Chapter 9
Multivariate Statistical and Computational Intelligence Techniques for Quality Monitoring of Production Systems

Tibor Kulcsár, Barbara Farsang, Sándor Németh and János Abonyi

Abstract The ISO 9001:2008 quality management standard states that organizations shall plan and implement monitoring, measurement, analysis and improvement processes to demonstrate conformity to product requirements. According to the standard, detailed analysis of data is required for this purpose. The analysis of data should also provide information related to characteristics and trends of processes and products, including opportunities for preventive action. The preliminary aim of this chapter is to show how intelligent techniques can be used to design data–driven tools that are able to support the organization to continuously improve the effectiveness of their production according to the Plan—Do—Check—Act (PDCA) methodology. The chapter focuses on the application of data mining and multivariate statistical tools for process monitoring and quality control. Classical multivariate tools such as PLS and PCA are presented along with their nonlinear variants. Special attention is given to software sensors used to estimate product quality. Practical application examples taken from chemical and oil and gas industries illustrate the applicability of the discussed techniques.

Keywords Multivariate statistics · Computational intelligence · Quality monitoring · Production systems · PDCA

9.1 Introduction

The modern definition of quality states that "quality is inversely proportional to variability". This definition implies that if variability in the important characteristics of a production system decreases, then the quality of the product increases. Statistical process control (SPC) provides techniques to assure and improve the

T. Kulcsár · B. Farsang · S. Németh · J. Abonyi (✉)
Department of Process Engineering, University of Pannonia, Veszprém 158 Hungary
e-mail: janos@abonyilab.com

C. Kahraman and S. Yanık (eds.), *Intelligent Decision Making in Quality Management*, Intelligent Systems Reference Library 97,
DOI 10.1007/978-3-319-24499-0_9

quality of products by reducing the variance of process variables. The role of these tools is illustrated in Fig. 9.1, which presents a manufacturing process. The control chart of SPC is a very useful process monitoring technique, when unusual sources of variability are present and important process variables will plot outside the control limits. In these cases some investigation of the process should be made and corrective action to remove these unusual sources of variability should be taken. Systematic use of a control chart is an excellent way to reduce variability (Montgomery 2009).

As new products are required to be introduced to the market over a short time scale to ensure competitive advantage, the development of process monitoring models of multi-product manufacturing environment necessitates the use of empirical based techniques as opposed to first-principles models since phenomenological model development is unrealizable in the time available. Hence, the mountains of data, that computer-controlled plants generate, must be used by the operator support systems to distinguish normal from abnormal operating conditions. Detection and diagnosis of faults and control of product quality are the pivotal tasks of plant operators. The aim of multivariate statistical based approaches is to reduce the dimensionality of the correlated process data by projecting them down onto a lower dimensional latent variable space where the operation can be easily visualized and hidden functional relationships among process and quality variables can be detected.

In modern production systems huge amount of process operational data are recorded. These data definitely have the potential to provide information for product and process design, monitoring and control (Yamashita 2000). This is especially important in many practical applications where first-principles modeling of complex "data rich and knowledge poor" systems are not possible (Zhang et al. 1997). The term knowledge discovery in databases (KDD) refers to the overall process of

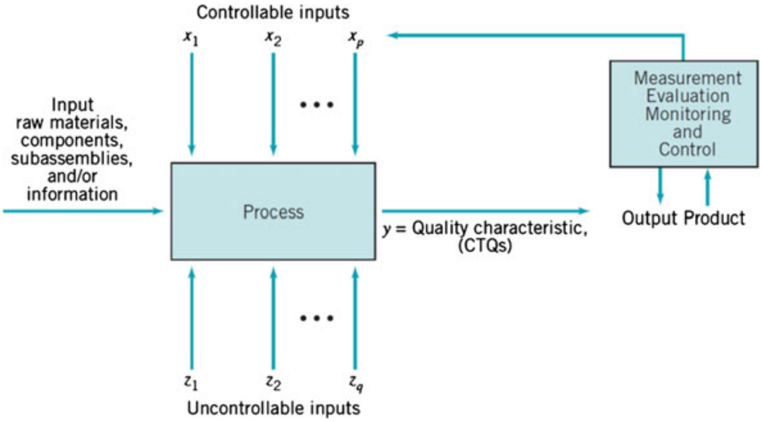


Fig. 9.1 Scheme of a production process where statistical process control (SPC) can be applied to improve the quality characteristic by adjusting and monitoring important process variables (Montgomery 2009)

discovering knowledge from data. KDD has evolved from the intersection of research fields such as machine learning, pattern recognition, databases, statistics, artificial intelligence, and more recently it gets new inspirations from soft computing. KDD methods have been successfully applied in the analysis of process systems, and the results have been used for process design, process improvement, operator training and so on (Wang 1999).

Application of knowledge discovery and data mining for quality development requires sophisticated methodology. Deming recommended the four steps (Plan, Do, Check, Act) based PDCA cycle as model to guide improvement. In the *Plan* step, we propose a change in the system that is aimed at improvement. In *Do*, we carry out the change, usually on a small or pilot scale to ensure that to learn the results that will be obtained. *Check* consists of analyzing the results of the change to determine what has been learned about the changes that we carried out. In *Act*, we either adopt the change or, if it was unsuccessful, abandon it. The process is almost always iterative, and may require several cycles for solving complex problems. It is interesting to note that the concept of PDCA is also applied in data mining. CRISP-DM stands for Cross Industry Standard Process for Data Mining (CRISP-DM 2000) (see Fig. 9.2). It is a data mining process model that describes commonly used approaches that expert data miners use to tackle problems.

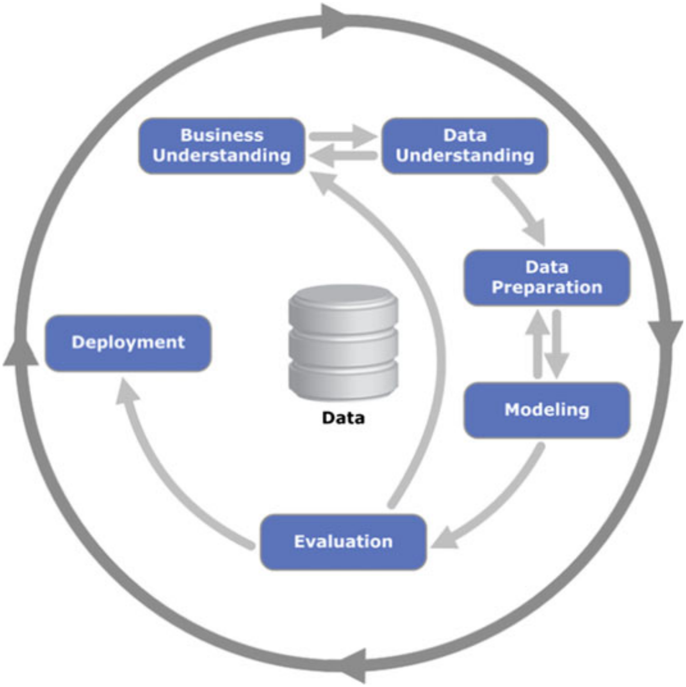


Fig. 9.2 The CRISP-DM methodology as continuous data-driven improvement process (CRISP-DM 2000)

Plan:
Business understanding: This initial phase focuses on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a data mining problem definition and a preliminary plan designed to achieve the objectives.
Data understanding: The data understanding phase starts with initial data collection and proceeds with activities that identify data quality problems, discover first insights into the data, and/or detect interesting subsets to form hypotheses regarding hidden information.

Do:
Data preparation: The data preparation phase covers all activities needed to construct the final dataset from the initial raw data.
Modeling: In this phase, various modelling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem. Some techniques have specific requirements on the form of data. Therefore, going back to the data preparation phase is often necessary.

Check:
Evaluation: At this stage, model (or models) is built that appears to have high quality from a data analysis perspective. A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.

Act:
Deployment: Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge, the knowledge gained should be organized and presented in a way that the customer can use it. It often involves applying "dynamic" models within an organization's decision making processes—for real-time control. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data mining process across the enterprise.

The previously presented data mining procedure should be embedded into the whole quality development process. As we mentioned, most of quality management methodologies are based on intensive analysis of data. Among the wide ranges of methodologies, we suggest the application of DMAIC (Define, Measure, Analyze, Improve, and Control) process (see Fig. 9.3). DMAIC is a structured problem-solving procedure extensively used in quality and process improvement.

Among the wide range of data mining tools, in this chapter we focus on multivariate statistical tools that are extensively applied in process monitoring and quality development.

Process monitoring based on multivariate statistical analysis of process data has recently been investigated by a number of researchers (MacGregor and Kourti 1995). The aim of these approaches is to reduce the dimensionality of the correlated process data by projecting them down onto a lower dimensional latent variable

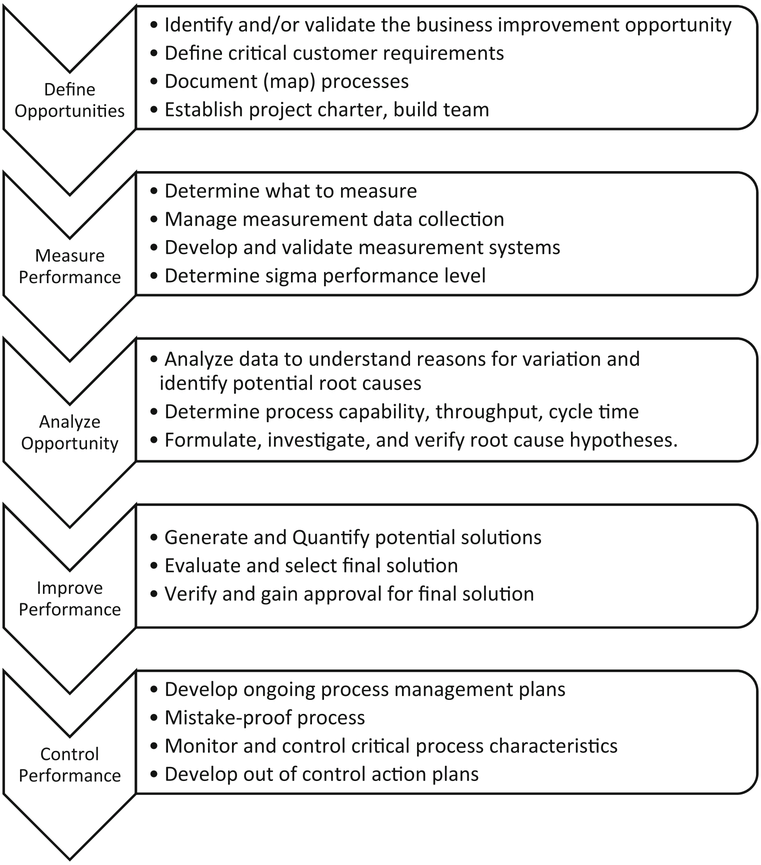

Define Opportunities
• Identify and/or validate the business improvement opportunity
• Define critical customer requirements
• Document (map) processes
• Establish project charter, build team
Measure Performance
• Determine what to measure
• Manage measurement data collection
• Develop and validate measurement systems
• Determine sigma performance level
Analyze Opportunity
• Analyze data to understand reasons for variation and identify potential root causes
• Determine process capability, throughput, cycle time
• Formulate, investigate, and verify root cause hypotheses.
Improve Performance
• Generate and Quantify potential solutions
• Evaluate and select final solution
• Verify and gain approval for final solution
Control Performance
• Develop ongoing process management plans
• Mistake-proof process
• Monitor and control critical process characteristics
• Develop out of control action plans

Fig. 9.3 The DMAIC process of quality development

space where the operation can be easily visualized. These approaches use the techniques of principal component analysis (PCA) or Partial Least Squares (PLS). Beside process performance monitoring, these tools can also be used for system identification (MacGregor and Kourti 1995), ensuring consistent production (Martin et al. 1996) and product design (Lakshminarayanan et al. 2000). Data analysis based formulation of new products was first reported by Moteki and Arai (Moteki and Arai 1986), who used PCA to analyze data from a polymer production. Jaeckle and MacGregor (1998) used PLS and principal component regression (PCR) to investigate the product design problem. Their methodology was illustrated using simulated data from a high-pressure tubular low-density polyethylene process. Borosy (1998) used artificial neural networks to analyze data from the rubber industry.

The large number of examples taken from polymer industry is not surprising. Formulated products (plastics, polymer composites) are generally produced from many ingredients, and high number of interactions between the components and the processing conditions has an effect on the final product quality. When these effects are detected, significant economic benefits can be realized. The major aims of monitoring plant performance are the reduction of off-specification production, the identification of important process disturbances and the early warning of process malfunctions or plant faults. Furthermore, when a reliable model is available that is able to estimate the quality of the product; it can be inverted to obtain the suitable operating conditions required for achieving the target product quality (Lakshminarayanan et al. 2000).

When we attempted to use standard data mining, KDD, and multivariate statistical tools for industrial problems such as extracting knowledge from large amount of data, we realized that production systems are typically ill-defined, difficult to model and they have large-scale solution spaces. In these cases, precise models are impractical, too expensive, or non-existent. Furthermore, the relevant available information is usually in the form of empirical prior knowledge and input-output data representing instances of the system's behaviour. Therefore, we need an approximate reasoning system capable of handling such imperfect information (Abonyi and Feil 2005). Computational Intelligence (CI) and Soft Computing (SC) are recently coined terms describing the use of many emerging computing disciplines. According to Zadeh (1994): "… in contrast to traditional, hard computing, soft computing is tolerant of imprecision, uncertainty, and partial truth." In this context Fuzzy Logic (FL), Probabilistic Reasoning (PR), Neural Networks (NNs), and Genetic Algorithms (GAs) are considered as main components of SC.

Most of the SC based models can be effectively used in data mining and lend themselves to transform into other traditional data mining or advanced SC-based model structures that allow information transfer between different models. For example, in Sethi (1990) a decision tree was mapped into a feed forward neural network. A variation of this method is given in Ivanova and Kubat (1995) where the decision tree was used only for the discretization of input domains. Another example is that as radial basis functions (RBF) are functionally equivalent to fuzzy inference systems (Jang and Sun 1993), tools developed for the identification of RBFs can also be used to design fuzzy models. The KDD process also includes the interpretation of the mined patterns. This step involves the visualization of the extracted patterns/models, or visualization of the data given the extracted models. Among the wide range of SC tools (Pal 1999), the Self-Organizing Map (SOM) is the most applicable for this purpose (Kohonen 1990). The main objective of this chapter is to propose an SOM based methodology that can be effectively used for the analysis of operational process data and product quality.

Nowadays, more and more articles deal with SOM-based data analysis (Astudillo and Oommen 2014; Poggy et al. 2013; Ghosh et al. 2014) that is a new, powerful software tool for the visualization of high-dimensional data. The SOM algorithm performs a topology preserving mapping from high dimensional space onto a two dimensional grid of neurons so that the relative distances between data points are preserved (Valova et al. 2013). The net roughly approximates the probability density function of the data and, thus, serves as a clustering tool

(Kohonen 1990). It also has the capability to generalize, i.e. the network can interpolate between previously encountered inputs. Since SOM is a special clustering tool that provides compact representation of the data distribution, it has been widely applied in the visualization of high-dimensional data (Kohonen 1990). The SOM facilitates visual understanding of processes so that several variables and their interactions may be inspected simultaneously. For instance, Kassalin used SOM to monitor the state of a power transformer and to indicate when the process was entering a non-desired state represented by a “forbidden” area on the map (Kassalin et al. 1992). Tryba and Goser (1991) applied the SOM in monitoring of a distillation process and discussed its use in chemical process control in general. Alander (1991) and Harris and Kohonen (1993) have used SOM in fault detection. Since the model is trained using measurement vectors describing normal operation only, a faulty situation can be detected by monitoring the quantization error (distance between the input vector and the best matching unit (BMU)), as large error indicates that the process is out of normal operation space. SOM can also be used for prediction, where SOM is used to partition the input space of piecewise linear models. This partitioning is obtained by the Voronoi diagram of the neurons (also called codebook) of SOM. The application of Voronoi diagrams of SOM has already been suggested in the context of time series prediction (Principe et al. 1998).

Based on the aforementioned beneficial properties of SOM, a new approach for process analysis and product quality estimation is proposed in this chapter. This approach is applied in an industrial polyethylene plant, where medium and high-density polyethylene (MDPE and HDPE) grades are manufactured in a low-pressure catalytic process, a slurry polymerization technology under license from Phillips Petroleum Company. The main properties of polymer products (Melt Index (MI) and density) are controlled by the reactor temperature, monomer, comonomer and chain-transfer agent concentration. The detailed application study demonstrates that SOM is very effective in the detection of typical operating conditions related to different products, and can be used to predict the product quality (MI and density) based on measured and calculated process variables.

The chapter is organized as follows. In Sect. 9.2, multivariate techniques for process monitoring and product quality estimation are overviewed. In Sect. 9.3, case studies are presented where the proposed methodologies are applied in real-life quality development problems of chemical industry. Finally, conclusions are given in Sect. 9.4.

9.2 Multivariate Techniques for Quality Development

Measurements on process variables $\boldsymbol{z}_k = \left[k_{k,1}, \ldots, z_{k,m}\right]^T$ such as temperatures, pressure, flow rates are available every second. Final product quality variables $\boldsymbol{y}_k = \left[y_{k,1}, \ldots, y_{k,n}\right]^T$, such as polymer molecular weights or melt index are available in much less frequent basis. All such data should be used to extract information in any effective scheme for monitoring and diagnosis operating performance. However, all of these variables are not independent of one another. Only a few underlying events are driving a process at any time, and all of these measurements are simply different reflections of these same underlying events. When the quality properties are not correlated, it is customary to build a model that relates z_k to each $y_{k,i}$ separately: $y_{k,i} = f_i(z_k)$. This approach is satisfactory in general if the model is just being used for calibration, inferential control or prediction. For monitoring purposes; however, since quality is a multivariate property, it is important to fit all the variables from the y-space in a single model in order to obtain a single low-dimensional monitoring space. Hence, the following multivariate models are introduced to model the joint distribution of the process and quality variables $\boldsymbol{x}_k = \left[x_{k,1}, \ldots, x_{k,l}\right]^T = \left[\boldsymbol{y}_k^T, z_k^T\right]^T$, where $l = n + m$.

9.2.1 Principal Component Analysis and Partial Least Squares

PCA and PLS are the most common algorithms used for the analysis of multivariate processes data (Jolliffe 2008). In the literature several papers deal with the application of these methods (Kano and Nakagawa 2008; Höskuldsson 1995; Godoy et al. 2014). Analysis of chemical and spectroscopic data mostly requires the utilization of these models. Results of the related research work are mostly published in Journal of Chemometrics (Kettaneh et al. 2003; Janné et al. 2001) and Chemometrics and Intelligent Laboratory Systems (Godoy et al. 2014; Nelson et al. 2006). Multivariate statistical methods can also be used in production systems to estimate unmeasured process and product quality variables (soft sensors) and fault detection. Chen et al. (1998) demonstrated how PLS and PCA are used for on-line quality improvement in two case studies: a binary distillation column and Tennessee Eastman process (Chen et al. 1998). Kresta (1992) showed how PLS and PCA are used to increase process operating performance in case of fluidized bed reactor and extractive distillation column (Kresta et al. 1992).

Fault detection and isolation algorithms detect outliers and isolate (root) causes of faults. PLS and PCA models can evaluate the consistency of multivariate data, characterize normal operation, and generate informative symptoms of deviations (Chiang et al. 2001; Wise and Gallagher 1996; Hu et al. 1995). It should be noted that these outliers may significantly reduce model accuracy when they are involved in the identification of PLS and PCA models. Therefore, data preprocessing and cleaning are important steps of model building (Wang and Srinivasan 2009; Fujiwara et al. 2012).

PLS and PCA are similar in that they are both factor analysis methods, and they both reduce the dimensionality of the variable space. This is done by representing the data matrix $(\boldsymbol{X})$ with a few orthogonal variables that explain most of the variance.

The main difference between PLS and PCA is that PLS can be referred to as a supervised technique that maximizes the covariance between the response $(\boldsymbol{Y})$ and input variables $(\boldsymbol{X})$ in as few factors as possible while PCA simply aims to maximize the covariance of $\boldsymbol{X}$ (Jolliffe 2008).

Mathematically, PCA reduces the data matrix using eigenvector decomposition of the covariance matrix of the data matrix. Essentially, the data matrix is broken down into principal components (PCs), represented by pairs of scores $(\boldsymbol{t})$ and loadings $(\boldsymbol{p})$ (Jolliffe 2008). The loading vectors are equivalent to the eigenvectors of the covariance matrix of $\boldsymbol{X}$, and the corresponding eigenvalues (λ) represent the variance of each corresponding PC. Suppose $\boldsymbol{X}$ is composed of n samples on q variables. The first PC is defined as $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{p}_1$ and explains the greatest amount of variance, while the second PC is defined as $\boldsymbol{t}_2 = \boldsymbol{X}\boldsymbol{p}_2$ having the next greatest amount of variance, and so on. Up to q PCs can be defined, but only the first few (M) are significant in explaining the main variability of the system (Jolliffe 2008). Selection of optimal number of PCs can be accomplished in various ways.

Partial Least Squares (PLS) regression combines principal component analysis and multivariate regression. PLS captures variance and correlates $\boldsymbol{X}$ and $\boldsymbol{Y}$ (Vinzi 2010). The first latent variable (LV) $\boldsymbol{t}_1 = \boldsymbol{X}\boldsymbol{w}_1$ is a linear combination of the $\boldsymbol{X}$ variables that maximizes the covariance of $\boldsymbol{X}$ and $\boldsymbol{Y}$, where $\boldsymbol{w}_1$ is the first eigenvector (called weight vector) of the covariance matrix. The columns of $\boldsymbol{X}$ are then regressed on t_1 to give a regression vector p_1. The original $\boldsymbol{X}$ matrix is then deflated as follows: $\boldsymbol{X}_2 = \boldsymbol{X} - \boldsymbol{t}_1\boldsymbol{p}_1^T$. $\boldsymbol{X}_2$ is the resultant data matrix after removing the element of the original data matrix $(\boldsymbol{X})$ that was most correlated with $\boldsymbol{Y}$. The second LV is then computed from $\boldsymbol{X}_2$, $\boldsymbol{t}_2 = \boldsymbol{X}\boldsymbol{w}_2$, where $\boldsymbol{w}_2$ is the first weight vector of the covariance matrix of $\boldsymbol{X}_2$ and $\boldsymbol{Y}$. These steps are repeated until q number of LVs is computed. As with PCA, the optimum number of LVs may be chosen via cross-validation methods.

PCA and PLS are widely applied tools of quality development. PCA can be used in monitoring of groundwater (Sánchez-Martos et al. 2001), essential oil (Ochocka et al. 1992), pig meat (Karlsson 1992), and soil quality (Garcia-Ruiz et al. 2008).

PLS is essentially a regression tool and may be used to relate process variables to product quality attributes. PCA can also be used as a regression tool in that the significant PCs may be used to generate a regression model that relates process variables to product quality attributes (when PCA is used in such a way, it is referred to as principal component regression—PCR) (Vinzi 2010). Application examples can be found from biotechnology (measure fruit and vegetable or vegetable oil quality) (Nicolai et al. 2007; Pereira et al. 2008), chemical industry (predict gasoline properties (Bao and Dai 2009), prediction of crude oil quality (Abbas et al. 2012), quality improvement of batch processes (Ge 2014), food industry (food quality improvement (Steenkamp and van Trijp 1996).

PLS can also be used for the visualization of the data. We apply the algorithm developed in Ergon (2004) for the two-dimensional visualization of the PLS model.

Two components that are informative for visualization may be obtained in several ways. One example is principal components of predictions (PCP), where in the scalar response case $\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{b}}$ normalization is used as one component, while residuals of $\boldsymbol{X}$ not contributing to $\boldsymbol{y}$ are suggested for use as the second component (Ergon 2004).

The basic idea behind the applied mapping is illustrated in Fig. 9.2. The estimator $\hat{\boldsymbol{b}}$ is found in the space spanned by loading weight vectors in $\widehat{\boldsymbol{W}} = [\hat{\boldsymbol{w}}_1, \hat{\boldsymbol{w}}_2, \ldots, \hat{\boldsymbol{w}}_A]$ i.e. it is a linear combination of these vectors. It is, however, also found in the plane defined by $\hat{\boldsymbol{w}}_1$ and a vector $\tilde{\boldsymbol{w}}_2$ orthogonal to $\hat{\boldsymbol{w}}_1$, which is a linear combination of the vectors $\hat{\boldsymbol{w}}_2, \hat{\boldsymbol{w}}_3, \ldots, \hat{\boldsymbol{w}}_A$.

The matrix $\widetilde{\boldsymbol{W}} = [\hat{\boldsymbol{w}}_1, \tilde{\boldsymbol{w}}_2]$ is thus the loading weight matrix in a two component PLS solution (2PLS) giving exactly the same estimator $\hat{\boldsymbol{b}}$ as the original solution using any number of components. What matters in the original PLS model is not the matrix $\widehat{\boldsymbol{W}}$ as such, but the space spanned by $\hat{\boldsymbol{w}}_1, \hat{\boldsymbol{w}}_2, \ldots, \hat{\boldsymbol{w}}_A$. In the 2PLS model, this represents the plane spanned by $\hat{\boldsymbol{w}}_1$ and $\tilde{\boldsymbol{w}}_2$ that is essential. Note that all samples in $\boldsymbol{X}$ (row vectors) in the original PLS model are projected onto the space spanned by $\hat{\boldsymbol{w}}_1, \hat{\boldsymbol{w}}_2, \ldots, \hat{\boldsymbol{w}}_A$.

Samples may thus be further projected onto the plane spanned by $\hat{\omega}_1$ and $\tilde{\omega}_2$, and form a single score plot containing all $\boldsymbol{y}$-relevant information. When for some reasons, for example, $\hat{\boldsymbol{w}}_2$ is more informative than $\hat{\boldsymbol{w}}_1$, a plane through $\hat{\boldsymbol{w}}_2$ and $\hat{\boldsymbol{b}}$ may be a better alternative. It will in any case result in a 2PLS model that gives the estimator $\hat{\boldsymbol{b}}$, as will in fact all planes through $\hat{\boldsymbol{b}}$ that are at the same time subspaces of the column space of $\widehat{\boldsymbol{W}}$ (Ergon 2004).

9.2.2 Self-organizing Map

Cluster analysis organizes data into groups according to similarities among them. In metric spaces, similarity is defined by means of distance based upon the length from a data vector to some prototypical object of the cluster. The prototypes are usually not known beforehand, and are sought by the clustering algorithm simultaneously with the partitioning of the data. In this chapter, the clustering of the operational data is considered. Hence, the data are the measured input and output process variables, parameters of the operating conditions, and laboratory measurements of the product quality. Each observation consists of l measured variables, grouped into an l-dimensional column vector $\boldsymbol{x}_i = \left[x_{i,1}, \ldots, x_{i,l}\right]^T$. A set of N observations is denoted by $\boldsymbol{X}$ and represented as a matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$. In pattern recognition terminology, the columns of $\boldsymbol{X}$ are called patterns or objects, the rows are called the features or attributes, and $\boldsymbol{X}$ is called the pattern matrix. The objective of clustering is to divide the data set $\boldsymbol{X}$ into c clusters.

The SOM algorithm is a kind of clustering algorithm which a performs a topology preserving mapping from high dimensional space onto map units so that relative distances between data points are preserved. The map units, or neurons, form usually a two dimensional regular lattice. Each neuron, i, of the SOM is represented by an l-dimensional weight, or model vector $\boldsymbol{m}_i = \left[m_{i,1}, \ldots, m_{i,l}\right]^T$. These weight vectors of the SOM form a codebook and can be considered as cluster prototypes. The neurons of the map are connected to adjacent neurons by a neighbourhood relation, which dictates the topology of the map. The number of neurons determines the granularity of the mapping, which affects the accuracy and the generalization capability of the SOM.

SOM is a vector quantizer, where the weights play the role of the codebook vectors. This means that each weight vector represents a local neighbourhood of the space, also called Voronoi cell. The response of a SOM to an input $\boldsymbol{x}_k = \left[x_{k,1}, \ldots, x_{k,l}\right]^T$ is determined by the reference vector (weight) $\boldsymbol{m}_{i^0}$ which produces the best match of the input

$$i_k^0 = arg(min_i \|\boldsymbol{m}_i - \boldsymbol{x}_k\|) \tag{9.1}$$

where i_k^0 represents the index of the Best Matching Unit (BMU) of the kth input.

During the iterative training of SOM, the SOM forms an elastic net that folds onto the “cloud” formed by the data. The net tends to approximate the probability density of the data; the codebook vectors tend to drift there where the data are dense, while there are only a few codebook vectors where the data are sparse.

The training of SOM can be accomplished generally with a competitive learning rule as

$$\boldsymbol{m}_i^{(t+1)} = \boldsymbol{m}_i^{(t)} + \eta \Lambda_{i_k^0,i}\left(\boldsymbol{x}_k - \boldsymbol{m}_i^{(t)}\right) \tag{9.2}$$

where $\Lambda_{i_k^0,i}$ is a spatial neighbourhood function and η is the learning rate, and the (t) upper index denotes the iteration step. Usually, the neighbourhood function is

$$\Lambda_{i_k^0,i} = \exp\left(-\frac{\left\|r_i - r_{i_k^0}\right\|^2}{2\sigma^{2,(t)}}\right) \tag{9.3}$$

where $\left\|r_i - r_{i_k^0}\right\|$ represents the Euclidean distance in the low dimensional output space between the ith vector and the winner neuron (BMU).

There are two phases during learning. First, the algorithm should cover the full input data space and establish neighbourhood relations that preserve the input data structure. This requires competition among the majority of the weights and a large learning rate such that the weights can orient themselves to preserve local relation-ships. Hence, in the first phase relatively large initial σ^2 is used. The second phase of learning is the convergence phase where the local detail of the input space is preserved. Hence the neighbourhood function should cover just one unit and the learning rate should also be small. In order to achieve these properties, both the neighbourhood function and the learning rate should be scheduled during learning (Kohonen 1990).

SOM is increasingly applied in quality development (Pölzlbauer 2004). Thanks to the robustness of the method SOM is applied water management (Kalteh et al. 2008; Juntunen et al. 2013) for soil and sediment quality estimation (Olawoyin et al. 2013), in pulp and paper processes (Alhoniemi et al. 1999), and in biotechnology (Mele and Crowley 2008).

In addition, SOM is capable of detection of faults. Since SOM is a gradient based iterative technique, it is less sensitive if outliers are in data sets. Since this technique is a mapping, the performance of whole procedure is not influenced by outliers because they grouped or they are on the edge of the map.

When a cell contains outliers the performance of the local model may decrease. Since this cell represents the edge of the normal operating region, outliers do not influence the global modelling performance. Hence, SOM is much less sensitive to outliers than PCA or PLS. Therefore, SOM is excellent for fault detection, because cells contain outliers and data related to malfunction of the process can be easily identified (Fustes et al. 2013; Munoz and Muruzábal 1998).

9.2.2.1 SOM for Piecewise Linear Regression

The goal of this section is to develop a data-driven algorithm for the identification of a model in the form of $\boldsymbol{y}_k = f(z_k)$, where z_k represents the model inputs (process variables) and $\boldsymbol{y}_k$ contains the product quality. In general, it may not be easy to find a global nonlinear model that is universally applicable to describe the relationships between the inputs and the outputs on the whole operating domain of the process. In that case, it would certainly be worthwhile to build local linear models for specific operating points of the process and combine these into a global model. This can be done by combining a number of local models, where each local model has a predefined operating region where the local model is valid. This results in the so-called operating regime based model. The applications and the possible identification of operating regime based modelling to the identification of dynamic systems are recent and rich (Murray-Smith 1997).

The operating regime based model is formulated as

$$\boldsymbol{y}_k^T = \sum_{i=1}^{s} \omega_i(z_k) \left[z_k^T, 1 \right] \boldsymbol{\Theta}_i \tag{9.4}$$

where $\omega_i(z_k)$ describes the operating regime of the ith local linear model de-fined by the $\boldsymbol{\Theta}_i$ parameter matrix (or vector if $\boldsymbol{y}_k^T$ is a scalar). The piecewise linear models are special case of operating regime-based models. If we denote the input space of the model by $T : z \in T \subset \Re^m$, the piecewise linear model consists of a set of operating ranges $T_1, T_2, \ldots, T_s$ which satisfy $T_1 \cup T_2 \cup \ldots \cup T_s = T$ and $T_j \cap T_i = \emptyset; \forall i \neq j$.

Hence, the model can be formulated as

$$\text{If}\, z_k \in T_i \text{ then } \boldsymbol{y}_k^T = \left[\mathbf{z}_k^T, 1\right]\boldsymbol{\Theta}_i \tag{9.5}$$

where $\boldsymbol{\Theta}_i$ denotes the parameter estimate vector used in the ith local model.

The identification of these models can be divided into two tasks: structure identification that generates the operating ranges and parameter identification of the local models. As the simultaneous combination of these steps results in complex nonlinear optimization problem, several heuristic, mainly iterative algorithms have been worked out for this purpose (Murray-Smith 1997).

When SOM is used to represent nonlinear systems, it is trained based on the N input-output data pairs arranged in the $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^T$ pattern matrix as $x_k = \left[y_k^T, z_k^T\right]^T$.

The SOM can be directly used for prediction of the output, $\boldsymbol{y}_k$ of the process given the input vector z_k. Regression is accomplished by searching for the BMU using the known vector components. As the output of the system is unknown, the BMU is determined as

$$i_k^0 = arg\left(min_i \left\|\boldsymbol{m}_i^* - z_k\right\|\right) \tag{9.6}$$

where $m_i^* = [m_{i,n+1}, \ldots, m_{i,l}]$.

The output of the model can be defined as the unknown component of the BMU, $\boldsymbol{y}_k^T = \boldsymbol{m}_{i^0}^+ = \left[m_{i^0,1}, \ldots, m_{i^0,n}\right]$, which results in a piecewise constant model.

The accuracy of this model can be increased by building local models for data in the Voronoi cells of the SOM,

$$\boldsymbol{y}_k^T = \left[z_k - \boldsymbol{m}_{i_k^0}^*\right]^T \boldsymbol{\Theta}_{i_k^0} + \left(\boldsymbol{m}_{i^0}^+\right)^T \tag{9.7}$$

or

$$\boldsymbol{y}_k^T = \left[z_k^T, 1\right]\boldsymbol{\Theta}_{i_k^0} \tag{9.8}$$

where the $\boldsymbol{\Theta}_{i_k^0}$ parameter matrix of the local model is calculated by least squares method based on the local data set on the operating regime T_i only, where T_i is the ith Voronoi cell of the Voronoi diagram of the codebook of the SOM, $M = \left\{\boldsymbol{m}_1^*, \ldots, \boldsymbol{m}_c^*\right\}$. $Vor(M)$ is defined as the subdivision of T into c cells $T_i, i = 1, \ldots, c$, with the property that a point z_k lies in the cell corresponding to the site $\boldsymbol{m}_{i_k^0}^*$ if and only if $i_k^0 = arg(min_i \|\boldsymbol{m}_i - \boldsymbol{x}_k\|)$. Thus, each cell of the diagram is the intersection of a number of half-planes.

When the process is nonlinear there is a need for local linear approximation of the operating regime of the system. Sliced Inverse Regression (SIR) and the related techniques are suitable for the extraction and characterization of local linear subspaces (Li 2012; Lue 2009; Kuentz and Saracco 2010). In this context these techniques are similar to SOM as SOM also defines local operating regimes that can be also considered Voronoi cells of SOM. As this section showed, these clusters can be used to build local linear models. In our case least squares regression is used to build local models. However, local models of the clusters can also be defined by local PCA (similar to SIR) or sub-PLS models. This approach also illustrates that local linear modelling and clustering can be effectively combined to get accurate and interpretable models (Kenesei and Abonyi 2013; Abonyi et al. 2002).

9.2.2.2 SOM for Classification of Product Grades and Operating Conditions

The SOM can be used for classification purposes by assigning a class for each codebook vector and deciding the class of a sample vector based on the class of its BMU. The rule-based classifier consists of rules that describe N_c number of classes, given n data points. The rule antecedent defines the operating region of the rule in the l-dimensional feature space and the rule consequent is a class label from the set $g_i = \{1, \ldots, N_c\}$.

$$\text{If}\, \boldsymbol{x}_k \in T_i \text{ then } \mathit{class}\,\text{is}\, g \tag{9.9}$$

The interpretability of classifier depends on the number of utilized features. For the selection of the most relevant features, we modify the Fischer interclass separability method which is based on statistical properties of labeled data. The importance of a feature is measured by leaving out a feature and calculating a cost function for the reduced model. The feature selection is made iteratively by leaving out the less needed feature.

9.3 Application Examples

In this chapter two examples are given to demonstrate the applicability of multivariate data-driven tools. In the first example, PLS is applied to visualize the production of a fuel mixing process and estimate the product quality. The second example is similar in the application point of view, SOM is applied to monitor product quality of a polymerization process.

9.3.1 Online NIR—PLS Example

Present research focuses on two tasks. Datasets collected at the Dune Refinery of MOL Ltd (Hungary) are analyzed. The first task is the development of a prediction model that can estimate product properties based on spectra taken by online NIR

Table 9.1 Effect of the number of latent variables to the performance of the model (correlation between the estimated and measured variables are shown)

Property	Latent dimensions					
	2	6	12	18	24	48
Density	0.776	0.988	0.993	0.993	0.993	0.989
T90	0.432	0.654	0.849	0.895	0.868	0.796
CFPP0	0.657	0.942	0.947	0.953	0.921	0.888
CFPP	0.516	0.755	0.769	0.728	0.703	0.610
Cloud Pt	0.668	0.924	0.950	0.958	0.955	0.943
Flash Pt	0.408	0.596	0.878	0.901	0.895	0.854

analyzers. The second task is the development a monitoring tool based on the visualization of the same spectra.

The prediction performance of the models is measured by the correlation coefficient defined as:

$$\boldsymbol{R}(i,j) = \frac{\boldsymbol{C}(i.j)}{\sqrt{\boldsymbol{C}(i,i)\boldsymbol{C}(j,j)}} \tag{9.10}$$

where $\boldsymbol{C}$ is the covariance matrix and it is calculated as $\boldsymbol{C} = \mathrm{cov}(\boldsymbol{y}, \widehat{\boldsymbol{y}})$. Table 9.1 shows that the number of the available samples, N, differs for each properties. Among the 651 spectra, only 560 were different and in most of the cases, only a fragment of the properties were measured. Firstly the effect of dimensionality of latent space of the PLS model was analyzed (from 2 to 48 dimensions). To perform an adequate comparison, leave-one-out and 10-fold cross validation technique was applied. As it is shown in this table, the accuracy of the model increases rapidly by increasing the dimensionality of the latent space from 2 to 6 dimensions; however, it reaches a maximum since when the complexity of the model is higher than the complexity of the modelled system.

In Sect. 9.2.2, a special method is presented that can map the PLS latent space into two dimensional space by orthogonal signal correction. This method is compared with Principal Component Analysis and Topological Near-Infrared Modeling (CRISP-DM 2000; Abonyi and Feil 2005) (TOPNIR) developed specifically to visualize NIRspectra and building topological prediction models with the help of resulted maps. As shown in Fig. 9.4, this technique is effective in visualization of high dimensional spectral space. This plot gives information about how summer and winter fuel samples are clustered.

9.3.2 Application in Polyethylene Production

To illustrate the proposed approach, the monitoring of a medium and high-density polyethylene (MDPE, HDPE) plant of the TVK Ltd. in Hungary is considered. HDPE is versatile plastic used for household goods, packaging, car parts and pipe,

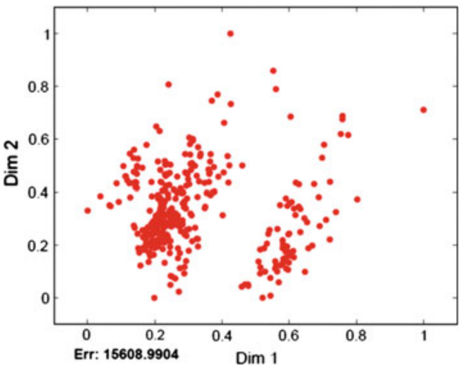


Fig. 9.4 Visualization of DS1 using PLS (CFPP0)

and TVK Ltd. is the largest Hungarian polymer production company (www.tvk.hu). A brief explanation of the Phillips license based low-pressure catalytic process is provided in the following section.

Figure 9.5 represents the Phillips Petroleum Co. suspension ethylene polymerization process. The polymer particles are suspended in an inert hydrocarbon. The melting point of high-density polyethylene is approximately 135 °C. Therefore, slurry polymerization takes place at a temperature below 135 °C; the polymer formed is in the solid state. The Phillips process takes place at a temperature between 85 and 110 °C. The catalyst and the inert solvent are introduced into the loop reactor where ethylene and an olefin (hexene) are circulating. The inert solvent

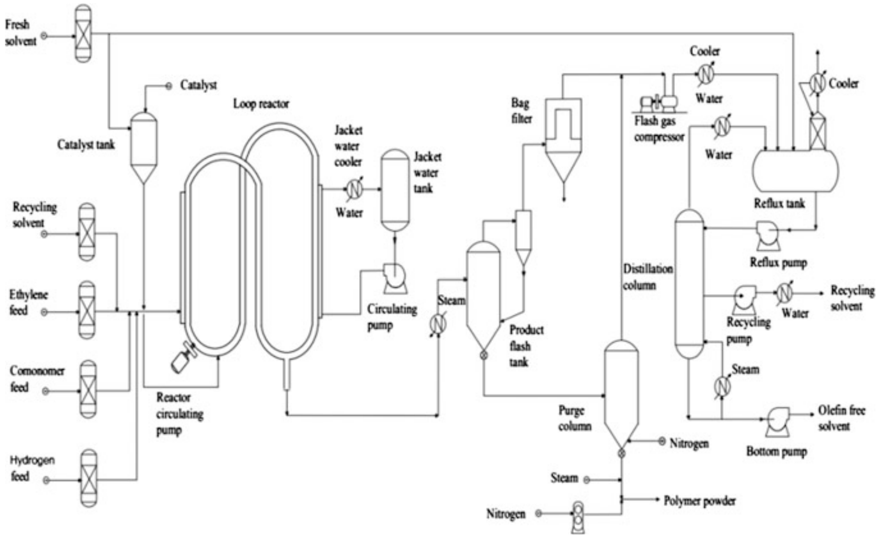


Fig. 9.5 Scheme of the Phillips loop reactor process (Nagy 1997)

(isobuthane) is used to dissipate heat as the reaction is highly exothermic. A cooling jacket is also used to dissipate heat. The reactor consists of a folded loop containing four long runs of pipe that are 1 m in diameter, connected by short horizontal lengths of 5 m. The slurry of HDPE and catalyst particles circulates through the loop at a velocity between 5 and 12 m/s. The reason for the high velocity is due to the fact that at lower velocities, the slurry will deposit on the walls of the reactor causing fouling. The concentration of polymer products in the slurry is 25–40 % by weight. Ethylene, olefin comonomer (if used), an inert solvent, and catalyst components are continuously charged into the reactor at a total pressure of 450 psig. The polymer is concentrated in settling legs to about 60–70 % by weight slurry and continuously removed. The solvent is recovered by hot flashing. The polymer is dried and pelletized. The conversion of ethylene to polyethylene is very high (95–98 %), eliminating ethylene recovery. The molecular weight of high-density polyethylene is controlled by the temperature of catalyst preparation (Nagy 1997). The main properties of polymer products (Melt Index (MI) and density) are controlled by the reactor temperature, monomer, comonomer and chain-transfer agent concentration.

9.3.2.1 Problem Description

An interesting problem with the process is that it is required to produce about ten product grades according to market demand. Hence, there is a clear need to minimize the time of changeover because off-specification product may be produced during transition. The difficulty of the problem comes from the fact that there are more than ten process variables to consider. Measurements are available in every 15 s on process variables $\mathbf{z}_k$, which are the $z_{k,1}$ reactor temperature (T), $z_{k,2}$ ethylene concentration in the loop reactor $(C2)$, $z_{k,3}$ hexene concentration $(C6)$, $z_{k,4}$ the ratio of the hexene and ethylene inlet flowrate $(C6/C2in)$, $z_{k,5}$ the flowrate of the isobutane solvent $(C4)$, $z_{k,6}$ the hydrogen concentration (H_2), $z_{k,7}$ the density of the slurry in the reactor (roz), $z_{k,8}$ polymer production intensity (PE), and $z_{k,9}$ the flowrate of the catalyzer (KAT). The product quality $\mathbf{y}_k$ is only determined later, in another process. The interval between the product samples is between half an hour and 5 h. The $y_{k,1}$ melt index (MI) and the $y_{k,2}$ density of the polymer (ro) are monitored by off-line laboratory analysis after drying and extrusion of the polymer that causes 1 h time-delay.

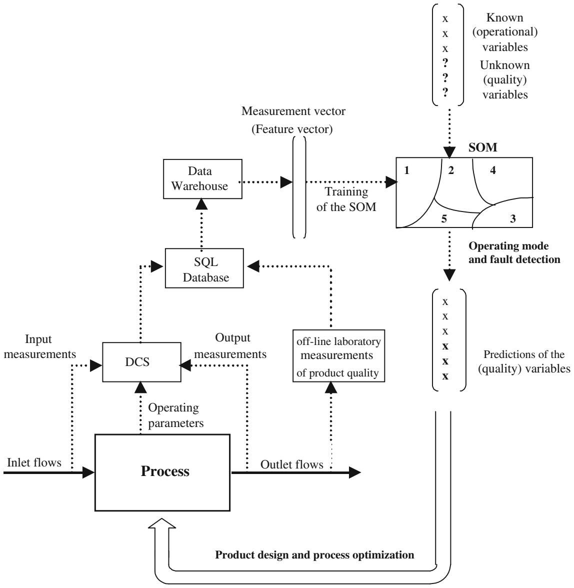
Since, it would be useful to know if the product is good before testing it, the monitoring of the process would help in the early detection of poor-quality product. There are other reasons why monitoring the process is advantageous. Only a few properties of the product are measured and sometimes these are not sufficient to entirely define the product quality. For example, if only rheological properties of polymer are measured (melt index), any variation in end-use application that arise due to variation of chemical structure (branching, composition, etc.) will not be captured by following only these product properties. In these cases, the process data may contain more information about events with special causes that may affect the product quality (Jeackle and MacGregor 1998).

9.3.2.2 SOM Based Historical Analysis of the Process

The modelling and monitoring of processes from data involves solving the problem of data gathering, preprocessing, model architecture selection, identification or adaptation and model validation. The process data analyzed in this chapter have been collected over 3 months of operation. The data have been extracted from the distributed control system (DCS) of the process. An SQL server has been installed to store and merge this data with the product quality database. According to the data warehousing methodology, the application relevant data have been extracted from this SQL database. As one of the objectives is to infer the values of product quality from process data obtained at different operating regions, a set of transition-free data is used that covers the whole range of specifications of the quality properties and the process variables over all the possible operating regions. The data were preprocessed by normalization performed on single variables. Scaling of variables is of special importance since the SOM algorithm uses Euclidean metric. In the current phase of the project, this data are processed by the modified version of the MATLAB SOM Toolbox (Vesanto et al. 2015). The whole methodology is illustrated in Fig. 9.6.

The SOM of the process has been applied to predict polymer properties from measured process variables and to interpret the behaviour of the process. The

Fig. 9.6 Scheme of the SOM based process analysis approach


x
x
x
?
?
?
Known (operational) variables
Unknown (quality) variables
Measurement vector (Feature vector)
SOM
Data Warehouse
Training of the SOM
1
2
4
5
3
Operating mode and fault detection
SQL Database
Input measurements
DCS
Output measurements
off-line laboratory measurements of product quality
x
x
x
x
x
x
Predictions of the (quality) variables
Operating parameters
Inlet flows
Process
Outlet flows
Product design and process optimization

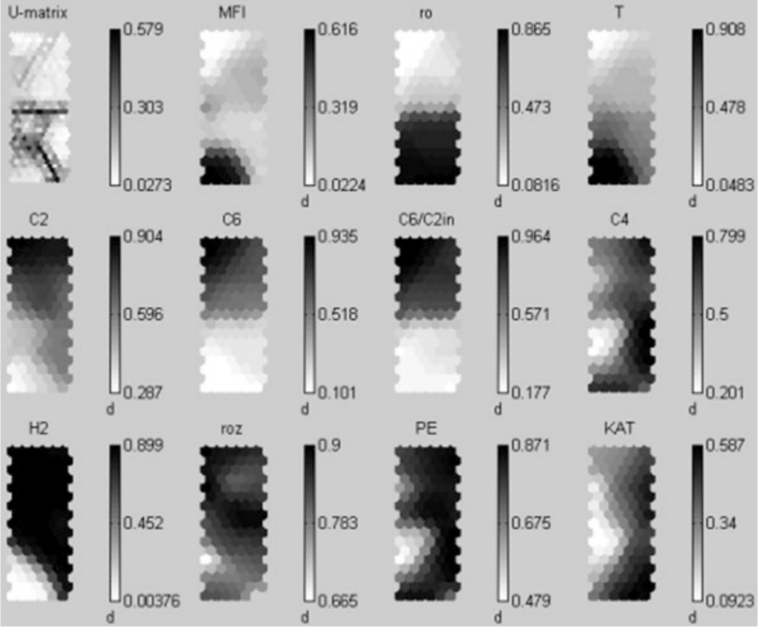


Fig. 9.7 Component planes of the polyethylene production map

constructed SOM (size 17 by 6 units) with eleven component planes is shown in Fig. 9.7. Based on the map the typical operating regions related to different product grades could be determined. Furthermore, the SOM is a good tool for hunting for correlation among the operating variables (Simula et al. 1999). For example, it can be easily seen that the melt index of the polymer (MFI) is highly correlated with reactor temperature (T).

Figure 9.8 shows the labels of the products and the distribution of the data marked by black hexagons with proportional size to the number of data in the operating regions of the clusters. This figure shows that the SOM is a useful tool for the visualization of multivariate data. A common procedure for reducing the dimensionality of the variable space is Principal Component Analysis (PCA) (MacGregor and Kourti 1995). For a comparison of the SOM with "standard" techniques, the historical data have been transformed into a two-dimensional space spanned by the first two principal components of the data. In Fig. 9.9, the grid of the transformed codebook of the SOM is shown to illustrate how the clusters approximate the density of the data. It is interesting to compare the SOM and the PCA model of the process (Fig. 9.11) as in both transformed spaces the regions of the different products are similar; the data points appear to cluster into four regions which corresponded to different product grades and operating conditions.

Since the distance preserving mapping property of the SOM, products that are close to each other on the map are similar. In the discrete two-dimensional output space of the SOM, the trajectory of the production can be effectively visualized by

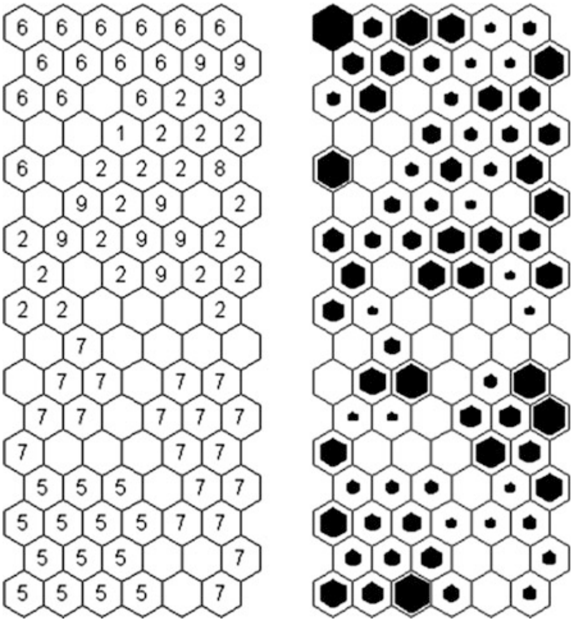


Fig. 9.8 Product labels (numbers) and distribution of the data

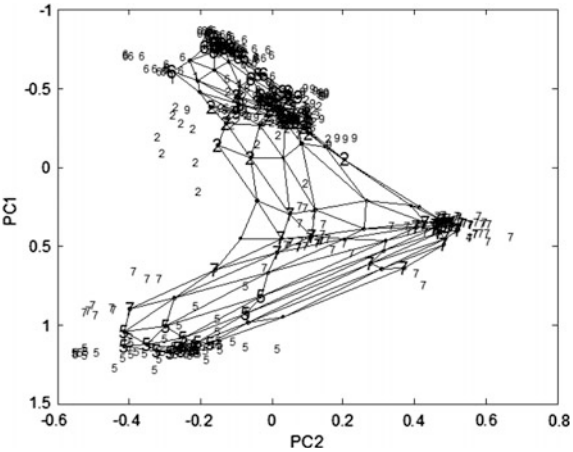


Fig. 9.9 PCA scores plots for three months of operation. The grid of the transformed codebook of the SOM is also shown to illustrate how the clusters approximate the density of the data

plotting the trajectory of the BMUs (Principe et al. 1998), which is especially useful in process monitoring and fault detection. Hence, the map can be effectively used for scheduling the different products by designing the trajectory of the production on the map of the products. An example for a grade transition is shown in Fig. 9.10, where the production of Product 6 is followed by the production of Product 7. The multivariate historical data of this transition depicted in Fig. 9.10 can be easily visualized by the PCA and the SOM model of the process as shown in Fig. 9.11.

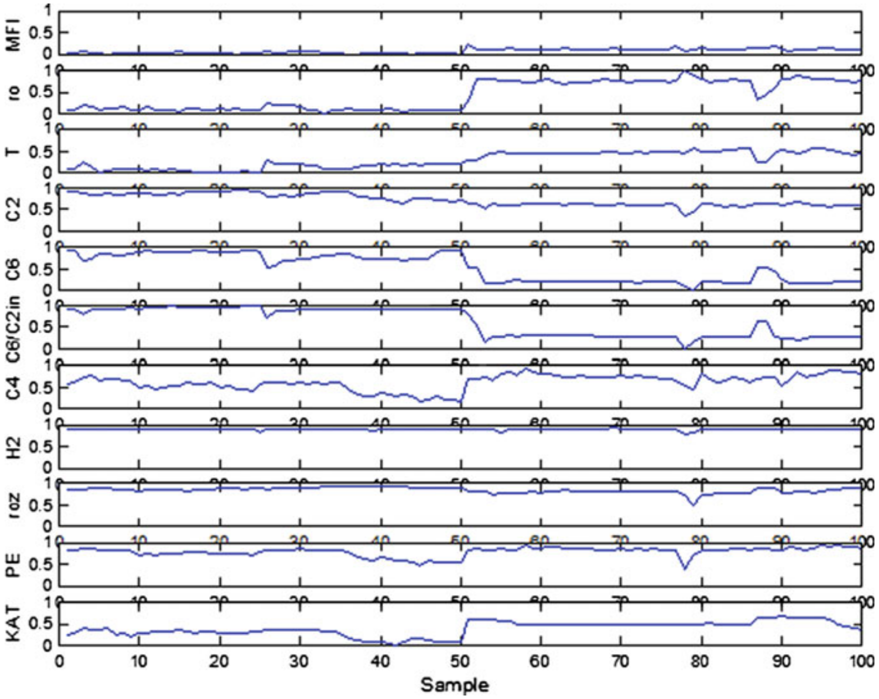


Fig. 9.10 Example of grade transition

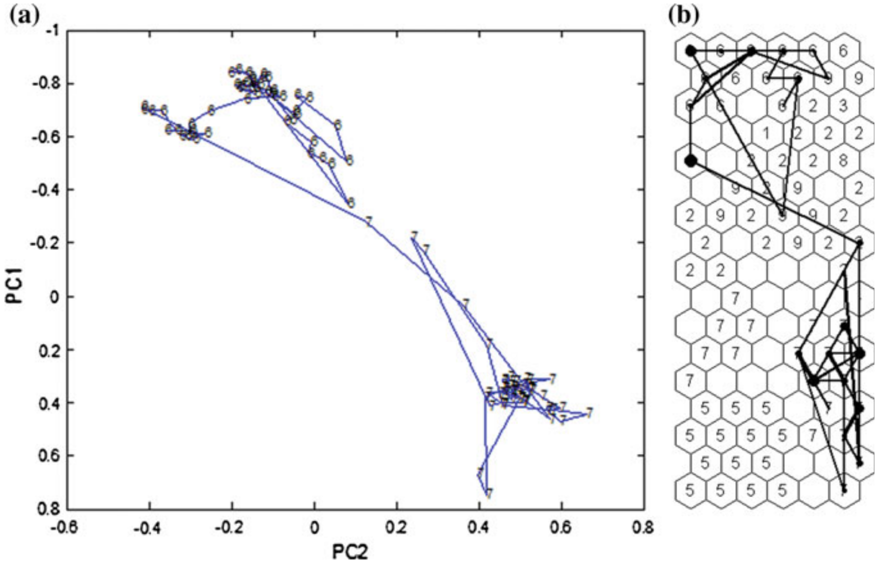


Fig. 9.11 Grade transition shown in Fig. 9.10 mapped into the two dimensional space by PCA (**a**) and SOM (**b**)

The previous example has shown that the SOM results in a good representation of the operating regions of the products. Hence, it can be also used for classification. When the whole SOM is used as a rule-based classifier system with rules like

$$\text{If } x_k \in T_i \text{ then } class \text{ is } g_i \tag{9.11}$$

it gives 8 % classification error. This can be considered as a good result taking into account that the data is quite noisy and not too much effort was put to select the training data related to normal operating conditions.

9.3.2.3 SOM Based Product Quality Estimation

The SOM has been also used to estimate the product quality variables. Figure 9.12 shows the estimation error of the linear model and the SOM presented in the previous section. Although in this case the SOM is used as a piecewise constant model, it gives better results than the linear model. The performances of the models have been measured by the Root Mean Square Error (RMSE) of the models. In Table 9.2, it can be seen that the SOM is more accurate than the linear model. This is not surprising since the linear model has only two times ten (20) parameters and cannot capture the nonlinearity of the process. The good performance of the BMU-based piecewise constant model shows that the SOM gives a good approximation of the density of the data, hence it can be considered as a good

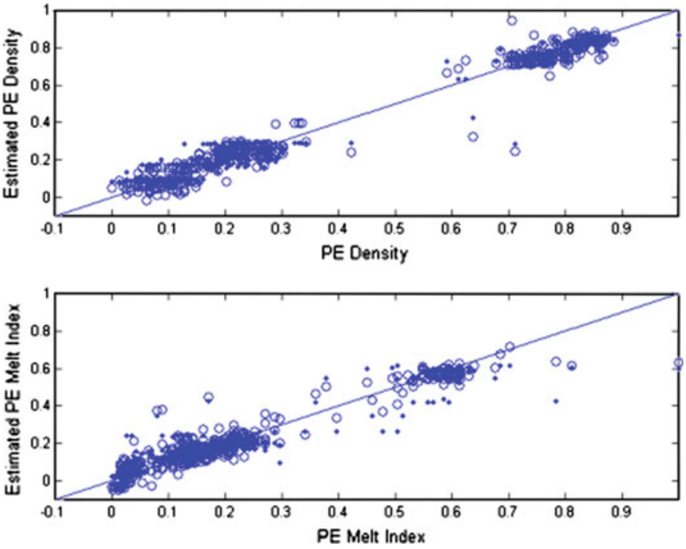


Fig. 9.12 Estimation performance of the SOM based piecewise constant model (.) and a multivariate linear model (o)

Table 9.2 Root Mean Square Errors (RMSE) achieved by different models

	Density (ro)	Melt index (MFI)
Linear	0.0355	0.0372
SOM pw constant (BMU)	0.0330	0.0341
Linear (4 variables)	0.0368	0.0387
SOM pw linear (4 variables)	0.0251	0.0312

Table 9.3 Relative importance of process variables obtained by OLS

Importance	Density (ro)	Melt index (MFI)
1	'C6/C2in'	'T'
2	'C4'	'C6/C2in'
3	'H2'	'C4'
4	'roz'	'H2'
5	'PE'	'C2'
6	'KAT'	'roz'
7	'C6'	'KAT'
8	'C2'	'C6'
9	'T'	'PE'

nonparametric model. Because of the large number of the clusters (size 17 by 6 units), the Voronoi cell based multiple linear model approach cannot be used. The reason is that the identification of the most local models become badly conditioned due to the small number of data related to the operating region of the models. Hence, in this chapter an approach based on the reduction of the original SOM is introduced, where for the regression purpose a smaller SOM is identified.

This task requires the selection of the most important variables having effect to the product quality variables. This can be done by analyzing SOM of the process (Fig. 9.7) to detect similarities between the component planes that shows the correlation of the variables. Another possible approach is to use Orthogonal Least Squares (OLS) techniques for ordering of the process variables. The ordering obtained is shown in Table 9.3. This ordering can then be easily used to select a subset of the inputs in a forward-regression manner. It is interesting to see that ordering of the OLS gives similar results that we can obtain from the visual inspection of the SOM of the process. Based on this model reduction approach, two independent SOMs with 24 clusters on four process variables (the first four shown in Table 9.2.) have been identified to estimate the density and the melt index of the product. As shown in Table 9.3, these compact models give good estimations of the quality variables.

9.4 Conclusion

Quality development intensively applies process data. The iterative data mining methodology effectively supports the PDCA cycle based quality development. Since several process variables have to be monitored and unknown functional relationships among process and quality variables have to be explored, multivariate statistical techniques are the most widely applied tools. Beside the classical principal component analysis (PCA) and partial least squares regression models (PLS), we applied soft computing based tools to handle uncertainty and nonlinearity and complexity of the problem.

We demonstrate that PLS based model is able to simultaneously predict unmeasured material properties and monitor the state of a complex production process. Process monitoring is realized in orthogonal two dimensional plots. These plots can also be used for the effective identification of outliers.

Self-Organizing Map (SOM) is a soft-computing based approach and it is used for the extraction of knowledge from the historical data of production. Since SOM provides a compact representation of the data distribution, the typical operating conditions of the process are efficiently detected. It has been shown that efficient process monitoring can be performed in the two-dimensional projection of the process variables. For the estimation of the product quality variables multiple local linear models are introduced, where the operating regimes of the local linear models are obtained by the Voronoi diagram of the prototype vectors of the SOM. The important process variables having effect to the product quality have been selected by orthogonal least squares method. The approach has been demonstrated by means of the analysis of a polyethylene production plant. The results show that the SOM is very effective in the detection of typical operating conditions related to different product grades and can be used to predict the product quality (melt index and density) based on the process variables measured. The proposed method is attractive in comparison with other advanced process monitoring schemes such as Principal Component Analysis.

The interested reader might want to know under what conditions these methods can be employed and what kind of diagnostic tests are available. Some books that are dealing with only one technique in detail are suggested for them: Handbook of Partial Least Squares (Vinzi et al. 2010), Principal Component Analysis (Jolliffe 2008), Introduction to Statistical Quality Control (Montgomery 2009) and Self-Organizing Maps (Kohonen 2001).

More technical details and illustrative examples and MATLAB program codes related to the application of intelligent tools for fault detection and quality estimation can be found at the website of the authors: www.abonyilab.com.

Acknowledgments The work was supported by the frames of TÁMOP-4.2.2.C-11/1/KONV-2012-0004—National Research Center for Development and Market Introduction of Advanced Information and Communication Technologies and TÁMOP 4.2.4. A/2-11- 1-2012-0001 "National Excellence Program—Elaborating and operating an inland student and researcher personal support system". These projects were subsidized by the European Union and co-financed by the European Social Fund.

References

Abbas, O., et al.: PLS regression on spectroscopic data for the prediction of crude oil quality: API gravity and aliphatic/aromatic ratio. Fuel **98**, 5–14 (2012)

Abonyi, J., Feil, B.: Computational intelligence in data mining. Informatica **29**, 3–12 (2005)

Abonyi, J., Babuska, R., Szeifert, F.: Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models. IEEE Trans. Syst. Man Cybern. B Cybern. **32**(5), 612–621 (2002)

Alander, J.T., et al.: Process error detection using self-organizing feature maps, Artif. Neural Netw. **2**, 1229–1232 (1991)

Alhoniemi, E., et al.: Process monitoring and modeling using the self-organizing map. Integr. Comput. Aided Eng. **6**, 3–14 (1999)

Astudillo, C.A., Oommen, B.J.: Self-organizing maps whose topologies can be learned with adaptive binary search trees using conditional rotations. Pattern Recogn. **47**, 96–113 (2014)

Bao, X., Dai, L.: Partial least squares with outlier detection in spectral analysis: a tool to predict gasoline properties. Fuel **88**(7), 1216–1222 (2009)

Borosy, A.P.: Quantitative composition-property modelling of rubber mixtures by utilizing artificial neural networks. Chemom. Intell. Lab. Syst. **47**, 227–238 (1998)

Chen, G., McAvoy, T.J., Piovoso, M.J.: A multivariate statistical controller for on-line quality improvement. J. Process Control **8**(2), 139–149 (1998)

Chiang, L.H., Russel, E.L., Braatz, R.D.: Fault Detection and Diagnosis in Industrial Systems. Springer, London (2001)

CRISP-DM Cross Industry Standard Process for Data Mining (2000). http://en.wikipedia.org/wiki/Cross_Industry_Standard_Process_for_Data_Mining

Ergon, R.: Informative PLS score-loading plots for process understanding and monitoring. J. Process Control **14**, 889–897 (2004)

Fujiwara, K., Sawada, H., Kano, M.: Input variable selection for PLS modeling using nearest correlation spectral clustering. Chemometr. Intell. Lab. Syst. **118**, 109–119 (2012)

Fustes, D., et al.: SOM ensemble for unsupervised outlier analysis. Application to outlier identification in the Gaia astronomical survey. Expert Syst. Appl. **40**(5), 1530–1541 (2013)

Garcia-Ruiz, R., et al.: Suitability of enzyme activities for the monitoring of soil quality improvement in organic agricultural systems. Soil Biol. Biochem. **40**(9), 2137–2145 (2008)

Ge, Z.: Two-level PLS model for quality prediction of multiphase batch processes. Chemometr. Intell. Lab. Syst. **130**, 29–36 (2014)

Ghosh, S., Roy, M., Ghosh, A.: Semi-supervised change detection using modified self-organizing feature map neural network. Appl. Soft Comput. **15**, 1–20 (2014)

Godoy, J.L., Vega, J.R., Marchetti, J.L.: Relationships between PCA and PLS-regression. Chemom. Intell. Lab. Syst. **130**, 182–191 (2014)

Harris, T., Kohonen, A.: S.O.M. based, machine health monitoring systems which enables diagnosis of faults not seen in the training set. Proc. Int. Conf. Neural Netw. (IJCNN'93) Nagoya, Japan **1**, 947–950 (1993)

Höskuldsson, A.: A combined theory for PCA and PLS. J. Chemom. **9**(2), 91–123 (1995). doi:10.1002/cem.1180090203

Hu, W., Storer, R., Georgakis, C.: Disturbance detection and isolation by dynamic principal component analysis. Chemometr. Intell. Lab. Syst. **30**(1), 179–196 (1995)

Ivanova, I., Kubat, M.: Initialization of neural networks by means of decision trees. Knowl. Based Syst. **8**, 333–344 (1995)

Jang, J.-S.R., Sun, C.T.: Functional equivalence between radial basis function networks and fuzzy inference systems. IEEE Trans. Neural Netw. **4**, 156–159 (1993)

Janné, K., et al.: Hierarchical principal component analysis (PCA) and projection to latent structure (PLS) technique on spectroscopic data as a data pretreatment for calibration. J. Chemom. **15**(4), 203–213 (2001). doi:10.1002/cem.677

Jeackle, C., MacGregor, J.: Product design through multivariate statistical analysis of process data. Am. Inst. Chem. Eng. J. **44**(5), 1105–1118 (1998)

Jolliffe, I.T.: Principal Component Analysis, 2nd edn. Springer Series in Statistics (2008)

Juntunen, P., et al.: Cluster analysis by self-organizing maps: an application to the modelling of water quality in a treatment process. Appl. Soft Comput. **13**(7), 3191–3196 (2013)

Kalteh, A.M., Hjorth, P., Berndtsson, R.: Review of the self-organizing map (SOM) approach in water resources: analysis, modelling and application. Environ. Model Softw. **23**(7), 835–845 (2008)

Kano, M., Nakagawa, Y.: Data-based process monitoring, process control, and quality improvement: recent developments and applications in steel industry. Comput. Chem. Eng. **32**(1–2), 12–24 (2008)

Karlsson, A.: The use of principal component analysis (PCA) for evaluating results from pig meat quality measurements. Meat Sci. **31**(4), 423–433 (1992)

Kassalin, M., Kangas, J., Simula, O.: Process state monitoring using self-organizing maps. Artif. Neural Netw. **2**, 1531–1534 (1992)

Kenesei, T., Abonyi, J.: Hinging hyperplane based regression tree identified by fuzzy clustering and its application. Appl. Soft Comput. J. **13**(2), 782–792 (2013)

Kettaneh, N., Berglund, A., Wold, S.: PCA and PLS with very large data sets. Comput. Stat. Data Anal. **48**, 69–85 (2003)

Kohonen, T.: The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)

Kohonen, T.: Self-Organizing Maps, 3rd edn. Springer Series in Information Sciences (2001)

Kresta, J.V.: The application of partial least squares to problems in chemical engineering, PhD Theis, McMaster University (1992).http://hdl.handle.net/11375/8576

Kresta, J.V., Macgregor, F.F., Marlin, T.E.: Multivariate statistical monitoring of process operating performance. Can. J. Chem. Eng. **69**(1), 35–47 (1992). doi:10.1002/cjce.5450690105

Kuentz, V., Saracco, J.: Cluster-based sliced inverse regression. J. Korean Stat. Soc. **39**, 251–267 (2010)

Lakshminarayanan, S., et al.: New product design via analysis of historical databases. Comput. Chem. Eng. **24**, 671–676 (2000)

Li, K.-C.: Sliced inverse regression for dimension reduction (2012). www.jstor.org/stable/2290563 . Accessed 19 Dec 2013

Lue, H.-H.: Sliced inverse regression for multivariate response regression. J. Stat. Plan. Inference **139**, 2656–2664 (2009)

MacGregor, J.F., Kourti, T.: Statistical process control of multivariate processes. Control Eng. Pract. **3**(3), 403–414 (1995)

Martin, E.B., et al.: Batch process monitoring for consistent production. Comput. Chem. Eng. **20**, S599–S605 (1996)

Mele, P.M., Crowley, D.E.: Application of self-organizing maps for assessing soil biological quality. Agric. Ecosyst. Environ. **126**(3–4), 139–152 (2008)

Montgomery D.C.: Introduction to Statistical Quality Control, John Wiley, New York (2009)

Moteki, Y., Arai, Y.: Operation planning and quality design of a polymer process. In: IFAC DYCORD, pp. 159–165 (1986)

Munoz, A., Muruzábal, J.: Self-organizing maps for outlier detection. Neurocomputing **18**, 33–60 (1998)

Murray-Smith, R., Johansen, T.A.: Multiple Model Approaches to Nonlinear Modeling and Control. Taylor & Francis, London (1997)

Nagy, G.: The polyethylene, Magyar Kémikusok Lapja (MKL). Hungary **52**(5), 233–242 (1997)

Nelson, P.R.C., MacGregor, J.F., Taylor, P.A.: The impact of missing measurements on PCA and PLS prediction and monitoring applications. Chemometr. Intell. Lab. Syst. **80**(1), 1–12 (2006)

Nicolai, B.M., et al.: Nondestructive measurement of fruit and vegetable quality by means of NIR spectroscopy: a review. Postharvest Biol. Technol. **46**(2), 99–118 (2007)

Ochocka, R.J., Wesolowski, M., Lamparczyk, H.: Thermoanalysis supported by principal component analysis (PCA) in quality assessment of essential oil samples. Thermochim. Acta **210**, 151–162 (1992)

Olawoyin, R., et al.: Application of artificial neural network (ANN)–self-organizing map (SOM) for the categorization of water, soil and sediment quality in petrochemical regions. Expert Syst. Appl. **40**(9), 3634–3648 (2013)

Pal, N.R.: Soft computing for feature analysis. Fuzzy Sets Syst. **103**, 201–221 (1999)

Pereira, A.F.C., et al.: NIR spectrometric determination of quality parameters in vegetable oils using iPLS and variable selection. Food Res. Int. **41**(4), 341–348 (2008)

Poggy, G., Cozzolino, D., Verdoliva, L.: Self-organizing maps for the design of multiple description vector quantizers. Neurocomputing **122**, 298–309 (2013)

Pölzlbauer, G.: Survey and comparison of quality measures for self-organizing maps. In: Fifth Workshop on Data Analysis (WDA) (2004). www.ifs.tuwien.ac.at/~poelzlbauer/publications/Poe04WDA.pdf. Accessed 17 Dec 2013

Principe, J.C., Wang, L., Motter, M.A.: Local dynamic modeling with self-organizing maps and applications to nonlinear system identification and control. Proc. IEEE **86**(11), 2241–2258 (1998)

Sánchez-Martos, F., Jiménez-Espinosa, R., Pulido-Bosch, A.: Mapping groundwater quality variables using PCA and geostatistics: a case study of Bajo Andarax, southeastern Spain. Hydrol. Sci. J.-des Sciences Hydrologiques **46**(2), 227–242 (2001)

Sethi, L.K.: Entropy nets: from decision trees to neural networks. Proc. IEEE **78**, 1605–1613 (1990)

Simula, O., et al.: Analysis and modeling of complex systems using the self-organizing map. In Neuro-Fuzzy Techniques for Intelligent Information Systems, pp. 3–22. Springer, New York (1999)

Steenkamp, J.B.E.M., van Trijp, H.C.M.: Quality guidance: a consumer-based approach to food quality improvement using partial least squares. Eur. Rev. Agric. Econ. **23**(2), 195–215 (1996). doi:10.1093/erae/23.2.195

Tryba, V., Goser, K.: Self-organizing feature maps for process control in chemistry. Artif. Neural Netw. 847–852 (1991)

Valova, I., et al.: Initialization Issues in Self-organizing Maps. Procedia Comput. Sci. **20**, 52–57 (2013)

Vinzi, V., et al.: Handbook of Partial Least Squares. In: Springer Handbooks of Computational Statistics (2010)

Wang, D., Srinivasan, R.: Eliminating the effect of multivariate outliers in pls-based models for inferring process quality. Comput. Aided Chem. Eng. **26**, 755–760 (2009)

Wang X.Z.: Data Mining and Knowledge Discovery for Process Monitoring and Control. Springer, New York (1999)

Wise, B.M., Gallagher, N.B.: The process chemometrics approach to process monitoring and fault detection. Journal of Process Control **6**(6), 329–348 (1996). doi:http://dx.doi.org/10.1016/0959-1524(96)00009-1

Vesanto, J., Himberg, J., Alhoniemi, E., Parhankangas, J.: SOM Tooolbox for MATLAB (2015). The Toolbox can be downloaded for free from http://www.cis.hut.fi/projects/somtoolbox

Yamashita, Y.: Supervised learning for the analysis of the process operational data. Comput. Chem. Eng. **24**, 471–474 (2000)

Zadeh, L.A.: Soft computing and fuzzy logic. Software, IEEE **11**(6), 48–56 (1994)

Zhang, J., Martin, E.B., Morris, A.J.: Process monitoring using non-linear statistical techniques. Chem. Eng. J. **67**, 181–189 (1997)