Discovering Student Preferences in E-Learning

Cristina Carmona[1], Gladys Castillo[2], Eva Millán[1]

[1] Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, Spain
{cristina,eva}@lcc.uma.es
[2] Department of Mathematics, University of Aveiro, Portugal.
gladys@mat.ua.pt

Abstract. Nowadays modeling user's preferences is one of the most challenging tasks in e-learning systems that deal with large volumes of information. The growth of on-line educational resources including encyclopaedias, repositories, etc., has made it crucial to "filter" or "sort" the information shown to the student, so that he/she can make a better use of it. To find out the student's preferences, a commonly used approach is to implement a decision model that matches some relevant characteristics of the learning resources with the student's learning style. The rules that compose the decision model are, in general, deterministic by nature and never change over time. In this paper, we propose to use adaptive machine learning algorithms to learn about the student's preferences over time. First we use all the background knowledge available about a particular student to build an initial decision model based on learning styles. This model can then be fine-tuned with the data generated by the student's interactions with the system in order to reflect more accurately his/her current preferences.

1 Introduction

Student modeling is the process whereby an adaptive learning system creates and updates a student model by collecting data from several sources *implicitly* (observing user's behaviour) or *explicitly* (requesting directly from the user). Traditionally, most of student modeling systems have been limited to maintain assumptions related with student's knowledge (acquired during evaluation activities) not paying too much attention to student's preferences. However, over the last years the growth of on-line educational data (encyclopaedias, repositories of learning resources, etc.) has made it necessary to "filter" or "sort" the information shown to the student, so he/she can make a better use of it. Since one of the first works in e-learning that suggested the use of learning styles for determining the student's preferences regarding multimedia materials [1], this research direction has been getting more and more attention.

Learning styles can be defined as the different ways a person collects, processes and organizes information. It is a fact that different people learn differently: some people tend to learn by doing, whereas others tend to learn concepts; some of them like better written text and/or spoken explanations, whereas others prefer learning by visual information (pictures, diagrams, etc). On the other hand, different learning resources can explain the same concept by implementing different learning activities in different multimedia formats. For example, a *geometric theorem proof* can be supported by a *static text* that describes this proof, or by an *animation* that explains this proof step by step. For a student who prefers visual representation, this proof should be presented as an animation. On the contrary, for a student who prefers verbal presentation the proof should be presented as a text. Thus, the student's learning style can influence the student's preferences for a particular learning resource. Therefore, an e-learning system could use the favourite learning style of a particular student to select the more interesting resources.

Students' learning styles can be acquired using one of existing psychometric instruments. Then, some decision rules are defined to establish the matches between learning styles and educational materials. Following this idea, some educational hypermedia systems have implemented several learning style models in order to better adapt their educational resources to their users: AES-CS [4] implements the Witkin's Field Dependent/ Field Independent Model to adapt the amount of control (program vs. learner), instructional support, navigational tools and feedback to assessment questions in Multimedia Technology Systems; INSPIRE [5] implements the Honey and Mumford model to adapt the method and order of presentation of multiple types of educational resources within educational material pages; iWeaver [6] implements the Dunn and Dunn model to adapt navigation and content presentation in an adaptive hypermedia system; TANGOW/WOTAN [7], WHURLE [8], CS383 [1] implement the Felder and Silverman model to adapt content presentation to the student.

However, as argued in [9]: *"There are no proven recipes for the application of learning styles in adaptation"*. In our opinion, this happened due to several issues: First, the information about the learning style acquired by psychometric instruments encloses some grade of uncertainty (it is very difficult to identify how a person learns). In spite of it all, in the majority of implemented approaches the assumptions about the student's learning style, once acquired, are no longer updated in the light of new evidences obtained from the student's interactions with the system. Second, the rules that match a learning style with a learning resource included in the decision models do not change either. This means that once the rules are defined, they are kept fixed, even when student behaviour might suggest that something could be wrong with these assumptions. Thus, the model is used for adaptation but it is unable to adapt itself in the light of new information.

But it could be the case that, during the interaction with the system, the student could change his/her preferences for another kind of learning resource that no longer matches with his/her inferred learning style. The problem of changes of the users' preferences is known as *concept drift* and has been discussed in several works about the use of machine learning for user modeling [10][11]. Concept drift can occur either because the acquired learning style information needs to be adjusted or because the student simply changes his/her preferences. In these scenarios, *adaptive decision models*, capable of better fitting the current student's preferences, are desirable.

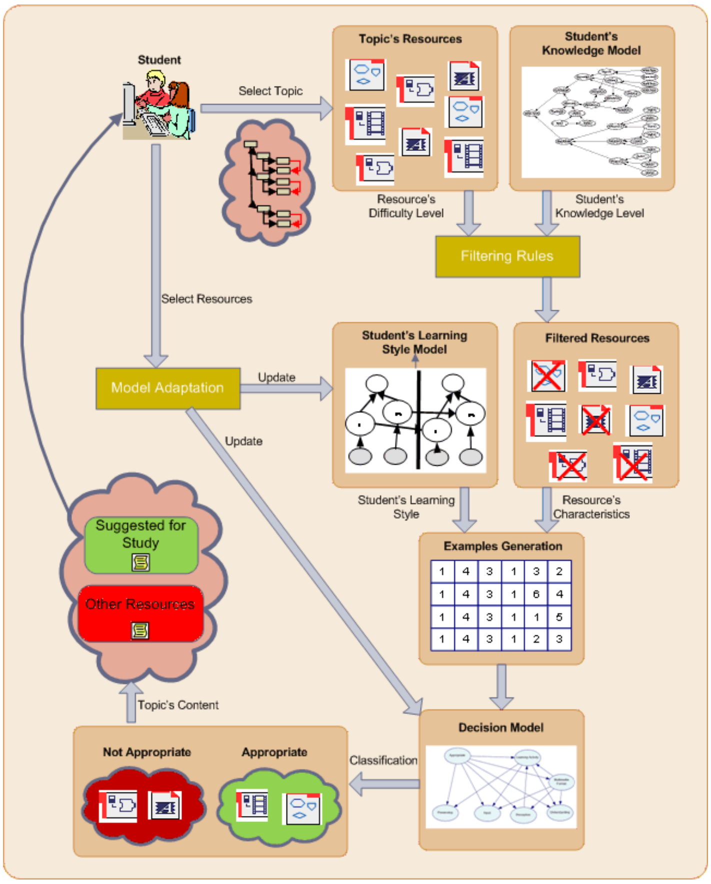
There are other adaptive e-learning systems that model student's preferences using machine learning techniques, like MANIC [12], where the student's learning style is not directly used, but it is approached by the student's preferences concerning the type of media, the instructional type and the level of abstraction of the content objects. The tutor learns the student's preferences via machine learning by observing which objects he/she shows or hides (a *stretch-text* technique is used to adapt the presentation). A Naïve Bayes classifier (NB) [17] predicts whether a student will want certain content objects. Those objects predicted as "*wanted*" will be shown to the user, while the others will not be shown.

The main difference between our approach and other related approaches is that we try to adapt and fine-tune the initial acquired information about the student's learning style and preferences by observing the student's interactions with the system (these observations provide the training examples that we attempt to incorporate to the current decision model). The rationale is as follows: we use all the background knowledge available to build an initial learning style model and decision model for each particular student. We design the learning style model using a Dynamic Bayesian Network (DBN) [13] (the structure and parameters are elicited a priori) that represents the Felder-Sylverman Learning Style Model (FSLSM) [2]. The initial beliefs about the learning styles can be acquired explicitly if the student chooses to answer to the Index of Learning Style Questionnaire (ILSQ) [3], otherwise, in the absence of information, a uniform distribution is used. Then, the student's selections are set as evidences in the DBN, triggering the evidence propagation mechanism and getting up-to-date beliefs for the learning styles. For the decision model, we use a model based on Bayesian Network Classifiers (BNC) [14] that represents the matches between learning styles and learning resources in order to decide if a resource is interesting to a student or not. We learn an initial classifier (structure and parameters) from data randomly generated by some pre-defined rules When the student selects a resource (and eventually gives feedback) we will incorporate this information to the model so that the latest observations are always more important than the oldest ones, thus reflecting more accurately the current preferences. Moreover, our decision model is adaptive in the sense that it is capable of adapting quickly to any change of the student's preferences. If a concept drift is observed, the model is adapted accordingly. This proposal is an improvement of the approach proposed in [15], where the learning style once acquired was no more refined and the decision model was modelled using an adaptive NB classifier. In this current proposal we use a DBN for modeling learning styles and a 2-DBC [16] classifier to initialize the decision model.

In the rest of the paper, we first explain the whole process aimed at selecting the learning resources to be shown to the student each time he/she makes a topic selection. Next, we explain the design of the learning style model and the decision model. Finally, we conclude with a summary and a description of ongoing and future work.

2 Selecting the Suitable Learning Resources

This paper is focused on the definition of the learning style model and the decision model that will be explained later in the following sections. But, for a better understanding on how these two models are used for adapting to the user's preferences, we first explain the whole process aimed at selecting the suitable learning resources for a given topic according to the *student's characteristics* (knowledge level, learning style and preferences) and the *characteristics of the resources* (learning activities and multimedia format).



1	4	3	1	3	2
1	4	3	1	6	4
1	4	3	1	1	5
1	4	3	1	2	3

Fig. 1 The selection of learning resources task

The whole process is shown in Fig. 1, and is performed according to the following steps:

- *Filtering***:** when a student selects a topic we apply some deterministic filtering rules to obtain the learning resources for this topic. This filtering process is performed according to the matches between the resource's difficulty level and the student's knowledge level.
- *Prediction***:** using the current decision model, each filtered resource is classified as '*appropriate*' or '*not appropriate*' for the student. With this purpose, examples including the learning style features (obtained from the learning style model) and the resource's characteristics are automatically generated and classified by the decision model. As a result the set of available resources is partitioned into these two classes.
- *Decision***:** since the classifier returns probabilities, all the resources of the same class can be ranked. Then, a document is sent to the student including two

separated ranked lists: *Resources suggested for study* (those classified as *appropriate*) and *Other resources for study* (those classified as *not appropriate*).

- *Adaptation*: when the student selects a resource in one of the two lists we assume that this resource is interesting to the student not by its content (since all the shown resources must explain the same concept), but by the learning activity and the multimedia format that this resource represents (each learning resource implements a learning activity in a multimedia format). Moreover, the user can explicitly rank a resource in order to obtain some confidence levels about how much does she/he like it. This way we can obtain a new labelled example that can be used to adapt both the learning style model and the decision model, accordingly.

3 The Learning Style Model

We have adopted the Felder-Sylverman Learning Style Model, since it is one of the more successful models and has been implemented in many e-learning systems. We use a DBN[1] to model the learning styles. A new time slice is instantiated whenever new evidences about the preferences of the student arrive (student's selections). Fig. 2 shows two time slices of a high level description of this DBN. The shaded node represents a random variable for which evidence is available to update the student model at a given time slice. We consider that the student's learning style influences the student's preferences and these preferences influence the student's selections. Besides, current preference for the learning resources depends on the previous preference.

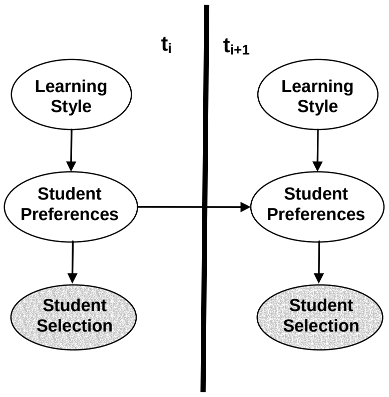


Fig. 2 DBN for modeling learning styles

We initialize this model with the scores obtained by the student in the ILSQ. Then, the student's selections are set as evidences in the DBN, triggering the evidence propagation mechanism and getting up-to-date beliefs for the learning styles. That

[1] A DBN is a model to describe a system that is dynamically changing or evolving over time. This model enables users to monitor and update the system as time proceeds.

makes it possible to refine the initial values for the student's learning style, becoming more confident as the student interacts with the system.

4 The Decision Model

The decision model helps to determine whether a given resource is appropriate for a specific learning style or not. This model uses a BNC and its behaviour is quite similar to a content-based recommender system[2]. The information about the resource (the item to recommend) and the user's learning style (the user's features) are presented to the classifier as input, having as output a probability that represents the appropriateness of the resource for this student (or how interesting the item is for this user). There are two issues that are crucial in the definition of the decision model. First, the *cold-start problem,* which is the problem of obtaining the data to build the initial model. Second, the procedure for updating the model in the light of new data.

To build the initial model, the system's authors must firstly establish the rules to match learning styles with the resource's characteristics in order to determine which resources are more appropriate to a particular learning style. In this implementation these rules are extracted from Table 1. We consider 6 learning activities (Lesson Objective, Simulation, Conceptual Map, Synthesis, Explanation and Example) and 6 multimedia formats (Text, Image, Audio, Video, Animation and Hypertext).

Table 1 Learning Resource Components and FSLSM

a. Learning Activities

	VIS	VER	SEN	INT	SEQ	GLO	ACT	REF
Lesson Objective	□	☑	□	☑	□	☑	□	☑
Simulation	☑	□	☑	□	☑	□	☑	□
Conceptual Map	☑	☑	☑	☑	☑	□	☑	☑
Synthesis	☑	□	□	☑	□	☑	☑	□
Explanation	□	☑	☑	□	☑	□	□	☑
Example	☑	□	☑	□	☑	☑	☑	□

b. Multimedia Formats

	VIS	VER	SEN	INT	SEQ	GLO	ACT	REF
Text	□	☑	□	☑	☑	□	□	☑
Image	☑	□	☑	☑	□	☑	□	☑
Audio	□	☑	☑	☑	☑	□	□	☑
Video	☑	☑	☑	☑	☑	□	□	☑
Animation	☑	□	☑	□	☑	□	□	☑
Hypertext	□	☑	□	☑	☑	☑	□	☑

[2] A recommender system tries to present to the user the information items he/she is interested in. To do this the user's profile is compared to some reference characteristics. These characteristics may be from the information item (the content-based approach) or the user's social environment (the collaborative filtering approach).

After that, the predefined matching rules are used to generate some training examples. The examples are described through 6 attributes: the first four represent the *student's learning style* and the last two represent the *learning resource*. The possible values for each attribute are presented in Table 2. For instance, the example *1,4,3,1,6,5,1* means that a student, with a *strong preference* for VISUAL, a *moderate preference* for INTUITIVE, a *mild preference* for SEQUENTIAL and a *strong preference* for ACTIVE, likes a resource implementing the learning activity EXAMPLE in the format ANIMATION. Finally, the generated examples can be used to learn the model that gives the minimum error rate, that is, to find the best classifier. Therefore, the acquired information about the student's learning style helps us to initialize the decision model.

Table 2. Establishing the Attributes and their Possible Values

Attributes	Values
Input	visualStrong (1); visualModerate (2); inputMild (3); verbalModerate (4); verbalStrong (5)
Perception	sensingStrong (1); sensingModerate (2); perceptionMild (3); intuitiveModerate (4); intuitiveStrong (5)
Understanding	sequentialStrong (1); sequentialModerate (2); undertandingMild (3); globalModerate (4); globalStrong (5)
Processing	activeStrong (1); activeModerate (2); processingMild (3); reflectiveModerate (4); reflectiveStrong (5)
Learning Activity	LessonObjective (1); Simulation (2); ConceptualMap (3); Synthesis (4); Explanation (5); Example (6)
Multimedia Format	Text (1); Image (2); Audio (3); Video (4); Animation (5); Hypertext (6)
Class	Appropriate (1); Not_ appropriate (0)

We choose the class of k-Dependence Bayesian Classifiers (k-DBCs) [16] to represent our decision model. A k-DBC is a Bayesian Network, with a NB[3] structure and that allows each attribute to have a maximum of k feature nodes as parents. To define the initial model we carried out some experiments with the aim to select the best classifier among the BNCs belonging to the class of k-DBCs (varying k from 0 to 5) that best fits the training examples generated from the pre-defined rules. We generated several datasets using an increasing number of instances (6250, 12500, 18750, 25000 and 31250) and generated 10 samples for each setting. For each dataset, all the possible learning styles are represented. Since there are 4 attributes for the learning style and each one has 5 values, we obtain 5^4=625 different learning styles. We generated datasets with 10, 20, 30, 40 and 50 examples for each learning style. The learning activity and the multimedia format were generated randomly and the obtained examples were classified accordingly to the rules extracted from Table 1.

We then learn different models from the generated data: the NB and several k-DBCs varying k from 1 to 5. To learn the k-DBCs, we apply, in conjunction with a score, a Hill Climbing procedure. In the experiments we use different scores (BAYES, MDL and AIC). Fig. 3 shows the errors obtained with each model and each score. These results are the average value for the 10 samples of the 25000 examples

[3] A NB is a Bayesian Network with a simple structure that has the class node as the parent node of all other feature nodes

(since with different sizes of datasets we obtain very similar results). The best model found was a 2-DBC using the BAYES score. As observed, from k>2 the accuracy does not improve significantly, which may indicate that we found a 2-degree of dependence in these domains.

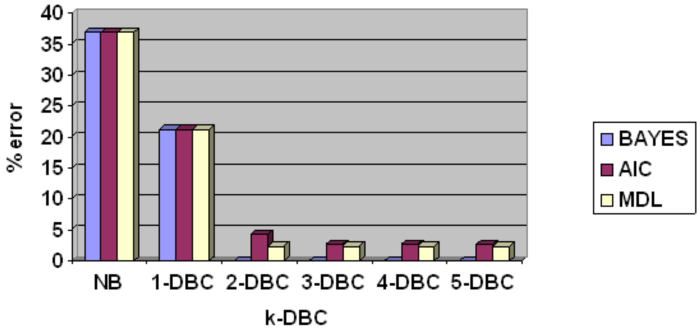


Fig. 3 Percentage of error with each model

The structure of the best model is shown in Fig 4. In addition to the relationships between the class and the attributes, we found other dependences between the attributes. For instance, the dependences between the multimedia format and all the dimensions of the learning style; the dependence between the learning activity and almost all the dimensions of the learning style and the dependence between a dimension (Perception) and the learning activity.

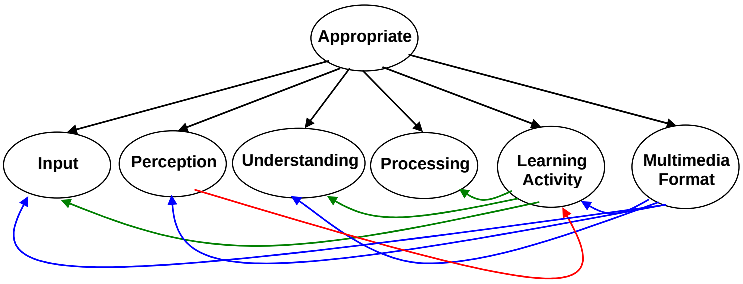


Fig. 4 Initial decision model

During the further interactions of the user with the system, the initial model is adapted using the data generated from the user behaviour. In order to compose the required examples with the correct class we need to obtain some feedback about how much does the student like/dislike a particular resource. In principle, there are two kinds of feedback: *positive examples* (items liked by the user) and *negative examples* (inferring features which the user is not interested in). We propose to obtain positive examples implicitly by observing the visited resources. However, obtaining a relevant set of *negative examples* is more difficult. To this aim we explicitly propose to the user to rate the resources (as *very good*, *good*, *bad*, and *very bad*). Whenever we obtain new labelled examples they can be used to update the model. Sequential updating of the parameters of BNs is straightforward: it only requires a simple scan through all the new examples in order to increment the frequency counters.

Nevertheless, we are very interested in adapting the model in such a way that the most recent observations gathered through relevance feedback represent the current user's preferences better than the older ones. To this end, we currently work on the adaptation of the Iterative Bayes (IB) algorithm [18] for BNC to this particular task.

IB performs an optimization process based on an iterative updating of the BN's parameters. In each iteration, and for each example, the corresponding conditional probabilities are updated so as to increase the probability on the correct class. The rationale is as follows: given an example, an increment is computed and added to all the corresponding counters of the predicted class and proportionally subtracted from the counters of all the other classes. If an example is correctly classified then the increment is positive and equal to 1−P(predicted|X), otherwise it is negative. Experimental evaluation using a NB classifier showed consistent reductions of the error rate. But the most important characteristic of the IB is its ability to adapt the model to new data. It was proved in [15] that this ability is very useful to deal with concept drift scenarios.

At the present we propose a modification of the IB algorithm. The main idea is to use the student's ranks instead of the categorical class values for the adaptation procedure. We consider different increment values according to the *quantitative differences* between the observed class and the predicted class. For instance, if a learning resource is classified as *appropriate* with a high probability (*very good*) and the student ranks this learning resource as *good*, then we use an increment with a value greater than the value used when the student ranks this resource as *very bad*.

5 Conclusions and Future Work

In this paper we have presented an adaptive user model aimed at discovering the student's preferences about the educational materials over time. This model is very suitable in e-learning systems that need to "filter" the great volumes of information available, so that their users can make a better use of it. To discover the user's preferences we use the information about learning styles represented in the student's learning style model (a DBN). The advantages of using a DBN is that this allows refining the initial beliefs acquired by the ILSQ by observing the student's selections over time thus computing up-to-date learning style for each student. On the other hand, we use an adaptive BNC as the decision model for determining whether a given resource is appropriate for a specific learning style or not. We described the experiments carried out to obtain an initial model thus solving the cold-start problem. For each student we initialize the decision model from data generated from a set of rules that represents the matches between learning styles and multimedia resources. Each individual decision model is then adapted from the observations of the student's selections and ranks over time. Moreover, the model is also able to adapt itself to changes in the student's preferences. At the present we are working on the adaptation of the Iterative Bayes procedure and also on the implementation of some experiments to prove that our approach works properly.

References

1. Carver, C.A., Howard, R.A., Lane, W.D.: Enhancing Student Learning Trough Hypermedia Courseware and Incorporation of Student Learning Styles, *IEEE Transactions on Education*, v.42, nº 1 (1999) 33-38
2. Felder, R.M., Silverman, L.K.: Learning and Teaching Styles in Engineering Education, *Engr. Education,* 78 (7), (1988) 674-681
3. Felder, R.M., Soloman, B.A. *Index of Learning Style Questionnaire*, available online at http://www.engr.ncsu.edu/learningstyles/ilsweb.html
4. Triantafillou, E., Pomportsis, A., Demetriadis, S.: The design and the formative evaluation of an adaptive educational system based on cognitive styles. *Computers & Education, 41* (2003) 87-103
5. Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G. D.: Personalizing the inter-action in a Web-based educational hypermedia system: the case of INSPIRE. *User-Modeling and User-Adapted Interaction 13* (3) (2003) 213-267
6. Wolf, C.: iWeaver: Towards Learning Style-based e-Learning in Computer Science Education. *Proceedings of the Fifth Australasian Computing Education Conference, ACE2003* (2003) 273-279
7. Paredes, P., Rodriguez, P.: The Application of Learning Styles in Both Individual and Collaborative Learning. *Proceedings of the sixth IEEE International Conference on Advanced Learning Technologies, ICALT'06* (2006) 1141-1142
8. Brown, E., Stewart, C., Brailsford, T.: Adapting for Visual and Verbal Learning Styles in AEH. *Proceedings of the sixth IEEE International Conference on Advanced Learning Technologies, ICALT'06* (2006) 1145-1146
9. Brusilovsky P., Millán, E.: User Models for Adaptive Hypermedia and Adaptive Educational Systems. *The Adaptive Web: Methods and Strategies of Web Personalization*, LNCS 4321 (2007) 3 – 53
10. Koychev, I., Schwab, I.: Adaptation to Drifting User's Interests. *Proceedings of ECML2000 Workshop: Machine Learning in New Information Age*, Spain (2000)
11. Webb, G., Pazzani, M., Billsus, D.: Machine Learning for User Modeling. *User Modeling and User-Adapted Interaction*, v.11 (2001) 19-29
12. Stern, M.K., Woolf, B.P.: Adaptive Content in an Online Lecture System. *Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web based Systems, AH2000*, (2000) 227-238
13. Dean, T., Kanazawa, K.: A model for reasoning about persistence and causation. *Computational Intelligence, 5* (1989) 142-150
14. Friedman, N., Geiger, D. and Goldszmidt, M.: Bayesian Network Classifiers. *Machine Learning, 29* (2-3) (1997) 131–163
15. Castillo, G., Gama, J., Breda, A.M.: An Adaptive Predictive Model for Student Modeling. *Advances in Web-based Education: Personalized Learning Environments*, (2005) Chapter IV
16. Sahami, M.: Learning Limited Dependence Bayesian Classifiers. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD-96* AAAI Press (1996) 335–338
17. Mitchell, T.: *Machine Learning*. McGraw Hill (1997)
18. Gama, J.: Iterative Bayes. *Discovery Science - Second International Conference*, LNAI 1721, (1999)