Int J Adv Manuf Technol (2006) 28: 1006–1011
DOI 10.1007/s00170-004-2434-7

ORIGINAL ARTICLE

Buhwan Jeong · Hyunbo Cho

Feature selection techniques and comparative studies for large-scale manufacturing processes

Received: 10 May 2004 / Accepted: 6 October 2004 / Published online: 15 March 2006

Abstract In order to efficiently and effectively control an overall process in the process industry, a few important parameters should be identified from high-dimensional, non-linear, and correlated data. Feature selection techniques can be employed to extract a subset of process parameters relevant to product quality. The performance of these techniques depends on the precision of the prediction model formulated to quantify the relationship between the process parameters and the quality characteristics. Although the neural network-based partial least squares (NNPLS) method has been proven to be effective in prediction models for the aforementioned industrial process data, feature selection techniques appropriate for NNPLS models have yet to appear. Here, several techniques for scoring the relevance of process parameters to product quality are proposed and validated by applying three datasets. These experiments show that the proposed techniques can discriminate relevant process parameters from irrelevant ones.

Keywords Feature selection · Neural network-based partial least squares (NNPLS) · Prediction model · Process data

1 Introduction

In general, the quality characteristics of products produced in the process industry, such as steelworks and photochemical manufacturing, are affected by numerous process parameters. However, not all process parameters are equally informative; some may be noisy, meaningless, correlated, or irrelevant to the quality characteristics of interest. Feature (or variable) selection techniques aim to discriminate between process parameters that are relevant to the product quality and those that are not. These techniques enable process operators to control a few vital process parameters rather than all parameters.

The philosophy of feature selection dates back to the 19th century, when the Italian economist Pareto formulated a general rule linking a small number of causes with a large number of corresponding effects. This general rule can be directly applied to the data collected from most process industries in order to identify a small number of vital parameters that significantly affect product quality [1–4]. Feature selection has become a major research topic, focusing on questions such as "which process parameters should be controlled?" and "how relevant and meaningful are they?" [5]. A prerequisite for successful feature selection is the construction of a good prediction model (or classifier) between process parameters and quality characteristics.

Multivariate projection approaches (e.g., partial least squares (PLS) and neural network-based PLS (NNPLS)) are used to build prediction models for highly correlated, high-dimensional datasets by reducing their input space to a lower dimensional space. The PLS method projects the original variables (process parameters and quality characteristics) down to a few latent variables between which a linear relationship can be established [6, 7]. However, this linear mapping may not yield a robust prediction model for high-dimensional and non-linear data. The NNPLS method, which uses neural networks for mapping the latent variables, avoids the problems of over-parameterization and convergence to local minima that commonly occur in multi-input/multi-output networks [8]. But despite the success of the NNPLS model in building prediction models, it has not been widely used in feature selection.

In this work, several NNPLS-based feature selection techniques for high-dimensional and non-linear data are proposed. Several other feature selection techniques are introduced for the purpose of comparison. Finally, the performance of the proposed techniques is compared with other techniques with known artificial datasets.

This paper is organized as follows: in Sect. 2, the notations used in the NNPLS model are briefly described; in Sect. 3, several feature selection techniques based on the NNPLS model

B. Jeong · H. Cho
Department of Industrial Engineering,
Pohang University of Science and Technology,
San 31 Hyoja, Pohang 790-784, Republic of South Korea
E-mail: hcho@postech.ac.kr
Tel.: +82-54-279-2204
Fax: +82-54-279-2870

are proposed; the performance of the proposed techniques is assessed in Sect. 4; and finally, concluding remarks are offered in Sect. 5.

2 NNPLS model and its transformation

2.1 Basic concept of the NNPLS model

Without a loss of generality, process parameters will be referred to as input variables $[x_1, \ldots, x_m]$, and quality characteristics will be referred to as output variables $[y_1, \ldots, y_n]$. Assuming that we have collected the data for each input and output variable, let $\mathbf{X}(d \times m)$ be the input data matrix, and $\mathbf{Y}(d \times n)$ be the output data matrix. Thus, the matrices $\mathbf{X} = (x_{ij})$ and $\mathbf{Y} = (y_{ik})$ represent the dataset of input variables and the corresponding output variables, respectively, where x_{ij} and y_{ik} represent the ith observation of input variable x_j and output variable y_k, respectively (where $i = 1, \ldots, d$; $j = 1, \ldots, m$; and $k = 1, \ldots, n$).

The basic concept of the NNPLS method is to reduce the original high-dimensional variables ($[x_1, \ldots, x_m]$ and $[y_1, \ldots, y_n]$) to lower dimensional, principal component-like latent variables ($[t_1, \ldots, t_A]$ and $[u_1, \ldots, u_A]$, where A is the number of latent variables), as follows:

$$\mathbf{X} = \mathbf{TP}^T + \mathbf{E} \tag{1}$$

$$\mathbf{Y} = \mathbf{UQ}^T + \mathbf{F} \tag{2}$$

$$\mathbf{U} = N(\mathbf{T}) \tag{3}$$

$$\mathbf{T} = \mathbf{XW} \tag{4}$$

where $\mathbf{P}(m \times A)$ and $\mathbf{Q}(n \times A)$ are the loading matrices; $\mathbf{T}(d \times A)$ and $\mathbf{U}(d \times A)$ are the score matrices; and $\mathbf{E}(d \times m)$ and $\mathbf{F}(d \times n)$ are the residual matrices for $\mathbf{X}$ and $\mathbf{Y}$, respectively. The weight matrix for $\mathbf{X}$, $\mathbf{W}(m \times A)$ makes the score vectors ($\mathbf{t}_a$) orthogonal to each other. The relationship function $N()$, which describes the inner relationship between $\mathbf{T}$ and $\mathbf{U}$, is constructed using A individual single-input/single-output (SISO) neural networks. Hence, a distinct neural network is used to build the connection between $\mathbf{T}$ and $\mathbf{U}$ for each latent variable (e.g. the ath neural network is used to build the connection of the ath pair of scores, $\mathbf{t}_a$ and $\mathbf{u}_a$ ($a = 1, \ldots, A$)) [9].

2.2 Transformation of a three-layer neural network

The NNPLL model can be viewed as a five-layer neural network, as shown in Fig. 1. It is well-known that the NNPLS model can be transformed into an equivalent three-layer neural network [9]. This transformation entails inner neural networks synthesis (Fig. 2), followed by outer model collapse (Fig. 3). It should be noted that the bias vectors are omitted.

First, the A individual inner neural networks are synthesized into one neural network, as shown in Fig. 2. The hidden nodes of the synthesized neural network are an exclusive aggregation of those of the inner neural networks, and therefore the number of hidden nodes (n_H) is the sum of the number of hidden nodes (H_a) of the inner neural networks. The weight matrices of input-to-hidden $\mathbf{\Phi}(A \times n_H)$ and hidden-to-output $\mathbf{\Gamma}\,(n_H \times A)$ can be constructed in sparse matrices in which the non-connected weights are filled with zeros (0).

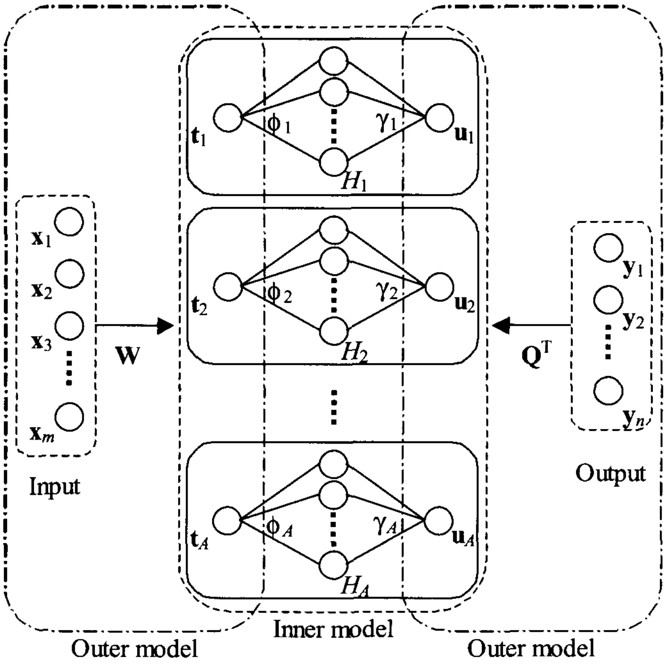

Fig. 1. NNPLS model as a five-layer neural network

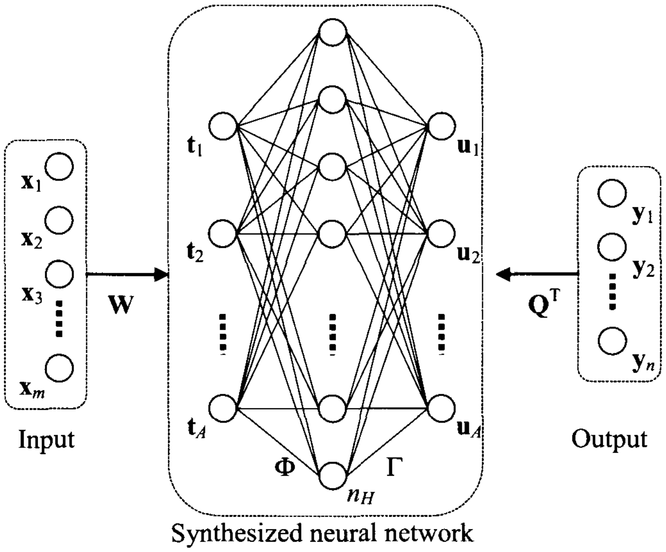

Fig. 2. Synthesis of inner neural networks

Then, a three-layer neural network is constructed using the synthesized neural network and the NNPLS outer model (e.g., $\mathbf{P}$ and $\mathbf{Q}$), as shown in Fig. 3. The weight matrix from the input nodes to the hidden nodes ($\mathbf{\Omega}$) is calculated as the product of the weight matrix for $\mathbf{X}$ (e.g., $\mathbf{W}$) and the input-to-hidden weight matrix of the synthesized neural network (e.g., $\mathbf{\Phi}$). Similarly, the weight matrix from the hidden nodes to the output nodes

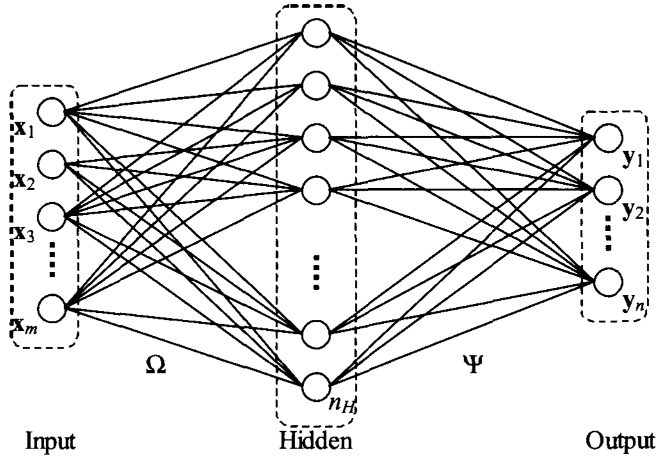

Fig. 3. Transformed three-layer neural network

(e.g., ψ) is computed as the product of $\mathbf{\Gamma}$ and $\mathbf{Q}^T$. The bias vectors can be computed in a similar way; however, the procedure was skipped because the bias vectors are not used in the description of the techniques proposed in our work. More detailed descriptions of the NNPLS method and the transformation procedure can be found in [9]. The transformation makes possible the rapid construction of a robust neural network model. In addition, since the NNPLS method resolves the co-linearity among $[x_1, \ldots, x_m]$ as well as between $[x_1, \ldots, x_m]$ and $[y_1, \ldots, y_n]$, the transformation naturally produces a co-linearity-free neural network [10].

3 Feature selection using NNPLS models

3.1 Using the transformed neural network model

Since the NNPLS model can be transformed into a three-layer neural network model, the feature selection techniques developed previously for neural network models can also be incorporated into those for the NNPLS model. These techniques can be classified into zero order (e.g. Garson's method [11]), first order (e.g. saliency based pruning, computation of output derivative), and second order (e.g. automatic relevance determination, optimal cell damage, early cell damage) methods. The first and second order methods use the first and second derivatives of the neural network parameters, respectively, whereas the zero order methods use the network parameters themselves. Some of these feature selection techniques and related issues are summarized in [12].

One well-known feature selection technique is Garson's method, which defines the relevance measure as the contribution of an input node to an output node, and computes this contribution by exploiting both the connection weight value and the neural network structure. The computational load for weight analysis is known to be low, especially when modeling large-scale datasets [5].

In Garson's method, the proportional contribution (S_{jk}) of the jth input node to the kth output node is calculated as follows:

$$S_{jk} = \frac{\sum_{l=1}^{n_H}\left(\frac{|\Omega_{lj}|}{\sum_{\alpha=1}^{m}|\Omega_{l\alpha}|}\,|\Psi_{kl}|\right)}{\sum_{\beta=1}^{m}\left(\sum_{l=1}^{n_H}\left(\frac{|\Omega_{l\beta}|}{\sum_{\alpha=1}^{m}|\Omega_{l\alpha}|}\,|\Psi_{kl}|\right)\right)},$$

$$j = 1, \ldots, m \text{ and } k = 1, \ldots, n \tag{5}$$

In this equation, an absolute scale of connection weights is used because the mixture of positive and negative weights can potentially produce an average close to zero.

The relevance measure (e.g., the overall contribution, S_j) of the jth input for all the output variables is defined as follows. The denominators of eqs. 5 and 6 function are normalizing factor, and the input matrix should be identically scaled, since the weights depend on the magnitude of the input.

$$S_j = \frac{\sum_{k=1}^{n}(S_{jk})}{\sum_{\alpha=1}^{m}\sum_{k=1}^{n}(S_{\alpha k})}, \quad j = 1, \ldots, m \tag{6}$$

3.2 Using original NNPLS models

3.2.1 Using latent variables

The latent variables of an NNPLS model can also be used for feature selection of the input variables. The latent variables $[t_1, \ldots, t_A]$ are representative of the input variables $[x_1, \ldots, x_m]$ because the scores t_a are the projections of $[x_1, \ldots, x_m]$ onto the loadings p_a, and the loading matrix $\mathbf{P}$ of $\mathbf{X}$ contains the projection information describing how the input variables are related to each other. The relevance measure (S_j) of the jth input variable can therefore be obtained by multiplying the jth row of the loading matrix by the relevance measure of the latent variables. The larger the magnitude of the relevance measure of an input variable, the greater its relevance. Since the latent variables are independent of each other, they can be ranked according to their individual relevance measures. Below, several methodologies that can be used to calculate the relevance measures of the latent variables are described.

R^2 for the explained variance of Y (R^2Y). R^2Y, the fraction of the sum of squares of all Ys explained by the extracted latent variables, is a standard measure of model fitness. Since the cumulative R^2Y approaches 1.0 as the number of extracted latent variables (A) increases, latent variables whose value of R^2Y is closer to 1.0 are more representative. Hence, R^2Y can be used as an index for determining the relevance of a latent variable. In general, the latent variables extracted earlier have larger R^2Y values than those extracted later. The relevance measure of the ath latent variable is calculated as

follows:

$$V_a = R^2Y(\mathbf{t}_a) = 1 - SS(\mathbf{F}_a)/SS(\mathbf{Y}) \tag{7}$$

where SS denotes the sum of squares, and F_a is the residual after a latent variables have been extracted. Therefore, the relevance measure (S_j) of the jth input variable is obtained as follows:

$$S_j = \sum_{a=1}^{A} \mathbf{p}_{ja} V_a = \sum_{a=1}^{A} \mathbf{p}_{ja}(1 - SS(\mathbf{F}_a)/SS(\mathbf{Y})) \tag{8}$$

Q^2 for the predicted variance of $Y(Q^2Y)$. Another standard measure of model fitness in terms of its predictive power is Q^2Y, the fraction of the sum of squares of all Ys that can be predicted by the model according to the cross-validation. Similar to R^2Y, the cumulative value of Q^2Y also approaches 1.0 as the number of latent variables (A) increases, and therefore variables with larger value of Q^2Y are more predictive. The cumulative Q^2Y (Q^2Y_{cum}) is calculated as follows:

$$Q^2Y_{\text{cum}}(a) = 1 - \prod_{\alpha=1}^{a} PRESS(\boldsymbol{F}_\alpha)/SS(\boldsymbol{F}_\alpha) \tag{9}$$

where PRESS, the prediction residual sum of squares, is the sum of squares of the cross-validation residuals (e.g., the squared difference between the observed values (y_{ik}) and the predicted values ($\hat{y}_{ik}$)):

$$PRESS = \sum_{i=1}^{d}\sum_{k=1}^{n}(y_{ik} - \hat{y}_{ik})^2 \tag{10}$$

PRESS is used to determine the number of latent variables and to prevent NNPLS models from being overfitted [13].

Thus, after computing the relevance measure of the ath latent variable (Eq. 11), the relevance measure (S_j) of the j^{th} input variable is calculated as Eq. 5:

$$\begin{aligned} V_a &= Q^2Y_{\text{cum}}(a) - Q^2Y_{\text{cum}}(a-1) \\ &= (1 - PRESS(\mathbf{F}_a)/SS(\mathbf{F}_a)) \prod_{\alpha=1}^{a-1} PRESS(\mathbf{F}_\alpha)/SS(\mathbf{F}_\alpha) \end{aligned} \tag{11}$$

$$\begin{aligned} S_j &= \sum_{a=1}^{A} \mathbf{p}_{ja} V_a \\ &= \sum_{a=1}^{A} \mathbf{P}_{ja} \\ &\quad \times \left((1 - PRESS(\mathbf{F}_a)/SS(\mathbf{F}_a)) \prod_{\alpha=1}^{a-1} PRESS(\mathbf{F}_\alpha)/SS(\mathbf{F}_\alpha) \right) \end{aligned} \tag{12}$$

Sensitivity analysis. The sensitivity analysis computes the variation of the predicted outputs by changing the value of one latent variable over its possible range while assigning nominal values to the other latent variables.

3.2.2 Using input variables

The relevance of a variable can be measured directly; however, direct measurement techniques are less efficient than those using latent variables.

Sensitivity analysis. This technique computes the variation of the predicted outputs by changing the value of one input variable over its possible range while assigning nominal values to the other input variables.

Saliency-based pruning method. The saliency-based pruning method has been used to evaluate the relevance of an input variable x_j in terms of the variation of the learning error (MSE) when x_j is replaced with its empirical mean $\bar{x}_j$ (that is, $S_j = MSE(x_j) - MSE(\bar{x}_j)$) [12]. Instead of the learning error, the cumulative R^2Y is used as a relevance measure for NNPLS models, as follows:

$$S_j = \sum_{a=1}^{A} R^2Y(\mathbf{X}) - \sum_{a=1}^{A} R^2Y(\bar{\mathbf{X}}_j) \tag{13}$$

where $\bar{\mathbf{X}}_j$ denotes the matrix in which the jth column is replaced with the empirical mean.

4 Comparative studies

The following abbreviations are used for the various techniques employed to calculate the relevance measure:

1. GAS – Garson's method
2. R2Y – R^2 for explained variance of Y
3. Q2Y – Q^2 for predicted variance of Y
4. SAL – sensitivity analysis with latent variables
5. SAO – sensitivity analysis with original variables
6. SBP – saliency-based pruning method

4.1 Case 1: three-class waveforms classification

The first case is a three-class waveforms classification problem [12]. Suppose that three waveforms (e.g., vectors) in 21 dimensions, H^i ($i = 1$, 2, and 3), are given, and that patterns in each class are defined as random convex combinations of two of these waveforms. Hence, three classes are available patterns defined by a random convex combination of H^1 and H^2 (class 1); patterns defined by a random convex combination of H^1 and H^3 (class 2); and patterns defined by a random convex combination of H^2 and H^3 (class 3). Thus, the problem is to classify the patterns into one of the three classes. Each component x_j in

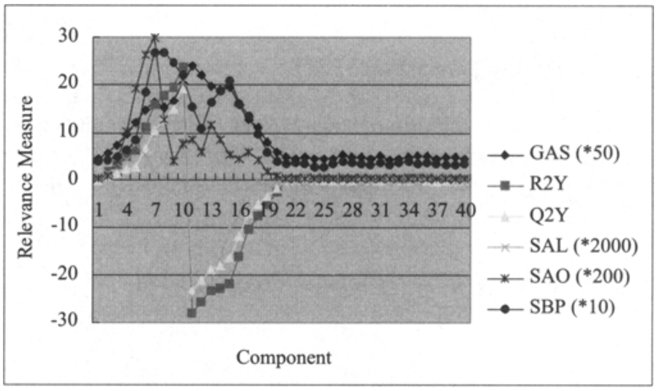

Fig. 4. Experimental results of the waveform classification problem

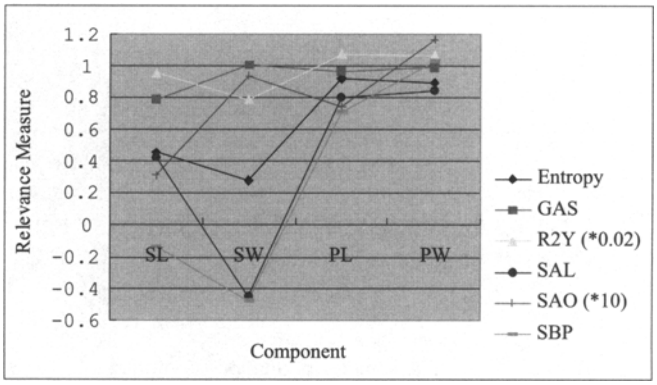

Fig. 6. Experimental results of the modified waveform classification problem

the pattern x is generated with 19 extra pure noise components according to the equation:

$$x_j = \begin{cases} \frac{uH_j^p+(1-u)H_j^q}{5} + \varepsilon_j & 1 \leq j \leq 21 \\ \varepsilon_j & 22 \leq j \leq 40 \end{cases} \tag{14}$$

where u is a uniform random variable in $[0, 1]$; ε_j is a white noise generated according to a normal distribution $N(0, 1)$; and p and q identify the two waves used to generate the class of pattern x. It should be noted that each component corresponds to an input variable, while each class corresponds to an output variable.

The experiments are performed with an open data (http://www.sgi.com/tech/mlc/db/waveform-40.all) generated according to Eq. 14, and the number of latent variables is set to five. Figure 4 shows the average relevance measure of each component. The relevant input variables (x_1 to x_{21}) have larger relevance measures than the irrelevant input variables; thus, the relevance measures can be used to discriminate the relevant variables from numerous input variables.

Additional experiments were performed to show that the relevant variables are effective to classify the waveforms. For each technique, the training set has 300 patterns, and the test set has 4700 patterns. Figure 5 lists the experimental results. First, all the techniques eliminate the pure noise components. Second, classifications using relevant input variables give slightly better performance (percentage of correct classification) than classification using all the input variables, except in the case of Bonnlander's technique. Third, feature selection techniques using the NNPLS model provide performance similar to neural networks techniques.

Feature selection techniques	Selected input variables	Performance	
		R^2Y	% of correctly classified patterns
None	111111111111111111111 1111111111111111111	93.55	71.15
Relevance only	111111111111111111111 0000000000000000000	91.41	75.36
Stepdisc[1]	000110111111111011100 0000000000000000000	88.31	73.36
Bonnlander[1]	000011101111111111000 0000000000000000000	88.18	69.30
Yaccoub, Moody[1]	000111111111111111100 0000000000000000000	87.13	74.77
Ruck, Dorizzi[1]	011111111111111111100 0000000000000000000	89.52	75.30
Czernichow[1]	010111111111111111100 0000000000000000000	88.31	76.15
Cibas[1]	000001111111011100000 0000000000000000000	83.74	72.72
Leray[1]	000001111111111100000 0000000000000000000	85.57	71.45
GAS	001111111111111111110 0000000000000000000	88.97	75.36
R2Y, Q2Y, SAO	011111111111111111110 0000000000000000000	89.42	76.09
SAL	001111011111111100000 0000000000000000000	89.10	72.57
SBP	000011111111111111000 0000000000000000000	87.61	72.81

[1] Feature selection techniques developed for neural networks

Fig. 5. Performance comparison of various feature selection techniques

4.2 Case 2: Modified waveform classification

Now, the feature selection techniques are applied to a modified waveform classification problem, in which some components in the pattern are generated by means of negative relevance to the waveform class. The patterns are generated according to the equation:

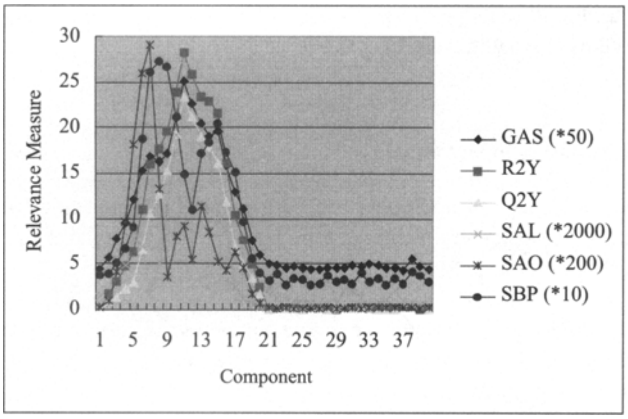

Fig. 7. Experimental results of IRIS classification data

$$x_j = \begin{cases} \frac{uH_j^p+(1-u)H_j^q}{5} + \varepsilon_j & 1 \le j \le 10 \\ -\left(\frac{uH_j^p+(1-u)H_j^q}{5}\right) + \varepsilon_j & 10 < j \le 21 \\ \varepsilon_j & 22 \le j \le 40 \end{cases} \tag{15}$$

One hundred experiments are repeated using 500 randomly selected patterns. It should be noted that the number of latent variables is five. The experimental results are depicted in Fig. 6. The relevance measures of components 11 to 21 obtained using R2Y, Q2Y, and SAL are similar in magnitude but opposite in sign to those of components 1 to 10. This implies that R2Y, Q2Y, and SAL can classify the components both in terms of the direction and the magnitude of the relevance measures, whereas GAS, SAO, and SBP can classify the components only in terms of the magnitude.

4.3 Case 3: IRIS dataset

The techniques are also applied to a dataset consisting of four input variables [sepal length (SL), sepal width (SW), petal length (PL) and petal width (PW)] and one output variable (e.g., type of iris: Setosa, Versicolor or Virginica), where the aim is to classify the type of iris. Each iris type is defined by 50 input data; hence there are 150 samples in total. Based on the concept of entropy, PL and PW are known to be much more relevant for classifying iris-type than SL and SW [14]. The number of latent variables is set to two.

Ten experiments were repeated with 50 randomly selected samples, and the remaining 100 samples were used as the validation data. As shown in Fig. 7, GAS and SAO do not identify SW as an irrelevant variable. According to the ANOVA results, R2Y discriminates between SL and PW with 99% confidence (the F-value is 237.0 and the F-critical is 8.29).

5 Discussion and conclusion

We have tested the feature selection techniques for identifying the input variables that are relevant to output variables. The techniques considered included neural network models and NNPLS models. The NNPLS models were further categorized depending on whether they use the latent variables or the original input variables. The NNPLS method has been widely used for building prediction models from industrial process data that are high-dimensional, non-linear, and correlated. The performance of these techniques depends on the quality of the prediction model used.

In the experiments, the proposed methods effectively identified the input variables relevant to the output variables of interest. Moreover, the experiments considering the modified waveform classification problem showed that the techniques based on latent variables can determine the direction of the relevance measure, as well as its magnitude. The results showed that some techniques are incapable of discriminating relevant and irrelevant variables for particular datasets. Thus, the characteristics of a dataset must be considered when choosing the most suitable technique to apply. In process management, relevance is not the only factor used to determine which process parameters are more important and must be optimized. For example, defects in products often result from instabilities in irrelevant process parameters. Thus, the domain knowledge of process operators and the process stability index [15] should be taken into consideration together with the relevance measures.

Acknowledgement The authors acknowledge the partial financial support of the BK21 project.

References

1. Fang K, Lin DK, Ma CX (2000) On the construction of multi-level supersaturated designs. J Stat Plann Inference 86(1):239–252
2. Hocking RR (1976) The analysis and selection of variables in linear regression. Biometrics 32:1–51
3. Park Y, Yoon B, Lim J (1999) Comparing fault prediction models using change request data for a telecommunication system. ETRI J 21(3):6–15
4. Rossi F (1996) Attribute suppression with multilayer perceptron. In: Proceedings of the CESA Multiconference on Robotics and Cybernetics, IMACS, Lille, France, pp 542–547
5. Laar P, Heskes T, Gielen S (1999) Partial training: a new approach to input relevance determination. Int J Neural Syst 9(1):75–85
6. Geladi P, Kowalski BR (1986) Partial least-squares regression: a tutorial. Analytica Chemica Acta 185:1–17
7. Wold S, Trygg J, Berglund A, Antti H (2001) Some recent developments in PLS modeling. Chemometrics Intell Lab Syst 58:131–150
8. Patwardhan R, Lakshminaraynan S, Shah S (1998) Constrained nonlinear MPC using Hammerstein and Winer models: PLS framework. Alche J 44(7):1611–1622
9. Qin SJ, McAvoy TJ (1992) Non-linear PLS modeling using neural networks. Comput Chem Eng 16(4):379–391
10. Qin SJ (1997) Neural networks for intelligent sensors and control – practical issues and some solutions. In: Omidvan O, Elliot DL (eds.) Neural systems control, Academic Press, San Diego. CA., pp 213–234
11. Garson GD (1991) Interpreting neural-network connection weights. AI Experts 6(4):47–51
12. Leray P, Gallinari P (1999) Feature selection with neural networks. Behavior metrika 26(1):145–166
13. Wold S (1978) Cross-validatory estimation of the number of components in factor and principal components models. Techometrics 20:397–405
14. Hong TP, Chen JB (1999) Finding relevant attribute and membership functions. Fuzzy Sets Syst 103:389–404
15. MacGregor JF, Kourti T (1995) Statistical process control of multivariate processes. Control Eng Pract 3(3):403–414