



HAL
open science

Identifying product failure rate based on a conditional Bayesian network classifier

Zhiqiang Cai, Shudong Sun, Shubin Si, Bernard Yannou

► **To cite this version:**

Zhiqiang Cai, Shudong Sun, Shubin Si, Bernard Yannou. Identifying product failure rate based on a conditional Bayesian network classifier. *Expert Systems with Applications*, 2011, 38 (5), pp.5036-5043. 10.1016/j.eswa.2010.09.146 . hal-00748716

HAL Id: hal-00748716

<https://hal.science/hal-00748716>

Submitted on 25 Mar 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Identifying product failure rate based on a conditional Bayesian network classifier

Zhiqiang Cai^{a,*}, Shudong Sun^a, Shubin Si^a, Bernard Yannou^b

^aMinistry of Education, Key Laboratory of Contemporary Design and Integrated Manufacturing Technology, School of Mechatronics, Northwestern Polytechnical University, Xi'an 710072, China

^bLaboratoire Genie Industriel, Ecole Centrale Paris, Chatenay-Malabry 92290, France

ARTICLE INFO

Keywords:

Maintenance management
Failure rate
Classifier
Bayesian network
Conditional independence

ABSTRACT

To identify the product failure rate grade under diverse configuration and operation conditions, a new conditional Bayesian networks (CBN) model is brought forward. By indicating the conditional independence relationship between attribute variables given the target variable, this model could provide an effective approach to classify the grade of failure rate. Furthermore, on the basis of the CBN model, the procedure of building product failure rate grade classifier is elaborated with modeling and application. At last, a case study is carried out and the results show that, with comparison to other Bayesian networks classifiers and traditional decision tree C4.5, the CBN model not only increases the total classification accuracy, but also reduces the complexity of network structure.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

In recent years, maintenance has been playing a more and more important role in industrial fields due to the high demand for system safety, operational efficiency and life cycle cost control. In China, we cooperated with some aircraft corporations to develop a maintenance management system for years. This maintenance system daily collects bulky failure data during the airplane operation which is in different formats. The challenge faced currently is how to discover the potential failure knowledge from these data for prediction and decision making.

Data mining, which is also referred to as knowledge discovery, means the process of extracting nontrivial, implicit, previously unknown and potentially useful information from databases (Witten & Frank, 2005). Depending on the types of knowledge derived, mining approaches may be classified as association rules mining, clustering, classification, prediction and others. In the area of product failure data mining, it has been used widely for the purpose of failure prediction, failure classification, and failure association. Al-Garni, Jamal, Ahmad, Al-Garni, and Tozan (2006) developed an artificial neural network (ANN) model for predicting the failure rate of De Havilland Dash-8 airplane tires utilizing the two layer feed-forward back-propagation algorithm. Using 6 years of data, the results show that the failure rate predicted by the ANN is closer to the actual data than the failure rate predicted by the Weibull regression model. Chen, Tseng, and Wang (2005) defined the

root-cause machine set identification problem of analyzing correlations between combinations of machines and the defective products and then proposed the Root-cause Machine Identifier (RMI) method using the technique of associating rule mining to solve the problem efficiently and effectively. Han, Kim, and Sohn (2007) applied sequential association rules to extract the failure patterns and forecast failure sequences of Republic of Korea Air Force (ROKAF) aircrafts for various combinations of aircraft types, location, mission and season, which could improve the utilization of aircrafts by properly forecasting the future demand of aircraft spare parts.

Because of the variety of each failure dataset and the diversity of each knowledge discovery mission, researchers have to build proper data mining models and processes according to the characteristic of target dataset and request. In this study, we limit the focus to product failure rate classification. Traditional product failure rate enactment is used to theoretically calculate the system reliability thanks to a static mathematical formula that ignores the actual application of each batch of products. Using the historical product failure data, we could provide a more accurate and effective classification of failure rate according to the configuration and operation. With such results, this model could satisfy the expectations of maintenance scheduling, spare parts supply chain management and product operation optimization.

From recent classification literature, with the characteristics of causality and conditional independence, the Bayesian networks (BN) have been recommended as a comprehensive method of indicating relationships among and influences of variables in system reliability domains (Boudali & Dugan, 2005; Langseth & Portinale, 2007; Mahadevan, Zhang, & Smith, 2001; Muller, Suhner, & Iung, 2008; Weber & Jouffe, 2006). It is a powerful technique for handling system uncertainty and it shows a high performance in

* Corresponding author. Tel.: +86 137 0924 7090.

E-mail addresses: caizhiqiang@nwpu.edu.cn, zhiqiangcai@hotmail.com (Z. Cai), sdsun@nwpu.edu.cn (S. Sun), sisb@nwpu.edu.cn (S. Si), bernard.yannou@ecp.fr (B. Yannou).

prediction and classification tasks. Friedman, Geiger, and Goldszmidt (1997) evaluated approaches for inducing classifiers from data, based on the theory of learning general Bayesian networks (GBN) and put forward a tree augmented Naïve Bayes (TAN) method, which outperforms Naïve Bayes (NB), yet at the same time maintains the computational simplicity and robustness that characterize NB. Cheng and Greiner (1999) learned BN augmented Naïve Bayes (BAN) and GBN using a conditional-independence (CI) based BN learning algorithm and evaluated the algorithms with NB and TAN. Experimental results show that the obtained classifiers are competitive with (or superior to) the other two classifiers. Madden (2002) introduced a new partial Bayesian network (PBN) and describes its constructing algorithm. The algorithm constructs an approximate Markov blanket around a classification node and the results indicate that PBN performs better than other Bayesian network classification structures on some problem domains. Because of the variety of collected data and application domains, researchers also have to focus on the individual case and choose the most effective classifier and modelling process. Baesens et al. (2004) compared and evaluated several Bayesian network classifiers with statistical and artificial intelligence techniques for the purpose of classifying customers in the binary classification problem. The experimental evidence showed that Bayesian network classifiers offer an interesting and viable alternative for customer lifecycle slope estimation problem.

This paper is organized as follows. In Sections 2.1 and 2.2, we discuss the principle of Bayesian networks and common Bayesian network classifiers. To deal with the weakness of present BN classifiers, a new conditional Bayesian network (CBN) classifier and its modeling process are described in Sections 2.3 and 2.4. In Section 3, the case study, the performance criteria and the comparison results are presented. Finally, Section 4 concludes the paper.

2. Modeling

2.1. Bayesian networks

Bayesian networks are directed acyclic graphs (DAGs) used to represent uncertain knowledge in artificial intelligence (Jensen, 1996). A Bayesian network is defined as a couple: $BN = (S, \Theta)$, where $S = (\mathbf{N}, \mathbf{A})$ represents the network structure.

\mathbf{N} describes the set of all the nodes in a BN. Each node represents a discrete variable having a finite number of mutually exclusive states. In our example, a node may be failure cause, failure mode or other factors.

\mathbf{A} is the set of all edges in a BN. Each edge represents the relationship of father and child by linking two nodes. In our example, an edge interprets as a causal relation such as failure cause node affects failure mode node.

Θ represents the set of probability distributions that are associated with each node. When node is a root node (i.e. it does not have a parent), Θ corresponds to the prior probability distribution of the node states. When a node is not a root node (i.e. when it has some parent nodes), Θ corresponds to a conditional probability distribution that quantifies the probabilistic dependency between that node and its parents. It is represented by a conditional probability table (CPT).

Fig. 1 illustrates the nodes, edges and probability distribution through an example. The piston valve has one failure mode which is locked in a closed position. The high temperature and high vibration are two failure causes of valve locked closure, and their joint probability of leading to a locked valve is given by CPT. At last, the closure of valve will result in high gas pressure as a failure effect. From the CPT of high gas pressure, we can see that this node is separated from high temperature and high vibration by the node of valve locked close, which means they are conditionally independent.

Through the complex application of the Bayesian probability theory, Bayesian networks are designed to obtain probabilities of unknown variables from known probabilistic relationships. It is believed that they are well suited for prediction and classification research.

With the network structure and probability distributions mentioned in Fig. 1, it is convenient to compute the posterior probability of target variable. For example, according to the Bayesian theory $P(A/B) = \frac{P(B/A)P(A)}{P(B)}$, where $P(A)$ is prior probability, $P(B/A)$ is conditional probability, $P(A/B)$ is the posterior probability, we could compute $P(C = True|G = True)$ by $\frac{P(G=True|C=True)P(C=True)}{P(G=True)}$.

Then, we calculate the original variable probability distributions as follows:

$$\begin{aligned}
 P(C = True) &= \sum_{TE,V} P(TE, V, C = True) = \sum_{TE,V} P(TE)P(V)P(C \\
 &= True|TE, V) = P(TE = True)P(V = True)P(C = True|TE \\
 &= True, V = True) + P(TE = True)P(V = False)P(C \\
 &= True|TE = True, V = False) + P(TE = False)P(V \\
 &= True)P(C = True|TE = False, V = True) + P(TE \\
 &= False)P(V = False)P(C = True|TE = False, V = False) \\
 &= 0.318
 \end{aligned} \tag{1}$$

$$\begin{aligned}
 P(G = True) &= \sum_{TE,V,C} P(TE, V, C, G = True) \\
 &= \sum_C P(G = True|C) \sum_{TE,V} P(TE)P(V)P(C|TE, V) \\
 &= \sum_C P(C)P(G = True|C) = P(C = True)P(G = True|C = True) \\
 &\quad + P(C = False)P(G = True|C = False) = 0.3172
 \end{aligned} \tag{2}$$

Finally, we got the posterior distribution easily as $P(C = True|G = True) = \frac{0.8 \times 0.318}{0.3172} = 0.802$.

In another case, suppose we have detected the evidence of high temperature and low vibration, the probability of bringing a high gas pressure is shown as:

$$\begin{aligned}
 P(G = true|TE = true, V = false) &= P(G = true|C = true) \\
 &\quad \cdot P(C = true|TE = true, V = false) + P(G = true|C = false) \\
 &\quad \cdot P(C = false|TE = true, V = false) \\
 &= 0.8 * 0.4 + 0.1 * 0.6 = 0.38
 \end{aligned} \tag{3}$$

But with the hypothesis of conditional independence, we could ignore the affections of (TE, V) on node G when C is detected. So, if we already know $(V = true, C = false)$, the result could be calculated directly as $P(G = True|V = True, C = False) = P(G = True|C = False) = 0.1$.

Because of the characteristics of causality and conditional independence, the Bayesian network provides a comprehensive method of representing relationships and influences among variables. It is also a powerful technique for handling uncertainty and shows a high performance in prediction domain. In addition, by presenting with graphical diagrams of nodes and edges, Bayesian network models can be more easily understood than many other techniques (Lee & Abbott, 2003).

2.2. Bayesian network classifiers

The application of Bayesian network classifiers is divided into two stages. First, the structure and parameters of Bayesian network are derived based on learning algorithm and some constraints. Secondly, the inference process is applied to compute the conditional probability of the target variable and classify it into certain classes based on the probability threshold. The time cost of classifier

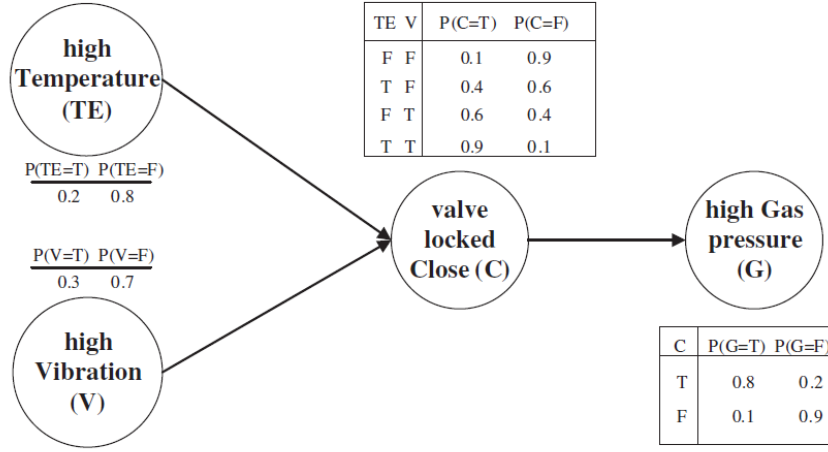


Fig. 1. A simple example of Bayesian networks.

learning and inference mainly depends on the numbers of variables and of the potential relationships among them. So it is usual to add some relationship constraints to simplify the Bayesian network learning process in actual applications.

The NB (Duda & Hart, 1973) supposes that all attribute variables X are conditionally independent given the target variable while it is not true in practice. The TAN (Friedman et al., 1997) makes a great improvement by relax the NB assumption with attribute variables to form a tree. But it is unnecessary to build such a complex structure with too many edges which could lead to the over fitting problems. The GBN learns the structure for global optimization not target node, so some weak but useful connections between target node and attribute variables are dismissed for the lowest minimum description length (MDL) (Friedman & Goldszmidt, 1996) score function. Because the target node is conditionally independent to all other variables given its MB, it could not apply all the useful information hidden in the dataset.

2.3. Conditional Bayesian network classifier

From the descriptions of traditional Bayesian networks classifiers, we could find that they all have their own weakness when dealing with different classification missions. We propose a new kind of Bayesian network classifier building algorithm named as conditional Bayesian networks (CBN). The CBN could utilize the conditional independence relationships among attribute variables given the target variable effectively. It could not only enhance the classification accuracy, but also lower the network structure complexity at the same time.

With a dataset including variables set, a DAG could be learned from it, see Fig. 2. If there is an arc from node X to node Y , then we name X as the parent node of Y (\mathbf{N}_p^Y) while Y is the child node of X (\mathbf{N}_c^X). The other parent nodes of a node's child node are spouse nodes (\mathbf{N}_w^X), like node J . The set of all the child nodes and parent nodes of one node X is called neighbor nodes ($\mathbf{N}_n^X = (\mathbf{N}_p^X, \mathbf{N}_c^X)$) and all the child nodes, parent nodes and spouse nodes of one node X are called its Markov Blanket nodes ($\mathbf{N}_{MB}^X = (\mathbf{N}_p^X, \mathbf{N}_c^X, \mathbf{N}_w^X)$). When a node has no parent node, it is called root node while a node without child node is named as leaf node. The ancestor nodes of a node are defined as its parent nodes and the parent nodes' ancestor nodes ($\mathbf{N}_A^X = (\mathbf{N}_p^X, \mathbf{N}_p^{\mathbf{N}_p^X}, \dots)$), like nodes A and E . The descendent nodes of a node are defined as its child nodes and the child nodes' descendent nodes ($\mathbf{N}_D^X = (\mathbf{N}_c^X, \mathbf{N}_c^{\mathbf{N}_c^X}, \dots)$), like nodes N , Y and P .

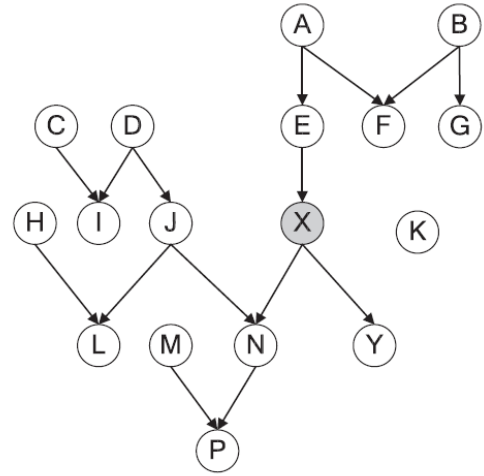


Fig. 2. An example of variables set and its directed acyclic graph.

For the purpose of describing the principle, algorithm and building process of CBN easily, we defined some other kinds of nodes.

Definition 1. Among all the original variables, the nodes that are independent of the learned Bayesian network, which means there is no arc to connect them, are defined as independent nodes (\mathbf{N}_I), like node K .

Definition 2. In a BN, the nodes that are connected with node X with edges are called the BN nodes of node X (\mathbf{N}_{BN}^X). When we delete one node Y and its edges from the BN, the rest nodes connected with X are named as X 's non- Y BN nodes (\mathbf{N}_{BN-Y}^X).

Definition 3. The \mathbf{N}_A^X 's descendent nodes which do not belong to nodes set $(X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_D^X, \mathbf{N}_A^{N_A^X}, \mathbf{N}_D^{N_D^X})$ are defined as sideward nodes ($\mathbf{N}_S^X = \mathbf{N}_D^{N_A^X}, \mathbf{N}_S^X \notin (X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_D^X, \mathbf{N}_A^{N_A^X}, \mathbf{N}_D^{N_D^X})$), like node F in Fig. 3. They have at least one common ancestor node with node X .

Definition 4. The non- X BN nodes of \mathbf{N}_S^X which do not belong to nodes set $(X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_D^X, \mathbf{N}_A^{N_A^X}, \mathbf{N}_D^{N_D^X}, \mathbf{N}_S^{N_S^X})$ are defined as non ancestor nodes ($\mathbf{N}_{NA}^X = \mathbf{N}_{BN-X}^X, \mathbf{N}_{NA}^X \notin (X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_D^X, \mathbf{N}_A^{N_A^X}, \mathbf{N}_D^{N_D^X}, \mathbf{N}_S^{N_S^X})$), like nodes B and G in Fig. 3. Normally, they are the spouse nodes of

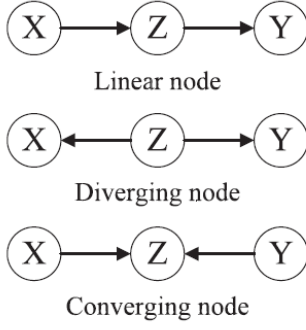


Fig. 3. Three kinds of connection relationships.

ancestor nodes and the spouse nodes' family member. So they do not have a kinship with node X .

Definition 5. All the non- X BN nodes of \mathbf{N}_D^X which do not belong to nodes set $(X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_S^X, \mathbf{N}_A^{N_W^X}, \mathbf{N}_D^{N_W^X}, \mathbf{N}_S^{N_W^X})$ are defined as non descendent nodes $(\mathbf{N}_{ND}^X = \mathbf{N}_{BN-X}^X, \mathbf{N}_{ND}^X \notin (X, \mathbf{N}_W^X, \mathbf{N}_A^X, \mathbf{N}_S^X, \mathbf{N}_A^{N_W^X}, \mathbf{N}_D^{N_W^X}, \mathbf{N}_S^{N_W^X}))$, like nodes C, H and M in Fig. 3.

All conditional independence statements can be derived from a BN structure by using the rules of directed separation (Charniak, 1991). In a BN, a path P between X and Y represents a set of nodes with edges connecting them from X to Y . There are three kinds of connections between a random variable Z and its two immediate neighbors in the path, X and Y . The three possibilities are shown in Fig. 4 and correspond to the possible combinations of arrow directions from Z to X and Y . In the first case, all the edges are in the same direction; in the second case, both are leaving Z ; and in the third, both point at Z . We can name Z as a linear, diverging or converging node in a path P depending on the situations according to Fig. 3.

The description of a block condition in the path is as below. A path P from X to Y is blocked with respect to the evidence nodes Z if Z has the property that either.

- (1) There is at least one linear node in P that belongs to Z .
- (2) There is at least one diverging node in P that belongs to Z .
- (3) There is a converging node in P , it and its descendent nodes do not belong to Z .

If all the paths between X and Y are blocked by Z , then we may say that Z directed separate X and Y , which means that X is conditionally independent of Y given Z , marked as $X \perp Y | Z$.

According to the description of conditional independence, we give the definition of CBN and infer some deductions for the purpose of building a CBN.

Definition 6. For a node X in the nodes set and the learned BN, its conditional Bayesian network is defined as $CBN^X = (\mathbf{N}_{CBN}^X, \mathbf{A}_{CBN}^X)$. The nodes set \mathbf{N}_{CBN}^X include X, \mathbf{N}_{MB}^X and the set of nodes which are conditionally independent with \mathbf{N}_{MB}^X given X ; the edges set \mathbf{A}_{CBN}^X include the edges in \mathbf{N}_{MB}^X , the edges from X to all conditionally independent nodes and the edges among the conditionally independent nodes themselves.

Deduction 1. With a BN learned from nodes set and a node X in it, \mathbf{N}_I are conditionally independent with \mathbf{N}_{MB}^X given X .

Because any node in \mathbf{N}_I does not connect with the BN with arc and could not form a path to $\mathbf{N}_{MB}^X, \mathbf{N}_{MB}^X$ could not affect it given X .

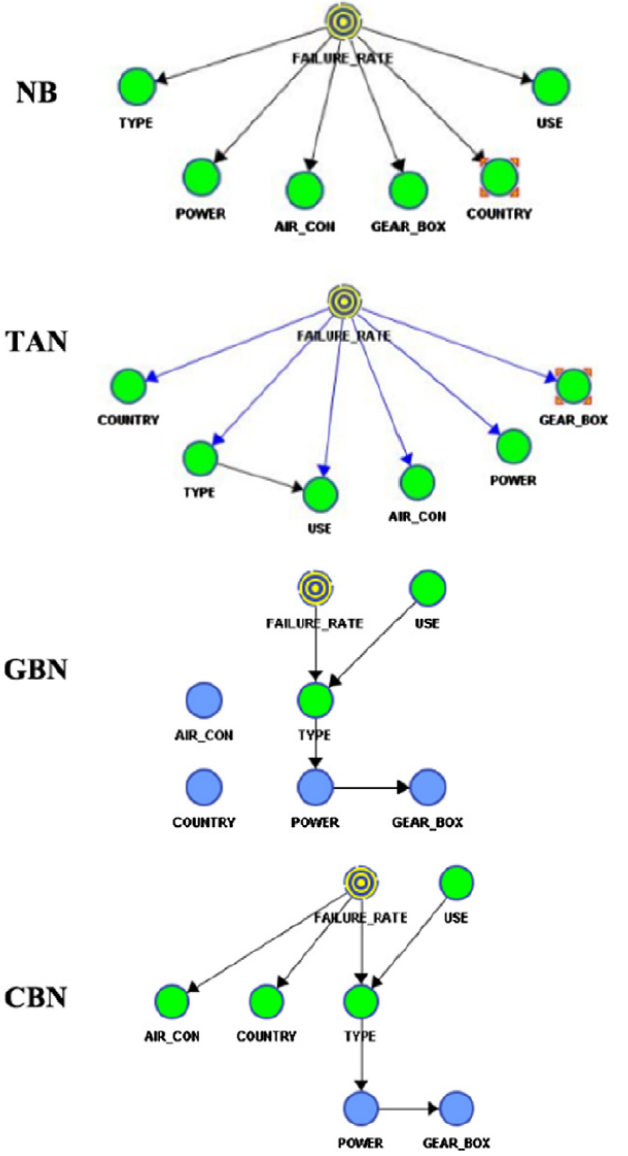


Fig. 4. The structures of learned Bayesian network classifiers.

Deduction 2. With a BN learned from nodes set and the node X in it, node X 's \mathbf{N}_{NA}^X are conditionally independent with \mathbf{N}_{MB}^X given X .

For any node in \mathbf{N}_{NA}^X , because the BN does not have a loop in it, all the paths from the node to $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$ will have at least one converging node in $(\mathbf{N}_S^X, \mathbf{N}_D^X, \mathbf{N}_S^{N_W^X}, \mathbf{N}_D^{N_W^X})$. Since $(\mathbf{N}_S^X, \mathbf{N}_D^X, \mathbf{N}_S^{N_W^X}, \mathbf{N}_D^{N_W^X})$ do not belong to X , we could say that X directed separate \mathbf{N}_{NA}^X and $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$. It is proven that \mathbf{N}_{NA}^X are independent with $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$ given node X .

Deduction 3. With a BN learned from nodes set and the node X in it, node X 's \mathbf{N}_{ND}^X are conditionally independent with \mathbf{N}_{MB}^X given X .

For any node in \mathbf{N}_{ND}^X , because the BN does not have a loop in it, all the paths from the node to $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$ will have at least one converging node in $(\mathbf{N}_S^X, \mathbf{N}_D^X, \mathbf{N}_S^{N_W^X}, \mathbf{N}_D^{N_W^X})$. Since $(\mathbf{N}_S^X, \mathbf{N}_D^X, \mathbf{N}_S^{N_W^X}, \mathbf{N}_D^{N_W^X})$

do not belong to X , we could say that X directed separate \mathbf{N}_{ND}^X and $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$. Then any node in set \mathbf{N}_{ND}^X is independent with $(\mathbf{N}_P^X, \mathbf{N}_C^X, \mathbf{N}_W^X)$ given node X .

Generally speaking, when we learn a directed acyclic graph from dataset, there are still some nodes which are conditionally independent with the target node's MB given target node. These nodes could also provide part mutual information for a more effective classification.

2.4. Product failure rate classification modeling

There are two main kinds of prediction technologies in machine prognostics. The most obvious and widely used prognostic is to predict the remaining useful life (RUL) before a failure occurs given the current machine condition and operation profile. Second it would be more interesting to predict the probability that a machine operates without a fault or a failure up to some future time given the current machine condition and operation profile (Jardine, Lin, & Banjevic, 2006). In our research, we focus on classification of product failure rate when a series of products with diverse configurations operate in different conditions. With this prediction, it could provide solid support for maintenance decision making, spare parts supply chain management and product optimization.

The procedure of built product failure rate classification model based on Bayesian networks is as shown below.

1. For the whole product family of a company, choose the i th series product ($P^i, i = 1, 2, \dots$) as the object of product failure rate classification.
2. According to P^i , search the related failure records from history failure database (including product configuration and application information variables set C , product failure rate variable R). Each record is marked as $\mathbf{D}_j^i = (C_j^i, R_j^i), j = 1, 2, \dots, n$.
3. To avoid training over fitting and perform well for new record, randomly divide failure dataset \mathbf{D}_i into two parts. While 2/3 of the dataset are used as a training set \mathbf{D}_{Train}^i for learning the classifier, the remaining 1/3 are used as a test set \mathbf{D}_{Test}^i for testing the general behavior of the classifier.
4. With the product failure rate R as target node, based on the train set \mathbf{D}_{Train}^i , learn the Bayesian networks classifier's structure \mathbf{G} and compute the conditional probability distributions Θ with Munteanu's EQ algorithm (Munteanu & Bendou, 2001).

Table 1
Variables used in the case study.

Variables	Name	States
Target variable	FAILURE_RATE	HIGH, LOW
Attribute variables	COUNTRY	C1, C2, C3, C4, C5, C6, C7, C8, C9, C10
	TYPE	V1, V2, V3, V4, V5
	USE	S, P
	AIR_CON	YES, NO
	POWER	P1, P2, P3, P4
	GEAR_BOX	HAND, AUTO

Table 2
Data set characteristics.

Data set size	235
Training set size	156
Test set size	79
Number of attributes	6

5. For the learned BN, set the failure rate node as the target node and search the nodes set $(\mathbf{N}_{NA}^R, \mathbf{N}_{ND}^R, \mathbf{N}_I)$ according to their definitions.
 6. Lock the nodes set $(R, \mathbf{N}_{MB}^R, \mathbf{N}_{NA}^R, \mathbf{N}_{ND}^R, \mathbf{N}_I)$ and delete all other nodes and their edges.
 7. According to the deductions before, $(\mathbf{N}_{NA}^R, \mathbf{N}_{ND}^R, \mathbf{N}_I)$ are conditionally independent with \mathbf{N}_{MB}^R given R . So adding edges from R to every node in nodes set $(\mathbf{N}_{NA}^R, \mathbf{N}_{ND}^R, \mathbf{N}_I)$ does not affect the original conditional independence relationship.
 8. Use the Maximum Likelihood Estimation (MLE) algorithm to update the conditional probability distributions of $(\mathbf{N}_{NA}^R, \mathbf{N}_{ND}^R, \mathbf{N}_I)$ resulting from the added edges.
- Finally, the CBN model for product failure rate classification is built and we could test and implement it for the practical applications.
9. According to the performance evaluation criteria, test the performance of learned CBN classifier with test set \mathbf{D}_{Test}^i and compare the results with other classifiers.
 10. Input the new product configuration and application record \mathbf{C}_{new} into CBN classifier and calculate the posterior probability distribution of target variable R to set its failure rate R_{new} .

3. Case study

3.1. Data set

The data we used for case study are all gathered from a French product manufacturer in a certain period (Ben Ahmed, Beranger, & Geslin, 2008). It describes the failure rate of product with different configurations and application conditions. The target variable is failure rate and the attribute variables include COUNTRY, TYPE, USE, AIR_CON, POWER and GEAR_BOX, see Table 1. Because of commercial confidence, we replace the original attribute variables states with corresponding symbols. To avoid over fitting, we split the data set according to process before and the detailed characteristics are shown in Table 2.

3.2. Evaluation criteria for classification

The confusion matrix (Hand, Mannila, & Smyth, 2001) is undoubtedly the most commonly used criteria for measuring performance of classifiers which is defined as $\mathbf{P} = [p_{ij}], i, j = 1, 2, \dots, n$. Number n represents the number of classes and p_{ij} represents the number of records which have been predicted to be in class j but which are actually in class i . When $i = j$, p_{ij} means the number of correct classification records and the total accuracy of model is

$Total = \frac{\sum_{i=1}^n p_{ii}}{\sum_{i=1}^n \sum_{j=1}^n p_{ij}}$. Besides accuracy, we use classification precision

$P_{ij} = \frac{p_{ij}}{\sum_{x=1}^n p_{ix}}, i, j = 1, 2, \dots, n$ and classification reliability

$R_{ij} = \frac{p_{ij}}{\sum_{y=1}^n p_{yj}}, i, j = 1, 2, \dots, n$ to describe distributions of true-positive rate, true-negative rate, false-positive rate and false-negative rate in detail.

Because the target variable has only two states, we also apply Receiver Operating Characteristic (ROC) to estimate the classification results. By moving the classification threshold, it could get coordinate points with true-positive rate as Y-axis and false-positive rate as X-axis. With a ROC curve connecting these points, we could evaluate the classifiers by comparing the curve to the 45° straight line and computing the area under the ROC curve (AUR-OC). It is calculated according to the curve displayed at the top of the graphic and represents the surface under the ROC curve divided by the total surface. If the curve is far from the 45° straight

Table 3
Comparison of classifiers' accuracy.

Classifiers	Criteria	True-positive (%)	False-negative (%)	True-negative (%)	False-positive (%)	Total accuracy (%)
NB	Precision	65.96	34.04	78.12	21.88	70.89
	Reliability	81.58	39.02	60.98	18.42	
TAN	Precision	72.34	27.66	93.75	6.25	81.01
	Reliability	94.44	30.23	69.77	5.56	
GBN	Precision	78.72	21.28	78.12	21.88	78.48
	Reliability	84.09	28.57	71.43	15.91	
CBN	Precision	76.6	23.4	93.75	6.25	83.54
	Reliability	94.74	26.83	73.17	5.26	
C4.5	Precision	80.85	19.15	65.63	34.37	74.68
	Reliability	77.55	30	70	22.45	

line, it means that the AUROC is larger and the performance of classifier is better (Bamber, 1975).

Moreover, the Bayesian network classifiers predict the target class label with its own probability, so the evaluation of the model could also be realized by using the gain curve and the lift curve (Bayesia Limited Company, 2010a). Cumulative gain curve and lift curve are graphical representations of the advantage of using a predictive model by measuring the difference between the results obtained with and without classifier model. The information can be used to determine whether we should use this model or one similar to it in the future.

The gain curve is generated by sorting the test individuals according to the target value probability returned by the classifier in the increasing order. We draw it using the actual positive rates to see how much the classifier model would have helped in this situation. The X-axis represents the rate of individuals that are taken into account. The Y-axis represents the rate of individuals with the target value that have been identified as such. In the gain curve, the yellow¹ line represents the percentage of individuals that have the target value in fact; the blue curve represents the gain curve of a pure random policy; the red curve represents the gain curve corresponding to the classifier. The Gini Index is computed according to the curve and displayed at the top of the graphic. It means the ratio of the surface under the red curve and above the blue curve divided by the surface above the blue curve. This interactive curve is not only an evaluation tool, but also a decision support tool that allows defining the best probability threshold by balancing gains and costs of the classification result.

The lift curve comes from the gain curve and highlights the improvement between the result returned by the current classifier and the result using no model. The X-axis represents the rate of test individuals that are taken into account. The Y-axis represents the lift factor defined as the ratio between the rate of the targeted population obtained with the current policy and the rate obtained with the random policy. The larger the lift factor value the better the accuracy, for a given model.

3.3. Simulation results

According to the modeling process listed in Section 2.4, we learn and test the data set with NB, CBN and GBN algorithms with BayesiaLab (Bayesia Limited Company, 2010b). The structures of four learned Bayesian network classifiers are show in Fig. 4. The confusion matrixes of classification results are listed in Table 3. For the purpose of comparison, we also train the decision tree induction algorithm C4.5 with Matlab toolbox. The Gain curves and Lift curves of all Bayesian network classifiers are shown in Figs.

¹ For interpretation of color in Figs. 4–7, the reader is referred to the web version of this article.

5 and 6, respectively. The ROC curves of each Bayesian network classifiers are shown as in Fig. 7.

From Table 3, we could see that the NB performs poorest with an accuracy of 70.89%. It is because its assumption ignores of the

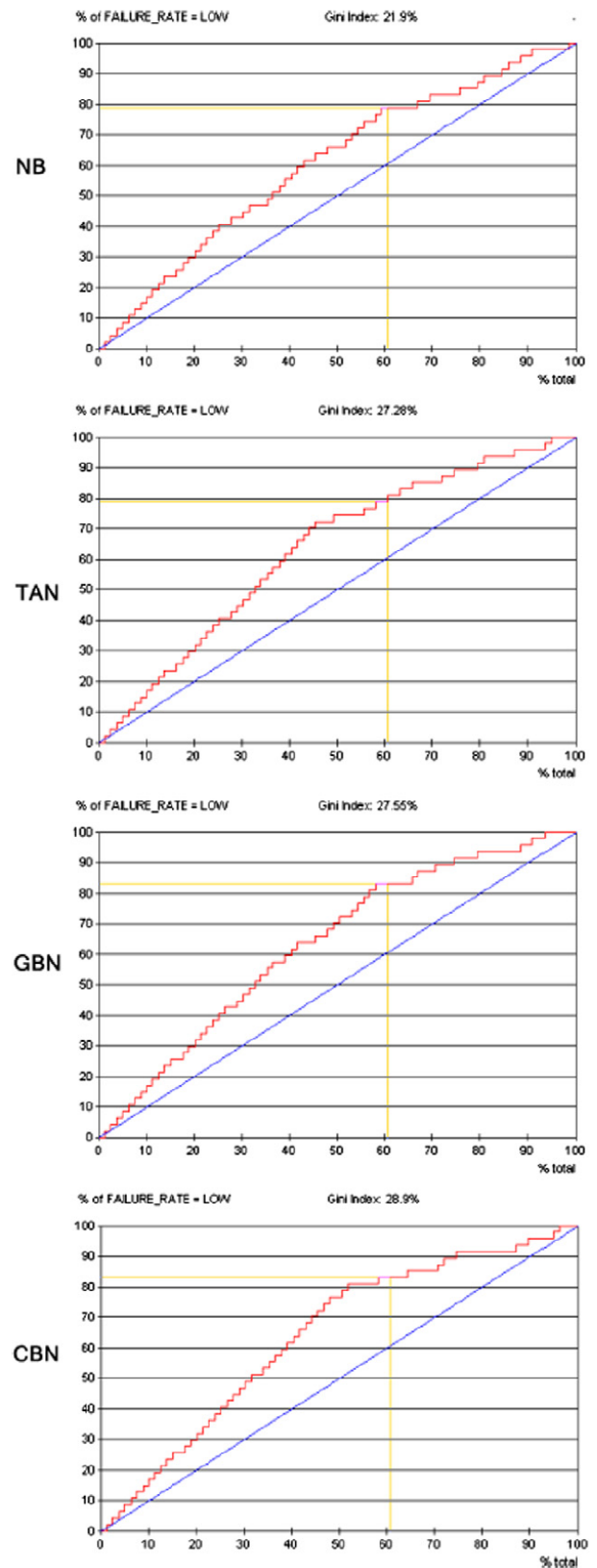


Fig. 5. The gain curves of four learned BN classifiers.

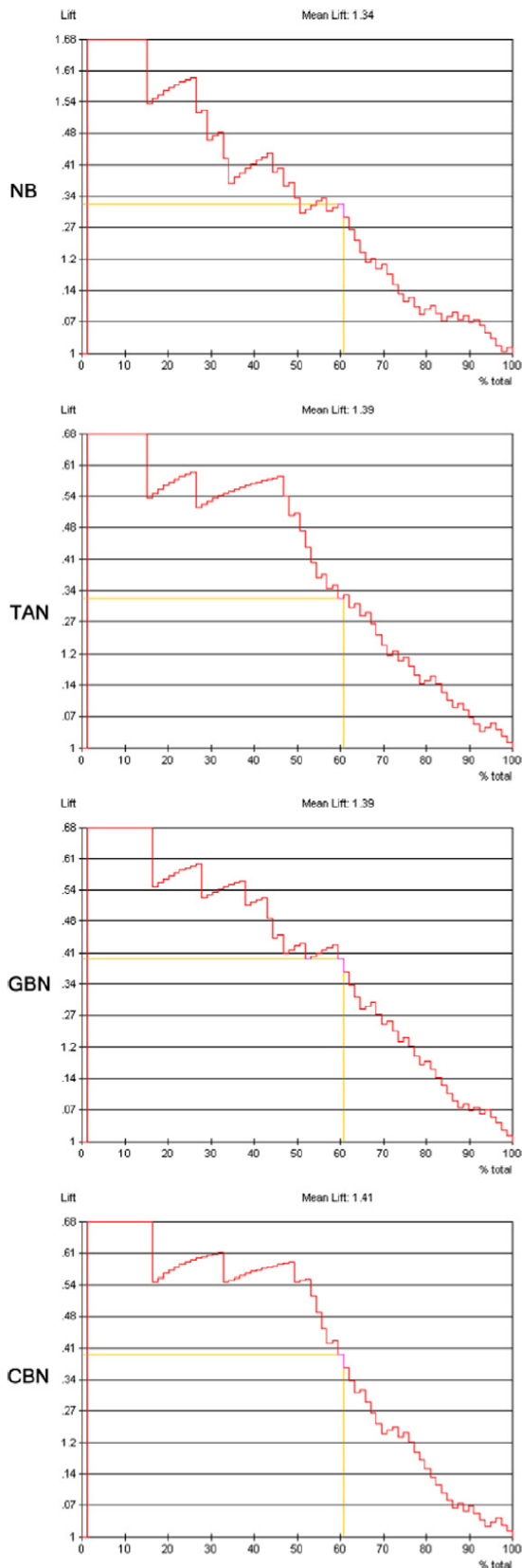


Fig. 6. The lift curves of four learned BN classifiers.

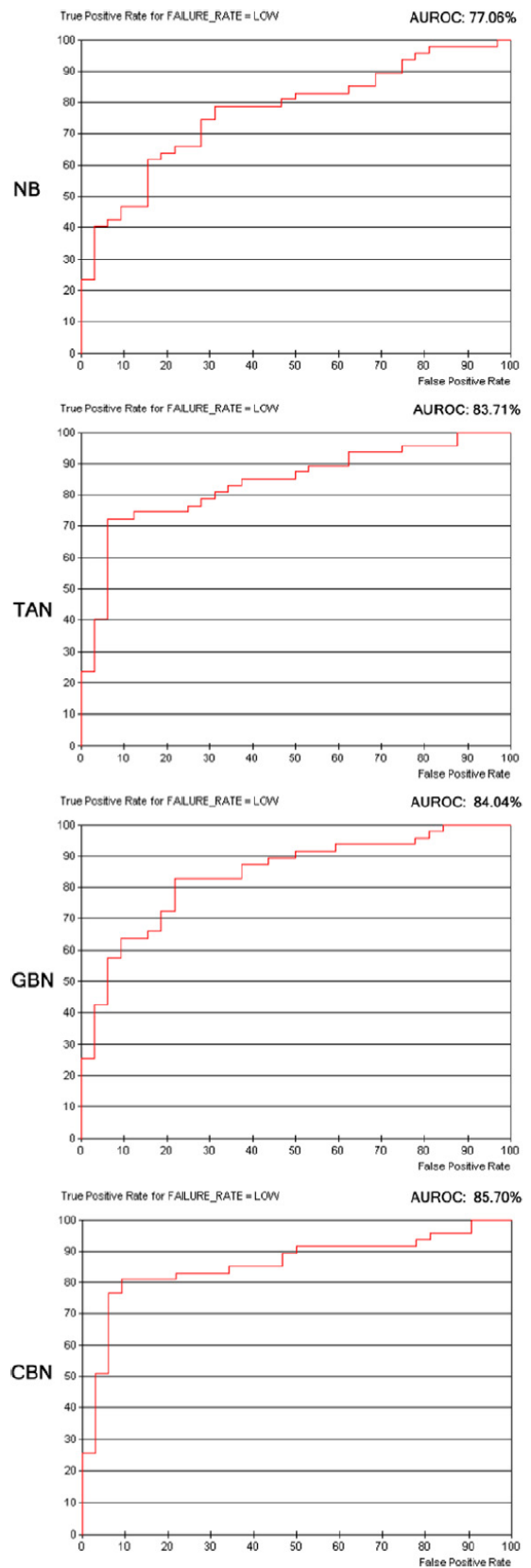


Fig. 7. The ROC curves of four learned BN classifiers.

relationships among attributes variables. The C4.5's result is moderate with 74.68% and still has a gap with the better ones. The GBN works better, but it pays more attention on full structure optimiza-

tion and balance between the precision and complexity. The TAN gets a good result of 81.01% since it relaxes the conditional independence assumption by considering the edges between attribute

Table 4
The structure complexity of BN classifiers.

BN classifiers	Nodes	Edges
NB	7	6
TAN	7	7
GBN	3	2
CBN	5	4

nodes. The CBN gets the highest score of 83.54% since it not only considers the conditional independence between attribute variables but also dismisses the useless direct relationships between attribute variables.

In the aspect of AUROC, the areas of TAN and GBN are almost the same but are much larger than NB's 77.06%. The AUROC of CBN is the best and reaches 85.70%. For the criteria of Gain curve and the Lift curve, the CBN also gets the highest performance with 28.9% in Gini index and 1.41 in mean Lift comparing to other three normal BN classifiers.

Besides, as shown in Fig. 4 and Table 4, the GBN classifier has the simplest structure with only three nodes and two edges because of a lowest MDL score. The TAN model has the most complex structure since it considers the relationship between the attribute nodes on the basis of a NB classifier. The CBN's structure complexity is in between with five nodes and four edges for an effective but economical utilization of all variables' information.

4. Conclusions

The paper proposes a new kind of conditional Bayesian network classifier on the basis of traditional NB, TAN and GBN model. The principle, algorithm and modeling of CBN are described in details to guide the application of identifying the product failure rate. Because it considers the conditional independence between attribute nodes and target node's Markov Blanket given the target node, the CBN could provide a more effective classification result. The case study shows that, with comparison to the BN classifiers and decision tree C4.5 classifier, the CBN got the highest performance in the all criteria of total accuracy, AUROC, Gini index and mean Lift.

Although the GBN classifier had the simplest network structure, the CBN made an acceptable balance between network complexity and promotion of classification performance. It may satisfy the expectations for maintenance decision making, spare parts supply chain management and product configuration optimization.

Acknowledgment

The authors gratefully acknowledge the financial support for this research from the China Civil Aircraft Advanced Research Project, the China Aeronautical Science Foundations (Grant No. 2009ZE53052), the Science and Technology Project of Shaanxi Province (Grant No. 2010K8-11) and the Hover Star project of Northwestern Polytechnical University, China (Grant No. 07XE0152).

References

- Al-Garni, A. Z., Jamal, A., Ahmad, A. M., Al-Garni, A. M., & Tozan, M. (2006). Neural network-based failure rate prediction for De Havilland Dash-8 tires. *Engineering Applications of Artificial Intelligence*, 19, 681–691.
- Baesens, B., Verstraeten, G., Van den Poel, D., Egmont-Petersen, M., Van Kenhove, P., & Vanthienen, J. (2004). Bayesian network classifiers for identifying the slope of the customer lifecycle of long-life customers. *European Journal of Operational Research*, 156, 508–523.
- Bamber, D. (1975). The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *Journal of Mathematical Psychology*, 12, 387–415.
- Bayesia Limited Company. (2010a). BayesiaLab user guide documentation. <www.bayesia.com>.
- Bayesia Limited Company. (2010b). BayesiaLab 4.5, Academic Edition. <www.bayesia.com>.
- Ben Ahmed, W., Beranger, M., & Geslin, C. (2008). A combined use of Bayesian network and logistic regression in reliability data mining: A new methodology. In *16ème Congrès de Maîtrise des Risques et de Sécurité de Fonctionnement*, Avignon, France, October 6–10.
- Boudali, H., & Dugan, J. B. (2005). A discrete-time Bayesian network reliability modeling and analysis framework. *Reliability Engineering and System Safety*, 87, 337–349.
- Charniak, E. (1991). Bayesian networks without tears. *AI Magazine*, 12(4), 50–63.
- Cheng, J., & Greiner, R. (1999). Comparing Bayesian network classifiers. In *Proceedings of the 15th annual conference on uncertainty in artificial intelligence*, Stockholm, Sweden, July 30–August 1, 1999 (pp. 101–108). San Francisco: Morgan Kaufmann.
- Chen, W. C., Tseng, S. S., & Wang, C. Y. (2005). A novel manufacturing defect detection method using association rule mining techniques. *Expert Systems with Applications*, 29, 807–815.
- Duda, R. O., & Hart, P. E. (1973). *Pattern classification and scene analysis*. New York: John Wiley.
- Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29, 131–163.
- Friedman, N., & Goldszmidt, M. (1996). Learning Bayesian networks with local structure. In *Proceedings of the 12th annual conference on uncertainty in artificial intelligence*, Portland, USA, July 31–August 4, 1996 (pp. 252–262). San Francisco: Morgan Kaufmann.
- Hand, D. J., Mannila, H., & Smyth, P. (2001). *Principles of data mining*. Cambridge: MIT Press.
- Han, H. K., Kim, H. S., & Sohn, S. Y. (2007). Sequential association rules for forecasting failure patterns of aircrafts in Korean airforce. *Expert Systems with Applications*. doi:10.1016/j.eswa.2007.10.012.
- Jardine, A. K. S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20, 1483–1510.
- Jensen, F. V. (1996). *An introduction to Bayesian networks*. London: UCL Press.
- Langseth, H., & Portinale, L. (2007). Bayesian networks in reliability. *Reliability Engineering and System Safety*, 92, 92–108.
- Lee, S. M., & Abbott, P. A. (2003). Bayesian networks for knowledge discovery in large datasets: Basics for nurse researchers. *Journal of Biomedical Informatics*, 36, 389–399.
- Madden, M. G. (2002). A new Bayesian network structure for classification tasks. In *Proceedings of the 13th Irish international conference on artificial intelligence and cognitive science*, Limerick, Ireland, 12–13 September, 2002 (pp. 203–208). London: Springer-Verlag.
- Mahadevan, S., Zhang, R., & Smith, N. (2001). Bayesian networks for system reliability reassessment. *Structural Safety*, 23, 231–251.
- Muller, A., Suhner, M. C., & lung, B. (2008). Formalisation of a new prognosis model for supporting proactive maintenance implementation on industrial system. *Reliability Engineering and System Safety*, 93, 234–253.
- Munteanu, P., & Bendou, M. (2001). The EQ framework for learning equivalence classes of Bayesian networks. In *Proceedings of the 2001 IEEE international conference on data mining*, San Jose, USA, November 29–December 2, 2001 (pp. 417–424). Washington DC: IEEE Computer Society.
- Weber, P., & Jouffe, L. (2006). Complex system reliability modelling with dynamic object oriented Bayesian networks (DOOBN). *Reliability Engineering and System Safety*, 91, 149–162.
- Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques* (2nd ed.). San Francisco: Morgan Kaufmann.