Process Fault Detection Employing Feature Selection and Linear Discriminant Analysis

Anderson da Silva Soares, engsoares@gmail.com
Programa de Pós-Graduação em Engenharia Eletrônica e Computação, ITA - São José dos Campos, SP
Roberto Kawakami Harrop Galvão, kawakami@ita.br
Divisão de Engenharia Eletrônica, ITA - São José dos Campos, SP

Abstract. *Classification methods such as linear discriminant analysis (LDA) have been widely applied to fault detection in industrial processes. In this case, the problem consists of classifying the operation as normal or faulty on the basis of monitored variables. If the number of such variables is large, selection techniques may be used to choose an informative subset of features in order to obtain a classifier with better generalization properties. In fact, the use of too many features may cause overfitting problems. In LDA, the presence of multicollinearity among the features may also lead to poor-conditioning issues, which are a known cause of generalization problems for the resulting classifier. This paper presents a fault detection approach that employs the Successive Projections Algorithm (SPA) as a feature selection technique for use with LDA. SPA is a recently proposed technique, which was specifically designed to minimize multicollinearity among the classifier inputs. The joint use of SPA and LDA has provided good results in several pattern recognition problems. However, their application in process fault detection has not been previously reported. The performance of the proposed approach was assessed in a simulated case study involving the Tennessee Eastman process, which is a reactor-separator-recycle system widely used as a benchmark in fault detection studies. The simulations involved 22 measured variables under normal operating conditions, as well as eight different types of faults. The LDA classifier inputs comprised present and time-lagged values of the measurements. SPA was thus applied to select not only the measurements to be considered for fault detection, but also the time lags to be employed. The results were evaluated in terms of overall classification performance, as well as sensitivity and false alarm rate. These metrics were obtained for a test set, which was not employed for feature selection or classifier training. As a result, a classification accuracy of 100% was obtained for six of the eight fault types. The accuracy for the remaining faults ranged from 75% to 85%. For comparison, two other classification techniques were also employed, namely κ-Nearest Neighbours (KNN) and LDA with feature selection by a Genetic Algorithm (GA-LDA). As a result, SPA-LDA was found to be superior to KNN and comparable to GA-LDA. In addition, SPA-LDA provided more parsimonious classification models as compared to GA-LDA.*

Keywords: *Fault Detection, Feature Selection, Successive Projections Algorithm, Linear Discriminant Analysis*

1. INTRODUCTION

In dynamic systems, a fault can be understood as a deviation from normal behaviour, which may arise for several reasons, such as design error, undue use or natural degradation (Isermann, 2006 and Venkatasubramanian et al., 2003). Since the occurrence of faults cannot be completely avoided, efficient methods for fault detection are very important to prevent economics loss or risk to human operators. In analytical redundancy schemes, a model is employed to perform dynamic consistency checks between the inputs and outputs of the system. However, such an approach may not be feasible if the *a priori* knowledge about the system is insufficient to build a suitable model. In contrast, data-driven methods (or process history based methods) only require the availability of historical process data. In this case, classification techniques such as κ-nearest neighbours (KNN) (Friedman, Baskett and Shustek, 1975 and He and Wang, 2007), and Linear Discriminant Analysis (LDA) (Chiang et al., 2001) can be used. For this purpose, fault detection can be formulated as a classification problem involving two classes: Class_1: normal operation and Class_2: operation with fault.

In classification problems, the results can be greatly affected by the choice of features employed as inputs to the classifier. A variety of different techniques have been applied to the problem of automated feature selection, including neural networks (Gascaa and Sánchezb, 2006), evolutionary algorithms (Chiang and Pell, 2004) and the Successive Projections Algorithm (SPA) (Pontes et al., 2005). SPA is a recently proposed method, which was specifically designed for use with

LDA classifiers. Application examples include the classification of vegetable oils from voltammetric data (Gambarra Neto et. al., 2009), diesel samples from near-infrared measurements (Pontes et al., 2005) and soil types using laser-induced breakdown spectroscopy (Pontes et al., 2009). The present paper proposes the use of SPA-LDA for fault detection. In this case, the LDA classifier inputs may comprise present and time-lagged values of measured process variables. Therefore, SPA will be applied to select not only the measurements to be considered for fault detection, but also the time lags to be employed.

The proposed SPA-LDA fault detection technique is illustrated by using simulated data from the Tennessee Eastman process (TEP), which is a reactor-separator-recycle system widely used as a benchmark in fault detection studies (Downs and Vogel, 1993). For comparison, two other classification techniques are also employed, namely: (1) KNN applied to the full set of available features and (2) LDA with feature selection by a Genetic Algorithm (GA-LDA) (Chiang and Pell, 2004). The results are evaluated in terms of overall classification performance, as well as sensitivity and false alarm rate.

2. Linear Discriminant Analysis

The purpose of Discriminant Analysis is to classify objects into one of several groups based on a set of features that describe the objects. Let $\mathbf{x} = [x_1, x_2, \ldots, x_d]^T$ be an object that must be assigned to one out of c possible classes. The Linear Discriminant Analysis method employs the Mahalanobis distance (Maesschalck, Rimbaud and Massart, 2000), which can be defined as follows. The squared Mahalanobis distance $r^2(\mathbf{x}, \mu_j)$ between $\mathbf{x}$ and the center of the j^{th} class $(j = 1, 2, \ldots, c)$ is defined as

$$r^2(\mathbf{x}, \mu_j) = (\mathbf{x} - \mu_j)^T \mathbf{\Sigma}_j^{-1} (\mathbf{x} - \mu_j); \tag{1}$$

where μ_j and $\mathbf{\Sigma}_j$ correspond to the mean and covariance matrix of the j^{th} class, respectively. If the true mean and covariance values for the population are unknown (which is usually the case), maximum likelihood estimates $\bar{\mathbf{x}}_j$ and $\mathbf{S}_j$ may be employed in place of μ_j and $\mathbf{\Sigma}_j$, respectively. These estimates can be obtained from a finite set of training objects of known classification. It is worth noting that LDA estimates a single pooled covariance matrix $\mathbf{S}$, instead of using a separate estimate for each class. This regularization procedure simplifies the classification model and results in linear decision surfaces (hyperplanes) in $\mathbb{R}^d$ (Duda, 2001). With this modification, the squared Mahalanobis distance between $\mathbf{x}$ and the center of the j^{th} class is calculated according to Eq. (2). Object $\mathbf{x}$ is then assigned to the class j for which $r^2(\mathbf{x}, \bar{\mathbf{x}}_j)$ has the smallest value.

$$r^2(\mathbf{x}, \bar{\mathbf{x}}_j) = (\mathbf{x} - \bar{\mathbf{x}}_j)^T \mathbf{S}^{-1} (\mathbf{x} - \bar{\mathbf{x}}_j); \tag{2}$$

2.1 Classification features for fault detection

Suppose that q process variables $w_1, w_2, \ldots, w_q$ are monitored for fault detection purposes. Let $w_i(t)$ be the measured value of w_i at time t. A possible approach consists of using $w_1(t), w_2(t), \ldots, w_q(t)$ as features to classify the operation as "normal" or "faulty" at time t. In this case, the object to be classified would be a vector $\mathbf{x}(t) = [w_1(t), w_2(t), \ldots, w_q(t)]^T$. Given an operation log of the system at times $t_1, t_2, \ldots, t_n$, the training data for the classifier could be disposed in matrix form as

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_n) \end{bmatrix} = \begin{bmatrix} w_1(t_1) & w_2(t_1) & \ldots & w_q(t_1) \\ w_1(t_2) & w_2(t_2) & \ldots & w_q(t_2) \\ \vdots & \vdots & \ldots & \vdots \\ w_1(t_n) & w_2(t_n) & \ldots & w_q(t_n) \end{bmatrix}$$

Now, consider that the measurements are acquired with a constant sampling period T. In this case, an alternative consists of using not only the present values, but also the past values of $w_1, w_2, \ldots, w_q$ as classifier inputs. In this case, the object at time t would be

$$\mathbf{x}(t) = [w_1(t) \quad w_1(t-T) \quad w_1(t-(p-1)T) \quad \ldots \quad w_2(t) \quad w_2(t-T) \quad w_2(t-(p-1)T) \ldots$$
$$\ldots \quad w_q(t) \quad w_q(t-T) \quad w_q(t-(p-1)T)]^T$$

where constant p defines the time window to be used. The classification features at time t would then be

$$\begin{aligned} &x_1(t) = w_1(t) \\ &\quad\vdots \\ &x_p(t) = w_1(t-(p-1)T) \\ &x_{p+1}(t) = w_2(t) \\ &\quad\vdots \\ &x_{2p}(t) = w_2(t-(p-1)T) \\ &\quad\vdots \\ &x_{qp}(t) = w_q(t-(p-1)T) \end{aligned}$$

3. The Successive Projections Algorithm for Classification Problems

An intrinsic limitation of LDA is the assumption that the classification features are not linearly dependent over the training data. Otherwise, the calculation of the matrix inverse in Eq. (2) becomes ill-conditioned (Krzanowski et al., 1995). Therefore, the use of LDA usually requires an appropriate feature selection procedure (Pontes et al., 2005; Krzanowski et al., 1995).

In a classification framework, the modelling information consists of a matrix of objects ($\mathbf{X}$) and a class index for each object. Matrix $\mathbf{X}$ has dimensions ($N \times K$), that is, N objects with K features each. The Successive Projections Algorithm for feature selection in classification problems was proposed in (Pontes et al., 2005) and comprises two main phases using two data sets: training and validation. Phase 1 consists of projection operations carried out on the training matrix $\mathbf{X}$, which generate K chains of M features each. The construction of each chain starts from one of the features $x_k, k = 1, \ldots, K$. Each subsequent feature included in the chain is selected in order to display the least collinearity with the previous ones. The maximum length M of the chain is given by

$$M = \min(N - c, K) \tag{3}$$

where c is the number of classes considered in the problem (Duda, Hart and Stork, 2001). However, a smaller value of M can be imposed by the user in order to save computational time.

In Phase 2, candidate subsets of features are extracted from the chains and evaluated according to the average risk G of misclassification by LDA when the subset of features under evaluation is employed. Such a cost is calculated in the validation set according to Equation (4).

$$G_{km} = \frac{1}{N_v} \sum_{n=1}^{N_v} g_{n,km}; \quad k = 1, \ldots, K; \; m = 1, \ldots, M \tag{4}$$

where N_v is the number of validation objects and G_{km} is calculated by using the first m features of the k^{th} chain. Moreover, $g_{n,km}$ is defined as

$$g_{n,km} = \frac{r^2_{km}(\mathbf{x}_n, \bar{\mathbf{x}}_{In})}{\min_{Ij \neq In} r^2_{km}(\mathbf{x}_n, \bar{\mathbf{x}}_{Ij})}; \quad n = 1, \ldots, N_v \tag{5}$$

where $r^2_{km}(\mathbf{x}_n, \bar{\mathbf{x}}_{In})$ is the squared Mahalanobis distance (Eq. 2) between object $\mathbf{x}_n$ (of class index In) and the sample mean $\bar{\mathbf{x}}_{In}$ of its true class. The denominator in Eq. (5) corresponds to the squared Mahalanobis distance between object $\mathbf{x}_n$ and the center of the closest wrong class. Ideally, $g_{n,km}$ should be as small as possible, that is, object $\mathbf{x}_n$ should be close to the center of its true class and distant from the centers of all other classes (in the statistical sense defined by the Mahalanobis distance).

4. Case Study

The proposed method will be applied to the same data set of (Pei et al., 2008), which was obtained from the Tennessee Eastman Process simulator (Downs and Vogel, 1993). The plant includes five major units: a reactor, a product condenser, a vapor-liquid separator, a recycle compressor and a product stripper. It converts four source materials (PA, PB, PC, PD) into two products (PG and PH).

Twenty-two process variables, listed in Table 1, are used for monitoring the operation. The sampling period T was set to three minutes. The faults under consideration consist of eight disturbances available in the simulator, which are listed in Table 2. In this table, the first seven disturbances are step changes and the last one is a random perturbation. A simulation lasting 200 sampling periods was carried out for each type of fault. In each simulation, the fault was inserted after 100 samples.

Constant p (width of the time window) was set to 40. Therefore, each object was described by 880 features (22 variables times 40 lags). The objects $\mathbf{x}(t)$ corresponding to normal operation were obtained for $t = 40T$ up to $t = 100T$. It is worth noting that $\mathbf{x}(t)$ could not be defined for $t < 40T$ as some of the time-lagged values would not be available. The objects objects $\mathbf{x}(t)$ corresponding to operation with fault were obtained for $t = 101T$ up to $t = 200T$. For each fault type, twenty objects were randomly selected for validation. In addition, another twenty objects were separated for test purposes. These test objects were only used in the final evaluation and comparative analysis of the classifiers.

Table 1. Process variables available for fault detection.

Process variable	Description
w_1	PA feed flowrate.
w_2	PD feed flowrate.
w_3	PE feed flowrate.
w_4	PA and PC feed flowrate.
w_5	Recycle flowrate.
w_6	Reactor feed rate.
w_7	Reactor pressure.
w_8	Reactor level.
w_9	Reactor temperature.
w_{10}	Purge rate.
w_{11}	Product separator temperature.
w_{12}	Product separator level.
w_{13}	Product separator pressure.
w_{14}	Product separator underflow.
w_{15}	Stripper level.
w_{16}	Stripper pressure.
w_{17}	Stripper underflow.
w_{18}	Stripper temperature.
w_{19}	Stripper steam flow.
w_{20}	Compressor work.
w_{21}	Reactor outlet temperature.
w_{22}	Condenser outlet temperature.

Table 2. Simulated Faults.

Fault	Disturbance
Type 1	PA/PC feed ratio.
Type 2	PB composition.
Type 3	PD feed temperature.
Type 4	Reactor cooling water inlet temperature.
Type 5	Condenser cooling water inlet temperature.
Type 6	PA feed loss.
Type 7	PC header pressure loss.
Type 8	PA, PB, PC feed composition

4.1 Evaluation Metrics

In fault detection, four results are possible:

1. True Positive (TP): the classifier indicates a fault when a fault is present;
2. False Positive (FP): the classifier indicates a fault when the operation is normal;
3. True Negative (TN): the classifier indicates a normal operation when the operation is normal;
4. False Negative (FN): the classifier indicates a normal operation when a fault is present;

The number of TP, FP, TN and FN can be used for sensitivity (SE) and specificity (SP) calculations as

$$\text{SE}(\%) = \left(\frac{TP}{TP + FN}\right) \times 100\% \tag{6}$$

$$\text{SP}(\%) = \left(\frac{TN}{TN + FP}\right) \times 100\% \tag{7}$$

In addition, a general Accuracy Rate (AR) can be defined as

$$\text{AR}(\%) = \left(1 - \frac{\text{Number of errors}}{\text{Number of objects}}\right) \times 100\% \tag{8}$$

4.2 Implementation details

SPA-LDA, GA-LDA and KNN classification routines were implemented in Matlab 7.6. In the KNN algorithm the number of neighbours κ was optimized according to the number of classification errors in the validation set. If two different values of κ provided the same result, the average number of errors obtained with $\kappa - 1$ and $\kappa + 1$ was considered to make the final decision. The GA employed standard binary chromosomes with length equal to the number of available features (a “1” gene indicates a selected feature). The fitness of each individual was taken as the inverse of the validation cost (Equation 4) calculated by using the features coded in the chromosome. The probability of a given individual being selected for the mating pool was proportional to its fitness (roulette method). One-point crossover and mutation operators were employed with probabilities of 60% and 10%, respectively. Population size was kept constant, each generation being completely replaced by its descendants. However, the best individual was automatically transferred to the next generation (elitism) to avoid the loss of good solutions. The GA was carried out for 100 generations with 200 chromosomes each.

The fault detection scheme adopted in the case study is illustrated in Figure 1. A classifier was obtained and tested for each type of fault.

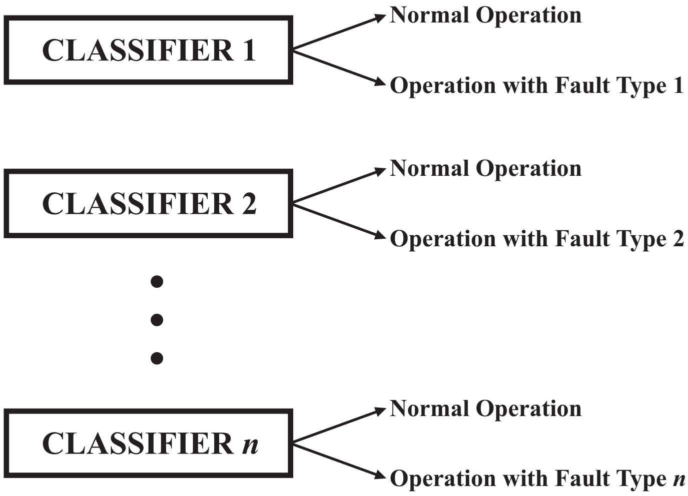


Figure 1. Fault detection scheme.

5. Results

For illustration, Figure 2 presents signal $w_1(t)$ resulting from a simulation with a fault of type 5 introduced after 100 sampling periods.

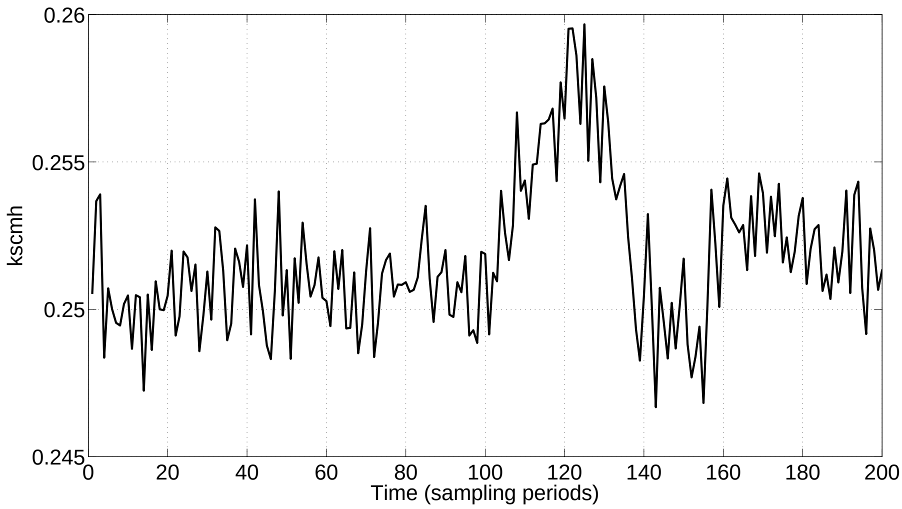
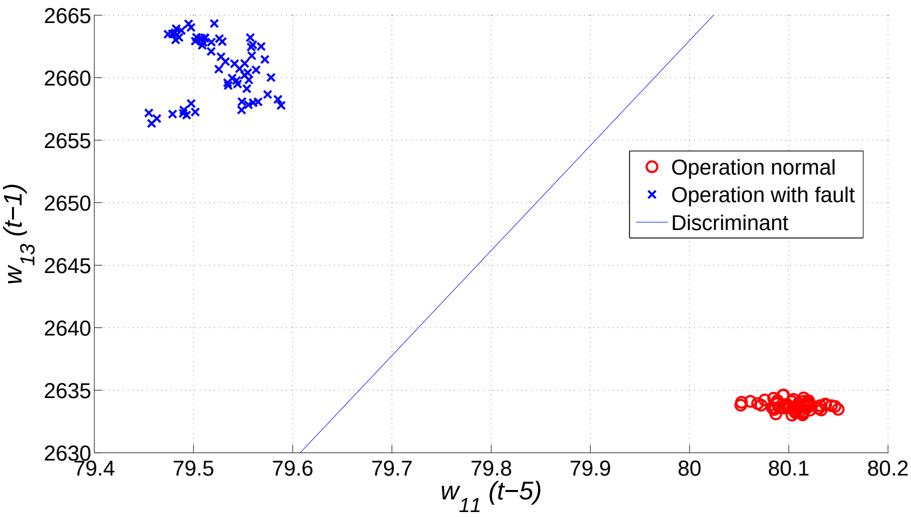


Figure 2. Signal w_1 (PA feed flowrate) with a fault of type 5 inserted after 100 sampling periods. The abbreviation kscmh referring to thousand standard cubic meters per hour.

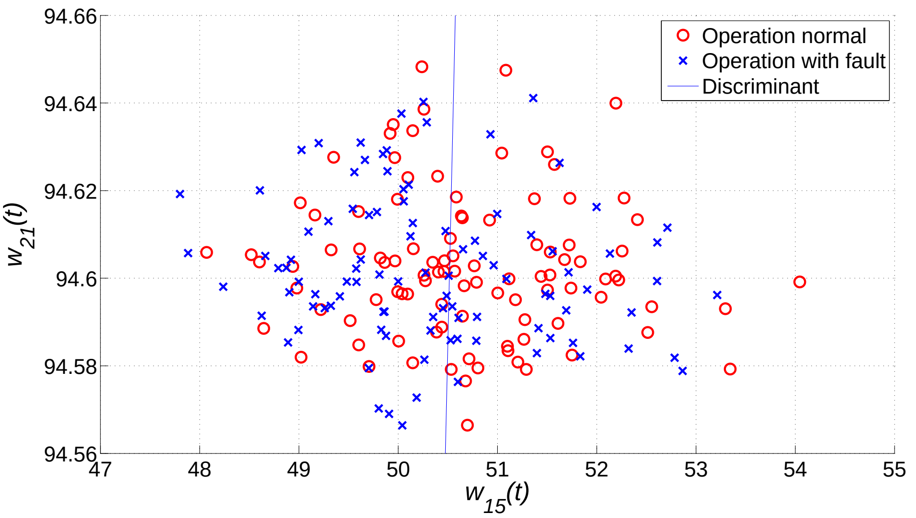
Table 3 presents the results obtained when no time lags are employed in the characterization of the objects. As can be seen, the performance of all three classification techniques (SPA-LDA, GA-LDA and KNN) was poor. In fact, the largest accuracy rate was 75% for fault type 8 employing SPA-LDA.

Table 4 presents the results obtained with the use of time lags. As can be seen, the classification performance was substantially improved. Such an improvement results from a better discrimination of the classes, as illustrated in Figures 3a and 3b for fault type 2.

As seen in Table 4, SPA-LDA and GA-LDA provided better overall results as compared to KNN. The performances of SPA-LDA and GA-LDA were comparable, with slight differences in fault types 3 and 8. However, the SPA-LDA classifiers were considerably more parsimonious (i.e. employed a much smaller number of features). Por example, for fault type 6, GA selected 42 features for the LDA classifier, whereas SPA selected 29 features (as indicated by the



(a) Discriminant frontier using features selected by SPA with time lags.



(b) Discriminant frontier using features selected by SPA without time lags.

Figure 3. Comparison of discriminant frontier with and without time lags for classification between normal operation and operation with fault type 2.

minimum point of the validation cost curve in Figure 4). The selected features for this fault type are shown in Figures 5(a) and 5(b). It is worth noting that the SPA-LDA classifier only employs six process variables (w_1, w_2, w_3, w_{11}, w_{14} and w_{15}, with various time lags), whereas the GA-LDA classifier requires the use of nineteen process variables (w_1, w_2, w_3, w_4, w_5, w_6, w_7 w_8, w_9, w_{10}, w_{11}, w_{14}, w_{15}, w_{16}, w_{17}, w_{18}, w_{19}, w_{21} and w_{22}, with various time lags). Therefore, the SPA-LDA solution would be more convenient for engineering purposes, as less sensors would be required to implement the fault detector.

6. Conclusions

This paper proposed a framework for use of SPA-LDA in process fault detection. For illustration, the proposed approach was employed in a simulated case study involving the Tennessee Eastman Process. In this example, better results were obtained by using both present and time-lagged values of the process variables as classification features. On the overall, the SPA-LDA results were superior to those provided by KNN and similar to those obtained by using LDA with features selected by a GA formulation. As compared to GA-LDA, the SPA-LDA classifiers were considerably more parsimonious, requiring the use of less sensors to implement the fault detectors.

These results suggest that the proposed methodology is a promising alternative for fault detection. Future works could extend the present investigation by using SPA with other classification techniques, such as QDA (Quadratic Discriminant Analysis).

Table 3. Results for fault detection without the use of time lags. The abbreviation NF and the symbol κ indicate the number of features in LDA and the number of neighbours in KNN, respectively.

	SPA-LDA				GA-LDA				KNN			
Fault	NF	AR (%)	SE (%)	SP (%)	NF	AR (%)	SE (%)	SP (%)	κ	AR (%)	SE (%)	SP (%)
Type 1	1	60	20	100	7	60	20	100	3	60	20	100
Type 2	3	50	0	100	2	50	60	40	4	55	100	10
Type 3	1	60	80	40	5	50	100	0	1	55	80	30
Type 4	2	65	80	50	4	70	90	80	6	60	60	60
Type 5	1	60	100	20	4	60	80	40	13	60	50	70
Type 6	1	60	100	20	3	60	80	40	5	55	100	20
Type 7	1	50	100	0	2	60	100	20	8	50	100	0
Type 8	2	75	85	65	3	60	80	40	6	50	100	0

Table 4. Results for fault detection with the use of time lags. The abbreviation NF and the symbol κ indicate the number of features in LDA and the number of neighbours in KNN, respectively.

	SPA-LDA				GA-LDA				KNN			
Fault	NF	AR (%)	SE (%)	SP (%)	NF	AR (%)	SE (%)	SP (%)	κ	AR (%)	SE (%)	SP (%)
Type 1	1	100	100	100	51	100	100	100	1	100	100	100
Type 2	2	100	100	100	64	100	100	100	3	100	100	100
Type 3	2	75	81	66	43	95	100	90	3	55	80	30
Type 4	1	100	100	100	44	100	100	100	29	55	10	100
Type 5	1	100	100	100	42	100	100	100	13	57	50	14
Type 6	29	100	100	100	42	100	100	100	31	100	100	100
Type 7	77	100	100	100	48	100	100	100	25	100	100	100
Type 8	2	85	62	100	64	70	80	90	1	100	100	100

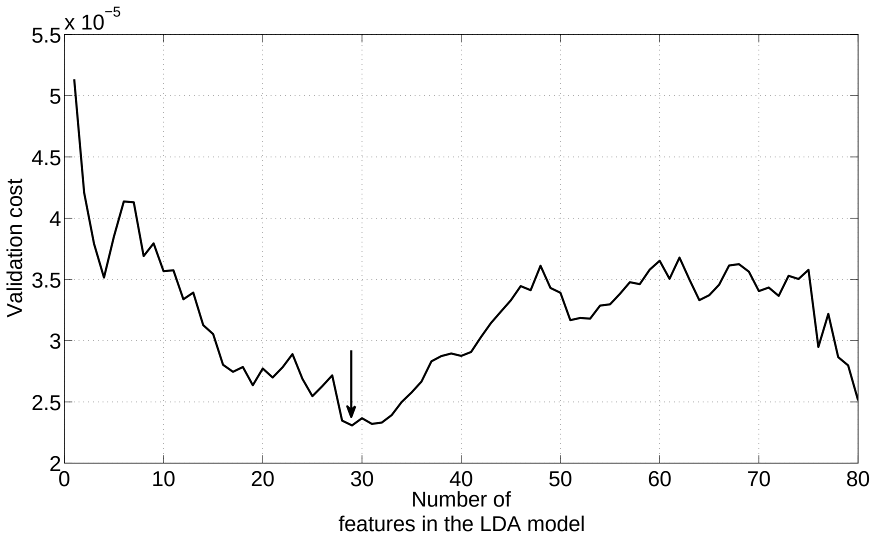
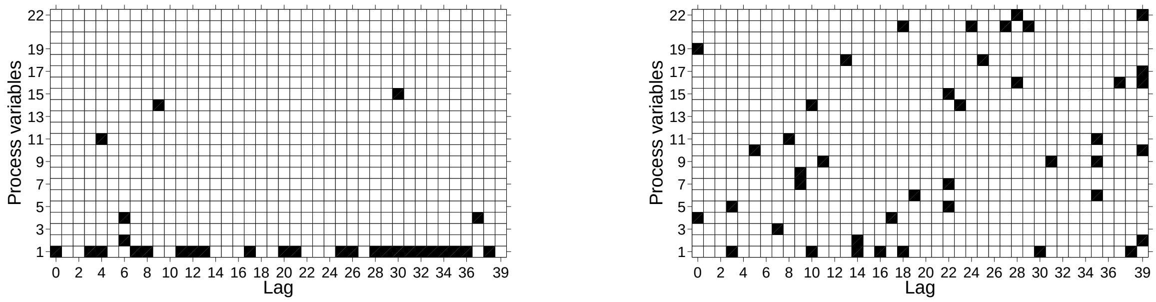


Figure 4. Validation cost as a function of the number of features selected by SPA for the detection of fault type 6. The arrow indicates the minimum point of the cost curve (29 features).



(a) Features selected by SPA.

(b) Features selected by GA.

Figure 5. Result comparasion of SPA and GA algorithm for LDA model for fault type 6. The black squares point out the selected features.

7. ACKNOWLEDGEMENTS

The authors acknowledge the support of FAPESP (Grant 2006/58850-6 and doctorate scholarship 2007/57803-7), PROCAD/CAPES (Grant 0081/05-1), CNPq (research fellowship) and CAPES (Pró-Engenharias).

8. REFERENCES

Casimir, R. , Boutleux, E. and Clerc, G., 2003, Fault diagnosis in an induction motor by pattern recognition methods, Proceedings of the 4th IEEE International Symposim on Diagnostics for Eletric Machines, 294-299.

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

Chiang, L. H. and Pell, R. J. , 2004, Genetic algorithms combined with discriminant analysis for key variable identification. *Journal of Process Control*, 14(1):143-155.

Duda R. O., Hart P. E. and Stork D. G., 2001, Pattern Classification. John Wiley, New York, 2 edition.

Downs J. J. and Vogel, E. F., 1993, A plant-wide industrial-process control problem, Computers and Chemical Engineering. Computers & Chemical Engineering, 17(3):245-255.

Friedman, J. H. , Baskett, F. and Shustek, L. J. . An Algorithm for Finding Nearest Neighbors, 1975, *IEEE Transactions on Computers.*, 24(10) 1000-1006.

Gambarra Neto, F.F., Marino, G., Araújo, M.C.U., Galvão, R.K.H., Pontes, M.J.C., Medeiros, E.P. and Lima, R.S., 2009, Classification of edible vegetable oils using square wave voltammetry with multivariate data analysis. Talanta, 77(5):1660-1666.

Gascaa, E. and Sánchezb, J. S. , 2006, Eliminating redundancy and irrelevance using a new MLP-based feature selection method. *Pattern Recognition*, 39(2) 313-315.

He, Q.P. and Wang, J., 2007, Fault Detection Using the k-Nearest Neighbor Rule for Semiconductor Manufacturing Processes *IEEE Transactions on Semiconductor Manufacturing*, 20(4):345-354.

Isermann, R., 2006, Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance *Springer*, New York, NY, USA.

Krzanowski, W.J., Jonathan, P., McCarthy, W.V., Thomas, M.R., 1995, Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data. Applied Statistics, 44:101-115.

Maesschalck, R., Rimbaud, D. J. and Massart, D.L., 2000, Chemom. Intell. Lab. Syst. 50:1Ű18.

Pei, X. , Yamashita, Y. , Yoshida, M. and Matsumoto, S. , 2008, Fault detection in chemical processes using discriminant analysis. *Journal of Chemical Engineering of Japan*, 41(1):25-31.

Pontes, M. J. C. , Galvão, R. K. H., Araújo, M. C. U., Nogueira, P. N. T. , Neto, O. D. P. , José, G. E. and Saldanha, T. C. B. ., 2005, The Successive Projections Algorithm for Spectral Variable Selection in Classification Problems. *Chemometrics and Intelligent Laboratory Systems*, 78(1):11-18.

Pontes, M. J. C., Cortez, J., Galvão, R. K. H., Pasquini, C., Araújo, M. C. U., Coelho, R. M., Chiba, M. K., Abreu, M.

F., Madari, B.E., 2009, Classification of Brazilian soils by using LIBS and variable selection in the wavelet domain. *Analytica Chimica Acta*, 642(12):12-18.

Venkatasubramanian, V. , Rengaswamy, R. , Yin, K. and Kavuri, S. N. , 2003, A review of process fault detection and diagnosis Part I: Quantitative model-based methods. *Computer and Chemical Engineering*, 27(1):293-311.

Wang, S. and Xiao, F. ., 2004, Sensor fault diagnosis using principal component analysis method. *Energy and Buildings*, 36(2) 147-160.

Wu, W. , Mallet, Y. , Walczak, B. , Penninckx, W. , Massart, D. L. and Heuerding, S. F., 1996, Comparison of regularized discriminant analysis, linear discriminant analysis and quadratic discriminant analysis, applied to NIR data. *Chemom. Intel. Lab. Syst.*, 329(3) 257-265.

9. Responsibility notice

The authors are the only responsible for the printed material included in this paper.