The 7th International Scientific Conference
eLearning and Software for Education
Bucharest, April 28-29, 2011

VIRTUAL TRAINER SOFTWARE: A BAYESIAN NETWORK APPROACH

Razvan-Daniel ALBU
University of Oradea, Faculty of Electrical Engineering and Information Technology, Romania
ralbu@uoradea.ro

Florin POPENTIU-VLADICESCU
University of Oradea , "UNESCO Chair" Department, Academy of Romanian Scientists, Romania
popentiu@imm.dtu.dk

Grigore ALBEANU
"Spiru Haret" University, Bucharest, Romania
galbeanu@yahoo.com

***Abstract**: The rapid expansion of information technologies in almost all sectors of the society induces lifelong learning (LLL) in our days. Learning something new can be an easy and quick task, or a very hard and long time procedure. The maximum efficiency in learning is achieved when you train the right skills at the right time, for the correct period of time (with the right intensity). Based on this idea, we believe that a virtual "software trainer" can help us make proficiency by instruction and practice efficiently ourselves, more on what is new for us. Also we can plan the training, so that we do not waste time on things we already know, or too much time on parts we can learn faster. These arguments could emphasize the future of e-learning.*

***Keywords:** Virtual trainer, Bayesian networks, lifelong learning, e-learning, belief, inference networks.*

I. INTRODUCTION

Lifelong learning (LLL) is the continuous building of skills and knowledge throughout the life of an individual. The rapid growth of information technology in almost all sectors of the society makes lifelong learning to be not just an option but becomes in nowadays a need. Modern e-learning tools, video tutorials, web-based learning, multimedia teaching, improve the learning process and the quality, making it easier, more attractive and can turn on the student's enthusiasm. But how much really improves learning efficiency?

The rapid growth of the technology means more information, more skills needed to learn in order to improve efficiency. When you have to learn something in a small period of time always something new appears .Therefore you must keep your skills updated and the most important element is the efficiency of the learning. Learning something new can be an easy and rapid task, or a very hard and long time one. The maximum efficiency in learning is achieved when you train the right skills at the right time for a correct period of time (with the right intensity). Based on this idea we believe that virtual trainer software can help us train efficiently, more on what is new for us. It can schedule the training, so that we do not waste time on things we already know, or too much time on parts, we can learn faster. This approach could be the future of e-learning. Making an efficient learning plan for humans is a very challenging task because each human brain is a unique neural network and needs specific and particular training. But an optimised training plan that tells us the correct order of training and the correct knowledge is not the end of story. To be really efficient for human brain training must consider key factors like feeding and rest.

This paper discusses a modern virtual trainer architecture based on a Bayesian network that improves learning efficiency. The virtual trainer assists individuals in maximizing their knowledge gains, through the use of the optimised and monitored training protocols. Virtual Trainer can predict and continually update their optimal training frequency and helps trainees to maximize their training progress, at a rate that is only limited by their own unique genetic potential. The main target of the virtual trainer is to help students to pass the final exam of a new training session, from the first attempt. This paper is organised as follows: after a short introduction we expound nearby relative to the choice of the Bayesian networks and argue about probabilistic Bayesian approach. In the next part we describe the virtual trainer architecture, functions and the “brain food” concept. The following section “The analytic module’s mystery” is dealing with the Bayesian networks design which can help students for efficient training advices. In the last sections we illustrate how our Bayesian network works, using Norsys Netica simulator. At the end we present some conclusions and future work developments.

II. BAYESIAN NETWORKS

It is well known that over the last decade the Bayesian network has become an esteemed representation for encoding uncertain knowledge. The techniques for learning Bayesian networks from data, which have been developed, are new and still evolving.

2.1 Why Bayesian networks?

There are a variety of representations available for data analysis. The best-known are: decision trees and neural networks. Also many techniques for data analysis have been proposed, for example: regression, clustering and classification. So what do Bayesian networks and Bayesian methods have to offer? First of all, the one that makes Bayesian networks slightly special is the following: they readily handle situations where some data entries are missing. The Bayesian model encodes dependencies among all variables. As another advantage, Bayesian networks can be also used to learn causal relationships. Therefore they can be used to predict the consequences of an intervention. A Bayesian network model has both a causal and probabilistic semantics. This argument makes Bayesian networks the perfect representation for combining prior knowledge and data. Also the last but not least, Bayesian statistical methods in conjunction with Bayesian networks offer an efficient approach for avoiding the over-fitting of the data.

2.2 Introduction to Bayesian interpretation of probability

First of all we have to state the precise meaning of the probability. A conventional probability is defined as a physical world property. In contrast a Bayesian probability is a property of the person who assigns the probability. It is your degree of belief that the event will take place [1]. The main difference between conventional probability and Bayesian probability is that when we assess the Bayesian probability we do not need repeated trials. For example, to make a visible representation of the repeated tosses of a Rubik cube, onto a rough surface. Every time the cube is thrown, its dimensions will change slightly. Although, the conventional probability that the cube will land with a particular colored face up will not take in considerations the dimension’s changes. On the contrary, the Bayesian simply restricts the attention to the next flip and assigns a probability.

2.3 The methodology for calculating Bayesian probabilities

We denote a variable by an upper-case letter (e.g., X; Y; X_i) and its value by that same letter in lower case (e.g., x; y; x_i). We use $P\ (X = x \mid E)$ or $P\ (x|E)$ to denote the probability that $X = x$ with state of information E. We name $\acute{\O}$ to be a variable whose values $\acute{\o}$ corresponds to the possible accurate value of the conventional probability. We express the uncertainty about $\acute{\O}$ using the probability density function $P\ (\acute{\o}|E)$. We use X_l to indicate the variable representing the result of the attempt to pass a new exam after l training sessions and exams, $l = 1 \dots n + 1$. We use $D = \{X_1 = x_1; \dots X_n = x_n\}$ to represent the set of our observations. As a result, in Bayesian terms, the exam passing problem reduces to computing P (x_{n+1}|D, E) from P (ǿ|E). First we need to obtain the probability distribution for Ǿ given D and background knowledge E. To do so, we use Bayes rule:

$$P\ (ǿ|D,\ E) = \frac{P(ǿ|E)P(D|ǿ,E)}{P(D|E)} \qquad (1)$$

where

$$P\ (D|E) = \int P(D|ǿ,E)\,P(ǿ|E)\,dǿ \qquad (2)$$

Now, we expand the term P (D|ǿ, E), the likelihood function, for binomial sampling [2]. In particular, given the value of Ǿ, the observations in D are mutually independent. The probability of pass (fail) on any one observation is ǿ (1-ǿ). Consequently, Equation (1) becomes:

$$P\ (ǿ|D,\ E) = \frac{P(ǿ|E)ǿ^{p}(1-ǿ)^{f}}{P(D|E)} \qquad (3)$$

where p and f are the number of pass and fails observed in D.

The quantities p and f are said to be adequate statistics for binomial sampling, because they provide a summarization of the data that is sufficient to compute the posterior from the prior [3]. Finally, we average over the possible values of Ǿ. We apply the expansion rule of probability in order to determine the probability that the (n + 1)th exam attempt will be “a pass”:

$$P\ (Xn_{+1}=pass\ |\ D,\ E) = \int P(X_{n+1} = pass\ |\ ǿ,E)P(ǿ|D,E)dǿ = \int ǿP(ǿ|D,E)dǿ \qquad (4)$$

III. VIRTUAL TRAINER SOFTWARE ARCHITECTURE AND FUNCTIONS

The virtual trainer assists students in maximizing their knowledge gains through the use of computer optimized and monitored training protocols. Virtual Trainer's built-in intelligence takes into consideration the training data and also through its analytical modeling notifies the trainee on the necessity to emphasize the training policy, on a specific exercise, to change routines and nutrients, or to take a rest.

The virtual trainer architecture is presented in the Figure 1.

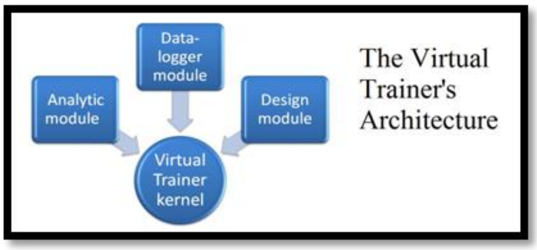


Figure 1. The virtual trainer software architecture

The design module contains three packages: SS (Skills Set), ML (Multimedia Lesson) and TQ (Test-Quiz), as shown in the Figure 2.

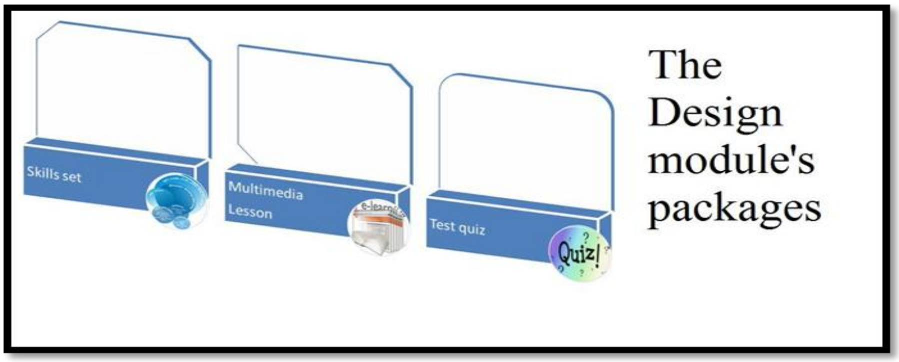


Figure 2. The packages of the design module

Using these packages a teacher can design and create lessons in the following order:

1. First, the teacher must define a skills set, the skills that the student will obtain in this training session. Each skill must be associated with a skills category, for example the skill defined as: “Calculating the probability distribution function” will be linked by the teacher with the skill category “Mathematics”.
2. The next step is to design and create a lesson. This virtual trainer’s software package allows the teacher to write lessons up to 50 pages, to insert images and videos and create simple animations using basic templates.
3. At the end of each lesson the student will take a test. In designing the test questions the teacher must allocate for each question a skill, from the skills set defined at the step 1. After the test, the TQ software package will provide a detailed score report. This report will show not only the final score, but also percentages for each skills category.

The data-logger module records relevant data from student’s tests’ results and from each student’s activity and creates student profiles. A student profile contains data about his activity, such as: tests’ scores, dates when the student has taken the tests, dates when the student have had the training sessions, for each lesson, also the order of training sessions. The data-logger module’s main function is to collect data and schedule the training program of each student based on his activity. Using the recorded data, the data-logger module can notify the student with efficient advices. For example, analyzing the tests’ scores, the virtual trainer software can calculate an average performance rate for each student. When the student starts having lower tests scores, by having an average performance rate, he will be notified to take a rest. The rest period will be at least three days but no more than two weeks, has been calculated on the basis of each student performance rate. A higher performance rate means a shorter rest period. But the greatest advantage of this virtual trainer software is the ability to give the advices about the skills on which the student must increase the training intensity, the recommended training and healthful diet. Those features are functions of the analytic module. The data-logger module records not only relevant data from student’s tests’ results but also the recommended nutrients from each student’s daily diet report.

The virtual trainer’s kernel provides basic services for all other parts. Its responsibilities include the management of the system's required resources and the communication between virtual trainer software’s modules.

3.1 The „Brain food” concept

Recent studies have shown that nutrients may have significantly positive effects on the brain. Actually, some foods can maximize your brain's potential and may remove obstacles in order to optimize the functioning. "Nutrients can have a tremendous impact on brain health," says Gary Small, M.D., director of the Center on Aging and the Memory Clinic at the University of California at Los Angeles and author of The Memory Bible (Hyperion, 2002).

Based on this idea the virtual trainer software tries to help trainees for maximizing their training progress, at a rate that is only limited by their own, unique genetic potential. To achieve this goal, three key elements are used: training strategy, nutrients and rest. Without a training strategy you will work hard and will not know what are your gains. If you will study hard but your brain will not get the needed nutrients, your hard work will bring fraying, instead of progress. Without enough rest but having a hard training, even having an optimal diet, you will have a quick progress but will end soon with overtraining. Overtraining is a physical, behavioral and emotional condition, which occurs when the volume and intensity of an individual's exercise exceeds their recovery capacity. First you will stop making progress, even beginning to lose strength, worst of that, getting a high risk diseases. In combination with folic acid, vitamins B6 and vitamin B12 could help to produce and release neurotransmitters. The brain uses omega-3 fatty acids for forming its cell structure and the carbohydrates for energy. In addition, foods rich in antioxidant nutrients help protect brain cells from free-radical damage caused by environmental pollution. The most important antioxidants for brain are vitamin C, vitamin E and beta-carotene,

The virtual trainer software implements a Bayesian network in the analytic module, which can give advices, about all the three key elements of a successfully training.

3.2 The analytic module’s mystery

A Bayesian network for a set of variables X = {X_1… X_n} consists of two main components [4]. The first one is a network structure S that encodes a set of conditional independence assertions about variables in X. The network structure S is a Directed Acyclic Graph DAG (Directed Acyclic Graph). The second component is a set P of local probability distributions associated with each variable. The nodes in S are in one-to-one correspondence with the variables X. We use x_i to denote both the variable and its corresponding node, also y_i to indicate the parents of node X_i in S, as well as the variables corresponding to those parents. Given structure S, the joint probability distribution for X is presented by:

$$P\ (s) = \prod_{i=1}^{n} P(x_i\ |y_i) \tag{5}$$

The probabilities encoded by a Bayesian network may be a Bayesian approach or conventional one. When we are building Bayesian networks, using only prior knowledge, the probabilities will be Bayesian. When we are learning these networks from data, the probabilities will be conventional.

We begin by determining the variables to the model. One possible choice of variables for our problem is: DER (Detailed Exam Report), DDR (Daily Diet Report), APR (Average Performance Rate), SKC (Skills Categories), N (Nutrients), CR (Change Routine), RP (Rest Period), the set of skills {s_1 … s_n} and the set of foods {f_1 … f_n}. In the next stage of Bayesian-network construction, we build a directed acyclic graph. The DAG encodes assertions of conditional independence [5]. The approach is based on the idea that people can often readily assert causal relationships among variables. In particular, to construct the Bayesian network for a given set of variables, we simply draw arcs from cause variables to their direct effects. In almost all cases, doing so results in a network structure that satisfies the definition Equation (5). In the final step of constructing a Bayesian network, we assess the local probability distributions P (x_i|y_i) [6]. A case study is presented in the next chapter.

The virtual trainee’s analytic module implements the Bayesian network in the Figure 3.

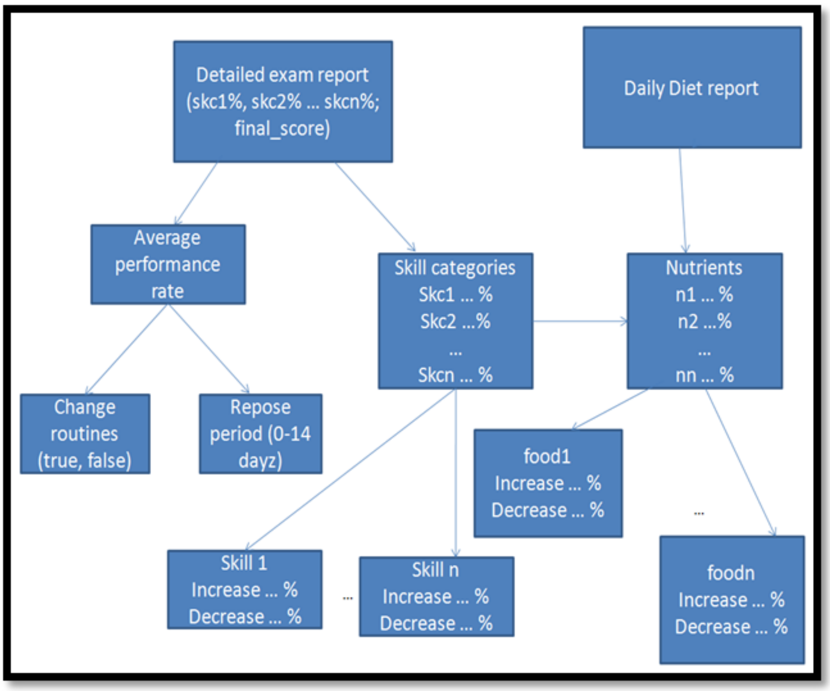

Detailed exam report (skc1%, skc2% ... skcn%; final_score)
Daily Diet report
Average performance rate
Skill categories Skc1 ... % Skc2 ...% ... Skcn ... %
Nutrients n1 ... % n2 ...% ... nn ... %
Change routines (true, false)
Repose period (0-14 dayz)
food1 Increase ... % Decrease ... %
...
Skill 1 Increase ... % Decrease ... %
...
Skill n Increase ... % Decrease ... %
foodn Increase ... % Decrease ... %

Figure 3. The proposed Bayesian network

Based on the detailed exam report and the daily diet report this Bayesian network can make useful advices for an efficient training. The advices includes skills on which the training must be increased, the recommended food and the rest period. Also we receive counsel if we need or not to change de order of training in order to avoid the progress saturation.

IV. A CASE STUDY USING NETICA APPLICATION

In this section we present the proposed Bayesian network implemented using Norsys Netica simulator. Netica is an easy-to-use and also a powerful application, developed by Norsys Software Corporation. Netica facilitates the work with belief networks and influence diagrams. It has an intuitive GUI (Graphical User Interface) for drawing the networks. The associations between variables may be entered in the form of equations, as individual probabilities or learned from data files. Netica uses the fastest and most modern algorithms to perform various kinds of inference. Netica could find the appropriate values or probabilities for all the unknown variables of a new given case of which we have limited knowledge. These values or probabilities may be displayed in a number of different ways, including bar graphs and meters. The case may be saved to a file and later brought back into the network or into a different network, for further querying.

4.1 The proposed Bayesian network

When a network is composed entirely of nature nodes it is called a Bayesian network or a “belief network”. If it also has decision and utility nodes, it is called a decision net or an “influence diagram”. As we can see in Figure 4, the proposed network is a BBN (Bayesian Belief Network). In this example we can see how the Bayesian network can help the student with useful advices. The student passes a networking exam with a final score of 75% and entered the daily diet report in a dialog window of virtual trainer software. Based on student’s exam answers the virtual trainer software calculates the detailed exam report. As we can see the student has a 20% correct answer rate on network configuration questions. This is his second lowest percentage. As a direct consequence the network calculates and tells student the probabilities for his configuration skills:

$$P\ (x_{i=\text{”false”}}=\text{”Implement OSPF”}\ |\ y_{i=\text{”network troubleshooting”}}=10,\ y_{i=\text{”network terminology”}}=30,\ y_{i=\text{”network configuration”}}=20,\ y_{i=\text{”network design”}}=40)=62 \qquad (6)$$

and

$$P\ (x_{i=\text{”true”}}=\text{”Implement OSPF”}\ |\ y_{i=\text{”network troubleshooting”}}=10,\ y_{i=\text{”network terminology”}}=30,\ y_{i=\text{”network configuration”}}=20,\ y_{i=\text{”network design”}}=40)=38 \qquad (7)$$

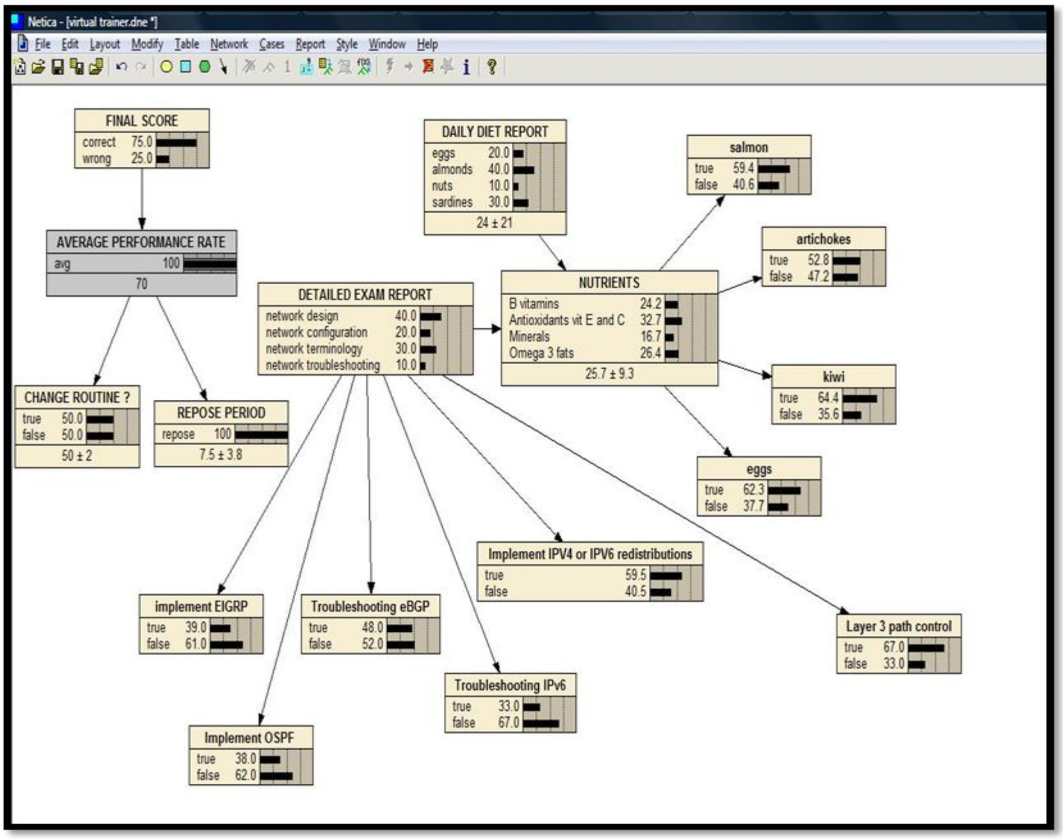
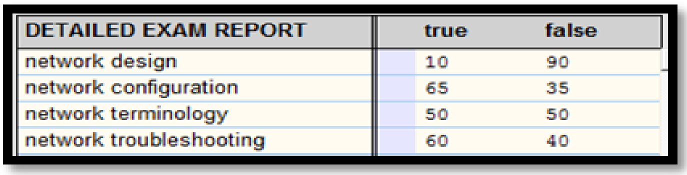


Figure 4. Modeling the proposed Bayesian network with Norsys Netica

This means the student need more training on configuring the OSPF (Open Shortest Path First) protocol, but the virtual trainer software can also provide an answer to the question: how much more training? The answer in this case is 38% more training. As we can see when one calculates the quantity of needed training on configuring the OSPF protocol skill, the Bayesian network consider not only the 10% score, on network troubleshooting questions, but also all the scores; skills on every exam are closely related to each other even if we catalogue then in skill categories. The probabilistic relations between the “Implement OSPF” node and the “Detailed exam report node” are shown in the TABLE 1.

TABLE 1. Probabilistic relations



DETAILED EXAM REPORT	true	false
network design	10	90
network configuration	65	35
network terminology	50	50
network troubleshooting	60	40

In a similar way, the virtual trainer can give advices about the diet and recommended nutrients. As we can see in Figure 4 the virtual trainer will advices the student to increase minerals in his diet:

$$P\ (x_{i=\text{”true”}}=\text{”Kiwi”}\ |\ y_{i=\text{”B vitamins”}}=24.2,\ y_{i=\text{”Antioxidants”}}=32.7,\ y_{i=\text{”Minerals”}}=16.7,\ y_{i\ =\text{”Omega 3 fats”}}=26.4) = 64.4 \qquad (8)$$

Other advices will be about the rest period and the change routines need. Because the student has a good average performance rate of 70% and P (xi =”true” | yi=70) = P (xi=”false”|yi=70) =50, the advised rest period will be 4 days (≈7.5-3.8).

V. CONCLUSIONS AND FUTURE WORK

The problem of the probability estimation is one of the precision. Are we sure that the probability for the event x is 0.3401, but not 0:4399? In most cases no. However, decisions are not sensitive to small variations in probabilities, so we can use probabilities to make decisions. Another problem with probability evaluation is the accuracy. For example, the way a question is phrased can lead to assessments that do not show a person's true beliefs. The virtual trainer can help students to pass the final exam of a new training session from the first attempt and could be the first step to a new e-learning technology model. As future works we plan to develop a complete version of Virtual trainer for both Windows 7 and Linux-like operating systems and optimize the Bayesian networks, for more efficient advices.

Acknowledgements

This work was partially supported by the strategic grant POSDRU/88/1.5/S/53501, Project ID53501 (2009), co-financed by the European Social Fund-Investing in People, within the Sectorial Operational Program Human Resources Development 2007-2013.

References

[1] Andrieu C, Doucet A, Holenstein R. 2010. Particle Markov chain Monte Carlo (with discussion). *Journal of the Royal Statistical Society*, Series B 72: 269–342.

[2] Li H. 2009. Sequential Bayesian analysis of time-changed infinite activity derivatives pricing models. *Working paper*, *ESSEC Business School*, Paris/Singapore.

[3] Li H, Wells MT, Yu CL. 2008. A Bayesian analysis of return dynamics with Lévy jumps. *Review of Financial Studies* 21: 2345–2378.

[4] Lopes HF, Carvalho CM, Johannes M, Polson NG. 2010. Particle learning for sequential Bayesian computation. *In Bayesian Statistics 9,* Bernardo JM, Bayarri MJ, Berger JO, Dawid AP, Heckerman D, Smith AFM, West M (eds). Oxford University Press: Oxford.

[5] Tenenbaum, J.B., Griffiths, T.L. & Kemp, C. (2006). Theory based Bayesian models of inductive learning and reasoning, *Trends in Cognitive Science* 10, 309–318.

[6] Aksoy, S. (2006). Parametric Models: Bayesian Belief Networks, *Lecture Notes*, Department of Computer Engineering Bilkent University.