# DOKTORI (PhD) ÉRTEKEZÉS

Kulcsár Tibor

Pannon Egyetem
2016

**Adatbányászati és gépi tanulási algoritmusok**
**szoftver szenzorok fejlesztésére**

Értekezés doktori (PhD) fokozat elnyerése érdekében
a **Pannon Egyetem Vegyészmérnöki- és Anyagtudományok Doktori Iskola**
Doktori Iskolájához tartozóan

Írta:

Kulcsár Tibor

Konzulens: Prof. Dr. habil. Abonyi János

Elfogadásra javaslom igen / nem .........................................
(aláírás)

A jelölt a doktori szigorlaton ................. %-ot ért el.

Az értekezést bírálóként elfogadásra javaslom:

Bíráló neve: ......................................... igen / nem .........................................
(aláírás)

Bíráló neve: ......................................... igen / nem .........................................
(aláírás)

A jelölt az értekezés nyilvános vitáján ................. %-ot ért el.

Veszprém, .........................................
a Bíráló Bizottság elnöke

A doktori (PhD) oklevél minősítése ............................

.........................................
az EDT elnöke

Pannon Egyetem

Doktori (PhD) Értekezés

# Adatbányászati és gépi tanulási algoritmusok szoftver szenzorok fejlesztésére

DOI:10.18136/PE.2016.636

*Szerző:*
Kulcsár Tibor

*Konzulens:*
Prof. Dr. habil. Abonyi János

*Értekezés doktori (PhD) fokozat elnyerése érdekében*
*a* ***Pannon Egyetem***

Vegyészmérnöki- és Anyagtudományok Doktori Iskola
*Doktori Iskolájához tartozóan*

Folyamatmérnöki Intézeti Tanszék

Pannon Egyetem
2016

University of Pannonia

Doctoral (PhD)Thesis

# Data mining and machine learning algorithms for soft sensor development

*Author:*
Tibor Kulcsár

*Supervisor:*
Prof. Dr. habil. János Abonyi

*A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy*

*in the*

Doctoral School in Chemical Engineering and Material Sciences
*of University of Pannonia*

Department of Process Engineering

University of Pannonia
2016

*“Always listen to experts. They’ll tell you what can’t be done, and why. Then do it!"*

Robert A. Heinlein

PANNON EGYETEM

# *Kivonat*

Mérnöki Kar
Folyamatmérnöki Intézeti Tanszék

Philosophiæ Doctor

**Adatbányászati és gépi tanulási algoritmusok szoftver szenzorok fejlesztésére**

írta: Kulcsár Tibor

A szoftver szenzorok költséghatékony alternatívát kínálnak a gyakran magas beruházási és jelentős karbantartási igényű hagyományos műszerezéssel szemben, így egyre inkább teret hódítanak a folyamatmérnöki gyakorlatban. A közvetett mérésen alapuló szoftveres megoldások fejlesztése a legtöbb esetben nem igényli a technológia módosítását, "csupán" a tulajdonságbecslést végző modell alkotása és identifikálása a kulcsfeladat. Az ismertetésre kerülő módszerek a rendelkezésre álló adatok elemzésével szolgáltatnak információt a technológia működéséről, a mért és a becsült változók közti kapcsolatokról, segítve a szoftver szenzorok alkotásához szükséges modellezési tevékenységet.

A dolgozat első harmada nemparametrikus regressziós feladatok hátteréül szolgáló adatok megjelenítésére alkalmas megoldásokat mutat be. A megjelenítés alapjául szolgáló bináris fákkal reprezentált egyenletek optimális struktúrájának meghatározására genetikus algoritmust alkalmazunk. Az optimált dimenzió-csökkentő leképezések nem csupán spektrális anyagjellemzőket becslő regressziós modellek fejlesztésére, hanem osztályozási problémák megoldására is alkalmazhatók. A következő fejezet parametrikus modellek fejlesztése céljából folyamatváltozók rangsorolására és kiválasztására alkalmas eszközök fejlesztését tárgyalja. Végül olyan idősor elemzési technikákat ismertetünk, melyekkel az előbbi két részben említett modellezési feladatok számára a célnak megfelelő és szűrt adatok választhatók ki.

A fejlesztett algoritmusok alkalmazhatóságát online NIR elemzők és energia monitoring rendszerek szoftver szenzorainak fejlesztéséhez kötődő esettanulmányok igazolják.

UNIVERSITY OF PANNONIA

# *Abstract*

Faculty of Engineering
Department of Process Engineering

Doctor of Philosophy

**Data mining and machine learning algorithms for soft sensor development**

by Tibor KULCSÁR

Soft sensors offer a cost efficient alternative to online analysers, so they become more and more prevalent in the process industry. The development and maintenance of software based sensors do not require modifications in the technology; the most critical task is only the identification of the model used for prediction. To support the monitoring and data-driven identification of models used in software sensors we developed tools that can be used to explore the hidden structure of the process data.

The first part of the thesis presents a genetic programming based methodology that can generate dimension reduction mappings to visualise the operation of online NIR analysers used for nonparametric regression model based product property estimation.

The next chapter presents feature transformation and selection methods to support parametric model identification. Finally, techniques for time series analysis are proposed to extract data relevant to model identification.

The applicability of the algorithms is proved by case studies related to online NIR analyser based product quality estimation and energy monitoring.

# *Acknowledgements*

Undertaking this Ph.D. has been a truly life-changing experience for me, and it would not have been possible to do without the support and guidance that I received from many people.

I would like first to say a big thank you to my supervisor Prof. Dr. habil. Janos Abonyi for all the support and encouragement he gave me during the time that I spent at the University. Without his guidance and steady feedback, the results presented in this Thesis would not have been realised.

My thanks also go out to the support I received from the collaborative work I undertook with colleagues of the Danube Refinery of MOL Ltd. Finally, I say many thanks to my former colleagues in the Department of Process Engineering, they helped me a lot during my Ph.D. studies and also since I left the University.

Last but not the least, I would like to thank my family: my parents, my gradparents and to my sister for supporting me spiritually throughout writing this thesis and my life in general.

*This thesis is dedicated*

*to the people who kept me going when I wanted to give up*

*and to You as a reader.*

# Contents

# Chapter 1

# Introduction

## 1.1 Motivation

Advanced process control systems should predict of product properties from operating conditions; optimise process variables to improve product quality and detect faults or malfunctions for preventing undesirable operation [42]. These functionalities require timely and accurate information about process variables characterising and influencing product quality. The control of measured process values (e.g. temperature, pressure, flow rate) does not always ensure that product properties (e.g. density, cloud point, flash point) will be in desired ranges. In chemical and oil industry some of these properties cannot be measured online (e.g. cetane index, aromatic field, sulphur content) or the frequency of the measurements is lower than required for real time control (see Fig. 1.1). On-line analysers have faster response time (1-4 minutes) [25], but due to their high instrumentation and maintenance costs and low reliability there is a need for an easily implementable, maintainable and robust alternative.

Soft sensors are frequently used to estimate difficult-to-measure process variables. It should be noted that soft sensors are not identical to smart sensors. Smart sensors are sensors that include a microprocessor which conditions the signals before its transmission. These devices are the indispensable enablers of the Internet of Things (IoT) solutions. Smart sensors are particularly useful because they can keep track log data, identify items, locate them and determine their environmental conditions, which information can be used as triggers for alarm and process management.

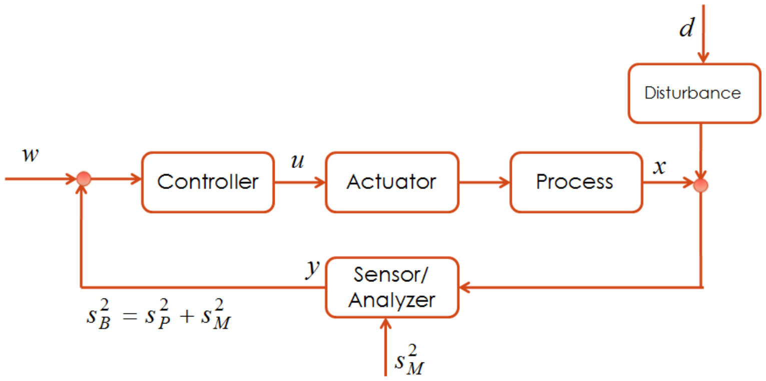

Figure 1.1. On-line analyser or (soft) sensor in feedback control.

A soft sensor or virtual sensor is a common name for software where several measurements are processed together. The interaction of signals can be used for calculating new quantities that do not need to be measured. Soft sensors can estimate unmeasured, but important variables from other easily measured variables using computational models. These inferential measurements can be used in fault diagnosis and control applications and for the validation of online analysers. Consequently, software sensors are models that can be realised in advanced control systems or in smart sensors to improve process control, optimisation and process safety by providing inferential measurements.

The implementation of soft sensors require proper mathematical models, whose development is a complex and time-consuming task. The motivation of our work is to develop tools for model identification and validation to support soft-sensor development. For non-parametric models, we worked out a genetic programming based methodology that can generate informative plots using nonlinear feature transformation. For parametric models, we present regression based techniques for feature selection and transformation methods to build adequate models with reduced complexity. Finally, we propose time series analysis methods to find local models of operating regimes.

## 1.2 Soft-sensor development

A very comprehensive review of the applications of soft sensors in process industry can be found in ref. [26, 27, 40]. As these implementations show, soft sensors are widely used in the hydrocarbon industry (e.g. in a fluid catalytic cracking process soft sensors are used to estimate catalyst circulation rate and heat of reaction

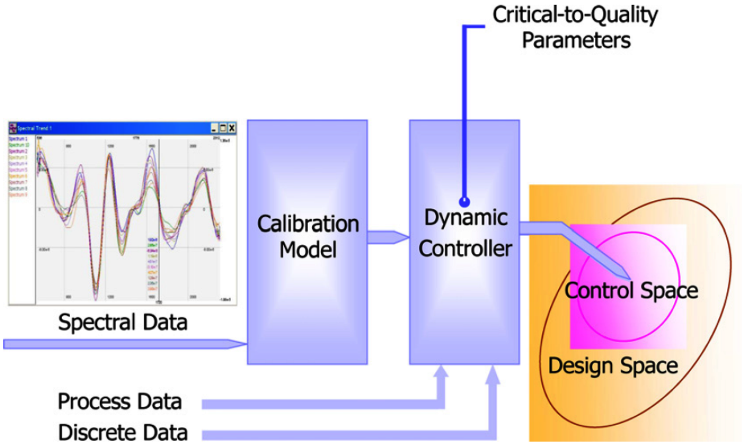

FIGURE 1.2. Scheme of NIR based process control [15].

to support model predictive control (MPC) [83]). Soft sensors are particularly widespread in the pharmaceutical industry in connection with Process Analytical Technology (PAT). A typical application is shown in ref. [76]. It is important to note that in PAT soft sensors are used to generate an on-line quality estimate based on on-line analytical measurements. An example of this concept can be seen in [16]. This PAT approach is critical in NIR based process control. Figure 1.2 shows an application where the spectral space is mapped into two dimensional space to define the control space representing good product quality.

Application of soft sensors and on-line analysers for (advanced) process control is not trivial. The book [23] deals with some key points of the soft sensors design procedure, starting from the necessary critical analysis of rough process data, to their performance analysis, and to topics related to on-line implementation.

There are several multivariate models and methods to support the prediction of product properties based on Near-Infrared (NIR) spectra. These methods can be separated into two groups: (i) parametric models (e.g. linear regression, multilinear regression, Partial Least Squares regression (PLS) ) and (ii) nonparametric methods (e.g. k-NN [81], False Nearest Neighbours (FNN), Neural Networks, Topological Near-Infrared Modeling [19, 72] - TOPNIR).

In the following, we give an overview of intelligent techniques that can support the development and maintenance of data-driven models used in soft sensors, and we show that soft-sensor models of advanced process control systems (APC) require sophisticated maintenance procedures.

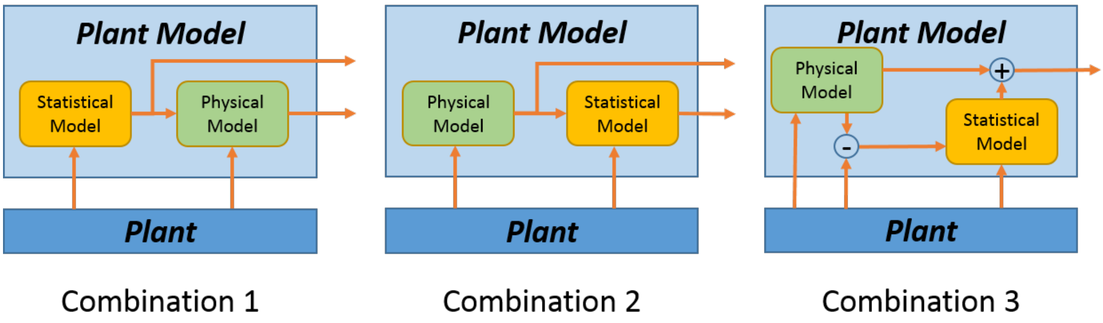

FIGURE 1.3. Soft sensor structures according to how property variables are estimated by the combination of white and black box models [59].

First-principle or data driven soft sensors can be distinguished according to the information used for soft-sensor development [2]. First-principle models (also referred as white-box, mechanistic or *a priori* models) are based on balance equation (mass, component, energy) and contain detailed chemical information about the system. Unfortunately, in practical applications processes are often too complex, uncertain and not sufficiently understood for complete model development. Therefore, the applicability of first-principle model based approach is very limited [40]. Furthermore, the development of first-principle models is very time-consuming procedure. In particular, it is hard to build precise first-principle models that can explain why defects appear in products. Model development time is a critical issue since product life cycles are getting shorter and the time available for improving product quality and yield requires fast and adaptive solutions [29].

Data driven (black-box or *a posteriori* ) model based soft sensors are built when no detailed knowledge is available about the process. In this case, data is used to create statistical models to determine the relationship between inputs and outputs. Statistical regression methods have become increasingly popular techniques for process modelling, and they are used for fault detection and quality estimation.

Data and *a priori* model driven approaches can have several synergistic combinations (see Figure 1.3).

- In the first case, a statistical model is the input of a physical model in the form of differential or algebraic equations or a complex flow sheeting simulator. In this case, a mathematical model is used to estimate parameters and phenomena that cannot be predicted easily.

- Combination 2 shows the case when outputs of the physical model are transformed into a statistical model.

- In the third option, the difference between measured and calculated variables are the inputs of a statistical model used for correction. Model-based data reconciliation techniques are similar to this approach.

*A posteriori* models can be distinguished into two different classes, parametric and nonparametric models. To describe the difference between the two modelling strategies let us define the standard form of *a posteriori* models. Let be $\mathbf{x}_k = [x_{1,k}, x_{2,k}, \dots, x_{n,k}]^T$ the vector of the $k = 1, \dots, N$th sample represented by the $i = 1, \dots, n$ (input) variables[1] and a $y_k$ the output of a system. We would like to give an estimated value $\hat{y}_k$ for the unmeasured $y_k$ using only the measured inputs $\mathbf{x}_k$. The common form of an *a posteriori* model is the following:

$$\hat{y}_k = f(x_{1,k}, x_{1,2}, \dots, x_{n,k}) = f(\mathbf{x}_k) \tag{1.1}$$

The most popular data-driven soft-sensor models are based on multivariate statistical techniques, i.e. the PCA and PLS, which together cover 38 % of the applications (see Figure 1.4). Soft sensors are regularly used for decision support. In these applications, appropriate classifier models have to be tuned. Figure 1.5 shows an example of this model structure. It is interesting to note that this scheme is also an example of a hybrid soft-sensor model, where different model types (PCA, SOM and RBFN) are combined.

Table 1.1. Numbers of known applications of soft sensors. Phys: White box models, MRA: PCA and multi linear regression models, ANN: Neural Networks. [43]

| | Methodology | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Process | Phys | MRA | PLS | O.L. | ANN | JIT | Gray | Toltal |
| Distillation | 20 | 256 | 41 | 6 | 0 | 5 | 3 | 331 |
| Reaction | 5 | 32 | 43 | 0 | 0 | 5 | 1 | 86 |
| Polymerisation | 0 | 4 | 8 | 0 | 3 | 0 | 5 | 2 |
| Others | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| Total | 25 | 293 | 93 | 6 | 3 | 10 | 9 | 439 |

According to Figure 1.5 we should note that Kalman on-line filters are often used for model updates as they have excellent performance characteristics, and they are

[1] We represent vectors in column form

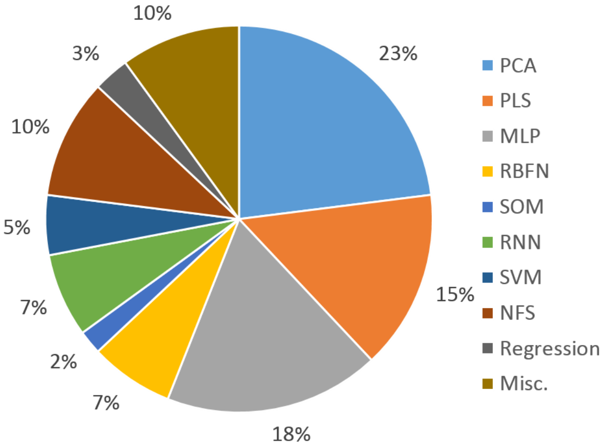

Figure 1.4. Distribution of model types used in soft sensors. PCA: Principal Component Analysis, PLS: Partial Least Squares regression, MLP: Multi Layer Perception - Neural Network, RBFN: Radial Basis Function Networks, SOM: Self-Organising Neural Models, SVM: Support Vector Machines. SPE stands for squared prediction error. [40].

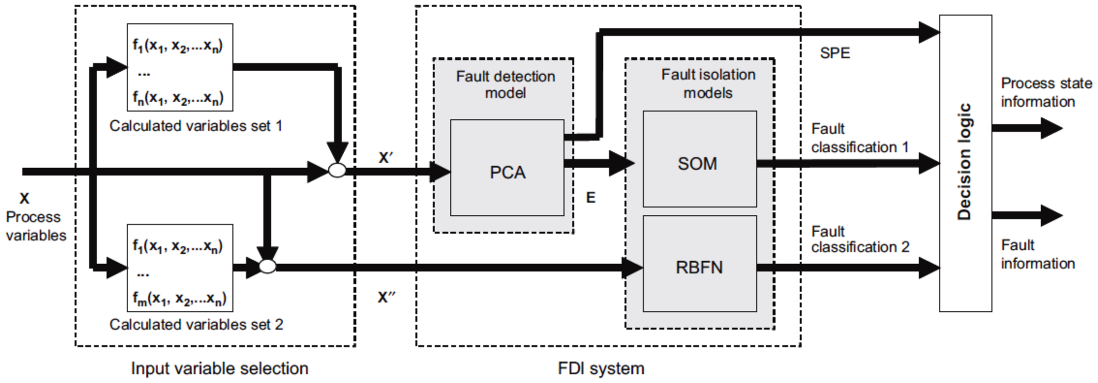

Figure 1.5. Hybrid soft-sensor model as combination of PCA, SOM and RBFN models. PCA is used for fault detection while classifier models are used for fault isolation [41].

simple to implement. In this way, all the process information can be included in the model leading to more precise predictions [40].

Building a high-performance soft-sensor is a very laborious task, since input variables and samples for model construction have to be selected carefully (e.g. by applying techniques of Design of Experiments), and parameters have to be tuned appropriately.

The general procedure for data-driven soft-sensor development is shown in Figure 1.6.

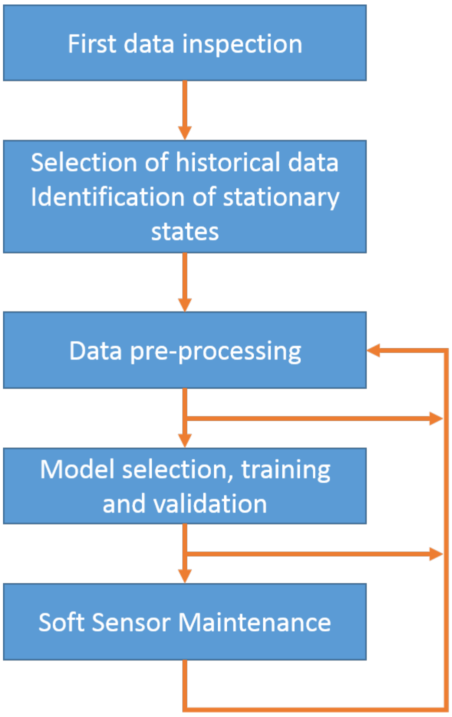
First data inspection
Selection of historical data
Identification of stationary states
Data pre-processing
Model selection, training and validation
Soft Sensor Maintenance

Figure 1.6. Main steps of data-driven soft-sensor development [40].

- *Data inspection*: In the first step it is necessary to specify the most important unmeasured variables and how we can calculate them from measured data. We should get an overview of the data structure and define problems which may be occurring. We collect the types of soft-sensor can be used during the next steps: simple regression model or complex model (e.g. PCA) or more complex (e.g. a neural network).

- *Selection of historical data and identification of stationary states*: In the second step, data to be used for the training and evaluation of the model is selected. Next, the stationary segments of the data have to be identified and selected. The identification of the stationary process states is usually performed by manual annotation of the data, [42], but time-series segmentation algorithms can also be used to automate this procedure [4].

- *Data preprocessing*: Since measured variables are on different scales, data pre-processing that includes standardisation of data (e.g. in the case of PCA it is inevitable). Usual steps are the handling of missing data, outlier detection, selection of relevant variables (i.e. feature selection and transformation) and data reconciliation. As shown in Figure 1.6, outlier removal, feature selection and transformation steps are repeatedly applied until the model developer considers the data as being ready to be used for the training and evaluation of the actual model [42].

- *Model selection, training and validation*: These are critical steps in the development of soft sensors. Selection depends on the application problem, nature of data and personal preferences since developers prefer models which are in their field of expertise. It is common practice to use a simple model if possible and gradually increase model complexity as long as significant improvement in the model's performance can be observed.

- *Validation and testing*: After finding the optimal structure and having trained the model, it is required to evaluate the performance of the model.

- *Maintanance*: Even a successfully developed performance of the soft sensors can deteriorate when process characteristics change. For example in chemical processes equipment characteristics can change by catalyst deactivation or adhesion. Soft sensors should be updated to follow these changes. Manually repeated construction of models should be avoided due to its heavy workload [43] .

## 1.3 Studied problems and the roadmap of thesis

In this thesis, we show three different aspects of model development for soft sensors and on-line analysers. In Chapter 2 we show a new genetic programming based method to support the development of non-parametric models for soft sensors and on-line analysers. In Chapter 3 we propose supporting tools for parametric model development. Data-based modelling requires proper raw data to build models having good performance, so in Chapter 4 we present some methods for time series analysis, model selection and validation.

We demonstrate the developed methods on two different problems. The first problem is a spectroscopy based process monitoring which is a widely applied technique in the chemical industry. Spectroscopy based modelling is multivariate by nature and difficult to manage. For these reasons supporting tools are required to build proper models for it. We show a method for visualisation of spectral databases (Chapter 2) and a regression based modelling technique (Chapter 3.) to predict quality measures during production. The second application example is energy monitoring (EM) of process systems. In EM systems the energy usage is estimated based on process variables using *a posteriori* models. We show visualisation tools and time series segmentation algorithms to help the development of these models. The application examples are detailed below.

### 1.3.1 Near infrared spectroscopy based process monitoring and modelling

In case of complex production systems the control of measured process values (e.g. temperature, pressure, flow rate) does not always ensure that the unmeasured product properties will be in predefined ranges of production orders or standards. E.g. in oil-industry cetane index and sulphur content are not measured online and the frequency of flash point, density, cold filter plugging point measurement are not enough for real time control. The objective of development of software sensors and online analysers is to support process control and monitoring by providing online information about these properties. The interaction of signals like temperatures, pressures - in our case absorption intensities - can be used for calculating new unmeasured quantities (like flash point, density etc.). These models can also be used for fault diagnosis and inferential control applications.

A widely used online measurement technique is near infrared spectroscopy. Information extracted from a spectral database can be used for process monitoring and to estimate unmeasured product properties. The main objective of spectroscopic modelling is to find the relations between recorded spectra and relevant material properties [36]. Topological modelling techniques are based on looking for similar spectra from a spectral database by nearest neighbourhood algorithms. Instead of performing explicit generalisation such as memory-based learning that compares new problem instances with instances seen in training. The class of such memory-based algorithms are called instance-based because it constructs hypotheses directly from training instances. One advantage that instance-based learning has over other methods is its ability to adapt its model to previously unseen data. While other methods generally require the entire set of training data to be re-examined when one instance is changed, instance-based learners may simply store a new instance or throw an old instance away. The disadvantage is that these non-parametric techniques cannot extrapolate.

The performance of instance based learning algorithms highly depends on the quality of the database used for estimation. Hence, data-driven modelling algorithms need a carefully designed and maintained database of training data (samples, instances). The coverage of the operating regimes and the structure of the indexed database should be consistent to support the fast searching for the nearest neighbours. Although the concept of this modelling scheme is quite simple, since spectral data is spread out in a high dimensional space it is difficult to decide which operating regimes need more training data and which samples (instances) should be removed to improve model prediction performance by inconsistency (noise) in the training data. Therefore, we not only developed learning algorithms to extract information from high dimensional spectral data but also developed several diagnostic tools to maintain and develop spectral databases.

As TOPNIR estimates a set of product properties, we developed tools to evaluate, estimate and improve the prediction performance of these models. Human supervision and intervention is always required in model development. In practical data mining and process monitoring applications high-dimensional data have to be analysed. In most of the cases it is very informative to visualise the hidden structure of complex data in a low-dimensional space.

Industrial applications require an easily implementable, interpretable and accurate projection. TOPNIR utilises heuristic nonlinear functions (aggregates) for the mapping of spectra as high dimensional object. These aggregates neither guarantee distance preserving nor neighbourhood preserving properties. We propose a performance metric to evaluate the quality of dimensional reduction. These techniques are applied and presented in Chapter 2.

The developed measures are based on the distance and neighbourhood preserving properties of mappings. We evaluate the quality of aggregate based mappings of TOPNIR and compare it to the most important dimensional reduction techniques (Multidimensional Scaling - MDS, Principal Component Analysis - PCA, Sammon Mapping, and Partial Least Squares - PLS model based two dimensional projection in Chapter 3). Results related to the online spectrometer of a fuel blending plant illustrate that the proposed approach is a useful way to visualise spectral databases. The suggested trustworthiness measure gives useful information about how topological information is preserved during the mapping of the aggregates and other techniques used to visualise the operating regions of the technology based on measured NIR spectrum.

A pair of aggregate functions implements feature selection and feature transformation. Finding the proper model structure is a complex nonlinear optimisation problem. We present a Genetic Programming (GP) based algorithm to generate optimal aggregate pairs (Chapter 2).

Regression based predictions are usually calculated by linear regression models, but Partial Least Squares regression (PLS) can also be used for modelling. We adapted a technique that allows the application of PLS to visualise spectral databases. The main benefit of this technique is that it allows the extension of operating region of the model as parametric models extrapolate better than nonparametric ones. This concept and the related results are discussed in Chapter 3.

Information hidden in the time-series of multivariate spectra can also be used to detect hidden changes in the operation of the technology and the measurement system status. We extended our former segmentation algorithm (published in [127]) to detect changes in the operation of a diesel fuel blending plant and the operation of a laboratory scaled reactor system. The whole methodology is presented in Chapter 4.

To demonstrate the above-mentioned novel techniques, multiple datasets provided from the Dune Refinery of MOL Ltd were analysed. Although the developed algorithms have been designed to support the model development of topological (TOPNIR) models, they can additionally be applied to build parametric models with good prediction and process monitoring performance. Besides the detailed analysis of the proposed framework, the thesis gives a detailed analysis of the TOPNIR algorithm and the related TopWin Software. Thanks to this analysis and the produced development tools the whole modelling and soft sensor maintenance procedure can be performed even more sophisticated manner.

### 1.3.2 Energy monitoring of process systems

Advanced production systems maximise the production and at the same time minimise cost and environmental impact [64]. The purpose of energy monitoring and targeting is to provide a better understanding of how energy is being used. The so-called energy portfolio allows the classification and prioritisation of energy consumers and the derivation of target-oriented action plans towards energy and resource efficiency improvement [74]. Energy efficiency can be improved based on data analysis [54]. Monitoring of energy consumption of industrial process systems requires sophisticated tools and methodologies.

The Industrial Technologies Program (ITP) is the lead government program working to increase the energy efficiency of the U.S. industry - which accounts for about one third of the U.S. energy usage. In partnership with industry, ITP helps to research, develop, and deploy innovative technologies that companies can use to improve their energy productivity, reduce carbon emissions, and gain a competitive edge [35]. ITP is developing methods that will help to quantify energy-efficiency improvements in the most energy-intensive process streams [21]. Analyses such as energy bandwidth studies will enable to focus on the processes or unit operations with the greatest potential for energy efficiency gains and maximise the benefit of research investments [24].

Energy bandwidth analyses provide a realistic estimate of the energy that may be saved in an industrial process by quantifying three measures of energy consumption [21]:

- *Theoretical minimum energy (TME)*: TME is a measure of the least amount of energy that a particular process would require under ideal conditions. TME calculations are based on the thermodynamic analyses of primary chemical reactions using the change in Gibbs free energy ($\Delta G$), and assume ideal conditions. In some cases, the TME values were obtained through industry publications or using the heat of reaction ($\Delta H_r$) due to insufficient Gibbs free energy data.
- *Practical minimum energy (PME).* The PME represents the minimum energy required to carry out a process in real-world, non-ideal conditions (e.g., temperature, pressure, selectivity and conversions less than 100%) that result in the formation of by-products, the need for product separation, catalyst and equipment fouling, and other factors. These conditions impose limitations that make it impossible to operate at the theoretical minimum. The energy savings considered for the practical minimum analysis are primarily based on best practices and state-of-the-art technologies currently available in the marketplace.
- *Current average energy (CAE).* CAE is a measure of the energy consumed by a process carried out under actual plant conditions. This measure exceeds

both the theoretical and practical minimum energies due to energy losses from inefficient or outdated equipment, process design, poor heat integration, poor conversion and selectivity, amongst other factors.

The bandwidth is the difference between PME and CAE and provides a snapshot of energy losses that may be recovered by improving current processing technologies, the overall process design, current operating practices and other related factors [21].

In this thesis we focus on the optimisation of CAE by the application of energy monitoring systems. Energy monitoring improves energy efficiency in process plants by helping plant operators, engineers and managers to track actual and target energy consumption [8]. Such system allows the user in the following tasks [57]:

1. Detect avoidable energy waste that might otherwise remain hidden. This is waste that occurs at random because of poor control, unexpected equipment faults or human error.
2. Quantify the savings achieved by energy projects and campaigns.
3. Identify fruitful lines of investigation for energy surveys.
4. Provide feedback for staff awareness, improve budget setting and undertake benchmarking.

Monitoring is based on continuous comparison of actual and estimated energy consumption. Energy efficiency has the following four components: performance efficiency, operation efficiency, equipment efficiency and technology efficiency [82]. A systematic overview of the state of the art in energy and resource efficiency increasing methods in the domain of discrete part manufacturing is given in reference [20].

Reducing energy consumption of machine tools can significantly improve the environmental performance of manufacturing systems. A structured approach at different system scale levels is presented in ref. [20]. Starting from a process unit, multi-machine, factory, multi-facility and supply chain levels are covered. To automate monitoring and analysis of energy consumption, event analysis techniques are reported in ref. [80]. Energy consumption characteristics of machine tools are compared and the potential of using the obtained data for energy labelling of machine tools is discussed in ref. [9]. Most of these developments are focused on discrete manufacturing [45]. Chemical industry is the largest energy consumer among different industrial sectors; it is responsible for around 4.7% of the total energy consumption in Europe. This thesis focuses on this domain.

Methods for calculating expected consumption fall into two categories. There are those based on precedent (comparison with previous periods) and activity-based methods that relate expected consumption to its driving factors (weather, production throughput, etc.). Precedent-based targeting models are most commonly used in monthly monitoring schemes, when expected consumption can be deduced from what was used in the corresponding month a year before. One weakness of this procedure is that it assumes that conditions were comparable in the two months. A more problematic issue is what happens when energy waste has occurred. The resulting excessive consumption erroneously raises the expected quantity a year later. To work effectively, abnormal months should be disqualified to prevent them being used for target setting. At the other end of the scale, some automatic monitoring and targeting schemes compare consumption at very short intervals (e.g. half an hour), with a target template derived from previous similar days. The same caveat applies: abnormal consumption patterns must be filtered out from the pool of data used as precedents to make this effective. Precedent-based targets, used with caution, may be the best method when consumption is seasonal, but unrelated to any measurable driving factor. Generally, however, precedent-based targeting models can be too simplistic and organisations may want to consider activity-based targeting. This is particularly appropriate when there are clear drivers for changing energy consumption, for example, changes in production throughput [75].

Activity based targeting models calculate the expected consumption based on models. These models estimate the expected values of the consumptions based on some measured values (driving factors). Activity Based Costing (ABC) is a widely used model for measuring the cost and performance of business and production processes. This model can be easily adapted to measure energy usage. The *activity based energy usage* model (ABE) has been presented in ref. [67]. The key idea is that once a model can estimate the energy consumption based on the activities and actual state of the processes, 'what if scenarios' can be analysed and energy abatement projects can be recommended.

Apart from the traditional volume-based accounting approach, the ABC approach is useful, especially when the rapid assessment of energy load curtailment is required. There were several studies to modify ABC with an intention to expand to include environmental factors. Jurek et al. proposed an APC-based energy consumption prediction model used to clarify the production and non-production energy loads rapidly, thereby being able to figure out the amount of possible load curtailment quickly [39]. The concept of ABC can also be applied to energy management and provide an energy usage distribution for the process to identify and evaluate energy consumption and cost saving opportunities [18]. A novel decision model based on ABC and stochastic programming has been developed to accurately evaluate the impact of load curtailments and determine as to whether or not accept an energy load curtailment offer in ref. [62]. Based on an activity-based targeting model we can use multi-objective optimisation to find sustainable energy usage mode [30] or to build an on-line energy management and optimisation system that identifies actionable cost saving opportunities in real time and empowers operators to take immediate action [63].

Analytical models using first-principle knowledge calculate the energy consumption based on driving actors are presented in ref. [12] and [70]. This approach performs well by a clear and good understanding of the mechanisms of the process.

Activity-based targeting models are applied to calculate expected the consumption reference of process units. This approach can be considered as a special software sensor [85]. A similar on-line neural network based software application has already been patented [68]. The patented neural network based methodology enables manufacturing facilities to meaningfully determine their energy performance, no matter how complex, taking the production rates and ambient conditions in consideration. Causes of statistically significant deviations are diagnosed and corrective actions highlighted. The software application is designed to be updated dynamically so that users can effectively manage performance on the basis of current information. Performance of data-driven targeting models depends on a complex set of process variables which are selected and ranked based on a heuristic and subjective evaluation of the operation. These models are generally statistical regression models. Partial least squares (PLS) is a perfect method for constructing predictive models from a large number of correlated input variables [10]. PLS was developed in the 1960s by Herman Wold as an econometric technique, but soon it become a widely applied tool in chemical engineering. These multivariate statistical and regression models can also be used for fault detection and isolation since complex processes can easily have hundreds of process variables [52]. In the proposed approach the monitoring of the processes is based on the difference between targeted and the measured energy consumption.

In Chapter 4 we present the concept and technical details of the proposed historical data based energy monitoring system and demonstrate its applicability at Heavy Naphtha Hydrotreater and Catalytic Cracking Reforming Units of MOL Hungarian Oil and Gas Company.

The dataset used for the identification of the targeting model is selected based on a heuristic and subjective evaluation of the process' operation. The disadvantages are: it is time-consuming, and it does not give any hint to the user how the targets given by the resulted models should be handled. We developed goal-oriented time-series segmentation techniques to automate this procedure. The algorithms are detailed in Section 4.2 and 4.3. Targeting-models for different operating regions can be automatically determined using the proposed novel segmentation algorithm.

All the programs used to generate the results in this thesis can be downloaded from
`https://github.com/kulcsartibor/phd-thesis-programs` and
`http://www.abonyilab.com/`

# Chapter 2

# Non-parametric model development

Nonlinear function based mapping of the feature space can visualise the hidden structure of high dimensional data. We developed a genetic programming (GP) algorithm to determine the optimal structure of these functions. The mapping can be tailored based on the intended use of the visualisation problem by designing goal-oriented fitness functions. We demonstrate the applicability of the method on a nonparametric regression and in a nonparametric classification problem. We designed NIR spectra based software sensor for product quality estimation with the use of the developed tools and compared the GP based visualisations to standard dimension reduction techniques like PCA and MDS.

## 2.1 Introduction

To decrease the complexity of regression and classification problems and to visualise high dimensional data dimensional reduction algorithms should be used. Dimensionality can be reduced by feature selection or feature extraction algorithms. *Feature selection* selects a subset of the features (variables) which contain the most important characters of data objects. The well known exhaustive search method [37] examines all possible subsets of the variables. Branch and bound [60] and floating search [66] techniques can reduce the enormous computational cost of this task by the introduction of a sophisticated search algorithm.

In contrast to feature selection *feature extraction methods* do not select the most relevant variables but they combine them into some new attributes. Usually, two or three attributes are generated to support the visualisation of the high dimensional data. The most commonly used linear dimensionality reduction methods are the Principal Component Analysis (PCA) [38], the Independent Component Analysis (ICA) [17], Multivariate Curve Resolutions (MCR) and the Linear Discriminant Analysis (LDA) [22]. These linear methods provide a poor representation of data when the analysed process is nonlinear, and a wider range of operating regime should be modelled. In these cases *nonlinear dimensionality reduction methods* may outperform the traditional linear techniques. The most widely applied nonlinear dimensionality reduction methods are Kohonen's Self-Organizing Maps (SOM) [47], Sammon mapping [71], Locally Linear Embedding (LLE) [69], Laplacian Eigenmaps or Isomap [73]. The application of these methods is an active research area of chemometrics and process monitoring; nowadays several successful applications are reported.

Nonlinear function (aggregate) can also form easily implementable and accurate projections. A pair of these functions simultaneously realises feature selection and transformation tasks. Finding the proper structure of this functions is a complex non-linear optimisation problem. We present a Genetic Programming (GP) based algorithm to generate nonlinear aggregates. This method is based on a *tree representation* based symbolic optimisation technique developed by John Koza [48]. This representation is extremely flexible; trees can represent computer programs, mathematical equations or complete models of process systems [5]. In ref. [14], GP is already applied in the visualisation of high-dimensional process data. Simple non-linear functions were identified to preserve the distances among the data-points. The drawback of this approach is that since the models were not parametrized only simple mappings with approximative distance preserving properties were generated.

Since the studied soft sensors realise $k$ nearest neighbour type nonparametric regression, the mapping should preserve the neighbourhood relations of the data. We developed a trustworthiness and continuity based problem-relevant measure to evaluate the quality of the visualisations since trustworthiness and continuity reflect the neighbourhood relations in the original and the mapped spaces. Since more than a hundred variables represents an NIR spectrum, finding the optimal mapping of this high dimensional space ($n >> 100$) into two dimensions is a complex problem. We propose a multi-chromosome based Genetic Programming (GP) algorithm for the structural optimisation of the mapping functions.

Our Genetic Programming (GP) based algorithm maximises the combination of the trustworthiness and continuity measures. To determine the optimal parameters of the mapping functions a pattern search [86] step is embedded into the GP. The resulted functions define aggregates of topological near infrared (TOPNIR) models usually used for product quality estimation of petroleum refinery products and the highlight the operating ranges of the process.

We applied the proposed method in a diesel fuel blending process at MOL Ltd. Duna Refinery to select and design pairs of aggregates that correctly reflect the hidden structure of the spectral database.

## 2.2 The modelling task

Topological Infrared Modelling (TOPNIR) is a widely applied method for predicting the $\mathbf{y}_i = \left[y_{i,1}, \dots, y_{i,n_y}\right]^T = \left[P_{i,1}, \dots, P_{i,n_y}\right]^T$ product or process stream properties (e.g. octane number, density) based on an $\mathbf{x}_i = [x_{i,1}, \dots, x_{i,n}]^T$ infrared spectrum composed of $l = 1, \dots, n$ absorbancies measured at different wavelengths. Usually $n \simeq 200$ samples contain enough information for modelling the functional relationship between absorbances and product properties:

$$\mathbf{y}_i = f\left(\mathbf{x}_i\right). \tag{2.1}$$

Instead of an explicit parametric function $\mathbf{y}_i = f\left(\mathbf{x}_i\right)$ TOPNIR utilises the well known $k$-nearest neighbour algorithm ($k$-NN). The $\mathbf{y}_i$ output of the model is estimated as the (weighted) average of the values of its $k$ nearest neighbours of the input vector $\mathbf{x}_i$. For modelling usually $N$ samples with known properties are available, $i = [1, \dots, N]$. For a new sample $\mathbf{x}_i$ we calculate $d_{i,j}$ distances of the spectra to each spectra in the database $\mathbf{x}_j$,.

As distance measure the *Minkowski distance* or its special cases, the Euclidean or Manhattan distances can be used:

$$d_{i,j} = d_p(\mathbf{x}_i, \mathbf{x}_j) = \left( \sum_{l=1}^{n} |x_{i,l} - x_{j,l}|^p \right)^{1/p} = \|\mathbf{x}_i - \mathbf{x}_j\|_p$$

where $p$ is a positive integer and $n$ yields the dimension of objects. For $p = 2$, the distance measure is refereed as *Euclidean*, while $p = 1$ defines the *Manhattan* distance. The differences may be weighted to take into account the different sensitivity of absorption to the property or to the spectrometer at a wavelength. Weights can also be used to reflect the reproducibility of the spectral measurement at wavelength $l$.

$$d_{i,j} = d\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sum_{l=1}^{n} \left|x_{i,l} - x_{j,l}\right| \lambda_l \tag{2.2}$$

When no prior knowledge is available $\lambda_l = \frac{1}{n}$.

To provide a robust estimate the weighting of the the samples is based on their distance from the $\mathbf{x}_i$ query point,

$$\mathbf{y}_i = f(\mathbf{x}_i) = \sum_{j=1}^{k} \left( \frac{\beta_{i,j}\left(d_{i,j}\right)}{\sum_{k=1}^{N} \beta_{i,j}\left(d_{i,j}\right)} \right) \mathbf{y}_j \tag{2.3}$$

In simplest case the $\beta_{i,j}\left(d_{i,j}\right)$ weighting function is a characteristic function, $1\left(d_{i,j} \leq R_i\right)$, where $R_i$ represents the region of the data point (usually the k-th largest $d_{i,j}$ distance measured from the $\mathbf{x}_i$ query point). The simplest yet effective nonlinear weighting is based on reciprocal weighting:

$$\beta_{i,j}\left(d_{i,j}\right) = 1\left(d_{i,j} \leq R_i\right) \frac{1}{d_{i,j}} \tag{2.4}$$

Exponential weighing can also be used. In this case often all of the samples are taken into account:

$$\beta_{i,j}\left(d_{i,j}\right) = \exp\left(-\lambda d_{i,j}\right) \tag{2.5}$$

where instead of $k$, the $\lambda$ parameter defines the locality of the model. In some cases it is beneficial to take into account the correlation among the features (absorbencies). The *Mahalanobis distance* uses the inverse of the $\mathbf{F}$ covariance matrix of the data as a distance norm (which can be considered as a special weighting function):

$$d_{i,j} = d_M(\mathbf{x}_i, \mathbf{x}_j) = \left(\mathbf{x}_i - \mathbf{x}_j\right)^T \mathbf{F}^{-1} \left(\mathbf{x}_i - \mathbf{x}_j\right) \tag{2.6}$$

In ideal case the locality region is independent from the actual query point $\mathbf{x}_i$, so $R = R_i \, \forall i$, and the key problem of model development is the careful selection of this value (or the selection of how $k$ that represents how many neighbours are used for the estimation). In the following we discuss how this parameter should be determined based on the expected accuracy of the model.

Let $E_v$ represent the experimental error or accuracy requirement of determining the $v$th property. To avoid unwanted mixing effects any two neighbours of the $i$th sample in the $R$ region (or among the $k$ nearest neighbours) should satisfy

$$||y_{a,v} - y_{b,v}||_2 < E_v\sqrt{2} \tag{2.7}$$

where $a$ and $b$ refer to the index of the two neighbours of the $i$-th query point.

The main concept of topological modelling is that samples having similar properties are also similar in the spectral space. This suggests that the required proximity region in the spectral space will differ property by property. Therefore, it is beneficial to separately determine $R_v$ based on the $E_v$ required prediction accuracies and use the smallest value as a minimal index:

$$i_m = R = \min_v (R_v) \tag{2.8}$$

When the density of the samples is sufficient, for every $k$ nearest neighbours of $\mathbf{x}_i$ the following inequality holds

$$d_{a,b} = d(\mathbf{x}_a, \mathbf{x}_b) < i_{min}, \ \forall a, b \tag{2.9}$$

that guarantees satisfactory prediction performance.

It can happen that the spectral database is incomplete due to the shortage of laboratory measurements (in a particular operating regime). In this case, it is possible to select a larger minimal index with e.g. $1-5$ times of the desired value, but it is anticipatory that the model will be less accurate for some properties.

### Illustrative example

The dataset used for model development contained 651 samples of spectra and 15 material properties. The spectra in the database were recorded at 195 discrete wavenumbers equally distributed in the range 4000 - 4776 $cm^{-1}$. After basline correction and normalization these spectra used as the input of the prediction model. The training outputs are the property values $\{y_1, y_2, \ldots, y_{n_y}\}$ of the samples. The properties were normalised into a range $[0, 1]$.

The performance of the TOPNIR model is considered as base case. The model performance was measured based on the correlation of the predicted and measured product properties. Table 2.1 shows that the $N$ number of the available samples differs for each property.

TABLE 2.1. The number of samples and the correlation of the property estimates of the TOPNIR model

| Property | $N$ | $R^2$ |
|---|---|---|
| Density | 441 | 0.971 |
| CI | 384 | 0.411 |
| CFPP0 | 229 | 0.964 |
| CFPP | 380 | 0.810 |
| CloudPt | 378 | 0.941 |
| FlashPt | 379 | 0.832 |
| T10 | 383 | 0.966 |
| T50 | 328 | 0.916 |
| T90 | 383 | 0.814 |
| E250 | 365 | 0.995 |
| E350 | 361 | 0.459 |
| E360 | 342 | 0.588 |
| PolyCycl | 331 | 0.559 |
| TotAro | 327 | 0.910 |
| VISC | 67 | 0.951 |

Table 2.2 shows the results of the $k$-NN estimation using leave-one-out validation (where $k = 3$). $R$ values show the average normalised distances of the $k$ neighbours, while $E_v$ represents the expectable accuracy of the model. These values should be compared to the standards and the requirements for accuracy related to the different properties. The applied FNN measure (see Section A.4) gives a reasonable estimate for the difficulty of the estimation problem. When the correlation between the $k$-NN and TOPNIR model is close to one than the FNN value is zero. It means that the prediction is better when the ratio of the false nearest neighbours is slow.

For comparison the performance of a well designed PLS model is also shown (details of this model will be presented in Chapter 3). It can be seen that in most of the cases the PLS model outperforms the TOPNIR and $k$-NN. This comparison illustrates the benefits and drawbacks of nonparametric and parametric modelling. When the number of samples is small, the parametric models - like PLS - give better performance. In the case of sufficient data (dense database), nonparametric models could perform better than linear models. However, it should be noted that both approaches require validated data. In the following section, we discuss the

Table 2.2. Weighted 3-NN estimation results - Comparison to PLS (The columns of )

| Property | $k$-NN | N | R | Ev | FNN | PLS |
|---|---|---|---|---|---|---|
| Density | 0.972 | 441 | 0.019 | 0.042 | 0.91 | 0.993 |
| CI | 0.284 | 384 | 0.023 | 0.039 | 6.25 | 0.190 |
| CFPP0 | 0.947 | 229 | 0.019 | 0.050 | 1.31 | 0.947 |
| CFPP | 0.631 | 380 | 0.015 | 0.163 | 0.00 | 0.769 |
| CloudPt | 0.932 | 378 | 0.024 | 0.047 | 0.26 | 0.950 |
| FlashPt | 0.803 | 379 | 0.024 | 0.056 | 0.26 | 0.878 |
| T10 | 0.964 | 383 | 0.023 | 0.042 | 1.31 | 0.908 |
| T50 | 0.904 | 328 | 0.026 | 0.070 | 0.00 | 0.970 |
| T90 | 0.815 | 383 | 0.023 | 0.060 | 0.00 | 0.849 |
| E250 | 0.852 | 365 | 0.024 | 0.067 | 0.00 | 0.955 |
| E350 | 0.156 | 361 | 0.023 | 0.038 | 1.94 | 0.308 |
| E360 | 0.484 | 342 | 0.023 | 0.145 | 0.29 | 0.431 |
| PolyCycl | 0.243 | 331 | 0.027 | 0.032 | 2.72 | 0.429 |
| TotAro | 0.904 | 327 | 0.027 | 0.054 | 1.53 | 0.905 |
| VISC | 0.978 | 67 | 0.014 | 0.047 | 0.00 | 0.999 |

role of the visualisation of a high dimensional spectral space in the selection and validation of the datasets used for model development.

## 2.3 Topological mapping based visualization

Instance-based prediction algorithms used for property estimation are based on the assumption that similar spectra represent samples having similar product properties. Therefore, to support the development of nonparametric models, it is beneficial to visualise the spectral database to check how the available set of spectra cover the operating region of the process.

The goal of *dimensionality reduction* is to map a set of observations from a high-dimensional space ($D$) into a low-dimensional space ($d$, $d \ll D$) preserving as much of the intrinsic structure of the data as possible. Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ be a set of the observed data, where $\mathbf{x}_i$ denotes the $i$-th observation ($\mathbf{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,D}]^T$). Each data object is characterized by $D$ dimensions, so $x_{i,j}$ yields the $j$-th ($j = 1, 2, \ldots, D$) attribute of the $i$-th ($i = 1, 2, \ldots, N$) data object. Dimensionality reduction techniques transform data set $\mathbf{X}$ into a new data set $\mathbf{Y}$ with dimensionality $d$ ($\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$, $\mathbf{y}_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,d}]^T$). In the reduced space many data analysis tasks (e.g. classification, clustering, image recognition) can be carried out faster than in the original data space.

### 2.3.1 Mapping based on aggregates

Industrial applications require easily implementable, interpretable and accurate projections. Nonlinear functions (often referred as aggregates) are useful for this purpose. A pair of these functions realises feature selection and transformation. Such mapping is used for the visualisation and indexing of spectroscopic databases in the Topological Mapping using Aggregates (TOPNIR) modelling framework [19]. TOPNIR performs a two-dimensional mapping of the spectral space to visualise the operation regimes of the process. There are 14 aggregates defined in the TOPWIN software used as a framework of the TOPNIR algorithm. The aggregates are equations that combine absorbances measured at significant wavelengths. Aggregates somehow reflect product properties. Since these properties can be dependent on different ranges of the spectra each aggregate built up several wavelengths to contain enough information related to a particular chemical property, e.g. the aromatic and the olefinic property have own ranges in the spectrum. Each aggregate builts up to six wavelengths to contain enough information related to a particular chemical property. For example, the aromatic and the olefinic property have own ranges in the spectrum (see Figure 2.1).

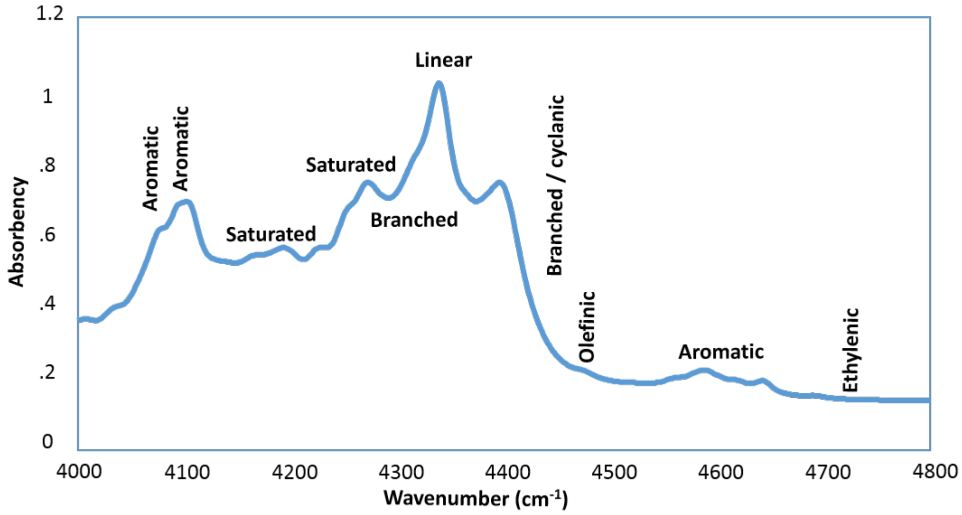

FIGURE 2.1. Significant wavelengths

The two main forms of the aggregates are shown by equation (2.10) and (2.11).

$$y_1 = a_{1,0}\frac{a_{1,1}x_{1,1} \cdot a_{1,2}x_{1,2}}{a_{1,3}x_{1,3} \cdot a_{1,4}x_{1,4}} \tag{2.10}$$

$$y_2 = a_{2,0}\frac{a_{2,1}x_{2,1} + a_{2,2}x_{2,2}}{a_{2,3}x_{2,3} \cdot a_{2,4}x_{2,4}} \tag{2.11}$$

where $x_l, i$ represents an absorbance value of the spectra at given wavelength.

Simultaneously two aggregates are used to give a two-dimensional mapping of the spectral space. Figure 2.2 shows a mapping defined by one possible combination of these aggregates (Naro and Parox) in the case of the previously presented case study. As it is illustrated on Figure 2.2, samples that are close to each other in the spectral space are also neighbours in the space of the property variables. Aggregates do not have a direct effect on the prediction performance. However, since boundaries (boxes) of operating regimes of the models are defined in this space, they have an indirect influence on the prediction performance. Therefore, the main issue to project high dimensional data into lower dimension is to discover the hidden structure of the original data set and the model coverage in the operating range. As can be seen on Figure 2.2 the visualisation highligths that the database contains samples from two different operating modes (summer and winter diesel).

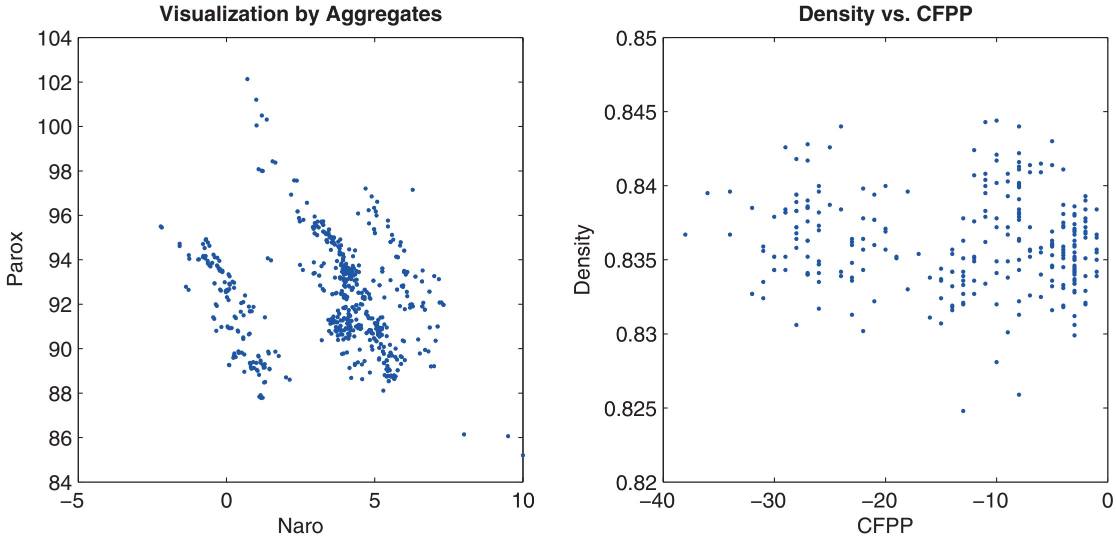

FIGURE 2.2. Mapping of the spectral space by Naro and Parox aggregates. The similarity of the figures shows that similar spectra have similar product property.

### 2.3.2 Evaluation of the mapping quality

In the previous session we show howed that the $E_v$ required accuracy of the estimation of a given property defines a region in the spectral space $i_m$ (see Figure 2.4). Since the structure of spectral database has an important effect to the estimation performance, it would be good to design a tool that enables the accurate and informative visualisation of these distances and topology of the data. Aggregates neither guarantee distance preserving nor neighbourhood preserving the property of the mapping. To provide valuable feedback about the quality of the mapping we propose a performance measure based on the distance and neighbourhood preserving properties of the mappings.

To measure the distance preservation of the mappings we propose the application of the classical MDS and the Sammon stress function based measures. The neighbourhood preservation of the mappings and the local and global mapping qualities are measured by functions of trustworthiness and continuity. Since Kaski and Venna pointed out that every visualisation method has to make a tradeoff between gaining excellent trustworthiness and preserving the continuity of the mapping [87] we propose a measure based on the combination of these.

A projection is said to be *trustworthy* when the nearest neighbours of a point in the reduced space are also close in the original vector space. Let $n$ be the number of the objects to be mapped, $U_k(i)$ be the set of points that are in the $k$ size neighbourhood of the sample $i$ in the visualization display but not in the original data space. The measure of trustworthiness of visualization can be calculated in the following way:

$$\begin{gathered} M_1(k) = 1 - \\ - \tfrac{2}{nk(2n-3k-1)} \textstyle\sum_{i=1}^{n} \sum_{j\in U_k(i)} \left(r\left(i,j\right) - k\right) \end{gathered} \tag{2.12}$$

where $r(i,j)$ denotes the ranking of the objects in input space.

The projection onto a lower dimensional output space is said to be *continuous* [87] when points near to each other in the original space are also nearby in the output space. The measure of continuity of visualization is calculated by the following equation:

$$M_2(k) = 1 - \tfrac{2}{nk(2n-3k-1)} \textstyle\sum_{i=1}^{n} \sum_{j \in V_k(i)} \left( s\left(i,j\right) - k \right), \tag{2.13}$$

where $s(i,j)$ is the rank of the data sample $i$ from $j$ in the output space, and $V_i(k)$ denotes the set of those data points that belong to the $k$-neighbours of data sample $i$ in the original space, but not in the mapped space used for visualization. When the mapping is based on geodesic distances, trustworthiness and continuity are calculated based on the geodesic distances.

Both trustworthiness and continuity functions are functions of the number of neighbours $k$. Usually, the qualitative measures of trustworthiness and continuity are calculated for $k = 1, 2, \ldots, k_{max}$, where $k_{max}$ denotes the maximum number of the objects to be taken into account. At small values of parameter $k$ the local reconstruction performance of the model can be tested, while at larger values of parameter $k$ the global reconstruction is measured.

The non-metric stress can be formulated as follows[1]:

$$E_{nonmetric} = \sqrt{\sum_{i<j}^{N} (\widehat{d}_{i,j} - d_{i,j})^2 / \sum_{i<j}^{N} d_{i,j}^2}, \tag{2.14}$$

where $\widehat{d}_{i,j}$ yields the disparity of $\mathbf{x}_i$ and $\mathbf{x}_j$, and $d_{i,j}$ denotes the distance between the vectors $\mathbf{y}_i$ and $\mathbf{y}_j$.

Figure 2.3 shows an example for the dimensional reduction into two-dimensional plain using Non-Classical Multidimensional Scaling (NMDS). The 2.3-A diagram shows the spectra of the samples (651 pcs. in this dataset). On this view, the samples cannot be distinguished because the difference between the spectra is relatively small. On the 2.3-B diagram one point represents on sample. The distance between two points is according to the distance of the corresponding two spectra.

A pattern can be recognised because the samples are separated into two distinct regimes which cannot be identified directly from the spectra.

[1] Traditionally, the non-metric stress is often called Stress-1 due to Kruskal [87]

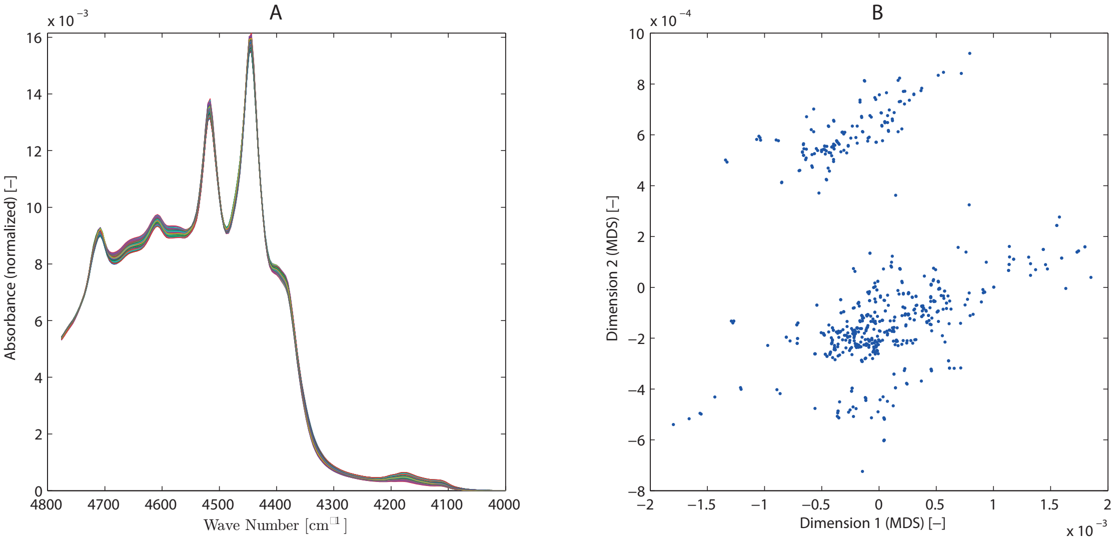

FIGURE 2.3. Structure of the data set in the (A) spectral space and in a (B) 2D mapped space

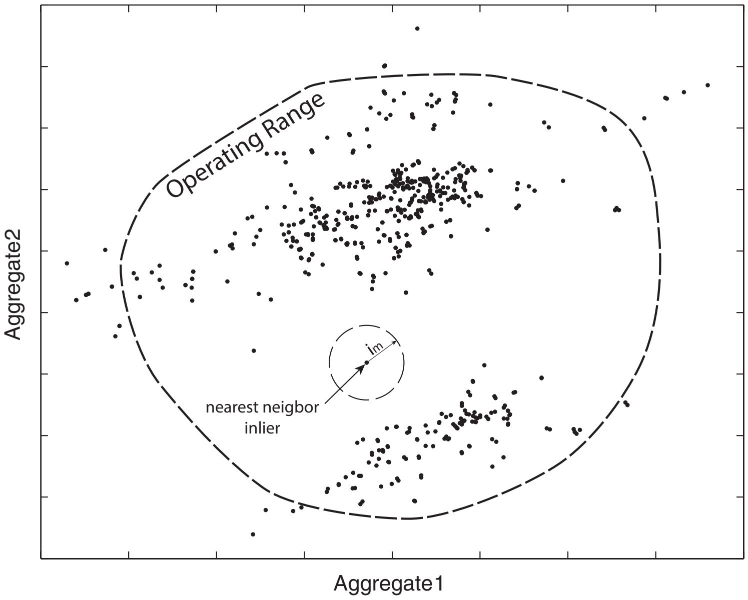

FIGURE 2.4. Model coverage in the operating range

Besides the 2D visualisation of data can show the model coverage too. In modelling aspect three type of samples are distinguished, *inlier*, *outlier* and *nearest neighbour distance outlier*

- *inlier* - a spectrum residing within the range of multivariate calibration space and the gap to the nearest neighbour is lower than a predefined limit ($i_m$).
- *outlier* - a spectrum residing out of range of multivariate calibration space.

- *nearest neighbour distance outlier* - a spectrum residing within the range of multivariate calibration space but the gap to the nearest neighbour is higher than a predefined limit ($i_m$).

Figure 2.4 illustrates examples of three types of samples. The operating ranges of the models and $i_m$ nearest neighbour distance limits can be easily defined based on this visualization, the number of samples and model coverage.

### 2.3.3 Genetic programming based visualisation

The proposed measures are proven to be useful for the selection of the best pairs of aggregates. It can happen that the user is not satisfied with the results, and he or she is interested in the design of new aggregates that minimises the proposed cost function. This task can be considered as a complex structural optimisation problem. Since aggregates are nonlinear functions of a small subset of hundreds of potential variables, the optimisation problem is so complex as a heuristic goal oriented optimisation algorithm is needed.

We developed a goal oriented genetic algorithm to automate the search for the optimal set of features ($x_{i,j}$) and the structure of the aggregate functions. The aggregates are represented by trees (see Figure 2.5) and we applied genetic operations among the potential solutions to get better and better mappings.

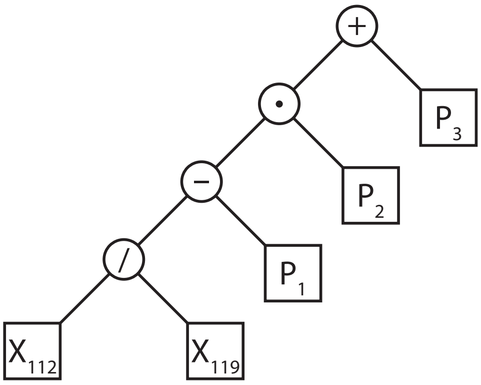

FIGURE 2.5. Decomposition of a tree to function terms

Because the algorithm of Genetic Programming is well-known, we will focus on the specific parts of the algorithm. Unlike standard optimisation methods, in which potential solutions are represented as numbers (usually vector of real numbers), the symbolic optimisation algorithms represent the potential solutions by the structured ordering of several symbols. One of the most popular methods for representing structures is the binary tree. A population member in GP is a hierarchically structured tree consisting of functions and terminals. The functions and terminals are selected from a set of functions (operators) and a set of terminals. For example, the set of operators $F$ can contain the basic arithmetic operations: $F = \{+, -, *, /\}$; however, it may also include other mathematical functions, Boolean operators, conditional operators or Automatically Defined Functions (ADFs). In this work, we used only arithmetic operations. The set of terminals $T$ contains the arguments for the functions. For example $T = \{x_1, \dots x_n, p_1, \dots p_m\}$ with $x_i$ represents the elements of possible input variables and $p_j$ represents the parameters. A potential solution may be depicted as a rooted, labelled tree with ordered branches, using operations (internal nodes of the tree) from the function set and arguments (terminal nodes of the tree) from the terminal set.

Genetic Programming is an evolutionary algorithm. It works with a set of individuals (potential solutions), and these individuals build up a generation. In every iteration (i) the algorithm evaluates the individuals and selects the best ones for reproduction according to their fitness value, (ii) generates new individuals by mutation (Figure 2.6), crossover (Figure 2.7) and direct reproduction, (iii) finally creates the new generation. The algorithm is illustrated in Figure 2.8. The fitness function reflects the goodness of a potential solution which is proportional to the probability of an individual's selection.

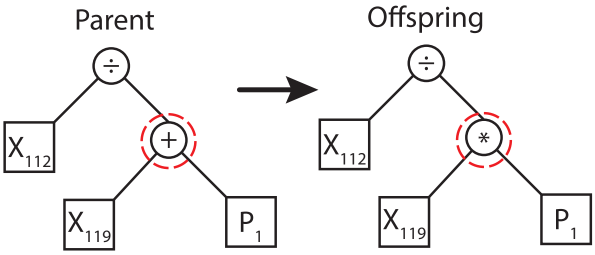

Figure 2.6. Effect of the mutation

At the beginning of this chapter, we showed that a pair of aggregate functions realise efficient feature selection and transformation, but there are no guidelines and tools that can be used to find the proper model structure. The proposed method is based on a *tree representation* based symbolic optimisation technique developed by John Koza [48]. This representation is extremely flexible; trees can

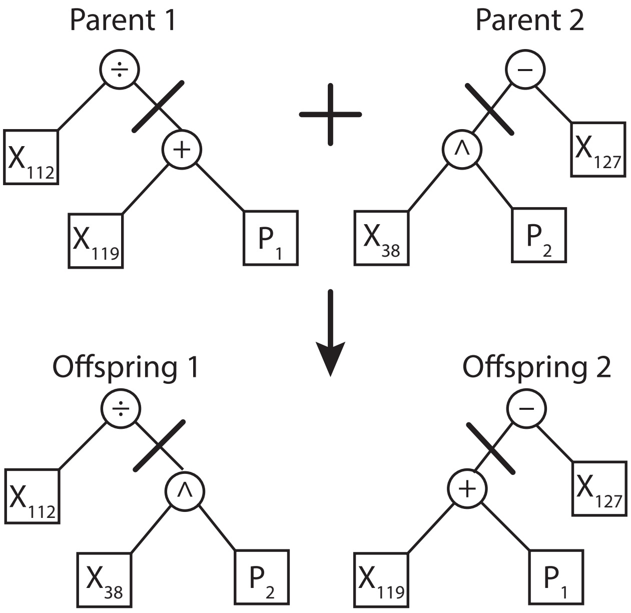

FIGURE 2.7. Effect of recombination

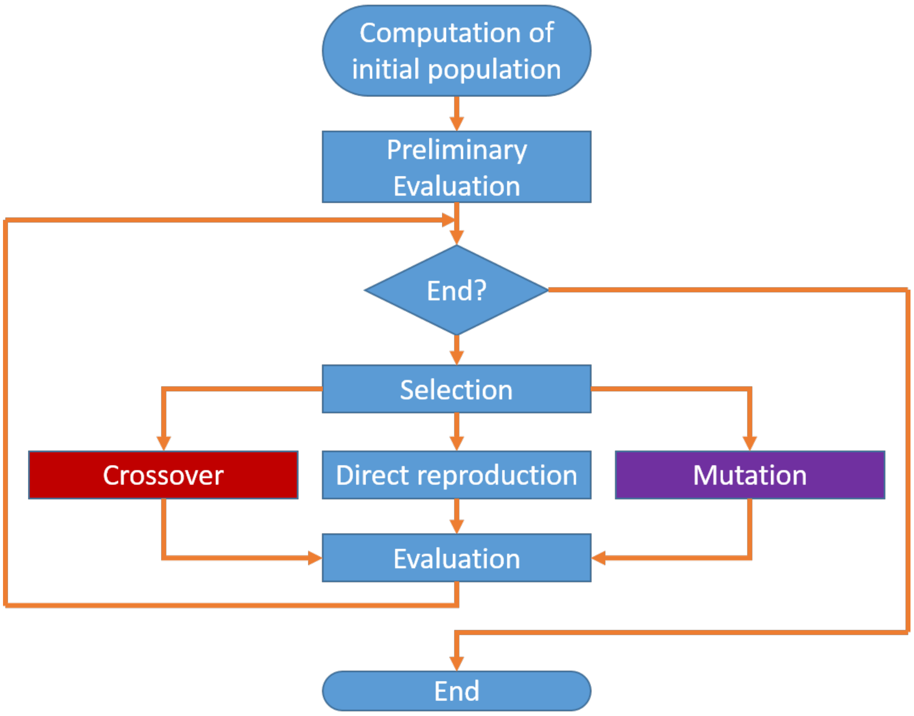

FIGURE 2.8. Scheme of the evolution loop followed by genetic programming

represent computer programs, mathematical equations or complete models of process systems. This scheme has been already used for circuit design in electronics, algorithm development for quantum computers, and it is suitable for generating model structures: e.g. identification of kinetic orders, steady-state models and differential equations. In [14] GP is applied to find simple nonlinear functions by minimising the distance preservation based Sammon stress function. The drawback of this approach is that since the models were not parametrized only simple mappings with approximative distance preserving properties were generated.

The parameters of the functions (aggregates) have an enormous impact on the mapping's performance. To find optimal values of these parameters we embedded a nonlinear parameter optimisation step into the GP. After the GP generated the new population of model structures a Pattern Search (PS) algorithm calculates the optimal values of the parameters. Furthermore, the cost function is based on the neighbourhood preserving properties of the mapping instead of distance preserving measures since this measure much closer reflects the application of the visualizer high dimensional instance-based models.

The proposed approach has been implemented in MATLAB. The user should only define the high dimensional data that should be mapped, one aggregate function which pair should be found by the optimisation, and the set of the terminal nodes (the set of the variables of the model and set of the internal nodes - mathematical operators. Based on our experiments we found that with the parameters given in Table 2.3 the GP can find good solutions for various problems. To give consistent results, these values have not been modified during the experiments presented in this chapter.

### 2.3.4 Topology preserving property based cost function

As dimensional reduction methods are based on the preservation of dissimilarities and the neighbourhood relation of objects, the numeral evaluation of mappings aims to measure the realisation of these principles. *The neighbourhood preservation of mappings* can be measured by functions of trustworthiness and continuity. Kaski and Vienna pointed out that every visualisation method has to make a tradeoff between gaining good trustworthiness and preserving the continuity of the mapping [78, 79]. A projection is said to be *trustworthy* [44, 79] when the nearest neighbours of a point in the reduced space are also close in the original vector space. Let $N$ be the number of the objects to be mapped, $U_k(i)$ be the set of points that are in the $k$ size neighbourhood of the sample $i$ in the visualisation display but not in the original data space. The measure of trustworthiness of visualisation can be calculated by Equation (2.12). The projection into a lower dimensional output space is said to be *continuous* [44, 79] when points near to each other in the original space are also nearby in the output space. The continuity of visualisation is calculated by the Equation (2.13).

Trustworthiness and continuity are depending on the number of neighbours $k$. Usually, the qualitative measures of trustworthiness and continuity are calculated for $k = 1, 2, \ldots, k_{max}$, where $k_{max}$ denotes the maximum number of the objects to be taken into account. At small values of parameter $k$ the local reconstruction performance of the model can be tested, while at larger values of parameter $k$ the global reconstruction is measured.

$$fitness = M_1 M_2 = \frac{1}{N} \sum_{k=1}^{N} M_1(k) \sum_{k=1}^{N} M_2(k), \tag{2.15}$$

where $M_1$ and $M_2$ are given by Equation (2.12) and (2.13). The $N$ is the number of projected data-points.

### 2.3.5 Multi-chromosome genetic programming for optimal 2D mapping

Optimal 2D mapping based on aggregates needs special representation for aggregate pairs. However in the common GP algorithms the chromosome and individual are identical - namely, one individual has only one chromosome. The presented method uses multi-chromosome genetic programming, which means that each individual in the population has two chromosomes.

The Figure 2.9 shows the schematic of the genetic representation. The $A$ and $B$ chromosomes produce together a fitness value (cost function) so the selection works on the individual level and takes the $A$ and $B$ chromosomes together. The selection uses roulette wheel method to generate survival probability of the individuals.

Crossover is allowed only on same type $A$ or $B$ chromosomes in the selected two individuals. Let be the two selected individuals $I_1$ with chromosomes $A_1, B_1$ and $I_2$ with chromosomes $A_2, B_2$. The crossover operation will be applied between $A_1$ - $A_2$ and $B_1$ -$B_2$. The crossover between $A_1$ - $B_2$ or $A_2$ - $B_1$ is forbidden.

The program flow is detailed in the Algorithm 1.

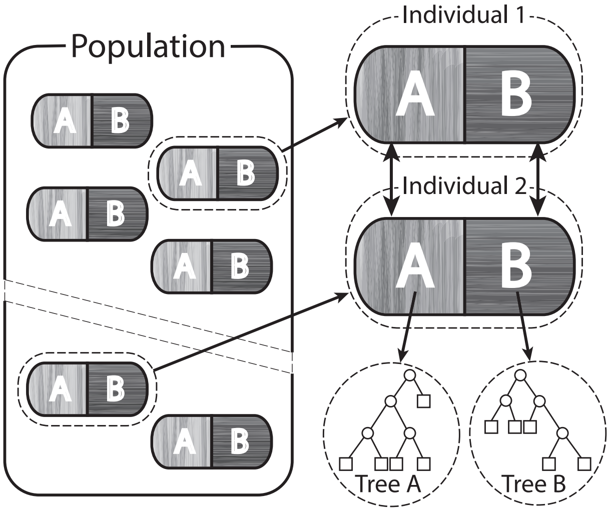

FIGURE 2.9. Multi-Chromosome representation of a 2D projection

## 2.4 Application examples

The first application study demonstrates how the proposed multi-chromosome genetic algorithm can be used to visualise a high dimensional spectral database. In the second example, we show that with the application specific fitness function the algorithm can also be applied in classification tasks.

### 2.4.1 Visualisation of spectral database

The first example focuses on product property prediction of a gas oil blending unit. The blending unit of the Dune Refinery (MOL Ltd, Hungary) is equipped with an online NIR analyser. The prediction method is implemented in the TOPNIR software framework provided by the supplier of the NIR analyser.

#### 2.4.1.1 Visualization results

As can be seen, the database contains samples from two different operating modes (summer and winter diesel), and some of these mappings can separate these operating regimes. It is interesting to see that there are also pairs of aggregates where correlation among them is too high to provide an informative mapping. We used the *topology preserving mapping based cost function* to select the pair of aggregates that is the best performer regarding the reflection of the hidden structure of the spectral database.

**Algorithm 1** Multi chromosome GP algorithm

```
Require: : random population $P[1 \dots N]$ with $N$ individual
Require: : $ps$: probability of selection, $pm$: probability of mutation
Ensure: : $ps + pm < 1$
procedure GP OPTIMIZATION
    $sel[1 \dots N] \leftarrow 0$
    $k \leftarrow 0$
    while stop criteria not met do
        for each $i \in P_k$ do
            $sel[i] \leftarrow$ ROULETTWHEEL($P_k[i]$)
        end for
        $Pc = selectPairs(P_k, ps)$      #Select pairs for crossover
        $Pm = select(P - Pc, pm)$      #Select invs for mutaation
        $Pu = P - Pc - Pm$             #Keep invs for direct reproduction
        for each $i \in Pm$ do
            $Pm[i] \leftarrow$ MUTATE($Pm[i]$)
        end for
        for each $i \in Pc$ do
            $Pc[i] \leftarrow$ CROSSOVER($Pc[i]$)
        end for
        $Pn = Pu \cup Pc \cup Pm$
        for each $i \in Pn$ do
            $fitness[i] \leftarrow$ EVALUATE($Pn[i]$)
        end for
    end while
end procedure

function EVALUATE(p)
    while $fit \neq optimal$ do
        $p \leftarrow$ OPTIMIZEPARAM($p$)
        $coords \leftarrow$ MAP($p$)
        $fit \leftarrow$ COSTFUNCTION($coords$)
    end while
    return $fit$
end function
```

Figure 2.10 shows the mappings defined by these aggregates called Naro and Parox. These aggregates were selected because this pair of equations - from the defined set in TOPNIR software framework - gives the best result related to our fitness function. The equations 2.16 and 2.17 show the mathematical expressions of these aggregates.

$$y_{Parox} = 550(x_{84}/(20x_{15} + x_{112}) - 0.0686) - 12.22 \tag{2.16}$$

$$y_{Naro} = 130((x_{112}/x_{119}) - 1.2462) + 55 \tag{2.17}$$

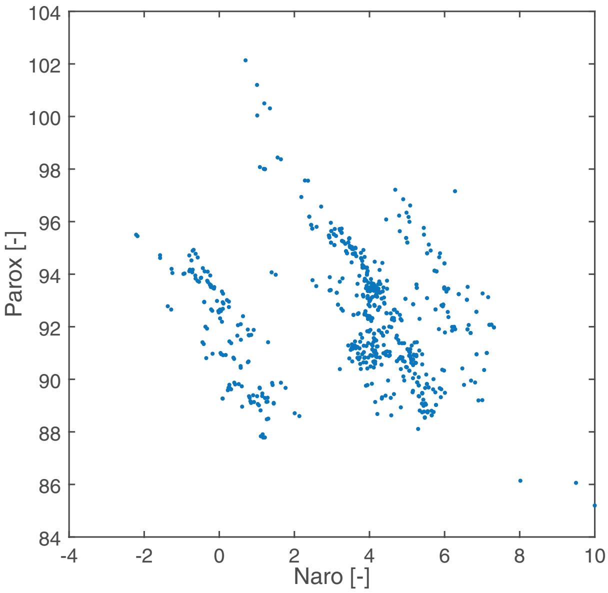

FIGURE 2.10. Best Pairing from Predefined Aggregates, $fitness = 0.9133$

As can be seen, among the $x_1 \ldots x_{195}$ possible input variables only four variables are used by these aggregates.

To obtain a much better model we applied GP to find an optimal pair for the Parox and also for the Naro aggregate. Based on our experiments we found that with the parameters given in Table 2.3 the GP can find a good solution for the problem. These parameters are the default parameters of our toolbox, and we have not modified them during the experiments presented in this thesis.

TABLE 2.3. Parameters of GP in the application examples

| Population size | 40 |
|---|---|
| Maximum number of evaluated individuals | 40000 |
| Type of selection | roulette-wheel |
| Type of mutation | point-mutation |
| Type of crossover | one-point (2 parents) |
| Type of replacement | elitist |
| Generation gap | 0.667 |
| Probability of crossover | 0.3 |
| Probability of mutation | 0.7 |

We applied genetic programming in three different cases. In the first and second cases we applied the one-chromosome based algorithm for looking for genetic pairs for predefined aggregates $Naro$ and $Parox$ separately.

We also applied Multi-Dimensional Scaling (MDS), Sammon projection and Principal Component Analysis (PCA) dimension reduction methods (see Section A.1) and compared the fitness values of these mappings (Figure 2.11. As this figure shows, the samples are separating into two different clusters belonging to summer and winter diesel fuels. We can also identify also a smaller third group, which represents the cluster of premium fuel samples.

The mappings not only separate these groups but also preserve the neighbourhood relations defined in the original spectral space. All of these methods have better performance than the best TOPNIR aggregate-pair. Although MDS and Sammon's mappings (see Section A.2) have good performance, the drawback of these methods is that these mappings have to be recalculated when new samples are coming into the dataset.

#### 2.4.1.2 Single-chromosome mapping - Naro

Firstly, we apply our algorithm in the single-chromosome mode to find an informative pair to the $Naro$ aggregate of the TOPNIR framework (see Equation 2.16) . In the single-chromosome mode, the first aggregate is fixed and only the second aggregate is optimised. Figure 2.12 shows the generated mapping with the parameters given in Table 2.3 after 1000 generation.

The equation of the generated aggregate (after mathematical simplification) is:

$$y_{Naro}^{G} = \frac{x_{37} + x_{108} + x_{176} - x_{86}}{2x_{169}x_{145}} \tag{2.18}$$

We can say that the pairing of equations 2.16 and 2.18 generates more suitable mapping related to the fitness function 2.15 than the original pairing of 2.16 and 2.17. As this figure shows, this mapping separates the groups of samples and defines the limits of local models of different product types (Figure 2.4).

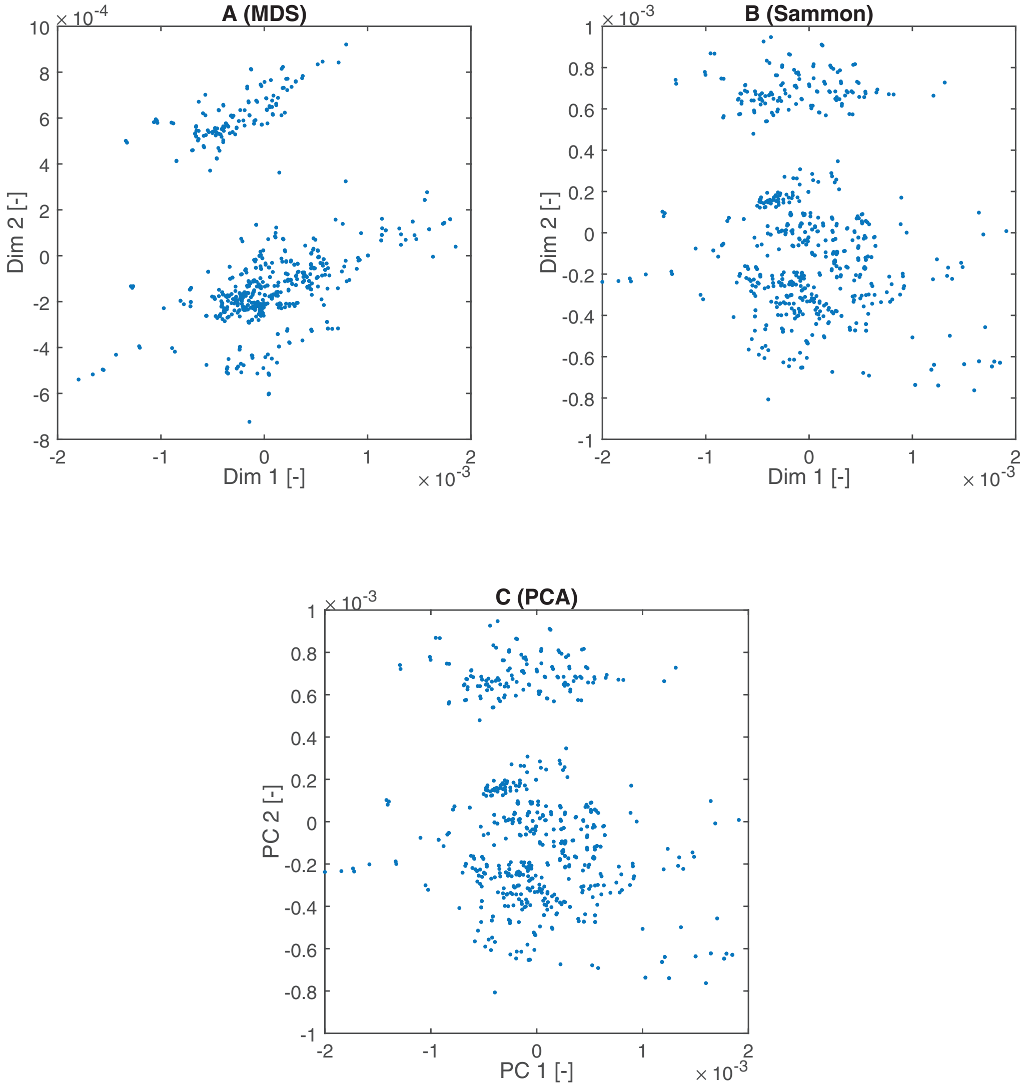

FIGURE 2.11. Mappings with standard methods,
(A) Non-classical Multi-Dimensional Scaling $fitness = 0.9196$,
(B) Sammon projection $fitness = 0.9625$,
(C) Principal Component Analysis with two Principal Component $fitness = 0.9192$

#### 2.4.1.3 Single-chromosome mapping - Parox

Similarly to the previous case study we also applied the single-chromosome algorithm to find the optimal pair of the Parox aggregate. The generated mapping can be seen on Figure 2.13.

The equation of the generated aggregate (after mathematical simplification) is:

$$y^G_{Parox} = 9\frac{x_{188}}{x_{70}} + 9\frac{x_{176}}{x_{84}} - 5\frac{x_{188}}{x_{70}} - 9\frac{x_{97}}{x_{156}} \qquad (2.19)$$

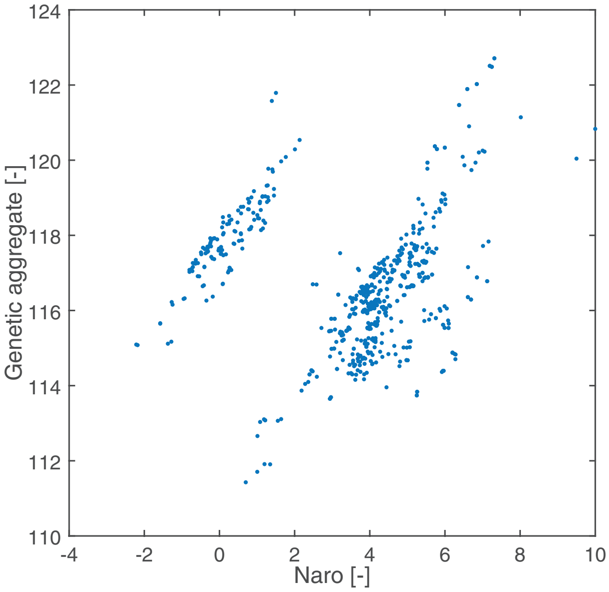

FIGURE 2.12. Single-Chromosome Result for the predefined *Naro* Aggregate. $fitness = 0.9470$

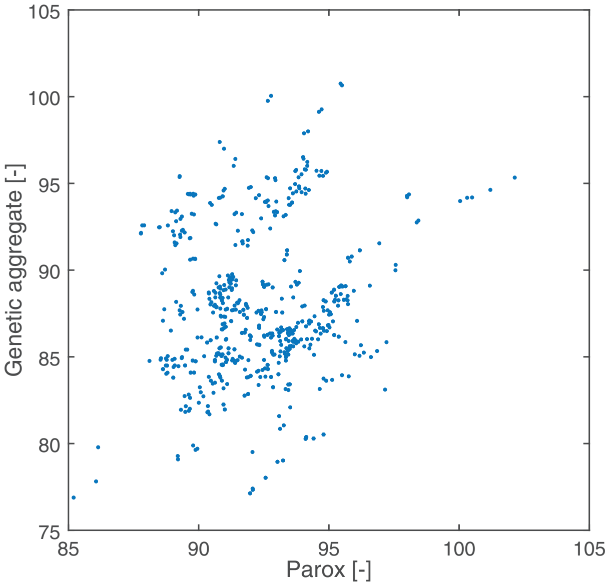

FIGURE 2.13. Single-Chromosome Result for the predefined *Parox* Aggregate. $fitness = 0.9473$

This mapping has better fitness value than the original pairing of Naro and Parox aggregates and the results obtained in the previous example.

Nevertheless, the sample groups are not separating on this mapping as were in the previous case. The usability of mappings depends on the aim of the application. The user has to supervise the process and review the results. When the result does not meet the requirements in some cases, it is worth to define a problem-specific cost function as it will be presented at the end of this session.

#### 2.4.1.4 Multi-chromosome mapping

In the third experiment, the two aggregates of the mapping are simultaneously optimised. The parameters were the same as were in the single-chromosome cases. In multi-chromosome mode, the algorithm generates a random population with aggregate pairs, and these pairs are managed together during execution. The stopping criterion was the same as in single-chromosome cases such as 1000 generation.

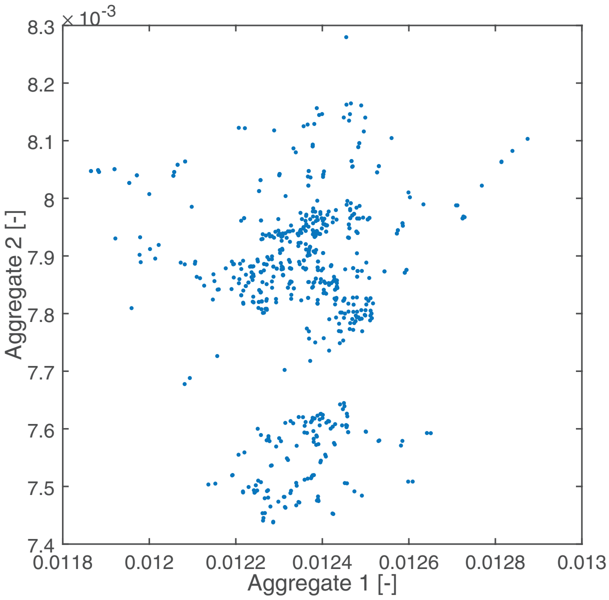

FIGURE 2.14. Multi-Chromosome Result without fixed aggregate. $fitness = 0.9686$

The generated aggregates (after mathematical simplification) are:

$$y_1^G = 2x_{160} - 3x_{56} + x_4 + 1x_{160} - x_{57} \tag{2.20}$$

$$y_2^G = x_{112} + 5x_{84} - 1x_{120} + x_{95}x_{165} - x_{120} + x_{133} \tag{2.21}$$

The equations are simple; they contain only a few variables and three basic operations. The fitness value is much higher than the standard mappings and single-chromosome optimizations. Since, the aggregates define explicit mapping new sample(s) can be easly visualied contrary to Sammon mapping where the projection of the complete dataset has to be recalculated when a new sample is added.

We can say that the generated equations give a proper mapping of the spectral database. The samples are separating into product groups and the neighbourhood preserving of the projection is better than any other methods or executions could provide.

#### 2.4.1.5 Discussion

Comparing the above results we can say that the algorithm can improve the explicit mapping of the spectral database.

Table 2.4. Fitness values of mapping using different methods.

| Projection | Fitness |
|---|---|
| Best Predefine Aggregate Pair | 0.9133 |
| Multidimensional Scaling | 0.9196 |
| Sammon Projection | 0.9625 |
| Principal Component Analysis | 0.9192 |
| Single-Chrom. Genetic (Naro) | 0.9470 |
| Single-Chrom. Genetic (Parox) | 0.9473 |
| Multi-Chromosome Genetic | 0,9686 |

Based on the results summarised in Table 2.4 we can conclude that the mappings of our genetic algorithm have better performance than standard dimensional reduction techniques.

The single-chromosome method can already improve an existing projection. This approach is useful when we would like put *aprori* knowledge into the model by defining at least one aggregate based on an existing model. Since we would like to visualise the spectral database to improve nonparametric regression models, the metric error of projection is less important than the topology preserving property of the mapping. This suggests that the proposed method can support several goal-oriented data visualisations by defining a cost function that reflects the expected use of the mapping. In the following session, such application example is given.

### 2.4.2 Application to classification problem

In this study we demonstrate how the proposed algorithm can be used to support the identification of classifier models. The examined a standard data mining benchmark dataset to present that our method is applicable for the improvement of other data mining tools like clustering and classification. Our analysis has two aspects. At first, we applied genetic programming based aggregate generation order to map the dataset into 2D to show the structure of the data. For visualisation, we used the same fitness function as it was in the case of the spectral database. The fitness function is defined in equation 2.15. In the second part of the analysis we utilised $c4.5$ decision tree based classification algorithm to improve its performance by using of aggregate based dimension reduction as a preprocessing step.

#### 2.4.2.1 Wine quality dataset

The wine dataset represents the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The initial data set had around 30 variables, but for reason that there ware missing values we only have a 13 dimensional version. The measured properties are the following:

Table 2.5. List of wine quality attributes.

| Number | Property |
|---|---|
| 1 | Alcohol |
| 2 | Malic acid |
| 3 | Ash |
| 4 | Alcalinity of ash |
| 5 | Magnesium |
| 6 | Total phenols |
| 7 | Flavanoids |
| 8 | Nonflavanoid phenols |
| 9 | Proanthocyanins |
| 10 | Color intensity |
| 11 | Hue |
| 12 | OD280/OD315 of diluted wines |
| 13 | Proline |

In a classification context, this is a well-posed problem with "well behaved" class structures. This is a good dataset for first testing of a new classifier. All attributes are continuous, and there are no missing values. The class labels are the cultivars. In the analysis, we applied topology (neighbourhood) preserving fitness function as well as the classification performance (ratio) fitness function.

#### 2.4.2.2 Visualization of wine database

To provide a reference for comparison we calculated standard mappings using the same methods as in the previous example. Fitness values are calculated for the $k = 10$ nearest neighbours using the Euclidean distance between samples. The attribute values are normalized into the range $[0, 1]$. It is necessary to avoid the effect the magnitudes. Figure 2.15 shows the projection generated by the three standard methods.

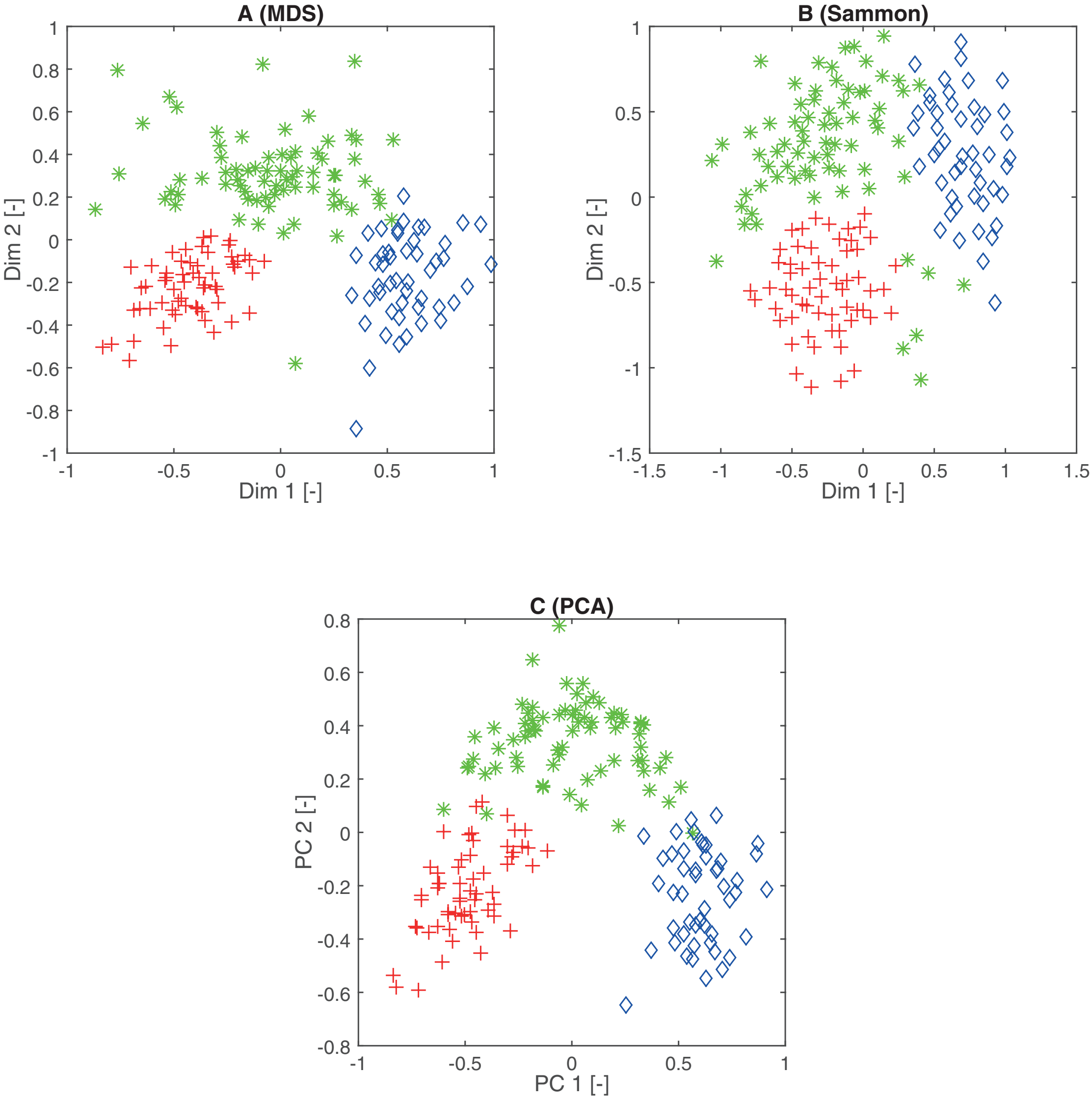

FIGURE 2.15. Mappings with standard methods,
(A) Non-classical Multi Dimensional Scaling $fitness = 0.8673$,
(B) Sammon projection $fitness = 0.8547$,
(C) Principal Component Analysis with two Principal Component $fitness = 0.8468$

The Table 2.6 shows the fitness values of the mappings. Considering the results, we can say that in this application Multi-Dimensional Scaling gives the better performance with metric stress cost function. Shammon projection and PCA have lower performance. To evaluate the genetic programming generated mapping we use the best of the standard methods' values, which is 0.8673.

Unlike the NIR spectral dataset in this application, we can apply only the multi-chromosome method and can generate new aggregates in pairs because literature does not define any aggregates for this dataset. Properties of the GP algorithm are the same as they were in the case of the spectral database given in Table 2.3. Stopping criteria is also the same, 1000 generation. Figure 2.16 shows the mapping using the aggregates that are generated by GP algorithm.

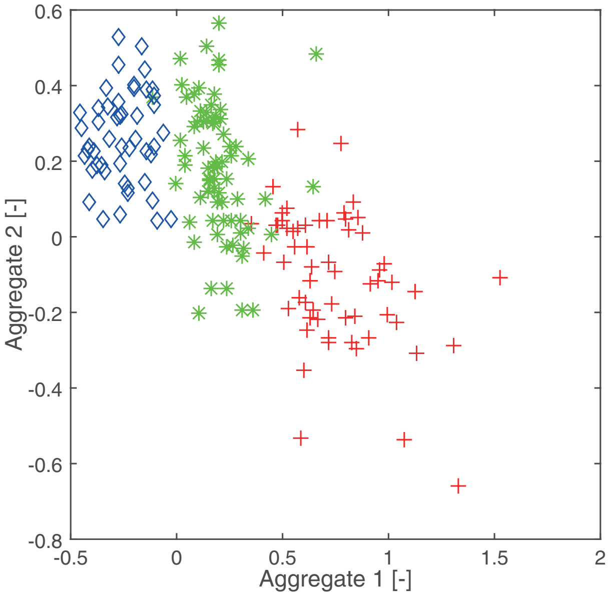

FIGURE 2.16. Multi-Chromosome Result
$fitness = 0.8730$

The equations of generated aggregates (after mathematical simplification) are:

$$y_1^G = 1.5x_{10}x_{12} + x_7 - 1.5x_{10} + 2.5x_{13}x_7\left(x_3 + x_1\right) \tag{2.22}$$

$$y_2^G = x_8 - x_9x_9 - 1.5\left(\left(x_1 - x_4\right)x_9\right) \tag{2.23}$$

According to mathematical and visual evaluation we can say that the aggregate based mapping has better performance than the standard methods. It has the highest fitness value and the groups of samples are separating on the 2D plain. From the user's point of view the second ascertainment is more important. The user should be able to differentiate samples and this 2D mapping gives a good support.

TABLE 2.6. Fitness values of different mapping methods in case of wine database.

| Projection | Fitness |
|---|---|
| Multi Dimensional Scaling | 0.8673 |
| Pricipal Component Analysis | 0.8468 |
| Sammon Projection | 0.8547 |
| Multi-Chromosome Genetic | 0.8730 |

#### 2.4.2.3 Supporting classification using aggregate based mapping

In this example, we used the aggregate based dimension reduction to design a classification algorithm and improve it's performance. The aim of this experiment is to demonstrate that our GP algorithm is applicable not only for visualisation but also for data preprocessing or feature transformation. To present this functionality of the GP, we applied a simple decision tree based classifier named C4.5 [88, 89]. A decision tree is a classifier which conducts recursive partition over the instance space. A typical decision tree is composed of internal nodes, edges and leaf nodes. Each internal node is called decision node representing a test on an attribute or a subset of attributes. Each edge is labelled with a specific value or range of value of the input attributes. In this way, internal nodes associated with their edges split the instance space into two or more partitions. Each leaf node is a terminal node of the tree with a class label. In this example, we have 13 splitting attributes (properties of wine), along with three class labels (cultivars).

We used the default properties without any change. The fitness function in this application was the ratio of the matching classifications. To compare the results of different methods we calculated the confusion matrix $C$ for classification. The form of $C$ is given in the Equation (2.24). In the diagonal of matrix are the numbers of correctly classified samples (denoted by $T$). The other matrix elements contains the number of falsely classified samples (denoted by $F$).

$$C = \begin{bmatrix} C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ C_{2,1} & C_{2,2} & \cdots & C_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n,1} & C_{n,2} & \cdots & C_{n,n} \end{bmatrix} \tag{2.24}$$

Let $N_s$ is the number of samples in the dataset and $N_c$ is the number of samples that are classified correctly.

$$N_c = \sum_{i \neq j} C_{i,j} \tag{2.25}$$

$$N_s = \sum_{i,j} C_{i,j} \tag{2.26}$$

For the genetic algorithm the fitness function is defined in the Equation (2.27).

$$fitness = \frac{N_c}{N_s} \tag{2.27}$$

The program flow is analogous to the Algorithm 1, except the inner *evaluation* function. The classification related application needs a special inner loop (C4.5 classification) which is detailed in Algorithm 2.

**Algorithm 2** Embedded clustering based fitness function with C4.5

```
Require: : $c_{ref}$ Class labels
function EVALUATE(p)
    while $fit \neq optimal$ do
        $p \leftarrow$ OPTIMIZEPARAM($p$)
        $coords \leftarrow$ MAP($p$)
        $mod \leftarrow$ C4.5($coords$)
        $c_{mod} \leftarrow$ CLASSIFY($coords$)
        $N_c \leftarrow$ COUNT($c_{mod} == c_{ref}$)
        $fit \leftarrow \frac{N_c}{N_s}$
    end while
    return $fit$
end function
```

- OPTIMIZEPARAM(p) is the inner loop call of parameter optimization algorithm. In our case the *pattersearch* method was used from MATLAB globa optimization toolbox.

- MAP(p) is the evaluation of the generated aggregate. This function calculates the two coordinate for each data point and gives a $N$ matrix.
- C4.5(coords) function teaches a classification model.
- COUNT call calculates the number of correctly classified samples

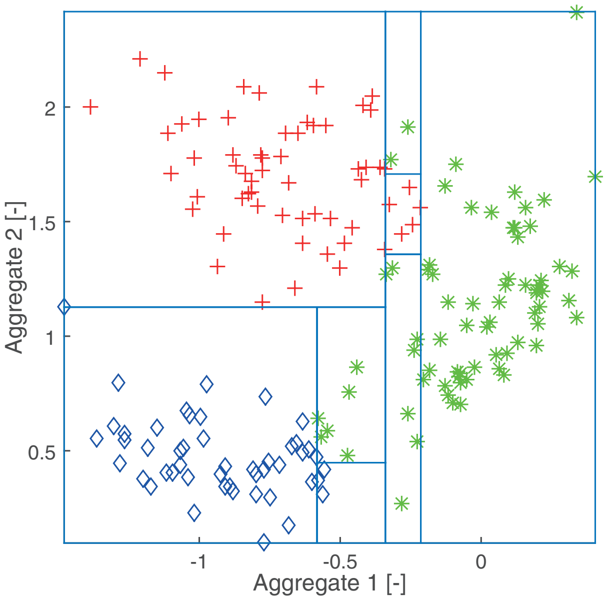

FIGURE 2.17. Multi-Chromosome of wine database and classification using C4.5 algorithm. The matching ratio on 2D plain is 0.9888%

For reference, we executed several standard classification algorithms, namely Linear Discriminant Analysis (LDA) [128], Quadratic Discriminant Analysis (QDA) [129] and C4.5 algorithm in the original variable space.

In the first step we executed the C4.5 algorithm with all attributes. In this case, the fitness value is 0.9213. After this experiment, we executed the GP algorithm, and we used the generated aggregate pair as new virtual attributes. In this step, the input of C4.5 algorithm consists these two new attributes. Properties of the GP ware the same as they ware in the case of the spectral database and wine dataset visualisation. The stopping criteria ware 1000 generation or the 100% correct classification or at least 50 generation without performance improvement. For wine dataset, the third criterion was reached after 231 generation. We used 10 times cross validation where the training and test subsets ware selected and fixed in the first step of execution.

Figure 2.17 shows the mapping result of this experiment. As it can be seen, the classification fitness is 98%. We can say that a proper dimension reduction can improve the performance of a standard classification algorithm.

The equations of generated aggregates (after mathematical simplification) are:

$$y_1^G = x_{12} + x_7 - x_{13} - 2x_{10} - x_{13} + x_1 \tag{2.28}$$

$$y_2^G = 3x_7 + x_5 + x_{12} + (x_1 + x_6)\,x_9 \tag{2.29}$$

TABLE 2.7. Results of different classification algorithms. The lines 1-3 show the confusion matrixes and fitness values of standard algorithms. The last line contains the results of our genetic algorithm where the embedded classifier ware the C4.5

| Method | Confusion matrix | Fitness value |
|---|---|---|
| Linear discriminant analysis | $\begin{bmatrix} 59 & 0 & 0 \\ 1 & 68 & 2 \\ 0 & 0 & 48 \end{bmatrix}$ | 0.9831 |
| Quadratic discriminant analysis | $\begin{bmatrix} 58 & 1 & 0 \\ 1 & 70 & 0 \\ 0 & 0 & 48 \end{bmatrix}$ | 0.9888 |
| C4.5 in original space | $\begin{bmatrix} 57 & 2 & 0 \\ 2 & 65 & 4 \\ 1 & 5 & 42 \end{bmatrix}$ | 0.9213 |
| C4.5 in transformed space | $\begin{bmatrix} 59 & 0 & 0 \\ 0 & 69 & 2 \\ 0 & 0 & 48 \end{bmatrix}$ | 0.9888 |

Based on the results shown in the Table 2.7, we can say that the data preprocessing with our genetic algorithm can improve the performance of standard classifiers since the worst performing C4.5 could be improved to have the same performance like the QDA which was the best performer of the reference algorithms.

## 2.5 Conclusions

Visualisation of high-dimensional data is an important task in process monitoring. In this chapter, we presented a genetic programming based algorithm that generates nonlinear functions to feature selection and transformation. We defined a novel cost function that relies on the topology preserving the property of the mappings. The resulted tool was applied to design new aggregates used for the visualisation of high-dimensional spectroscopic databases. The results illustrated that the algorithm can generate compact and accurate mappings having better performance than PCA, MDS, Shammon projections or classical aggregate based models.

Furthermore, we also applied the GP-based method to support a standard classification tool (C4.5) to improve its performance. We provided a transformed feature set for the classifier, and we demonstrated that the classification performance increased.

We can conclude that the developed method can map high-dimensional data into lower dimensional space with satisfying a properly defined fitness function. Application of our GP algorithm is limited by the formulation of these fitness functions and computational resources. In the presented examples, the running time were about 72 hours. Further development opportunities are the extension the algorithm to support other types of chromosomes, and rewriting the code in a more efficient programming language like C/C++ and the utilization the concept of parallel computing.

# Chapter 3

# Parametric model development

In Chapter 2 we presented novel methodologies that can support non-parametric model development or can act as individual modelling approaches. In this chapter we focus on parametric models and because most of the industrial solutions rely on these.

## 3.1 Feature transformation based modelling for prediction and visualisation

In this section, we show a novel approach for prediction and visualisation by using the feature transformation functionality of Partial Least Squares regression (PLS) [93]. The dataset - that we used for the demonstration of our technique is described in Section 2.4.1. The PLS model is applied to estimate the cold filter plugging point, density and one property of distillation. For monitoring purposes, the latent space of the PLS model is used. A special orthogonalisation algorithm was applied that can visualise the data and give information about the distribution of the operating regimes and the model's quality.

### 3.1.1 PLS Concept

PLS is a method for constructing of predictive models from numerous and correlated input variables [92]. PLS was developed in the 1960s by Herman Wold as an econometric technique, but it became soon a widely applied tool in chemical engineering [93]. In addition to spectrometric calibration, PLS is often applied to monitor and control industrial processes; since complex processes can easily have hundreds of variables [90]. PLS finds the multidimensional direction in the **X** space of the input variables that explains the maximum multidimensional variance direction in the **Y** space of the output variables. PLS regression is particularly suited when the matrix of predictors has more variables than observations and when there is multicollinearity among **X** variables. By contrast, standard regression cannot be applied in these cases.

The general underlying model of multivariate PLS [133] is

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T + \mathbf{E} \tag{3.1}$$

$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^T + \mathbf{F} \tag{3.2}$$

Let $n$ is the number of samples, $m$ is the number of input variables, $p$ is the number of output variables and $l$ is the number of of dimensions in the latent space. The **X** is an $n \times m$ matrix of predictors, **Y** is an $n \times p$ matrix of responses; **T** and **U** are $n \times l$ matrices that are, respectively, projections of **X** (the X score, component or factor matrix) and projections of **Y** (the Y scores); **P** and **Q** are, respectively, $m \times l$ and $p \times l$ orthogonal loading matrices; and matrices **E** and **F** are the error terms, assumed to be i.i.d. normal. The decompositions of **X** and **Y** are made so as to maximize the covariance of **T** and **U**.

### 3.1.2 2D PLS based visualization

For the two-dimensional visualisation of the PLS model we applied the algorithm that is developed by Rolf Ergon and is described in ref. [90]. In this subsection only the most important details of this technique are summarised.

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

The basic idea behind the applied mapping is illustrated in Figure 3.1. The estimator $\hat{\mathbf{b}}$ is found in the space that is spanned by the loading weight vectors in

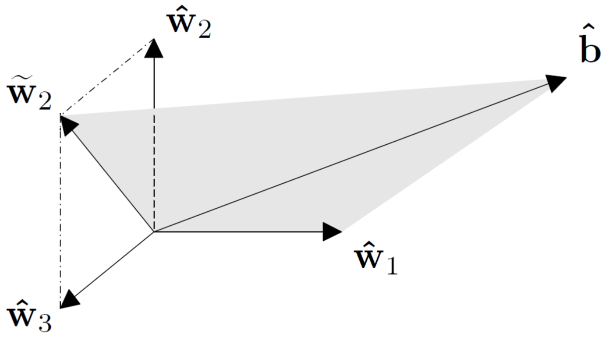

FIGURE 3.1. Generic graphical representation of the 2D compression of the PLS predictors.

$\hat{\mathbf{W}} = [\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \ldots, \hat{\mathbf{w}}_A]$, where $A$ is the number of dimensions in the PLS latent space. i.e. it is a linear combination of these vectors. It is also found in the plane defined by $\hat{\mathbf{w}}_1$ and a vector $\tilde{\mathbf{w}}_2$ orthogonal to $\hat{\mathbf{w}}_1$, which is a linear combination of the vectors $\hat{\mathbf{w}}_2, \hat{\mathbf{w}}_3, \ldots, \hat{\mathbf{w}}_A$.

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

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

### 3.1.3 Prediction of product properties

The presented research focuses on two tasks. The first task is the development of a prediction model that can estimate product properties based on spectra taken by online NIR analysers. The second task is the development of a monitoring tool that relies on the visualisation of the same spectra [94].

We present our methodology on two different NIR datasets. The first one ( "$DS_1$") is described in Section 2.4.1. The second data set ( "$DS_2$") consists of 67 samples collected from a different process which is a laboratory scale experimental reactor in the Duna Refinery of MOL Ltd. The prediction performances of the models are measured by the correlation coefficient defined in the equation (4.7). All the presented algorithms have been implemented in MATLAB.

Firstly the effect of the dimensionality of PLS model's latent space has been analysed (from 2 to 48 dimensions). To valiadate the model leave-one-out and 10-fold cross validation techniques were applied. On Figure 3.2 the performances (correlation coefficients) [95] of the PLS models are shown.

As Figure 3.2 shows, the accuracy of the model increases rapidly by increasing the dimensionality of the latent space from 2 to 6 dimensions. It should be noted that the model correlation has a maximum and it decreases when the complexity of the model is higher than the complexity of the modelled system.

Table 3.1. Effect of the number of latent variables to the performance of the model (correlation between the estimated and measured variables are shown).

| | Latent dimensions | | | | | |
|---|---|---|---|---|---|---|
| Property | 2 | 6 | 12 | 18 | 24 | 48 |
| Density | 0.776 | 0.988 | 0.993 | 0.993 | 0.993 | 0.989 |
| CI | 0.130 | 0.204 | 0.190 | 0.420 | 0.344 | 0.272 |
| 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 |
| CloudPt | 0.668 | 0.924 | 0.950 | 0.958 | 0.955 | 0.943 |
| FlashPt | 0.408 | 0.596 | 0.878 | 0.901 | 0.895 | 0.854 |
| T10 | 0.428 | 0.732 | 0.908 | 0.946 | 0.941 | 0.938 |
| T50 | 0.694 | 0.922 | 0.970 | 0.971 | 0.957 | 0.910 |
| T90 | 0.432 | 0.654 | 0.849 | 0.895 | 0.868 | 0.796 |
| E250 | 0.660 | 0.879 | 0.955 | 0.954 | 0.927 | 0.904 |
| E350 | 0.044 | 0.077 | 0.308 | 0.259 | 0.174 | 0.006 |
| E360 | 0.115 | 0.374 | 0.431 | 0.397 | 0.341 | 0.190 |
| PolyCycl | 0.169 | 0.377 | 0.429 | 0.441 | 0.434 | 0.381 |
| TotAro | 0.765 | 0.885 | 0.905 | 0.880 | 0.862 | 0.771 |
| VISC | 0.898 | 0.991 | 0.999 | 0.999 | 0.999 | 0.999 |

The Table 3.1 and Fig. 3.2 show that the complexity of the best performer model varies according to the estimated property. For example the best performer model of T50 property has a 24 dim latent field while this value is 18 for E250. We have utilized the Sum of Ranking Differences (SRD)[130] algorithm to select a common model which has an overall good performance for all the features. SRD is a fast and general method that compares alternative solutions to the same problem. We

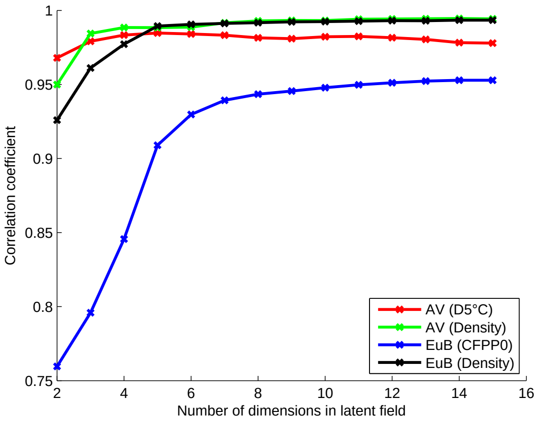

Figure 3.2. Effect of PLS latent field's dimensionality

have multiple PLS models to compare with different complexity that estimate the same material properties. SRD takes the matrix of correlations where the rows are containing different properties and the columns are models with increasing complexity.

We have executed the SRD algorithm [130] with two different ideal ranking objective. In the first execution we used the maximum correlation of a model as objective. Figure 3.3 shows that the PLS model with 12 latent dimensions gives an overall good performance we take all the properties into account.

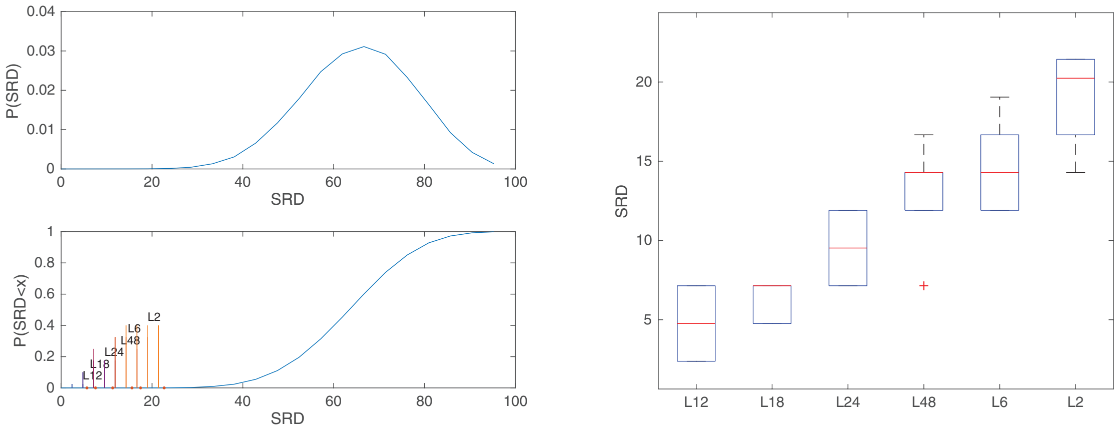

Figure 3.3. SRD results with maximum ranking objective

### 3.1.4 Visualization of operating regimes

In section 3.1.2 a unique method was presented that can map the PLS latent space into two-dimensional space by two component orthogonal combination of

PLS predictors. Here we show how the 2D PLS could be used to visualise the operating regimes of the modelled process. We compare the method with Principal Component Analysis and Topological Near-Infrared Modelling, which was detailed in Chapter 2.

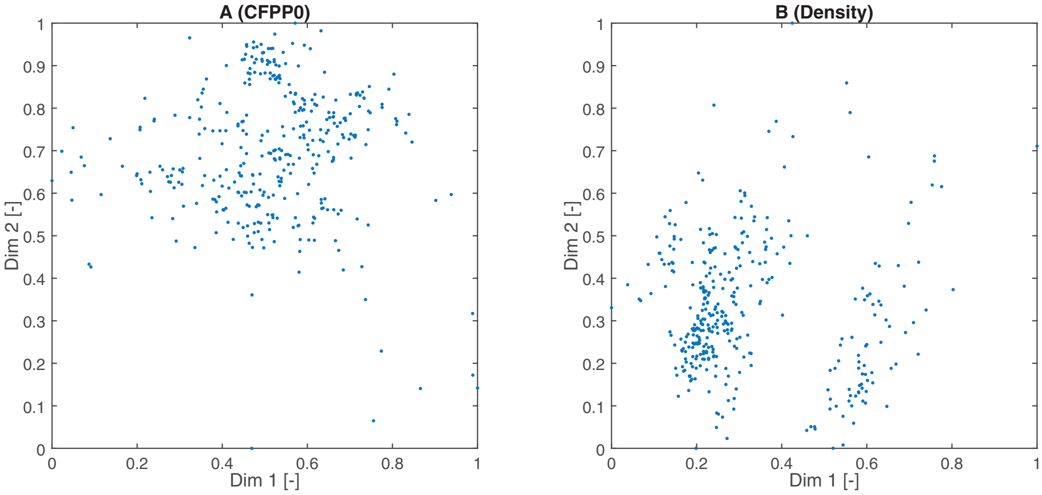

FIGURE 3.4. Visualization of Diesel Blending spectral database using 2D PLS compression.
(A) using PLS regression for Cloud Filter Plugging Point property (CFPP)
(B) using PLS regression for Density property

Figure 2.11 C shows the mapping of PCA with the first two principal components [96, 132]. We can say that PCA can separate the operating regimes similarly to the topological models as it was presented in Chapter 2.

Results of 2D PLS can be seen on Figure 3.4 A and B. The PLS model is more informative since it also utilises the output variables for the mapping. Figure 3.4 A shows the mapping using the Cloud Filter Plugging Point as the estimated property. Comparing this mapping with the mapping of obtained using Density (see Figure 3.4 B) one can easily see that the operating regimes have much more impact on the density than to the CFPP.

As it can be seen PLS correctly reflects the operating regions and can detect more effectively the outliers than the aggregate based mappings.

In the second part of the case study we demonstrate how the outlier samples can be identified in the mapped space. As it can be seen on the Figure 3.5 the $DS_2$ contains two samples which are far from the normal operational range (top right corner). The aggregate based mapping can not identify these samples exactly; it finds only one outlier of the two.

As we show on Figures 3.6 A and B the 2D PLS gives detailed information for the outlier detection. Comparing these plots and PCA (Figure 3.5) we can conclude that the 2PLS technique is the most efficient to detect outliers in the spectral or the property space.

Such analysis gives information to the user not only about operating regimes but also about quality of the models; the presented mapping can give hints the modeller how to enhance model performance by the proper selection of the training data.

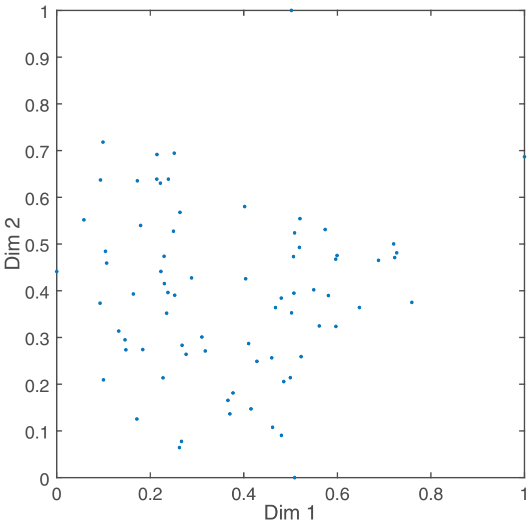

Figure 3.5. Visualization of Distillation spectral database by PCA

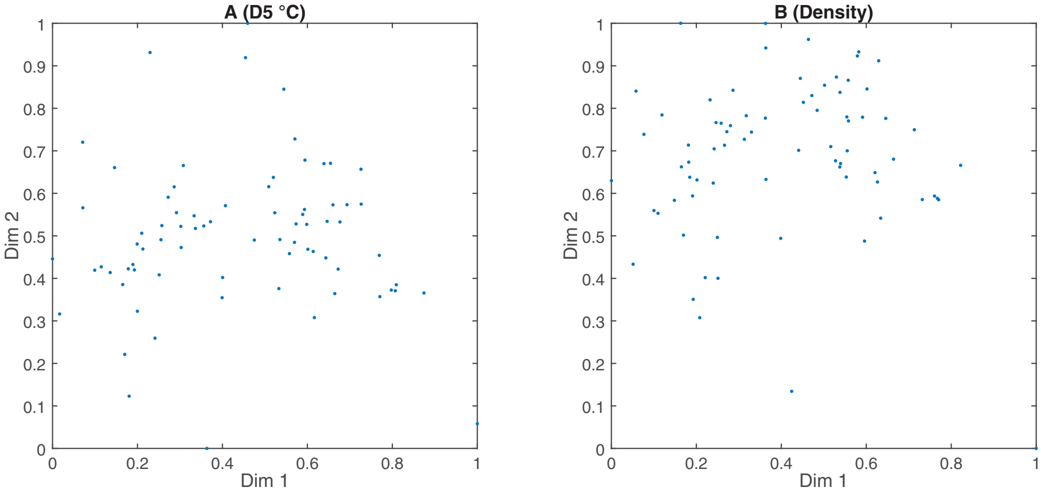

Figure 3.6. Visualization of Distillation spectral database using 2D PLS compression.
(A) using PLS regression for D5 distillation property
(B) using PLS regression for Density property

### 3.1.5 Conclusion

On-line analysers use indirect measurement combined with a prediction model to support process control and monitoring. Several multivariate models and methods can help the prediction of product properties based on NIR spectra. Model development cannot be a fully automated, human supervision and intervention is always needed. We have proven that it is very informative to visualise the hidden structure of the complex spectral database in a low-dimensional space because it could support the model development. Industrial applications require easily implementable, interpretable and accurate projection. TOPNIR utilises nonlinear heuristic functions (aggregates) for the mapping of spectra as a high-dimensional object. We proposed a much more sophisticated approach that can be used simultaneously for prediction and visualisation. We adapted a technique that allows the application of PLS also for visualisation of a spectral database.

Datasets taken from the Duna Refinery of MOL Ltd were analysed. The PLS model is applied to estimate cold filter plugging point, density and one property (E250) of distillation. The main benefit of this technique is that it allows us to add extrapolation functionality to calculate product properties of samples that are out of the known operating region. The proposed PLS based model can simultaneously predict the unmeasured material properties and monitor the state of the process. The monitoring is realised in orthogonal two-dimensional plots. These plots can also be used for the efficient identification of outliers.

## 3.2 Feature Selection Based Root Cause Analysis for Energy Monitoring and Targeting

Energy Monitoring (EM) systems are based on the monitoring of the difference between targeted and measured energy consumption. Data-driven dynamic targeting models can be used to estimate values of key energy indicators (KEI). In some cases it is difficult to determine, which process variables influence the KEIs. We developed an automated root cause analysis (RCA) technique to find the most important driving factors of the energy efficiency. The proposed concept is based on the application of feature selection algorithms. We applied Orthogonal Least Squares (OLS) and Random Forest Regression (RFR) to find a proper set of the input variables for the targeting models. The concept of the resulted energy monitoring system is applied at the Duna Refinery of MOL Hungarian Oil and Gas Company. The basic concepts of energy monitoring are described in Section 1.3.2.

In some cases, it is difficult to determine which process variables influence the KEIs. In these situations, the input variables of the targeting models should be selected based on root cause analysis of the operation. Unfortunately, this procedure is subjective, time-consuming and the good prediction performance is also not guaranteed.

Root Cause Analysis (RCA) is a method of problem-solving that identifies the root causes of faults and problems. We applied the RCA approach to find the driving factors of energy efficiency of process plants. There are many ways to implement RCA. For example, Bayesian networks can be applied to determine the root causes of deviations during the operation of complex processes [99]. Digraph models were proven to be useful to identify discrete events (faults) [101]. Multivariate statistical process monitoring (MSPM) with some extensions is a useful technique to isolate not only the effects of the faults but also the underlying causes. For this purpose, MSPM and "fuzzy-signed directed graphs" were combined to identify the root causes [102]. These methods have developed for discrete event systems. Building energy monitoring models requires knowing the driving factors of the energy efficiency [138]. The techniques as mentioned above are designed to analyse discrete events and do not manage continuous process variables. To support root cause analysis of energy efficiency, we proposed a fully automated feature selection based approach.

The concept of the resulted energy monitoring system is applied at the AV2 unit of the Duna Refinery of MOL Hungarian Oil and Gas Company. The Key Energy Indicators were calculated based on one-year historical data because we assumed that the range of this dataset is broad enough to cover operation ranges with high and low energy consumption periods and contains information about the significant malfunctions. The results show that the proposed approach can determine useful and informative sets of driving factors having a large impact on the energy efficiency.

### 3.2.1 Targeting model based energy monitoring

Activity-based energy targets are usually calculated by linear regression models,

$$\hat{y}_k = (\mathbf{x}_k, \theta) = \left[\mathbf{x}_k^T 1\right] \theta \tag{3.3}$$

where the calculated output $\hat{y}_i$ is the linear combination of process variables (drivers), $x_k = [x_{1,k}, \ldots, x_{n,k}]$ , where $k$ represents the $k$-th sampling time and $n$ stands for the number of process variables could have effect to the energy consumption. At the development of this model it is important to ensure that data are synchronised as closely as possible with the required assessment intervals. Based on a synchronized set of data $\{y_k, x_k\}, k = 1, \ldots, N$ linear least squares method can be applied to find optimal parameters of the model $\theta$ that minimizes the $\sum (y_k - \hat{y}_k)^2$ quadratic cost function.

$$\theta = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{3.4}$$

where $\mathbf{X}$ is $n \times N$ matrix of historical process variables and $\mathbf{y}$ is an $N \times 1$ vector of measured output variable (energy consumption or efficiency measure). When the predicted consumption $\hat{y}_k$ is higher than the measured value $y_i$; the technology is considered to be efficient regarding historical data. The relation $\hat{y}_k < y_k$ suggests that the technology could work with lower energy consumption.

### 3.2.2 Orthogonal least squares based feature selection

The performance of data-driven targeting models depend on a complex set of process variables. When no proper *a priori* knowledge is available for the selection of a KEI model's driving factors, feature selection algorithms can be used for sophisticated and automated root cause analysis.

The OLS algorithm is an effective tool to determine, which terms are significant in a linear-in-parameters model since it is based on the error reduction ratio ($err$) which is a measure of the decrease in the variance of output by a given term. In the following the details of this algorithm are presented. The compact matrix corresponding to the linear-in-parameters model is $\mathbf{y} = \mathbf{X}\theta + \mathbf{e}$, where the $\mathbf{X}$ is the regression matrix, $\theta$ is the parameter vector, $\mathbf{e}$ is the error vector. The OLS technique transforms the columns of the $\mathbf{X}$ matrix into a set of orthogonal basis vectors to inspect the individual contributions of each term.

The OLS algorithm decomposes the regression matrix $\mathbf{X}$ orthogonally as $\mathbf{X} = \mathbf{WA}$, where $\mathbf{A}$ is an $n \times n$ upper triangular matrix (it means $A_{i,j} = 0$ if $i > j$) and $\mathbf{W}$ is an $N \times n$ matrix with orthogonal columns in the sense that $\mathbf{W}^T\mathbf{W} = \mathbf{D}$ is a diagonal matrix. ($N$ is the length of $\mathbf{y}$ vector and $n$ is the number of regressors.) After this decomposition one can calculate the OLS auxiliary parameter vector $\mathbf{g}$ as

$$\mathbf{g} = \mathbf{D}^{-1}\mathbf{W}^T\mathbf{y} \tag{3.5}$$

where $g_i$ is the corresponding element of the OLS solution vector. The output variance ($\mathbf{y}^T\mathbf{y}$) can be explained as

$$\mathbf{y}^T\mathbf{y} = \sum_{i=1}^{M} g_i^2 w_i^T w_i + \mathbf{e}^T\mathbf{e} \tag{3.6}$$

The error reduction ratio, $[err]_i$ of the $i$-th input variable can be expressed as

$$[err]_i = \frac{g_i^2 w_i^T w_i}{\mathbf{y}^T\mathbf{y}} \tag{3.7}$$

This ratio offers a simple mean to order and select the model terms of a linear-in-parameters model according to their contribution to the performance of the model.

### 3.2.3 Random forest regression based feature selection

The drawback of OLS is that it assumes a linear relationship between the inputs and the output. Regression trees are simple, transparent and easily interpretable nonlinear models. The combination of these trees results in a forest of these models. When the regression trees are statistically independent, the average of these models' prediction will be better than the prediction of the individual models. Furthermore, the analysis of the forest can be used to select the most important process variables. In the following the theoretical background of this technique will be presented.

The concept of random forest was developed by Leo Breiman [103]. Andy Liaw implemented Breiman's concept in R. We used the MATLAB hosted version of this R package. The method combines Breiman's "bagging" idea and the random selection of features. Random forests for regression are formed by growing trees depending on random matrix $\mathbf{\Theta}$. The $\mathbf{\Theta}$ consist of a number of independent random integers between 1 and $M$, where $M$ is the number of trees in the forest. The nature and dimensionality of $\mathbf{\Theta}$ depends on it's use in the tree construction. A random forest is a predictor consisting a collection of tree-structured predictors $\{h_i(x, \mathbf{\Theta}_i), i = 1 \ldots\}$ where the $\mathbf{\Theta}_k$ are independent identically distributed random vectors and each tree cast a unique estimation for output $\hat{y}$ at input $\mathbf{x}$. The output values are numerical and we assume that the training set is independently drawn from the distribution of the given $\mathbf{y}, \mathbf{X}$ dataset. The mean-squared generalization error for any numerical predictor $h_i(\mathbf{X}) = h(X, \mathbf{\Theta}_i)$ is

$$E_{X,y}\left((y - h_i(X))^2\right) \tag{3.8}$$

where $E_{\mathbf{X},\mathbf{y}}$ denotes the expected value and $(y - h(X))^2 = (y - h(X))^T (y - h(X))$, and we use this substitution in the following. The random forest predictor is formed by taking the average over $M$ of the trees $\{h(x, \mathbf{\Theta}_i)\}$. Use of the proof [103] of Almost Sure Convergence theorem, as the number of the trees in the forest goes to infinity, mean-squared generalization error goes to a limit value almost surely as:

$$E_{\mathbf{X},\mathbf{y}}\left(\left(\mathbf{y}-\frac{\sum_{i=1}^{M}h\left(\mathbf{X},\mathbf{\Theta}_i\right)}{M}\right)^2\right)\rightarrow E_{\mathbf{X},\mathbf{y}}\left(\left(\mathbf{y}-E_{\mathbf{\Theta}}\left(h\left(\mathbf{X},\mathbf{\Theta}\right)\right)\right)^2\right) \tag{3.9}$$

Denote the right hand side (limit value) of 3.9 as $PE^*(forest)$ - the generalization error of the forest. Define the average generalization error of a tree as:

$$PE^*\left(tree\right)=E_{\mathbf{\Theta}}\left(E_{\mathbf{X},\mathbf{y}}\left(\left(\mathbf{y}-h\left(\mathbf{X},\mathbf{\Theta}\right)\right)^2\right)\right) \tag{3.10}$$

The concept is based on the fact $PE^*(forest) < \bar{\rho}PE^*(tree)$, where $\bar{\rho}$ (3.11) is the mean value of the correlation between the residuals $(\mathbf{y}-h\left(\mathbf{X},\mathbf{\Theta}\right))$ and $(\mathbf{y}-h\left(\mathbf{X},\mathbf{\Theta}'\right))$, where $\mathbf{\Theta}$, $\mathbf{\Theta}'$ are independent.

$$\bar{\rho}=\frac{E_{\mathbf{\Theta}}\left(E_{\mathbf{\Theta}'}(\rho(\mathbf{\Theta},\mathbf{\Theta}')S(\mathbf{\Theta})S(\mathbf{\Theta}'))\right)}{E_{\mathbf{\Theta}\mathbf{\Theta}'}(S(\mathbf{\Theta})S(\mathbf{\Theta}'))} \tag{3.11}$$

where $S\left(\mathbf{\Theta}\right)=\sqrt{E_{\mathbf{X},\mathbf{y}}\left(\left(\mathbf{y}-h\left(\mathbf{X},\mathbf{\Theta}\right)\right)^2\right)}$ is the standard deviation of prediction errors.

To obtain accurate regression forest, this theorem requires a low correlation between residuals and low error trees. The random forest decreases the average error of the trees employed by the factor $\bar{\rho}$. The randomization employed needs to aim at low correlation [103].

To rank the process variables and select a proper subset we used the importance measures which are defined in the following way. The first measure is computed from a random permutation of the data: For each tree, the prediction error (MSE) is recorded. Then the same is done after permutation of predictor variables. The difference between the two is then averaged over the trees and normalised by the standard deviation of differences. If the standard deviation of the differences is equal to 0 for a variable, the division is not done (but the average is almost always equal to 0 in that case) [103]. The second measure is the total decrease in node impurities from splitting on a variable. We used the first common form [131] of impurity, which is measured by the residual sum of squares and averaged over all the trees.

### 3.2.4 OLS based feature selection on fuel gas consumption

The proposed technique is applied to support the targeting model development project of the MOL Hungarian Oil and Gas Company. In this thesis results related to two Key Energy Indicators (KEIs) of the AV2 plant are presented. The applicability of the orthogonal least squares based feature selection is demonstrated on the total fuel gas consumption of the AV2 plant's furnaces, while the random forest based feature selection is applied to model the plant-wide electric power consumption of AV2.

The OLS model was used to find the most relevant variables influencing the gas consumption of the furnaces among 620 historical process variables. Figure 3.7 shows how the accuracy of the model increases by adding more and more input variables. The variables are introduced to the model by the decreasing series of relevance given by OLS. The model performance is measured by the model correlation ($R^2$). Figure 3.7 shows that the model's performance which is built using the first two most relevant variables has already $R^2 = 0.88$ correlation.

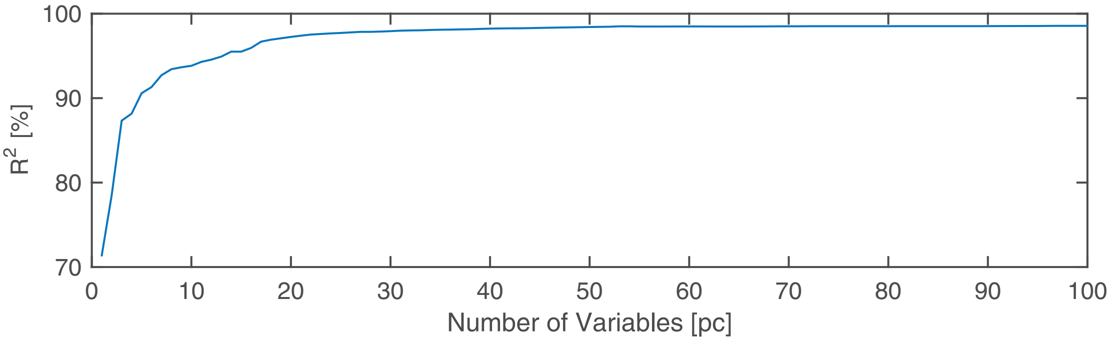

FIGURE 3.7. Model accuracy for fuel gas consumption in function of the increasing number of the relevant input variables

The first five variables - that are selected by OLS - make a compact sufficient model because the fuel gas consumption can be predicted with $R^2 = 0.92$. These most important variables are:

1. Main boiler temperature
2. Temperature of heating steam
3. Liquid level in the main boiler
4. Density of fuel gas

5. Total crude oil feed

This list of variables reflects the knowledge and expertise of the process engineers. However, it should be noted that statistical correlation does not necessarily result in informative features, we often neglected statistically significant variables from the final model based on the suggestions of the engineers. Therefore, the proposed tool should be considered as only a tool for decision support. A proper way to use OLS based feature selection is the following:

1. Let OLS select a large set of variables.
2. Among these potential inputs select a smaller set based on prior knowledge of the process.

### 3.2.5 RF based feature selection on electric power consumption

We used random forest feature selection to select a proper set of variables, which are relevant to the complete electric power consumption (KEI) of the AV2 unit. Based on prior knowledge of the process engineers we know that almost all the electric power is consumed by the main process pumps (total feed, inlet tower streams, cooling water and product streams). Based on this prior knowledge we expect that the feature selection algorithm should highlight the importance of flow rates and pressures.

For our calculations we used the MATLAB hosted R package implementation of the FORTRAN77 program created by Leo Breiman. The forest contained 500 regression trees. Each tree was grown using five randomly selected process variables from the original variable set. Figure 3.8 shows the normalised importance of each variable in alphabetical order (top), and ordered according to their importance level (bottom). As the results show, the relevance of variables is decreasing exponentially.

The total crude oil feed, the inlet pipe pressures and the flows of main process streams were proven to be the most important variables which ordering was also confirmed by the process engineers. We analysed the prediction performance of the random forest using validation samples. On the validation set, the model

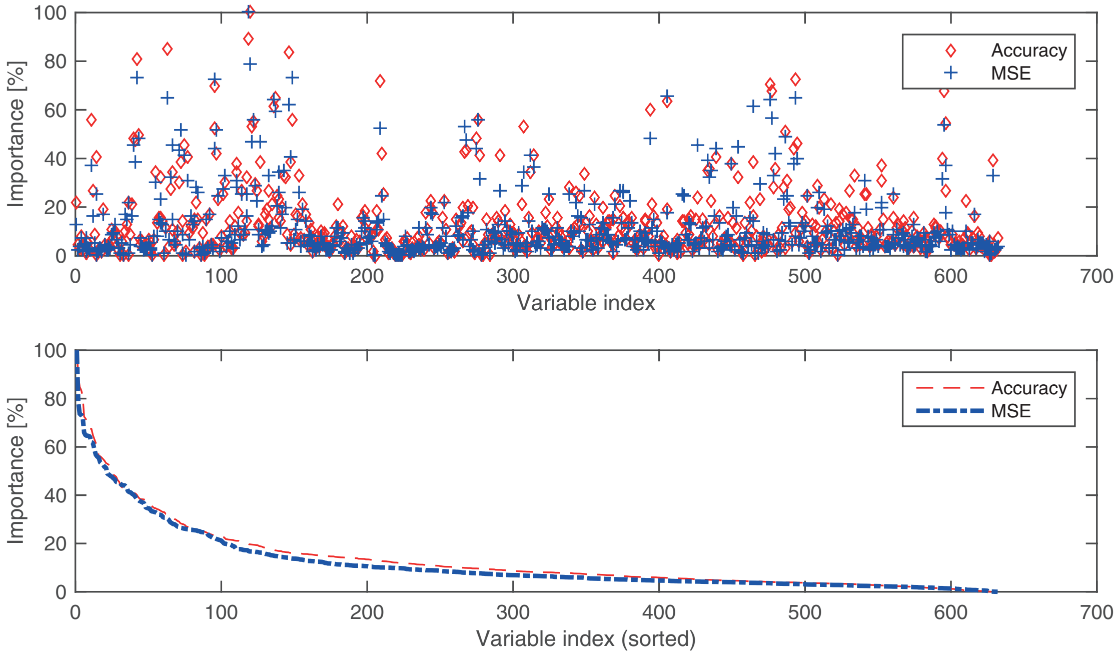

Figure 3.8. Relevance of process variables given by random forest regression

correlation was excellent, $R^2 = 0.97$. The selected variables were also used to formulate a linear model. The linear model with the ten most significant variables was also quite accurate, $R^2 = 0.92$. Based on the opinion of engineers and the patent related to feature selection for energy monitoring [104], the model having $R^2 = 0.92$ satisfies the minimum needs.

### 3.2.6 Conclusions

Energy Monitoring is based on monitoring the difference between targeted and measured energy consumption. In some cases, it is problematic to develop accurate and informative targeting models since it is difficult to determine, which process variables influence the KEIs. We developed an automated cause analysis (RCA) technique to find the most important driving factors of energy efficiency. The proposed concept is based on the application of feature selection algorithms. We examined two regression methods with feature selection capability for energy monitoring applications. We applied orthogonal least squares regression and random forest regression to predict key energy indicators and select the most important process variables which are relevant to the KEIs. The applicability of these methods was demonstrated on two KEI of AV2 plant in MOL Duna Refinery. Based on the results we can conclude that both methods can predict the KEI values and can select the most relevant process variables.

## 3.3 Parametric model based statistical process control

Following the aim of energy monitoring that is described in Section 1.3.2 We developed partial least squares regression based targeting models that predict the expected value of energy consumption and also visualise the operating regimes of the process. The development of PLS models could be problematic because a preliminary feature selection should be included in the development.

Since the complex set of process, variables determines Key Energy Indicators (KEIs) we applied Self-Organizing Map (SOM) models that support visualisation and feature selection of the process variables. Local linear target-models of different operating regions can be automatically determined based on the Voronoi diagram of the codebook of the SOM. We used Statistical Process Control (SPC) techniques to monitor the difference between the targeted and the measured energy consumption. We applied the concept of the resulted energy monitoring system at Heavy Naphtha Hydrotreater and CCR Reforming Units of MOL Hungarian Oil and Gas Company.

### 3.3.1 Basic concepts

Since many companies have built integrated databases to store historical process data from all plants, and in many cases no detailed knowledge is available about the process we should build data driven (black-box or *aposteriori*) models. The basics of energy monitoring are detailed in Section 1.3.2.

Self-Organizing Maps, which we use for modelling and data visualisation, performs a topology preserving mapping from high dimensional space onto a two-dimensional grid of neurones so that the relative distances between data points are preserved. As SOM provides a compact representation of the data distribution, it has been widely applied in analysis and visualisation of high-dimensional data [47]. It should be noted that since historical process data is extensively used our methodology can be considered as a mixture of precedent and activity-based targeting approaches.

The monitoring of the process is based on the difference between the targeted and the measured energy consumption (see Section 1.3.2). To provide a sophisticated analysis of this deviation we propose the application of statistical process control (SPC) techniques [105]. Control charts are industry-accepted methods to ascertain the in-statistical-control status of the process [52]. As we will show this technique - connected to the data-driven targeting models - is also suitable to provide informative feedback about the energy consumption.

### 3.3.2 SOM based models of energy monitoring

Data driven activity-based energy monitoring is based on the predicted value of energy consumption, $\hat{y}_k$. The structure of the model is given in Equation (3.3), where the calculated output $\hat{y}_k$ that represents an energy consumption or efficiency related variable is modelled by the linear combination of process variables (drivers), $\mathbf{x}_k = [x_{1,k}, \ldots, x_{n,k}]]$, where $k$ represents the $k$-th sampling time and $n$ stands for the number of process variables having significant effect to energy consumption. Based on a set of data $\mathbf{z}_k = [y_k, \mathbf{x}_k], k = 1, \ldots, N$ least squares method can be applied. In this case the application of operating regime based models could be beneficial:

$$\hat{y}_k \sum_{i=1}^{s} \omega_i\left(\mathbf{x}_k\right)\left(\mathbf{a}_i^T \mathbf{x}_k + b_i\right) \tag{3.12}$$

where $\omega_i\left(\mathbf{x}_k\right)$ describes the operating regime of the $i$-th local linear model defined by the parameter vector $\theta_i = \left[\mathbf{a}_i^T b_i\right]^T$. Piecewise linear models are special case of operating regime based models. If we denote the input space of the model by $T : z \in T \subset \mathbb{R}^n$, the piecewise linear model consists of a set of operating ranges $T_1, T_2, \ldots, T_s$ which satisfy $T_1 \cup T_2 \cup \cdots \cup T_s = T$ and $T_j \cap T_i = \emptyset$ when $i \neq j$. Hence, the model can be formulated as

$$\boldsymbol{If}\ \ \mathbf{x}_k \in T_i\ \ \boldsymbol{then}\ \ \hat{y}_k = [\mathbf{x}_k 1]\, \theta_i \tag{3.13}$$

where $\theta_s$ denotes the parameter estimate vector used in the $i$-th local model. SOM performs a topology preserving mapping from high dimensional space onto map units so that relative distances between data points are preserved. The map units (also referred as neurones or codebooks) usually form a two-dimensional regular lattice. Each neuron $i$ of the SOM is represented by an $l$-dimensional weight, or model $m_i = [m_{i,1}, \ldots, m_{i,l}]$. These weigh vectors of the SOM form a codebook. The partitioning is obtained by the Voronoi diagram of the codebook of the SOM. Our idea is to quantize the available input-output data to get a set of operating regimes and use the obtained regimes to identify parameters of local targeting models. SOM can be used to predict the output $\hat{y}_k$ of the process from the input vector $\mathbf{x}_k$. Regression is accomplished by searching for the Best Matching Unit(BMU) using the known vector components $\mathbf{x}_k$ (please remember SOM was trained based on $\vec{z}_k = [y_k \; \mathbf{x}_k]$. Since the output of the system is unknown, the BMU is determined as

$$i_0 = arg\ min\,||\mathbf{p}_i - \mathbf{x}_k|| \tag{3.14}$$

where $\mathbf{p}_i = [m_{i,2}, \ldots, m_{i,n+1}]$. The output of the model can be estimated by the local model of BMU, which could be piecewise constant model ($\hat{y}_k = d_i$) or piecewise linear regression model $\left(\hat{y}_k = \left[\mathbf{x}_k^T \; 1\right] \theta_i\right)$. (1) The piecewise constant output model results a $d_i$ constant value for each Voronoi cell; (2) The piecewise linear regression model estimates $y_k$ using the parameter estimate vector $\theta_i$, where $i$ is the index of BMU.

### 3.3.3 Results and discussion

The concept of historical data based energy monitoring system is demonstrated at Heavy Naphtha Hydrotreater and CCR Reforming Units of MOL Hungarian Oil and Gas Company. The plant's heating steam production is analysed as a demonstrating an example. The steam is produced in a furnace operated by fuel gas from the fuel gas network of the refinery. The energy content is calculated based on flow, density, heat capacity and temperature of the steam, so the unit of production is [GJ/h]. The targeting model is identified based on one-year historical data.

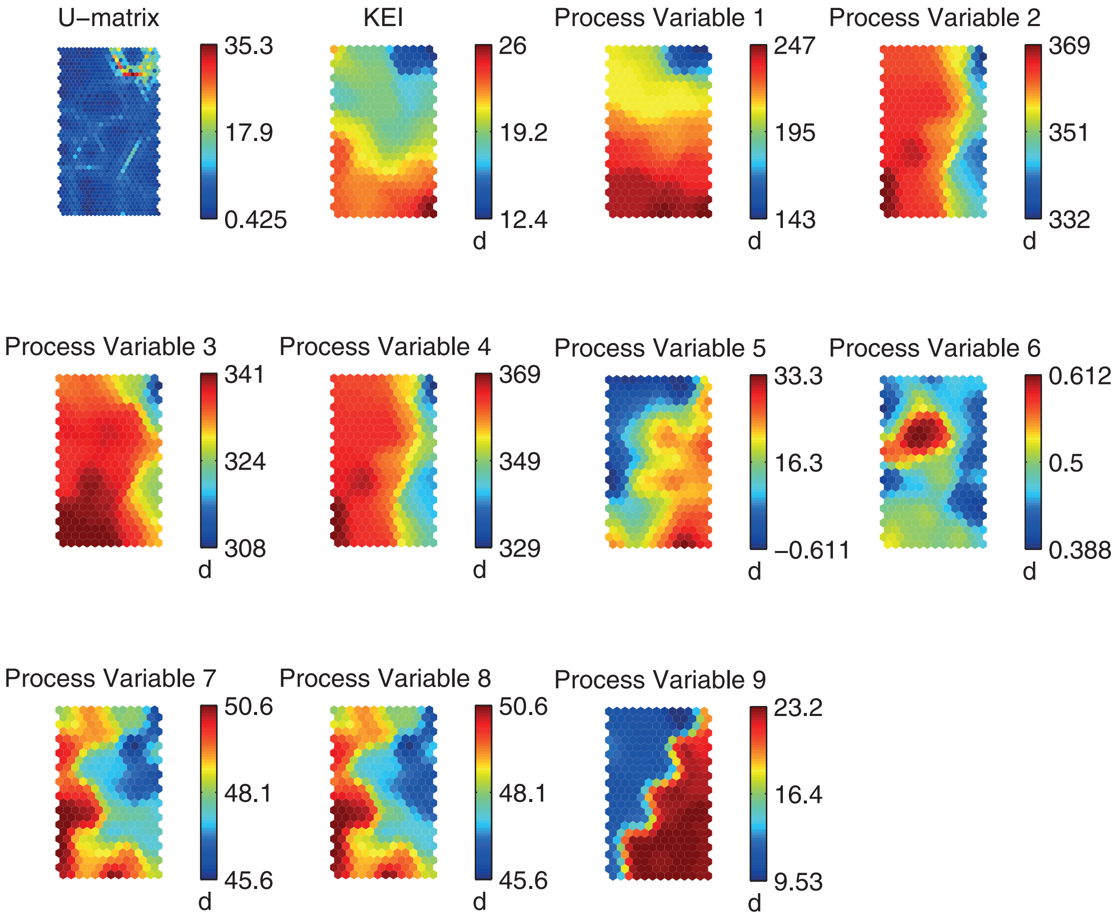

FIGURE 3.9. Self-Organizing Maps of process variables related to the Key Energy Indicator (KEI). The first Map shows the dissimilarity matrix (Euclidean distance) of Voronoi Cells. The 2nd Map show the Key Energy Indicator (KEI) that is the heating steam production.

We applied Self-Organizing Maps to identify the most relevant driving factors of heating steam production. These maps (matrixes) are useful to find correlated variables. Figure 3.9 compares process variables related to the Key Energy Indicator (KEI). We ranked the variables based on the similarity of maps measured by the absolute value of the 2D correlation coefficient ($C_2$) of two matrixes **A** and **B** given by Equation (3.15)

$$C_2 = \frac{\sum_i \sum_j \left(A_{i,j}-\bar{A}\right)\left(B_{i,j}-\bar{B}\right)}{\sqrt{\left(\sum_i \sum_j \left(A_{i,j}-\bar{A}\right)^2\right)\left(\sum_i \sum_j \left(B_{i,j}-\bar{B}\right)^2\right)}} \tag{3.15}$$
$$\bar{A} = \frac{\sum_i \sum_n A_{i,j}}{ij}$$
$$\bar{B} = \frac{\sum_i \sum_n B_{i,j}}{ij}$$

Using this method we can also find the variables with opposite behaviour to KEI, which is as well important as the variables with same behaviour (red -> blue is similar to blue -> red). As Figure 3.9 shows the Process Variable 1 is the most similar to the Key Energy Indicator, 2nd is Variable 7, the 3rd is Variable 8, etc Following the industrial practice, we applied PLS regression to obtain a targeting model. We identified a SOM model based on the same one-year historical data. This model estimates the steam production using the method described in the previous section.

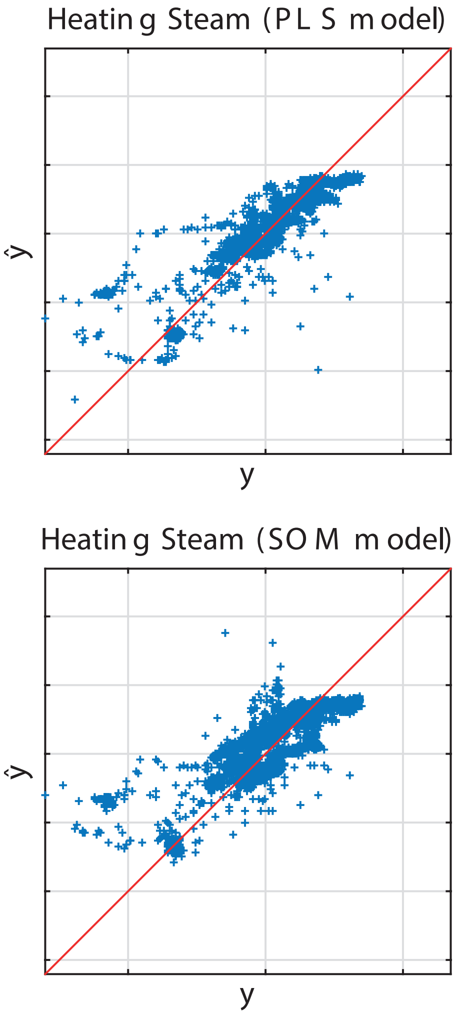

Figure 3.10. Time series of the latent variables and modelling error of PLS model.

We can identify the operating regimes of the technology and identify a local model for each Voronoi cell. Historical data related to the regimes are used to build local models. The prediction performance and the SPC charts based on this targeting model are shown in Figure 3.11. It should be noted that this nonlinear model gives almost the same prediction performance than the linear PLS model. Figure 3.10

shows the correlation diagrams of the examined models. SOM based model has almost the same result than PLS, but it can also show the operating regimes (See Figure 3.9).

The statistical process control practice provides useful tools [105] for monitoring like I-Charts and MR-Charts, which are well supported in the most monitoring systems. To examine the prediction error we used standard I-Charts with control limits. The limits were calculated using the six-sigma rule. With SPC we can detect outlier samples [105] and indicate that the technology could operate more efficiently.

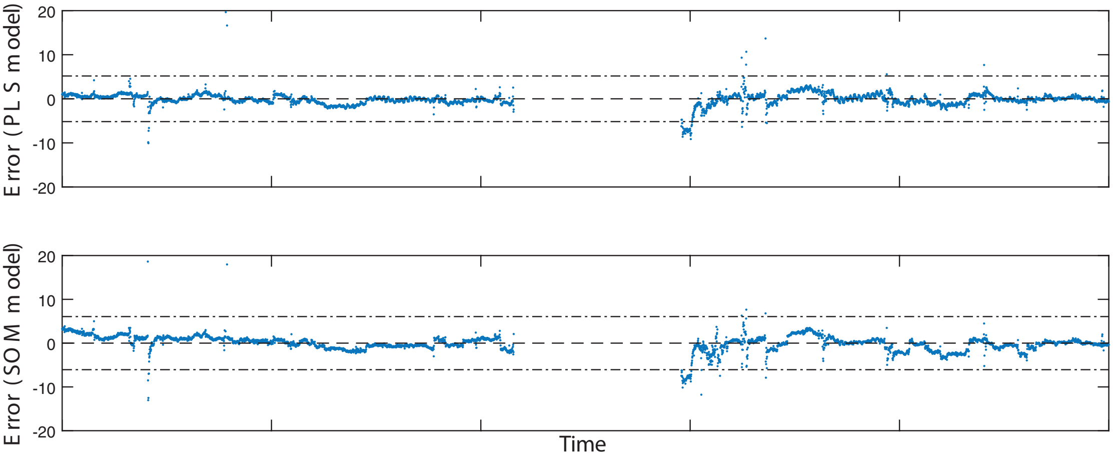

Figure 3.11. I-Chart of the modelling error for the two different models (PLS, SOM). The control limits are also shown determined using six-sigma rule. It should be noted that tracking only single value of energy consumption does not give detailed information about the operating situation.

### 3.3.4 Conclusion

Energy monitoring improves energy efficiency in process plants by helping plant operators, engineers and managers to track actual and target energy consumption. Energy monitoring is based on the comparison of Key Energy Indicators (KEIs) and their target values. These targets depend on operating regimes determined by a complex set of process variables. We developed advanced data-driven modelling techniques to support on-line targeting. We have utilized Self-Organizing Map and partial least squares to predict key energy indicators. We have shown that SOM can identify operating regimes and generate local models for different operating regimes. SOM is applicable also for feature selection based on the similarity measure between property maps. In addition we have used orthogonal least squares and random forest regression methods to select and rank input variables. Based on the selected subsets of energy driving factors we have identified compact and sufficient targeting models. Summarizing the results, we have combined regression and feature selection methods resulting a set of tools than can be effectively used together to support the systematic improvement of energy efficiency.

# Chapter 4

# Model validation and time-series segmentation

Process performance monitoring is based on models. In Chapter 2 and 3 we presented several techniques and methods to build non-linear models that include multiple operating modes and process transitions. Considering the process changes that are common in the chemical industry, we can improve the modelling performance if we build separate models for the different operating modes and regimes. It could be beneficial to use simplified linear models for various operation regimes instead of a complex non-linear one that covers all of those and use these local linear models for fault detection and analysis.

Monitoring plant performance at process transitions can help to reduce the off-specification production. Identification of critical process disturbances and the early warning of process malfunctions or plant faults can also reduce losses. Manual process supervision is largely based on visual monitoring of characteristic process trends. Although humans are excellent at visual detection of such patterns, it is a difficult problem for a control system software. This issue is particularly challenging when the process is complex, and large sets of multivariate signals should be monitored. The first step toward building an automated decision support system is the intelligent analysis of archive process data [106, 107].

We use time series segmentation for the identification of the operating modes. Time series segmentation is often used to extract internally homogeneous segments from a given time series to locate stable periods of time, to identify change points or to compress the original time series into a more compact representation [108].

Most time series segmentation algorithms manage only one time-variant variable and do not care on the relation between those [107]. An univariate time series can contain data in a time-ordered structure originated from a given sensor. Although accurate and frequent measurements are taken, it happens often that even the main changes of the system cannot be detected from the signal of a single sensor. This is because sometimes the changes in the correlation structure between the variables (sensor signals) are interesting since such fused information reflects the hidden change of the system. This chapter deals with the problem of multivariate time series segmentation and shows new algorithms that can manage time-varying multivariate data of sensors and analysers to detect changes in the state of the monitored processes.

## 4.1 Time series segmentation

A time series $T = \{\mathbf{x}_k = [x_{1,k}, x_{2,k}, \ldots, x_{n,k}]^T | 1 \leq k \leq N\}$ is a finite set of $N$ $n$-dimensional samples labelled by time points $t_1, \ldots, t_N$. A segment of $T$ is a set of consecutive time points $S(a,b) = \{a \leq k \leq b\}$, $\mathbf{x}_a, \mathbf{x}_{a+1}, \ldots, \mathbf{x}_b$. The $c$-segmentation of the time series $T$ is a partition of $T$ to $c$ non-overlapping segments $S_T^c = \{S_i(a_i, b_i) | 1 \leq i \leq c\}$, such that $a_1 = 1, b_c = N$, and $a_i = b_{i-1} + 1$. In other words, an $c$-segmentation splits $T$ to $c$ disjoint time intervals by segment boundaries $s_1 < s_2 < \ldots < s_c$, where $S_i(s_i, s_{i+1} - 1)$.

The goal of the segmentation procedure is to find internally homogeneous segments from a given time series. To formalize this goal a cost function $cost(S(a,b))$ - describing the internal homogeneity of individual segments - should be defined. Usually this cost function $cost(S(a,b))$ is defined based on the distances between the actual values of the time series and the values given by a simple function (constant or linear function, or a polynomial of a higher but limited degree) fitted to the data of each segment. For example in [32, 123] the sum of variances of variables in the segment was defined as $cost(S(a,b))$:

$$\begin{aligned} cost(S_i(a_i, b_i)) &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \parallel \mathbf{x}_k - \mathbf{v}_i \parallel^2, \qquad (4.1) \\ \mathbf{v}_i &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \mathbf{x}_k, \end{aligned}$$

where $\mathbf{v}_i$ is the mean of the segment.

Cost functions of several segmentation algorithms can be minimized by dynamic programming (e.g. [32]), which is unfortunately computationally intractable for many real data sets. Hence, usually one of the the following heuristic approaches are followed:

- **Search for inflection points:**
  Searching for primitive episodes located between two inflection points [106].
- **Sliding window:**
  A segment is grown until it exceeds some error bound. The process repeats with the next data point not included in the newly approximated segment. For example a linear model is fitted on the observed period and the modelling error is analysed [46].
- **Top-down method:**
  The time series is recursively partitioned until some stopping criteria is met [46].
- **Bottom-up method:**
  Starting from the finest possible approximation, segments are merged until some stopping criteria is met [46].
- **Clustering based method:**
  Time series segmentation may be viewed as clustering, but with a time-ordered structure. In ref. [124] a new fuzzy clustering algorithm has been proposed, which can be effectively used to segment large, multivariate time series.

In data mining, the bottom-up algorithm has been used extensively to support a variety of time series data mining tasks [46], hence in this thesis this approach will be followed. The algorithm begins with creating a fine approximation of the time series and iteratively merge the pair of segments having the lowest merge cost until a stopping criteria is met. When adjacent segments $S_i$ and $S_{i+1}$ are merged, the costs of the new segment's merging with it's left $S_{i-1}$ and right ($S'_{i+1}$ neighbours must be calculated. The $S'_{i+1}$ in $k+1$-th is equivalent to $S_{i+2}$ in $k$-th iteration.

## 4.2 Regression based time series segmentation

The first method we developed is a regression model based times series segmentation algorithm. The method follows the strategy of a bottom-up scheme to detect changes in multivariate time-series of model inputs (drivers) and outputs (energy consumption or efficiency measure). The algorithm works offline and requires historical process data for a period when the operating regimes and modes occurred. As we mentioned in Section 4.1, time series segmentation algorithms need cost functions or goals to be reached during the segmentation. The proposed segmentation algorithm is based on simple least squares regression since we used this cost function to evaluate the quality of our models. In the following sections, we describe the cost function and the algorithm in details.

### 4.2.1 Cost function formulation

Since segments are defined to represent homogeneous periods of operation in which a local linear model can efficiently describe the functional relationships among process variables, we defined the cost function based on the mean square error of the local models:

$$\begin{aligned} cost_i\left(a_i, b_i\right) &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \left(y_k - \hat{y}_k\right)^2 \\ &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \left(y_k - \mathbf{x}_k^T \theta_i\right)^2 \end{aligned} \tag{4.2}$$

where $\hat{y}_k$ is given by the Equation (3.3) and the $\theta_i$ parameters are determined based on a dataset of the segment,

$$\theta_i = \left(\mathbf{X}_i^T \mathbf{X}_i\right)^{-1} \mathbf{X}_i^T \mathbf{y}_i \tag{4.3}$$

where $\mathbf{X}_i = \left[\mathbf{x}_{a_i}, \ldots, \mathbf{x}_{b_i}\right]^T$ and $\mathbf{y}_i = \left[y_{a_i}, \ldots, y_{b_i}\right]^T$.

The algorithm automatically finds the segment borders and the related model parameters. The user has to evaluate only the result of segmentations and analyse the differences among the model parameters. Since the method uses linear regression model it simultaneously determines the $\theta_i$ parameters of the models used to approximate the behaviour of the system in the segments and the $a_i, b_i$ borders of the segments by minimising the sum of the costs of the individual segments:

$$cost_T^c = \sum_{i=1}^{c} cost_i \tag{4.4}$$

To reduce the computational cost of a segmentation we use the Bottom-Up heuristic approach. The pseudo code of Bottom-Up strategy is shown in Algorithm 3.

**Algorithm 3** Bottom-Up segmentation algorithm

```
procedure BU–SEGMENTATION
    for each segment pair $i \in [1,\ N-1]$ do
        #Find the cost of merging for each pair of segments
        $mergecost(i) = cost(S(a_i, b_{i+1}))$
    end for
    while $min(mergecost) < maxerror$ do
        #Find the cheapest pair to merge:
        $i = argmin_i(mergecost(i))$

        #Merge the two segments, update the $a_i, b_i$ boundary indices, and recalculate the merge costs:
        $mergecost(i) = cost(S(a_i, b_{i+1}))$
        $mergecost(i-1) = cost(S(a_{i-1}, b_i))$
    end while
end procedure
```

This algorithm is quite powerful since the merging cost evaluations requires simple identification of Linear Regression models, which is easy to implement and computationally cheap. Because of this simplicities the proposed approach can be considered as a multivariate extension of the piecewise linear approximation (PLA) based time series segmentation and analysis tools developed by Keogh [46, 118].

### 4.2.2 Application in energy monitoring

The concept of the resulted data based energy monitoring system is demonstrated at Heavy Naphtha Hydrotreater and CCR Reforming Units of MOL Hungarian Oil and Gas Company. The same dataset was evaluated as was in the Chapter 3.

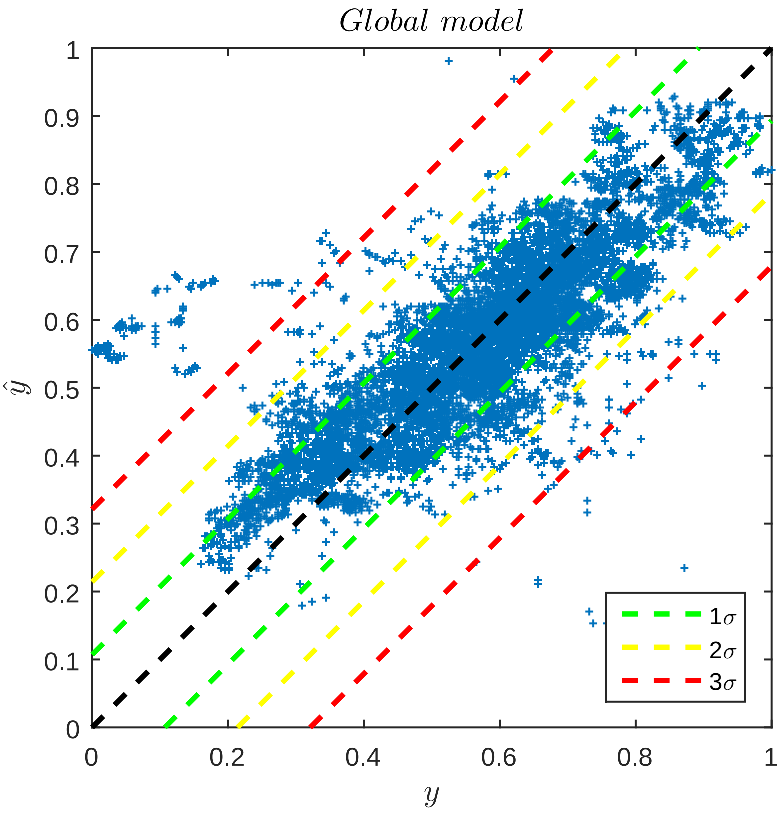

Figure 4.1. Correlation diagram of measured and predicted energy consumption. The black line shows the ideal prediction; the dashed lines show $Q$ levels based on standard deviation ($\sigma$) of prediction error ($y - \hat{y}$). Colored lines belong to $1\sigma, 2\sigma, 3\sigma$ $Q$ levels.

The fuel gas consumption target was calculated based on one-year historical data. We scaled the data into [0 1] interval since the nominal values of process variables are confidential. The following drivers of the fuel gas consumption were identified based on the analysis of the technology and data: total feed, inlet temperature, the density of fuel gas and ambient temperature. These drivers were selected by feature selection algorithms that were presented in Chapter 3. The result of the data the mining procedure was validated by process experts.

We applied linear least squares regression to obtain the parameters of a model that covers the whole operation period. This model is referred as a global model since it is based on the entire available dataset, segmentation was not applied for the selection of relevant operating regimes. Figure 4.1 compares the targeted (predicted) energy flows and actual data of a model which has the parameters that are given in Table 4.2.

The correlation diagram helps to qualify the energy efficiency of the technology.

When all the data is taken into account, the target model estimates the average energy consumption. This means when the estimation of the target model is higher than the measured value the process operates well. Otherwise, it should be checked what is the reason of a higher consumption than the expected with the current operating parameters. For a sophisticated decision support, the confidence of the model should be taken into account. In this case for an estimated target value, $\hat{y}$ , $[\hat{y} - \delta(y) \ldots \hat{y} + \delta(y)]$ bounds (related to a given confidence level $\alpha$) can be calculated.

The confidence interval calculation is based on Student distribution with $n - 2$ degree of freedom. Where $1 - \alpha$ is the confidence level (probability).

$$\begin{gathered}\hat{y}^* = \hat{y} \pm t_{1-\frac{\alpha}{2}} * s'_{\hat{y}_e} = \hat{y} \pm \delta(\hat{y}_e) \\ P\left(y + \Delta y_e - \delta(y) < \hat{y} < y + \Delta y_e + \delta(y)\right) = 1 - \alpha \end{gathered} \tag{4.5}$$

$$s'_{\hat{y}_e} = \sigma_e^2 \sqrt{\mathbf{x}_k^T \left(\mathbf{X}^T\mathbf{X}\right)^{-1} \mathbf{x}_k}, s_e = \sqrt{\frac{\sum (y - \hat{y})^2}{N - 2}} \tag{4.6}$$

As this consideration shows, the tuning (how aggressive or conservative the target model will be) is realised by the shifting of the predicted output based on the variance of the modelling error. The confidence bound is also based on this measure. To reduce this variance and increase the model accuracy the abnormal consumption patterns must be filtered out from the pool of data related to different operating regimes.

Before the application of the proposed segmentation algorithm, we also applied PLS model to visualise the operating regimes. PLS is a method for constructing predictive models from a large number of correlated input variables and 2D visualisation of high dimensional process data as we described in Chapter 3. It should be noted that the performance of a PLS model is slightly worse than a linear regression model (see Table 4.1) due to the regularisation effect of the latent variables. The latent variables of the PLS model can be used to visualise the operating regimes in two dimensions. As Figure 4.2 shows, several operating regimes can be distinguished. Operating regimes requiring more energy than the average value are well separated from operating regimes requiring energy less than the average. We can assume that the process was operated under significantly different conditions. As the following analysis will show, a single linear targeting model is sufficient to cover all these operating modes.

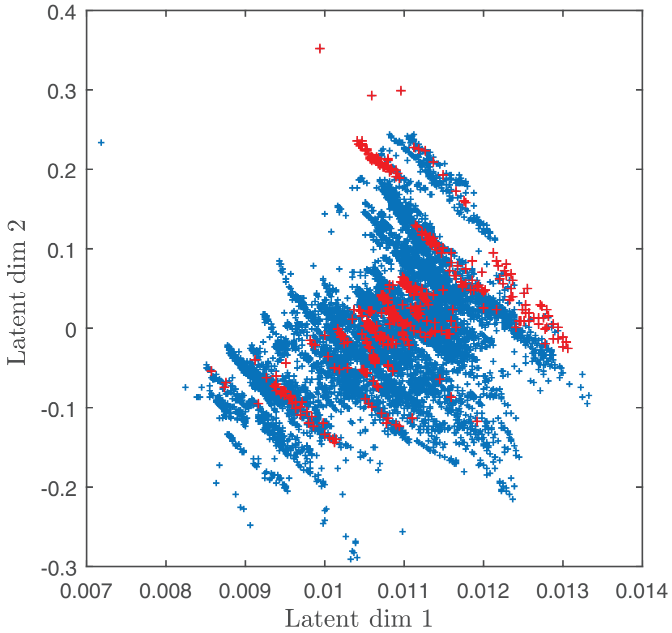

Figure 4.2. 2D PLS mapping of the operating regimes. Red '+' represents operating points requiring more energy than average.

The segmentation algorithm was implemented in MATLAB. Our implementation is compatible with version 7. Fig. 4.3 shows the result of the segmentation of the one-year historical process data. The boundaries of the segments are highlighted with vertical lines. Based on the analysis of the results the most important operating modes of the process can be automatically determined. Four significant segments were detected. Shorter segments represent major changes in the process. These operating periods are related to grade transitions or process malfunctions. The parameters of the models related to the first two longer segments are given in Table 4.2. The performances of these local targeting models are shown in Fig. 4.4.

The following figures indicate that the identified local linear models have much better prediction performance (and smaller variance) on the related operating regions (also see Table 4.1). The practical benefit of identifying of these modes is that more accurate models can give more relevant information that can support the energy monitoring system.

It should be noted that detailed analysis of the parameters of the local models can support the improvements of the targeting model. In our case, it is interesting to

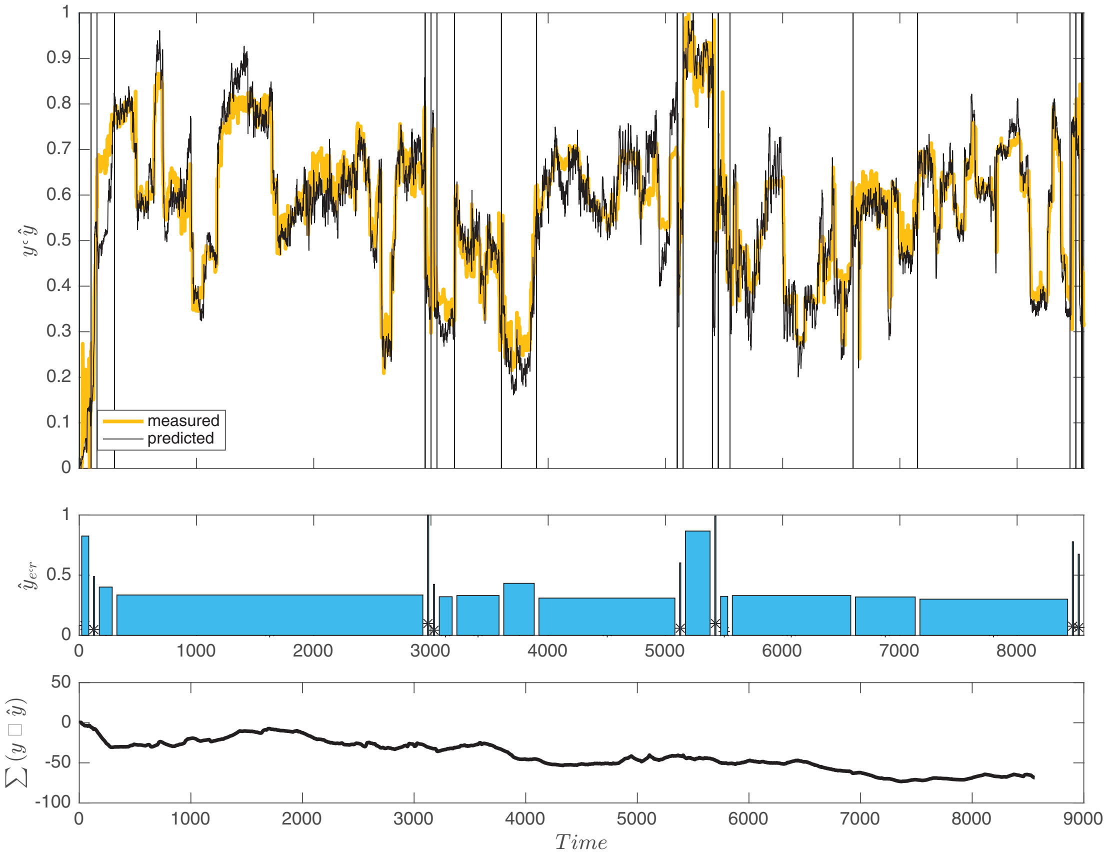

FIGURE 4.3. Results of the segmentation . The boundaries of the segments are shown by vertical lines. Estimated and measured energy consumption values are shown. The bottom figure shows the cost of the segments that are the average mean square errors of the models.

TABLE 4.1. Performance of the models (std deviation of the modelling error)

| Model | $\sigma$ |
|---|---|
| PLS | 0.0912 |
| Linear | 0.0893 |
| Segment 4 | 0.0663 |
| Segment 17 | 0.0648 |

see that the largest variation in the parameters of the local models is related to the inlet and ambient temperatures. This variation can be easily explained by the differences among operating strategies of the winter and summer periods.

Based on the proposed concept, the accuracy of the targeting models can be significantly improved. For the evaluation of this development, the variance of the targeting model's prediction can be used. This means that the proposed technique can improve the efficiency of bandwidth analysis, as better targeting models can provide better estimates of the practical minimum energy (PME) requirements.

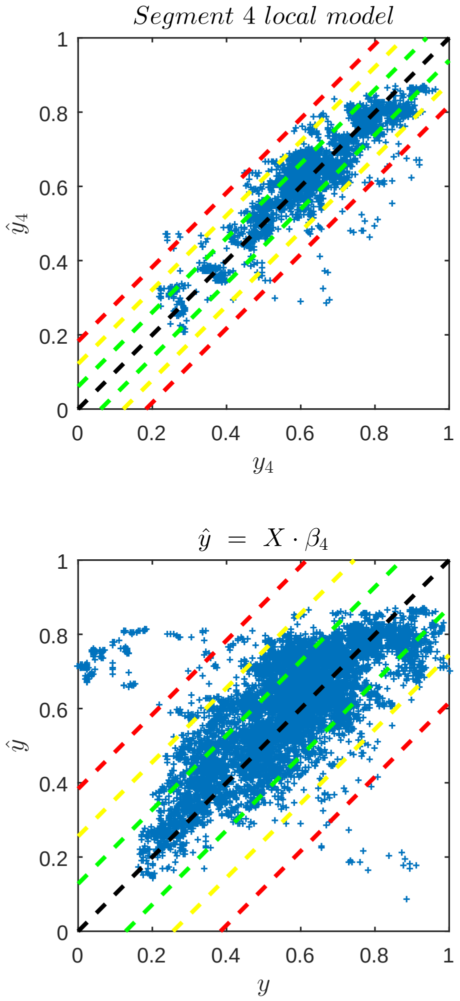

FIGURE 4.4. Comparison of the local target models in their segments. As can be seen the local models have much smaller variance in their segments, so they can provide more accurate information for energy monitoring.

TABLE 4.2. Parameter values in global linear model and in the best two segments.

| $\hat{y}$ = | Total feed×$\theta_1$ | Inlet temp. ×$\theta_2$ | Density of fuel gas ×$\theta_3$ | Ambient temp. ×$\theta_4$ | $\theta_5$ (bias) |
|---|---|---|---|---|---|
| Global | 0.63 | -0.79 | -0.32 | -0.05 | 0.99 |
| Segment 4 | 0.57 | 0.02 | -0.3 | -0.17 | 0.44 |
| Segment 17 | 0.63 | -0.47 | -0.37 | 0.02 | 0.75 |

## 4.3 Correlation based time series segmentation

In the previous section, we described a regression based times segmentation algorithm. This method works well if a prediction model could be applied so there are process variables that are measured and also could be predicted. In exceptional cases when regression based segmentation could not be implemented because the predicted variable is not measured so we cannot calculate the prediction error or there is no predictable variable at all, we have to use such algorithm that does not use prediction or estimation. We developed a second approach for multivariate time series segmentation to eliminate this drawback of regression-based technique. The designed segmentation algorithm can detect changes in correlation structure among several measures.

Our method follows the basic concept of Xuan's and Murphy's observation likelihood models [126] but the implementation is simplified and uses a goal oriented similarity function instead of feature models.

We demonstrate the capability of this technique in a near infrared spectrum series that is collected in diesel blending unit in Duna Refinery of MOL Ltd. between Mid September and of December in 2013. This time series contains about 15.000 spectra recorded between $4000 - 4076$ $[cm^{-1}]$ wave numbers in every 5 minutes.

### 4.3.1 Dissimilarity measure formulation

Similarly to regression based segmentation our correlation based method also requires a goal-oriented cost function, but it is better to name it distance or dissimilarity function. Since segments are defined to represent homogeneous periods of operation in which the internal correlation can describe the statistical relationship between process variables. We define the dissimilarity of segments based on the dissimilarity correlation coefficient matrices. We have compared the correlation coefficient matrixes using the Frobenius distance.

Let $\mathbf{X}_k$ is the time series of variables in the $k$-th time segment. The $\mathbf{R}_k$ is the matrix of correlation coefficients of variables in the $k$-th time segment. The calculation of $\mathbf{R}_k$ for continues variables is detailed in Equation 4.7.

$$\mathbf{R}_k(i,j) = \frac{C_k(i,j)}{\sqrt{C_k(i,i)\,C_k(j,j)}} \tag{4.7}$$

where $C_k(i,j)$ is the covariance. $C_k(i,i)$ and $C_k(j,j)$ are standard deviations of $i$-th and $j$-th variable in $k$-th segment.

The $d_k$ distance of $k$-th segment to it's right neighbour $k+1$ is calculated as Frobenius distance of correlation matrices.

$$d_k = \sqrt{trace\left((\mathbf{R}_{k+1} - \mathbf{R}_k)(\mathbf{R}_{k+1} - \mathbf{R}_k)^T\right)} \tag{4.8}$$

The algorithm is quite powerful since merging cost evaluation requires simple calculation of Frobenius norms which is easy to implement.

### 4.3.2 Application in spectral time series segmentation

The applicability of the algorithm is demonstrated at Diesel Blending Unit of MOL Hungarian Oil and Gas Company. In this application, we have identified changes in production based on the recorded spectrum. We do not use any other measurements or estimates. Our dataset contained about 15000 recorded spectra from time range of mid-September to end of October in 2013. Each spectrum contains 195 absorbance values. We consider the spectral series as a multivariate time series of 195 variables.

On the Fig. 4.5 we show the implied spectra series on a 3D surface. As the graph shows, there are only small changes in the series so the application of basic algorithms like change detection in the average or variance analysis is problematic.

The developed algorithm uses correlation coefficient matrices. To show the correlation structure we use colour map like Fig. 4.6. On the map we can see the inner correlation of the time series of absorbance values. On $x$ and $y$ axis are the wave numbers and the colour shows the value of the correlation coefficient.

On Fig. 4.6 we show the correlation structure which, is calculated on the whole time series. Our algorithm finds segments in time when the correlation structure changes.

Following the bottom up strategy, we make an initial resolution. We partition the series into 200 small segments. The algorithm merges the neighbouring segments having the lowest dissimilarity value that is calculated by the equation 4.8. The merging will be continued until some stopping criterion is met. In this experiment, the stopping criterion is the number of the final segments. Using 20 segments as stopping criteria the algorithm provide the segmentation that can be seen on Figure 4.7.

The Fig. 4.7 shows the correlation coefficient matrix $R$ of the identified segments. As the maps show, the algorithm can detect changes in the correlation structure of the absorbance values at different wavelengths. For reference, we compared the found segment borders to the dates of product changes.

The Fig. 4.8 shows the timestamps of product changes (black) and also the determined segment borders (red). As we can see most of found segment boarders are close to the product change timestamps. When we compare the Figure 4.7 and Figure 4.8 we can see that most of the short segments (e.g. 2, 16, 18) contain a product change. In these segments the correlation matrix has a higher average and

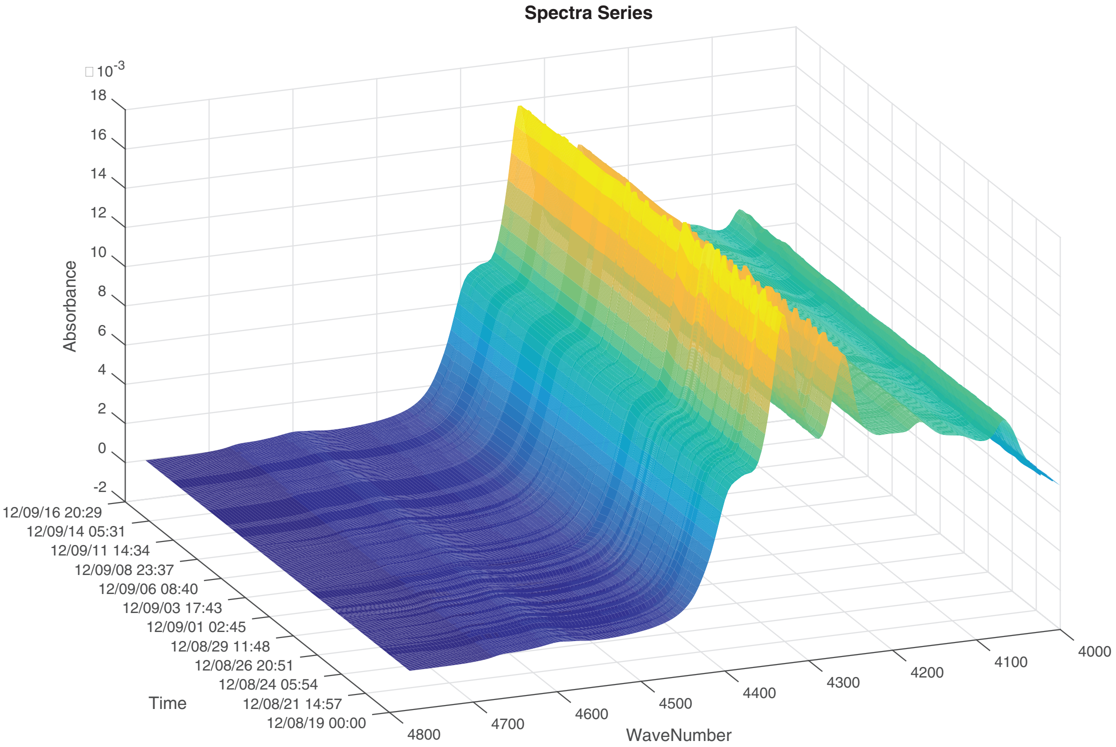

Figure 4.5. Infra red spectrum time series. First dimension($x$-axis) is time when the sample was taken, second dimension ($y$-axis) is wave number, third dimension ($z$-axis) is absorbance value.

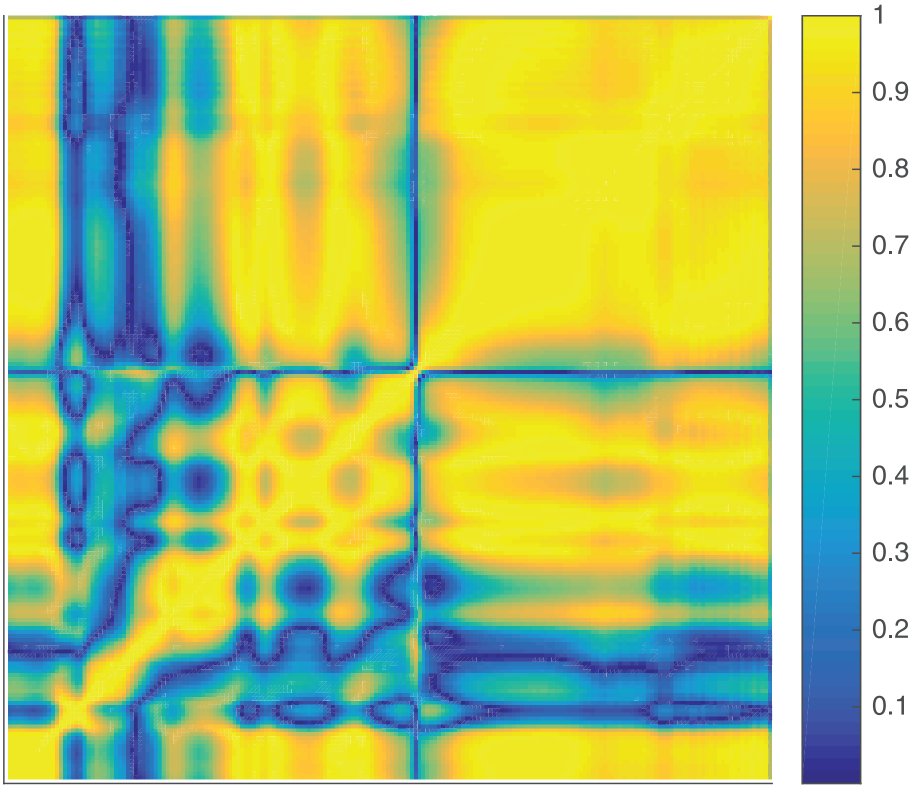

Figure 4.6. Colour map of correlation coefficient matrix of whole time series.

the standard deviation of the spectral series is higher with one order of magnitude. It means that the segmentation algorithm finds the ranges of process transitions.

## 4.4 PCA based time series segmentation

The last time-series segmentation algorithm we use is based on Principal Component Analysis based multivariate time series segmentation. PCA based segmentation is highly related our correlation based algorithm described in the previous section and we can consider the PCA based method as an extension of it. PCA based segmentation was developed by Dobos and Abonyi and detailed in ref. [127]. This approach uses their results and algorithm with some modification and customization to fit spectral time series.

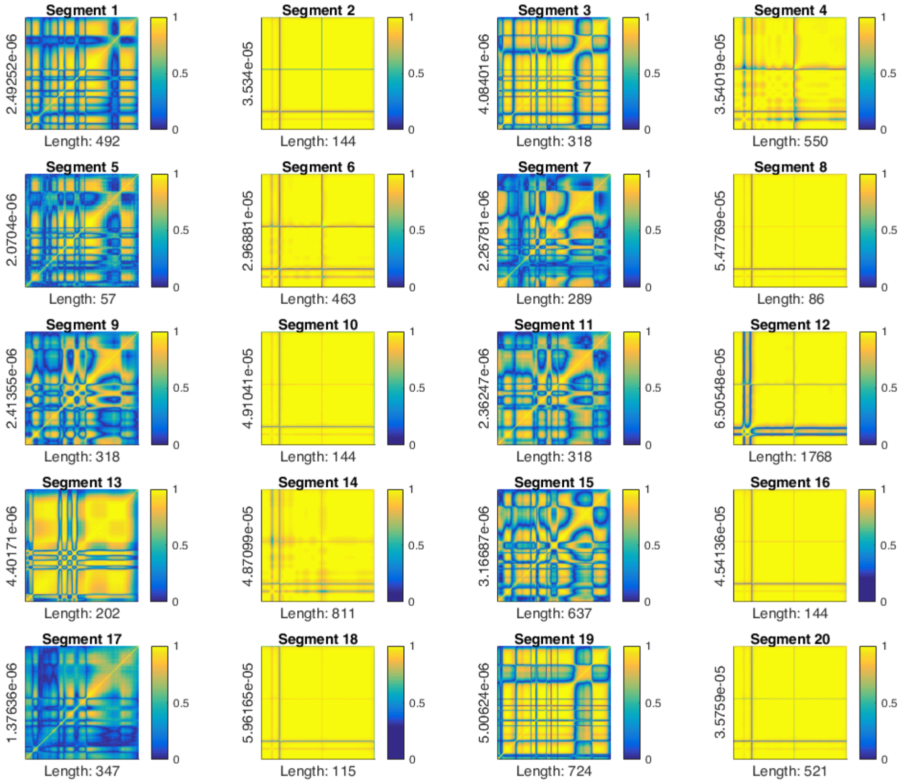

FIGURE 4.7. Colour map of correlation coefficient matrix of first 20 identified segments. The colour denotes correlation value, the number on $y$ axis is the average of standard deviation of spectrum

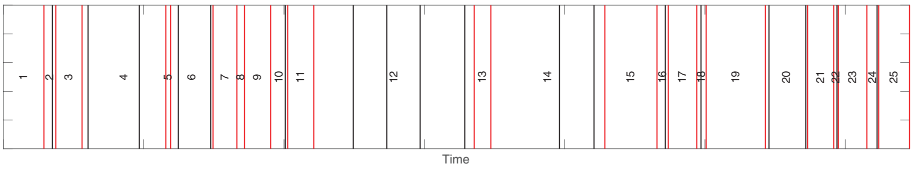

FIGURE 4.8. Timestamps of product changes and found segment borders. Black lines are the product change dates, the red lines are the found segment boarders

### 4.4.1 Cost function formulation using principal components

The cost function of the segmentation is based on the Principal Component Analysis of the $\mathbf{F}_i$ covariance matrices of the segments:

$$\mathbf{F}_i = \frac{1}{b_i - a_i} \sum_{k=a_i}^{b_i} (\mathbf{x}_k - \mathbf{v}_i)(\mathbf{x}_k - \mathbf{v}_i)^T . \tag{4.9}$$

PCA is based on the decomposition of the $\mathbf{F}_i$ covariance matrix $\mathbf{F}_i = \mathbf{U}_i \Lambda_i \mathbf{U}_i^T$ into a $\Lambda_i$ matrix which includes the eigenvalues of $\mathbf{F}_i$ in it's diagonal in decreasing order, and into a $\mathbf{U}_i$ matrix which includes the eigenvectors corresponding to the eigenvalues in it's columns. With the use of the first few ($p < n$) nonzero eigenvalues and the corresponding eigenvectors, the PCA model projects the correlated high-dimensional data onto a hyperplane, which is useful for the visualization of multivariate data:

$$\mathbf{y}_{i,k} = \Lambda_{i,p}^{-\frac{1}{2}} \mathbf{U}_{i,p}^T \mathbf{x}_k \tag{4.10}$$

When the PCA model has adequate number of dimensions, the distance of the data from the $p$-dimensional hyperplane of the PCA model is resulted by measurement failures, disturbances and negligible information. Hence, it is useful to analyse the reconstruction error of the projection:

$$Q_{i,k} = (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T (\mathbf{x}_k - \hat{\mathbf{x}}_k) = \mathbf{x}_k^T (\mathbf{I} - \mathbf{U}_{i,p} \mathbf{\Lambda}_{i,p} \mathbf{U}_{i,p}^T) \mathbf{x}_k. \tag{4.11}$$

The analysis of the distribution of the projected data is also informative. The Hotelling $T^2$ measure is often used to calculate the distance of the mapped data from the center of the linear subspace

$$T_{i,k}^2 = \mathbf{y}_{i,k}^T \mathbf{y}_{i,k}. \tag{4.12}$$

Fig. 4.9 illustrates these measures in case of two variables and one principal component.

These $T^2$ and $Q$ measures are often used for the monitoring of multivariate systems and for the exploration of the errors and the causes of the errors.

The main idea of this section is to use these measures as the measure of the homogeneity of the segments:

$$\begin{aligned} cost_{T^2}(S_i(a_i, b_i)) &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} T_{i,k}^2 \\ cost_Q(S_i(a_i, b_i)) &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} Q_{i,k} \end{aligned} \tag{4.13}$$

### 4.4.2 Application in process monitoring using spectra series

The proposed tool has been applied to the same time series of spectra as in the case of correlation based method. As Fig. 4.10 shows the proposed tools can

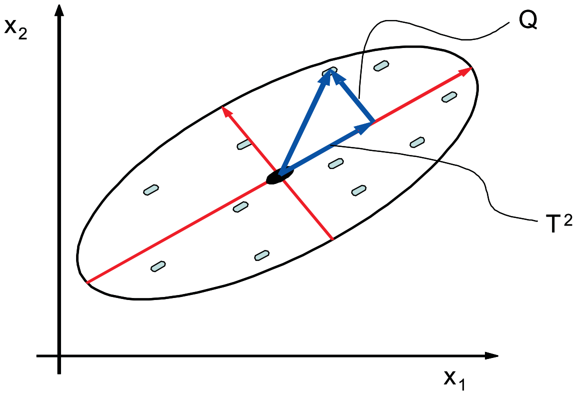

FIGURE 4.9. Distance measures based on the PCA model.

detect significant changes in the process. It is interesting to see that some of these changes related to product changes (denoted by vertical lines).

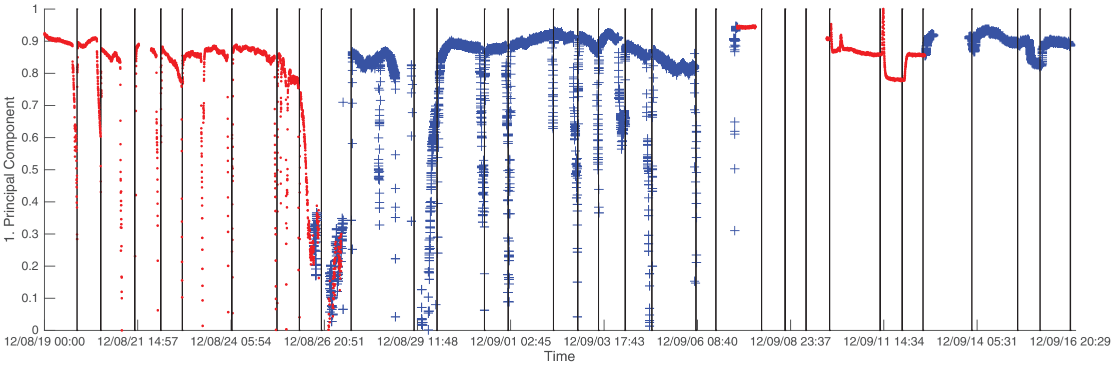

FIGURE 4.10. Segmentation results of the diesel blending process. The red and blue ranges after each other show the found segments in time series.

The tool has been applied to on-line monitoring. The results here are even more convincing; we were able to detect malfunctions of the spectrometer and also changes in the parameters of the operations without knowing any background information. Details of the PCA models and the projected latent variables are given in Fig. 4.11. Although the bottom-up strategy cannot find the segment reliably because the time series contained samples from the ranges when the spectrometer was not operating correctly, PCA is a quite powerful tool as the figure shows. Segments can be identified easily also by operators because principal components have significant deviations when changes occur.

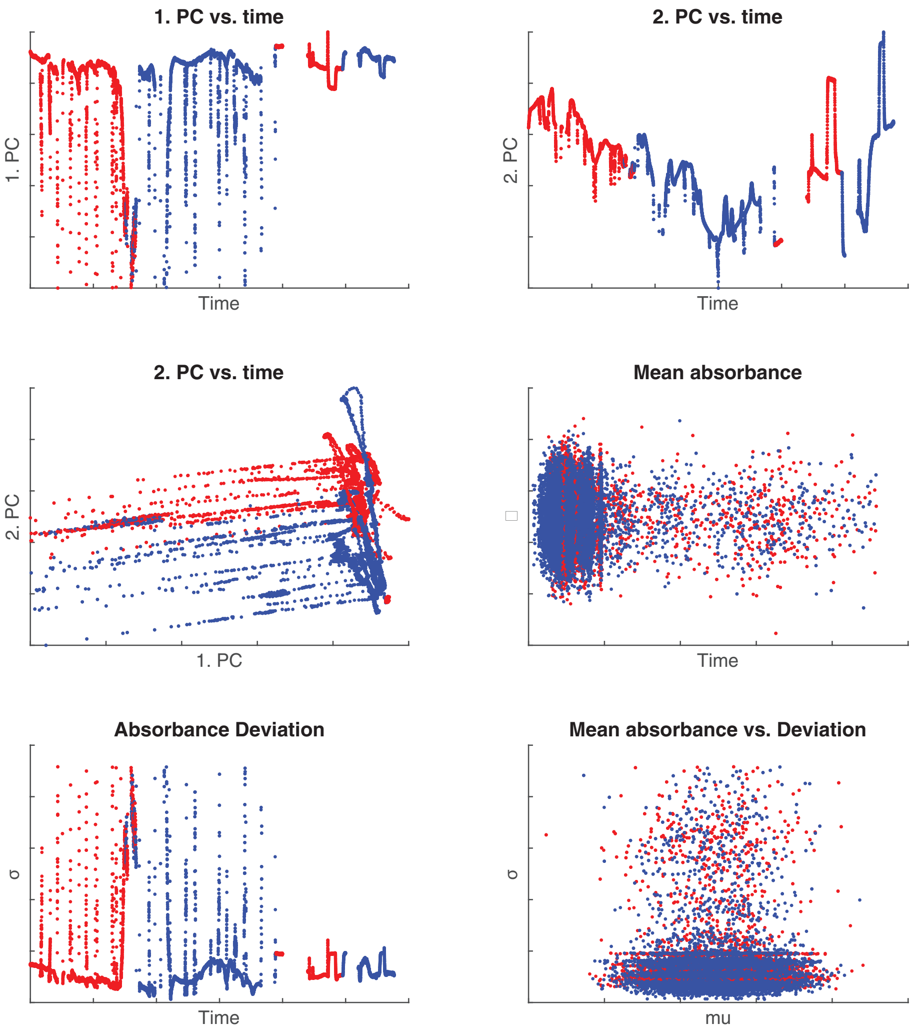

Figure 4.11. Details of the latent PCA variables and the results of the segmentation.

## 4.5 Conclusions

Data based process monitoring requires well-built mathematical models of the technology. Usually, these empirical models are identified based on historical data but the recorded data series could be affected by measurement errors or can cover time ranges, which are not relevant for the current modelling task. For example, if we want to include only a particular operating mode with a specific model we care on historical data, when the technology was in that special mode. It is also plausible that we want to exclude data from a downtime or a maintenance window. Until now the selection of proper training data is performed manually based on a heuristic and subjective evaluation of the operation of the process. This practice is not efficient and very time-consuming. In this chapter, we presented three different goal-oriented time-series segmentation techniques to automate this procedure.

Regression-based time-series segmentation is a powerful tool for those cases when linear prediction models cannot be applied e.g. targeting model based energy monitoring. The real benefit of this method is: during the segmentation the algorithm builds predictor models also that can be used in the energy monitoring solution as starting point of creation of final target models. With the proposed tool target-models for different operating regions can be automatically determined. The presented case study shows the applicability of the proposed methodology since we were able to build a set of accurate models and identify a set of operating regimes showing different impacts of the drivers of energy consumption and efficiency. Once the proposed scheme has been set up, building and analysis of targeting models is routine operation and should be neither time-consuming nor complex procedure. Further analysis should focus on the detailed comparison of the operating regimes and models. For such cases when prediction models could not be built or applied, we developed two statistics based segmentation algorithms, and we have proven the concept on one of the most difficult to manage data type which is spectral time series. Our correlation bases segmentation could be applied when we want to detect structural changes in technology like switching product or operation mode. It is useful because using the developed algorithm we can select those time ranges when the operation was stable and can focus on process transitions also that is complicated to handle and can cause losses. With proper segmentation, we can detect the stabilised operation or we can build special models for transition monitoring. Using these models the transition time could be shortened, and losses could be reduced. Principal Component Analysis is a sophisticated tool and not just for segmentation but detection of any changes in technology. We applied PCA to segment online spectral time series, but we proved that PCA can detect the changes immediately so can also be used for fault detection. We can conclude that the developed segmentation algorithms can help to select proper historical data for the building of nonlinear process models like we presented in Chapter 2 and Chapter 3, and can also be applied independently for monitoring of changes and fault detection.

All the programs used to generate the results in this thesis can be downloaded from
`https://github.com/kulcsartibor/phd-thesis-programs` and
`http://www.abonyilab.com/`

# Chapter 5

# Summary and Theses

Nowadays industry and in particular process industry more and more relies on new information theory and artificial intelligence related solutions that can help to improve the technology and reduce the cost of instrumentation, automation and maintenance. Software sensors are capable of extending or even replace the classical instrumentation of technology. We developed techniques and tools to support the identification and maintenance of data-driven models of soft sensors used for product quality and energy usage estimation.

The dissertation describes three different aspects of soft-sensor development. The first two chapters are focusing on parametric and non-parametric modelling, while the third chapter deals with the selection and preprocessing of the data.

Regarding non-parametric models, I proposed a genetic programming based algorithm to generate dimension reduction mappings. I showed the applicability of these mappings in spectroscopic modelling and in solving the classical Wine classification benchmark problem. Finally, I presented tools to for the selection of the input variables of these models and data segments. I demonstrated the capability of the methods in spectroscopic modelling and energy monitoring of chemical processes.

## 5.1 Experimental tools and technologies

We developed the dimension reduction mapping algorithms to extract useful information from the data are originated from the Diesel Blending Unit and the product development laboratory of MOL Duna Refinery. The datasets contained spectra recorded by ABB and Bruker spectrometers. For the processing of the spectra, we used the software of ABB and Bruker. We used MATLAB to implement our algorithms. For the development of parametric modelling related algorithms, we extracted the times series of process variables from the OSIsoft PI central data collection system of MOL Duna Refinery.

## 5.2 Theses

1. **I have developed a genetic programming based solution to visualise high dimensional datasets. The method can explore the operating regimes of online NIR analysers and can support the identification of classifier models.**
   (Related publications: [137, 142, 144, 145, 148, 150, 151])

   (a) I developed a multi-chromosome representation based genetic algorithm to find explicit multi-dimensional projections of high-dimensional input spaces into lower dimensions. I applied the method to visualise spectral databases of soft sensors by preserving neighbourhood and distance relations of NIR spectra. [137, 145, 148]

   (b) I modified the cost function of the genetic programming to support the visualisation of classification problems. The results confirm that the performance of traditional classifiers improves, when we apply them on goal-oriented projected data. Additionally, I have defined a new classifier that uses convex polygons and also generates an informative view to the user. I have shown that the algorithm can separate the operational regimes of a technology hence helps to define local models of soft sensors. [137, 142, 150, 151]

2. **I developed parametric models for spectrometric applications and target calculation in energy monitoring systems. I worked out methods to accelerate the modelling process and can generate informative vislualisations to show the hidden structure and the validity range of the models.**
   (Related publications: [136, 138, 139, 141–143, 146, 147, 149])

   (a) I developed a validation process that can qualify the modelling performance and can determine the validity range of spectrometric models. I presented models that can visualise and explore the hidden structures in the training dataset. I compared data from a flow-through cell, and a fibre optic spectrometer and proved that the more cost efficient fibre optic system has similar performance as the flow through cell system has. [136, 141, 147, 149]

   (b) I proved that Self-Organizing-Maps (SOM) can separate the operating modes in energy monitoring (EM) systems and I worked out a SOM based feature ranking and selection tool. [138, 139, 141]

   (c) I implemented a Random Forest (RF) regression based feature selection algorithm as an extension of the developed framework. I used the RF to select relevant input variables for EM targeting models. [139]

3. **I developed goal-orineted multivariate time series analysis methods to support the identification of parametric and non-parametric models used in NIR analyser based soft sensors and energy monitoring systems.**

   (Related publications: [138–140, 147, 149, 150])

   (a) I demonstrated that Principal Component Analysis based times series segmentation can be used to find consistent operating periods of production and events affecting the dynamical behaviour of the process. [138, 141]

   (b) I developed a novel regression-based times series algorithm to detect homogeneous periods of operation based on the prediction accuracy of energy targeting models. The method is applicable to identify events when energy efficiency differs significantly. [138, 140]

   (c) I demonstrated that Self-Organizing-Maps can not only be used to isolate operating modes and to define local models for the individual operating regimes, but it can also be applied to feature selection. The results illustrate that all of these functionalities support the building of compact models used in energy monitoring. [140]

## 5.3 Utilization of results

A part of the results presented in the dissertation has been already utilised. A new practice has been introduced to maintenance and to upgrade the model that is running in the ABB spectrometer of the Diesel Blending unit in MOL Duna Refinery. This method includes the generation of new explicit mapping equations (aggregates). A fibre optic spectrometer was installed into one of the experimental reactors of the product development department, and it is being used to trace the experiments. In this system, partial least squares models are providing the estimations for the material properties of the product. The feature selection methods are being utilised to build new targeting models into the energy monitoring systems.

The dissertation presented the application of the genetic algorithm to generate dimensional reduction mappings of spectral datasets. However besides of chemical applications this tools could be used effectively in any other data mining problems.

The presented time series evaluation methods have been already utilised in telecommunication, more specifically in the platforms that are supplied by I-New Unified Mobil Solutions. Hence, these practices can detect several fault and incident patterns before they cause the complete service outage. The time series analysis has been included into the monitoring system of the Mobil Virtual Network Operators (MVNOs) of Virgin Mobile Colombia, Virgin Mobile Chile, Compass and other providers.

## 5.4. Tézisek

1. **Sokváltozós adathalmaz megjelenítésére alkalmas genetikus programozáson alapuló algoritmust fejlesztettem és alkalmaztam szoftver szenzorok működési tartományainak feltérképezésére és osztályozási feladatok megoldására.**
   (Kapcsolódó publikációk: [137, 142, 144, 145, 148, 150, 151])

   (a) Többkromoszómás reprezentációra alapozva létrehoztam egy sokdimenziós térből alacsonyabb dimenziós térbe történő explicit leképezések keresésére alkalmas genetikus algoritmust. A módszert első lépésben spektrális adatbázisok dimenzió csökkentésére használtam fel úgy, hogy a leképezés a lehető legnagyobb mértékben megőrizze az eredeti tér távolsági és szomszédossági viszonyait. Megmutattam, hogy a genetikus algoritmus alkalmazható spektrometriai modellek fejlesztésében és karbantartásában. [137, 145, 148]

   (b) A kifejlesztett genetikus algoritmust osztályozási feladatok támogatásra is alkalmazhatóvá tettem. Az osztályozó algoritmusok módosítás nélküli felhasználásán és az osztályozók teljesítményére alapuló költségfüggvényén alapuló módszert referencia adatsorokon alkalmaztam, és számítási példákkal igazoltam, hogy a genetikus algoritmussal létrehozott explicit leképezéssel kiegészítve a hagyományos osztályozó eljárások osztályozási pontossága nagy mértékben javítható. Megmutattam továbbá, hogy célirányos költségfüggvényeket alkalmazva a genetikus algoritmus működési tartományok elkülönítésére is alkalmas, egyaránt támogatva ezzel a parameterikus és nemparametrikus modellek fejlesztését. [137, 142, 150, 151]

2. **Parametrikus modellek fejlesztését támogató keretrendszert hoztam létre, melyet spektroszkópiai regressziós modellek készítésére és az energiamonitoring rendszerekben az energiafelhasználás célértékének meghatározására használtam fel. Megmutattam, hogy a keretrendszert felhasználva a modellek fejlesztése gyorsítható és informatív megjelenítés készíthető a modellek struktúrájáról és érvényességi tartományáról.**
(Kapcsolódó publikációk: [136, 138, 139, 141–143, 146, 147, 149])

    (a) A közeli infravörös online és labor spektrométerekre támaszkodó anyagi jellemzők becslésére alkalmazott parametrikus regressziós modellek validálására alkalmas eljárást fejlesztettem. Megmutattam, hogy ortogonális jelkompenzációt alkalmazó regressziós modellel a szoftver szenzorok érvényességi tartománya behatárolható. [136, 141, 147, 149]

    (b) Működési tartományok vizsgálata kapcsán megmutattam, hogy az önszervező térkép (SOM) kiválóan alkalmazható Energia Monitoring (EM) rendszerekben az egyes működési módok elkülönítésére és az energiafelhasználást meghatározó folyamatváltozók azonosítására. [138, 139, 141]

    (c) A keretrendszer részeként implementáltam egy Random Forest (RF) változó szelekciós algoritmust, melyet EM modellek bemeneti változóinak kiválasztására használtam fel. Az eljárás alkalmazhatóságát az adott üzem esetében a legnehezebben becsülhető energia jellemző esetén is sikeresen igazoltam. [139]

3. **Szoftver szenzorok modelljeinek fejlesztése céljából többváltozós idősorok elemzésére alkalmas algoritmusokat hoztam létre és sikeresen alkalmaztam spektrális adatbázisok elemzésében és energia monitoring rendszerben alkalmazott modellek identifikálásában.** (Kapcsolódó publikációk: [138–140, 147, 149, 150])

   (a) Kimutattam, hogy a főkomponens-elemzésen alapuló többváltozós idősorszegmentáló algoritmus alkalmas spektrális idősorok alapján homogén működési szakaszok meghatározására, illetve az üzemmenet jelentősebb változásait eredményező események azonosítására. [138, 141]

   (b) Regressziós modelleken alapuló idősorszegmentáló algoritmust fejlesztettem azoknak az időintervallumoknak az azonosítására melyeken belül a szoftver szenzor (lokális) modellje az elvárt becslési pontossággal rendelkezik. Az algoritmust sikeresen alkalmaztam energia monitoring rendszerek lokális érvényességi tartományú modelljeinek identifikálására. [138, 140]

   (c) Megmutattam, hogy az önszervező térkép működési tartományok szegmentálására és az adott szegmenseken belüli lokális modellek készítésére is alkalmazható. A fejlesztett módszert integráltam SOM alapú változó kiválasztással, hatékonyságát energia monitoring példán demonstráltam. [140]

# Appendix A

# Dimensional Reduction and false nearest neighbor method

## A.1 Principal Component Analysis

One of the most widely applied dimensionality reduction method is the *Principal Component Analysis* (PCA) [96]. PCA is also known as Hotteling or as Karhunen-Loéve transformation [96]. PCA differs from metric and non-metric dimensionality reduction methods, because instead of the preservation of the distances or the global ordering relations of the objects (in this case spectra) it tries to preserve the variance of the data. PCA represents the data as linear combinations of a small number of basis vectors. This method finds the projection that stores the largest variance possible in the original data and rotates the set of the objects such that the maximum variability becomes visible. Geometrically, PCA transforms the data into a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate, the second greatest variance on the second coordinate, and so on. If the data set ($\mathbf{X}$) is characterized with $D$ dimensions and the aim of the PCA is to find the $d$-dimensional reduced representation of the data set, the PCA works as follows:

1. PCA subtracts the mean from each of the data dimensions,
2. then it calculates the $D \times D$ covariance matrix of the data set,

3. following this PCA calculates the eigenvectors and the eigenvalues of the covariance matrix,
4. then it chooses the $d$ largest eigenvectors,
5. and finally it derives the new data set from the significant eigenvectors and from the original data matrix.

The corresponding $d$-dimensional output is found by linear transformation: $\mathbf{Y} = \mathbf{QX}$, where $\mathbf{Q}$ is the $d \times D$ matrix of linear transformation composed of the $d$ largest eigenvectors of the covariance matrix, and $\mathbf{Y}$ is the $d \times D$ matrix of the projected data set. *Independent Component Analysis* (ICA) is similar to PCA, except that it tries to find components that are independent.

## A.2 Multidimensional scaling (MDS)

*Multidimensional scaling* (MDS)[11] refers to a group of unsupervised data visualization techniques. Given a set of data in a high-dimensional feature space, MDS maps them into a low-dimensional (generally 2-dimensional) data space in a way that objects that are very similar to each other in the original space are placed near each other on the map, and objects that are very different from each other are placed far away from each other. There are two types of MDS: (i) *metric MDS* and (ii) *non-metric MDS.*

The classical MDS algorithm is an algebraic method that rests on the fact that matrix $\mathbf{Y}$ containing the output coordinates can be derived by eigenvalue decomposition from the scalar product matrix $\mathbf{B} = \mathbf{YY}^T$. Matrix $\mathbf{B}$ can be found from the known distances using Young-Householder process [84].

The *metric (or classical) MDS* discovers the underlying structure of data set by preserving similarity information (pairwise distances) among the data objects. Similarly to the Sammon mapping the metric multidimensional scaling also tries to minimize a stress function. If the square-error cost is used, the objective function (stress) to be minimized can be written as:

$$E_{metric\_MDS} = \frac{1}{\sum\limits_{i<j}^{N} d_{i,j}^{*2}} \sum_{i<j}^{N} (d_{i,j}^{*} - d_{i,j})^2, \tag{A.1}$$

where $d^*_{i,j}$ denotes the distance between the vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, and $d_{i,j}$ between $\mathbf{y}_i$ and $\mathbf{y}_j$ respectively. The only difference between the stress functions of the Sammon mapping (see A.3) and the metric MDS (see A.1) is that the errors in distance preservation are normalized by the distances of the input data objects. Because of this normalization the Sammon mapping emphasizes the preservation of small distances.

In *non-metric MDS* only the ordinal information of the proximities is used for constructing the spatial configuration, thereby the non-metric MDS attempts to preserve the rank order among the dissimilarities. The non-metric MDS finds a configuration of points whose pairwise Euclidean distances have approximately the same rank order as the corresponding dissimilarities of the objects. Equivalently, the non-metric MDS finds a configuration of points, whose pairwise Euclidean distances approximate a monotonic transformation of the dissimilarities. These transformed values are known as the disparities. The non-metric MDS stress can be formulated as follows[1]:

$$E_{nonmetric\_MDS} = \sqrt{\sum_{i<j}^{N} (\widehat{d}_{i,j} - d_{i,j})^2 / \sum_{i<j}^{N} d^2_{i,j}}, \tag{A.2}$$

where $\widehat{d}_{i,j}$ yields the disparity of $\mathbf{x}_i$ and $\mathbf{x}_j$, and $d_{i,j}$ denotes the distance between the vectors $\mathbf{y}_i$ and $\mathbf{y}_j$.

It can be shown, that the metric and non-metric MDS mappings are substantially different methods. While the metric MDS algorithm is an algebraic method, the non-metric MDS is an iterative mapping process.

1. Let the searched coordinates of $n$ points in a $d$-dimensional Euclidean space be given by $\mathbf{y}_i \quad (i = 1, \ldots, n)$, where $\mathbf{y}_i = [y_{i,1}, \ldots, y_{i,d}]^T$. Matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^T$ is the $n \times d$ coordinates matrix. The Euclidean distances $\{d_{i,j} = (\mathbf{y}_i - \mathbf{y}_j)^T (\mathbf{y}_i - \mathbf{y}_j)\}$ are known. The inner product of matrix $\mathbf{Y}$ is denoted $\mathbf{B} = \mathbf{Y}\mathbf{Y}^T$. Find matrix $\mathbf{B}$ from the known distances $\{d_{i,j}\}$ using Young-Householder process[84]:

   (a) Define matrix $\mathbf{A} = [a_{i,j}]$, where $a_{i,j} = -\frac{1}{2} d^2_{i,j}$,

[1] Traditionally, the non-metric MDS stress is often called Stress-1 due to Kruskal [49]

(b) Deduce matrix $\mathbf{B}$ from $\mathbf{B} = \mathbf{HAH}$, where $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{l}\mathbf{l}^T$ is the centering matrix, and $\mathbf{l}$ is an $n \times 1$ column vector of $n$ one's

2. Recover the coordinates matrix $\mathbf{Y}$ from $\mathbf{B}$ using the spectral decomposition of $\mathbf{B}$:

   (a) The inner product matrix $\mathbf{B}$ is expressed as $\mathbf{B} = \mathbf{Y}\mathbf{Y}^T$. The rank of $\mathbf{B}$ is $r(\mathbf{B}) = r\left(\mathbf{Y}\mathbf{Y}^T\right) = r(\mathbf{Y}) = d$. $\mathbf{B}$ is symmetric, positive semi-definite and of rank $d$, and hence has $d$ non-negative eigenvalues and $n - d$ zero eigenvalues.

   (b) Matrix $\mathbf{B}$ is now written in terms of its spectral decomposition, $\mathbf{B} = \mathbf{V}\Lambda\mathbf{V}^T$, where $\Lambda = diag\left[\lambda_1, \lambda_2, \ldots, \lambda_n\right]$ the diagonal matrix of eigenvalues $\lambda_i$ of $\mathbf{B}$, and $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ the matrix of corresponding eigenvectors, normalized such that $\mathbf{v}_i^T\mathbf{v}_i = \mathbf{l}$,

   (c) Because of the $n - d$ zero eigenvalues, $\mathbf{B}$ can now be rewritten as
   $\mathbf{B} = \mathbf{V}_1\Lambda_1\mathbf{V}_1^T$, where
   $\Lambda_1 = diag\left[\lambda_1, \lambda_2, \ldots, \lambda_d\right]$ and $\mathbf{V}_1 = [\mathbf{v}_1, \ldots, \mathbf{v}_d]$,

   (d) Finally the coordinates matrix is given by $\mathbf{Y} = \mathbf{V}_1\Lambda_1^{\frac{1}{2}}$, where $\Lambda_1^{\frac{1}{2}} = diag\left[\lambda_1^{\frac{1}{2}}, \ldots, \lambda_d^{\frac{1}{2}}\right]$.

## A.3 Sammon Mapping

*Sammon mapping* [71] (SM) is a metric, nonlinear dimensionality reduction method which maps the set of points in a high-dimensional vector space onto a $d$-dimensional output space. While PCA attempts to preserve the variance of the data during the mapping, Sammon's mapping try to preserve the interpattern distances [55, 65]. The Sammon mapping tries to optimize the cost function that describes how well the pairwise distances in a data set are preserved. The Sammon stress function (distortion of the Sammon projection) can be written as:

$$E_{SM} = \frac{1}{\sum\limits_{i<j}^{N} d_{i,j}^*} \sum_{i<j}^{N} \frac{\left(d_{i,j}^* - d_{i,j}\right)^2}{d_{i,j}^*}, \tag{A.3}$$

where $d^*_{i,j}$ denotes the distance between the vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, and $d_{i,j}$ respectively for $\mathbf{y}_i$ and $\mathbf{y}_j$.

The minimization of the Sammon stress is an optimization problem. When the gradient-descent method is applied to search for the minimum of Sammon stress, a local minimum can be reached. Therefore a significant number of runs with different random initializations may be necessary.

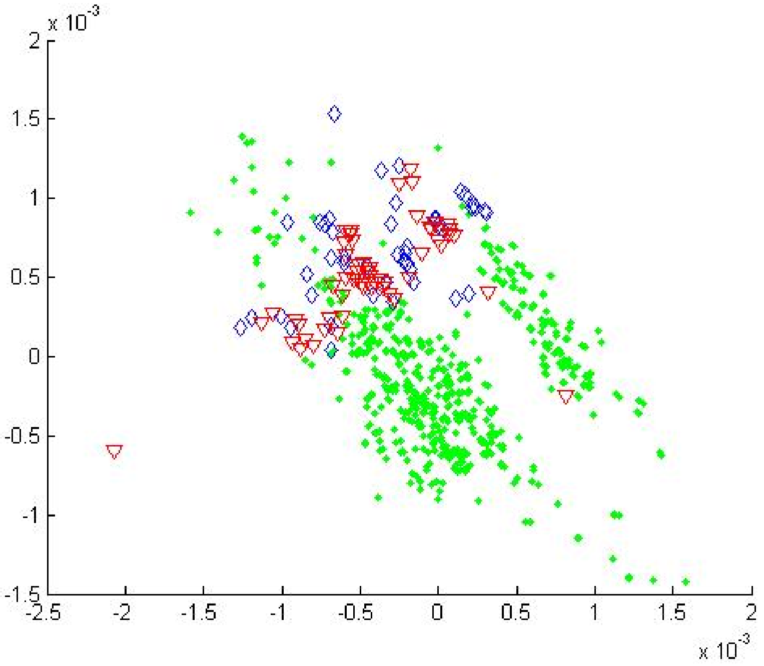

FIGURE A.1. Sammon mapping of the TOPNIR database and samples collected from online (blue) and laboratory (red) analysers

## A.4 False Nearest Neighbor (FNN) Method

The main idea of the FNN algorithm stems from the basic property of a function. If there is enough information in the regression vector to predict the future output, then any of two regression vectors which are close in the regression space will also have future outputs which are close in some sense. For all regression vectors embedded in the proper dimensions, for two regression vectors that are close in the regression space and their corresponding outputs are related in the following way:

$$y_i - y_j = df\left(\mathbf{x}_i\right)\left[\mathbf{x}_i - \mathbf{x}_j\right] + o\left(\left[\mathbf{x}_i - \mathbf{x}_j\right]\right)^2 \qquad (A.4)$$

where $df\left(\mathbf{x}_i\right)$ is the jacobian of the function $f(.)$ at $\mathbf{x}_i$.

Ignoring higher order terms, and using the Cauchy-Schwarz inequality the following inequality can be obtained:

$$|y_i - y_j| \leq \|df\,(\mathbf{x}_i)\|_2\,\|\mathbf{x}_i - \mathbf{x}_j\|_2 \tag{A.5}$$

$$\frac{|y_i - y_j|}{\|\mathbf{x}_i - \mathbf{x}_j\|_2} \leq \|df\,(\mathbf{x}_i)\|_2 \tag{A.6}$$

If the above expression is true, then the neighbors are recorded as true neighbors. Otherwise, the neighbors are false neighbors.

Based on this theoretical background, the outline of the FNN algorithm is the following.

1. Identify the nearest neighbor to a given point in the regressor space. For a given regressor: $\mathbf{x}_i$ find the nearest neighbor $\mathbf{x}_j = \mathbf{x}_{(i,1)}$.

2. Determine if the following expression is true or false

$$\frac{|y_i - y_j|}{||\mathbf{x}_i - \mathbf{x}_j||_2} \leq R$$

   where $R$ is a previously chosen threshold value. If the above expression is true, then the neighbors are recorded as true neighbors. Otherwise, the neighbors are false neighbors.

3. Continue the algorithm for all times $i$ in the data set.

The FNN algorithm is sensitive to the choice of the $R$ threshold. In the threshold value was selected by trial and error method based on empirical rules of thumb, $10 \leq R \leq 50$. However, choosing a single threshold that will work well for all data sets is impossible task. In this case, it is advantageous to estimate $R$ based on A.6 using the the maximum of the Jacobian, $R = max_i\,\|df\,(\mathbf{x}_i)\|$, as it was suggested by Rhodes and Morari.

While this method uses data based models for the estimation of$\|df\,(\mathbf{x}_i)\|$, the performance and the capabilities of this identified model can deteriorate the estimate of $max(df)$. When $df$ is over estimated the model orders could be under estimated, and vice-versa. Hence, the modeler has to be careful at the construction of this model (e.g. the model can be over or under parameterized, etc.).

# List of Figures

# List of Tables

# Acronyms

| | |
|---|---|
| **ABC** | **A**ctivity **B**ased **C**osting |
| **ABE** | **A**ctivity **B**ased **E**nergy |
| **ANN** | **A**rtificial **N**eural **N**etwork |
| **CAE** | **C**urrent **A**verage **E**nergy |
| **EM** | **E**nergy **M**onitoring |
| **FNN** | **F**alse **N**earest **N**eighbors |
| **GA** | **G**enetic **A**lgorithm |
| **GP** | **G**enetic **P**rogramming |
| **ITP** | **I**ndustrial **T**echnologies **P**rogram |
| **KEI** | **K**ey **E**nergy **I**ndicator |
| **KNN** | **K** **N**earest **N**eighbors |
| **LLE** | **L**ocally **L**inear **E**mbedding |
| **MDS** | **M**ulti **D**imensional **S**caling |
| **MLP** | **M**ulti **L**ayer **P**ercepcion |
| **MPC** | **M**odel **P**redictive **C**ontrol |
| **MSPM** | **M**ultivariate **S**tatistical **P**rocess **M** Monitoring |
| **NIR** | **N**ear **I**nfra**R**ed |
| **NFS** | **N**euro-**F**uzzy **S**ystem |
| **OLS** | **O**rthogonal **L**east **S**quares |
| **PAT** | **P**rocess **A**nalytical **T**echnology |
| **PCA** | **P**rincipal **C**omponenet **A**nalyisis |
| **PLS** | **P**artial **L**east **S**quares |
| **PME** | **P**ractical **M**inimum **Energy** |

| | |
|---|---|
| **PS** | **P**attern **S**earch |
| **RBFN** | **R**adial **B**asis **F**oundation **N**etworks |
| **RCA** | **R**oot **C**ause **A**nalysis |
| **RF** | **R**andom **F**orest |
| **RFR** | **R**andom **F**orest **R**egression |
| **RNN** | **R**ecurrent **N**eural **N**etwork |
| **SPC** | **S**tatistical **P**rocess **C**ontrol |
| **SPE** | **S**quared **P**rediction **E**rror |
| **SOM** | **S**elf-**O**rganising **M**ap |
| **SVM** | **S**upport **V**ector **M**achine |
| **TME** | **T**heoretical **M**inimum **E**nergy |
| **TOPNIR** | **TOP**ological **N**ear **I**frared **R**ed |

# List of Programs

| Reference | Run command |
|---|---|
| [PR1] | Single chromosome evaluation that generates a second aggregate if one of the aggregates is fixed to the predefined *Naro*.<br>Chapter_2/A_SpectraMap/SingleChromNaro/AgrSearchPairNaro.m |
| [PR2] | Single chromosome evaluation that generates a second aggregate if one of the aggregates is fixed to the predefined *Parox*.<br>Chapter_2/A_SpectraMap/SingleChromParox/AgrSearchPairParox.m |
| [PR3] | Multi-chromosome aggregate generation for spectral database mapping. In this evaluation there is no fixed aggregate.<br>Chapter_2/A_SpectraMap/MultiCrom/Breeding.m |
| [PR4] | Multi-chromosome evaluation that creates mapping that preprocesses data to improve the embedded C4.5 classification.<br>Chapter_2/A_Wine/B_Cluster_DT_MC/Breeding.m |
| [PR5] | Standard discriminant analysis of the standard Wine datasets. The program calculates the classification metrics to give reference values for [PR4].<br>Chapter_2/A_Wine/E_Discriminance/WineDiscriminant.m |
| [PR6] | This program creates regression models for the EUBio dataset. The algorithm generates PLS models and exports data for [PR8].<br>Chapter_3/Results/Results_EUBio_v2.m |
| [PR7] | This program creates regression models for the AV2 dataset. The algorithm generates PLS models and exports data for [PR8].<br>Chapter_3/Results/Results_AV2_v2.m |

| | |
|---|---|
| [PR8] | This program visualise the result os [PR6] and [PR7]<br>Chapter_3/Results/effplot.m |
| [PR9] | Self Organizing Map Random Forest and OLS based feature selection for energy monitoring systems.<br>Chapter_4/C_EM_Feature_Selection/Teszt_03_Futoanyag.m<br>Chapter_4/C_EM_Feature_Selection/Teszt_03_Uzemgoz.m<br>Chapter_4/C_EM_Feature_Selection/Teszt_03_Villany.m |
| [PR10] | Correlation based segmentation of spectral time series.<br>Chapter_4/AB_CorrSegment/CorrSegment_01.m |

# Symbols

| | |
|---|---|
| $k$ | index of sample |
| $N$ | number of samples in dataset $k \in [1, \dots, N]$ |
| $x_{i,k}$ | $i$th variable of $\mathbf{x}_k$ |
| $i$ | index of variable, $i \in [1, \dots, n]$ |
| $n$ | number of variables |
| $\vec{x}_k$ | vector of variables of $k$th sample |
| $\mathbf{X}$ | matrix of variables with size $N \times n$ |
| $\mathbf{F}$ | covariance matrix of $x_i$ variables |
| $\mathbf{\Lambda}$ | diagonal matrix of eigenvalues $\lambda_k$ of $X^T X$ |
| $\mathbf{U}$ | matrix whose columns are the eigenvectors of $X^T X$ |
| $I$ | identity matrix |
| $s$ | standard deviance |
| $\hat{y}$ | estimated value |
| $\theta$ | regression model's parameter vector |
| $R_i$ | nearest neighbors range of $i$th sample |
| $d_{i,j}$ | distance measure |
| $\beta_{i,j}$ | weighting function |
| $E_v$ | experimental error |
| $i_m$ | minimal index |
| $D$ | dimensionality, number of dimensions |
| $g_i$ | OLS solution vector element |
| $\Theta$ | Random matrix for tree growing |

# References

[1] Janos Abonyi. *Fuzzy Model Identification*. Springer, 2003.

[2] Janos Abonyi. Fuzzy model identification. In *Fuzzy Model Identification for Control*. Birkhäuser Boston, 2003.

[3] Janos Abonyi, Robert Babuska, and Ferenc Szeifert. Modified gath-geva fuzzy clustering for identification of takagi-sugeno fuzzy models. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 32(5): 612–621, 2002.

[4] Janos Abonyi, Balazs Feil, Sandor Nemeth, and Peter Arva. Modified gath-geva clustering for fuzzy segmentation of multivariate time-series. *Fuzzy Sets and Systems*, 149(1):39–56, 2005.

[5] J. Madar, J. Abonyi, and F. Szeifert. Genetic programming for the identification of nonlinear input-output models. *Industrial and Engineering Chemistry Research*, 44(9):3178–3186, 2005.

[6] Mark A. Abramson. *Pattern Search Filter Algorithms for Mixed Variable General Constrained Optimization Problems*. Phd thesis, Air Force Institute of Technology, Department of Mathematics and Statistics, 2002. PhD Thesis, Department of Computational and Applied Mathematics, Rice University.

[7] Charles Audet and John E. Dennis Jr. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2003.

[8] Clive Beggs. Chapter 9 - energy monitoring, targeting and waste avoidance. In Clive Beggs, editor, *Energy: Management, Supply and Conservation (Second Edition)*, pages 158 – 172. Butterworth-Heinemann, Oxford, second edition edition, 2009. ISBN 978-0-7506-8670-9. doi: 10.1016/B978-0-7506-8670-9.00009-8.

[9] Thomas Behrendt, André Zein, and Sangkee Min. Development of an energy consumption monitoring procedure for machine tools. *CIRP Annals - Manufacturing Technology*, 61(1):43 – 46, 2012. ISSN 0007-8506. doi: 10.1016/j.cirp.2012.03.103.

[10] Dayal Bhupinder and MacGregor. Improved pls algorithms. *Journal of Chemometrics*, 11(1):73–85, 1997.

[11] I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer Series in Statistics. Springer Verlag, New York, 1997.

[12] Jean-Florian Brau, Matteo Morandin, and Thore Berntsson. Hydrogen for oil refining via biomass indirect steam gasification: energy and environmental targets. *Clean Technologies and Environmental Policy*, 15(3):501–512, 2013. ISSN 1618-954X. doi: 10.1007/s10098-013-0591-9.

[13] Christopher J. C. Burges. *Dimension Reduction: A Guided Tour*, volume 2. Springer, 2010. doi: 10.1561/2200000002.

[14] T. P. Chemaly and C. Aldrich. Visualization of process data by use of evolutionary computation. *Computers and Chemical Engineering*, 25(9-10): 1341–1349, 2001.

[15] Zengping Chen, David Lovett, and Julian Morris. Process analytical technologies and real time process control a review of some spectroscopic issues and challenges. *Journal of Process Control*, 21(10):1467 – 1482, 2011.

[16] Christian Cimander, Thomas Bachinger, and Carl-Fredrik Mandenius. Integration of distributed multi-analyzer monitoring and control in bioprocessing

based on a real-time expert system. *Journal of Biotechnology*, 103(3):237 – 248, 2003.

[17] P. Comon. Independent component analysis: a new concept? *Signal Processing*, 36(3):287–317, 1994.

[18] William Cox, Toby Considine, and TC Principal. Price communication, product definition, and service-oriented energy, 2009.

[19] B. Descales, D. Lambert, J.R. Llinas, A. Martens, S. Osta, M. Sanchez, and S. Bages. Method for determining properties using near infra-red (nir) spectroscopy, 2000. US6.070.128.

[20] Joost R. Duflou, John W. Sutherland, David Dornfeld, Christoph Herrmann, Jack Jeswiet, Sami Kara, Michael Hauschild, and Karel Kellens. Towards energy and resource efficient manufacturing: A processes and systems approach. *CIRP Annals - Manufacturing Technology*, 61(2):587 – 609, 2012. ISSN 0007-8506. doi: 10.1016/j.cirp.2012.05.002.

[21] Incorporated Energetics. Energy bandwidth for petroleum refining processes. Prepared by Energetics Incorporated for the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Industrial Technologies Program, 2006.

[22] R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[23] Luigi Fortuna. *Soft sensors for monitoring and control of industrial processes*. Springer, 2007.

[24] E. Giacone and S. Mancò. Energy efficiency measurement in industrial processes. *Energy*, 38(1):331 – 345, 2012. ISSN 0360-5442. doi: 10.1016/j.energy.2011.11.054.

[25] K. Kumar Gidwani and Robert F. Beckman. Evaluation of refinery control systems. *SA Transactions*, 33(3):217 – 225, 1994.

[26] G.D. Gonzalez. Soft sensors for processing plants. In *Intelligent Processing and Manufacturing of Materials, 1999. IPMM '99. Proceedings of the Second International Conference on*, volume 1, pages 59–69, 1999. doi: 10.1109/IPMM.1999.792454.

[27] G.D. Gonzalez, J.P. Redard, R. Barrera, and M. Fernandez. Issues in soft-sensor applications in industrial plants. In *Industrial Electronics, 1994. Symposium Proceedings, ISIE '94., 1994 IEEE International Symposium on*, pages 380–385, 1994. doi: 10.1109/ISIE.1994.333086.

[28] A.N. Gorban, B. Kegl, D.C. Wunsch, and A. Zinovyev. *Principal Manifolds for Data Visualization and Dimension Reduction*, volume 58 of *Lecture Notes in Computational Science and Engineering*. Springer, 2008. ISBN 978-3-540-73749-0.

[29] R. Grbic, D. Sliskovic, and P. Kadlec. Adaptive soft sensor for online prediction based on moving window gaussian process regression. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, volume 2, pages 428–433, 2012. doi: 10.1109/ICMLA.2012.160.

[30] CésarG. Gutiérrez-Arriaga, Medardo Serna-González, JoséMaría Ponce-Ortega, and MahmoudM. El-Halwagi. Multi-objective optimization of steam power plants for sustainable generation of electricity. *Clean Technologies and Environmental Policy*, 15(4):551–566, 2013. ISSN 1618-954X. doi: 10.1007/s10098-012-0556-4.

[31] Isabelle Guyon, Steve Gunn, Masoud Nikravesh, and Lotfi A. Zadeh. *Feature Extraction: Foundations and Applications*, volume 207 of *Studies in Fuzziness and Soft Computing*. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-35487-1.

[32] J. Himberg, K. Korpiaho, H. Mannila, J. Tikanmaki, and H.T. Toivonen. Time-series segmentation for context recognition in mobile devices. In *International Conference on Data Mining*, pages 466–473, 2001.

[33] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Education Psychology*, 24:417–441, 1933.

[34] Dan Hou, Shuai Shao, Yun Zhang, SuLing Liu, Yu Chen, and ShuShen Zhang. Exergy analysis of a thermal power plant using a modeling approach. *Clean Technologies and Environmental Policy*, 14(5):805–813, 2012. ISSN 1618-954X. doi: 10.1007/s10098-011-0447-0.

[35] Industrial Technology Program. Industrial technology program, tools to improve energy efficiency, June 2014. URL `http://energy.gov/eere/downloads/amo-software-tools`.

[36] T. Kourtl J.F. MacGregor. Statistical process control of multivariate processes. *Control Fag. Practice*, 3(3):403–414, 1995.

[37] A. Jain and D. Zongker. Feature selection: evaluation, application, and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 192:153–158, 1997.

[38] T. Jolliffe. *Principal Component Analysis*. Springer, New York, 1996.

[39] Paul Jurek, Bert Bras, Tina Guldberg, Jim D'Arcy, Seog-Chan Oh, and Stephan Biller. Activity-based costing applied to automotive manufacturing. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–7. IEEE, 2012.

[40] Petr Kadlec, Bogdan Gabrys, and Sibylle Strandt. Data-driven soft sensors in the process industry. *Computers and Chemical Engineering*, 33(4):795 – 814, 2009.

[41] Petteri Kampjarvi, Mauri Sourander, Tiina Komulainen, Nikolai Vatanski, Mats Nikus, and Sirkka-Liisa Jamsa-Jounelac. Fault detection and isolation of an on-line analyzer for an ethylene cracking process. *Control Engineering Practice*, 16:1–13, 2008.

[42] Manabu Kano and Yoshiaki Nakagawa. Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry. *Computers and Chemical Engineering*, 32:12 – 24, 2008.

[43] Manabu Kano and Morimasa Ogawa. The state of the art in chemical process control in japan: Good practice and questionnaire survey. *Journal of Process Control*, 20(9):969 – 982, 2010.

[44] S. Kaski, J. Nikkilä, M. Oja, J. Venna, J. Törönen, and E. Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC Bioinformatics*, 4(48), 2003.

[45] K. Kellens, W. Dewulf, B. Lauwers, J.-P. Kruth, and J.-R. Duflou. Environmental impact reduction in discrete manufacturing: Examples for non-conventional processes. *Procedia CIRP*, 6:27 – 34, 2013. ISSN 2212-8271. doi: 10.1016/j.procir.2013.03.003.

[46] E. Keogh, S. Chu, D. Hart, and H. Pazzani. An online algorithm for segmenting time series. *IEEE International Conference on Data Mining*, 2001. URL: http://citeseer.nj.nec.com/keogh01online.html.

[47] T. Kohonen. *Self-Organizing Maps*. Springer, third edition, 2001.

[48] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992. ISBN 0-262-11170-5.

[49] J.B. Kruskal. Multidimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis. *Psychometrika*, 29:1–29, 1964.

[50] I. Lee and J. Yang. Common clustering algorithms. *Comprehensive Chemometrics*, pages 577–618, 2009.

[51] Huan Liu and Hiroshi Motada. *Feature Extraction, Construction and Selection: A Data Mining Perspective*. Springer, first edition, 1998. ISBN 978-1-4615-5725-8. doi: 10.1007/978-1-4615-5725-8.

[52] JF MacGregor and Th Kourti. Statistical process control of multivariate processes. *Control Engineering Practice*, 3(3):403–414, 1995.

[53] Oded Maimon and Lior Rokach. *Data Mining and Knowledge Discovery Handbook*. Springer, second edition, 2010. ISBN 978-0-387-09823-4. doi: 10.1007/978-0-387-09823-4.

[54] SabujKumar Mandal and S. Madheswaran. Energy use efficiency of indian cement companies: a data envelopment analysis. *Energy Efficiency*, 4(1): 57–73, 2011. ISSN 1570-646X. doi: 10.1007/s12053-010-9081-7.

[55] J. Mao and A.K. Jain. Artifical neural networks for feature extraction and multivariate data projection. *IEEE Trans. Neural Networks*, pages 629–637, 1995.

[56] S. Mimaroglu and E. Erdil. Combining multiple clusterings using similarity graph. *Pattern Recognition*, 44(3):694–703, 2011.

[57] Keith Moss. *Monitoring and Targeting*, chapter Chapter 10. Routledge, 2010. ISBN 9780415353915. doi: 10.4324/9780203349021.

[58] Roderick Murray-Smith and T Johansen. *Multiple Model Approaches to Nonlinear Modelling and Control*. CRC press, 1997.

[59] Makoto Nakaya and Xinchun Li. On-line tracking simulator with a hybrid of physical and just-in-time models. *Journal of Process Control*, 23(2):171 – 178, 2013.

[60] P. Narendra and K. Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, C-269:917–922, 1977.

[61] Oliver Nelles. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer, 2001.

[62] Seog-Chan Oh and Alfred J Hildreth. Decisions on energy demand response option contracts in smart grids based on activity-based costing and stochastic programming. *Energies*, 6(1):425–443, 2013.

[63] Solomon Oji. Case study - online energy management and optimisation of utility generation assets on industrial sites. In Ian David Lockhart Bogle and Michael Fairweather, editors, *22nd European Symposium on Computer Aided Process Engineering*, volume 30 of *Computer Aided Chemical Engineering*, pages 337 – 341. Elsevier, 2012. doi: 10.1016/B978-0-444-59519-5.50068-X.

[64] Abass A. Olajire. The brewing industry and environmental challenges. *Journal of Cleaner Production*, 2012. ISSN 0959-6526. doi: 10.1016/j.jclepro.2012.03.003.

[65] N.R. Pal and V.K. Eluri. Two efficient connectionist schemes for structure preserving dimensionality reduction. *IEEE Transactions on Neural Networks*, 9:1143–1153, 1998.

[66] P. Pudil, J. Novovičová, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15(1):1119–1125, 1994.

[67] S Ramasubramanian, Y Avinash, S Pragathi Chitra, T Geetha, and S Anand. An activity based approach to minimize energy usage of service sector infrastructure. In *Infrastructure Systems and Services: Developing 21st Century Infrastructure Networks,(INFRA), 2009 Second International Conference on*, pages 1–6. IEEE, 2009.

[68] T. Retsina. Method and system for targeting and monitoring the energy performance of manufacturing facilities, September 5 2006. URL `http://www.google.com/patents/US7103452`. US Patent 7,103,452.

[69] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[70] R. Saidur, M.T. Sambandam, M. Hasanuzzaman, D. Devaraj, S. Rajakarunakaran, and M.D. Islam. An energy flow analysis in a paper-based industry. *Clean Technologies and Environmental Policy*, 14(5):905–916, 2012. ISSN 1618-954X. doi: 10.1007/s10098-012-0462-9.

[71] J.W. Sammon. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18(5):401–409, 1969.

[72] Yaser R. Sonbul. Topological near infrared analysis modeling of petroleum refinery products, 2005. US6.897.071 B2.

[73] J.B. Tenenbaum, V. Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

[74] S. Thiede, G. Bogdanski, and C. Herrmann. A systematic method for increasing the energy and resource efficiency in manufacturing companies. *Procedia CIRP*, 2:28 – 33, 2012. ISSN 2212-8271. doi: 10.1016/j.procir.2012.05.034.

[75] Carbon Trust. Monitoring and tarfetinng. http://www.carbontrust.com/, 2006.

[76] Cenk Undey, Sinem Ertuni, Thomas Mistretta, and Bryan Looze. Applied advanced process analytics in biopharmaceutical manufacturing: Challenges and prospects in real-time monitoring and control. *Journal of Process Control*, 20(9):1009 – 1018, 2010.

[77] K. Vasko and H.T.T. Toivonen. Estimating the number of segments in time series data using permutation tests. *IEEE International Conference on Data Mining*, pages 466–473, 2002.

[78] J. Venna and S. Kaski. Local multidimensional scaling. *Neural Networks*, pages 889–899, 2006.

[79] J. Venna and S. Kaski. Local multidimensional scaling with controlled tradeoff between trustworthiness and continuity. In *Proceedings of the workshop on self-organizing maps*, pages 695–702, 2005.

[80] A. Vijayaraghavan and D. Dornfeld. Automated energy monitoring of machine tools. *CIRP Annals - Manufacturing Technology*, 59(1):21 – 24, 2010. ISSN 0007-8506. doi: 10.1016/j.cirp.2010.03.042.

[81] Govindaraju V. Wu Y., Ianakiev K. Improved k-nearest neighbor classification. *Pattern Recognition*, 35(1):2311–2318, 2002.

[82] X. Xia and J. Zhang. Energy efficiency and control systems-from a poet perspective. *Control Methodologies and Technology for Energy Efficiency*, 1 (1):255 – 260, 2010. doi: 10.3182/20100329-3-PT-3006.00047.

[83] S.H. Yang, X.Z. Wang, C. McGreavy (Fellow), and Q.H. Chen. Soft sensor based predictive control of industrial fluid catalytic cracking processes. *Chemical Engineering Research and Design*, 76(4):499 – 508, 1998.

[84] G. Young and A.S. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3(1):19–22, 1938.

[85] O. Zaouaka, A.B. Daouda, M. Fagesb, J-L. Fanlob, and B. Auberta. High performance cost effective miniature sensor for continuous network monitoring of h2s. *Chemical Engineering Transactions*, 30:325 – 330, 2012. ISSN 1974-9791. doi: 10.3303/CET1230055.

[86] Robert Michael Lewis, Anne Shepherd, and Virginia Torczon. Implementing generating set search methods for linearly constrained minimization. *SIAM Journal on Scientific Computing*, 29:2507 – 2530, 2007.

[87] Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC bioinformatics*, 4(1):48, 2003.

[88] John Ross Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan kaufmann, 1993.

[89] J. Ross Quinlan. Improved use of continuous attributes in c4. 5. *arXiv preprint cs/9603103*, 1996.

[90] Rolf Ergon. Informative pls score-loading plots for process understanding and monitoring. *Journal of Process Control*, 14(889-897), 2004.

[91] M.J. Stillman W.R. Browett. Computer-aided chemistry ii. a spectral database management program for use with microcomputers. *Computers and Chemistry*, 11:73–82, 1987.

[92] G. Tutz N.Krdzmer, A.L. Boulesteix. Penalized partial least squares with applications to b-spline transformations and functional data. *Chemometrics and Intelligent Laboratory Systems*, 94:60–69, 2008.

[93] Herman Wold. Partial least squares. *Encyclopedia of statistical sciences*, 1985.

[94] B. Theodoulidis I. Kopanakis. Visual data mining modeling techniques for the visualization of mining outcomes. *Journal of Visual Languages and Computing*, 14:543–589, 2003.

[95] E. Fukusaki H. Yamamotoa, H. Yamaji. Canonical correlation analysis for multivariate regression and its application to metabolic fingerprinting. *Biochemical Engineerin Journal*, 40:199–204, 2008.

[96] P. Geladi K.H. Esbensen. Principal component analysis: Concept, geometrical interpretation, mathematical background, algorithms, history,practice. *Comprehensive Chemometrics*, pages 211–226, 2009.

[97] T. Hastieb M. Greenacrea. Dynamic visualization of statistical learning in the context of high-dimensional textual data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8:163–168, 2010.

[98] J. Sanchis X. Blasco, J.M. Herrero. A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. *J. Information Sciences*, 178:3908–3924, 2008.

[99] G Weidl, AL Madsen, and S Israelson. Applications of object-oriented bayesian networks for condition monitoring, root cause analysis and decision support on operation of complex continuous processes. *Computers & chemical engineering*, 29(9):1996–2009, 2005.

[100] Kew Hong Chew, Sharifah Rafidah Wan Alwi, Jiří Jaromír Klemeš, and Zainuddin Abdul Manan. Process modification potentials for total site heat integration. *CHEMICAL ENGINEERING*, 35, 2013.

[101] Yiming Wan, Fan Yang, Ning Lv, Haipeng Xu, Hao Ye, Weichang Li, Peng Xu, Liming Song, and Adam K Usadi. Statistical root cause analysis of novel faults based on digraph models. *Chemical Engineering Research and Design*, 91(1):87–99, 2013.

[102] Bo He, Tao Chen, and Xianhui Yang. Root cause analysis in multivariate statistical process monitoring: Integrating reconstruction-based multivariate contribution analysis with fuzzy-signed directed graphs. *Computers & Chemical Engineering*, 64:167–177, 2014.

[103] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[104] Theodora Retsina. Method and system for targeting and monitoring the energy performance of manufacturing facilities, September 5 2006. US Patent 7,103,452.

[105] DC Montgomery. *Introduction to statistical quality control.* John Wiley & Sons, Inc., 2009. ISBN 9780470169926.

[106] G. Stephanopoulos and C. Han. Intelligent systems in process engineering: A review. *Comput. Chem. Eng.*, 20:743–791, 1996.

[107] S. Kivikunnas. Overview of process trend analysis methods and applications. *ERUDIT Workshop on Applications in Pulp and Paper Industry*, page CD ROM, 1998.

[108] M. Last, Y. Klein, and A. Kandel. Knowledge discovery in time series databases. *IEEE Transactions on Systems, Man, and Cybernetics*, 31(1):160–169, 2000.

[109] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal components analysis. *Neural Computation*, 11:443–482, 1999.

[110] B. Karlsson, J.-O. Jarrhed, and P. Wide. A fusion toolbox for sensor data fusion in industrial recycling. *IEEE Tansactions on Instrumentation and Measurement*, 51(1):144–149, February 2002.

[111] G. L. Marcialis and F. Roli. Fusion of lda and pca for face verification. *Springer-Verlag, London, UK*, pages 30–38, 2002.

[112] F. Samadzadegan. Fusion techniques in remote sensing. *The International Archives of the Photogrammetry, Remote Sensing*, 2003.

[113] A. Negiz and A. Cinar. Monitoring of multivariable dynamic processes and sensor auditing. *Journal of Process Control*, 8(5):375–380, 1998.

[114] C.D. Natale, R. Paolesse, A. Macagnano, A. Mantini, A. DAmico, A. Legin, L. Lvova, A. Rudnitskaya, and Y. Vlasov. Electronic nose and electronic tongue integration for improved classification of clinical and food samples. *Sensors and Actuators B*, 64:15–21, 2000.

[115] A. Charef, A. Ghauch, P. Baussand, and M. Martin-Bouyer. Water quality monitoring using a smart sensing system. *Measurement*, 28:219–224, 2000.

[116] C. Cimander, T. Bachinger, and C.-F. Mandenius. Integration of distributed multi-analyzer monitoring and control in bioprocessing based on a real-time expert system. *Journal of Biotechnology*, 103:237–248, 2003.

[117] C. Cimander, M. Carlsson, and C.-F. Mandenius. Sensor fusion for on-line monitoring of yoghurt fermentation. *Journal of Biotechnology*, 99:237–248, 2002.

[118] E. Keogh and M.J. Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. *4th Int. Conf. on KDD.*, pages 239–243, 1998.

[119] W.J. Krzanowsky. Between group comparison of principal components. *J. Amer. Stat. Assoc.*, pages 703–707, 1979.

[120] A. Singhal and D.E. Seborg. Matching patterns from historical data using PCA and distance similarity factors. *Proceedings of the American Control Conference*, pages 1759–1764, 2001.

[121] W. Elmenreich. An introduction to sensor fusion. Research Report 47/2001, Technische Universität Wien, Institut für Technische Informatik, Treitlstr. 1-3/182-1, 1040 Vienna, Austria, 2001.

[122] G.T. McKee. What can be fused? *Multisensor Fusion for Computer Vision, Nato Advanced Studies Institute Series F*, (99):71–84, 1993.

[123] K. Vasko and H.T.T. Toivonen. Estimating the number of segments in time series data using permutation tests. *IEEE International Conference on Data Mining*, pages 466–473, 2002.

[124] J. Abonyi, B. Feil, S. Nemeth, and P. Arva. Fuzzy clustering time series segmentation. *IDA 2003 Conference*, page http://www.fmt.vein.hu/softcomp, 2003.

[125] J. Lin, E. Keogh, and W. Truppel. Clustering of streaming time series is meaningless: Implications for previous and future research. *SIGKDD'03*, 2003.

[126] Xiang Xuan and Kevin Murphy. Modeling changing dependency structure in multivariate time series. In *Proceedings of the 24th international conference on Machine learning*, pages 1055–1062. ACM, 2007.

[127] Laszlo Dobos and Janos Abonyi. On-line detection of homogeneous operation ranges by dynamic principal component analysis based time-series segmentation. *Chemical Engineering Science*, 75:96–105, 2012.

[128] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[129] Jerome H Friedman. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

[130] Károly Héberger. Sum of ranking differences compares methods or models fairly. *TrAC Trends in Analytical Chemistry*, 29(1):101–109, 2010.

[131] Glenn De'ath and Katharina E Fabricius. Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology*, 81(11): 3178–3192, 2000.

[132] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[133] Paul Geladi and Bruce R Kowalski. Partial least-squares regression: a tutorial. *Analytica chimica acta*, 185:1–17, 1986.

[134] Károly Héberger and András Péter Borosy. Comparison of chemometric methods for prediction of rate constants and activation energies of radical addition reactions. *Journal of chemometrics*, 13(3-4):473–489, 1999.

[135] András Péter Borosy, Katalin Keserű, and Péter Mátyus. Application of nonlinear and local modeling methods for 3d qsar study of class i antiarrhythmics. *Chemometrics and Intelligent Laboratory Systems*, 54(2):107–122, 2000.

# Personal References

[136] Tibor Kulcsar, Gabor Sarossy, Gabor Bereznai, Robert Auer, and Janos Abonyi. Partial least squares model based process monitoring using near infrared spectroscopy. *Periodica Polytechnica Chemical Engineering*, 57(1-2):15–20, 2013. doi: 10.3311/PPch.2165.

[137] Tibor Kulcsar and Janos Abonyi. Development of a modelling framework for nir spectroscopy based on-line analyzers using dimensional reduction techniques and genetic programming. *Chemical Engineering Transactions*, 32, 2013. ISSN 1974-9791. doi: 10.3303/CET1332207.

[138] Janos Abonyi, Tibor Kulcsar, Miklos Balaton, and Laszlo Nagy. Historical process data based energy monitoring-model based time-series segmentation to determine target values. *Chemical Engineering Transactions*, 35:931–936, 2013. ISSN 1974-9791. doi: 10.3303/CET1335155.

[139] Tibor Kulcsar, Miklos Balaton, Laszlo Nagy, and Janos Abonyi. Feature selection based root cause analysis for energy monitoring and targeting. *Chemical Engineering Transactions*, 39:709–714, 2014. ISSN 2283-9216. doi: 10.3303/CET1439119.

[140] Janos Abonyi, Tibor Kulcsar, Miklos Balaton, and Laszlo Nagy. Energy monitoring of process systems: time-series segmentation-based targeting models. *Clean Technologies and Environmental Policy*, 16(7):1245–1253, 2014. ISSN 1618-954X. doi: 10.1007/s10098-014-0808-6. URL `http://dx.doi.org/10.1007/s10098-014-0808-6`.

[141] Tibor Kulcsar and Janos Abonyi. Statistical process control based performance evaluation of on-line analysers. *Hungarian Journal of Industry and Chemistry*, 41(1):77–82, 2013.

[142] Tibor Kulcsar, Barbara Farsang, Sandor Nemeth, and Janos Abonyi. Multivariate statistical and computational intelligence techniques for quality monitoring of production systems. In Cengiz Kahraman and Seda Yanık,

editors, *Intelligent Decision Making in Quality Management*, volume 97 of *Intelligent Systems Reference Library*, pages 237–263. Springer International Publishing, 2016. ISBN 978-3-319-24497-6. doi: 10.1007/978-3-319-24499-0_9.

[143] Tibor Kulcsar, Peter Koncz, Miklos Balaton, Laszlo Nagy, and Janos Abonyi. Statistical process control based energy monitoring of chemical processes. In Petar Sabev Varbanov Jiří Jaromír Klemeš and Peng Yen Liew, editors, *24th European Symposium on Computer Aided Process Engineering*, volume 33 of *Computer Aided Chemical Engineering*, pages 397 – 402. Elsevier, 2014. doi: 10.1016/B978-0-444-63456-6.50067-3.

[144] Tibor Kulcsar, Gabor Bereznai, Gabor Sarossy, Robert Auer, and Janos Abonyi. Visualisation of high dimensional data by use of genetic programming: Application to on-line infrared spectroscopy based process monitoring. In *Soft Computing in Industrial Applications*, volume 223 of *Advances in Intelligent Systems and Computing*, pages 223–231. Springer International Publishing, 2014. ISBN 978-3-319-00929-2. doi: 10.1007/978-3-319-00930-8_20.

[145] Tibor Kulcsar, Gabor Sarossy, Gabor Bereznai, Robert Auer, and Janos Abonyi. Visualization and indexing of spectral databases. In *Proceedings of International Conference on Computational and Statistical Sciences*, volume 6, pages 860 – 865. World Academy of Science, Engineering and Technology, 2012.

[146] J. Abonyi, B. Farsang, and T. Kulcsar. Data-driven development and maintenance of soft-sensors. In *Applied Machine Intelligence and Informatics (SAMI), 2014 IEEE 12th International Symposium on*, pages 239–244. IEEE, Jan 2014. doi: 10.1109/SAMI.2014.6822414.

[147] Tibor Kulcsar and Janos Abonyi. Partial least squares model based process monitoring using near infrared spectroscopy. In *Chemical Engineering Days '12*, Veszprem, Hungary, 2012.

[148] Tibor Kulcsar, Gabor Sarossy, Gabor Bereznai, Robert Auer, and Janos Abonyi. Visualization and indexing of spectral databases. In *International Conference on Computational and Statistical Sciences*, Zurich, Switzerland, 2012.

[149] J. Abonyi, B. Farsang, and T. Kulcsar. Data-driven development and maintenance of soft-sensors. In *2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Herl'any, Slovakia, 2014.

[150] Tibor Kulcsar, Gabor Bereznai, Gabor Sarossy, Robert Auer, and Janos Abonyi. Data-driven development and maintenance of soft-sensors. In *Visualisation of High Dimensional Data by Use of Genetic Programming: Application to On-line Infrared Spectroscopy Based Process Monitoring*, 2012.

[151] Tibor Kulcsar and Janos Abonyi. Development of a modeling framework for nir spectroscopy based on-line analysers using dimensional reduction techniques and genetic programming. In *ICheaP12 - International Conference on Chemical and Process Engineering*, 2014.

[152] Janos Abonyi, Tibor Kulcsar, Miklos Balaton, and Laszlo Nagy. Historical process data based energy monitoring - model based time-series segmentation to determine target values. In *PRES'13 - Conference Process Integration, Modelling and Optimisation for Energy Saving and Pollution Reduction*, Rhodes, Greece, 2013.

[153] Janos Abonyi, Tibor Kulcsar, Miklos Balaton, and Laszlo Nagy. Feature selection based root cause analysis for energy monitoring and targeting. In *PRES 2014 - Conference Process Integration, Modelling and Optimisation for Energy Saving and Pollution Reduction*, Prague, Czech Republic, 2014.

[154] Tibor Kulcsar, Peter Koncz, Miklos Balaton, Laszlo Nagy, and Janos Abonyi. Statistical process control based energy monitoring of chemical processes. In *ESCAPE 24 - 24th European Symposium on Computer Aided Process Engineering*, Budapest, Hungary, 2014.