ Probability Bayesian Network, Hidden Markov Model (Dynamic Bayesian Net), Density Estimator, Fuzzy Methods
- Clustering
  - ✓ Partitional Methods: K-means and K-medians square-error methods, Mode-seeking methods, Graph based method, Mixture Distribution Models, Fuzzy c-means Methods
  - ✓ Hierarchical Methods: Bottom-up and Top-down methods
- Database mining
  - ✓ Relational data mining
  - ✓ Document warehouse
  - ✓ Data warehouse
- Graph mining
- Mining Association Rules
- Concept Mining
- Sequence mining
- Tree mining

- Web mining
- Software mining
- Text mining

| | Data Type | | | | Data Mining Problem | |
|---|---|---|---|---|---|---|
| Data Mining Methodology | Labeled Data | Unlabeled Data | Separate Data Records | Time Series Data | Predication and Classification | Discovery of Data Patterns,Associations, and Structure |
| Decision trees | X | | X | | X | X |
| Association rules | | X | X | | | X |
| Artificial neural networks | X | X | X | X | X | |
| Bayesian network | X | X | X | X | X | X |
| Hidden Markov Model | X | X | | X | X | |
| Clustering | | X | X | | | X |
| Support vector machines | X | X | X | | X | |

**Figure 1.1:** Typical Use of Data Mining Methodologies for VariousData Types and Problems

## 1.4. Educational data mining

In the past few years, lots of educational systems have been deployed to provide more flexible courses. These systems are not only based on static learning materials, but also some of them take into account the diversity of students using adaptive and intelligent techniques. To offer personalized learning environments, these systems build up learner models based on learners' goals, preferences and knowledge. Further, data mining and knowledge discovery techniques can play an important role in extracting the interesting and useful patterns about learners from logs of their behaviours.

Educational data mining (EDM) integrates data mining and knowledge discovery methods into educational environments. EDM is "concerned with developing methods for exploring the unique types of data that come from educational settings, and using those methods to better understand students, and the settings which they learn in". [2] Educational data mining is a process of converting raw data from educational systems to useful information that can be used to inform design decisions and answer research questions.

### 1.4.1. Data Mining Techniques in Educational Systems

In the educational field, data mining techniques can find useful patterns that can be used both by educators and learners. Not only may EDM assist educators to improve the instructional materials and to establish a decision process that will modify the learning environment or teaching approach, but it may also provide recommendations to learners to improve their learning and to create individual learning environments.

Romero and Ventura (2007) introduced an educational data mining cycle model showing that the application of educational data mining is an iterative cycle of hypothesis formation, testing and refinement. The knowledge mined from educational data mining should be used to facilitate and enhance the whole learning process. From this cycle model, we can see that the application of educational data mining can be oriented to different actors each with their own views [3]:

- Oriented toward learners: Purposes for EDM are to recommend to learners good learning experiences, effective learning sequences, useful resources, successful tasks carried out by other similar learners, and activities that would favour and improve their learning based on the tasks already done by other learners.
- Oriented toward educators: Purposes for EDM are to get more feedback for instructors, classify learners into groups based on their behaviours and needs, find effective learning patterns, find more effective activities, discover the most frequently made mistakes, organize the contents efficiently for instructors to adopt instructional plans, evaluate the learning process, and evaluate the structure of course contents.

Many data mining algorithms can be applied in educational systems with different data sources and purposes. The majority of applications use classifcation, clustering, association rules and text mining algorithms. Here we briefly describe these techniques with an emphasis on the applications of these techniques in various educational systems.

## 1.4.2. Classification

**Decision Tree**

A decision tree is a special type of calssifier, where each internal node denotes a test on a splitting attribute, each branch represents an outcome of the test, and leaf nodes represent classes or class distributions. [4] Let $X_1$,...., $X_m$ ,C be random variables where $X_i$ has domain dom($X_i$); a classifier is a function

$$d : dom(X_1) \times ... \times dom(X_m) \rightarrow dom(C)$$

There are two phases in constructing a classfication tree from nominal attributes. In the growth phase, an overly large decision tree is constructed from the training data. To minimize the misclassification rate, impurity-based split selection methods will find the splitting criterion by minimizing an impurity function such as the information gain (entropy), the gini-index or the index of correlation $\chi^2$. For scalable data access, in which the training dataset is large, the main algorithms are: Sprint, which removes all relationships between main memory and size of the training dataset; Rainforest, a generic tree induction schema; and Boat, the only tree construction algorithm that constructs several levels of the tree in a single scan over the dataset.

In the pruning phase, pruning algorithms are used to prevent ovefitting and to construct the final tree with minimized misclassification rate. The minimum description length (MDL) principle states that the best classification tree can be encoded with the least number of bits. The PUBLIC pruning algorithm combines the growth and pruning phase by computing a lower bound based on MDL principles.

For numerical attributes, we can construct regression trees by applying decision tree algorithms. The decision tree applications include SPSS, SAS, and C4.5.

Adopting decision tree and data cube technology to web log portfolios in assessing performances of students, Chen et al. (2000) discovered potential student groups based on similar characteristics to develop more effective pedagogical strategies. Murray and Vanlehn (2000) used dynamic decision theory to select rational and interesting actions within satisfied response time. Utilizing a machine-learned detector, Baker (2009) predicts students' off-task behaviours within an intelligent tutor, and finds best predicting models. Using simple and intuitive classifiers based on decision tree, Dekker et al.(2009) describes an educational data mining case study aimed at predicting students drop out cases.

### 1.4.3. Association Rules

Association rules discovery focuses on detecting and characterizing unexpected interrelationships between data elements [5]. It typically returns all rules that satisfy user specified constraints with user defined good measures. An association rule is composed of two datasets, antecedent (A) and consequent (C). Two statistical terms, support and confidence, are used to describe these relationships. For an association a → C,

$$\text{Support } (A \rightarrow C) = \text{support } (A \cup C)$$

$$\text{Confidence } (A \rightarrow C) = \text{support } (A \cup C)/\text{support } (A)$$

If support is high enough, the confidence is a reasonable estimate of the association rule.

The problem with finding association rules in this naive way is the number of possible combinations of antecedents and consequents are very large; it is impossible to check all combinations for very large datasets. The Apriori algorithm uses user defined min-support such that support $(A \rightarrow C) \leq \min - \text{support}$, to reduce the number of datasets that are considered. The frequent item set strategy further uses both min-support and min-confidence to reduce considered datasets.

There are two objective measures of "interestingness" used to identify the most interesting rules from thousands of association rules that satisfy user specified constraints on support and confidence. The most popular measure is lift that is the ratio of the frequency of the consequent.

$$\text{Lift } (A \rightarrow C) = \text{confidence } (A \rightarrow C)/\text{support}(C)$$

The other measure is leverage that captures both the volume and the strength of the effect.

$$\text{Leverage } (A \rightarrow C) = \text{support } (A \rightarrow C) - \text{support } (A) \times \text{support}(C)$$

Association rules can be used in numerical datasets by discretizing a numeric field into subranges.

Markellou et al. (2005) proposed an ontology-based framework to use the priori algorithm to discover association rules whose purpose was to provide personalized experiences to users. Their work has distinguished two stages in the whole process: one, offline, that includes data preparation, ontology creation and usage mining, and one, online, that concerns the production of recommendations. Lu (2004) used association fuzzy rules to discover associations between requirements of learners and a list of learning materials, which would then help learners find learning materials they would need to read.

Association rules mining was used by (Monk, 2005) to reveal patterns from large volume datasets. Mostow et al.(2005) built a tool to find association rules from tutor-student interactions. By processing the vast quantity of data generated by students, Agapito et al. (2009) detects potential symptoms of low performance in e-Learning courses in two steps: generating the production rules of the C4.5 algorithm and filtering the most representative rules, which could indicate low performance of students. By using a pairwise test to search for the relationships between learning curves, Pavlik et al.(2009) show that test results can be expressed in a Q-matric domain model. Prata et al.(2009) present a model which can detect various students' speech acts within a computer supported collaborative learning environment. They found interpersonal conflict is associated with positive learning. Mercer and Yacef (2008) provided a case study to show how teachers can easily interpret association rules mined from educational data.

### 1.4.4. Bayesian Network

Bayesian networks are a well developed technique in the machine learning area. Bayesian learning calculates the probabilities of hypotheses using training data, and makes predictions based on probabilities of hypotheses. Let d be the observed data; the probability of each hypothesis h is obtained by Bayes' rule: [6]

$$P(h_i \mid d) = \alpha P(d \mid h_i) P(h_i)$$

We can then make a prediction about unknown X:

$$P(X \mid d) = \sum_i P(X \mid h_i) P(h_i \mid d)$$

Here, Bayesian learning is an optimal result. The main issue is that the hypothesis space is usually very large or infinite in real learning problems. An approximating hypothesis called maximum a posteriori (MAP) is adopted such that $P(X \mid d) \approx P(X \mid h_{MAP})$, and predictions are made based on this single probable hypothesis. The maximum-likelihood (ML) hypothesis, $h_{ML}$ , is a simplification of MAP that chooses an $h_i$ to maximize $P(d \mid H_i)$.

### Naive Bayes

In the Naive Bayes model, the output variable, which is to be predicted, is the root node, and attribute variables are leaf nodes, with the assumption that attributes are conditionally independent of each other. Given complete data, which is a training dataset of attributes with respect to outputs, the entire joint probabilities distribution of a Naive Bayes net can be easily learned by the maximum-likelihood parameter learning method, which maximizes the log likelihood. The task of prediction simply becomes choosing the most likely class by inference through the Bayes net. In this way, Naive Bayes net is the most common Bayesian network model and has a wide range of applications.

### Expectation Maximization

The expectation maximization (EM) algorithm can learn hidden variables using observed variables. EM has been widely used in clustering, Bayesian networks with hidden variables, and hidden Markov models. In general, EM computes expected values of hidden variables for each data element and then recomputes the parameters using the expected values. Let x be the observed values, element Z denote the hidden variables, and let H be parameters of the probability model. The EM algorithm is:

$$H^{i+1} = \text{argmax}_{\text{H}} \sum_{z} P(Z = z \mid x, H^{(i)}) L(x, Z = z \mid H)$$

In the EM algorithm, the E-step computes the summation, which is the expectation of the log likelihood of complete data with respect to the posterior distribution over the hidden variables. The M-step is the maximization of expected log likelihood with respect to the parameters.

### Hidden Markov Model

A Hidden Markov Model (HMM) is a finite mixture model whose distribution is a Markov chain. Let E represent a sequence of observed data, X be a sequence of states, the transition model is $P(X_i \mid X_{t-1})$ for the first-order process, and the observation model is $P(E_i \mid X_i)$. For any finite t, we have a complete joint distribution over all variables:

$$P(X_0, X_1, \ldots, X_t, E_1, \ldots, E_t) = P(X_0) \prod_{i=1}^{t} P(X_i \mid X_{i-1}) P(E_i \mid X_i)$$

From this model, we can make predictions that are tasks to compute posterior distributions over future states; and find the most likely explanation which is the sequence of states that is most likely to have generated those observations.

HMM is a very general type of time series model since X can be any data structure. Since HMM has a wide range of applications, it is a good tool for time series mining and sequence patterns mining for use in data classification and data clustering methods. HMM is a specific case of a dynamic Bayesian network and can be represented by a dynamic Bayes net with a single discrete state variable. In both the EM algorithm and Markov Chain Monte Carlo (MCMC) data augmentation can be used to estimate HMM parameters.

Bayesian networks are the most popular algorithm used in mining learners information; and many researchers use Bayesian networks to build student models and to extract hidden information. (Conati et al., 2002) used a Bayesian network to manage a student model that provides long-term knowledge assessment, plan recognition, prediction of students' actions, and assessment of students' understanding and learning levels. Baker et al. (2004; 2006) developed a classifier to identify if a student is gaming the system in a way that leads to poor learning. They obtained significant success in detecting student misuse of cognitive tutors from students' response models. (Mayo and Mitrovic, 2001) provided a method to manage a long term student model using a Bayesian network, and to select the next rational tutorial action based on decision theory. (Jonsson et al., 2005) expanded this model and learned the model's parameters by using the HMM method and the EM algorithm. Arroyo et al. (2004) constructed a dynamic Bayesian network that infers unobservable learning variables to predict students' positive and negative learning attitudes from ITS log files. They used the maximum likelihood method to learn conditional probabilities that predict relations among variables. Winters (2005) found the fundamental topic of a course and the proficiencies of each student by using collaborative filtering techniques such as non-negative matrix factorization, sigmoidal factorization, and common-factor analysis. Using dynamic Bayesian network, Gong et al.(2009) create a Knowledge Tracing model to make inferences about students' knowledge and learning based on self-discipline. He found that high self-discipline students had significantly higher initial knowledge. Using Markov Decision Process technique, Stamper and Barnes (2009) promise a domain-independent use of data mining method to automatically generate adaptive hints in an intelligent tutor system. Based on the usage data and marks information, Romero et

al.(2008) compared different data mining methods to classify students, and found the classifier model appropriate for educational use with both accuracy and comprehension. Bayesian graphical models are commonly used in building student models. Desmarais et al. (2008) designed a Naive Bayes Framework to explore the heuristic selection leading to better performance. Using data-driven modelling, Mavrikis (2008) compared Bayesian network, decision trees and logistic regression methods to reveal the possible educational consequences.

## 1.4.5. Clustering

Clustering is a process of grouping data into distinct clusters (groups, categories, or subsets) based on similarity among data. There is no universally accepted definition of clustering. Most researchers describe a cluster by considering the internal homogeneity and external separation [7] for example features of objects are similar to each other in the same cluster, while features of objects are not similar when they are in different clusters. Three necessary steps in implementing clustering algorithms are: the representation of data, the deffinition of proximity among data, and applying clustering methods. The first step obtains features or attributes to appropriately represent raw data. The second step chooses a measurement to judge the similarity between two pieces of data such as Euclidean distance between two data points, or Mahalonobis distance. Both the similarity and dissimilarity should be examinable in a clear and meaningful way (Xu and Wunsch, 2005). When data becomes more complex such as in trees or sequences, determining similarity or dissimilarity becomes more difficult. Since clustering algorithms are highly data dependent, the choice of a measurement for similarity has a profound impact on clustering quality.

Here are two simple mathematical descriptions of two typical types of clustering. Given a set of input $X = x_1,\ldots,x_j,\ldots x_n$, where $x_j = (x_{j1}, x_{j2},\ldots,x_{jd})^T \in R^d$ and each measure $x_{ji}$ is said to be a attribute.

- Partitional clustering seeks a K partition of X, $C = C1, \ldots, Ck (K \leq N)$, such that
  1. $C_i \neq \phi, i = 1,\ldots,k;$
  2. $\bigcup_{i=1}^{k} C_i = X;$
  3. $C_i \cap C_j = \phi, i, j = 1,\ldots,K$ and $i \neg j.$

- Hierarchical clustering attempts to construct a tree like nested structure of X, $H=H_1,H_2,\ldots,H_Q(Q \leq N)$, such that $C_i \in H_m$, $C_j \subset H_l$, and m limply $C_i \in C_j or C_i \cap C_j = \phi$ for all i,j $\neq$ i,m,l = 1,…,Q.

For clustering algorithms, there are four basic steps in the clustering procedure. These steps are closely related to each other and affect the derived clusters.

- Feature selection: Feature selection chooses distinguishing features from a set of candidate features or transforms to new and useful features from original ones. It should be used to reduce noise, to be easy to select and interpret.
- Clustering algorithm selection: The selection of clustering algorithm is essentially the selection of a proper proximity measure. The proximity measure directly impacts on the formation of the resulting clusters. The partition of clusters is an optimization problem once the proximity measure is determined. However, there is no clustering algorithm that can be universally used to solve all problems. Therefore, it is very important to select an appropriate clustering strategy to solve the problem properly.
- Clustering validation: The different algorithms usually have different clusters. Therefore, it is necessary to use effective evaluation criteria to provide a certain degree of confidence for resulting clusters from selected clustering algorithms. Usually, there are three types of testing criteria: external indices, internal indices, and relative indices and they are defined from three clustering structures, known as partitional clustering, hierarchical clustering, and individual clustering.
- Results interpretation: The final purpose of clustering is to provide meaningful insights to solve a problem so that the partition needs to be interpreted carefully.

**Clustering Algorithm Categories**

Traditionally, accepted clustering algorithms are classified as hierarchical clustering or partitional clustering based on the properties of generated clusters. [8] In recent years, there are new categories of clustering algorithms based on techniques, theories and applications. The following is a mixture of these categorizations.

- Hierarchical clustering
  - Agglomerative: single linkage, complete linkage, median linkage, centroid linkage...
  - Divisive: divisive analysis, monotheistic analysis...
- Squared error-based (Vector quantization): K-means, genetic K-means algorithm...
- Mixture density-based: expectation-maximization, Gaussian mixture density decomposition...
- Graph theory-based: highly connected subgraphs...
- Fuzzy: Fuzzy c-means, fuzzy c-shells...
- Neural network-based: learning vector quantization...
- Kernel-based: support vector clustering, kernel k-means...
- Sequence data: sequence similarity...
- Large-scale data: CURE...
- High dimensional data: PCA...

The techniques-based clustering algorithms such as graph based, neural network based and kernel based clustering algorithms can be used both in partition and hierarchical clusterings. We will brifley discuss the k-means clustering and expectation-maximization clustering in the following sections.

### Distance and Similarity Measures

The first important issue of clustering algorithms is how to measure the distance or the similarity between two objects, or à pair of clusters in order to determine the closeness. In mathematics, a distance or similarity function on a data set is defined to satisfy the following conditions:

1. Symmetry: $D(x_i, x_j) = D(x_j, x_i)$;
2. Positivity: $D(x_i, x_j) \geq 0$ for all $x_i$ and $x_j$ ;
3. Triangle inequality: $D(x_i, x_j) \leq D(x_i, x_k) + D(x_k, x_j)$ for all $x_i$, $x_j$ and $x_k$;
4. Reexivity: $D(x_i, x_j) = 0$ iff $x_i = x_j$ ;

There are some typical measures which can be used in determining the distance or the similarity.

The Euclidean distance is the most commonly used metric to measure the distance between a pair of elements and a pair of clusters such as in hierarchical linkage

| | |
|---|---|
| Minkowski distance | $D_{ij} = (\sum_1^d \mid x_{il} - x_{jl} \mid^{1/n})^n$ |
| Euclidean distance | $D_{ij} = (\sum_1^d \mid x_{il} - x_{jl} \mid^{1/2})^2$ |
| City-block distance | $D_{ij} = \sum_1^d \mid x_{il} - x_{jl} \mid$ |
| Sup distance | $D_{ij} = \max_{1 \le l \le d} \mid x_{il} - x_{jl} \mid$ |
| Pearson correlation | $D_{ij} = (1 - r_{ij}) / 2$ |

**Table 1.1** Typical measures of distances

Clustering, squared error-based k-means clustering and mixture densities EM clustering. (Han and Kamber, 2001) The Minkowski distance, City-block distance and Sup distance are normally used in fuzzy clustering algorithms.

**Partitional method**

A Partitional method divides a dataset into k clusters. K-means and k-medians are two popular square-errors based hard partitional clustering methods that are iterative algorithms to minimize the least square error criteria. The mode-seeking method and grid-based methods are density based clustering methods that regard clusters as dense regions of data or grid of data structure in the data space separated by regions of relatively low density. Graph based methods transform clustering problems into a combinational optimization problem that can be solved by using graph algorithms and related heuristics such as cutting the longest edge of the minimum spanning tree to create clusters of nodes. Mixture distribution models assume data are generated based on probabilities that are dependent on certain distributions such as Gaussian distribution. A fuzzy c-means algorithm assigns a degree of certainty that specific data belong to certain clusters.

### Hierarchical method

A hierarchical method creates a hierarchical decomposition of data and uses methods such as the following. The agglomerative (bottom-up) approach starts with each data element forming a separate group and then merges groups that are closest according to some distance measure (e.g. single link and complete link methods). The divisive approach starts with all data in the same cluster and then splits into smaller clusters in each iteration according to some measures.

Ayers et al. (Ayers et al., 2009) compare the performance of the three estimates of student skill knowledge under a variety of clustering methods including hierarchical agglomerative clustering, K-means clustering and model-based clustering using simulated data with varying levels of missing values. To support teacher's reection and adaptation in intelligent tutoring systems, Ben-Naim et al. (2009) used a solution trace graph method to help teachers understand students' behaviours in an adaptive tutorial by post-analysis of the systems' data-log. In constructing an intelligent tutoring system, finding the level of learners' background knowledge is an unresolved problem. Antunes (2008) used sequential pattern mining methods to automatically acquire that knowledge. Baker and Carvalho (2008) compared hierarchica classifiers and non-hierarchical classifiers to identify exactly when a special behaviour occurs in the gaming detector systems.

### K-means Clustering Algorithm

The k-means clustering algorithm is the best-known squared error-based algorithm which minimizes the squared error while partitioning elements into k clusters. The basic steps of k-means clustering algorithm are:

1. Initialize k clusters randomly or based on some prior knowledge. Calculate the cluster properties $M = [m_1,\ldots,m_k]$
2. Assign each element into the nearest cluster $C_i$ based on the distance between elements and clusters
3. Recalculate the cluster properties based on the current partitioning
4. Repeat steps 2 and 3 until convergence is reached

The k-means clustering algorithm is very simple and works well in many practical problems. It is well studied and can be easily implemented in many applications. Obviously, the k-means clustering algorithm also has some disadvantages and lots of variants of k-

means clustering algorithms have been developed in order to overcome its drawbacks. The main drawbacks include lack of a universal method to identify the initial partitions and the number of clusters, lack of a guarantee to converge to a global optimum, and sensitivity to outliers and noise.

### Mixture Gaussian Model

The mixture Gaussian clustering model is a classic statistical method with an assumption that data is generated based on a mixture of k components that have multivariate Gaussian distributions.

Expectation Maximization is the most popular method of mixture densitiesbased clustering. From the probabilistic view, clustering presumes that the data objects are generated from a mixture distribution such that this distribution has k components and each component is a distribution in its own right. If the distributions are known, finding the clusters of a given data set is equivalent to estimating the parameters of underlying models. Let the random variable C denote the cluster components, with value 1,...,k; then, the mixture distribution (probability density) is defined as

$$p(x \mid \theta) = \sum_{i=1}^{k} p(x \mid C_i, \theta_i) P(C_i)$$

Here, $\theta$ is the parameter vector, $P(C_i)$ is the prior probability and $\sum_{i=1}^{k} P(C_i) = 1$, and $p(x \mid C_i, \theta_i)$ is the component density. The posterior probability for assigning a data point into a cluster can be calculated with Bayes's theorem while the parameter vector $\theta$ is decided. Multivariate Gaussian is the natural choice to construct the mixture density for continuous data.

The maximum likelihood estimation is an important statistical method to estimate parameters because it maximizes the probability of generating all the observations by giving the joint density function in logarithm form

$$\iota(\theta) = \sum_{j=1}^{N} lnp(x_i \mid \theta)$$

The best estimate can be achieved by solving the log-likelihood equations.

The expectation maximization algorithm is the most effective method to approximate the maximum likelihood since the analytical solution of the likelihood equation cannot be obtained in most circumstances. The standard expectation maximization algorithm calculates a series of parameter estimates $\theta^o, \theta^1, ..., \theta^T$ through the following steps:

1. initialize $\theta^o$ and set t = 0
2. e-step: compute the expectation of the log-likelihood

$$Q(\theta, \theta^t) = E[Logp(C \mid \theta) \mid X, \theta^t];$$

3. m-step: select a new parameter estimate that maximizes the Q-function,

$$\theta^{t+1} = \arg\max_{\theta}(\theta, \theta^t);$$

4. Increase t = t+1; repeat steps 2 - 3 until the convergence condition is satisfied.

In ITS systems, following work that identifies which items produce the most learning, Pardos and Heffernan (2009) proposed a Bayesian method using similar permutation analysis techniques to determine if item learning is context sensitive and if so which orderings produce the most learning. Nugent et al. (2009) has implemented an approach to estimate each student's individual skill knowledge. The method has adapted several clustering algorithms including hierarchical agglomerative clustering, k-means clustering, and model-based clustering to introduce an automatic subspace method which first identifies skills on which students are wellseparated prior to clustering smaller subspace. Using data mining techniques, Sacin et al. (2009) propose the use of a recommendation system to help students take academic decisions based on available information.

## 1.4.6. Regression

Regression problems involve the prediction of continuous outcomes. Linear regression is the simplest form of regression in which data is modelled using a straight line. The coefficients of linear regression can be solved by using the least squares method. Multiple regression is an extension of linear regression that involves more than one predictor variable. Nonlinear regression such as polynomial regression can be transformed to linear regression.

The common regression methods used in data mining are: linear regression, polynomial regression, robust regression, cascade correlation, radial basis functions (RBFs), regression trees, multilinear interpolation, group methods data handling (GMDH), and multivariate adaptive regression splines (MARS).

Regression methods were used by Feng (2005) to predict learners' knowledge levels from error sources. To determine the relative efficacy of different instructional strategies, Feng et al. (2009) used learning decomposition, an educational data mining technique, and logistic regression methods to determine how much learning is caused by different methods of teaching the same skill.

### 1.4.7. Support Vector Machines

Support Vector Machines (SVMs, also called kernel machines) use efficient training algorithms and can represent complex and/or nonlinear functions. The idea of SVMs is that data will always be linearly separable if they can be mapped into a space with sufficiently high dimensions. Various kernel functions can be used to map data into some corresponding feature spaces, which have support vectors to hold separating planes.

Support vector machines were used by Joachims (2001) in building learning models of text calssification. A method called"power law of learning" was used by Koedinger and Mathan (2004) in determining whether students can flexibly use knowledge learned from a course. Makrehchi and Kamel (2005) combined together a vector space machine, weight regression, and similarity measures to build a social network in a small community.

Other data mining algorithms include instance-based learning (such as the nearest-neighbor model and the kernel model), neural networks, and distributed data mining methods.

Chen et al. (2004) proposed an approach to automatically construct an e-textbook via web content mining. They used a ranking strategy to evaluate the web pages and extract concept features from web contents. To extract from the discussion forum with viewable and useful information for instructors, Dringus et al. (2005) proposed a strategy using text mining to assess an asynchronous discussion forum. Their experiment shows that using text mining techniques in the query process could improve the instructor's ability to evaluate the progress of a threaded discussion. Hershkovitz and Nachmias (2009) investigate the consistency of students' behaviours regarding their pace of actions over sessions within an

online course. The results suggest that pace is something not consistent. Web mining methods also were used by Merceron and Yacef (2003) to improve learning and teaching. By combining data mining and text mining methods together, Nagata et al. (2009) proposes a novel method for recommending books to pupils based on only local histories with an accuracy 60% performance. Rai et al. (2009) propose a technique that directly produces parameters to improve the understanding students' knowledge. Ritter et al. (2009) find a way to reducing the knowledge tracing space and parameters to improve great efficiency gains in interpreting specific learning and performance from students' models. Using heuristic classification model, Hubscher and Puntambekar (2008) described a method to integrate the knowledge discovered with data mining techniques, pedagogical knowledge and linguistic knowledge together.

## 1.5. Main Applications of EDM methods

EDM methods have a large variety of applications, including individual learning from educational software, computer supported collaborative learning, computer-adaptive testing and the factors that are associated with student failure or non-retention in courses.

Many of these methods can be also applied to study teacher's behaviours, learning about teacher's criterions and predicting the success or failure of some course, also based on previous students feedback and grades.

One of the most important applications is the improvement of student models. They represent information about student's characteristics, such as student's current knowledge, motivation, attitudes... Researchers have used EDM methods to infer whether a student is gaming the system [9], experiencing poor self-efficacy, off-task, or even if a student is bored or frustrated.

Another important application is the discovery or improvement of models of a domain's knowledge structure. Researchers have been able to develop a way to discover domain structure models, directly from data, through combinations of psychometric modelling frameworks, like algorithms which can automatically discover a QMatrix from data or algorithms for finding partial order knowledge structure (POKS) models that explain the interrelationships of knowledge in a domain.

In another area of application, EDM methods has been helping in studying pedagogical support, to discover which types of pedagogical support are most effective [10] ,

Learning decomposition is a popular method for studying pedagogical support, relating a student‟s later success to the amount of each type of pedagogical support the student received up to that point.

## 1.6. Open Issues in EDM

Being EDM a relatively recent subject, it still has many open issues.Dealing with the amount of data that is collected and the small amount that is taken as a sample is still an issue. In a classroom there are, at most, few hundreds of students and they probably don't take the same exercises. Collecting years of data can be an option, although course offerings change from year to year. The best option is to be careful and try to avoid measures and parameters where sample size has a big effect on the results. Far from consensus is also to know which data to log and how to do it.

The acquisition of background knowledge, like teaching strategy and student's frequent behaviour, has been a main problem in EDM, although a methodology was proposed in [11]. However, an open issue remains, which is the correlation between students‟ results and a teaching strategy. If the strategy and the expected results are known previously, the methodology can be used to identify which steps do not work as expected.

There are still problems about dealing with real temporal data, there isn't consensus about the best solution and even if Sequential Pattern Mining is one of the few techniques that lead with it, it still discovers large amounts of useless patterns in the real world problems.