Procedure based on mutual information and bayesian networks for fault diagnosis of industrial systems

Sylvain Verron, Teodor Tiplica, Abdessamad Kobi

▶ **To cite this version:**

HAL Id: inria-00517019

https://hal.inria.fr/inria-00517019

Submitted on 13 Sep 2010

Procedure based on mutual information and bayesian networks for the fault diagnosis of industrial systems

Sylvain VERRON, Teodor TIPLICA, Abdessamad KOBI

Abstract— The aim of this paper is to present a new method for process diagnosis using a bayesian network. The mutual information between each variable of the system and the class variable is computed to identify the important variables. To illustrate the performances of this method, we use the Tennessee Eastman Process. For this complex process (51 variables), we take into account three kinds of faults with the minimal recognition error rate objective.

I. INTRODUCTION

In the industrial systems or processes, due to the multiplication of sensors, we can obtain an important amount of data. An active research field is on the utilization of these data in order to control the process. We can view the process control as a four-step procedure. In the first step, the goal is to detect an abnormal situation, a fault in the process: this step is named fault detection. The goal of the second step, the fault identification, is to identify the most relevant variables for the diagnosis of the fault. The third step is the fault diagnosis (which is the purpose of this article), it consists to determine which fault has occurred in the process. Finally, the last step is the process recovery which aim is to return the process in a normal state.

Process control can be achieved by three major ways: data-driven, analytical and knowledge-based [1]. The analytical methods construct mathematic models of the process. The knowledge-based category represents methods based on qualitative models. Finally, the data-driven methods are techniques based on statistical development of the process data. Theoretically, we can say that analytical methods are the most rigorous ones and so the methods giving the best results. But, for large systems (lots of inputs, outputs and states), obtaining detailed models is almost impossible. So, mathematic models of the process are not well fitted to the real process and can conduct to false conclusions on the system. Our preference is on the application of data-driven techniques which are quantitative models based on rigorous statistical developments of the process data.

Many data-driven techniques for fault detection can be found in the literature: univariate statistical process control (Shewhart charts) [2], multivariate statistical process control (T^2 and Q charts) [3], and some PCA (Principal Component Analysis) based techniques [4] like the Multiway PCA or the Moving PCA [5]. In [6], authors make comparisons between these different techniques. Other important approaches are

the PLS (Projection to Latent Structures) based approaches [7] like the Multiway PLS [8]. For the fault identification, one of the most relevant statistical techniques is the MYT decomposition [9] in which authors decompose the T^2 statistic into orthogonal components in order to determine which variable or group of variables contribute to an out-of-control status.

The fault diagnosis procedure can be seen as a classification task. Indeed, as we said, today's processes give many measurements. These measurements can be stored in a database when the process is in control, but also in the case of identified out-of-controls. Assuming that the number of types of fault (classes) and that the belonging class of each observation is in the database (learning sample), the fault diagnosis can be viewed as a supervised classification task whose objective is to class new observations to one of the existing classes. Many classifiers have been developed, we can cite FDA (Fisher Discriminant Analysis) [10], SVM (Support Vector Machine) [11], kNN (k-nearest neighborhood) [12], ANN (Artificial Neural Networks) [10] and bayesian classifiers [13]. But, performances of all these classifiers are reduced in the space described by all the variables of the process. So, before the classification, a feature selection is required in order to obtain better performances.

In this article, we present a new data-driven procedure to diagnosis the faults of an industrial system. This procedure includes a feature selection of the most informative variables of the system. Then, a bayesian classifier can easily discriminate between the types of fault of the process. The article is structured in the following manner. In section II, we introduce bayesian network (BN) classifiers and the notion of mutual information. In section III, we present a new procedure for the fault diagnosis of a system, based on mutual information and a bayesian classifier. Section IV is an application of this procedure on a benchmark problem: the Tennessee Eastman Process. Finally, section V presents conclusions and outlooks of the fault diagnosis with bayesian networks.

II. BAYESIAN NETWORKS AND MUTUAL INFORMATION

A. Bayesian network classifiers

A Bayesian Network (BN) [14], [15], [16] is an acyclic graph. In this graph, each variable is a node that can be continuous or discrete. Edges of the graph represent dependence between linked nodes. A formal definition is given here.

A bayesian network is a triplet $\{\mathbf{G}, \mathbf{E}, \mathbf{D}\}$ where:

{**G**} is a directed acyclic graph, $\mathbf{G} = (V, A)$, where V is the ensemble of nodes of **G**, and A is the ensemble of edges of **G**,

{**E**} is a finite probabilistic space (Ω, Z, p), where Ω is a non-empty space, Z is a collection of subspace of Ω, and p is a probability measure on Z with $p(\Omega) = 1$,

{**D**} is an ensemble of random variables associated to the nodes of **G** and defined on **E** such as:

$$p(V_1, V_2, \ldots, V_n) = \prod_{i=1}^{n} p(V_i | C(V_i)) \qquad (1)$$

where $C(V_i)$ is the ensemble of causes (parents) of V_i in the graph **G**.

Bayesian network classifiers are particular bayesian networks [17], [13]. They always have a discrete node C coding the r different classes of the system. The other variables $(X_1, \ldots, X_p)$ represent the p descriptors (variables) of the system. A Naïve Bayesian Network (NBN) is a particular BN classifier [18], [13], also known as the Bayes classifier. A NBN links the class node to all other variables of the system (descriptors) as indicated on the figure 1.
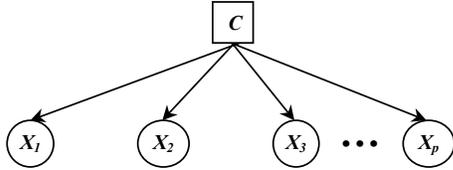

Fig. 1.   Naïve bayesian network

The NBN is called naïve because it makes the naïve (but strong) assumption that all the descriptors (variables of the system) are statistically independent (no correlation between each other). Inza [19] made a comparative study of classification algorithms used in artificial intelligence. Performances of NBN were compared with other methods like hierarchic classification, k-nearest neighborhood, C4.5, decision tree and so on. In the case that the independence assumption of the descriptors is verified and that the probabilities are estimated with enough precision, Yang [20] demonstrated that the NBN is an optimal classifier in term of misclassification rate. This optimality is obtained when continuous variables are discretized in such a way that every distribution can be well approximated by discretization. Of course, the discretization of the variables is a loss of information because it is a reduction of the variables space. But, assuming that continuous variables follow normal probability density function, we can deal with them directly in the network. And, if this assumption is verified, we keep this optimality. In our case (an industrial system), many variables are normally distributed. This assumption is verified in the majority of industrial systems. From the historical database of our system, we will be able to estimate the parameters of the continuous (normal) variables. The Maximum Likelihood Estimation (MLE) and the bayesian approach are two different techniques of estimation of these parameters. In this paper, the parameters of the network are estimated with the MLE method. In the case that all the descriptors are continuous variables (with normal distribution assumption) and that the parameters are estimated with the MLE, we obtain the known classifier: diagonal quadratic discriminant analysis.

But, this classifier is not well fitted for industrial systems. Indeed, in industrial systems, it is very frequent to have high correlations between the variables, and a NBN will not take into account these correlations. Some extensions of the NBN have been developed in order to solve this problem.

A first interesting extension is the TAN (Tree-Augmented bayesian Network) [13]. In a TAN, a maximum weighted spanning tree is constructed with the p descriptors following the algorithm of Chow et Liu [21]. So, each descriptor will have at most one other descriptor as parent. After that, the edges from the class node to each descriptor are added (like a NBN). An example of a TAN is given on the figure 2.
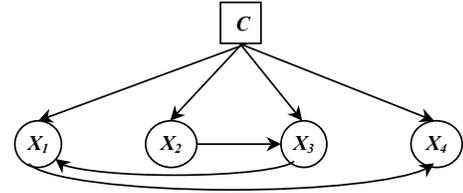

Fig. 2.   Tree-Augmented bayesian Network

Another extension is the k-dependence Bayesian classifier structure (kDB structure) [22] that extends the TAN structure in allowing a maximum of k predictor parents plus the class for each predictor variable (TAN structures are equivalent to kDB structures with k = 1). Finally, the bayesian multinets [23] introduces a different TAN structure for each values of the class variable.

Friedman shows that these classifiers give a lower misclassification rate than the NBN [13]. But, these classifiers do not take into account all the correlations between variables. An other extension to the NBN is the Condensed Semi Naïve Bayesian Network (CSNBN) [24]. The principle of this classifier is to represent some variables in a joint node. So, some normally distributed variables can be modeled with a node representing a multivariate normal distribution. In that way, all the correlations of the system will be taken into account. A CSNBN will be composed of two nodes: the class node and a multivariate node. An example of a CSNBN is given on the figure 3.

Fig. 3.   Condensed Semi Naïve Bayesian Network

As the NBN giving a discriminant rule equivalent to a diagonal quadratic discriminant analysis, the CSNBN is

equivalent to the discriminant rule of the quadratic discriminant analysis. Although this classifier is well fitted for the classification task of faults in industrial systems, it remains a problem. If we have non-informative descriptors, the performances (in term of correct classification rate) are poor. So, if we want to diagnosis a system with many variables, even though only few are really important for classification, other (less important) must not be taken into account. We have to do a selection of the important variables for the classification. We will use mutual information for representing the importance of a variable in the classification (discrimination) task.

B. Mutual Information

In the information theory [25], the Mutual Information (I) of two random variables is a quantity measuring the mutual dependence of these two variables. In our case, the objective is to find important descriptors as well as to quantify this importance. For that, we must compute, for $i = 1, 2, \ldots, p$, $I(C; X_i)$ the mutual information between the class variable C and each descriptor X_i.

In [26], authors demonstrate that the mutual information between a gaussian (normaly distributed) variable and a multinomial (discrete) variable can be computed as indicated by equation 2. For this equation, it is assumed that: C is a multinomial random variable with r possible values and a probability distribution given by $P(C = c) = P(c)$; X is a random variable with a normal density function of parameters μ and σ^2; X conditioned to $C = c$ follows a normal density function with parameters μ_c and $\sigma_c{}^2$.

$$I(X; C) = \frac{1}{2} \left[\log(\sigma^2) - \sum_{c=1}^{r} P(c) \log(\sigma_c^2) \right] \qquad (2)$$

This equation can be reduced on the form of the equation 3.

$$I(X; C) = \log(\sigma) - \sum_{c=1}^{r} P(c) \log(\sigma_c) \qquad (3)$$

So, the mutual information (I) can be computed for all the variables (descriptors) of the industrial system. The most important variables for the classification task will be those having an important I value.

We have seen a tool able to classify (CSNBN) and a tool able to quantify the discriminant information of each descriptor (mutual information I). Now, we will see how to use these two tools for the diagnosis of a fault in an industrial system.

III. THE PROPOSED APPROACH

This approach sorts the variables in decreasing order of $I(X_i; C)$. After that, we will iteratively add variables to the CSNBN and evaluate the misclassification rate. First we will observe that the misclassification rate decreases by adding variables into the model. But once a certain number of variables is overpassed, the misclassification rate starts to increase again. If the misclassification rate increases at
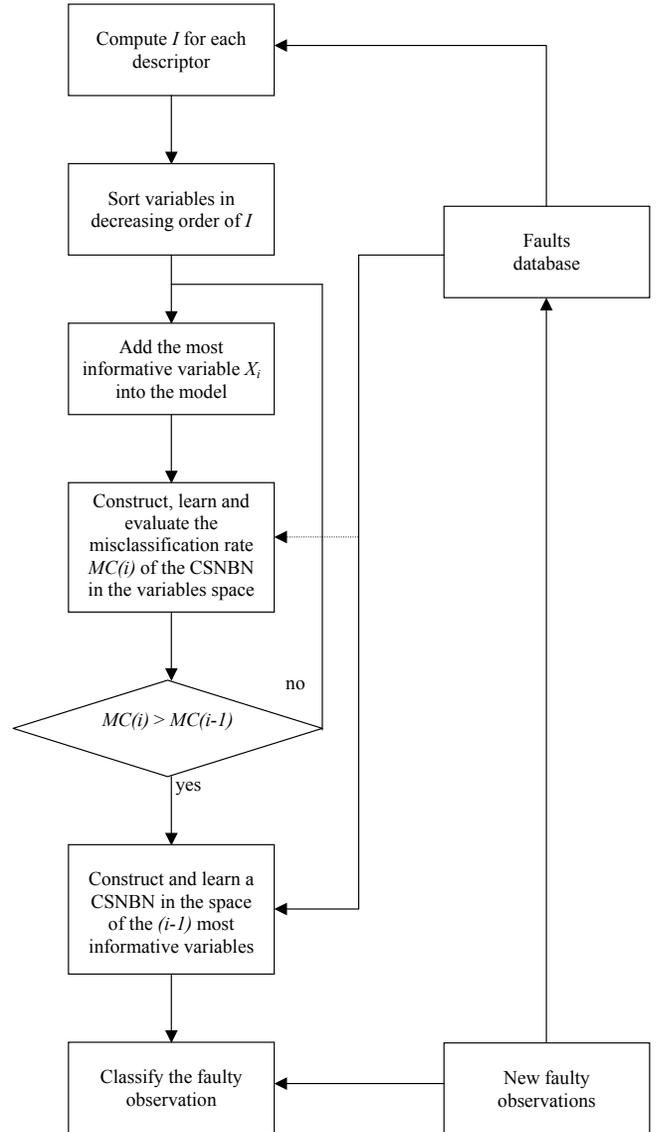

Fig. 4.   Procedure of the proposed approach

step i, the best model will be the one with the $(i - 1)$ first informative variables. So, we obtain the following algorithm:

1) Compute $I(X_i; C)$ for all X_i
2) Sort X_i in decreasing order of $I(X_i; C)$
3) Initialize structure S to a Condensed Semi Naïve Bayesian Network (CSNBN) structure with a unique joint node containing the first informative variable
4) Evaluate this classifier through the estimated classified percentage
5) do {
6) Add the next informative variable to the joint node
7) Evaluate this classifier through the estimated classified percentage
8) }until No option improves the inducted classifier.

Of course, the evaluation of classifiers is done only on the training dataset. For this evaluation procedure, we apply a
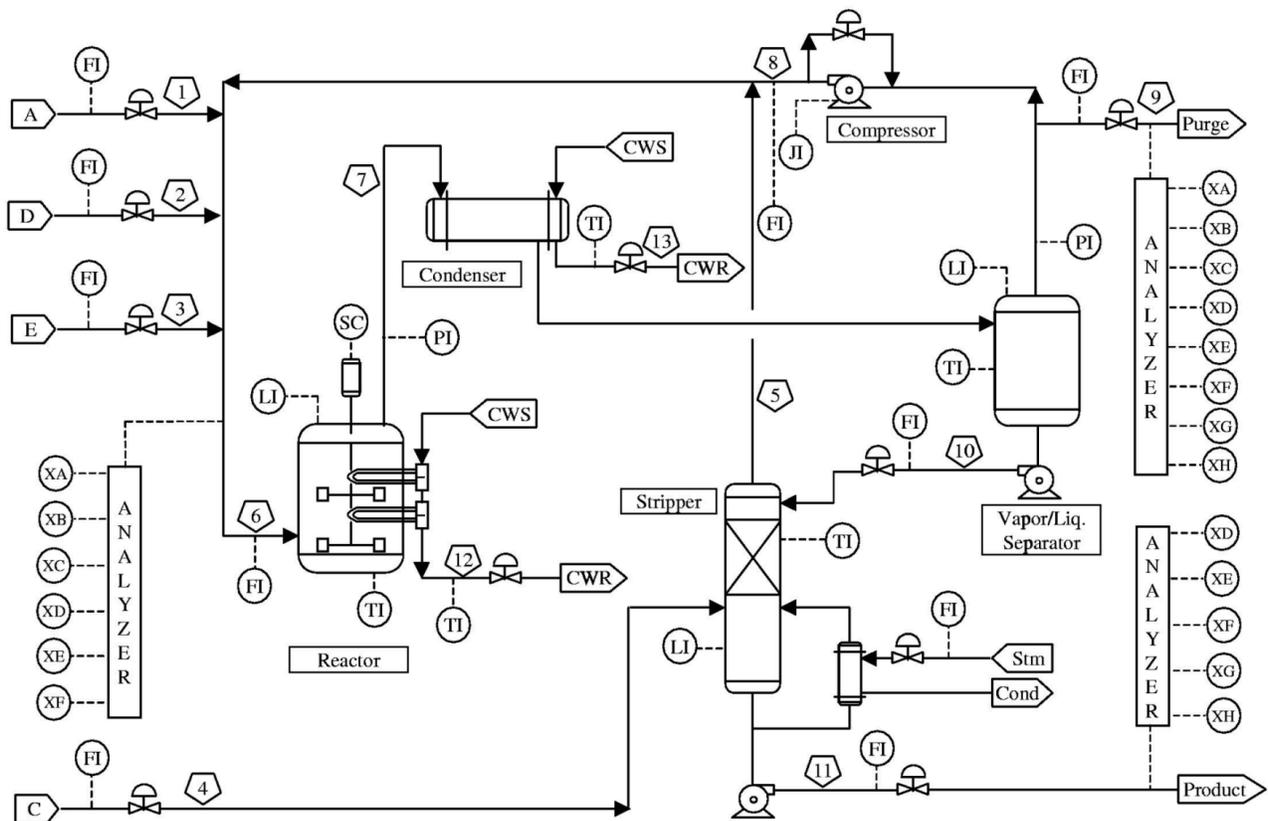
Fig. 5.   Process flowsheet of the TEP

well known technique: the m-fold cross validation [27]. In the m-fold cross validation, the training dataset is divided into m subsets. So, for each m, one of the m subsets is used as the testing set, and the m-1 other subsets are put together to form the training set. Then the average error across all m trials is computed [10]. The entirely procedure of our approach is given on the figure 4. Now, we will see an application of this approach on a benchmark problem: the Tennessee Eastman Process (figure 5).

IV. APPLICATION

A. Presentation of the TEP

The Tennessee Eastman Process (TEP) is a chemical process. It is not a real process but a simulation of a process that was created by the Eastman Chemical Company to provide a realistic industrial process in order to evaluate process control and monitoring methods. Article of Downs and Vogel [28] entirely describes this process. Authors also give the Fortran code of the simulation of the process. Ricker [29] has implemented the simulation on Matlab. The TEP is composed of five major operation units: a reactor, a condenser, a compressor, a stripper and a separator. Four gaseous reactant A, C, D, E and an inert B are fed to the reactor where the liquid products F, G and H are formed. This process has 12 input variables and 41 output variables. The TEP has 20 types of identified faults. So, this process is ideal to test monitoring methods. But, it is also a benchmark problem for

control techniques because it is open-loop unstable. Many articles (authors) present the TEP and test their approaches on it. For example, in fault detection, we can cite [6], [30]. Some fault diagnosis techniques have also been tested on the TEP [1], [31], [32], [33] with the plant-wide control structure recommended in Lyman and Georgakis [34]. In [31], [32], [33], authors focus on only 3 types of fault and give the datasets they used. For this reason, we will take the same data that in these articles and compare our approach to those of the others.

As we said, we have taken into account 3 types of faults: fault 4, 9 and 11 (see table I). These three types of fault are good representations of overlapping data and so, are not easy to classify.

Class	Fault type	Train data	Test data
1	Fault 4: step change in the reactor cooling water inlet temperature	480	800
2	Fault 9: random variation in D feed temperature	480	800
3	Fault 11: random variation in the reactor cooling water inlet temperature	480	800

TABLE I

DESCRIPTION OF FAULT DATASETS

As indicated on the table I, each type of fault have 2 datasets: a training sample and a testing sample, containing respectively 480 and 800 observations.

We have tested our approach on the Tennessee Eastman Process. We precise that all computations have been made on Matlab with the BNT (BayesNet Toolbox) developed by Murphy [35]. We also precise that the approach presented in the third section save a new faulty observation in the database. But, for a correct comparison with the results of other authors, we keep in the database just the 480 observations by fault.

B. Proposed approach on the TEP

Now, we will apply our approach on the TEP application. We first compute I for all variables and sort variables in decreasing order of I. After that, we iteratively add variables to the model and compute misclassification rate of a 10-fold cross validation on the training dataset. Result of these misclassification rates are given in table II.

Variable(s) in the model	Misclassification rate
51	12.5
51, 9	6.46
51, 9, 50	6.04
51, 9, 50, 19	5.90
51, 9, 50, 19, 18	**5.62**
51, 9, 50, 19, 18, 20	5.70

TABLE II

MISCLASSIFICATIONS AT EACH STEP OF THE APPROACH

On this table II, we can see that the best result (on training dataset) is obtained for the selection of the 5 first informative variables (variables 51, 9, 50, 19 and 18). So, we construct a CSNBN in the space of these determined key variables and learn the parameters of the CSNBN with MLE. Finally, we can classify the 2400 observations of the testing dataset. Results of this classification can be seen on the confusion matrix of the table III.

Class	Fault 4	Fault 9	Fault 11	Total
Fault 4	788	0	31	819
Fault 9	0	778	74	852
Fault 11	12	22	695	729
Total	800	800	800	2400

TABLE III

CONFUSION MATRIX OF APPROACH B ON THE TEP DATA

Based on this table, we can compute the misclassification rate. Its value is 5.79%. We can now analyze our results and compare them to those of other authors.

C. Analysis and comparison of results

We will compare the results of our approach to those of other methods on the same data (see table IV). We notify that the results for the methods FDA (Fisher Discriminant Analysis), SVM (Support Vector Machines) and PSVM (Proximal Support Vector Machines) are extracted from [32], and the results for the ISVM (Independent Support Vector

Machines) from [33]. In these articles, the classification was made in the space of the variables 9 and 51. In the table IV, we have added the misclassification rate of the CSNBN in the space of all the variables (53 variables) $CSNBN_{all}$. We have also computed the misclassification rate that would be obtained by the use of a Naïve Bayesian Network, firstly in the space of all the variables (NBN_{all}), secondly in the space described by the variables 9 and 51 ($NBN_{9,51}$) and finally in the same space that the result of our approach ($NBN_{9,18,19,50,51}$).

Method	Misclassification rate
NBN_{all}	20.83%
$CSNBN_{all}$	18.83%
$FDA_{9,51}$	16%
$PSVM_{9,51}$	6.0%
$SVM_{9,51}$	6.5%
$ISVM_{9,51}$	6.0%
$NBN_{9,51}$	7.79%
$CSNBN_{9,51}$	5.87%
$NBN_{9,18,19,50,51}$	13.12%
$CSNBN_{9,18,19,50,51}$	**5.79%**

TABLE IV

RESULTS OF THE APPLICATION

We can see that the classification methods in the selected variables space give better results than in the space of all the variables. Second interesting remark is the fact that on the same space (variables 9 and 51), the best classifier is the CSNBN. The fact that CSNBN outperforms FDA is not surprising because FDA is a linear technique while CSNBN is a quadratic one. More, it is also logic than CSNBN outperforms NBN because NBN do not take into account the correlation between variable 9 and variable 51 (correlation of 0.42). Although, CSNBN requires quite simple computation, its results are quite similar to SVM based techniques (SVM, PSVM, ISVM), which are techniques requiring more computational potential. Finally, we can see that the best result is obtained with our proposed approach. Indeed, this method gives the best result of misclassification rate for this example of the TEP. But, it will be interesting to have misclassification rates of others techniques in the same space of variables (9, 18, 19, 50, 51).

Our approach gives good results on this example because fault classes are normally distributed (ellipsoidal shapes). Results of the CSNBN would not be so efficient if the data were not normal. But, an interesting way for taking into account not-normally distributed variables is the gaussian mixture model that, on the based of few gaussian distributions, can represent any type of distribution. It will be a good answer to the problem of normally distributed data assumption. More, it

is easy to represent a gaussian mixture model with a bayesian network.

Another way of interest is the diagnosis of a not identified fault type (no historical data on this fault type). Indeed, the major drawback of our procedure is the selection of the variables able to discriminate identified faults. But, if a new type of fault appeared, not discriminable on the selected variables, the procedure randomly attributes to this fault an identified type.

V. CONCLUSION

The main interest of this article is the presentation of a new procedure for the fault diagnosis of industrial processes. This procedure uses a faults database to compute the mutual information between each descriptor and the fault type variable. In our approach, the informative variables are iteratively added to the model and the misclassification rates are estimated with a 10-fold cross validation. The iteration stops as soon as the estimated misclassification rate increases. The performances of this approach were tested on a concrete example: the Tennessee Eastman Process. The results of this method are good and outperform some previous results of other published methods.

Extensions of this procedure will be the application of similar techniques, based on bayesian networks, in order to improve the fault diagnosis (cases of a non-identified fault, case of non-normally distributed data), but also in order to study in which way the fault detection step can be made with a bayesian network. The final goal is to obtain on the same procedure the fault detection and the fault diagnosis of a system.

REFERENCES

[1] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault detection and diagnosis in industrial systems*. New York: Springer-Verlag, 2001.

[2] W. A. Shewhart, *Economic control of quality of manufactured product*. New York : D. Van Nostrand Co., 1931.

[3] H. Hotelling, "Multivariate quality control," *Techniques of Statistical Analysis*, vol. , pp. 111–184, 1947.

[4] E. J. Jackson, "Multivariate quality control," *Communication Statistics - Theory and Methods*, vol. 14, pp. 2657 – 2688, 1985.

[5] B. R. Bakshi, "Multiscale PCA with application to multivariate statistical process monitoring," *AIChE Journal*, vol. 44, no. 7, pp. 1596–1610, 1998.

[6] M. Kano, K. Nagao, S. Hasebe, I. Hashimoto, H. Ohno, R. Strauss, and B. Bakshi, "Comparison of multivariate statistical process monitoring methods with applications to the eastman challenge problem," *Computers and Chemical Engineering*, vol. 26, no. 2, pp. 161–174, 2002.

[7] J. MacGregor and T. Kourti, "Statistical process control of multivariate processes," *Control Engineering Practice*, vol. 3, no. 3, pp. 403–414, 1995.

[8] B. Wise and N. Gallagher, "The process chemometrics approach to process monitoring and fault detection," *Journal of Process Control*, vol. 6, no. 6, pp. 329–348, 1996.

[9] R. L. Mason, N. D. Tracy, and J. C. Young, "Decomposition of T^2 for multivariate control chart interpretation," *Journal of Quality Technology*, vol. 27, no. 2, pp. 99–108, 1995.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification 2nd edition*. Wiley, 2001.

[11] V. N. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1995.

[12] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, pp. 21–27, 1967.

[13] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[14] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, 1988.

[15] E. Charniak, "Bayesian networks without tears," *AI Magazine*, vol. 12, no. 4, pp. 50–63, 1991.

[16] F. V. Jensen, *An introduction to Bayesian Networks*. Taylor and Francis, London, United Kingdom, 1996.

[17] P. Langley, W. Iba, and K. Thompson, "An analysis of bayesian classifiers," in *National Conference on Artificial Intelligence*, 1992.

[18] P. Domingos and M. J. Pazzani, "Beyond independence: Conditions for the optimality of the simple bayesian classifier," in *International Conference on Machine Learning*, 1996.

[19] I. Inza, P. Larranaga, B. Sierra, R. Etxeberria, J. Lozano, and J. Pena, "Representing the behaviour of supervised classification learning algorithms by bayesian networks," *Pattern Recognition Letters*, vol. 20, no. 11-13, pp. 1201–1209, 1999.

[20] Y. Yang and G. I. Webb, "Discretization for naive-bayes learning: Managing discretization bias and variance," Tech. Rep. 131, School of Computer Science and Software Engineering, Monash University,, 2003.

[21] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *Information Theory, IEEE Transactions on*, vol. 14, no. 3, pp. 462–467, 1968.

[22] M. Sahami, "Learning limited dependence bayesian classifiers," in *Second International Conference on Knowledge Discovery in Databases*, 1996.

[23] D. Geiger and D. Heckerman, "Knowledge representation and inference in similarity networks and bayesian multinets," *Artificial Intelligence*, vol. 82, no. 1-2, pp. 45–74, 1996.

[24] I. Kononenko, "Semi-naive bayesian classifier," in *EWSL-91: Proceedings of the European working session on learning on Machine learning*, pp. 206–219, 1991.

[25] C. E. Shannon, "A mathematical theory of communication," *Bell Sys. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

[26] A. Perez, P. Larranaga, and I. Inza, "Supervised classification with conditional gaussian networks: Increasing the structure complexity from naive bayes," *International Journal of Approximate Reasoning*, vol. 43, pp. 1–25, 2006.

[27] T. M. Cover, *Learning in pattern recognition*. NY: Methodologies of Pattern Recognition, s. watanabe (ed.) ed., 1969.

[28] J. Downs and E. Vogel, "Plant-wide industrial process control problem," *Computers and Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[29] N. Ricker, "Decentralized control of the tennessee eastman challenge process," *Journal of Process Control*, vol. 6, no. 4, pp. 205–221, 1996.

[30] U. Kruger, Y. Zhou, and G. Irwin, "Improved principal component monitoring of large-scale processes," *Journal of Process Control*, vol. 14, no. 8, pp. 879–888, 2004.

[31] L. Chiang, E. Russell, and R. Braatz, "Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 50, no. 2, pp. 243–252, 2000.

[32] L. Chiang, M. Kotanchek, and A. Kordon, "Fault diagnosis based on fisher discriminant analysis and support vector machines," *Computers and Chemical Engineering*, vol. 28, no. 8, pp. 1389–1401, 2004.

[33] A. Kulkarni, V. Jayaraman, and B. Kulkarni, "Knowledge incorporated support vector machines to detect faults in tennessee eastman process," *Computers and Chemical Engineering*, vol. 29, no. 10, pp. 2128–2133, 2005.

[34] P. Lyman and C. Georgakis, "Plant-wide control of the tennessee eastman problem," *Computers and Chemical Engineering*, vol. 19, no. 3, pp. 321–331, 1995.

[35] K. P. Murphy, "The bayes net toolbox for matlab," in *In Computing Science and Statistics : Proceedings of Interface*, 2001.